Janich & Klass Computertechnik GmbH

J&K IMAGING

DpuScan 7

Reference Manual

DpuScan

Copyrights

Trademarks

Disclaimer

The instructions and descriptions in this manual were accurate at the time of this manual's printing. However, we reserve the right to alter the description and/or the product at anytime without prior notice. As per the actual state of software technique it is not possible to develop programs that will work trouble-free under all conditions and in any configuration. © Janich und Klass Computertechnik GmbH assume no liability for damages incurred directly or indirectly from errors, omissions, or discrepancies between this manual and the product.

Actuality

It may happen that a more recent version of this manual for DpuScan is available for download from the Internet. Therefore, it is recommended that you should compare the version by means of the date printed on this page with the version on the Internet. You should please use the most up-to-date version of the manual.

Table of Contents

1 Installation

When purchasing the program, you will usually receive a printed quick guide for installation, a data carrier (CD) and a USB dongle.

If you want to use a version without a dongle, we will send you a download link to the current version of the CD.
On the data carrier or in the installation folder you will find the file "Getting started" (DE / EN / FR) which describes the installation and licensing steps in the current version.

You should keep the CD or the installation folder at a safe place.The same applies to any key supplied for the program itself and the additionally purchased PlugIns. This allows to re-install the program again later.

If you use a computer-bound license, please pay particular attention to the Activation / Deactivation section of the Licensing chapter. A transfer of the license to another computer is not possible without the deactivation process.

2 Introduction

The program is a high-performance Document Capturing and Image Processing Software Package that can be customized to almost any requirement. It can be used very efficiently every time, where large lots of paper must be scanned during short periods of time.

The unique DPU technology enables rotation, data compression and barcode recognition ”on the fly”.

The program meets all requirements for high performance production scanning. So you can define multiple configurations (Base profiles and subprofiles), each for different scanner models, different document types, different processes or just for different customers.The settings are stored in the form of profiles, to be available for later applications without each parameter must be reset.

A series of consecutive commands is referred to as a task. Once the profiles have been defined, you can start such a task by a simple mouse click.You do not need to edit files, but only select the task from a list and start.
The program can produce a variety of paper document formats, from simple letter to continuous forms are processed. The barcode search recognizes 12 different types of barcodes. The imprinter support works with barcodes, user-specific Indices, or several scanners with their image header information.

Documents can be separated during the scanning process by event rules that you can set according to the respective profile. This provides an improved document and file management, enabling the transfer of image files to following applications such as DMS, ECM or workflow programs.
The program exports scanned images as a package of well grouped files. File names and directories are automatically generated by the program and can also be customized.

The program-export functions and tools provide the ability to adapt to all the leading document management systems (DMS).
By editing function an image enhancement (as rotaion, deskew or despeckle) is available after the scan. Documents of poor quality can be rescanned; the new image will automatically replace the old one.

The program generates a statistics file and it provides an image counter.

A log file with information about the captured image is automatically created as a basis for subsequent processing.

The program will be offered depending on the version in different variants and operating modes.

See also Help overview

2.1 Variants of the Program

The version 6 of the program is shipped in several variants, these are distinguished by speed and the number of features included.

Standard	for complex scan tasks.
Professional	for complex scan tasks and advanced requirements.
Gold	for complex scan tasks and advanced requirements, color filtering and background finalization
QSI	For quality assurance and manual indexing
Gold Limited 10	for scanners with a speed up to 10 pages per minute (10ppm / 20ipm)
Gold Limited 16	for scanners with a speed up to 16 pages per minute (16ppm / 32ipm)
Gold Limited 25	for scanners with a speed up to 25 pages per minute (25ppm / 50ipm)
Gold Limited 32	for scanners with a speed up to 32 pages per minute (32ppm / 64ipm)

Gold Limited 50	for scanners with a speed up to 50 pages per minute (50ppm / 100ipm)
Gold Limited 100	for scanners with a speed up to 100 pages per minute (100ppm / 200ipm)
Gold Limited 150	for scanners with a speed up to 150 pages per minute (150ppm / 300ipm)

The details of this classes changes with the technical progress, an actual description can be found here.

For the post processing or the handover of the scanned images to a following process there are two further user-free versions:

DpuFinalize	For the handover of the data in the background.
DpuProcess	For the time consuming enhancement or recognition procedure in the background.

The variant will be determined by the bought license License.

See also introduction.

2.2 Licensing Models

The variants of the program can be licensed in different ways:

Dongle
Buying the program, usually you get a box with a CD, a short manual and a dongle. Connect this dongle to a free USB port. Use if possible a port "close" to the motherboard, on desktop PCs this is block on the rear side.
The license is bound to this dongle, the program will work only on the PC with this dongle.

Product Key & Activation Code
If you got the program without a dongle, you may contact us for a **Product Key** for a certain variant of the program. With entering this key you can try out the program for a certain time. After this time span you can order an **Activation Code** which allows the further usage of the software.
The license is bound to this computer and can only be transferred after it has been **deactivated**. Please contact us about this.

DpuServer
If it is not useful or not possible to work with a dongle, the station can be connected to a DpuServer. Just install the program on the PC. Then in the DpuControlPanel connect this PC to the DpuServer. In the Group Management of the DpuEnterpriseManager you can now support the station with licenses and profiles.
This type of license will work on any PC connected to the DpuServer as long as there are free licenses.

See also introduction.

2.3 Profile Storage

The application uses a separate memory module, the DpuLocalServer. This service stores all configuration data.

In addition, this storage module receives configurations from a DpuServer and makes them available locally. So configurations can be maintained at a central location. Then they can then be reloaded on the individual scan stations each time the scanning application starts automatically.

It can be set how the program intends to confine the profiles of various users against each other. This can be done in a setup dialog during the installing, or later in the DpuServer Settings in the system's control panel.

Common profile storage for all users

This is probably the most common case, for example, when the computer is used by one person. Here you can see all users all available profiles and can use and change them.

🖥 The monitor marks a profile that has been created for all users of the computer. Everyone can choose, use and modify it.

? The question mark stands for the system profile, that is the default values that are accepted if previously not been set.

Profile storage per user

When this mode is selected, each user sees only his own profiles. In addition, he sees even those that have been copied in a common area.

In the corresponding setting dialog, a private profile can be **published locally** and, vice versa, a published profile can be **copied** to the local area. Likewise own and shared profiles can also be **moved** or **deleted** entirely.

👤 The person icon represents a profile that has created the currently logged in user of the computer. Other people do not have access to this profile.

🖥 The monitor is a profile that was created by a specific user of the computer and then published locally. After publishing it can be used any other user.

👤🖥 The combination of person and monitor stands for a profile that was created for all users of the computer and was subsequently modified by the logged in user. So there are now differences from the version published originally.

DpuServer profil storage

If this storage mode was elected, there is no local common area. Instead, the shared profiles can be copied on the DpuServer (published) and downloaded (adjusted) from there automatically. The publishing can be done using the DpuEnterpriseManager. Even in this mode it is possible to copy own profiles and shared profiles as well.

But only own profiles can also be deleted. Shared profiles can only be erased by the DpuEnterpriseManager.

🌐 The world icon is for a profile that was loaded from the server.

👤 The person icon represents a profile that has been created by the current user. Other users can not see or use this profile.

👤🌐 The combination of person and world is a configuration that was initially loaded from the server and was subsequently modified by the logged in user. So there are now changes from the original version of DpuServer.

2.4 Operating modes

Officemode

The Office-Mode offers the user a all control elements in a ribbon bar, which are used frequently in the normale office business. For example the scanning of the incomimg mails or the creation of a brochure or the archiving of old folders. It's possible to use special properties like barcode or patchcode to split the scans into documents.

To start the program on office mode just use the corresponding commandline paramter OFFICE.

Expertmode

The expert mode is suitable to set up complex or multi-stage processes. For example, on the one hand a complex document structure to create for a specific target system. On the other hand, the program should be easy to use, that is, the person on the scanner needs no more changes to make.

This focus implies that adjustments are made by an administrator and must be deposited while an operator later completely can concentrate on the scanning or indexing activity. Additional toolbars are used for this purpose, which can be configured as desired. All elements of the ribbon are stored here as buttons.

The program starts automatically in Expert mode, unless otherwise specified..

See also introduction.

2.5 Working Modes

The program can be operated in different ways:

With user interactions
This is the normal working mode, e.g. when scanning or when indexing. It requires the attention of an operator, for example, to clean exceptional situations (No paper jam, etc.) or to get general data from the user. To interact with the operator, the program offers a user interface.
Depending on the strategy this operating mode can be distinguished between Direct Mode and OpenJob mode, between scanning and postprocessing and between manual and automated work, for more details see also tasks

DpuFinalize
The program can finalize Open Jobs in the background. At the same time the next stack can already be scanned. If an error occurs in a stack, it continues with the next batch.
DpuFinalize can be configured in the normal program interface. DpuFinalize is contained in the program variant DpuScan Gold. For other program variants it may also be purchased optionally.

DpuProcess
As DpuProcess the program can **handle** OpenJobs in the background. Unlike **DpuFinalizer** here almost all actions in the system are available, for example, the call to the OCR. If an error occurs in one batch , it continues with the next.
Also DpuProcess can be configured in the normal program interface. DpuProcess can be purchased at all program versions optionally.

See also introduction.

2.5.1 DirectMode and OpenJobMode

The application distinguishes the two modes **DirectMode** and **OpenJob** Mode.

DirectMode
The image files are stored immediately after scanning directly in the target directory. The opportunities of postprocessing of images here are limited and only during scanning and in pause mode possible immediately after scanning. Inserting or moving images is not possible. Also, the stack can not be temporarily closed and reopened.
The Direct Mode is predominantly used when a highly automated detection is performed without any manual intervention. For example, information for further processing can be introduced by applying barcodes at work preparing next to the document separation.

OpenJobMode
This mode provides against the DirectMode numerous advantages in the flexible process design, since the images are first stored temporarily and stored after finalizing in its destination directory. So, for example, images can be moved or inserted into the Batch. Likewise, the manual calling of the PlugIn is only possible in this mode.

The stack can be transported to other stations in order to decouple the scanning process of further, time-consuming process steps. For example, scanning at a high-performance device can be carried out. The steps image enhancement and indexing can be done with the QSI variant of the program temporally decoupled at other workplaces.
Thus, the scanner can be operated at maximum speed and the downtime of the unit are minimized.

Please note that the QSI variant is used for quality assurance and indexing and therefore **mandatory** to use the OpenJob mode presupposes.

Definition: Direct Mode or OpenJob Mode
The setting, in which mode the software works, is coupled to the task configuration. Here, DirectMode and OpenJob mode distinguished by the Task step "Load batch". By inserting this command an open batch is generated. The configuration dialog for the OpenJob (base path, batch filters, etc.) can be opened by double clicking on the command in the task list or at "Data Source" ind base profile settings.

See also introduction.

2.5.2 DpuFinalize Configuration

The process of external finalizing may be initiated by a statement in the Task Profile or by pressing a corresponding configured button in the toolbar.

A click on the instruction in the task, or a right-click on the button in the toolbar opens the dialog for the configuration of the call.

Start background process	**On:** Starts a new instance of the process in the background. This is similar to open the program but without starting a task. **Off:** Doesn't start the process. Instead the batch will be just appended to the list of the batches to finalize.
Display signon window	Show a startup screen during the initialization phase of the background process.

Display logging window

Opens a window to log the current processing. The newest entry is the top of the list.

Start processing

On:
Determines the start of the finalization immediately. This is only useful if the background process runs, see above.
(This corresponds to the starting of a task in the open program)

Off
The batch will be just appended to the list of the batches to finalize.

Task profiles

Here the task profile will be selected the DpuFinalizer should use.
The buttons act as follows::

⚙ Open the dialog to configure the task steps.
🗋 Creates a new task subprofile
✕ Deletes the selected task subprofile

This task profile ***always*** works with that base profile, which also used the defining task.
The further steps of the task arising from the requirements, typical for the DpuFinalizer is:

Load batch extern
Finalize batch
Restart task

It can be added in addition to the step **Process Event rules on batch**. The frequently used functions such as **Print Info** or queries and **set variables** for the whole batch are possible. The stack may be additionally or alternatively **exported**

But all further steps such as image editing or OCR, however, are reserved for the DpuProcess and can not be used here

If the DpuFinalizer is active, an additional icon in the notification area is displayed to the right of the taskbar. The point in the symbol indicates if the DpuFinalizer is working (red) or is ready for the next batch (green). A **right-click** opens the context menu with entries

Stop Processing Hold the Finalizer after the ending of the current batch but doesn't close the process.
Show/Hide Logging Hide or shows the logging window.
Exit DpuFinalise 1 Terminates the DpuFinalizer immediately and removes the programs instance for the memory.
Close Closes the menu

Please also note the warning about using variables in multiple instances.

Note: DpuFinalize can also be started from the command line or as a link. For that pass the parameter FINALIZE after the program name. To start the process immediately an additional STARTTASK is required.

DpuFinalize can be purchased as additional license. If there are free licenses at the start of the command line, it will be tried to start a new instance. In contrast, at the start as a task step or as a button, see above, it only creates a new instance if this option is selected. Otherwise, the batch remains lying for the running instance.

See also introduction.

2.5.3 DpuProcess Configuration

The configuration of DpuProcess largely corresponds to the configuration of DpuFinalize.

However, there is the possibility to add other task steps in the DpuProcess task. In principle, all actions are allowed in the interest of image processing and recognition of information (OCR, barcode, patch code, plug-in).

Some restrictions are also here:

Because the process is running in the background, all steps are disallowed that might require a user action, such as User Dialogs, counter or directory queries, etc. For the same reason please do not display messages or questions on the screen in the event rule or in PlugIn- or broker scripts.

Likewise, actions are prohibited, that would generate additional images such as copying or filtering because the file name generation can not be prepared for it.

Please also note the warning about using variables in multiple instances.

If the DpuProcess is active, an additional icon in the notification area is displayed to the right of the taskbar. The point in the symbol indicates if the DpuFinalizer is working (red) or is ready for the next batch (green). A **right-click** opens the context menu with entries

Stop Processing	Hold the Finalizer after the ending of the current batch but doesn't close the process.
Show/Hide Logging	Hide or shows the logging window.
Exit DpuProcess 1	Terminates the DpuProcess immediately and removes the programs instance for the memory.
Close	Closes the menu

Note: DpuProcess can also be started from the command line or as a link. For that pass the parameter PROCESS after the program name. To start the process immediately an additional STARTTASK is required.

DpuProcess has to be purchased as additional license. If there are free licenses at the start of the command line, it will be tried to start a new instance. In contrast, at the start as a task step or as a button, see above, it only creates a new instance if this option is selected. Otherwise, the batch remains lying for the running instance.

See also introduction.

3 Users Guide

3.1 Working with the program

The program is very versatile, most commonly it is however used to scan stacks of paper.
This sequence provides essentially like:

Before you can start scanning, the paper must be prepared so that the scanner can capture and transport it. It may not folded or stapled.

Equally important is the selection of an appropriate scan profile in the program. Choosing the right profile is determined by the set task. For example, to capture an incoming mail as an electronic document or to archive already stored invoices electronically.

If there is no matching profile, as such has to be created.

A new profile can be created by selecting a similar one on the tab Profiles and to copy and adjust it afterward.

The details of the profile can be adjusted with the corresponding buttons on tab Settings or a toolbar button with the same name. That means the details for the task profile and the assigned base profile can be set.If there is a matching profile the scanning can start.

If the matching profile is set up, it can be selected. The scanning can be controlled via the tab Start at the ribbonbar.
In Expert mode, for that there are buttons in the toolbar control too.

Because of certain characteristics, such as barcode, patch code, etc., the images can be grouped together to documents or folders and files. Important images or images that are later to be controlled or discarded, can be provided in the event rules with colored marks

After scanning the stack can be displayed for quality control. Here, in the Tree View or in the Document Explorer the view to the important documents can be restricted. For that there are buttons, which marks, shows or hides marked images.

With the help of the corresponding buttons pictures can be inserted, rescanned, rotated or improved. Provided that the appropriate PlugIn is installed, additional information may be given to the image, the file or folder resp.

When the stack is in order, that is, no more blue marks are in, it can be passed on to the subsequent system. For this purpose a descriptive file can be generated and another program can be started or notified, which receives and further processes the batch then. Before, for safety, a copy of the stack can be archived too.

3.1.1 Preparatory work

Before you can start scanning, the paper must be prepared so that the scanner can capture and transport it. It may not folded or stapled.

Equally important is the selection of an appropriate scan profile in the program. Choosing the right profile is determined by the set task. For example, to capture an incoming mail as an electronic document or to archive already stored invoices electronically.

If there is no matching profile, as such has to be created.

3.1.2 Creating Profiles

A new profile can be created by selecting a similar one on the tab Profiles and to copy and adjust it afterward.

The details of the profile can be adjusted with the corresponding buttons on tab Settings or a toolbar button with the same name. That means the details for the task profile and the assigned base profile can be set.If there is a matching profile the scanning can start.

3.1.3 Scanning

If the matching profile is set up, it can be selected. The scanning can be controlled via the tab Start at the ribbonbar.
In Expert mode, for that there are buttons in the toolbar control too.

Because of certain characteristics, such as barcode, patch code, etc., the images can be grouped together to documents or folders and files. Important images or images that are later to be controlled or discarded, can be provided in the event rules with colored marks

3.1.4 Quality control

After scanning the stack can be displayed for quality control. Here, in the Tree View or in the Document Explorer the view to the important documents can be restricted. For that there are buttons, which marks, shows or hides marked images.

With the help of the corresponding buttons pictures can be inserted, rescanned, rotated or improved. Provided that the appropriate PlugIn is installed, additional information may be given to the image, the file or folder resp.

3.1.5 Passing the batch

When the stack is in order, that is, no more blue marks are in, it can be passed on to the subsequent system. For this purpose a descriptive file can be generated and another program can be started or notified, which receives and further processes the batch then. Before, for safety, a copy of the stack can be archived too.

3.1.6 Working in the background

In order to speed up workflow at the scanner, work steps can also be performed automatically in the background. That is, while the program scans the next batch in the foreground, the previous batch is edited in the background and then passed to the subsequent program.

Depending on the working mode, time-consuming tasks such as the text recognition can be moved to the background. A corresponding license is required for working in the background.

3.2 User Interface

The interface of the program is divided into these areas

Captionbar The narrow bar at the top indicates which **program variant** is executed, for example, Gold or QSI. Should there be problems when reading out the dongle or when allocating a license from the server, here is a corresponding note appears.

In addition to the program version, the current task is displayed.

The title bar contains the usual set of operating system controls, for minimize, maximize, close and help.

Ribbonbar The ribbon, contains all important controls to operate the program. It is the preferred control element in **Office Mode**.

Toolbar The toolbars show buttons arranges horizontally or vertically. Which toolbars are to be displayed with which buttons is freely configurable.

Workspace This is the remaining area in where no control elements. It can be divided arbitrarily and are used to display the images or the structure of the batch. Certain parts of the program can also accommodate here own controls, for example, Indexing dialogs.

Statusbar The line at the bottom indicates what the program does at the moment or what it is waiting for.

See help overview also

3.2.1 Ribbonbar

The ribbon will appear below the title bar and above the working area. They can be folded and unfolded by a switch on the right side. It is clearer than the menu bar used in earlier versions. It is in the operating mode OFFICE the preferred control because it includes all the major commands.

The Ribbonbar shows four tabs:

Profiles
Start
Document
Settings

˅˄ ? Right top there are the usual buttons to show or hide the bar or to open the help.

The Ribbonbar is part of the user interface

3.2.1.1 Profile

The tab Profile in the ribbon bar switches to the welcome screen of the program. It shows a menu at the left side and the details for the selected item on right side.

Profile Selection Swithes the right side to the selection of profiles and lists

- the most recently used profiles
- and all profiles

A double click selects the corresponding profile and returns to the tab Start in the ribbon bar

Profile Management Offers on the right side a set of functions for the profile management.

Rights Management Opens a dialog to manage the user rights and grants.

Save Profile Opens a dialog to save the configuration in form of a DAT-File.

License Opens the dialog to enter a product key

System information Informs about several parts of the program.

Help Shows this help file

About
Shows the version information

Exit Program
If necessary asks how to proceeed with the batch currently opened and closes the program.

See also Start, Document and Settings

3.2.1.1.1 Profile Management

This sub menu offers these options:

Load Profile
Displays a File-Open-dialog for DAT-Files. When reading such a file all contained profiles an subprofiles will be imported.

Save Profile
Displays a Save-As-dialog for DAT-Files to save the current settings as a DAT-File

Profile management
In Expertmode the function opens the management tab in the DpuEnterpriseManager. Here several profiles ca be saved, loaded or removed **at once**.

Copy Profile
Copies the current profile and all associated Subprofiles. A dialog will open, asking for the new profile name and displays the previously assigned name. If there are several active subprofiles of the same type, a number is appended to the name, e.g. NEW_01, NEW_02, ... You can replace these "automatically" names by "speaking" names later.

Restore Baseprofile
Re-creates the start profile, i.e. for the Office Mode the profile **OFFICE** and for the Expert Mode the profile **DPS**.

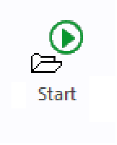
Create Expert Profile
In Officemode this function make the current profile available for the Expertmode.

back

3.2.1.2 Start

The tab **Start** of the ribbon bar is used to control the process and has these groups

Group Start

Symbol	Meaning
	Runs the loaded task, usually: load profiles, open stack, scan, ...

Group Scan

Symbol	Meaning

Continue	Restarts the scanner after a pause again.
Pause	Stops the scanner, retrieves the remaining images and changes the screen layout to pause mode.
Insert	Opens a menu for inserting images.
Replace	Opens a menu for replacing images.
Löschen	Opens a menu for deleting images.

Group Save

Symbol	Meaning
Stop	When running, it stops the scanner and processes the remaining images. Than it ends the current task and closes image batch. This remains in the OPEN_JOB directory.
Finalize	Finalizes the batch currently opened. Than it switches from pause mode to ready mode.
Purge	End s the current task, deletes all images and removes the batch from the OPEN_JOB folder.

Siehe auch Profile, Dokument und Einstellungen

3.2.1.3 Document

The tab **Document** of the ribbon bar serves to postprocess the current batch and has these groups:

Group Actions

Symbol	Meaning
RotR90	Rotates the image clockwise by 90°
Rot180	Rotates the image by 180°.
RotL90	Rotates the image counter clockwise by 90°.
Invert	Inverts the image colors.
Deskew	Deskews the image and removes black borders.

Group Document

Symbol	Meaning
Connect	Packs the following images to the document currently selected.
Split	Creates a new document, beginning with the selected image.
Löschen	Opens a menu for deleting images.

Group Mark

Symbol Meaning

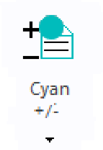
Switches the red marker. Red marked images will not be stored. Clicking on the arrow opens the menu red marks.

Switches the blue marker. A blue marked image prevents the batch to be finalized..
Clicking on the arrow opens the menu blue marks.

Switches the green marker Use green marks to highlight an image. Clicking on the arrow opens the menu green marks.

Switches the magenta marker Use this marks to highlight an image. Clicking on the arrow opens the menu magenta marks.

Switches the orange marker Use this marks to highlight an image. Clicking on the arrow opens the menu orange marks.



Switches the cyan marker Use this marks to highlight an image. Clicking on the arrow opens the menu cyan marks.

See also profile, Document and Settings

3.2.1.4 Settings

The tab **Settings** of the ribbon bar serves to prepare the work flow for the the current batch and has different groups of commands, depending on the operation mode:

Settings in Office Mode

Settings in Expert Mode

See also Profile, Start and Document

3.2.1.4.1 Settings Expert Mode

The Tab "Settings" in the Ribbonbar allows to setup the scanner and some further sub profiles.

Settings

Symbol Meaning

Opens the setup dialog for the application layout i.e. the toolbars.

Task

Opens the setup dialog for the task profile.

Base profil

Opens the setup dialog for the base profile.

Source image

Symbol Meaning

Scanner-1

Activates the first (or only) scanner, usually a document scanner.Click on the arrow to open the document scanner menu.

Scanner-2

Activates the second scanner (if connected) , for example a flatbed scanner. Click on the arrow to open the flatbed scanner menu.

Image Selection

Symbol Meaning

All

Opens a menu where you can specify which of the available color formats to save. The symbol and text of this button correspond to the last color selection made in the menu.

For interactive enhancement, it may be useful to fetch color and black-and-white images from the scanner, but in the end, only black-and-white on the hard drive to save.

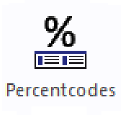
The prerequisite for saving is that the scanner generates the corresponding color type or it will be created by filtering. Is this not possible If this is not the case, this selection is not possible and is displayed in gray.

If the color type is created, an additional storage format must be specified. If this format is missing, this is indicated by the warning:

The color format is not available!

An automatic selection is only possible if the scanner provides this information or if the PlugIn color detection is used.

Setting

Symbol	Meaning
 Percentcodes	Opens an overview for the program variables also called %-Codes. This variables collects all information the program gathers during the work, e.g. counter, date, time, codes or further image information.
Scanner	Opens the dialog for the scanner setup. Possible settings are resolution, paper size, orientation and so on.
Display	Opens the setup dialog for the splitting of the working area. For example the number of visible images when scanning and in the pause mode can be given here. This button is available only when a job is opened currently. The settings are **temporarily**, the general settings can be made in the base class setup.

3.2.1.4.2 Settings Office Mode

Settings in Office Mode

Image Source

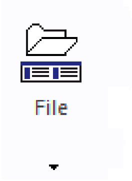
Symbol	Meaning
Scanner-1	Activates the first (or only) scanner, usually a document scanner.Click on the arrow to open the document scanner menu.
Scanner-2	Activates the second scanner (if connected) , for example a flatbed scanner. Click on the arrow to open the flatbed scanner menu.
 File	Selects a picture folder as input. The files are loaded in alphabetical order, with multi-page files always all images are loaded. Clicking on the arrow opens the Load-Menu.

Image Selection

Symbol	Meaning
All	Opens a menu where you can specify which of the available color formats to save. The symbol and text of this button correspond to the last color selection made in the menu. For interactive enhancement, it may be useful to fetch color and black-and-white images from the scanner, but in the end, only black-and-white on the hard drive to save.

The prerequisite for saving is that the scanner generates the corresponding color type.

Doc Separation

Symbol	Meaning
Image ▾	Opens a menu to enable the separation after each sheet.
Dpucode ▾	Opens a menu to activate the separation on a DpuCode.
Image ▾	Opens a menu to enable the separation after each sheet.
Barcode ▾	Opens a menu to activate and set up the separation by Barcodes.

Image correction

Symbol	Meaning
Punch hole remove	Activates the filling of punch holes near the edge of the paper.
Detect orientation	Rotates the image to be in an upright readable orientation. Also small angles will be corrected.

Delete

Symbol	Meaning

Empty pages	Activates the removing of empty pages

Target Image

Symbol	Meaning
Path	Defines the destination folder for the Images. The destination folder can be a local directory or a network path. Also the handling of existing files and folders can be given.
Filename	Defines the names for the image files. An assistant helps to compile the name from counters, times and found barcodes.
File format PDF	Opens a menu to select the type of file to store the images
Batch file	Creates an additional file that describes the documents. This file is formatted in the XML standard and describes the structure of the stack, the scanned documents and the barcodes found.

3.2.2 Toolbars

As part of the user interface in Expertmode up to eight floating toolbar-windows can be defined. They are free configurable, that means one can put the most frequently used actions in form of buttons on this windows. The toolbars can be docked to the right, left or upper border of the working area.
The toolbar configuration serves the keyboard configuration also. That means the commands in a toolbar attached to shortcuts. The shortcuts will work always, nevertheless the corresponding buttons are not placed on the toolbars.

To keep it comfortable for the operator it is useful to reduce the count of buttons to the one really needed.

As long there is a shortcut to get back to the configuration, it is possible to hide the ribbon bar itself. If locked out yourself accidently you can start the program with a special switch from the commandline <Win>-<R>:

```
"C:\Program Files (x86)\JuK\DpuScan\DpuScan.exe" FORCERIBBONON.
```

As part of the user interface up to eight floating toolbar-windows can be defined. They are free configurable, that means one can put the most frequently used actions in form of buttons on these windows. The toolbars can be docked to the right, left or upper border of the working area.

The toolbar configuration servers the keyboard configuration also. That means the commands in a toolbar can be attatched to hotkeys. The hotkeys will work always, althoug the corresponding buttons may not be placed on the toolbars.

To keep it comfortable for the operator it is useful to reduce the count of buttons to those one really needed.

As long there is a hotkey to get back to the configuration, it is possible to hide the ribbon bar itself. If locked out yourself accidently you can start the program with a special switch from the commandline <Win>-<R>:

```
"C:\Program Files (x86)\JuK\DpuScan\DpuScan.exe" FORCERIBBONON.
```

Then the ribbonbar will be visible again and the application layout will be accessible.

3.2.3 Working Area

The visible area of t?he program, which is not occupied by the ribbon, toolbars, or title and status bars, is the working area. It is used to show the incoming images during scanning and in the pause the structure of the data. In the display setup it can be arbitrarily divided into several panes.

The essential parts of the work area are:

Document explorer	Displays the document structure
Tree	Displays the folder and file structure
Thumbnails	Shows a small preview of scanned images
Image group thumbnails	Displays the preview of the images in one file.
Edit Window	Displays the newly scanned image and the selected image in the pause
Info Window	Can be used to display any information
Tools Window	Two dimensional control for image enhancement
Scan settings	Allows quickly setting the main parameters
Warn Window	Shows warnings or Event Rule outcomes
Batchwindow	Displays a preview to be generated log file.

If possible the windows are synchronized, that means the selection of an image in the tree will move automatically set the selection mark in the document explorer and all other visible windows too.

In earlier versions there were the window types brightness slider, contrast slider or marker information window. These rarely used types will no longer be supported in version 6 of the program.

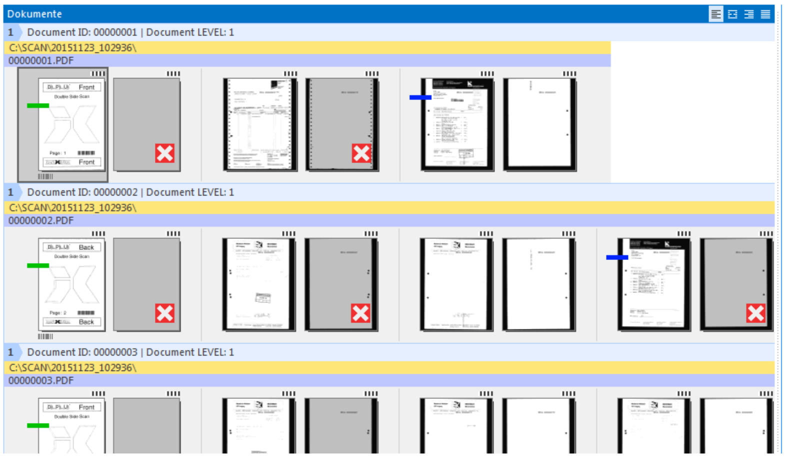
See also user interface and help overview.

3.2.3.1 Document Explorer

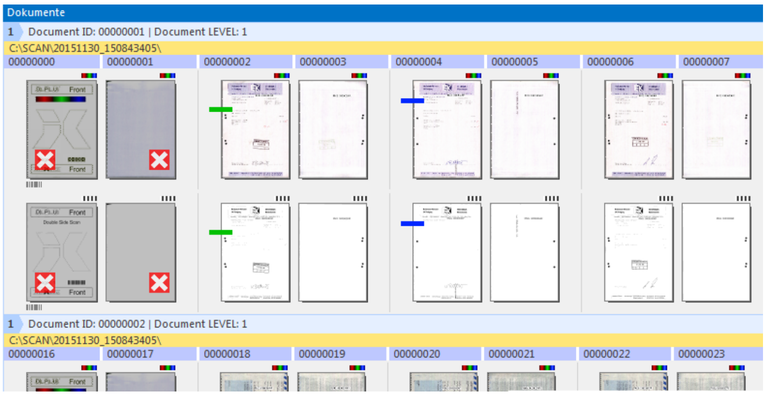
This window shows the document structure of the batch. For single images to be stored the color format to save can be set. Pages and sheets can be moved via drag & drop. Depending on the set document level the documents will be indented.

Document

Each document will be shown as one row: A light yellow bar indicates the location, a light blue bar shows the name of the file. Both sides of a sheet are displayed side by side close, the separation between two sheets is indicated by the greater distance.


Dokumente
1 Document ID: 00000001 | Document LEVEL: 1
C:\SCAN\20151123_102936\
00000001.PDF
1 Document ID: 00000002 | Document LEVEL: 1
C:\SCAN\20151123_102936\
00000002.PDF
1 Document ID: 00000003 | Document LEVEL: 1
C:\SCAN\20151123_102936\
00000003.PDF

If in office mode **all color formats** are chosen for output the display will use two or three lines (for Dual or Multi Stream)


Dokumente
1 Document ID: 00000001 | Document LEVEL: 1
C:\SCAN\20151130_150843405\
00000000
00000001
00000002
00000003
00000004
00000005
00000006
00000007
1 Document ID: 00000002 | Document LEVEL: 1
C:\SCAN\20151130_150843405\
00000016
00000017
00000018
00000019
00000020
00000021
00000022
00000023

If there are more images in the document than the screen's width, the display ca be adjusted. For example to show the begin and/or the ende of the document.

Documents will be shown left aligned. Always the beginning images are visible. Use the horizontal slider to see the remaining images. If an image **is selected** it will stay in the **visible** range always.

Documents will be shown right aligned. Always the last images are visible. Use the horizontal slider to see the preceding images. If an image **is selected** it will stay in the **visible** range always.

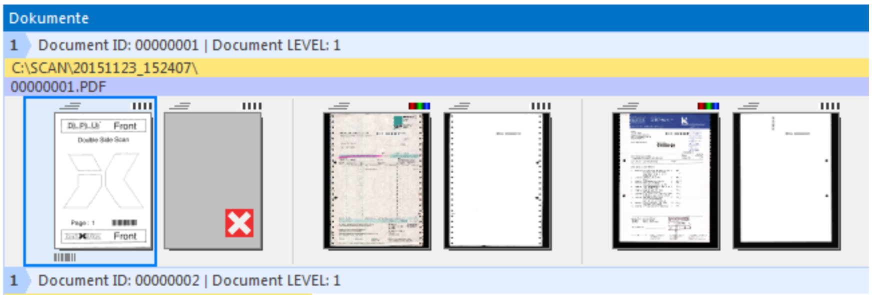
Show the first and the last images of the document, the images in between begin and end will be hidden.Watch out: If an image "in the middle" **was selected** it will **hidden** automatically!

In OfficeMode a multistream scanner output can be limited in a simple manner to one image type (for example, black/white). The other image (color image) is kept in the background to improve the visible image, for example, by filtering or interactive image enhancement. Using the block mode button, these images can be displayed also.

Blockmode

The block mode button displays all images of a multistream in a grid. If an image is missing, a free space will be displayed instead. Of course this button doesn't work for singlestream scanning.

In the following sample the machine returns bitonal and color images. But except some tables or photos, all other images shall be saved b/w later. In the following graphic only image 3 and 5 are set to color the rest is b/w.



Dokumente

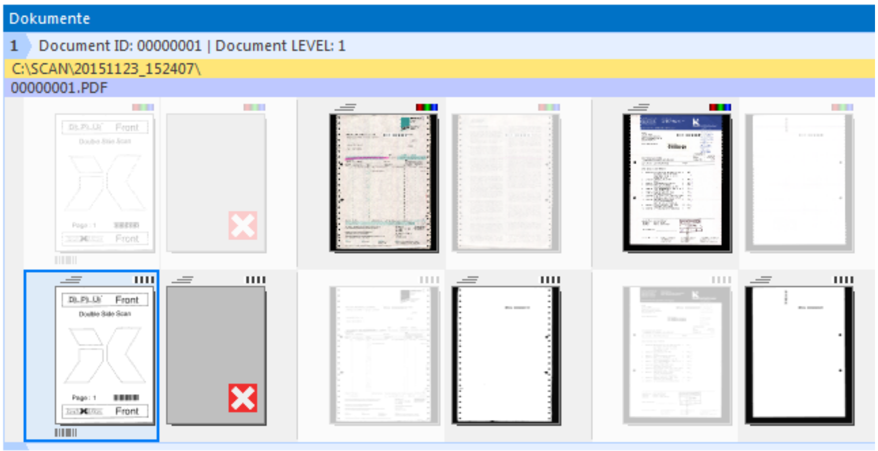
1 Document ID: 00000001 | Document LEVEL: 1

C:\SCAN\20151123_152407\

00000001.PDF

1 Document ID: 00000002 | Document LEVEL: 1

If the block mode is enabled, all images are shown, also the hidden ones:



Dokumente

1 Document ID: 00000001 | Document LEVEL: 1

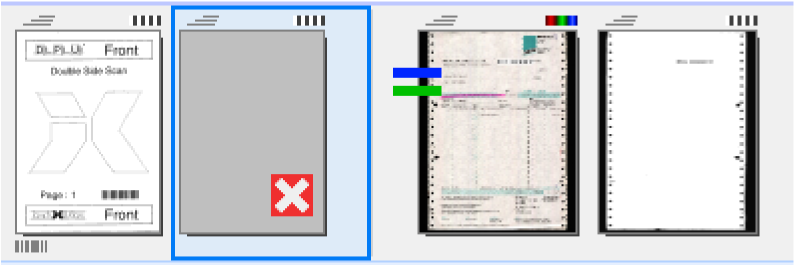
C:\SCAN\20151123_152407\

00000001.PDF

In Expert mode the block mode doesn't work, because the number of images can vary depending on the settings.

Pages and sheets

In the document explorer the front side and the rear side are summarized as a sheet and be displayed close together.

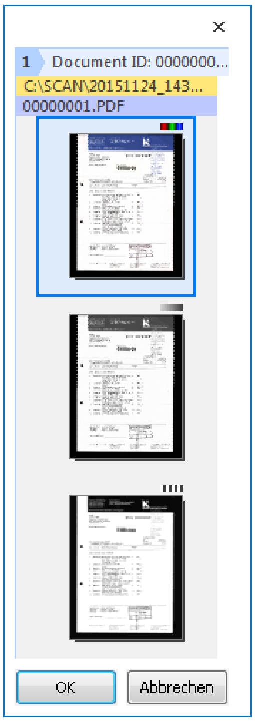


Above and below the preview of the image there are symbols pointing to the properties of the image.

≡	top left	For this image are existing one or more color separations (multistream). If they are hidden currently they can be shown by using the block mode button. This symbol here can be used to toggle the color type.

▮▮▮▮	top right	This indicates a bitonal image. (black/white image) 1 Bit color depth, toggles by clicking.
▬	top right	This indicates a grayscale image. 8 Bit color depth, toggles by clicking.
🟥🟩🟦	top right	This indicates a color image. 24 Bit color depth, toggles by clicking.
‖‖‖	bottom left	On this image a barcode was found
≡	bottom right	On this image a Patchcode was found
🟩	at the side	This image has a green mark (important)
🟦	at the side	This image has a blue mark (must be inspected)
❌	inside the image	This image has a red mark (should not be saved)

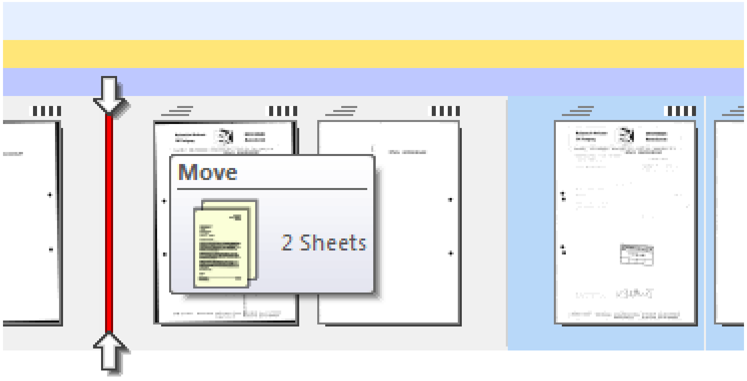
Toogeling the color type
If the scanner sends several color separations (multistream) one of them can be selected for the output. To toggle them a click on the **top right symbol** of the image opens an image menu:


1 Document ID: 0000000...
C:\SCAN\20151124_143...
00000001.PDF

Choose here the type of picture to be saved.

Selecting Images
By a simple left mouse click an image can be selected. Like in other Windows dialogs the selection can be enlarged by holdinge the keys **Ctrl** or **Shift** before the next click.

Moving Images
In Officemode: If one or more images are selected they can be moved by dragging the mouse. They can be inserted in between two sheets. The inserting position will be indicated by a red bar. It is not possible insert an image between front and rear side of a sheet.
In Expertmode this restriction does not apply.



See also other parts of the working area and user interface.

3.2.3.2 Tree View

The directory tree displays the storage structure of the data, that is, the directories and files that are created in the destination folder later.

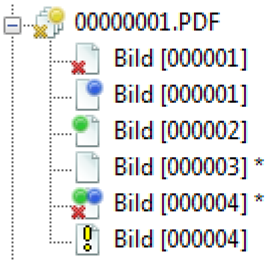
The root of the tree has the name of the batch, including the directories are listed. Below the directories the files are listed and within the files again the images.

Directories and Files

Directories are displayed simply indented. If the name of the directory or file is not yet known, because the destination directory is not empty, then a counter is displayed, for example, "+++++++ 3"for the third "automatic directory".

Images and Marks

The tree shows for all images the marks:



Red, blue and green Marks will be shown in a corner of the image. The white mark will be show as a star after the name.

Once an image has a deletion marker (red cross), the parent file icon has a yellow cross, when all the images are marked in red, and the file icon is highlighted in red too. The same applies to blue and green markings

A yellow exclamation mark indicates a missing storage format for this image.

It may be that not all images are visible, for example, if this is set, or if they have been hidden by the corresponding buttons in the toolbar markings.

Selecting and editing

A single image can be selected by a mouse click. Than it can be further processed, for example, in the Edit window, or by an action in the command group Edit Image.

If this is not prohibited, the selection can be extended by pressing the key <Ctrl> or <Shift> to a multiple images (multi-selection).

If one or more images are selected, the actions of the toolbar user actions (rotating, straightening, inverting, ..) can apply all selected images at the same time. The same applies to the actions in the Ribbon Document,

Possibly a button is disabled when the use only on certain images is allowed.

Moving and Copying

If this is not prohibited, selected images can also be moved by drag & drop to a different location. If this images are moved, where there is a separation, it is useful to rebuild the batch, that is, to repeat the event rules or to generate the new filename.

Copying is possible only for a single image. To do this, use the button in the command group process.

The usual Windows keyboard shortcuts for Copy, Cut and Paste <Ctrl> C, <Ctrl> -X and <Ctrl> -V *do not work* in this version of the software.

See also other parts of the working area and user interface.

3.2.3.3 Thumbnails

The preview window shows reduced representations of the images received, also called "thumbnails". It is used for rapid control and easy navigation in the batch. Since the display of thumbnails requires computation time, they are not created until the scanning is done. Also they will be created only for the currently visible part of the batch.

Image markings

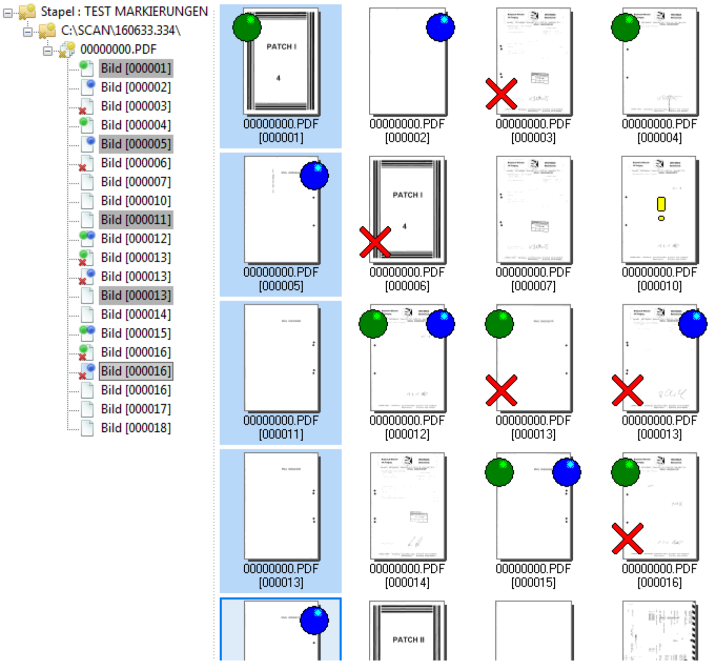
The images are displayed in rows and columns. The marks are shown in the corners of the thumbnails. Here the same rule apply as for the tree, especially for the presentation of the marks and the visibility of the images. By moving the right border of the window the line break after a certain number images can be enforced. So, e.g., a four-page document can build one line. Document directories or boundaries are not represented.

The text below the thumbnails is the file name, for multipage files can be displayed in the file here the number of the image.

Selecting and editing

A single image can be selected by a mouse click. Than it can be further processed, for example, in the Edit window, or by an action in the toolbar "Edit Image" (Filter, PlugIn) or "Actions" (paste, cutting, joining).

If this is not prohibited, the selection can be extended by pressing the key Ctrl or Shift to a multiple images (multi-selection).



.

If one or more images are selected, the actions of the toolbar user actions (rotating, straightening, inverting, ..) can apply all selected images at the same time. The same applies to the actions in the Ribbon Document,

Possibly a button is disabled when the use only on certain images is allowed.

Moving and Copying

If this is not prohibited, selected images can also be moved by drag & drop to the tree view. If this images are moved, where there is a separation, it is useful to rebuild the batch, that is, to repeat the event rules or to generate the new filename.

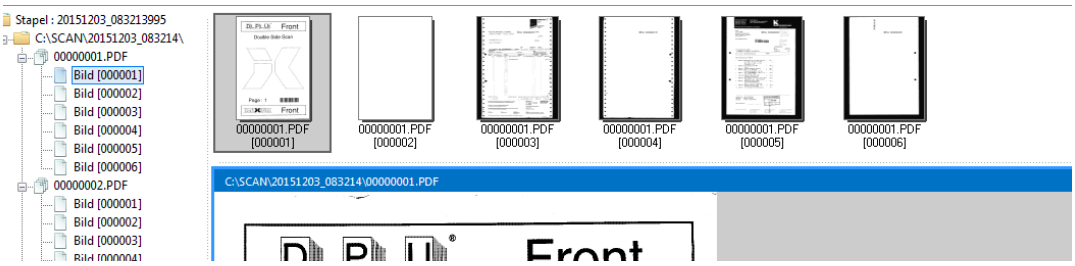
Copying is possible only for a single image. To do this, use the button in the action toolbar.

The usual Windows keyboard shortcuts for Copy, Cut and Paste <Ctrl> C, <Ctrl> -X and <Ctrl> -V *do not work* in this version of the

See also other parts of the working area and user interface.

3.2.3.4 Image Group Thumbnails

The Image Group Thumbnail shown - like the regular Thumbnails - a scaled down representations of the images received. Like this it is used for rapid control and easy navigation in the stack. It has the same features and limitations.
In contrast to the preview window, the view can be limited to a file or a folder.

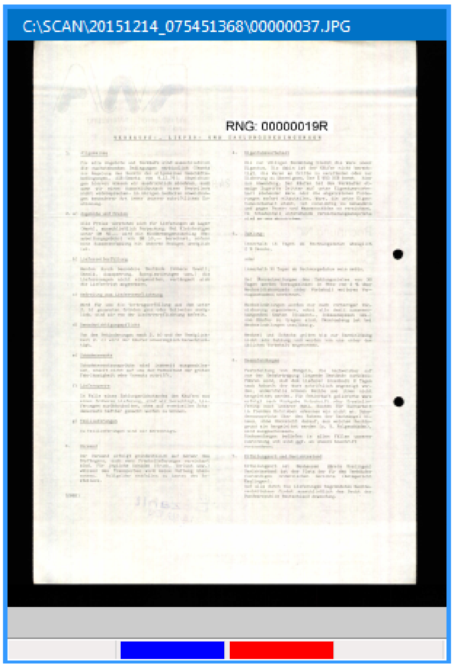


See also other parts of the working area and user interface.

3.2.3.5 Edit Window

This part of the working area displays the images when scanning and allows to inspect them in the interactive mode. If necessary they can be reworked with the help of the buttons in the toolbar Edit Menu. For example the can be rotated or deskewd manually or searched for barcodes or texts.

The image can be saved to a file or be replaced by another image from the hard disk.



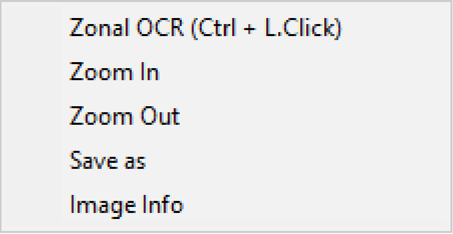
If the status bars are displayed, the marks are indicated by corresponding color boxes at the bottom.

In addition, these actions can be triggered with the mouse in the Edit window:

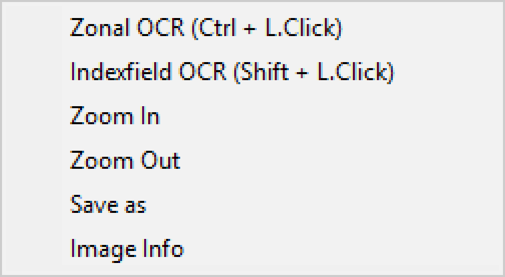
Mouse wheel
Turning the mouse wheel enlarges or reduces the representation of the image.

Left mouse button
In the Edit window, it is possible to move the image by clicking on it with the left, holds and moves the mouse.

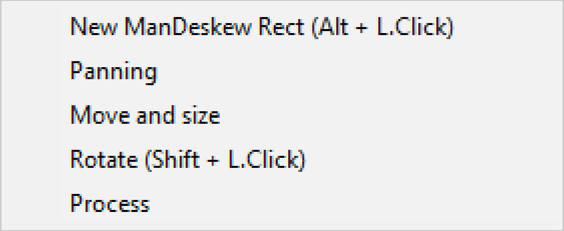
Right mouse key
The right mouse button opens a context menu. Normally, this menu shows the options available for **zooming in and out** as well as the option of raising a rectangle in which a **text recognition** is performed. So you can quickly see if the image quality for OCR is sufficient.



In addition, when an embedded index dialog is shown and configured accordingly, a frame can be drawn up, its content is transferred directly into the current index field.



If the function of the **manual deskew** has been activated, the context menu changes and now allows you to switch between the moving of the image (panning) and moving the deskew frame.

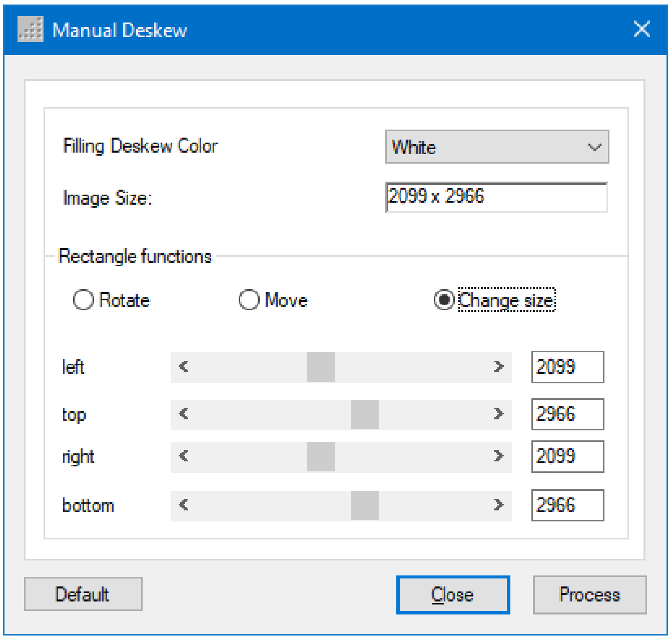


See also other parts of the working area and user interface.

3.2.3.5.1 Manual Deskew

Sometimes the automatic deskew can not find the borders of the paper in the image because the paper doesn't differ from the scanner's background. For example scanning magazines sometimes a page appears nearly complete black - except the picture of the product in the middle of the page. So the border of the paper cannot be found if the scanner has a black background too. The same situation comes up when the page could be deskewed but is not processed because of a user defined condition.

In this case the image can be straightened and cut manually. A click on the button manual deskew or the choice of the corresponding context menu will open an auxiliary dialog:



Filling Deskew Color: White

Image Size: 2099 x 2966

Rectangle functions: ○ Rotate ○ Move ● Change size

left: 2099

top: 2966

right: 2099

bottom: 2966

Filling Deskew Color

If the hand-reared area will be rotated by a great angle, it can happen that the corners extend beyond the actual image. If the image is straightened now, triangles will come up for which there is no image information.

These triangles can be filled with either black or white

Image Size

This field specifies the size of the cut-out in 1/10 mm.

Rotate
Move
change size

Here you can set which part function is operated by the sliders at the bottom of setup dialog.

Rotation

Allows when **rotating** the cutout in 1/10 degrees to turn left or right.

left/right
top/bottom

Allows the precise **positioning** of the frame when moving it.

left
right
top
bottom

These slides allow the exactly placing of the edges when **resizing** the cutout range..

The displayed numerical value is the length or width of the resulting image.

Close

Closes the dialog.
.

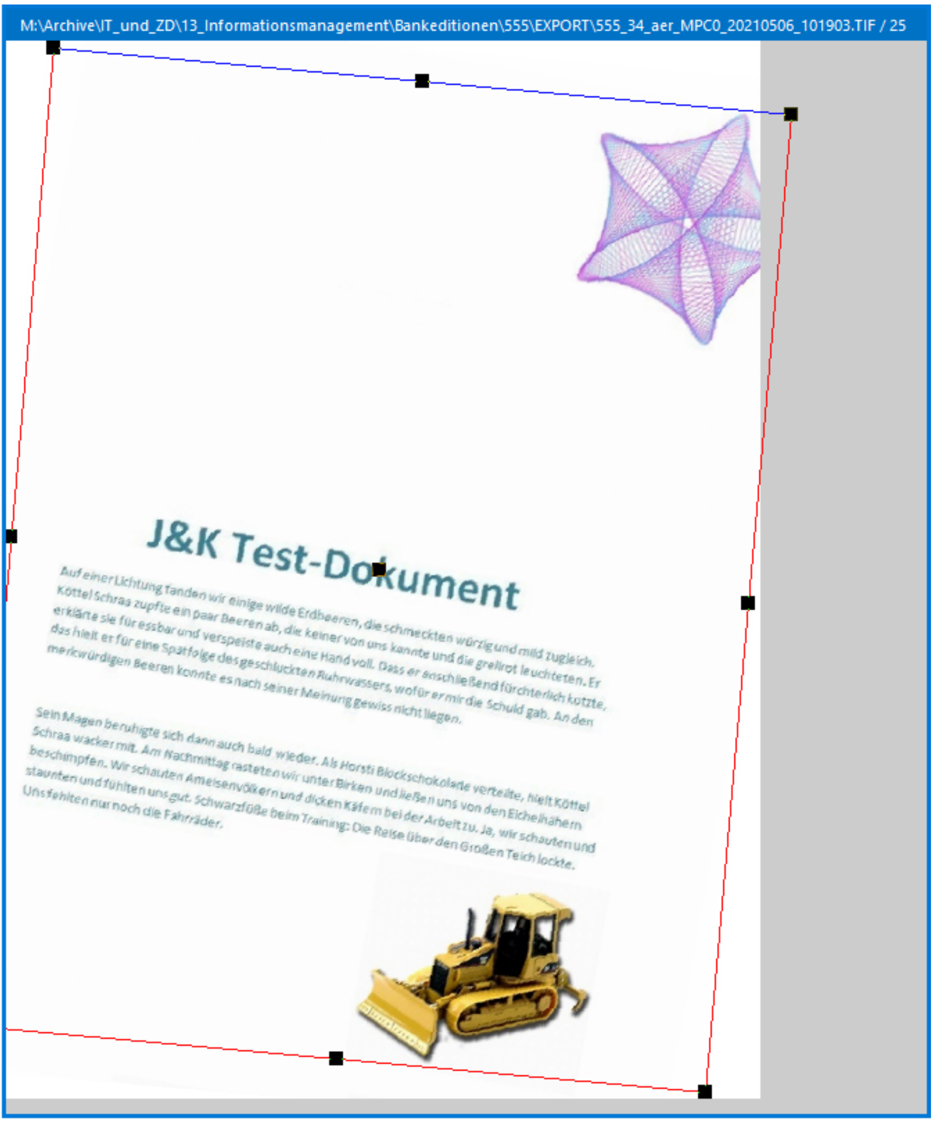
Process

Starts the deskew process, the dialog will stay open..

In addition, in selected Edit window a frame will be appear. The blue edge represents the upper edge of the cut out region, the red ones, the others. By "handles" on the edges of the frame it can be **drawn**. The handle in the center of the rectangle is designed to **move** it.

If the **shift key** is hold simultaneously, the frame can be **rotated** with the handles at the corners.

To fine tune the appropriate controls can be used in auxiliary dialog.



M:\Archive\IT_und_ZD\13_Informationsmanagement\Bankeditionen\555\EXPORT\555_34_aer_MPC0_20210506_101903.TIF / 25

J&K Test-Dokument

Auf einer Lichtung fanden wir einige wilde Erdbeeren, die schmeckten würzig und mild zugleich. Köttel Schraa zupfte ein paar Beeren ab, die keiner von uns kannte und die grellrot leuchteten. Er erklärte sie für essbar und verspeiste auch eine Hand voll. Dass er anschließend fürchterlich kotzte, das hielt er für eine Spätfolge des geschluckten Ruhrwassers, wofür er mir die Schuld gab. An den merkwürdigen Beeren konnte es nach seiner Meinung gewiss nicht liegen.

Sein Magen beruhigte sich dann auch bald wieder. Als Horsti Blockschokolade verteilte, hielt Köttel Schraa wacker mit. Am Nachmittag rasteten wir unter Birken und ließen uns von den Eichelhähern beschimpfen. Wir schauten Ameisenvölkern und dicken Käfern bei der Arbeit zu. Ja, wir schauten und staunten und fühlten uns gut. Schwarzfüße beim Training: Die Reise über den Großen Teich lockte. Uns fehlten nur noch die Fahrräder.

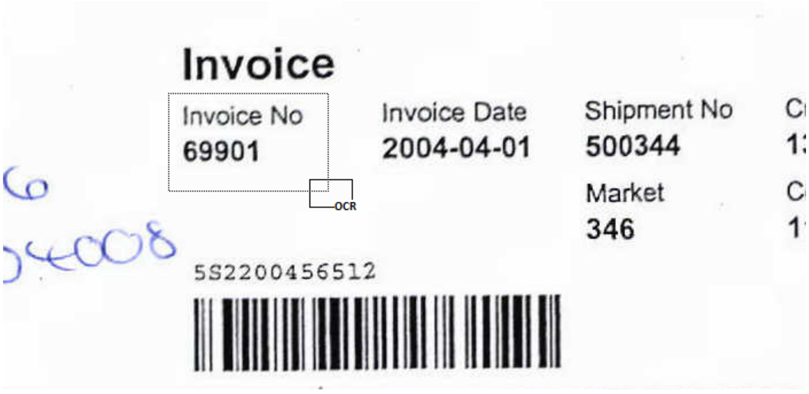
t

This process **keeps track of the last selected edge** for the rest of the stack. A distinction is made between the front and back: If the frame on the **front** turned to the left, it turned to the right appears on the **back** at the same angle.

See also other parts of the working area and user interface.

3.2.3.5.2 Zonal Text Recognition

By holding the **Control**-Key a frame can be spawned with the left mouse button:



Invoice

Invoice No
69901

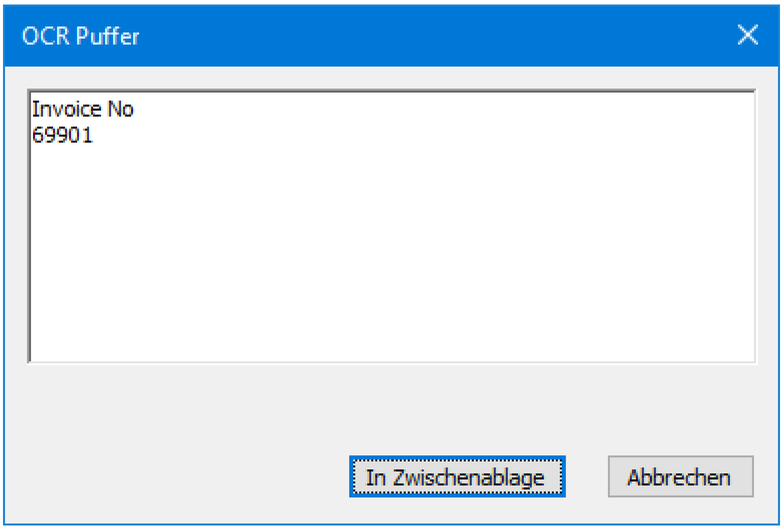
Invoice Date
2004-04-01

Shipment No
500344

Market
346

5S2200456512

As soon as the mouse is released the result will be displayed in a box:



OCR Puffer

Invoice No
69901

See also other parts of the working area and user interface.

3.2.3.6 Info Window

The Info Window can be displayed as a pop-up window or as part of the working area. It serves mostly display information in the form of %-Codes.

The %-Codes are updated for the last clicked image in the Directory Tree or or in the Document Explorer.

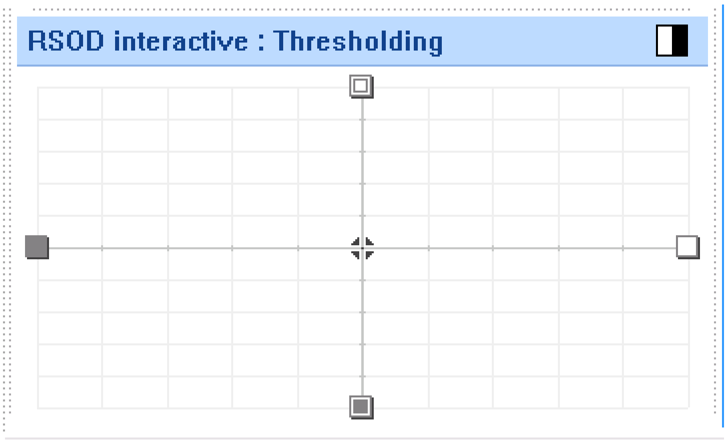
Here, the scope will be respected, as the F-variables for the file are displayed, which includes the last selected image.

See also other parts of the working area and user interface.

3.2.3.7 Tools Window

This part of the working area can be used by PlugIns that e.g. offer an image enhancement. The Tools window lists the tools as bars one under the other. A klick on the bar will show the dialog window and hide the others.

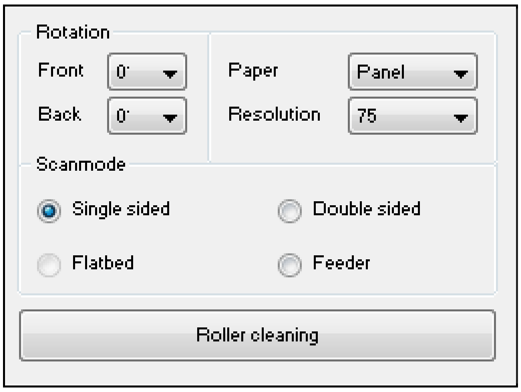
In this example, the plug-in "Interactive image enhancement" the method "RSOD interactively" available



See also other parts of the working area and user interface.

3.2.3.8 Scan Settings

The scanner parameter window can be displayed as a pop-up window or as part of the working area. It is a simple way to change some of the most important scan parameters temporarily for following images. With the start of the next task, the original settings are restored.



Rotation

Front: 0°

Back: 0°

Paper: Panel

Resolution: 75

Scanmode

● Single sided ○ Double sided

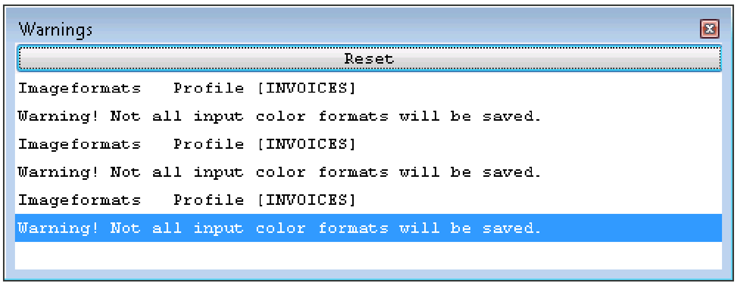
○ Flatbed ○ Feeder

While scanning a batch this window is **visible but inactive**. Only in a scan **pause** it can be operated. Its use is therefore useful when working with manual feeding, or scanning a prepared batch so that the task pauses when a change of settings is necessary.

The switch between flatbed and automatic feeder "Feeder" only works for devices with these properties. Also starting the pick rollers without additional paper transport is only for scanners that support this.

3.2.3.9 Warn Window

The Info Window can be displayed as a pop-up window or as part of the working area. It displays error messages from the program, for example, that not all the images can be saved, because a storage format for the incoming images is missing.



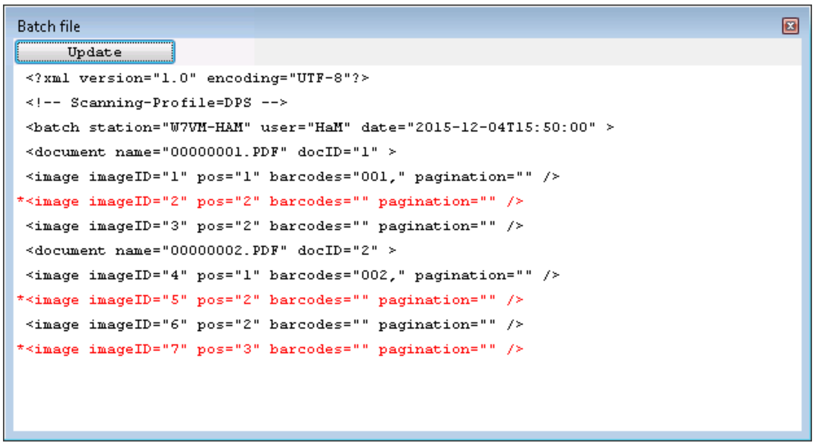
```
Imageformats   Profile [INVOICES]
Warning! Not all input color formats will be saved.
Imageformats   Profile [INVOICES]
Warning! Not all input color formats will be saved.
Imageformats   Profile [INVOICES]
Warning! Not all input color formats will be saved.
```

In addition, the processing of the event rules can here be tracked, if this option is turned on.

See also other parts of the working area and user interface.

3.2.3.10 Batch Window

The Info Window can be displayed as a pop-up window or as part of the working area. It shows a preview of the resulting status file:



```
 <?xml version="1.0" encoding="UTF-8"?>
 <!-- Scanning-Profile=DPS -->
 <batch station="W7VM-HAM" user="HaM" date="2015-12-04T15:50:00" >
 <document name="00000001.PDF" docID="1" >
 <image imageID="1" pos="1" barcodes="001," pagination="" />
*<image imageID="2" pos="2" barcodes="" pagination="" />
 <image imageID="3" pos="2" barcodes="" pagination="" />
 <document name="00000002.PDF" docID="2" >
 <image imageID="4" pos="1" barcodes="002," pagination="" />
*<image imageID="5" pos="2" barcodes="" pagination="" />
 <image imageID="6" pos="2" barcodes="" pagination="" />
*<image imageID="7" pos="3" barcodes="" pagination="" />
```

When an image to delete is highlighted in red, the corresponding line is also shown in red in the status file preview.

□ **Tip**: Avoid to show this window permanently at larger batches, because the calculation always requires the evaluation of all previously scanned images.

See also other parts of the working area and user interface.

3.2.3.11 Other Windows

PlugIn window
PlugIns can reserve a portion of the working area for themselves, for example, to display a cotrol window, with which the image can be improved. Also the input of information through an embedded UDD is possible in this type of window.

Empty Window
Empty windows can be used as a placeholder for a non-modal, "topmost" window of another program, for example, the always visible status indication of an import program.

Empty windows appear automatically when a PlugIn that would have this place actually, can not be displayed.

See also other parts of the working area and user interface.

3.2.4 Statusbar

The statusbar will be shown under the working area. It can be hidden and shown in the application layout.

Left part

The left part is used to display system information. In pause mode, also called interactive mode, here the button text appears when you hold the mouse over one of the buttons in the toolbar. These texts can be defined for each button on the tab actions in the application layout.
During the scan procedure, also called process mode here the current action is shown, e.g. Load batch, Finalize batch, Redo Event rules, etc.

Middle part

The middle part can be used to display user information, for example the last barcode found. This part can be configured in the subprofile for the info window
When a batch is loaded, automatically processed or finalized, this part is used to display a progress bar.

Right part

⊟ Here the program informs about its current state.

The program can be in one of the following states:

Ready

At the moment no action occurs, the program is waiting for a user action. For example a click on a control element that starts the task.

Processing

It will be scanned or the loaded images will be processed.

Task stopped

A command was given to end the task and to close the batch. e.g. suspend in OpenJob Mode or finalize in DirectMode.

The scanner was yet stopped but the remaining images will be fetched from the buffer and processed.

Task paused

A currently opened or scanned batch is ready and will be presented on the screen. This is the usual state for the interactive mode.

Testing license

The program loads a new profile and checks for all task steps the licenses needed.
For this the dongle or the DpuServer will queried. This action may take a few moments.

No valid license found!

The program license was not found. Please check the program is opened by another user in the background. Also check if the dongle is plugged in correctly, or if the DpuServer is reachable and there are free licenses.

Plugin licenses missing!

The license for one of the PlugIns was not found. Please check if the dongle is plugged in correctly, or if the DpuServer is reachable and there are free licenses.

Polling for file

The program waits for images to process in a automatic post-processing task.

Polling for batch The program waits for a batch to process in a automatic post-processing task.

In a Gold-Limited version a field indicates how fast the program may run.

As long as this field is green all components of the process works with full speed. If the color turns to red, it means that the program is delaying the execution of the next steps. This should not occur during the scan procedure as long the bought license type fits the scanners speed. But it may occur - on the same machine- if images were read from a fast mass storage or a virtual scanner is used instead of the real one.

3.2.5 Sound

In general the program works completely quiet.

But an acoustical feedback can be defined for some special events like Patchcode/Barcode found. For that the Function-DLL SOUND can be called in the eventrules.

3.3 Officemode

The Officemode provides simple setting for the most important parameters. A configuration of the base profile or the task profile as in Expert mode is not necessary here.

The settings for the scanner in the office mode comply with the scanner settings in expertmode.

The setting for the search carried out in most cases only a simple choice, for example, to look for which Barcodes or which Patchcodes.

Similarly, the steps for image enhancement can be activated by simply selecting.

The settings for the data destination are somewhat simpler compared to the Expert mode:

- Target Path
- Filename

3.3.1 Target Path

Here you can define the folder where the images will be stored.

Handle Existing Directory Specify, how the program treats existing directories.

Don't care and continue All files of the existing directory is overwritten if a filename is identical with an already existing filename. If you are sure that no file names are assigned twice, this option will simply add image files to an existing directory.

Prompt to continue The program warns, if a directory exists, before data is overwritten. You have the opportunity to continue or to stop the scan process.

Always delete directory All files in an existing directory are deleted before the first scan.

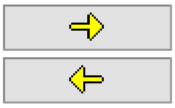
Prompt to delete directory This option warns before unwanted deleting of data. You have the opportunity to stop the scan job before it even starts.

Always cancel process If a directory with the same name already exists, the process is always stopped. Overwriting or deleting of data is thus excluded.

Main Path Defines the the folder for the images to store.

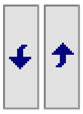
List (left) Here the placeholders are listed you can choose to build the target path. Available are

- Barcode (represented as BBBB) in the preview
- Date: Year 4 digits, month, day
- Time: Hour Minute, Second
- User name, computer name
- Dash and underscore



With these buttons the placeholders can be selected or de-selected resp.

List (right) Here the current parts of the target path definition are listed.



With these buttons the sequence of the placeholders can be re-ordered.

Preview Shows a preview of the path to create.

OK Saves the changes and closes the dialog.

Cancel Closes the dialog without saving the changes made to the settings.

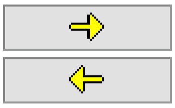
Help Opens the help topic for this dialog

3.3.2 Filename

Here one can specify the names of the files to be stored:

Starting filename for a new directory

List (left) Here the placeholders are listed you can choose to build the filenames.



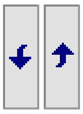
With these buttons the placeholders can be selected or de-selected resp.

List (right) Here the current parts of the filename definition are listed. Available are

- Barcode (represented as BBBB) in the preview
- Date: Year 4 digits, month, day
- Time: Hour Minute, Second

- Counter (each image, not sheet)
- Side: F or B
- User name, computer name
- Dash and underscore

The file extension is either ".PDF" or ".TIF", depending on the file type chosen.



With these buttons the sequence of the placeholders can be re-ordered.

Preview Shows a preview of the filename to create.

OK Saves the changes and closes the dialog.

Cancel Closes the dialog without saving the changes made to the settings.

Help Opens the help topic for this dialog

Hint: Make sure that each file has a different name, otherwise files can be overwritten and information gets lost. To be save, choose a counter in the file name.

3.4 Task Profiles

In Office mode, the program works with a **fixed predefined workflow** with specific steps that can be switched on or off and configured partially.

In Expert mode, a workflow can be freely configured. To perform a scan task many individual steps can be combined into one overall process. The list of all individual steps is the **task profile**, or simply "the **task**".

From the nature of a single step, it depends on whether it is running in the scanning process only once or repeatedly. The order of individual steps is fixed in some cases, in other cases, they can be changed according to the specific requirements. The task therefore describes the workflow during scanning.

The task will be executed after the start step by step from top to bottom. Using a **#if** statement can this process branch, for example, to run the OCR only on the first image after a document break. The flow can also be interrupted or completely halted.

It can be set up any number of tasks, tasks can pass control to other tasks. It is therefore e.g. not useful to map all assigned tasks in only one single task.

Scanned batches can be processed in the background or transferred to other stations.

Depending on the requirements can be distinguished:

⊟ **Course of events in a scan task**

A scan task consists essentially of three parts, one part of which is performed once at the beginning of the task (preparition) and a part which is repeatedly executed for each image (scanning loop).
It may end up giving a further part (overrun), which is also run through only once.

⊟ Course of events in an automatic post processing task

In the automatic post processing already scanned images will be edited again, e,g. converted into text with the OCR. The structure corresponds to a regular scan task, but the step "scan" will be replaced only by other processing steps. Although such a task must be started by an operator it requires then no further interventions (as long there is no exceptional situation).
A special type of automatic post processing are called polling tasks which can repeat themselves and automatically edit various source directories sequentially.

⊟ Course of events in a manual post processing task

In the manual post processing the scannig loop is omitted. The images are only loaded and presented to the operator.
The operator can improve, move, replace the images, or insert new images and delete existing ones.
If there is an index dialogue, he can also enter additional information.

⊟ Course of events in a background task

Depending on the working mode, the images can be re-edited here. At the end, they are either transferred to another station or the subsequent target system. It works automatically and if an error occurs in a batch it will continue with the next one.
The background task is firmly bonded to the task of which it is invoked after scanning, that is, it works with the **same** base profile.

Selecting, copying and deleting Tasks

In Office Mode and Expert Mode, Task profiles can be selected, copied, and deleted in ribbon field profiles.

For each mode the program provides a basic task profile.New profiles can be created by copying the basic task profile or an already adapted own one.

Only in Office mode, there is a task called **[OFFICE]**, which contains the most common operations. It can be there in office mode but neither viewed nor modified.

In Expert mode, a basic configuration too, named **DPS**. It also contains some basic steps. But in this mode it can be viewed and changed.

Change a task

In Office Mode the task can be customized via the Settings tab of the Ribbon. It is possible to check each step and disable, or configure them. e.g. you can adjust to search for what barcodes.

In Expert mode, there is a configuration dialog for the task it can be reached via the corresponding buttons on the ribbon or the toolbar. Here can also tasks selected, copied, deleted or changed. Especially, further actions can be added.

Starting a Task

One a task was chosen one can start it with the associated button on the ribbon tab start or the toolbar or the shortcut <Return>.

See also Help overview

3.4.1 Actions before the scanning

The part of the task, which is executed once before the actual scanning loop, is intended for loading the basic parameters and create or open a batch.

Loading parameters

Essential Actions:

- Load base profile

Commonly used actions:

- Load layout profile
- Set value of a variable (for example, the Endorser counter)
- Launch a dialogue
- Call a PlugIn (for example, a script)

Occasionally used actions:

- Set the counter for the expected images.
- Configure delay time.
- Load global counter
- Load or save a global string
- Show Info window
- Pause
- Message box (for example, the task can not or may not start)

Create or open a batch

In OpenJob mode

- Load batch
- Import batch (from another station)
- Polling for a batch (Watch a directory)
- Load batch externally (in a background task)
-

In Direct Mode

- Polling for files (Watch a directory)

3.4.2 Actions at the scanning

Obtaining the images in OpenJob

- Load from scanner
- Import batch from a directory
- Import batch from a list
- Import batch specific Files (individually select)
- Process images Images of batch again (automatic post processing)

Obtaining the images in Direct Mode

- Load from scanner
- Loading batchs of directory
- Load batch of protocol (list)
- Loading batchs of certain files (individually select)

Improving images

- De-skew images
- Rotate image
- Scale image
- Image enhancement

- Invert image colors
- Transform image with filter
- PlugIn start for each image (gamma, edge enhancement, color noise ...)
- Copy image
- Generate missing color formats

Examine the images

- Search barcodes
- Search patchcodes
- Text recognition
- PlugIn start for each image (2D barcodes, ...)
- Start a dialog for each image
- Set variable for each image

Sorting and grouping the images

- Running event rules (scanning)
- Running Event rules (automatic post processing)
- Keep input path, filename and format
- Keep input filename and format
- Set pathname
- Set filename

Storing the images

- Save Images
- Print Images

3.4.3 Actions to control the workflow

To be able to react to certain situations, it is possible to intervene in the process.

The following actions are available:

- Canceling Scan loop (if there follows a task overrun)
- Pause task
- End task
- Cancel task
- Start task again
- Exit program

3.4.4 Actions after the scanning

The actions after scanning are used to pass the batch of images to another station or to the target system. For automatic post processing tasks then the cleanup is to do.

- Print Info

- Export batch
- Finalize batch
- Export and finalize batch

- Finalize batch extern (starts DpuFinalize)
- Process batch extern (startet DpuProcess)

- Delete batch

- Delete images in source path
- Delete the full source path

Apart from the special overrun actions, other actions can be appended, which are otherwise provided for the task preparation at begin. For example, displaying a message or the info window.

Since not all of these actions signals automatically the end of the scanning loop to the program, it is possible that they will be ignored.
In this case it is useful to perform a "blind" action at first one that will be executed at all times, e.g .:

Set value of a variable with `%(dummy)=Overrun`

3.4.5 Actions for DpuFinalize

Not all actions are available for the DpuFinalize working mode, especially those ones which force a user action.

Following actions are available:

- Checking an #iIF-THEN-ELSE condition
- PlugIn calls (e.g. a script)
- Set value of a variable
- Load a globalen counter
- Load and save a global character string
- Lade Batch extern
- Print Info
- Redo event rule on the batch
- Export batch
- Finalize batch
- Restart task
- Close the program.
-

Please also ensure that external program parts, e.g. PlugIns do not interrupt the workflow through queries or messages on the screen.

3.4.6 Actions for DpuProcess

Not all actions are available for the DpuProcess working mode, especially those ones which force a user action.

Following actions are available:

- Checkin an #IF-THEN-ELSE condition
- Call a PlugIn aufrufen
- Set value of a variable
- Load global counter
- Load and save a global character string
- Load batch extern
- **Reload images for processing**
  - De-skew images
  - Rotate images
  - Scale images
  - Do an enhanced modification on the images
  - Revers image colors
  - Start a PlugIn for each image
  - Set a value for variable for each image
  - Search for barcodes
  - Search for Patchcodes

  - Text recognition
  - Event rules
  - Set path name
  - Set file name
  - Save the image
  - Print Info sheet
- Redo the event rules on the batch
- Export the batch
- Finalize the batch
- Restart the Task n
- End the program

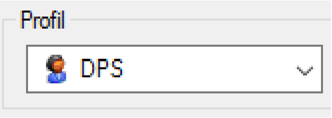
The action "**Reload images for processing**" MUST be set if one of the indented actions is to take place.

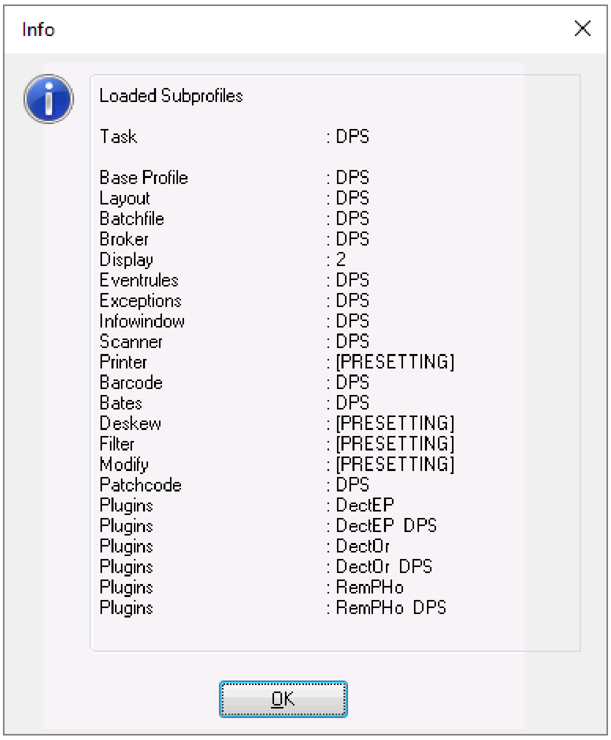
Please also ensure that external program parts, e.g. PlugIns do not interrupt the workflow through queries or messages on the screen.

3.4.7 Task Settings

In Expert Mode a dialog exists to manage and change a task profile. In Office Mode there is no such option.

This dialog box offers in the upper part the options to create or delete a task profile. Also the changes can be saved or the list of the connected subprofiles can be shown.

 Profil DPS	With the selection here the task profile will be loaded. Hence the attached base profile, subprofiles and the drivers for scanner and printer are **loaded** too, this may take a while.
	Opens the dialog to specify a new profile name. Then a copy of the profile currently selected will be created. Initially only the task is copied, the individual Subprofiles are not copied. That the old and the new task using initially **the same subprofiles**.
✗	Deletes the chosen task profile after a confirmation dialog.
✓	Saves the changes without closing the dialog.
ⓘ	Lists the used subprofiles:



Info

```
Loaded Subprofiles

Task                 : DPS

Base Profile         : DPS
Layout               : DPS
Batchfile            : DPS
Broker               : DPS
Display              : 2
Eventrules           : DPS
Exceptions           : DPS
Infowindow           : DPS
Scanner              : DPS
Printer              : [PRESETTING]
Barcode              : DPS
Bates                : DPS
Deskew               : [PRESETTING]
Filter               : [PRESETTING]
Modify               : [PRESETTING]
Patchcode            : DPS
Plugins              : DectEP
Plugins              : DectEP  DPS
Plugins              : DectOr
Plugins              : DectOr  DPS
Plugins              : RemPHo
Plugins              : RemPHo  DPS
```

In the lower part the dialog show the on the **left side** the available task steps and on the **right side** the currently used steps. The table on the rigt side has these columns:

Active This checkbox indicated if the step will be executed.

This corresponds to the set of an option in the tab settings of the In Office Mode. this

Actionlist Name of the task step.

Parameterlist Parameters for this action. This is the used subprofile and, if available, the set of images the action should process.

The buttons above the task list works as follows::

➔ Add	Inserts the selected action from left side **under** the currently marked action on the right side.
{IF} Condition	Inserts a condition under the marked line on right side.
☒	Deletes all task actions.
✕	Deletes the marked task action only
⬆	Moves the selected task action one position up. The lines can be shifted by drag & drop with the mouse too.
⬇	Moves the selected task action one position down.

OK	Closes the task configuration, changes will be saved.
Cancel	Closes the task configuration, changes will not be saved.
Help	Opens this help file.

3.4.8 List of all actions

Sort by activities

Load profiles, forerun, polling, Open batches, Getting images in OpenJob, Getting images in DirectMode, Scanning, Editing images, Recognize images, Sort images, Save images, Controlling the scan loop, Overrun.

Sorted by objects

Image, Folder and files, batch, task, profiles, program

Sorted by activities:

Load profiles

Load base profile
Load layout profile

Forerun

Setup image counter
Setup delay time
Pause
Start user defined dialog
Call PlugIn
Start messagebox
Show Infowindow
Set value of a variable
Set global counter
Load gloabal string
Save gloabal string

Polling

Poll for file

Open batches

Load batch
Load batch extern
Import batch
Poll for batch

Getting images in OpenJob

Reload images for processing from batch
Import from directory to batch
Import from batchfile to batchi
Import special files to batch

Getting images in DirectMode

Laod from directory
Load from batch file

Load particular files

Scanning

Load from Scanner

Editing images

Copy image
Deskew images
Rotate Images
Modify images
Bates stamping
Reverse images colors
Scale images
Transform images with filter
Start user defined dialog for every image
Call PlugIn for every image
Set value of a variable for every image

Recognizing images

Search barcodes on images
Search Patchcode on images
Search text on images

Sorting images

Check event rules
Keep input path, filename and format
Keep filename and format
Set pathname
Set filename

Saving images

Save image to disk
Print image

Controlling the scan loop

Exit scan loopn

Overrun

Pause task
Close task
Cancel Task
Print Info
Process Event rules on batch
Delete images in source path
Delete source directory
Export batch
Finalize batch
Export and finalize batchl
Finalize batch extern
Process batch extern
Delete batch
Restart task
Exit program

#IF-#ELSE-#ENDIF

Sorted by objects

Image

File and folder

Batch

- Delete a batch
- Pausing
- Polling on files
- Polling on batches
- Make a break
- Waiting for files
- Waiting for a batch

Task

- Cancel the work
- End the work
- Satrt a dialog
- Restart again
- Load a global counter
- Load a gloabal character string
- Signal a message
- Pausing
- PlugIn call
- Polling on files
- Leaving the the scan loop
- Specifieing a delay time
- Foreward a batch
- Repeat
- Set the value of a variable
- Save a global character string

- #IF-#ELSE-#ENDIF

Profiles

- Load the base profile
- Load a layout profile

Program

- Ending Program

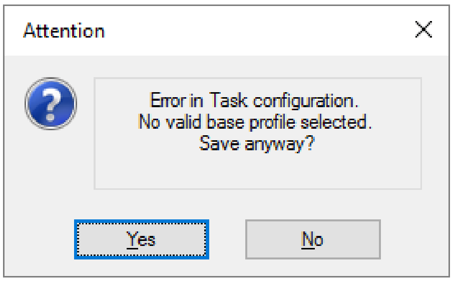
3.4.8.1 Load base profile

With this command one task is assigned to one Base Profile. The base profile with its subprofiles describes exactly how to perform the single task steps, for example, where and how the image files are stored.

Notes

This action is **always required** and must **precede all other** actions. It can not be deleted or moved.

Also the base profile must be specified, otherwise the task can not start . If an old task is imported from older versions of the program, missing this line, an error message cames up:



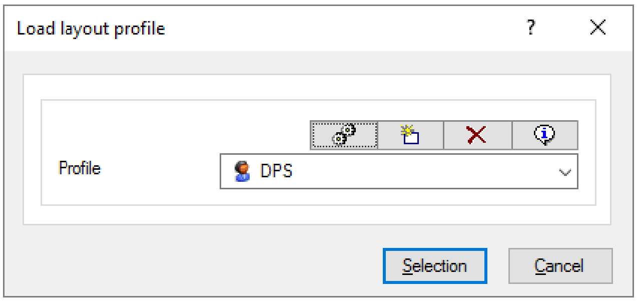
Attention

Error in Task configuration.
No valid base profile selected.
Save anyway?

An exception are the Background Tasks, which are started by the command Filalize batch extern or Process batch extern. Those types of tasks uses the same base profile as their "foreground Task" which triggers them.

3.4.8.2 Load layout profile

A layout profile contains the configuration of the **toolbars**, **buttons**, keyboard **shortcuts** and **macros**. Thus an own user interface can be collected for each task.



Load layout profile

Profile: DPS

Note

This action should be used for each profile in the beginning.

3.4.8.3 Setup image counter

This action opens the window to input the image counter before scanning begins.

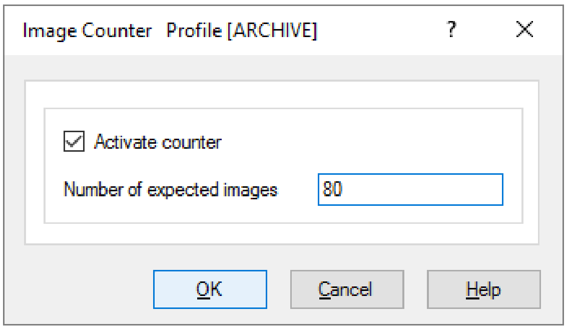


Image Counter Profile [ARCHIVE]

☑ Activate counter

Number of expected images: 80

The program shows a corresponding message if there more images or if the scanning ends before the number expected is reached.

Caution: In this version the **images** are counted, not the papers. In duplex scanning in dual stream one paper generates four images

Entering this amount at the beginning of the task is e.g. useful if the papers are previously passed through a counting machine and the batches containing different numbers of papers.

When in the job preparation, the papers are always combined into a batch of a *certain size*, so the image counter can also be given directly determined in the base profile.

If the value is additionally interrogated at the beginning of the task, it overrides the value in the base profile. So always the last entered value will apply.

