



Image Import Application Package

Image Import Application

HPII - High Performance Image Import
MRII - Medium-Range Image Import

Installation and Configuration Guide

January 21, 2008
Version 3.3.8

Table of Contents

1. INTRODUCTION & OVERVIEW	1
2. DEFINITIONS	2
3. HARDWARE & SOFTWARE REQUIREMENTS	3
4. SOFTWARE INSTALLATION.....	4
4.1 UNIX PLATFORMS	4
4.2 WINDOWS PLATFORMS	5
4.3 COMPLETING INSTALLATION ON UNIX AND WINDOWS OPERATING SYSTEMS	6
4.3.1 “hosts” file entry	6
4.3.2 Environment PATH.....	7
4.3.3 ISTK file ownership and permissions on UNIX	7
5. CONFIGURING IMAGE IMPORT FACILITY	8
5.1 USER ENVIRONMENT	8
5.2 CACHE PARAMETERS.....	9
5.3 HP11.CFG/MR11.CFG.....	ERROR! BOOKMARK NOT DEFINED.
5.4 LOGON ATTRIBUTES.....	12
5.5 CONFIGURING ENCRYPTED PASSWORD (OPTIONAL)	13
5.6 DOCCCLASS ATTRIBUTES.....	15
5.6.1 F_DOCFORMAT System Field	15
5.6.2 F_DOCTYPE System Field	16
5.7 BATCH ATTRIBUTES	16
5.8 DIRECTORIES	22
6. CONFIGURING UTILITIES TO RUN FROM A 3RD PARTY SERVER	23
6.1 REQUIREMENTS.....	23

6.2	CONFIGURATION.....	23
6.3	RUNTIME.....	23
7.	HPII/MRII OPERATION.....	24
7.1	COMMAND LINE STARTUP	24
7.2	WINDOWS NT/2000 SERVICE SETUP.....	24
7.3	PROCESSES	25
7.4	RUNTIME.....	27
8.	PROGRAMS	29
9.	INTERFACE REQUIREMENTS	31
9.1	DIRECTORY STRUCTURE	31
9.2	FILE FORMATS.....	33
10.	TROUBLESHOOTING	37
10.1	UTILITY / ISTK VERSION RELATIONSHIP	37
10.2	ERROR LOGGING FILES	37
10.3	PROCEDURE FOR REPORTING PROBLEMS.....	38
10.4	WHEN TO USE WAL_PURGE – UNIX ONLY.....	39
10.5	HOW TO COMPUTE THE MAXSUBBATCHSIZE.....	39
10.6	FAST BATCH COMMITAL	41
10.7	DOCUMENT TYPE SETTING LIMITATION.....	41
10.8	OS PATH ISSUE	41
10.9	HP CDROM MOUNTING FOR ISO9660 CD TYPE.....	41
10.10	ERROR CODES THAT NEED SPECIFIC ACTIONS	41
11.	QUESTIONS AND ANSWERS	46
11.1	MRII GENERAL PRODUCT QUESTIONS	46

12.	APPENDIX A—EPHEMERAL PORT SETTINGS	47
12.1	DEFINITION	47
12.2	WINDOWS PLATFORM.....	48
12.3	<i>MAXUSERPORT</i>	48
12.4	<i>TCPMAXCONNECTTRANSMISSIONS</i>	48
12.5	<i>TCPMAXCONNECTRETRANSMISSIONS</i>	49
12.6	<i>TCPTIMEDWAITDELAY</i>	49
12.7	<i>FN_COR_QLEN</i>	50
12.8	UNIX PLATFORM.....	50
12.9	IBM AIX	50
12.10	HP-UX	52
12.11	SUN SOLARIS	52

1. INTRODUCTION & OVERVIEW

Installation by a FileNet Certified Professional (FCP) Recommended. For more information on the FCP program, refer to the FileNet Web site (<http://filenet.com>), Customer Center > Global Learning Services > Certification Programs.

This document describes the installation, configuration, and use of the Image Import products. The two products currently available are High-Performance Image Import (HP II) and Mid Range Image Import (MR II). Because of the similarities between the two products they will both be covered in this document. There are functionality differences between HP II and MR II. These differences will be documented in the specific sections where they occur. Read all sections before installing and running HP II/MR II to verify space and configuration requirements.

This document is intended to be read by people who are familiar with UNIX or Windows operating systems as well as FILENET's Image Services Server Software.

There is no difference in HP II/MR II functionality when running on UNIX or Windows. However, the method for installing HP II/MR II does differ on the two platforms. For installation details, see the section on installation specific to the target Operating System.

NOTE HP II and MR II do not specifically restrict document type or document structure. FILENET Image Services Server and/or the retrieval/display software define the limitations. For example, check the FILENET Viewer documentation for file formats that it will display.

2. DEFINITIONS

HPII – High Performance Image Import

MRII – Medium Range Image Import

Domain – The name of the FILENET Image Services.

IS – FILENET Image Services

ISTK – Image Services Toolkit (WorkFlo Application Library)

SCR – Software Change Request

OS – Operating System

WorkingDirectory – Formerly known as the ParentDirectory

3. HARDWARE & SOFTWARE REQUIREMENTS

HPII/MRII provides version releases compiled for currently supported ISTK versions. When installing on a UNIX operating system, the HPII/MRII installation program will determine the ISTK version release and install the HPII/MRII version, which coincides with the ISTK version found on the server. (Example: HPII/MRII 3.6 compile will be installed for ISTK 3.6 and HPII/MRII 4.0 compile will be installed for ISTK 4.0.) Installation on a Windows operating system will require the user to know the version of ISTK they have installed. The section on installation explains how to determine your ISTK version. Any other requirements are dependent upon the ISTK and Image Services Server supported hardware and software requirements. Image Services Server and ISTK must already be installed, configured, and running before installing HPII/MRII. See the Image Services Server and ISTK documentation for those requirements.

The HPII/MRII software installation requires a minimum of 10 MB of hard disk space. Runtime directories' space usage depends on the size of the user files, volume of batches, and other factors such as file retention schedules.

4. SOFTWARE INSTALLATION

4.1 UNIX Platforms

This section describes the installation procedure for UNIX systems **only**. (Refer to the next section “Windows Platforms” for installation instructions when installing HP II/MR II on a Windows Server.)

The following procedure installs HP II/MR II into the directory **/fnsw/local/bin**.

1. Log in as the root user, to the server where HP II/MR II be installed.
2. Insert the CD., execute the appropriate mount command for your UNIX Operating System, and change the directory to the CD-ROM mount point. This should place the user at the root of the CD where the “ps_install” executable is located.

NOTE These utilities are released on a CD type that is created for all supported operating systems. HP operating systems require a special mount command to read an ISO9660 CD type.

```
mount -o cdcase /dev/cd0 /cdrom
```

3. Run the installation program for HP II/MR II using the following command.

```
> csh ./ps_install.
```

NOTES The periods are needed before and after the command.

The “csh” shell specification in the command is needed if the current user environment is not the csh shell. If the user is already using the csh shell, including “csh” in the command will not cause any problems for the install.

There can be more than one csh (c shell) available on a UNIX system. If you use the csh (c shell) command to execute the ps_install program, it could fail depending upon the default csh. If a failure occurs during installation, issue the ./ps_install. command without specifying a shell or issue the command specifying the path to an alternate csh on the system.

4. Verify that the following files were written to the install directory (/fnsw/local/bin):

CheckSSN
HP11_maint or MR11_maint
HP11_import or MR11_import
HP11_start or MR11_start
HP11_stop or MR11_stop
HP11_val or MR11_val
HP11_cfg.sample or MR11_cfg.sample
TVDR (AIX)
libTVDR.so (Solaris)
libTVDR.sl (HP/UX)
PS_Password.exe
PSs (AIX)
LibPSs.sl (HP/UX)
LibPSs.so (Solaris)
PSs.dll (Windows)
timer.awk

NOTE **HP11_cfg.sample** and **MR11_cfg.sample** are sample configuration files. After the applicable cfg file is modified, copy or rename it to **HP11.cfg** or **MR11.cfg** to be used as the runtime configuration file.

Following installation, HP11/MR11 can be moved to another directory, if desired, by simply moving all of the above files from /fnsw/local/bin to a new directory.

4.2 Windows Platforms

This section describes the installation procedure for Windows systems only.

1. Log in as an Administrator user, to the server where you will install HP11/MR11.
2. Insert the CD and installation will begin via the Autorun program found in the root of the CD. If Autorun is turned off, use Explorer to access the root of the CD and start the install by Double clicking on PS_Install.exe or by selecting Start>Run> and input **<CD ROM drive letter>: PS_Install.exe**
3. Select **Continue** on the Application Package Installer dialog box.
4. Select the ISTK version installed on your system from the **Available Releases** frame and select **Install**.

NOTE To check your ISTK version level on a Windows Server stamp the following module. The stamp information will look similar to the output below.

Windows Server DOS prompt:

```
stamp <drive>:\fnsw\client\shobj\*SysV*
```

Example of an ISTK 3.5 module stamp:

```
\fnsw\client\shobj\ISTK_sysv.dll (NT bin):  
system 3.5.0.174(1) (lib, Tue Nov 28 06:59:51 2000)  
developer 3.5.0.0.0 (lib, Tue Nov 28 06:59:47 2000)  
SubSys: mv, Rel_type: ISTK, SCR#: 118396, mode: 100666, size: 363912
```

5. To verify the installation path, select **Yes**.

The standard installation path for HP11/MR11 is: **C:\fnsw_loc\bin**

This installation path is usually changed to a drive letter location where the IS software and ISTK software are normally installed. To specify this preferred area for the installation path, select **No**.

