

Meta Information using XML for the AXIS 7000

| | |
|---|----|
| Introduction | 1 |
| XML | 1 |
| AXIS 7000 Network Document Server | |
| Possible applications | 1 |
| System Requirements | 3 |
| How to do the integration | 4 |
| 1 Installing an FTP Server..... | 4 |
| 2 Adding an XML user..... | 4 |
| 3 Creating an XML DD file..... | 4 |
| Configuring the AXIS 7000..... | 5 |
| 4 Adding a Destination..... | 5 |
| 5 Meta Information configuration..... | 6 |
| Example of creating an XML DD file . | 7 |
| Creating the XML DD file..... | 7 |
| AXIS MetaData DTD | 9 |
| Pattern | 13 |
| Description..... | 13 |
| Internally defined variables | 14 |
| Default Output format | 17 |
| References | 18 |
| Axis Communications | 19 |

<http://www.axis.com>

Axis Communications (USA)
Tel: (978) 614-2000
Fax: (978) 614-2100

Axis Communications (Europe)
Tel: +46 46 272 18 00
Fax: +46 46 13 61 30

Axis Communications (Japan)
Tel: +81 3 3545 8282
Fax: +81 3 3545 8280

Introduction

This document describes how the AXIS 7000 can be integrated with different kinds of applications using its Meta Information functionality. It is intended for System Integrators, Developers, and Administrators.

The Meta Information functionality allows the integrator to define a list of information to be requested (it can be required or optional information) from the user at the moment of scanning. The user will be prompted for this information before the document is scanned and sent over the network.

The Meta information will be sent together with the scanned image in the Information File or as a separate document. The administrator can also choose to send information set by the AXIS 7000 itself, e.g. a time stamp and user identification. This attached Meta information can then be used by different applications such as accounting software or for storing in a database server under a unique name.

XML (eXtensible Markup Language) is used as the language to define the requested information and can also be used as the output format along with almost any other text based format.

XML

The XML (eXtensible Markup Language) specification defines a standard way to add Meta Information to documents. Please see the XML references in the end of this document for further information about XML.

AXIS 7000 Network Document Server

The AXIS 7000 offers the convenience of allowing end users to distribute and store documents digitally. The AXIS 7000 offers many options to do business in a more effective and cost-efficient way. With a wide variety of formats (TIFF, JPEG/JFIF, PDF, PCL) and transport methods (SMTP, FTP, LPD, Raw TCP) to choose from, an AXIS 7000 provides increased value to digital copiers and scanners. Included in its architecture are the embedded Web pages. Administrators can use an existing Web browser for configuration and management purposes, thus eliminating the need to load extra software.

Possible applications

There are numerous examples of how the Meta Information functionality can be used. Many work or business processes with documents involved can be improved by adding an AXIS 7000 integrated with an application server. The application server can be Knowledge Management Systems, Document management Systems, Imaging Systems, ERP Systems, Databases and Internet Services. Most application servers connected to a network can be integrated with the AXIS 7000.

Overview

Figure 1 shows the different components involved when using Meta Information and XML together with an AXIS 7000. A scanner or digital copier is connected to the network using an AXIS 7000. The AXIS 7000 can then be configured to prompt the user for information at the moment of scanning a document. The document will be scanned and sent as an image file to a server on the network together with an Information File. The Information File will include the values entered by the user and can be in a format defined by the administrator. Almost any format is possible. For instance XML, HTML or a list of values (separated by commas) would all be possible. XML is used as the language to define the requested information that the AXIS 7000 downloads from an FTP server on the network.