This counter can also be queried and set in the variable

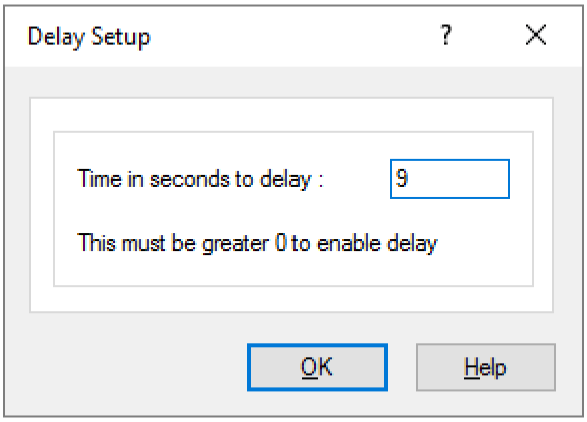
%(S.SCANSEXPECTED) or %e.

Note

This Task step should be carried out **before** the scan loop.

3.4.8.4 Setup delay time

Here, a time from 0 to 9 seconds can be given, the wait between two scans.



Delay Setup

Time in seconds to delay : 9

This must be greater 0 to enable delay

This is useful when large documents must be manually inserted, or if bound documents must be turned over in a book scanner manually.

Note
This Task step should be performed before the scanning loop.

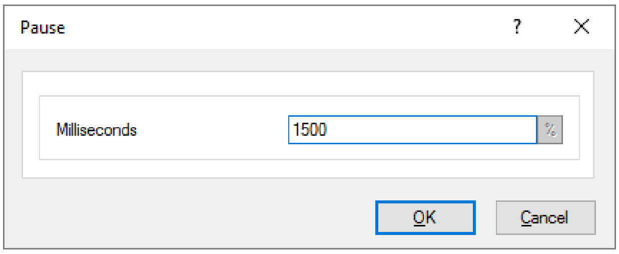
3.4.8.5 Pause

Here a time can be specified in milliseconds, to wait before starting the task.

While waiting, the processor will be charged as few as possible, so that other programs can be executed.

A programmed pause makes sense e.g. in an automatic post-processing task if you want to wait for a preceding process.

Note
The pause can be **before** or **after** the scan loop.



Pause

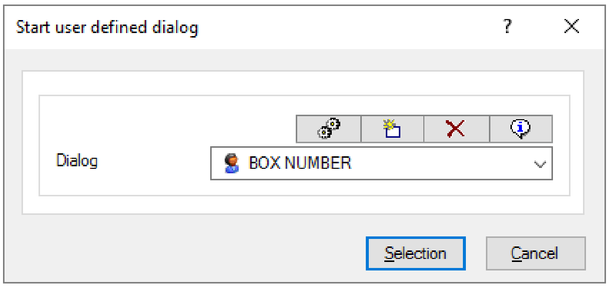
Milliseconds 1500

3.4.8.6 Start userdefined dialog

By calling a user-defined dialog (UDD) information can be obtained, which can not be detected by the program in another way.

This can e.g. be a handwritten note on a cover sheet at the beginning of the task, or the number of the box in which the paper was delivered. After scanning is a UDD e.g. to query the status of the stack.

To configure this task step, it is sufficient to specify a UDD:



Start user defined dialog

Dialog: BOX NUMBER

Notes

The UDD can be called **before** or **after** the scan loop; within the loop this statement has no effect! Please use in this case the task step Start userdefined dialog for every image.

To set the default values for the variables may e.g. the task step Set value of a variable can be used.

If a user defined dialog with the **Cancel button** closes, this cancels the task too!

For this task step, a person really must **operate** the program, that is, it should not be used in an automatic post-processing task. The use in a background task isn't possible anyway.
This function **can not** be **defined or changed** with the license type "Standard". But it can be used, if the setting was made with a license "Gold" or "Professional".

3.4.8.7 Call a PlugIn

Invokes a PlugIn. Selectable are PlugIns that have been previously loaded and configured for use with the Base Profile. If no PlugIns are loaded, this line remains disabled. A typical PlugIn call can e.g. be before scanning the execution of a script, which determines the job number or makes an entry in a database after scanning.

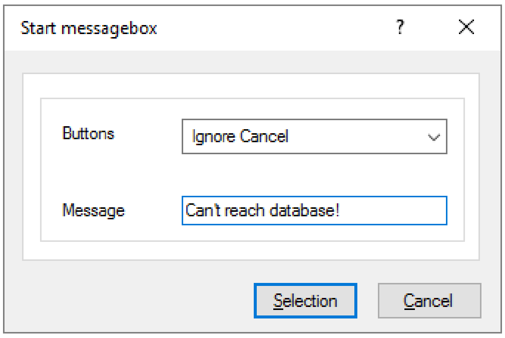
To set the parameters, there is a dialogue.

Note

This action can only be performed **before** or **after** a scan loop. For use within a scan loop there is the appropriate action Call PlugIn for every image.

3.4.8.8 Start Messagebox

With this instruction a simple message or question can be displayed.



Buttons: Ignore Cancel

Message: Can't reach database!

The outcome is stored in the variable %(S.MSGBOX), possible answers are:

OK	**1**
Cancel	**2**
Exit	**3**
Retry	**4**
Ignore	**5**
Yes	**6**
No	**7**

Note

This action can only be performed **before** or **after** a scan loop.

See also warning against the use in the scanning loop.

3.4.8.9 Show info window

Displays the info window with the selected subprofile as a popup window. Does nothing if the info window is already integral and visible part of the working area.

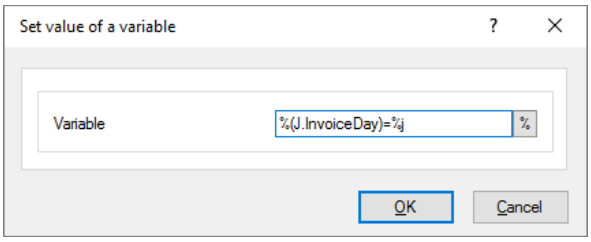
If it was opened by this Task action, it may be closed and re-opened by the appropriate button.

Note

This action should be carried out **before** a scan loop. If this action is to be executed **after** the scanning loop, it must **not be the first action,** see remark. See also warning against the use in the scanning loop.

3.4.8.10 Set value of a variable

Sets the value of a variable. Fix values or other %-Codes can be assigned.



Variable: %(J.InvoiceDay)=%j

Please regard the rules for assignments.

Notes

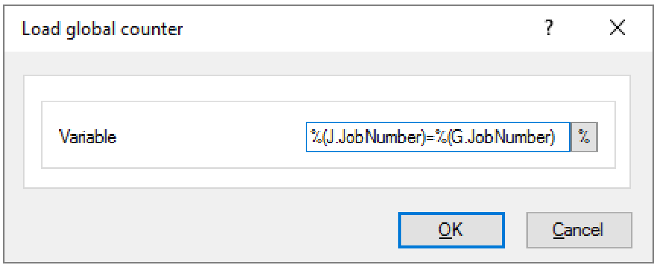
This action can only be performed **before** or **after** a scan loop. For use within a scan loop there is the appropriate action Set value of a variable for each image.

Please note that an assignment is only useful if the object already exists, referred to by the scope. For example, is assigning meaningless to a image variable %(I.Number), if one has not yet been scanned and there is still no image. On the other hand, one should not count on that the variables are immediately "forgotten" when there is no longer the object.

It is not possible to overwrite so-called system information values, e.g. the current day or the computer name.

3.4.8.11 Load global Counter

This Task step copies the value for a global counter in a local variable in the form of an assignment. For example,



Load global counter

Variable: %(J.JobNumber)=%(G.JobNumber)

The value is picked up from DpuServer and there then automatically increased by one. If the counter does not exists yet, it is created and returned as 0. then its value is set to 1.

The prefix G. in the above example has specified the right variable as a **global variable**. Global variables can be queried or **set only with the appropriate task steps**. Elsewhere, e.g. at a normal value assignment or the output to a log file, they will simply be ignored and replaced by an empty string!

With the help of a global counter may e.g. a unique batch number are assigned when scanning at different locations.

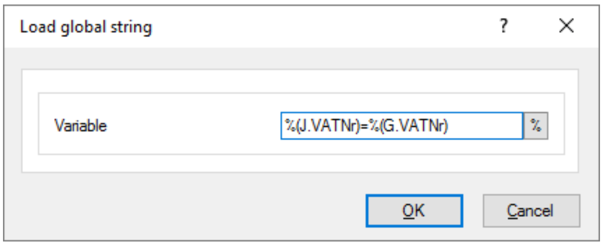
The value of this counter can be reset on the DpuServer using the DpuEnterpriseManager. In future releases, this behavior is emulated for operation without a DpuServer.

Note:

This action should be carried out **before** a scan loop. If this action is to be executed **after** the scanning loop, it must **not be the first action,** see remark. See also warning against the use in the scanning loop.

3.4.8.12 Load gloabal string

This Task step copies the value of a global variable in a local variable by assigning a value. For example:



Load global string

Variable: %(J.VATNr)=%(G.VATNr)

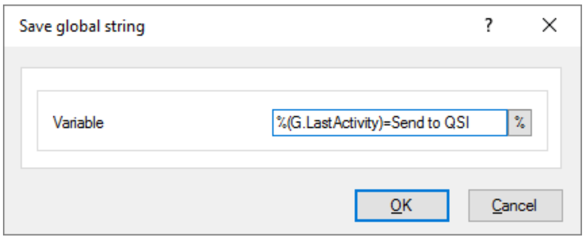
In this assignment the value **`%(J.VATNr.)=%(G.VATNr)`** the prefix of the prefix G. has made the right variable as a global variable. Global variables can be queried or set only with the appropriate task steps. Elsewhere, e.g. at a normal value assignment or the output to a log file, it will simply be ignored and replaced by an empty string.

Note:

This action should be carried out **before** a scan loop. If this action is to be executed **after** the scanning loop, it must **not be the first action,** see remark. See also warning against the use in the scanning loop.

3.4.8.13 Save global string

This Task step copies the value of a local variable in a global variable on the DpuServer by assigning a value. For example:.



In this assignment `%(G.LastActivity)=Send to QSI` the prefix G. of the left value has made this variable as a global variable. Global variables can be queried or set **only with the appropriate task steps**. Elsewhere, e.g. at a normal value assignment or the output to a log file, it will simply be ignored and replaced by an empty string!

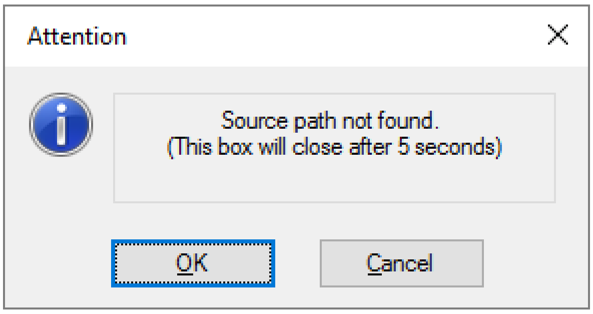
Note:

This action should be carried out **before** a scan loop. If this action is to be executed **after** the scanning loop, it must **not be the first action,** see remark. See also warning against the use in the scanning loop.

3.4.8.14 Poll for file

This statement puts the program in a waiting state in which it seeks in a certain input folder for image files, which it then processes. When all the images have been processed, the triggering file is moved to a specified output folder. The directory containing the images will be automatically deleted to make room for the next round.

The configuration of these parameters is performed in the base profile. While the program waits for new images, a message appears:



Source path not found.
(This box will close after 5 seconds)

Usually, such a polling task is executed repeatedly, that at the end of such a task the last statement is Restart task.

The polling process works like any normal scan task, in the foreground, that is the program can not be used at this time in any other way. The process can be stopped by the "End Task" button. For unattended processes in the background, the operating modes DpuFinalize or DpuProcess are provided.

Notes:
This statement only works in Direct Mode, for the Open Job mode there are the corresponding task step Polling on batch.

This action can only be performed **before** a scan (or process) loop.

This function **can not** be **defined or changed** with the license type "Standard". But it can be used, if the setting was made with a license "Gold" or "Professional".

3.4.8.15 Load batch

With this instruction a stack is created or opened, which allows further processing steps. Possible are setting marks, drag and drop, manual breaks or interrupting and the subsequent continuation of the work on such a batch.

To set the parameters, there is a detailed setting dialog. This dialog will be offered (in a reduced form) to the operator for selecting a batch displayed.

Notes:
This action can only be performed **before** a scan loop.

By inserting this step, the task is operating in OpenJob mode. Some actions are available now, while other actions are tailored to the Direct Mode no longer available.

It makes sense to give the operator the opportunity to check whether he has opened the right batch. This should be done after loading the batch with the task step Pause task.

3.4.8.16 Load batch extern

This Task step is for a background task and corresponds to step Load batch as e.g. is used in a scan task. The Task step is performed without any parameters, because these are taken from the settings in the Base Profile.

Note:
This step is only available if a background task is created by the task steps Finalize batch extern or Process batch extern or by pressing of of the appropriate buttons in the toolbar.

This statement is the **first statement** in a background task. It can neither be **deleted nor moved**.

3.4.8.17 Import batch

This Task step imports a an existing batch from a specified directory. This directory can be defined in the base profile. In this procedure, the directory is always copied first and then the source directory is removed. Of course the deletion will occur only after complete and successful copying.

Notes:

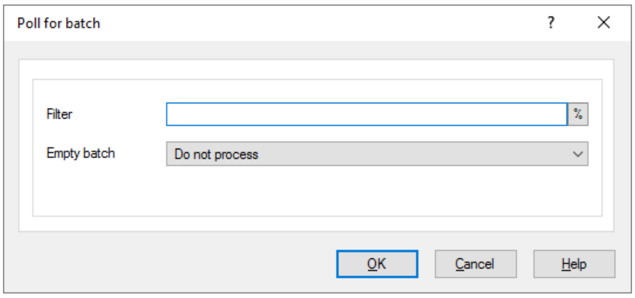
This action can only be performed **before** a scan loop. For sending a batch after the loop, there is the command Task export.

By inserting this step, the task is operating in OpenJob mode. Some actions are available now, while other actions are tailored to the Direct Mode no longer available.

It makes sense to give the operator the opportunity to check whether he has opened the right batch. This should be done after importing the batch with task step Pause task.

3.4.8.18 Poll for batch

This Task step causes the program in a given directory to search for open batches. If such a batch is found, the further processing starts automatically. The selection of the loaded batches may be restricted by this dialog:



Poll for batch

Filter:

Empty batch: Do not process

This action is useful for an automatic post-processing task that requires no operator. As with other polling tasks, it is common that the task re-starts again at the end.

Notes:

This action can only be performed **before** a scan loop, or processing loop.

By inserting this step, the task is operating in OpenJob mode. Some actions are available now, while other actions are tailored to the Direct Mode no longer available.

This function **can not** be **defined or changed** with the license type "Standard". But it can be used, if the setting was made with a license "Gold" or "Professional".

3.4.8.19 Reload images for processing from batch

This action is intended for automatic post-processing tasks. It must be inserted where other image-related actions are to be executed, e.g. an image enhancement or recognition.

It initiates the processing loop and is used in place of the action Load from scanner.

That is, for example, useful if the time-consuming OCR is to be executed for all images of a stack not on scan workstation but later on another station.

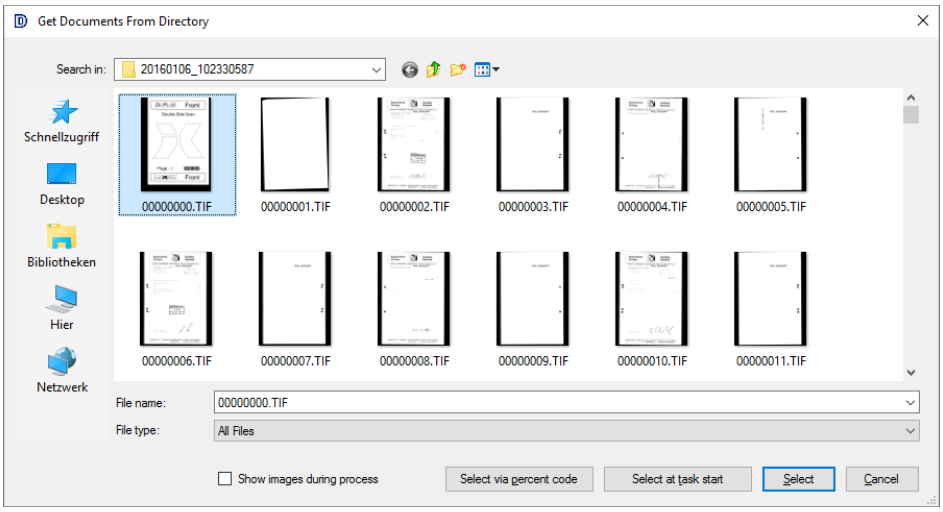
Note:

This action can only be performed **within** a processing loop. It must be **the first action** there.

3.4.8.20 Import from directory to batch

A series of images can also be loaded instead of a scanner directly from a directory. In this way, scanned images can already be later edited or only displayed. For this task step a fix directory can be specified when defining the task step.

Unlike the system dialog for opening a **directory** here a dialog for opening an image **file** comes up:



If there (still) are no files, the selection can be made via Percentcodes, or this dialogue will be displayed to the operator **at task start**. With the flag **Show images during process** this can be watched.

In this case the dialog can then be closed with OK only, if the operator selects an image file. This file **and all other files** in this folder will be loaded then.

The start directory for this selection can be specified in the Base Profile. With the specification of file name masks the file can be filtered and to distinguish between front and back sides.

Notes

This task step is reserved for the OpenJob mode only, e.g. an open job must be loaded before. For the DirectMode there is a corresponding step named Load from directory.

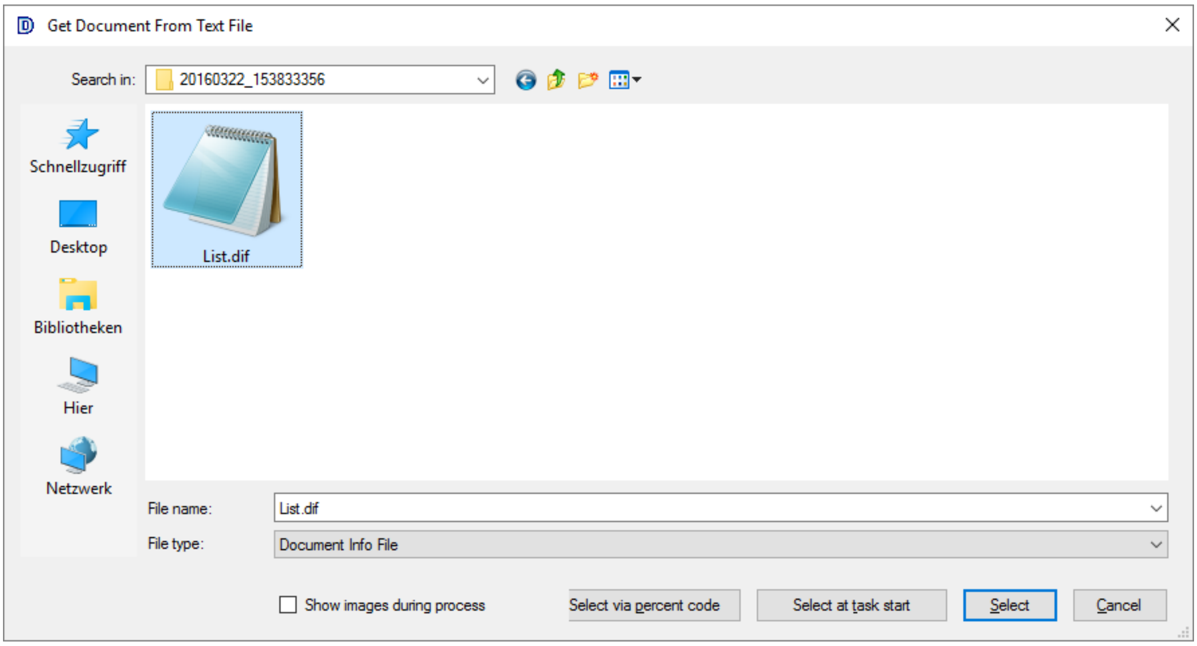
This step must stand **inside** the processing loop. It must be the **first step** there.

3.4.8.21 Import from batch file to batch

This Task step causes the reading of all the files in a **document info files (* .DIF)** are listed. DIF files are the standard type of output for batch files (report files) generated by the program.

Basically, however, any text file can be used, which lists an image file per line of text. So the file list can be filtered by TXT or CSV too.

You must specify the absolute path of the batch file.



Relative paths or wildcards are not allowed, but if there is no file (yet), the selection can be made via Percentcodes, or this dialogue will be displayed to the operator **at task start**. With the flag **Show images during process** the loading can be watched.

With the specification of file name masks the file can be filtered and to distinguish between front and back sides.

Notes

This task step is reserved for the OpenJob mode only, e.g. an open job must be loaded before. For the DirectMode there is a corresponding step named Load from directory.

This step must stand **inside** the processing loop. It must be the **first step** there..

Hint:

Lists of files can be produced subsequently by the redirection of DOS commands you to a file, e.g:

```
C:\SCAN>dir *.TIF /S/B >List.TXT
```

3.4.8.22 Import special files to batch

The program loads a single or a custom amount of image files from one folder. All required files must be selected in the file dialog box. This Task step does not have any additional parameters.

When run, it opens a file selection dialog where the user can select individual files. The home directory for this selection can be specified in the Base Profile. With the specification of file name masks the file can be filtered and to distinguish between front and back sides.

Notes

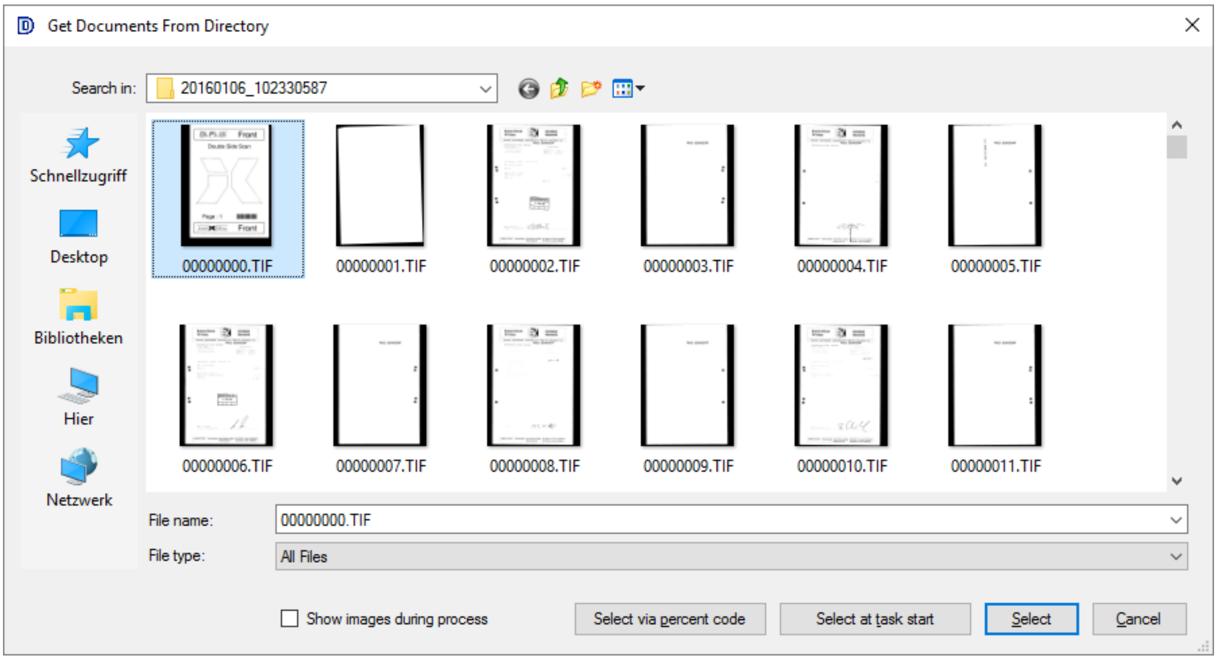
This task step is reserved for the OpenJob mode only, e.g. an open job must be loaded before. For the DirectMode there is a corresponding step named Import special files to batch

This step must stand **inside** the processing loop. It must be the **first step** there.

3.4.8.23 Load from Directory

A series of images can also be loaded instead of a scanner directly from a directory. In this way, scanned images can already be later edited or only displayed. For this task step a fix directory can be specified when defining the task step.

Unlike the system dialog for opening a **directory** here a dialog for opening an image **file** comes up:



If there (still) are no files, the selection can be made via Percentcodes, or this dialogue will be displayed to the operator **at task start**. With the flag **Show images during process** this can be watched.

The dialog can then be closed with OK only, if the operator selects **one** single image file. This file **and all other files** in this folder will be loaded then.

The start directory for this selection can be specified in the Base Profile. With the specification of file name masks the file can be filtered and to distinguish between front and back sides.

Notes

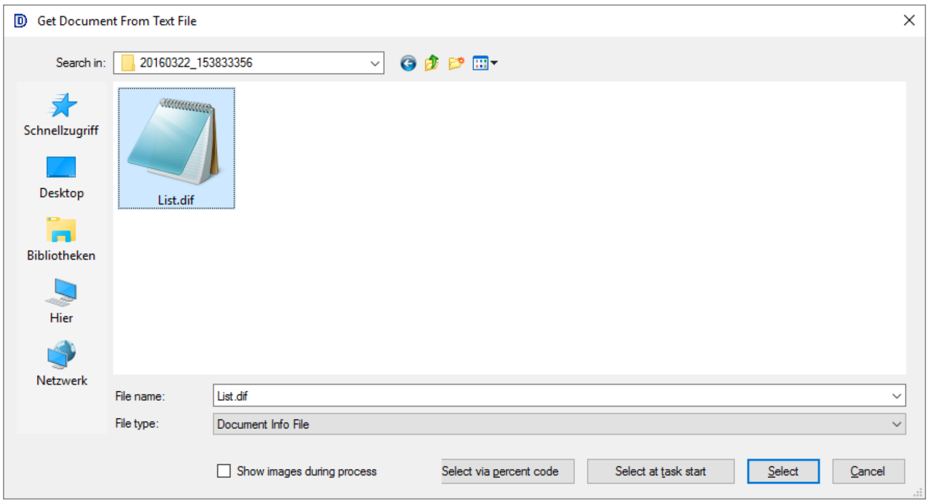
This task step is reserved for the DirectMode only, e.g. no open job can be loaded before. For the OpenJob mode there is a corresponding step named Import from directory to batch.

This step must stand **inside** the processing loop. It must be the **first step** there.

3.4.8.24 Load from batch file

This Task step causes the reading of all the files in a **document info files (* .DIF)** are listed. DIF files are the standard type of output for batch files (report files) generated by the program.

Basically, however, any text file can be used, which lists an image file per line of text. So the file list can be filtered by TXT or CSV too.



Relative paths or wildcards are not allowed, but if there is no file (yet), the selection can be made via Percentcodes, or this dialogue will be displayed to the operator to **Select at task start**. With the flag **Show images during process** the loading can be watched.

With the specification of file name masks the files can be filtered and to distinguish between front and back sides.

Notes

This task step is reserved for the DirectMode only, e.g. no open job can be loaded before. For the OpenJob mode there is a corresponding step named Import from batch file to batch.

This step must stand **inside** the processing loop. It must be the **first step** there.

Hint:

Lists of files can be produced subsequently by the redirection of DOS commands you to a file, e.g:

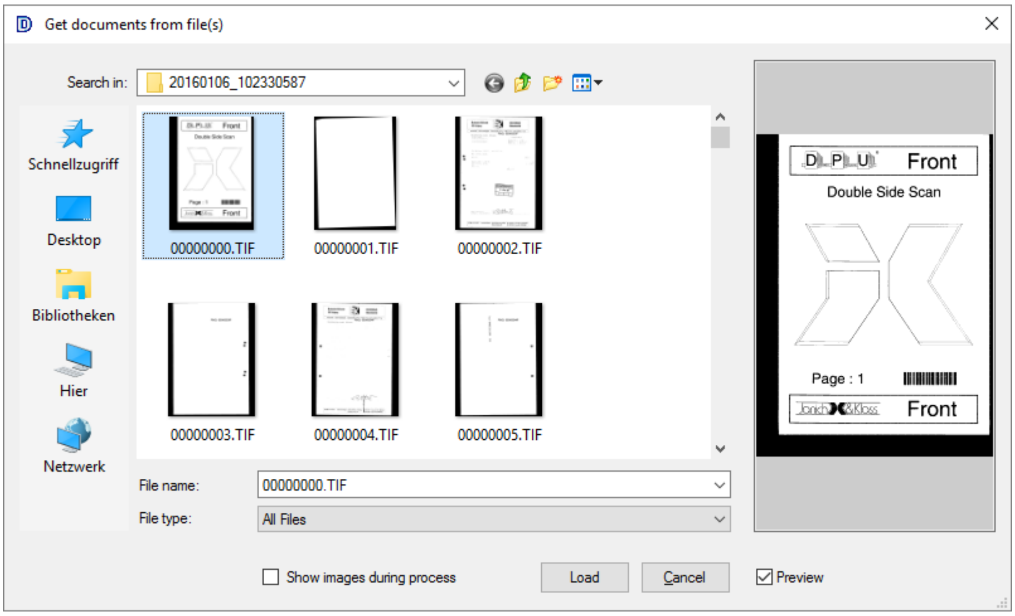
```
C:\SCAN>dir *.TIF /S/B >List.TXT
```

3.4.8.25 Load particular files

The program loads a single or a custom amount of image files from one folder. All required files must be selected in the file dialog box. This Task step does not have any additional parameters.

When run, it opens a file selection dialog where the user can select individual files. The starting folder for this selection can be specified in the Base Profile. With the specification of file name masks the file can be filtered and to distinguish between front and back sides.

Later in the task run the operator will see a select file dialog:



File name: 00000000.TIF

File type: All Files

□ Show images during process

☑ Preview

Notes

This task step is reserved for the DirectMode only, e.g. no open job can be loaded before. For the OpenJob mode there is a corresponding step named Import from batch file to batch.

This step must stand **inside** the processing loop. It must be the **first step** there.

3.4.8.26 Load from Scanner

This function starts the collection of images from the scanner. When two scanners are given it uses the activaed one. The parameters for this task step serve the scanner settings, as defined in the Base Profile.

The scan usually ends when no more paper is there, an error occurs or the operator presses a stop button. It may also be terminated by an action in the event rules.

Note:

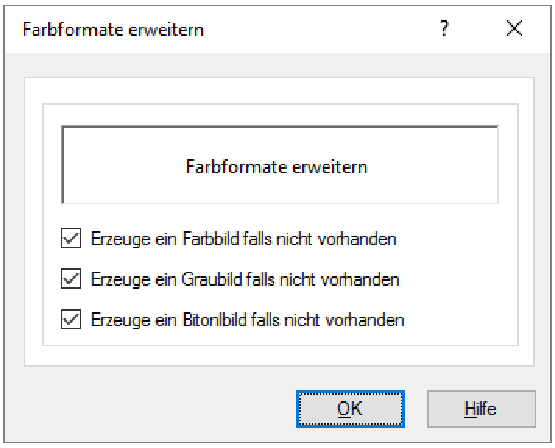
This step must stand **inside** the processing loop. It must be the **first step** there.

The parameters for this step can be set **after** a scanner subprofile was selected or created before in the base class.

3.4.8.27 Farbformate erweitern

In this step, missing color formats can be supplemented.This occurs e.g. when the scanner alternately send sometimes color, sometimes black-and-white and sometimes both. It also occurs, when a multi-page file is imported with images of different color types.

For many subsequent steps (and subsequent systems), it is easier to work with a continuous data stream and not have to check before each step whether the image exists.
The settings dialog allows you to specify which images are to be created.



☑ Erzeuge ein Farbbild falls nicht vorhanden

☑ Erzeuge ein Graubild falls nicht vorhanden

☑ Erzeuge ein Bitonlbild falls nicht vorhanden

When this step is executed, the program creates the corresponding image as a copy, not as a blank image.

Notes:

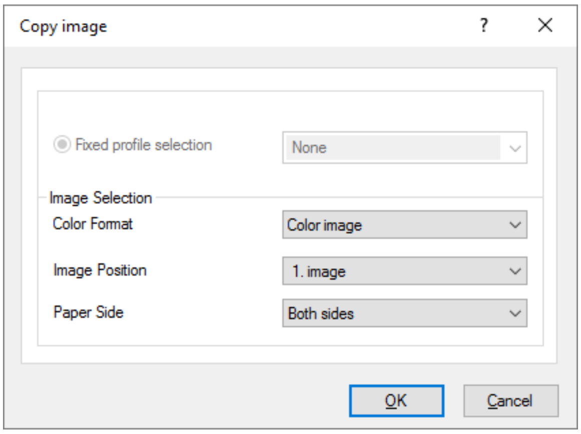
The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

3.4.8.28 Copy image

With this Task step an image can be copied. This is e.g. useful if you want to archive images as authentically as possible but on the other hand, must process the image for recognition. Then a copy can be created, improved, recognized and can be deleted later.

One can specify the image type number and side to copy:



○ Fixed profile selection: None

Image Selection

Color Format: Color image

Image Position: 1. image

Paper Side: Both sides

⊟ For an image to be processed it can be distinguished:

Color Format — **Bitonal image**, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position — The position defines the number, if there are more than one image of this

color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

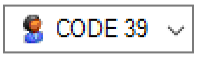
3.4.8.29 Deskew images

By this action, an image is straightened while the black edges are removed or "exempt". The image is practically not only straightened by this action, but also moved to the top left corner of the image window. By cutting off the black edges, the image is cropped at the same time to its original size.

In batches of documents with differt sizes it defined simplistically the maximum sheet (for example, A3). All smaller documents are then still cut to the right size.

The parameters for these actions are specified in the relevant Subprofile of the base profile. The task configuration just specifies which subprofile is used and on which images it should be applied.

⊟ The control elements for selecting and handling subprofies have the following meanings:

 CODE 39	The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.
⚙	Opens a dialog to change the currently selected subprofile or parameter set.
✳	Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.
✕	Removes the selected subprofile or parameter set, if it is not used elsewhere.
ⓘ	Shows the usage and further information of the currently selected subprofile or parameter set.

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this

color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

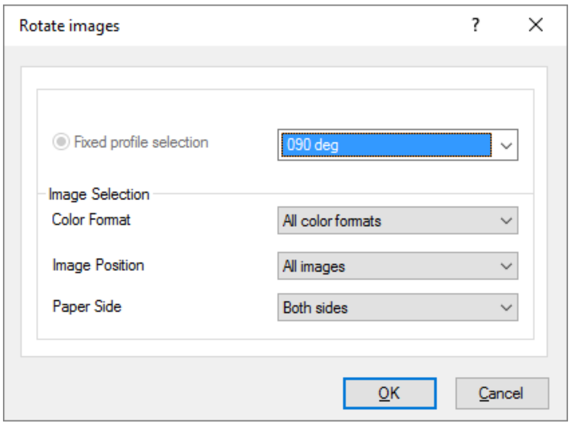
The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

3.4.8.30 Rotate images

This task step performs a rotation of 90 °, 180 ° or 270 ° from each image. It provides a quick and easy way is to straighten images that, for example, were scanned in landcape mode (on the short way) to save time.

There no subprofile for this step, the setup dialog is limited to the angle of rotation:



Rotate images

○ Fixed profile selection: 090 deg

Image Selection

Color Format: All color formats

Image Position: All images

Paper Side: Both sides

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side **Both sides** selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

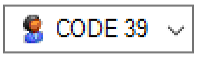
For a fine rotation or a precise rotation with an upstanding result the usage of the PlugIn **DectOR** is recommended.

3.4.8.31 Modify Images

This function starts the advanced image editing process to remove dirt, lines or shadows of a black and white image.

These actions are specified in the relevant Subprofile. The task configuration specifies which subprofile is used and on which images it should be applied

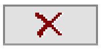
⊟ The control elements for selecting and handling subprofies have the following meanings:



The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.

Opens a dialog to change the currently selected subprofile or parameter set.

Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.


Removes the selected subprofile or parameter set, if it is not used elsewhere.

Shows the usage and further information of the currently selected subprofile or parameter set.

⊟ For an image to be processed it can be distinguished:

Color Format **Bitonal image**, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

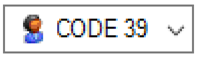
This step works with **black and white images only**.

The functions *remove lines* or *remove points* may spoil the search for barcodes or QR-Code for example. To avoid this please use this function **after** the recognition phase.
This function **can neither** be **defined nor** be **used** with the license type "Standard".

3.4.8.32 Bates stamping

This task step brings a text or image stamp in an image. The parameters for these actions are specified in the relevant Subprofile of the base profile. The task configuration just specifies which subprofile is used and on which images it should be applied.

⊟ The control elements for selecting and handling subprofies have the following meanings:

 CODE 39	The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.
⚙	Opens a dialog to change the currently selected subprofile or parameter set.
✱	Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.
✕	Removes the selected subprofile or parameter set, if it is not used elsewhere.
ⓘ	Shows the usage and further information of the currently selected subprofile or parameter set.

Please note that the color formats - are defined in Subprofile and here "all color formats" is selected fixed - notwithstanding other task steps.

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

When selecting the color types here, don't forget to enable the same color types in the subprofile.

The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

Stamped text will be recognized at the OCR like regular text. To avoid this please use this function **after** the recognition phase.
This function **can neither** be **defined nor** be **used** with the license type "Standard".

3.4.8.33 Reverse image colors

During this action, the colors of the image are reversed, this is as useful if the binarization of a color image large proportions would result from white text on a black background. The conversion has no further parameters, so here can only be set which images to be processed.

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

The stak step must stand **inside** a processing loop.

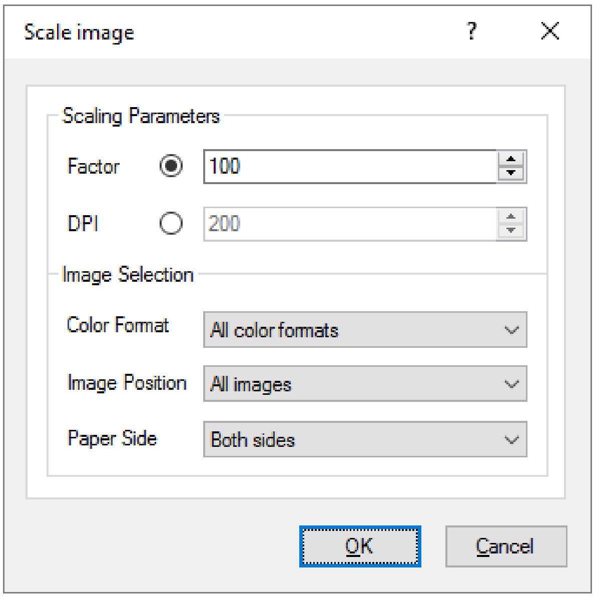
For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

3.4.8.34 Scale image

In this task, step, the selected images are scaled. This is e.g. useful if was a very high resolution is scanned to allow the detection of fine text or barcodes. But on the other hand, only an image with a normal resolution should be archived.

A scale is also necessary, if images was scanned in multistream with a unique resolution, but the different image types shall bes stored with different resolutions. For example the 200 dpi for black and white and 150 dpi for color.

There no subprofile for this step, the setup dialog is limited to the factor or the target DPI value::



Scale image

Scaling Parameters

Factor ● 100

DPI ○ 200

Image Selection

Color Format: All color formats

Image Position: All images

Paper Side: Both sides

In the field scaling parameter either a fixed target value for the resulting resolution in **DPI** can be given or the **factor** in percent.

A rising of the scaling factor over 100% is possible but it doesn't better the image quality.

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

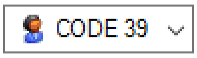
The reduction of the resolution may spoil the recognition of texts and other codes.To avoid this please use this function **after** the recognition phase.

3.4.8.35 Transform image with filter

During this action color filtering is applied to the scanned image. The scanned image will be altered by using filters to optimize it for further processing or storage. For this purpose certain colors are emphasized and other colors are suppressed as background colors.

Which filter is used is specified in the relevant Subprofile. The task configuration specifies which subprofile is used and on which images it should be applied. Here, the color format is limited to color images and gray images.

⊟ The control elements for selecting and handling subprofies have the following meanings:



The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.

⚙ Opens a dialog to change the currently selected subprofile or parameter set.

✱ Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.

✕ Removes the selected subprofile or parameter set, if it is not used elsewhere.

ℹ Shows the usage and further information of the currently selected subprofile or parameter set.

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes:

The stak step must stand **inside** a processing loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action
This function can only be **configured** with the license "Gold". But it can be **used** with "Gold" or "Gold limited".

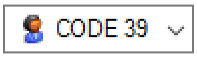
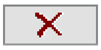
3.4.8.36 Start user defined dialog for every image

This step opens a dialog box to accept some user input, for example a replacement for an unreadable barcode. For that the custom dialogs, UDDs, are utilized. These can be configured in the base profile.

The pop-up of a dialog during the scan procedure is unusual since the scanning is halted. So it should be used in exceptional situations only. An exception may occur, if an insured property of an image is missing and user is asked to continue the work anyway. For a permanently dialog with the user a UDD may be shown as embedded part of the workspace. For that the use of the PlugIn JK_INDEX is recommended.

Here in the task configuration the UDD can be selected like a Subprofile:

⊟ The control elements for selecting and handling subprofies have the following meanings:

	The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.
	Opens a dialog to change the currently selected subprofile or parameter set.
	Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.
	Removes the selected subprofile or parameter set, if it is not used elsewhere.
	Shows the usage and further information of the currently selected subprofile or parameter set.

Notes:

The stak step must stand **inside** a processing loop. For the call before or after the loop, please use the step Start user defined dialog.

If a UDD was closed with the **cancel button**, the task will be stop too.

For the work with a UDD a person must **handle** the dialog. So it should not be used in an automatic post processing task. Also the usage in a background task is not possible.

If a UDD pops up when the scanner works in queuing mode the triggering image may no longer be visible.
This function **can not** be **defined or changed** with the license type "Standard". But it can be used, if the setting was made with a license "Gold" or "Professional".

3.4.8.37 Call a PlugIn for every image

Calls a PlugIn in a scan loop for each image. Selectable are PlugIns that have been previously loaded and configured for use with the Base Profile. To set the parameters, there is a dialog box.

Notes:

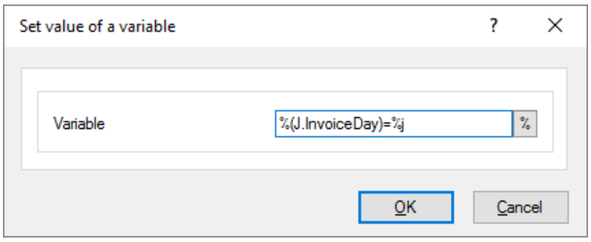
The stak step must stand **inside** a processing loop. For the call before or after the loop, please use the step Call PlugIn

For PlugIns changing or inspecting images there must be an images, i.e. this step has to occur **after** a **Load from...** action.

PlugIns expecting **user actions** should not be called without a user, i.e neither in a post processing task nor in a background task.

3.4.8.38 Set value of a variable for every image

Sets the value of a variable. Fix values or other %-Codes can be assigned.



Set value of a variable

Variable: %(J.InvoiceDay)=%j

Please regard the rules for assignments.

Notes

This action can only be performed **within** a scan loop. For use before and after a scan loop there is the appropriate action Set value of a variable.

Please note that an assignment is only useful if the object already exists, referred to by the scope. For example, is assigning meaningless to a image variable %(I.Number), if one has not yet been scanned and there is still no image. On the other hand, one should not count on that the variables are immediately "forgotten" when there is no longer the object.

It is not possible to overwrite so-called system information values, e.g. the current day or the computer name.

3.4.8.39 Barcode search on images

Through this Task step is searched on the image for a barcode. This barcode (or codes) can be sent in a barcode file, status file, or to the broker. They can also be used for further sequential control or for the name of the stored files.

The details for a barcode search are summarized in a subprofile which can be configured in the base profile. The task configuration specifies which subprofile is used and on which images it should be applied

⊟ The control elements for selecting and handling subprofies have the following meanings:

CODE 39 ⌄	The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.
⚙	Opens a dialog to change the currently selected subprofile or parameter set.

Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.

Removes the selected subprofile or parameter set, if it is not used elsewhere.

Shows the usage and further information of the currently selected subprofile or parameter set.

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

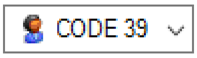
After the barcode search the image may be rotated by 90°,180° or 270°, see options for barcode search.

3.4.8.40 Patchcodes search on images

Through this task step is searched on the image for a patchcodes. The found patchcode can be used for further sequential control.

The details for a patchcode search are summarized in a subprofile which can be configured in the base profile. The task configuration specifies which subprofile is used and on which images it should be applied

⊟ The control elements for selecting and handling subprofies have the following meanings:



The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.

Opens a dialog to change the currently selected subprofile or parameter set.

Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.

Removes the selected subprofile or parameter set, if it is not used elsewhere.

Shows the usage and further information of the currently selected subprofile or parameter set.

⊟ For an image to be processed it can be distinguished:

Color Format **Bitonal image**, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side **Both sides** selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes

This action can only be performed **within** a scan loop.

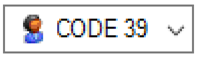
For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

3.4.8.41 OCR on images

This Task step starts in the image a text search. Depending on the parameters set a field search at certain places is either started in the image or a transformation of the whole page.

For this option is available, the OCR PlugIn JKFR60 must be loaded. The details for the text recognition are summarized in a subprofile which can be configured the base profile. The task configuration specifies which subprofile is used and on which images it should be applied

⊟ The control elements for selecting and handling subprofies have the following meanings:


CODE 39

The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.

Opens a dialog to change the currently selected subprofile or parameter set.

✳	Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.
✕	Removes the selected subprofile or parameter set, if it is not used elsewhere.
ⓘ	Shows the usage and further information of the currently selected subprofile or parameter set.

⊟ For an image to be processed it can be distinguished:

Color Format

Bitonal image, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

Notes

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

This function **can neither** be **defined nor** be **used** with the license type "Standard".

3.4.8.42 Check event rules

In this task, step previously defined event rules are processed. They responds to specific event with a particular action, such as with creating a new folder when finding a patchcode.

In Base Profile, there in the dialog to the event rules the different conditions and associated actions can be defined.

In the task configuration, this step can only be inserted, there is therefore no further adjustments.

Like all other actions in the task list the rules will be executes at once for all images of an image group of one side of the paper.

Notes

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

By event rules normally the file and directory-breaks are defined. Thus, this step must occur prior to the actions **Set pathname** and **Set filename**.
This function **can not** be **defined or changed** with the license type "Standard". But it can be used, if the setting was made with a license "Gold" or "Professional".

3.4.8.43 Keep input path, filename and format

This Task step is intended for automatic post-processing tasks in DirectMode.

If you want to read in already scanned images from the hard drive , for example, to rotate it or edit it in any other way, then by this action it can be determined that the images remain under the same name in the directory from which they were read.

Notes

This action is available for **DirectMode only**. The *similar action* Keep filename and format is available in DirectMode and OpenJob Mode.

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

Please regard also the rules for generation of path and filenames.

3.4.8.44 Keep filename and format

This Task step is intended for automatic post-processing tasks. If you want to read in already scanned images from the hard drive , for example, to rotate it or edit it in any other way, then by this action it can be determined that the images keep their name in the new destination directory.

Notes

This action is available for **DirectMode** and **OpenJob Mode**. The *similar action* Keep input path, filename and format is available in DirectMode **only**.

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

Please regard also the rules for generation of path and filenames.

3.4.8.45 Set path name

The program generates a storage path where the scanned image is to be stored. The options for generating the path can be made in the dialog box for the target directory.

Notes

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

Basically, it is always given the pathname **first** and **then** the file name! Please also regard the rules for generation of path and filenames.

3.4.8.46 Set file name

The program generates files where the scanned image is to be stored. The options for generating the filename can be made in the dialog box for the.

Notes

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

Basically, it is always given the pathname **first** and **then** the file name! Please also regard the rules for generation of path and filenames.

3.4.8.47 Save image to disk

With this task step, the image is saved. There is no separate subprofile for this action, because the storing options are part of the base profile and can be specified there.

Notes

This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

Also a **file name** and a **path name** must exist, so this action must stand **after** the steps doing this.

This action is usually the **last step within** the scan loop.

It must not be ommited in scan tasks, automatic post-prosssing tasks or background tasks. Only in a manual post-processing task it can be dispensed.

3.4.8.48 Print image

Using this function images will be printed. If this step becomes part of a scan task, the program works like a copier.

The parameters for these actions are specified in the relevant Subprofile of the base profile. The task configuration just specifies which subprofile is used and on which images it should be applied.

⊟ For an image to be processed it can be distinguished:

Color Format	**Bitonal image**, also black & white, 1 BPP color depth **Gray image**, also Grayscale image, 8 BPP **Color image**, 24 BPP **Automatic**, takes the best one for this procedure **All,** chooses all formats.
Image Position	The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying. **That is not the number in a file!** Depending on the action also **all images** may be chosen
Paper side	**Both sides** selects images for front and rear side aus **Front side** chooses images captured by the front side camera of the scanner. **Rear side** chooses the back side images.

Notes

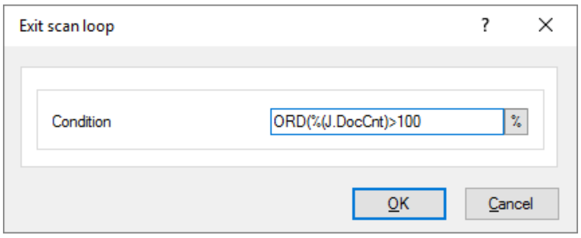
This action can only be performed **within** a scan loop.

For this step an image must exist, i.e. it must occur **after** a **Load from...** action.

This action prints all images (or a selected subset). To print a single image please use button in the Edit-Menu.

3.4.8.49 Exit scan loop

By this task step the end of the loop can be specified. If a block of instruction should occur after the scan loop (overrun) it will be executed then. The exit of the loop can be bound to a condition:



Exit scan loop

Condition: ORD(%(J.DocCnt)>100

The scanner will be halted immediately but all images buffered in the scanner will be still collected and processed. For that all steps inside the loop will be executed.

If exists, the first step after the scan loop will be executed then. If not the batch will be closed, i.e. in OpenJob Mode it will be suspended in DirectMode it will be finalized.

This construction can be used to exit a scan loop without using the event rules

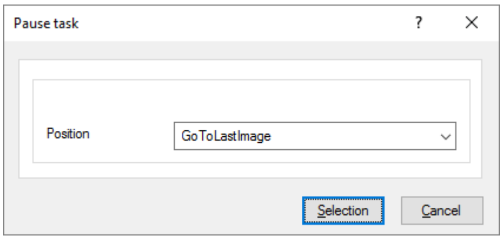
Note

This step must stand **after** the scan loop, it builds the **first step** there.

Please regard also the notes for checking a condition.

3.4.8.50 Pause task

This action stops the further processing of task steps. The program goes into interactive mode and selects a particular image.



Pause task

Position: GoToLastImage

It can, for example, be specified to go to the last image. This is useful after scanning if you want to control the last images or if you want to continue an existing batch. If on the other hand a finished stack is opened for inspection (but not to resume), it may be useful to start with the first image. It is also possible to go to the first or last marked images.

Note

This step can stand **before** or **after** the scan loop but not within, see warning.

Within the loop the execution can be paused by the corresponding step Exit scan loop or via the equivalent action in the event rules.

Hint:

If the step Load batch allows to open an exiting non-empty batch, the action **Pause task** should be placed immediately after the loading. So the operator gets a chance to inspect the batch and to close it, if he opened a wrong one. Otherwise the scanner would start working instantly an appending - wrong - images to this batch.

3.4.8.51 Task close

This task step is obsolete and **does not work** in the current version of the program.

Note

This step can stand **before** or **after** the scan loop but not within, see warning.

Within the loop the execution can be paused by the corresponding step Exit scan loop or via the equivalent action in the event rules.

3.4.8.52 Task cancel

This step stops the task immediately. No further task steps will be executed, the switches to the ready state.

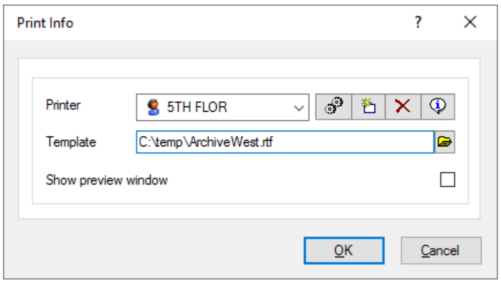
Notes

This step can stand **before** or **after** the scan loop but not within, see warning.

Within the loop the execution can be paused by the corresponding step Exit scan loop or via the equivalent action in the event rules.

3.4.8.53 Print info

Through this task step the print out of an info sheet is triggered. This sheet is archived then together with the completed scanned paper. The details for printing are set in Printer subprofile. In the task configuration is to select just the subprofile and the template file:



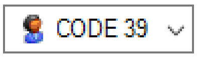
Printer: 5TH FLOR

Template: C:\temp\ArchiveWest.rtf

Show preview window □

Optional a preview can be ordered, this gives the chance to accept or to cancel the print out.

⊟ The control elements for selecting and handling subprofies have the following meanings:

	The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.
	Opens a dialog to change the currently selected subprofile or parameter set.

✳	Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.
✕	Removes the selected subprofile or parameter set, if it is not used elsewhere.
ℹ	Shows the usage and further information of the currently selected subprofile or parameter set.

Notes

This step can stand **before** or **after** the scan loop only, but not within, see warning.

The print out *after* the scanning uses current values, e.g. the number of documents. The print out *before* the scanning allows to scan the sheet too.

3.4.8.54 Process event rules on batch

This action is designated for automatic post-processing tasks. It will be inserted immediately after the instruction **Load from Directory** (batch file, special images). It causes the event rules to processed for all images of the stack again.

This allows to regard changes made on the batch at other stations.

Note

This step can stand **instead** or **after** the scan loop only, but not within, see warning.

3.4.8.55 Delete images in source path

This action is intended for automatic post-processing tasks. With this Task step, the images are deleted in the source path after reading it into the application. They are deleted "hard" and not be moved to the trash bin.

Note

This step can stand **after** the scan loop only, but not within, see warning.

It deletes the images only, **no other files**. Also the image directory is not removed, even if it had only contain pictures.

There is a safety mechanism that prevents this action is performed in important directories, for example, Windows.

If this action fails, the program halts and displays a message which will inform about the reason. In unattended working modes DpuFinalize and DpuProcess, this message will be suppressed.

3.4.8.56 Delete source directory

This action is intended for automatic post-processing tasks. With this Task step, the directory is deleted from which the images have been read. It is deleted "hard" and not be moved to the trash bin.

Note

This step can stand **after** the scan loop only, but not within, see warning.

It deletes the folder **including all** sub-folders and files, also non-images!

There is a safety mechanism that prevents this action is performed in important directories, for example, Windows.

If this action fails, the program halts and displays a message which will inform about the reason. In unattended working modes DpuFinalize and DpuProcess, this message will be suppressed.

3.4.8.57 Export Batch

With this Task step, in OpenJob mode a scanned batch is moved from the own OpenJob directory to another directory. This directory can be specified in the base profile.

Note

This Task step is available only in the Open Job mode.

In this procedure, the directory is always copied first. After a complete and successful copying the source directory will be removed.

This step can stand **after** the scan loop only, but not within, see warning. For obtaining a batch *before* the scanning loop, there is the command Import batch.

3.4.8.58 Finalize batch

With this task step more actions are executed

- the provided folder structure in the target directory is created (only in OpenJob mode)
- the scanned images are stored there permanently (only in OpenJob mode)
- it is written to the batch file (status file, protocol file).
- it is called the broker.
- the temporary files are deleted.

Note

In OpenJob mode the target folder will be created first. After a complete and successful copying the OpenJob will be removed.

This step can stand **after** the scan loop only, but not within, see warning.

During the creating of folders and files in the target directory the protocol file is written and the broker is call. The exact timing of these calls is defined in the sub-profile of the status file file or the broker.

If the program has a Gold license. this step can occur in the background. This is also possible with an installed DpuFinalize or DpuProcess.

3.4.8.59 Export and finalize batch

With this Task step, in OpenJob mode a scanned batch will be exported first. Afterward it will be finalized. In this procedure, the OpenJob is always copied first. After a complete and successful copying the finalization starts.

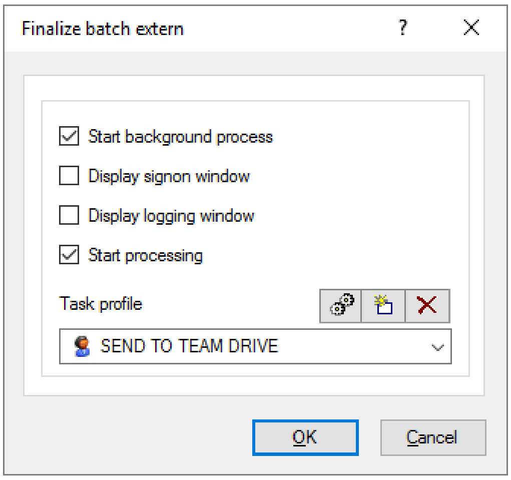
Note

This Task step is available only in the **Open Job mode**. It can be used in a DpuProcess and DpuFinalize task and can run in the background this way.

This step must stand **after** the scan loop only, but not within, see warning.

3.4.8.60 Finalize batch extern

With this Task step a batch is finalized. For this purpose, but in contrast to "normal" Finalizing starts a second process running in the background. This process can be configured in the system configuration:


Finalize batch extern

☑ Start background process

☐ Display signon window

☐ Display logging window

☑ Start processing

Task profile: SEND TO TEAM DRIVE

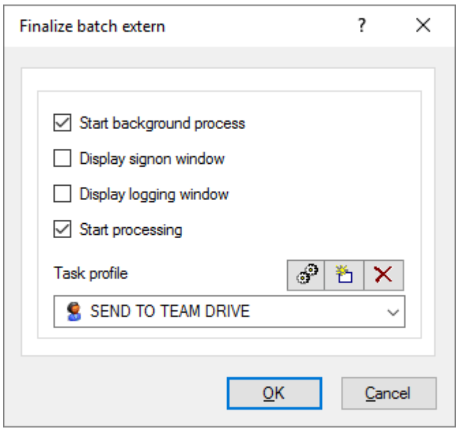
Note

This Task step is available only in the OpenJob mode. This step can stand **after** the scan loop only, but not within, see warning.

To use this step either Gold license is necessary or an installed DpuFinalize or DpuProcess.

3.4.8.61 Process batch extern

With this Task step, a background process is started, that is, the program can scan a new batch while the current batch is further processed in the background. This background process can be configured here:


Finalize batch extern

☑ Start background process

☐ Display signon window

☐ Display logging window

☑ Start processing

Task profile: SEND TO TEAM DRIVE

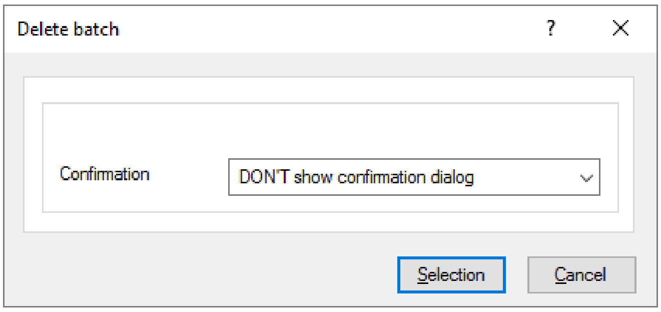
Note

This Task step is available only in the OpenJob mode. This step can stand **after** the scan loop only, but not within, see warning.

To use this step a DpuFinalize license is obligatory.

3.4.8.62 Delete batch

This action deletes the currently opened batch. Usually this instruction is restricted by a condition. A confirmation dialog can be shown optional:



Delete batch

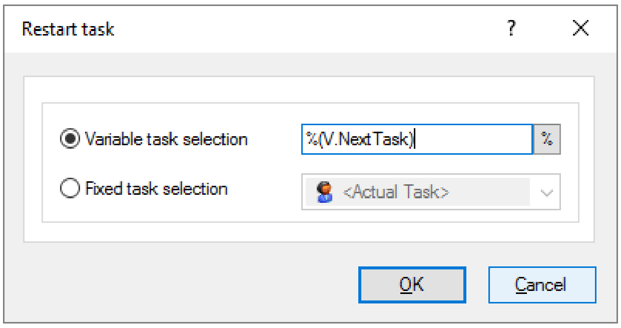
Confirmation: DON'T show confirmation dialog

Note

This Task step is available only in the OpenJob mode. This step can stand **after** the scan loop only, but not within, see warning.

3.4.8.63 Restart task

This action starts a task. It can be used either to start again the current task, or to pass the control to another task:



Restart task

● Variable task selection: %(V.NextTask)

○ Fixed task selection: <Actual Task>

With this statement, it is possible to have an automatic post-processing task that always repeat itself. This is especially of interest for tasks waiting with a polling command for a new batch.

Another common application is the **Task Chaining**, wherein a cover sheet is scanned separately and then run the task as specified on the cover page.

Note

This Task step is available only in the OpenJob mode. This step can stand **after** the scan loop only, but not within, see warning.

3.4.8.64 End program

This instruction ens the program. Together with an #IF-Instruction it can be used to end an self-repeating automatic post-processing task.

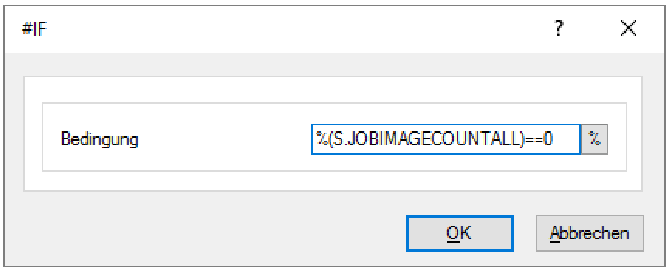
A further task may be a run-once construction which starts automatically with the command line parameters end end atomatically too with this instruction.
wieder automatisch aufhört.

Note

This step can stand **before** or **after** the scan loop only, but not within, see warning.

3.4.8.65 #IF-ELSE-END IF

Here, some of the statements are linked to a condition in the system, for example, a time-consuming process performed only on certain images or certain preassignments only when first opening the stack. When inserting this action in the Task, a dialog opens in which the condition can be entered:



Bedingung %(S.JOBIMAGECOUNTALL)==0

Here the rules for checking a condition are regarded. if the condition is true all steps between **#IF** and **#ELSE** will be executes, otherwise those between **#ELSE** and **#ENDIF**. the block between this keywords are called *clauses*.

Samples

Do the presettings only if there are no images yet:

```
...
#IF %(S.JOBIMAGECOUNTALL)==0
   Set value of a variable %(J.Stapelnumme)+1
   Set value of a variable %(J.ScanDatum)=%d.%m.%Y
#ELSE
#ENDIF
...
```

Read the adress field with the OCR only on pages with a barcode:

```
   ...
   #IF %R1!=""
      OCR on images
   #ELSE
      %(I.Adresse)=""
   #ENDIF
   ...
```

Hinweise

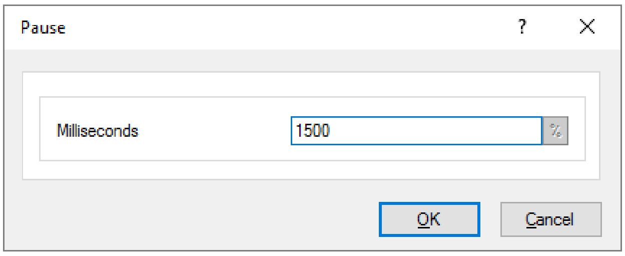
This step can be used everywhere in task, but after loading the base profile.

#IF statements can be nested up a depth of nine levels. Usually clauses are displayed in another color.

If a **Load from...** instruction is within a clause, the rest of the scan loop until **Save image** should stay in the **same** clause.

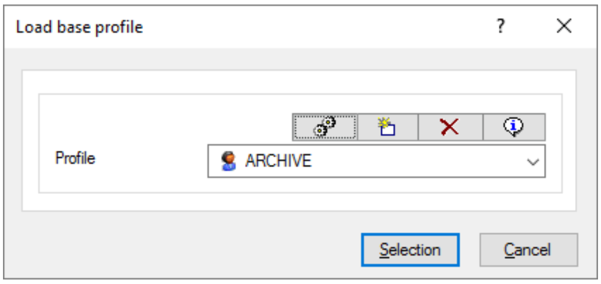
3.4.9 Task parameters

For many actions in the Task you can specify parameters directly, for example, how many milliseconds to wait for a pause. If the parameters are simple, there is for their own setting dialog boxes::



Milliseconds 1500

If the parameters are combined in one profile or subprofile, only the selection of the profile is necessary:



Profile ARCHIVE

⊟ The control elements for selecting and handling subprofies have the following meanings:

CODE 39 ⌄ The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.

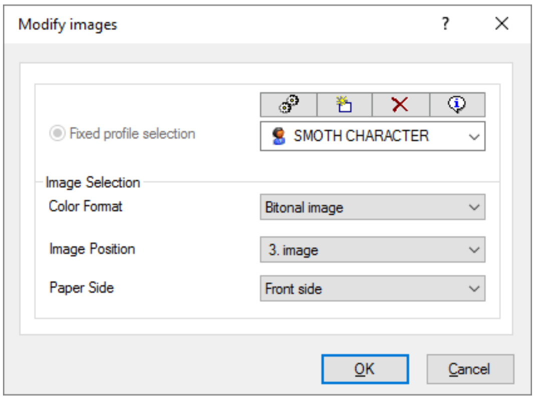
⚙ Opens a dialog to change the currently selected subprofile or parameter set.

✱ Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.

✕ Removes the selected subprofile or parameter set, if it is not used elsewhere.

ⓘ Shows the usage and further information of the currently selected subprofile or parameter set.

When a task step can be applied to a single image, a dialog is displayed which provides the possible combinations to choose from, for example, for Advanced Image Processing:



○ Fixed profile selection SMOTH CHARACTER

Image Selection

Color Format Bitonal image

Image Position 3. image

Paper Side Front side

⊟ For an image to be processed it can be distinguished:

Color Format **Bitonal image**, also black & white, 1 BPP color depth

Gray image, also Grayscale image, 8 BPP

Color image, 24 BPP

Automatic, takes the best one for this procedure

All, chooses all formats.

Image Position

The position defines the number, if there are more than one image of this color type. Those may come up by cutting, filtering or copying.
That is not the number in a file!

Depending on the action also **all images** may be chosen

Paper side

Both sides selects images for front and rear side aus

Front side chooses images captured by the front side camera of the scanner.

Rear side chooses the back side images.

To call PlugIns and load a batch there are respectively somewhat more detailed dialogs.

3.4.9.1 Load batch

This dialog is used to chose the folder which the program uses to work on batches. Besides this, batches can be created or deleted in this dialog.
Please note: Make sure that target folder and open job folder are different, especially if variables are used to determine the folder names.

This dialog is called from several points of the program. Some controls are, depending on the program state, not available in every situation.

Batches

In this list are displayed all available batches. The color in the first column indicates, whether the batch is empty, open, or locked.
🔵 A batch is empty, if it is created newly.
🟢 It is open if it is opened the second time.
🔴 It is locked, if another user has opened this batch.
🟡 It has a problem, if it is open but isn't locked actually.

New Batch Date

Adds date and time in the editfield

Add Batch

Applies a new batch name (according to the name definition) to the Batches listbox.

Purge Batch

The marked batch is removed from the list.

Update

The view oft the listbox Batches will be refreshed.

Batch name to be created

Insert a name for a new batch. Pressing Add batch, the name is applied to the listbox and the batch can be opened.
The button % allows to insert %-Codes.

Base path for batches

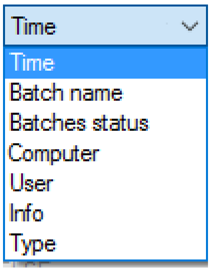
Shows the path, where the listed batches are stored

The button 📂 opens a system dialog to pick up a folder.

Batch Filter

This editfield is only available, if the dialog is called in the task configuration. The list of batches can be restricted with a filter. Example: If the content of the editfield is "%(J.JOBSTATUS)=QSI", only batches with the status "QSI" are listed

ad hoc Filter by:



*This editfield is only available, if the dialog is called in the task configuration or as result of the task step **Load batch***

In the field to the right a variable or text can be entered

Hence only those batches are shown which met the condition.

For example if "Time" with the search text "-%m-%d" would show batches from today only.

Type of batch selection

This list is only available, if the dialog is called in the task configuration

Automatic create at task start

The name of the batch will be created at task start according the settings in the edit field batch name, the user cannot select a batch.

If the batch name is **fix**, each run of the task will use the same batch. If the batch is finalized the next run will create it again.

If the name is **variable** (because of the used %-Codes) at each task start a new batch is created. An old one (closed but not finalized) cannot be opened again with this task. So be aware to finalize or export the batch after work!

Select or create at task start

A dialog will be shown at task start. The batch can be selected from the list of available batches or can be created new.

Select now

Selects the marked batch name from the list of the existing ones above.

Options

This button is only available, if the dialog is called in the base profile configuration

Opens a dialog with additional options:

Hide the "Add Batch" button, useful for post processing tasks like QA or Index
Hide the "New Batch Date", useful if batches with well defined names are obligatory
Hide the "Batch Name" inputbox, useful to avoid wrong batch names

OK

The dialog box is closed. All settings are saved.

Cancel

The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also
Data Source-Dialogs
Task definition
Help overview.

3.4.9.2 Call a PlugIn

In this dialog can be specified how a PlugIn should be invoked

PlugIn Select a PlugIn from the list of loaded PlugIns.

Subprofile Selection Select an existing Subprofile for the previously selected PlugIn.
The button over the selection list works this way:

⚙ Opens a dialog to change the currently selected subprofile or parameter set.

✱ Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.

✕ Removes the selected subprofile or parameter set, if it is not used elsewhere.

ⓘ Shows the usage and further information of the currently selected subprofile or parameter set.

Paper side You can select front side images, back side images or both images sides for image processing.

Select Input Images Specify here which images to be sent to the PlugIn.

Depending on the type of plug-in color, gray and bitonal images can be processed. If there are multiple images of a color depth, for example, by copying or cutting, you can determine what is to be taken of them.

Select Output Images Here you can specify what to do with the images that produced the PlugIn.

In the simplest case, the incoming image is replaced, for example, at a gamma correction.

If additional images are formed, for example, at Split Image PlugIn, can here be determined whether they should be appended.

Please note that this option is only activated if ...
- Output formats are defined in the PlugIn
- The PlugIn is supplied with the matching input formats

Possible output may, for example this

In the process mode:
- Appendices - An additional image is appended to the image group
- Replace (1-10) - The returned image replaces the submitted image.

In interactive mode:
- Appendices - An additional image is appended to the image group and displayed in the tree

- Replace Selection - The returned image replaces the selected image.

This dialog is called when pressing settings in the PlugIn dialog.

3.5 Base Profiles

In Officemode, the program works with a given base configuration, which can be adjusted in individual parts, for example, the type of barcode search or the file storage. For this purpose, the most commonly used features can be changed on the Ribbon tab settings.

In Expertmode, there is a more comprehensive impact those settings. These parameters are summarized in the so-called Base Profile and made available in a setup dialog.

For each workflow such a basic configuration must be specified, in other words: Each task must initially load a Base Profile.

Together with the program a standard task "DPS" is delivered, which in turn contains the standard base profile "DPS".

In the properties dialog of the task action "Load base profile" base profiles can be created as a copy of the standard base profile or other already changed profiles. In this dialog base profiles can also be modified or removed.

Some parts of the base profile are simple parameters that are **fixedly connected** to the base profile and stored therein.

Output filename	Names of the image files
Target Path	destination folder for the images
Imageformats	File formats (TIFF, PDF, JP2,...) and Compression (G4, JPG, LZW,...)
Image counter	Counting the images and comparing with the expected value.
Filename Masks	Filtering the images read to distinguish front and rear sides.
Source Path	Place of the images to load
Measurement Unit	Switch mm and inches
Polling	Waiting for a batch of images
Batch Import	Picking up a batch from another station
Batch Export	Dispose a batch to another station
Batch Data	Batch name, station name, document type,..
Batch Path	Place for the temporary storage.
Statistic File	Common protocol of the activities at the workplace

Other parts of the base profile are combined to **subprofiles** that are stored separately and are used as in various basic profiles.

Exception Handling	Behavior at job end

Barcodes	Search for bar codes
User Dialogs	Versatile entering of other data
Screen Layout	Displaying different image types during scan and in pause mode.
Bates	Stamping a text or graphics into scanned images
Broker	Calling an import program after the scan.
Deskew	De-skewing, black border removal
Event Rules	Detecting empty pages or document breaks
Image Enhancement	Removing lines or speckles
Filter	Suppressing colored background
Info Window	Displaying counters, search results etc.
OCR	Text recognition
Patchcode	Search for special divider sheets
PlugIns	Search or enhancement with external program parts
Status File	Output of a file describing the batch
Scanner Settings	Paper size, resolution, image quality, ...

Most of the settings stored as subprofiles can be configured directly form the task setup.

3.5.1 Base Profile Settings

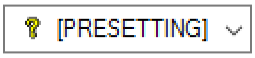
The Settings dialog for the base profile does not impose controls but merely refers to the various settings in certain fields. These settings are grouped according to their importance in various tabs.

You can set parameters for a **base profile configuration** in the following categories. Each of these categories has their own page for changing and adjusting parameters On the tabs these settings are represented by a symbol and a description

A red cross on the symbol indicates that although there is a sub-profile, but there is no corresponding step in the task active.

The button with the wheels opens the appropriate settings dialog

If the respective parameters can be combined to subprofiles, there are the buttons to manage the Sub-Profiles:



The drop-down list allows the selection of a subprofile. It is always a subprofile with the defaults provided.

Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.

Removes the selected subprofile or parameter set, if it is not used elsewhere.

Shows the usage and further information of the currently selected subprofile or parameter set.

Categories

General Dialogs

Units

Imageformats

Screen Layout

Info Window
Exception Handling

Batch Dialogs

Image Counter

Job Data

Event Rules

User Dialog

Data Source Dialogs

Source Path

Batch Path

Polling

Filename Masks

Batch Impor

Scanner Selection

Process Dialogs

PlugIns

Enhanced Modify

Bates Stamping

Deskew

Filter

Data Target Dialogs

Target Path

Filename

Batch File

Broker

Printer

Statistic File

Batch Export

Recognition Dialogs

Barcodes

Patchcode

OCR

OK Saves the changes and closes the dialog.

Cancel Closes the dialog without saving the changes made to the settings.

Help Opens the help topic for this dialog

3.5.1.1 General Settings

Units

Choice of the units, used in the program. You can choose between millimeters, tenth of millimeters and thousandth of an inch.

Imageformats

Selection of input, display and storing formats.

Screen Layout

In this dialog the settings for the representation of the scanned images in different windows can be adjusted. For the parameter settings the dialog has 3 pages.

Treeview

Process

Interactive

Info Window

Configuration dialog to set the parameter for the info window.

Exception Handling

You can set the behavior for error messages that happen during scanning or finalizing.

3.5.1.1.1 Units Settings

Select a measurement (unit) for some dialogs of the program. This measurement is used for length specification in the scanner- barcode- or patchcode dialog. The following measurements are available:

1 mm	The measurement is millimeter.
0,1 mm	The measurement is tenth of millimeter.
0,001 inch	The measurement is thousandth of an inch.
OK	The dialog box is closed. All settings are saved.
Cancel	The dialog box is closed. The settings will not be saved.
Help	Opens this help screen.

See also Help overview.

3.5.1.1.2 Image Format

Selection of input, display and storing formats.

Input formats

Shows the input format (bitonal, gray or color) of the images. Change these parameters by clicking **Scanner** or **Filter**. Some scanner perform more than one image format. The filter dialog supports the filter color to color color to gray and gray to bitonal, so it is possible to get more than one image format with one image scan. All created formats are displayed in this dialog control

Scanner

Opens the scanner setup.

Display formats

Shows which images are displayed in the scan windows during scanning. Change these setting by clicking **Display**. The display dialog opens to change the displayed image formats.

Display

Open the display setup.

Temporary formats

This dialog control is only available in OpenJobMode. Select the temporary storing format of the images. In OpenJobMode images are stored temporary until the batch is finalized. It is possible to store color or gray images temporary uncompressed, because some modifications will probably be performed. Uncompressed images have no quality loss by loading and saving. With finalization, probably the images are stored in compressed format (**Output formats**)

Output formats

This section lists the storage options for the images. Depending on the color type are different file formats available. The images in these files can be compressed with different methods in order to save space.

For **one color type** only **one storage** format can be used. Once a format is selected for a colort ype the other formats are no longer available. It follows that when changing the format the old one has to be de-selected prior choosing the new one.

The following formats can be selected:

For bitonal images: TIFF, PDF, PDF/A, PDF/A-2, JPEG2000, Raw data and OCR PDF/A

For gray images: TIFF, PDF, PDF/A, PDF/A-2, JPG, JPEG2000, Raw data and OCR PDF/A

For color images: TIFF, PDF, PDF/A, PDF/A-2, JPG, JPEG2000, Raw data and OCR PDF/A

Attention: the JPEG/JFIF file format can not be selected for multi page files. Multi page files must be multi-tiff or multi-pdf but not multi-jfif.

Format Options

If you have chosen a storage format that uses a configurable compression, you can use a sub-profile that the parameters for compression contains. There are the usual buttons for managing subprofiles available:

⚙	Opens the configuration dialog for the subprofile selected.
✳	Creates a new subprofile as copy of the selected one.
✕	Deletes the subprofile if it is no longer in use.

These options are enabled, if a format is selected that uses JPEG, JPG(GDI+), JPEG2000 or LZW compression. The selected sub profile contains the compression parameters.

Tiff Header

This button is only active, if a TIFF output format is used. You can define own tags to write it in the created TIFF files. A tag is a memory block in a tiff file with information.

Pressing this button a dialog opens to define own tags. The tag ID should get a value, which is not used for standard tags. Please read appropriate literature for numbers of standard tags. You can use values from 0x8000, which are not used by standard tags. If you have defined a number, used by a standard tag, the program ignores this value. This means your TIFF file will not be unreadable, but your defined information is not written. In the combobox Tag Type you can select BYTE, ASCII or UNDEFINED. Write your information in the edit field tag value.

Use output formats as temporary formats for batches

Checking this box the settings for **Temporary formats** are the same as the settings for **Output formats** in the OpenJobMode. The temporary control is disabled.

OK

The dialog box is closed. All settings are saved.

Cancel

The dialog box is closed. The settings will not be saved.

Help

Opens this help screen.

See also Process-Dialogs
Help overview.

3.5.1.1.3 Screen Layout

The visible area of the program, which is not occupied by the the ribbon, the toolbars, or title and status bars, is the working area. It is used to display the incoming images during scanning and in the pause the structure of the data.

It can be divided arbitrarily here. The setup dialog has three property pages to summarize the parameters

Treeview

Type and number of images to display, subtitles, preview type and size

Process	Distribution of the working area while scanning or automatic processing .
Interactive	Distribution of the working area after scanning in the pause

The buttons have the common meaning:

OK	The dialog box is closed. All settings are saved.
Cancel	The dialog box is closed. The settings will not be saved.
Help	Opens this help screen.

See also Help overview
General Dialogs

3.5.1.1.3.1 Treeview

On this page, parameter settings for the treeview, the thumbnails and the image window are set.

Treeview

Visible images at start Determines, which images are visible at job start. You can display blue, green, red or not marked images after change to pause mode or fade them out.
White marked images are invisible at start always. They can be shown by a corresponding toolbar command.

Icon size Choose between Large and Small icons.

Automatic hidden images If selected, images can be hidden immediately after manual marking, if hide the option corresponding tagged images is active.
If the image is hidden in this way, so there is an automatic selection switch to the nearest visible image.

Allow Drag&Drop Activates or deactivates drag and drop function of images in the tree. If enabled, the images can be moved to other positions in the tree.

Allow Multiselection Activates or deactivates the selection of mulitple images in the tree or the preview window. Please press the control or shift key to select more than one image.

Apply limited area view of thumbnails With this option the view will be limited the same way like the thumbnail view. I.e. all Thumbnails or the Image Group Thumbnails. Also the setting the **Visible elements at start** are regarded.

Thumbnails

Thumbnail size The size of the thumbnail display windows can be set. Select between the following settings:
Extra small,
Small,
Normal,
Large,
Extra large

Icon spacing This setting determines the distance between the thumbnails. Determine settings for the horizontal and vertical space by choosing values between 10 and 90 points

Visible range Possibility to choose whether all images, images of a directory or images in one file (in case of multipage files) are displayed in the preview windows.

Show filenames Offers the possibility to switch on/off the filename of an image under the preview window.

Show index Offers the possibility to switch on/off the index of an image (using multi tiffs) under the preview window.

Visible Elements at start For better orientation within the displayed images the view can be restricted. It is possible to show the images of one **file** or one **folder** only.

Window

Show title bar Checking this checkbox, every image window gets a title bar which shows the filename and the number of the image.

Statusbar visible Checking this option, a statusbar is displayed at the bottom of each image window.

If enabled, also the font size can be set in the drop-down box below. Choose between small, normal, medium and large.

Scaling The scaling determines the size of an image in the image window.

Frame This slider sets the size of spaces between the image windows.

Fit to window Fit to window calculates the zoom factor of an image from the size of the image window.

Fixed Fixed requires a setting in Fixed Scaling to choose the scaling factor for an image in percent.

Fixed scaling Sets the scaling factor of the scanned images in percent. This option is only enabled, if Scaling is set to fixed.

Framesize With this slider the distance in between the elements on the working area can be adjusted. Also their distance to the toolbars is set.

See also Help overview
General Dialogs
Screen Layout Overviewl

3.5.1.1.3.2 Process

On the page process the segmentation of the screen is adjusted. With the two Separate button and with Remove Window you can define the number of windows on the screen. Adjust every window by clicking it. Select the properties and size on the left side

Show window

horizontal Determines how many image windows will be shown side by side.

vertical Determines how many image windows will be shown one below the other.

Bitonal Images
Gray Images
Color Images

Used to suppress the display oft color formats.
For example when scanning in multistream mode (bitonal, gray and color at the same time) the display of gray images and color images can be suppressed and only the bitonal images are shown. This allows an easier quality check during the scan process.

Process- and interactive layout are equal

By activating this option this settings are the same for the page Interactive.

Window Properties

Document Explorer — Shows the structure of the documents.

Tree — Tree view of all images

Thumbnail — Small windows to display the images of the batch

Image Group Thumbnails — Small windows to display the images of a defined group

Edit window — Bigger windows for the images and image post processing.

Info window — Can be used to display several information.

Tool window — Displays tools to rise the image quality

Scanner Parameter — Window to change values for front and backside rotation, paper format, resolution, single side or double side scan, feeder choice and roller cleaning. Please note that some settings are not supported by all scanners.

Warn window — To display some warnings.

Batch window — To display the batchfile (protcol file) in a pan

Empty window — To reserve place for other purposes

Other windows — PlugIns can offer here their own windows, e.g. for indexing.

Exact Size

Exact size of the marked window.

Optimal Size

Optimal size of the marked window.

Minimum Size

Minimum size of the marked window.

Input Focus in Initialization

The marked window gets the input focus during process mode if this option is enabled.

Relative Separation

Is enabled, if a splitter is clicked. Insert a percent value for the active splitter. This belongs to the screen area, which is not assigned yet. For example if you have three windows and two spitters and the first splitter has a relative separation of 50 %, then the first window has 50 % of the screen area. If the second splitter has 50 % as well, then the other two windows have 25 % of the screen area each (this is 50 % of the remaining space). Relative separations keep its percent value, even if the main window changed its size

Fix Separation

Is enabled, if a splitter is clicked. The appropriate splitter cannot change its size, it stays fix as it is initialized (either by relative separation or by a fix size).

Layout Shows the segmantation of the windows.

Separate The marked window can be splitted horizontal and vertical.

Remove window The marked window is removed.

See also Help overview
General Dialogs
Screen Layout Overviewl

3.5.1.1.3.3 Interactive

The page Interactive represents the segmantation of the screen in pause mode. The dialog items are the same as in the page Process and are not described here.

See also Help overview
General Dialogs
Screen Layout Overview

3.5.1.1.4 Info Window

Use this property page to set which information shall display in the info window and in the info / status line at the bottom of the window while the Task is processed.
The information to display can freely be defined and serves the user for his survey purposes.
You may chose any text strings, but system variables to display. These may also be variables that that receive their value only during the scan process, like for example the actual filename or the number of scanned images per minute. Click the **%** button to open the selection dialog for the System Variables .

By typing in the edit box in the **Information in the info window** the definition for a line is added or modified.

In order to define the lines to display in the info window in the desired form, the following buttons are used:

Replace ⇩ Replaced the selected line with the contents of the edit box.

Add ⇩ Adds the contents of the edit box to the list, behind the actually selected line

Edit ⇧ Displays the actually selected line in the edit box. After editing, the modified line can be returned to the list by a click to the (Replace?) button.

☒ Deletes all lines from the list.

✗ Deletes the selected line from the list.

↑ Moves the selected line upwards in the list by one position. The line can also be moved with your left mouse button pressed down.

↓ Moves the selected line downwards in the list by one position. The line can also be moved with your left mouse button pressed down.

Also the **Status line** may contain any text string and variables. The set status line is displayed during the scan process with constantly refreshed values.

Displaying the Info Window

The information as set for this base profile can be displayed during the scan process either constantly or only on demand.

For a constant display, the related Action must be added in the Task.
For an on-demand display, just click the Info button ⓘ.
The next click to the Info button hides the info window again.

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also General Dialogs
Help overview
Print Info Page.

3.5.1.1.5 Exception Handling

Geben Sie hier den Text ein.You can set the behavior for error messages that happen during scanning or finalizing.
The parameters set in this dialog are saved in common under one name.

Error Messages during Scanning Here, you can determine the general behavior for all error messages of the scan process.

MessageBox and output to logfile
The message is displayed, like no paper is left in the feeder, for example. Here, all buttons are displayed that were defined below the dropdown list.
The messages will remain on the screen until the user clicks on one of these buttons.

MessageBox closes by timer and output to logfile
A message is displayed, like no paper is left in the feeder, for example. Here, all buttons are displayed that were defined below the dropdown list.
The message remains on the screen for several seconds. If no user action is done during this period, the action is automatically executed that is assigned to the default button.

Output to logfile only
No message is displayed on the screen. The action that is assigned to the default button is immediately executed.

Buttons A message / error message usually contains three buttons that allow different reactions to the error message. The following actions are available:

Retry
Pause Job and
Close Job

Button 1 (Default) This button is displayed at the left side. It is handled as default.
This means that this button cannnot be switched to inactive.

In case of automatic closing of the message window (options two and three in the dropdown list above) the action defined here will be executed.

In case that the message window is closed without using the mouse, simply by the Return key, the action defined here will be executed.

Button 2 Also to this button, you can assign an action, or it may be set as inactive. In the message window, the button then is not visible and not assigned to an action.

Button 3 Also to this button, you can assign an action, or it may be set as inactive. In the message window, the button then is not visible and not assigned to an action.

Reset This button resets all settings in this areas to their default values.

A treatment of these error messages while finalization is foreseen for the upcoming versions.

These settings can achieve that eventually unwanted options in the message window are no longer displayed. It can be used to reduce the hazards of false operation.

Moreover, the configuration can be made so that less manual operation must be done, and that the grade of automatization is increased.

See also General Dialogs
Help overview.

3.5.1.2 Data Source Settings

Path

To define with which directory the program should start, if files are loaded from harddisk. This dialog is only available in DirectMode.

Batch Path Configuration

This dialog serves to the administration of the batches and the data source of the batch.

Polling

In this dialog the settings for the polling file can be adjusted. In Polling Mode the program can make image postprocessing without intervention of the user. See Polling for general information about the polling process.

The Polling configuration dialog is subdivided in two pages:

Polling file Image files

Filename masks

This dialog allows the declaration of masks. If, instead of scanning, the images are loaded from the file system the masks can be used for restricted loading of image files. Additional an expression for the backside recognition can be entered. If this expression matches, the image will be treated as a backside.

Batch Import

A default path can be defined in this dialog to import batches. This dialog is only available in OpenJobMode.

Scanner Configuration

In the Scanner Configuration dialog you can set specific scanner parameter to adjust the program with the appropriate scanner. On different property pagess the settings for the specific scanner model and the scanner options can be adjusted.

3.5.1.2.1 Source Path

Here can be stated, with which path the program should start the search for files to open. This is used if the task steps Load from Directory or Load particular files are configured to ask the user.

Start with last selected path	Here it starts with the path which was used at the last call of **Load from...** .
Start with last created path	The search starts with the last used target directory
Start with fixed path	The search starts always in this directory.

See also Data Source Dialogs
Help overview

3.5.1.2.2 Batch Path

This dialog is used to chose the folder which the program uses to work on batches. Besides this, batches can be created or deleted in this dialog.
Please note: Make sure that target folder and open job folder are different, especially if variables are used to determine the folder names.

This dialog is called from several points of the program. Some controls are, depending on the program state, not available in every situation.

Batches	In this list are displayed all available batches. The color in the first column indicates, whether the batch is empty, open, or locked. 🔵 A batch is empty, if it is created newly. 🟢 It is open if it is opened the second time. 🔴 It is locked, if another user has opened this batch. 🟡 It has a problem, if it is open but isn't locked actually.

New Batch Date

Adds date and time in the editfield

Add Batch

Applies a new batch name (according to the name definition) to the Batches listbox.

Purge Batch

The marked batch is removed from the list.

Update

The view oft the listbox Batches will be refreshed.

Batch name to be created

Insert a name for a new batch. Pressing Add batch, the name is applied to the listbox and the batch can be opened.

The button % allows to insert %-Codes.

Base path for batches

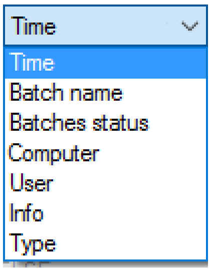
Shows the path, where the listed batches are stored

The button 📂 opens a system dialog to pick up a folder.

Batch Filter

This editfield is only available, if the dialog is called in the task configuration. The list of batches can be restricted with a filter. Example: If the content of the editfield is "%(J.JOBSTATUS)=QSI", only batches with the status "QSI" are listed

ad hoc Filter by:



*This editfield is only available, if the dialog is called in the task configuration or as result of the task step **Load batch***

In the field to the right a variable or text can be entered

Hence only those batches are shown which met the condition.

For example if "Time" with the search text "-%m-%d" would show batches from today only.