6. If you selected **No** in the above step, you will be asked to specify/select a new installation path. Select/create the new installation path and select **Accept**.

A standard installation path for HP11/MR11 is:

< drive letter>: \fnsw_loc\bin

NOTE Do not select an installation path like “\Program Files”. The space in the path name will not work with HP11/MR11.

7. Verify the new installation path by selecting **Yes**, or return to the **Destination Path Selection** dialog box by selecting **No**
8. Click **OK** and **Quit**, when the installation is complete.

4.3 Completing Installation on UNIX and Windows Operating Systems

Complete the following steps before the initial configuration and execution of HP11/MR11. These steps will direct first time users on how to update the system “hosts” file, verify that HP11/MR11 can communicate with FILENET ISTK Runtime, and insure that HP11/MR11 is able to logon to the FILENET system.

4.3.1 “hosts” file entry

The “hosts” file can be found in the following areas on your server:

UNIX: /etc/hosts

Windows: <drive>:\WINDOWS\system32\drivers\etc\hosts

ISTK applications require specific entries in the “hosts” file in order to correctly execute the ISTK logon call. This entry is specific to ISTK applications and must be added to the “hosts” file before running a FILENET utility. The FILENET domain name entry must appear exactly as it appears in the FILENET Application Executive. If the name appears in capital letters with underscores then you must enter the name with that same case and character composition.

To create an entry in the “hosts” file, enter:

- the TCPIP address followed by a tab
- the server name, followed by a space
- the FILENET domain name followed by another space
- the 4 part FILENET NCH server name

NOTE Enter only one line for each server. Do not add multiple lines for the same server as this could cause inconsistent results during runtime.

Examples:

```
192.48.11.12    servername domainname domainname-filenet-nch-server
```

```
183.52.10.11    SERVERNAME DOMAIN_NAME DOMAIN_NAME-filenet-nch-server
```

NOTE If you have an Image Services Server requirement for a domain name entry of a different case, add two 4 part NCH server names to the line entry for your FILENET server.

4.3.2 Environment PATH

The PATH in the user environment (usually the fnsw user) must be setup so that the ISTK shared libraries are accessed by an ISTK call and not the IS shared libraries. This is explained in the ISTK install readme file and is also covered in this document to emphasize and include the utility PATH location. The UNIX configuration of the PATH does differ from the Windows Server configuration of the PATH. It is important not to confuse the Windows configuration with the UNIX environment configuration. The PATH should not be limited to the examples below as there are other software and OS requirements found in each environment PATH that must be retained. The examples below show the ISTK vs. IS PATH hierarchy.

To configure the PATH in UNIX:

```
/fnsw/client/bin:/fnsw/local/bin:/fnsw/bin
```

To configure the PATH in Windows:

```
<drive>:\fnsw\client\bin;<drive>:\fnsw\client\shobj;<drive>:\fnsw\bin;<drive>:\fnsw\lib\shobj;
```

NOTE IS systems running eProcess require that the IS shared library path be exposed thus causing a potential conflict with the ISTK shared libraries. Should this occur, contact CSS for details on alternate ways to run HPII/MRII that will avoid this conflict.

4.3.3 ISTK file ownership and permissions on UNIX

FILENET Image Services Server Toolkit (ISTK) provides an installation readme, which outlines the UNIX environment permissions and ownership settings on the ISTK modules. Follow the instructions in the readme file, to ensure correct access of ISTK modules during application runtime. The ISTK modules can not be owned by root:system. There are two modules that are owned by root:fnusr. Refer to the ISTK installation readme file for complete instructions.

5. CONFIGURING IMAGE IMPORT FACILITY

Once the software is installed, you must logon as the System Administrator to do the following:

- set search paths

- set user security

- setup mandatory and optional settings by editing the HP11/MR11 configuration file which is initially named HP11_cfg.sample or MR11_cfg.sample and will be renamed at the time of editing to **HP11.cfg** or **MR11.cfg** for runtime use

Additionally you might need to modify the system Cache parameters. For more information, see the Cache Parameters section below.

5.1 User Environment

UNIX Operating System User only

Although the *fns* user is often used to run HP11/MR11, FILENET supports the use of a separate UNIX user for running HP11/MR11. This user must have read, write, and execute privileges for all directories, files, and programs used by HP11/MR11. The UNIX user must be a member of the groups: *fnsr*, *fnp*, *fndadmin*, *dba*, and *root*. Refer to the section entitled "Environment PATH" to setup this *newuser's* PATH correctly. The HP11/MR11 modules in the install point must also have the "sticky bit" set to provide for proper module access.

Use the UNIX `chmod` command on all HP11/MR11 runtime modules as follows:

```
cd /fns/local/bin
chmod g+s *
chmod u+s *
```

Use the UNIX `chmod` command on `wal_purge` and `wal_daemon` modules as follows:

```
cd /fns/client/bin
chmod g+s wal_purge wal_daemon
chmod u+s wal_purge wal_daemon
```

All ISTK and HP11/MR11 modules should be owned by `fns:fnsr`. When running HP11/MR11 you will login as the *newuser*. If HP11/MR11 was run by another user, check all data input and output files for permissions to ensure proper file access.

Windows and UNIX Image Services Server User

In addition, an Image Services Server user must be created that has read, write, and execute privileges for documents on FILENET Image Services Server library. The name and password of this Image Services Server user must be entered in the **HP11.cfg** or **MR11.cfg** file. This user must also have been configured with at least 3 concurrent logins. The default user for this entry is SysAdmin. To configure a special user for HP11/MR11, consult with your implementation consultant.

5.2 Cache Parameters

Cache Parameters are configured through **fn_edit** and will not take effect until an **fn_build** is done and the FILENET software has been restarted. Since cache usage varies by system, we can not specify default settings for HP11/MR11.

HP11 and MR11 differences

HP11 uses fast batch committal, which means that it must have access to a contiguous amount of Retrieval Cache (Page Cache) to operate. When using HP11 the user will not be able to specify the cache. The `page_cache` is tied to an IS family causing `page_cache` selection to only be controllable through IS configurable items. See the IS System Administrators Guide and the IS Configuration Guide for more information on `page_cache` and family settings.

MR11 uses Batch Entry Services for Batch Assembly and Page Cache for committal. The BES space used is a configurable item through the MR11.cfg file keyword `BESDomain`. See the "Batch Attributes" section in this document that describes the `BESDomain` keyword.

If there is insufficient cache available, HP11/MR11 will wait and then retry the batch a limited number of times while cache services attempts to satisfy the request. In extreme cases, this could give the appearance that HP11/MR11 has ceased operating. Should Cache continue to be unavailable HP11/MR11 will continue attempts to process each *.eob file located in the WorkingDirectory and perform its waits and retries on each *.eob file. Once the limit of retries has been reached the *.eob file to be processed will be renamed to *.err. The *.err files will contain text which indicates that the Cache is full. The user will need to wait for the optical (jukebox) to catch up or configure additional Cache if system resources will support this addition.

The following table lists default and recommended system parameter settings that most affect HP11/MR11.

CACHE PARAMETER	DEFAULT	RECOMMENDED
% Locked	80	90
% daemon	80	60
% write	90	70

If the default settings do not provide adequate performance, the test settings can be tried. However, other system requirements such as pre-fetch requirements and ad-hoc retrieval frequency should be taken into account before making any changes.

5.3 HP11.cfg/MR11.cfg

HP11.cfg/MR11.cfg is a text file and can be modified using a text editor. The sample configuration file, HP11_cfg.sample/MR11_cfg.sample, is shown below:

```

;*****
; This module contains the configuration information for the XXX.      *
;*****
; $Author: giffj $
; $Date: 10/.0/.0 .1:.3:.3 $
; $Header: II_cfg.sample,v 1.3 10/.0/.0 .1:.3:.3 giffj Exp $
; $Locker: $
; $Revision: 1.3 $
; $Source: /usr/ProfServ/lbush/II_Three/src/RCS/II_cfg.sample,v $
; $State: Exp $
;*****
; INSTALL HISTORY:
;INSTALLED BY          DATE          DESCRIPTION
;-----
;
;*****
;
;          Logon Information
;*****
LogonAttribute {
    UserName="SysAdmin" <MUST CHANGE TO YOUR ENVIRONMENT>
    Password="SysAdmin" <MUST CHANGE TO YOUR ENVIRONMENT>
;The ImpTerm and ValTerm are no longer required. User may set these to
;a naming convention supported by the IS or leave them commented out and
;the utility will default to the devices names as listed below.
;
;    ImpTerm="HP11_IMPORT" <MUST CHANGE TO MR11_IMPORT WHEN USING MR11>
;    ValTerm="HP11_VAL" <MUST CHANGE TO MR11_VAL WHEN USING MR11>
;    Domain="profserv" <MUST CHANGE TO YOUR ENVIRONMENT>
;    Organization="FileNET" <MUST CHANGE TO YOUR ENVIRONMENT>
}
;*****
;          DOCUMENT CLASS AND INDEXING INFORMATION
;*****
;The following DocClassAttributes are only samples and must be changed to valid
;Doc Classes, Indexes, and values found in your system environment.
DocClassAttribute {
    ClassName=HP11_AppTest
    ClassCode=01
    IndexName=Reference_Number
    IndexName=Soc_Sec
    IndexName=Account_Number
    IndexName=Name
}
DocClassAttribute {
    ClassCode=02