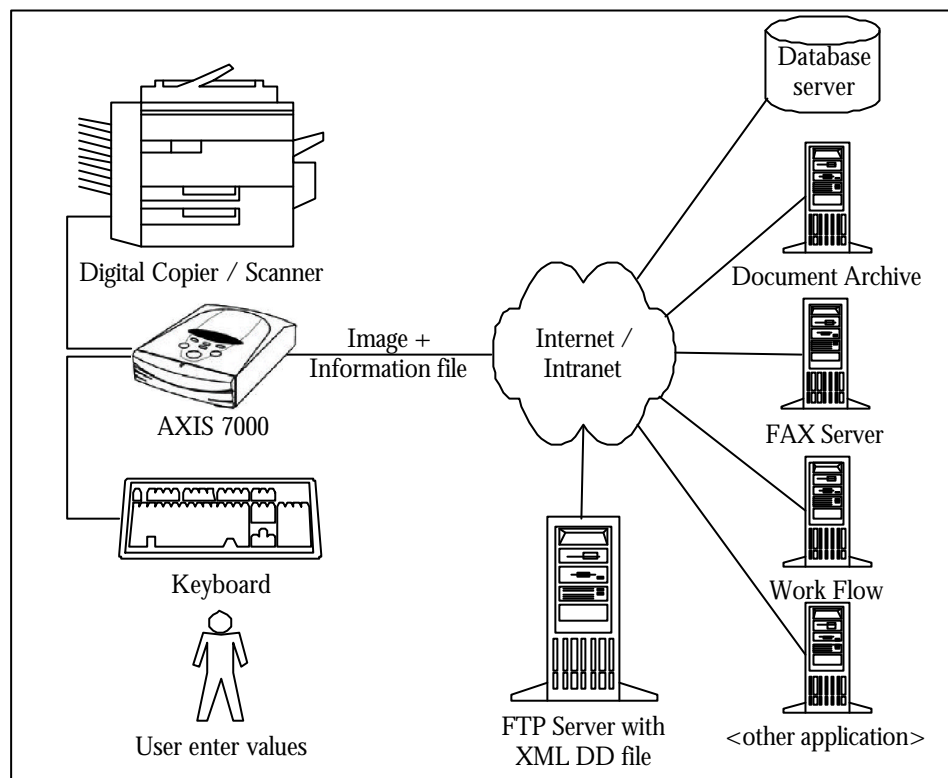
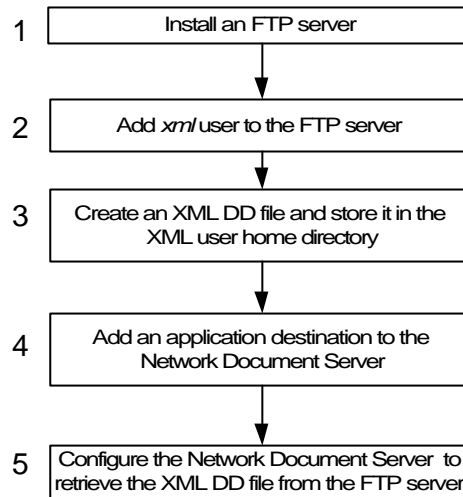


Figure 1: The components involved in an integration scenario.

How to do the integration

This section describes the general workflow for integrating the AXIS 7000 with an application. There is an overview in the flow diagram, followed by a more detailed description of each step in the process.



1 Installing an FTP Server

For the installation of the FTP server, please refer to the FTP server manual. The FTP server will be used for making the XML DD file available for the AXIS 7000. It can also be used for storing the scanned images and Information Files if a File Destination is used.

2 Adding an XML user

A new user, *xml*, will be added to the FTP server user list. The creation of new users and configuration of their properties differ from each FTP server. Please refer to the FTP server manual for further information. The *xml* user can have any directory as a home directory but it must be appropriate for storing setting files. It is recommended that you create a new directory for use as the *xml* user home directory.

3 Creating an XML DD file

The XML DD file specifies what the user will be prompted for and the format and contents of the output Information File. It can be created and edited in any text editor. The AXIS MetaData DTD defines the syntax of the XML DD file. The XML DD file will be stored in the home directory of the *xml* ftp-user, where the AXIS 7000 will be configured to retrieve it.

The XML DD file will differ depending on which kinds of applications the AXIS 7000 will be integrated with. There are two issues that need to be resolved prior to writing the file:

- What will the user be prompted for
 - Is the user required to enter the value?
 - What are the permitted values?
 - Is there only a set of valid values?
- The format of the Information File

An example of an XML DD file is shown below:

```

<?xml version='1.0'?>
<!DOCTYPE MetaDataFile SYSTEM "metadata.dtd">
<MetaDataFile>
  <Ask><When><Destination Description="Invoice" /></When>
    <MetaData Required = "yes"
      Id = "INVOICE_NB"
      Prompt = "Invoice Nb:"
      ErrorMessage = "Wrong Nb!"
      DefaultValue = "00000"
      Pattern = "[0-9]{5}"/>
    <MetaData Required = "yes"
      Id = "PRIORITY"
      Prompt = "Priority:"
      DefaultValue = "Medium"
      Editable = "no">
      <Value>Very High</Value>
      <Value>High</Value>
      <Value>Low</Value>
      <Value>Very Low</Value>
    </MetaData>
  <Output>
    <Format>
      InvoiceNb = "$INVOICE_NB$"
      Priority = "$PRIORITY$"
      ImageURL = "$INT_FILE_URL$"
    </Format>
  </Output>
</Ask>
</MetaDataFile>