Type of batch selection

This list is only available, if the dialog is called in the task configuration

Automatic create at task start

The name of the batch will be created at task start according the settings in the edit field batch name, the user cannot select a batch.

If the batch name is **fix**, each run of the task will use the same batch. If the batch is finalized the next run will create it again.

If the name is **variable** (because of the used %-Codes) at each task start a new batch is created. An old one (closed but not finalized) cannot be opened again with this task. So be aware to finalize or export the batch after work!

Select or create at task start

A dialog will be shown at task start. The batch can be selected from the list of available batches or can be created new.

Select now

Selects the marked batch name from the list of the existing ones above.

Options	*This button is only available, if the dialog is called in the base profile configuration* Opens a dialog with additional options: Hide the "Add Batch" button, useful for post processing tasks like QA or Index Hide the "New Batch Date", useful if batches with well defined names are obligatory Hide the "Batch Name" inputbox, useful to avoid wrong batch names
OK	The dialog box is closed. All settings are saved.
Cancel	The dialog box is closed. The settings will not be saved.
Help	Opens this help screen.

See also

Data Source-Dialogs
Task definition
Help overview.

3.5.1.2.3 Polling

The program allows to run tasks without a user intervention. This is useful to proceed time consuming steps like text recognition or database querying not at scan time.

Configuration:

The parameters for the polling can be set on the following pages:

Polling file

Image files

Process:

The program watches a particular directory and start executing the other task steps when a particular file, the "Polling file", is stored in this directory. Thereafter, the processed image files will be **deleted**, and the polling file **copied** to another directory. By the Task step Restart Task a repeating loop can be established.

It must be determined, where the image data can be found and where the modified files are to be written down. The Polling file describes in a special form, where the image files to process are located.

All other **non-image** files that are **copied 1:1** from the source directory to the destination directory. So through the modification of the image files through the polling process, no other data will be lost, who belongs to the stack (for example, results of barcode search etc).

Indicators:

Polling tasks are automatic post processing tasks with this properties:

- **Endless Loop** The task should be defined as an endless loop.
- **Trigger** The task (each loop run) is triggered by an event that notifies the readiness of image data for processing

- **Delete Source** All source images are deleted after ALL images have been processed successfully.

  No source image is deleted if any failure occurs.

- **Rename Polling File** The Polling File is renamed at start of Task. This way several workstations may work on the same source directory performing parallel tasks..

- **Delete**

  Polling File The renamed Polling File is deleted after all images have been processed successfully.

- **On Errors** It is possible that a task crashes for some reason, or is interrupted or terminated.

  In such cases, the polling file in the source directory will remain, however, with a new name.

- **No Questions** Since this method does not require users, all actions should be avoided, which could trigger an inquiry to the operator!

The major difference from other Tasks is

Example

Simple Task

Load base profile

Poll For File

Modify Images

Save Images to Disk

Restart Task

If the filenames should be kept (not create new one) the task may be changed like this:

Load base profile

Poll for File

Set pathname

Keep input path and filename

Save image to disk

Restart Task

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also

Data Source Dialogs
Help overview
Polling file
Image files

3.5.1.2.3.1 Polling File

In this page the file name, the source and the targetpath of the polling files are defined.

Source file Defines the filename of the Polling File. Placeholders like *.DIF may be used to define the filename.

Source path Defines the general path for the Polling File. This means, from this path all subdirectories are searched, so it can be found several Polling Files, which are processed one after another.

Target file Set a * to store the Polling File with the same name as the Source file after processing successfully.

An empty field deletes the Polling File at the end of the process.

Every other input is treated as a new name for the Polling File.

The Polling File is deleted from the Source Path at the end of the process in any case. This definition is used for moving the file to another directory or just for deleting.

Target path Defines the path of the Polling File at the end of the process. An empty field means, the directory of the image path is taken for the Polling File, but it is also possible to define a separate directory.

Polling file is in image file path Selecting this option, all image files are searched in the same path as the Polling File.

Polling file describes image file path Selecting this option, the Polling File has to define the place where source image files are stored with the line

SourcePath=pathname

The pathname may not be enclosed by quotation marks. Trailing spaces are to eliminate.

Polling file has no image filenames Is always selected in this release: No image filenames are defined in the Polling File.

Polling file includes image filenames The option is disabled for this release.

See also Data Source Dialogs
Help overview.
Polling Overview.
Image files

3.5.1.2.3.2 Image files

Target path	Shows the pathname defines in the path dialog. This setting can only be changed in the path dialog.
Event rules	Opens the Event rules dialog box.
Path	Opens the path dialog box.
Add source image path to target path	Activating this checkbox, subdirectories of the found image files are appended to the destination path of the image files. If the image files are in the same directory as the polling file, it is the subdirectory, which is defined on the page pollingfile in source path. Otherwise the directory is described in the polling file. A drive letter in the source path is deleted before appending.
Delete source image path	Activating this checkbox the source image path will be deleted when the batch has been finished successfully.

See also Data Source Dialogs
Help overview.
Polling Overview.
Polling file

3.5.1.2.4 Filename Mask

This dialog allows the declaratin of filename masks. The entered masks will restrict the choice of the filenames when using the task steps **Load from directory** and **Load particular files**.

The List **File masks**
This List contains two columns. Into the column **Description** a shortname will be entered. This name describes the expression in the column **Mask**. Every line oft this list will be available in the combo-box **File type** of the file-open dialog. According to the chosen mask, the dialog shows a filtered file-list.
In the process there is to choose exactly one of the declared masks.

For the declaration process the characters * and ? can be used.
? is a placeholder for exact one character.
***** is a placeholder for zero or any number of characters.
Example:

The mask `?0?.TIF` will find the files	`001.TIF`
	`002.TIF`
	`100.TIF`
	`201.TIF`
not found files are	`001.JPG`
	`011.TIF`
	`ABC.TIF`
The mask `PA*.TIF` will find the files	`PA.TIF`
	`PAABC.TIF`

	PA123.TIF
	PA1234567.TIF
not found files are	PA.TIF
	PA.TIF
	PA.TIF

The editfield **Mask rearside recognition**
will contain a declaration, which classifies the matching files as backsides. The syntax is the same as that for the filename masks.

When using the backside recognition, please note that both filenames have to have the same length. There only may differ the backside value.

Example:

Valid:	F000001.TIF R000001.TIF F000002.TIF R000002.TIF	Mask:	R*.TIF
Valid:	000003F.TIF 000003R.TIF 000004F.TIF 000004R.TIF	Mask:	??????R.TIF
Not valid:	000001FRONT 000001REAR	.	

The strings "FRONT" and "REAR" do not have the same length. The program is not able to determine the matching frontside page for the given backside page.

See also Datasource-Dialogs
Help overview

3.5.1.2.5 Batch Import

Not finalized but exported batches can be reloaded for indexing. The Task must define **Import batch**. In this dialog you can define where the program will search for batches. and where the modified data is stored.

Import source	Directory of the batches. The program will search in this directory for exported batches.
Import target	Defines where an imported and indexed batch is stored.
	It is also possible to select a path by mouse click.
OK	Closes the dialog box. All settings will be saved.
Cancel	Closes the dialog box. The settings will not be saved.

Help Opens this help screen.

See also Data Source Dialogs
Help overview

3.5.1.2.6 Scanner Selection

Clicking the Scanner button in the basic profile or on the ribbon, opens the dialog for selecting the scanner. This dialog will also pop up when you start the Program for the very firs time.

Two scanner configurations are available:

First Scanner
Mostly this is connected directly to the PC scanner, for example, a sheet-fed scanner. Depending on the application, size and throughput one talks of a *production* or *departmental* or *office* scanner.If only one scanner is connected to the PC, this is the first scanner. This is the most commonly encountered device constellation in normal operation.

Second Scanner
A second scanner is required if there are documents in the batch that the "normal" Scanner can not scan or should not scan.. For example, large pages, valuable documents which may not be kinked or bound together by sealing string to a file. In technical or medical fields here are added also drawings, radiographs or writer flags. Another example is confidential mail that should not be opened for scanning but at least to recognize that envelope.

The Configure button each opens the setup dialog for the scanner properties.

If the scanner is not yet been turned on, so it can be found by turning the button **Search Scanner again** now.

See also Data Source Dialogs
Help overview

3.5.1.3 Data Target Settings

Target Path

Defines the output path for the scanned images.

Filename

Defines the rule, which generate the filenames for the scanned images.

Batch File

A Batch File stores information about the scan process. Following processes uses the Batch File as a import file. Therefore the Batch File can be defined by own request.

Broker

The broker is another method to transmit data from the scan process to other software components.

Printer

Dialog for the printer settings, used to print images with the program.

Statistic File

In a statistic file can be stored additional informations about the workplace.

Batch Export

Open batches can be exported. Data, which belong to an open batch are copied from a source directory to a target directory.

See also Help overview
Base profile overview
General Dialogs
Data Source Dialogs
Batch dialogs
Process Dialogs
Recognition Dialogs

3.5.1.3.1 Target Path

You can customize the directory to which the program will save the captured documents.

Warning: Please do not select the Open Job directory or the program directory as target path.

Handle Existing Directory	Specify, how the program treats existing directories.
Don't care and continue	All files of the existing directory is overwritten if a filename is identical with an already existing filename. If you are sure that no file names are assigned twice, this option will simply add image files to an existing directory.
Prompt to continue	The program warns, if a directory exists, before data is overwritten. You have the opportunity to continue or to stop the scan process.
Always delete directory	All files in an existing directory are deleted before the first scan.
Prompt to delete directory	This option warns before unwanted deleting of data. You have the opportunity to stop the scan job before it even starts.
Always cancel process	If a directory with the same name already exists, the process is always stopped. Overwriting or deleting of data is thus excluded.

Main Path

The box under **Main Path** lists the main directory to which your documents are saved. If you want to change this directory, or no directory is listed, click the **Browse** button. Select your main directory and choose **OK**. The program will list your choice in the **Main Path**:

Additional Path

For better file management, the program lets you define an additional Sub-Path together with the Main Path.

If the additional path does not exist, it will be created before the image files are saved.

Default setting for the additional path is the combination of follwing variables:

```
%Y%m%d_%H%M%S%@
```

(means: YearMonthDay_HourMinuteSecondMillisecond)

To generate additional pathname almost all valid variables can be used. Please make sure that the result of the definition must provide valid path and file name.

Problematic are variable with the results of recognition processes (for example, barcodes and OCR text), as they can not be predicted with certainty.

The indication of several different additional paths is not provided in this version!!

The content of this field can be queried during scanning with %(S.ADDPATH). This variable is recalculated each directory break and is empty between two jobs.

Resulting Path

In this field, the resulting overall path displayed. Where possible, the program variables are evaluated, barcode results are represented by BBBBB and Patchcode results by TTTTT.

Main path and additional path are concatenated, removing double slashes.

If the resulting path is invalid, a warning message is displayed.

Event rules

Polling

If by polling or by the event rules for an extension of the path done, a corresponding message is displayed, otherwise the sentence "The path is valid for the whole batch."

OK

The dialog box is closed. All settings are saved.

Cancel

The dialog box is closed. The settings will not be saved.

Help

Opens this help screen.

Example:

Assume the main path is **C:\SCAN** and the additional path **%Y%m%d_%H%M%S%**

When the scan process starts, the program inserts the current time and date values in the additional path.

This means for 25 July 2015 10h 34min 11sec you will get the path:

```
C:\SCAN\20150725_103411\
```

See also Data Target Dialogs
Help overview

3.5.1.3.2 Filenames

You can customize the file name, in which the program saves the captured images.

Starting filename for a new directory

Format of filename

This format may contain fixed text, %-Codes, and the # character as a placeholder for the counter.

Please make sure that the result of the definition must provide valid path and file name and must be of no more than **255 characters.**

The rules for forming the file names depend strongly on how it is generated. Possible are:

- Automatic numbering
- Evaluation of the program variables

No matter which option file names are generated, you should always make sure that every file gets a different file name. Use to a counter or a varying %-Codes. Otherwise, the program will **overwrite** existing files **without asking**!

As you enter the next name formed from this format is shown below as an example file name.

File extension: If the file name contains a **period**, the text after the point is used as the file extension. If no point is given, the normal ending of the file format is attached, e.g. ".PDF" for PDF or "JPG" for JPEG.

Multistream: If the images are saved in one file when scanning in multistream, simply the file name specified here is used. If they are stored in individual files, a dedicated file name must be created for each file of multi-stream.

If **only one filename** is given, it applies to all images/files. Then a label is appended automatically. This consists of a letter and a number, enclosed by underscores.

The letter defines the color type:

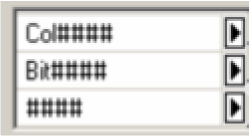
- **C Color**, color image
- **G Grayscale**, grayscale image
- **M Monochrome**, Bitonal, black-white image

The number indicates the image position, that the number of color separation, if there are several. The count starts with zero.

If, on the contrary, the files should have separate names, one can specify **multiple filenames**.

The assignment of the list items to the individual images occurs via the output sequence. First all color images will come from the scanner, then all gray images and finally all bitonal ones. Thus the file names must be specified in the same order.

Example:



Col####

Bit####

####

From Scanner come a color image and bitonal image, in addition a cleaned bitonal image for detection is generated.

The color image from the scanner is named **`Col####.JPG`**, the monochrome image from the scanner will be **`Bit####.TIF`** and the black and white image for recognition will be just **`####.TIF`**.

Counter start value

For a # -Placeholder optionally selected for Format of Filename a start value for the counter can be entered. This may be a fix value or a %-Code too.

For example, %n sets the start value to the number of scanned images since batch start. When the batch starts is this number zero. But if the directory is changed by an event rule and "Reset to starting value ..." is selected, the start value in the new directory is equal to the number of already scanned images. The counter therefore counts then consistently!

Resulting filename

The selected file name is displayed in this field, the #-Placeholders are already generated.

In the preview, the file names are displayed, which are used later. Missing data are replaced by placeholders if possible, e.g. the barcode is represented by BBBBB in the preview. Similarly, a patch code% T is indicated by PPPPP.

File options

See also examples of filenames.

Use definition for the first scan then increment # counter only

The program expects the # sign, in the above-defined file names that are replaced by a counter. There is an automatic numbering of the file names. It is important that the format has at least as many # as the largest expected number places has digits. Otherwise, there is an overflow, and the count within the Format field starts again at 0. This could produce two files with the same name, here the second file will overwrite the first one.

Use definition for every filename but don't use # counter

It will not change any # characters and the file name is generated by evaluating the program variables. This is only useful if the %-Codes differ from file to file, e.g. Counter or times. Otherwise, files could be generated twice, thereby overwrites the second file the first.

Write singlepage files for single page

Each file contains exactly one image. If per page only one image is retrieved from the scanner, so this is simply written to a file. If per page results in several images, this must be considered in the creation of the file names..

Write multipagefile for single page

Each file contains all the pictures that have been collected or generated for a page.

This option **does not mean** that both sides of a sheet are summarized. Only all the images from one camera (plus copies, cut outs etc.) are combined.

Write multipagefile for multi pages

There is only one file that contains all the resulting images. But using the event rules, on certain events, a separation can be triggered to begin a new multi-image-file

Note: Grouping multiple images is possible only if the file format TIF, PDF or PDF/A is selected. The JPG format doesn't allow to storage of multiple images in one file!

Counter Options

Reset to starting value for a new directory.

This option sets the counter at a new folder always back to the starting value defined above.

Continue with next free filenumber in the new directory.

If the highest number of a file in a directory and increased it to 1 to create the file name for the first document. This is the best way to prevent overwriting files.

When determining the relevant files for that texts before or after the counter-part are considered. If the format as "Inv ####.TIF" reads, a file namend "Del_1234.TIF" file will be ignored.

Count base 10

It used only the ten digits 0..9 of the decimal number system to form a number in the filename. With a limitation to a maximum of 3 digits, thus 1000 different names are generated.

Count base 16

To form a number the ten digits 0-9 and the letters A-F are used. Thus 16 values are available for one digit. So with a limitation to a maximum of 3 digits, 4096 different names can be generated.

OK

Saves the changes and closes the dialog.

Cancel

Closes the dialog without saving the changes made to the settings.

Help

Opens the help topic for this dialog

See also Data Target Dialogs
Help overview

3.5.1.3.2.1 Automatic numbering

If the option Use definition for the first scan then increase the # counter is selected, the # character is regarded as a placeholder for one digit. Several numbers (depending on the number of # characters) together form a number that is incremented between each scan.

At least one # character in file name (before the point) has to be specified, # in the file extension (after the period) are not converted!

For this counter, an initial value can be specified, and the behavior when there is a directory break.

If there are several separate sequences of # characters, *only the first one* is converted and incremented.

The format is not blown up, that if the counter is larger than the size, it starts again at zero.

If the # character is not to be interpreted as a number, it can be substituted by the variable %(S.ASCII.X035) or %\X035.

Also %-Codes can be used for the definition, the selection dialog can be opened by the corresponding button nearby. The evaluation of the codes is done with the **first saving operation** in the (new) folder.

Warning: If a %-Codes is used which contains a point, a file extension will be specified this way.

See also examples of filenames.

3.5.1.3.2.2 Evaluation of the program variables

If the option Use definition for every filename but don't use # counter is selected, the # sign cannot be used as counter. A warning message below the preview will inform you about that.

Instead the %-Codes will be evaluated for every filename. A problem may be those variables containing a recognition result, as they can not be predicted with certainty.

Please make sure that the result of the definition must provide valid path and file name and must be of no more than **255 characters.**

Often used Variables

Name	Synonym	Meaning
%(S.ACTUALTASK)	**%$d**	Name of the current task profile
%(S.ACTUALCLASS)	**%$A**	Name of the current base profile
%(S.SCANSJOB)	**%n**	Number of images in the batch
%(S.SCANCOUNT1)		Number of scanned images in the batch
%(S.PAPERSCANNED)		Number of scanned sheets in the batch
%(S.JOBCOUNT)	**%$N**	Number of scanned batches
%(S.BAR1)	**%R1**	Barcode 1-9
%(S.PATCH)	**%T**	Patchcode
%(S.OCR1)		Result of the first OCR field

`%(S.IMAGE.INFO)`	`%K`	Image inform
`%(S.IMAGE.WIDTH)`		Width of the image in pixel
`%(S.IMAGE.HEIGHT)`		Height of the image in pixelation
`%(S.IMAGE.SIDE)`	`%s`	Side 1 or 2
`%(S.IMAGE.COLOR)`		Color type of the image 0: unknown 1: Bitonal 2: Gray 3: Color
`%(S.IMAGE.COLORINDEX)`		Number of the image, if there are more than one of the same color type.

`%(S.TIME.DAY)`
`%(S.TIME.MONTH)`
`%(S.TIME.YEAR)`
	`%d, %m, %y`	Day, month, year 2 digits

`%(S.TIME.YEARLONG)`	`%Y`	Year 4 digits
`%(S.TIME.DAYYEAR)`	`%j`	Day of the year

`%(S.TIME.HOUR)`
`%(S.TIME.MINUTE)`
`%(S.TIME.SECOND)`
	`%H, %M, %S`	Hour, minute, second

`%(S.TIME.MILLISEC)`	`%@`	Milliseconds

See also examples of filenames.

3.5.1.3.2.3 Examples for filenames

The name and contents of the files depend on whether the front and back side are scanned simultaneously, the number of images produced during each scan and how they are combined.:

	Use definition for the first scan then increment # counter only	**Use definition for every filename but don't use # counter**
Write singlepage files for single page	Example 1	Example 2
Write multipage file for single page	Example 3	Example 4
Write multipage file for multi pages	Example 5	Example 6

Example 1

- Write singlepage files for single page.
- Use definition for the first scan then increment # counter only.

A) Simplex Singlestream, that is single sided, only one color image per side, file format JPG

Definition	Remark	Images	Files	Description

`#########`	Starting value of the counter = 0	1. Front side
		2. Front side
		3. Front side
		…
			`00000000.JPG`
			`00000001.JPG`
			`00000002.JPG`
			…
				If unchanged, 8 digits will be put out.

`Nr###.DOC`	Starting value of the counter = 1	1. Front side
		2. Front side
		3. Front side
			`Nr001.DOC`
			`Nr002.DOC`
			`Nr003.DOC`
				3 digits enumerated with a constant prefix and a fix extension

`%O_###.DOC`	Starting value of the counter = 1
	Jobname="20190720_0086
		1. Front side
		2. Front side
		3. Front side
			`20190720_0086001.DOC`
			`20190720_0086002.DOC`
			`20190720_0086003.DOC`
				3 digits enumerated with a prefix build by a %-Code. Extension is a fix text.

B) Duplex, Singlestream, that is double sided, one B/W-Image per side, Dateityp TIF

Definition	Remark	Images	Files	Description

`#########`	Starting value of the counter = 0	1. Front side
		1. Rear side
		2. Front side
		2. Rear side
		3. Front side
		…
			`00000000.TIF`
			`00000001.TIF`
			`00000002.TIF`
			`00000003.TIF`
			`00000004.TIF`
			…
				If nothing is specified, it will be enumerated with 8 digits.

C) Simplex, Multitream, 3 Images per side, two Images from the scanner in Dualstream , one color image and one black and white image, additionally a cleaned BW-Image by copy and image enhancement, color images saved as JPG, B/W Images as TIF.

Definition	Remark	Images	Files	Description

`########`	Starting value of the counter = 1	1. Front side, Color
		1. Front side, 1. B/W
		1. Front side, 2. B/W
		2. Front side, Color
		2. Front side, 1. B/W
			`000000001_C0_.JPG`
			`000000001_M0_.TIF`
			`000000001_M1_.TIF`
			`000000002_C0_.JPG`
				enumerated with 8 digits
				Automatic suffixes:
				_C0_ =
				Color image from scanner
				_M0_ =

		2. Front side, 2. B/W 3. Front side, Color 3. Front side, 1. B/W 3. Front side, 2. B/W ...	000000002_M0_.TIF 000000002_M1_.TIF 000000003_C0_.JPG 000000003_M0_.TIF 000000003_M1_.TIF	BW-Image from scanner _M1_ = BW-Image cleaned
%j[03]_%\x035####	scanned at 11. February = 42. day of the year ASCII 35 it "#" Starting value of the counter = %(S.SCANSDAY) Scans of the day with current count 644 at start	1. Front side, Color 1. Front side, 1. B/W 1. Front side, 2. B/W 2. Front side, Color 2. Front side, 1. B/W 2. Front side, 2. B/W 3. Front side, Color 3. Front side, 1. B/W 3. Front side, 2. B/W ...	042_#0645_C0_.JPG 042_#0645_M0_.TIF 042_#0645_M1_.TIF 042_#0646_C0_.JPG 042_#0646_M0_.TIF 042_#0646_M1_.TIF 042_#0647_C0_.JPG 042_#0647_M0_.TIF 042_#0647_M1_.TIF	Prefix from %-Code, Special character "#" 4 digit counter Automatic suffixes: _C0_ = Color image from scanner _M0_ = BW-Image from scanner _M1_ = BW-Image cleaned
DOC_#### **Arc_####** **Rec_###**	Starting value of the counter = %(S.SCANSDAY) Scans of the day with current value of 647 before the start	1. Front side, Color 1. Front side, 1. B/W 1. Front side, 2. B/W 2. Front side, Color 2. Front side, 1. B/W 2. Front side, 2. B/W 3. Front side, Color 3. Front side, 1. B/W	Doc_0648.JPG Arc_0648.JPG Rec_648.JPG Doc_0649.JPG Arc_0649.JPG Rec_649.JPG Doc_0650.JPG Arc_0650.JPG Rec_650.JPG	Prefix as fixed text, no suffix required Watch out! If here are **only two** names specified, no automatic suffix will be used. So the third image will just **overwrite** the second.

		3. Front side, 2. B/W ..		

D) Duplex, Multitream, 3 Images per side, color image, gray image and black and white image , color images saved as JPG, gray images as PDF and B/W Images as TIF.

Definition	Remark	Images	Files	Description

`####`	Starting value of the counter = 1	1. Front side, Color
		1. Front side, gray
		1. Front side, black and white
		1. Rear side, Color
		1. Rear side, gray
		1. Rear side, black and white
		2. Front side, Color
		2. Front side, gray
		2. Front side, black and white
		2. Rear side, Color
		2. Rear side, gray
		2. Rear side, black and white
		...
			`0001.JPG`
			`0001.PDF`
			`0001.TIF`
			`0002.JPG`
			`0002.PDF`
			`0002.TIF`
			`0003.JPG`
			`0003.PDF`
			`0003.TIF`
			`0004.JPG`
			`0004.PDF`
			`0004.TIF`
			`...`
				4 digit enumerated
				No suffix required, because there is only one image per color type and the color types get different file extensions.

Back to top

Example 2

- Write singlepage files for single page.
- Use definition for every filename but don't use # counter

A) Simplex, Singlestream: only one black and white image per side, file format TIF.

Definition	Remark	Images	Files	Description

%n	%n= Scans in the Job	1. Front side
		2. Front side
		3. Front side
		...
			`1.TIF`
			`2.TIF`
			`3.TIF`
			...
				Without leading zeros

`%(S.SCANSJOB)[08].DOC`	like above,
	%n is a synonym
		1. Front side
		2. Front side
			`00000001.DOC`
			`00000002.DOC`
				With leading zeros and fixed file

	of %(S.SCANSJOB)	3. Front side ...	00000003.DOC ...	extension
%R1	%R1 is the first barcode found	1. VS with BC=2435863 2. VS with BC=2437863 3. VS with BC=2935863 ..	2435863.TIF 2437863.TIF 2935863.TIF ...	Watch out! Barcodes may miss or exist twice.

B) Duplex, Singlestream: only one black and white image per side, file format TIF.

Definition	Remark	Images	Files	Description
%K[000000]%K[007016][04]	Image Information, first character F or R Image Information character 8 to 17 Scanner's paper counter Value before start = 234	1. Front side 1. Rear side 2. Front side 2. Rear side 3. Front side …	F0235.TIF R0235.TIF F0236.TIF R0236.TIF F0237.TIF …	4- digits Watch out! This format will be blown up, i.e. Counter 123435 results F12345.TIF
%(S.PAPERSCANNED)[06]_%s[F,B]	Count of scanned pages Side indicator, either F or B	1. Front side 1. Rear side 2. Front side 2. Rear side 3. Front side	000235_F.TIF 000235_B.TIF 000236_F.TIF 000236_B.TIF 000237_F.TIF ..	6 digits

C) Simplex, Multitream, 3 Images per side, black and white image and two cut outs made by the image processor for address and account data, file format TIF.

Definition	Remark	Images	Files	Description

%n	%n= Scans in the Job	1. Front side
		1. VS, cut out 1 (address)
		1. VS, cut out 2 (account)
		2. Front side
		2. VS, cut out 1 (address)
			1_M0_.TIF
			1_M1_.TIF
			1_M2_.TIF
			2_M0_.TIF
			2_M1_.TIF
			2_M2_.TIF
			3_M0_.TIF
				Without leading zeros
				automatic suffixes:
				_M0_ = ganze Side
				_M1_ = address
				_M2_ = account

		2. VS, cut out 2 (account) 3. Front side 3. VS, cut out 1 (address) 3. VS, cut out 2 (account) ...	3_M1_.TIF 3_M2_.TIF ...	

#%(S.SCANCOUNT2)[03]	Scan counter 2 2	1. Front side
		1. VS, cut out 1 (address)
		1. VS, cut out 2 (account)
		2. Front side
		2. VS, cut out 1 (address)
		2. VS, cut out 2 (account)
		3. Front side
		3. VS, cut out 1 (address)
		3. VS, cut out 2 (account)
		...
			001_M0_.TIF
			001_M1_.TIF
			001_M2_.TIF
			002_M0_.TIF
			002_M1_.TIF
			002_M2_.TIF
			003_M0_.TIF
			003_M1_.TIF
			003_M2_.TIF
			...
				Without leading zeros
				automatic suffixes:
				_M0_ = ganze Side
				_M1_ = address
				_M2_ = account

Doc_%n[03] **Adr_%n[03]** **Rec_%n[03]**	Three definitions	1. Front side 1. VS, cut out 1 (address) 1. VS, cut out 2 (account) 2. Front side 2. VS, cut out 1 (address) 2. VS, cut out 2 (account) 3. Front side 3. VS, cut out 1 (address) 3. VS, cut out 2 (account)	Doc_001.TIF Adr_001.TIF Bnk_001.TIF Doc_002.TIF Adr_002.TIF Bnk_002.TIF Doc_003.TIF Adr_003.TIF Bnk_003.TIF	Prefixe as fixed text, no suffix required Watch out! If here are **only two** names specified, no automatic suffix will be used. So the third image will just **overwrite** the second.

D) Duplex, Singlestream. By the OCR the invoice number is read and copied to the variable %(J.InvNo). By the event rules a new multi-image file is created if the number changes. Images inside an invoice will be enumerated and stored in %(I.PC) . Empty pages will get a red mark.

Definition	Remark	Images	Files	Description
%(J.InvNo).%(I.PC)[03]		1. Front side "123456" 1. Rear side (T&C) 2. Front side 2. Rear side (empty) 3. Front side" 3. Rear side 4. Front side "456987" 4. Rear side ...	123456.001 123456.002 123456.003 red mark, not stored 123456.005 123456.006 456987.001 456987.002 ...	The invoice number stored in an J-variable will be the same for all images until the next invoice. The image number builds the extension within an invoice. Because of the empty rear side there is a gap, 005 is missing. If the numbering shall be continous, the incrementing for %(I.PC) has to be paused when a red mark is set.

Back to top

Example 3

- Write multipage file for single page.
- Use definition for the first scan then increment # counter only.

A) Simplex, Dualream, 2 Images, Color image and black and white image, file format PDF.

Definition	Remark	Images	Files	Description
########	Starting value of the counter is Null	1. Front side Color 1. Front side black and white 2. Front side Color 2. Front side black and white 3. Front side Color 3. Front side black and white ...	00000000.PDF (1 Image) 00000000.PDF (2 Images) 00000001.PDF (1 Image) 00000001.PDF (2 Images) 00000002.PDF (1 Image)	If nothing is specified, it will be enumerated in 8 digits.

			`00000002.PDF (2 Images)` ...	

B) Duplex, Dualream, 2 images per Side, that is 4 images per sheet, Color image and black and white image, file format PDF/A.

Definition	Remark	Images	Files	Description

`%$J_####`	%$J is the number of the job, here = 33
	Starting value of the counter is 1
		1. Front side Color
		1. Front side black and white
		1. Rear side Color
		1. Rear side black and white
		2. Front side Color
		2. Front side black and white
		2. Rear side Color
		2. Rear side black and white
		...
			`33_0001.PDF (1 Image)`
			`33_0001.PDF (2 Images)`
			`33_0001.PDF (3 Images)`
			`33_0001.PDF (4 Images)`
			`33_0002.PDF (1 Image)`
			`33_0002.PDF (2 Images)`
			`33_0002.PDF (3 Images)`
			`33_0002.PDF (4 Images)`
			...
				Packing all images of a sheet (front and rear side) in one file.

Back to top

Example 4

- Write multipage file for single page.
- Use definition for every filename but don't use # counter

Simplex, Dualream, 2 Images, color image and black and white image, file format PDF.

Definition	Remark	Images	Files	Description

`%m%d_%H%M%M%@`	Month, day
	Hour, minute, second
	Milliseconds
		1. Front side Color
		1. Front side black and white
		2. Front side Color
			`0524_084533462.PDF (1)`
			`0524_084533462.PDF (2)`
				Watch out!
				The scanner works so fast that two images come within one second.

	Scan will be at 24.Mai 2018, at 08:45:33	2. Front side black and white 3. Front side Color 3. Front side black and white ...	0524_084534004.PDF (1) 0524_084534004.PDF (2) 0524_084534882.PDF (1) 0524_084534882.PDF (2) ...	Without the milliseconds, here the third file would overwrite the second.

Back to top

Example 5

- Write multipage file for single page.
- Use definition for the first scan then increment # counter only .

Duplex, Singlestream: only one black and white image per side, file format TIF. By the event rules on each front side a new multi-image file shall start

Definition	Remark	Images	Files	Description
########	Starting value of the counter is Null	1. Front side black and white	00000000.PDF (1 Image)	If nothing is specified, it will be enumerated in 8 digits.
		1. Rear side black and white	00000000.PDF (2 Images)	
		2. Front side black and white	00000001.PDF (1 Image)	
		2. Rear side black and white	00000001.PDF (2 Images)	
		3. Front side black and white	00000002.PDF (1 Image)	
		3. Rear side black and white	00000002.PDF (2 Images)	
		...	...	

Back to top

Example 6

- Write multipage file for single page.
- Use definition for every filename but don't use # counter

A) Duplex, Singlestream. black and white, simply everything to one file, file format PDF

Definition	**Remark**	**Images**	**Files**	**Description**
`%(J.JOBNAME)`	Jobname is "ORDER_15"	1. Front side black and white	ORDER_15.PDF (1 Bilde)	Simply scanning all input to one file.
		1. Rear side black and white	ORDER_15.PDF (2 Images)	
		2. Front side black and white	ORDER_15.PDF (3 Images)	
		2. Rear side black and white	ORDER_15.PDF (4 Images)	
		3. Front side black and white	ORDER_15.PDF (5 Images)	
		3. Rear side black and white	ORDER_15.PDF (6 Images)	
		…	…	

B) Duplex, Singlestream. Through OCR an proposal number is read and stored in %(J.PROP_NO). By the event rules every time the proposal number changes a new file will be created. Empty rear sides will be marked red.

Definition	**Remark**	**Images**	**Files**	**Description**
`%(J.PROP_NO).DOC`	Read number as file name, fixed text as file extension	1. Front side "123456"	`123456.DOC (1 Image)`	Attention! The proposal number doesn't have to exist twice. It has to be unique. For proposal form sheets this may be granted usually, but watch out for copies in the stack e.g.
		1. Rear side (AGB)	`123456.DOC (2 Images)`	
		2. Front side	`123456.DOC (3 Images)`	
		2. Rear side (empty)	`123456.DOC (3 Images, bleibt)`	
		3. Front side "456987"	`456987.DOC (1 Image)`	
		3. Rear side	`456987.DOC (2 Images)`	
		…	…	

Back to top

3.5.1.3.3 Batch File

Batch File can be generated to store all important information about a scan process. The **Batch File** is an ASCII file that can be easily customized to all leading document management and workflow applications. The **Batch File** is the important link from scanning to subsequent applications. Usually, the

Batch File is written into the same directory as the images, with the extension *Filename.*dif. But, remember any filename can be set in any directory.

Activate batch file

Activating this checkbox, the batch file is stored in the specified path. The image right beside indicates, whether the batch file is created or not.

Path & filename for batch file

You can enter a path and/or filename in **Path & filename for Batch File** for moving the **Batch File** at the end of the scan process and/or rename it. Click on the **% Codes** button to insert pre-defined or user-defined field codes into the path.

By inserting a filename, the **Batch File** is renamed in the same directory.

To move the file into another directory, insert a path. If the entry does not end with **"\"** the last part of the entry will be the new name for the **Batch File**. Otherwise the file will only be moved, but not ***renamed.*** Renaming the **Batch File** is required if the existence of a file with a certain name signalizes the completion of the scan job to another process**.**

One batch file per job

Depending on the file structure a scan job can have several sub directories with image files. Enabling this checkbox only one batch file is created in the superior directory. Otherwise a batch file is created in each subdirectory.

Delete batch file in image path

If the batch file is copied to another directory by defining a new path in path and filename the original batchfile can be deleted by enabling this checkbox. However in this program version the checkbox is always enabled, so the batch file is only at one position at the end of the scan job.

Editfield for line in batch file

Here you can define individual lines of the **batch file**. These lines can contain any text or any field code that receives actual values as variables at scanning time.

Click on the **% Codes** button to insert pre-defined or user-defined field codes into the **batch file**. You can also use letters, numbers, and special characters to customize the information you need for your application.

You can assign different saving options to every single line:

Lines can be created

With the first image

[01]

This line is written into the **batch file** only once during the first scan related to the current batch file. Here, you can write title information as batch name, station name, start time, or any desired start text at the beginning of the **batch file**.

If the task produces subfolders and each of the subfolders shall contain a batch file, the line will be written to each batch file.

Update after last image group
[02]

This line is initially written into the **batch file** during the first scan. It will be filled with spaces to a line length of 71 characters. The line is actualized again after the last scan. This way, the number of scanned pages, or the time of scan end can be written in a second line of the **batch file.**

Not available, if there are *more* than one batch file (one per folder)

For every image group

[03]

This line is written into the **batch file** after every scan for every scanned page. So that you can enter information that is important for every scanned image into the **batch file**.

As event rule

[04]

This line is written into the **batch file** only if an event happened for which an event mark shall be written. The related event must have been defined in the Event Rules. The action must be **Mark in Batch File**.

After the last image

[05]

This line is written into the **batch file** only after the last scan. This gives the possibility to write defined end lines into the **batch file**.

If there are several directories and thus multiple **batch files** (by event rules), so this line is entered as the last in the respective **batch file**.

With file creation

[06]

The defined line is written if a new file is created.

With file closing

[07]

The defined line is written if a file is closed (for example a multi tiff file). Please note the event order with file closing.

With path creation

[08]

The defined line is written if a new path is created.

With path closing

[09]

The defined line is written if a path is closed (for example if a new subdirectory is created).

With document creation

[16]

This event and the following one are supported from version 6.11 on. A document is created when the document level is set as an action in the event rules.

A document is automatically opened at the first image: In the log file this happens after the first path has been opened and before the first file is opened

With document closing

[17]

A document is closed when the document level is set as an action in the event rules. The value of the new document level does not matter.

With the last image, the last document is automatically closed. This happens in the log file after closing the last file and before closing the last path. Except at the beginning and at the end, these two events are always triggered in pairs and one directly after the other.

At task start

[10]

The defined line is sent at the beginning of task, this event always occurs before anyone else.

The task start occurs when the Start Task button was pressed in the toolbar or the corresponding menu item is selected

Not available, if there are *more* than one batch file. Use "With the first image [01]" instead.

At task end

The defined line is sent at the end of task.

[11]

In DirectMode, the task end is reached, when

- the last image was written to a file,
- This file has been closed,
- the final list was closed and
- was the batch is closed too.

In OpenJob the task end is reached, when a batch has been finalized

Not available, if there are *more* than one batch file. Use "After the first image [06]" instead.

With batch creation

[12]

The defined line is written after a batch is created.

Not available, if there are *more* than one batch file.

With batch finalization

[13]

The defined line is written after batch finalization and before the last broker calls.

Not available, if there are *more* than one batch file.

Several lines of each above type can be arranged. So it is possible to define, for example, five start lines, three lines per scan, two lines per event mark, or four lines at the end.

The edit field allows to specify a text to be written to the batchfile at the given event.. Fixed texts and %-Codes can be used as well.

But please note in OpenJob-Mode the %-Codes will be evaluated twice, so a %-Sign in the output has to masked twice too. For example the output "%(S.JOBCOUNT=44" has to be defined as

%%%%(S.JOBCOUNT)=%(S.JOBCOUNT)

In the first round the red and blue pairs will be reduced to a single % each. Then in the second run the resulting **%%**will be replaced to the wanted single **%**.

For DirectMode, this behavior takes no effect.

Defined batch file Lines

You can add or insert new lines, change their order, or delete them - each by using the buttons at the bottom right hand corner:

Replace will replace the highlighted line with the new line.

Add will add the defined line after the highlighted line.

Edit click on any defined line from the listbox + **Edit**, and the adjusted configurations are shown in the **New Line** window. Now you can also change the definition and save the modified line by clicking the **Replace** button.

☒ will delete all defined lines

✕	will delete the highlighted line
↑ ↓	moves the selected line up or down resp. With pressed left mouse button on a line in the listbox you can change the order of the defined lines.

OK	The dialog box is closed. All settings are saved.
Cancel	The dialog box is closed. The settings will not be saved.
Help	Opens this help screen.

See also Data Target Dialogs
Help overview.

xt ein.

3.5.1.3.4 Broker

A broker is a mediator between different systems. The usage of a broker is a method to send data to another software application during the scan process. The corresponding application for example may be archiving systems, document management systems or DBMS.

A broker can be a either a DLL with a special interface or a Visual Basic Script with a special function. In the manual you find a more detailed description of the interface and samples for both, DLL and VBS.

The broker can return information as a result of a broker call in two ways

- numerical return, presented by %(S.BrokerStatus)
- text string, presented by %(S.BrokerString)

Return value 0x9999
If the broker returns 0x9999 (decimal 39321) the program accepts data in **%(S.BokerString).** Hence it works as input and output parameter as well. The returned text may not exceed 9000 characters.

Return value greater than or equal to Zero
All other non-negative values are stored in **%(S.BrokerStatus)** and can be used to control the further workflow or output. The broker parameter is treated as input value only and it will be assumed %(S.BrokerString) wasn't changed by the broker.

Return value less than Zero
If the broker returns a negative value it will be treated as an error.

Hint: Displaying the current content of %-Codes in the status bar or the info window helps to develop a profile.

The definition of the broker service requires that the DLL or the VBS is choosen. Only then the box **Activate** can be checked. The base installation of the software contains some broker DLL's, e.g. BRO_UTILS.DLL, BRO_UTILS.VBS or ExeBroker.DLL.

The definition of the broker lines is analogous to the batch lines. Every line is associated to an event of the software application. Both fixed text and program variables can be used to create a broker line.

Activate

Enables the broker function. The image right beside indicates, whether the broker is active or not. Changes affect all classes where this configuration is used.

Path & filename for brokerfile

You can enter a path and filename for the broker DLL. Use the **Browse** button to select the DLL by mouse click.

Editfield for line in batch file

Here you can define individual lines of the broker file. These lines can contain any text or any variable.

The available types of lines are listed below.

In general: Broker events will be fired directly at scan time in DirectMode

and at finalization in OpenJob mode.

Those events especially react on OpenJobs will not be fired for DirectMode tasks.

Following line types can be created, they are attached to this broker events:

Broker Events

With the first image group

[01]

This line is sent to the broker only once during the first scan. Here, you can send title information like batch name, station name, start time, or any desired start text.

For every image group

[02]

Here you can place information which are important for one side.

This line is sent to the broker after every scan for front side and rear side separately. But there is only one call for all images of one side (filtered copies, cut-outs,...).

After the last image group

[03]

This line is sent to the broker file only after the last scan. This gives the possibility to write defined end lines.

As event rule

[04]

This line will only be sent to the Broker when an event has occurred, for that an event mark shall be sent.

The related event must have been defined in the Event Rules, the action must be "**Mark in statusfile or broker**".

This line type is not available for the OpenJob mode!

With file creation

[05]

The defined line is sent after a new file is created and before the first image is stored in the new file.

With file closing

The defined line is sent after a closing a file, for example a Multi-TIF file. A

[06]

file will be closed when opening a new file, opening a new folder or finalizing a batch.

With path creation

[07]

The defined line is sent when you create a new path. If a new path is created, a new file is generated inevitably and the new image is stored there.

With path closing

[08]

The defined line is sent if a path is closed. this occurs for example if a new subdirectory is created or the batch gets finalized.

With document creation **- not in use at the moment -**

[16]

With document closing **- not in use at the moment -**

[17]

At task start

[09]

The defined line is sent at the beginning of task, this event always occurs before any other event..

The task start occurs when the Start Task button was pressed in the toolbar or the corresponding menu item is selected .

.

At task end

[10]

The defined line is sent at end of task.

In DirectMode the Task End is reached when

- the last image was written to a file
- the last file was closed
- the last folder was closed
- the batch was closed.

In OpenJob the Task End is reached when a batch was closed, exported or finalized.

So "At Task End" can occur more then once.

With batch creation

[11]

The defined line is sent at batch creation.

After batch finalization

[12]

The defined line is sent **after** the finalization of an OpenJob.

If the command "export then finalize" is used the sequence at the end looks like this:

```
1. ...
2. [03] With the last image
3. [08]       With Path Closing
4. [17]    With Batch Closing
```

```
5. [10] At Task End
6. [12] After Batch Finalization
7. [14] After Batch Export
8. [20] At Task Final
```

Before Batch Finalization [13]

This line will be sent immediately **before** finalizing an OpenJob.This event occurs when the Finalize button was pressed. It occurs too if a corresponding command is executed in the Task or in a macro.

If the command "export then finalize" is used the sequence at the begin looks like this:

```
1. [15] Before Batch Export
1. [17]    With Batch Closing
2. [10] At Task End
3. [13] Before Batch Finalizing
4. ...
```

After Batch Export

[14]

The defined line is sent **after** the export of an OpenJob.

Before Batch Export

[15]

This line will be sent immediately **before** exporting an OpenJob.This event occurs when the Export button is pressed or a corresponding command is executed in the Task or in a macro.

With Batch Opening

[16]

The defined line is sent **after** a batch was opened.

With Batch Closing

[16]

This line is sent **after** a the last path was closed.

Error on Batch Export

[17]

The defined line will be sent to PlugIns with a Broker interface only.

It will be send if the operating system signals an error at the batch export, e.g. disk full, network drive not found, insufficient rights,...

Error at Batch Finalization

[18]

The defined line will be sent to PlugIns with a Broker interface only.

It will be send if the operating system signals an error at the batch finalization.

At Task Final

[20]

This event is fired when all commands of the task list are processed. So it occurs always as the last event.

Several lines of each above type can be arranged. So it is possible to define, for example, five start lines, three lines per scan, two lines per event rule, or four lines at the end.

Defined Broker Lines You can add or insert new lines, change their order, or delete them - each by using the buttons at the bottom right hand corner:

Replace Replaces the highlighted line with the content of the input field above.

Add Adds the defined line after the highlighted line.

Edit Click on any defined line from the listbox + **Edit**, and the content of the selected line is shown in the input field. Now you can also change the definition and save the modified line by clicking the **Replace** button.

☒ Deletes all defined lines

✕ Deletes the highlighted line

↑ Moves the highlighted line one position up. The line can also be moved using the mouse with a pressed left button.

↓ Moves the highlighted line one position down. The line can also be moved using the mouse with a pressed left button.

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also Data Target Dialogs
Help overview.

3.5.1.3.5 Printer

These settings are only for printing *info pages* and *scanned images*. Please see print Info Page for the setting when printing the Info Page.

Pinter settings This button opens a dialog to select and configurate a printer.

Activate Deactivate the printer if you want to use the profile at a station without this printer connection, e.g. for testing.. This prevents that the program looks at every start and every change of profile for the printer.

Output to file Check the **enable** button for writing the printer output to a file. Click the button beside input field to open a dialog for typing in the name or to choose a variable which contains folder and name of the file.

Auto/Manual scale page Optionally, you can automatically or manually re-scale the image for printer output.
Click on **Manual scale page** and enlarge or reduce the image by choosing the percentage from a drop-down list.

Auto/Manual center page Optionally, you can automatically or manually modify the position of the images on the print output.

	Click on **Manual center page** and select the new coordinates (x,y) for the center from the drop-down list. Unit is 0.1 mm.
Info	Shows an info window with parameters of the chosen printer.
Test	Sends the current displayed images to the printer.
OK	Closes the dialog and saves the parameters.
Cancel	Closes the dialog without saving the parameters.
Help	Opens this help page.

See also Data Target Dialogs
Help overview
Print Info Page

3.5.1.3.6 Statistic File Settings

To store a statistic file in a separate directory check the checkbox **Enable statistic file**

If activated a statistic file is stored in the directory and name, which is defined in the field **Path and filename for statistic file.**

You can use the **%-Code** button to generate the statistic file. To select a path for the statistic file press **Browse** and a dialog opens and let you select a directory.

Contents of the statistic file

Each line of a statistic file has the same structure, the contents of them are different for each batch of course. The information are written in one line and the items are separated by comma. The following data is stored in the statistic file.

Column	*Column in Excel*	*Meaning*
1	A	Type of line "Direct" means that the batch is processed in DirectMode. "Open" indicates that the batch is processed in OpenJobMode. "Renumber" means the renumbering of the file names by re-applying the event rules to all images. "Final" indicates the finalizing of the batch. "Export" means the batch export.
2	B	Station name, Content of the field **Station** name in the Dialog **Profiles | Base profile | Batch | Job Data**. If there is no entry in that field, the content of the variable %(SYS.STATIONNAME) will be used for the output.
3	C	User name, Content of the field **User** in the Dialog **Profiles | Base profile | Batch | Job Data**. If there is no entry in that field, the content of the variable %(SYS.USERNAME) will be used for the output.
4	D	Task name, as indicated (variable %(S.ACTUALTASK))
5	E	Batch name, as indicated (variable %(S.JOBNAME))

6	F	Task start date (variable %(S.TIME.DATE))
7	G	Task start time (variable %(S.TIME.TIME))
8	H	Task end date (variable %(S.TIME.DATE))
9	I	Task end time (variable %(S.TIME.TIME))
10	J	Number of scans (0 at Renumber and Final)
11	K	Number of saved images (0 at Open and Renumber)
12	L	Number of saved files (0 at Open and Renumber). This number is identical with 11 with single tiff, but different with multi tiff.
13	M	Number of processed images (identical with 10 at Direct, temporary number of available images at Renumber, Open and Final)
14	N	Number of insert operations (0 at Renumber and Final). If more than on image is inserted with the insert operation, the number is incremented by 1 because it is only one operation.
15	O	Number of modified images in edit window (0 at Renumber and Final), including rescans.
16	P	Number of deleted images (0 at Renumber, red marked image at Final). Specially the clicks are count. If more than one image is selected and and the delete button is clicked it will be count as one click.
17	Q	Number of real scan errors (0 at Renumber and Final)
18	R	Number of UDD calls (0 at Renumber and Final)
19	S	Number of manual tags (red, green, blue). (0 at Direct, Renumber and Final). Specially the clicks are count. If more than one image is selected and one toggle tag is clicked it will be count as one click.
20	T	Number of moved images and split and connect operations (0 at Direct, Renumber and Final)
21-29	U - AC	Command line parameters 1-9

See also Data Target Dialogs
Help overview.

3.5.1.3.7 Batch Export

Open batches can be moved from the own OpenJob folder to another folder. After scanning this can be triggered by the a task action or by klick on the corresponding button in the toolbar.

The export of batch is useful to make post processes of batches at indexing stations. While the scan station scans a document pile the time intensive processes are done at indexing stations. This requires batch data at the indexing stations, so the batch is not finalized at the scan station but exported. The finalization is done at the indexing station, after additional data is set.

In the batch export dialog some settings can be done for exporting batches

This dialog box is only available in OpenJobMode.

Export name

Name of the batch. You can use button % to insert %-Codes here.

If the edit field is empty the program takes the batch name for the export name, too.

Export target

Directory of the exported batch. You can use percent codes for the directory. In that case press the % button.

Seek

To define the directory by mouse click.

Don't export
but delete the batch
if no images are existingt

Doesn't move an empty batch. A batch contains images and management data. If this check box is activated and there are no images in the batch, the management data will not be written to the export.

Delete the batch
if there are only
red marked images left

Doesn't move a batch which will save no files at finalizing.

OK

Closes the dialog box. All settings will be saved.

Cancel

Closes the dialog box. The settings will not be saved.

Help

Opens this help screen.

See also Main index
Data Target Dialogs

3.5.1.4 Batch Settings

Image Counter

If you know exactly the number of scanned images for a batch, you can check these number with the image counter during the scan process.

Job Data

Define general settings for your scan job like the batch name, the user name or the scan station in the job data dialog.

Event Rules

Events appearing during the scan process like found patch- or barcodes can affect the scan process in some respects.
You an define different conditions of event rules and appropriate actions.

User defined Dialogs

Create your own Message and Input dialogs here

3.5.1.4.1 Image Counter

If you know the exact number of images to be scanned in your batch, you can have this number checked during scanning with the image counter.

Activate counter Checking this box, activate the image counter check.

Number of expected images Insert the number of expected images. After all steps of the scan loop the program checks this he number of scanned images. A dialog box will inform you if the number of expected images differs from the number of actual images.

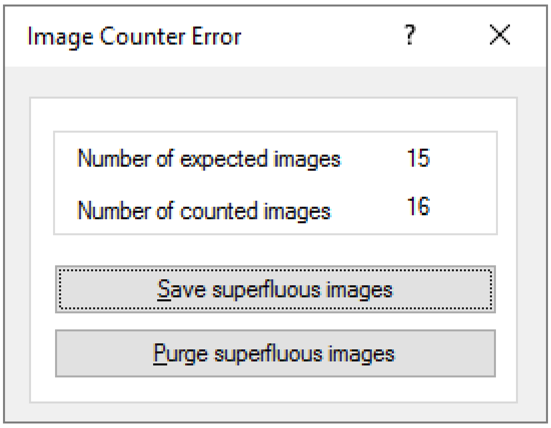


Image Counter Error

Number of expected images 15
Number of counted images 16

The batch can then be continued (Save superfluous images) or the batch can be stopped after the expected number of images (Purge superfluous images).

If you define the image counter here, it will remain constant for a certain **base profile**. If, however, the Image counter has a different value for every batch at the same **base profile** definition, it can also be defined within a **Task** as an edit field. In this case, the user is queried to enter the Image counter at the start of a batch.

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also Batch dialogs
Help overview.

3.5.1.4.2 Job Data Settings

In this dialog box some of the often used properties and counters can be set. To set other variables please use the %-Code Dialog.

batch name Type a **batch name** in the respective box. The content of this field can be requested as the program variable %(S.JOBNAME), briefly %O or

%o.

The entered value is only in DirectMode of meaning, it can be used in the path, the batchfile or in the info window for example.

In OpenJob Mode the content of this field will be always *overwritten* at the task action Load Batch with the job name given there.

Length of batch name

- This parameter is ignored in the current version -

In earlier versions the program could check the length of the **batch name** for plausibility.

Accepted were batch names, which were

less - less than the defined length

equal - equal as the defined length.

greater - greater than the defined length

Counter

Several types of counters are available, which are loaded at the start of a job, incremented during the batch and be saved at the end.

All counter can be set manually by editing the appropriate edit field.

Base profile Scan Counter

This Counter is different for every **base profile** and is incremented only if this **base profile** is assigned to the actual scan job.

The percent code of the scan counter of the class: %C or %(S.SCANSCLASS)

Scan counter for the day

This Counter is steadily incremented independent from the actual **base profile** for a whole day and is set back to Zero at the beginning of every new day. The reset occurs at the begin of the first job at this day.

The percent code of the scan counter for the day: %$N or %(S.SCANSDAY)

Global scan counter

This Counter is steadily incremented independent from the actual **base profile** and is never automatically set back.

The percent code of the global scan counter: %$C or %(S.SCANSTOTAL)

Bates counter

This Counter is steadily incremented by the bates stamping process.

The percent code of the Bates counter: %$s or %(S.BATETOTAL)

Workplace

Here, five different workplace variables are available. These variables can be set for every **base profile**, or be transferred as a parameter when the program is called.

Task name

Certain user profiles require the **Task Name** for displaying on the screen. Please do not mix up the **Task Name** with the **Task**. If you define no **Task Name** here, the name of the actual **Task** will also be used as **Task Name**. If you, however, define the name here, it may have a different value, which will not select a different **Task!**

The percent code of the task name: **%$D** or **%(S.TASKNAME)**

To set the variable via the command line parameters, use **TASKNAME.**

User name

Here you can enter a **user name**. The **user name** is required at certain user profiles for displaying on the screen.

The percent code of the user name: **%$U or %(S.USERNAME)**

To set the variable via the command line parameters, use **USERNAME.**

Station

You can enter the name of a **Station**. The **Station** name is required for certain user profiles for displaying on the screen.

The percent code of the station name: **%$S** or **%(SYS.COMPUTERNAME)**

To set the variable via the command line parameters, use **COMPUTERNAME.**

Document

You can enter a **document** name here. The **document** name is required for certain user profiles for displaying on the screen.

The percent code of the document name: **%$c** or **%(S.DOCCLASS)**

To set the variable via the command line parameters, use **DOCCLASS.**

Job counter

Here, you can enter a **Job counter** as a numeric value. The **Job counter** is read at the start of a batch, incremented by one and saved again. Thus, every batch will receive a new value. You can define the start value here. If the **Job counter** is given as a parameter at the program start, one will use this value without being incremented. The **Job counter** is required for certain user profiles for displaying on the screen.

The percent code of the job counter**: %$J** or **%(S.JOBCOUNT)**

To set the variable via the command line parameters, use **JOBCOUNT.**

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also Batch dialogs
Help overview

3.5.1.4.3 Event rules

With the work flow the program registeres certain events, for example, that a barcode has been found on one page. Here you have the ability to define, how to respond to that. You could, for example, trigger a document break on the correct barcode . To formulate this statement, there are the event rules that can be defined in a dialog.

To apply the rules the task step check event rules must be active. The rules are then performed for each front and back side for each sheet. The actions always affect all images on a side, for example, at multistream.

An event rule usually consists essentially of an **event** and an **action**. Moreover, there are still some more features which characterize an event rule:

Name After entering a name for the rule, all other properties can be set.

Context Event rules can be checked at different times.

Event These rules can react on certain events.

Condition Defines a a condition to be true to force an action.

  Option Specifies whether the condition or its opposite will be proofed.

Action If the condition matches a certain action will take place.

Image Option For the inspected images can be stated to let them stay in batch or not.

Message Some actions can be combined with a **message** to the operator.

Event rules are used to indicate breaks or to reject certain images like empty pages. Once an event is identified, it's possible that various actions to be carried out. Generally, a **new event rule** is required for **each action**.

See also context, events, conditions, actions, image options and samples and hints.
Furthermore see job dialogs and base profiles

3.5.1.4.3.1 Context

The time when an event rule has to be regarded can be given here. Possible are:

⊟ Only in a task loop

The event usually runs only when scanning or when reading files form a directory. But it will not run when the event rules are executed again in the quality control or after indexing.This restriction is useful when automatic decisions were taken during scanning, but have been withdrawn in quality assurance, does not want to be overwritten. Example: When you scan any barcode triggers a file break. Unfortunately, in the stack was a foreign barcode.In the quality assurance this accidental break is removed and the images attached to the previous file. Would the rule now run again, the separation will occur again.

⊟ Only if reprocessing rules

The event rule is executed only if the command Check event rules was given. This context is very rarely used and is e.g. is useful if the information should be considered only after the scanning is over.

⊟ Always

Select this if the event rules should be regarded at scanning and later when redoing the rules.
This is the default.

⊟ Never

Select this option if you want to suppress the test but not to delete the entire rule. This can be useful when an event rule subprofile was changed only slightly for a new task.

See also events, conditions, actions, image options and samples and hints.
Furthermore see job dialogs and base profiles.

3.5.1.4.3.2 Events

The program recognizes the following events and offers it as a source of an action:

Barcode search

This event occurs when the barcode search was successful.

Please note that the barcode search must be configured and enabled if you want to use it here as an event source.

Patchcode search

This event occurs when the patchcode search was successful.

Please note that the patchcode search must be configured and enabled if you want to use it as an event source.

Files in path counter

The event is to reach a certain number of files in the current path. The number is specified in the text box below.

Image processed

This event is simply occurs for **each image**. Usually, in this event a condition is evaluated, for example, whether the OCR has delivered a certain result..

Start of event rules

This event occurs after the first image is scanned or loaded and now the event rules are to be performed for the first time. This also applies when repeatedly executing the event rules. For the further job this event no longer occurs.

Use this event for presettings of job-wide counters or flags.

See also context, events, conditions, actions, image options and samples and hints.
Furthermore see job dialogs and base profiles

3.5.1.4.3.3 Conditions

You can use program variables, numbers and operators to define the condition that occurs (or not occurs) to produce a particular resulting action.

For that type in the condition in the **and**-field. To pick up variables, right to this field there is a button for the %-Code selection.

As in the #IF-Clause in the task here the rules for checking a condition are to be considered

Please note for the events **Barcode search** / **Patchcode search** a comparison occurs implicitly when you just enter a value. For example, at **Patchcode search** entering just PATCH2 is treated like %(S.PATCH)==PATCH2.

Match above condition

This is the default. Please change it only if it helps to achieve a significant advantage, see example in the following chapter.

Match NOT above condition

This is the reverse of the set condition. For example, a new document is to begin, if the bar code has 7 digits and starts with 347 or if it has 12 digits ending with 901, then the condition is as follows:

% R1 [006006]! = "" &&% R1 [007007] == "" &&% R1 [000002] == "347" || % R1 [011011]! = "" &&% R1 [012012] == "" &&% R1 [008011] == "901"

For all other pages a counter has to be incremented. Now, instead of to formulate the reverse condition, it suffices to copy the above condition and select the option **Match NOT above condition.**

Btw, the opposite would be:

(%R1[006006]=="" || %R1[007007]!="" || %R1[000002]!="347") && (%R1[011011]=="" || %R1[012012]!="" || %R1[008011]!="901")

See also context, events, actions, image options and samples and hints.
Furthermore see job dialogs and base profiles

3.5.1.4.3.4 Actions

You can select from a drop down list box, the resulting action to be performed if the set condition is met:

Abort even rules loop

Skips the further event rules.

- The further event rules for the current image will NOT be processed!
- The further task steps for the current image will be processed.
- The program stays in state processing and the scan process will continue.

Broker/Batchfile message

Writes a mark in the batch file or to the broker there is a corresponding action is defined there.

The text can be specified in the event message box.

Endorserstring setting

Sends the specified endorser string to the scanner. This is useful only in only in unbuffered operation of the scanner.

In buffered operation of the scanner more sheets are already passed through the scanner until the command has an effect.

Folder renaming

Use this action if you want to group the images in directories and the eponymous feature is only found on an image after the break.

You can, for example, in the incoming mail scan envelopes as well as their content and identified them by the format and start a new directory. The name or number, after which the directory is to be named, is located somewhere within the consignment. If you have found the information, you can rename the folder automatically generated now

Function call

Calls a specific function. When you select this action, an appropriate setting dialog opens.

Image options only

Only the options are executed, but it will not cause a another action for the image.

Mark image

Sets or removes a mark. For that there is a setup dialog too.

New automatic folder

Creates a new subdirectory that is appended to the main path. The first subdirectory is '00000000'. This value is each time incremented by 1 when a new subdirectory is generated.

Note: The event must be triggered at the **first stored image**, or an error message is displayed.

Please note that **directory variables** that relate to the new directory to be applied, can be used only with the **next image**.

New multipage file

Closes a multi-page file that is used for storing the images. The current image is first image of the new file.

New named folder

You can specify the name of the subdirectory that is to be generated by the event. Tap the new folder into the box path, or select the %-Code button to use variables.

Note: The event must be triggered at the **first stored image**, or an error message is displayed.

Please note that **directory variables** that relate to the new directory to be applied, can be used only with the **next image**.

Set new Documentlevel

By specifying a new document levels arises at this point a document separation. Through different levels creates a document hierarchy, for example, for a sending with attachments.

It can also always the same level to be specified if only a separation is desired.

This information can be used in the batch file or at a broker call.

Set PDF-Bookmark

Creates a new Bokokmark in a PDF file. A Dialog offers the further settings.

Task abort

Cancels the task if the condition is satisfied.

- The further event rules for the current image will **NOT** be processed.
- The further task steps for the current image will **NOT** be processed.
- The current image will NOT be saved.
- If in duplex scan this action is taken on a **front side**, the corresponding rear side will be dropped.
- The further images in the scanner buffer will be ignored, but not deleted, regardless of the option flush queuing buffer.
- The block after the scan loop will be NOT processed.
- After all the program closes the batch and goes in the state ready.

Task end

Ends the task if the condition is satisfied.

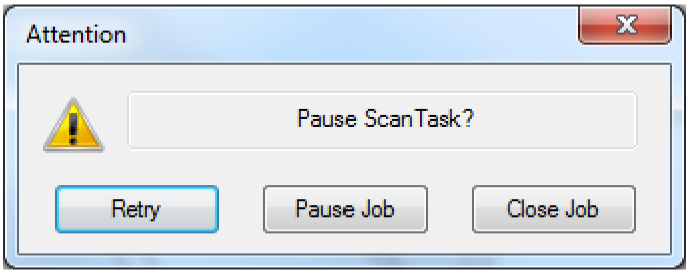
- The further event rules for the current image will **NOT** be processed!
- The further task steps for the current image will be processed.

- If the scanner option Flush queuing buffer is set, the statement block after the scan loop is processed immediately.
- If this option is NOT set, that is the default, the rest of the pictures from the scanner buffer are processed, including all event rules. Afterward the task overrun is processed.
- After all, in OpenJob mode, the program closes the batch and goes in the state ready.
- Otherwise in DirectMode, the batch will be finalized and all intermediate files removed.

Note: This action suppresses the following rules. If this is not intended, it should therefore be performed as the last action if possible.

Task pause

If the condition is satisfied the program makes a break and asks the operator how to proceed:



Attention

Pause ScanTask?

The prompting text can be set in the field Message.

Choosing **Retry**

- The further event rules for the current image will be processed!
- The further task steps for the current image will be processed.
- The program stays in state processing and the scan process will continue.

Choosing **Pause**

- The further event rules for the current image will **NOT** be processed!
- The further task steps for the current image will be processed
- If the scanner option Flush queuing buffer is set, the program will break immediately.
- If this option is NOT set, that is the default, the rest of the images from the scanner buffer are processed, including all event rules.
- Afterward the program switches to interactive mode and the state task paused.

Choosing **Close Job**

- The further event rules for the current image will NOT be processed!
- The further task steps for the current image will be processed
- If the scanner option Flush queuing buffer is set, the program will break immediately.

- If this option is NOT set the rest of the images from the scanner buffer are processed, including all event rules.
- After all the program closes the batch and goes in the state ready.

Note: This action suppresses the following rules. If this is not intended, it should therefore be performed as the last action if possible

Userdialog

This action opens a user defined dialog. UDDs can be defined in the base class.

Before the dialog opens, all variables of the UDD will be updated to the values of the current image.

Please note, that this image may be not visible on the screen, for example if there are several additional copies or cut-outs in the image group. Please note also that this image isn't accessible, for example it can't be zoomed or panned. The behavior is completely different to the one at indexing with the Index-PlugIn.

If the UDD has been closed with **OK**, the new values of the variables are copied back and the scan process continues.

If the UDD was closed with Cancel:

- The further event rules for the current image will be processed!
- The further task steps for the current image will be processed.
- If the scanner option Flush queuing buffer is set, the program will break immediately.
- If this option is NOT set the rest of the images from the scanner buffer are processed, including all event rules.
- Afterward the program switches to interactive mode and the state task paused.

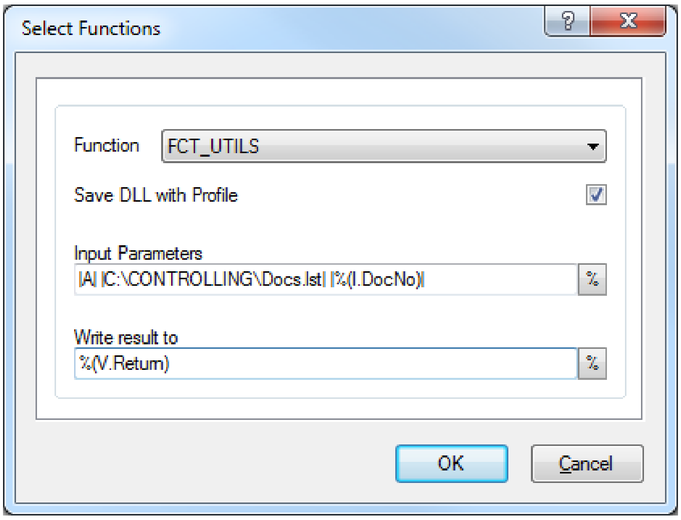
Variable setting

Use this option to change the value of a variable. So you can, for example, count certain images or assign certain properties to an image.

The assignment is always made for all (!) Images from this group of pictures.

See also context, events, conditions, image options and samples and hints.
Furthermore see job dialogs and base profiles

Taking this action a dialog opens to choose the VB-Script or the DLL



Function: FCT_UTILS

Save DLL with Profile ☑

Input Parameters: |A| |C:\CONTROLLING\Docs.lst| |%(I.DocNo)|

Write result to: %(V.Return)

Function

All available external functions are listed in this combo box. It shows all the VB scripts and all DLLs with C interface. The place for the scripts or the DLLs is the subfolder \FCT of the program folder.

Examples of such scripts or those DLLs can be found in the manual. Briefly: In the script, there must be a function named **`ExtFunction`** and having exactly one argument. This function must return a value.

Save DLL with profile

If this option is enabled, the script is included with in the sub-profile. If such a profile at another station loaded, then:If the script does not exist, it will be copied to the subdirectory FTC of the program directory.

If there is already an older version of the script, it's also copied.

If there already is a newer version, it's **not copied**.

Worked will be always with the script, which is located in this directory.

Tip: If you have made a change to the script and now want to save the profile, open this selection dialog, and turn on this hook off and back on.

Input Parameters

Here the parameters for the function can be defined if the function requires parameters.

To select the program variables, there is a corresponding button next to this field.

The character string entered there is passed to the function. Therefore, the function has to define the designated syntax. The FCT_UTILS used a text in which each parameter is enclosed in delimiters. In the above example:

```
|A| |C:\CONTROLLING\Docs.lst| |%(I.DocNo)|
```

The function takes the first argument "A" and then branched into AppendFile function (see script). There, on the file (2nd parameter) is appended a text (3rd parameter).

Note: This box can not be empty, even if the function needs no input. Please enter in such a case, either a variable or a fixed text, even if this is not evaluated.

Write result to	In this field, the return value of the function can be saved if this is to be further used. Here, too, %-Codes can be used. For this, the corresponding button is next to the box. If the script returns a **text**, so is this just assigned to the given %-Codes. This also applies to empty strings. If it returns a **negative number**, this triggers an interrupt: The remaining event rules and the further steps in the scanning loop are skipped and the program goes into the pause mode. If the script returns a **zero** or a **positive number**, operation continues. However, an assignment does not occur.

See also Actions, Job-Dialogs and Basierofiles

If the action **Mark Image** is chosen the corresponding button get active. It opens a dialog to set or remove the marks, Choose the color type and the position of the image in the image group.

Pictures that have certain characteristics, for example, are blank, contain a bar code or a keyword can be marked:

Red marks
Images with red tag will not be written when finalizing the destination directory. However, it is possible to remove previously set mark in the quality control or to set them for other images manually.

Red marks are usually set to **empty rear sides**, but it can also be labeled sheets that were not originally in the stack of paper and were only inserted to separate (Patchcodes, other divider sheets or control sheets).

Blue marks
If still images with blue marks in a stack, it can not be finalized. This is only when the corresponding image is selected and displayed, and the mark was manually removed.
Blue marks are used to draw attention to exceptions, i.e. **errors**. For example, that a barcode is missing; but should actually be on page:

Green marks
Images that are especially **important**, can be highlighted with green marks.
When a stack of documents is composed, which in turn may consist of several images, one can e.g. select the first one of each green.

White marks
White tags serve to mark images **unimportant**. They are not displayed after scanning, regardless of whether they still have other marks.
In contrast images with multiple marks only disappears when ALL markings should be hidden.

Color
Since the program can create multiple images for each page, e.g. by scanning Multistream or by copying or cutting, you can here specify the selection yet:

- **All color formats** chooses all formats
- **Bitonal iamge**, also black/white image, 1 BPP color depth
- **Gray image**, also grayscale image, 8 BPP
- **Color image**, 24 BPP

Image position
The image position is used to determine which copy or extract is to be marked.
A multiple-selection is unfortunately not possible. For example to mark the first and third image only, two event rules are necessary for this.

See also Actions, Job-Dialogs and Basierofiles

3.5.1.4.3.5 Image Options

To dispense with further event rules, you can specify if the image is to be saved or not. This can be done with the image option.

This option is the only way to remove a page automatically from the stack. Basically, a once removed image can not be restored without rescan. In OpenJob mode it's more convenient, not to remove the page, but to mark red only. On finalizing these files are not saved. But you can reverse this decision anytime by removing the mark in the pause.

Tip: You should use the default if possible. It is clearer to explicitly set the markers with the designated action.

Save Image after rule handling
This is the default, do nothing. It should not be changed without good reason.

Purge actual and backside.
When two-sided scanning, this setting is useful for separation sheets, as Patchcode sheets. This option also works when only one side is scanned. So the following front is NOT deleted.

Attention! If this option is selected, the back of the sheet is picked up by the scanner only, but no further task actions or event rules will be run on it!

Purge actual image only
This action simply removes the side on which the event is resolved. For example at single-sided scanning the divider sheets can be removed removed.

Purge backside image only
This action removes the back on the affected side. For example blank back sides of cover sheets can be removed this way.

Red mark on actual and backside
This action marks that side where the event was triggered and its rear side with a red mark. Thus, these pages are not stored when finalizing.

Red mark on actual image only.
This action marks that page where the event was triggered. Thus, these pages will afterwards not saved.

Red mark on backside image only
This action marks the back of the affected side.

See also context, events, conditions, actions and samples and hints.
Furthermore see job dialogs and base profiles

3.5.1.4.3.6 Editing event rules

Event rules enable the sorting of documents during the scanning process to store these in different subdirectories to stop the scanning process or add flags. Event rules are perfect for all requirements of mass scanning tasks. Document management and sharing of image files into workflow or OCR applications are considerably simplified.

The setup dialog on the right shows a list of event rules and left an area with the details of the currently selected rule. When opening the first rule is automatically selected and displayed.

Context

Defines the time when to regard the rule

Rule Name

This is an arbitrary text which can be freely selected.

Here you can select a "speaking" name, that describes briefly event and action, for example, "Empty page-> Red mark".

Using the options the results of the rules can be displayed the warning window.

The names of all defined event rules are listed in the **Active Rules** list box.

Event

Selection of the event which should trigger an action. This ay be a barcode, a patchcoe a counter or simply all images.

and

Define here the condition which should be checked when the event occurs. You can use texts, numbers or program variables to specify the condition.

Option

Following options in the handling of the above condition are available:

Match above condition:

The condition is met (true), if the text was found, reached the counter or apply the formulated comparisons.

Match NOT above condition:

This is the inverse of "Match above condition".

The evaluation is carried out just as there, only the result is inverted thereafter, that "True" from is "False" and vice versa.

Action

The action the program shall take.

Image options

Here you can define what shall happen with the image that triggered the rule.

Message

Here you can post a message, which is always displayed on the screen if the scan process has stopped.

The output message is also assigned to the variable **%$M, %(S.EVENTMESSAGE)** or **%(S.MESSAGE)** and can e.g. be used in the batch file at "as event rule".

In addition, this message can be used as a comment in the list of active rules.

On the right side, a list of defined rules will be displayed.

Here the following options are available:

Replace Replaces the highlighted rule by the newly defined rule.

Add Inserts the new rule after the highlighted one.

Edit Click here to view the selected event rule on the left side to edit..

☒ Remove all elements from the list

✕ Removers the selected element from the list

↑ Moves the element one position up

↓ Moves the selected element one position down.

The list contains columns on the properties of the event rules. The column **option** uses the character **==** for Match above condition and ! = for Match NOT above condition.

In addition, the first column displays a checkbox, with which you can enable the rule. Remove the check mark if you want to set rules shortly suspended during testing.

Options This button opens a dialog box to setup quickly certain options. Please change these settings only if you are familiar with their meaning or if you prompted to do so in a support case.

EventLogging

0 (default) or 1, 1 shows the result of the test f the rules in the warning window.

EventResetMark

0 or 1 (default) , 1 removes all marks before the execution of the command Redo event rules

OK Saves the changes and closes the dialog.

Cancel Closes the dialog without saving the changes made to the settings.

Help Opens the help topic for this dialog

See also context, events, conditions, actions, image options and samples and hints.
Furthermore see job dialogs and base profiles

3.5.1.4.3.7 Examples

Empty page

Event	Image processed
Condition	ORD(%k)<3500
Action	Mark image: Set red mark

Separation on Patchcode

Event	Patchcode found
Condition	
Action	Write new multipage file

Separation on Barcode (or with the first image)

Event	Image processed
Condition	%R1!="" || %n==1
Action	New automatic folder

See also context, events, conditions, actions, image options and hints.
Furthermore see job dialogs and base profiles

3.5.1.4.3.8 Hints

-
- If you have **several conditions** that characterize a situation, and also have to perform multiple actions, you should formulate the condition only once, check it and hold the result in **a flag**, for example, % (I.NewDoc)=1 . The other actions are then executed only if the flag is set.
- If you want to enter a **more complicated** issue, for example, "3. page after the patchcode but only if the barcode ends with 123 or is only 5 characters long, where the second digit is not an even number ..." you should **call a function** and perform this calculation there
- Preassignments of variables that are not retrieved in the event rules and are the same for each sheet, can just be set with the task step Set value of a variable.
- If you have **many event rules** and / or the profile is changed by **several people**, it makes sense to add a **comment** with the most recent version in the set of rules. For that use value assignment, for example, % (Comment) = "last updated 19/05/2016, A.M."

See also context, events, conditions, actions, image options and samples.
Furthermore see job dialogs and base profiles

3.5.1.5 Process Settings

PlugIns

Dialog for selection and configuration of PlugIns. PlugIns are external program parts. They are embedded to the program by an open interface. The symbol shows wether PlugIns are activated or not.

Enhanced Modify

A scanned image can be modified to make a quality improvement. You can delete lines, remove noise, move the image and much more.

Bates Stamping

The Bates stamp is an electronic stamp, merged into the image. It contains a text and an image. Of course it’s possible to use %-Codes and program counter in the text stamp.

Deskew

Configuration Dialog for the color and gray deskew.

Filter

Configuration Dialog to define different color filter. A color filter transforms an image in another color format, for example color images to gray images. The parameters for the filter process are defined in this dialog.

3.5.1.5.1 PlugIns

PlugIns are external program parts. They are embedded to the program by an open interface. In this dialog the plugins can be selected and the parameters can be defined.

PlugIns activated

Enables the plugin function. The image right beside indicates, whether the plugin is active or not.

List of used PlugIns

Shows a list of all Plugins which are used within the current base profile

Please note:
For a correct working configuration **the order of the lines may be important**. For example the PlugIns can get events and change variables. Then it will be important that the next PlugIn call gets valid variable contents.

Status

Shows that the PlugIn is loaded and a valid licence is available.

🟢 green: the PlugIn is loaded and licensed.

🔴 red: the PlugIn is loaded without having a valid license assigned to it.

🟡 yellow: PlugIns are loaded but are disabled by the overall control 'PlugIns activated

⚫ black: PlugIns could not found or loaded

PlugIn

Shows the name of the PlugIn.

Description

Shows a short description of the PlugIn.

Mode

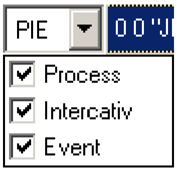
Shows which mode is supported by the PlugIn.

There are three modes:
P: process mode (while scanning)
I: interactive mode (when scanning is paused)
E: event mode (the PlugIn gets events like the broker)

If the selected PlugIn only can be used when scanning only a "P" would be shown in the column:

P

If one of the available modes shall be suppressed it can be deactivated.:



PIE 00"J

☑ Process

☑ Intercativ

☑ Event

Parameter Shows the configuration which shall be used by default, e.g. when a button is pressed in the edit window.

Add Opens a dialog box to add a new PlugIn to the list.

Setting Opens a dialog box to create, change or delete configurations.

Buttons:

L Starts a license check for the PlugIns. If a PlugIn is not licensed the key can be entered. If all licenses are ok the user will get no further messages.

i Opens a window with further information for the PlugIn and the chosen configuration.

☒ Deletes all lines from the list.

✕ Deletes the selected line from the list.

↑ Moves the selected line one position up.

↓ Moves the selected line one position down.

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

3.5.1.5.1.1 PlugIn Selection

Here PlugIns can be loaded in to the current Profile. The load PlugIns, please **check mark** them in the list and leave the dialog with **OK**. If necessary, a dialog pops up to ask for a license key.

List The list shows all available PlugIns which are located in the EXT subfolder of the application's program folder. The columns show the name, the short name (ID) and a brief description.

If, nevertheless it was installed, a PlugIn isn't listed here it couldn't be loaded. The reason may be a missing software component. For example the OCR-PlugIn can work only if the OCR-Engine is installed too.

The ID will be used at several places in the program to identify the PlugIn.

Update This button causes the program to search the folder for newly installed PlugIns.

OK The dialog box is closed. All changes are saved.

Cancel The dialog box is closed. The changes will not be saved.

Help Opens this help screen.

This dialog is called when clicking **Add** in the PlugIn dialog.

Some parts of the regular work in the program are moved to external libraries. They are free and can be used always in version 6 and higher.

- 1D Barcode by Pixtools
- 2D Barcode by Pixtools
- Automatic Orientation
- Erasing parts of the image
- Starting an executable File EXE
- Empty page detection
- MDIT Binarization
- Removing Punchholes
- Select an existing OpenJob
- 1D Barcode by Softek
- Calling a VBScript
- Using common system dialogs

Another part is created by us and uses special algorithms to better images or to control the workflow. These PlugIns are optional and liable to pay costs. Some of them are included in certain versions of the program, please refer to the actual documentation for that.

- Color detection / Marker detection
- Edge Enhancement
- Manual Indexing
- Interactive Image Enhancement \RsoD
- ODBC Database access
- Color noise reduction
- Gamma correction
- OMR Optical mark recognition
- Split Images
- Extract handwritten signatures

A third part are PlugIns that deals with 3rd party libraries, like 2D-Barcode search. Depending of the library those are liable to pay costs too.

- 2D Barcode: Axtel Datamatrix
- 2D Barcode: Axtel PDF417
- 2D Barcode: Inlite Datamatrix
- 2D Barcode: Inlite QR
- Abbyy FineReader 6
- Digital Signature by Authentidate
- Digital Signature by Digiseal 100/250/500

- Dunord Binarization
- FaceSnap Batch
- FaceSnap Interactive
- ICR Iris

See also PlugIns.

3.5.1.5.1.2 Plugin Profile Information

This dialog shows the properties of he PlugIn in form of a tabbed dialog.

General

The tab **General** shows detailed information about the PlugIn, e.g. version and its manufacturer.

Percent Code

The Percent Code tab lists the variables that can be used by the PlugIn or within the configuration.

The table lists, next to the name of the variable, the property that marks with symbols how the variables shall be processed by the system. For OMR, the following symbols are displayed:

D DLL ⟷	The program and the PlugIn exchanges this variable as input and output
D DLL →	The variable is set by the program and is sent to the PlugIn.
D DLL ←	The variable is set by the PlugIn and is handed over to the program.
↓	In process mode data will be sent as a task action or as a macro action.
←	In interactive mode the data will be send **before** the next image gets the focus.
→	In interactive mode the data will be send **after** the new image has been selected.

Information

This page offers, in form of a tree view, information about the name of the PlugIn, its manufacturer and the version.

The ID lists the windows, images and variables that are used by the PlugIns.

This dialog is called when clicking Info in the PlugIn dialog.

3.5.1.5.1.3 PlugIn Call

In this dialog can be specified how a PlugIn should be invoked

PlugIn Select a PlugIn from the list of loaded PlugIns.

Subprofile Selection
Select an existing Subprofile for the previously selected PlugIn.
The button over the selection list works this way:

⚙ Opens a dialog to change the currently selected subprofile or parameter set.

✳ Opens a dialog for a new subprofile or parameter set. At the begin the new subprofile will be a copy of the current subprofile.

✕ Removes the selected subprofile or parameter set, if it is not used elsewhere.

ⓘ Shows the usage and further information of the currently selected subprofile or parameter set.

Paper side
You can select front side images, back side images or both images sides for image processing.

Select Input Images
Specify here which images to be sent to the PlugIn.

Depending on the type of plug-in color, gray and bitonal images can be processed. If there are multiple images of a color depth, for example, by copying or cutting, you can determine what is to be taken of them.

Select Output Images
Here you can specify what to do with the images that produced the PlugIn.

In the simplest case, the incoming image is replaced, for example, at a gamma correction.

If additional images are formed, for example, at Split Image PlugIn, can here be determined whether they should be appended.

Please note that this option is only activated if ...
- Output formats are defined in the PlugIn
- The PlugIn is supplied with the matching input formats

Possible output may, for example this

In the process mode:
- Appendices - An additional image is appended to the image group
- Replace (1-10) - The returned image replaces the submitted image.

In interactive mode:
- Appendices - An additional image is appended to the image group and displayed in the tree
- Replace Selection - The returned image replaces the selected image.

This dialog is called when pressing settings in the PlugIn dialog.

3.5.1.5.2 Bates Stamping

By Bates Stamping graphics and text can are placed at certain points of the scanned document. So may be provided scanned pages with own stamps. For example, you can state when and where a paper was scanned

It can be stamped black/white, gray or color images. To enable this select the appropriate option **here**. The usual selection of the image for task step Bates Stamping is therefore limited to "All Images". The settings here on this page therefore take **precedence**.

The stamp is inserted only in the **electronic image**, not on the paper. To stamp the **paper** when scanning, the scanner must have an endorser.

- Text
- Bitmap

Test Click **Test** to test your settings for the bitmap and the text. Place the frame area to the correct position of the image, which can be load from harddisk or scanner.

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

3.5.1.5.2.1 Text

On this page the parameters for the text can be set:

Stamp bitonal images Enables the Bates text stamping for bitonal images

Stamp gray images Enables the Bates text stamping for gray images

Stamp color images Enables the Bates text stamping for color images

Stamping position The **X** and **Y** field defines the upper left corner of the text, the **size** defines the height of the text, the appropriated width is calculated.

Size Defines the height of the text. The appropriate width is calculated.

Position relative to Select a corner of the Document. All field coordinates are relative to this corner.

Rotation The text can be rotated by 0, 90, 180 and 270 degrees clock wise.

Background Sets the background color for a frame around the text. Choose between black, white or transparent.
Black or white means the background of the stamped area will be black or white.
Transparent means, the background of the stamp area remains unchanged.
Choosing transparent the characters will be printed in black color.
The setting "white text" on "transparent background" is not allowed. In that case no stamping will be done.

Border Defines the size of the frame around the text. The defined area in the window will be larger of smaller.

Text Insert a text in this field to be moved to the scanned image. You can also use **% Codes**, which are placeholders for other information.

Pleas note: In case of changes of the text color or background color, please check the corresponding setting to make sure that the settings will produce readable stamps.

See also Help overview
Process-Dialogs
Bates Overview
Bates Bitmap

3.5.1.5.2.2 Graphic

Files in Windows bitmap format can be placed on the scanned pages. Graphic image stamp can be stamped **only as a black and white** image on the scanned pages. If you select a Windows bitmap in color or gray format, this bitmap is **converted** to black and white.

Stamp bitonal images Enables the Bates stamping for bitonal images

Stamp gray images Enables the Bates stamping for gray images

Stamp color images Enables the Bates stamping for color images

Stamping position The X and Y field defines the upper left corner of the bitmap, the graphic size defines the height of the graphic, the appropriated width is calculated.

Graphic size Shows the height of the graphic. The appropriate width is calculated. To adjust the image size use the mouse and change the size of the frame on the preview window.

Position relative to Select a corner of the Document. All field coordinates are relative to this corner.

Rotation The graphic can be rotated by 0, 90, 180 and 270 degrees clock wise.

Background Sets the background color for a frame around the graphic. Choose between black, white or transparent. Black or white means the background around the stamped image will be black or white. The image area of the stamp is inverted, if the background is black. Transparent means only the black area of the stamp overwrites the image. The background of the stamp area remains unchanged

Border Defines the size of the frame around the graphic. The defined area in the window will be larger of smaller.

File Insert the path and the name of the graphic file to be moved to the scanned image. You can also press Browse to search for the graphic file.

See also Help overview
Process-Dialogs
Bates Overview
Bates Text

3.5.1.5.3 Deskew

If during the scan process the paper wasn't fed accurately aligned usually edges and corners of the scanner's background will become part of the image. With the method described here, these edges can be cut off, this is called "cropping". If the image is rotated previously so far that the paper's edges are parallel to the edges of the image, the resulting strips can be completely removed completely. This rotation is referred to as de-skew or "deskew" shortly.

The deskew parameter get defined separately for bitonal, gray and color images.

The setup is split into these steps:

- Parameter for detecting borders and calculating the destination rectangle, including manipulations to the rectangle and rectangle size checking.
- Image processing parameters based on the rectangle.
- Additional image manipulation settings.

Find Rectangle Parameter

Detection Mode

Brightness (Black&White only)

Skip black pixel (the background), find white pixel (the document).

Brightness (manually) (Gray or Color only)

Skip dark pixel (the background), find bright pixel (the document);

define the background color level by parameter Threshold.

Brightness (auto) (Gray or Color only)

Skip dark pixel (the background), find bright pixel (the document);

the background color level is decided by the rectangle processor, independently for every image.

Color Sensitive (manually) (Color only)

Skip pixel near the defined Background Color, find pixel color outside the defined Tolerance (the document).
Color Sensitive (auto) (Color only)

Same as above, but Background Color and Tolerance are decided by the rectangle processor, independently for every image

Background Color (Color only)

Default: Black

Defines the scanners background color within the 24 bit color space. For non-standard colors format: 0xrrggbb is used (with rr (red), gg (green), bb (blue) in range of [00..FF]).

Threshold

(Gray or Color only)

Range [0..255], Default: 20

Threshold for searching the borders. A border point is found, if a stream of more than **MinColorPixel** (see dialog "more...") points are brighter than Threshold value.

Tolerance

(Color only)

Range [0..255], Default: 20

Similar to threshold: while threshold gives a value based on color "Black", gives Tolerance a color space around defined Background Color.

Required Borders

Range [1..4], Default: 2

The detector needs to find [1..4] borders to accept the destination rectangle. For example, a sales slip may have only two straight edges at left and right, while the begin and the end of the slip is cracked.

more ...

The button opens the dialog "More Settings".
(These special settings should be changed by experts only.)

Modify Rectangle Size

Check, if the detected rectangle should be manipulated.

Use button "Modify settings ..." to setup manipulation parameters.

Check Rectangle Size

Check, if the rectangle size should be inspected.

Use button "Check settings ..." to setup inspection parameters.

Process Parameter

Process Selector

Default: Deskew

Cropping: The corner points of the rectangle are used to cut off the image frame. Normally, this process reduces the image size in a short processing time.

Deskew: The rectangle is cut out of the image and turned into the nearest perpendicular position. This replaces the original image.

Mode

Range [Fast / 1 Point (Default for Black&White), 1 Point, 2 Point (Default for Gray and Color), 4 Point, Bicubic (16 Point)]

When turning the image, most destination points don't have a corresponding source point. Example: The new image point at [line 9 column 15] is calculated to get from [line 47.35 column 34.86].