```

```

        ClassName=HP11_Test
        IndexName=BatchInfo
    }
DocClassAttribute {
    ClassName=HP11_Test
    ClassCode=03
    IndexName=BatchID
    IndexName=BatchInfo
}
;
;*****
;    BATCH INFORMATION
;*****
;
BatchAttribute {
; These first 3 items are required:
    MaxSubBatchSize=3072
;   MaxSubBatchSize is size of batch in 1K units (1024=1Mb)(default=1024)
    MaxDocPerSubBatch=999
    WorkingDirectory=/datafiles
;   BESDomain=<besserverdomainname> CAN ONLY BE USED WITH MR11
;   BESObject=<besobjectname> CAN ONLY BE USED WITH MR11
;
;
; The following are OPTIONAL paramters--defaults are indicated:
;
;   Display format for dates, also affects lengths of some filenames
;   YearFmt="%Y"
;
;   VerifyTiffFrequency=0
;   0 = No Validation
;   1 = Validation AND Correction
;   >1= Spot validation
;
;   DocReportFrequency=999
;   JournalDirectory=/parent
;   MultiDoc=1
;
;   FieldDelimiter=":" See section on "File Formats" for more
;   ItemDelimiter="," information on these two values.
;
;   IgnoreChkSum=<False=off>
;
;   Minimum for the following items is 5 seconds
;   SleepInterval=60
;   ConfirmSleepInterval=60
;
;   DelayProcess=30
;   OSNice=0 <0 is valid>
;
;   MaxErrorsAllowed=<unlimited> <default=65535>

```

```
;      ErrorDelay=<optional - default 0 = skip batch, do next one>
;
;
;      FALSE = 0, TRUE = 1
;
;      Timing=<FALSE=off>
;      CheckIndexDB=<TRUE=on>
;      CleanCache=<FALSE=off>
;      Checksumming=<FALSE=off>
;      TransactDatLineLength=<255>
;      ProgressFileFlag=<0>
;      FNProgressDirectory=<no output generated>
;      ExternalProgressDirectory=<no output generated>
;      ExternalPassDirectory=<no output generated>
;      ExternalErrorDirectory=<no output generated>
;      ExternalReportProgram=<no program launched>
;      ExternalRptDirectory=<no output generated>
;      PassCopyDirectory=<optional>
}
;***** END OF CONFIGURATION FILE *****
```

NOTES A line beginning with a semi-colon (;) is a comment line and is ignored by the program. Blank lines are not valid in the **HP11.cfg/MR11.cfg** file. If a blank line is introduced into the **HP11.cfg/MR11.cfg** file then HP11/MR11 will not run.

The above example shows directory paths in the UNIX directory format. For a Windows system, the directory path entry in the **HP11.cfg/MR11.cfg** file should follow the correct Windows directory convention.

5.4 Logon Attributes

The logon attributes should match the Image Services Server user, Terminal IDs, and standard logon information for the FILENET Image Services Server system. The separate and distinct Terminal IDs for **HP11_import/MR11_import** and **HP11_val/MR11_val** are automatically created by Security Services when HP11/MR11 is run.

5.5 Configuring Encrypted Password (optional)

The mechanism to provide values for an IS login through the Utilities is to set the values of the `UserName` and `PassWord` in the `LogonAttributes` of the `HPII.cfg/MRII.cfg` file. There are two ways to enter this information. You can enter the values for the `UserName` and `PassWord` using `ascii` text form. This means that the password for the user will be visible to anyone with read access to the `HPII.cfg/MRII.cfg` file. The second way to configure these login attributes is to use the Encrypted Password functionality. Follow the instructions below to use this functionality.

Use the `PS_Password` program to create the `.ps_passwd` file which will contain the encrypted records. The first time the `PS_Password` program is run you must provide a password for the creation of the `.ps_passwd` file as well as the password(s) for any IS Utility Application and User Name. Logged on as System Administrator, you must use this `.ps_passwd` to run the `PS_Password` program to change the `.ps_passwd` file.

Login as `fns` and run the `PS_Password` program:

UNIX:

```
cd /fns/local/bin
>PS_Password
```

Windows:

```
Open a DOS window
cd <drive>:\fns_loc\bin
PS_Password
```

The first time you run the `PS_Password` program, you will be notified that it is running in *Initialization Mode*. You will then be asked to Enter and ReEnter (validate) the password for the *Password Application Key*. (The character limit for this password is 8 characters and it is not changeable once it has been set.) The *Password Application Key* is the password that you will use to run the `PS_Password` program a second time. Subsequent runs of the program will require you to enter this password, and the *Initialization Mode* message will not be displayed.

The `PS_Password` program will then ask for the *PS Application Name*, a *Password* for the login being used and the *User Name* of that login. You must provide a valid user ID that has SysAdmin privileges within the IS domain.

Once the application name, password, and user name have been entered, the program will display the application and user name with validation that the password has been set. Another application password can be set at this time. When you have added all of the applications and user names, select the **Enter key** at the next prompt to stop the `PS_Password` program and display a list of all applications and user names that have been entered.

Example on a UNIX system:

```

/fnsw/local/bin> PS_Password
Creating PS Password File - Initialization Mode
Enter Password Application Key      :
ReEnter Password Application Key    :
Enter PS Application Name           : HP11
Application (39)'HP11'
Enter New Password for Application HP11 :
ReEnter Password for Application HP11 :
Enter User Name                     : SysAdmin
PS Program 'HP11': User=SysAdmin, Password Set
Enter PS Application Name           :
**** Current Applications Set ****
** Application (39)'HP11' [User='SysAdmin']
*****

```

Now that the **PS_Password** program has been run and the *.ps_passwd* file has been created you can configure the **UserName** and **PassWord** values in the **HP11.cfg** file.

Set the **UserName** value in the configuration file to the application name entered into the *.ps_passwd* file. In this case that would be **UserName="HP11"**.

Remove or comment out the **PassWord** value in the **LoginAttribute** section of the **HP11.cfg** file. Otherwise the HP11/MR11 application will try to use the **PassWord** parameter in the **HP11.cfg** file during start-up, it will not use the encrypted password that was entered when the **PS_Password** program was run, and HP11/MR11 will not run.

Example of the **HP11.cfg LogonAttribute** section when using the encrypted password functionality:

```

;*****
; LOGON INFORMATION
;*****
;HP11.cfg.sample
LogonAttribute {
    UserName="HP11"
;    Password="" <this parameter has been commented out>
; For a source system the Domain should be the source domain
; For any other system (target or 3rd party) the Domain
; should be the target domain
    Domain="profserv" (must change to correct domain)
    Organization="FileNet "
}

```

NOTE If you forget the the password to **ps_passwd** , you must delete and recreate the *.ps_passwd* file with the **PS_Password** program Initialization Mode.

5.6 DocClass Attributes

The **HP11.cfg/MR11.cfg** file contains a **DocClassAttribute** field, which will be used by HP11/MR11 during runtime. These Document Class Attributes must define each Document Class that is used by HP11/MR11 with each assigned a unique two digit **ClassCode**. Refer to the examples listed in the **HP11_cfg.sample/MR11_cfg.sample** file. **ClassCode=00** is reserved for future HP11/MR11 development use and must not be used in a customer environment. For each Document Class used during HP11/MR11 runtime, the user will need to list the **ClassName**, create a unique 2 digit **ClassCode**, and list the **IndexName** fields found in the specific **DocClassAttribute**.

NOTE You must define the related index fields in the same order as the values for each Index Field will be listed in the *transact.dat* file. (If your index values are not listed in the correct order, the values will be saved to incorrect fields, or an error will occur when there is an attempt to save incorrect data format values to an incorrect field.) Be sure to include all required fields for the document class as well as any others that will be provided. (All indexes need not be used unless they are required fields.) See the definition for the *transact.dat* file for more information on this attribute.

5.6.1 F_DOCFORMAT System Field

Use the System Field **F_DOCFORMAT** to add document format information to an object file before import, if the object file is not the FILENET standard TIFF file type. For information on the values for the **F_DOCFORMAT** field, see the industry standards document, "Media Types", at: <http://www.iana.org/assignments/media-types/>

Examples:

Your **HP11.cfg/MR11.cfg** file must have the following IndexName configured for your Document Class.

```
IndexName=F_DOCFORMAT
```

Your *transact.dat* file must have a value for the IndexName=F_DOCFORMAT using the correct "Media Types" document information for your document type.

An example of the object file type "plain text" is:

```
Text/plain;name="filename.txt"
```

An example of the object file type "Application pdf" is:

```
Application/pdf;name="filename.pdf"
```

The file extension must be part of the image name. The file extension for the image file must also be registered through your client Operating System and the specific Application must be loaded in order for that Application to render the image.

NOTE The Type and Subtype for the file format must be determined by the standards found in the "Media Types" document. Test this setting to insure correct display of a document after import before adding this feature to a production system.

5.6.2 F_DOCTYPE System Field

The System Field **F_DOCTYPE** must be used to change the Document Type from the default of “Image” to anything else.

The defined values are:

“IMAGE”
“TEXT”
“FORM”
“MIXED”
“SHEET”
“ANNOT”
“OTHER”

NOTE Any non-defined value (other than “IMAGE,” “TEXT”, and so on) will internally have F_DOCTYPE set to the first character of the non-defined value.

Unacceptable values to Index Services can result in the batch failing to commit.

Examples:

Your **HP11.cfg/MR11.cfg** file must have the following IndexName configured for your Document Class.

```
IndexName=F_DOCTYPE
```

Your *transact.dat* file must have a value for the IndexName=F_DOCTYPE chosen from the document type values listed above.

5.7 Batch Attributes

The first three batch attributes, **MaxSubBatchSize**, **MaxDocPerSubBatch**, and **WorkingDirectory**, are required parameters in the **HP11.cfg** file. The rest of the “keywords” are for convenience in configuring the import parameters for your environment’s requirements.

