

USER MANUAL

PrintManager

version 2.2

Erstellt von: Sebastian Weber
Geändert von: Sebastian Weber
Gespeichert am: 25. September 2006

Version: 81

Radebeul, den 05/04/06

Preamble

This is a legal agreement between you (either an individual person or a company) and Hartmann media solutions *H media*. You agree to be bound by this agreement by installing, copying or using the PrintManager software.

If you do not have a valid license for the PrintManager software, you are not authorized to install, copy or otherwise use the software.

Provided that you have a valid license and you agree with the terms of this agreement, Hartmann media solutions grants you the following rights:

1. You may install as many instances of the PrintManager software as you want, if this is in accordance with this agreement.
2. You may copy the PrintManager software or documents that accompany this software in accordance with this agreement.

The information contained in this document and any other document belonging to the PrintManager software is subject to change without notice and does not contain any liabilities to incur by Hartmann media solutions. Hartmann media solutions takes no charge for any errors that may appear in these documents. Errors and omissions are excepted. Any documentation that accompanies the PrintManager software is licensed for internal, non-commercial reference purposes only.

No part of the PrintManager software or any PrintManager software artefacts may be in any form given to third parties.

Hartmann media solutions reserves all rights not expressly granted in this license.

You may not reverse engineer, decompile or disassemble this software except that you have a written permission by Hartmann media solutions to do so.

This license applies to all updates and supplements that Hartmann media solutions will provide.

Hartmann media solutions may terminate your license if you fail to comply with the terms and conditions of this license. In this case you have to uninstall all your instances of the PrintManager software and you have to destroy all copies of the software and any of its artefacts.

THIS SOFTWARE IS PROVIDED „AS IS“ AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HARTMANN MEDIA SOLUTIONS OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes software developed by The Apache Software Foundation (<http://www.apache.org/>).

© Copyright 2006

H media

Andreas Hartmann media solutions
Ludwig-Richter-Allee 6
01445 Radebeul - Germany
mail@hmedia.de

All rights reserved.
Printed in Europe

All trademarks and registered trademarks used herein are the property of their respective owners.

Processing Note

This document is written with OpenOffice.org 2.0. The original file is saved as User Manual_060920.odt

Versions

Date/Version	Editor	Changes
10 th Apr 06 / 13	Sebastian Weber	Preamble, Pages
11 th Apr 06 / 29	S. Weber A. Hartmann	Final State, License
18 th Apr 06 / 32	S. Weber	Updates
20 th Sep 06 / 36	S.Weber	Version 2.2

Table of Contents

Preamble	iii
Processing Note	iv
Versions	iv
Preface	vii
1 General	vii
2 About this document	vii
3 Symbols and conventions	viii
3.1 Bold words or characters	viii
3.2 Hints	viii
3.3 General information	viii
3.4 Warnings	viii
3.5 Links	viii
3.6 Console conventions	viii
1 System Overview	9
2 The PrintServer	11
1 Server Preparation	11
1.1 General	11
1.2 Java Runtime Environment Installation	11
1.3 FTP Server Installation	11
2 Enable PrintClient Access	12
3 PrintServer Installation	12
4 PrintServer Configuration	13
5 PrintServer Commands	18
6 PrintServer Control (Windows only)	19
3 Fonts and Character Sets	21
1 Introduction	21

2 Custom Font Embedding	21
3 Changing the Character Set	22
4 Switching the FOP Distribution	23
4 The PrintClient	25
1 PrintClient Installation	25
2 PrintClient Command-Line Mode	26
3 PrintClient Graphical Mode	27
5 Creating PrintStyles	31
1 The PDF Creation Process	31
2 NSML Conversion	33
3 PrintStyle Deployment	35
6 Error Messages	37
1 PrintClient Error Messages	37
2 PrintServer Messages	39
7 Bibliography	41

Preface

1 General

The iNEWS Newsroom Computer System does not contain a function to print the contents of a queue in a fashionable manner. The PrintManager is a printing system that can be used as an iNEWS add-on that provides an easy to use easy to adapt printing function. Hence iNEWS users are able to make printouts to plan a show, to have paper-based backup material or to make the contents of a show available in situations or on places where the iNEWS client is not accessible.

2 About this document

The chapters of this manual cover the following topics:

- Chapter 1 – System Overview: Gives a general description of the components of the PrintManager and how these components act together.
- Chapter 2 – The PrintServer: Explains the installation and configuration of the PrintServer component in detail.
- Chapter 3 – Fonts and Character Sets: Explains how to insert custom fonts into the printout, how to change code pages the PrintServer uses to interpret the content of the iNEWS database and how the switch between different versions of the Formatting Objects Processor.
- Chapter 4– The PrintClient: Explains the installation and configuration of the PrintClient component.
- Chapter 5 – Creating PrintStyles: Explains how PrintStyles are used to convert the content of an iNEWS queue into a printable document and how such PrintStyles are structured.
- Chapter 6 – Error Messages: Gives a description of possible error messages separated by messages of the PrintServer and messages of the PrintClient.

3 Symbols and conventions

3.1 Bold words or characters

Characters and words in bold face type are used by the authors to **emphasize** a special region in the text.

3.2 Hints

Hints and additional information appear in this area.

The hints in the marginal left area help you to work effectively.

3.3 General information

Information like these help you not to miss important facts.

3.4 Warnings

WARNINGS LIKE THESE SHOULD PROTECT YOU FROM COMMON PROBLEMS AND FAILURES.

3.5 Links

Hypertext links are contained in the text as bibliography entries. The address of the link can be found in the bibliography.

3.6 Console conventions

Console commands, directory paths and console output are typed as in the following example:

The response of typing the command `java -version` on the command prompt results in the console output:

```
h:\>java -version
java version "1.5.0_01" Java(TM) 2 Runtime Environment, Standard
Edition (build 1.5.0_01-b08)
Java HotSpot(TM) Client VM (build 1.5.0_01-b08, mixed mode)
```


1 System Overview

The PrintManager is a printing system developed for the use with iNEWS. It facilitates the printing of rundown queues out of the iNEWS Client.

The PrintManager consists of two parts. The PrintServer that is installed on a server machine and processes print requests and the PrintClient that is installed on every iNEWS client PC. The overall print process contains the following steps:

1. With the help of the PrintClient an iNEWS user can choose a queue to print and an appropriate PrintStyle, which specifies format and style of the resulting document.
2. The PrintClient opens an FTP connection to the PrintServer and puts the users print request as a file in a specific directory on the PrintServer machine. The PrintServer scans this directory for incoming print requests.
3. If a request is found the PrintServer downloads the eligible queue from the iNEWS server and converts it into PDF according to the given PrintStyle. The result is saved as a PDF file in a dedicated output directory on the server.
4. The PrintClient scans this output directory. If the requested PDF file appears, it is downloaded by the PrintClient and is printed or displayed as desired by the user.