There are different processors to calculate the destination points. Use parameter Mode to select the processor you like:

Fast / 1 Point (Black&White only):

If skew angle is less than ±7 degrees, a special, very fast processor is used.

If the skew angle is larger, 1 Point processor is used automatically.

1 Point:

Processor just takes the nearest original point (in given example, it would be [line 47 column 35]. Advantage: fast processor; disadvantage: clean outlines may become rough.

2 Point:

The destination point is calculated as the mean value of the two nearest original points.

4 Point:

The destination point is calculated as the mean value of the four nearest original points. This processor produces smooth outlines best, but is not so fast and the resulting image may loose sharpness.

Bicubic (16 Point) (Gray or Color only): This is an enhanced method with a special border smoothing.

Fill Color

Range [Black, White, 0x...], Default: Black

Areas of the destination rectangle outside the scanned image (typically corner areas, because scan area was smaller than the skewed document), have to be created by the deskew process. These areas will be filled by the chosen color.

Aside from standard colors Black and White, experts are free to enter any color of the available color space: 0xpp on gray, 0xrrggbb on color images (with pp, rr, gg, bb in range of [00..FF]).

Note: If Windows color selection dialog is used to select a color for gray images, the corresponding gray level will be used.

Fill Border

If checked, the background color will be replaced by the given Border Color, starting from the image borders, till non-background color is detected. This process is controlled by parameters in "Fill settings ..." dialog.

Border Color

Range [Black, White, 0x..., Image Color*], Default: Black

Defines the color that replaces background color near the image borders.

Aside from standard colors Black and White, experts are free to enter any color of the available color space: 0xpp on gray, 0xrrggbb on color images (with pp, rr, gg, bb in range of [00..FF]).

If Image Color* is used, the processor steps through background color inside the image, takes a color near this point, and fills up the background area using this color. It should not be used to fill large areas!

Additional Features

Adjust Image Size

If checked, the resulting image can be adjusted in size. Enter "Adjust settings ..." dialog to setup size values.

Frame Color

Möglich [Schwarz, Weiß, 0x…, Bildfarbe], Standardwert: Schwarz

Range [Black, White, 0x…, Image Color *], Default: Black

Define the color of added image areas (frame). Aside from standard colors Black and White, experts are free to enter any color of the available color space: 0xpp on gray, 0xrrggbb on color images (with pp, rr, gg, bb in range of [00..FF]).

If Image Color * is used, the processor steps through background color inside the image, takes a color near this point, and fills up the background area using this color. It should not be used to fill large areas!

Global

Default

Set all parameter back to default values.

OK

Saves the changes and closes the dialog.

Abbrechen

Closes the dialog without saving the changes made to the settings.

Apply

This button has no meaning here.

Help

Opens the help topic for this dialog

3.5.1.5.3.1 Find Rectangle

These parameters describe how non-ideal edges are to be treated. In general the default values provide the best result. So don't change these settings unless you are familiar with the operation or be prompted by us in case of support.

Min Border Pixel

Range [0..255], Default: 5

The border detector starts Min Border Pix points away from the page border.
This is to minimize the influence of noise at the scan area margins.
Decrease this value, if the images have extremely narrow page borders.

Min Color Pixel

Range [1..16], Default: 4

Number of points which must exceed, at a stretch, the Threshold value (or must be white in bitonal images) to detect a border point.
Increase this value if the image background is noisy.

Gradient Error

Range [1..20], Default: 2

By definition, all 4 rectangle borders are right-angled. The rectangle detector compares all normalized gradients of the found borders, and finaly processes only those borders, which normalized gradient is less than the given Gradient Error value.

Increase this value if the papers are not exactly right-angled or if one or more border(s) are not straight (which disallows calculation of a fix gradient).

Max. Error

Range [1..99], Default: 3

The optimal image border line will be created by a linear regression of the detected border points. This regression line will be accepted if all point

distances to this line is smaller than Max Error. Otherwise, the worst point is removed from detection list and regression process is restarted.
Increase this value if the image borders are not straight lines.

Defect factor

Range [1..99], Default: 4

If the destination rectangle holds dark areas near the border, the border detector will identify a border point inside the rectangle (that is the areas end). To prefer removement of these inner points, their distance to the regression line is weight by Defect Factor.
Increase this value if the images have dark areas near the border and the process cuts too much.

See also Deskew Parameter Setup.

3.5.1.5.3.2 Modify Rectangle Size

The borders of the detected rectangle can be moved manually by the following settings. A positive value moves the corresponding border out (enlarge rectangle), a negative value moves it in (cut more).

Left, Right, Top, Bottom

Default: 0

Value to move the corresponding border.

Unit

Range: [Pixel, mm, Inch]
Note: Use values in absolute measurement (mm or Inch) to be independent from changes in resolution.

unit2all

Click this button to set units for Right, Top and Bottom value to the unit selected for Left value.

Mirror horizontal values on backside images

Check to exchange Left and Right value when processing back sides.

See also Deskew Parameter Setup.

3.5.1.5.3.3 Check Rectangle Size

Use this option to protect your images from merciless cutting. If one or more of the following checked rules becomes TRUE, the process will be canceled without changing the image.

No Deskew/Cropping, if ...

Ratio becomes smaller than xxx : 100

xxx : 100
Lower limit for side ratio.

The ratio is calculated as: large side / small side.

Here are some ratio values of standard paper formats:

All A-Formats:	141 : 100
Letter:	129 : 100

Double Letter:	156 : 100
Legal:	165 : 100

Ratio becomes larger than xxx : 100

xxx : 100
Upper limit for side ratio.

See lower limit description above.

Width or Length becomes smaller than xxx %

Border relation limit.

Width becomes smaller than

Absolute lower horizontal limit.

Width becomes larger than

Absolute upper horizontal limit.

Length becomes shorter than

Absolute lower vertical limit.

Length becomes larger than

Absolute upper vertical limit.

Unit

Range: [Pixel, mm, Inch]

Note: mm and Inch values are independent from changes in resolution.

See also Deskew Parameter Setup.

3.5.1.5.3.4 Fill Border

The "Fill Border" processor detects background color, starting from the image borders. The detected areas are filled with chosen border color. This is e.g. useful if the paper edge is slightly damaged by tearing. For the repair of larger damages, this method is not suitable.

Filling Limit

Use limit parameters to prevent the images from running the processor to much inside the document. By default, all are unchecked; that is: unlimited processing.

left

Left border processing limit.

unit2all

Click this button to set units for Right, Top and Bottom value to the unit selected for Left value.

right

Right border processing limit.

top

Upper border processing limit.

bottom

Lower border processing limit.

unit

Range: [Pixel, mm, Inch]

Note: mm and Inch values are independent from changes in resolution.

Mirror horizontal values on backside images Check to exchange Left and Right value when processing back sides.

Expert Settings

Min Corner Pixel Range [0..32], Default: 8

For filling it's important to identify the document corners carefully, because fillings from two directions will meet there. Enter the number of pixels that must surpass the threshold subsequently so that a document corner can be supposed. Higher values will better filter out interferences from the background.

Go Inside Range [0..32], default = 5

After deskewing, the pixel color very close to the image border often do neither belong to the background color, nor match the color of the document. Taking this into account, the detection algorithm will start the given pixels inside the image.

See also Deskew Parameter Setup.

3.5.1.5.3.5 Adjust Image Size

After cutting of the black borders the image can be resized bach to a given format:

Resize Image to Check to size-up the image to a specified format.

Original Image Size Select to bring the image back to the size it had before deskewing/cropping.

Format Select to bring the image to the specified absolute format.

From selection list, choose a standard format, or choose "Custom Size" and enter image dimensions (unit [Pixel, mm, Inch]).

Landscape Check to select landscape size of a standard format.

Resize Mode Choose, how to bring the deskewed/cropped image to the selected size.

Add Frame:

Borders are added arround the image; or, in other words, the image is placed into a cleared image of the selected size. See position parameters below.

Scale Image:

The image is scaled into the selected format. To prevent from undesirable distortion it's recommended to enable "**Skip Resizing, if Size Relation exceeds ...**" (see below).

Horizontal Range [Left, Horiz. Centered, Right] default: Horiz. Centered

Available if resize mode "Add Frame" is chosen. Choose, on what horizontal position inside the new size the deskewed/cropped image should be placed.

Vertical Range [Top, Vert. Centered, Bottom] default: Vert. Centered

Available if resize mode "Add Frame" is chosen. Choose, on what vertical position inside the new size the deskewed/cropped image should be placed.

Skip Resizing, if Size Relation exceeds ... % If checked, the resize process is canceled if size relation exceeds the given value.

It's urgently recommended to enable this inspection with resize mode "Scale Image".

Add Borders Check to add one or more fix-sized borders to the image.

Left Size of border to add to the left side of the image.

unit2all Click this button to set units for Right, Top and Bottom value to the unit selected for Left value.

Right Size of border to add to the right side of the image.

Top Size of border to add to the upper side of the image.

Bottom Size of border to add to the lower side of the image.

Unit Range: [Pixel, mm, Inch]

Note: mm and Inch values are independent from changes in resolution.

Mirror horizontal values on backside images Check to exchange Left and Right value when processing back sides.

See also Deskew Parameter Setup.

3.5.1.5.3.6 Combine Front and Rear Side

The double side deskew processor increases deskew quality when scanning both sides: Front and back side images of one scan show the same document (the rectangle to found) with the same size and the same skew. So it's save to detect the rectangle on any image, and to process process deskew on all images by these result.

Sometimes it's difficult to detect one (or more) borders of the rectangle, because the background color is used in large parts of the original document near the border(s). The best solution is, to detect both side rectangles, align the results and use them to de-skew.

Front/Back Side Rectangle

Define independently for each side, where to detect the rectangle:

- Search Front Side
- Search Back Side
- Search Both Sides, take Best Result (default)

Offset

If there is a static difference in front and back side document position in scanned images (because scanners front and back side camera are not mounted at the same position), this can be corrected by offset setting.

It's recommended to make settings in absolute measurement (mm or Inch) for to be independent from chosen resolution.

Horizontal Enter a positive value to move the back side to the left (if using front side rectangle to deskew back side).

Vertical Enter a positive value to move the back side upwards (if using front side rectangle to deskew back side).

Tolerance

It may happen when aligning both side detection results, that the rectangles differ a little in size or skew angle. The result is accepted anyway, if the differences are in the given tolerance.

Size	Range [0..10.0%], default: 2.0%
Angle	Range [0..10.00°], default: 0.50°

See also Deskew Parameter Setup.

3.5.1.6 Recognition Settings

Barcode Configuration

In this dialog the barcode search can be adjusted. Select the right barcode types and the barcode parameters for the barcode search.

Patchcode Configuration

In this dialog the patchcode search can be adjusted. Select the right patchcode parameters for the patchcode search.

OCR-Configuration

The OCR-configuration contains the setting for area search an full page search.

Please note: If not active maybe there is a newer version of the OCR engine installed. Please check the available PlugIns because newer versions will work as a PlugIn..

3.5.1.6.1 Barcode

The program can evaluate barcodes, for example, to carry out a separation in the event rules or to use the barcode found as the file name.

The program library used here is looking for "normal" one-dimensional barcodes (Code39, Code 2/5, ..). The search for two-dimensional barcodes (QR, Datamatrix) may be performed by a PlugIn

The barcode Dialog has the following pages:

Barcode Types

Barcode Parameter

Barcode Area

Barcode Options

Barcode Enhanced

Test In the displayed edit window is searched for a barcode with the parameters selected in the property pages.

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also Recognition-Dialogs
Help overview
Barcode Parameter
Barcode Types
Barcode Area
Barcode Options

3.5.1.6.1.1 Barcode Types

In the dialog box you can define the barcode types to search for by checking the respective boxes next to the barcode types.

The program can search simultaneously for several different barcodes. Some barcodes, however, can not be selected with others. In that case, the program will disable the selection.

Choose only the necessary barcodes to avoid a time-consuming search.

By marking the checkbox in front of the barcode typename, this barcodetype is considered in the next barcode search on a document.

The following barcode types are available:

- Type EAN 8
- Type CODE 39
- Type CODE 39 with checksum (Modulo 43)
- Type CODE 2/5 (interleaved)
- Type CODE 2/5 (5 lines / IATA)
- Type Codabar
- Type EAN 13
- Type CODE 39 full ASCII
- Type CODE 39 full ASCII with checksum (Modulo 43)
- Type CODE 2/5 (matrix)
- Type CODE 2/5 (industrial)
- Type Codabar (start/stop)
- Type UPC-E
- Type CODE 128
- Type CODE 2/5 (datalogic)

- Type CODE 2/5 (interleaved) with checksum (Modulo 10 with a weight of 3)

See also Help overview, Recognition -Dialogs, Barcode (overview), Parameter, Barcode fields, Options and enhanced options

3.5.1.6.1.2 Barcode Parameter

You can change the default settings to adapt to some special cases where the barcodes are extraordinarily small, of bad quality, or if they appear only in a specified area.

Number of barcodes to search for Specifies the maximum number of barcodes in your document. The maximum number possible is nine.

Minimum barcode height Enter the minimum height for a barcode. Smaller values will increase the search time but make it possible to recognize partially corrupted barcodes. Measuring unit is 0.1mm.

Ratio from largest to smallest stripe Gives the ratio of the width of a broad strip to the width of a narrow strip. Normally, this value is given in the definition of a specific barcode type. You can also enter it manually.

Search order Select the prefered search order, depending on the orientation of the barcode. Please note, that “horizontal” tolerate errors up to 45 degrees in both directions, vertical too. If you want to search in all directions, you should begin with the direction, the most barcodes have. This decrease the search time.

Automatic black-white correction Automatically regards the black and white division.

Min # strips The minimum number of strips to form a meaningful barcode.

Max # strips The maximum number of strips for a specific barcode.

Retries You can type the number of retries into the box.

A value above zero enables recursive search with modified parameters in areas where barcodes are suspected to be. This method practically influences the persistence of the barcode search, charging the search time.

Timeout This option gives you the opportunity to stop a barcode search after a predetermined time. This is useful for scan processes where barcodes are only on a small number of the documents scanned. A search time stops redundant search on a sheet that does not have a barcode.

Get Defaults This button loads the default search parameters for the barcode types you specified.

See also Help overview, Recognition -Dialogs, Barcode (overview), Barcode types, Barcode fields, Options and enhanced options

3.5.1.6.1.3 Barcode Area

Here you can set parameters about the position of the barcode in order to search for barcodes in certain areas of the image.

Barcode areas activated

When activated, the barcode search is executed, using the defined areas. The **% code** column determines via which variable the search result can be queried.
The radio button can be activated only if at least one search area exists.

□

This button adds an area to the list of barcode areas. Position and size are indicated by a colored frame in the preview pane; they can be modified by Windows-conform mouse action.

For proper positioning of the frame, load a document with a barcode on it into the edit window.
For closer information about the functions of the Edit Window, please see the page Edit window.

☒

Deletes all defined barcode areas

✕

Deletes the actually selected barcode area

↑

Moves the actually selected barcode area in the list one line upwards. The variable is automatically changed to the new list position. If, for exemple, the area is moved from position 3 to position 2, the area now has the variable %R2 while the now lower area is bound to the variable %R3.

↓

Moves the actually selected barcode area in the list one line downwards. The variable is automatically changed to the new list position. If, for exemple, the area is moved from position 2 to position 3, the area now has the variable %R3 while the now lower area is bound to the variable %R2.

x

Position of the left border of the barcode area, seen from the left image rim. The value can be changed by direct input, or by moving the frame in the preview pane.

y

Position of the top border of the barcode area, seen from the image top. The value can be changed by direct input, or by moving the frame in the preview pane.

Width

Width of the barcode area. The value can be changed by direct input, or by moving the frame in the preview pane..

Height

Height of the barcode area. The value can be changed by direct input, or by moving the frame in the preview pane..

%code

Name of the variable that is used to evaluate the recognition result of the relative area. The name is assigned automatically and cannot be altered. When you move the area inside the list, its name is newly assigned. If, for exemple, the area is moved from position 2 to position 3, the area now has the variable %R3 while the now lower area is bound to the variable %R2.

See also Help overview, Recognition -Dialogs, Barcode (overview), Barcode types, Parameter, Options and enhanced options

3.5.1.6.1.4 Barcode Options

Here you can set parameters about the additional workflow and output of teh search.

Further Options There are some more options for the barcode search:

Write barcode info file Activate this option to create a text file with the barcode information. This file has the same name as the corresponding image file. The extension is .BAR. Each recognized barcode value will be stores in a new line of the file. In case of multipage image files each found barcode value will be added until a new file begins. The file does not contain any additional values like position or type.

In Basic version this is the only possibility to use the barcode information.

Rotate 90/180 degrees if no barcode was found Activate this option to search for a barcode again on a 90/180 degrees rotated image, if the search has failed the first time.

This option forces a new barcode search with other search direction, if no barcode was found. It is also useful, if an area is defined and the paper has the wrong direction to correct the image direction automatically.

Undo rotation Activate this option to rotate images back, which has been rotated for barcode search (last option)

- if a barcode was found but no barcode area was defined.
- no barcode was found even after a rotation.

No search on backside if frontside has barcode This option is useful to prevent errors with the option **never seach on backside/frontside**. No barcode is missed, even if the paper has swapped front and backside.

Never search on backside/frontside Barcode search need much time, if no barcode is on the image. If it is known, that no barcode is on the backside, the search on the backside should be disabled. On the other hand it is possible to miss a barcode, if the paper has swapped front and backside.

Image size Options Enable these options by setting the values to a number different from zero. The number represents the compressed G4 size of the image in bytes.

Maximum size of image The barcode search will then ignore all images that are larger than this value, in compressed condition. This is helpful if barcodes are on divider sheets that do not contain any other information.

Minimum size of image The program will search for barcodes only when images that in compressed condition are at least this size. You can use this function to stop the system for manually entering a barcode (or its index) if the DPU did not find a barcode.

See also Help overview, Recognition -Dialogs, Barcode (overview), Barcode types, Parameter, Barcode fields, and enhanced options

3.5.1.6.1.5 Barcode Enhanced

Here some further parameters can be set for the search for barcodes of type 2/5, Code39 and Code93.

Character Relation (Code2/5)

Controls the relation tolerated maximally from the wides to the narrowest bar within one character for the 2/5 barcode family. The thickest stroke must not wider be than n (here specified value) times thinnest stroke width.
Possible value range: 2 to 15, default is 5

Magnitude Tolerance (Code2/5)

Controls for the barcodes 2/5 family, the tolerance threshold when deciding between thick and thin strokes.

By definition, there is a 2/5 bar from "thick" and "thin" strokes; accurately: per group of digits (= 5 bars) 2 thick - hence the name TWO of five.

Thick lines are defined as: at least 2 times, less than 3 times the width of thin strokes. Thick lines must therefore be theoretically at least 100% more than thin lines.
In practice: the DPU-Decoder accepts a minimum thickness margin of 30%.

Character Relation (Code39

Controls the relation tolerated maximally from the wides to the narrowest bar within one character for the 39 barcode family. The thickest stroke must not wider be than n (here specified value) times thinnest stroke width.
Possible value range: 2 to 15, default is 5

Magnitude Tolerance (Code39)

Controls for the barcodes 39 family, the tolerance threshold when deciding between thick and thin strokes.

By definition, there is a 39 bar from "thick" and "thin" strokes; accurately: per group of digits (= 9 bars) 3 thick - hence the name THREE of NINE.

Thick lines are defined as: at least 2 times, less than 3 times the width of thin strokes. Thick lines must therefore be theoretically at least 100% more than thin lines.

In practice: the DPU-Decoder accepts a minimum thickness margin of 30%.

Tolerance (Code93)

This value applies to Code93 barcodes. This type uses four different line thicknesses, so that there would not be enough to describe a relation between thick and thin strokes.

Instead, defines this value, how much may vary the width of a line from the normal width.

Possible values are 0% to 50%, default is 20%.

Default values

Resets all parameters on this page but those on the other pages.

See also Help overview, Recognition -Dialogs, Barcode (overview), Barcode types, Parameter, Barcode fields and Options

3.5.1.6.2 Patchcode

Patchcodes are a good method for the process control. The options for the search can be set here. This process doesn't affect the Patchcodes-Search inside the scanner, iff the machine is equipped with a patchcode reader.

Options

Patchcode has bad quality If the patchcode has bad quality, enable this option to make a thorough patchcode search.

Patchcode has oversize Patchcode should have a height of 20mm. Enabling this option accepts larger Patchcodes (20-25mm)

Patchcode has small strips Patchcode should have a height of 20mm. Enabling this option accepts smaller Patchcodes (15-20mm).

Patchcode Search area

Patchcode area enable Activate this checkbox to search the patchcodes in the defined area. This options is helpful, if a patchcode is placed at the same location on every image.

Patchcode area This list shows the patchcode area definition. The current patchcode area is displayed in the edit window

The size and the position of the frame can be changed in the edit window.

Press and hold the left mouse button in a frame to move it. Click the frame margin to change the size of the frame.

Type Selection Here the search can be limited to certain codes, normally all codes are reacted to.

Buttons

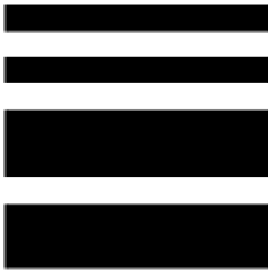
Test In the displayed edit window is searched for a patchcode with the parameters selected in the property pages.

OK The dialog box is closed. All settings are saved.

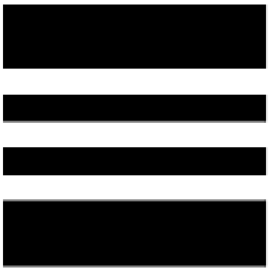
Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

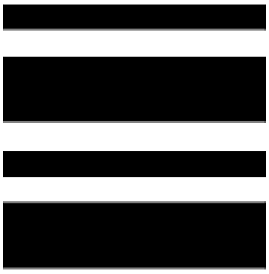
3.5.1.6.2.1 Patchcode_Types



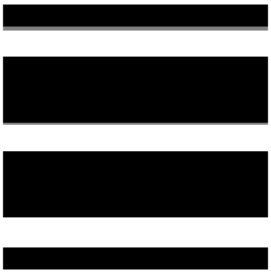
Patch I



Patch II

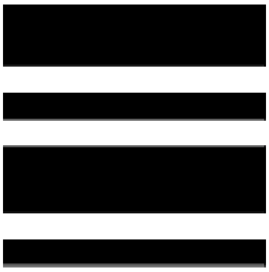


Patch III



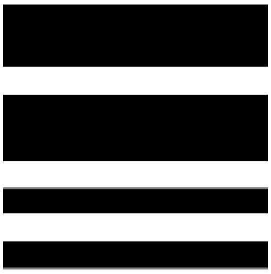
Patch IV

"Toggle"

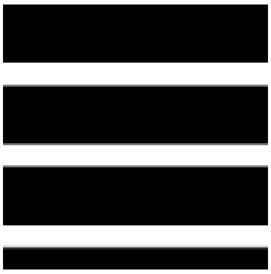


Patch T

"Transfer"

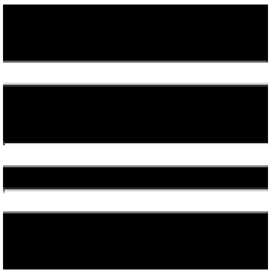


PatchVI



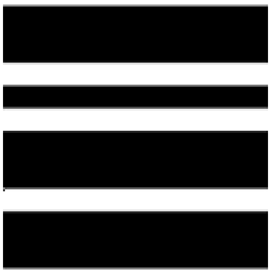
Patch VII

"Bitonal"



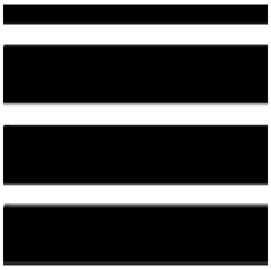
Patch VIII

"Color"

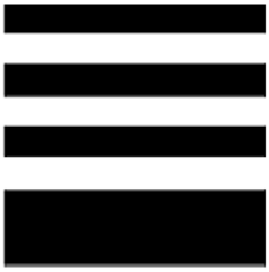


Patch IX

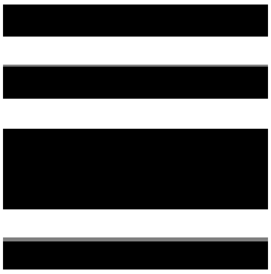
"Gray"



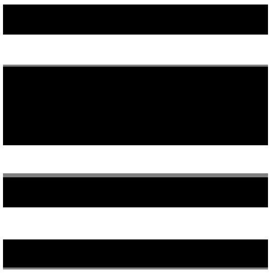
Patch X



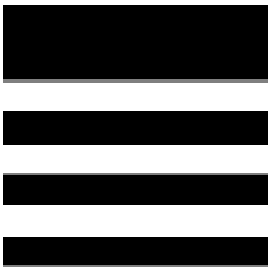
Patch XI



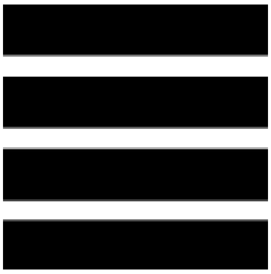
Patch XII



Patch XIII



Patch XIV



Patch XV

3.6 Application Layout

The program can be controlled via the menu or via a series of buttons. Several buttons are displayed together in so-called toolbars. Several Toorbars together form a layout profile for the application. Such Application Layout Profiles can be loaded within a Task.

Together with the program will be delivered two Application Layout Profiles, one profile with few buttons ,"MINIMAL_UI", and one with the most frequently used buttons,"EXTENDED_UI". Application Layout Profile can be managed here.

In addition to the list of profiles are these buttons

✳	Creates a new Application Layout Profile
✗	Removes a new Application Layout Profile, if it is not used in other task profiles
✓	Accept the changes and stores the profile
ⓘ	Shows the tasks where this Application Layout Profile is used.

Changes to the current Application Layout Profile can be made on the tabs below:

Toolbars
Actions
XINO_Buttons

The standard buttons at the bottom will work in the known manner:

OK	Saves all changes and closes the dialog.
Cancel	Closes the dialog without changing the settings.
Apply	Saves all changes without to close the dialog.
Help	Opens this screen.

Please note:

The program allows for very complex settings in the base profile definition and the task. Wrong settings may have catastrophic consequences. Therefore, it must always be secured that not every user can change these settings. Therefore, a specific Application Layout Profile can be set for every single scan station, where either no or only selected settings made by the user are possible.

See also Help overview

3.6.1 Toolbars

The program allows for very complex settings in the base profile and the task. Wrong settings may have catastrophic consequences. Therefore, it must always be secured that not every user can change these settings.

Therefore, a specific User Profile can be set for every single scan station, where either no or only selected settings made by the user are possible.
You can change the toolbar of the program by making other user profiles which doesn't perform the whole functionality of the program.
In the application layout configuration all these settings can be made.

User Profile The User Profile can be selected by clicking the button, if it has been configurated. The appropriate settings will be taken. The green button on the left indicates, which User Profile is configurated

Delete actual User Profile Deletes the selected User Profile.

Toolbar list All available toolbars are listed. Select the toolbar you want to have in your User Level.

Column: Toolbar To switch the marked toolbar on or off.

Text The display of the text in the toolbar can be switched on or off. It must be switched on either text or symbol. If text is switched off the symbols are switched on in any case, even if they was switched off previously.

Symbols The display of the symbols in the toolbar can be switched on or off. Switching off is only available, if Text is switched on.

Size You can set the size of the symbols in the toolbar. Possible selections are small symbols, normal symbols and large symbols.

Menu actions Restricts the access via context menu to the following actions: to hide the toolbar, toggle the symbol display mode or the text display mode, change of the symbol size, close or move the toolbar.

Apply standard commands to actual Toolbar Undo all changes of the marked toolbar.

Apply standard commands to all Toolbars Undo all changes of all toolbars.

Menu visible Sets the display status of the menu bar.

Statusline visible Sets the display status of the statusline.

The related combobox allows to set the font size of the statusline.

Maximize Dialogs Allows the automatic adjustment to the current display resolution.

See also

- Help Overview.
- Applikation Layout
- Toolbars
- Actions
- XINO-Buttons

3.6.2 Actions

All available commands in the program are listed on this page. They are grouped by functionality. It is possible to assign hotkeys to these commands to execute them with the keyboard. It can be specified when and where these hotkeys are available.

It is also possible to execute many commands in a row, see Macros and Macro Editor.

By the User Rights Settings all commands can be allowed or denied for a single user or a group of users. It may be useful to show a toolbar button despite its corresponding action is forbidden, e.g. to grant a common user interface.

Command Tree

To find commands easier, the number of visible commands in the right window can be restricted. Depending on the command group in the tree the appropriate commands to this category are displayed in the right window.

Please note:

The commands for image selection are divided into two categories. The Nodes "Skip disallowed" and "Skip allowed" contain commands with similar names and similar icons but different colors. The behaviour is depending on the settings for the display of thumbnails. Details on the difference are explained on the page "Difference: Skip Allowed - Skip Disallowed".

Table with commands

The table view allows you to configure the individual commands.

Column Command

Shows the icon and the name of the command

Column ID

For commands using parameters this column shows the name of the parameter set or subprofile

Column Status

The command can be activated or disabled. When you disable the command, its according button is gray and can not be clicked. The command is then generally deactivated, independent of conditions.

The corresponding menu entry, if there is one, is enabled or disabled too.

Column Hotkey

You may assign a hotkey to every command; the key combination can freely be selected, using individual keys or combinations with Ctrl, Alt and Shift and the relative letter.

You can delete an already assigned hotkey by pressing the Ctrl, Alt or shift key.

If your newly defined hotkey is already assigned to another command, the program will throw a warning message. You can then define a different hotkey, or overwrite the hotkey of the old command.

Column LegalArea

You can set a legal area for the command (and thus restrict it to certain areas of the program). The set legal area is shown by a certain letter. Click to the line to open a dialog where settings can be made, using check boxes. Possible legal areas are:

Application (A) including toolbars
Tree (T)
ThumbNail (N)
Scanwindow (S)
External Dll (E) (Windows owned by a PlugIn)
Infowindow (I)
Warnwindow (W)
Batchwindow (B)
Toolbox (T)

Column Condition

Here, you can enter a condition. The command will be executed only if the result of this condition is "true".

Column Message

If the expression in the column Condition becomes false the command is not available. When you check this box, the according command will be activated nevertheless. If the command is called although the conditions do not allow it, the text from the **Message Text** column will display as a message on the screen.

Example:
As long as blue marks exist in a batch, it can not be finalized. The corresponding button is inactive.
If the check box is activated in the **Message** column, also the **Finalize** button is active again.
If the **Finalize** command is executed although blue marks still exist, the message will be thrown as entered in the **Message Text** column. If no text is entered there, a standard message will be reported. In this actual example, the software would hint to the fact that finalization is not possible as blue marks still exist.

Column Message Text

The text that you can enter here will display as a message in case that the box is checked in the **Message** column, but the command can not be executed.

Column Button Text

Contains the text that displays under the button if you have enabled that such texts are generally displayed.

Column Status Text

Contains the text that will display in the status line.

Column ToolTip

Contains the text that will display as tool tip when the mouse pointer is rested for some time on the button.

Icon

You can change the standard icon of the marked command by selecting another icon in a list.

Parameter

Some commands need additional parameters. This can be adjusted by this button

Macro

This button defines a macro. Several commands are performed in a row, so a combination of all available commands is assigned to one macro.

Reset

The marked command is set to the default value. You can choose the command columns (Command, Status, Hotkey, Legal Area, Event, EventMsg, MsgText, Button Text, Status Text, ToolTip) which should be set back.

Reset all

All available commands are set to the default value. You can choose the command columns (Command, Status, Hotkey, Legal Area, Event, EventMsg, MsgText, Button Text, Status Text, ToolTip) which should be set back.

See also

Help Overview.
Application Layout
Toolbars
XINO-Button

3.6.2.1 Macro Editor

The program allows it to combine commands to so-called macros that can be executed by a click on a toolbar button or a keystroke.

This dialog is opened to create a new macro or to change an existing one. It shows at the upper part the available commands and the lower part the active commands. In between, there are the usual switches to activate the available action and if necessary to describe the actions more precisely:

Available Commands

The available commands are shown on the left as a tree and on the right as a list. In the list only the actions offered belonging to the selection of the command group in the tree. Also, certain commands are not available, which could lead to a recursion.

The selection is also limited for actions that are not allowed when more than one image is selected.

Switches and buttons

Working with multi-selections:

Process every step on all images

Executes the first command for the first image, then for the second image, then for the third etc. Then the second instruction is done on the first image, then on the second image etc.

Process all steps on every image

Executes all commands for the first image, then all commands for the second image, then on the third etc.

In this mode, all actions are prohibited which have an effect on the entire batch, e.g: Insertion or removal of images, repeating the event rules, exit the program program, etc.

Process all steps on the focused image

Executes all commands only for that image, that was selected as last and has currently the input -focus.

Forbid macro for multiselection

Doesn't execute the macro if more than one image is selected.

Parameter

This button opens a dialog to select a subprofile if this process step needs parameters in form of a subprofile, e.g. filter, PlugIn, Enhanced Modify, ...

List buttons

⇩ Add	Inserts the new command after the highlighted one.
☒	Remove all elements from the list
✕	Removers the selected element from the list
↑	Moves the element one position up
↓	Moves the selected element one position down.

Active Commands

The list of the active commands shows the properties of the process step in these 7 columns:

Active

Indicates that this command is activated. The hook can be removed e.g.to perform a test without this command.

Name

here a description can be given.

Command

Name of the command

ID

name of the subprofile, which is used for this action.

Properties

When a command is executed, the following values will be returned:

Action was successful	1
Action was not successful	0
Action was not executed	-1

Action was not executed means that the execution has been prevented by a given condition.

Cancel macro, if error

The macros ends after this instruction, if this command returns an error. For example don't print a batch sheet if the finalizing process failed because there were still blue marks in batch.

Cancel macro, if success

The macros stops here if no errors were returned, e.g. don't export the batch to the error folder if the finalizing has succeeded before.

Cancel macro, if ignored

In some cases an instruction doesn't make sense, e.g. "Go to the next green mark" if there aren't any marks in the batch. In this case the program disables the corresponding button and draws it in gray color. The same happens to commands which are forbidden by a user defined condition when this condition becomes true. This state of inactivity will be checked here.

Ask

There are control commands having a confirmation dialog, e.g. Close Program. This confirmation will be displayed when the **button in the toolbar** is clicked. On the other hand **in a macro** this message doesn't come. The reason is to prevent an interruption of the automated work flow in a macro.

With this option the confirmation is enabled again.

Only execute if

Here a condition can be defined to allow the execution of the instruction.

Return value

Here an assignment of a variable can be given, which is executed after finishing the action successfully.

A double click on a line in the list of the active commands will synchronize automatically the selection in the upper part.

3.6.2.2 Command Groups

The commands can be combined in groups. In settings dialog for application layout is the following classification used:

System

Image Position

Data transfer

See also application layout and User Interface for further controls.

3.6.2.2.1 System

In this command group following commands are combined:

Icon	Command	Description	Hotkey	Restriction
	UserRights	Defines which persons and groups may use which tasks and comannds.		Only in pause mode
	SetupTask	Shows setup dialog for task profile		
	SetupBase	Shows setup dialog for base profile		
	SetupLayout	Shows application layout setup dialog (toolbars & macros)		

	SetupDisplay	Shows display setup dialog (pans & windows)	
	SetupScanner	Shows scanner setup dialog (First & second scanner)	
%	SetupPercent	Shows system variables dialog, (%-Codes)	
DAT ?	ProfileExport	Exports chosen profiles to a newly created DAT-File	Only in pause mode
DAT	ProfileExport Quick	Exports the current profile to a newly created DAT-File	Only in pause mode
DAT ?	ProfileImport	Opens a DAT-File to select profiles and subprofiles to be imported. **Note: Existing Profiles will be overwritten!**	Only in pause mode
DAT	ProfileImport Quick	Reads and imports a profile from a a DAT-File. **Note: Existing Profiles will be overwritten!**	Only in pause mode
	License	Opens the dialog to enter license information.	
	LicenseTest	Retries to get a license, for example after a reinitialization of a lost connection to a DpuServer.	
i	ApplicationInfo	Shows the program version	Only in pause mode
i	System Information	Opens the dialog for the systeminformataion, the logging setup and the profile management.	
🟥	Application Exit	Closes the program. OpenJobs will be closed but not finalised.	
▶	AppStart	This is a deny-only-icon: It does not work, but it can be used in the User Rights and Grants settings. to deny the corresponding action for groups of users. For example you may forbid the program start for Non-Scan-Operators. Attention: Be very careful when forbidding the program start. If you lock out the last user accidently, you have reinstall the software!	Deny only

	Macro Configuration	This is a deny-only-icon: It does not work, but it can be used in the User Rights and Grants settings. to deny the corresponding action for groups of users.		Deny only

In addition to these buttons or commands, the **separator** is also offered in this branch of the configuration dialog. The Sseparator is a bar to group command groups on a toolbar.

3.6.2.2.2 Image Position

In this command group following commands are combined:

Icon	Command	Description	Hotkey	Restriction
	ZoomOut	Zooms to the previous lower step.	NUM5	
	PanCenter	Moves the visible area to the middle of the image.		
	PanUp	Moves the visible area up.	NUM8	
	PanDown	Moves the visible area down.	NUM2	
	PanLeft	Moves the visible area to the left.	NUM4	
	PanRight	Moves the visible area to the right.	NUM6	
	PanTopLeft	Moves the visible area to the top left corner of the image.	NUM7	
	PanTopRight	Moves the visible area to the top right corner of the image.	NUM9	
	PanBottomLeft	Moves the visible area to the bottom left corner of the image.	NUM1	
	PanBottomRight	Moves the visible area to the bottom right corner of the image.	NUM3	

3.6.2.2.3 Data transfer

In this command group following commands are combined:

Icon	Command	Description	Hotkey	Restriction
	DataReceive	Transfer from the PlugIns to the Program: Receives from all external modules (PlugIns) all data offered and assigns them to the current image/file/folder/dokument.		
?	DataReceiveAsk	Transfer from the PlugIns to the Program: Asks all loaded external modules (PlugIns) about data updates. If so, it receives all data and assigns them to the current image/file/folder/dokument..		
	DataSend	Transfer from the Program to the PlugIns Sends all data of the currently selected image/file/folder/document to all external modules (PlugIns) loaded.		

Usually the data will be exchanged on each change of the selection. These commands are intended to force an exchange without leaving the currently selected image.

Icon	Command	Description	Hotkey	Restriction
UDD f(x)	SetVariable	Sets the Return-Variable for the selected image.		Macro only
UDD f(x)	SetVariable ImageGroup	Sets the Return-Variable for all images of the image group of the selected image. Please consider the rules for variable assigment.		Macro only
UDD f(x)	SetVariable ImagePaper	Sets the Return-Variable for all images of front and rear side of the selected image. Please consider the rules for variable assigment.		Macro only

3.6.2.2.4 User macros

In this command group the macros are combined:

Icon	Command	Description	Hotkey	Restriction
M 1	UserMacro	Executes a macro. These sequences of instructions can be programmed in the Macro-Editor.		

The program can handle up to 32 macros.

3.6.2.2.5 Editwindow menu

In this command group commands are summarized with which a picture can be examined or subsequently edited.

Please regard:

- All actions apply only to the last selected and displayed image.
- All commands in this group use the subprofile select in the base profile.
- Changes to the settings made here are temporary and will be discarded when starting the next batch.

A distinction is made between commands with the optional option of setting the parameters and those that are executed immediately:

Parameter commands
Immediately commands

The changes made can be undone or accepted:

Icon	Command	Description	Hotkey	Restriction
✓	ParamApply	Accepts the changes made to the selected image.		
↶	ParamUndo	Rejects the changes made to the selected image.		

3.6.2.2.5.1 Parameter Commands

In detail, these steps are available:

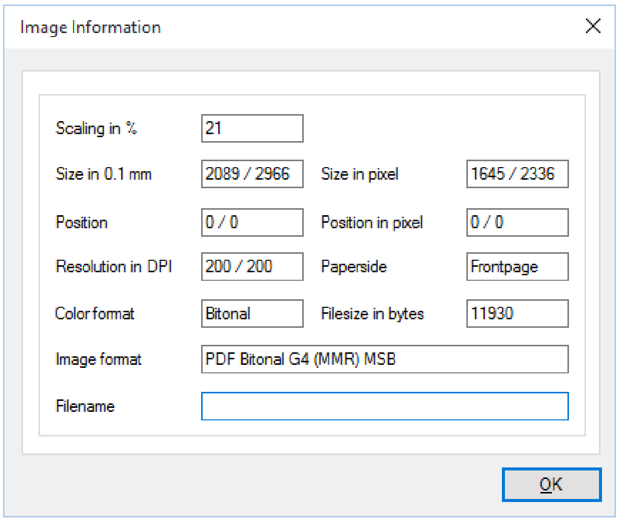
Icon	Command	Description	Hotkey	Restriction
	BarcodeMenu	Searches for barcodes and shows the result. The barcode variables of the image will not be changed A click on the arrow will open the Barcode-menu.		
ABC	BatesMenu	Inserts a stamp into the currentyl selected image. A click on the arrow will open the Bates Stamping menu.		
	FilterMenu	**Replaces** the selected grayscale or BW image with new one filtered from the corresponding color- or grayscale image of the same image group. A click on the arrow will open the Bates Stamping menu.		Color or Gray image in the background

	InfoMenu	Shows a window with the image information. A click on the arrow opens the corresponding menu.		
	Image LoadSaveMenu	Opens a menu to load or save an image		
	ModifyMenu	Improves image quality through noise removal, line removal, etc. A click on the arrow opens the corresponding menu		Bitonal images only
	OCRMenu	Searches for Texts and shows the result. The OCR variables of the image will not be changed A click on the arrow will open the OCR-menu.		OCR PlugIn loaded
	PatchcodeMenu	Searches for Patchcodes and shows the result. The Patchcode variables of the image will not be changed A click on the arrow will open the Patchcode-menu.		
	PrintMenu	Prints the currently selected image on the default printer. A click on the arrow opens the corresponding menu		
	OCRMenu	Searches for Texts and shows the result. The OCR variables of the image will not be changed A click on the arrow will open the OCR-menu.		OCR PlugIn loaded
	DeskewMenu	Opens a menu to rotate or de-skew an image		
	ScanMenu	Replaces the currently selected image with an image from scanner used lately. The click on the arrow opens the corresponding menu.		

If a setting dialog has fallen into the background, it can be brought to the foreground again.

Icon	**Command**	**Description**	**Hotkey**	**Restriction**
	WindowsSwap	Brings an accidentally hidden setup dialog back to front.		

3.6.2.2.5.2 Immediately Commands

Icon	Command	Description	Hotkey	Restriction
	Barcode	Searches for barcodes with the currently loaded set of parameters.		
	Barcode Configuration	Opens the setup dialog for the barcode search.		
	Bates	Stamps the actual image using the currently loaded set of parameters.		
	Bates Configuration	Opens the setup dialog to change the current settings for the bates stamping process.		
	FilterAnwenden	**Replaces** the selected grayscale or BW image with new one filtered from the corresponding color- or grayscale image of the same image group.		Color or Grayscle images exists
	Filter Configuration	Opens the setup dialog for the filtering process.		
	ShowImage Info	Shows a window with the image information  Image Information Scaling in % 21 Size in 0.1 mm 2089 / 2966 Size in pixel 1645 / 2336 Position 0 / 0 Position in pixel 0 / 0 Resolution in DPI 200 / 200 Paperside Frontpage Color format Bitonal Filesize in bytes 11930 Image format PDF Bitonal G4 (MMR) MSB Filename OK		
	ImageInvert	Inverts the cloors of the actual image.		
	ImageLoad	Opens a dialog to load an image from the disk.		
	ImageSave	Saves the current image.		

?	ImageSaveAs	Opens a dialog box to save the current image with a new name.	
	Modify	Modifies the actual image using the currently loaded set of parameters.	b/w only
	Modify Configuration	Opens the setup dialog to change the current settings for the Enhanced Modify process	
Ocr	OCR	Searches for text on the actual image using the currently loaded set of parameters.	OCR PlugIn loaded
Ocr	OCR Konfiguration	Opens the setup dialog to change the current settings for the text recognition.	OCR PlugIn loaded.
	Patchcode	Searches for patchcodes with the currently loaded set of parameters	
	Patch Configuration	Opens the setup dialog to change the current settings for the patchcode search.	
	Print	Prints the actual image using the currently loaded set of parameters.	
	Printer Configuration	Opens the setup dialog to change the current settings for the printout process.	
90°	Image RotationL90	Rotates the image 90° clockwise.	
180°	Image Rotation180	Rotates the image upside down.	
90°	Image RotationR90	Rotates the image 90° counterclockwise.	
	Deskew	Aligns, de-skews, the image and removes the black borders.	
	Deskew Configuration	Opens the setup dialog to change the current settings for the deskew process	
	Deskew Manual	Opens a help window to rotate and cut an image manually.	

3.6.2.2.6 View

In this command group following commands are combined:

Icon	Command	Description	Hotkey	Restriction
1 + −	ChangeDisplay Frontside	Displays or hides frontsides		
1 −	SetDisplay FrontsideOff	Hides front sides.		
1 +	SetDisplay RearsideOn	Displays front sides.		
2 + −	ChangeDisplay Backside	Displays or hides rearsides		
2 −	SetDisplay BacksideOff	Hides rear sides.		
2 +	SetDisplay BacksideOn	Displays rear sides.		
+ −	ChangeDisplay BitonalImages	Shows or hides black/white images.		
−	SetDisplay BitonalImagesOff	Hides black/white images.		
+	SetDisplay BitonalImagesOn	Displays black/white images.		
+ −	ChangeDisplay GrayImages	Shows or hides gray images.		
−	SetDisplay GrayImagesOff	Hides gray images		

−	SetDisplay GrayImagesOn	Displays grayscale images.
+ −	ChangeDisplay ColorImages	Shows or hides color images.
−	SetDisplay ColorImagesOff	Hides color images..
+	SetDisplay ColorImagesOn	Displays color images.
+ −	ChangeDisplay RedMarked	Displays or hides red marked images. Images with red marks will be rejected when finalizing the job.
−	SetDisplay RedMarkedOff	Hides images with red marks. Bilder mit roten Marken werden beim Finalisieren verworfen
+	SetDisplay RedMarkedOn	Displays images with a red mark. Bilder mit roten Marken werden beim Finalisieren verworfen
+ −	ChangeDisplay BlueMarked	Displays or hides blue marked images. Blue marks prevent the finalization of a batch.
−	SetDisplay BlueMarkedOff	Hides images with blue marks. Blue marks prevent the finalization of a batch.
+	SetDisplay BlueMarkedOn	Displays images with a blue mark. Blue marks prevent the finalization of a batch.
+ −	ChangeDisplay GreenMarked	Displays or hides green marked images. Green tags identify important images.
−	SetDisplay GreenMarkedOff	Hides images with green marks. Green tags identify important images.
+	SetDisplay GreenMarkedOn	Displays images with a green mark. Green tags identify important images.
+ −	ChangeDisplay WhiteMarked	Displays or hides white marked images. Images with white marks can be hidden - regardless of other marks.

−	SetDisplay WhiteMarkedOff	Hides images with white marks. Images with white marks can be hidden - regardless of other marks
+	SetDisplay WhiteMarkedOn	Displays images with a white mark. Images with white marks can be hidden - regardless of other marks.
+ −	ChangeNormal	Displays or hides unmarked images.
−	SetDisplay NormalOff	Hides images without marks.
+	SetDisplay NormalOn	Displays images without marks.
	ColorFormats DisplaySave	Saves the display options for the current view
	ColorFormats DisplayLoad	Restores the display options for the current view

3.6.2.2.6.1 Working Area

This command group combines the commands that can be used to update the workspace or to show and hide certain windows.

Icon	Command	Description	Hotkey	Restriction
i	ToggleInfo Window	Shows or hides a free configurable window holding information, e.g. %-Codes	Shift+ Backspace	
P	ToggleBatchfile Window	Shows or hiodes a preview window for the batchfile	Shift+Backspace	
	ToggleToolbox Window	Shows or hides the toolbox window.	Ctrl+Backspace	
!	ToggleWarnings Window	Shows or hides the window for warnings.	Ctrl+Alt+ Backspace	There are messages
	ViewUpdate	Updates the tree view, image and image preview windows.The event rules are NOT executed.		

	Window SelectNext	Goes to the next pan of the workspace.		
	Window SelectPrevious	Goes to the previous pan of the workspace.		

See also Image Area and Tree View.

3.6.2.2.6.2 Image Area

In this command group the commands are summarized, with which the view on certain pictures can be restricted. This is e.g. useful in quality control.

Icon	Command	Description	Hotkey	Restriction
1 + −	ChangeDisplay Frontside	Displays or hides frontsides		
1 −	SetDisplay FrontsideOff	Hides front sides.		
1 +	SetDisplay RearsideOn	Displays front sides.		
2 + −	ChangeDisplay Backside	Displays or hides rearsides		
2 −	SetDisplay BacksideOff	Hides rear sides.		
2 +	SetDisplay BacksideOn	Displays rear sides.		
+ −	ChangeDisplay BitonalImages	Shows or hides black/white images.		
−	SetDisplay BitonalImagesOff	Hides black/white images.		
+	SetDisplay BitonalImagesOn	Displays black/white images.		

ChangeDisplay GrayImages	Shows or hides gray images.
SetDisplay GrayImagesOff	Hides gray images
SetDisplay GrayImagesOn	Displays grayscale images.
ChangeDisplay ColorImages	Shows or hides color images.
SetDisplay ColorImagesOff	Hides color images..
SetDisplay ColorImagesOn	Displays color images.
ChangeDisplay RedMarked	Displays or hides red marked images. Images with red marks will be rejected when finalizing the job.
SetDisplay RedMarkedOff	Hides images with red marks. Bilder mit roten Marken werden beim Finalisieren verworfen
SetDisplay RedMarkedOn	Displays images with a red mark. Bilder mit roten Marken werden beim Finalisieren verworfen
ChangeDisplay BlueMarked	Displays or hides blue marked images. Blue marks prevent the finalization of a batch.
SetDisplay BlueMarkedOff	Hides images with blue marks. Blue marks prevent the finalization of a batch.
SetDisplay BlueMarkedOn	Displays images with a blue mark. Blue marks prevent the finalization of a batch.
ChangeDisplay GreenMarked	Displays or hides green marked images. Green tags identify important images.
SetDisplay GreenMarkedOff	Hides images with green marks. Green tags identify important images.

+	SetDisplay GreenMarkedOn	Displays images with a green mark. Green tags identify important images.
+ −	ChangeDisplay MagentaMarked	Displays or hides magenta marked images. Magenta tags identify important images.
−	SetDisplay MagentaMarked Off	Hides images with magenta marks. Magenta tags identify important images.
+	SetDisplay MagentaMarked On	Displays images with a magenta mark. Magenta tags identify important images.
+ −	ChangeDisplay OrangeMarked	Displays or hides orange marked images. Orange tags identify important images.
−	SetDisplay OrangeMarked Off	Hides images with orange marks. Orange tags identify important images.
+	SetDisplay OrangeMarked On	Displays images with a orange mark. Orange tags identify important images.
+ −	ChangeDisplay CyanMarked	Displays or hides cyan marked images. Cyan tags identify important images.
−	SetDisplay CyanMarked Off	Hides images with cyan marks. Cyan tags identify important images.
+	SetDisplay CyanMarked On	Displays images with a cyan mark. Cyan tags identify important images.
+ −	ChangeDisplay WhiteMarked	Displays or hides white marked images. Images with white marks can be hidden - regardless of other marks.
−	SetDisplay WhiteMarkedOff	Hides images with white marks. Images with white marks can be hidden - regardless of other marks
+	SetDisplay WhiteMarkedOn	Displays images with a white mark. Images with white marks can be hidden - regardless of other marks.
+ −	ChangeNormal	Displays or hides unmarked images.

	SetDisplay NormalOff	Hides images without marks.
	SetDisplay NormalOn	Displays images without marks.
	ColorFormats DisplaySave	Saves the display options for the current view
	ColorFormats DisplayLoad	Restores the display options for the current view

See also Tree View and Working Area

3.6.2.2.6.3 Tree View

In this command group the commands are summarized, with which the view on certain pictures can be restricted. This is e.g. useful in quality control.

Icon	Command	Description	Hotkey	Restriction
	TreeExpand	Explodes all branches in the tree view below the selected node.		
	TreeExpandAll	Explodes all branches in the tree view below the selected node.		
	Treebranch InOut	Toggles the currently open branch in the tree view.		
	Treebranch InOutAll	Toggles all branches in the tree view below the selected node.		
	TreeCollaps	Collapses the currently open branch in the tree view.		
	TreeCollapsAll	Collapses all open branches in the tree view.		

See also Image Area and Working Area

3.6.2.2.7 Control

In this command group following commands are combined:

Icon	Command	Description	Hotkey	Restriction
▶	StartTask	Runs the loaded task, usually: load profiles, open stack, scan, ...	Return	
⊗	CancelTask	Stops the scanner and each further processing. The steps in and after the scan loop will not be processed. In OpenJob mode the batch will be closed automatically. In DirectMode the intermediate files will be removed. Goes into state "Ready".		
▶	ContinueTask	Starts the scanner again		Pause mode
	Process EventRules	Removes all marks and all file or directory variables. Then it processes the event rules for each picture again.		Eventrules defined
	ExportBatch	Copies the current batch to a specified export folder and removes it afterwards from the own Open_Job-Folder		OpenJob Mode
	Export FinalizeBatch	Finalize the batch and copies the open job to the given export folder.		OpenJob Mode
	FinalizeBatch	Finalizes the current batch.		
	FinalizeBatch Extern	Starts a Background-Task to finalize the currently scanned images.		License requested
⏸	PauseTask	Stops the scanner, fetches and processes the last scanned images, but doesn't execute the overrun. After clicking this button the program goes to mode "Stopping" and after fetching the last image it goes to "Pause".		
	ImportBatch	Copies a batch from a specified import folder to the own Open_Job-Folder.		OpenJob Mode
▶⏸	StartStopTask	Starts and stops the task. Makes it possible to handle start/stop with one key on the keyboard..	Spacebar	
↻	TaskRestart	Ends the current Task and restarts it again. Reserved for use in a macro.		

	ProcessBatch Extern	Starts a Background-Task to process the the currently scanned images, e.g. to perform a text recognition.		License requested
	PurgeBatch	Deletes the batch from the OPEN_JOB folder.		
	Renumber Filenames	Numbers the files through again and considers deleted and inserted images.		Automatic #-File names defined
	StopTask	OpenJob mode When scanning it stops the scanner, fetches and processes the last scanned images from the buffer. Then it closes the batch When pausing it just closes the batch. Direct Mode: When scanning it stops the scanner, fetches and processes the last scanned images from the buffer. Then it processes the overrun and finalizes the batch. In all cases the overrun will be executed.		

3.6.2.2.8 Process

These groups combines the commands to change single images or the whole batch:

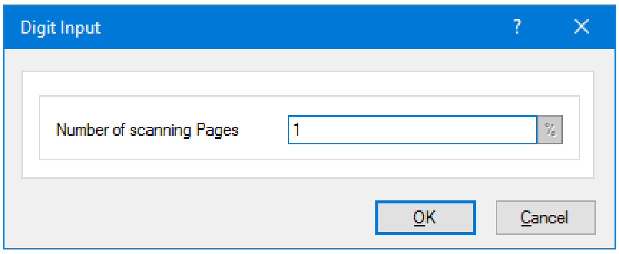
Insert Images
Delete Images
Mark Images
Modify Images
Group Images

3.6.2.2.8.1 Insert Images

In this command group the commands are combined which allows to insert images to a batch.

Note:
Inserting images is meaningful and allowed only if a single image is selected. The insertion occurs always *after* the image selected currently.

Icon	Command	Description	Hotkey	Restriction
	InsertEmptyImage	Inserts an empty image of the same size and color type after the actual one. In Multistream it inserts a group of images behind the last image of the currently selected group Image .		
	InsertImage FromCopy	Inserts a copy of the currently selected image after this image. In multistream it inserts inserts a copy of the whole image group.		

	InsertImage FromFilter	This button works if the selected image can be filtered with filter currently loaded in the base class. If so, a new filtered image will be inserted after the selected one.
	InsertImage FromFile	Opens a file selection dialog, loads the chosen file and insert the image (only the **first,** if there are more than one) and inserts it after the current image or image group.
	InsertImage FromFileN	Opens a file selection dialog, loads one or more chosen files and insert the images after the current image or image group.
	InsertImage FromScanner	Scanns an Image and inserts it after the selected image or image group
	InsertImage FromScanner2	Scanns a page (front & rear) and inserts it after the selected image or image group.
	InsertImage FromScannerN	Openes a diloag to specify the number of pages to scan..  Number of scanning Pages: 1 Then the number od scans needed will take place and the resulting images will be inserted after the image group currently selected.
	InsertImage FromScannerX	Scans all papers in the scanner and inserts the images after the image currently selected.

3.6.2.2.8.2 Mark Images

In this command group hold s the commands to mark images in a batch.

There are red, blue, green, magenta, orange and cyan as well as white marks

Images with red tag marks will be deleted when the is finalized. They won't be copied to the target directory.

Icon	Command	Description	Hotkey	Restriction
	ToggleRedMark	Sets or removes the red mark. Images with red marks will be rejected when finalizing the batch.		

ClearRedMark	Sets or removes the red mark from all images selected. Images with red marks will be rejected when finalizing the batch..
SetRedMark	Sets the red mark on all images selected. Images with red marks will be rejected when finalizing the batch..

Images with blue marks prevent a batch to be finalized.

ToggleBlueMark	Sets or removes the blue mark. Images with blue marks will prevent the finalization of the batch.
ClearBlueMark	Removes the blue mark from all selected images. Images with blue marks will prevent the finalization of the batch..
SetBlueMark	Sets the blue mark to all selected images. Images with blue marks will prevent the finalization of the batch.

Green, magenta, orange and cyan marks are intended to highlight special documents.

ToggleGreenMark	Sets or removes the green mark. Important images can be marked green.
EntferneGrüne Markierung	Removes the green mark from all images selected. Important images can be marked green.
SetzeGrüne Markierung	Sets the green mark to all images selected. Important images can be marked green.
Toggle MagentaMark	Sets or removes the Magenta mark. Important images can be marked Magenta.
Clear MagentaMark	Removes the magenta mark from all images selected. Important images can be marked Magenta.
Set MagentaMark	Sets the magenta mark to all images selected. Important images can be marked Magenta.
Toggle OrangeMark	Sets or removes the orange mark. Important images can be marked Orange
Clear OrangeMark	Removes the orange mark from all images selected. Important images can be marked orange.
Set OrangeMark	Sets the orange mark to all images selected. Important images can be marked orange.

	Toggle CyanMark	Sets or removes the cyan mark. Important images can be marked cyan.
	Clear CyanMark	Removes the cyan mark from all images selected. Important images can be marked cyan.
	Set CyanMark	Sets the cyan mark to all images selected. Important images can be marked cyan.

When the batch is displayed, images with white marks will be hidden always.

	ToggleWhiteMark	Sets or removes the white mark. Images with white marks can be hidden - regardless of other marks.
	ClearWhiteMark	Removes the white mark from all images selected Images with white marks can be hidden - regardless of other marks.
	SetWhiteMark	Sets the white mark for all images selected. Images with white marks can be hidden - regardless of other marks

3.6.2.2.8.3 Group Images

In this command group hold s the commands to re-arrange images to groups.

Icon	Command	Description	Hotkey	Restriction
	ImageConnect	Appends the actual file to the file before.		First image of a file
	ImageSplit	Closes the actual file before the selected image and starts a new file beginning with this image		First image of a group
	ImagePaper SwapSides	Swaps front and rear side image or image groups resp.		

3.6.2.2.8.4 Modify Images

These types of commands include these command groups:

Simple image operations, e.g. rotaion and deskew

Parameter commands, steps for which a subprofile or an image selection can be specified.

Undo and accept

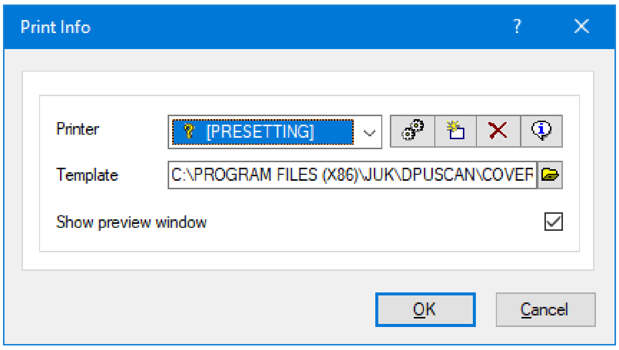
Please note all these commands will work in pause mode only.

In this command group, the commands are summarized, which require no further information

Icon	Command	Description	Hotkey	Restriction
↻ 90°	ImageRotationR90	Rotates the selected Image 90° clockwise		
↺ 90°	ImageRotationL90	Rotates the selected Image 90° counterclockwise		
180°	ImageRotation180	Rotates the selected Image by 180°		
	ImageDeskew	Deskews the image an removes the black borders		
	ImageInvert	Ivertes the image color		

In this command group the all commands are listed which uses a set of parameter or a subprofile. For example the call of a PlugIn for a certain image.

Icon	Command	Description	Hotkey	Restriction
1 ABC	BatesParam01	Stamps the image with the settings from the stored parameter set 1-16		
1	DeskewParam01	De-skews the image with the stored subprofile 1 to 16		
1	PlugInParam01	Uses the PlugIn to better an image with the stored subprofile 1 to 16, e.g. Gamma Correction or Color Noise Reduction.		
1	FilterParam01	Uses a Color Filter to better an image with the stored filter subprofile 1 to 16.		
i 1	PrintInfo01	Prints a cover page with the stores setting 1-16.		



Print Info

Printer: [PRESETTING]

Template: C:\PROGRAM FILES (X86)\JUK\DPUSCAN\COVEF

Show preview window ☑

As a **printer**, use a subprofile that points to a specific printer.

If you use the **PRESETTING** instead, the default printer of the operating system will be activated.

As a template, enter the template file in RTF format.

Before printing, a preview can be displayed.

	MarkParam01	Marks the images of the image group as defined in the settings 1 to 8.		
	ModifyParam01	Uses the Enhanced Modify process to better an image with the stored subprofile 1 to 16		
	TaskStart Param01	Starts a task as defined in the parameter set 1-16. This button can be used to start an often used task quickly.		
	UserDialog Param01	Displays the user defined dialog 1-16.		

In this command group, the commands are summarized to accept or undo the image operations.

Icon	Command	Description	Hotkey	Restriction
✓	ParamApply	Accepts the changes made to the selected image.		
↶	ParamUndo	Rejects the changes made to the selected image.		

3.6.2.2.8.5 Delete Images

Icon	Command	Description	Hotkey	Restriction

	ImageRemove	Removes the selected image	Del.	
	ImageRemove Group	Removes all images of the actual image group, i.e. the group the current image is belonging to.		
	ImageRemove Paper	Removes all images of the actual paper (front and rear side)		
	Image Remove Document	Removes the document currently selected.		There is a document structure

3.6.2.2.9 Selection

These groups collect all commands to set the selection to a specific image. These commands cannot be used during scan. When scanning the selection is set automatically to the just incomming image.

But on quality control in a manual post-processing task it is useful to simplify the navigation to interesting pictures. For example, always to the first page of a document. Therefore, there are a variety of commands that causes exactly this.

One can move the next image, the next file or the next folder. Also the a jump to the next marked image is possible. As long as all images are displayed, this is easy to handle. It gets somewhat more complicated, if the view is limited to the images of only one file or folder and the rest will be hidden.

Then one differentiates the behavior at the end of a file:
- staying at this file (Yellow arrows)
- skipping the file border (Blue arrows)

If the selection has reached the last image there are two options again:
- staying at the last image. (without repeat icon)
- skipping to the begin of the range (with repeat icon)

The combination of these options forms the following command groups:

Skip disallowed
- Range limited
- Range unlimited

Skip allowed
- Range limited
- Range unlimited

In addition, there are other rather global options to select images::

Icon	**Command**	**Description**	**Hotkey**	**Restriction**

GoToBatchStart	Go to the begin of the batch (to the topmost document)	
GoToFile	Go to the current file (to current document on second level)	
Select ActualSream	Selects all images of the current image group.	
Select AktualImage	(Re-) Selects the images of the current.. This command can be used if several images are selected.	
SelectAll	Selects the complete batch.	Ctrl-A
SelectToEnd	Selects all images down to the last one.	Ctrl-Shift-End
SelectToStart	Selects all images up to the first one	Ctrl-Shift-Pos1
GoToFirstImage	Moves to the first image of the batch	
Select FirstImage OfFile	Selects the first image o a file.	
Select FirstImage OfFolder	Selects the first image o a folder.	
GoToLastImage	Moves to the last image of the batch	
Select LastImage OfFolder	Goes to the last image in the current folder	
Select LastImage OfFile	Goes to the last image of the current file.	
SelectionSave	Notes all images that are currently selected.	In a macro only
Selection Restore	Re-selects all images again which were noted before.	In a macro only

3.6.2.2.9.1 Skip disallowed (yellow arrows)

If the view in the preview window or in the directory tree to a file or directory is restricted, these commands allow the navigation only within this visible file or folder.

Please use for the jump to to the next/previous file or folder the commands of the group skip allowed.

Within this group can be distinguished, how to process at the first or last image:

Innerhalb dieser Gruppe kann unterschieden werde, wie beim letzten bzw. ersten Bild verfahren werden soll:

- Range limited (remain staying)
- Range unlimited (wrap around)

If the view in the preview window or in the directory tree to a file or directory is restricted, then

- It is not possible to go to the next or previous hidden file. This is only possible with the appropriate commands or by clicking the mouse in the tree view.
- The movement ends at the latest, or the first image in the visible region. It will not automatically go back to the first, respectively, the last image.

Icon	Command	Description	Hotkey	Restriction
⇧	GoToPrevious Image Range	Moves to the previous image. Remains standing on the first picture of the visible region.		
⇩	GoToNext Image Range	Moves to the next image. Remains standing on the last picture of the visible region.		
⊗⇧	GoToPrevious RedMark Range	Moves to the previous image with a red mark. Remains standing on the first picture of the visible region.		
⇩⊗	GoToNext RedMark Range	Moves to the next image with a red mark. Remains standing on the last picture of the visible region.		
●⇧	GoToPrevious BlueMark Range	Moves to the previous image with a blue mark. Remains standing on the first picture of the visible region.		
⇩●	GoToNext BlueMark Range	Moves to the next image with a blue mark. Remains standing on the last picture of the visible region.		

🟢⇧	GoToPrevious GreenMark Range	Moves to the previous image with a green mark. Remains standing on the first picture of the visible region.
⇩🟢	GoToNext GreenMark Range	Moves to the next image with a green mark. Remains standing on the last picture of the visible region.
🟣⇧	GoToPrevious MagentaMark Range	Moves to the previous image with a magenta mark. Remains standing on the first picture of the visible region.
⇩🟣	GoToNext MagentaMark Range	Moves to the next image with a magenta mark. Remains standing on the last picture of the visible region.
🟠⇧	GoToNext OrangeMark Range	Moves to the previous image with a orange mark. Remains standing on the first orange picture of the visible region.
⇩🟠	GoToNext OrangeMark Range	Moves to the next image with a orange mark. Remains standing on the last orange picture of the visible regio
🔵⇧	GoToPrevious CyanMark Range	Moves to the previous image with a cyan mark. Remains standing on the first cyan picture of the visible region.
⇩🔵	GoToNext CyanMark Range	Moves to the next image with a cyan mark. Remains standing on the last cyan picture of the visible region.

The following commands are based on the packing of images into files and directories.

📂⇧	GoTo PreviousFolder Range	Go to the previous folder (to previous document on top level). Remains standing on the first picture of the visible region.
⇩📂	GoTo NextFolder Range	Go to the next folder (to next document on top level) Remains standing on the last picture of the visible region..
📄⇧	GoTo PreviousFile Range	Go to the previous file (to previous document on second level). Remains standing on the first picture of the visible region.
⇩📄	GoTo NextFile Range	Go to the next file (to next document on second level) Remains standing on the last picture of the visible region.

Moreover, there are commands that are not geared to the characteristics of the images in the batch. They are oriented to how many images can be displayed simultaneously in a certain part of the work area:

⇧	GoToPrevious Page Range	Moves one screen page backward that is the number of images visible in the **Scan/Edit window** Remains standing on the first picture of the visible region.
⇩	GoToNext Page Range	Moves one screen page forward that is the number of images visible in the **Scan/Edit window.** Remains standing on the last picture of the visible region.
⇧	GoToPrevious ScanPageRange	Moves one screen page backward that is the number of images visible in the **current window**. Remains standing on the first picture of the visible region.
⇩	GoToNext ScanPageRange	Moves one screen page forward that is the number of images visible in the **current window**. Remains standing on the last picture of the visible region.

If the view in the preview window or in the directory tree to a file or directory is restricted, then:

- Browsing downward after the last image of the visible file or folder it moves to the **next hidden** file or folder. Browsing upward after the first image of the visible file or folder it moves to the **previous hidden** file or folder. The the current file will be hidden and the next or previous file will be **shown**.
- The movement doesn't end at the latest, or the batch. Instead it will skip back to the begin or forward to end of the batch.

Icon	Command	Description	Hotkey	Restriction
⬆	GoToPrevious Image Endless	Moves to the previous image. At the begin it skips back to the last image of the batch.		
⬇	GoToNext Image Endless	Moves to the next image. At the end it skips back to the first image of the batch.		
●⬆	GoToPrevious BlueMark Endless	Moves to the previous blue mark. At the begin it skips back to the last blue mark of the batch.		
●⬇	GoToNext BlueMark Endless	Moves to the next blue mark. At the end it skips back to the first blue mark of the batch.		
●⬆	GoToPrevious GreenMark Endless	Moves to the previous green mark. At the begin it skips back to the last green mark of the batch.		

GoToNext GreenMark Endless	Moves to the next green mark. At the end it skips back to the first green mark of the batch.
GoToPrevious RedMark Endless	Moves to the previous red mark. At the begin it skips back to the last red mark of the batch.
GoToNext RedMark Endless	Moves to the next red mark. At the end it skips back to the first red mark of the batch.

The following commands are based on the packing of images into files and directories.

GoToPrevious Folder. Endless	Moves to the previous folder. At the begin it skips back to the last folder of the batch.
GoToNext Folder Endless	Moves to the next folder. At the end it skips back to the first folder of the batch.
GoToPrevious File Endless	Moves to the previous file. At the begin it skips back to the last file of the batch.
GoToNext File Endless	Moves to the next file. At the end it skips back to the first file of the batch.

Moreover, there are commands that are not geared to the characteristics of the images in the batch. They are oriented to how many images can be displayed simultaneously in a certain part of the work area:

GoToPrevious Page Endless	Moves one screen page backward that is the number of images visible in the **Scan/Edit window** At the begin it skips back to the last page of the batch.
GoToNext Page Endless	Moves to the next page.Moves one screen page forward that is the number of images visible in the **Scan/Edit window** At the end it skips back to the first page of the batch.
GoToPrevious ScanPage Endless	Moves one screen page backward that is the number of images visible in the **current window**. At the begin it skips back to the last page of the batch.
GoToNext ScanPage Endless	Moves one screen page forward that is the number of images visible in the **current window**. At the end it skips back to the first page of the batch.

3.6.2.2.9.2 Skip allowed (blue arrows)

If the view in the preview window or in the directory tree to a file or directory is restricted, these commands allow the navigation through the batch, regardless of hidden or shown regions.
Within this group can be distinguished, how to process at the first or last image:

- Range limited (remain staying)
- Range unlimited (wrap around)

If the view in the preview window or in the directory tree to a file or directory is restricted, then:

- Browsing downward after the last image of the visible file or folder it moves to the **next hidden** file or folder. Browsing upward after the first image of the visible file or folder it moves to the **previous hidden** file or folder. The the current file will be hidden and the next or previous file will be **shown**.
- This movement ends at the end or begin of the complete batch.

Icon	Command	Description	Hotkey	Restriction
⬆	GoToPrevious Image	Moves to the previous image. Remains standing on the first picture of the batch.		
⬇	GoToNext Image	Moves to the next image. Remains standing on the last picture of the batch.		
⊗⬆	GoToPrevious RedMark	Moves to the previous red mark. Remains standing on the first red mark of the batch.		
⬇⊗	GoToNext RedMark	Moves to the next red mark. Remains standing on the last red mark of the batch.		
🔵⬆	GoToPrevious BlueMark	Moves to the previous blue mark. Remains standing on the first blue mark of the batch.		
⬇🔵	GoToNext BlueMark	Moves to the next blue mark. Remains standing on the last blue mark of the batch.		
🟢⬆	GoToPrevious GreenMark	Moves to the previous green mark. Remains standing on the first green mark of the batch.		

GoToNext GreenMark	Moves to the next grenn mark. Remains standing on the last green mark of the batch.
GoToPrevious MagentaMark	Moves to the previous magenta mark. Remains standing on the first magenta mark of the batch.
GoToNext MagentaMark	Moves to the next grenn mark. Remains standing on the last magenta mark of the batch.
GoToPrevious OrangeMark	Moves to the previous orange mark. Remains standing on the first orange mark of the batch.
GoToNext OrangeMark	Moves to the next grenn mark. Remains standing on the last orange mark of the batch.
GoToPrevious CyanMark	Moves to the previous cyan mark. Remains standing on the first cyan mark of the batch.
GoToNext CyanMark	Moves to the next grenn mark. Remains standing on the last cyan mark of the batch.

The following commands are based on the packing of images into files and directories.

GoToPrevious Folder.	Moves to the previous folder. Remains standing on the first folder of the batch.
GoToNext Folder.	Moves to the next folder. Remains standing on the last folder of the batch.
GoToPrevious File	Moves to the previous file. Remains standing on the first file of the batch.
GoToNext File	Moves to the next file. Remains standing on the last picture of the file.

Moreover, there are commands that are not geared to the characteristics of the images in the batch. They are oriented to how many images can be displayed simultaneously in a certain part of the work area:

GoToPrevious Page	Moves one screen page backward that is the number of images visible in the **Scan/Edit window** Remains standing on the first page of the batch.

⬇	GoToNext Page	Moves one screen page forward that is the number of images visible in the **Scan/Edit window** Remains standing on the last page of the batch.		
⬆	GoToPrevious ScanPage RangeEndless	Moves one screen page backward that is the number of images visible in the **current window**. At the begin it skips back to the last page of the visible region.		
⬇	GoToNext ScanPage RangeEndless	Moves one screen page forward that is the number of images visible in the **current window**. At the end it skips back to the first page of the visible region.		

If the view in the preview window or in the directory tree to a file or directory is restricted, then

- It is not possible to go to the next or previous **hidden** file. This is only possible with the appropriate commands or by clicking the mouse in the tree view.
- The movement doesn't end at the latest, or the first image image in the visible region. Instead it will skip back to the begin or end.

Icon	**Command**	**Description**	**Hotkey**	**Restriction**
⬆	GoToPrevious Image RangeEndless	Moves to the previous image. At the begin it skips back to the last image of the visible region.		
⬇	GoToNext Image RangeEndless	Moves to the next image. At the end it skips back to the first image of the visible region..		
⊗⬆	GoToPrevious RedMark RangeEndless	Moves to the previous red mark. At the begin it skips back to the last red mark of the visible region.		
⊗⬇	GoToNext RedMark RangeEndless	Moves to the next red mark. At the end it skips back to the first red mark of the visible region..		
●⬆	GoToPrevious BlueMark RangeEndless	Moves to the previous blue mark. At the begin it skips back to the last blue mark of the visible region.		
●⬇	GoToNext BlueMark RangeEndless	Moves to the next blue mark. At the end it skips back to the first blue mark of the visible region..		

GoToPrevious MagentaMark RangeEndless	Moves to the previous magenta mark. At the begin it skips back to the last magenta mark of the visible region.
GoToNext MagentaMark RangeEndless	Moves to the next magenta mark. At the end it skips back to the first magenta mark of the visible region.
GoToPrevious OrangeMark RangeEndless	Moves to the previous orange mark. At the begin it skips back to the last orange mark of the visible region.
GoToNext OrangeMark RangeEndless	Moves to the next orange mark. At the end it skips back to the first orange mark of the visible region.
GoToPrevious CyanMark RangeEndless	Moves to the previous cyan mark. At the begin it skips back to the last cyan mark of the visible region.
GoToNext CyanMark RangeEndless	Moves to the next cyan mark. At the end it skips back to the first cyan mark of the visible region.

The following commands are based on the packing of images into files and directories.

GoToPrevious Folder RangeEndless	Moves to the previous folder. At the begin it skips back to the last folder of the visible region.
GoToNext Folder RangeEndless	Moves to the next folder. At the end it skips back to the first folder of the visible region..
GoToPrevious File RangeEndless	Moves to the previous File. At the begin it skips back to the last File of the visible region.
GoToNext File RangeEndless	Moves to the next file. At the end it skips back to the first file of the visible region..

Moreover, there are commands that are not geared to the characteristics of the images in the batch. They are oriented to how many images can be displayed simultaneously in a certain part of the work area:

GoToPrevious Page RangeEndless	Moves one screen page backward that is the number of images visible in the **Scan/Edit window.** At the begin it skips back to the last page of the visible region.
GoToNext Page RangeEndless	Moves one screen page forward that is the number of images visible in the **Scan/Edit window.** At the end it skips back to the first page of the visible region.

⇧	GoToPrevious ScanPage RangeEndless	Moves one screen page backward that is the number of images visible in the **current window**. At the begin it skips back to the last page of the visible region.
⇩	GoToNext ScanPage RangeEndless	Moves one screen page forward that is the number of images visible in the **current window**. At the end it skips back to the first page of the visible region.

3.6.2.2.10 Further Commands

The following command groups belong to this kind of commands:

Set variable
Device control
User events

3.6.2.2.10.1 Set Variable

In this command group the commands are combined used to set variable for a single image, all images of image group or all image groups of one sheet of paper.

Icon	Command	Description	Hotkey	Restriction
UDD f(x)	SetVariable	Sets the Return-Variable for the selected image.		Macro only
UDD f(x)	SetVariable ImageGroup	Sets the Return-Variable for all images of the image group of the selected image. Please consider the rules for variable assigment.		Macro only
UDD f(x)	SetVariable ImagePaper	Sets the Return-Variable for all images of front and rear side of the selected image. Please consider the rules for variable assigment.		Macro only

3.6.2.2.10.2 Device Control

This command group contains at the moment only the function to set the text for the endorser:

Icon	Command	Description	Hotkey	Restriction
ABC	SetEndorser	Stopps the scanner to set a new text for the built-in printer head (endorser/imprinter).		

3.6.2.2.10.3 User Events

In this command group the commands are combined which triggers a userdefined action:

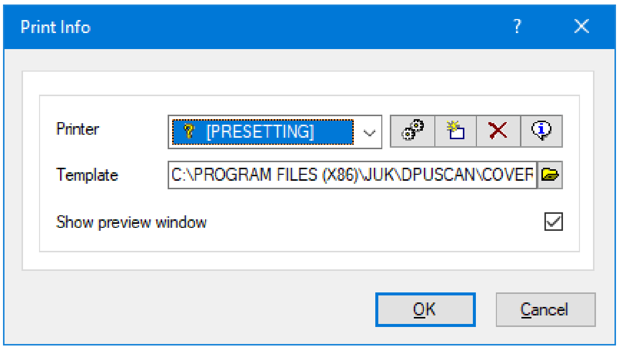
User events are also recorded during the scan process and remain for one side and are automatically reset on the next side.

Icon	Command	Description	Hotkey	Restriction
	UserEvent01 to UserEvent32	Triggers the user defined event 1 to 32. It also sets the value of the variables %(S.USEREVENT1) to %(S.USEREVENT32) to 1 . These will be reset to 0 on the next side. To hold the event until the next the variables %(S.USEREVENTLAST1) to %(S.USEREVENTLAST32) will be set to 1 too. Those will not be reset with the next side. The index-less Variables %(S.USEREVENT) and %(S.USEREVENTLAST) will be set to 1 on *each* user event		

3.6.2.2.10.4 Print

In this command group the commands are combined which triggers a print. Please distinguish between the printing of pictures and texts:

Icon	Command	Description	Hotkey	Restriction
	PrintInfo01	Prints a cover page with the stores setting 1-16.		



Printer: [PRESETTING]

Template: C:\PROGRAM FILES (X86)\JUK\DPUSCAN\COVEF

Show preview window ☑

As a **printer**, use a subprofile that points to a specific printer.

If you use the **PRESETTING** instead, the default printer of the operating system will be activated.

As a template, enter the template file in RTF format.

Before printing, a preview can be displayed.

	Print	Prints the actual image using the currently loaded set of parameters.

3.6.2.2.10.5 Placeholder and outdated Commands

In this command group the commands from older versions of program are listet.

und mittlerweile abgelöst wurden.

Icon	Command	Description	Hotkey	Restriction
Task	Defined Profiles	The functionality has changed since version 6 From version 6: Opens the Profiles page in the Ribbon. There a profile can be selected. Older versions: Opens the drop-down list for selecting the task profile.	Return	
	Wizard	The functionality is no longer available in version 6. Older versions: Opens a step-by-step wizard to create a profile.		
M	Macro Configuration	This is a deny-only-icon: It does not work, but it can be used in the User Rights and Grants settings. to deny the corresponding action for groups of users.		Deny only

3.6.3 XINO-Buttons

If a XINO scanner is connected to the computer, so here can be defined how the program should react when a certain button is pressed on the scanner
To make this buttons visible at the scanner's display please refer to the current users manual of the scanner.

▶	When this button is pressed on the XINO scanner, the currently selected task is started in the program. Note: If the task contains the Task Pause command at the beginning, the start button on the scanner must be pressed again.
⏸	If this button is pressed on the XINO scanner, the program interrupts the task and goes into pause mode (without displaying another message).
■	This key stops the scanner and carries out all further steps in the scan loop (detection, event rules). If there is a post processing at the end (e.g., exporting the stack to an indexing station), these steps will be performed too.

Afterwards the task is finished and the program goes into the Ready state

This key stops the scanner and triggers the status message "Emergency Break". The program displays this and waits for further instructions (Continue, Pause, Close)

...

These buttons can each be linked to a macro of the program. Thus, the subsequent process can be controlled, e.g. a rotation can be switched on or it can be toggled between single and multiple image files.

Hint: Usually in macros while scanning it's not possible to set %-Codes. Instead, fire a user event and evaluate it in the event rules.

See also

Help Overview.
Applikation Layout
Toolbars
Actions

3.7 Scanner

After selecting a scanner you can specify the settings for your scanner model here:

The scanner setup dialog shows these property pages:

- General, color types, resolution,...
- Paper, feeder, size, ...
- Misc, buffers, workflow,...
- Endorser, printer unit in the scanner
  - Endorser Control, handling after a break,...
  - Endorser Formats, alphabetical list of placeholders
- For TWAIN-Scanner
  - TWAIN, User Interface and data transfer.
  - TWAIN Specials, direct controlling by the program
- For ISIS-Scanner
  - ISIS, User Interface and data transfer.
- Other Scanner
- Virtual Scanners
- Index
- Image Processor

Common control elements:

Send Parameter Sends all settings one after the other to the device. This way and using a log function, failures may be identified.

Info Opens a dialog showing the driver's version.

Test Scan Sends the current values to the device and performs a scan. This can be used to optimize the settings. In multistream mode, each test scan returns one color format.

Note: With some scanners, you have to additionally press the start key on the scanner for running a test scan.

OK The dialog box is closed. All settings are saved.

Cancel The dialog box is closed. The settings will not be saved.

Help Opens this help screen.

See also General, Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.1 General

This page shows the most commonly used settings for a scanner. Depending on the scanner used, parts of these settings can be hidden.

Resolution Chooses the resolution

The values given will depend on the capabilities of your scanner.

Panel With panel the resolution can be set at the scanner.

200 A smaller resolution like **200** dpi, reduces the amount of data to store.

300 A higher resolution like **300** dpi yields better results for Barcode or OCR-search.

Scan Mode

One Side It will be scanned in simplex mode.

Two Sides Front and rear side will be scanned.

Brightness Sats the brightness.

Front Sets the value for the front side camera.

Rear Sets the value for the rear side camera if available and separately configurable.

Threshold Sets the threshold for the black/white dsitinction.

Front Sets the value for the front side camera.

Rear Sets the value for the rear side camera.

Kontrast Sets the contrast.

Front Sets the value for the front side camera.

Rear Sets the value for the rear side camera.

Images to fetch

Front Black+White
Rear Black+White States 1 Bit- black+white images to be fetched from the scanner.

Front Gray
Rear Gray States 8 Bit gray scale images to be fetched from the scanner.

Front Color
Rear Color States 24 Bit color images to be fetched from the scanner.

F: Auto / Color Detect.
R: Auto / Color Detect. Specifies that the scanner decides what type of image it delivers. This is e.g. useful if among many ordinary black and white images are only a few colored images and the scanner is equipped with a color recognition.

Set Parameters Sets the current settings to the machine.

Info Opens a message box holding verion information about the driver file.

Select Scanner This function known from earlier versions of the program was replaced by be scanner selection dialog.

See also Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.2 Paper

This page shows the paper settings. More precisely, here the size of the area to scan is specified, that is how many pixels wide and long the image should be. Is a **smaller** piece of paper is scanned, so the image trimming is achieved with the **deskew** function. If a **longer** piece of paper is scanned, depending on the scanner, it is simply cut off or an error is thrown. If a **wider** sheet is scanned, the left and right parts of the image will miss. Nevertheless, the maximum settings should not just always be used. The scanner would lose time trying to scan edges that need to be cut off again later.

Source Select the feeding here. The offered values depends on your scanner model.

- **Flatbed** to feed manually
- **Feeder 1, Feeder 2, ...** To use the automatic document feeder ”ADF”
- **Panel** To select the feeder from the scanner’s panel

Centered If checked the image is taken from the feeders center; value X-Pos sets the difference (+/-) between center points then.

If centered feeding is not supported by the scanner, the check box "Centered" is not displayed. In this case or if "Centered" is not checked, X-Pos gives the distance from left border.

Size

Select a standard format (**A4, A5, Letter, Legal**,...) from the list or a free format (User #).

If a free format is selected: Enter values for position and size of the **scan area** in User defined size ... edit fields!

Landscape

The formats in size list are portrait formats. Check this box if paper is feeded in landscape format.

The option changes with and length, but it does **not** rotate the image automatically.

Scanning on the "the short way" will save time. But the saving will be reduced if the images must be rotated later.

Overscan

Select, which kind of scan area size enlargement should be used. Overscan is useful, if a deskew or cropping process follows.

Automatic all available dimensions are enlarged.

Horizontal only the width is enlarged.

Vertical only the height is enlarged.

Off no enlargement

Size Tolerance

If checked scanning of a too long and/or too short document will be treated as a scan error.

+ plus: Scanning a document longer than the length of the selected paper size plus the given value here will be treated as a scan error. With this some kind of double-feeds can be detected.

- minus: Scanning a document shorter than the length of the selected paper size minus the given value here will be treated as a scan error. With this destroyed papers can be detected.

User defined size (in 0.1 mm)

Changeable, if a free format (User #).is selected: The used measurement unit can be set in the base class.

Since the settings for these values are highly dependent on the scanner; We recommend urgently to perform a test scan and then to adjust the values until a satisfactory outcome is achieved.

X-Pos X-Pos: Horizontal position of the scan area. If Centered is selected (see above), the value gives the distance of the center points (+/-); otherwise it gives the distance from the left border

Y-Pos Position of the top border of the scan area.

Required minimum for Width and Length is 5mm (0.197Inch)!

Length Length of the paper.

Width Width of the paper.

Rotation Select an angle (clockwise for front image) to rotate scanned images:

Front

Rear

For **regular** documents, bound on the left side, front an rear angle will be **equal**.

90º turns a quarter to the right

180° turns a half

270° turn a quarter to the left

See also General, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.3 Miscellaneous

This page collects miscellaneous settings which are accessible for many scanners.

Image Processing

Scan Mode Selects how gray shades from the original document shall be processed.

Text mode is Standard

Photo mode simulates gray shades in the resulting bitonal image by dithering The size of the scanned image file will be significantly larger.

Panel means that settings will not be transmitted to the scanner

Inverted Check this box to get inverted images (black - white exchanged)

Cropping Check this box to cut off the (black!) borders from scanned images. Depending on the used scanner, the horizontal, the vertical, or all borders are removed.

Deskew Check this box to get de-skewed images. This function will work if the scanner produces images with black background only!

Flow Control

Start key Check this box if the start key at the scanner must be pressed before scanning each paper.

Display start button If this box is checked and flatbed is selected as scanning source, the program shows a dialog box with a [Start] button before every scan is done. This allows preparation of papers if Start Key function is not available.

Note: This function is available only, if the Image Processor is available and enabled.

Queuing Check this box to allow pre-scanning images to a buffer (batch mode). (Recommendable because this increases efficiency)

Flush Queuing Buffer on Error In case of an error, which does not automatically lead to a direct stop of the scanner, further pages may be in the queuing buffer after the faulty page is scanned. This complicates restarting the scan job in an error case. Therefore, it may make sense to delete these pages from the buffer. With an activated Imprinter, such pages have possibly already been printed.

Map Error Codes

Map Error Codes: Some of the scanner error codes may cause in special actions of the application. Sometimes, these actions are not usefull, or an other action should be processed then. Use the following options to define, as what code the scanner error code reaches the application:

No Paper If scanner sends ”No Paper”, the application is notified as

Report (default): code ”No Paper” is send to the application.

Ignore: the application gets not notified - scanning runs continuously.

End of Job: code ”End of Job” is send to the application.

Scanning done: code ”Scanning done” is send to the application.

Scanning done If scanner sends ”Scanning done”, the application is notified as

Report (default): code ”Scanning done” is send to the application.

Ignore: the application gets not notified - scanning runs continuously.

End of Job: code ”End of Job” is send to the application.

No Paper: code ”No Paper” is send to the application.

End of Job If scanner sends ”End of Job”, the application is notified as

Report (default): code ”End of Job” is send to the application.

Ignore: the application gets not notified - scanning runs continuously.

No Paper: code ”No Paper” is send to the application.

Scanning done: code ”Scanning done” is send to the application.

