

Mergefields | Placeholders

The following mergefields are supported:

```

"%0": PatID
    Patient ID
    e.g.: 08/15
"%1": PatID
    Patient ID modified for use as file name
    e.g.: 08~615    (invalid character "/" replaced by "~6")
    The following invalid characters are replaced as described:
        "\" -> "~0"
        "*" -> "~1"
        "?" -> "~2"
        "@" -> "~3"
        ":" -> "~4"
        "\"" -> "~5"
        "/" -> "~6"
        "<" -> "~7"
        ">" -> "~8"
        "|" -> "~9"
"%2": Name
"%3": First Name
"%4": Birthday
    Format: JJJJMMTT
    e.g.: 19960313  for 13.3.1996
"%5": Sex
    possible values:
    "X":  unknown
    "M":  Male
    "F":  Female
    "MC": Male Converted
    "FC": Female Converted
"%8": Name w/o empty spaces
"%9": First name w/o empty spaces
"%A": Cast number
"%B": Birthday - Mediadent required format: TTMMJJJJ
    e.g.: 13031996 for 13.3.1996
"%C": Additional patient ID
"%D": Social insurance number
"%F": Number of the current client
    Can be extended to the format 0015 with <FL|%F|4|0>
"%M" or "%{pat:notes}": Comment on the patient (!! Can be multiline !!)
    With <VAL|%M|TagName> A value can be read from the comment
    This value must be in the form "TagName=Value" in the comment
"%{pat:attributes}": Attributes separated with semicolon ;
"%{pat:caseid}": Case ID of the patient.
    If several case IDs are known, a selection dialog is displayed.
"%{pat:caseid_1}": First case ID of the patient in the case ID list
  
```

Data of the patient's treating doctor

The data of the primary doctor is returned, which is the first entry in the "Treating doctor" list.

```
"%{treat:short}": Short name of the doctor
"%{treat:firstname}": First name of the doctor
"%{treat:lastname}": Last name of the doc
```

Placeholder for system data

```
"%U": Username: Will be replaced with the short name of the current OnyxCeph login
"%W": Workstation: Replaced with the current Windows login: Username [Computer
name]
"%X" or "%{time:ansi}": Timestamp: current date as well as time (server) in format:
JJJJMMTTHHMMSS (yyyymmddhhnnss)
    e.g.: The current year can be output with <SS|%X|1|4>
"%%": Percent sign "%" (in case the percent sign is needed in the text)
```

Placeholder when an image is selected

```
"%6": Image file name starting from the image directory on the server:
    z.B.: 08~615\08~615SBIX.jpg
"%7": Image Code
    z.B.: 08~615SBIX
"%E": Image date in format JJJJMMTT
"%I": Image type
"%T" or "%{img:notes}": Comment on the image (!! Can be multiline !!)
    With <VAL|%M|TagName> a value can be read from the comment
    This value must be in the form "TagName=Value" in the comment
"%{img:attributes}": Attributes of the image separated with semicolon ;
"%G" or "%{img:seriesname}": Name of the session to which the image belongs
    e.g.: Final-01.11.2016
"%{img:seriesdate}": Date of the session to which the image belongs in the format
YYYYMMDD
"%{img:seriesnotes}": Comment on the session to which the picture belongs (!! Can
be multiline !!)
```

Recursive search with * in image acquisition

If a "*" character is appended to the path, the subdirectories of the path are also searched.
Optionally, the depth of the search can be specified after the "*" character and the "<" marker.

Examples:

Search subdirectories in the patient directory:

```
"D:\Images\%1\*"
```

For the patient 123456 to become all images from the directory

```
"D:\Images\123456\"
```

and all subdirectories are returned.