```

The components of the XML DD file are the following:

- Internal variables: In the Information File, it is not only the values entered by the user that can be included but the internal AXIS 7000 values as well. Examples of variables can be the date, scanner name etc.
- Format section: The format of the Information file is specified here.
- List of values: The Administrator can specify a set of valid values the User can choose from. It is also possible to specify whether the User is allowed to enter his/her own values
- Pattern: It is possible to determine what the user is permitted to enter by specifying a pattern. Regular Expressions are used for this.
- Prompts: Each prompt needs a MetaData element entry.

4,5 Configuring the AXIS 7000

The AXIS 7000 can be configured through its internal Web-interface. General knowledge of the product is prerequisite. Please refer to the AXIS 7000 User's Manual for information regarding the basic configuration.

4 Adding a Destination

A new destination is added in the **Destination** section of the **Admin** part of the Web-interface of the AXIS 7000. Figure 3 shows an example of how it can be configured for a File Destination.

Note: The *Destination Name* must match a Destination Description in the XML DD file.

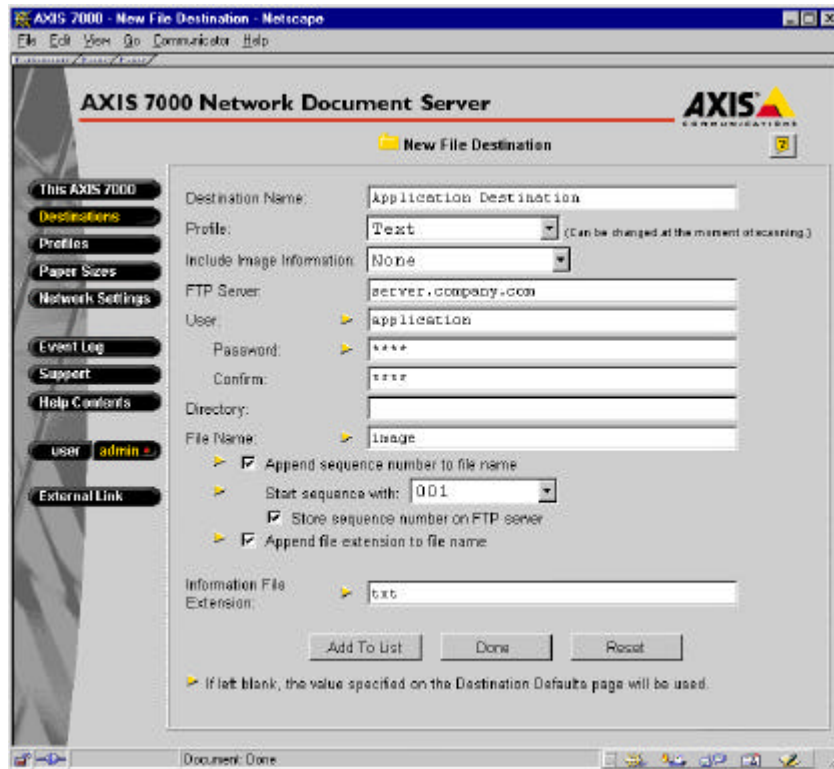


Figure 3

The value of the parameter Information File Extension will differ depending on which application is used. It could be *html* or *xml* for instance. The default value is *txt*.

5 Meta Information configuration

The AXIS 7000 has to be configured to retrieve the XML DD file from an FTP server in the network. This is done on the **Meta Information** tab in the **Edit** window accessible from the first **Admin** page of the Web-interface. Figure 4 shows an example of how it can be configured. A complete information concerning the parameters is available online.