See also General, Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.4 Endorser

Some scanner models have a built-in endorser that prints a text on every scanned document. This endorser sometimes also called **Imprinter**. Some scanners have an **Imprinter** (which prints before scanning) and an **Endorser** (which prints after scanning) . Unfortunately, the denominations are not uniform. Therefore, we use both denominations as synonyms. If an endorser exists, you can activate it by clicking the corresponding box.

Enable

Check this box if the specified text to be printed on each document.

String

This text will be printed on every page scanned. Can can consist of fixed text, program variables (%-Codes) and other placeholders, which depends on the scanner model.

The program variables will be evaluated only once at the start of the batch. So it's *not possible* to calculate and change the text for each image scanned.

Please also regard scanner's placeholdes starting with a percent sign will need a **double percent sign** here, e.g:

```
Scan_%(S.STATIONNAME)_%%u
```

Here the the text to print consist of a fix text at the begin, the program variable for the workstation's name and a special placeholder for a consecutive counter.

Counter

Start value for the counter, if there is counter placeholder used. That may be a fix value or a %-Code. It may be also an empty string, but there are scanners which use this as a special start value, see the option *don't set counter* below.

Remark: The current value of the endorser counter can be queried by the program variable %(S.ENDORSERCOUNT).

Multiple counters: If the scanner controls more than one counter, then starting values for those

counters can be given in field Counter. Starting values should be separated by semicolon and given in this order:

Sheet counter ; Level1 counter ; Level2 counter ; Level3 counter.

%(S.ENDORSERCOUNT) is always the the sheet counter, the other values can be determined from the image address.

don't set counter

To prevent the initializing of the endorser counter explicitly this option can be turned on. This is useful to prevent scanners to interpret an empty value at *Counter* as a special command.

Y-Offset

Distance of the starting position of the print from the begin of the paper.

Orientation

Orientation of the text to print. If the selection here is restricted to plain numbers please refer to the scanners manual or the endorser overview.

Font

Selects the font for the text to print. If the selection here is restricted to plain numbers please refer to the scanners manual or the endorser overview.

Print onto...

If the scanner has a patchcode recognition, printing can be restricted to certain sheets. E.g. It can be specified that only the first sheet after the patchcode is printed or that patchcode sheets are generally not printed in order to reuse them.

Please note: The patchcode search in the program has no effect on this option.

...Page with Patch

Set this option if you want to print patchcode sheets. If you switch this option off, these sheets will not be printed on and can therefore be used several times.

...Page after Patch

Set this option if you want to print the first sheet after a patchcode sheet. If you also switch off the "All other sheets" option, only the first sheet after the patch is endorsed, the subsequent sheets remain unchanged.

...all other pages

Set this option if you want to print on sheets that neither have a patchcode themselves nor are directly behind a patchcode sheet.

If you enable this option off and also disable the option "Page after patch", you can ensure that as many papers as possible remain unprinted.

Patch controlled print

Activate this option if you want to suppress printing between two patchcodes, e.g. to protect valuable originals.

...Off

Select here the patch code from which endorser printing should be switched off for the subsequent papers.

...On

Select here the patch code from which endorser printing should be turned on for the subsequent papers.

Enhanced Handling

If checked the enhanced endorser/imprinter control is activated. See the setup dialog for the further description

Setup

Opens the setup dialog for the enhanced handling.

See also General, Paper, Miscellaneous, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.4.1 Endorser Control

Here can be specified as the endorser counter should behave in certain actions:

Printer / Counter

During program run - depending on the situation - a setup dialog can be displayed where the user can change the value (pure number) of the counter manually.

The display can be triggered in the following events:

Batch Startt

before the first Document of a batch will be scanned.

on Rescan

before a Rescan.

on Insertscan

before an Insertscan.

after Pause

before a paused Job (for Rescan(s), Insertscsan(s), etc.) will start again.

after Scan Error

before a Job will start again that was stopped by a scanner error

For the situations above it can be stated how the Endorser shall react:

Continue Printing (default)

Counter value is not changed by application; scanned images get printed.

Restore Counter

Counter value is set by application to the value the document was printed before; scanned images get printed.

Disable Printer

printer is switched off while Rescanning.

The checkbox **Show Counter Setup** enables this for corresponding situation

Set String and Counter with every scan (slow scanning)

Normally, the text to be printed is sent only at the start of a job to the scanner.During scanning, the text remains unchanged, only the counter continues.

If selected here, the text before each scan is evaluated by the user program (%-Codes will be replaced) and sent to the scanner.

This function is only available when the image-buffering is not active (setting page options).

Throughput is thus low!.

See also Endorser, Endorser Formats and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.4.2 Endorser Formats

The operation of the endorser is highly dependent on the scanner. Some scanner will print onto the front side, other will print onto papers back side, and some have two printers to print on both sides. Most will print after scanning, but sometimes the printer is mounted before scanners camera: the printed text is included in scanned document. Also, the amount of extended functions (Counter, Time/Date) differs between scanner types.

The text to be printed can also contain placeholders. Valid placeholders will be replaced at start of the scanning by the application (e.g. by the content of an environment variable) or by the scanner (e.g. by the consecutive number of the scan).

Valid placeholders of the program are the %-Codes.

For valid placeholders of the scanner please refer to the scanners's manual or - for some models - this list.

The placeholders are successively replaced first by the program and then by the scanner.

Please note that in the preceding replacing by the application may delete, modify or add control characters, which are provided for the scanner.

There are various strategies to circumvent this problem. This will depend on how the scanner is driven.

Endorser for Scanners controlled via TWAIN.

Endorser for Scanners controlled via ISIS.

Endorser for Scanners controlled directly.

See also alphabetical list of scanners and Scanner Configuration.

3.7.4.2.1 Endorser for Scanners controlled via TWAIN

For scanners that are controlled via the TWAIN interface, the setup dialog displays two additional tabs: TWAIN and TWAIN Specials. The TWAIN Specials page, you can specify which parts of the settings can be set via the scanner's TWAIN dialog and which may additionally be placed on the other side of the scanner dialog. If the scanner has an endorser, allows the control and set to the TWAIN Specials page of the corresponding hook, so an additional page Endorser for the text to be printed is displayed.

The text to be printed can contain placeholders. These placeholders are thereby successively replaced by the program and then by the scanner.

Please note that in the preceding replacing by the application may delete, modify or add control characters, which are provided for the scanner.

A common problem thereby forms the use of the % character. For the workaround, specify this character ***twice***.

Example:
With DpuScan an Inotec scanner shall print a text, consisting of batch name and a five digit consecutive number of the page. For this number the scanner's counter %05u shall be used, the batch name comes from DpuScan, in form of the variable %O.

So enter on the page **Endorser** this text: "**`%O_%%05u`**". In the begin DpuScan replaces the %O with the batch name, say "INV045", and then the double percent with a single one.

The text sent to the scanner now reads "**`INV045_%05u`**". In this text the scanner now replaces its own part, say for the seventh page **`"INV045_00007".`** Finally, this text will be printed.

Recommended approach:
First, in the TWAIN dialog, generate the part of a text to be interpreted by the scanner. This is especially meaningful if the scanner in its own settings dialog has no input field for text and instead offers only buttons or lists with counter formats.
Then close the TWAIN dialog and, on the Endorser tab, add the text with the required DpuScan percent codes.

Also note that the maximum length of the text can be exceeded by the replacements.

List:
For the following scanner models the placeholders are listed here:

⊟ Inotec M06 per TWAIN-Interface

Valid for InoTec Scamax 402, 403, 412, 413 also 422, 423, 433 and 801, 811, 821

%u	Counter without leading zeros
%0*n*u	n-digit counter with leading zeros (n = [2..10])

%[Lson]	Landscape on
%[Lsoff]	Landscape off
%[Rot]	Text rotation 180°

%[Cd00]	0 character distance in pixel (default)
%[Cd01]	1 character distance in pixel
%[Cd02]	2 character distance in pixel
%[Cd03]	3 character distance in pixel
%[Cd04]	4 character distance in pixel
%[Cd05]	5 character distance in pixel
%[Cd06]	6 character distance in pixel
%[Cd07]	7 character distance in pixel
%[Cd08]	8 character distance in pixel
%[Cd09]	9 character distance in pixel
%[Cd10]	10 character distance in pixel

%[Pt1]	1 Pixel wide (default)
%[Pt2]	2 Pixel wide
%[Pt3]	3 Pixel wide
%[Pt4]	4 Pixel wide
%[Pt5]	5 Pixel wide

%[Seqid]	sequential ID
%[Seqid0]	sequential ID with leading zeros

%[DD.MM.CCYY]	Date DD.MM.YYYY
%[DD.MM.YY]	Date DD.MM.YY
%[MM/DD/YY]	Date MM/DD/YYYY

%[hh:mm]	Time HH:MM
%[hh:mm:ss]	Time HH:MM:SS
%[hh:mm AM]	Time HH:MM AM
%[hh:mm:ss AM]	Time HH:MM:SS AM
%[DOY]	Day des Jahres DDD
%[CC]	Jahrhundert
%[YY]	Year YY
%[DD]	Day DD
%[MM]	Month MM
%[Month]	Name of the month

%[Cd>]	Calculation start
%[<Cd]	Calculation end

%[BC25>]	Barcode 2/5 Start
%[<BC25]	Barcode 2/5 Stop

%[M10/3]	Modulo10/3 character
%[M7]	Modulo10/3 character

⊟ Kodak i1840 and i1860 under TWAIN

Valid for Kodak i1840 and i1860

The maximum lengt of the text to print is 40 characters.
These placeholders are available:

%0*n*u	*n*-digit counter with leading zeros [*n* = 1..9]
%*n*u	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<*n*u	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%0*n*u	*n*-stelliger Zähler mit führenden Nullen [*n* = 1..9]
%*n*u	*n*-stelliger Zähler mit führenden Leerzeichen [*n* = 1..9]
%u	Zähler, entspricht %<9u
%<*n*u	Zähler, linksbündig (*n* digit) [*n* = 1..9]

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY
%ijD	other date formats with : [i = 1..3], [*j* = 0..4], where i indicates the type and j the punctuation character

i=1	MMDDYYYY month 2 digits, day 2 digits, year 4 digits (j see below)
i=2	DDMMYYYY day 2 digits, month 2 digits, year 4 digits (j see below)
i=3	YYYYMMDD year 4 digits, month 2 digits, day 2 digits (j see below)
i=4	DDD day of the year 3 digits (j **not** given)
i=4	YYYYDDD year 4 digits, day of the year (j see below)

j=0	no punctuation
j=1	slash
j=2	minus
j=3	dot
j=4	space

If two placeholders of the same type are used, the last set formatting applies to both.

⊟ Kodak i4200 and i4600 under TWAIN

Valid for Kodak i4200 and i4600

The maximum length of the text to print is 40 characters.
These placeholders are available:

%0*n*u	*n*-digit counter with leading zeros [*n* = 1..9]
%*n*u	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<*n*u	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%0*n*u	*n*-stelliger Zähler mit führenden Nullen [*n* = 1..9]
%*n*u	*n*-stelliger Zähler mit führenden Leerzeichen [*n* = 1..9]
%u	Zähler, entspricht %<9u
%<*n*u	Zähler, linksbündig (*n* digit) [*n* = 1..9]

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY

%ijD	other date formats with : [i = 1..3], [*j* = 0..4], where i indicates the type and j the punctuation character

i=1	MMDDYYYY month 2 digits, day 2 digits, year 4 digits (j see below)
i=2	DDMMYYYY day 2 digits, month 2 digits, year 4 digits (j see below)
i=3	YYYYMMDD year 4 digits, month 2 digits, day 2 digits (j see below)
i=4	DDD day of the year 3 digits (j **not** given)
i=4	YYYYDDD year 4 digits, day of the year (j see below)

j=0	no punctuation
j=1	slash
j=2	minus
j=3	dot
j=4	space

If two placeholders of the same type are used, the last set formatting applies to both.

⊟ Kodak i5000 via TWAIN

Valid for Kodak i5200, i5250, i5600, i5650, i5650S, i5800, i5850, i5850S

The maximum lengt of the text to print is 40 characters.
These placeholders are available:

%0*n*u	*n*-stelliger Zähler mit führenden Nullen [*n* = 1..9]
%*n*u	*n*-stelliger Zähler mit führenden Leerzeichen [*n* = 1..9]
%u	Zähler, entspricht %<9u
%<*n*u	Zähler, linksbündig (*n* digit) [*n* = 1..9]

%0T	Uhrzeit HH:MM
%0D	Datum DD.MM.YYYY
%2D	Datum MM/DD/YYYY
%ijD	other date formats with : [i = 1..3], [*j* = 0..4], where i indicates the type and j the punctuation character i

i=1	MMDDYYYY month 2 digits, day 2 digits, year 4 digits (j see below)
i=2	DDMMYYYY day 2 digits, month 2 digits, year 4 digits (j see below)
i=3	YYYYMMDD year 4 digits, month 2 digits, day 2 digits (j see below)
i=4	DDD day of the year 3 digits (j **not** given)
i=4	YYYYDDD year 4 digits, day of the year (j see below)
i=5	DDMMYY day 2 digits, month 2 digits, year 2stellig (j see below)
i=6	MMDDYY month 2 digits, day 2 digits, year 2stellig (j see below)
i=7	YYDDD year 2 digits, day of the year (j see below)
i=8	YYMMDD year 2 digits , month 2 digits, day 2 digits (j see below)

j=0	no punctuation
j=1	slash
j=2	minus
j=3	dot
j=4	space

If two placeholders of the same type are used, the last set formatting applies to both.

Older models:

Kodak i800 Models via TWAIN Interface

Valid for older i800

%1L	Level_1 Text follows
%2L	Level_2 Text follows
%3L	Level_3 Text follows

The last %*n*L in string selects for which "document level" the described printing will occur.
If %nL isn't specified the printing will occur for all document levels (always)

%0nu	*n*-digit counter with leading zeros [*n* = 1..9]
%nu	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<nu	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%6B	ImageAddress Feld A
%7B	ImageAddress Feld B
%8B	ImageAddress Feld C
%9B	ImageAddress Feld D

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY
%ijD	Several date formats with: [*i* = 1..3], [*j* = 0..4]

%0S	Style/Font: large comic
%1S	Style/Font: large cinema
%4S	Style/Font: small comic
%5S	Style/Font: small cinema

If there is no %nS in the string, the font selected in TWAIN UserInterface will be used.

Maximum number of printable characters: 40

Kodak 3520, 4500 und i200 viaTWAIN Interface

Valid for older Kodak 3520 and other

%0nu	*n*-digit counter with leading zeros [*n* = 1..9]
%nu	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<nu	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY
%*ij*D	Several date formats with: [*i* = 1..3], [*j* = 0..4]

%0S	Style/Font: large comic
%1S	Style/Font: large cinema
%4S	Style/Font: small comic
%5S	Style/Font: small cinema

If there is no %nS in the string, the font selected in TWAIN UserInterface will be used.
Beside the given codes, and blanks, only one unique text element can be processed!

Maximum number of printable characters: 40

⊟ Kodak 5000, 7000 and older 500, 500A Models via TWAIN Interface

%0P	Front side text follows
%1P	Rear side text follows
%1L	Level_1 Text follows
%2L	Level_2 Text follows
%3L	Level_3 Text follows
	The last %*n*L code on the text defines the current document level. If there is no %*n*L in the text, the text is used for all levels
%u	Counter (9 characters)
%nu	Counter (9 characters)
%0A	Complete IA, no delimiter
%1A	Complete IA, all delimiter
%2A	Complete IA, fixed delimiter
%0T	Time HH:MM
%D	Date
%y	Day number (Day of the year)
%1B	Fixed Field IA
%2B	Index Field IA
%3B	Current Level IA
%M	Delimiter
%0S	Style/Font: large comic
%1S	Style/Font: large cinema
%2S	Style/Font: large comic, 180° gedreht
%3S	Style/Font: large cinema, 180° gedreht
%4S	Style/Font: small comic
%5S	Style/Font: small cinema
%6S	Style/Font: small comic, 180° gedreht
%7S	Style/Font: small cinema, 180° gedreht
	If no %*n*S is given in the text the Font set in the TWAIN User Interface font is used.

Maximum number of characters is 40.

See also scanner setup dialog.

3.7.4.2.2 Endorser for Scanners controlled via ISIS

For scanners that are controlled via the ISIS interface, the setup dialog displays an additional tab ISIS. On this propertypage a dialog Settings can be opened to control the setup of the endorser by the application. So if the scanner has an endorser and allows the setup by the application - after a restart of the application - an additional tab Endorser for the text to be printed is shown.

The text to be printed can contain placeholders. These placeholders are thereby successively replaced by the program and at then by the scanner.

Please note that in the preceding replacing by the application may delete, modify or add control characters, which are provided for the scanner.

To work around this problem, there is the possibility ***to mask*** these characters in the settings. For example, the percent sign % commonly used in DpuScan can be replaced by another character when the text is transferred to the scanner. Conversely, a control character, needed by the scanner, can also be masked when the text is displayed on the Endorser tab.

Recommended approach:
First, in the **ISIS** dialog, generate the part of a text to be interpreted by the scanner. This is especially meaningful if the scanner in its own settings dialog has no input field for text and instead offers only buttons or lists with counter formats.
Then close the **ISIS** dialog and, on the **Endorser** tab, add the text with the required DpuScan percent codes.

Also note that the maximum length of the text can be exceeded by the replacements.

See also scanner setup dialog.

3.7.4.2.3 Endorser for Scanners controlled directly

For scanners that are controlled directly and have an endorser, the setup dialog shows the page Endorser for the text to be printed.

The text to be printed can contain placeholders. These placeholders are thereby successively replaced by various application modules and at least by the scanner.

Please note that in the preceding replacing by the application may delete, modify or add control characters, which are provided for the scanner.

A common problem thereby forms the use of the % character. For the workaround, specify this character ***twice***.

Example:
With DpuScan a XINO scanner shall print a text, consisting of profile name and a four digit consecutive number of the page. For this number the scanner's counter %04u shall be used. The profile name comes from DpuScan, in form of the variable %$A.

So enter on the page **Endorser** this text: "**`%$A_%%04u`**". In the begin DpuScan replaces the %$A with the profile name, say "DN", and then the double percent with a single one.

The text sent to the scanner now reads "**`DN_%04u`**". In this text the scanner now replaces its own part, say for the seventh page **`"DN_0007".`** Finally, this text will be printed.

Also note that the maximum length of the text can be exceeded by the replacements.

List:
For the following scanner models the placeholders are listed here:

Current models:

⊟ XINOscan S700

%*n*u	n-digit counter without leading zeros, n=1..10
%0*n*u	n-digit counter with leading zeros, n=1..10
%0&	Toggle between endorser and imprinter
%1&	Set Endorser (after the Scan, rear side)

%2&	Set Imprinter (before the Scan, front side)
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%2T	Time HH:MM a.m./p.m.
%3T	Time HH:MM:SS a.m./p.m.
%0N	Username
%1N	Scanner name
%0*n*i %*n*i	Level 1 Index-Counter print (n-digits, with/without leading zeros)
%0*n*j %*n*j	Level 2 Index-Counter print (n-digits, with/without leading zeros)
%0*n*k %*n*k	Level 3 Index-Counter print (n-digits, with/without leading zeros)
%0*n*l %*n*l	Level 4 Index-Counter print (n-digits, with/without leading zeros)
%I	Print all active Index counter, see Index for formatting
%*nn*M	Set Check sum mode (nn = [00..99] check sum type) **0**: 231-weights, modulo 11, 10 & 11 = 0, see example. **1:** Code 39-reference numbers with Modulo 43, see example.
%S	Begin of range for check sum calculating
%E	End of range for check sum calculating
%P	prints check sum
	The control codes **%*nn*M**, **%S**, **%E**, **%P** must occur in this *order* in the text.
%*n*W	Character width: **0**: Default width **1:** Double Width
%*n*F	Type of font **0**: Standard font **1:** Loadable Font An additional font can be edited and loaded using a special font editor for this device. Please contact the vendor for this tool.
%*n*B	Reduce the character's width: **0:** print all 12 columns **1:** print only the first 10 columns **2:** print only the first 8 columns **3:** print only the first 6 columns For a special small font this increases the number of possible characters to print. The font must be loaded and activated with **%*1*F**

%*n*X **Character spacing**

0: Regular type

1: Spaced type

%R Text printing rotated 180°

Over all the length of the text may not exceed 200 characters.

⊟ Microform S415, S455, S615 and S655

%*n*u maximum *n* digit counter without leading nulls or blanks (*n* = [1..8])
%0*n*u *n* digit counter with leading nulls (*n* = [1..8])

%0S Style: Normal (Inverted OFF)
%1S Style: Inverse ON

%0D Date DD.MM.YYYY
%1D Date DD.MM.YY
%2D Date MM/DD/YYYY
%3D Date MM/DD/YY

%0T Time HH:MM
%1T Time HH:MM:SS

%I Formatierter Index-String
Warning: The %*n*I codes are available only when the scanner is equipped with an additional patch code reader! The formatting of the index strings can be determined on the setting page "Index"..

%0B Barcode OFF (normal text ON)
%1B Barcode Code 2/5 Interleaved
%2B Barcode Code 39

%! "<ESC>!" Sequenz - see Endorser manual
%* "<ESC>*" Sequenz - see Endorser manual

%& Only, if the scanner is equipped with **two endorsers / imprinters**: Switches text definition between the primary and secondary printer back and forth.
Sample: The text "AAAAAAA%&BBBBBBBB" will be printed as: "AAAAAAA" with the primary imprinter and "BBBBBBBB" and with the secondary endorser.
All % Codes are allowed for AAAAAAA as well as BBBBBBBB.

One of these fonts can be chosen:

0	Standard
1	IBM2 schmal
2	OCR-A schmal
3	System schmal
4	Terminal schmal
5	Time New Roman schmal
6	IBM2 normal
7	OCR-A normal
8	System normal
9	Terminal normal
10	Time New Roman normal
11	IBM2 wide
12	OCR-A wide

13	System wide
14	Terminal wide
15	Time New Roman wide

One of these orientations can be chosen:

0	normal
1	rotated
2	mirrored
3	rotated and mirrored

Maximum length of the text to print: 64 if no control codes are used. The value is reduced accordingly when using control codes (font selection, etc.).

⊟ Inotec M03

If "Index Mode" is activated, the user-defined text is not available. In "Index Mode" a text defined in the scanner will be used.

%1F	Font: IBM2 (default)
%2F	Font: System
%3F	Font: Terminal
%4F	Font: Times New Roman
%nS	Font width: *n* Pixel (*n* = [1..5], default = 1)
%nW	Character spacing: *n* Pixel (*n* = [1..10], default = 2)
%1L	Orientation: horizontal
%2L	Orientation: vertical (default)
%0nu	*n* digit counter with leading nulls (*n* = [1..10])
%nu	*n* digit counter without leading nulls (*n* = [1..10])
%u	6 digit counter without leading nulls
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%4D	Tag Nummer (day of the year)
%5D	Century CC
%6D	Year YY
%7D	Month MM
%8D	Month Month
%9D	Tag DD
%0I	ImageAdress Fields: FixText:Volume:Chapter:Page
%01I / %1I	IA-Feld: Fix Text (with/without leading nulls)
%02I / %2I	IA-Feld: Page Counter (with/without leading nulls)
%03I / %3I	IA-Feld: Chapter counter (with/without leading nulls)
%04I / %	IA-Feld: Volume Counter (with/without leading nulls)

4I	
%05I / %5I	IA-Feld: Roll Number (with/without leading nulls)
%06I / %6I	IA-Feld: Sequential ID (with/without leading nulls)
%07I / %7I	IA-Feld: Level (with/without leading nulls)
%08I / %8I	IA-Feld: Mode (with/without leading nulls)
%09I / %9I	IA-Feld: Momentary Flag (with/without leading nulls)
%010I / %10I	IA-Feld: Latched Flag (with/without leading nulls)

%0T	Time (24h) HH:MM
%1T	Time (24h) HH:MM:SS
%2T	Time (12h am/pm) HH:MM
%3T	Time (12h am/pm) HH:MM:SS
%4T	Time (24h) HHMM
%5T	Time (24h) HHMMSS

%0B	Barcode OFF (normal / Text ON)
%1B	Barcode Code 39
%2B	Barcode Code 2/5 Interleaved
%3B	Barcode Code 2/5 Interleaved with Prüfsumme Modulo 10

%nnn**A**	Attribut *nnn* (see Endorser-Manual)
%nnn**V**	Variable *nnn* (see Endorser-Manual)

Maximum length of the text to print:: 255

Older Models

⊟ AGFA Admis S31

%06u	6 digit counter with leading nulls
%6u	6 digit counter without leading nulls
%08u	8 digit counter with leading nulls
%8u	8 digit counter without leading nulls
%..u	all other %u-Codes will be treated as %06u or %08u resp..

%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS

%I	Text "*Advanced Indexing*" (not for all models)

AGFA S35C

This list shows the available control codes when the scanner is equipped with an Inotec endorser.
If the scanner is equipped with a NICA endorser: see own NICA endorser section below.

%0*n*u	*n* digit counter with leading nulls (*n* = [1..8])
%0*n*U	*n* digit counter with leading nulls (*n* = [1..8])
%*n*u	*n* digit counter without leading nulls (*n* = [6 oder 8])
%*n*U	*n* digit counter without leading nulls (*n* = [6 oder 8])

%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%4C	1-stelliger Zähler
%5C	2 digit counter with leading nulls
%6C	3 digit counter with leading nulls
%7C	4 digit counter with leading nulls
%8C	5 digit counter with leading nulls
%9C	7 digit counter with leading nulls

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS

%0B	Barcode OFF (normal / Text-format ON)
%1B	Barcode Code 39
%2B	Barcode Code 2/5 Interleaved

Agfa Admis S61, Microform S61 and Microform S655

%*n*u	maximal *n* digit counter without leading nulls or blanks (*n* = [1..8])
%0*n*u	*n* digit counter with leading nulls (*n* = [1..8])

%0S	Style: Normal (Invers OFF)
%1S	Stylel: Invers ON

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS

%I	Formatted Index-String Warning: The %*n*I codes are available only when the scanner is equipped with an additional patch code reader! The formatting of the index strings can be determined on the setting page "Index".

%0B	Barcode OFF (normal / Text-format ON)
%1B	Barcode Code 2/5 Interleaved
%2B	Barcode Code 39

%! "<ESC>!" Sequenz - see Endorser manual
%* "<ESC>*" Sequenz - see Endorser manual

%& Only, if the scanner is equipped with two endorsers / imprinters: Switches text definition between the primary and secondary printer back and forth.

Sample: The text "AAAAAAA%&BBBBBBBB" will be printed as: "AAAAAAA" with the primary imprinter and "BBBBBBBB" and with the secondary endorser.
All % Codes are allowed for AAAAAAA as well as BBBBBBBB.

One of these fonts can be chosen:

0	Standard
1	IBM2 schmal
2	OCR-A schmal
3	System schmal
4	Terminal schmal
5	Time New Roman schmal
6	IBM2 normal
7	OCR-A normal
8	System normal
9	Terminal normal
10	Time New Roman normal
11	IBM2 wide
12	OCR-A wide
13	System wide
14	Terminal wide
15	Time New Roman wide

One of these orientations can be chosen:

0	normal
1	rotated
2	mirrored
3	rotated and mirrored

Banctec TDC2610 or S-Series

%*n*u	*n* digit counter without leading nulls
%0*n*u	*n* digit counter with leading nulls

Bell+Howell Copiscan II

%05u	5 digit counter with leading nulls
%*n*u	*n* digit counter with leading nulls

The counter is always printed at the end of the text, regardless of the position of **%..u**!

Bell+Howell Copiscan 8000

%05u	5 digit counter with leading nulls
%*n*u	*n* digit counter with leading nulls
%*n*C	*n* digit counter with leading nulls

The counter is **always** printed at the end of the text, regardless of the position of **%..u**!

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS
%2T	Time HH:MM am/pm
%3T	Time HH:MM:SS am/pm

Fujitsu

%05u	5 digit counter with leading nulls
%08u	8 digit counter with leading nulls, only some models
%..u	all other %u-Codes will be treated as %5u or %8u if possible. For the most models the counter is always printed at the end of the text, regardless of the position of **%..u**!

One of these fonts can be chosen:

0	Regular
1	Bold

One of these orientations can be chosen:

0	top down
1	top down, characters rotated
2	bottom up
3	bottom up, characters rotated

Maximum length of the text to print

FI-4860C: 40 with 5-digit counter, 43 with 8-digit counter.
M4099, **FI-4990**: 30 with 5-digit counter, 33 with 8-digit counter.

Inotec Scamax 2500/2600/5000

%06u	6 digit counter with leading nulls
%6u	6 digit counter without leading nulls
%08u	8 digit counter with leading nulls
%8u	8 digit counter without leading nulls
%..u	all other %u-Codes will be treated as %06u or %08u resp.
%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%I	Substring *Advanced Indexing* (only some models)

Inotec Scamax 4000

%0*n*u	*n* digit counter with leading nulls (*n* = [1..9])
%0*n*U	*n* digit counter with leading nulls (*n* = [1..9])
%0*n*c	*n* digit counter with leading nulls (*n* = [1..9])
%*n*u	*n* digit counter without leading nulls (*n* = [1..9])

%nU	*n* digit counter without leading nulls (*n* = [1..9])
%nc	*n* digit counter without leading nulls (*n* = [1..9])
%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%4C	1 digit counter
%5C	2 digit counter with leading nulls
%6C	3 digit counter with leading nulls
%7C	4 digit counter with leading nulls
%8C	5 digit counter with leading nulls
%9C	7 digit counter with leading nulls
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%0f	Use default font
%0F	Use default font
%1f	Use font loaded
%1F	Use font loaded
%0s	Typeface : normal (Attributes off)
%0S	Typeface : normal (Attributes off)
%1s	Typeface : Bold
%1S	Typeface : Bold
%2s	Typeface : Double width
%2S	Typeface : Double width
%3s	Typeface : Underline
%3S	Typeface : Underline

Microform S61

%*n*u	maximal *n* digit counter without leading nulls or blanks (*n* = [1..8])
%0*n*u	*n* digit counter with leading nulls (*n* = [1..8])
%0S	Style: Normal (Invers OFF)
%1S	Stylel: Invers ON
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%I	Formatted Index-String Warning: The %*n*I codes are available only when the scanner is equipped with an additional patch code reader! The formatting of the index strings can be determined on the setting page "Index".
%0B	Barcode OFF (normal / Text-format ON)

%1B	Barcode Code 2/5 Interleaved
%2B	Barcode Code 39
%!	"<ESC>!" Sequenz - see Endorser manual
%*	"<ESC>*" Sequenz - see Endorser manual
%&	Only, if the scanner is equipped with two endorsers / imprinters: Switches text definition between the primary and secondary printer back and forth. Sample: The text "AAAAAAA%&BBBBBBBB" will be printed as: "AAAAAAA" with the primary imprinter and "BBBBBBBB" and with the secondary endorser. All % Codes are allowed for AAAAAAA as well as BBBBBBBB.

One of these fonts can be chosen:

0	Standard
1	IBM2 schmal
2	OCR-A schmal
3	System schmal
4	Terminal schmal
5	Time New Roman schmal
6	IBM2 normal
7	OCR-A normal
8	System normal
9	Terminal normal
10	Time New Roman normal
11	IBM2 wide
12	OCR-A wide
13	System wide
14	Terminal wide
15	Time New Roman wide

One of these orientations can be chosen:

0	normal
1	rotated
2	mirrored
3	rotated and mirrored

⊟ NICA-Endorser

%0*n*u	*n* digit counter with leading nulls (*n* = [1..9])
%0*n*U	*n* digit counter with leading nulls (*n* = [1..9])
%0*n*c	*n* digit counter with leading nulls (*n* = [1..9])
%*n*u	*n* digit counter without leading nulls (*n* = [1..9])
%*n*U	*n* digit counter without leading nulls (*n* = [1..9])
%*n*c	*n* digit counter without leading nulls (*n* = [1..9])
%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%4C	1 digit counter
%5C	2 digit counter with leading nulls
%6C	3 digit counter with leading nulls
%7C	4 digit counter with leading nulls
%8C	5 digit counter with leading nulls
%9C	7 digit counter with leading nulls

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%0f	Use default font
%0F	Use default font
%1f	Use font loaded
%1F	Use font loaded
%0s	Typeface : normal (Attributes off)
%0S	Typeface : normal (Attributes off)
%1s	Typeface : Bold
%1S	Typeface : Bold
%2s	Typeface : Double width
%2S	Typeface : Double width
%3s	Typeface : Underline
%3S	Typeface : Underline

⊟ Panasonic KV-S2055

%07u	7 digit counter with leading nulls
%7u	7 digit counter without leading nulls
%..u	all other %u-Codes will be treated as %07u
%0C	7-stelliger Counter with leading spaces
%1C	7 digit counter with leading nulls
%0D	Date YY/MM/DD
%1D	Date YYYY/MM/DD
%2D	Date MM/DD/YY
%3D	Date DD month name short, YYYY
%0T	Time HH:MM
%1T	Time HH:MM:SS

⊟ Photomatrix

%05u	5 digit counter with leading nulls
%..u	all other %u-Codes will be treated as %05u

⊟ Ricoh IS410/420/430/4x0

Text only - no counters.

See also scanner setup dialog.

3.7.4.2.4 Endorsers (Alphabetical List)

List of endorser formatting characters sorted by scanner models, older models grayed out.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

AGFA Admis S31

%06u	6 digit counter with leading nulls
%6u	6 digit counter without leading nulls
%08u	8 digit counter with leading nulls
%8u	8 digit counter without leading nulls
%..u	all other %u-Codes will be treated as %06u or %08u resp..

%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS

%I	Text "*Advanced Indexing*" (not for all models)

AGFA S35C

This list shows the available control codes when the scanner is equipped with an Inotec endorser. If the scanner is equipped with a NICA endorser: see own NICA endorser section below.

%0*n*u	*n* digit counter with leading nulls (*n* = [1..8])
%0*n*U	*n* digit counter with leading nulls (*n* = [1..8])
%*n*u	*n* digit counter without leading nulls (*n* = [6 oder 8])
%*n*U	*n* digit counter without leading nulls (*n* = [6 oder 8])

%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%4C	1-stelliger Zähler
%5C	2 digit counter with leading nulls
%6C	3 digit counter with leading nulls
%7C	4 digit counter with leading nulls
%8C	5 digit counter with leading nulls
%9C	7 digit counter with leading nulls

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS

%0B	Barcode OFF (normal / Text-format ON)
%1B	Barcode Code 39
%2B	Barcode Code 2/5 Interleaved

Agfa Admis S61, Microform S61 and Microform S655

%*n*u	maximal *n* digit counter without leading nulls or blanks (*n* = [1..8])
%0*n*u	*n* digit counter with leading nulls (*n* = [1..8])
%0S	Style: Normal (Invers OFF)
%1S	Stylel: Invers ON
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%I	Formatted Index-String Warning: The %*n*I codes are available only when the scanner is equipped with an additional patch code reader! The formatting of the index strings can be determined on the setting page "Index".
%0B	Barcode OFF (normal / Text-format ON)
%1B	Barcode Code 2/5 Interleaved
%2B	Barcode Code 39
%!	"`<ESC>!`" Sequenz - see Endorser manual
%*	"`<ESC>*`" Sequenz - see Endorser manual
%&	Only, if the scanner is equipped with two endorsers / imprinters: Switches text definition between the primary and secondary printer back and forth. Sample: The text "AAAAAAA%&BBBBBBBB" will be printed as: "AAAAAAA" with the primary imprinter and "BBBBBBBB" and with the secondary endorser. All % Codes are allowed for AAAAAAA as well as BBBBBBBB.

One of these fonts can be chosen:

0	Standard
1	IBM2 schmal
2	OCR-A schmal
3	System schmal
4	Terminal schmal
5	Time New Roman schmal
6	IBM2 normal
7	OCR-A normal
8	System normal
9	Terminal normal
10	Time New Roman normal
11	IBM2 wide
12	OCR-A wide
13	System wide
14	Terminal wide
15	Time New Roman wide

One of these orientations can be chosen:

0	normal
1	rotated
2	mirrored
3	rotated and mirrored

⊟ Banctec TDC2610 or S-Series

%*n*u	*n* digit counter without leading nulls
%0*n*u	*n* digit counter with leading nulls

⊟ Bell+Howell Copiscan II

%05u	5 digit counter with leading nulls
%*n*u	*n* digit counter with leading nulls

The counter is always printed at the end of the text, regardless of the position of **%..u**!

⊟ Bell+Howell Copiscan 8000

%05u	5 digit counter with leading nulls
%*n*u	*n* digit counter with leading nulls
%*n*C	*n* digit counter with leading nulls
	The counter is **always** printed at the end of the text, regardless of the position of **%..u**!

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS
%2T	Time HH:MM am/pm
%3T	Time HH:MM:SS am/pm

⊟ Canon DR 3020

Not controllable by driver.

⊟ Fujitsu

%05u	5 digit counter with leading nulls
%08u	8 digit counter with leading nulls, only some models
%..u	all other %u-Codes will be treated as %5u or %8u if possible. For the most models the counter is always printed at the end of the text, regardless of the position of **%..u**!

One of these fonts can be chosen:

0	Regular
1	Bold

One of these orientations can be chosen:

0	top down
1	top down, characters rotated
2	bottom up
3	bottom up, characters rotated

Maximum length of the text to print

FI-4860C: 40 with 5-digit counter, 43 with 8-digit counter.
M4099, **FI-4990**: 30 with 5-digit counter, 33 with 8-digit counter.

⊟ Inotec M03

If "Index Mode" is activated, the user-defined text is not available. In "Index Mode" a text defined in the scanner will be used.

%1F	Font: IBM2 (default)
%2F	Font: System
%3F	Font: Terminal
%4F	Font: Times New Roman

%nS	Font width: *n* Pixel (*n* = [1..5], default = 1)

%nW	Character spacing: *n* Pixel (*n* = [1..10], default = 2)

%1L	Orientation: horizontal
%2L	Orientation: vertical (default)

%0nu	*n* digit counter with leading nulls (*n* = [1..10])
%nu	*n* digit counter without leading nulls (*n* = [1..10])
%u	6 digit counter without leading nulls

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%4D	Tag Nummer (day of the year)
%5D	Century CC
%6D	Year YY
%7D	Month MM
%8D	Month Month
%9D	Tag DD

%0I	ImageAdress Fields: FixText:Volume:Chapter:Page
%01I / %1I	IA-Feld: Fix Text (with/without leading nulls)
%02I / %2I	IA-Feld: Page Counter (with/without leading nulls)
%03I / %3I	IA-Feld: Chapter counter (with/without leading nulls)
%04I / %4I	IA-Feld: Volume Counter (with/without leading nulls)
%05I / %5I	IA-Feld: Roll Number (with/without leading nulls)
%06I / %6I	IA-Feld: Sequential ID (with/without leading nulls)
%07I / %7I	IA-Feld: Level (with/without leading nulls)
%08I / %8I	IA-Feld: Mode (with/without leading nulls)
%09I / %9I	IA-Feld: Momentary Flag (with/without leading nulls)
%010I / %10I	IA-Feld: Latched Flag (with/without leading nulls)

%0T	Time (24h) HH:MM
%1T	Time (24h) HH:MM:SS
%2T	Time (12h am/pm) HH:MM
%3T	Time (12h am/pm) HH:MM:SS
%4T	Time (24h) HHMM

%5T	Time (24h) HHMMSS

%0B	Barcode OFF (normal / Text ON)
%1B	Barcode Code 39
%2B	Barcode Code 2/5 Interleaved
%3B	Barcode Code 2/5 Interleaved with Prüfsumme Modulo 10

%nnn**A**	Attribut *nnn* (see Endorser-Manual)
%nnn**V**	Variable *nnn* (see Endorser-Manual)

Maximum length of the text to print:: 255

⊟ Inotec M06 per TWAIN-Interface

Valid for InoTec Scamax 402, 403, 412, 413 also 422, 423, 433 and 801, 811, 821

%u	Counter without leading zeros
%0*n*u	n-digit counter with leading zeros (n = [2..10])

%[Lson]	Landscape on
%[Lsoff]	Landscape off
%[Rot]	Text rotation 180°

%[Cd00]	0 character distance in pixel (default)
%[Cd01]	1 character distance in pixel
%[Cd02]	2 character distance in pixel
%[Cd03]	3 character distance in pixel
%[Cd04]	4 character distance in pixel
%[Cd05]	5 character distance in pixel
%[Cd06]	6 character distance in pixel
%[Cd07]	7 character distance in pixel
%[Cd08]	8 character distance in pixel
%[Cd09]	9 character distance in pixel
%[Cd10]	10 character distance in pixel

%[Pt1]	1 Pixel wide (default)
%[Pt2]	2 Pixel wide
%[Pt3]	3 Pixel wide
%[Pt4]	4 Pixel wide
%[Pt5]	5 Pixel wide

%[Seqid]	sequential ID
%[Seqid0]	sequential ID with leading zeros

%[DD.MM.CCYY]	Date DD.MM.YYYY
%[DD.MM.YY]	Date DD.MM.YY
%[MM/DD/YY]	Date MM/DD/YYYY

%[hh:mm]	Time HH:MM
%[hh:mm:ss]	Time HH:MM:SS
%[hh:mm	Time HH:MM AM

AM]	
%[hh:mm:ss AM]	Time HH:MM:SS AM
%[DOY]	Day des Jahres DDD
%[CC]	Jahrhundert
%[YY]	Year YY
%[DD]	Day DD
%[MM]	Month MM
%[Month]	Name of the month
%[Cd>]	Calculation start
%[<Cd]	Calculation end
%[BC25>]	Barcode 2/5 Start
%[<BC25]	Barcode 2/5 Stop
%[M10/3]	Modulo10/3 character
%[M7]	Modulo10/3 character

⊟ Inotec Scamax 2500/2600/5000

%06u	6 digit counter with leading nulls
%6u	6 digit counter without leading nulls
%08u	8 digit counter with leading nulls
%8u	8 digit counter without leading nulls
%..u	all other %u-Codes will be treated as %06u or %08u resp.
%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%I	Substring *Advanced Indexing* (only some models)

⊟ Inotec Scamax 4000

%0nu	*n* digit counter with leading nulls (*n* = [1..9])
%0nU	*n* digit counter with leading nulls (*n* = [1..9])
%0nc	*n* digit counter with leading nulls (*n* = [1..9])
%nu	*n* digit counter without leading nulls (*n* = [1..9])
%nU	*n* digit counter without leading nulls (*n* = [1..9])
%nc	*n* digit counter without leading nulls (*n* = [1..9])
%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%4C	1 digit counter
%5C	2 digit counter with leading nulls
%6C	3 digit counter with leading nulls

%7C	4 digit counter with leading nulls
%8C	5 digit counter with leading nulls
%9C	7 digit counter with leading nulls

%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY

%0T	Time HH:MM
%1T	Time HH:MM:SS

%0f	Use default font
%0F	Use default font
%1f	Use font loaded
%1F	Use font loaded

%0s	Typeface : normal (Attributes off)
%0S	Typeface : normal (Attributes off)
%1s	Typeface : Bold
%1S	Typeface : Bold
%2s	Typeface : Double width
%2S	Typeface : Double width
%3s	Typeface : Underline
%3S	Typeface : Underline

⊟ Kodak 5000, 7000 and older 500, 500A Modelle via TWAIN Interface

%0P	Front side text follows
%1P	Rear side text follows

%1L	Level_1 Text follows
%2L	Level_2 Text follows
%3L	Level_3 Text follows
	The last %*n*L code on the text defines the current document level.
	If there is no %*n*L in the text, the text is used for all levels

%u	Counter (9 characters)
%nu	Counter (9 characters)

%0A	Complete IA, no delimiter
%1A	Complete IA, all delimiter
%2A	Complete IA, fixed delimiter

%0T	Time HH:MM
%D	Date
%y	Day number (Day of the year)

%1B	Fixed Field IA
%2B	Index Field IA
%3B	Current Level IA

%M	Delimiter

%0S	Style/Font: large comic
%1S	Style/Font: large cinema
%2S	Style/Font: large comic, 180° gedreht
%3S	Style/Font: large cinema, 180° gedreht

%4S	Style/Font: small comic
%5S	Style/Font: small cinema
%6S	Style/Font: small comic, 180° gedreht
%7S	Style/Font: small cinema, 180° gedreht
	If no %*n*S is given in the text the Font set in the TWAIN User Interface font is used.

Maximum number of characters is 40.

Kodak i800 Models via TWAIN Interface

Valid for older i800

%1L	Level_1 Text follows
%2L	Level_2 Text follows
%3L	Level_3 Text follows

The last %*n*L in string selects for which "document level" the described printing will occur.
If %nL isn't specified the printing will occur for all document levels (always)

%0*n*u	*n*-digit counter with leading zeros [*n* = 1..9]
%*n*u	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<*n*u	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%6B	ImageAddress Feld A
%7B	ImageAddress Feld B
%8B	ImageAddress Feld C
%9B	ImageAddress Feld D

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY
%ijD	Several date formats with: [*i* = 1..3], [*j* = 0..4]

%0S	Style/Font: large comic
%1S	Style/Font: large cinema
%4S	Style/Font: small comic
%5S	Style/Font: small cinema

If there is no %nS in the string, the font selected in TWAIN UserInterface will be used.

Maximum number of printable characters: 40

Kodak 3520, 4500 and i200 Modelle via TWAIN Interface

Valid for older Kodak 3520 and other

%0*n*u	*n*-digit counter with leading zeros [*n* = 1..9]
%*n*u	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<*n*u	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY
%*ij*D	Several date formats with: [*i* = 1..3], [*j* = 0..4]

%0S	Style/Font: large comic

%1S	Style/Font: large cinema
%4S	Style/Font: small comic
%5S	Style/Font: small cinema

If there is no %nS in the string, the font selected in TWAIN UserInterface will be used.
Beside the given codes, and blanks, only one unique text element can be processed!

Maximum number of printable characters: 40

⊟ Kodak i1800 via TWAIN

Valid for Kodak i4200 and i4600

The maximum length of the text to print is 40 characters.
These placeholders are available:

%0*n*u	*n*-digit counter with leading zeros [*n* = 1..9]
%*n*u	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<*n*u	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%0*n*u	*n*-stelliger Zähler mit führenden Nullen [*n* = 1..9]
%*n*u	*n*-stelliger Zähler mit führenden Leerzeichen [*n* = 1..9]
%u	Zähler, entspricht %<9u
%<*n*u	Zähler, linksbündig (*n* digit) [*n* = 1..9]

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY
%ijD	other date formats with : [i = 1..3], [*j* = 0..4], where i indicates the type and j the punctuation character

i=1	MMDDYYYY month 2 digits, day 2 digits, year 4 digits (j see below)
i=2	DDMMYYYY day 2 digits, month 2 digits, year 4 digits (j see below)
i=3	YYYYMMDD year 4 digits, month 2 digits, day 2 digits (j see below)
i=4	DDD day of the year 3 digits (j **not** given)
i=4	YYYYDDD year 4 digits, day of the year (j see below)

j=0	no punctuation
j=1	slash
j=2	minus
j=3	dot
j=4	space

If two placeholders of the same type are used, the last set formatting applies to both.

⊟ Kodak i4000 via TWAIN

Valid for Kodak i4200 and i4600

The maximum length of the text to print is 40 characters.
These placeholders are available:

%0*n*u	*n*-digit counter with leading zeros [*n* = 1..9]

%*n*u	*n*-digit counter without leading zeros [*n* = 1..9]
%u	Counter, without leading zeros (9 digit)
%<*n*u	Counter, without leading zeros (*n* digit) [*n* = 1..9]

%0*n*u	*n*-stelliger Zähler mit führenden Nullen [*n* = 1..9]
%*n*u	*n*-stelliger Zähler mit führenden Leerzeichen [*n* = 1..9]
%u	Zähler, entspricht %<9u
%<*n*u	Zähler, linksbündig (*n* digit) [*n* = 1..9]

%0T	Time HH:MM
%0D	Date DD.MM.YYYY
%2D	Date MM/DD/YYYY
%ijD	other date formats with : [i = 1..3], [*j* = 0..4], where i indicates the type and j the punctuation character

i=1	MMDDYYYY month 2 digits, day 2 digits, year 4 digits (j see below)
i=2	DDMMYYYY day 2 digits, month 2 digits, year 4 digits (j see below)
i=3	YYYYMMDD year 4 digits, month 2 digits, day 2 digits (j see below)
i=4	DDD day of the year 3 digits (j **not** given)
i=4	YYYYDDD year 4 digits, day of the year (j see below)

j=0	no punctuation
j=1	slash
j=2	minus
j=3	dot
j=4	space

If two placeholders of the same type are used, the last set formatting applies to both.

Kodak i5000 via TWAIN

Valid for Kodak i5200, i5250, i5600, i5650, i5650S, i5800, i5850, i5850S

The maximum lengt of the text to print is 40 characters.
These placeholders are available:

%0*n*u	*n*-stelliger Zähler mit führenden Nullen [*n* = 1..9]
%*n*u	*n*-stelliger Zähler mit führenden Leerzeichen [*n* = 1..9]
%u	Zähler, entspricht %<9u
%<*n*u	Zähler, linksbündig (*n* digit) [*n* = 1..9]

%0T	Uhrzeit HH:MM
%0D	Datum DD.MM.YYYY
%2D	Datum MM/DD/YYYY
%ijD	other date formats with : [i = 1..3], [*j* = 0..4], where i indicates the type and j the punctuation character i

i=1	MMDDYYYY month 2 digits, day 2 digits, year 4 digits (j see below)
i=2	DDMMYYYY day 2 digits, month 2 digits, year 4 digits (j see below)
i=3	YYYYMMDD year 4 digits, month 2 digits, day 2 digits (j see below)
i=4	DDD day of the year 3 digits (j **not** given)
i=4	YYYYDDD year 4 digits, day of the year (j see below)
i=5	DDMMYY day 2 digits, month 2 digits, year 2stellig (j see below)
i=6	MMDDYY month 2 digits, day 2 digits, year 2stellig (j see below)
i=7	YYDDD year 2 digits, day of the year (j see below)
i=8	YYMMDD year 2 digits , month 2 digits, day 2 digits (j see below)

j=0	no punctuation
j=1	slash
j=2	minus
j=3	dot
j=4	space

If two placeholders of the same type are used, the last set formatting applies to both.

Microform S61

%*n*u	maximal *n* digit counter without leading nulls or blanks (*n* = [1..8])
%0*n*u	*n* digit counter with leading nulls (*n* = [1..8])
%0S	Style: Normal (Invers OFF)
%1S	Stylel: Invers ON
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%I	Formatted Index-String Warning: The %*n*I codes are available only when the scanner is equipped with an additional patch code reader! The formatting of the index strings can be determined on the setting page "Index".
%0B	Barcode OFF (normal / Text-format ON)
%1B	Barcode Code 2/5 Interleaved
%2B	Barcode Code 39
%!	"`<ESC>!`" Sequenz - see Endorser manual
%*	"`<ESC>*`" Sequenz - see Endorser manual
%&	Only, if the scanner is equipped with two endorsers / imprinters: Switches text definition between the primary and secondary printer back and forth. Sample: The text "AAAAAAA%&BBBBBBBB" will be printed as: "AAAAAAA" with the primary imprinter and "BBBBBBBB" and with the secondary endorser. All % Codes are allowed for AAAAAAA as well as BBBBBBBB.

One of these fonts can be chosen:

0	Standard
1	IBM2 schmal
2	OCR-A schmal
3	System schmal
4	Terminal schmal
5	Time New Roman schmal
6	IBM2 normal
7	OCR-A normal
8	System normal
9	Terminal normal
10	Time New Roman normal

11	IBM2 wide
12	OCR-A wide
13	System wide
14	Terminal wide
15	Time New Roman wide

One of these orientations can be chosen:

0	normal
1	rotated
2	mirrored
3	rotated and mirrored

Microform S415, S455, S615 and S655

%*n*u	maximum *n* digit counter without leading nulls or blanks (*n* = [1..8])
%0*n*u	*n* digit counter with leading nulls (*n* = [1..8])
%0S	Style: Normal (Inverted OFF)
%1S	Style: Inverse ON
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%I	Formatierter Index-String Warning: The %*n*I codes are available only when the scanner is equipped with an additional patch code reader! The formatting of the index strings can be determined on the setting page "Index"..
%0B	Barcode OFF (normal text ON)
%1B	Barcode Code 2/5 Interleaved
%2B	Barcode Code 39
%!	"<ESC>!" Sequenz - see Endorser manual
%*	"<ESC>*" Sequenz - see Endorser manual
%&	Only, if the scanner is equipped with **two endorsers / imprinters**: Switches text definition between the primary and secondary printer back and forth. Sample: The text "AAAAAAA%&BBBBBBBB" will be printed as: "AAAAAAA" with the primary imprinter and "BBBBBBBB" and with the secondary endorser. All % Codes are allowed for AAAAAAA as well as BBBBBBBB.

One of these fonts can be chosen:

0	Standard
1	IBM2 schmal
2	OCR-A schmal
3	System schmal
4	Terminal schmal
5	Time New Roman schmal
6	IBM2 normal
7	OCR-A normal
8	System normal
9	Terminal normal

10	Time New Roman normal
11	IBM2 wide
12	OCR-A wide
13	System wide
14	Terminal wide
15	Time New Roman wide

One of these orientations can be chosen:

0	normal
1	rotated
2	mirrored
3	rotated and mirrored

Maximum length of the text to print: 64 if no control codes are used. The value is reduced accordingly when using control codes (font selection, etc.).

NICA-Endorser

%0*n*u	*n* digit counter with leading nulls (*n* = [1..9])
%0*n*U	*n* digit counter with leading nulls (*n* = [1..9])
%0*n*c	*n* digit counter with leading nulls (*n* = [1..9])
%nu	*n* digit counter without leading nulls (*n* = [1..9])
%nU	*n* digit counter without leading nulls (*n* = [1..9])
%nc	*n* digit counter without leading nulls (*n* = [1..9])
%0C	6 digit counter with leading nulls
%1C	6 digit counter without leading nulls
%2C	8 digit counter with leading nulls
%3C	8 digit counter without leading nulls
%4C	1 digit counter
%5C	2 digit counter with leading nulls
%6C	3 digit counter with leading nulls
%7C	4 digit counter with leading nulls
%8C	5 digit counter with leading nulls
%9C	7 digit counter with leading nulls
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%0f	Use default font
%0F	Use default font
%1f	Use font loaded
%1F	Use font loaded
%0s	Typeface : normal (Attributes off)
%0S	Typeface : normal (Attributes off)
%1s	Typeface : Bold
%1S	Typeface : Bold
%2s	Typeface : Double width

%2S	Typeface : Double width
%3s	Typeface : Underline
%3S	Typeface : Underline

⊟ Panasonic KV-S2055

%07u	7 digit counter with leading nulls
%7u	7 digit counter without leading nulls
%..u	all other %u-Codes will be treated as %07u
%0C	7-stelliger Counter with leading spaces
%1C	7 digit counter with leading nulls
%0D	Date YY/MM/DD
%1D	Date YYYY/MM/DD
%2D	Date MM/DD/YY
%3D	Date DD month name short, YYYY
%0T	Time HH:MM
%1T	Time HH:MM:SS

⊟ Photomatrix

%05u	5 digit counter with leading nulls
%..u	all other %u-Codes will be treated as %05u

⊟ Ricoh IS410/420/430/4x0

Nur Text - kein Zähler.

⊟ XINOscan S700

%*n*u	n-digit counter without leading zeros, n=1..10
%0*n*u	n-digit counter with leading zeros, n=1..10
%0&	Toggle between endorser and imprinter
%1&	Set Endorser (after the Scan, rear side)
%2&	Set Imprinter (before the Scan, front side)
%0D	Date DD.MM.YYYY
%1D	Date DD.MM.YY
%2D	Date MM/DD/YYYY
%3D	Date MM/DD/YY
%0T	Time HH:MM
%1T	Time HH:MM:SS
%2T	Time HH:MM a.m./p.m.
%3T	Time HH:MM:SS a.m./p.m.
%0N	Username
%1N	Scanner name

%0*n*i %*n*i Level 1 Index-Counter print (n-digits, with/without leading zeros)
%0*n*j %*n*j Level 2 Index-Counter print (n-digits, with/without leading zeros)
%0*n*k %*n*k Level 3 Index-Counter print (n-digits, with/without leading zeros)
%0*n*l %*n*l Level 4 Index-Counter print (n-digits, with/without leading zeros)

%I Print all active Index counter, see Index for formatting

%*nn*M Set Check sum mode (nn = [00..99] check sum type)
0: 231-weights, modulo 11, 10 & 11 = 0, see example.
1: Code 39-reference numbers with Modulo 43, see example.
%S Begin of range for check sum calculating
%E End of range for check sum calculating
%P prints check sum

The control codes **%*nn*M**, **%S**, **%E**, **%P** must occur in this *order* in the text.

%*n*W Character width:
0: Default width
1: Double Width

%*n*F Type of font
0: Standard font
1: Loadable Font

An additional font can be edited and loaded using a special font editor for this device. Please contact the vendor for this tool.

%*n*B Reduce the character's width:
0: print all 12 columns
1: print only the first 10 columns
2: print only the first 8 columns
3: print only the first 6 columns

For a special small font this increases the number of possible characters to print. The font must be loaded and activated with **%*1*F**

%*n*X **Character spacing**
0: Regular type
1: Spaced type

%R Text printing rotated 180°

Over all the length of the text may not exceed 200 characters.

See also endorser formats for ISIS-Scanners, TWAIN-Scanners and directly controlled Scanners and further Scanner Einstelldialog

3.7.5 TWAIN-Scanner

TWAIN scanners are scanners that are accessed via a driver corresponding to the TWAIN standard.

With a few exceptions, the scanner will be delivered with a TWAIN driver always. Please check whether the driver delivered is up to date and download possibly a newer version from the scanner manufacturer.

The program is able to operate with the help of a special library such scanners. Here you can set how you want to communicate with the scanner and which values, such as Brightness to be controlled directly by the program.

For this purpose two tabs are displayed:

TWAIN, User Interface and data transfer

TWAIN Specials, direct control by the program

See also General, Paper, Miscellaneous, Endorser, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.5.1 TWAIN

This page allows to set the behavior of the TWAIN User Interface and the image transfer.

Show User Interface .. Opens the setup dialog of the scanner driver. For the other settings there consult the manual of the scanner manufacturer for help.

Note: Some TWAIN scanner offer to save your settings when you **exit the setup dialog**. This is useful for other programs that don't have own storage options. So if you use the scanner with other software too, you should ensure that the settings are not overwritten. In this case you should just cancel the Save dialog, or leave it with "no".

Show User Interface while Scanning Displays the setup dialog when scanning to permanent. It may happen that the driver of the scanner requires this procedure. In general, it makes more sense to hide the dialog, because the functions start / stop can also be performed by the program.

Show Indicators Shows - if exists - a status dialog of the scanner on the screen while scanning. This can be, for example, a progressbar for a flatbed scanner.

TWAIN Setup The following values defines how the communication shall carried out between the program and the driver. These settings are determined automatically and should be changed only in a case of a failure.

Image Transfer Mode

Memory

In this mode the images can be transferred compressed. This may save time.

Native

This basic mode is defined for all TWAIN scanner and should work always.

Bitonal Transfer Compression

G4

Transfers black/white images compressed by the CCITT method G4.

Uncompressed

Transfers black/white images uncompressed

Panel

Uses the parameters as set in the scanner driver or at the machine.

Gray/Color Transfer Compression

JPEG

Transfers gray images and color images compressed by the JPEG method

Uncompressed

Transfers gray images and color images uncompressed.

Panel

Uses the parameters as set in the scanner driver or at the machine.

Bitonal Photometric

Uncomp.

G4

Defines the interpretation of the image data for black/white images. This can be set for G4 and uncompressed images separately.

normal: Ones will be interpreted as black, zeros as white.

Reverse: Zeros will be interpreted as black, ones as white

If the scanner serves the TWAIN-Capabilitiy **TWEI_PIXELFLAVOR** , also this values may be used.

XImg Info: Interpretation as given in the XImg information

rev. XImg Info: Reversed interpretation as given in the XImg information

Paper Side Info

Defines how descition is

Specifies how to determine whether the side is a front or back.

by DS:

Only the information from the DS (also know as DATASOURCE) is used..

by DS/Counter:

First it will be tried to get the information from the DS. Then, if this doesn't exists, the internal counter is used.

by DS/Counter

The internal counter is used.

more ...

Opens a further setup dialog to determine the procedures after leaving the driver's setting.

See also TWAIN-Scanner, TWAIN Specials and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.5.1.1 TWAIN more

This setting is normally preset already correctly when loading the driver. Please change it only if you have detailed knowledge of the processes or if you are prompted to after consultation with us.

set CAP_AUTOFEED

If switched on, the scanner automatically pulls in the next page from the document feeder after a successful scan.

set CAP_AUTOSCAN

Sometimes a scanner is able to scan in auto mode (queuing mode), but does not confirm this capability at startup phase.

Enabling this checkbox will override this and mark the scanner as ready for auto scan.

get DAT:IMAGEINFO in.

Choose, when to operate DAT_IMAGEINFO / MSG_GET to read the general image description information. Standard is State 7 (that's when image data transfer is ready).
Some TWAIN driver handle over incorrect informations in State 7; image data interpretation fails then. Choose State 6 (that's just before image data transfer starts) in those cases.

send MSG_STOPFEEDER

The Program can stop the scanning in queuing mode with the TWAIN command MSG_STOPFEEDER only.
This command should be available if the scanner has the capability CAP_AUTOSCAN (scanning in queuing mode).

Define here how this command should be sent:

normal

The command will just be sent.

never

The command will never be sent. This is useful for **older scanner** models not knowing this command and throwing an error instead.

always

There are scanners knowing the command but doesn't working in queuing mode. If such a device doesn't work correctly in **normal** mode use **always**.

Alternatively send PENDINGXFERS/RESET to stop feeder

If activated, in case of a stop feeder request from aplication, and if MSG_STOPFEEDER does not work or is disabled (send never), MSG_RESET is send to the TWAIN scanner.

This will stop most scanner models, but buffered prescanned images will be lost.

Avoid PENDINGXFERS/RESET

Disables the attempt to stop the scanner with the Reset-Command when the scan application wants to break the scanning process.

Thus will keep the prescanned images.

To break the process the operator has to press the stop button at the device.

Limit MemXFer Block Size

In case of errors during the image transfer from the scanner to the receiving application, the size of the transferred memory blocks can be specified.

For that the data source can be asked for the smallest, largest and preferred size of the buffer.

unlimited / Complete Image Size

Send the image in one block if possible.

DS:Preferred Value

Send the image in blocks of the preferred size as given by the driver (data source)

DS:MinBufferSize

Send the image in smallest blocks as possible and allowed by the by the driver (data source).

DS:MaxBufferSize

Send the image in largest blocks as possible and allowed by the by the driver (data source).

While MemXfer ignore DS:MinBufferSize ignore DS:MaxBuffersize limit

In case of a transfer errors you can try to override the values suggested by the driver (data source). You can try to ignore the minimum the maximum or both values for the block size.

Realloc MemXfer Blocks in steps of

Here one can specify the size steps which shall be used to enlarge or reduce the memory allocated currently for the next (smaller or larger) image.

store TWEI Barcode Information in Image Header

Allows to hand over the barcode search result in the Image Header.

BC-Setup...

This button will open a configuration dialog for the barcode search options.

always get Extra&Imageinfo (TWEI)

HIDC_GETTWEIALWAYS_CHECK

Add ICC profile...

Allows loading of ICC profiles for the front and rear cameras. These are tables that assign precisely defined color values to the scanned color values.

The control elements below helps to locate and load the ICC profile files.

Delay Endxfer

Delays the check for the end of transfer.

Throw No Paper after unbuffered scan.

Throws the error „No Paper“ after each scan in non-queuing mode. It’s up to the scan application to handle this error, for example to display a message. Otherwise the scanner will wait for the next paper or a timeout.

After scanning unload and reload DS

This option unloads and reloads the driver after scanning. This will release locked resources.

on MSG_CLOSEDSOK

Choose what to process when TWAIN command MSG_CLOSEDSREQ occures (when User Interface closes):

D+C+O+P (default)

The TWAIN-driver will be **d**isabled, then **c**losed, then **o**pened again and then the **p**arameters will be read again.

This works with all scanners but takes a little time because the driver will be reloaded.

D+P

The TWAIN-driver will be **d**isabled then and then the **p**arameters will be read again. This is faster but not all TWAIN-Scanners support this method.

D+P+C

The TWAIN-driver will be **d**isabled then and then the **p**arameters will be read

Load PresetFile before ENABLE_ DS

Loads the settings from a Preset File, formerly exported by the driver in the user interface.
The following controls will help to locate the file and to specify the behavior if the file cannot be found at run time. "MessageBox will show a message, where throw ScanError 05556 will just inform the scan application.

3.7.5.2 TWAIN Specials

Only experts should select other values than the defaults here!

This configuration is global for ALL TWAIN-Scanners using this interface.

The changes take place only after closing and restarting the application.

Some of this settings are no longer important for the version 6 of the program because the corresponding control elements no longer exists.

Special Extra Settings

These settings defines how the TWAIN-User-Interface will be bound to the program.

Subclass Dialog

If this option is enabled the TWAIN-Dialog will be opened as SUBCLASS dialog. So it will be part of its message queue.

Otherwise it will created with its own window class.

send Timer MSGs

These settings determine whether permanently windows of the driver, such as a progressbar, will received messages for synchronization.

Allow Control by Application

With this option important settings can be made directly by the program. If disabled changes can be made in the TWAIN-User-Interface only.

To avoid side effects these options should be enabled only for values that are really used by the program.

Color Mode

Determines whether the images to fetch (color, gray, black and white) can be set also on the relevant page of the scanner parameters.

Resolution

Determines whether resolution can be set also on the relevant page of the scanner parameters.

Brightness Determines whether the brightness can be set also on the relevant page of the scanner parameters.

... reversed Values Should be switched if the sliders in the program works in a reverse way.

Threshold Determines whether the threshold can be set also on the relevant page of the scanner parameters.

... reversed Values Should be switched if the sliders in the program works in a reverse way.

Contrast Determines whether the contrast can be set also on the relevant page of the scanner parameters.

... reversed Values Should be switched if the sliders in the program works in a reverse way.

Paper Source Specifies if the paper source (flatbed or feeder) can be set also on the relevant page of the scanner parameters.

Paper Size Specifies if the paper size (A5,A4,...) can be set also on the relevant page of the scanner parameters.

Endorser Describes if the endorser can be set also on the corresponding page of the scanner parameters.

See also TWAIN-Scanner, TWAIN and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.6 ISIS-Scanner

ISIS scanners are scanners that are accessed via a driver that corresponds to the ISIS standard.

The program is able to operate with help of a special library such scanners. Here you can set how you want to communicate with the scanner and which values, such as brightness or contrast to be controlled directly from the program.

For this purpose this tab is displayed:

ISIS, User Interface and data transfer

See also General, Paper, Miscellaneous, Endorser, TWAIN-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.6.1 ISIS

This page allows to open the ISIS User Interface and to specify the image transfer.

Scan Dialog Opens the settings dialog of the scanner with the most important parameters, for example, size, brightness, contrast, etc. For the settings there, please consult the manual of the scanner manufacturer for help.

More Dialog Opens a dialog for the further and special parameters such as the details for the binarization algorithm.

If the scanner driver doesn't support a more-dialog the common setup dialog will be shown.

Color JPEG

Determines whether the color images are retrieved already JPEG-compressed from the scanner.

If adjustable, the compression rate and the quality can be specified by a slide here.

If the scanner driver offers detailed setting for this by **Panel** can be set that the settings in the driver take precedence. The values set here are ignored then.

Gray JPEG

Determines whether the grayscale images are retrieved already JPEG-compressed from the scanner.

If adjustable, the compression rate and the quality can be specified by a slide here.

If the scanner driver offers detailed setting for this by **Panel** can be set that the settings in the driver take precedence. The values set here are ignored then.

G4

Determines whether the bitonal images are retrieved already G4-compressed from the scanner.

If the scanner driver offers detailed setting for this by **Panel** can be set that the settings in the driver take precedence. The values set here are ignored then.

Settings

Opens a dialog where you can specify whether and how certain values used elsewhere in the program and can be adjusted.

See also ISIS-Scanner and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.6.1.1 ISIS-Options

The settings on this page are highly dependent on the equipment of the device. If adjustment is not possible, the corresponding control elements are hidden here.

Show User Interface

Determines which parameters are to be available in the usual pages of the scanner setting dialog.

Resolution

Determines whether resolution can be set also on the relevant page of the scanner parameters.

Feeder

Specifies if the paper source (flatbed or feeder) can be set also on the relevant page of the scanner parameters.

Brightness

Determines whether the brightness can be set also on the relevant page of the scanner parameters.

If the range of this parameter is inversely to the conventional structure, this is indicated by the addition of **Rev**. It can be specified whether the general settings for color, gray and black and white images to be applied for **color**, **gray** and **black and white** images.

Threshold

Determines whether the threshold can be set also on the relevant page of the scanner parameters.

If the range of this parameter is inversely to the conventional structure, this is indicated by the addition of **Rev**. It can be specified whether the general settings for color, gray and black and white images to be applied for **color**, **gray** and **black and white** images.

Contrast

Determines whether the contrast can be set also on the relevant page of the scanner parameters.

If the range of this parameter is inversely to the conventional structure, this is indicated by the addition of **Rev**. It can be specified whether the general settings for color, gray and black and white images to be applied for **color**, **gray** and **black and white** images.

Papersize

Specifies if the paper size (A5,A4,...) can be set also on the relevant page of the scanner parameters.

Area

Describes whether a "Userformat" can be set also on the relevant page of the scanner parameters.

Rotation

Describes whether the rotation in 90° steps can be set also on the relevant page of the scanner parameters.

Imagetype Select

Determines whether the images to fetch (color, gray, black and white) can be set also on the relevant page of the scanner parameters.

Endorser

Enables the configuration options for the printer unit in the scanner

Token Replacement

The scanner-dependent placeholder in an endorser often begin with a characteristic letter, as "&". On the side of the program - in accordance with the %-Codes - is often worked with the percent sign.

Here you can specify how these characters are to be replaced on the respective side..

Large Transfer Buffer

In the SIS standard a fixed buffer size for the handing over of the image data was specified. Meanwhile the most scanners can handle larger blocks of memory, this reduces the time consumption a little bit.

So, In case of a failure, please disable this option.

Prefer TIFF

Determines whether the image data to be transmitted along with a TIFF tag. Otherwise, only the JPEG compressed data is transmitted.

3.7.7 Other Scanners

Scanner can also be driven with a driver that does not support TWAIN and ISIS standards corresponds, for example, if the scanner manufacturer provides a direct access to the functions of the driver, for example the XINO scanner.

In this case, the program uses a setup dialog that is adapted for the particular scanner. Depending on the capabilities of the scanner several tabs are then displayed, for example:

Device

Controls as the double feed detection, or the behavior of the feeder and stacker or the power saving mode.

Image

Sets certain image parameters, such as gamma correction or the distinction between image and text for some scanners, this can be set for the front and

Front, Rear back separately.

Marker recognition For XINO S700 and other devices the detection of colored stickers can be set here.

Color recognition Also for XINO S700 and other devices can be set the automatic detection of colored documents..

Please check the manual of the manufacturer or online help for help in this case.

See also General, Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.8 Index

The index print is a special scanner endorser / imprinter function. Thus, the index counters are printed at the desired position, the text and must contain the formatter **%I**.

Please also note that Scanner placeholder beginning with the % character, requires a **double percent sign**, so **%%I**

The index is made up of several counters ("Level"). From the time of archiving on microfilm submitted the names of the counters: page counter, chapter counter, volume counter and roll counter. The counting is controlled by the built-in scanner patch code reader and by the scanning counter. If the indexing is turned on, according to the defined rules, the four counters are incremented.

Together, read from right to left, they give the so-called image address. This image address and the individual counters can be read from the image information %(S.IMAGE.INFO) or% K, e.g. is the chapter counter in% K [131140].

Enable Check this box to enable the indexing

Index Format Definition

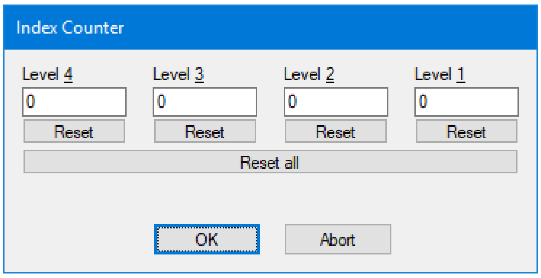
Configure Counter This button opens the Index Counter Configuration. Here you can determine how the counters are initialized, counted and reset.

Separator The character given here will be inserted in between the counters, i.e. between Level1 and Level2, Level2 and Level3, ...

In the image information a period is used always, regardless of a separator specified here.

Counter Values

Setup Values Opens a dialog box to set or reset the counters:



Index Counter

Level 4: 0

Level 3: 0

Level 2: 0

Level 1: 0

Show Setup Panel when Scanning Starts
Specifies that the settings dialog is displayed before scanning the batch. Thus, the operator has the option to correct the values after an interruption, where appropriate.

Printer Control

Don't print onto Patchcode Sheets
Patch code sheets can be sorted out and reused. Then it is useful not to print them.

See also General, Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.8.1 Index Counter Setup

This dialog shows for each of the four counters a tab on which the the size and the counting can be specified.

Field Width
This setting specifies the count of maximum digits which will be used to print the level counter value. Also, the counter can be disabled here: It is not printed and not incremented then

Field Format
Select of:

Compressed

The counter value will be printed without any additional character. The printed field width will be in range of 1 to given Field Width (depending on the current counter value).

Leading Blanks

Counter value printing always takes exactly Field Width digits. Unused digits are filled by SPACE character(s) on the left side.

Leading Zeros

Counter value printing always takes exactly Field Width digits. Unused digits are filled by 0 character on the left side

Reset Value
Enter that value the counter should get if a **Reset Event** occurs.

Trigger Value
When the counter reaches this value, a Level_ # counter event is triggered

Event
There are events where the counter is counted up, and events where it is reset.

Increment Event
Select events for incrementing the counter. Depending on scanner type, some events may be always selected, also the number of events may be limited.

Reset Event
Select events for resetting the counter. Depending on scanner type, some events may be always selected, also the number of events may be limited.

See also Index and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.9 ImagePorcessor

The image processor offers several features for image processing before passing it to the scan program such as rotating, straightening, cutting and merging. The corresponding settings can be made on this tab.

Enable

If checked, the processor is enabled.

If not checked, the scanned images bypass the processor. Note that some other functions are not available then (for example: "Display Start Button for Flatbed scanning" on page MISC).

Image Improvement

Deskew

Check this box if you want the scanned images to be straightened or cropped by the image processor. The other settings can be made here.

Note: It's recommended to set paper width and length to values greater than the physical width of your paper (Overscan).

Use Double Side Algorythm

If checked and if scanning in duplex mode, the skew detector can use informations from front and back side.

Note: With each scan, this function can process only one (!) Front image and one (!) Back image.

If the scanner delivers more than one color type per page, only the two images with the highest color depth are processed here; all other pictures are deskewed individually one by one.
Note: With every scan, this function is able to process only one front and one back side. If scanner sends more than one image color type per side (parameter Camera on page General), the images with the highest color information (bit per pixel) are processed by the double side processor; all other images are deskewed in single mode (image by image).

Deskew Parameter

This button opens a dialog to configure the deskew/cropping parameter.

Rotation

Select an angle to rotate (clockwise for front image) scanned images:

- 90º turn a quarter to the right,
- 180º turn a half,
- 270º turn a quarter to the left.

Note: With most scanners efficiency can be increased by feeding paper in landscape format and 90° rotation afterwards.

more ...

Check this box to enable enhanced features.

more Parameter ...

This button opens a dialog to configure the enhanced parameter.

Extra Images

Split Images

Additional images can be created by splitting scanned images. This button opens the Split Images Manager.

Combined Images Additional images can be created by combining scanned images. This button opens the Combine Images Manager.

Image Sequence

All Images If checked, all available images, i.e. all scanned and all generated ones, are returned in internal order to the application.

Specify Sequence If selected, only the defined images (in the defined order) are transmitted to the application. Use the "Configure List." button to edit the definition.

Configure List This button opens the Returned Image Sequence Editor.

See also Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.9.1 Deskew

If during the scan process the paper wasn't fed accurately aligned usually edges and corners of the scanner's background will become part of the image. With the method described here, these edges can be cut off, this is called "cropping". If the image is rotated previously so far that the paper's edges are parallel to the edges of the image, the resulting strips can be completely removed completely. This rotation is referred to as de-skew or "deskew" shortly.

The deskew parameter get defined separately for bitonal, gray and color images.

The setup is split into these steps:

- Parameter for detecting borders and calculating the destination rectangle, including manipulations to the rectangle and rectangle size checking.
- Image processing parameters based on the rectangle.
- Additional image manipulation settings.

Find Rectangle Parameter

Detection Mode **Brightness** (Black&White only)

Skip black pixel (the background), find white pixel (the document).

Brightness (manually) (Gray or Color only)

Skip dark pixel (the background), find bright pixel (the document);

define the background color level by parameter Threshold.

Brightness (auto) (Gray or Color only)

Skip dark pixel (the background), find bright pixel (the document);

the background color level is decided by the rectangle processor, independently for every image.

Color Sensitive (manually) (Color only)

Skip pixel near the defined Background Color, find pixel color outside the defined Tolerance (the document).
Color Sensitive (auto) (Color only)

Same as above, but Background Color and Tolerance are decided by the rectangle processor, independently for every image

Background Color (Color only)

Default: Black

Defines the scanners background color within the 24 bit color space. For non-standard colors format: 0xrrggbb is used (with rr (red), gg (green), bb (blue) in range of [00..FF]).

Threshold

Range [0..255], Default: 20

(Gray or Color only)

Threshold for searching the borders. A border point is found, if a stream of more than **MinColorPixel** (see dialog "more...") points are brighter than Threshold value.

Tolerance

Range [0..255], Default: 20

(Color only)

Similar to threshold: while threshold gives a value based on color "Black", gives Tolerance a color space around defined Background Color.

Required Borders

Range [1..4], Default: 2

The detector needs to find [1..4] borders to accept the destination rectangle. For example, a sales slip may have only two straight edges at left and right, while the begin and the end of the slip is cracked.

more ...

The button opens the dialog "More Settings".
(These special settings should be changed by experts only.)

Modify Rectangle Size

Check, if the detected rectangle should be manipulated.

Use button "Modify settings ..." to setup manipulation parameters.

Check Rectangle Size

Check, if the rectangle size should be inspected.

Use button "Check settings ..." to setup inspection parameters.

Process Parameter

Process Selector

Default: Deskew

Cropping: The corner points of the rectangle are used to cut off the image frame. Normally, this process reduces the image size in a short processing time.

Deskew: The rectangle is cut out of the image and turned into the nearest perpendicular position. This replaces the original image.

Mode

Range [Fast / 1 Point (Default for Black&White), 1 Point, 2 Point (Default for Gray and Color), 4 Point, Bicubic (16 Point)]

When turning the image, most destination points don't have a corresponding source point. Example: The new image point at [line 9 column 15] is calculated to get from [line 47.35 column 34.86].

There are different processors to calculate the destination points. Use parameter Mode to select the processor you like:

Fast / 1 Point (Black&White only):

If skew angle is less than ±7 degrees, a special, very fast processor is used.

If the skew angle is larger, 1 Point processor is used automatically.

1 Point:

Processor just takes the nearest original point (in given example, it would be [line 47 column 35]. Advantage: fast processor; disadvantage: clean outlines may become rough.

2 Point:

The destination point is calculated as the mean value of the two nearest original points.

4 Point:

The destination point is calculated as the mean value of the four nearest original points. This processor produces smooth outlines best, but is not so fast and the resulting image may loose sharpness.

Bicubic (16 Point) (Gray or Color only): This is an enhanced method with a special border smoothing.

Fill Color

Range [Black, White, 0x...], Default: Black

Areas of the destination rectangle outside the scanned image (typically corner areas, because scan area was smaller than the skewed document), have to be created by the deskew process. These areas will be filled by the chosen color.

Aside from standard colors Black and White, experts are free to enter any color of the available color space: 0xpp on gray, 0xrrggbb on color images (with pp, rr, gg, bb in range of [00..FF]).

Note: If Windows color selection dialog is used to select a color for gray images, the corresponding gray level will be used.

Fill Border

If checked, the background color will be replaced by the given Border Color, starting from the image borders, till non-background color is detected. This process is controlled by parameters in "Fill settings ..." dialog.

Border Color

Range [Black, White, 0x..., Image Color*], Default: Black

Defines the color that replaces background color near the image borders.

Aside from standard colors Black and White, experts are free to enter any color of the available color space: 0xpp on gray, 0xrrggbb on color images (with pp, rr, gg, bb in range of [00..FF]).

If Image Color* is used, the processor steps through background color inside the image, takes a color near this point, and fills up the background area using this color. It should not be used to fill large areas!

Additional Features

Adjust Image Size

If checked, the resulting image can be adjusted in size. Enter "Adjust settings ..." dialog to setup size values.

Frame Color

Möglich [Schwarz, Weiß, 0x..., Bildfarbe], Standardwert: Schwarz

Range [Black, White, 0x..., Image Color *], Default: Black

Define the color of added image areas (frame). Aside from standard colors Black and White, experts are free to enter any color of the available color space: 0xpp on gray, 0xrrggbb on color images (with pp, rr, gg, bb in range of [00..FF]).

If Image Color * is used, the processor steps through background color inside the image, takes a color near this point, and fills up the background area using this color. It should not be used to fill large areas!

Global

Default Set all parameter back to default values.

OK Saves the changes and closes the dialog.

Abbrechen Closes the dialog without saving the changes made to the settings.

Apply *This button has no meaning here.*

Help Opens the help topic for this dialog

See also Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.9.1.1 Find Rectangle

These parameters describe how non-ideal edges are to be treated. In general the default values provide the best result. So don't change these settings unless you are familiar with the operation or be prompted by us in case of support.

Min Border Pixel

Range [0..255], Default: 5

The border detector starts Min Border Pix points away from the page border. This is to minimize the influence of noise at the scan area margins.

Decrease this value, if the images have extremely narrow page borders.

Min Color Pixel

Range [1..16], Default: 4

Number of points which must exceed, at a stretch, the Threshold value (or must be white in bitonal images) to detect a border point.

Increase this value if the image background is noisy.

Gradient Error

Range [1..20], Default: 2

By definition, all 4 rectangle borders are right-angled. The rectangle detector compares all normalized gradients of the found borders, and finaly processes only those borders, which normalized gradient is less than the given Gradient Error value.

Increase this value if the papers are not exactly right-angled or if one or more border(s) are not straight (which disallows calculation of a fix gradient).

Max. Error

Range [1..99], Default: 3

The optimal image border line will be created by a linear regression of the detected border points. This regression line will be accepted if all point distances to this line is smaller than Max Error. Otherwise, the worst point is removed from detection list and regression process is restarted.

Increase this value if the image borders are not straight lines.

Defect factor

Range [1..99], Default: 4

If the destination rectangle holds dark areas near the border, the border detector will identify a border point inside the rectangle (that is the areas end). To prefer removement of these inner points, their distance to the regression line is weight by Defect Factor.

Increase this value if the images have dark areas near the border and the process cuts too much.

See also Deskew Parameter Setup.

3.7.9.1.2 Modify Rectangle Size

The borders of the detected rectangle can be moved manually by the following settings. A positive value moves the corresponding border out (enlarge rectangle), a negative value moves it in (cut more).

Left, Right, Top, Bottom

Default: 0

Value to move the corresponding border.

Unit

Range: [Pixel, mm, Inch]

Note: Use values in absolute measurement (mm or Inch) to be independent from changes in resolution.

unit2all

Click this button to set units for Right, Top and Bottom value to the unit selected for Left value.

Mirror horizontal values on backside images

Check to exchange Left and Right value when processing back sides.

See also Deskew Parameter Setup.

3.7.9.1.3 Check Rectangle Size

Use this option to protect your images from merciless cutting. If one or more of the following checked rules becomes TRUE, the process will be canceled without changing the image.

No Deskew/Cropping, if ...

Ratio becomes smaller than xxx : 100

xxx : 100
Lower limit for side ratio.

The ratio is calculated as: large side / small side.

Here are some ratio values of standard paper formats:

All A-Formats:	141 : 100
Letter:	129 : 100
Double Letter:	156 : 100
Legal:	165 : 100

Ratio becomes larger than xxx : 100

xxx : 100
Upper limit for side ratio.

See lower limit description above.

Width or Length becomes smaller than xxx %

Border relation limit.

Width becomes smaller than Absolute lower horizontal limit.

Width becomes larger than Absolute upper horizontal limit.

Length becomes shorter than Absolute lower vertical limit.

Length becomes larger than Absolute upper vertical limit.

Unit Range: [Pixel, mm, Inch]

Note: mm and Inch values are independent from changes in resolution.

See also Deskew Parameter Setup.

3.7.9.1.4 Fill Border

The "Fill Border" processor detects background color, starting from the image borders. The detected areas are filled with chosen border color. This is e.g. useful if the paper edge is slightly damaged by tearing. For the repair of larger damages, this method is not suitable.

Filling Limit

Use limit parameters to prevent the images from running the processor to much inside the document. By default, all are unchecked; that is: unlimited processing.

left Left border processing limit.

unit2all Click this button to set units for Right, Top and Bottom value to the unit selected for Left value.

right Right border processing limit.

top Upper border processing limit.

bottom Lower border processing limit.

unit Range: [Pixel, mm, Inch]

Note: mm and Inch values are independent from changes in resolution.

Mirror horizontal values on backside images Check to exchange Left and Right value when processing back sides.

Expert Settings

Min Corner Pixel Range [0..32], Default: 8

For filling it's important to identify the document corners carefully, because fillings from two directions will meet there. Enter the number of pixels that must surpass the threshold subsequently so that a document corner can be supposed. Higher values will better filter out interferences from the background.

Go Inside

Range [0..32], default = 5

After deskewing, the pixel color very close to the image border often do neither belong to the background color, nor match the color of the document. Taking this into account, the detection algorithm will start the given pixels inside the image.

See also Deskew Parameter Setup.

3.7.9.1.5 Adjust Image Size

After cutting of the black borders the image can be resized bach to a given format:

Resize Image to

Check to size-up the image to a specified format.

Original Image Size

Select to bring the image back to the size it had before deskewing/cropping.

Format

Select to bring the image to the specified absolute format.

From selection list, choose a standard format, or choose "Custom Size" and enter image dimensions (unit [Pixel, mm, Inch]).

Landscape

Check to select landscape size of a standard format.

Resize Mode

Choose, how to bring the deskewed/cropped image to the selected size.

Add Frame:

Borders are added arround the image; or, in other words, the image is placed into a cleared image of the selected size. See position parameters below.

Scale Image:

The image is scaled into the selected format. To prevent from undesirable distortion it's recommended to enable "**Skip Resizing, if Size Relation exceeds ...**" (see below).

Horizontal

Range [Left, Horiz. Centered, Right] default: Horiz. Centered

Available if resize mode "Add Frame" is chosen. Choose, on what horizontal position inside the new size the deskewed/cropped image should be placed.

Vertical

Range [Top, Vert. Centered, Bottom] default: Vert. Centered

Available if resize mode "Add Frame" is chosen. Choose, on what vertical position inside the new size the deskewed/cropped image should be placed.

Skip Resizing, if Size Relation exceeds ... %

If checked, the resize process is canceled if size relation exceeds the given value.

It's urgently recommended to enable this inspection with resize mode "Scale Image".

Add Borders

Check to add one or more fix-sized borders to the image.

Left

Size of border to add to the left side of the image.

unit2all	Click this button to set units for Right, Top and Bottom value to the unit selected for Left value.
Right	Size of border to add to the right side of the image.
Top	Size of border to add to the upper side of the image.
Bottom	Size of border to add to the lower side of the image.
Unit	Range: [Pixel, mm, Inch] Note: mm and Inch values are independent from changes in resolution.
Mirror horizontal values on backside images	Check to exchange Left and Right value when processing back sides.

See also Deskew Parameter Setup.

3.7.9.1.6 Combine Front and Rear Side

The double side deskew processor increases deskew quality when scanning both sides: Front and back side images of one scan show the same document (the rectangle to found) with the same size and the same skew. So it's save to detect the rectangle on any image, and to process process deskew on all images by these result.

Sometimes it's difficult to detect one (or more) borders of the rectangle, because the background color is used in large parts of the original document near the border(s). The best solution is, to detect both side rectangles, align the results and use them to de-skew.

Front/Back Side Rectangle

Define independently for each side, where to detect the rectangle:

- Search Front Side
- Search Back Side
- Search Both Sides, take Best Result (default)

Offset

If there is a static difference in front and back side document position in scanned images (because scanners front and back side camera are not mounted at the same position), this can be corrected by offset setting.

It's recommended to make settings in absolute measurement (mm or Inch) for to be independent from chosen resolution.

Horizontal	Enter a positive value to move the back side to the left (if using front side rectangle to deskew back side).
Vertical	Enter a positive value to move the back side upwards (if using front side rectangle to deskew back side).

Tolerance

It may happen when aligning both side detection results, that the rectangles differ a little in size or skew angle. The result is accepted anyway, if the differences are in the given tolerance.

Size	Range [0..10.0%], default: 2.0%
Angle	Range [0..10.00°], default: 0.50°

See also Deskew Parameter Setup.

3.7.9.2 Image Improvement

For further image processing, only the Invert function is currently available:

Invert Black & White Images (before Deskew)
Check this box to invert black-and-white images.

If inversion is enabled, it is done before any other function of the Image Processor (as Deskew, etc.) is done.

See also Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.9.3 Split Images

This dialog shows all defined split images in a list box and allows split image managing.

New...
On pressing this button, a new split image parameter set is created and the Split Parameter Editor opens. Define parameters of this additional image split there.

This function is not available, if the maximum count of supported splits is obtained.

Edit ...
Available, if one split definition line is marked in the list window. Pressing this button opens the Split Parameter Editor, where the split parameters can be changed.

Delete
Available, if one or more split definition lines are marked. Pressing this button deletes all marked split definitions.

Close
Closes the dialog.

See also Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.9.3.1 Split Parameters

In this dialog, the origin, dimensions and other characteristics of a new sub-image can be set.