NOTE **MR11.cfg** requires the **WorkingDirectory** parameter value **MaxSubBatchSize** and **MaxDocPerSubBatch**.

The following **MaxSubBatchSize** and **MaxDocPerSubBatch** are for HP11 users ONLY:

- **MaxSubBatchSize** is the disk space required for any given sub-batch in 1 KB units (for example, 5120 is 5 MB). This value needs to be large enough to hold an entire batch if the batch will not be sub-divided, or large enough to hold an entire sub-batch. The maximum number of sub batches that can be created is 3.

HP11 will request this amount as contiguous cache from cache services. If the amount of cache requested is too large, the request will not be satisfied. Equally, setting this number too low will negatively impact performance or could force the sub-batch limit to be exceeded.

For more information on this setting, see the section on “How to compute the MaxSubBatchSize”.

- MaxDocPerSubBatch** is the maximum number of documents allowed in a sub-batch. This parameter works in conjunction with **MaxSubBatchSize**. When configuring this parameter it is important to note IS limitations in regards to Batches, Documents, and Pages, and also the fact that HP II uses Fast Batches. The IS will allow the user to put 10,000 documents in a batch with a maximum of 30,000 pages (images and document headers). One document can not have more than 1000 pages (999 images + 1 document header). Each document has the overhead of 1 page, the document header, regardless of the number of images it contains. These constraints create the HP II maximum values described below.

The maximum value for the **MaxSubBatchSize** parameter calculated in k-bytes is the maximum available contiguous cache space. **MaxDocPerSubBatch** is 1000 (999 images + 1 document header) with the default value of 1000. HP II has a limit of 1000 documents per sub-batch with no more than 9,997 images (pages) in a sub-batch. Therefore, the maximum batch size (3 sub-batches) is 3000 documents including document headers.

NOTE While **MaxSubBatchSize** and **MaxDocPerSubBatch** can be set without consideration for the other, they are tuning parameters and can affect each other. HP II works most efficiently when it does not have to create sub-batches. As a starting point, generate batches such that the batch size corresponds to approximately 1/10 of the available free space in retrieval cache. If batches are smaller than 1000 documents, set **MaxDocPerSubBatch** to 999. If batches are larger than 1000 documents, set the value to a little more than 1/2 or 1/3 of the maximum expected batch size. The limit of 9,997 images (pages) per sub-batch must not be violated. In addition, HP II will not process a batch where the last sub-batch (if created) exceeds **MaxDocPerSubBatch**.

- The table below shows the maximum number of pages, documents, or images that can be placed in a batch for each product.

	MR II	HP II
Number of pages per document	999	1000
Number of documents per batch	5000	1000 docs per sub batch with a max of 15,000 pages per batch
Number of images per batch	5000	3000 docs each contain 1 image 3 docs containing 999 images

- WorkingDirectory** is the directory where the batch status files and *.eob files will be located. The value for this parameter must be a fully qualified path. The WorkingDirectory directory name was formerly known as the “ParentDirectory”. The keyword “ParentDirectory” can be changed to WorkingDirectory when upgrading to this release in an effort to keep your configuration files current.

NOTE The **WorkingDirectory** path along with internal commands and spaces has a character limit of 64. (A path exceeding 35 characters could exceed the path limit and cause a 202,0,9 error. The HP Operating System seems to be the most sensitive to this issue.)

- **BESDomain** is found in MRII functionality only and allows for the ability to specify a different Batch Entry Service Domain.
- **BESObject** is found in MRII functionality only and allows for the ability to specify a specific Batch Entry Object found in the Batch Entry Service Domain selected.

NOTES MRII can run on a "Remote Entry Server" (RES) but HP11 cannot.

MR11 has a maximum throughput of 2 pages per second as defined by the IS supported load levels.

A batch name cannot exceed 20 characters or IS will produce an error and the batch will not be imported.

The remaining batch attributes are optional. Defaults are enforced unless the option is specifically uncommented and the value is entered or changed to a new value.

- **YearFmt** is the display format for the year portion of the date which is appended to the HP11/MR11 working files. This parameter can change the length of a file name. The default is '%Y' which displays in the 4-digit format "yyyy". A 2-digit "yy" format can be displayed by changing the default to '%y'
- **VerifyTiffFrequency=<0>** is the number of TIFF images processed between image verifications within a batch. (This process verifies that the image can be decompressed.) This value must be set lower than the total number of images in a batch or no checking will occur. A suggested value is 1/2 the batch size in images. Note that low values will negatively impact performance.

This parameter can be set to one of 3 valid values:

- 0 = No validation
- 1 = Validation and correction: every image is checked and a decompression/compression is forced, thus ensuring that the image conforms to the FILENET TIFF specification. This parameter will slow throughput considerably. This option is not supported in a Windows environment. Both utility functionality as well as IS functionality require a UNIX environment.
- >1 = Spot validation checks the TIFF header and performs a decompression on the image data.

NOTES IS TIFF validation must also be turned on.

Do not use the **VerifyTiffFrequency** setting if all files being imported are not TIFF type image files.

HP11 uses SC_CONVERT for the VerifyTiffFrequency=1 option. SC_CONVERT verifies only FILENET TIFF type 4 formats. If the TIFF type is not type 4, and if that alternate TIFF type is readable, HP11 will rewrite it to the minimum standard (TIFF 4). HP11 decompresses and recompresses each image object. Due to

overhead, performance, and image type, you should do this verification outside of HP11.

- **DocReportFrequency** is the frequency with which entries are made into the journal files. The value of this attribute represents the number of documents between reports. This can be used as a progress indicator. The journal log entry would enter the batch status, the number of documents in the batch, and the memory used to process the batch. A suggested setting is 250. The default is 999.
- **Journal Directory** is the directory where the HP11/MR11 log files will be written. If not specified, the journals directory will be automatically created in the WorkingDirectory.

Log file names:

besyymmdd is the log file for MR11 (where *yymmdd* is the date stamp)

impymmdd and *valymmdd* are the log files for HP11 (where *yymmdd* is the date stamp)

- **MultiDoc** is used to combine multiple documents into a single FILENET document. You can specify a maximum of 1,000 pages per document. The default is 0.

NOTE Do not use this option without FILENET support. Documents could become irretrievable if the option is used incorrectly.

- **FieldDelimiter=<”:>** sets the major delimiter in the *transact.dat* file. The default is colon.
- **ItemDelimiter=<”,>** sets the minor delimiter in the *transact.dat* file. The default is comma.
- **Sleep Interval** is the total number of seconds that **HP11_import/MR11_import** will wait from the initial search for available *batchname.eob* files to the next search for available *batchname.eob* files. This interval will not be affected by how long it takes HP11 to process its current work unless the current work exceeds the set **Sleep Interval**. Should the work exceed the sleep interval then another sleep interval will be appended to the end of the last sleep interval before HP11 does another search for an available *batchname.eob* file. The default and recommended setting is 60 (seconds). The maximum setting is 3600 (1 hr)
- **ConfirmSleepInterval** is the same as the Sleep Interval above except that it affects the search interval for **HP11_val/MR11_val**. This is the confirmation that the batch has been written from cache to optical disk. A suggested setting is 360 (6 minutes). The maximum setting is 36000 (10 hr). The default setting is 60 seconds.
- **DelayProcess** can be used to create a delay in processing between the physical time stamp on a *batchname.eob* and when **HP11_import/MR11_import** recognizes the *batchname.eob* file for processing. Setting this keyword to 300 will cause a 5-minute delay between the actual creation of the *batchname.eob* file and when **HP11_import/MR11_import** will read it for processing. (Commonly referred to as the *batchname.eob* age.) Normally, this should be set to 0 (seconds). The maximum setting is 3600 (1 hr). The default is 30 (seconds).

- **OSNice=<0>** allows HP11/MR11 to “sleep” for the indicated number of seconds, between each batch. This slows down HP11/MR11 and allows other processes to get processing time. This parameter is useful when HP11/MR11 is run in normal production time. The default is zero seconds (0) and the maximum is 21600 seconds (6 hours).

Example: If the OSNice value is set to 300, then it will take 5 minutes to complete the processing of one eob file. First the contents designated in an eob file will be imported and then there will be a 5-minute wait. After the wait the eob import that occurred will be confirmed. When the confirmation is done another eob file will be imported.

- **MaxErrorsAllowed=<unlimited>** sets the number of errors that are allowed within the period specified by **ErrorDelay**. If this number is exceeded, the program will stop. The default is 65535.

Example: A setting equal to 1 will allow 2 eob files to error in the time allotted by the **ErrorDelay**. A second eob error within the specified time period will stop **HP11_import**.

NOTE If the limit is reached, **HP11_import** will stop but **HP11_val** will continue to run. You must run **HP11_stop** to stop **HP11_val**.

- **ErrorDelay=<optional - default 5 seconds = skip batch, do next one>** sets the window of time (in seconds) for which **MaxErrorsAllowed** is counted. The minimum is 0 seconds. A setting of zero will cause **MaxErrorsAllowed** to never be reached and thus allow **HP11_import** to continue no matter how many batches fail with errors.
- **Timing=<FALSE=off>** provides for log output used in performance analysis. The default value is set to FALSE. Setting the value of this variable to TRUE will cause additional information to be logged. The timer.awk program can be used with this added log information for performance analysis.

Usage for UNIX Operating Systems and Windows Operating Systems with UNIX Tools for DOS loaded: (Some newer UNIX systems require the use of nawk instead of awk.)

```
awk -f timer.awk journals/imp20020328
```

```
nawk -f timer.awk journals/imp20040129
```

NOTES Timing=False will be overridden and internally set to True by debug when turned on using the HP11_start -D option.