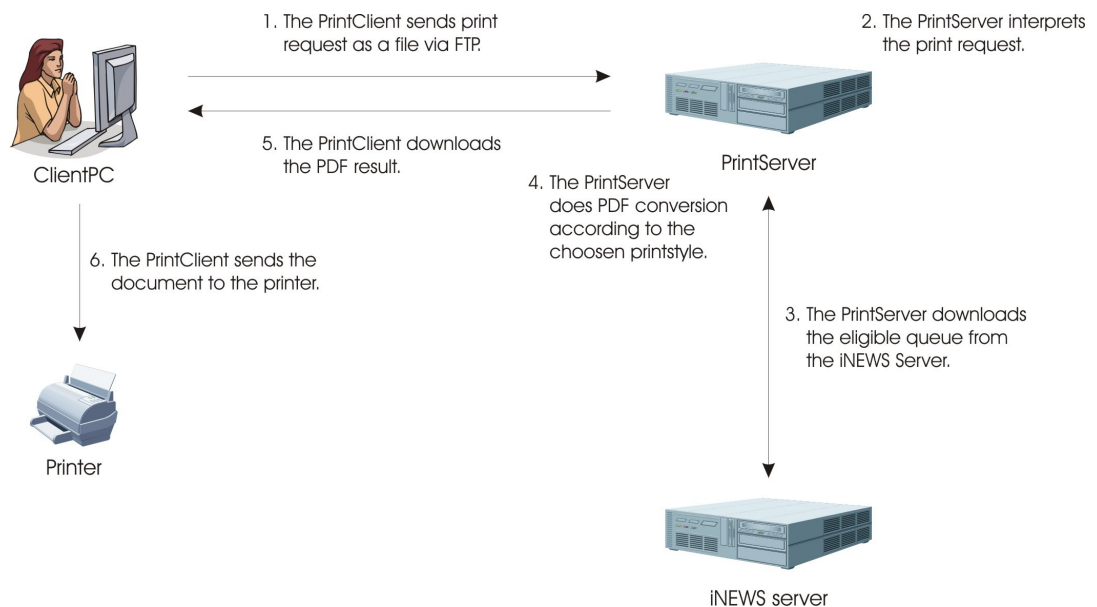


Figure 1: The printing process

2 The PrintServer

1 Server Preparation

1.1 General

The PrintServer is independent from the underlying operating system as it is implemented in Java. The PrintServer machine should comply with the following minimum system requirements: Pentium II, 256MB RAM, FTP connection to the iNEWS server. As the PrintClient connects to the PrintServer via FTP, an FTP server has to be installed on the PrintServer machine.

The Java Runtime Environment (JRE) Version 5 or higher has to be installed on the PrintServer machine.

The following chapters give a detailed description of how to prepare the PrintServer machine and how to install the PrintServer.

1.2 Java Runtime Environment Installation

The PrintServer needs a Java Runtime Environment (JRE) 5 or higher. The JRE can be downloaded at the [\[Java\]](#) website.

Make sure that the binary directory of the JRE is entered in the path variable.

The commands `java` and `rmiregistry` have to be executable from the command line. The version of the installed JRE can be checked with the command `java -version`. The output of this command should look similar to this:

```
...# java -version
java version "1.5.0_01"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0_01-b08)
Java HotSpot(TM) Client VM (build 1.5.0_01-b08, mixed mode, sharing)
...#
```

1.3 FTP Server Installation

The PrintClient needs FTP access to the server machine to send print requests to the PrintServer.

Thus an FTP server such as the VSFTP Server (for Linux) has to be installed on the server machine. See your operating system manual for further information.

2 Enable PrintClient Access

The communication between PrintClient and PrintServer is realized over the FTP protocol. That means that print requests from the PrintClient are sent as plain text files to the PrintServer and the resulting document is downloaded by the PrintClient via FTP. To log in to the server machine the PrintClient needs a username and password to establish an FTP connection. Furthermore the PrintClient expects a certain directory structure in the default directory it is in after logging in to the server.

In this documentation we assume that there is a user named **pclientuser** with password **pclientpass**. The home directory of this user is `{printclient_home} = /home/printclient`. This is the directory where the printclient user is in after logging in to the server machine. Now the PrintClient expects three subdirectories within the `{printclient_home}` directory. Those are:

- `{printclient_home}/incoming`
where it can store the print request. (The print request itself is a text file.) It is important that the printclient user has write access to this directory.
- `{printclient_home}/output`
where it can find the resulting PDF file. After downloading the PDF file the PrintClient deletes the file via FTP. Therefore it is important that the printclient user has write access to this directory.
- `{printclient_home}/status`
where it can find status information about the PrintServer. Status information is stored within text files. See end of chapter 2.4 for details. The printclient user does not need write access to this directory.

3 PrintServer Installation

To install the PrintServer copy the PrintServer directory from the installation medium into an appropriate directory of your server machine. In this documentation we will refer to this directory as the PrintServer home directory or `{printserver_home}`.

MAKE SURE THAT THE PATH TO THE `{PRINTSERVER_HOME}` DIRECTORY DOES NOT CONTAIN WHITESPACES.

In addition an environment variable called `PRINTSERVER_HOME` pointing to the `{printserver_home}` directory has to be created.

4 PrintServer Configuration

The `conf.xml` file within the `{printserver_home}` directory contains the configuration of the PrintServer. The configuration includes information about the iNEWS systems the PrintServer has to serve and the directories where the PrintServer will find print requests and store its output and temporary files in.

To configure the iNEWS systems the PrintServer has to serve, the following information have to be provided in the configuration at least:

- the names of the iNEWS systems
- the IP number for every server an iNEWS system consists of.
- the user name and password the PrintServer has to use if it connects to a specific iNEWS server. This user is an iNEWS user that has access to all the queues that might be printed.

Additionally the following directories have to be configured in the `conf.xml` configuration file:

- the directory where the PrintServer will find the PrintStyles
- the directory where the print requests will be filed. This is `{printclient_home}/incoming`
- the directory where the output will be filed. This is `{printclient_home}/output`
- the directory where the PrintServer can save its status information. This is `{printclient_home}/status`
- the directory for temporary files that the PrintServer will create
- the directory and file name for the log file.