Name
Internal assigned name of this parameter set.

Description
An optional user defined text describing this parameter set.

Source Image
Source image where the split image is taken from, e.g. the color image from the front camera.

Must exist
Defines handling if source image does not exist (while runtime) :

- if checked, an error is generated;

- if not checked, the split process is skipped / no split image is generated.

Horizonta

Start Defines the left border position of the split image inside the source image. The position can be specified in absolute units or relative to the source image width (in percent).

from The left border position of the split image can be defined relative to the left or to the right border of the source image.

width Width of the generated image.

The image generation will be stop at the right border of the original image, if the given value is too large.

min Specifies that the cut-out must have a minimum width. If the image is smaller than the predetermined value, an error message comes up. So the image will NOT be padded.

This allows to give a relative width in percent and to specify a minimum as an absolute value.

Vertical

Start Defines the top border position of the split image inside the source image.The position can be specified in absolute units or relative to the source image length (in percent).

from The top border position of the split image can be defined relative to the top or to the bottom border of the source image.

length Length of the generated image.

The image generation will be stop at the bottom border of the original image, if the given value is too large.

min Specifies that the cut-out must have a minimum length. If the image is shorter than the predetermined value, an error message comes up. So the image will NOT be padded.

This allows to give a relative lengt in percent and to specify a minimum as an absolute value.

Assign Side Info Choose, what side attribute (front or back) gets assigned to the generated image

Generate Image Header If checked, a special Image Header will be assigned to the generated image.

If creation fails, suppress error reporting If checked, and generating the image fails, no error message is generated.

Note: Due to a missing image, however, error messages may occur in the further program sequence

See also Image Processor, Scanner Configuration and Split Imagesr

3.7.9.4 Combine Images

This dialog shows all defined combined images in a list box and allows combine image managing.

New... On pressing this button, a new combine image parameter set is created and the Combine Parameter Editor opens. Define parameters of this additional combined image there.

This function is not available, if the maximum count of supported combined images is obtained.

Edit ... Available, if one combine image definition line is marked in the list window. Pressing this button opens the Combine Parameter Editor, where the combined image parameters can be changed.

Delete Available, if one or more combined image definitions are marked. Pressing this button deletes all marked definitions.

Close Closes the dialog.

See also Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

3.7.9.4.1 Combine Parameters

In this dialog, the origin, the position and other characteristics of a new combined image can be set.

Name Internal assigned name of this parameter set.

Description An optional user defined text describing this parameter set.

Source Image 1 First source image; gets placed at the upper/left position.

Source Image 2 Second source image; gets placed at the lower (vertical mode) or right (horizontal mode) position.

Both must exist Defines handling if one of the source images does not exist (while runtime):

- if checked, an error is generated;
- if not checked, the combine process is skipped / image is not created.

Combine Direction Choose the direction to combine the source images. Select of:

- Vertical Images are placed one below the other.
- Horizontal Images are placed side by side.

Background Color Choose a color to fill areas of the created image, which are not filled by source images.

Size Choose, how the size of the created image is calculated:

Auto The image size becomes just as large that both source images fit in.

Max-Doubled The created image becomes twice as large like larger source image size. This format allows easy restoring of the source images by cutting in the middle.

Assign Side Info Choose, what side attribute gets assigned to the created image.

If creation fails, suppress error reporting

If checked, and image creating fails, no error message is generated.

See also Image Processor, Scanner Configuration and Combine Images.

3.7.9.5 Returned Image Sequence

In this dialog, the order in which the images are returned to the program can be specified..

Dropped Images

The images in this list are not returned to the scan program.

Returned Images

The images in this list are returned to the scan program in the given sequence: the image listed in the first line is returned first, the one in the next line then, ...

This list must contain at least one image!

Regard when running with older DPUSCAN versions: in order to avoid disturbances, first return all front side images, then the back side images!

Add

Moves all marked drop list images to the return list.

Remove

Moves all marked return list images to the drop list.

Up

Moves all marked return list images one line up.

Down

Moves all marked return list images one line down.

Separator

Inserts a separation mark in return list. This mark can be evaluated by application.
(This functionality is not available with all applications.)

All Images must exist

Specifies the procedure, if at run-time (while scanning) one or more images of the return list do not exist:

- if checked, an error is generated;
- if not checked, missing return list images do not generate an error - these images are just skipped.

Simulate

Range: [Disabled, Single Side Scanner, Double Side Scanner], Default: Disabled

If *Simulation* is enabled, the images defined in return list are sent to the application in groups. This is usefull, if a set of additional images is generated (Split Image, ...) for to make processing by application easier.

A group is built of one side (*Single Side Scanner*) or two sides (*Double Side Scanner*).

Each side is built of one to three images with different color type (black&white, gray, color).

All images in the return list must exist, and they must be sorted in a way that allows grouping. It is recommended, especially in Double Side Simulation, that the images have Front/Back side attribute set in correct manner.

See also Paper, Miscellaneous, Endorser, TWAIN-Scanner, ISIS-Scanner, Other Scanner, Index, Image Processor and furthermore Scanner Configuration, Scanner Selection and Base Profiles

4 Technical Notes

In this part of the manual you will find keyboard commands, command line parameters and variables.

4.1 System Information

In the main menu can be found under the help point the system information. This opens up a dialog from which you get important information about the installed hardware to the individual software modules and to set profiles.It also allows changes to some general settings, and moreover, there is a profile management.

The dialog has tabs for different functional areas. Depending on the mode, not all tabs visible..

Configuration	Installed soft- and hardware
Application	Basic program settings
Logging	Logging of certain processes
Profile Management	Local management of profles
Processor	List of batches processed external
Finalizer	List of batches finalized external
Logfile	Filtered view of the log file

See also help overview

4.1.1 Configuration

This tab displays the versions of the software used in a tree structure.
The branches DpuMod32 and ScanSub also contain information about the connected scanner, if possible.

□ Please deliver ready this information when you contact the hotline.

See also system information and help overview.

4.1.2 Application

The behavior of the program is dependent on both the use of profiles and profile-cross settings. Profile-cross settings should be changed only on the advice of technical support.

In most applications, the default settings are to be used. Only in exceptional cases, you are prompted to change this parameter.

□ Change parameters of this dialog please only after an explicit request

Please note that changes in this dialog affect all profiles on the computer. Settings from this dialogue are computer-specific and are not transported in an export / import process or when you publish a DpuServer on a target computer.

After changes it is wise to quit and restart the program.

See also system information and help overview.

4.1.3 Logging

The program logs always basic actions and possible errors in a log file. When analyzing a scan process, it may be necessary to track the behavior of the program more accurately. This can be seen on the Logging tab set.

Please change the parameters of this dialogue only after an explicit request, as a log can be very extensive.

In general, it is sufficient for the analysis to choose the basic settings. However, it can be useful on the advice of the Hotline, added switch other parameters.

When the analysis is complete, the state **Minimum** should be selected definitely be back.
If you want to discard changes, so you can do this with the **Restore** button.
If you have made changes, this will not take effect until you click **Save**.

Please note that are more possibilities to trace the process, e.g. , the results of the event rules can be displayed in the alert box.

See also system information and help overview.

4.1.4 Profile Management

This tab shows the existing profiles at the station. There are various types of profiles, depending on how the program is operated.

The application uses a separate memory module, the DpuLocalServer. This service stores all configuration data.

In addition, this storage module receives configurations from a DpuServer and makes them available locally. So configurations can be maintained at a central location. Then they can then be reloaded on the individual scan stations each time the scanning application starts automatically.

It can be set how the program intends to confine the profiles of various users against each other. This can be done in a setup dialog during the installing, or later in the DpuServer Settings in the system's control panel.

Common profile storage for all users

This is probably the most common case, for example, when the computer is used by one person. Here you can see all users all available profiles and can use and change them.

🖥 The monitor marks a profile that has been created for all users of the computer. Everyone can choose, use and modify it.

? The question mark stands for the system profile, that is the default values that are accepted if previously not been set.

Profile storage per user

When this mode is selected, each user sees only his own profiles. In addition, he sees even those that have been copied in a common area.

In the corresponding setting dialog, a private profile can be **published locally** and, vice versa, a published profile can be **copied** to the local area. Likewise own and shared profiles can also be **moved** or **deleted** entirely.

The person icon represents a profile that has created the currently logged in user of the computer. Other people do not have access to this profile.

The monitor is a profile that was created by a specific user of the computer and then published locally. After publishing it can be used any other user.

The combination of person and monitor stands for a profile that was created for all users of the computer and was subsequently modified by the logged in user. So there are now differences from the version published originally.

DpuServer profil storage

If this storage mode was elected, there is no local common area. Instead, the shared profiles can be copied on the DpuServer (published) and downloaded (adjusted) from there automatically. The publishing can be done using the DpuEnterpriseManager. Even in this mode it is possible to copy own profiles and shared profiles as well.

But only own profiles can also be deleted. Shared profiles can only be erased by the DpuEnterpriseManager.

The world icon is for a profile that was loaded from the server.

The person icon represents a profile that has been created by the current user. Other users can not see or use this profile.

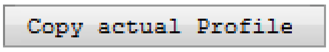
The combination of person and world is a configuration that was initially loaded from the server and was subsequently modified by the logged in user. So there are now changes from the original version of DpuServer.

The button for this profile management can found on this tab, the meaning of the buttons are:

The Profile / Subprofile is stored in the user region.

The Profile / Subprofile is stored in the common (user-independent) region.

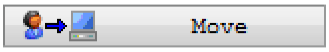
The subprofile is used in several profiles.



Copies the currently selected profile. All sub-profiles associated with the new profile have the same name so that an assignment of the sub-profiles for task or base profile is easy. The source data will remain unchanged, they will not be deleted.

Because all sub profiles be created, the resulting copy of the profile can be changed without affecting the original.

On the other hand, "speaking" subprofile names get lost and are replaced by the new profile name-number, e.g. can the macro "Forward to QSI" copied "INVOICES_05".

	Removes the currently selected user profile,
	Deletes the currently selected user profile,
	Copies the user profile to the common region.
	Copies the common profile to the user region.
	Moves the user profile to the common region, the former user profile will be removed.
	Moves the common profile to the user region, the former common profile will be removed

See also system information and help overview.

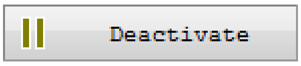
4.1.5 Processor

Using a DpuProcess version of the program can shift time-consuming operations on a downstream process. So you can, for example, do the image rotation and the OCR in the background. This tab lists all batches which are currently be processed in the background, or waiting for that.

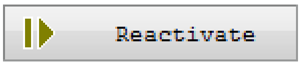
The list shows for each batch:

Status Processor	Shows the working state: WAITING (waits for processing) WORKING (is on work currently) PROBLEM (deactivated or locked by an error)
Time	Time when this batch was sent to the DpuProcess queue
Job	name of the batch
Task	Task profile which is used to process this batch. If this task isn't available it will be displayed red.
Base Profile	Base Profile which is used to process this batch. If this base profile isn't available it will be displayed red.
Station	Name of the Computer working on this batch.
User	Name of the Name working on this batch.

The buttons below the list work this way:

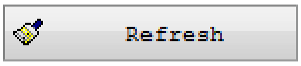


Marks the selected batch to be ignored by the DpuProcess. It gets the state PROBLEM.



Removes the previously set mark , so DpuProcess can process this batch

Removes the batch form the working list for DpuProcess. The batch itself remains unchanged in the OpenJob-Folder, it will be NOT deleted.



Updates the List.

See also system information and help overview.

4.1.6 Finalizer

Using a DpuFinalize version of the program can shift the finalize operation to a background process.

The list shows for each batch:

Status Processor — Shows the working state:

WAITING (waits for processing)

WORKING (is on work currently)

PROBLEM (deactivated or locked by an error)

Time — Time when this batch was sent to the DpuProcess queue

Job — name of the batch

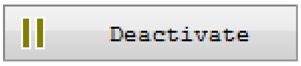
Task — Task profile which is used to process this batch. If this task isn't available it will be displayed red.

Base Profile — Base Profile which is used to process this batch. If this base profile isn't available it will be displayed red.

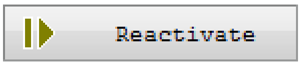
Station — Name of the Computer working on this batch.

User — Name of the Name working on this batch.

The buttons below the list work this way:

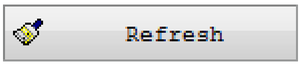


Marks the selected batch to be ignored by the DpuFinalize. It gets the state PROBLEM.



Removes the previously set mark , so DpuFinalize can process this batch

Removes the batch form the working list for DpuFinalize. The batch itself remains unchanged in the OpenJob-Folder, it will be NOT deleted.



Updates the List.

See also system information and help overview.

4.1.7 Logfile

This tab displays a log file. Either the current file are displayed or the log file from another computer. It is possible to filter the lines and store the result.

□ Depending on the selected logging conditions Log files can be very large.

Therefore, a file will only be displayed if Load is clicked. The loading process can take some time to complete.

The log file is displayed in a list with these columns:

ID

Shows the process type of the program:

`DPU`	Entries of DpuScan
`F01`	Entries of DpuFinalize
`P01`	Entries of DpuProcess

DpuFinalize and DpuProcess could be started several times, in this case the number indicates the logging program: P01, P02, etc.

Time

Time stamp of the line

Line

The text itself.t.

The buttons works this way:

Loads the own log file, or searches for a foreign one.

Saves the log file or opens a dialog to select another folder.

Searches for the next line holding the given text.

Displays onla lines holding the given text

In the search or filter field plain texts can be entered, e.g. `error` in this case only lines will be shown holding an error (usually error messages from the scanner)

Also regular expression can be used like `[/Start.*INVOICES]` Here all lines are listed where the task INVOICES has been started (but no other tasks) and not those lines working (but not starting) this task.

See also system information and help overview.

4.2 Shortcuts

In Officemode and Expertmode the program can be controlled via the keyboard. A keyboard command triggers the same action as the cklick on the corresponding button in the toolbar.

The shortcut will work nevertheless the button is visible or not. Because of this the default shortcuts will be available in Officemode too.

Shortcuts are assigned to the corresponding buttons on the toolbars. Buttons on the ribbon control are not accessible this way in the moment.

Because the buttons are not accessible in Officemode there is not chance to configure them. On the other hand in Expertmode the can be set up in the application profile.

As long as unchanged by the user this shortcuts will work

Space	Start, hold and continue scanning
Shift+Ctrl+Tab	Switch to the next window.
Shift+Backspace	Show or hide Info Window
Return	Start of the Task
NUM 9	Image corner top right
NUM 8	Move image up
NUM 7	Image corner top left
NUM 6	Move image right
NUM 5	Bild einpassen in Fenster
NUM 4	Move image left
NUM 3	Image corner bottom right
NUM 2	Move image down
NUM 1	Image corner bottom left
NUM +	Zoom in
NUM -	Zoom out
Entf.	Delete the image
Ctrl+Tab	Switch to the previous window.
Ctrl+Backspace	Show or hide the temporary Scanner Settings
Ctrl+Alt+Backspace	Show or hide the Warnings
Ctrl-A	Select all
Alt-Return	List of the defined tasks.
Alt-+Backspace	Show or hide the Status file.

4.3 Command Line Parameters

At start value can be assigned to some field codes at the program start of the program via the command line. For these field codes, keywords have been defined.

The program invocation can have the following form:

DpuScan *cmd1=value1, cmd2=value2, ... , cmdn=valuen*

The parameter pairs are separated by commas or spaces. Values that contain spaces must be set in quotation marks. Values can be fixed texts or variables, e.g. environment variables. *Cmdn* is to be replaced with one of following keywords.

Some of the command line parameters can be used in the program as variables, some them will appear in the statistic file automatically.

Operating Mode

OFFICE

Starts the program in Office-Mode. If this parameter is omitted the program start in Expertmode.

Example:

```
DpuScan OFFICE
```

Environment

LANGUAGE

Sets the language for the user interface and the messages of the program.

07 = German

09 = English

0A = Spanish

0C = French

0E = Hungarian

19 = Russian

1D = Swedish

The language can be referenced by the variable %(S.LANGUAGE).

Not all versions of the program support all languages. If a language is not supported, the dialog boxes of the program are displayed in the language of the operating system or in English.

Example:

```
DpuScan LANGUAGE=0C
```

Actions

With the following parameters the program can started in a desired mode. These parameters neither have a corresponding variable in the program nor will be reported in the statistic file.

STARTTASK

Using this parameter, the program executes the actual task. This task can be selected simultaneously by using the parameter TASKFILE. The parameter STARTTASK is equivalent to pressing the go button.

Example:

```
DpuScan TASKFILE="INVOICES 200" STARTTASK
```

FINALIZE

Starts DpuFinalize instead of the scan application. In this mode the program finalizes all batches scheduled for the currently loaded task. There is no further user interaction required.

Example:

```
DpuScan FINALIZE TASKFILE="MAILROOM"
STARTTASK
```

PROCESS

Starts DpuProcess instead of the scan application. In this mode the program works without a user interaction and processes all batches scheduled for the currently loaded task. In comparison to FINALIZE the mod PROCESS allows more task steps, like image enhancement or recognition.

Example:

```
DpuScan PROCESS TASKFILE="OCR AND PDF"
STARTTASK
```

Rem.: To run in operation modes DpuFinalize or DpuProcess a special license is required . If this license is missing, the command line parameter will take no effect. The program's Gold license includes a DpuFinalize license, for all other program licenses it has to be ordered separately. DpuProcess has always to be ordered.

Profil Selection

Directly at the start the profile can be given.

TASKFILE or

TODOFILE

TASKFILE defines the name of the task profile to be loaded at start. The task will be loaded but not started at begin. To start the task immediately use the parameter STARTTASK.

The parameter can be referenced as %(S.TASKFILE) or %(S.ACTUALTASK) or %$d

Column 4 in Statistic File.

Example:

```
DpuScan TASKFILE="MAIL STANDARD"
```

Rem: The obsolete parameter CLASSFILE will be ignored, because the task loads a base profile at the begin always. The formerly assigned variables %(S.ACTUALCLASS) and $A will hold the name of the current base profile.

Variables of the Jobdata Dialog

The following parameters can be set at the Jobdata Dialog in the base profile. They are additional descriptions of the working place and the kind of documents:

STATIONNAME

This parameter describes the location of the scan place. It has nothing to do with the system's variable %(SYS.COMPUTERNAME) which is set in the computers control panel.

STATIONNAME can be referenced as %(S.STATIONNAME) or %$S

Column 2 in Statistic File.

Example:

```
DpuScan STATIONNAME="XINO_2"
```

USERNAME

This parameter describes the user of the scan application. It has nothing to do with the logged on person on the computer which can be referenced in the system's variable %(SYS.USERNAME),

USERNAME can be found as %(S.USERNAME) or %$U

Column 3 in Statistic File.

Example:

```
DpuScan USERNAME="JOHN DOE"
```

DOCCLASS

DOCCLASS is an additional description of the document type. It doesn't refer to the base profile name, see above.

This parameter can be referenced as %(S.DOCCLASS) or %$c

Example:

```
DpuScan DOCCLASS="SHIPPING NOTE"
```

JOBCOUNT

JOBCOUNT sets the number of jobs ran an the machine. The counter ha a value from 0 to 999,999,999 and will be increased at each start of a job automatically. So to use a job count of 100 the command line parameter has to be 99.

JOBCOUNT can be referenced as %(S.JOBCOUNT) or %$J

Example:

```
DpuScan JOBCOUNT=99
```

TASKNAME oder
TODONAME

TASKNAME is additional description of the task, not to confuse with the current the task profile, see TASKFILE above.

It can be referenced as %(S.TASKNAME) or %$D

Example:

```
DpuScan TASKFILE="INVOICES 300"
TASKNAME=GTC
```

Additional Parameters

Moreover the parameters above there is a set of nine private parameters which can be set at the command line too.

CMDPARAM1 to CMDPARAM9

This parameters can be used by the users discretionary.

They can be found as %(S.CmdParam1) to %(S.CmdParam9)

Column 21 to 29 in statistic File.

Example:

```
DpuScan CMDPARAM1="On the fly"
```

USERPARAM1 to USERPARAM9

This parameters can be used by the users discretionary too.

They can be found as %(S.USERPARAM1) to %(S.USERPARAM9)

In comparison to the CMDPARAMs ther aren't listed in the statistic file.

Example:

```
DpuScan USERPARAM1="DRINGEND" USERPARAM2=%ALLUSERSPROFILE%
```

Rem: Here USERPARAM2 holds the path to the common program data, usually "C:\ProgramData"

Special Commands

The ribbon bar of the program can be hidden. If this happens in too many profiles, it may be that the operator "excludes himself". To regain access, there are these parameters:

FORCERIBBONON

Shows the ribbon bar, the current application layout profile is changed accordingly, i.e. the check for menu bar is set.

Example

```
DpuScan FORCERIBBONON
```

FORCERIBBONOFF

Hides the ribbon bar, the current application layout profile is changed accordingly, the check for menu bar is unset.

Example:

```
DpuScan FORCERIBBONOFF
```

See also help overview

4.4 Glossary

4.4.1 Administrator

When working with the program two groups of persons can be decided: Administrators and operators.

Administrators are well acquainted with the program, can deny access to some functions of DpuScan for other persons (Please refer to the chapter "user profiles" in the manual) define Base Profiles and Task Profiles.

4.4.2 Barcode Direction

Return Value	Text	Reading direction	Image rotation	Range

1	Abc	from left to right	~ 0°	Ca. 337,5° - 22,5°
2	Abc	from left top to right bottom	~ 45°	Ca. 22,5 -° 67,5°
3	Abc	from top to bottom	~ 90°	Ca. 67,5° - 112,5°°
4	Abc	from right top to left bottom	~ 135°	Ca. 112,5° - 157,5°
5	Abc	from right to left	~ 180°	Ca. 157,5° - 202,5°
6	Abc	from right bottom to left top	~ 225°	Ca. 202,5° - 247,5°
7	Abc	from bottom to top	~ 270°	Ca. 247,5° - 292,5°
0	Abc	from left bottom to right top	~ 315°	Ca. 292,5° - 337,5°

4.4.3 Barcode Serach

The successful search for barcodes assumes that the black bars and white spaces are **at least two pixels wide**. While this does not have to be guaranteed on all the height of the bar code, but at least in a portion thereof. The barcode must not too close to the edge or another line (for example, table) are, it needs a "quiet zone".

The direction of the barcode on the sheet doesn't matter, also inclinations are well recognized. Also up to a degree extent variations in the width of the individual lines can compensated.

Overall, the bar code should be uniform but, that means when printing and scanning, the paper should be transported at a constant speed, without "slipping".

Common reasons why a bar code is not found, are:

- It's searched on the wrong image (BW scanned, searched on Color)
- It's searched in a certain frame on the page but the code is outside (partikular).
- The barcode doesn't have start and stop code (e..g. at Code39 an asterisk * at begin and end)
- It will be searched for the wrong type of code (e.g.. Code25 with check sum instead Code25 without)
- The barcode is printed too fat, the bars "stick" together.
- The barcode is printed regularly on the paper, but looks too fat in the image (check contrast)
- The barcode is printed too sparsely, the bars doesn't reach the the width needed.
- The barcode is printed regularly on the paper, but looks too thin in the image (check contrast)
- The black bars of the bar code are "excavated" (check contrast)
- The barcode is overall too small (Check resolution, 200 DPI at least, better 300 DPI)

The **result** of the barcode search is stored in the variable **%R** also **%(BARCODE)** or **%(S.BARCODE)**. If no barcode was found, **%R** contains exactly one comma.

Up to 99 Barcodes can be found in one image. If there are more than one, they will be separated by commas.Watch out: Some types of barcode may contain commas themselves

In addition, the variables **%R1** to **%R9** (**%(BAR1)**,... or **%(S.BAR1)**,..) will be filled, for each barcode one variable. That means the *first nine* codes can be addressed separately.

If at least one barcode was found also the **Event Barcode found** is fired.

Also the parameters **%(S.BARDIRECTION1)**,... and **%(S.BARINFO1)**,...will be set. They inform about position, type and angle of the code.

The last found values will be stored in the variables **%$R** also **%(BAROLD)** or **%(S.BAROLD)** resp. **%$R1**,... also **%(BAROLD1)**,... or **%(S.BAROLD1)**,... These values will be kept and copied from one image to the next until new codes will befound.

If a barcode was found which is particularly incorrect, e.g.a checksum or a start/stop-code is missing , the variable **%(S.BARPOSSIBLE)** will be filled.

Attention: Beside the embedded search routine of the program external program libraries can be called to search for barcodes, e.g. the PlugIns SofBar or PixBar.
These modules uses their own variables , but not **%R**, **%R1...** Also the **Event Barcode found** will not be triggered!

But it's possible to set (or overwrite) the values of **%R**, **%R1**,... This variables aren't write protected. At the same time as **%R** or **%R1** will be set, the **Event Barcode found** will be triggered too.

4.4.4 Barcodes

Barcode: A barcode is a representation of a text coded as a sequence of stripes and gaps in between the stripes. Stripes an gaps may have a different width, depending an the used barcode type. The content of a barcode can be an alphanumeric text or numbers only, also depending on the type. To avoid errors when decoding a barcode they may have special characters at begin and/or at the end.
For the same reason they may have a check digit somewhere embedded in the text.

Barcode are often used to mark the begin of a document

4.4.5 Base Profile

Base Profile:Within the program a base profile refers a set of configurations. It describes parameters, which will be kept for a particular function.

Base profiles contain basic settings for

- Data source: resolution, paper size, ...
- Data target: target path, file format, ...
- Process: rules for document separation, parameters for recognition and image processing
- View: number and properties of scan windows etc.

Often this settings are grouped and stored as SubProfiles.

Once defined, base profiles can be copied or extended in any way. They can be reused in different tasks for similar batches.

4.4.6 Bates Stamping

Bates Stamping: The process to merge a text or a graphic into the scanned image is called Bates Stamping. This can be used to state the document was scanned in a certain company place on a certain station at a certain time. For digital images this is the equivalent for endorsing a paper during the scan by an embedded printer head The name Bates goes back to a stamp with mechanical index and its inventor Edwin G. Bates..

4.4.7 Binarization

Binarization means the transforming of a gray image with a color depth of 8 bpp to a black/white image with 1 bpp. The resulting image will refered to as "bitonal image" or "monochrome image" also.

Scanners usually have a color camera, in rare cases a gray camera. Those cameras create color or gray images. On the other side only black/white images are used for the most common recognition processes like barcode search or OCR. Also scanning business correspondence or similar documents a good readability is more important than a precise reproduction of color tones. Hence the most scanners have a built-in binarization and offer a bitonal output as well as the color one.

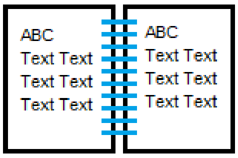
The program can handle scanner-created bitonal images as well as creating own bitonal images by filtering them from color images.

4.4.8 Binding

The binding describes how the orientation of sides is related.

Book

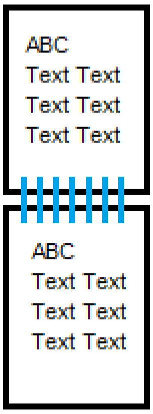
This is the most often case for documents to scan,



For this type the Rotation can be set to the same value for front and rear side.

Folding block

This is type is used for tables or long lists.



For tis type the rotation has to be set with a 180° difference for front and rear side, e.g. 90° Front, 270° Back.

4.4.9 Bitmap

A Windows Bitmap is a computer graphic format. It consist of a fixed format data block called Bitmap Header, an optional palette and the image data Nevertheless there are options to compress the image data a bitmap is in the most cases uncompressed. That allows a fast display but causes a a longer lasting transfer.

Bitmaps are usually stored in in bitmap files which has a special bitmap file header and the ending .BMP.

4.4.10 bitonal

Bitonal is here a synonym for "black/white" or - sometimes - "monochrom"

4.4.11 Brightness

Brightness: A high brightness results in an overall brighter image, a low value results in an overall darker image.
If the documents to be digitized are dark, the darkest area can be given as black again, even if it has different shades of gray. By increasing the brightness this gradation can be made visible. If an image is reproduced almost entirely white, the brightness should be reduced.

4.4.12 Broker

Broker: A broker is a mediator between different systems. The usage of a broker is a method to send data to another software application during the scan process. The corresponding application for example may be archiving systems, document management systems or DBMS.

A broker can be a either a DLL with a special interface or a Visual Basic Script with a special function. In the manual you find a more detailed description of the interface and samples for both, DLL and VBS

It can receive

- Broker events
- Texts in form of variables

But in comparison to PlugIns it cannot deal with images.

A broker can send a return value and a broker message

A versatile sample broker called BRO_UTILS is shipped with the program. It can be found in the BRO subfolder of the program, together with the VBS-Script and an informational text.

4.4.13 Broker Mode

In Broker Mode a PlugIn receives the Broker Events an reacts on them. Usually there is no further user action required, the PlugIn acts like a broker.

Compare with Interactive Mode and Process Mode

4.4.14 Broker Return Values

The broker can return information as a result of a broker call in two ways

- numerical return, presented by %(S.BrokerStatus)
- text string, presented by %(S.BrokerString)

Return value 0x9999
If the broker returns 0x9999 (decimal 39321) the program accepts data in **`%(S.BokerString).`** Hence it works as input and output parameter as well. The returned text may not exceed 9000 characters.

Return value greater than or equal to Zero
All other non-negative values are stored in **`%(S.BrokerStatus)`** and can be used to control the further workflow or output. The broker parameter is treated as input value only and it will be assumed %(S.BrokerString) wasn't changed by the broker.

Return value less than Zero
If the broker returns a negative value it will be treated as an error.

4.4.15 Broker-Events

Broker Events

With the first image group

[01]

This line is sent to the broker only once during the first scan. Here, you can send title information like batch name, station name, start time, or any desired start text.

For every image group

[02]

Here you can place information which are important for one side.

This line is sent to the broker after every scan for front side and rear side separately. But there is only one call for all images of one side (filtered copies, cut-outs,...).

After the last image group

[03]

This line is sent to the broker file only after the last scan. This gives the possibility to write defined end lines.

As event rule

[04]

This line will only be sent to the Broker when an event has occurred, for that an event mark shall be sent.

The related event must have been defined in the Event Rules, the action must be "**Mark in statusfile or broker**".

This line type is not available for the OpenJob mode!

With file creation

[05]

The defined line is sent after a new file is created and before the first image is stored in the new file.

With file closing

[06]

The defined line is sent after a closing a file, for example a Multi-TIF file. A file will be closed when opening a new file, opening a new folder or finalizing a batch.

With path creation

[07]

The defined line is sent when you create a new path. If a new path is created, a new file is generated inevitably and the new image is stored there.

With path closing

[08]

The defined line is sent if a path is closed. this occurs for example if a new subdirectory is created or the batch gets finalized.

With document creation **- not in use at the moment -**

[16]

With document closing **- not in use at the moment -**

[17]

At task start

[09]

The defined line is sent at the beginning of task, this event always occurs before any other event..

The task start occurs when the Start Task button was pressed in the toolbar or the corresponding menu item is selected .

.

At task end

[10]

The defined line is sent at end of task.

In DirectMode the Task End is reached when

- the last image was written to a file
- the last file was closed
- the last folder was closed
- the batch was closed.

In OpenJob the Task End is reached when a batch was closed, exported or finalized.

So "At Task End" can occur more then once.

With batch creation

[11]

The defined line is sent at batch creation.

After batch finalization

[12]

The defined line is sent **after** the finalization of an OpenJob.

If the command "export then finalize" is used the sequence at the end looks like this:

```
1. ...
2. [03] With the last image
3. [08]       With Path Closing
4. [17]    With Batch Closing
5. [10] At Task End
6. [12] After Batch Finalization
7. [14] After Batch Export
8. [20] At Task Final
```

Before Batch Finalization
[13]

This line will be sent immediately **before** finalizing an OpenJob.This event occurs when the Finalize button was pressed. It occurs too if a corresponding command is executed in the Task or in a macro.

If the command "export then finalize" is used the sequence at the begin looks like this:

```
1. [15] Before Batch Export
1. [17]    With Batch Closing
2. [10] At Task End
3. [13] Before Batch Finalizing
4. ...
```

After Batch Export

[14]

The defined line is sent **after** the export of an OpenJob.

Before Batch Export

[15]

This line will be sent immediately **before** exporting an OpenJob.This event occurs when the Export button is pressed or a corresponding command is executed in the Task or in a macro.

With Batch Opening

[16]

The defined line is sent **after** a batch was opened.

With Batch Closing

[16]

This line is sent **after** a the last path was closed.

Error on Batch Export

[17]

The defined line will be sent to PlugIns with a Broker interface only.

It will be send if the operating system signals an error at the batch export, e.g. disk full, network drive not found, insufficient rights,...

Error at Batch Finalization

[18]

The defined line will be sent to PlugIns with a Broker interface only.

It will be send if the operating system signals an error at the batch finalization.

At Task Final

[20]

This event is fired when all commands of the task list are processed. So it occurs always as the last event.

4.4.16 ConnectionString

A ConnectionString is a character string used to define a connection to a database server.

Its syntax depends on the vendor of the database and could be found in the documentation. In many cases it can be found in the internet too.

Usually it holds the vendor of the database service, the place where this service is running and the name of the database. Depending on the security mechanism a user name or a password is required to enable the connection.

As a precaution the ConnectionString will be stored encrypted always!

For Example the ConnctionString to the Microsoft sample database "Northwind", hosted by the SQLServer Express service, running on the computer MyPC (without any special rights) is like this:

```
Driver={SQL Native Client}; Server=MyPC\SQLEXPRESS;
Database=Northwind; Trusted_Connection=yes;
```

4.4.17 Contact Address

Janich & Klass Computertechnik GmbH
Zum Alten Zollhaus 24
42281 Wuppertal
Germany

Tel: +49 202 2708 0
info@janichklass.com
http://www.janichklass.com

4.4.18 Contrast

Contrast: A high contrast value raises out especially White and Black and preserved only a few intermediates. In low contrast value the gray intermediates at the expense of black and white are highlighted vice versa.
To obtain a sharp image, the contrast should be raised. For softer contours you should choose a lower value.

4.4.19 Cover sheet

Cover Sheet or **Control Sheet:** This is a page of paper which is put at the work preparation place on the top of the batch to scan . It is mostly used as a control sheet for the stack and specifies what to do with the stack, for example, on which scanners and which task it is to scan.

4.4.20 Database

A database is a collection of information and data which are well organized and can be retrieved by a Database-Application.

To exchange data from one database to another the ODBC standard is defined.

4.4.21 DAT-File

DAT-File: A DAT file contains a configuration, i.e. Profiles and subprofiles. DAT files are usually created from the scan program. They include the current task profile, the basic profile loaded in it, as well as all the subprofiles required for execution.

If the DAT file has been created with the Profile Management -> Profile Management function or with the DpuEnterpriseManager, it can also contain *several* profiles, or even *individual* subprofiles.

The DAT file also contains a data snapshot with all version of the installed software as well as the current log file of the program.

The DAT file is compressed and usually does not need to be further packed (zipped).

4.4.22 Deskew

Deskew:
The deskew process straightens images which are skewed and cuts off black borders. The skewing can be caused by a bad paper guidance during the transport through the scanner. De-skewd images are easier to read for the OCR and are smaller in the most cases.

4.4.23 DirectMode

The **DirectMode** is an operating mode that is designed for fast, uncomplicated workflows. A task profile works in DirectMode when no stack is created, i.e. It lacks the steps "Load Batch" or "Import Batch from ...".

In DirectMode the images are scanned directly into the target directory. All automatic processing steps such as improvement, detection or event rules can be performed. Likewise, a text file can be created and a subsequent import can be triggered.

After the last image, the batch is displayed, but it is **not possible** to move or insert images, or to enter index data. It is also possible to delete **only the last** picture.

In return, the time for finalizing is eliminated.

If the images are to be checked or resorted after scanning, or if an OCR or manual indexing is required after scanning, OpenJob is the more appropriate mode.

4.4.24 Dll

DLL: (Microsoft Windows - Dynamic Link Library). A compiled collection of computer functions that is not directly linked with an executable program, as there are normal libraries. These libraries are managed at runtime by Windows, they are available for all running programs simultaneously.

4.4.25 Document

Document: In the context of this program, a document called a belonging together number of images. Typically, the document contains exactly the images that captured the program when scanning a paper document, so for example, all sides of every page of a multipage bill.

The filing of documents in the file system of Windows can take place in one or more files that are in turn in one or more folders. Since the recovery and analysis of documents not usually done by hand, documents scanned are often passed to a document management system. This DMS used next to the pictures mostly a descriptive file when importing the document. This file can be created with the program. Here the beginning of a new document can be identified.

4.4.26 Document level

Document Level: Documents can have different hierarchy levels, they are mapped in the program through the document level.

For example, document can consist of an informal letter, along with a completed application form and different attachments. Using different levels in this form the attachments can be subordinated to the actual writing.
The program provides the ability to set a level at a certain event.

4.4.27 Dongle

Dongle: This is a USB device, which is connected to the computer. It contains license information and identifies the program clearly.

Please use the dongle to an interface that is as "close" connected to the motherboard of the computer, for example, the interfaces on the back side.

A dongle can only work if there is a USB port **available** and the user has **access** to it. Since this rarely is the case with **virtual machines**, is suggested to use a DpuServer there instead for licensing.

4.4.28 DpuEnterpriseManager

The DpuEnterpriseManager, DEM, is a tool to define groups and rules for copying profiles and licenses to the connected stations.
It allows to copy (publish) profiles to a DpuServer.

4.4.29 DpuFinalize

As DpuFinalizer the program can **finalize** OpenJobs in the background. Unlike DpuProcess here only a few actions in the system are available

If an error occurs in a batch, it continues with the next batch.

4.4.30 DpuJobHandler

The DpuJobHandler is a program for managing open job. One can move open batches that are not currently being worked on. Accidentally locked batches can be released with the DpuJobHandler.

In addition, the "Status" property can be set. By querying variables at the job level, e.g. the selection can be restricted when opening a batch in DpuScan.

4.4.31 DpuLocalServer

The DpuLocalServer is a local service that handles all storage operations of DpuScan. If DpuScan works in a DpuServer-Environment it also interacts with a DpuServer on the server machine.

The connection can be defined at the system control panel DpuServerSettings. Once it is installed, the handling of the license and profile distribution can be managed with the DpuEnterpriseManager.

4.4.32 DpuProcess

As DpuProcess the program can **work** with OpenJobs in the background. Unlike DpuFinalizer are here almost all actions in the system are available, eg the call to the OCR.

If an error occurs in a batch, it continues with the next batch.

4.4.33 DPU-S

The DPU-S Document Processing Unit software is a set of libraries that allows to obtain images, edit, search, and save it again.

This layer can connect to a scanner and uses TWAIN or ISIS drivers. Scanners, which are provided with an interface from our company can also be contacted directly. Likewise images can be read from the disk of course.

Images captured this way may straightened, rotated, filtered, converted or packed together. There is the ability to search for barcodes or patchcodes in these pictures. The DPU-S does also the (non-trivial) memory management for image objects.

4.4.34 DpuScan

DpuScan is the most common name of the scan application. It acts with the DpuLocalServer and the DPU-S

4.4.35 DpuServer

The **DpuServer** is a service installed on a Computer somewhere in the network. This service handles the distribution of licenses an profiles. It interacts on the client machines with the so called DpuLocalServer.

It can be managed with the tool DpuEnterpriseManager.

4.4.36 DpuServerSettings

The **DpuServerSettings** is a control panel, reachable from the system settings. It is the visible part of the DpuLocalServer. If you want to use it, be aware to have sufficient rights starting control panels.

It allows to start and stop the service on the local machine and to connect it to a DpuServer if there is one. Also it allows to setup the logging options to watch the communication between DpuScan, DpuLocalServer and DpuServer.

4.4.37 Duplex

The duplex mode means, the scanner capures front and rear side of a sheet simultaneously. This is known as two-sided or **double-sided** scanning. Scanner with this property are called duplex scanner.

Most modern duplex scanners have installed two cameras, there are also few scanners have a turning gadget. Normally the capturing of the rear side can be turned off, then the devices will function as simplex scanner.

Duplex scanning is always useful for documents with information is on the back, for example, Questionnaires. For ordinary correspondence the backs are rarely printed. Therefore, the program offers to remove the blank rear sides.

4.4.38 Endorser

Endorser: The endorser is a printer inside the scanner which prints a given text on the paper. To avoid stains at the paper transport inside the scanner the print often occurs immediately before the paper leaves the housing. To keep the text on paper readable the endorsing often occurs on the rear side. An endorser which prints papers before the camera scans the paper is called **Imprinter**.

This devices stamps the paper; to get an electronic stamp in the image you may use the Bates Stamping function additionally

The endorser text can be set as part of the scanner settings in the base profile.

4.4.39 Expertmode

Expertmode

The expert mode is suitable to set up complex or multi-stage processes. For example, on the one hand a complex document structure to create for a specific target system. On the other hand, the program should be easy to use, that is, the person on the scanner needs no more changes to make.

This focus implies that adjustments are made by an administrator and must be deposited while an operator later completely can concentrate on the scanning or indexing activity. Additional toolbars are used for this purpose, which can be configured as desired. All elements of the ribbon are stored here as buttons.

The program starts automatically in Expert mode, unless otherwise specified..

4.4.40 Export DAT

Profiles an subprofiles can be exported to a so called DAT-File. DAT-Files can be read by another Station to import the profiles. Also they can be used as a backup for the local station.

The DAT contains all settings made in the Task profile, the Base Profile and the Application Layout. Also all external files belonging to the subprofile will be packed too as long they are stored in the program folder or below.

Files aside from this folder will **not** be exported.

Be aware that importing this files on another machine, for example with a different scanner, needs to be checked

4.4.41 Feeder

Feeder or **Hopper:** This device serves to pull each one sheet of paper from the stack and feed the scanner. Scanners with such a device are generally called document scanner, as opposed to flatbed scanners, which have a surface to which the paper is placed.
Depending on the features of the scanner the feeder can be set to automatic feeding or manual feeding. In manual mode, it may have advantages, disable the image buffering.

4.4.42 Finalize Mode

The program works in Finalize Mode when it is finalizing batches automatically. As in Process Mode no user action is required unless an exceptional situation occurs (disk full, error in broker execution)

4.4.43 Foreign barcode

Foreign Barcode: In a stack of documents located on particular papers are barcodes, for example, barcode labels. These barcodes indicate the beginning of a new document, for example, an incoming invoice. Now it may happen that a supplier uses on its own barcodes on the invoice. If these barcodes are from the same type randomly as the expected barcode, according to this undesired break would occur there.

To avoid such errors, it is useful to inspect the content of the barcode for plausibility, so as a certain *length* or a certain *structure*.

4.4.44 ImageAddress

The ImageAddress is a vector of counters for images. These counters counts on special events, e.g. each image, images with patchcodes or if another counter reaches a given value. Like digits in numbers they are grouped for right to left. The ImageAddress is part of the ImageHeader, position 120 to 151.

4.4.45 ImageGroup

Image Group: The Image Group contains all the images for a single side scan.

Example 1:
-> Bitonal double-sided scan, no further processing
The one front side image makes up the Image Group
The one rear side image makes up another Image Group

Example 2:
-> Double-sided dual-stream scanning (bitonal + color), no further processing
The bitonal image and the color image of the front side make up the Image Group
The bitonal image and the color image of the rear side make up another Image Group

Example 3:
-> Double-sided dual-stream scanning (bitonal + color)
-> The front side color image is filtered to suppress certain background shades
-> From the modified color image a filtered bitonal is created
-> From the filtered bitonal image a new image is created by copying out the address field
-> From the filtered bitonal image a new image is created by copying out the account field
-> The color front side image is scaled down save storage space

The front-side image group now contains five images:
The scaled color image
the original bitonal image
the filtered bitonal image
the image with the address field
the image with the account field

The rear-side image group contains two images
The color rear side image from the scanner
The bitonal rear side image from the scanner

4.4.46 ImagePosition

Images in a Image group are indicated by the type and the position. The position indicates the number of pictures of the same type.

4.4.47 ImageType

Images in a Image group are indicated by the type and the position. The type indicates whether it is a color, gray or black and white image.

4.4.48 Import DAT

Profiles an subprofiles can be imported from a so called DAT-File. Import will overwrite existing profiles or subprofiles with the same name.

Be aware that importing a DAT on another machine, for example with a different scanner, needs to be checked.

4.4.49 Info Sheet

Info sheet: An info sheet (also cover sheet, stack cover sheet, stack sheet) is a printed paper that is created on the basis of a template and is placed on the finished paper stack after scanning.

Usually it contains information that will be **determined during** the scan. It is not to be confused with a cover sheet that was previously on the paper stack and may contain information that is **needed for** scanning.

The template file must be in 2007 Richtext format, "RTF". It can be easily created using the Microsoft Wordpad program. It can contain normal text or images and also %-Codes. So it is e.g. possible to update and print the number of documents in the template after scanning.

The printed info sheet is then normally used to be stored together with the scanned paper. Typical information on such a sheet is the location and **date of the scan**, the **number of papers**, or the first and last found barcodes.

4.4.50 Insertscan

Insertscan: It is inserted before the next front image. In multistream always all images are to be inserted. It is therefore not possible, e.g. to insert a single color image only.

Likewise, it is not possible to insert images between the front and rear sides .

In an Insertscan all task steps of the scan loop are executed for the new image. Please note that although the S-Variables are reset, for example, % (S.IMAGE.WIDTH). But the I-Variables (Image variables) are taken from the image, before the actually is inserted. So not the one image that was last clicked!

If the images are stored in individual files, as well as a new file is created in addition to the image. It gets the name of the file prior to which it was added, extended with a ~ date-time stamp. This gives a unique new name.

The batch structure is *not re-calculated* automatically.

If event rules are given, you can do the toolbar command "Process Event rules" execute.

If the file names are generated by using an automatic #-counter, and the command can "Renumber filenames" be used.

4.4.51 Interactive Mode

DpuScan goes into Interactive Mode when scanning or loading has been completed and the images are displayed. This may also be the case if an existing Open Job has been opened, e.g. to continue scanning.

Here, the work area is divided differently, that is certain part of windows and controls are displayed or hidden. Certain commands are now available, while others are no longer available. The operator can now scroll through the images by selecting them with the mouse or keyboard.

Compare with Process Mode and Broker Mode

4.4.52 Macro

Macro or **Usermacro**

A macro is a sequence of toolbar actions, i.e. actions, available as buttons in the Application Layout Profile.

For example a macro can
- redo the event rules
- finalize the batch
- print an info sheet

4.4.53 Marks

Pictures that have certain characteristics, for example, are blank, contain a bar code or a keyword can be marked:

Red marks
Images with red tag will not be written when finalizing the destination directory. However, it is possible to remove previously set mark in the quality control or to set them for other images manually.

Red marks are usually set to **empty rear sides**, but it can also be labeled sheets that were not originally in the stack of paper and were only inserted to separate (Patchcodes, other divider sheets or control sheets).

Blue marks
If still images with blue marks in a stack, it can not be finalized. This is only when the corresponding image is selected and displayed, and the mark was manually removed.

Blue marks are used to draw attention to exceptions, i.e. **errors**. For example, that a barcode is missing; but should actually be on page:

Green marks
Images that are especially **important**, can be highlighted with green marks.
When a stack of documents is composed, which in turn may consist of several images, one can e.g. select the first one of each green.

White marks
White tags serve to mark images **unimportant**. They are not displayed after scanning, regardless of whether they still have other marks.
In contrast images with multiple marks only disappears when ALL markings should be hidden.

4.4.54 Multistream

Multi Stream: A scanner works in multi-stream mode, when multiple types of the scanned image delivered with every scan, for example, a black/white image plus a grayscale image and plus a color image.

A well-known special case is the **Dual Stream** i.e. the simultaneous scanning of the color and the black / white image

4.4.55 OCR

OCR: **O**ptical **C**haracter ecognition; Automatic recognition of printed text in images. Originally an optical method, today released electronically. The recognized text will be made available to other processes for further processing.

4.4.56 Officemode

Officemode

The Office-Mode offers the user a all control elements in a ribbon bar, which are used frequently in the normale office business. For example the scanning of the incomimg mails or the creation of a brochure or the archiving of old folders. It's possible to use special properties like barcode or patchcode to split the scans into documents.

To start the program on office mode just use the corresponding commandline paramter OFFICE.

4.4.57 OpenJob

An OpenJob is a set of folders and files which contains the result of an image capturing process, e.g. a batch of scanned or loaded images.

It consists of the images themselves as well as the meta-data registered during the process like date, time, counters, sizes, barcodes or userdefined indexes

4.4.58 OpenJob Folder

The OpenJob folder is the folder in which the images are temporarily stored. The program stores unfinished batches there and deletes them when finalizing succeded.

The OpenJob folder can be specified as desired, the default is **C: \ OPEN_JOB\ .**

To avoid delays when saving in the network, a **local folder** should be selected on the computer to which the scanner is connected.Warning: Do not use this folder as a destination folder or as a storage location for other files!

Tip: If possible, exclude this folder from the virus scan. The program saves images and XML files in this directory, both are usually checked by the virus scanner.

4.4.59 OpenJobMode

OpenJob Mode: The OpenJob mode is suitable if time-consuming processes such as detecting, controlling and indexing should not be done at the scanner workstation.

The scanned images are saved in a special OpenJob folder and can be picked up by another station for further processing. After the last step, the batch will be copied into the target folder and then deleted from the Open Job directory. This process is known as **finalizing**.

A task containing the instruction "Load Batch" or "Import Batch", operates in Open Mode job.

One possible workflow could be

1. Scan on Scanstation
2. Quality check at the supervisor's place
3. Indexing at other Stations.

If there is no reason for an inspection or post-processing the mode DirectMode is better suited.

4.4.60 Operator

When working with the program two groups of persons can be decided: Operators and administrators.

Usually operators are familiar only with that part of DpuScan that is needed to do a scan job. They may also be able to work with the **interactive post processing** functions like a rescan with other scan parameters.

But they can’t change the basic setting for the automatic upload to the DMS or change the user profile.

4.4.61 Patchcode

Patchcode: A special barcode is used for scanner control or separation within the batch.
A patch code consists of only four bars. Usually they are printed along the edges of the paper.
Patchcodes are read orthogonally to the transport direction only or top-down resp.
So Patchcodes printed horizontally will be recognized, printed vertically ones not.

4.4.62 PatchcodeSearch

The successful search for a patch code assumes that the code is possible exactly 2 cm high and is applied transversely to the direction of reading.

Since patch codes consist of only two thick and two thin black bars, there are only six different combinations. Of these two are mirror images of each other: PATCH1-PATCH6 and PATCH3-PATCHT.

The patch code search will be performed from top to bottom starting at the left edge, so it is found on the top left of the first horizontal code. To speed up the search, so the bar code along the four edges is often printed, so that it does not matter if the sheet is longitudinally or transversely withdrawn.

Common reasons why a patch code is not found, are:

- It searches on the wrong image (SW scanned, searched on color)
- It searches only in a specific image area, but the patch code is (partially) outside.
- The black bars of a patch code are "hollowed out" (Check Contrast)

- The patch code has been reduced or enlarged during photocopying and is not 20mm high now.
- The patch code is rotated by 90 ° (wrong direction).

The **result of the search** is saved in the variable `%T` also `%(PATCHCODE)` or `%(S.PATCHCODE)`.The search returns one of these texts: PATCH1, PATCH2, PATCH3, PATCH4, PATCHT and PATCH6.

Iff a patchcode was found also the event Patchcode found will be raised.

It will be always search for **only one** patchcode, if there are more the search stops after the the first one.

The last found patchcode will be stored in `%$T` also `%(PATCHOLD)` or `%(S.PATCHOLD)` for the further images following the the patchcode page.

The variable `%T` or `%(PATCHCODE)` or `%(S.PATCHCODE)` isn't write-protected and can be set arbitrarily. At the moment the variable is set also the event Patchcode found will be set. From the following image on also the variable `%$T` is set.

4.4.63 PercentCode

Program Variables: Much of the information DpuScan collects during operation are provided to the user as program variables. These variables can be specified in queries, definitions or other issues. Here the notation % (variable name) is used. The leading percent mark at the beginning of the name has meant that these variables are also called **Percentcodes** or **%-Codes**.

A detailed description of the almost 400 predefined variables can be found in the help file to DpuScan. In addition, you can create as many custom variable.

The name of a program variable can consist of any **letters**, **numbers**, **underscores** and **dots**. Uppercase and lowercase is NOT considered program-internally.

Usually the part before the first point in the name defines the scope of the variable and thus its location.

4.4.64 PlugIn

PlugIns are external program parts in form of DLLs. They are embedded to the program by an open interface.
PlugIns can be used for Image Enhancement, Recognition or Meta Data Exchange.

It can

- receive and send texts with variables
- receive and send Images
- react on Broker events
- capture a part of the workspace to interact with the user.

A PlugIn can act in process mode, interactive mode or in broker mode. To optimize several co-working PlugIns, an additional Profiling-Mode can be enabled.

4.4.65 process mode

The program works in Process Mode when it is scanning or working with images from the disk. It works automatically, no user action is required unless an exceptional situation occurs (no paper, jam, ...). A special form of process is restricted to finalization only

Compare with Interactive Mode and Broker Mode

4.4.66 Profiles

A Collection of Parameters is called **Profile**.

There are these types of profiles.

- Task Profile
  describes which job step in which order has to be executed.
- Base Profile
  describes the basic settings like file formats, file names etc.
- Sub Profile
  describes settings related to one topic like barcode recognition, display settings and so on. It will be saved with a separate name and can be reused in many tasks.
- Application Layout Profile
  describes the position and contents of the toolbars.

4.4.67 Program folder

The program directory is the folder where the EXE file of this program is. Since this program can be installed in various forms, under different names, these can also be different directories.

If this program is installed e.g. as DpuScan, then the path on a 64-bit system is

```
C:\Program Files (x86)\JuK\DpuScan\
```

and on a 32-bit system

```
C:\Program Files\JuK\DpuScan\
```

4.4.68 Publish

If DpuScan works in a DpuServer environment, the profiles can copied to the DpuServer, this is called "publishing". From the DpuServer they can be shipped to other connected stations, this procedure is called "adjustment".

The behavior of publishing and adjusting can be managed with the administrating tool DpuEnterpriseManager.

4.4.69 Queuing

Queuing (Image Buffering): When actitivate image buffering documents are scanned and stored on stock. Only when activated, this method it is possible to achieve the full speed of a high-performance scanner.
Please note that between the scanning a document and its display on the screen can pass a few seconds. Likewise, there may be a "lag"
On the other hand, it is possible only when the image buffering is turned off, to stop the operation before the next scan or influence.

4.4.70 Queuing Mode

Queuing: The scanner tries to work **as fast as possible**. For that it uses buffers the images in the machine and on the computer. The scanner doesn't care about the outcome of the further process in the scan application. Usually a production scanner runs with full speed and it's up to the software to shift time consuming steps to a post processing place.

When queuing is disabled, the scanner scans only one image and waits for the next scan command. This may be triggered by a foot switch, for example. This is useful for manual paper feeding when scanning sensitive paper documents, large formats, books or transparencies in so called jackets.

4.4.71 Regular Expression

A Regular Expression is a character string used on search or replacement operations in texts.

It has a search pattern which consists of text and placeholders. In case of a replacement operation it also has replacement pattern which consists of text and placeholders for found texts by the search pattern. It als may have options for the behavior during the search or replacement operation.

4.4.72 Rescan

Rescan: A previously scanned image is selected in the batch and replaced by a new image from the scanner.
The color depth must match, that is, a color image can be replaced by a color image, a gray image only by a gray image and a black and white image only by a black and white image.

In a rescan *no further operations* are carried out, for example, OCR. Please note that although the S-Variables are reset, for example, %(S.IMAGE.WIDTH). However, the I-variables (Image variables) remain unchanged.

Likewise, the batch structure is *not* automatically recalculated. To perform this, use the toolbar command "Process Event rules".

4.4.73 Resolution

Resolution: The resolution determines how many image points will be used for a certain distance. It's given usually in "dots per inch", DPI.

In scanning process there is a distinction between horizontal and vertical resolution. The horizontal resolution (transverse to the direction of the sheet) is determined by the **accuracy** of the CCD camera of the scanner, are usual 200-400 dpi. The vertical resolution is determined by the **speed** with which the paper is passed in the direction of the camera. Since usually vertical and horizontal resolution **are the same**, at a high horizontal resolution, the paper is transported correspondingly slower.

The higher the resolution, the finer the structures are depicted and the more time and space required.

4.4.74 Simplex

Simplex mode means that the scanner scans only the front side. One speaks also of single side scanning.
Scanners capture both sides of the sheet are called duplex scanners.

4.4.75 Singlestream

Singlestream: Unlike Multistream there is **only one** image per paper page.

4.4.76 Stacker

Paper tray (Stacker): This is the part of the scanner, the scanned and ejected papers are collected.

4.4.77 Storagemode

When installing the program or later in the DpuServer Settings it can be specified how the profiles are stored and shared

In the simplest case, "**Common profile storage for all users**" all users see all available profiles and can use and change them.

If the operating mode "**Separate profile storage per user** " is selected, each user sees only his own profiles. In addition, he sees even those that have been copied in a common area. The tab Profile Management allows a private profile to be published locally and - vice versa - a locally published profiles to be copied in the private area. Likewise own and shared profiles can also be moved or deleted entirely.

If the mode "**DpuServer profile storage**" has been selected, this common area doesn't exist. Instead, the shared profiles be copied to the DpuServer (published) and automatically downloaded from there at start of the program. A profile can be publish by the DpuEnterpriseManager,

Even in this mode it is possible to copy private and shared profiles. Own profiles can be deleted as well, but shared profiles can be deleted only by DpuEnterpriseManager.

4.4.78 SubProfiles

Sub Profiles: Sub Profiles are subsets of settings for a certain processing step, e.g. the scanning, filtering, or the barcode search. Subprofile are independent and can be used in several Base Profiles or Application Layout Profiles.

Hence deleting a base profile or Application Layout Profile doesn't delete the attached base profiles. On the other hand the attempt to delete a Subprofile fails if it is still used in another base profile.

4.4.79 Task Profile

Task Profile: A task profile, or shortly **task,** is a list of instructions the program should execute. It defines the sequence of the processing steps and objects to work with. (E.G.First search for barcode on bitonal images, then remove lines on bitonal images, then make a copy of the color image.)

Usually a task consist of this steps at least.

Load Base Profile
Scan or load Images
Set path name
Set file name
Save the image

4.4.80 Threshold

Threshold: This is the value at which a pixel is reproduced in binarization as *white*, values below this value are interpreted as *black*.

4.4.81 Toolbar

Toolbar: A toolbar is an elongated surface with controls that can be docked to the edge of the workspace. In the application, these control elements can be used with any actions of the program. For example, Delete image, batch finalize, call a PlugIn, etc.

4.4.82 Variable's Scope

Variable's Scope

New variables can be created in the %Code-Dialog or simply by writing them in an assignment statement in Task, Macro or Eventrules.
In general the syntax for a variable is %-character, opening round bracket, name, closing round bracket.
Depending of the meaning, variables can be assigned to different type of objects in the scan environment. For that a prefix can be specified, which has to be separated by a dot.

Prefix	**Meaning**	**Scope**	**Source / Storage**	**Comments**
A.	Application	Valid for the station	Station part of the Gobal.XML	Will be saved when saving a profile as a DAT file, but not read in when loading it.
T.	Task	Valid for the task profile	2DO part in the XML	Will be copied when copying a task profile
C.	Class	Valid for the base profile (formerly called "class")	CLA part in the XML	Will be copied when copying a base profile
J.	Job	Valid for the current job (batch)	Job-XMLs	Will be created when used the first time and stored in the open job. Will not be inherited. Will be kept after a job ends, but will be deleted when a new job is started.
D.	Directory	Valid for all files in the current folder (directory)	Directory-XMLs	Will be created when used the first time and stored in the open job. Will be copied (inherited) from the previous folder. Will be kept after a job ends, but will be deleted when a new job is started.
F.	File	Valid for all images in the current file	File-XMLs	Will be created when used the first time and stored in the open job. Will be copied (inherited) from the previous file Will be kept after a job ends, but will be deleted when a new job is started.
I.	Image	Valid for the current image	Image-XMLs	Will be copied (inherited) from the previous scan. Will be kept after a job ends, but will be deleted when a new job is started.
V.	Volatile	Valid at runtime	memory of the current instance	Volatile, can be used by different profiles, jobs, folders, files and variables. Will not be copied when starting a new instance (DpuFinaize, DpuProcess) Lost after program end.
	Ad hoc-Variables Without ore unknown scope indicator	Valid at runtime, but saved for each image separately.	Image-XMLs AND with the Global.XML	Will be copied from the previous scan or from the station, see remark.

SYS.	Computer System	Global system variables like computer name	Served by the operating system	Are not stored, cannot be overwritten.
S.	DpuScan-System	States, flags and counters of DpuScan.		Will exist when program starts. Information (like Time) are write protected, Variables (like Barcode) can be overwritten. Will be kept after a job ends. Will be deleted when a new job is started (all counters) or a new image is scanned (Barcode,Patch)
G.	DpuServer (global)	Valid for all stations connected to this server.	DpuServer-table	

Remark for I-Variables:
If a scan consist of more than one image, e.g. a copy or a cut-out, the same value will be assigned to all images of this scan.
Vice versa only the value of the last image of this group can be recalled.

Remark for ad hoc variables (scope less and unknown variables):
Use them very sparely! Once they are created, they will be inherited from image to image and from job to job and finally from station to station, if the job is shared by several stations.
To avoid this, please set the parameter NoScopelessVars=1 at the system information.

Remark for persistence variables
Many scanning configurations require information assigned to certain objects, or information to be stored **permanently**, require information to outlast the life of a single scan job.
To be sure use A. C. or T. variables in such case.

4.4.83 VBS

VBS stands for Visual Basic Script. A VBS is a text file that can be run from the operating system by a so-called scripting host.

The scripting language VBS has a fixed set of commands for repeating instructions and for branching in different processes. It has also a COM interface that enables access to a plurality of system components. So you can, for example, after scanning connect to a database and submit the finished job there as completed.

4.5 Variables (%-Codes)

4.5.1 Overview

The program collects a lot of information from different sources and makes it available as program variables. Since these variables start with a percentage sign, they are also called percent codes or %-Codes. All program variables can be queried and some of them can be assigned too.

Variables can be grouped according to their access type, origin, scope, or usage:

Access type	Source	Scope	Usage
System Information	Recognition	Station	Endorser-Codes
System Variables	Date and time	Taskprofile	Special Characters

User definied Variables	Operating System	Basisprofile
	Counters	Batch
	Image Header	Directory
	Marks	File
	Command Line	Image

See also formatting and percent code selection.

4.5.1.1 System Information

System information are usually values of functions that are executed when the %-Code is queried, for example the clock is queried to get the current hour. They are thus not variables, i.e their values aren't saved and can not be overwritten.

The system information begins with the prefix S.

Available %-Codes in alphabetical order:

Variable	Description
%(S.ACTUALCLASS) %$A	The base profile holding all the task parameters of the task that is currently executed.
%(S.ACTUALTASK) %(S.TASKFILE) %$d	The task, that is currently executed. The task can be pre-selected automatically by the command line parameter as **TASKFILE**.or as **TODOFILE.** Please do not confuse this variable with %(S.TASKNAME) which is just a commenting text.
%(S.ADDPATH) %(ADDPATH) %(S.ADDDIR	Additional path of the images, defined in the path dialog, with %-codes replaced at start of task. It can be given as addendum to the Additional directory of saved images (without file names). In addition to the main directory , a subdirectory can also be specified.
%(S.APPTYPE)	3-letter shortcut for the working mode of the program. **DPU** as ccan application with user interface, visible on the screen **Fnn** as background process of type DpuFinalize. **Pnn** as background process of type DpuProcess. If there are more than one instance in the background the **nn** gives the number of the process, e..g. **F01** for background finalizer started first.
%(S.ASCII.BELL) %\a	Bell, beep
%(S.ASCII.BSLASH) %\\	Backslash (inverse slash)
%(S.ASCII.BSPACE)	Backspace

%\b

%(S.ASCII.DQUOTE)
%\"

Double quote

%(S.ASCII.ESCAPE)
%\e

Escape sign (dec. 27)

%(S.ASCII.FFEED)
%\f

Form feed

%(S.ASCII.HTAB)
%\t

Horizontal tab

%(S.ASCII.LFEED)
%\n

Line feed

%(S.ASCII.NULL)
%\0

0 (ASCII sign zero)

%(S.ASCII.QMARK)
%\?

Questionmark

%(S.ASCII.RETURN)
%\r

Carriage return

%(S.ASCII.SMARK)
%\'

Single quote

%(S.ASCII.VTAB)
%\v

Vertical tab.

%(S.ASCII.Xn)
%\X
%\x

Where **n** is a one to five digit number, which is the UTF-16-Code in **decimal** interpretation.

e.g.

%\x8364 for €

%\x169 for ©

%\x174 for ®

%\x177 for ±

%\x34 " for quotation mark

%\x39 ' for apostrophe

%\x91 and %\x93 for [ ] (square brackets)

%\x60 and %\x62 for < > (angle brackets)

%\x123 and %\x125 for { } (curly brackets)

%(S.BARDIRECTION_)

Returns information about the direction of the current barcode number (1-9). The returned values are summarized in this table:

Return Value	Text	Reading direction	Image rotatio

1	→	from left to right	~ 0°
2	↘	from left top to right bottom	~ 45°
3	↓	from top to bottom	~ 90°
4	↙	from right top to left bottom	~ 135
5	←	from right to left	~ 180
6	↖	from right bottom to left top	~ 225
7	↑	from bottom to top	~ 270
0	↗	from left bottom to right top	~ 315

A normal, horizontally applied barcode should be recognized with "1", but in exceptional cases it can also be "0" or "2"

%(S.BARDIRECTIONOLD_)

Returns information about the direction of the **last** barcode (1-9).

%(S.BARFIRST)

Returns the first, found barcode value of the current image. So it is possible to check whether a barcode was found at all. If not, no value is provided.

If the bar code search is **not restricted** to several fixed fields, this value is equal to %(S.BAR1). **Otherwise**, this is the first field found.

%(S.BARPOS_)

Returns the position of the barcode (1-9) found on the current page.

The format of the returned text is:

"left edge from the left, right edge from the left, top edge from top, bottom edge from top, unit"

The numbers are padded with leading zeros.

%(S.BARPOSOLD_)

Returns the position of the last barcode (1-9) found, see %(S.BARPOS_)

%(S.BARPOSSIBLE)

Returns a barcode that has been recognized, but does not meet the specification for that type, e.g. a Code 39 where the start/stop characters are missing.

%(S.BROKERSTATUS)

%(S.BROKERSTAT)

%$B

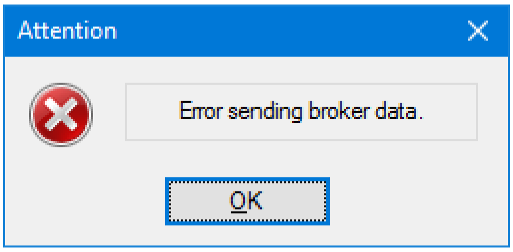
Numerical return value of the broker call.

Values **greater than or equal to 0** describe a status of the broker.

A return value of **0x9999** fills the variable %(S.BROKERSTRING) with the broker buffer.

Values **smaller than 0** signal errors and interrupt the further processing!.

!



Error sending broker data.

%(S.BROKERSTATUSPLUGIN)

Numeric return value of a PlugIn call, if this call is made in broker mode for PlugIns.

The returned values are the same as for %(S.BROKERSTATUS)

%(S.BROKERSTRING)
%$b

If the broker call returns **0x9999**, this variable will be filled with the broker buffer.

%(S.CMDMACROSTATUS)

Status of the last macro command. Not in use at the moment.

%(S.CMDTASKSTATUS)
%(S.COMMANDSTATUS)

Status of the last task command called. E.g. this can be the return value when printing a cover page or starting the finalization process.

1 success

0 an error occurred, e.g. a jump destination wasn't found

-1 the command couldn't be executed because a restriction denied this.

%(S.CPU)

Returns a text indicating the load of all CPUs. in %

%(S.CPU.LOAD)

Returns a number indicating the load of all CPUs. in %

%(S.DIR)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.DIR.*)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.G.2) for the second grayscale image.

The entry without suffix corresponds to the variable %(S.DIR).

%(S.DOCCLASS)
%$c

Document name assigned to the batch of documents. It can be specified in a field of the job data dialog. Also it can be given by a command line parameter referred to as **DOCCLASS**.

%(S.DRIVE)

Drive for the images to save. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C: \Folder\SubFolder\File.EXT

\\Server\Share \Folder\SubFolder\File.EXT

%(S.DRIVE.*)

Drive for the images to save. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C: \Folder\SubFolder\File.EXT

\\Server\Share \Folder\SubFolder\File.EXT

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.C.1) for the first color image.

However, the value of this variable is the same for all images in a stack, since the drive specification is a fix part of the profile.

The entry without suffix corresponds to the variable %(S.DRIVE).

%(S.DRIVEDIR)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\ File.EXT

\\Server\Share\Folder\SubFolder\ File.EXT

%(S.DRIVEDIR.*)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\ File.EXT

\\Server\Share\Folder\SubFolder\ File.EXT

If multiple images are to be saved under the same filename, the exact name of the copy can be specified by specifying the letter for the color type and a number for the location. De facto, however, this value is the same for all image files in a scan because another path can only be created between two scans.

The entry without suffix corresponds to the variable %(S.DRIVEDIR)

%(S.ENDORSERCOUNT)

%(S.ENDORSERCOUNT) stores the maximum value of the %(S.ENDORSERCOUNTLAST). While %(S.ENDORSERCOUNTLAST) can be reset with the start of a new batch, %(S.ENDORSERCOUNT) retains the last value until %(S.ENDORSERCOUNTLAST) returns a higher value. It stores the maximum value over all Scan batches and also over several running times of DpuScan.

This variable behaves like a station variable. It will be the Maximum value of any scan profiles on the scan station.

%(S.ENDORSERCOUNTLAST)

Latest value of the endorser in the current batch.

Saves the last value of the variable %(S.IMAGE.ENDORSER) **increased by 1**.

This variable is treated as a J.-Variable, i.e. it belongs to the batch.

%(S.ENDORSERSTRINGLAST)

Last text printed by the printer unit of the scanner.

This variable is treated as a J.-Variable, i.e. it belongs to the batch.

During the scan a copy of this value will be saved on image level as %(S.IMAGE.ENDORSERTEXT).

%(S.EVENTMESSAGE)
%(S.MESSAGE)
%$M

Last output message of an event rule: If an event rule: "Broker/Batchfile message" has been defined, the corresponding output message can be changed via %(S.EVENTMESSAGE), for example in the log file as a line with the attribute "as event rule". This value can also be used as a description for the last event rule in the info window.

%(S.EVENTPATH)

Part of the target path created by the event rules.

This is the result of "New named folder" or "New automatic folder" action in the eventrules.

%(S.EVENTSTART)

Is 0 if the event rule were be processed at least once. In the time before that it indicates the state of the event rule processing.

%(S.EXPORTPATH)

Path which will be used for batch export.

%(S.EXT)

Type of the file to save, **without** the dot.

C:\Folder\SubFolder\File.**EXT**

If there are multiple color formats with the same name in the image group, but no special definition for this case, the automatic ending is indicated by three stars:

C:\Folder\SubFolder\File.*******

%(S.EXT.*)

Type of the file to save, **without** the dot.

C:\Folder\SubFolder\File.**EXT**

It is possible to specify different file formats for different color formats. If not specified, the regular ending for this file formats will be used, e.g. TIF for a TIFF.

Then %(S.EXT.B.1) would return "TIF".

If on the other hand the file extension is defined too, this value will be returned. For example: The second bitonal snippet (e.g. the address filed) shall be saved as #####.ADR, then %(S.EXT.B.2) will return

"ADR"

The entry without suffix corresponds to the variable %(S.EXT).

%(S.EXTERNALDATASTATUS)

Return value of a PlugIn-DLL

%(S.FILE)

Name and type of the file to save, without drive and folder.

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\**File__*_.Ext**

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\**********.Ext**

%(S.FILE.*)

Name and type of the file to save, without drive and folder.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.FILE).

%(S.FILEIMAGECOUNT)

Number of images in the current file, but without images marked red. This counter is incremented only after saving the file. It behaves as %(S.SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.

%(S.FILEIMAGECOUNTALL)

Number of images in the current file, including red marked ones.

%(S.FILEIMAGECOUNTIN)

Number of images in a file loaded from disk.

%(S.IMAGE.CODING.*)

Coding mode (G4,JPG,JP2) of the current image. This is **not** the file type (TIFF, JFIF, PDF)

%(S.IMAGE.COLOR.*)

Color depth of the current image
0: unknown
1: Bitonal
2: Gray
3: Color

%(S.IMAGE.COLORINDEX.*)

Ordinal number of an image in an image group if there are several images of the same color type.

For example if there two images in a scan

1: first image

2: second image.

Together with the color type S.IMAGE.COLOR this forms a unique index inside the group.

%(S.IMAGE.DPI.*)

Resolution in dots per inch DPI, some samples are here.

%(S.IMAGE.DPIX.*)

Horizontal resolution in dots per inch DPI

The old syntax of this variable %(S.IMAGE.DPI_X) should **no longer** be used..

%(S.IMAGE.DPIY.*)

Vertical resolution in dots per inch DPI

Usually, horizontal and vertical resolution are always the same when scanning. For images from other sources, the resolutions may be different.

The old syntax of this variable %(S.IMAGE.DPI_Y) should **no longer** be used.

%(S.IMAGE.ENDORSER.*)

Endorser number of the current image. This value is queried from the machine for each scan. It is not necessarily continuous, since the scanner can set it e.g. in response to a user intervention or a counter event.

This value is the same for all images of one scan, i.e. all front and all back pictures.

%(S.IMAGE.ENDORSERSTRING.*)

Endorser text of the current image.

This value is the same for all images of one scan, i.e. all front and all back pictures.

%(S.IMAGE.HEIGHT.*)

Height of an image in pixels. The height depends on the selected resolution

%(S.IMAGE.ID.*)

Unique image counter within the batch.

%(S.IMAGE.INFO)

Returns the ImageHeader for the current image.

%K

%(S.IMAGE.INFOOLD)

Returns the ImageHeader for the previous image.

%$K

Comparing the previous and the current ImageHeader **%(S.IMAGE.INFO)**, short **%K**, in the Event rules allows to react on a change and to form a document break.

%(S.IMAGE.OCRFILE)

Filename of the text recognition (Full-Page-OCR).

%(S.IMAGE.POSINFILE.*)

Position of the image with in a file when loading files.

%(S.IMAGE.SAVED.*)

This function returns a 1 if the image was saved 0 otherwise.

%(S.IMAGE.SIDE)
%(S.PAGESIDE)
%s

Image information of the previously scanned image.

Sheet Information (Front or Back): The front is marked with "1" and the back with "2". If the sheet information is unknown (e.g., loading from directory), "0"; is output.

Note that this variable has a special behavior when enumerating formatting, ignoring zero and assigning the first value to the 1.

%s[Front, Back] returns 1 = front and 2 = back.

For other variables, the count starts at zero.

%(S.IMAGE.SIZE.*)

Size of the image in bytes, this value depends on color type and compression.

%(S.IMAGE.SIZEBITONAL)
%k

This function returns the size in bytes of the bitonal image. If there are only color and gray images it will be calculated nevertheless.

%(S.IMAGE.TIME)

Returns the time when the images was scanned.

%(S.IMAGE.WIDTH.*)

Width of an image in pixels. The height depends on the selected resolution

%(S.IMPORTPATH)

Path which will be used for batch import.

%(S.IN.PATH)
%(S.INPUTPATH)

Full path to the image source location, containing drive, directory file name and file type

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DRIVEDIR)
%(S.INPUTDRIVEDIR

Path to the image source location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DRIVE)
%(S.INPUTDRIVE)

Drive of the image source location. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\Name.Ext
\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DIR)
%(S.INPUTDIR)

Name of the folder containing the files to read without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

Verzeichnis, in dem die in dem die einzulesende Datei liegt, ohne Laufwerk und Dateiname.

C:**\Folder\SubFolder\**Name.Ext

\\Server\Share**\Folder\SubFolder\**Name.Ext

Die Angabe erfolgt **mit** einleitendem und abschließendem Schrägstrich.

%(S.IN.NAME)
%(S.INPUTNAME)

Name of the file to load, without drive, directory, dot and file type.

C:\Folder\SubFolder\**Name**.Ext

%(S.IN.EXT)
%(S.INPUTEXT)

Type of the file to load, without the dot.

C:\Folder\SubFolder\Name.**Ext**

%(S.ISVISIBLE.BITONAL)

Indicates whether black-and-white images are displayed.

%(S.ISVISIBLE.COLOR)

Indicates whether color images are displayed.

%(S.ISVISIBLE.GRAY)

Indicates whether gray scale images are displayed.

%(S.ISVISIBLE.MARK.BLUE)

Indicates whether images are displayed that are marked blue.

%(S.ISVISIBLE.MARK.GREEN

Indicates whether images are displayed that are marked green.

%(S.ISVISIBLE.MARK.NORMAL)

Indicates whether images are displayed that are not marked.

%(S.ISVISIBLE.MARK.RED

Indicates whether images are displayed that are marked red.

%(S.JOBCOUNT)
%$J

Job counter; Number of batches started. It is increased by one when a new batch is created, so already existing batches are not counted.

The counter is set independently of the selected profile.

The initial value can be specified at the start of the program.

%(S.JOBFILECOUNT)
%(S.FILECOUNT)

Returns the number of created files in the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after saving the file. It behaves as %(SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.

%(S.JOBFILECOUNTALL)
%(S.FILECOUNTALL)
%(S.IMAGEFILE)

Returns the number of created files in the current batch.

This counter is incremented only after saving the file.

%(S.JOBIMAGECOUNT)
%(S.IMAGECOUNT)

Returns the number of images in the current batch, **without** the ones deleted by **red marks**.

%(S.JOBIMAGECOUNTALL)
%(S.IMAGECOUNTALL)
%(S.IMAGEINFILEJOB)

Returns the number of images in the current batch.

%(S.JOBPATHCOUNT)
%(S.PATHCOUNT)

Returns the number of sub folders created n the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after closing a folder.

%(S.JOBPATHCOUNTALL)
%(S.PATHCOUNTALL)

Returns the number of sub folders created n the current batch.

%(S.LINEEND)

Character sequence defining the end of a line in a text output. The valid values are CRLF, LF or LFCR can be specified at the system information, see **LineEnd.**

%(S.MAINPATH)
%(MAINPATH)
%(S.MAINDIR

Main path of the images to store. It must be specified in the path dialog and hold at least a drive specification. In this dialog an additonal folder can be specified too.

%(S.MARK.BLUE)
%(S.MARKBLUE)

Indicates whether an image has a blue mark, that means it causes an error prevents finalization. If there are several images in the image group, %(S.MARK.BLUE) returns a 1 if at least one of the images within the image group has a blue mark.

%(S.MARK.BLUE.*)

The variable can also be used with the color format and the picture number. Thus,%(S.MARK.BLUE.C.1) supplies the information to the first color image of the image group.

%(S.MARK.BLUEACTIVE)

Indicates whether images with blue marks are set in the batch.

%(S.MARK.BLUECOUNT)

Returns the number of blue marked images in the batch.

%(S.MARK.GREEN)
%(S.MARKGREEN)

Indicates whether an image has a red mark, that means it is important. If there are several images in the image group, %(S.MARK.GREEN) returns a 1 if at least one of the images within the image group has a green mark.

%(S.MARK.GREEN.*)

The variable can also be used with the color format and the picture number. So %(S.MARK.GREEN.C.1) supplies the information to the first color image of the image group.

%(S.MARK.GREENACTIVE)

Indicates whether images with green marks are set in the batch.

%(S.MARK.GREENCOUNT)

Returns the number of green marked images in the batch.

%(S.MARK.RED)
%(S.MARKRED)

Indicates whether an image has a red mark, that means it discarded during finalization. If there are several images in the image group, %(S.MARK.RED) returns a 1 if at least one of the images within the image group has a red mark.

%(S.MARK.RED.*)

The variable can also be used with the color format and the picture number. Thus,%(S.MARK.RED.C.1) supplies the information to the first color image of the image group.

%(S.MARK.REDACTIVE)

Indicates whether images with red marks are set in the batch.

%(S.MARK.REDCOUNT)

Returns the number of red marked images in the batch.

%(S.MARK.WHITE)
%(S.MARKWHITE)

Indicates whether an image has a white mark, that means it is hidden. If there are several images in the image group, %(S.MARK.WHITE) returns a 1 if at least one of the images within the image group has a white mark.

%(S.MARK.WHITE.*)

The variable can also be used with the color format and the picture number. So %(S.MARK.WHITE.C.1) supplies the information to the first color image of the image group.

%(S.MARK.WHITEACTIVE)

Indicates whether images with white marks are set in the batch.

%(S.MARK.WHITECOUNT)

Returns the number of white marked images in the batch.

%(S.MEMORY)

Information about the computer's memory as a text.

%(S.MEMORY.FREE)

Memory free in MB.

%(S.MEMORY.LOAD)

Memory loaded in Percent.

%(S.MEMORY.TOTAL)

Total memory in MB.

%(S.MEMORY.USED)

Memory used in in MB.

%(S.NAME)

Name of the file to save, without drive, directory, dot and file type.

C:\Folder\SubFolder\**File**.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\**File__*_**.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\************.Ext

%(S.NAME.*)

Name of the file to save, without drive, directory, dot and file type.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\**Name__C1_**.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\**Bank**.Ext

The entry without suffix corresponds to the variable %(S.NAME).

%(S.OPENJOBBASE)

Contains the base path wich will be used to create or reopen a batch in OpenJob-Mode.

%(S.OPENJOBPATH)

Path of an open job, i.e. the open job base and the open job name. Please note: The structure of an open job folder differs from the target folder!

%(S.OUT.PATH)
%(S.REALPATH)
%F

This function provides the complete output path for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.PATH)**

C:\Folder\SubFolder\File.Ext
\\Server\Share\Folder\SubFolder\File.Ext

The "Real" part of the name should indicate that it is the name under which the file is actually saved when finalizing an open batch.

%(S.OUT.DRIVEDIR)
%(S.REALDRIVEDIR)

This function provides the storage forlder for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.DriveDir).**

This value will be returned **with** a trailing backslash.:

C:\Folder\SubFolder\File.Ext
\\Server\Share\Folder\File.Ext

%(S.OUT.FILE)
%(S.REALFILE)

This function returnes the file of the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.File).**

C:\Folder\SubFolder\File.Ext

%(S.OUT.DRIVE)
%(S.REALDRIVE)
%D

Returns the drive for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Drive).**

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.Ext
\\Server\Share\Folder\SubFolder\File.Ext

%(S.OUT.DIR)
%(S.REALDIR)
%P

Returns the folder for the images to save without drive an filespecification. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Dir).**

The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext
\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.OUT.NAME)
%(S.REALFILENAME)
%(S.REALNAME)
%N

Returns the file name for the images to save without drive, folder and extension. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%**

(S.Name).

C:\Folder\SubFolder\**File**.Ext

%(S.OUT.EXT)
%E

Returns the file type (extension) **without** the dot. for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Ext).**

C:\Folder\SubFolder\File**.Ext**

%(S.OUT.EXT+)
%(S.REALEXT)

Returns the file type (extension) **inclusive** the dot.

C:\Folder\SubFolder\File**.Ext**

%(S.PATH)

Full path to the image storing location, containing drive, directory file name and file type

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\File__*_.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\******.Ext**

%(S.PATH.*)

Full path to the image storing location, containing drive, directory file name and file type

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.PATH.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.PATH).

%(S.PATHFILECOUNT)
%(S.FILESINPATH)

This function returns the number of files in path, without those files which will be deleted because of red marked images.

%

This function returns the number of files in path.

(S.PATHFILECOUNTALL)

%(S.PATHIMAGECOUNT)

This function returns the number of images in the current path, without those files which will be deleted because of red marked images.

%(S.PATHIMAGECOUNTALL)

This function returns the number of images in the current path.

%(S.POLLBATCHGETFOUND)

This function returns the batch name of the found batch when the program searches for batches in polling mode.

%(S.POLLFILEFOUND)

When working in Polling mode and searching for files, this function returns the full path to the polling file found. This is not necessary the image location too.

%(S.POLLPATHBASE)

This function returns the defined path where the program starts to search for batches in polling mode.

The returned path is **without** trailing backslash

%(S.POLLPATHFOUND)

When working in Polling mode and searching for files, this function returns the folder holding the images to process. Depending on the settings the polling file may or may not be in this folder too.

The returned path is **with** trailing backslash.

%(S.PROGRAM.PATH)

Full path to the program location, containing drive, directory file name and file type

%(S.PROGRAM.DRIVEDIR)

Path to the program location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

%(S.PROGRAM.FILE)

Name and type of the program, without drive and folder.

%(S.PROGRAM.DRIVE)

This function returns the drive on which the program is installed.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.PROGRAM.DIR)

This function returns the folder in which the program is installed, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.PROGRAM.NAME)

Name of the program, without drive, directory, dot and file type.

%(S.PROGRAM.EXT)

Type of the program file, without the dot.

%(S.SCANSIDES)

Returns a 2 for duplex scanning and 1 for simplex scanning.

%(S.SCANSMINUTE)
%f

Number of scans per minute. This value is constantly updated every few seconds and gives the actual value for the speed, similar to a speedometer.

Since several images can be created during one scan, this value does not correspond to the number of images per minute.

%(S.STATUS.PATH)
%(S.STATUSPATH)
%(S.STATUS)
%(STATUSFILE)

Full path to the location of the status file, containing drive, directory file name and file type.

%(S.STATUS.DRIVEDIR)
%(S.STATUSDRIVEDIR)

Path to the location of the status file including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

%(S.STATUS.FILE)
%(S.STATUSFILE)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.DRIVE)
%(S.STATUSDRIVE)

This function returns the drive on which the status file is located.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.STATUS.DIR)
%(S.STATUSDIR)

This function returns the folder in which the on which the status file is located, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.STATUS.NAME)
%(S.STATUSNAME)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.EXT)
%(S.STATUSEXT)

Returns the extension (type) of the status file, without the dot.

%(S.TARGETBASEPATH)
%(S.STARTPATH)

Returns the destination path for the saved images,without the parts created by event rules or the polling process.

The returned text is **inclusive** the trailing backslash.

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

This value is the concatenation of %(S.MAINPATH) and %(S.ADDPATH).

%(S.TARGETPATH)

Targetpath for the images, **with** trailing Backslash.

This is the concatenation of %(S.TARGETBASEPATH) and %(S.EVENTPATH).

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

Because of the settings in the polling mode in between these components there may be the folder holding the images to read in.

%(S.TASKNAME)
%$D

Optional name of the document batch. It can be set in the field taskname in the job data setup.

It can be also set as a command line parameter called **TASKNAME**.

If it is not set, it is equal to the content of the variable %(S.TASKFILE).

%(S.TASKSTATUS)

Returns the current program state, possible values are:

"ERROR"	If an error occurs
"START"	at start of the job, before the first image is received.
"HALT"	when pausing the job
"STOP"	when a job was ended
"EXIT"	when leaving the program
"CONTINUE"	when continuing a batch, before the first image is received.
"WORK"	when processing images
"READY"	when waiting for the next job or other user action.

%(S.TIME.AMPM)
%p

Current locale's AM/PM for a 12-hour clock. Ante meridiem – post meridiem, before and after noon.

%(S.TIME.DATE)
%x

Date representation for current locale (e.g. dd.mm.yyyy for Germany or mm/dd/yyy in the United States of Amerika or yyyy-mm-dd in the People's Republic of China).

%(S.TIME.DATETIME)
%c

Date and time, representation appropriate for the locale.

%(S.TIME.DAYLONG)
%A

Full name of the weekday

%(S.TIME.DAYMONTH)
%d

Day of month as decimal number, always two digits (01-31)

%(S.TIME.DAYSHORT)
%a

Name of the weekday, abbreviated.

%(S.TIME.DAYWEEK)
%w

Full name of the weekday, .(0-6; **Sunday** is 0)

%(S.TIME.DAYYEAR)
%j

Day of the year as decimal number (001-366)

%(S.TIME.HOUR)
%H

Hour in 24 hours format (00-23)

%(S.TIME.HOURAM)
%I

Hour in 12 hours format (01-12)

%(S.TIME.MILLISEC)
%@

Milliseconds as decimal number

%(S.TIME.MINUTE)
%M

Minute as decimal number(00-59)

%(S.TIME.MONTH) %m	Month as decimal number(01-12)
%(S.TIME.MONTHLONG) %B	Full name of the month
%(S.TIME.MONTHSHORT) %b	Name of the month, abbreviated.
%(S.TIME.SECOND) %S	Second as dezimal number (00-59)
%(S.TIME.TIME) **%X**	Time in local format (e.g. hh:mm:ss for the USA)
%(S.TIME.TIMEZONE) %Z	Name of the time zone if known, empty otherwise
%(S.TIME.WEEKYEARM) %W	Number of the week, week starts on **Monday** (00-51)
%(S.TIME.WEEKYEARS) %U	Number of the week, week starts on **Sunday** (00-51)
%(S.TIME.YEARLAST) %z	Last digit of the year
%(S.TIME.YEARLONG) %Y	Year as dezimal number.
%(S.TIME.YEARSHORT) %y	Year without century as dezimal number (00-99)
%(S.UNICODE.BOM)	This function returns a Unicode byte order mark. This string indicates which encoding corresponds to the following data, e.g. EF BB BF for UTF-8 or FF FE for UTF-16 Little Endian.
%(S.UNIQUE)	Postfix-String for folders and files which are created by a manual split operation. These files are now in between two other files numbered consecutively. This value can be set in the system information of the program, there it is denoted as **FileUniquePostfix**.
%(S.USEREVENT_)	Returns the status of the uservent (1 - 32), contains "1" if the event has occured, otherwise "0". User events are triggered by the corresponding toolbar command. They are automatically reset after processing the event rules for a group of pictures. In this way, actions can be executed depending on specific events.

%(S.USEREVENTLAST_)

Indicates whether the corresponding user event has occurred on the previous image.

That is, the value is set once for the image after the event and is reset to 0 after processing the event rules.

%(S.USERNAME)
%$U

This function returns the given username. It can be set in the field username in the job data setup. It can be also set as a command line parameter called **USERNAME**.