Solaris 5.8 requires the use of nawk, as awk will give poor results. The nawk program can also be used instead of awk on other UNIX Operating Systems when present.

For this functionality to work within a Windows Operating System environment the “awk” executable must be available. If you are unfamiliar with this executable contact your FILENET System Consultant for information.

- **CheckIndexDB=<TRUE=on>** allows HP11/MR11 to be used on a conversion system by setting this to FALSE. **HP11_val/MR11_val** normally will validate that the document is on optical and that the Index information exists.

- **CleanCache** allows HP11/MR11 to delete objects that are placed in cache by HP11/MR11 for verification purposes. The delete will only be completed on documents that are not locked. Default is FALSE=off

NOTE - HP11 ONLY: When the user is running high volumes they will need to have the **CleanCache** option set to the "TRUE" value. This will keep page cache space available for HP11 batches to process.

The **CleanCache** option should be set to FALSE under the following conditions:

1. On a system with an OSAR – if Fast Batch Breakup (setting in `fn_edit`) is enabled. Use this if you require documents to stay in cache for immediate accessibility. The documents will remain in cache until IS removes them. To complete this capability Fast Batch Breakup (FBB) must be enabled in IS. If you are not familiar with this setting, contact your TC or FILENET Customer Service and Support for assistance.
2. On an OSARless system – set this option to FALSE or documents could be deleted from cache and therefore lost.

- **Checksumming=<FALSE=off>** turns image checksumming on or off. Usually checksumming is off. Its purpose is as a diagnostic aid when the FILENET Image Services Server system is set to validate checksums. Validate checksums must be turned on for this variable to take effect. Output will be created for Checksumming only if there is an error and the output will be written to the IS log files and not to the HP11/MR11 output files.
- **TransactDatLineLength=<Default 255>** sets the maximum line length in the *transact.dat* file. When increased beyond the 255 default, this figure must be in 1024 increments and cannot exceed available memory on the system. Most operating systems limit the alloc for this buffer to 4mb. (4mb line could be 1,073,741,824 chars long)

This keyword setting (TransactDatLineLength) only affects each actual line length in the *transact.dat* file that HP11 must read and parse out in order to do the import. If you set TransactDatLineLength to 4mb, then whenever HP11 processes a line in a *transact.dat* file, it will always allocate 4mb even if the line is very short. HP11 processes one line at a time and will reuse the 4mb for each line processed.

- **ExternalReportProgram** is the name of a user-defined program that is executed every time **HP11_import/MR11_import** successfully creates a *batchname.pass* file. If an entry exists in the **HP11.cfg/MR11.cfg** file, this program will be initiated. To suppress this action, comment out this line. The syntax to invoke the external program is *programname batchname.rpt batchname*.
- **ProgressFileFlag** is used to indicate whether *batchname.pro* files are desired. If the attribute is true (1), the *batchname.pro* files will be created in the WorkingDirectory. If the flag is false (0), the *batchname.pro* files will not be created. The purpose of *batchname.pro* files is to provide additional indicators to external programs of the progress within HP11/MR11. The *batchname.pro* file is created in the directory specified (**FNProgressDirectory** and **ExternalProgressDirectory**) and is deleted when the batch has been confirmed.
 - **FNProgressDirectory** is the directory where a copy of the *batchname.pro* files will be written (usually a local directory).

- **ExternalProgressDirectory** is the directory where a copy of the *batchname.pro* files will be written (usually an externally mounted directory visible to third-party programs).
- **ExternalPassDirectory** is the directory where a copy of the *batchname.pass* and *batchname.confirm* files will be written (usually an externally mounted directory visible to third-party programs).
- **ExternalErrorDirectory** is a directory where a copy of the *batchname.err* files will be written (usually an externally mounted directory visible to third-party programs).
- **ExternalRptDirectory** is the directory where a copy of the *batchname.rpt* files will be written (usually an externally mounted directory visible to third-party programs).
- **PassCopyDirectory** is a directory where a copy of the *batchname.pass* files will be written (usually a local directory).

5.8 Directories

All directory attributes are configurable. The defaults shown above are for representation purposes only. The **ParentDirectory** value is mandatory. All other directory values are optional. The **journals** directory is mandatory but will be automatically created under the **ParentDirectory** if a value is not specified. To inhibit the creation of files in a directory, comment out the corresponding line in **HP11.cfg/MR11_cfg**. If an entry exists for these directories, HP11/MR11 will create the directory if it does not exist.

6. CONFIGURING UTILITIES TO RUN FROM A 3RD PARTY SERVER

The FILENET Image Services Server Utilities (HP11/MR11) can be run remotely from a 3rd Party Server. Use of the 3rd Party Server environment can reduce memory, CPU, and Disk I/O overhead from the Image Services Server. However it can increase network traffic.

6.1 Requirements

The system used to run any utility must be on the same network as the Image Services Server it will be communicating with. The system will require the installation of the correct version of ISTK and the correct compile of the utility for the Operating System/ISTK environment. Follow the ISTK installation readme instructions for the Operating System where you will be installing ISTK. Follow the utility installation instructions in the previous sections for the Operating System on which you will be installing the utility.

6.2 Configuration

Configure the utility as you would when actually running on the IS server to ensure that permissions to access all runtime file repositories are correct. Follow the standard Configuration instructions in the section above. Add the hosts file entries to facilitate proper communication with the target IS server.

6.3 Runtime

The utility does not require any special commands during runtime because of the 3rd party environment. Follow the section below regarding Operation.

7. HP11/MR11 OPERATION

HP11/MR11 takes documents, usually in the form of TIFF images, from an external process and commits them to optical disk. This is performed as a batch process where documents are written to sub-directories under the WorkingDirectory along with a file (*transact.dat*) that provides index information. To signal a batch is ready for import, a *batchname.eob* file is then written to the WorkingDirectory. The *batchname.eob* file name is arbitrary although the *.eob* extension is necessary for HP11/MR11 to identify the file and process the work.

NOTE The *transact.dat* file used in HP11 operations can contain different document classes per batch line item. All document classes used must belong to the same family. This functionality differs from the MR11 utility as it ONLY allows for one document class in a *transact.dat* file.

On UNIX systems, start HP11/MR11 with **HP11_start/MR11_start**, and stop them with **HP11_stop/MR11_stop**. You can invoke the programs manually from a terminal or automatically on a scheduled basis in */etc/crontab*.

On Windows systems, start and stop HP11/MR11 from a command line with **HP11_start.exe/MR11_start.exe** and **HP11_stop.exe/MR11_stop.exe**, or set the application as a Windows service.

7.1 Command line startup

HP11/MR11 uses **HP11_start/MR11_start** in the UNIX environment and **HP11_start.exe/MR11_start.exe** in the Windows environment to start. Stopping HP11/MR11 uses **HP11_stop/MR11_stop** in the UNIX environment and **HP11_stop.exe/MR11_stop.exe** in the Windows environment. These programs can be initiated through the same command line options available on the UNIX system as they can in the Windows environment. Using the start and stop programs at a command prompt offers more options and they must always be initiated in the HOME (install point) directory directly. When started from a command prompt, **HP11_start/MR11_start** initiates two background programs, **HP11_import/MR11_import**, which asynchronously commits the batches to optical disk, and **HP11_val/MR11_val**, which validates that committal has occurred. Both of these programs use standard file I/O and FILENET ISTK calls to perform their functions. There are no special interfaces or engineering entry points used.

See the “Runtime” section below on specific command line options.

7.2 Windows NT/2000 Service setup

You can start and stop HP11/MR11 using Window’s Services when running HP11/MR11 on a Windows platform. This feature can be implemented once HP11/MR11 has been installed.

1. In a DOS prompt, change directory to the HP11/MR11 installation location:

Default installation location:

<drive>:\fnsw_locbin

2. Enter the HP11/MR11 command to create the Windows Service: (This command will NOT start HP11/MR11. It will only create the HP11/MR11 Service.)

```
HPII_start -service  
or  
MRII_start -service
```

The display will report back after the Service has been installed:

```
"HPII Service successfully installed"  
or  
"MRII Service successfully installed"
```

3. Open the Control Panel / Administrative Tools / Services icon:

In the Services window select the "FILENET HPII Service" or "FILENET MRII Service" with a right mouse click and then select the "Properties" menu item

Select the "Log On" tab

Under "Log on as:" select the "This account:" radio button

Go to step #4

4. In the "This Account" text box enter the "fnsw" user or a user in the fnusr group specifically created for running HPII/MRII. Enter and confirm the password for the account and click "OK".
5. Once the login properties have been set and all configuration items have been completed, start HPII/MRII by selecting the "FILENET HPII Service" or "FILENET MRII Service" in the server window and then selecting start.

NOTES Configure the Service Logon Account before the Service is used to start HPII/MRII or the startup will fail. The notice of a failure could take more than a minute to report back to the user.

When using the Service Applet to stop HPII/MRII the applet might receive premature notice that HPII/MRII has stopped. However, some processes could still be shutting down. Immediate restart through the applet can cause corrupted multiple processes to be run. Check the Windows Task Manager to ensure all HPII/MRII processes have stopped.

6. The following procedure will remove the FILENET HPII or MRII Service:

Check the Service through the Service applet window and ensure the service has been stopped.

open a DOS window

```
cd <drive>:\fnsw_loc\bin
```

```
HPII_start -noservice
```

```
or
```

```
MRII_start -noservice
```

7.3 Processes

The **HPII_import/MRII_import** program scans the WorkingDirectory for *batchname.eob* files. When a *batchname.eob* file is located, the related batch directory will be processed. The index information file (*transact.dat*), found in the batch directory, will be parsed, and is used to control the generation of documents from the other files contained in the batch directory.