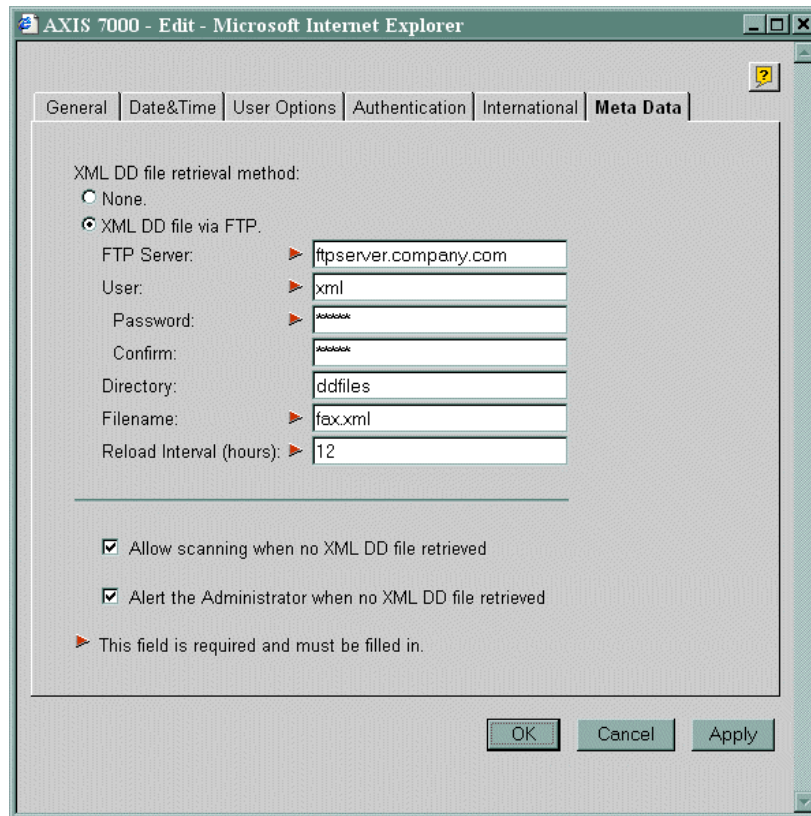


Figure 4

Example of creating an XML DD file

This section describes an example of how to create an XML DD file for an invoice application. The user will be prompted for a few specific values when scanning to a specific *invoice* destination. These values will be included in an XML formatted Information File sent together with the image file.

Creating the XML DD file

It is good practice to make a table with all prompts and their properties prior to writing the XML DD file. A simple example for this particular case is shown in Table 1.

| Prompt | Allowed values | Required |
|----------------|---|----------|
| Invoice Number | 00000-99999 | Yes |
| Due Date | A valid date in mm/dd / yyyy formats. | Yes |
| Issuer | A string | Yes |
| Amount | 0-9999999999 | No |
| Importance | Enumerated – not editable by the User. E.g. :Very High, High, Medium, Low, Very Low. | No |

Table 1

The Administrator has also the possibility to specify a list value the User can choose from. Apart from specifying the prompts, the format of the Information File must also be decided upon. If there is no <FORMAT> section of the XML DD file a default format will be used as described in *Default Output format*. In this case the Information File will be in XML format and therefore a CDATA section is used to prevent the XML parser in the AXIS 7000 from parsing that part of the code. Otherwise, normally only a <FORMAT> section is needed.

```

<?xml version='1.0'?>
<!DOCTYPE MetaDataFile SYSTEM "metadata.dtd">
<MetaDataFile>
<Ask><When><Destination Description="Invoice" /></When>
<MetaData Required      = "yes"
      Id                = "INVOICE_NB"
      Prompt            = "Invoice Nb:"
      ErrorMessage      = "Wrong Nb!"
      DefaultValue     = "00000"
      Pattern           = "[0-9]{5}"
/>
<MetaData Required      = "yes"
      Id                = "DUE_DATE"
      Prompt            = "Due Date:"
      ErrorMessage      = "Wrong Date!"
      DefaultValue     = "01/01/2000"
      Pattern           = "[0-9]{2}/[0-9]{2}/[0-9]{4}"
/>
<MetaData Required      = "yes"
      Id                = "ISSUER"
      Prompt            = "Issuer:"
      ErrorMessage      = "Enter the name."
      Pattern           = ".+"
/>
<MetaData Required      = "no"
      Id                = "AMOUNT"
      Prompt            = "Amount:"
      ErrorMessage      = "Enter the amount."
      Pattern           = "[0-9]{,10}"
/>
<MetaData Required      = "no"
      Id                = "IMPORTANCE"
      Prompt            = "Importance:"
      DefaultValue     = "Medium"
      Editable          = "no">
  <Value>Very High</Value>
  <Value>High</Value>
  <Value>Low</Value>
  <Value>Very Low</Value>
</MetaData>
<Output>
<Format>
<![CDATA[<?xml version="1.0"?>
<!DOCTYPE Invoice SYSTEM "invoice.dtd">
<Invoice InvoiceNb = "$INVOICE_NB$"
      DueDate      = "$DUE_DATE$"
      Issuer       = "$ISSUER$"
      Amount      = "$AMOUNT$"
      ImageURL    = "$INT_FILE_URL$">
</Invoice>
]]>
</Format>
</Output>
</Ask>
</MetaDataFile>