%(S.UTC.AMPM)

Current locale's AM/PM for a 12-hour clock. Ante meridiem – post meridiem, before and after noon

Use the **UTC**-Variables if the program is used at locations in different time zones

In the **GMT** zone (London) the local time is equal to the **UTC**-Time. In the **CET** zone (Berlin) the local time differs by 1 hour or 2 hours in the summer.

%(S.UTC.DATE)
%x

Date representation for current locale (e.g. dd.mm.yyyy for Germany or mm/dd/yyy in the United States of Amerika or yyyy-mm-dd in the People's Republic of China).

%(S.UTC.DATETIME)
%c

Date and time, representation appropriate for the locale.

%(S.UTC.DAYLONG)
%A

Full name of the weekday

%(S.UTC.DAYMONTH)
%d

Day of month as decimal number, always two digits (01-31)

%(S.UTC.DAYSHORT)
%a

Name of the weekday, abbreviated.

%(S.UTC.DAYWEEK)
%w

Full name of the weekday, .(0-6; **Sunday** is 0)

%(S.UTC.DAYYEAR)
%j

Day of the year as decimal number (001-366)

%(S.UTC.HOUR)
%H

Hour in 24 hours format (00-23)

%(S.UTC.HOURAM)
%I

Hour in 12 hours format (01-12)

%(S.UTC.MILLISEC)
%@

Milliseconds as decimal number

%(S.UTC.MINUTE)
%M

Minute as decimal number(00-59)

%(S.UTC.MONTH)
%m

Month as decimal number(01-12)

%(S.UTC.MONTHLONG)
%B

Full name of the month

%(S.UTC.MONTHSHORT)
%b

Name of the month, abbreviated.

%(S.UTC.SECOND)
%S

Second as dezimal number (00-59)

%(S.UTC.TIME)
%X

Time in local format (e.g. hh:mm:ss for the USA)

%(S.UTC.TIMEZONE)
%Z

Name of the time zone if known, empty otherwise

%(S.UTC.WEEKYEARM)
%W

Number of the week, week starts on **Monday** (00-51)

%(S.UTC.WEEKYEARS)
%U

Number of the week, week starts on **Sunday** (00-51)

%(S.UTC.YEARLAST)
%z

Last digit of the year

%(S.UTC.YEARLONG)
%Y

Year as dezimal number.

%(S.UTC.YEARSHORT)
%y

Year without century as dezimal number (00-99)

%(SYS.COMPUTERNAME)
%(SYS.STATIONNAME)

Content of the system variable COMPUTERNAME, that is the name of the computer as specified in the operating system.

%(SYS.DESKTOP)

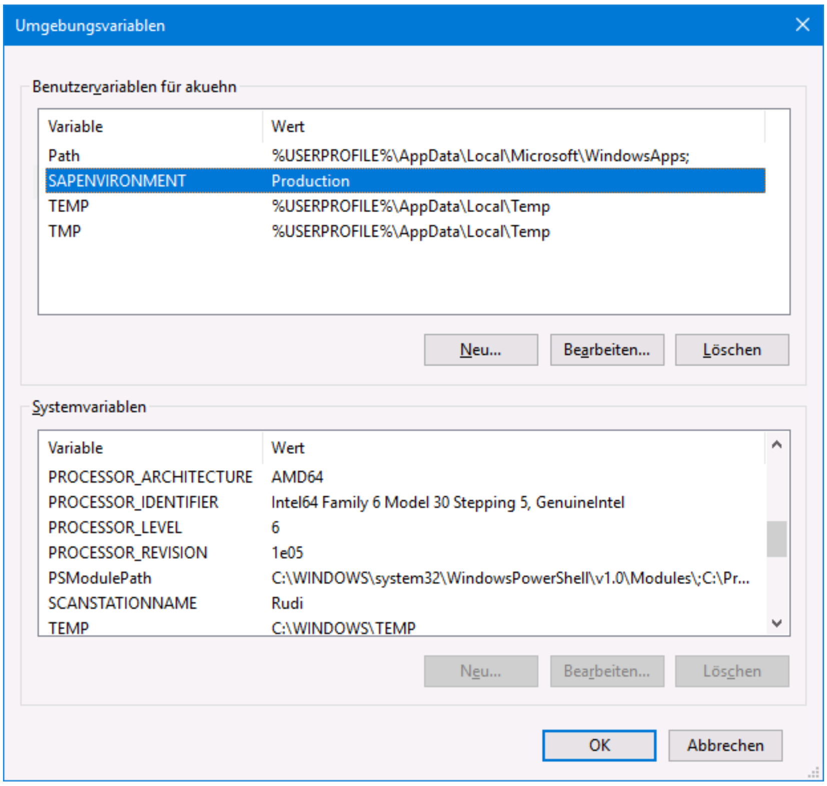
Returns the content of the environment variable DESKTOP, that is the desktop folder of the user logged on currently. The returned path has 8.3-convention and is **without** a trailing backslash, e.g.:

```
C:\Users\EBERHA~1\Desktop
```

%(SYS.ENV.*)

Returns the contents of any environment variable, e.g. the input request PROMPT by %(SYS.ENV.PROMPT).

User-defined environment variables can also be queried. In the example below, %(S.SYS.SAPENVIRONMENT) would return the value "Production"



Umgebungsvariablen

Benutzervariablen für akuehn

Variable	Wert
Path	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
SAPENVIRONMENT	Production
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Systemvariablen

Variable	Wert
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 30 Stepping 5, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	1e05
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
SCANSTATIONNAME	Rudi
TEMP	C:\WINDOWS\TEMP

%(SYS.MYDOCUMENTS) Returns the content of the environment variable DOCUMENTS, that is the documents folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

```
C:\Users\EBERHA~1\Documents
```

%(SYS.TEMP) Returns the content of the environment variable TEMP, that is the temporary folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

```
C:\Users\EBERHA~1\AppData\Local\Temp
```

%(SYS.USERNAME) Returns the content of the environment variable USERNAME, that is the name of the user logged on currently, for example

This variable is not to be confused with the program variable %(S.USERNAME) which can be set arbitrarily by command line or dialog.

4.5.1.2 System Variables

System variables are values that the program determines and then stores. They can be overwritten.

Available %-Codes in alphabetical order:

Variable	Description
%(S.BAR) %R	Returns the barcode for the current image.

%(S.BAR_)
%(BAR_)
%R_

If *several barcodes* are searched, the results are listed and separated by comma.

The individual results can then be queried by specifying a number 1..9 at the end. The barcode search can detect up to 100 barcodes per image, but only the first nine can be addressed via an index.

This variable can be overwritten.

The value assignment to %(S.BAR) (without index) must be followed by a comma after each value, even if there is only one value. Only then is this variable set, as well as all dependent variables with index, or the OLD variables for the subsequent images.

The value assignment to %(S.BAR_) (with index) does not require a comma. %(S.BAR) and the subsequent OLD variables are set automatically.

In either case, the "barcode found" event is also triggered.

%(S.BARCOUNT)
%r

Number of found barcodes from batch start to the current image.

This variable is stored as an I-variable with the image.

%(S.BARFIRST)

Returns the first, found barcode value of the current image. So it is possible to check whether a barcode was found at all. If not, no value is provided.

If the bar code search is **not restricted** to several fixed fields, this value is equal to %(S.BAR1). **Otherwise**, this is the first field found.

%(S.BARINFO)
%(S.BARINFO_)

Returns information about the current barcode (bar code type and position).

If *several barcodes* are searched, the results are listed and separated by comma.

The individual results can then be queried by specifying a number 1..9 at the end. The barcode search can detect up to 100 barcodes per image, but only the first nine can be addressed via an index.

The information in the variable can be splitted by the formatting:

Type	`%(S.BARINFO1)[>"("][<"Start"]`
left edge	`%(S.BARINFO1)[>"Start"][<" /"]`
right edge	`%(S.BARINFO1)[>"Stop"][<" /"]`
upper edge	`%(S.BARINFO1)[>"Start"][>" / "][<" "]`
lower edge	`%(S.BARINFO1)[>"Stop"][>" / "][<" "]`

%(S.BARINFOOLD)
%(S.BARINFOOLD_)

Returns information about the *last* barcode found, see %(S.BARINFO_).

%(S.BAROLD)
%$R

Returns the list of the barcode for the last image. See %(S.BAR)

%(S.BAROLD_)
%(BAROLD_)

%$R_

%(S.BATESTOTAL)
%$s

Returns the counter for the bates stamp

%(S.CMDPARAM_)

Returns the contents of the command line parameter 1 to 9.

These values are also output in the statistics file in columns 21-29.

%(S.JOBDAYCOUNT)

Counter of all scanned batches of the day.

%(S.JOBNAME)
%O
%o

Liefert den Inhalt des Kommandozeilenparameters JOBNAME bzw. des eingestellten Wertes in den Jobdaten.

%(S.OCR_)
%(OCR_)

Returns the result of the text recognition for the given search area 1..99. The underline at the end of the variable's name has to be replaced by the corresponding number of the field.

%(S.PAPERSAVED)
%(PAPERSAVED)

Returns the number of papers stored during the current job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved.

%(S.PAPERSAVEDDAY)

Returns the number of papers stored today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved.

The counter will be reset with the first start of a job after midnight.

%(S.PAPERSCANNED)
%(S.PAPERCOUNT)

%(PAPERSCANNED)
%(PAPERCOUNT)

Returns the number of papers scanned or loaded during the current job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned.

%(S.PAPERSCANNEDDAY)

Returns the number of papers scanned or loaded today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned.

The counter will be reset with the first start of a job after midnight.

%(S.PATCH)

Returns the Patchcode for the current image.

The search always ends after the first code found. Because of that one cannot search for multiple codes hence there are also no variables for multiple results.

%(S.PATCHOLD)

Returns the last Patchcode found, see %(S.PATCH)

%(S.PATCHPOSSIBLE)

If the search for Patchcodes is restricted to only a few types the search will recognize all 6 codes anyway. A code which is found but not of interest will returned by this function.

%(S.SAVECOUNT_)
%(SAVECOUNT_)

Counter for the stored images. There are counters 1 to 9.

These counters can be set arbitrarily by a value assignment.

They are increased automatically when an image is saved

%(S.SCANCOUNT_)
%(SCANCOUNT_)

Counter for the scanned images. There are counters 1 to 9.
These counters can be set arbitrarily by a value assignment.
They are automatically increased when an image is scanned

All images that the scanner delivers from a page count as one scan. This includes the color separations as well as the partial images and composite images from the Image Processor.

%(S.SCANSCLASS)
%C

Total scans for the actual base profile. For each defined base profile, an individual scan counter is generated and is saved together with base profile definition.

%(S.SCANSDAY)
%$N

Number of scanned images since the beginning of the day.

This counter is reset to zero after the first start of a job after midnight.

%(S.SCANSEXPECTED)
%e

This function returns the number of images expected by the program for a batch.

This number can be specified in the basic profile or queried in the task at the beginning.

If this number is specified, it is checked for each scan and, if necessary, the process is stopped.

%(S.SCANSJOB)
%n

Number of scanned Images in the batch.

%(S.SCANSTOTAL)
%$C

This function returns the number of all scanned images in all profiles.

%(S.USERCOUNT_)
%(USERCOUNT_)

User-defined counters (1..9) can be defined in the event rules or sa task step.

They are used to count documents for example.

The value of the counter is stored to the current image. The variable behaves like an I-Variable. However, when the event rules are started, the counter is reset to 0.

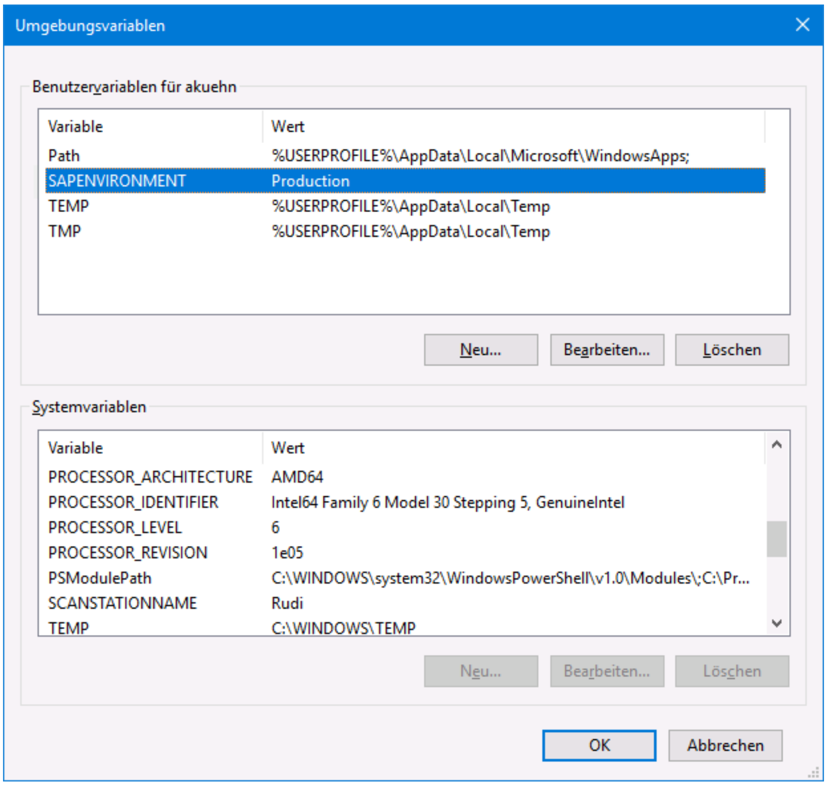
%(S.USERPARAM_)

Returns the contents of the command line parameter USERPARAM 1 to 9.
In contrast to CMDPARAM 1 to 9, these values are not passed to the statistics file.

%(SYS.ENV.*)

Returns the contents of any environment variable, e.g. the input request PROMPT by %(SYS.ENV.PROMPT).

User-defined environment variables can also be queried. In the example below, %(S.SYS.SAPENVIRONMENT) would return the value "Production"



Umgebungsvariablen

Benutzervariablen für akuehn

Variable	Wert
Path	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
SAPENVIRONMENT	Production
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Systemvariablen

Variable	Wert
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 30 Stepping 5, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	1e05
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
SCANSTATIONNAME	Rudi
TEMP	C:\WINDOWS\TEMP

4.5.1.3 Usedefined Percent Codes

In addition to the variables generated by the program, you can create any of your own variables. This is useful if you want to hold a certain value, to output or evaluate it later.

A user-defined variable must start with a percent sign and an opening parenthesis and end with a closing parenthesis. It should always have a prefix that indicates the scope of validity. Suffixes for certain images in the image group are **not allowed** and will be ignored.

New custom variables can be specified for a value assignment to the left of the equals sign, e.g. In the event rules or in the task. If the variable is not to be used immediately, it can also be created in the selection dialog.

Variable	Description
%(P.xxxx)	The prefix p defines the scope of the variable, that means where this variable is stored and where it is accessible. The name xxxx can be replace by any meaningful name. It can consist of any **letters**, **numbers**, **underscores** and **dots**. Uppercase and lowercase is NOT considered program-internally.
Caution:	**Do not use userdefined variable with the prefix S or SYS, because they will not be saved.**

4.5.1.4 Percent Codes for Recognition

The variables listed here come from the built-in recognition of Barcodes, Patchcodes and text recognition in fields. Recognition results from other PlugIns, e.g. the data matrix recognition are not listed here.

Available %-Codes in alphabetical order:

Variable | **Description**

%(S.BAR)
%R

Returns the barcode for the current image.

If *several barcodes* are searched, the results are listed and separated by comma.

%(S.BAR_)
%(BAR_)
%R_

The individual results can then be queried by specifying a number 1..9 at the end. The barcode search can detect up to 100 barcodes per image, but only the first nine can be addressed via an index.

This variable can be overwritten.

The value assignment to %(S.BAR) (without index) must be followed by a comma after each value, even if there is only one value. Only then is this variable set, as well as all dependent variables with index, or the OLD variables for the subsequent images.

The value assignment to %(S.BAR_) (with index) does not require a comma. %(S.BAR) and the subsequent OLD variables are set automatically.

In either case, the "barcode found" event is also triggered.

%(S.BARCOUNT)
%r

Number of found barcodes from batch start to the current image.

This variable is stored as an I-variable with the image.

%(S.BARDIRECTION_)

Returns information about the direction of the current barcode number (1-9). The returned values are summarized in this table:

Return Value	Text	Reading direction	Image rotatio
1	Abc	from left to right	~ 0°
2	Abc	from left top to right bottom	~ 45°
3	Abc	from top to bottom	~ 90°
4	Abc	from right top to left bottom	~ 135
5	Abc	from right to left	~ 180
6	Abc	from right bottom to left top	~ 225
7	Abc	from bottom to top	~ 270
0	Abc	from left bottom to right top	~ 315

A normal, horizontally applied barcode should be recognized with "1", but in exceptional cases it can also be "0" or "2"

%

Returns information about the direction of the **last** barcode (1-9).

(S.BARDIRECTIONOLD_)

%(S.BARFIRST)

Returns the first, found barcode value of the current image. So it is possible to check whether a barcode was found at all. If not, no value is provided.

If the bar code search is **not restricted** to several fixed fields, this value is equal to %(S.BAR1). **Otherwise**, this is the first field found.

%(S.BARINFO)
%(S.BARINFO_)

Returns information about the current barcode (bar code type and position).

If *several barcodes* are searched, the results are listed and separated by comma.

The individual results can then be queried by specifying a number 1..9 at the end. The barcode search can detect up to 100 barcodes per image, but only the first nine can be addressed via an index.

The information in the variable can be splitted by the formatting:

Type	`%(S.BARINFO1)[>"("][<"Start"]`
left edge	`%(S.BARINFO1)[>"Start"][<" /"]`
right edge	`%(S.BARINFO1)[>"Stop"][<" /"]`
upper edge	`%(S.BARINFO1)[>"Start"][>" / "][<" "]`
lower edge	`%(S.BARINFO1)[>"Stop"][>" / "][<" "]`

%(S.BARINFOOLD)
%(S.BARINFOOLD_)

Returns information about the *last* barcode found, see %(S.BARINFO_).

%(S.BAROLD)
%$R

Returns the list of the barcode for the last image. See %(S.BAR)

%(S.BAROLD_)
%(BAROLD_)
%$R_

%(S.BARPOS_)

Returns the position of the barcode (1-9) found on the current page.

The format of the returned text is:

"left edge from the left, right edge from the left, top edge from top, bottom edge from top, unit"

The numbers are padded with leading zeros.

%(S.BARPOSOLD_)

Returns the position of the last barcode (1-9) found, see %(S.BARPOS_)

%(S.BARPOSSIBLE)

Returns a barcode that has been recognized, but does not meet the specification for that type, e.g. a Code 39 where the start/stop characters are missing.

%(S.IMAGE.INFO)

Returns the ImageHeader for the current image.

%K	
%(S.IMAGE.INFOOLD) %$K	Returns the ImageHeader for the previous image. Comparing the previous and the current ImageHeader **%(S.IMAGE.INFO)**, short **%K**, in the Event rules allows to react on a change and to form a document break.
%(S.IMAGE.OCRFILE)	Filename of the text recognition (Full-Page-OCR).
%(S.OCR_) %(OCR_)	Returns the result of the text recognition for the given search area 1..99. The underline at the end of the variable's name has to be replaced by the corresponding number of the field.
%(S.PATCH)	Returns the Patchcode for the current image. The search always ends after the first code found. Because of that one cannot search for multiple codes hence there are also no variables for multiple results.
%(S.PATCHOLD)	Returns the last Patchcode found, see %(S.PATCH)
%(S.PATCHPOSSIBLE)	If the search for Patchcodes is restricted to only a few types the search will recognize all 6 codes anyway. A code which is found but not of interest will returned by this function.

4.5.1.5 Percent Codes for Date and Time

The variables listed here result from the query of the system time. It is distinguished according to local time and standard world time UTC.

Available %-Codes in alphabetical order:

Variable	Description
%(S.TIME.AMPM) %p	Current locale's AM/PM for a 12-hour clock. Ante meridiem – post meridiem, before and after noon.
%(S.TIME.DATE) %x	Date representation for current locale (e.g. dd.mm.yyyy for Germany or mm/dd/yyy in the United States of Amerika or yyyy-mm-dd in the People's Republic of China).
%(S.TIME.DATETIME) %c	Date and time, representation appropriate for the locale.
%(S.TIME.DAYLONG) %A	Full name of the weekday
%(S.TIME.DAYMONTH) %d	Day of month as decimal number, always two digits (01-31)
%(S.TIME.DAYSHORT) %a	Name of the weekday, abbreviated.
%(S.TIME.DAYWEEK) %w	Full name of the weekday, .(0-6; **Sunday** is 0)
%(S.TIME.DAYYEAR)	Day of the year as decimal number (001-366)

%j	
%(S.TIME.HOUR) %H	Hour in 24 hours format (00-23)
%(S.TIME.HOURAM) %I	Hour in 12 hours format (01-12)
%(S.TIME.MILLISEC) %@	Milliseconds as decimal number
%(S.TIME.MINUTE) %M	Minute as decimal number(00-59)
%(S.TIME.MONTH) %m	Month as decimal number(01-12)
%(S.TIME.MONTHLONG) %B	Full name of the month
%(S.TIME.MONTHSHORT) %b	Name of the month, abbreviated.
%(S.TIME.SECOND) %S	Second as dezimal number (00-59)
%(S.TIME.TIME) **%X**	Time in local format (e.g. hh:mm:ss for the USA)
%(S.TIME.TIMEZONE) %Z	Name of the time zone if known, empty otherwise
%(S.TIME.WEEKYEARM) %W	Number of the week, week starts on **Monday** (00-51)
%(S.TIME.WEEKYEARS) %U	Number of the week, week starts on **Sunday** (00-51)
%(S.TIME.YEARLAST) %z	Last digit of the year
%(S.TIME.YEARLONG) %Y	Year as dezimal number.
%(S.TIME.YEARSHORT) %y	Year without century as dezimal number (00-99)
%(S.UTC.AMPM)	Current locale's AM/PM for a 12-hour clock. Ante meridiem – post meridiem, before and after noon Use the **UTC**-Variables if the program is used at locations in different time zones

	In the **GMT** zone (London) the local time is equal to the **UTC**-Time. In the **CET** zone (Berlin) the local time differs by 1 hour or 2 hours in the summer.
%(S.UTC.DATE) %x	Date representation for current locale (e.g. dd.mm.yyyy for Germany or mm/dd/yyy in the United States of Amerika or yyyy-mm-dd in the People's Republic of China).
%(S.UTC.DATETIME) %c	Date and time, representation appropriate for the locale.
%(S.UTC.DAYLONG) %A	Full name of the weekday
%(S.UTC.DAYMONTH) %d	Day of month as decimal number, always two digits (01-31)
%(S.UTC.DAYSHORT) %a	Name of the weekday, abbreviated.
%(S.UTC.DAYWEEK) %w	Full name of the weekday, .(0-6; **Sunday** is 0)
%(S.UTC.DAYYEAR) %j	Day of the year as decimal number (001-366)
%(S.UTC.HOUR) %H	Hour in 24 hours format (00-23)
%(S.UTC.HOURAM) %I	Hour in 12 hours format (01-12)
%(S.UTC.MILLISEC) %@	Milliseconds as decimal number
%(S.UTC.MINUTE) %M	Minute as decimal number(00-59)
%(S.UTC.MONTH) %m	Month as decimal number(01-12)
%(S.UTC.MONTHLONG) %B	Full name of the month
%(S.UTC.MONTHSHORT) %b	Name of the month, abbreviated.
%(S.UTC.SECOND) %S	Second as dezimal number (00-59)
%(S.UTC.TIME) **%X**	Time in local format (e.g. hh:mm:ss for the USA)
%(S.UTC.TIMEZONE) %Z	Name of the time zone if known, empty otherwise
%(S.UTC.WEEKYEARM) %W	Number of the week, week starts on **Monday** (00-51)

%(S.UTC.WEEKYEARS) %U	Number of the week, week starts on **Sunday** (00-51)
%(S.UTC.YEARLAST) %z	Last digit of the year
%(S.UTC.YEARLONG) %Y	Year as dezimal number.
%(S.UTC.YEARSHORT) %y	Year without century as dezimal number (00-99)

4.5.1.6 Percent Codes for Counter

Here are all variables listed that are used to count objects.

Available %-Codes in alphabetical order:

Variable	Description
%(S.BARCOUNT) %r	Number of found barcodes from batch start to the current image. This variable is stored as an I-variable with the image.
%(S.BATESTOTAL) %$s	Returns the counter for the bates stamp
%(S.FILEIMAGECOUNT)	Number of images in the current file, but without images marked red. This counter is incremented only after saving the file. It behaves as %(S.SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.
%(S.FILEIMAGECOUNTALL)	Number of images in the current file, including red marked ones.
%(S.ENDORSERCOUNT)	%(S.ENDORSERCOUNT) stores the maximum value of the %(S.ENDORSERCOUNTLAST). While %(S.ENDORSERCOUNTLAST) can be reset with the start of a new batch, %(S.ENDORSERCOUNT) retains the last value until %(S.ENDORSERCOUNTLAST) returns a higher value. It stores the maximum value over all Scan batches and also over several running times of DpuScan. This variable behaves like a station variable. It will be the Maximum value of any scan profiles on the scan station.
%(S.ENDORSERCOUNTLAST)	Latest value of the endorser in the current batch. Saves the last value of the variable %(S.IMAGE.ENDORSER) **increased by 1**. This variable is treated as a J.-Variable, i.e. it belongs to the batch.
%(S.JOBCOUNT) %$J	Job counter; Number of batches started. It is increased by one when a new batch is created, so already existing batches are not counted. The counter is set independently of the selected profile. The initial value can be specified at the start of the program.

%(S.JOBDAYCOUNT)

Counter of all scanned batches of the day.

%(S.JOBFILECOUNT)
%(S.FILECOUNT)

Returns the number of created files in the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after saving the file. It behaves as %(SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.

%(S.JOBFILECOUNTALL)
%(S.FILECOUNTALL)
%(S.IMAGEFILE)

Returns the number of created files in the current batch.

This counter is incremented only after saving the file.

%(S.JOBIMAGECOUNT)
%(S.IMAGECOUNT)

Returns the number of images in the current batch, **without** the ones deleted by **red marks**.

%(S.JOBIMAGECOUNTALL)
%(S.IMAGECOUNTALL)
%(S.IMAGEINFILEJOB)

Returns the number of images in the current batch.

%(S.JOBPATHCOUNT)
%(S.PATHCOUNT)

Returns the number of sub folders created n the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after closing a folder.

%(S.JOBPATHCOUNTALL)
%(S.PATHCOUNTALL)

Returns the number of sub folders created n the current batch.

%(S.MARK.BLUECOUNT)

Returns the number of blue marked images in the batch.

%(S.MARK.GREENCOUNT)

Returns the number of green marked images in the batch.

%(S.MARK.REDCOUNT)

Returns the number of red marked images in the batch.

%(S.MARK.WHITECOUNT)

Returns the number of white marked images in the batch.

%(S.PAPERSCANNED)
%(S.PAPERCOUNT)

%(PAPERSCANNED)
%(PAPERCOUNT)

Returns the number of papers scanned or loaded during the current job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned.

%

Returns the number of papers scanned or loaded today. Paper-sheets will be counted here, regardless of single or double side scanning. A

(S.PAPERSCANNEDDAY)

paper counts a scanned if at least one of the sides was scanned.

The counter will be reset with the first start of a job after midnight.

%(S.PAPERSAVED)
%(PAPERSAVED)

Returns the number of papers stored during the current job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved.

%(S.PAPERSAVEDDAY)

Returns the number of papers stored today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved.

The counter will be reset with the first start of a job after midnight.

%(S.PATHFILECOUNT)
%(S.FILESINPATH)

This function returns the number of files in path, without those files which will be deleted because of red marked images.

%(S.PATHFILECOUNTALL)

This function returns the number of files in path.

%(S.PATHIMAGECOUNT)

This function returns the number of images in the current path, without those files which will be deleted because of red marked images.

%(S.PATHIMAGECOUNTALL)

This function returns the number of images in the current path.

%(S.SAVECOUNT_)
%(SAVECOUNT_)

Counter for the stored images. There are counters 1 to 9.

These counters can be set arbitrarily by a value assignment.

They are increased automatically when an image is saved

%(S.SCANCOUNT_)
%(SCANCOUNT_)

Counter for the scanned images. There are counters 1 to 9.
These counters can be set arbitrarily by a value assignment.
They are automatically increased when an image is scanned

All images that the scanner delivers from a page count as one scan. This includes the color separations as well as the partial images and composite images from the Image Processor.

%(S.SCANSCLASS)
%C

Total scans for the actual base profile. For each defined base profile, an individual scan counter is generated and is saved together with base profile definition.

%(S.SCANSDAY)
%$N

Number of scanned images since the beginning of the day.

This counter is reset to zero after the first start of a job after midnight.

%(S.SCANSEXPECTED)
%e

This function returns the number of images expected by the program for a batch.

This number can be specified in the basic profile or queried in the task at the beginning.

If this number is specified, it is checked for each scan and, if

	necessary, the process is stopped.
%(S.SCANSJOB) %n	Number of scanned Images in the batch.
%(S.SCANSMINUTE) %f	Number of scans per minute. This value is constantly updated every few seconds and gives the actual value for the speed, similar to a speedometer. Since several images can be created during one scan, this value does not correspond to the number of images per minute.
%(S.SCANSTOTAL) %$C	This function returns the number of all scanned images in all profiles.
%(S.USERCOUNT_) %(USERCOUNT_)	User-defined counters (1..9) can be defined in the event rules or sa task step. They are used to count documents for example. The value of the counter is stored to the current image. The variable behaves like an I-Variable. However, when the event rules are started, the counter is reset to 0.

4.5.1.7 Special Character Codes

These variables are used to format output, e.g. oo create a tab-delimited list.

Available %-Codes in alphabetical order:

Variable	Description
%(S.ASCII.BELL) %\a	Bell, beep
%(S.ASCII.BSLASH) %\\	Backslash (inverse slash)
%(S.ASCII.BSPACE) %\b	Backspace
%(S.ASCII.DQUOTE) %\"	Double quote
%(S.ASCII.ESCAPE) %\e	Escape sign (dec. 27)
%(S.ASCII.FFEED) %\f	Form feed
%(S.ASCII.HTAB) %\t	Horizontal tab
%(S.ASCII.LFEED) %\n	Line feed
%(S.ASCII.NULL) %\0	0 (ASCII sign zero)

%(S.ASCII.QMARK) %\?	Questionmark
%(S.ASCII.RETURN) %\r	Carriage return
%(S.ASCII.SMARK) %\'	Single quote
%(S.ASCII.VTAB) %\v	Vertical tab.
%(S.ASCII.Xn) **%\X** **%\x**	Where **n** is a one to five digit number, which is the UTF-16-Code in **decimal** interpretation. e.g. %\x8364 for € %\x169 for © %\x174 for ® %\x177 for ± %\x34 " for quotation mark %\x39 ' for apostrophe %\x91 and %\x93 for [ ] (square brackets) %\x60 and %\x62 for < > (angle brackets) %\x123 and %\x125 for { } (curly brackets)
%(S.UNICODE.BOM)	This function returns a Unicode byte order mark. This string indicates which encoding corresponds to the following data, e.g. EF BB BF for UTF-8 or FF FE for UTF-16 Little Endian.

4.5.1.8 Percent Codes for marks

These variables describe the marks in the batch or on the individual images.

Verfügbare %-Codes in alphabetischer Reihenfolge:

Variable	Description
%(S.MARK.BLUE) %(S.MARKBLUE)	Indicates whether an image has a blue mark, that means it causes an error prevents finalization. If there are several images in the image group, %(S.MARK.BLUE) returns a 1 if at least one of the images within the image group has a blue mark.
%(S.MARK.BLUE.*)	The variable can also be used with the color format and the picture number. Thus,%(S.MARK.BLUE.C.1) supplies the information to the first color image of the image group.
%(S.MARK.BLUEACTIVE)	Indicates whether images with blue marks are set in the batch.

%(S.MARK.BLUECOUNT)	Returns the number of blue marked images in the batch.
%(S.MARK.GREEN) %(S.MARKGREEN)	Indicates whether an image has a red mark, that means it is important. If there are several images in the image group, %(S.MARK.GREEN) returns a 1 if at least one of the images within the image group has a green mark.
%(S.MARK.GREEN.*)	The variable can also be used with the color format and the picture number. So %(S.MARK.GREEN.C.1) supplies the information to the first color image of the image group.
%(S.MARK.GREENACTIVE)	Indicates whether images with green marks are set in the batch.
%(S.MARK.GREENCOUNT)	Returns the number of green marked images in the batch.
%(S.MARK.RED) %(S.MARKRED)	Indicates whether an image has a red mark, that means it discarded during finalization. If there are several images in the image group, %(S.MARK.RED) returns a 1 if at least one of the images within the image group has a red mark.
%(S.MARK.RED.*)	The variable can also be used with the color format and the picture number. Thus,%(S.MARK.RED.C.1) supplies the information to the first color image of the image group.
%(S.MARK.REDACTIVE)	Indicates whether images with red marks are set in the batch.
%(S.MARK.REDCOUNT)	Returns the number of red marked images in the batch.
%(S.MARK.WHITE) %(S.MARKWHITE)	Indicates whether an image has a white mark, that means it is hidden. If there are several images in the image group, %(S.MARK.WHITE) returns a 1 if at least one of the images within the image group has a white mark.
%(S.MARK.WHITE.*)	The variable can also be used with the color format and the picture number. So %(S.MARK.WHITE.C.1) supplies the information to the first color image of the image group.
%(S.MARK.WHITEACTIVE)	Indicates whether images with white marks are set in the batch.
%(S.MARK.WHITECOUNT)	Returns the number of white marked images in the batch.

4.5.1.9 Percent Codes for the station

The following variables are assigned to a workstation, they are always available and valid for **all** profile.

⊟ **Userdefined variables**

Variables that are to refer to the whole station have the **prefix A**. A-Variables can be used by all tasks and all basic profiles. However, they are stored on the station and can not be transferred to a different station by transferring a stack or exporting a configuration. Only if the imported configuration is executed, new A-Variable will be created.

For example, the workgroup to which a scanstation belongs may be defined by **%(Workgroup).**

⊟ **System variables**

System variables are values that the program determines and then stores. They can be overwritten. The system variables start with the **prefix S**.

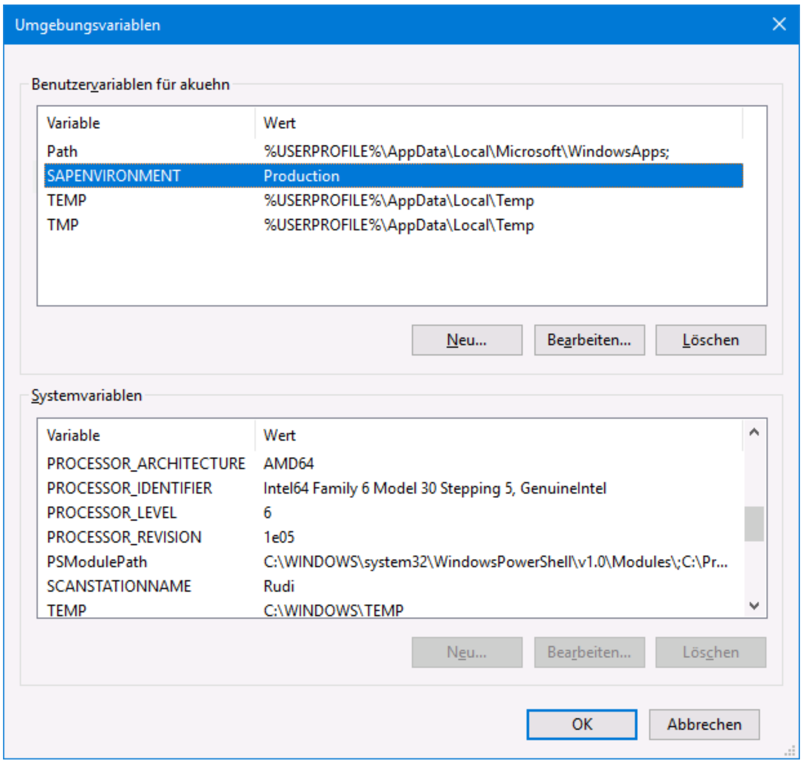
Available %-Codes in alphabetical order:

Variable	**Description**
%(S.BATESTOTAL) %$s	Returns the counter for the bates stamp
%(S.CMDPARAM_)	Returns the contents of the command line parameter 1 to 9. These values are also output in the statistics file in columns 21-29.
%(S.JOBNAME) %O %o	Liefert den Inhalt des Kommandozeilenparameters JOBNAME bzw. des eingestellten Wertes in den Jobdaten.
%(S.PAPERSAVEDDAY)	Returns the number of papers stored today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved. The counter will be reset with the first start of a job after midnight.
%(S.PAPERSCANNEDDAY)	Returns the number of papers scanned or loaded today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned. The counter will be reset with the first start of a job after midnight.
%(S.SCANSDAY) %$N	Number of scanned images since the beginning of the day. This counter is reset to zero after the first start of a job after midnight.
%(S.SCANSTOTAL) %$C	This function returns the number of all scanned images in all profiles.
%(S.USERPARAM_)	Returns the contents of the command line parameter USERPARAM 1 to 9. In contrast to CMDPARAM 1 to 9, these values are not passed to the statistics file.

%(SYS.ENV.*)

Returns the contents of any environment variable, e.g. the input request PROMPT by %(SYS.ENV.PROMPT).

User-defined environment variables can also be queried. In the example below, %(S.SYS.SAPENVIRONMENT) would return the value "Production"


Umgebungsvariablen

Benutzervariablen für akuehn

Variable	Wert
Path	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
SAPENVIRONMENT	Production
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Systemvariablen

Variable	Wert
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 30 Stepping 5, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	1e05
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
SCANSTATIONNAME	Rudi
TEMP	C:\WINDOWS\TEMP

System information

System variables are values that the program determines and then stores. They can be overwritten. The system variables start with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.BATESTOTAL) %$s	Returns the counter for the bates stamp
%(S.CMDPARAM_)	Returns the contents of the command line parameter 1 to 9. These values are also output in the statistics file in columns 21-29.
%(S.JOBNAME) %O %o	Liefert den Inhalt des Kommandozeilenparameters JOBNAME bzw. des eingestellten Wertes in den Jobdaten.
%(S.PAPERSAVEDDAY)	Returns the number of papers stored today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved. The counter will be reset with the first start of a job after midnight.

%(S.PAPERSCANNEDDAY)

Returns the number of papers scanned or loaded today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned.

The counter will be reset with the first start of a job after midnight.

%(S.SCANSDAY)
%$N

Number of scanned images since the beginning of the day.

This counter is reset to zero after the first start of a job after midnight.

%(S.SCANSTOTAL)
%$C

This function returns the number of all scanned images in all profiles.

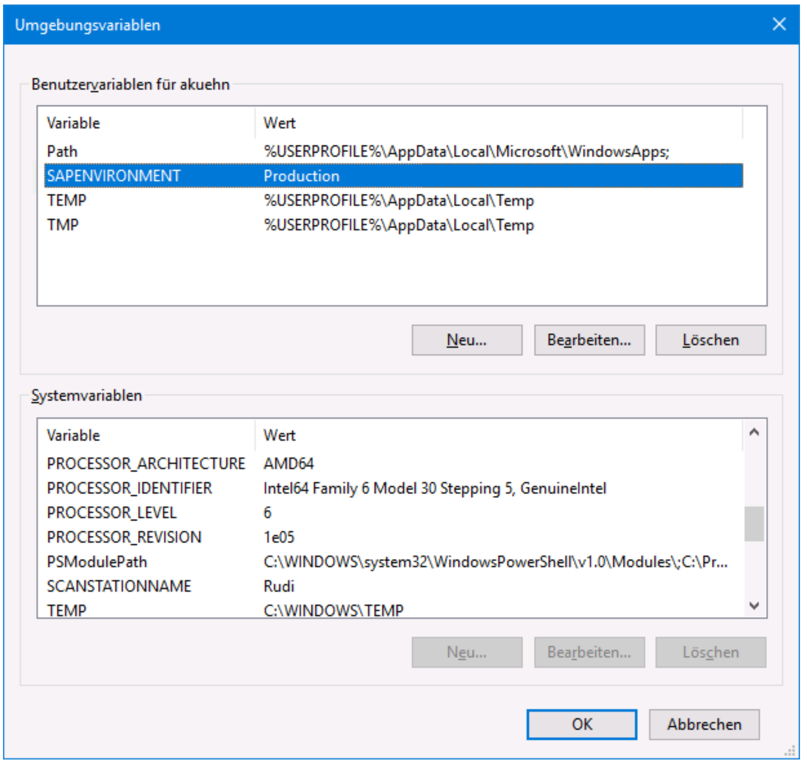
%(S.USERPARAM_)

Returns the contents of the command line parameter USERPARAM 1 to 9.
In contrast to CMDPARAM 1 to 9, these values are not passed to the statistics file.

%(SYS.ENV.*)

Returns the contents of any environment variable, e.g. the input request PROMPT by %(SYS.ENV.PROMPT).

User-defined environment variables can also be queried. In the example below, %(S.SYS.SAPENVIRONMENT) would return the value "Production"



Umgebungsvariablen

Benutzervariablen für akuehn

Variable	Wert
Path	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
SAPENVIRONMENT	Production
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Systemvariablen

Variable	Wert
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 30 Stepping 5, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	1e05
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
SCANSTATIONNAME	Rudi
TEMP	C:\WINDOWS\TEMP

4.5.1.9.1 Userdefined variables for the station

Variables that are to refer to the whole station have the **prefix A**. A-Variables can be used by all tasks and all basic profiles. However, they are stored on the station and can not be transferred to a different station by transferring a stack or exporting a configuration. Only if the imported configuration is executed, new A-Variable will be created.

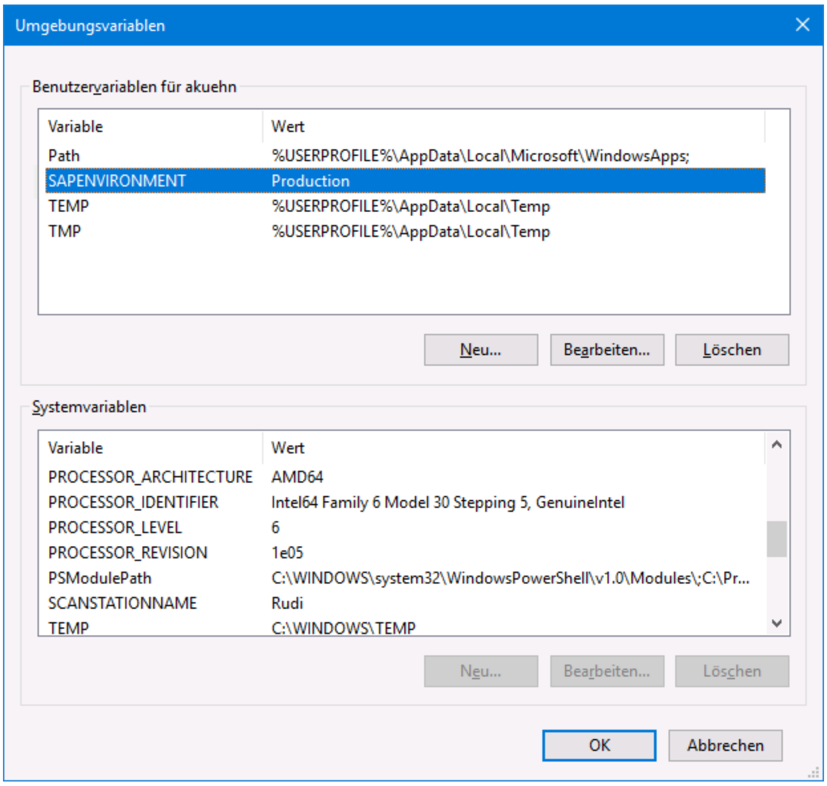
For example, the workgroup to which a scanstation belongs may be defined by **`%(Workgroup).`**

4.5.1.9.2 System variables for the station

System variables are values that the program determines and then stores. They can be overwritten. The system variables start with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	**Description**
`%(S.BATESTOTAL)` %$s	Returns the counter for the bates stamp
`%(S.CMDPARAM_)`	Returns the contents of the command line parameter 1 to 9. These values are also output in the statistics file in columns 21-29.
`%(S.JOBNAME)` %O %o	Liefert den Inhalt des Kommandozeilenparameters JOBNAME bzw. des eingestellten Wertes in den Jobdaten.
`%(S.PAPERSAVEDDAY)`	Returns the number of papers stored today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved. The counter will be reset with the first start of a job after midnight.
`%(S.PAPERSCANNEDDAY)`	Returns the number of papers scanned or loaded today. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned. The counter will be reset with the first start of a job after midnight.
`%(S.SCANSDAY)` %$N	Number of scanned images since the beginning of the day. This counter is reset to zero after the first start of a job after midnight.
`%(S.SCANSTOTAL)` %$C	This function returns the number of all scanned images in all profiles.
`%(S.USERPARAM_)`	Returns the contents of the command line parameter USERPARAM 1 to 9. In contrast to CMDPARAM 1 to 9, these values are not passed to the statistics file.
`%(SYS.ENV.*)`	Returns the contents of any environment variable, e.g. the input request PROMPT by %(SYS.ENV.PROMPT). User-defined environment variables can also be queried. In the example below, %(S.SYS.SAPENVIRONMENT) would return the value "Production"



Variable	Wert
Path	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
SAPENVIRONMENT	Production
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Variable	Wert
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 30 Stepping 5, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	1e05
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
SCANSTATIONNAME	Rudi
TEMP	C:\WINDOWS\TEMP

4.5.1.9.3 System information for the station

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.ADDPATH) %(ADDPATH) %(S.ADDDIR	Additional path of the images, defined in the path dialog, with %-codes replaced at start of task. It can be given as addendum to the Additional directory of saved images (without file names). In addition to the main directory , a subdirectory can also be specified.
%(S.APPTYPE)	3-letter shortcut for the working mode of the program. **DPU** as ccan application with user interface, visible on the screen **Fnn** as background process of type DpuFinalize. **Pnn** as background process of type DpuProcess. If there are more than one instance in the background the **nn** gives the number of the process, e..g. **F01** for background finalizer started first.
%(S.ASCII.BELL) %\a	Bell, beep

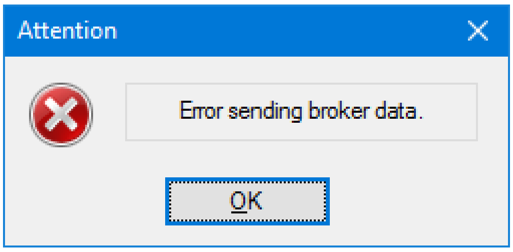
%(S.ASCII.BSLASH) %\\	Backslash (inverse slash)
%(S.ASCII.BSPACE) %\b	Backspace
%(S.ASCII.DQUOTE) %\"	Double quote
%(S.ASCII.ESCAPE) %\e	Escape sign (dec. 27)
%(S.ASCII.FFEED) %\f	Form feed
%(S.ASCII.HTAB) %\t	Horizontal tab
%(S.ASCII.LFEED) %\n	Line feed
%(S.ASCII.NULL) %\0	0 (ASCII sign zero)
%(S.ASCII.QMARK) %\?	Questionmark
%(S.ASCII.RETURN) %\r	Carriage return
%(S.ASCII.SMARK) %\'	Single quote
%(S.ASCII.VTAB) %\v	Vertical tab.
%(S.ASCII.Xn) **%\X** **%\x**	Where **n** is a one to five digit number, which is the UTF-16-Code in **decimal** interpretation. e.g. %\x8364 for € %\x169 for © %\x174 for ® %\x177 for ± %\x34 " for quotation mark %\x39 ' for apostrophe %\x91 and %\x93 for [ ] (square brackets) %\x60 and %\x62 for < > (angle brackets) %\x123 and %\x125 for { } (curly brackets)
%(S.BROKERSTATUS) %(S.BROKERSTAT)	Numerical return value of the broker call. Values **greater than or equal to 0** describe a status of the broker.

%$B

A return value of **0x9999** fills the variable %(S.BROKERSTRING) with the broker buffer.

Values **smaller than 0** signal errors and interrupt the further processing!.

!



Attention

Error sending broker data.

%(S.BROKERSTATUSPLUGIN)

Numeric return value of a PlugIn call, if this call is made in broker mode for PlugIns.

The returned values are the same as for %(S.BROKERSTATUS)

%(S.BROKERSTRING)
%$b

If the broker call returns **0x9999**, this variable will be filled with the broker buffer.

%(S.CMDMACROSTATUS)

Status of the last macro command. Not in use at the moment.

%(S.CMDTASKSTATUS)
%(S.COMMANDSTATUS)

Status of the last task command called. E.g. this can be the return value when printing a cover page or starting the finalization process.

1 success

0 an error occurred, e.g. a jump destination wasn't found

-1 the command couldn't be executed because a restriction denied this.

%(S.CPU)

Returns a text indicating the load of all CPUs. in %

%(S.CPU.LOAD)

Returns a number indicating the load of all CPUs. in %

%(S.DIR)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.DIR.*)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.G.2) for the second grayscale image.

The entry without suffix corresponds to the variable %(S.DIR).

%(S.DRIVE)

Drive for the images to save. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C: \Folder\SubFolder****File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

%(S.DRIVE.*)

Drive for the images to save. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C: \Folder\SubFolder****File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.C.1) for the first color image.

However, the value of this variable is the same for all images in a stack, since the drive specification is a fix part of the profile.

The entry without suffix corresponds to the variable %(S.DRIVE).

%(S.DRIVEDIR)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\ File.EXT

\\Server\Share\Folder\SubFolder\ File.EXT

%(S.DRIVEDIR.*)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\ File.EXT

\\Server\Share\Folder\SubFolder\ File.EXT

If multiple images are to be saved under the same filename, the exact name of the copy can be specified by specifying the letter for the color type and a number for the location. De facto, however, this value is the same for all image files in a scan because another path can only be created between two scans.

The entry without suffix corresponds to the variable %(S.DRIVEDIR)

%(S.ENDORSERCOUNT)

%(S.ENDORSERCOUNT) stores the maximum value of the %(S.ENDORSERCOUNTLAST). While %(S.ENDORSERCOUNTLAST) can be reset with the start of a new batch, %(S.ENDORSERCOUNT) retains the last value until %(S.ENDORSERCOUNTLAST) returns a higher value. It stores the maximum value over all Scan batches and also over several running times of DpuScan.

This variable behaves like a station variable. It will be the Maximum value of any scan profiles on the scan station.

%(S.ENDORSERCOUNTLAST)

Latest value of the endorser in the current batch.

Saves the last value of the variable %(S.IMAGE.ENDORSER) **increased by 1**.

This variable is treated as a J.-Variable, i.e. it belongs to the batch.

%(S.ENDORSERSTRINGLAST)

Last text printed by the printer unit of the scanner.

This variable is treated as a J.-Variable, i.e. it belongs to the batch.

During the scan a copy of this value will be saved on image level as %(S.IMAGE.ENDORSERTEXT).

%(S.EVENTMESSAGE)
%(S.MESSAGE)
%$M

Last output message of an event rule: If an event rule: "Broker/Batchfile message" has been defined, the corresponding output message can be changed via %(S.EVENTMESSAGE), for example in the log file as a line with the attribute "as event rule". This value can also be used as a description for the last event rule in the info window.

%(S.EVENTPATH)

Part of the target path created by the event rules.

This is the result of "New named folder" or "New automatic folder" action in the eventrules.

%(S.EVENTSTART)

Is 0 if the event rule were be processed at least once. In the time before that it indicates the state of the event rule processing.

%(S.EXPORTPATH)

Path which will be used for batch export.

%(S.EXT)

Type of the file to save, **without** the dot.

C:\Folder\SubFolder\File.**EXT**

If there are multiple color formats with the same name in the image group, but no special definition for this case, the automatic ending is indicated by three stars:

C:\Folder\SubFolder\File.*****

%(S.EXT.*)

Type of the file to save, **without** the dot.

C:\Folder\SubFolder\File.**EXT**

It is possible to specify different file formats for different color formats. If not specified, the regular ending for this file formats will be used, e.g. TIF for a TIFF.

Then %(S.EXT.B.1) would return "TIF".

If on the other hand the file extension is defined too, this value will be returned. For example: The second bitonal snippet (e.g. the address filed) shall be saved as #####.ADR, then %(S.EXT.B.2) will return "ADR"

The entry without suffix corresponds to the variable %(S.EXT).

%(S.EXTERNALDATASTATUS)

Return value of a PlugIn-DLL

%(S.FILE)

Name and type of the file to save, without drive and folder.

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\**File__*_.Ext**

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\**********.Ext**

%(S.FILE.*)

Name and type of the file to save, without drive and folder.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.FILE).

%(S.IMPORTPATH)

Path which will be used for batch import.

%(S.IN.PATH)
%(S.INPUTPATH)

Full path to the image source location, containing drive, directory file name and file type

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DRIVEDIR)
%(S.INPUTDRIVEDIR

Path to the image source location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\Name.Ext
\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DRIVE)
%(S.INPUTDRIVE)

Drive of the image source location. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\Name.Ext
\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DIR)
%(S.INPUTDIR)

Name of the folder containing the files to read without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

Verzeichnis, in dem die in dem die einzulesende Datei liegt, ohne Laufwerk und Dateiname.

C:**\Folder\SubFolder\**Name.Ext

\\Server\Share**\Folder\SubFolder\**Name.Ext

Die Angabe erfolgt **mit** einleitendem und abschließendem Schrägstrich.

%(S.IN.NAME)
%(S.INPUTNAME)

Name of the file to load, without drive, directory, dot and file type.

C:\Folder\SubFolder\**Name**.Ext

%(S.IN.EXT)
%(S.INPUTEXT)

Type of the file to load, without the dot.

C:\Folder\SubFolder\Name.**Ext**

%(S.LINEEND)

Character sequence defining the end of a line in a text output. The valid values are CRLF, LF or LFCR can be specified at the system information, see **LineEnd.**

%(S.MAINPATH)
%(MAINPATH)
%(S.MAINDIR

Main path of the images to store. It must be specified in the path dialog and hold at least a drive specification. In this dialog an additonal folder can be specified too.

%(S.MEMORY)

Information about the computer's memory as a text.

%(S.MEMORY.FREE)

Memory free in MB.

%(S.MEMORY.LOAD)

Memory loaded in Percent.

%(S.MEMORY.TOTAL)

Total memory in MB.

%(S.MEMORY.USED)

Memory used in in MB.

%(S.NAME)

Name of the file to save, without drive, directory, dot and file type.

C:\Folder\SubFolder\**File**.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\**File__*_**.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\*********.Ext

%(S.NAME.*)

Name of the file to save, without drive, directory, dot and file type.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\**Name__C1_**.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\**Bank**.Ext

The entry without suffix corresponds to the variable %(S.NAME).

%(S.OPENJOBBASE)

Contains the base path wich will be used to create or reopen a batch in OpenJob-Mode.

%(S.OPENJOBPATH)

Path of an open job, i.e. the open job base and the open job name. Please note: The structure of an open job folder differs from the target folder!

%(S.OUT.DIR)
%(S.REALDIR)
%P

Returns the folder for the images to save without drive an filespecification. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Dir).**

The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.OUT.DRIVE)
%(S.REALDRIVE)
%D

Returns the drive for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Drive).**

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.Ext

\\Server\Share\Folder\SubFolder\File.Ext

%(S.OUT.DRIVEDIR)
%(S.REALDRIVEDIR)

This function provides the storage forlder for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.DriveDir).**

This value will be returned **with** a trailing backslash.:

C:\Folder\SubFolder\File.Ext

\\Server\Share\Folder\File.Ext

%(S.OUT.EXT)
%E

Returns the file type (extension) **without** the dot. for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Ext).**

C:\Folder\SubFolder\File**.Ext**

%(S.OUT.EXT+)
%(S.REALEXT)

Returns the file type (extension) **inclusive** the dot.

C:\Folder\SubFolder\File**.Ext**

%(S.OUT.FILE)
%(S.REALFILE)

This function returnes the file of the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.File).**

C:\Folder\SubFolder\File.Ext

%(S.OUT.NAME)
%(S.REALFILENAME)
%(S.REALNAME)
%N

Returns the file name for the images to save without drive, folder and extension. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Name).**

C:\Folder\SubFolder\**File**.Ext

%(S.OUT.PATH)
%(S.REALPATH)
%F

This function provides the complete output path for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.PATH)**

C:\Folder\SubFolder\File.Ext

\\Server\Share\Folder\SubFolder\File.Ext

The "Real" part of the name should indicate that it is the name under which the file is actually saved when finalizing an open batch.

%(S.PATH)

Full path to the image storing location, containing drive, directory file name and file type

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\File__*_.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\********.Ext

%(S.PATH.*)

Full path to the image storing location, containing drive, directory file name and file type

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.PATH.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.PATH).

%(S.POLLBATCHGETFOUND)

This function returns the batch name of the found batch when the program searches for batches in polling mode.

%(S.POLLFILEFOUND)

When working in Polling mode and searching for files, this function returns the full path to the polling file found. This is not necessary the image location too.

%(S.POLLPATHBASE)

This function returns the defined path where the program starts to search for batches in polling mode.

The returned path is **without** trailing backslash

%(S.POLLPATHFOUND)

When working in Polling mode and searching for files, this function returns the folder holding the images to process. Depending on the settings the polling file may or may not be in this folder too.

The returned path is **with** trailing backslash.

%(S.PROGRAM.PATH)

Full path to the program location, containing drive, directory file name and file type

%(S.PROGRAM.DRIVEDI

Path to the program location including drive and directory but without file name and file type. This value will be returned **with** a trailing

R)

backslash.

%(S.PROGRAM.FILE)

Name and type of the program, without drive and folder.

%(S.PROGRAM.DRIVE)

This function returns the drive on which the program is installed.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.PROGRAM.DIR)

This function returns the folder in which the program is installed, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.PROGRAM.NAME)

Name of the program, without drive, directory, dot and file type.

%(S.PROGRAM.EXT)

Type of the program file, without the dot.

%(S.STATIONNAME)
%$S

This function returns the given station name. It can be set in the field station in the job data setup. It can be also set as a command line parameter.

This value can be used independently from the name of the computer named **STATIONNAME**.

%(S.STATUS.PATH)
%(S.STATUSPATH)
%(S.STATUS)
%(STATUSFILE)

Full path to the location of the status file, containing drive, directory file name and file type.

%(S.STATUS.DRIVEDIR)
%(S.STATUSDRIVEDIR)

Path to the location of the status file including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

%(S.STATUS.FILE)
%(S.STATUSFILE)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.DRIVE)
%(S.STATUSDRIVE)

This function returns the drive on which the status file is located.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.STATUS.DIR)
%(S.STATUSDIR)

This function returns the folder in which the on which the status file is located, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.STATUS.NAME)
%(S.STATUSNAME)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.EXT)
%(S.STATUSEXT)

Returns the extension (type) of the status file, without the dot.

%(S.TARGETBASEPATH)

Returns the destination path for the saved images,without the parts created by event rules or the polling process.

%(S.STARTPATH)

The returned text is **inclusive** the trailing backslash.

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

This value is the concatenation of %(S.MAINPATH) and %(S.ADDPATH).

%(S.TARGETPATH)

Targetpath for the images, **with** trailing Backslash.

This is the concatenation of %(S.TARGETBASEPATH) and %(S.EVENTPATH).

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

Because of the settings in the polling mode in between these components there may be the folder holding the images to read in.

%(S.TASKNAME)
%$D

Optional name of the document batch. It can be set in the field taskname in the job data setup.

It can be also set as a command line parameter called **TASKNAME**.

If it is not set, it is equal to the content of the variable %(S.TASKFILE).

%(S.TASKSTATUS)

Returns the current program state, possible values are:

"ERROR"	If an error occurs
"START"	at start of the job, before the first image is received.
"HALT"	when pausing the job
"STOP"	when a job was ended
"EXIT"	when leaving the program
"CONTINUE"	when continuing a batch, before the first image is received.
"WORK"	when processing images
"READY"	when waiting for the next job or other user action.

%(S.TIME.AMPM)
%p

Current locale's AM/PM for a 12-hour clock. Ante meridiem – post meridiem, before and after noon.

%(S.TIME.DATE)
%x

Date representation for current locale (e.g. dd.mm.yyyy for Germany or mm/dd/yyy in the United States of Amerika or yyyy-mm-dd in the People's Republic of China).

%(S.TIME.DATETIME)
%c

Date and time, representation appropriate for the locale.

%(S.TIME.DAYLONG)
%A

Full name of the weekday

%(S.TIME.DAYMONTH)

Day of month as decimal number, always two digits (01-31)

%d	
%(S.TIME.DAYSHORT) %a	Name of the weekday, abbreviated.
%(S.TIME.DAYWEEK) %w	Full name of the weekday, .(0-6; **Sunday** is 0)
%(S.TIME.DAYYEAR) %j	Day of the year as decimal number (001-366)
%(S.TIME.HOUR) %H	Hour in 24 hours format (00-23)
%(S.TIME.HOURAM) %I	Hour in 12 hours format (01-12)
%(S.TIME.MILLISEC) %@	Milliseconds as decimal number
%(S.TIME.MINUTE) %M	Minute as decimal number(00-59)
%(S.TIME.MONTH) %m	Month as decimal number(01-12)
%(S.TIME.MONTHLONG) %B	Full name of the month
%(S.TIME.MONTHSHORT) %b	Name of the month, abbreviated.
%(S.TIME.SECOND) %S	Second as dezimal number (00-59)
%(S.TIME.TIME) **%X**	Time in local format (e.g. hh:mm:ss for the USA)
%(S.TIME.TIMEZONE) %Z	Name of the time zone if known, empty otherwise
%(S.TIME.WEEKYEARM) %W	Number of the week, week starts on **Monday** (00-51)
%(S.TIME.WEEKYEARS) %U	Number of the week, week starts on **Sunday** (00-51)
%(S.TIME.YEARLAST) %z	Last digit of the year
%(S.TIME.YEARLONG) %Y	Year as dezimal number.

%(S.TIME.YEARSHORT)
%y

Year without century as dezimal number (00-99)

%(S.UNICODE.BOM)

This function returns a Unicode byte order mark. This string indicates which encoding corresponds to the following data, e.g. EF BB BF for UTF-8 or FF FE for UTF-16 Little Endian.

%(S.UNIQUE)

Postfix-String for folders and files which are created by a manual split operation. These files are now in between two other files numbered consecutively.

This value can be set in the system information of the program, there it is denoted as **FileUniquePostfix**.

%(S.USEREVENT_)

Returns the status of the uservent (1 - 32), contains "1" if the event has occured, otherwise "0".

User events are triggered by the corresponding toolbar command.

They are automatically reset after processing the event rules for a group of pictures.

In this way, actions can be executed depending on specific events.

%(S.USEREVENTLAST_)

Indicates whether the corresponding user event has occurred on the previous image.

That is, the value is set once for the image after the event and is reset to 0 after processing the event rules.

%(S.USERNAME)
%$U

This function returns the given username. It can be set in the field username in the job data setup. It can be also set as a command line parameter called **USERNAME**.

%(S.UTC.AMPM)

Current locale's AM/PM for a 12-hour clock. Ante meridiem – post meridiem, before and after noon

Use the **UTC**-Variables if the program is used at locations in different time zones

In the **GMT** zone (London) the local time is equal to the **UTC**-Time. In the **CET** zone (Berlin) the local time differs by 1 hour or 2 hours in the summer.

%(S.UTC.DATE)
%x

Date representation for current locale (e.g. dd.mm.yyyy for Germany or mm/dd/yyy in the United States of Amerika or yyyy-mm-dd in the People's Republic of China).

%(S.UTC.DATETIME)
%c

Date and time, representation appropriate for the locale.

%(S.UTC.DAYLONG)
%A

Full name of the weekday

%(S.UTC.DAYMONTH)
%d

Day of month as decimal number, always two digits (01-31)

%(S.UTC.DAYSHORT)
%a

Name of the weekday, abbreviated.

%(S.UTC.DAYWEEK) %w	Full name of the weekday, .(0-6; **Sunday** is 0)
%(S.UTC.DAYYEAR) %j	Day of the year as decimal number (001-366)
%(S.UTC.HOUR) %H	Hour in 24 hours format (00-23)
%(S.UTC.HOURAM) %I	Hour in 12 hours format (01-12)
%(S.UTC.MILLISEC) %@	Milliseconds as decimal number
%(S.UTC.MINUTE) %M	Minute as decimal number(00-59)
%(S.UTC.MONTH) %m	Month as decimal number(01-12)
%(S.UTC.MONTHLONG) %B	Full name of the month
%(S.UTC.MONTHSHORT) %b	Name of the month, abbreviated.
%(S.UTC.SECOND) %S	Second as dezimal number (00-59)
%(S.UTC.TIME) **%X**	Time in local format (e.g. hh:mm:ss for the USA)
%(S.UTC.TIMEZONE) %Z	Name of the time zone if known, empty otherwise
%(S.UTC.WEEKYEARM) %W	Number of the week, week starts on **Monday** (00-51)
%(S.UTC.WEEKYEARS) %U	Number of the week, week starts on **Sunday** (00-51)
%(S.UTC.YEARLAST) %z	Last digit of the year
%(S.UTC.YEARLONG) %Y	Year as dezimal number.
%(S.UTC.YEARSHORT) %y	Year without century as dezimal number (00-99)
%(SYS.COMPUTERNAME) %(SYS.STATIONNAME)	Content of the system variable COMPUTERNAME, that is the name of the computer as specified in the operating system.
%(SYS.DESKTOP)	Returns the content of the environment variable DESKTOP, that is the desktop folder of the user logged on currently. The returned path has

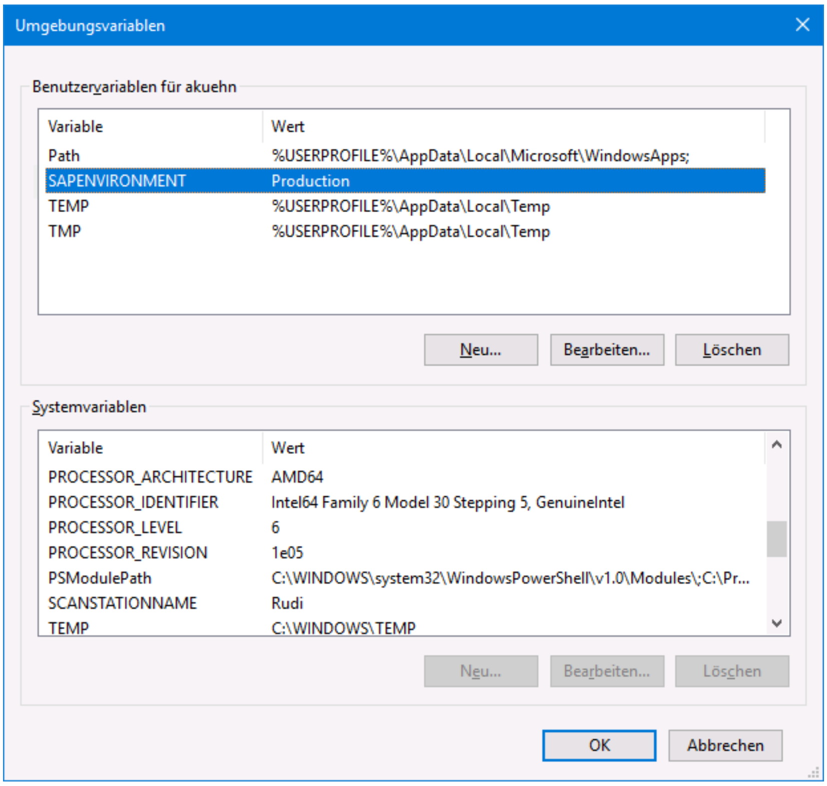
8.3-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\Desktop`

%(SYS.ENV.*)

Returns the contents of any environment variable, e.g. the input request PROMPT by %(SYS.ENV.PROMPT).

User-defined environment variables can also be queried. In the example below, %(S.SYS.SAPENVIRONMENT) would return the value "Production"



Umgebungsvariablen

Benutzervariablen für akuehn

Variable	Wert
Path	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
SAPENVIRONMENT	Production
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Systemvariablen

Variable	Wert
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 30 Stepping 5, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	1e05
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
SCANSTATIONNAME	Rudi
TEMP	C:\WINDOWS\TEMP

%(SYS.MYDOCUMENTS)

Returns the content of the environment variable DOCUMENTS, that is the documents folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\Documents`

%(SYS.TEMP)

Returns the content of the environment variable TEMP, that is the temporary folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\AppData\Local\Temp`

%(SYS.USERNAME)

Returns the content of the environment variable USERNAME, that is the name of the user logged on currently, for example

This variable is not to be confused with the program variable %(S.USERNAME) which can be set arbitrarily by command line or dialog.

4.5.1.10 Percent Codes of the Task

The following variables are assigned to a **task profile**.

⊟ **Userdefined variables**

Variables that are to relate to the current task profile have the **prefix T**. T variables can only be used in the current task. To see the contents of another task variable, you have to load the corresponding task.

Task and base profile variables are loaded immediately at the beginning of the task. Since each task profile must load a basic profile and this is almost always adapted to the corresponding task, a distinction is rarely made between task and basic profile variables. Only if you want to depict a slightly different sequence with the same basic data, different tasks can load the same basic profile.

For example, the distinction between the test environment and the production environment can be defined by task variables. So the storage location can be set by a T variable, which is then used in the *common* base profile to define the storage path.

⊟ **System variables**

Es gibt keine Systemvariablen, die sich besonderer Weise auf die Task beziehen.

⊟ **System information**

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.ACTUALTASK) %(S.TASKFILE) %$d	The task, that is currently executed. The task can be pre-selected automatically by the command line parameter as **TASKFILE**.or as **TODOFILE.** Please do not confuse this variable with %(S.TASKNAME) which is just a commenting text.
%(S.EVENTMESSAGE) %(S.MESSAGE) %$M	Last output message of an event rule: If an event rule: "Broker/Batchfile message" has been defined, the corresponding output message can be changed via %(S.EVENTMESSAGE), for example in the log file as a line with the attribute "as event rule". This value can also be used as a description for the last event rule in the info window.
%(S.EVENTSTART)	Is 0 if the event rule were be processed at least once. In the time before that it indicates the state of the event rule processing.
%(S.ISVISIBLE.BITONAL)	Indicates whether black-and-white images are displayed.
%(S.ISVISIBLE.COLOR)	Indicates whether color images are displayed.

%(S.ISVISIBLE.GRAY) Indicates whether gray scale images are displayed.

%(S.ISVISIBLE.MARK.BLUE) Indicates whether images are displayed that are marked blue.

%(S.ISVISIBLE.MARK.GREEN Indicates whether images are displayed that are marked green.

%(S.ISVISIBLE.MARK.NORMAL) Indicates whether images are displayed that are not marked.

%(S.ISVISIBLE.MARK.RED Indicates whether images are displayed that are marked red.

%(S.TASKNAME)
%$D

Optional name of the document batch. It can be set in the field taskname in the job data setup.

It can be also set as a command line parameter called **TASKNAME**.

If it is not set, it is equal to the content of the variable %(S.TASKFILE).

%(S.TASKSTATUS) Returns the current program state, possible values are:

"ERROR"	If an error occurs
"START"	at start of the job, before the first image is received.
"HALT"	when pausing the job
"STOP"	when a job was ended
"EXIT"	when leaving the program
"CONTINUE"	when continuing a batch, before the first image is received.
"WORK"	when processing images
"READY"	when waiting for the next job or other user action.

4.5.1.10.1 Userdefined variables for the task profile

Variables that are to relate to the current task profile have the **prefix T**. T variables can only be used in the current task. To see the contents of another task variable, you have to load the corresponding task.

Task and base profile variables are loaded immediately at the beginning of the task. Since each task profile must load a basic profile and this is almost always adapted to the corresponding task, a distinction is rarely made between task and basic profile variables. Only if you want to depict a slightly different sequence with the same basic data, different tasks can load the same basic profile.

For example, the distinction between the test environment and the production environment can be defined by task variables. So the storage location can be set by a T variable, which is then used in the *common* base profile to define the storage path.

4.5.1.10.2 System information for the task

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.ACTUALTASK) %(S.TASKFILE) %$d	The task, that is currently executed. The task can be pre-selected automatically by the command line parameter as **TASKFILE**.or as **TODOFILE.** Please do not confuse this variable with %(S.TASKNAME) which is just a commenting text.
%(S.EVENTMESSAGE) %(S.MESSAGE) %$M	Last output message of an event rule: If an event rule: "Broker/Batchfile message" has been defined, the corresponding output message can be changed via %(S.EVENTMESSAGE), for example in the log file as a line with the attribute "as event rule". This value can also be used as a description for the last event rule in the info window.
%(S.EVENTSTART)	Is 0 if the event rule were be processed at least once. In the time before that it indicates the state of the event rule processing.
%(S.ISVISIBLE.BITONAL)	Indicates whether black-and-white images are displayed.
%(S.ISVISIBLE.COLOR)	Indicates whether color images are displayed.
%(S.ISVISIBLE.GRAY)	Indicates whether gray scale images are displayed.
%(S.ISVISIBLE.MARK.BLUE)	Indicates whether images are displayed that are marked blue.
%(S.ISVISIBLE.MARK.GREEN	Indicates whether images are displayed that are marked green.
%(S.ISVISIBLE.MARK.NORMAL)	Indicates whether images are displayed that are not marked.
%(S.ISVISIBLE.MARK.RED	Indicates whether images are displayed that are marked red.
%(S.TASKNAME) %$D	Optional name of the document batch. It can be set in the field taskname in the job data setup. It can be also set as a command line parameter called **TASKNAME**. If it is not set, it is equal to the content of the variable %(S.TASKFILE).
%(S.TASKSTATUS)	Returns the current program state, possible values are:

"ERROR"	If an error occurs
"START"	at start of the job, before the first image is received.
"HALT"	when pausing the job
"STOP"	when a job was ended
"EXIT"	when leaving the program
"CONTINUE"	when continuing a batch, before the first image is received.
"WORK"	when processing images
"READY"	when waiting for the next job or other user action.

4.5.1.11 Percent Codes of th Base Profile

The following variables are assigned to a **base profile**.

⊟ **Userdefined variables**
Variables that are supposed to refer to the current base profile have the **prefix C**. The C is derived from the earlier designation Class.

C-variables can be used in all tasks that load the corresponding base profile at the beginning. To see the contents of another base profile variable, you must load the corresponding base profile in a task.

Frequent use for C-variables are counters, e.g. **`%(C.DocCount)`**, if you would like to count the documents across jobs.

⊟ **System variables**
System variables are values that the program determines and then stores. They can be overwritten. The system variables start with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
`%(S.SCANSCLASS)` %C	Total scans for the actual base profile. For each defined base profile, an individual scan counter is generated and is saved together with base profile definition.

⊟ **System information**
System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
`%(S.ACTUALCLASS)` %$A	The base profile holding all the task parameters of the task that is currently executed.

%(S.DOCCLASS) %$c	Document name assigned to the batch of documents. It can be specified in a field of the job data dialog. Also it can be given by a command line parameter referred to as **DOCCLASS**.
%(S.EVENTMESSAGE) %(S.MESSAGE) %$M	Last output message of an event rule: If an event rule: "Broker/Batchfile message" has been defined, the corresponding output message can be changed via %(S.EVENTMESSAGE), for example in the log file as a line with the attribute "as event rule". This value can also be used as a description for the last event rule in the info window.
%(S.EVENTSTART)	Is 0 if the event rule were be processed at least once. In the time before that it indicates the state of the event rule processing.
%(S.SCANSIDES)	Returns a 2 for duplex scanning and 1 for simplex scanning.

4.5.1.11.1 Userdefined variables for the base profile

Variables that are supposed to refer to the current base profile have the **prefix C**. The C is derived from the earlier designation Class.

C-variables can be used in all tasks that load the corresponding base profile at the beginning. To see the contents of another base profile variable, you must load the corresponding base profile in a task.

Frequent use for C-variables are counters, e.g. **%(C.DocCount)**, if you would like to count the documents across jobs.

4.5.1.11.2 System variables for the base profile

System variables are values that the program determines and then stores. They can be overwritten. The system variables start with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.SCANSCLASS) %C	Total scans for the actual base profile. For each defined base profile, an individual scan counter is generated and is saved together with base profile definition.

4.5.1.11.3 System information for the base profile

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description

%(S.ACTUALCLASS) %$A	The base profile holding all the task parameters of the task that is currently executed.
%(S.DOCCLASS) %$c	Document name assigned to the batch of documents. It can be specified in a field of the job data dialog. Also it can be given by a command line parameter referred to as **DOCCLASS**.
%(S.EVENTMESSAGE) %(S.MESSAGE) %$M	Last output message of an event rule: If an event rule: "Broker/Batchfile message" has been defined, the corresponding output message can be changed via %(S.EVENTMESSAGE), for example in the log file as a line with the attribute "as event rule". This value can also be used as a description for the last event rule in the info window.
%(S.EVENTSTART)	Is 0 if the event rule were be processed at least once. In the time before that it indicates the state of the event rule processing.
%(S.SCANSIDES)	Returns a 2 for duplex scanning and 1 for simplex scanning.

4.5.1.12 Percent Codes of the Batch

The following variables are assigned to the current job resp. of its batch of images.

Userdefined variables

Variables that are supposed to refer to the current batch or job have the **prefix J**.

J variables can only be used in the current task when the batch has been loaded. To have J variables available even after closing the stack, e.g. for output on the screen or in a statistics, they are not deleted until a new task is started.

Job variables are stored with the image batch and can thus be passed on to another station by export or import.

A typical job variable is the scan date when scanning incoming mail.

System variables

System variables are values that the program determines and then stores. They can be overwritten. The system variables start with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	**Description**
%(S.JOBDAYCOUNT)	Counter of all scanned batches of the day.
%(S.JOBNAME) %O %o	Liefert den Inhalt des Kommandozeilenparameters JOBNAME bzw. des eingestellten Wertes in den Jobdaten.
%(S.PAPERSAVED) %(PAPERSAVED)	Returns the number of papers stored during the current job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved.
%(S.PAPERSCANNED)	Returns the number of papers scanned or loaded during the current

%(S.PAPERCOUNT) **%(PAPERSCANNED)** %(PAPERCOUNT)	job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned.
%(S.SAVECOUNT_) %(SAVECOUNT_)	Counter for the stored images. There are counters 1 to 9. These counters can be set arbitrarily by a value assignment. They are increased automatically when an image is saved
%(S.SCANCOUNT_) %(SCANCOUNT_)	Counter for the scanned images. There are counters 1 to 9. These counters can be set arbitrarily by a value assignment. They are automatically increased when an image is scanned All images that the scanner delivers from a page count as one scan. This includes the color separations as well as the partial images and composite images from the Image Processor.
%(S.SCANSEXPECTED) %e	This function returns the number of images expected by the program for a batch. This number can be specified in the basic profile or queried in the task at the beginning. If this number is specified, it is checked for each scan and, if necessary, the process is stopped.
%(S.SCANSJOB) %n	Number of scanned Images in the batch.
%(S.USERCOUNT_) %(USERCOUNT_)	User-defined counters (1..9) can be defined in the event rules or sa task step. They are used to count documents for example. The value of the counter is stored to the current image. The variable behaves like an I-Variable. However, when the event rules are started, the counter is reset to 0.

Job variables

Job variables are variables that the program automatically creates and stores with each new job. They can be changed at any time. Like all job variables, they begin with the **prefix J**. and have a second prefix **JOB**, e.g. %(J.JOB.STATUS)

These values can be used to filter the list of available Open Jobs. They are also listed in the table in the DpuJobHandler

Available %-Codes in alphabetical order:

Variable	Description
%(J.JOB.COMPUTER)	This variable is pre-set with the name of the computer on which the batch was created. This value can be changed at any time. This value can be used to filter the batch list when loading a batch.
%(J.JOB.INFO)	This is not preset and variable can be changed at any time. It can be

	used to filter the batch list when loading a batch. This variable may be used to specify property of the batch e.g. the priority "Important" "Regular", ...
%(J.JOB.MODIFIED)	This variable indicates whether a batch has been modified. If the stack has not been modified, the variable contains the string "0". In the actions - Clear image, - mark / remove and remove image - Insert image the value is automatically changed to "1". When processing the event rules, the value is reset to "0". See also example
%(J.JOB.STATUS)	The program pre-sets this variable with OPEN but the value can be changed at any time. This value can be used to filter the batch list when loading a batch. This value can also be set in the DpuJobHandler. One can use any values specified in the JobStatus.txt file in the DpuJobHandler directory.
%(J.JOB.TYPE)	This is not preset and variable can be changed at any time. It can be used to filter the batch list when loading a batch. This variable is intended to signal a special property of the batch, e.g. the origin "External", "Scan Departement", "FAX Service", "Mail Service" ..."
%(J.JOB.USER)	This variable is preset with the name of the logged in user when creating the batch. It can be changed at any time. It can be used to filter the batch list when loading a batch.

System information

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	**Description**
%(S.ENDORSERCOUNTLAST)	Latest value of the endorser in the current batch. Saves the last value of the variable %(S.IMAGE.ENDORSER) **increased by 1**. This variable is treated as a J.-Variable, i.e. it belongs to the batch.
%(S.ENDORSERSTRINGLAST)	Last text printed by the printer unit of the scanner. This variable is treated as a J.-Variable, i.e. it belongs to the batch.

During the scan a copy of this value will be saved on image level as %(S.IMAGE.ENDORSERTEXT).

%(S.JOBCOUNT)
%$J

Job counter; Number of batches started. It is increased by one when a new batch is created, so already existing batches are not counted.

The counter is set independently of the selected profile.

The initial value can be specified at the start of the program.

%(S.JOBFILECOUNT)
%(S.FILECOUNT)

Returns the number of created files in the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after saving the file. It behaves as %(SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.

%(S.JOBFILECOUNTALL)
%(S.FILECOUNTALL)
%(S.IMAGEFILE)

Returns the number of created files in the current batch.

This counter is incremented only after saving the file.

%(S.JOBIMAGECOUNT)
%(S.IMAGECOUNT)

Returns the number of images in the current batch, **without** the ones deleted by **red marks**.

%(S.JOBIMAGECOUNTALL)
%(S.IMAGECOUNTALL)
%(S.IMAGEINFILEJOB)

Returns the number of images in the current batch.

%(S.JOBPATHCOUNT)
%(S.PATHCOUNT)

Returns the number of sub folders created n the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after closing a folder.

%(S.JOBPATHCOUNTALL)
%(S.PATHCOUNTALL)

Returns the number of sub folders created n the current batch.

%(S.MARK.BLUEACTIVE)

Indicates whether images with blue marks are set in the batch.

%(S.MARK.BLUECOUNT)

Returns the number of blue marked images in the batch.

%(S.MARK.GREENACTIVE)

Indicates whether images with green marks are set in the batch.

%(S.MARK.GREENCOUNT

Returns the number of green marked images in the batch.

)	
%(S.MARK.REDACTIVE)	Indicates whether images with red marks are set in the batch.
%(S.MARK.REDCOUNT)	Returns the number of red marked images in the batch.
%(S.MARK.WHITEACTIVE)	Indicates whether images with white marks are set in the batch.
%(S.MARK.WHITECOUNT)	Returns the number of white marked images in the batch.

Job information

This variable informs about a job. It cannot be set directly

Variable	Description
%(J.JOB.MODIFIED)	This variable indicates whether a batch has been modified. If the stack has not been modified, the variable contains the string "0". In the actions - Clear image, - mark / remove and remove image - Insert image the value is automatically changed to "1". When processing the event rules, the value is reset to "0". See also example

4.5.1.12.1 Userdefined variables for a batch

Variables that are supposed to refer to the current batch or job have the **prefix J**.

J variables can only be used in the current task when the batch has been loaded. To have J variables available even after closing the stack, e.g. for output on the screen or in a statistics, they are not deleted until a new task is started.

Job variables are stored with the image batch and can thus be passed on to another station by export or import.

A typical job variable is the scan date when scanning incoming mail.

4.5.1.12.2 System variables for a batch

System variables are values that the program determines and then stores. They can be overwritten. The system variables start with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	**Description**
%(S.JOBDAYCOUNT)	Counter of all scanned batches of the day.
%(S.JOBNAME) %O %o	Liefert den Inhalt des Kommandozeilenparameters JOBNAME bzw. des eingestellten Wertes in den Jobdaten.
%(S.PAPERSAVED) %(PAPERSAVED)	Returns the number of papers stored during the current job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a saved if at least one of the sides was saved.
%(S.PAPERSCANNED) %(S.PAPERCOUNT) **%(PAPERSCANNED)** %(PAPERCOUNT)	Returns the number of papers scanned or loaded during the current job. Paper-sheets will be counted here, regardless of single or double side scanning. A paper counts a scanned if at least one of the sides was scanned.
%(S.SAVECOUNT_) %(SAVECOUNT_)	Counter for the stored images. There are counters 1 to 9. These counters can be set arbitrarily by a value assignment. They are increased automatically when an image is saved
%(S.SCANCOUNT_) %(SCANCOUNT_)	Counter for the scanned images. There are counters 1 to 9. These counters can be set arbitrarily by a value assignment. They are automatically increased when an image is scanned All images that the scanner delivers from a page count as one scan. This includes the color separations as well as the partial images and composite images from the Image Processor.
%(S.SCANSEXPECTED) %e	This function returns the number of images expected by the program for a batch. This number can be specified in the basic profile or queried in the task at the beginning. If this number is specified, it is checked for each scan and, if necessary, the process is stopped.
%(S.SCANSJOB) %n	Number of scanned Images in the batch.
%(S.USERCOUNT_) %(USERCOUNT_)	User-defined counters (1..9) can be defined in the event rules or sa task step. They are used to count documents for example. The value of the counter is stored to the current image. The variable behaves like an I-Variable. However, when the event rules are started, the counter is reset to 0.

4.5.1.12.3 Job variables for a batch

Job variables are variables that the program automatically creates and stores with each new job. They can be changed at any time. Like all job variables, they begin with the **prefix J**. and have a second prefix **JOB**, e.g. `%(J.JOB.STATUS)`

These values can be used to filter the list of available Open Jobs. They are also listed in the table in the DpuJobHandler

Available %-Codes in alphabetical order:

Variable	Description
%(J.JOB.COMPUTER)	This variable is pre-set with the name of the computer on which the batch was created. This value can be changed at any time. This value can be used to filter the batch list when loading a batch.
%(J.JOB.INFO)	This is not preset and variable can be changed at any time. It can be used to filter the batch list when loading a batch. This variable may be used to specify property of the batch e.g. the priority "Important" "Regular", ...
%(J.JOB.MODIFIED)	This variable indicates whether a batch has been modified. If the stack has not been modified, the variable contains the string "0". In the actions - Clear image, - mark / remove and remove image - Insert image the value is automatically changed to "1". When processing the event rules, the value is reset to "0". See also example
%(J.JOB.STATUS)	The program pre-sets this variable with OPEN but the value can be changed at any time. This value can be used to filter the batch list when loading a batch. This value can also be set in the DpuJobHandler. One can use any values specified in the JobStatus.txt file in the DpuJobHandler directory.
%(J.JOB.TYPE)	This is not preset and variable can be changed at any time. It can be used to filter the batch list when loading a batch. This variable is intended to signal a special property of the batch, e.g. the origin "External", "Scan Departement", "FAX Service", "Mail Service" ..."
%(J.JOB.USER)	This variable is preset with the name of the logged in user when creating the batch. It can be changed at any time. It can be used to filter the batch list when loading a batch.

4.5.1.12.4 System information for a batch

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.ENDORSERCOUNTLAST)	Latest value of the endorser in the current batch. Saves the last value of the variable %(S.IMAGE.ENDORSER) **increased by 1**. This variable is treated as a J.-Variable, i.e. it belongs to the batch.
%(S.ENDORSERSTRINGLAST)	Last text printed by the printer unit of the scanner. This variable is treated as a J.-Variable, i.e. it belongs to the batch. During the scan a copy of this value will be saved on image level as %(S.IMAGE.ENDORSERTEXT).
%(S.JOBCOUNT) %$J	Job counter; Number of batches started. It is increased by one when a new batch is created, so already existing batches are not counted. The counter is set independently of the selected profile. The initial value can be specified at the start of the program.
%(S.JOBFILECOUNT) %(S.FILECOUNT)	Returns the number of created files in the current batch, **without** the ones deleted by **red marks**. This counter is incremented only after saving the file. It behaves as %(SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.
%(S.JOBFILECOUNTALL) %(S.FILECOUNTALL) %(S.IMAGEFILE)	Returns the number of created files in the current batch. This counter is incremented only after saving the file.
%(S.JOBIMAGECOUNT) %(S.IMAGECOUNT)	Returns the number of images in the current batch, **without** the ones deleted by **red marks**.
%(S.JOBIMAGECOUNTALL) %(S.IMAGECOUNTALL) %(S.IMAGEINFILEJOB)	Returns the number of images in the current batch.
%(S.JOBPATHCOUNT) %(S.PATHCOUNT)	Returns the number of sub folders created n the current batch, **without** the ones deleted by **red marks**. This counter is incremented only after closing a folder.

%(S.JOBPATHCOUNTALL) %(S.PATHCOUNTALL)	Returns the number of sub folders created n the current batch.
%(S.MARK.BLUEACTIVE)	Indicates whether images with blue marks are set in the batch.
%(S.MARK.BLUECOUNT)	Returns the number of blue marked images in the batch.
%(S.MARK.GREENACTIVE)	Indicates whether images with green marks are set in the batch.
%(S.MARK.GREENCOUNT)	Returns the number of green marked images in the batch.
%(S.MARK.REDACTIVE)	Indicates whether images with red marks are set in the batch.
%(S.MARK.REDCOUNT)	Returns the number of red marked images in the batch.
%(S.MARK.WHITEACTIVE)	Indicates whether images with white marks are set in the batch.
%(S.MARK.WHITECOUNT)	Returns the number of white marked images in the batch.

4.5.1.12.5 Job information for a batch

This variable informs about a job. It cannot be set directly

Variable	**Description**
%(J.JOB.MODIFIED)	This variable indicates whether a batch has been modified. If the stack has not been modified, the variable contains the string "0". In the actions - Clear image, - mark / remove and remove image - Insert image the value is automatically changed to "1". When processing the event rules, the value is reset to "0". See also example

4.5.1.13 Percent Codes of the Image File Folder

The following variables are assigned to the **directory** or folder holding the image.