HPII_import/MRII_import will process the batch according to the following rules:

1. The program will begin parsing the *transact.dat* file when it detects the existence of the corresponding *batchname.eob* file in the WorkingDirectory. The file structure for *transact.dat* can be found in Section 6 of this manual.
2. If any error occurs in processing a batch, the *batchname.eob* file is renamed as *batchname.err*, and error information is appended to the file.
3. If one or more specified files (usually TIFF) associated with a document cannot be found, the program will skip committing that batch and log the entry as an error in a *batchname.err* file where the filename is the same name as the *batchname.eob* file.
4. The program will not check for duplicate index information.
5. When the batch processing is completed, all documents within the batch will be sent to cache for committal to optical disk.
6. If the request for committal was successful (for example, no errors occurred in processing the batch), the *batchname.lck* file is deleted from the WorkingDirectory. A *batchname.pass* and *batchname.rpt* is created in the WorkDirectory. In addition, a *batchname.conf* is created in the Home directory.
7. If related optional Batch Attributes are configured, then HPII/MRII will copy the *batchname.pass*, *batchname.err*, and *batchname.rpt* files to the directories specified in **HPII.cfg/MRII.cfg**.
8. An entry is made to the *yymmdd.eod* file in the Home directory for each batch processed.

Sub-Batches (HPII ONLY)

If a batch is submitted to HPII that exceeds the value set by **MaxDocPerSubBatch**, HPII breaks the batch into sub-batches to meet this limit, and processes them. Information on sub-batches is written to the *batchname.pass* file.

Validation

HPII_val/MRII_val scans the Home directory (location of **HPII.cfg**) looking for *batchname.conf* files that indicate that **HPII_import/MRII_import** has submitted a batch for committal. If the batch has been committed, **HPII_val/MRII_val** will create *batchname.confirm* in the WorkingDirectory (formerly known as the ParentDirectory) where the original *batchname.eob* files were located.

File and Directory Maintenance

File and directory maintenance occurs on the basis of the **HPII_maint/MRII_maint** command. This command is manually initiated, or can be started by entries in */etc/crontab* or the Windows Task Scheduler.

HPII_maint/MRII_maint looks for *batchname.confirm* files older than the specified number of days, and deletes corresponding files from the WorkingDirectory as well as the corresponding batch's directories. If optional directories are used, **HPII_maint/MRII_maint** ignores them.

Shutdown

If you must shut down HP11/MR11 for system maintenance or administrative purposes, you can invoke a stop command as described below. Whenever a stop is invoked, HP11/MR11 will attempt to shut down gracefully by completing the processing of any active batches prior to shutdown.

On UNIX systems: **HP11_stop/MR11_stop** is initiated from the command line. If system unavailability or other major problems interrupt the import process, you can restart **HP11_start/MR11_start** manually, or it will be re-started by */etc/crontab* at the next scheduled time (if defined).

On Windows systems: The shutdown is available through the HP11/MR11 program window with shutdown options available for regular shutdown as well as in the event of system unavailability or other major problems interrupting the import process. Refer to Appendix A for complete details. If the utility was started from the command line, it must be stopped from the command line, using the **HP11_stop/MR11_stop** program.

Once restarted, **HP11_import/MR11_import** will scan the WorkingDirectory for any *batchname.lck* files, and resume processing the incomplete batches. If there are *batchname.err* files, these can be manually renamed to *batchname.eob* files, and HP11/MR11 will attempt to re-process them. (*Batchname.lck* files will not exist if HP11/MR11 is shutdown cleanly.)

7.4 Runtime

HP11_import/MR11_import and **HP11_val/MR11_val** can be invoked with the following parameters if you include the specific flags listed below with the **HP11_start/MR11_start** command. The default mode specifies no parameters (Verbose Mode). This mode causes **HP11_import/MR11_import** and **HP11_val/MR11_val** to log the status during the initialization and configuration time, to the *nohup* files. Once the programs are scanning directories, they switch to a quiet mode.

- h Help
- h *homedir* Location of the HP11.cfg/MR11.cfg file and PLSLF license key if other than the directory HP11/MR11 program files are located
- q Quiet mode This will remove all initial output to the *nohup* files.
- v Display Version

The flags below should only be used when directed by FILENET CSS and will enable the internal debugging code.

- S Spy mode
- D Debug mode (will override Timing=False keyword value)
- M Debug Memory mode
- C <mins> Clock Timeout

NOTE When starting either application using its 'start' program, by default it will create a file in the current directory called '<app>.out'. This file will be updated with information throughout the start process. The applications 'stop' program will also append information to this file when you use it to shutdown the application.

The 'start' program also establishes the location of the 'nohup' (program's stdout data) files. To determine the 'nohup' file location, it examines the environment variables, in the following order:

"NOHUP_DIR", "ISTK_LOG_DIR", "TEMP", "TMP", "TMPDIR",

If none of these are found, then the default location is **'/tmp'**.

NOTE This is important on systems where the /tmp folder is also the 'dump' location, or is cleaned out at boot time. By examining the environment values in this manner, we allow the customer to 'relocate' the nohup files.

8. PROGRAMS

The following section gives a more detailed description of the operation of the various HP11/MR11 programs.

HP11_start / MR11_start - starts **HP11_import/MR11_import** and **HP11_val/MR11_val**. It does this using the *NOHUP* option.

HP11_stop / MR11_stop - shuts down HP11/MR11 gracefully. When executed, no new batches will be processed, but current batches will be completed before shutdown.

HP11_stop /MR11_stop must be stopped using the *-h homedir* flag if **HP11_start/MR11_start** was started with the *-h homedir* flag. Also, **HP11_stop/MR11_stop** will cause the *yymmdd.eod* file in the Home directory to be copied to the WorkingDirectory as *yymmddfn.eod*

HP11_import / MR11_import - When **HP11_start/MR11_start** is initially invoked, it will scan the **WorkingDirectory** looking for *batchname.lck* files (incomplete batches). This scan is performed *only* at start-up. The program then scans the **WorkingDirectory** for *batchname.eob* files. Subsequent to the initial scan, the program wakes up and scans for *batchname.eob* files at a default time of every 3600 seconds (or the amount of sleep time specified in the **SleepInterval** field of the **HP11.cfg/MR11.cfg** file).

When a *batchname.eob* file is detected, the **HP11_import/MR11_import** program is invoked to process the batch. The *batchname.eob* file is renamed as a *batchname.lck* file, indicating that the batch is being processed. When the batch is successfully processed, the *batchname.lck* is replaced with *batchname.pass* and *batchname.rpt* files.

If the system malfunctions while **HP11_import/MR11_import** is in progress, the *batchname.lck* file will be retained. Upon system start-up, those batches with corresponding *batchname.lck* files will be reprocessed.

If an error occurs, the *batchname.lck* file is renamed as *batchname.err*

HP11_val / MR11_val - When **HP11_start/MR11_start** is initially invoked, it will start **HP11_val/MR11_val** looking for *batchname.conf* files in the **Home** directory (directory where **HP11.cfg/MR11.cfg** file is located). This file indicates that **HP11_import/MR11_import** has submitted a batch for committal. If the batch was committed successfully, **HP11_val/MR11_val** will be able to use the information found in the *batchname.conf* file to confirm the physical existence of the document and then write the *batchname.confirm* file in the WorkingDirectory (formerly known as the ParentDirectory). Subsequent to the initial scan, the program wakes up and scans for more *batchname.conf* files at a default time of every 3600 seconds (or the amount of sleep time specified in the **SleepInterval** field of the **HP11.cfg/MR11.cfg** file).

HP11_maint / MR11_maint – *directoryname #days sleeptime*

Directoryname specifies the directory that **HPII_maint/MRII_maint** will process.

#days is the number of days before the current time. For example, if the parameter is set to 3, **HPII_maint/MRII_maint** will process files older than 3 days **from the current time**. This parameter is optional and defaults to 7 days. If the parameter is set to 0, **HPII_maint/MRII_maint** will process all files found in the target directory without taking the file date into consideration.

Sleeptime specifies the number of minutes that **HPII_maint/MRII_maint** waits before reprocessing *directoryname*. This parameter is optional and defaults to 5 minutes. If this parameter is used, a value for *#days* must be set. Note that the current time is reset every time that **HPII_maint/MRII_maint** runs. This value can be set to zero (0) which will cause the maint program to terminate once it has deleted all available files and directories found in the target directory.

HPII_maint/MRII_maint will look for any *batchname.confirm* files in the specified directory with a date earlier than the number of days specified. The following will then be deleted:

- Any corresponding *batchname.pass*, *batchname.rpt*, *batchname.lck*, *batchname.err*, *batchname.pro*, *batchname.conf* files.
- Any *batchname.confirm* files within the specified directory. (Files in sub-directories will not be affected.)
- The batch directory as well as all the contents of the batch directory indicated in the *batchname.confirm* file. (This will delete the image files and the *transact.dat* file.)

In a UNIX environment **HPII_maint/MRII_maint** has been designed to run from a cron job. When using a command line, stop the program with a Ctrl-C, if it's not a background job. If the program is running in the background, use `kill -term` or `kill -9` to stop it. A `kill -9` will immediately terminate the program no matter what it is doing. A `kill -term` will be captured by the program and when **HPII_maint/MRII_maint** has completed the work available it will stop. If there are thousands of *.confirm files available then **HPII_maint/MRII_maint** will not stop until the work created by all available *.confirm files has been completed.

9. INTERFACE REQUIREMENTS

The purpose of this section is to specify the interface architecture and standards necessary to support the image import process between an Image Capture System and the image import operation.