```

For further information regarding writing an XML DD file please refer to the AXIS 7000 documentation and the sections *AXIS MetaData DTD*, *Pattern syntax* and *Internal variables*.

AXIS MetaData DTD

The DTD (Document Type Definition) is a specific definition of the rules and syntax that the DD (Document Description) file must follow. It contains the definition of the element hierarchy and relation between these elements in the hierarchy. When your XML DD file is processed, it is compared to its associated DTD to be sure that it is structured correctly and that all tags are correctly used. The AXIS MetaData DTD file defines the syntax of the XML DD file used for specifying the prompts that will be shown on the AXIS 7000 control panel display as well as the format of the Information File with the entered values. As from the 2.23 release of the firmware, you have the possibility to download from the AXIS 7000 into your favorite validating editor, the DTD used internally. This mechanism allow you to create, validate and upload the AXIS 7000 XML DD file in a straight forward process. See the User Manual for more information.

```

<?xml version="1.0" ?>
<!--
=====
FILE:      metadata.dtd
ROOT:      MetaDataFile
DESCRIPTION: This file contains the DTD for the Meta Data functionality. This
            functionality allows the Administrator to define custom information that will
            be required from the User at the moment of scanning. This custom information
            can either be related to a particular destination (for instance an Invoice
            number when an invoice destination is selected) or it can be requested
            regardless of the destination.
===== --
>
<!--
=====
ENTITY:      MD_DTD_VERSION
DESCRIPTION: This entity contains the version number of this DTD.
===== -->
<!ENTITY MD_DTD_VERSION "1.00">
<!--
=====
ENTITY:      MD_DOMAIN
DESCRIPTION: This entity contains the domain of this DTD.
===== --
>
<!ENTITY MD_DOMAIN "XML meta data">
<!--
=====
ELEMENT:      MetaDataFile
PARENT:      None
CHILD:      AlwaysAsk (optional)
            Ask (zero or more)
ATTRIBUTES:  None
DESCRIPTION: This element is the root element. It allows the Administrator to define meta
            data that will always be requested (gathered in the "AlwaysAsk" section)
            regardless of the destination, and meta data that will be requested only when
            a particular destination is selected (gathered in the "Ask" section).
===== --
>
<!ELEMENT MetaDataFile (AlwaysAsk?, Ask*)>
<!ATTLIST MetaDataFile
            Version CDATA #FIXED "&MD_DTD_VERSION;"
            Domain CDATA #FIXED "&MD_DOMAIN;">
<!--
=====
PARENT:      MetaDataFile

```

```

CHILD:      MetaData (one or more)
            Output (optional)
ATTRIBUTES: None
DESCRIPTION: This element gathers the meta data that is requested from the user
            regardless of the destination. This does not imply that the different meta
            data is required but just that it is not destination dependent as the meta
            data gathered in the "Ask" section is. For instance the Administrator can
            always ask the User to identify himself. This is not destination dependent.
===== --
>
<!ELEMENT AlwaysAsk (MetaData+, Output?)>
<!--
=====
ELEMENT:      Ask
PARENT:      MetaDataFile
CHILD:      When (one)
            MetaData (one or more)
            Output (optional)
ATTRIBUTES: None
DESCRIPTION: This element gathers the different meta data that is Destination dependent.
            The Administrator is able to specify the meta data which is to be requested
            from the User on selecting a specific destination (gathered in the "When"
            element). This section also allows the Administrator to define the specific
            format of the output for this set of MetaData. For instance, the
            Administrator can ask for an invoice number when the User selects the invoice
            destination. He can also define a particular output format that can easily be
            processed (by a database for instance).
===== --
>
<!ELEMENT Ask (When, MetaData+, Output?)>
<!--
=====
ELEMENT:      MetaData
PARENT:      AlwaysAsk
            Ask
CHILD:      Value (zero or more)
ATTRIBUTES: Required:
            This attribute is an enumerated value that defines if the User must enter
            the MetaData (="yes") or if it can be skipped (="no") or if it is hidden to
            the User (="hidden"). MetaData with the "Required" attribute defined as
            "yes", is mandatory and must be entered before the User can proceed with
            the scanning. MetaData with the "Required" attribute defined as "hidden",
            is useful when the administrator wants to define a default value that will
            help in processing the information entered by the User.
            Id:
            This attribute indicates the system and name the MetaData is known by. This
            identifier must be unique among all the MetaData elements.
            Prompt:
            This attribute contains the string that will be displayed on the LCD
            display when requesting this particular MetaData from the User. If the
            Administrator does not specify the text the User will be prompted with the
            "Id" attribute.
            Label:
            This attribute contains the output string when the Administrator has not
            defined an output format. If no labels are defined, the "Id" attribute will
            be used instead.
            ErrorMessage:
            This attribute contains the string that will be displayed when the value
            entered by the User does not match the "Pattern" attribute. If the
            Administrator does not specify this attribute, the default string
            "Incorrect value!" will be displayed instead.
            DefaultValue:

```

This attribute contains the output string for when the MetaData is not required and not entered by the User (attribute "Required" equal "no") or hidden (attribute "Required" equal "hidden"). The default value for this attribute is the empty string.

Pattern:
This attribute contains the regular expressions that describe the expected format in the value entered by the User. For more details on the Regular Expression syntax see the User's Manual. The default value for this attribute is the match-everything pattern (".*").

Editable:
This attribute is an enumerated value that defines if the User is allowed to enter a value different from the one purposed in the list of value defined for this Meta Data. The User will be allowed if the attribute is set to "yes" and will not be if the attribute is et to "no"

DESCRIPTION: This element contains the information specific to the meta data itself. See the attribute descriptions for more details.

```

===== --
>
<!ELEMENT MetaData (#PCDATA | Value)*>
<!ATTLIST MetaData
    Required      (yes | no | hidden) "no"
    Id            ID      #REQUIRED
    Prompt       CDATA #IMPLIED
    Label        CDATA #IMPLIED
    ErrorMessage CDATA #IMPLIED
    DefaultValue CDATA #IMPLIED
    Pattern      CDATA #IMPLIED
    Editable     (yes | no) "yes">
<!--
===== --
ELEMENT:      Value
PARENT:      MetaData
CHILD:       None
ATTRIBUTES:  None
DESCRIPTION:  This element contains a specific value for a Meta Data. This allow the User
              to choose the value of a certain Meta Data among several value.
===== --
>
<!ELEMENT Value (#PCDATA)>
<!--
===== --
ELEMENT:      When
PARENT:      Ask
CHILD:       Destination (one or more)
ATTRIBUTES:  None
DESCRIPTION:  This element contains a list of the different destinations that trigger the
              request of the dependent MetaData elements from the User. When one or more of
              the specified destinations is selected by the User, the following MetaData
              will be requested
===== --
>
<!ELEMENT When (Destination+)>
<!--
===== --
ELEMENT:      Destination
PARENT:      When
              To
CHILD:       None
ATTRIBUTES:  Description:
              This is the Id of the Destination. One restriction is that the
              destination's description attribute must be a valid destination
===== --

```

```

description. This means that the Administrator cannot define a new
destination here.
DESCRIPTION: This element contains the description of the defined destination
===== --
>
<!ELEMENT Destination (#PCDATA)>
<!ATTLIST Destination
        Description          CDATA          #REQUIRED>
<!--
=====
ELEMENT:      Output
PARENT:      AlwaysAsk
              Ask
CHILD:       To (optional)
              Format (optional)
ATTRIBUTES:  FileName
              This attribute indicate to the AXIS 7000 which template it should use in
              order to determine the remote file name. This template can contain
              references to Meta Data's Id, in order to create a file name that is
              created with the values entered by the User. This attribute overrides the
              value set for the Destination.
DESCRIPTION: This element contains the specific information for when the Administrator
              wants to specify a format different from the traditional one. The
              Administrator can also specify a destination for the output file.
===== --
>
<!ELEMENT Output (To | Format | (To, Format))>
<!--
=====
ELEMENT:      To
PARENT:      Output
CHILD:       Destination
ATTRIBUTES:  None
DESCRIPTION:  This element contains the particular destination to which the output file
              should be sent.
===== --
>
<!ELEMENT To (Destination)>
<!--
=====
ELEMENT:      Format
PARENT:      Output
CHILD:       None
ATTRIBUTES:  None
DESCRIPTION:  This element contains the custom declaration that defines the output format
              defined by the Administrator for this specific destination.
===== --
>
<!ELEMENT Format (#PCDATA)>