⊟ **Userdefined variables**
Variables which are to be referenced to the current directory have the **prefix D**. D variables can only be used in the current task when the batch has been loaded.

Directory variables are stored with the image stack and can thus be passed on to another station by export or import.

A typical use is the number of the consignment when scanning incoming mail (which in turn may contain several documents (files) each with several pages (images).

⊟ **System variables**

There are no system variables that are specific to the current directory.

⊟ **System information**
System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.ADDPATH) %(ADDPATH) %(S.ADDDIR	Additional path of the images, defined in the path dialog, with %-codes replaced at start of task. It can be given as addendum to the Additional directory of saved images (without file names). In addition to the main directory , a subdirectory can also be specified.
%(S.DIR)	Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash. C:**\Folder\SubFolder\**File.Ext \\Server\Share**\Folder\SubFolder\**File.Ext
%(S.DIR.*)	Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash. C:**\Folder\SubFolder\File.Ext \\Server\Share\Folder\SubFolder\**File.Ext If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.G.2) for the second grayscale image. The entry without suffix corresponds to the variable %(S.DIR).
%(S.DRIVE)	Drive for the images to save. For drives and mapped drives this value

will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

%(S.DRIVE.*)

Drive for the images to save. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.C.1) for the first color image.

However, the value of this variable is the same for all images in a stack, since the drive specification is a fix part of the profile.

The entry without suffix corresponds to the variable %(S.DRIVE).

%(S.DRIVEDIR)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

%(S.DRIVEDIR.*)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

If multiple images are to be saved under the same filename, the exact name of the copy can be specified by specifying the letter for the color type and a number for the location. De facto, however, this value is the same for all image files in a scan because another path can only be created between two scans.

The entry without suffix corresponds to the variable %(S.DRIVEDIR)

%(S.EVENTPATH)

Part of the target path created by the event rules.

This is the result of "New named folder" or "New automatic folder" action in the eventrules.

%(S.EXPORTPATH)

Path which will be used for batch export.

%(S.IMPORTPATH)

Path which will be used for batch import.

%(S.IN.DIR)
%(S.INPUTDIR)

Name of the folder containing the files to read without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

Verzeichnis, in dem die in dem die einzulesende Datei liegt, ohne Laufwerk und Dateiname.

C:**\Folder\SubFolder\**Name.Ext

\\Server\Share**\Folder\SubFolder\**Name.Ext

Die Angabe erfolgt **mit** einleitendem und abschließendem Schrägstrich.

%(S.IN.DRIVE)
%(S.INPUTDRIVE)

Drive of the image source location. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\Name.Ext
\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DRIVEDIR)
%(S.INPUTDRIVEDIR

Path to the image source location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\Name.Ext
\\Server\Share\Folder\SubFolder\Name.Ext

%(S.MAINPATH)
%(MAINPATH)
%(S.MAINDIR

Main path of the images to store. It must be specified in the path dialog and hold at least a drive specification. In this dialog an additonal folder can be specified too.

%(S.OPENJOBBASE)

Contains the base path wich will be used to create or reopen a batch in OpenJob-Mode.

%(S.OPENJOBPATH)

Path of an open job, i.e. the open job base and the open job name. Please note: The structure of an open job folder differs from the target folder!

%(S.OUT.DIR)
%(S.REALDIR)
%P

Returns the folder for the images to save without drive an filespecification. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Dir).**

The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.OUT.DRIVE)
%(S.REALDRIVE)
%D

Returns the drive for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Drive).**

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.Ext

\\Server\Share\Folder\SubFolder\File.Ext

%(S.OUT.DRIVE)
%(S.REALDRIVE)
%D

Returns the drive for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Drive).**

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.Ext

\\Server\Share\Folder\SubFolder\File.Ext

%(S.OUT.NAME)
%(S.REALFILENAME)
%(S.REALNAME)
%N

Returns the file name for the images to save without drive, folder and extension. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Name).**

C:\Folder\SubFolder\**File**.Ext

%(S.OUT.PATH)
%(S.REALPATH)
%F

This function provides the complete output path for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.PATH)**

C:\Folder\SubFolder\File.Ext
\\Server\Share\Folder\SubFolder\File.Ext

The "Real" part of the name should indicate that it is the name under which the file is actually saved when finalizing an open batch.

%(S.PATH)

Full path to the image storing location, containing drive, directory file name and file type

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\File__*_.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\******.Ext**

%(S.PATH.*)

Full path to the image storing location, containing drive, directory file name and file type

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.PATH.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.PATH).

%(S.JOBPATHCOUNT)
%(S.PATHCOUNT)

Returns the number of sub folders created n the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after closing a folder.

%(S.JOBPATHCOUNTALL)
%(S.PATHCOUNTALL)

Returns the number of sub folders created n the current batch.

%(S.JOBFILECOUNT)
%(S.FILECOUNT)

Returns the number of created files in the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after saving the file. It behaves as %(SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.

%(S.JOBFILECOUNTALL)
%(S.FILECOUNTALL)
%(S.IMAGEFILE)

Returns the number of created files in the current batch.

This counter is incremented only after saving the file.

%(S.JOBIMAGECOUNT)
%(S.IMAGECOUNT)

Returns the number of images in the current batch, **without** the ones deleted by **red marks**.

%(S.JOBIMAGECOUNTALL)
%(S.IMAGECOUNTALL)
%(S.IMAGEINFILEJOB)

Returns the number of images in the current batch.

%(S.POLLPATHBASE)

This function returns the defined path where the program starts to search for batches in polling mode.

The returned path is **without** trailing backslash

%(S.POLLPATHFOUND)

When working in Polling mode and searching for files, this function returns the folder holding the images to process. Depending on the settings the polling file may or may not be in this folder too.

The returned path is **with** trailing backslash.

%(S.PROGRAM.DIR)

This function returns the folder in which the program is installed, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.PROGRAM.DRIVE)

This function returns the drive on which the program is installed.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.PROGRAM.DRIVEDIR)

Path to the program location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

%(S.PROGRAM.PATH)

Full path to the program location, containing drive, directory file name and file type

%(S.STATUS.DIR)
%(S.STATUSDIR)

This function returns the folder in which the on which the status file is located, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.STATUS.PATH)
%(S.STATUSPATH)
%(S.STATUS)
%(STATUSFILE)

Full path to the location of the status file, containing drive, directory file name and file type.

%(S.STATUS.DRIVE)
%(S.STATUSDRIVE)

This function returns the drive on which the status file is located.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.STATUS.DRIVEDIR)
%(S.STATUSDRIVEDIR)

Path to the location of the status file including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

%(S.TARGETBASEPATH)
%(S.STARTPATH)

Returns the destination path for the saved images,without the parts created by event rules or the polling process.

The returned text is **inclusive** the trailing backslash.

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

This value is the concatenation of %(S.MAINPATH) and %

(S.ADDPATH).

%(S.TARGETPATH)

Targetpath for the images, **with** trailing Backslash.

This is the concatenation of %(S.TARGETBASEPATH) and %(S.EVENTPATH).

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

Because of the settings in the polling mode in between these components there may be the folder holding the images to read in.

%(SYS.DESKTOP)

Returns the content of the environment variable DESKTOP, that is the desktop folder of the user logged on currently. The returned path has 8.3-convention and is **without** a trailing backslash, e.g.:

```
C:\Users\EBERHA~1\Desktop
```

%(SYS.MYDOCUMENTS)

Returns the content of the environment variable DOCUMENTS, that is the documents folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

```
C:\Users\EBERHA~1\Documents
```

%(SYS.TEMP)

Returns the content of the environment variable TEMP, that is the temporary folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

```
C:\Users\EBERHA~1\AppData\Local\Temp
```

4.5.1.13.1 Userdefined variables for a folder

Variables which are to be referenced to the current directory have the **prefix D**. D variables can only be used in the current task when the batch has been loaded.

Directory variables are stored with the image stack and can thus be passed on to another station by export or import.

A typical use is the number of the consignment when scanning incoming mail (which in turn may contain several documents (files) each with several pages (images).

4.5.1.13.2 System variables for a folder

There are no system variables that are specific to the current directory.

4.5.1.13.3 System information for a folder

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable **Description**

%(S.ADDPATH)
%(ADDPATH)
%(S.ADDDIR

Additional path of the images, defined in the path dialog, with %-codes replaced at start of task. It can be given as addendum to the

Additional directory of saved images (without file names). In addition to the main directory , a subdirectory can also be specified.

%(S.DIR)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.DIR.*)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.G.2) for the second grayscale image.

The entry without suffix corresponds to the variable %(S.DIR).

%(S.DRIVE)

Drive for the images to save. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder****File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

%(S.DRIVE.*)

Drive for the images to save. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder****File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.C.1) for the first color image.

However, the value of this variable is the same for all images in a stack, since the drive specification is a fix part of the profile.

The entry without suffix corresponds to the variable %(S.DRIVE).

%(S.DRIVEDIR)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

%(S.DRIVEDIR.*)

Path to the image storing location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\File.EXT

\\Server\Share\Folder\SubFolder\File.EXT

If multiple images are to be saved under the same filename, the exact name of the copy can be specified by specifying the letter for the color type and a number for the location. De facto, however, this value is the same for all image files in a scan because another path can only be created between two scans.

The entry without suffix corresponds to the variable %(S.DRIVEDIR)

%(S.EVENTPATH)

Part of the target path created by the event rules.

This is the result of "New named folder" or "New automatic folder" action in the eventrules.

%(S.EXPORTPATH)

Path which will be used for batch export.

%(S.IMPORTPATH)

Path which will be used for batch import.

%(S.IN.DIR)
%(S.INPUTDIR)

Name of the folder containing the files to read without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

Verzeichnis, in dem die in dem die einzulesende Datei liegt, ohne Laufwerk und Dateiname.

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

Die Angabe erfolgt **mit** einleitendem und abschließendem Schrägstrich.

%(S.IN.DRIVE)
%(S.INPUTDRIVE)

Drive of the image source location. For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.DRIVEDIR)
%(S.INPUTDRIVEDIR

Path to the image source location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.MAINPATH)
%(MAINPATH)
%(S.MAINDIR

Main path of the images to store. It must be specified in the path dialog and hold at least a drive specification. In this dialog an additonal folder can be specified too.

%(S.OPENJOBBASE)

Contains the base path wich will be used to create or reopen a batch in OpenJob-Mode.

%(S.OPENJOBPATH)

Path of an open job, i.e. the open job base and the open job name. Please note: The structure of an open job folder differs from the target folder!

%(S.OUT.DIR)
%(S.REALDIR)
%P

Returns the folder for the images to save without drive an filespecification. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Dir).**

The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.OUT.DRIVE)
%(S.REALDRIVE)
%D

Returns the drive for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Drive).**

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.Ext

\\Server\Share\Folder\SubFolder\File.Ext

%(S.OUT.DRIVE)
%(S.REALDRIVE)
%D

Returns the drive for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Drive).**

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash.

C:\Folder\SubFolder\File.Ext

\\Server\Share\Folder\SubFolder\File.Ext

%(S.OUT.NAME)
%(S.REALFILENAME)
%(S.REALNAME)
%N

Returns the file name for the images to save without drive, folder and extension. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Name).**

C:\Folder\SubFolder\**File**.Ext

%(S.OUT.PATH)
%(S.REALPATH)
%F

This function provides the complete output path for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.PATH)**

C:\Folder\SubFolder\File.Ext
\\Server\Share\Folder\SubFolder\File.Ext

The "Real" part of the name should indicate that it is the name under which the file is actually saved when finalizing an open batch.

%(S.PATH)

Full path to the image storing location, containing drive, directory file name and file type

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\File__*_.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\********.Ext

%(S.PATH.*)

Full path to the image storing location, containing drive, directory file name and file type

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.PATH.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.PATH).

%(S.JOBPATHCOUNT)
%(S.PATHCOUNT)

Returns the number of sub folders created n the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after closing a folder.

%(S.JOBPATHCOUNTALL)
%(S.PATHCOUNTALL)

Returns the number of sub folders created n the current batch.

%(S.JOBFILECOUNT)
%(S.FILECOUNT)

Returns the number of created files in the current batch, **without** the ones deleted by **red marks**.

This counter is incremented only after saving the file. It behaves as %(SAVECOUNT). This behavior must be taken into account when the variable is used, e.g. for Bates stamps.

%(S.JOBFILECOUNTALL)
%(S.FILECOUNTALL)
%(S.IMAGEFILE)

Returns the number of created files in the current batch.

This counter is incremented only after saving the file.

%(S.JOBIMAGECOUNT)
%(S.IMAGECOUNT)

Returns the number of images in the current batch, **without** the ones deleted by **red marks**.

%(S.JOBIMAGECOUNTALL)
%(S.IMAGECOUNTALL)
%(S.IMAGEINFILEJOB)

Returns the number of images in the current batch.

%(S.POLLPATHBASE)

This function returns the defined path where the program starts to search for batches in polling mode.

The returned path is **without** trailing backslash

%(S.POLLPATHFOUND)

When working in Polling mode and searching for files, this function returns the folder holding the images to process. Depending on the settings the polling file may or may not be in this folder too.

The returned path is **with** trailing backslash.

%(S.PROGRAM.DIR)

This function returns the folder in which the program is installed, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.PROGRAM.DRIVE)

This function returns the drive on which the program is installed.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.PROGRAM.DRIVEDIR)

Path to the program location including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

%(S.PROGRAM.PATH)

Full path to the program location, containing drive, directory file name and file type

%(S.STATUS.DIR)
%(S.STATUSDIR)

This function returns the folder in which the on which the status file is located, without drive and file name.

The returned value will have both, a **leading** and a **trailing** backslash.

%(S.STATUS.PATH)
%(S.STATUSPATH)
%(S.STATUS)

Full path to the location of the status file, containing drive, directory file name and file type.

%(STATUSFILE)

%(S.STATUS.DRIVE)
%(S.STATUSDRIVE)

This function returns the drive on which the status file is located.

For drives and mapped drives this value will be returned with the drive letter and a colon, but without the file. For UNCs the server and share will be returned. In both cases there is **no** trailing backslash

%(S.STATUS.DRIVEDIR)
%(S.STATUSDRIVEDIR)

Path to the location of the status file including drive and directory but without file name and file type. This value will be returned **with** a trailing backslash.

%(S.TARGETBASEPATH)
%(S.STARTPATH)

Returns the destination path for the saved images,without the parts created by event rules or the polling process.

The returned text is **inclusive** the trailing backslash.

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

This value is the concatenation of %(S.MAINPATH) and %(S.ADDPATH).

%(S.TARGETPATH)

Targetpath for the images, **with** trailing Backslash.

This is the concatenation of %(S.TARGETBASEPATH) and %(S.EVENTPATH).

C:\MainFolder\AddFolder\EventFolder\File.EXT

\\Server\Share\MainFolder\AddFolder\EventFolder\File.EXT

Because of the settings in the polling mode in between these components there may be the folder holding the images to read in.

%(SYS.DESKTOP)

Returns the content of the environment variable DESKTOP, that is the desktop folder of the user logged on currently. The returned path has 8.3-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\Desktop`

%(SYS.MYDOCUMENTS)

Returns the content of the environment variable DOCUMENTS, that is the documents folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\Documents`

%(SYS.TEMP)

Returns the content of the environment variable TEMP, that is the temporary folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\AppData\Local\Temp`

4.5.1.14 Percent Codes of the Image File

The following variables are assigned to the **file** holding one or more images.

⊟ **Userdefined variables**

Variables that are to be referenced to the current file have the **prefix F**. F variables can only be used in the current task when the batch has been loaded.

File variables are stored with the image batch and can be passed to another station by export or import.

A typical usage is the invoice number for multi-page invoices that are saved as a multi-page file.

⊟ **System variables**

There are no system variables that are specific to the current file.

⊟ **System information**

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.DIR)	Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash. C:**\Folder\SubFolder\**File.Ext \\Server\Share**\Folder\SubFolder\**File.Ext
%(S.DIR.*)	Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash. C:**\Folder\SubFolder\File.Ext \\Server\Share\Folder\SubFolder\**File.Ext If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.G.2) for the second grayscale image. The entry without suffix corresponds to the variable %(S.DIR).
%(S.EXT)	Type of the file to save, **without** the dot. C:\Folder\SubFolder\File.**EXT** If there are multiple color formats with the same name in the image group, but no special definition for this case, the automatic ending is indicated by three stars: C:\Folder\SubFolder\File.*******

%(S.EXT.*)

Type of the file to save, **without** the dot.

C:\Folder\SubFolder\File.EXT

It is possible to specify different file formats for different color formats. If not specified, the regular ending for this file formats will be used, e.g. TIF for a TIFF.

Then %(S.EXT.B.1) would return "TIF".

If on the other hand the file extension is defined too, this value will be returned. For example: The second bitonal snippet (e.g. the address filed) shall be saved as #####.ADR, then %(S.EXT.B.2) will return "ADR"

The entry without suffix corresponds to the variable %(S.EXT).

%(S.FILE)

Name and type of the file to save, without drive and folder.

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\File__*_.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\********.Ext

%(S.FILE.*)

Name and type of the file to save, without drive and folder.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.FILE).

%(S.IN.DIR)

%(S.INPUTDIR)

Name of the folder containing the files to read without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

Verzeichnis, in dem die in dem die einzulesende Datei liegt, ohne Laufwerk und Dateiname.

C: **\Folder\SubFolder\** Name.Ext

\\Server\Share**\Folder\SubFolder\** Name.Ext

Die Angabe erfolgt **mit** einleitendem und abschließendem Schrägstrich.

%(S.IN.EXT)
%(S.INPUTEXT)

Type of the file to load, without the dot.

C:\Folder\SubFolder\Name.**Ext**

%(S.IN.FILE)
%(S.INPUTFILE)

Name and type of the file to load, without drive and folder.

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.NAME)
%(S.INPUTNAME)

Name of the file to load, without drive, directory, dot and file type.

C:\Folder\SubFolder\**Name**.Ext

%(S.IN.PATH)
%(S.INPUTPATH)

Full path to the image source location, containing drive, directory file name and file type

C:\Folder\SubFolder\Name.Ext
\\Server\Share\Folder\SubFolder\Name.Ext

%(S.NAME)

Name of the file to save, without drive, directory, dot and file type.

C:\Folder\SubFolder\**File**.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\**File__*_**.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\**********.Ext

%(S.NAME.*)

Name of the file to save, without drive, directory, dot and file type.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\**Name__C1_**.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\**Bank**.Ext

The entry without suffix corresponds to the variable %(S.NAME).

%(S.OUT.EXT)
%E

Returns the file type (extension) **without** the dot. for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Ext).**

C:\Folder\SubFolder\File**.Ext**

%(S.OUT.EXT+)
%(S.REALEXT)

Returns the file type (extension) **inclusive** the dot.

C:\Folder\SubFolder\File**.Ext**

%(S.OUT.FILE)
%(S.REALFILE)

This function returnes the file of the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.File).**

C:\Folder\SubFolder\File.Ext

%(S.OUT.NAME)
%(S.REALFILENAME)
%(S.REALNAME)
%N

Returns the file name for the images to save without drive, folder and extension. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Name).**

C:\Folder\SubFolder\**File**.Ext

%(S.OUT.PATH)
%(S.REALPATH)
%F

This function provides the complete output path for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.PATH)**

C:\Folder\SubFolder\File.Ext
\\Server\Share\Folder\SubFolder\File.Ext

The "Real" part of the name should indicate that it is the name under which the file is actually saved when finalizing an open batch.

%(S.PROGRAM.EXT)
Type of the program file, without the dot.

%(S.PROGRAM.FILE)
Name and type of the program, without drive and folder.

%(S.PROGRAM.NAME)
Name of the program, without drive, directory, dot and file type.

%(S.PROGRAM.PATH)
Full path to the program location, containing drive, directory file name and file type

%(S.OUT.EXT)
%E

Returns the file type (extension) **without** the dot. for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Ext).**

C:\Folder\SubFolder\File**.Ext**

%(S.OUT.EXT+)
%(S.REALEXT)

Returns the file type (extension) **inclusive** the dot.

C:\Folder\SubFolder\File**.Ext**

%(S.STATUS.EXT)
%(S.STATUSEXT)

Returns the extension (type) of the status file, without the dot.

%(S.STATUS.FILE)
%(S.STATUSFILE)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.NAME)
%(S.STATUSNAME)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.PATH)
%(S.STATUSPATH)
%(S.STATUS)
%(STATUSFILE)

Full path to the location of the status file, containing drive, directory file name and file type.

4.5.1.14.1 Userdefined variables for a file

Variables that are to be referenced to the current file have the **prefix F**. F variables can only be used in the current task when the batch has been loaded.

File variables are stored with the image batch and can be passed to another station by export or import.

A typical usage is the invoice number for multi-page invoices that are saved as a multi-page file.

4.5.1.14.2 System variables for the a file

There are no system variables that are specific to the current file.

4.5.1.14.3 System information for a file

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable **Description**

%(S.DIR)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

%(S.DIR.*)

Name of the target folder for the files to write without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

C:**\Folder\SubFolder\**File.Ext

\\Server\Share**\Folder\SubFolder\**File.Ext

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.DRIVE.G.2) for the second grayscale image.

The entry without suffix corresponds to the variable %(S.DIR).

%(S.EXT)

Type of the file to save, **without** the dot.

C:\Folder\SubFolder\File.**EXT**

If there are multiple color formats with the same name in the image group, but no special definition for this case, the automatic ending is indicated by three stars:

C:\Folder\SubFolder\File.***

%(S.EXT.*)

Type of the file to save, **without** the dot.

C:\Folder\SubFolder\File.**EXT**

It is possible to specify different file formats for different color formats. If not specified, the regular ending for this file formats will be used, e.g. TIF for a TIFF.

Then %(S.EXT.B.1) would return "TIF".

If on the other hand the file extension is defined too, this value will be returned. For example: The second bitonal snippet (e.g. the address filed) shall be saved as ######.ADR, then %(S.EXT.B.2) will return "ADR"

The entry without suffix corresponds to the variable %(S.EXT).

%(S.FILE)

Name and type of the file to save, without drive and folder.

C:\Folder\SubFolder\File.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\File__*_.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\********.Ext

%(S.FILE.*)

Name and type of the file to save, without drive and folder.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.FILE).

%(S.IN.DIR)
%(S.INPUTDIR)

Name of the folder containing the files to read without drive and file name. The returned value will have both, a **leading** and a **trailing** backslash.

Verzeichnis, in dem die in dem die einzulesende Datei liegt, ohne Laufwerk und Dateiname.

C:**\Folder\SubFolder\**Name.Ext

\\Server\Share**\Folder\SubFolder\**Name.Ext

Die Angabe erfolgt **mit** einleitendem und abschließendem Schrägstrich.

%(S.IN.EXT)
%(S.INPUTEXT)

Type of the file to load, without the dot.

C:\Folder\SubFolder\Name.**Ext**

%(S.IN.FILE)
%(S.INPUTFILE)

Name and type of the file to load, without drive and folder.

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.IN.NAME)
%(S.INPUTNAME)

Name of the file to load, without drive, directory, dot and file type.

C:\Folder\SubFolder\**Name**.Ext

%(S.IN.PATH)
%(S.INPUTPATH)

Full path to the image source location, containing drive, directory file name and file type

C:\Folder\SubFolder\Name.Ext

\\Server\Share\Folder\SubFolder\Name.Ext

%(S.NAME)

Name of the file to save, without drive, directory, dot and file type.

C:\Folder\SubFolder\**File**.Ext

If there are multiple files with the same name in the image group, but no special definition for this case, the automatic ending is indicated by a suffix with a star:

C:\Folder\SubFolder\File__*_.Ext

If several file names have been defined for the image group, the name is indicated by stars:

C:\Folder\SubFolder\********.Ext

%(S.NAME.*)

Name of the file to save, without drive, directory, dot and file type.

If several images are to be stored under the same file name, e.g. because snippets or copies have been made, the exact name of the copy can be indicated by specifying the letter for the color type and a number for the position. For example one can use %(S.FILE.C.1) for the first color image.

While not specified at the file name definition in another way an automatic suffix will be used:

C:\Folder\SubFolder\Name__C1_.Ext

If there is a specification for a copy or snippet this one will be returned. For example on an invoice the part with the bank data can build the second cut-out and can be titled as "Bank".

Then %(S.PATH.B.2) would return:

C:\Folder\SubFolder\Bank.Ext

The entry without suffix corresponds to the variable %(S.NAME).

%(S.OUT.EXT)
%E

Returns the file type (extension) **without** the dot. for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Ext).**

C:\Folder\SubFolder\File.Ext

%(S.OUT.EXT+)
%(S.REALEXT)

Returns the file type (extension) **inclusive** the dot.

C:\Folder\SubFolder\File.Ext

%(S.OUT.FILE)
%(S.REALFILE)

This function returnes the file of the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.File).**

C:\Folder\SubFolder\File.Ext

%(S.OUT.NAME)
%(S.REALFILENAME)
%(S.REALNAME)
%N

Returns the file name for the images to save without drive, folder and extension. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Name).**

C:\Folder\SubFolder\**File**.Ext

%(S.OUT.PATH)
%(S.REALPATH)
%F

This function provides the complete output path for the image to be saved. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.PATH)**

C:\Folder\SubFolder\File.Ext
\\Server\Share\Folder\SubFolder\File.Ext

The "Real" part of the name should indicate that it is the name under which the file is actually saved when finalizing an open batch.

%(S.PROGRAM.EXT)

Type of the program file, without the dot.

%(S.PROGRAM.FILE)

Name and type of the program, without drive and folder.

%(S.PROGRAM.NAME)

Name of the program, without drive, directory, dot and file type.

%(S.PROGRAM.PATH)

Full path to the program location, containing drive, directory file name and file type

%(S.OUT.EXT)
%E

Returns the file type (extension) **without** the dot. for the images to save. It is unambiguous, i.e. it can only be called for a single image. It cannot be called for specific image in an image group like **%(S.Ext).**

C:\Folder\SubFolder\File.**Ext**

%(S.OUT.EXT+)
%(S.REALEXT)

Returns the file type (extension) **inclusive** the dot.

C:\Folder\SubFolder\File**.Ext**

%(S.STATUS.EXT)
%(S.STATUSEXT)

Returns the extension (type) of the status file, without the dot.

%(S.STATUS.FILE)
%(S.STATUSFILE)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.NAME)
%(S.STATUSNAME)

Name of the status file, without drive, directory, dot and file type.

%(S.STATUS.PATH)
%(S.STATUSPATH)
%(S.STATUS)
%(STATUSFILE)

Full path to the location of the status file, containing drive, directory file name and file type.

4.5.1.15 Percent Codes of the Image

The following variables are assigned to an **image** group or to a single image.

⊟ **Userdefined variables**
Variables that are to be related to the current image have the **prefix I**. I variables can only be used in the current task when the batch has been loaded.

Image variables are saved with the image file and can be passed on to another station by export or import.

A typical use is to hold a property, e.g. a counter, an OCR result, or a bar code. In general, it is recommended to save characteristics (first sheet of an invoice, a questionary, etc.) in this user-defined variable. In this way, separations can be reset or set manually.

System variables

There are no system variables that are specific to the current image.

System information

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	Description
%(S.IMAGE.CODING.*)	Coding mode (G4,JPG,JP2) of the current image. This is **not** the file type (TIFF, JFIF, PDF)
%(S.IMAGE.COLOR.*)	Color depth of the current image 0: unknown 1: Bitonal 2: Gray 3: Color
%(S.IMAGE.COLORINDEX.*)	Ordinal number of an image in an image group if there are several images of the same color type. For example if there two images in a scan 1: first image 2: second image. Together with the color type S.IMAGE.COLOR this forms a unique index inside the group.
%(S.IMAGE.DPI.*)	Resolution in dots per inch DPI, some samples are here.
%(S.IMAGE.DPIX.*)	Horizontal resolution in dots per inch DPI The old syntax of this variable %(S.IMAGE.DPI_X) should **no longer** be used..
%(S.IMAGE.DPIY.*)	Vertical resolution in dots per inch DPI Usually, horizontal and vertical resolution are always the same when scanning. For images from other sources, the resolutions may be different. The old syntax of this variable %(S.IMAGE.DPI_Y) should **no longer** be used.
%(S.IMAGE.ENDORSER.	Endorser number of the current image. This value is queried from the

***)**

machine for each scan. It is not necessarily continuous, since the scanner can set it e.g. in response to a user intervention or a counter event.

This value is the same for all images of one scan, i.e. all front and all back pictures.

%(S.IMAGE.ENDORSERSTRING.*)

Endorser text of the current image.

This value is the same for all images of one scan, i.e. all front and all back pictures.

%(S.IMAGE.HEIGHT.*)

Height of an image in pixels. The height depends on the selected resolution

%(S.IMAGE.ID.*)

Unique image counter within the batch.

%(S.IMAGE.INFO)

%K

Returns the ImageHeader for the current image.

%(S.IMAGE.INFOOLD)

%$K

Returns the ImageHeader for the previous image.

Comparing the previous and the current ImageHeader **%(S.IMAGE.INFO)**, short **%K**, in the Event rules allows to react on a change and to form a document break.

%(S.IMAGE.OCRFILE)

Filename of the text recognition (Full-Page-OCR).

%(S.IMAGE.POSINFILE.*)

Position of the image with in a file when loading files.

%(S.IMAGE.SAVED.*)

This function returns a 1 if the image was saved 0 otherwise.

%(S.IMAGE.SIDE)

%(S.PAGESIDE)

%s

Image information of the previously scanned image.

Sheet Information (Front or Back): The front is marked with "1" and the back with "2". If the sheet information is unknown (e.g., loading from directory), "0"; is output.

Note that this variable has a special behavior when enumerating formatting, ignoring zero and assigning the first value to the 1.

%s[Front, Back] returns 1 = front and 2 = back.

For other variables, the count starts at zero.

%(S.IMAGE.SIZE.*)

Size of the image in bytes, this value depends on color type and compression.

%(S.IMAGE.SIZEBITONAL)

%k

This function returns the size in bytes of the bitonal image. If there are only color and gray images it will be calculated nevertheless.

%(S.IMAGE.TIME)

Returns the time when the images was scanned.

%(S.IMAGE.WIDTH.*)

Width of an image in pixels. The height depends on the selected resolution

%(S.MARK.BLUE)
%(S.MARKBLUE)

Indicates whether an image has a blue mark, that means it causes an error prevents finalization. If there are several images in the image group, %(S.MARK.BLUE) returns a 1 if at least one of the images within the image group has a blue mark.

%(S.MARK.BLUE.*)

The variable can also be used with the color format and the picture number. Thus,%(S.MARK.BLUE.C.1) supplies the information to the first color image of the image group.

%(S.MARK.GREEN)
%(S.MARKGREEN)

Indicates whether an image has a red mark, that means it is important. If there are several images in the image group, %(S.MARK.GREEN) returns a 1 if at least one of the images within the image group has a green mark.

%(S.MARK.GREEN.*)

The variable can also be used with the color format and the picture number. So %(S.MARK.GREEN.C.1) supplies the information to the first color image of the image group.

%(S.MARK.RED)
%(S.MARKRED)

Indicates whether an image has a red mark, that means it discarded during finalization. If there are several images in the image group, %(S.MARK.RED) returns a 1 if at least one of the images within the image group has a red mark.

%(S.MARK.RED.*)

The variable can also be used with the color format and the picture number. Thus,%(S.MARK.RED.C.1) supplies the information to the first color image of the image group.

%(S.MARK.WHITE)
%(S.MARKWHITE)

Indicates whether an image has a white mark, that means it is hidden. If there are several images in the image group, %(S.MARK.WHITE) returns a 1 if at least one of the images within the image group has a white mark.

%(S.MARK.WHITE.*)

The variable can also be used with the color format and the picture number. So %(S.MARK.WHITE.C.1) supplies the information to the first color image of the image group.

%(S.UNIQUE)

Postfix-String for folders and files which are created by a manual split operation. These files are now in between two other files numbered consecutively.

This value can be set in the system information of the program, there it is denoted as **FileUniquePostfix**.

4.5.1.15.1 Userdefined variables for an image

Variables that are to be related to the current image have the **prefix I**. I variables can only be used in the current task when the batch has been loaded.

Image variables are saved with the image file and can be passed on to another station by export or import.

A typical use is to hold a property, e.g. a counter, an OCR result, or a bar code. In general, it is recommended to save characteristics (first sheet of an invoice, a questionary, etc.) in this user-defined variable. In this way, separations can be reset or set manually.

4.5.1.15.2 System variables for an image

There are no system variables that are specific to the current image.

4.5.1.15.3 System information for an image

System information is usually the **return value of functions** that are executed when the %-Code is queried. Thus, they are not real variables, i. Its value is not saved and can not be overwritten. The system information begins with the **prefix S**.

Available %-Codes in alphabetical order:

Variable	**Description**
%(S.IMAGE.CODING.*)	Coding mode (G4,JPG,JP2) of the current image. This is **not** the file type (TIFF, JFIF, PDF)
%(S.IMAGE.COLOR.*)	Color depth of the current image 0: unknown 1: Bitonal 2: Gray 3: Color
%(S.IMAGE.COLORINDEX.*)	Ordinal number of an image in an image group if there are several images of the same color type. For example if there two images in a scan 1: first image 2: second image. Together with the color type S.IMAGE.COLOR this forms a unique index inside the group.
%(S.IMAGE.DPI.*)	Resolution in dots per inch DPI, some samples are here.
%(S.IMAGE.DPIX.*)	Horizontal resolution in dots per inch DPI The old syntax of this variable %(S.IMAGE.DPI_X) should **no longer** be used..
%(S.IMAGE.DPIY.*)	Vertical resolution in dots per inch DPI Usually, horizontal and vertical resolution are always the same when scanning. For images from other sources, the resolutions may be different.

The old syntax of this variable %(S.IMAGE.DPI_Y) should **no longer** be used.

%(S.IMAGE.ENDORSER.*)

Endorser number of the current image. This value is queried from the machine for each scan. It is not necessarily continuous, since the scanner can set it e.g. in response to a user intervention or a counter event.

This value is the same for all images of one scan, i.e. all front and all back pictures.

%(S.IMAGE.ENDORSERSTRING.*)

Endorser text of the current image.

This value is the same for all images of one scan, i.e. all front and all back pictures.

%(S.IMAGE.HEIGHT.*)

Height of an image in pixels. The height depends on the selected resolution

%(S.IMAGE.ID.*)

Unique image counter within the batch.

%(S.IMAGE.INFO)

%K

Returns the ImageHeader for the current image.

%(S.IMAGE.INFOOLD)

%$K

Returns the ImageHeader for the previous image.

Comparing the previous and the current ImageHeader **%(S.IMAGE.INFO)**, short **%K**, in the Event rules allows to react on a change and to form a document break.

%(S.IMAGE.OCRFILE)

Filename of the text recognition (Full-Page-OCR).

%(S.IMAGE.POSINFILE.*)

Position of the image with in a file when loading files.

%(S.IMAGE.SAVED.*)

This function returns a 1 if the image was saved 0 otherwise.

%(S.IMAGE.SIDE)

%(S.PAGESIDE)

%s

Image information of the previously scanned image.

Sheet Information (Front or Back): The front is marked with "1" and the back with "2". If the sheet information is unknown (e.g., loading from directory), "0"; is output.

Note that this variable has a special behavior when enumerating formatting, ignoring zero and assigning the first value to the 1.

%s[Front, Back] returns 1 = front and 2 = back.

For other variables, the count starts at zero.

%(S.IMAGE.SIZE.*)

Size of the image in bytes, this value depends on color type and compression.

%(S.IMAGE.SIZEBITONAL)
%k

This function returns the size in bytes of the bitonal image. If there are only color and gray images it will be calculated nevertheless.

%(S.IMAGE.TIME)

Returns the time when the images was scanned.

%(S.IMAGE.WIDTH.*)

Width of an image in pixels. The height depends on the selected resolution

%(S.MARK.BLUE)
%(S.MARKBLUE)

Indicates whether an image has a blue mark, that means it causes an error prevents finalization. If there are several images in the image group, %(S.MARK.BLUE) returns a 1 if at least one of the images within the image group has a blue mark.

%(S.MARK.BLUE.*)

The variable can also be used with the color format and the picture number. Thus,%(S.MARK.BLUE.C.1) supplies the information to the first color image of the image group.

%(S.MARK.GREEN)
%(S.MARKGREEN)

Indicates whether an image has a red mark, that means it is important. If there are several images in the image group, %(S.MARK.GREEN) returns a 1 if at least one of the images within the image group has a green mark.

%(S.MARK.GREEN.*)

The variable can also be used with the color format and the picture number. So %(S.MARK.GREEN.C.1) supplies the information to the first color image of the image group.

%(S.MARK.RED)
%(S.MARKRED)

Indicates whether an image has a red mark, that means it discarded during finalization. If there are several images in the image group, %(S.MARK.RED) returns a 1 if at least one of the images within the image group has a red mark.

%(S.MARK.RED.*)

The variable can also be used with the color format and the picture number. Thus,%(S.MARK.RED.C.1) supplies the information to the first color image of the image group.

%(S.MARK.WHITE)
%(S.MARKWHITE)

Indicates whether an image has a white mark, that means it is hidden. If there are several images in the image group, %(S.MARK.WHITE) returns a 1 if at least one of the images within the image group has a white mark.

%(S.MARK.WHITE.*)

The variable can also be used with the color format and the picture number. So %(S.MARK.WHITE.C.1) supplies the information to the first color image of the image group.

%(S.UNIQUE)

Postfix-String for folders and files which are created by a manual split operation. These files are now in between two other files numbered consecutively.

This value can be set in the system information of the program, there it is denoted as **FileUniquePostfix**.

4.5.1.16 Percent Codes Image Header

With some scanners, additional information can be queried from the scanner like date, time, counters or pressed keys at the device

These information are kept in a unique formatted data block called ImageHeader.

Variable	Description
%(S.IMAGE.INFO)	Returns the ImageHeader for the current image.
%K	
%(S.IMAGE.INFOOLD)	Returns the ImageHeader for the previous image.
%$K	Comparing the previous and the current ImageHeader **%(S.IMAGE.INFO)**, short **%K**, in the Event rules allows to react on a change and to form a document break.

Hints:

The program can pick out a certain information by cutting out a part of the image header using a special formatter. for the following samples the short form of the variable is used:

%K[000000] 'F' (Front) or 'R' (Rear)

%K[007016] Sequential ID, with leading spaces

%K[104151] ImageAdress, where the counters (level 1 - 4) will have 10 digit, with leading spaces and separated bay dots. Depending on the device, the counters in the image address can be set by the index function for certain events.

%K[175180] Date (Month/Day/Year)

%K[189194] Time (Hour/Minute/Second)

If the scanner has the appropriate detection capabilities, these values are also often queried:

%K[298298] Found Patchcode: 0,T,2,3,1,4 or 6

%K[300300] Found Patchcode, 0,1,2,3,4,5 or 6

%K[348348] Recognized color type, 0= unknown, 1=black & white, 2=color

%K[388388] Found Marker, 0, 1-3

4.5.1.16.1 ImageHeader

The ImageHeader is a 512 byte block of text that is supplied with each scanned image. It contains information about the printed text from the endorser, counters, recognition results (patch code), image parameters, etc.

The first 256 bytes usually correspond to this structure:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17						
F	**r**	**o**	**n**	**t**		**#**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**0**							
18	19	20	21	22	23	24	25	26	27	88	29	30	31	32	33	34	35	36					
L	**e**	**n**	**g**	**t**	**h**		**=**		**-**	**-**	**-**	**-**	**-**	**-**	**-**	**0**							
37	38	39	40	41	42	43	44	45	46	47													
L	**e**	**v**	**e**	**l**		**=**		**-**	**0**														
48	49	50	51	52	53	54	55	56															
M	**o**	**d**	**e**	**=**		**-**	**0**	**lf**															
57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
L	**i**	**n**	**e**		**L**	**e**	**n**	**g**	**t**	**h**		**=**		**-**	**-**	**-**	**-**	**-**	**-**	**-**	**0**		
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	
P	**a**	**g**	**e**		**L**	**e**	**n**	**g**	**t**	**h**		**=**		**-**	**-**	**-**	**-**	**-**	**-**	**-**	**0**	**lf**	
104	105	106	107	108	109	...	118	119	120	...	129	130	131	...	140	141	142	...	151	152	153		
I	**A**		**=**		**-**	**-**	**0**	**.**	**-**	**-**	**0**	**.**	**-**	**-**	**0**	**.**	**-**	**-**	**0**				
154	155	156	157	158																			
-	**0**	**-**	**0**	**lf**																			
159	160	161	162	163	164	165	166	167															
C	**m**	**p**		**=**		**-**	**0**																
168	169	170	171	172	173	174	175	176	177	178	179	180	181										
D	**a**	**t**	**e**		**=**		**m**	**m**	**d**	**d**	**y**	**y**											
182	183	184	184	186	187	188	189	190	191	192	193	194	195										
T	**i**	**m**	**e**		**=**		**h**	**h**	**m**	**m**	**s**	**s**											
196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215				
R	**o**	**l**	**l**		**#**		**=**		**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**0**					
216	217	218	219	220	221	222	223																
R	**e**	**s**	**=**	**-**	**-**	**0**																	
224	225	226	227	228	229																		
B	**o**	**=**	**-**	**0**																			
230	231	232	233	234	235	236	237	238															
S	**k**	**=**	**-**	**-**	**-**	**0**		**lf**															
239	...	255																					
nu	**nu**	**nu**																					

The further structure depends heavily on the scanner, not all values are always supported:

256	257	258	...	287												
:	lf	nu	nu	nu												
288	289	290	291	292	293	294	295	296	297	298	299	300	301			
P	a	t	c	h	c	o	d	e	=	T	-	1	-			
302	303	304	305	306	307	308	309	310	311	312	313	314				
E	n	d	o	r	s	e	d	=	-	-	0	-				
315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331
B	i	t	s	P	e	r	P	i	x	e	l	=	-	2	4	-
332	333	334	335	336	337	338	339	340	341							
D	u	s	t	=	-	-	0	-	-							
342	343	344	345	346	347	348	349									
T	y	p	e	=	-	0	-									
350	351	352	353	354	355	356	357									
D	r	o	p	=	-	0	-									
358	359	360	361	362	363	364	365	366	367	368	369	370	371			
C	o	m	p	=	-	0	.	-	0	.	-	0	-			
372	373	374	375	376	377	378	379									
C	o	l	s	=	-	0	-									
380	381	382	383	384	385	386	387	388	389	390						
M	a	r	k	e	r	=	-	0	-	-						
391	392	393	394	395	396											
D	f	=	-	0	-											
431	432	433	434													
E	p	=	0													

The fields not listed are generally filled with the NULL character.

Remark

Following units are used here:

18 – 36	Length	Image size in Bytes
37 – 47	Level	Document Level 0 to 3
48 – 56	Mode	Processing Mode 0 to 15
57 – 80	Line Length	Line length in Pixel
81 – 103	Page Length	Page length in Pixel
104 – 153	IA	Image address, where
109-118		Counter 4 "Rollcounter"
120-129		Counter 3 "Volume Counter"
131-140		Counter 2 "Chapter Counter"
142-151		Counter 1 "Page Counter"

155		Momentary Flag, 0 oder 1
157		Latched (permanent) Flag, 0 oder 1
159 – 167	Cmp	Compression 0=uncompressed, 1=G3(1-dim), 2=G3(2-dim), 4=G4
168 – 181	Date	Date in format MMDDYY
182 – 195	Time	Date in format HHMMSS
196 – 215	Roll	Addidional Counzter (film roll)
216 – 223	Res	resolution in in DPI
224 – 229	BO	unused
230 – 237	Sk	Skew, 1=Image is skewed, but de-skew is not done.
238 – 243	Pol	unused

If available, the following applies to the other scanner-dependent values:

288 – 301	Patchcode	Byte 298: 0,T,2,3,1,4 or 6 Byte 300: 0,1,2,3,4,5 or 6
302 – 314	Endorsed	1, if page was endorsed (printed)
315 – 331	BitsPerPixel	1, 8 or 24 BPP (B/W, Gray, Color)
332 - 341	Dust	Probably dust in the scanner, vertical lines in images.
342 – 349	Type	Color Detection: 0= Unknown, 1=Non Color(bitonal), 2=Color
350 – 357	Drop	Color detection: 1= superfluous image (can be deleted)
358 – 371	Comp	Color Detection: 0-99% each. Nonwhite parts of the image: From left to right Bitonal, Gray and Color. Parts may be counted more than once.
372 – 379	Cols	Color recognition: Number of clearly distinguishable colors
380 – 388	Marker	Color Marker Detection: Byte 388: Marker 1,2,3 found, 0 = no Marker
391 – 395	Df	Doublefeed Byte 396: 0 = no doublefeed, 1 = at page start, 2 = at page end
431 – 434	Ep	Empty Page Byte 434: 0 or 1 = Page is empty.

Thees values can be queried by the corresponding percent code using the cutout format specifier.

4.5.1.17 Endorser Codes

The% codes in this area are **format descriptors** within an imprintertext. They are not variables in the actual sense, since they are neither stored nor calculated. They are passed on to the scanner instead. If necessary and possible, they are converted into the corresponding command of the respective device.

Double percent sign
There are different placeholders for the endorser text, depending on the device. If these placeholders begin with a percentage sign, these percentages must be masked by another percentage sign. For example, The Endorser control character %D (for example, to output the date), it must be specified as %%D on the endorser page in the scanner settings so that it is not interpreted as the %Code for the target drive.

For many devices the "u" is used for a page counter:

Variable	Description
%%u	%u within an imprinter text indicates the position of the automatic **imprinter counter**. The scanner driver automatically increases this counter after each use. Without further indications behind this field code, the length of the number is variable and from the value of the counters
%%0nu	As %u, however always **n**-digits long and with leading **zeros**. Example: %04u and counter=11 result as: 0011
%%nu	As %u, however always **n**-digits long but without leading zeros Example: %4u and counter=11 result as: " 11"
%%-nu	As %nu, however the number flush left within the defined text field: Example: %-4u and counter=11 result as: "11 "

These placeholders work only for the endorser text, they are not converted at any other point in the program. You can find more placeholders here.

The printed text can be queried by the corresponding variables: **`%(S.ENDORSERCOUNT), %(S.ENDORSERCOUNTLAST), %(S.ENDORSERTEXTLAST`**)

4.5.1.18 Frequently used Percent Codes

These variables are often used:

Short	Alternative	Description
%n	**%(S.SCANSJOB)**	Number of the image in the batch
%s	**%(S.IMAGE.SIDE)**	Side 1 or 2
%k	**%(S.IMAGE.SIZEBITONAL)**	Size of the image, less than ~3500 may be empty

%R1	%(S.BAR1)	Barcode 1-9
%T	%(S.PATCH)	Patchcode
%K	%(S.IMAGE.INFO)	ImageHeader
	%(S.OCR1)	Text recognition field 1-n
	%(S.SCANCOUNT1)	Number of scanned images
	%(S.PAPERSCANNED)	Number of scanned papers
%$N	%(S.JOBCOUNT)	Number of jobs
%$A	%(S.ACTUALCLASS)	Name of the current base profile
%O	%(S.JOBNAME)	Name of the current job
%F	%(S.OUT.PATH)	Full path to the image file
%D	%(S.OUT.DRIVE)	Drive of the image file
%P	%(S.OUT.DIR)	Path (without name) of the image file
%N	%(S.OUT.NAME)	Name of the image file
%E	%(S.OUT.EXT)	Extension or type of the image file
	%(S.STATUS.PATH)	Path of the status file (= Index file, Import file, ...)
%c	%(S.TIME.DATETIME)	Date and time (watch out the colons!)
%H%M%S	%(S.TIME.xxx)	Time stamp without colon
%s%m%y	%(S.TIME.xxx)	date stamp without the century
%\t	%(S.ASCII.TAB)	Horizontal Tab
%\f	%(S.ASCII.FFEED)	form feed
%%		the percent sign itself.

4.5.1.19 Short forms of some variable

For these variables, there is a short notation without decoration, i.e. without leading % (at begin and) at end.

Short	Meaning
%$A	Name of the base profile
%$B	Broker status
%$C	Scanns in all profiles
%$D	Name of the Task profile
%$J	Job -Counter
%$K	ImageHeader of the previous image.

%$M	Message fired by the eventrules
%$N	Scanns of the day
%$R	Barcode of the previous image.
%$S	Station name
%$T	Patchcode of the previous image.
%$U	Username
%$b	Broker message
%$c	Optional document class
%$d	Name of Taskprofile
%$s	Batestamp counter

%@	Milliseconds (000-999)
%A	Name of the Weekday (long)
%B	Name of the month (long)
%C	Scanns of the current base profile
%D	Destination drive
%E	Extension (file type) of the image file
%F	Full path to the image file
%H	Hour of day (00-23)
%I	Hour of day (01-12)
%K	ImageHeader
%M	Minute of the hour (00-59)
%N	Image file name w/o Drive, folder and extension
%O	Job name
%P	Target directory without drive and file
%R	Barcode on the current image
%S	Second of the minute (00-59)
%T	Patchcode on the current image
%U	Week of the year (Sunday = start of the week)
%W	Week of the year (Monday = start of the week)
%X	Time stamp (watch out for the colon!)
%Y	Year with century (4 digits)
%Z	Name of the time zone

%a	Name of the Weekday (abbreviated)
%b	Name of the month (abbreviated)
%c	Date and time stamp (watch out for the colon!)
%d	Tag des Monats (01-31)
%e	Expected scans
%f	Scanns per minute
%j	Day o the year (001-266)
%k	Site of the black and whit image
%m	Month fo the year (01-12)
%n	Scanns in the batch
%o	Job name
%p	Daytime indicator AM/PM
%r	Number of barcodes found
%s	Side (front, rear) of the image
%w	Weeday as number Sunday=0
%x	Date
%y	Year without century 2 digits

%z Last digit of the year.

4.5.1.20 Percent Codes of the Operating System

The variables listed here result from the interrogation of the environment variables of the operating system:

%(SYS.USERNAME)

Returns the content of the environment variable USERNAME, that is the name of the user logged on currently, for example

This variable is not to be confused with the program variable %(S.USERNAME) which can be set arbitrarily by command line or dialog.

%
(SYS.COMPUTERNAME)
%(SYS.STATIONNAME)

Content of the system variable COMPUTERNAME, that is the name of the computer as specified in the operating system.

%(SYS.DESKTOP)

Returns the content of the environment variable DESKTOP, that is the desktop folder of the user logged on currently. The returned path has 8.3-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\Desktop`

%(SYS.MYDOCUMENTS)

Returns the content of the environment variable DOCUMENTS, that is the documents folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\Documents`

%(SYS.TEMP)

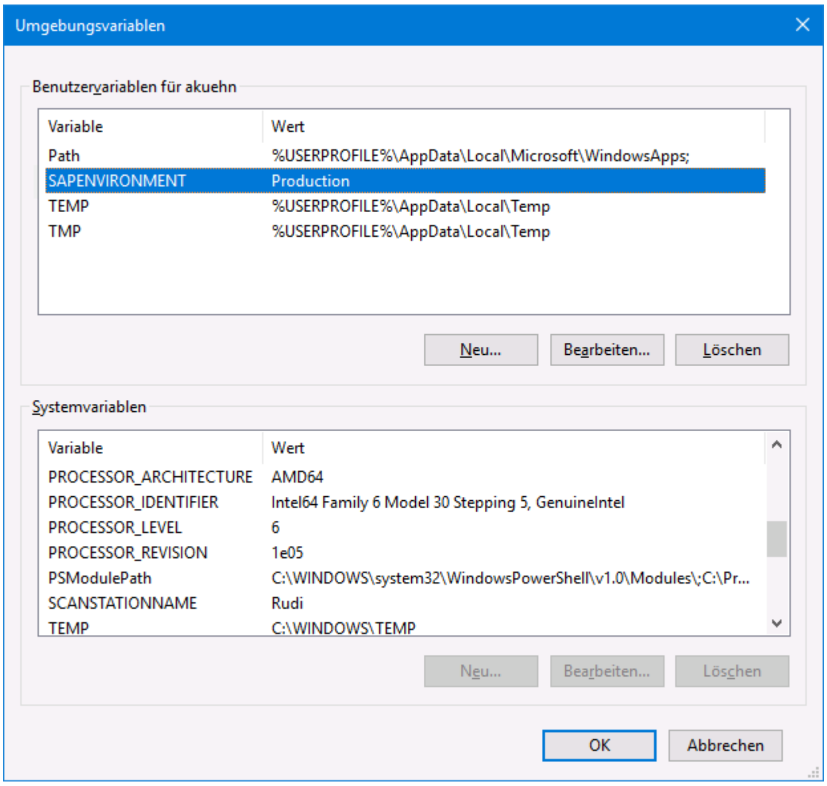
Returns the content of the environment variable TEMP, that is the temporary folder of the user logged on currently. The returned path has 8.8-convention and is **without** a trailing backslash, e.g.:

`C:\Users\EBERHA~1\AppData\Local\Temp`

%(SYS.ENV.*)

Returns the contents of any environment variable, e.g. the input request PROMPT by %(SYS.ENV.PROMPT).

User-defined environment variables can also be queried. In the example below, %(S.SYS.SAPENVIRONMENT) would return the value "Production"



Benutzervariablen für akuehn

Variable	Wert
Path	%USERPROFILE%\AppData\Local\Microsoft\WindowsApps;
SAPENVIRONMENT	Production
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

Systemvariablen

Variable	Wert
PROCESSOR_ARCHITECTURE	AMD64
PROCESSOR_IDENTIFIER	Intel64 Family 6 Model 30 Stepping 5, GenuineIntel
PROCESSOR_LEVEL	6
PROCESSOR_REVISION	1e05
PSModulePath	C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\;C:\Pr...
SCANSTATIONNAME	Rudi
TEMP	C:\WINDOWS\TEMP

4.5.1.21 Percent Codes of the Command Line

These variables can be specified in the command line as parameter.

`%(S.ACTUALTASK)`

%(S.TASKFILE)

%$d

The task, that is currently executed.

The task can be pre-selected automatically by the command line parameter as **TASKFILE**.or as **TODOFILE.**

Please do not confuse this variable with %(S.TASKNAME) which is just a commenting text.

`%(S.LANGUAGE)`

Returns a 2 character string for the language:

07 = German

09 = English

0A = Spanish

0C = French

0E = Hungarian

19 = Russian

1D = Swedish

It can be also set as a command line parameter named **LANGUAGE**.

This value can be used independently from the name of the computer.

Not all versions of the program support all languages. If a language is not supported, the dialog boxes of the program are displayed in the language of the operating system or in English.

%(S.STATIONNAME)
%$S

This function returns the given station name. It can be set in the field station in the job data setup. It can be also set as a command line parameter.

This value can be used independently from the name of the computer named **STATIONNAME**.

%(S.USERNAME)
%$U

This function returns the given username. It can be set in the field username in the job data setup. It can be also set as a command line parameter called **USERNAME**.

%(S.DOCCLASS)
%$c

Document name assigned to the batch of documents. It can be specified in a field of the job data dialog. Also it can be given by a command line parameter referred to as **DOCCLASS**.

%(S.TASKNAME)
%$D

Optional name of the document batch. It can be set in the field taskname in the job data setup.

It can be also set as a command line parameter called **TASKNAME**.

If it is not set, it is equal to the content of the variable %(S.TASKFILE).

%(S.CMDPARAM_)

Returns the contents of the command line parameter 1 to 9.
These values are also output in the statistics file in columns 21-29.

%(S.USERPARAM_)

Returns the contents of the command line parameter USERPARAM 1 to 9.
In contrast to CMDPARAM 1 to 9, these values are not passed to the statistics file.

4.5.2 Creating and selecting Variables

Automatic creation of %-Codes

Program variables are automatically created when they are entered on the left side of a value assignment. They are also created when they are stored in a stack that comes from another station. The same applies when a profile is imported or adjusted.

Manual creation and deletion of %-Codes

Wherever program variables are used, there is a small button with a percentage sign %. Clicking on it opens the dialog for creating and deleting %-Codes.

In the upper section on the left, the dialog shows a tree view of the various areas where there are percentage codes. To the right there is a table with these columns

%Code

Name of the program variables in decorated notation., That means with leading %(at the begin and trailing) at the end, for example **%(S.OUT.PATH)**

The single-character codes are not listed here, they can be found in column **Synonym**

Info

A short escription of the variable as additional information. When creating a new variable, see below, these information can be defined in the field Info.

Synonym

Here alternative and formerly used names of the variable can be found. Please note: For variable names with multiple characters the notation with %(and) is obligatory. For example **`%(S.REALPATH)`** oder **`%(REALPATH)`** would substituted with the full path to the stored image, where just REALPATH would just stay as the plain text REALPATH.

If there is a single-character Variable available it will be shown here too, e.g. **`%F.`**

Some single-character variables may have a dollar sign $ before or an index number 1-9 after the character. For example **`%$R4`**, points to the *fourth* barcode on the *previous* page.

Value

Shows the current value

Edit / create / delete a percent code

By a simple selection with the mouse in the list, the corresponding variable is copied into the lower area of the dialog. There are some control elements available. Above, there is an input field for the variable and a field for the description below.

✕

Removes the variable and its value from the list. If the variable is still somewhere in use, it will be added to the list once the corresponding task is executed. However, it has no value, or is an empty text.

✓

Adds a new variable to the list if - and only if - the given text in the field nearby begins with a percent sign followed by an opening round bracket and ends with closing round bracket **%(...)**

In addition a value can be given, for example **`%(A.SCANNER)=S4`**

If only the value is changed during editing, only this change is accepted. If the name of the variable is changed, a new variable is created.

Select / Format Percent Code

If the %-Code selection dialog is called to use it **in a definition**, the fields in the lower area are also switched on:

The upper input field allows you to **edit** the definition directly, in the lower field there is a **preview**.

⇨ Add

Inserts the selected program variable in the edit field at the cursor position. The preview field is also updated immediately with the current variable contents.

See also Overview and Formatting

4.5.3 Formatting Commands

Special commands can be used for the formatted output of variable contents. Please keep in mind that the contents of the variable remains **unchanged**, nevertheless. The syntax will only influence the output of the contents of the variable at the moment when it is used.

Common Syntax

A formatting command consists of

- An opening square bracket immediately behind the variable,
- the formatting pattern and
- a closing square bracket.

Example

`%(S.SCANCOUNT1)[04]` or `%R3[002004]`

Here the first scan counter will be formatted to four digits with leading zeros and from the third barcode the third to the fifth character will be cut out.

Concatenation

Formatters can be used in a row:

`%K[007016][06]`

The formatting will occur from left to right, here cutting out the page counter from the image header and formatting the found number to six digits with leading zeros.

Restrictions

- Nesting is not allowed, so don't use square brackets into other square brackets, not even in quotation marks!.
- Variables inside the formatter are not allowed, so don't use %-Codes within the square brackets.

In both cases the result will be an empty string.

Groups of Formatters

• **Formatting of Numbers**	Padding a numeric value to a given number of digits `47 --> 0000047`
• **Formatting of Character Strings**	Aligning texts `"Barcode" --> " Bar   "`

- Formatting of Enumerative Variables

  Listing the barcodes with a different separator

  `Text1,Text2 -> Text1;Text2`

- Assigning strings to numeric codes

  Assigning a text value to a list index

  `2 -> Rearside`

- Getting Parts of a Character String

  Using text from position 7 to 11
  `DATE071212A_BACK --> 1212A`

- Managing Substrings

  Removing unwanted characters

  `Ex pres sion --> Expression`

- Regular Expressions

  Finding a date in a OCR result

  `oice 20.07.2015 Your Sign --> 20.07.15`

- Functions

  Getting the length of a text

  `five --> 4`

  Converting text to numbers

  `4004 decimal --> FA4`

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.1 Formatting of Numbers

Numeric values and string values can also be shown formatted, with fixed length. Numeric values enable selection of the number of digits and shall put zeros in front of it, if the figure is below the foreseen number of digits. String values can be filled with spaces at the beginning or at the end.

If you do not use formatting, the length of the resulting number string will be different.

Please keep in mind that the contents of the variable remain **unchanged**, nevertheless.
The syntax will only influence the output of the contents of the variable at the moment when it is used.

Any formatting command is to place directly behind the variables name:

%..[**0cd**]: **Adding leading Zeros**
0 Choose the character zero
c (counter) is a numerical value indicating the number of digits. Up to three digit values are allowed. whereas the first must be different from 0.

d is a numerical value.
Numbers can only be integers in the range between -2147483648 and +2147483647. For better look it is possible to fill digits with leading zeros. The value remains unchanged.

Example:
The image counter has a value of 500.
%e[06d] will result: ‚000500‘

%..[cd]:

Adding leading Spaces

 First character after the opening bracket is a **space**.
c (counter) is a numerical value indicating the number of digits. Up to three digit values are allowed. whereas the first must be different from 0.
d is a numerical value.
Numbers can only be integers in the range between -2147483648 and +2147483647. For better look it is possible to fill digits with leading spaces. The value remains unchanged.

Please notice: The variables content will be treated as a numeric value even if there are leading zeros..

Example:
The variable contains the string ‚0029‘. The content will be treated as the numeric value 29.
%(MYVAR)[6d] will result: ‚   29‘

%..[xcd]:

Output a decimal value as a hexadecimal value

x The output is hexadecimal, alpha characters are lower case.
c (counter) is a numerical value indicating the number of digits. Up to three digit values are allowed. whereas the first must be different from 0.
d is a numerical value.
Numbers can only be integers in the range between -2147483648 and +2147483647. For better look it is possible to fill digits with leading zeros. The value remains unchanged.

Example:
The image counter has a value of 500.
%e[x10d] will result: ‚00000001f4‘

%..[Xcd]:

Output a decimal value as a hexadecimal value

X The output is hexadecimal, alpha characters are upper case.
c (counter) is a numerical value indicating the number of digits. Up to three digit values are allowed. whereas the first must be different from 0.
d is a numerical value.
Numbers can only be integers in the range between -2147483648 and +2147483647. For better look it is possible to fill digits with leading zeros. The value remains unchanged.

Example:
The image counter has a value of 500. %e[X10d]
will result: ‚00000001F4‘

%..[#cd]:

Output a decimal value as a hexadecimal value
# The output is hexadecimal with the leading characters "0X".
c (counter) is a numerical value indicating the number of digits. Up to three digit values are allowed. whereas the first must be different from 0.
d is a numerical value.
Numbers can only be integers in the range between -2147483648 and +2147483647. For better look it is possible to fill digits with leading zeros. The value remains unchanged.

Example:
The image counter has a value of 500. %e[#10d]
will result: ‚0X000001F4‘

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.2 Formatting of Texts

String variables can also be formatted.

Please note that the contents of the variables remain **unchanged**. Only the output of the contents of the corresponding variables at the time of the formatting is influenced by the syntax.

%..[<cs]:

Add blanks at the beginning of a string until the length is c characters
< means blanks are filled at the beginning of the string.
c is a numerical value indicating the number of digits. Up to three digit values are allowed.
s means the variable must be string value filled with blanks. It does not work with numeric values.

Example:
The batch name is ”Risotto”
%O[<10s] will result: "   Risotto"

%..[>cs]::

Add blanks at the end of a string until the length is c characters
> means blanks are filled at the end of the string.
c is a numerical value indicating the number of digits. Up to three digit values are allowed.
s means the variable must be string value filled with blanks. It does not work with numeric values.

Example:
The batch name is ”Risotto”
%O[>10s] will result: "Risotto   "

%..[<cz]:

Add zeros at the beginning of a string until the length is c characters
< means zeros are filled at the beginning of the string.
c is a numerical value indicating the number of digits. Up to three digit values are allowed.

z means the variable must be string value filled with zeros. It does not work with numeric values.

Example:
The batch name is ”Risotto”
%O[<10s] will result: "000Risotto"

%..[>cz]:

Add zeros at the end of a string until the length is c characters
> means zeros are filled at the end of the string.
c is a numerical value indicating the number of digits. Up to three digit values are allowed.
z means the variable must be string value filled with zeros. It does not work with numeric values.

Example:
The batch name is ”Risotto”
%O[>10s] will result: "Risotto000"

%..[<c|]:

Output of the first c characters of a string
< The formatting refers to the beginning of the string
c (counter) is a numerical value to define the number of characters. Up to three digit values are allowed.
| The string is cut after the number of characters, defined by the c value.

Example:
The batch name is ”Risotto ”
%O[<2|] will result: "Ri"

%..[>c|]:

Output of the last c characters of a string
> The formatting refers to the end of the string
c (counter) is a numerical value to define the number of characters. Up to three digit values are allowed.
| The string is cut after the number of characters, defined by the c value.

Example:
The batch name is ”Risotto ”
%O[>2|] will result: "to"

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.3 Formatting of Enumerative Variables

In the program, several data can be listed in one variable.

For example, the variable `%(S.BARCODE)` (or synonym %R) contains all found barcodes according to the settings made for the search.
The simple use of the variable name leads to an output of all barcodes, which are then separated by a comma.

If a different separator is to be used in the collection, this can be achieved with the following syntax:

%(S.BARCODE)
%R

If the variable name is used without an additional string for formatting, a ',' (comma) is automatically selected as a separator.

%(S.BARCODE)[+]
%R[+]

Adding the **[+]** string causes all found barcodes to be listed, but separated by a '+' (plus sign). All other reasonable characters can be used here also. To get a tabulator please copy and paste it from a text editor, like NotePad. The usual notation with backslash-t doesn't work here.

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.4 Naming list elements

The numerical content of a variable can be used as index for a list of texts:

%(p.xxx)
[Text0,Text1,Text2,...]

The value of the variable will be replaced by a text from a list.
The list is enclosed in square brackets.
The values are separated by commas.
The index of the list starts at **Zero, a** value of **0** will return the **Text0.**
The returned text includes **all characters** in between two separators **including** the spaces

Example:
The variable **%(I.COLOR)** has the value 2
The format **%(I.COLOR)[Red   ,Green,Blue ,Black]**

will return the word „**Blue** “.
Using a value beyond the defined indexes will return an empty string (but no error). In the sample above a 4 will return an empty string.

%(S.IMAGE.SIDE)
[Text1,Text2]
%s[Text1,Text2]

The formatting of the variable **%(S.IMAGE.SIDE)** works similar to other numerical variables but has a different start index

The index of the list starts at **One, a** value of **1** will return the **Text1.**

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.5 Getting parts of a Character String

It is possible to get substrings of a percent code. The definition has the following form:

%..[nnnmmm]:

nnn stands for a 3-digit number defining the begin of the text to cut out.
The counting starts at **Zero**
mmm stands for a 3-digit number defining the end of the text to cut out.

Examples:

Assume the name of a the loaded file is INV29072017.JPG consisting of a prefix and a date in ddmmyyyy form.

Then the following assignment will create a date in yymmdd order

%(I.MyDate)=%(S.IN.NAME)[009010]%(S.IN.NAME)[005006]%(S.IN.NAME)[003004]
So the result will be 170729

Getting images from a XINO scanner, the image header will hold the result of the Color-Marker detection.
%K[388388] returns the 388th character in the image header holding the Indicator for the found Color-Marker

The following condition tests the length of a barcode (at least ten but not 11 characters)
%R1[009009]!="" && %R1[010010]==""

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.6 Trimming Substrings

There are specific commands for adding, suppressing or keeping parts of a string. Same as with formatting commands, such commands are written between square brackets and are placed directly behind the name of the variable.

The syntax following below it is possible, at the moment when the variable is used, to add characters to the relative contents, or to suppress specific strings.

Please keep in mind that the contents of the variable remain **unchanged**, nevertheless. The syntax will only influence the output of the contents of the variable at the moment when it is used.

Add substrings

A substring can be added to a percent code. The plus sign signals to a string to add, an optional angle bracket signals where to do this. There are the following possibilities:

%[+"Text"]

(Plus) adds a string before and after the value of the variable.

'+' is for adding a substring.

Text is the substring to add. It is to enclose by quotation marks.

Example:

```
%(A.Code39)       =  "031120"
%(A.Code39)[+"*"] -> "*031120*"
```

%[+>"Text"]

(Plus,Greater) adds a string after the value of the variable.

Example:

```
%(A.Env)             =  "Job_0234"
%(A.Env)[+>"_PROD"] -> "Job_0234_PROD"
```

%[+<"Text"]

(Plus,Smaller) adds a string before the value of the variable.

Example:

```
%(A.Env)               =  "Job_0234"
%(A.Env)[+<"TEST_"] -> "TEST_Job_0234"
```

Suppress substrings

A substring can be removed from a character string. The tilde sign signals to remove all occurrences, the minus sign only one. A following angle bracket indicates where to do this:

%[~"Text"]

(Tilde) suppress all occurrences of the given character string (not the single characters)

Text is the character string to search for and remove. The argument must be quoted.

Example:

```
%(A.TEST)         =  " Sp a c es "
%(A.TEST)[~" "] -> "Spaces"
%(A.TEST)[-" "] -> "Sp a c es"
```

%[-"Text"]

(Minus) Suppresses a substring at the beginning and/or at the end of the content of the variable.

Text is the substring to suppress. It is to enclose by quotation marks.

"**Text**" will be suppressed at the begining of the content of the variable as often as it can be found.

"**Text**" will be suppressed at the end of the content of the variable as often as it can be found.

Example:

```
%(A.Code39)        =  "*041320*"
%(A.Code39)[-"*"] -> "041320"
```

%[->"Text"]

(Minus,Greater) Suppresses a substring at the end of the content of the variable.

Text is the substring to suppress. It is to enclose by quotation marks.

"**Text**" will be suppressed at the end of the content of the variable as often as it can be found.

Example:

```
%(A.Price)        =  "13.59$"
%(A.Price)[->"$"] -> "13.59"
%(A.SMS)          =  "xoxoCUxoxo"
%(A.SMS)[->"xo"]  -> "xoxoCU"
```

%[-<"Text"]

(Minus,Smaller) Suppresses a substring at the beginning of the content of the variable.

Text is the substring to suppress. It is to enclose by quotation marks.

"**Text**" will be suppressed at the begining of the content of the variable as often as it can be found.

Example:

```
%(A.No)          =  "++++++22"
%(A.No)[-<"+"]   -> "22"
%(A.SMS)         =  "xoxoCUxoxo"
%(A.SMS)[-<"xo"] -> "CUxoxo"
```

Keep a substring, suppress the rest of the string

A substring can be kept while the rest of a string is removed. The angle bracket indicates the direction in which the characters should be kept. A trailing equal sign indicates that the search text should be retained during the deletion:

%[<"Text"]

(Greater) Returns all characters before "**Text**".

Text is the substring to be kept.

All characters from "**Text**" to the end of the character string will be suppressed.

If the variable does not contain the string "**Text**" the full text of the variable will be returned.

If the "**Text**" occurs more than once, only the first one is treated.

Example:

```
%(A.Test)         =  zzzABZyyyCDZxxx
%(A.Test)[<"CD"]  -> zzzABZyyy
%(A.Test)[<"24"]  -> zzzABZyyyCDZxxx
%(A.Test)[<"Z"]   -> zzzAB
```

% [<=" Text "]

(Greater,Equal) Returns all characters up to and including "**Text**"

Text is the substring to be kept.

All characters beyond "**Text**" will be suppressed.

If the variable does not contain the string "**Text**" the full text of the variable will be returned.

If the "**Text**" occurs more than once, only the first one is treated.

Example:

```
%(A.Test)          =  zzzABZyyyCDZxxx
%(A.Test)[<="CD"]  -> zzzABZyyyCD
%(A.Test)[<="03"]  -> zzzABZyyyCDZxxx
%(A.Test)[<="Z"]   -> zzzABZ
```

% [>" Text "]

(Smaller) All characters beyond "**Text**" will be returned.

Text is the substring to be kept.

Alle character before and including "**Text**" will be suppressed.

If the variable does not contain the string "**Text**" the full text of the variable will be returned.

If the "**Text**" occurs more than once, only the first one is treated.

Example:

```
%(A.Test)         =  zzzABZyyyCDZxxx
%(A.Test)[>"AB"]  -> ZyyyCDZxxx
%(A.Test)[>"30"]  -> zzzABZyyyCDZxxx
%(A.Test)[>"Z"]   -> yyyCDZxxx
```

% [>=" Text "]

(Smaller,Equal) All characters from "**Text**" to the end of the string will be returned

Text is the substring to be kept.

All characters before "**Text**" will be suppressed.

If the variable does not contain the string "**Text**" the full text of the variable will be returned.

If the "**Text**" occurs more than once, only the first one is treated.

Example:

```
%(A.Test)          =  xxxABZyyyCDZzzz
%(A.Test)[>="AB"]  -> ABZyyyCDZzzz
%(A.Test)[>="11"]  -> xxxABZyyyCDZzzz
```

```
%(A.Test)[>="Z"]   -> ZyyyCDZzzz
```

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.7 Regular Expressions

With the help of so-called regular expressions (short "RegExp"), strings can be searched for and, if necessary, replaced.

A detailed description is available in the Reference Manual.

Regular expressions

`%..[/rx]`

Returns the part of the string that matches the search pattern:

The first OCR-Result may hold the text fragment "cedate 20.07.2015 Deliv'

`%(S.OCR1)[/\d{2}\.\d{2}\.\d{4}]` will find **20.07.2015**

`%...[/rx/rp/]`

Replaces the found text with the replacement text:

The userdefined variable `%(I.Times)` holds the word "thrice"

`%(I.Times)[/thrice/triple/]` returns triple

`%...[/rx//]`

Deletes the found text

The name of the XML-File has the value "20.07.2015 14:35" which is not a valid file name

`%(J.XMLFile)[/:\.//]` removes the colons and returns: **20072015 1435**

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

4.5.3.7.1 Search Pattern

Search Pattern

A search pattern is made of ordinary characters (letters a-Z 0-1) and meta-characters. Ordinary characters must match exactly, meta-characters replace classes of characters and can give counts.

Ordinary Characters

Ordinary Characters are all characters except `.*?+(){}[]^$\`, which are meta-characters.

Should a meta-character be treated as an ordinary character, that character must be escaped by a leading backslash \

Searching for a path C:\Temp\File.txt needs escaping because there are 3 meta-characters in it:

```
C:\\Temp\\file\.txt
```

To avoid having to escape long sequences of text, the special switches \Q \E may enclose a text area without meta-character interpretation:

```
\QC:\Temp\Datei.txt\E
```

Meta-character

Meta-characters represent classes of text.

Replacements for a set of characters

.	The character "**.**" matches any single character but newline (`\n`). If flag s (SINGLELINE) is given, "." also matches newline, see 10.2.3 Flags
\w	A backslash followed by the lower case letter "w" represents any character a-z, A-Z and the underline character _.
\W	A backslash followed by the upper case letter "W" represents any character not being in the set a-z, A-Z and the underline character _..
\s	A backslash followed by the lower case letter "s" represents any white space character, such as space, tabulator, newline, carriage return.
\S	A backslash followed by the upper case letter "S" non-white space character.
\d	Represents a digit (0-9)
\D	Represents a non-digit, anything else but 0-9.

There are also some POSIX expressions available that define a class:

[:alnum:]	Represents a character a-zA-Z or digit 0-9
[:alpha:]	Represents a letter a-zA-Z
[:upper:]	Represents an upper case character.
[:lower:]	Represents a lower case character.
[:digit:]	Represents a digit 0-9, same as \d
[:blank:]	Represents a white space character, same as \s
[:punct:]	Represents a punctuation character ,.!?”’ or parentheses, brackets etc.
[:space:]	Represents a white space characters, same as \s
[:print:]	Represents a printable character, ANSI 0x20 to 0x7E.
[:graph:]	Represents a printable character without whitespace characters **[:blank:]**
[:ascii:]	Represents a character of the ASCII character set
[:cntrl:]	Represents a control character from the ASCII character set 0x00 to 0x1F and 0x7F.
[:xdigit:]	Represents a hexadecimal digit, such as 0-9, A-F.

By the bracket notation a set of characters can be defined.

[CDEFGAH]	Any of the characters given in the brackets will match. Meta-characters are allowed, so [AB\d] will match the upper case letters A and B and any digit character 0-9
[AC-H]	A range can be given. The given expression matches the character A and one of the characters C-H-
[A-Za-zÄÖÜäöüß]	The given expression matches a German letters.
[^AC-H]	^ denotes a negation. The expression matches any character by ACDEFGH

Quantifier

A quantifier denotes how many occurrences are required of a given expression.

*	The expression may occur 0 or more times.
+	The expression must occur 1 or more times.
?	The expression may or may not occur.

{n} n is a number that requires the expression to occur exactly the given count times.

{n,} n is a number that requires the expression to occur at least the given count times.

{n, m } n is a number that requires the expression to occur at least the given count times, and m is the number that requires the expression at most the given count times.

Quantifiers work greedy. * takes the maximum numbers of characters that match the expression.
`\d{3,6}` does not stop at 3 digits, but takes 6 digits if there are more than 5 digits.

Line and word boundaries

The search pattern uses operators to locate word boundaries and line boundaries.
There is start of line and end of line available.

^ Denotes the beginning of the line or the beginning of the search string.

If flag m **`MULTILINE`** is given, the search target may consists of several lines separated by newline characters. In such case, ^ denotes the beginning of a line. Otherwise ^ is the beginning of the complete search target

$ Denotes the very end of the search target.

\b Denotes the word boundary.

Please also refer to the manal for more Details.

See also Numbers, Enumerative Variables, Character Strings, Getting Parts, Managing Substrings, Regular Expressions and Formatting Overview

4.5.3.7.2 Replacement

If the format **`%..[/rx/rp/]`** is used, **rp** represents the so-called replacement expression. Replacement expressions can contain plain text and access to certain found subexpressions.

$n Returns the nth group, **`$0`** contains the entire expression found..

The search pattern **`(\d{1,2})\/(\d{1,2})\/(\d{1,4})`** finds an American format date, e.g. in "ABC 07/20/2018 XYZ"

In the replacement text, the found result can be re-sorted **`$2.$1.$3`** that then returns "ABC 20.07.2018 XYZ".

If you only want the date, then the test before and behind it must also be searched as a parenthesized expression:
`(.*\b)(\d{1,2})\/(\d{1,2})\b\/(\d{1,4})(\b.*)`

the expression **`$2.$1.$3`** will only return "20.07.2018"

${Name} You can also use names. They correspond to the assignments in the respective group:
`(.*\b)(?<Month>\d{1,2})\/(?<Day>\d{1,2})\b\/(?<Year>\d{1,4})(\b.*)`

The resorted result **`${year}${month}${day}`** then returns "20180720"

$+ Stands for the group with the highest number.

$_ A dollar sign followed by an underscore represents the entire original text.

$` A dollar sign followed by Accent Grave (0x60, 96dec, on the QWERTY keyboard above the TAB key) represents the text **before the first** phrase found.

$' A dollar sign followed by a quotation mark represents the text **behind the last** expression found.

$$ Is in substitution for a single $.

Here, therefore, there is *no masking with the backslash*

4.5.3.8 Functions

The content of a variable can be examined or converted to another type by using special formatting templates. These function are new and will not work for Version 5 or earlier.

Examination functions, Version 6.03 or newer

%...**[LEN]** Returns the length of a text.

Set %(V.Test1)=Fourtyfour

`%(V.Test1)[LEN]` returns **`10`**

If the text is an empty string a zero will be returned.

Conversion functions, Version 6.03 or newer

%...**[NUM]** Converts the numerical part at the begin of a string into a decimal number.

Set %(V.Test1)=32
Set %(V.Test2)="15 milliseconds"
Set %(V.Test2)="Type 33"

%(V.Test1)[NUM] returns **32**
%(V.Test2)[NUM] returns **15**
%(V.Test3)[NUM] returns **0**

If the text doesn't start with a number a **zero** is returned, not an empty string.

%...**[HEX]**

Converts the numerical part at begin of a string into a hexadecimal number

Set %(V.Test1)=32
Set %(V.Test2)="15 milliseconds"
Set %(V.Test2)="Type 33"

%(V.Test1)[HEX] returns **20**
%(V.Test2)[HEX] returns **F**
%(V.Test3)[HEX] returns **0**

The result is without any decorations, so there are no leading "0x", or trailing "h"

If the text doesn't start with a number a **zero** is returned, not an empty string.

%...**[XML]**

Replaces some XML control characters from a string with their &-notation

Set %(V.Test1)="We found an "&" in the <text> !"

%(V.Test2)[XML] returns **We&032;found&032;an&032;&quot;&amp;&quot**

Conversion functions, Version 6.11 or newer

%...**[PATH]**

Replaces characters that are not allowed in filenames with underscores. This conversion is appropriate when e.g. OCR results should serve as a file name.

Set **%(V.FN)=C:\D<>cs\|mage035.jpg**

%(V.FN)[PATH] returns **C:\D__cs_mage035.jpg**

It is not mandatory to specify the function when defining file or directory name. There the conversion happens automatically. Only if the name is e.g. in the batch file, this function ensures that the "correct" name is used.

%...**[UPR]**

Returns the text in uppercase letters.

Set **%(V.Txt)=Local food as well as Crème fraîche and Heiße Würstchen.**

%(V.Txt)[UPR]returns: **LOCAL FOOD AS WELL AS CRÈME FRAÎCHE AND HEIßE WÜRSTCHEN.**

%...[**LWR**]

Returns the text converted to lowercase letters.

Set **`%(V.Txt)=The Catcher In The Rye`**

`%(V.Txt)[UPR]`returns: t**`he catcher in the rye.`**

%...[**PRP**]

Returns the text in lowercase, each word starting with a capital letter

Set **`%(V.Txt)=TERMS AND CONDITIONS (EU-VERSION).`**

`%(V.Txt)[RRP]`returns: **`Terms And Conditions (Eu-Version).`**

See also Numbers, Strings, Enumerators, Lists, Segmentation, Substrings, Regular Expressions and Functions.

5 Safety Instructions

To ensure that the program runs smoothly, the operator must have access to the resources used: If scanners and printers are used, they must be accessible and ready. This is especially true for network devices.

With some scanners, loading the scanner driver will automatically wake up the devices from standby mode. In all other cases, the devices should be switched on manually before starting the program.

The directory for temporary storage, usually C:\OPEN_JOB\, and the target directory must be accessible to the logged on operator.
If it is allowed and possible, the intermediate directory should be excluded from the virus scanner, otherwise the read/write operation will be slowed down.

If the scan profile used accesses templates, e.g. for stack sheets, these must also be accessible.

Under no circumstances should the intermediate directory be used as a storage location by other programs. Choosing the intermediate directory as the target directory should also be avoided.

The program directory is generally not read-only. However, it should not be used by other programs. The output of additional log or statistics files should not take place here either.

Please also observe the regulations for operating the hardware!

6 Error Handling

We try hard to provide you with a good product. Nevertheless, errors can occur in the program sequence.

In the event of an error, the program can usually display a message that indicates the cause of the error, e.g. that a target directory in the network cannot be reached or that the cover sheet cannot be printed because the template file cannot be found. Such errors can be easily rectified by ensuring that these directories and files are accessible to the logged on operator. The same applies when reading in from the directory or when importing batches.

When using the scanner, the frequent errors such as "jam", "double sheet" or "housing not closed" are also only displayed and can be rectified by the operator.

If there are fundamental errors when operating the scanner, most of these will be displayed with an error number in the 200 or 300 range. If a cold start of the scanner and computer does not solve the problem, technical support should be contacted.

However, there are also errors that the program does not notice, e.g. if a document is not separated because the separating feature, e.g. barcode, is not recognized. In such a case, please check the settings for the search with the image on which the search is carried out. A changed image quality is also often the cause of impairment of the recognition performance
The first step here is always to display the value found in the info window or in the status line.

In the case of very complex profiles, it can be helpful to log the task steps in the program's log file or to display the event rules in the warning window.

If the cause of an error cannot be found, our support is available to you within the framework of a maintenance contract. Please note, however, that we only help with troubleshooting and suggest suitable measures. The complete analysis of an existing profile or the creation of a new profile always represent a fee-based effort.

If this is possible and allowed, you should use our remote maintenance program TeamViewer. You can find this in our download area under Diagnostic Tools.

If access is not possible, we always need the current configuration in the form of a DAT file. You can find this in the main menu profile, Save as...

In addition to the profile, this file also contains information on the installed versions and the current log files of the program.

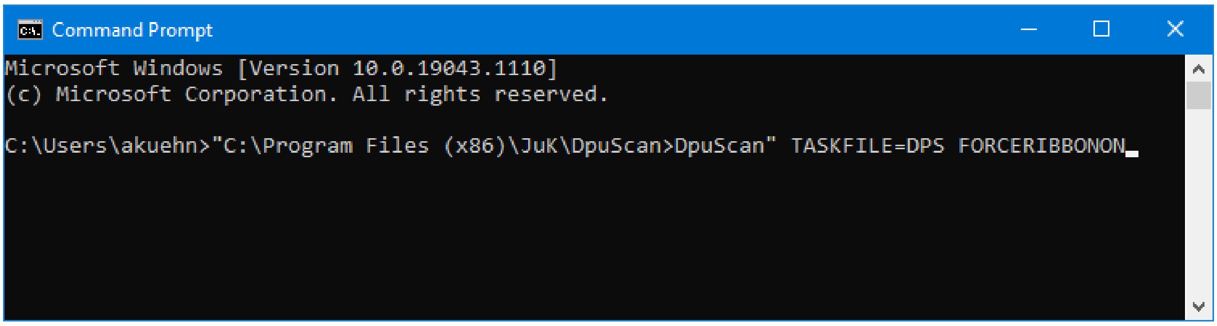
In the case of recognition problems, we also need the image on which the recognition is to take place. A screenshot or a mobile phone photo will not help here. Therefore, please scan the paper with the current configuration and save it from the display window by right-clicking -> "Save as ..."

In addition to the DAT file, a description of the target and error status is essential, as well as a description how to reproduce the error.

A description of how to create a test batch or, ideally, a test batch scanned with the program, will simplify the support's work and shorten response times.

If an unexpected error (crash) occurs, try to start the program again. If the error does not appear at the start directly, please immediately save the profile as described above.

If the program does not start, there is probably an error in one of the loaded modules or scripts. Try to start the program with the standard profile DPS from the command prompt:



```
Microsoft Windows [Version 10.0.19043.1110]
(c) Microsoft Corporation. All rights reserved.

C:\Users\akuehn>"C:\Program Files (x86)\JuK\DpuScan>DpuScan" TASKFILE=DPS FORCERIBBONON
```

Start of the Default Profile

The additional parameter FORCERIBBONON forces the display of the menu bar (again).

If it is not possible to start the program at all, please use the DpuEnterpriseManager to save all profiles in the form of a DAT file; the last log file of the program is also saved.

Then start the program DpuStat32.EXE in the DpuScan folder. It determines the installed versions of the required program modules and during this time shows a progress dialog in the form of a moving magnifying glass. At the end it offers to save the result of the analysis or to send it directly by email.

7 Maintenance and Update

Claim

As part of a maintenance contract, we allow you to use the current version of your software in the form of an update. This

Updates require the installation of a licensed version and an ongoing maintenance contract for this license.

Updates are offered up to 2 years after the life cycle of a program version has expired.

You need a maintenance contract to upgrade to a higher version. The installation program detects computers (dongles) that are authorized to upgrade.

Sources

You can find the current updates in the download area on our website www.DpuScan.com or www.Xino.de. Please contact us if you need a special version of the software.

Workflow

Please close and finalize all open batches first. Then create a backup of all profiles and sub-profiles used. Keep this backup in case you want to revert to the old version later.

Uninstall the old version of the scanning program. It is not necessary to uninstall the other DPU components.

Install the new version. It is not necessary to read in the previously saved profiles. Migration is carried out automatically when the profiles are loaded.

In the case of an update, the configuration of external modules, e.g. the OCR, may have to be adapted because there may have been changes in the newer version.

Basically, all profiles must be carefully checked after an update!

After each installation you should back up the profiles and keep them.

DPU-S

Regardless of the version of the scan program, you should use the most up-to-date version of the "DPU-S" driver sets. You can find the updates for this in our download area. The update is free of charge and can be used without restrictions.

8 Frequently Asked Questions

⊟ What are the differences between the yellow and the cyan colored navigation buttons?

If the settings for the visible elements of the thumbnails and the tree view at start are reduced to the current folder or current file, see **visible elements at task start**, the navigation buttons are acting in a different way. At the end of an area the yellow navigation buttons will be disabled or select the first image of the current area. The cyan navigation buttons will change the selected image group.

⊟ The scanner is working but I cannot see the scanned images!

That can have the following reasons:

- Please make sure that a file format for the output is defined.
  To set up the file format see **Profile - Base Profile - tab sheet General - Imageformats - Outputformats**
- Please make sure that the desired color format is set up to be displayed.
  See **Profile - Base Profile - tab sheet General - Screen Layout - tab sheets Process and Interactive - check boxes Bitonal/Gray/Color Images**

⊟ The application can be closed during a scan pause even if an open batch is still loaded. Can I avoid that?

Set a condition to prevent the execution of the command **Application Exit**.
Open **Profile - Application Layout Profile - tab sheet Actions** and choose the line **Application Exit**.
Edit the cell in the ***column Condition***. Insert: **%(S.TASKSTATUS)="READY"**.
Now, the application can only be closed if there is no batch or configuration in process.

⊟ The program has a corrupt configuration and does not start with this last used profile. How to start the program with another profile?

Try to use the command line to start the program using the command line parameter ***taskfile=***, for example **dpuscan.exe taskfile=default.**

⊟ Within a multi stream batch I am going to change an index value for a color image using a UDD. The event rules seem not to use the changed value.

Make sure that the value is stored to all images of the image group by using the command **SetVariableImageGroup.**

In the case of using image variables with a multi stream batch every image has its own value.
For example: For each image the variable %(I.MYVAR) has been preset with a "0" while scanning.
If a value for one color image is changed and the value for the bitonal image will not be changed automatically.

Because event rule checking always uses the variables of the last image of the group, the changed value of the color image will not be checked by an event rule. To ensure that those changes will be noticed by the event rule check, they have to be stored to all images of the group. Please use the macro command **SetVariableImageGroup**.

⊟ Why do I get the message: "Can't create directory because of a missing barcodes / patchcode or ocr result" at scan start?

Make sure that the first image of the batch fits the rule for creating the new directory. Please note that it has to be the first image which will not be deleted.

..

9 Content

About this Help

Index

DpuScan

Reference Manual DpuScan

Wuppertal, Germany, 2024