Table 1 shows the XML structure of this file.

Element <i>Parent element</i> Description	Variables	Sub elements
PrintManagerConfiguration root element		serverList stylesheets stylesheetDirectory precompiledDirectory requestDirecotory outputDirectory statusDirectory workingDirektory logfile
serverList <i>PrintManagerConfiguration</i> Contains the configurations of the iNEWS systems that can be connected by this PrintServer.		iNews_System
iNews_System <i>serverList</i> Contains the configuration of an iNEWS System.	name = {name} - the name of the iNEWS system server	
server <i>iNews_System</i> Contains the configuration of an iNEWS Server within an iNEWS system.	ip = {string} - the IP address of the iNEWS server. maxConnections = {integer} - the max. number of connections that can be established by this PrintServer to this iNEWS server. usePrinterUser="ALWAYS" - other values than ALWAYS are not supported yet.	printerUser
printerUser <i>server</i> The user the PrintServer uses to login to the iNEWS server.		login password
login <i>printerUser</i> The login name of the printerUser.		

Element <i>Parent element</i> Description	Variables	Sub elements
password <i>printerUser</i> The password of the printerUser.		
stylesheets <i>PrintManagerConfiguration</i> The max. number of PrintStyles that the PrintServer holds in memory to speed up PDF conversion.		
stylesheetDirectory <i>PrintManagerConfiguration</i> The directory where PrintStyles are stored.		
requestDirectory <i>PrintManagerConfiguration</i> The directory on the PrintServer machine where the PrintClient stores its print requests. This is the <code>{printclient_home}/incoming</code> directory.		
outputDirectory <i>PrintManagerConfiguration</i> The directory on the PrintServer machine where the PrintServer stores the PDF files. This is the <code>{printclient_home}/output</code> directory.		
statusDirectory <i>PrintManagerConfiguration</i> The directory on the PrintServer machine where the PrintServer saves status information files. This is the <code>{printclient_home}/status</code> directory.		

Element <i>Parent element</i> Description	Variables	Sub elements
workingDirectory <i>PrintManagerConfiguration</i> The directory on the PrintServer machine where the PrintServer stores temporary files.		
logfile <i>PrintManagerConfiguration</i> The log file name is the value of the parameter 'name' of this element.	name = {logfile_name} - The log file name.	

Table 1: Structure of PrintServer configuration file conf.xml

The following is a sample configuration for a PrintServer that serves two iNEWS systems, named NRCS1 which is an A-B system and NRCS2 that consists of only one server. All three servers can be accessed by the PrintServer with the user named **printuser** and password **printuser**. This user is itself an iNEWS user who has to have access to all the queues on the iNEWS system an iNEWS user may want to print using the PrintClient.

```
<?xml version="1.0" encoding="UTF-8"?>
<PrintManagerConfiguration>
  <serverList>
    <iNews_system name="NRCS1">
      <server ip="10.32.24.1" maxConnections="100"
usePrinterUser="ALWAYS">
        <printerUser>
          <login>printuser</login>
          <password>printuser</password>
        </printerUser>
      </server>
      <server ip="10.32.24.2" maxConnections="100"
usePrinterUser="ALWAYS">
        <printerUser>
          <login>printuser</login>
          <password>printuser</password>
        </printerUser>
      </server>
    </iNews_system>
    <iNews_system name="NRCS2">
      <server ip="10.34.22.1" maxConnections="100"
usePrinterUser="ALWAYS">
        <printerUser>
          <login>printuser</login>
          <password>printuser</password>
        </printerUser>
      </server>
    </iNews_system>
```



```
</serverList>
<stylesheets>10</stylesheets>

<stylesheetDirectory>/usr/Printserver/stylesheets</stylesheetDirectory>

<requestDirectory>/home/printclient/incoming</requestDirectory>
  <outputDirectory>/home/printclient/output</outputDirectory>
  <statusDirectory>/home/printclient/status</statusDirectory>
  <workingDirectory>/usr/Printserver/temp</workingDirectory>
  <logfile name="/usr/Printserver/logfile.xml"/>
</PrintManagerConfiguration>
```

Additionally to the PrintServer configuration with the `conf.xml` file, PrintStyles have to be configured for each iNEWS system that can be connected by the PrintServer.

This can be done with the `iNewsSystems.txt` file within the `{printclient_home}/status` directory. In the `iNewsSystems.txt` file you can tell the PrintServer which PrintStyle is available for which iNEWS system and what the human readable names and file names of these PrintStyles are.

What follows is an example for the `iNewsSystems.txt` file. As in the example above, there are two iNEWS systems named NRCS1 and NRCS2. Both appear in brackets. For every PrintStyle its file name and its human readable name are given. Notice that the PrintStyle file name and the readable name are separated by a single space. The readable name of a PrintStyle must not contain spaces. The readable name of a PrintStyle given here will be shown in the PrintStyle box of the PrintClient.

```
[NRCS1]
eos-runorder.xsl Eye_on_Springfield_Runorder
eos-diary.xsl Eye_on_Springfield_Diary
eos-cg_playlist.xsl Eye_on_Springfield_Playlist
[NRCS2]
smartline-runorder.xsl Smartline_Runorder
smartline-diary.xsl Smartline_Diary
```

5 PrintServer Commands

The PrintServer can be controlled with console commands. Those commands have different names on Linux and Windows systems. As Linux commands can be found directly in the PrintServer directory; the Windows commands are located in the `/win` directory.

Linux	Windows	Description
<code>prt_startup</code>		Starts the PrintServer on Linux. This command first starts the Java RMI server and then the PrintServer. All messages of the PrintServer can be found in <code>/var/log/messages</code> . Exceptions thrown at start up, if any, can be found there.
	<code>printserver_.bat</code>	Starts the PrintServer on Windows. The Java RMI registry must be started first.
	<code>startrmi.bat</code>	Starts the Java RMI registry.
<code>prt_shutdown</code>	<code>shutdown.bat</code>	Disables the PrintServer.
<code>prt_status</code>	<code>status.bat</code>	Prints status information on the console and updates the status files in the <code>{printclient_home}/status</code> directory.
<code>prt_xmloutput</code>	<code>xmloutput.bat</code>	Enables/Disables the output of the result of the NSML to XML conversion. If enabled the resulting XML file can be found in the <code>{printclient_home}/output</code> directory under the same name as the resulting PDF file but with a <code>.xml</code> ending.
<code>prt_version</code>	<code>version.bat</code>	Print product version.
<code>prt_license</code>	<code>license.bat</code>	Print product license.