Possible results:

```
"D:\Images\123456\Img1.jpg"
```

```
"D:\Images\123456\2010\Img2.jpg"
```

```
"D:\Images\123456\2011\Retension\Img3.jpg"
```

```
...
```

With the path:

```
"D:\Images\%1\*<1"
```

Only the following file will be found:

```
"D:\Images\123456\Img1.jpg"
"D:\Images\123456\2010\Img2.jpg"
This file is not found:
"D:\Images\123456\2011\Retension\Img3.jpg"
Also allowed: Search all images on the D: drive:
"D:\*"
Please avoid, may take a very long time.
```

Wildcards

Wildcards "*" and "?" are also possible in all directory names.
Wildcards are used according to the normal Windows conventions:

```
? stands for a character
* stands for any number of characters
```

OnyxCeph³ searches for all matching paths and returns the files they contain.

For the patient Müller,Jan, for example, at the import path:

```
D:\Scans\Day????????\%8,%9\*<1
```

searches the following directories (if available):

```
D:\Scans\Day20160301\Müller,Jan\
D:\Scans\Day20160301\Müller,Jan\Head\
D:\Scans\Day20170313\Müller,Jan\
D:\Scans\Day20170313\Müller,Jan\Teeth\
```

Module-specific placeholders

```
%{module:id} also %V
Code of the triggering module
```

```
%{module:name}
Name of the triggering module
```

```
%E
Image date in all modules derived from TfrmTreat3DBase.
Image date in ThirdPartyToolStart (if started from image)
```

```
%J
Jaw in export 3D, if only objects of the upper or lower are exported
```

```
%N
Filename container in EventContainerImportDone
Filename container in EventContainerExportDone
File name container in EventContainerUploadDone
File name Report in EventReportUploadDone
File name of the report EXE in EventReportExportDone
Filename in EventToothMovementExportDone
Filename of the 3D file in Event3DExportDone
WebAddress in EventWebExportDone
```

File names separated with spaces in Event3DSeriesExportDone

When exporting multiple files, the wildchar * stands for the step number

%O

FTPCode in EventContainerUploadDone

FormName in SaveCarbonCopy and in PDF-Export

ProjectName in EventProjectSaved

%P

Passwort in EventReportUploadDone

%Q

Download- Address in EventReportUploadDone

ProjectCode in EventProjectSaved

%S

Step-Number in Export3D ot Aligner

Number of files in Event3DSeriesExportDone

"New" for new or "Upd" for updated records in EventEvaluationDone

%V

ID of the module in EventModulOpened und EventModulClosed as well as in PDF-export (also %{module:id})

Name of the anaysis in EventEvaluationDone

%Z

Placeholders for image data type (2D|3D) for events EventEvaluationDone, EventImageImportDone

Macros for special interfaces

Only for image acquisition the following macros are supported:

"<FDK_D>": The name of the data path according to KF0 Office conventions with date of birth is used.

"<FDK_I>": The name of the data path according to the KF0 Office conventions with patient ID is used.

Example:

C:\PatientData\ is the path configured in KF0 Office for patient images.

In Onyx, the path C:\PatientData\<FDK_I> would then have to be configured for the third-party programs.

(Assuming FDK uses the patient number, if birthday instead of <FDK_I> then use the macro <FDK_D>).

"<TNTand>": SurnameFirstName is inserted without spaces (for TNTand).

Example: Hans Peter Müller-Herbig becomes "Müller-HerbigHansPeter".

Macros for string manipulation

The following macros start with "<" and end with ">". They use multiple parameters separated by "|". Syntax::

<C|T|P0|P1|P2|...>

C: Macro command e.g. "SS" for SubString

T: Text, may contain placeholders like %1, the placeholders are always replaced first and

only then the macro is executed.

P0, P1: Parameters, depending on command different number and usage

In general, macros may be nested. That means T, P0, P1, ... can contain macros. The macros are resolved from the inside to the outside.

SubString (SS)

"<SS|T|Pos|Count>"

A substring of length Count is returned from the text T, starting from Pos.

If Pos + Count exceed the length of the string, the string is returned starting from Pos.