The Image Capture System must be able to generate image and transaction data in accordance with the following directory structure and associated conventions. If the physical interface between the systems is LAN-based, the directory structure can either be on shared magnetic drives, or copied to the image import server. If tape is used, the Image Capture System must produce a UNIX TAR tape for UNIX systems, or standard Windows backup tape for Windows systems. These tapes must be copied to magnetic disk space available to the image import server before HP/II/MR/II can access the data.

9.1 Directory Structure

The directory and file structure used for import can vary depending on the site requirements. However, it is important to note the following:

1. The *batchdirectory* containing the images and the *transact.dat* file must be present before the *.eob files are exposed to HP/II/MR/II.
2. If a *sequence* directory structure is used as described below you must stop HP/II/MR/II and edit the **WorkingDirectory** path in the **HP/II.cfg/MR/II.cfg** file with each new sequence introduced into the system. If your site doesn't require a sequence structure, we highly recommended that a common directory be used for the *.eob files so that HP/II/MR/II will not need to be stopped to adjust for a new *.eob location.

Common *.eob directory structure:

/parent

batchname.eob (used by HP/II/MR/II to begin processing batches)
batchname.lck (will be removed by HP/II/MR/II once a batch is processed)
batchname.rpt (produced when a batch first begins processing)
batchname.err (produced when a batch cannot be processed)
batchname.pass (produced when a batch is successfully processed)
batchname.confirm (produced when validation of images being written out is completed)
batchname.pro (optional file configured in **HP/II.cfg/MR/II.cfg**)
yyyymmddfn.eod (contains batch by batch image count of successful and failed batches)

/batchdirectory(1)

individual TIFF image files
transact.dat - transaction file

/batchdirectory(n)

individual TIFF image files
transact.dat - transaction file

/journal

impyyyymmdd for HP/II or *besyyyymmdd* for MR/II
valyyyymmdd

Sequence *.eob directory structure:

For LAN-connected on-line import, there should be three types of sub-directories to the root directory depicted in the following diagram: *parent*, *sequence*, and *batchdirectory*. The WorkingDirectory should be named in some meaningful way. The *sequence* directory is used for tracking purposes; we suggest batch creation date in *yyyymmdd* format or by tape sequence number. The *sequence* directory contains one *batchname.eob* file for each batch to be processed. Each *batchdirectory* contains the input batch data (images and the *transact.dat* file) for each batch.

Files shown underlined are created by HP11/MR11 during processing. On Windows systems, the same structure applies, although the UNIX “/” will be a “\”.

/parent

yyyymmddf.n.eod (contains batch by batch image count of successful and failed batches)

/sequence(1)(yyyymmdd)

batchname.eob (used by HP11/MR11 to begin processing batches)

batchname.lck (will be removed by HP11/MR11 once a batch is processed)

batchname.rpt (produced when a batch first begins processing)

batchname.err (produced when a batch cannot be processed)

batchname.pass (produced when a batch is successfully processed)

batchname.confirm (produced when validation of images being written out is completed)

batchname.pro (optional file configured in **HP11.cfg/MR11.cfg**)

/batchdirectory(1)

individual TIFF image files

transact.dat - transaction file

/batchdirectory(n)

individual TIFF image files

transact.dat - transaction file

/journal

impyyyyymmdd for HP11 or besyyyymmdd for MR11

valyyyymmdd

HP11/MR11 cannot read directly from tape. In the case of tape input, there must be two types of sub-directories to the root directory created with the tape copy for HP11/MR11 to read from: *batchdirectory* and *sequencedirectory*.

The first directory structure and its contents to be created must be the *batchdirectory*. This *batchdirectory* contains the input batch data (images and the *transact.dat* file) for each batch. The second directory structure along with its contents to be created is the *sequence* directory. The *sequencedirectory* contains one *batchname.eob* file for each batch to be processed. All *batchdirectory* contents must be in place before the corresponding *batchname.eob* file is in place.

Reminder: When using a *sequencedirectory* for *.**eob** files you must stop HP11/MR11 to change the “**ParentDirectory**” (also known as the “WorkingDirectory”) path in the **HP11.cfg/MR11.cfg** file for each sequence copied from tape. The user will need to set the **ParentDirectory** path value to the appropriate *sequencedirectory* name to be processed. This corrected path change is needed as new tapes are copied to the system.

/parent

yyyymmddf.n.eod (contains batch by batch image count of successful and failed batches)

/sequence(1)(yyyymmdd)
batchname.eob (used by HP11/MR11 to begin processing batches)
batchname.lck (will be removed by HP11/MR11 once a batch is processed)
batchname.rpt (produced when a batch first begins processing)
batchname.err (produced when a batch cannot be processed)
batchname.pass (produced when a batch is successfully processed)
batchname.confirm (produced when validation of images being written out is completed)
batchname.pro (optional file configured in **HP11.cfg/MR11.cfg**)
/batchdirectory(1)
 individual TIFF image files
transact.dat - transaction file
/batchdirectory(n)
 individual TIFF image files
transact.dat - transaction file
/journal
impyyyyymmdd for HP11 or *besyyyymmdd* for MR11
valyyyymmdd

9.2 File Formats

The file formats, as well as a description of their functions, is listed below. All text files are UNIX format (terminated with Line Feed only) and assume no white space at the end of each line.

transact.dat

This text file contains one row (line) of information for each document within a batch. This line can wrap within the text editor used but must begin with a "class code". The line length is configurable but any changes made in the **HP11.cfg/MR11.cfg** regarding the line length will apply to all *transact.dat* files. If the line length exceeds the maximum configurable (255) for the line size value (TransactDatLineLength) you must then use the file list option in the format example below.

A document belongs to an Image Services Document Class, has index information associated with it, and can have more than one image per document. Each image represents a single page within a document. Documents within a batch can be committed to different document classes, but any one document (row/line in the *transact.dat* file) can only be committed to one document class. If a batch contains documents for more than one document class, these doc_classes must map to one—and only one—optical disk surface.

This multiple Document Class functionality differs in regards to MR11. MR11 can only import to one document class per batch. The format of the transact.dat file is:

Using a list of file names:

**ClassCode:FILENETIndexInfo1,FILENETIndexInfo2:Document
Data:ImageFileName,ImageFileName,ImageFileName**

Using an image list file:

**ClassCode:FILENETIndexInfo1,FILENETIndexInfo2:Document
Data:+ImageListFile**

The sample above uses the DEFAULT Field and Item Delimiters. These delimiters can be set to customized delimiters in the **HP11.cfg/MR11.cfg** file.

Defaults found in the **HP11.cfg/MR11.cfg** file

FieldDelimiter=","

ItemDelimiter=","

ClassCode is a 2-character document class code that is mapped to a defined FILENET Document Class in the **HP11.cfg/MR11.cfg** file.

FILENET Index information (optional if no required index values) The format of the FILENET Index information is: IndexValue1, IndexValue2,....., IndexValue*n*, where each value is represented as a string literal separated by a configurable item delimiter (the default is comma). The import process will make the necessary conversion to the corresponding FILENET index data type.

NOTE The string literal for dates must match the date mask definition for the corresponding FILENET index. Numeric values must not contain non-numeric symbols.

The maximum number of index fields is determined by the **DocClass** definition in the **HP11.cfg/MR11.cfg** file. **IndexValues** are mapped (by positional) to the FILENET index field in the same order as they are defined in the **HP11.cfg/MR11.cfg** file. The **HP11.cfg/MR11.cfg** file can define a subset of FILENET index fields for a given document class. If an index field is defined as required in the FILENET system, a valid value must be supplied.

Empty fields are permitted thus: " ," (where the item delimiter is set to a comma).

NOTE HP11 Support of Asian Double Byte Fonts

Double byte fonts (or a single character composed of two keystrokes) are a type of font found in Chinese, Japanese, and Korean languages. Single byte fonts are used for all other languages and only refer to a basic set of 256 characters.

HP11 can support 'double-byte' characters in the input data lines of the *transact.dat* file for the Index field information.

We use the hex, octal, or decimal values for each of the two bytes required for each.

The syntax is simply '\0xyy\0zzz' where yy & zz = the hex values for the two bytes of the character.

The values can be provided in "octal" by simply specifying a leading 0 instead of 0x, or in "decimal" by specifying just the number, with no leading characters.

DocumentData is a free format text field. HP11/MR11 does not parse it.

The field content is written into the *batchname.rpt* file for the corresponding document. This field is typically used to provide a copy or subset of the index information associated with a document for use by other programs once a batch has been committed.

If the **DocumentData** field is not used, 2 colons (“:”) are used to separate the **IndexValues** from the image filenames.

Image File Names comprise the filenames of the items to be imported (possibly including their path location). Each filename is separated by the item delimiter (the default is a comma). The order of the filenames defines the page order for the document. Image files are single-image TIFF files in FILENET TIFF format.

Example of a line in a *transact.dat* file:

```
02:Smith,608-68-0600,45000::Doccover.tif,Docpage2.tif, nextpage3.tif,
Docpage4.tif
```

A **Listfile** could also represent image file names. The **Listfile** is identified by a plus sign (+) before the filename. The **Listfile** contains the filenames (one per line) of the image files that comprise the document.

An example of a row in a *transact.dat* file that uses this feature is:

```
01:Test01,1245::+../common/header.lst,page1.tif,page2.tif,+../common/trailer.lst
```

This example uses the UNIX directory structure format and Windows users should conform to the Windows format. However, HP/II/MR/II will automatically switch the directory delimiter to the correct format on the target OS. That is, on Windows a “/” will automatically be translated to a “\”.