Table 2: PrintServer management commands

6 PrintServer Control (Windows only)

The PrintServer Control is a Graphical User Interface to control the PrintServer. All commands listed in Table 2 can be activated by a click on a button. The output messages of the PrintServer are printed into the PrintServer Control message window. Figure 2 shows the PrintServer Control after starting the PrintServer.

The PrintServer Control can be started with

```
printserver.exe
```

located in the `/win` directory of the PrintServer directory. Alternatively the PrintServer Control can be launched with the `start` option

```
printserver.exe start
```

Then it automatically starts the PrintServer after the PrintServer control is launched. This option can be used in Windows autostart if the PrintServer should be started automatically after power up.

If the PrintServer Control window is minimized it is automatically removed from the Windows taskbar and put into the system tray. If you want to open the PrintServer Control just double-click on the PrintServer Control tray icon. If the PrintServer Control window is closed while the PrintServer is running, the PrintServer will be stopped too. Therefore a warning message appears if the PrintServer Control is closed. If you answer this warning message with OK, then the PrintServer will be stopped and the PrintServer Control will be closed.

NEVER CLOSE THE PRINTSERVER CONTROL WINDOW IF YOU DON'T WANT THE PRINTSERVER TO BE STOPPED.

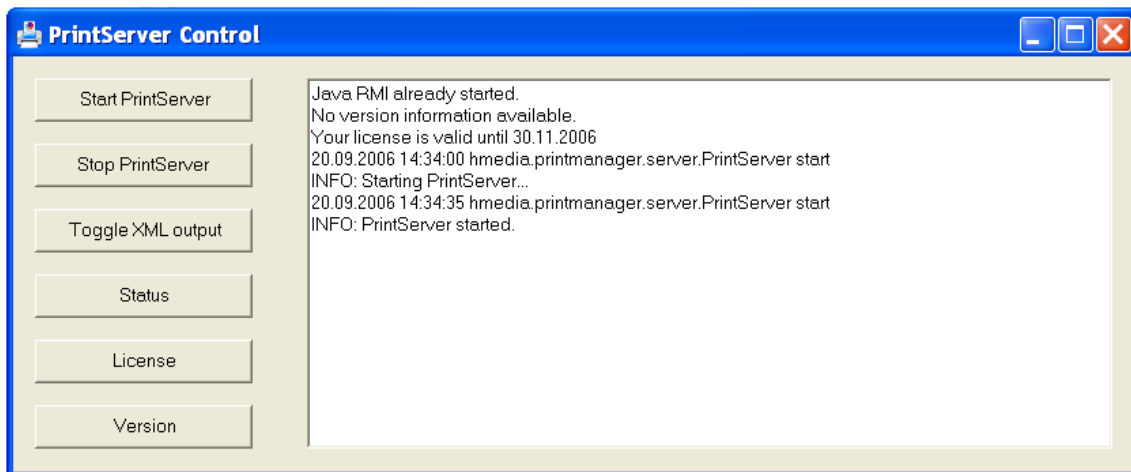


Figure 2: The PrintServer Control

3 *Fonts and Character Sets*

1 Introduction

A Formatting Objects Processor (FOP) is used to convert XML files to other file types such as PDF by using an XSL stylesheet. The PrintServer uses a FOP to convert the contents of an iNEWS queue represented in XML format to PDF by using a PrintStyle which is an XSL file. The FOP implementation the PrintServer uses is the Apache Formatting Objects Processor. The term FOP in this documentation will refer to the Apache FOP unless mentioned otherwise.

As the FOP creates the PDF output, changing its configuration can have an impact on the resulting PDF files. Special fonts can be added to the PDF output, for example to represent Cyrillic or Arabic script. As the development of the Apache FOP is still continuing the functionality of the Apache FOP can differ between the different FOP distributions. Some functions, for example right-to-left printing, are not available in any of the official FOP distributions. Therefore *H^{media}* offers a special distribution of the FOP to enable the PrintServer to print Arabic script.

The following sections describe how to use a custom font, how to use non-standard character sets and how to switch the FOP distribution that the PrintServer uses.

2 Custom Font Embedding

Custom fonts can be embedded in the printout. Currently only Type 1 and TrueType fonts are supported. If you want to embed custom fonts into your printout you have to make these fonts available to FOP. This is done in two steps.

1. Copy the existing Type 1 or TrueType font file into the `{printserver_home}/fonts` directory of the PrintServer. Now an XML file containing the font description has to be created and made available to FOP. This file can be automatically created using the PFMReader for Type 1 fonts or the TTFReader for TrueType fonts. Both programs are contained in the FOP Java archive located in the `{printserver_home}/lib/fop.jar` file. A detailed description of how to use these Reader programs can be found at [\[Font205\]](#) for FOP version 0.20.5 or [\[Font092\]](#) for FOP version 0.92. To find out which version of FOP you are using take a look into the `fopconfig.xml` file in your `{printserver_home}` directory. If this file contains the line

```
<fop version="1.0">
```

you know that you are using FOP version 0.92 or above. If the file does not contain a FOP version information at all, you are using FOP version 0.20.5.

2. After making the font file and its XML description file available in the `{printserver_home}/fonts` directory you have to edit the FOP configuration file `{printserver_home}/fopconfig.xml` to tell FOP the name of the font and where to find it. A description of how to edit the FOP configuration file to make custom fonts available to FOP can be found on the websites mentioned in step 1. A detailed description of the FOP configuration file can be found on [\[conf205\]](#) or [\[conf092\]](#) for FOP 0.20.5 and FOP 0.92 respectively.

Restart the PrintServer after embedding a new font.

The font made available to FOP as explained before can be used in a PrintStyle by referencing it with the `font-family` attribute on a formatting object block level element (see [W3Co3]).

3 Changing the Character Set

The iNEWS database uses ASCII encoding. Therefore a story saved in the database has exactly the same encoding as the encoding the iNEWS client PC uses. Which code page the PrintServer uses depends on the language and country configuration of the PrintServer machine. Because these language configurations can differ between the iNEWS clients and the PrintServer machine the printout can contain pointless characters. In this case you have to adjust the character set of the PrintServer.

First make sure that your Java installation contains the `charsets.jar` file in its `lib` directory. If not, reinstall the Java Runtime Environment or change its software configuration and add the Support for additional languages.