Example:

"This is a <SS|new MacroXX|5|5> in text"

turns into:

"This is a Macro in text"

Replace String (RS)

"<RS|T|SearchString|ReplaceString>"

In text T substring SearchString is searched and replaced with ReplaceString.

Example::

"This is <RS|a old macro|old|new> in text"

turns into:

"This is a new macro in text"

Fill Left (FL)

"<FL|T|Size|Char>"

In the text T is filled on the left side with the character Char that T has the length Size.

Example::

Fill patient number to 8 characters with zeros:

"ID<FL|%1|8|0>"

becomes for patient 123456 to:

"ID00123456"

Fill Right (FR)

"<FR|T|Size|Char>"

In the text T is filled on the right side with the character Char that T has the length Size

Example:

Fill patient number to 6 characters with zeros:

```
"ID<FR|%1|8|0>"
```

becomes for patient 123456 to:

```
"ID12345600"
```

Trim Left (TL)

```
"<TL|T|Size|Char>"
```

Im Text T werden auf der linken Seite so viele Zeichen Char gelöscht, dass T die Länge Size hat

If the last character on the left side is not Char, the T remains unchanged

Example:

Delete leading zeros in patient number up to 8 characters:

```
"ID<TL|%1|8|0>"
```

becomes for patient 0000123456 to:

```
"ID00123456"
```

Trim Right (TR)

```
"<TR|T|Size|Char>"
```

In the text T on the right side so many characters Char are deleted that T has the length Size

If the last character on the right side is not Char, the T remains unchanged

Example:

Delete appended X in patient number up to 8 characters:

```
"ID<TR|%1|8|X>"
```

becomes for patient 123456XXXX to:

```
"ID123456XX"
```

```
"ID<TR|%1|8|X>"
```

becomes for patient 1234567890XXXX to:

```
"ID12345667890"
```

Crop Left (CL)

```
"<CL|T|Count>"
```

In the text T on the left side Count characters are deleted.

If T is shorter than Count, an empty string is returned.

Example:

Delete leading text "ID" in patient id:

```
"<CL|%1|2>"
```

becomes for patient ID123456:

```
"123456"
```

Crop Right (CR)

```
"<CR|T|Count>"
```

In the text T, Count characters are deleted on the right side.

If T is shorter than Count, an empty string is returned.

Example:

Delete year number in a patient ID:

```
"ID<CR|%1|5>"
```

becomes for patient 123456-2021 to:

```
"ID123456"
```

Contains Text (CTX)

```
"<CTX|T|Tag|Found Text|Not Found Text>"
```

In the text T the text Tag is searched. If it is found, the "Found text" is returned,

if not the "Not found text" is returned.

Example:

Check if a tag is included in the attributes:

```
"<CTC|{%pat:attributes}|;CLP;|cleft palate|>"
```

becomes for a patient with the attribute ;ORTH0;CLP;CHIR; to:

```
"cleft palate"
```

Position (POS)

```
"<POS|T|Tag>"
```

In the text T the text Tag is searched. If it is found, the position of the first character is returned,

if it is not found, an empty text.

Example:

Find out at which position a tag is included in the comment:

```
"<POS|{%pat:notes}|station:>"
```

becomes for a patient with the comment "Dr. Meier station:B5" to:

```
"11"
```

Conversion to UTF-7 (U7)

```
"<U7|T>"
```

The text T converted to UTF-7 encoding

Example:

```
"Name: <U7|%3 %2>"
```

becomes for the patient with the name "Björn Müller" to:

```
"Name: Bj+APY-rn M+APw-ller"
```

Encoding for URL (URLE)

```
"<URLE|T>"
```

The text T is converted to UTF-8 and then escaped for transmission as a URL.

See also: <https://en.wikipedia.org/wiki/URL-Encoding>

Example:

```
"<URLE|Name: %3 %2>"
```

becomes for the patient with the name "Björn Müller" to:

"Name%3A%20Bj%C3%B6rn%20M%C3%BCller"

Conversion to a valid file name (FILE)

"<FILE|T>"

In the T text, all characters that are not allowed for file or directory names are replaced.