```

Pattern

Each MetaData element is defined in the XML DD file. In this file a pattern is specified. The pattern is a set of regular expressions used to check the lexical correctness of the value. The format is enclosed into a string delimited by the "" (quote) character, new lines are not accepted in the regular expression. Infix notation is a structure notation where the operator is located between the operands, e.g. (A+B). In postfix notation the operator follows the operands, e.g. (A,B)+.

| Character | Description |
|-----------|--|
| . | Matches any character except new line. |
| + | (postfix) matches the previous expression one or several times |
| * | (postfix) matches the previous expression zero, one or several times |
| ? | (postfix) matches the previous expression once or not at all |
| [..] | Character set; ranges are denoted with -, as in [a-z]; an initial ^, as in [^0-9], complements the set |
| R{1,5} | r could be repeated between 1 and 5 times |
| R{1} | r must be repeated exactly once. |
| R{,2} | r could be repeated twice (i.e. 0, 1 or 2 times). |
| R{2,} | r must be repeated at least twice. |
| ^ | Matches at beginning of line |
| \$ | Matches at end of line |
| "a b" | "a" or "b" |
| | (infix) Alternative between two expressions |
| \ | \ quotes special characters. |

The following table shows a few examples of regular expressions

| Example | Description |
|---|--|
| "[0-9]{2}/[0-9]{2}/[0-9]{4}" | Date DD/MM/YYYY |
| "[a-zA-Z+] ([a-zA-Z]+.[a-zA-Z]+)@[a-zA-Z]+.[a-zA-Z]{2,3}" "(\([0-9]{3}\))?[0-9]{3}[1{,1}[0-9]{4}" | E-mail address firstname@companyname.com firstname@companyname.se Telephone number "(123)1231234" "(123)1231234" "123 1234" "1231234" |
| "[a-zA-Z]+" | Name "John Doe" "John" |
| "[0-9]{2} ?[0-9]{3}" | ZIP code "12345" "12 345" |
| "(1 2 3 4 5 6 7 8 9 10 11 12):[0-9]{,2}(:[0-9]{0,2}) ?(AM am Am PM pm Pm)" | Time "1:05:59 AM" |
| "(^((http://) (ftp://) (https://) (gopher://))?([a-zA-Z0-9]+.)*[a-zA-Z0-9]+(/([a-zA-Z0-9] - .)+)*" | URL "http://www.axis.com/prod/index.html" "ftp://ftp.axis.com/pub/" "www.axis.com" |

Internally defined variables

There are several internal variables defined by the AXIS 7000 that can be output in the Information File. They can be used as any other variable in the XML DD file. Example:

```
...
<Format>
The image is located at $INT_FILE_URLS
The time at the moment of scanning was $INT_TIMES
</Format>
...
```

| Variable name | Description |
|--------------------|--|
| INT_BITS_PER_PIXEL | This reference name will be used to output the number of bits per pixel of the image file. Example : 1 |
| INT_CONTRAST | This reference name will be used to output the contrast of the image file. Example : 50 |
| INT_DATA_TYPE | This reference name will be used to output the data type of the image file. Example : Black & white |
| INT_DATE | This reference name will be used to output the date of the image file. Example : 2000-05-04 |
| INT_DESTINATION | This reference name will be used to output the destination description that originated the Information File. Example : Invoice |
| INT_DOUBLE_SIDED | This reference name will be used to output the orientation of the image file. Example : Off |
| INT_FILE_NAME | This reference name will be used to output only the file name of the image file. Example : image.tif |
| INT_FILE_SIZE | This reference name will be used to output the size of the image file. Example : 54490 |
| INT_FILE_URL | This reference name will be used to output the file URL of the image file. In the case of a file on an FTP server, this will correspond to the complete URL of the file. In the case of an Email to a pre-set destination it will correspond to the URL specified on the web interface. Example : ftp://server/image.tif |
| INT_FORMAT | This reference name will be used to output the format of the image file. Example : CCITT G.4 |
| INT_HEIGHT | This reference name will be used to output the height of the image file. Example : 3300 |