To change the the code page the PrintServer uses you have to change the PrintServer start script file. The name of this file depends on the operating system your PrintServer is installed on (see Table 2). If the PrintServer runs on Linux open the file

```
{printserver_home}/prt_startup
```

on windows systems open the file

```
{printserver_home}/win/printserver_.bat.
```

Both files contain a line with the `java` command followed by some options like `-Djava.rmi.server.codebase=...` and so on. These are parameters given to the Java Runtime Environment. If the PrintServer

should use a certain code page use `-Dfile.encoding=xxx` where the `x` stand for a code page listed in the supported encodings by Java 5 on [\[enc\]](#). For example if the contents of your database contain Cyrillic letters the `java` command would be followed by the name of the windows Cyrillic encoding set:

```
java -Dfile.encoding=windows-1251 -Djava.rmi.server...
```

If you want to print Arabic script the important part of the start script file may look like this:

```
java -Dfile.encoding=windows-1256 -Djava.rmi.server...
```

Besides changing the character set you also have to switch the FOP to the Arabic FOP if you want to print Arabic. Changing the FOP library will be explained in the next section.

Restart the PrintServer after changing the character set.

4 Switching the FOP Distribution

Currently the PrintServer ships with two different versions of FOP. Both versions are contained in your `{printserver_home}/FOP` directory or in the `PrintServer/FOP` directory on the installation medium. Table 3 gives an overview of these versions.

To switch the PrintServer to another distribution of FOP do the following:

- Disable the PrintServer
- Remove the current FOP library the PrintServer uses. That means `remove fop.jar` from the `{printserver_home}/lib` directory.
- Copy the library file of the FOP distribution you want the PrintServer to use from `{printserver_home}/FOP` to `{printserver_home}/lib` and rename it to `fop.jar`.
- If necessary update the FOP configuration file. The configuration files of different FOP distributions differ in structure. A detailed description of the FOP configuration file can be found on [\[conf205\]](#) or [\[conf092\]](#) for FOP 0.20.5 and FOP 0.92 respectively.
- Restart the PrintServer.

FOPversion	Library filename	Description
FOP 0.20.5	fop-0.20.5.jar	This is an older but stable release of FOP directly from Apache. It complies with most of the XSL-FO 1.0 standard. An overview of supported XSL-FO objects can be found at the FOP compliance page [comp] . <i>H^{media}</i> recommends to use this version of FOP.
FOP 0.92 arabic	fop-0.92-ar.jar	This is a mostly stable release of FOP patched by <i>H^{media}</i> for printing Arabic text right-to-left. It has the same compliance to XSL-FO as the official FOP 0.92 release from Apache has, but has added support for writing-mode. <i>H^{media}</i> recommends that this version of FOP is used in an Arabic environment only.

Table 3: Different FOP versions shipped with the PrintServer

4 *The PrintClient*

1 PrintClient Installation

The PrintClient is installed on the iNEWS client PC. It needs the Microsoft .NET runtime environment. Hence make sure it is installed on the PrintClient machine before continuing with the PrintClient installation procedure.

To display and print PDF files the Adobe Acrobat Reader program is used by the PrintClient. Make sure that this program is installed on the PrintClient machine. It can be downloaded for free from the [\[Adobe\]](#) website.

The `PrintClient.reg` file on the PrintClient installation medium contains information about available PrintServers. During installation this information is stored in to the Windows registry. It contains the names of the iNEWS systems and the PrintServers that are connected with these iNEWS systems. For every iNEWS system there must be at least one PrintServer. If two or more PrintServers are given for one iNEWS system the PrintClient first tries to connect the first PrintServer from the list. If no connection can be established it tries to connect with the second PrintServer and so on. Here is an example for the `PrintClient.reg` file.

```
Windows Registry Editor Version 5.00
[HKEY_LOCAL_MACHINE\SOFTWARE\HMedia]
[HKEY_LOCAL_MACHINE\SOFTWARE\HMedia\PrintClient]
[HKEY_LOCAL_MACHINE\SOFTWARE\HMedia\PrintClient\System]
"NRCS1"="PrintServer1;login1_1;password1_1;PrintServer2;login1_2;password1_2"
"NRCS2"="PrintServer2;login1_2;password1_2"
```

The last two lines should be adapted to your needs. Both lines start with the name of an iNEWS system. In our example the first system NRCS1 has two PrintServers available namely PrintServer1 and PrintServer2. For both PrintServers a user name and password is given that allows the PrintClient to connect to the PrintServer machine via FTP. The second iNEWS system NRCS2 has one PrintServer associated. In this configuration PrintServer2 serves as the primary PrintServer for NRCS2 and as the secondary PrintServer for NRCS1. PrintServer1 and PrintServer2 are the DNS names of the PrintServer machines. As an alternative to the DNS names the IP addresses of the PrintServer machines can also be used.

After customizing the `PrintClient.reg` file the installation process can be started with the `install.bat` file. The PrintClient will be installed in your program files directory. You can change the `install.bat` file, if you want the PrintClient to be installed into another directory.

2 PrintClient Command-Line Mode

The `PrintClient.exe` file can be started with command-line parameters. If the iNEWS system, the rundown queue and the PrintStyle are specified by the parameters the PrintClient starts printing without showing its graphical user interface. The following parameters are available:

Parameter	Description
-s [iNEWSServer]	Specifies the iNEWS system.
-q [queueName]	Specifies the queue.
-t [PrintStyleName]	Specifies the file name of the PrintStyle .
-c	<p>If this parameter is given the PrintClient tries to get the iNEWS server name and the queue name from the Windows clipboard.</p> <hr/> <p><i>In this case the -s and -q parameters are ignored.</i></p> <hr/>
-p [printerName]	Specifies the printer.
-o	If this parameter is given the output is shown in the Adobe Reader.

Table 4: Command line parameter of the PrintClient

For example if you want to print the queue `NEWS.RUNDOWN` on iNEWS system `NRCS` with PrintStyle `rundown.xml` you invoke the PrintClient from the command-line as follows:

```
PrintClient.exe -s NRCS -q NEWS.RUNDOWN -t rundown.xml
```

If you want to print the output on another but the standard printer you can specify the printer by **-p**.

For example if the output has to address a printer named `myPrinter` in the domain `mydomain` you invoke the PrintClient from command-line as follows:

```
PrintClient.exe -s NRCS -q NEWS.RUNDOWN -t rundown.xml -p
\\mydomain\myPrinter
```

If started with parameter **-c** the PrintClient expects the iNEWS system name and the queue name in the Windows clipboard. The data in the clipboard has to be normal text in the following format:

```
[iNEWSServer]queueName
```

For our example the clipboard entry looks like this:

```
[NRCS]NEWS.RUNDOWN
```

With this data in the Windows clipboard you can invoke the PrintClient from command-line as follows:

```
PrintClient.exe -c -t rundown.xml
```

To bring iNEWS system name and queue name into the Windows clipboard you can use an iNEWS macro and link this macro to a toolbar button or a keyboard shortcut.