Invalid characters are replaced (as already for the placeholder "%1") during correction as follows:

"*" -> "~1"

"?" -> "~2"

"@" -> "~3"

":" -> "~4"

""" -> "~5"

"/" -> "~6"

"<" -> "~7"

">" -> "~8"

"|" -> "~9"

Example:

"Model<FILE|%A>.PDF"

becomes the valid file name for the patient with the model number "123\A":

"Model123~0A.PDF"

Increment (INC)

The integer T is increased by INCVAL. The result is also an integer.

The texts in T and INCVAL must be convertible to an integer. Negative numbers are permissible.

Example:

Increase step number %S by 8:

"Step: <INC|%S|8>"

becomes for Step 4:

"Step: 12"

Value (VAL)

"<VAL|T|Tag>"

The text T should be a multiline (e.g.: %M Notes).

A search is made for a line that starts with the tag followed by "=", the text after the "=" is returned

Example:

Read out order number "Order" from the patient comment:

"Order #<VA|%M|ORDER>"

If there is a line "Order=0-8-15" in Notes (merge command %M), the result is:

"Order #0-8-15"

DateTimeFormat (DTF)

"<DTF|T|FS>"

The text T should be a date or time in one of the following formats:
yyyymmdd, yyyymmddhhnss or yyyymmddhhnn

In FS a format string must be passed, according to which date (and time) are formatted.

The syntax is described here:
<http://docwiki.embarcadero.com/Libraries/Sydney/en/System.TDateTime.FormatString>

Example:
<DTF|%4|dd.m.yyyy> -> "19.9.1990"
<DTF|%X|dddd mmmm d, yyyy hh:mm:ss AM/PM> -> "Tuesday June 15, 2021 10:39:00 AM"

Age (AGE)

"<AGE|STARTDATE|ENDATE|FORMAT>"

Der Text in STARTDATE und ENDATE sollte ein Datum or time in one of the following formats:
yyyymmdd, yyyymmddhhnss or yyyymmddhhnn

To output the patient age, %4 should always be transferred as STARTDATE.

As ENDATE, for example, "%{img:seriesdate}" or "%E" for the picture date can be transferred.

The formatting of the result can be selected with FORMAT. The following values are possible here:
SHORT, LONG, YEARS, DECI, DAYS or YYMMDD.

Examples:

Age on the date of the session:
<AGE|%4|{%img:seriesdate}|SHORT> -> "19.5 Years"
<AGE|%4|{%img:seriesdate}|LONG> -> "19.5 (19 Years, 6 Months)"

Age at the date of the picture:
<AGE|%4|%4|DECI> -> "19.5"
<AGE|%4|%4|DAYS> -> "7128"

Current age of the patient:
<AGE|%4|{%time:ansi}|YYMMDD> -> "190613"

Overview of placeholders whose validity

Description	Platzhalter	Short	Comment	3D Export Done	3D Series Export Done	Container Export Done	Container Import Done	Container Upload Done	Evaluation Done	Image Import Done	Module Closed	Module Opened	Project Saved	Report Export Done	Report Upload Done	Tooth Movement Export Done	Treated Image Saved	User Login	User Logout	Web Export Done	Drittprogramme	Export 3D (Files, Labeling)	Export Aligner-Protocol
Patient: Values are available when a patient is active.																							
ID		%0		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
ID, for filenames		%1	Patient ID corrected for file names	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Last name		%2		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
First name		%3		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Birthday		%4	Format: JJJJMMTT	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Gender		%5		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Last name (without space)		%8	von Schulz → vonSchulz	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
First name (without spaces)		%9	Anna Lena → AnnaLena	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Model number		%A		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Birthday		%B	Format: TTMMJJJJ	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Additional Patient ID		%C		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
Insurance number		%D		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

Links



Last update: **2024/04/08 13:20**