| | |
|-----------------------|---|
| INT_HOST_ADDRESS | This reference name will be used to output the IP address of the AXIS 7000 that produced the image file. Example: 10.13.3.231 |
| INT_HOST_NAME | This reference name will be used to output the host name of the AXIS 7000 that produced the image file. Example : computer.company.com |
| INT_INTENSITY | This reference name will be used to output the intensity of the image file. Example : 50 |
| INT_MAC_ADDRESS | The MAC-address of the AXIS 7000. Example: 00:40:8c:18:04:e2 |
| INT_NUMBER_OF_PAGES | This reference name will be used to output the number of page of the image file. Example : 2 |
| INT_PAPER_ORIENTATION | This reference name will be used to output the paper orientation of the image file. Example : Portrait |
| INT_PAPER_SIZE | This reference name will be used to output the paper size used for the scanning. Example : Letter |
| INT_PROFILE | This reference name will be used to output the profile used for the scanning. Example : Text |
| INT_SCANNER | This reference name will be used to output the descriptive name of the scanner that produced the image file. Example : RICOH IS450 |
| INT_SUBJECT | This reference name will be used to output the selected subject. Example : Here comes the document |
| INT_TIME | This reference name will be used to output the time of the image file. Example : 14:25:25 |
| INT_USER_ID | This reference name will be used to output the description of the user. Example : John Smith |
| INT_USER_MAIL | This reference name will be used to output the email address of the user. Example : user@company.com |
| INT_WIDTH | This reference name will be used to output the width of the image file. Example : 2528 |
| INT_X_RESOLUTION | This reference name will be used to output the horizontal resolution of the image file. Example : 300 |
| INT_Y_RESOLUTION | This reference name will be used to output the vertical resolution of the image file. Example : 300 |
| INT_LAST_RESET_DATE | This reference name will be used to output the date of the last time the counter has been reset. Example : "Thu Mar 22 15:20:54 2001" |

| | |
|------------------------|---|
| INT_COUNTER_LAST_RESET | This reference name will be used to output the number of page scanned since the last reset of the counter. Example : 102 |
| INT_COUNTER_TOTAL | This reference name will be used to output the number of page scanned since the beginning of times Example : 9261 |

Default Output format

The Information File includes all values that the user entered at the moment of scanning to a destination that specifies Meta Information. It is possible to either use a default format of the Information File or to specify an own. If there is no <FORMAT> section in the XML DD file the default format will be used. It is formatted like <ID> = <VALUE>. An example of an Information File is shown below. Except for the Meta Information it also includes the normal image information. It is possible to only get the Meta Information by setting the *Include Image Information* parameter for the destination to *None*.

```
; AXIS 7000 Parameter List, V2.20 June 12 2000
[General]
File name           = image.tif
File size           = 100164
Date                = Thu Jun 12 12:55:06 2000
Host address        = 10.13.3.231
Scanner             = RICOH IS450
Destination         = Invoice
Profile             = Text
Paper size          = Letter
Number of pages     = 1

[Image Information]
Width               = 2528
Height              = 3300
X Resolution (DPI) = 300
Y Resolution (DPI) = 300
Bits per pixel      = 1
Data type           = Black & white
Paper orientation   = Portrait
Format              = CCITT G.4
Double-sided        = Off
Intensity           = 50
Contrast            = 50

[Metadata]
INVOICE             = 12456
DATE                = 01/25/2000
ISSUER              = ACME Corp
AMOUNT              = 5020
```

References

There are several different resources on the Internet with information regarding XML. A few of them are listed below for reference.

www.w3.org/XML

www.xml.org

www.xml.com

Axis Communications

Axis is a world leader in the rapidly growing network-attached peripherals industry. Headquartered in Sweden, Axis designs and manufactures network-attached print servers, network document servers, CD/DVD servers, camera servers and storage devices based on its ThinServer™ Technology.

All Axis products leverage Axis' own ThinServer Technology, which allows any peripheral device to be directly attached to the network without a file server or PC. These devices are displayed on any desktop without changing client software and can be accessed and used by multiple clients using virtually any desktop and network operating system. The core of ThinServer Technology is the embedded software consisting of self-contained, "thin" versions of popular operating systems, a Web server for consistent, network-wide management, and an optimized 32-bit RISC chip complete with device I/O and network controllers for high-speed data transport.