The option to get the iNEWS system name and the queue from the Windows clipboard is useful if you want to print different queues with the same PrintStyle. In this case you copy the iNEWS system name and the queue the user is actually working in to the Windows clipboard and invoke the PrintClient.

The PrintClient switches to graphical-mode if any of the information about the iNEWS system name, the queue or the style sheet needed for printing is incomplete.

An error window is shown if any error occurs during the printing process. For error messages and possible solutions see chapter 6.

3 PrintClient Graphical Mode

If `PrintClient.exe` is started without any parameters or without all parameters required to start the printing process (see section 4.2) it will pop-up its graphical user interface (GUI).

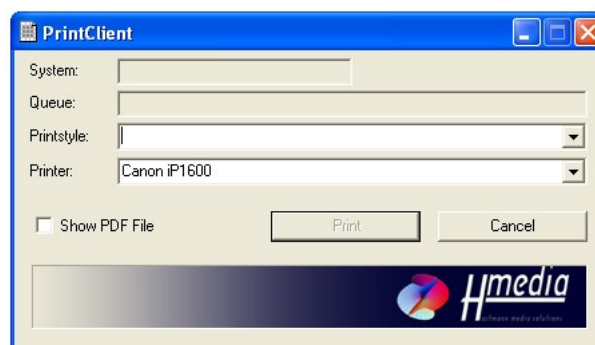


Figure 3: The PrintClient GUI

In the GUI there are two text fields named **System** and **Queue** which contain the iNEWS system name and the queue name of the queue that should be printed, respectively.

The box named **PrintStyles** is initially left blank because the available PrintStyles depend on the iNEWS system. It will be filled after entering an iNEWS system and queue name.

The box named **Printer** contains the available printers on your system. Initially the standard printer is chosen.

With the checkbox named **Show PDF file** one can chose to view the output file in the Adobe Acrobat Viewer. In this case the file is not printed. Hence the Printer box is disabled.

Furthermore there are two buttons. The **Print** button to start printing. This button is initially disabled and will be enabled if all necessary fields are filled. The **Cancel** button which always can be pressed to stop the printing process and close the PrintClient GUI.

Users can start the print procedure with filling the **System** field and **Queue** field. Both fields are write protected. There are two ways to fill in these fields. Either by starting the PrintClient with command-line options for system and queue name or by drag&drop the queue out of the iNEWS Client into the PrintClient.

Figure 4 - 6 show the actions needed to drag&drop the queue into PrintClient GUI.

After filling the **System** and **Queue** fields the PrintStyles are loaded from the PrintServer. This can take some seconds depending on your network speed. While loading the PrintStyles the PrintClient shows the message **Loading stylesheets. Please wait...** and is disabled for any input except the option to cancel. The message disappears if the PrintStyles are present and available in the **PrintStyle** box.

Now one can select an appropriate PrintStyle. Because the standard printer is selected per default all information necessary to start the printing process is now complete. Therefore the **Print** button is now enabled.

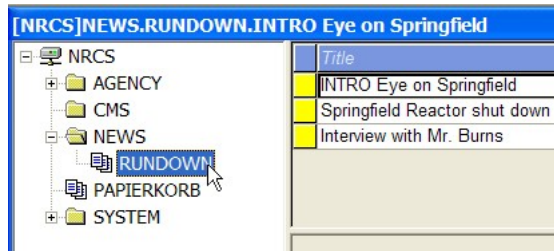


Figure 4: Click and hold the left mouse button pressed

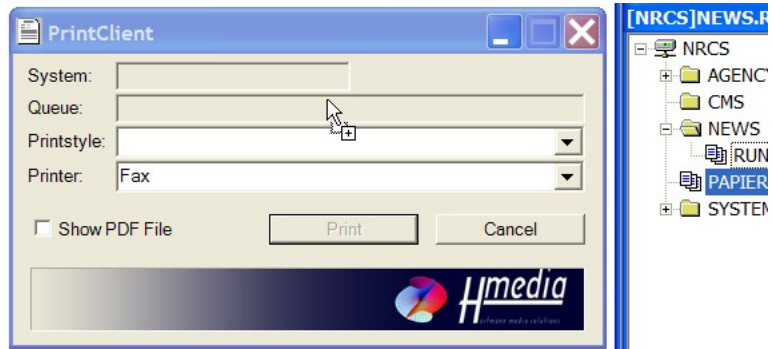


Figure 5: Drag the cursor over the PrintClient window

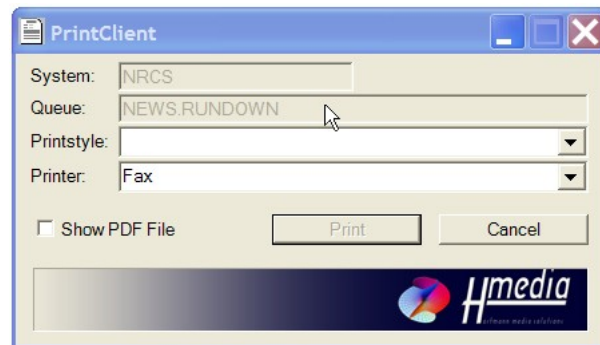


Figure 6: Drag the cursor over the PrintClient window

5 *Creating PrintStyles*

1 The PDF Creation Process

To create own PrintStyles it is necessary to understand how data from an iNEWS queue is converted into a printable document. Figure 7 shows this process, which consists of four steps:

1. After getting a print request the PrintServer pulls the complete rundown data out of the iNEWS database and writes it into a file.
2. This file contains the whole rundown in NSML format (see [AVIDo1]) and is parsed and converted to XML by the NSMLParser.
3. An XSLT processor can use this XML file and an XSL style sheet (the PrintStyle) to create an XSL-FO file. The XSL-FO file contains the rundown data together with formatting information, the so-called formatting objects (FO).
4. The Apache Formatting Objects Processor (FOP) is then used to convert the XSL-FO file into a PDF file which is finally downloaded and printed or displayed by the PrintClient/Adobe Acrobat Reader.

Hence creating own PrintStyles means to write an XSL style sheet file that can be used by the XSLT processor in step 3.

Therefore profound knowledge about XML, XSL and XSL-FO is needed to develop own PrintStyles. These topics are not covered in this documentation because they are not PrintManager specific and there are many websites and books devoted to. For additional information on the three topics you can consult one of the websites with tutorials by the W3C [W3Co1], [W3Co2], [W3Co3].

There are some differences between the original NSML as specified by [AVIDo1] and the converted NSML. They are explained in the next chapter. Chapter 5.2 gives an example of how a PrintStyle for a specific queue has to look like. In Chapter 5.3 the deployment of a PrintStyle will be explained.

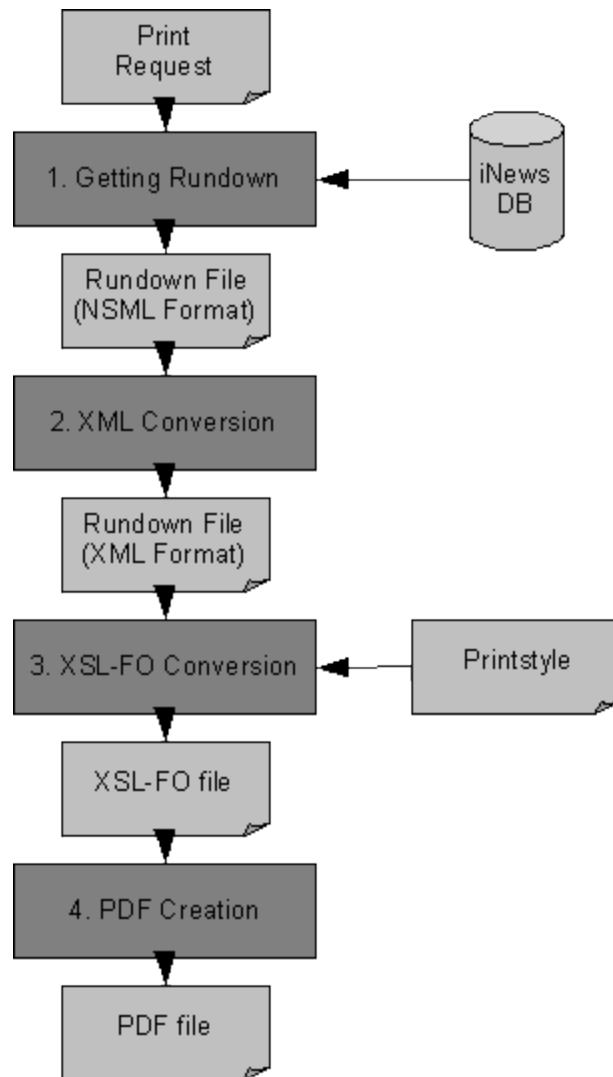


Figure 7: The PDF creation process

2 NSML Conversion

The idea of the PrintManager conversion process is to convert an XML file into a PDF file with the help of an XSL style sheet.

The source XML file in this case is, of course, the NSML file you get out of the iNEWS database. But NSML does not comply with basic XML syntax. Therefore some conversions have to be made by the PrintManager.

These conversions are made by the NSMLParser program and are part of step 1 in the conversion process explained previous in chapter 4.1.

If you want to write your own PrintStyle you have to know how converted NSML differs from original NSML. The differences arise from two problems:

1. Optional closing tags in NSML.

Explanation: The closing of some NSML elements with a closing tag is optional (as specified in [AVIDO1]) whereas XML requires closing tags for every start tag.

Solution: The NSMLParser inserts the closing tag.

2. Overlapping style tags

Explanation: Because font style tags in the story text are not closed, the NSMLParser can only assume where a specific region ends (e.g. a region of bold or green text).

Solution: The NSMLParser processes the story text by the following rules:

- a starting paragraph ends all preceding style regions
- red, green and normal text cannot be nested.

3. Boolean Variables without values.

Explanation: In NSML the presence of a variable name without any value means that this variable is true. But in XML every variable has to have a value.

Solution: The NSMLParser adds the value '1' to any variable that has no value set.

In addition to the conversions mentioned above, the NSMLParser encloses the stories of a queue with a root element called `NSMLRoot`. Therefore each converted NSML file starts and ends with the `NSMLRoot` element, containing an `nsml` element for each story in the queue. The following two examples show the differences between a story in NSML format and the same story converted by the NSMLParser.

Example 1: A story in NSML format.

```
<nsml>
<head>
<meta words=5 rate=135 float>
<storyid>03c917d4:00000471:42c9225e
<formname>default-form
<story>
<fields>
<f id=title>INTRO Eye on Springfield
<f id=modify-date>1120482530
<f id=modify-by>sebastian
<f id=bemerkungen>
<f id=ready>READY
<f id=audio-time>2
<body>This is some story text.</body></nsml>
```

Example 2: A story converted by NSMLParser.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<NSMLRoot>
<nsml version="-//iNEWS//DTD NSML 2.0//EN">
<head>
<meta words="5" rate="135" float="1"></meta>
<formname>default-form</formname>
<storyid>03c917d4:00000c9d:42c934e2</storyid>
</head>
<fields>
<f id="title">INTRO Eye on Springfield</f>
<f id="modify-date">1120482530</f>
<f id="modify-by">sebastian</f>
<f id="bemerkungen"></f><f id="ready">READY</f>
<f id="audio-time">2</f>
</fields>
<body>
<p>This is some story text.</p>
</body>
</nsml>
</NSMLRoot>
```

3 PrintStyle Deployment

The deployment of a PrintStyle is made in three steps:

1. Put the PrintStyle in the style sheet directory of the PrintServer.
2. Put any resources the PrintStyle needs into the directory referenced within the PrintStyle. Relative paths start within the home directory of the PrintServer.
3. Enter the name of the PrintStyle in the system status file. This file is located in the status folder of the PrintClient home directory on the PrintServer machine and is named `iNewsSystems.txt` (see chapter 2.4 for an explanation of this file).

Repeat these steps on each PrintServer the PrintStyle has to be installed on.

The next time you start the PrintClient the new PrintStyle appears in the PrintStyle box.

If the PrintServer holds PrintStyles in its memory to speed up the PDF conversion process then restart the PrintServer before the new or altered PrintStyle is used the first time to make sure that it is properly reloaded from disk.

THE STYLE SHEET DIRECTORY MUST NOT CONTAIN SUBDIRECTORIES BECAUSE PRINTSTYLES WITHIN SUBDIRECTORIES CANNOT BE LOADED BY THE PRINTSERVER.

6 Error Messages

1 PrintClient Error Messages

Message	Description
Can't load data from Windows registry.	The PrintClient configuration cannot be loaded from the Windows registry. Check the registry and make sure that there is a node <i>HKEY_LOCAL_MACHINE\SOFTWARE\HMedia\PrintClient\System</i> s. Inside this node there has to be a description of the PrintServer that this PrintClient can connect to. See chapter 3.1 for the structure of this description.
The Acrobat Reader cannot be found.	The program path of the Adobe Acrobat Reader program cannot be found in the Windows registry. Make sure that this program is installed on the PrintClient machine.
Cannot load data from clipboard.	If the PrintClient is started with option <i>-c</i> then it awaits a string in the Windows clipboard that describes an iNEWS system and a queue (see section 3.2 for details). If the clipboard content is no system and queue description this error is thrown.
The PrintServer is offline: [iNEWS system name]	All PrintServers for the iNEWS system displayed are offline.
The system is not configured: [iNEWS system name]	The iNEWS system with the given name is not configured in the configuration file <i>conf.xml</i> of the PrintServer that is connected by the PrintClient to execute print requests for this iNEWS system. (Restart the PrintServer after making changes in the <i>conf.xml</i> file)
The system is offline: [iNEWS system name]	The PrintServer cannot connect to the iNEWS system. Hence it assumes that the iNEWS system is offline.
Couldn't resolve remote host: [host name]	The host name of the PrintServer cannot be resolved. If this is not due to a network error check the PrintClient configuration for wrong host names or enter the IP address of the PrintServer instead of the host name.

Message	Description
<p>Couldn't connect to remote server: [server name]</p>	<p>The PrintClient could not establish an FTP connection to the PrintServer with the given name. If this is not due to a network error check the PrintClient configuration.</p>
<p>An error occurred: [error message]</p>	<p>The PrintClient shows an error message from the PrintServer. See chapter 5.2 for PrintServer error messages.</p>
<p>Cannot download PrintServer status file.</p>	<p>The status file of the PrintServer contains status information of the PrintServer. This file is set up automatically by the PrintServer. Make sure it exists in the right directory and can be downloaded by the PrintClient (see chapter 2.2).</p>
<p>Cannot load stylesheets.</p>	<p>The iNEWS systems status file of the PrintServer contains information about the iNEWS systems and their PrintStyles. Without these information the PrintClient cannot show the PrintStyles list. Make sure that the iNewsSystems.txt file exists (as described at the end of section 2.4) and can be downloaded by the PrintClient via FTP.</p>
<p>The PrintServer doesn't respond to the print request.</p>	<p>Though the status file of the PrintServer indicates that it is running the print request is not processed. Check if the PrintServer is running and is working correct.</p>
<p>The PrintServer is not configured: [iNEWS system name]</p>	<p>This error is thrown if the user started a print job for an iNEWS system that the PrintClient is not configured for. That means that this iNEWS system is not entered in the <i>HKEY_LOCAL_MACHINE\SOFTWARE\HMedia\PrintClient\System</i>s node in the Windows registry, where the PrintClient configuration is stored. See chapter 3.1 on how to configure the PrintClient.</p>

2 PrintServer Messages

Message	Description
SEVERE: The connection to the ftp server 'xyz' could not be established.	<p>The PrintServer could not connect to an iNEWS system FTP server. Shutdown the PrintServer and check these points.</p> <ol style="list-style-type: none"> 1. Check if the iNEWS system is running. 2. Check if the iNEWS systems FTP server can be accessed from the PrintServer machine. 3. Check the configuration file <i>conf.xml</i> of the PrintServer (see chapter 2.4)
java.io.FileNotFoundException: foo.xml (No such file or directory)	<p>A PrintStyle could not be found.</p> <ol style="list-style-type: none"> 1. Check if the PrintStyle exists in the style sheet directory. 2. Check if the file is misspelled in the <i>iNewsSystems.txt</i> file. 3. Check if the style sheet directory is entered correctly in the PrintServer configuration file <i>conf.xml</i>. <p>Restart the PrintServer.</p>
[Fatal Error] foo.xml:x:y: [...]	<p>Error messages that refer to a line in an xsl file are messages from the XML parser. They indicate a failure in a PrintStyle, mostly with additional text. Repair the PrintStyle and restart the PrintServer to make sure it is reloaded.</p>
Could not read the configuration file.	<p>The configuration file could not be found. Kill the PrintServer process. Make sure that a valid PrintServer configuration file is in the same path as the PrintServer program.</p>
java.io.IOException: Polling /foo/bar does not exist.	<p>The directory for incoming print requests does not exist. Check the element <i>requestDirectory</i> in the PrintServer configuration file <i>conf.xml</i>. Restart the PrintServer.</p>
SEVERE: The renderer can not create the output	<p>The PrintServer cannot write into the output directory. Check the element <i>outputDirectory</i></p>

Message	Description
file 'foo.pdf'.	in the PrintServer configuration file <code>conf.xml</code> . Make sure that the PrintServer has write access to this directory. Restart the PrintServer.
SEVERE: Error occurred while writing the status file: status.txt.	The PrintServer cannot write into the status directory. Check the element <code>statusDirectory</code> in the PrintServer configuration file <code>conf.xml</code> . Make sure that the PrintServer has write access to this directory. Restart the PrintServer.
SEVERE: The temporary file 'temp_bar.txt' could not be created.	The PrintServer cannot write into its temporary directory. Check the element <code>workingDirectory</code> in the PrintServer configuration file <code>conf.xml</code> . Make sure that the PrintServer has write access to this directory. Restart the PrintServer.

7 *Bibliography*

[Java] Sun Java Technology website, <http://java.sun.com>

[Font205] FOP 0.20.5 Fonts website, <http://xmlgraphics.apache.org/fop/0.20.5/fonts.html>

[Font092] FOP 0.92 Fonts website, <http://xmlgraphics.apache.org/fop/0.92/fonts.html>

[conf205] FOP 0.20.5 configuration website,
<http://xmlgraphics.apache.org/fop/0.20.5/configuration.html>

[conf092] FOP 0.92 configuration website,
<http://xmlgraphics.apache.org/fop/0.92/configuration.html>

[W3C03] W3C – XSL-FO Tutorial, <http://www.w3schools.com/xslfo/>

[enc] Java encodings website,
<http://java.sun.com/j2se/1.5.0/docs/guide/intl/encoding.doc.html>

[comp] FOP compliance page, <http://xmlgraphics.apache.org/fop/compliance.html>

[Adobe] Adobe website, www.adobe.com

[AVID01] iNews NRCS Engineering Specification News Story Markup Language 2.7, 2002

[W3C01] W3C – XML Tutorial, <http://www.w3schools.com/xml/>

[W3C02] W3C – XSLT Tutorial, <http://www.w3schools.com/xsl/>