The files *header.lst* and *trailer.lst* are files found in *../common*, and contain image file names. The names found in those files are also expected to relate to the batch directory.

batchname.eob

This file indicates that the batch is ready to be processed. The content of the *.eob* file, points to the path where the images and *transact.dat* reside. It also provides a total count of documents and a total count of pages within the batch. The format of the *batchname.eob* file is:

PathName Doc_count Page_count, where:

PathName points to the directory of the batch relative to the WorkingDirectory (It should not include the directory where the *batchname.eob* file is located).

Doc_count is the total count of documents within the batch.

Page_count is the total count of the pages within the batch.

NOTE The *.*eob* file name is limited to 20 characters.

Example for LAN 19960729/01 900 3600:
19960729 is the *sequence* directory name

01 is the **batchdirectory** name
 900 is the number of documents in the batch
 3600 is the number of pages in the batch

Example for TAPE 1996072901 900 3600:
 1996072901 is the **batchdirectory** name
 900 is the number of documents in the batch
 3600 is the number of pages in the batch

batchname.confirm This file indicates that the batch has been successfully committed to optical disk. The file will first be created as *batchname.conf*. Once the confirmation of the batch migration to optical disk or page cache has been confirmed, the file will be renamed to *batchname.confirm*

batchname.err This has the same format as the *batchname.eob* file, plus additional lines that contain error codes and messages for failed batches.

batchname.pass This text file contains information on batches successfully written to cache. **HPII_import/MRII_import** will create a file for each batch successfully written to cache with the following format:
Doc_count Page_count Doc_ID range Start_Time End_Time

batchname.pro This empty file is optional. The **HPII.cfg/MRII.cfg** contains a flag to turn it on or off. Its purpose is to allow external programs to be aware that a batch is being processed.

batchname.rpt This report file gives information on batch contents. This file is created by **HPII_import/MRII_import**, however, its contents will not be used until the creation of the corresponding *batchname.confirm* file. (As the *batchname.confirm* file indicates successful migration to optical disc.) The format of this file is shown below.

The first line gives the three-part NCH name for the destination Image Services. Subsequent lines give the **Document Class Name**, **DocumentID**, and **Document Data** (from the *transact.dat* file) for each document created in a batch:

```
DefaultIMS:jose:FileNet
TestDocClass:104157: Document Data for 104157
TestDocClass:104158:
TestDocClass:104160: Document Data for 104160
```

yymmddfneod This is the FILENET end-of-day file. FILENET will create this report with the following format:
 (One line per batch)

	Send	Commit/Process
Batchname status	#docs # pgs	#docs #pgs
Batchname status	#docs # pgs	#docs #pgs
Batchname status	#docs # pgs	#docs #pgs
Batchname status	#docs # pgs	#docs #pgs

10. TROUBLESHOOTING

10.1 Utility / ISTK version relationship

The Utility build version selected will depend on the ISTK version in use on the IS Server. The Utilities have been compiled to run with ISTK 3.6 as well as with ISTK 4.0. The installation CD will detect the version of ISTK and install the correct compilation of the selected Utility when installing on a UNIX system. When installing on a Windows system the user must know the ISTK version and select the correct Utility compilation when asked during the setup procedure. If the user needs to download a version of a Utility off the CSS download site they must verify the ISTK version by running a stamp command on an ISTK module in the ISTK install directory.

Examples:

UNIX system:

```
cd /fnsf/client/shobj
stamp *SysV*
```

NT system:

```
cd <drive letter>:\fnsf\client\shobj
stamp *SysV*
```

10.2 Error Logging Files

Errors that can occur during the operation of an IS Utility Application can be reported in several places, depending on the error type. When an Image Services related error is encountered, the FILENET error tuple and error text, if available, will be reported. If a Utility or File System error is encountered, a Utility error code will be reported. The Utility error codes are generally related to an incorrect configuration, bad input file format, or a problem with the file system. File system errors (Code 1024) will almost always be the responsibility of the System Administrator to resolve. When running HP11 and MR11, a *batchname.err* file will be created in the WorkingDirectory if problems are encountered.

The following is an outline of error types and locations related to all utilities:

1. Nohup logs – all standard out / standard error messages (system errors, preamble and postamble logging from the utility, output generally captured with standard out that would be suppressed when using `-q` for quiet mode, and so on.)
This applies to HP11 and MR11, on UNIX ONLY.

Default UNIX - /tmp/{product}.YYYYMMDD

Default Windows – Windows Event logs / Applications and System

2. IS error logs – all IS related messages and errors (OSAR, database, security, and so on.)
This applies to ALL utilities.

Default UNIX - /fnsf/local/tmp/logs or /fnsf/local/logs

Default Windows - \fnsf_loc\logs\elogs

3. ISTK error logs – any internal ISTK error (an IS error log specifically for the ISTK environment.) This applies to ALL utilities.

Default UNIX - `/fnsw/client/logs`

Default NT - `<drive>:\fnsw\client\logs\ISTKYYYYMMDD`

- Utility error logs – any batch progress and general error messages are reported in the journal files

Journal files will be entitled as follows:

HP11 = `impyyyyymmdd` and `valyyyymmdd`

MR11 = `besyyyymmdd` and `valyyyymmdd`

Journal log files will be found in the journals directory. The default location of the journals directory can be found under the WorkingDirectory. (Your location could vary depending on the path set for the Journals keyword in the **HP11.cfg** or **MR11.cfg** file)

Default UNIX - `/fnsw/local/bin/journals`

Default NT - `\fnsw_loc\bin\journals`

- HP11 and MR11 failure to process an `eob` file, when the `*.eob` file has been renamed to `*.err` - This `*.err` file will contain the original line entry contained in the `*.eob` file as well as specific error output information relating to why the batch failed.

10.3 Procedure for Reporting Problems

When a problem with the functionality of an IS Utility Application Product is perceived, execute a sample run with the minimum amount of data. You can specify the debug option '-D', the spy option '-S', and/or the memory option '-M'. These debugging messages will be added to the journal log file. Once you have collected the messages, send the log file along with a description of the problem encountered, to CSS.

Files to send (all that apply): See the Error logging section above that explains the log file as well as default paths.

- Site name or CSS Case Number
- System Configuration
 - Operating System platform and version
 - Database platform and version
Run `"fn_util whichfn"` and supply output to CSS
 - IS Domain, Organization, ssn, and so on
Run `"nch_check"` and send output to CSS
 - IS version and stamp for module:
`stamp /fnsw/lib/shobj/*SysV*`
 - ISTK version and stamp for module:
`stamp /fnsw/client/shobj/*SysV*`
 - Directory listing of ISTK installation point to verify permissions
(`ls -alR /fnsw/client > filename`)

- Is ISTK/Utility installed directly on IS or on a remote system? If remote, supply the Operating System for remote environment.
 - PS Utility version (i.e. **HPII_start -v** or **MRII_start -v**)
 - PS Utility stamp on UNIX systems only (i.e. stamp **HPII_start** or **MRII_start**)
 - Directory listing of PS Utility installation point to verify permissions (`ls -al /fnsw/local/bin > filename`)
3. CheckSSN output from display in DOS prompt or Command window
 4. State problem
 5. Outline steps on how to reproduce problem
 6. Specify the exact command used to start/stop the PS Utility
 7. Output from display in DOS prompt or Command window of Utility runtime
 8. Nohup files - applicable for HP11 and MR11 ONLY (i.e. `/tmp/Utility_Name.datetime`)
 9. Utility Journal logs (`/fnsw/local/bin/journals/*`)
 10. Utility Configuration file (**HP11.cfg**, **MR11.cfg**)
 11. Input files used (DocID list files, eob files, *transact.dat* files, images, etc.)
 12. ISTK Error logs (`/fnsw/client/tmp/ISTKyyyymmdd`)
 13. IS Error Logs (`/fnsw/local/logs/elogs/elog#####`)
 14. Stack trace of core (if a core file is produced)

10.4 When to use wal_purge – UNIX ONLY

The ISTK program `/fnsw/client/bin/wal_purge` can clean up numerous problems but it can also be used incorrectly when ISTK programs are running, and can cause application/program failures. Only run `wal_purge` when all ISTK applications are stopped. You can use a script that will run immediately after bringing up the FILENET software but before starting any ISTK applications/programs.

The following cases are times when `wal_purge` can be used to clean up an application problem.

1. If the FILENET software is shut down before the utility (HP11/MR11) is stopped.
2. If any changes are made to the **HP11.cfg/MR11.cfg** file.
3. If a newly created doc class or index is not recognized by HP11/MR11.
4. Various security problems with HP11/MR11.

10.5 How to compute the MaxSubBatchSize

The `MaxSubBatchSize` value can vary greatly between batch contents. The method described below is merely a starting point and does not guarantee the values will be correct the first time. There are many variables to consider in the calculations. Some trial and error testing might be necessary to find the best value.

The following example has been explained using commands found in a UNIX environment. There are numerous ways of collecting the same information in a Windows environment.

Estimate the total byte count of all the pages and the *transact.dat* file for the largest batch. In other words, the **MaxSubBatchSize** value must be greater than the largest sub-batch. Run "wc" on UNIX to determine the total byte count, in the directory that contains the images.

```
wc *
```

Using the total byte count for the images, add the images sum to the byte count of the *transact.dat* file and then divide by 1024 to get the kb size. Now working in kb values, add 1kb for each image file and another 1kb for the *transact.dat* to determine the **MaxSubBatchSize** value.

Images bytes + *transact.dat* bytes = total bytes / 1024 = total kb

6128838 + 826 = 6129664 / 1024 = 5986 kb

Total kb + 1kb for each image + 1kb for the *transact.dat* file = Grand Total in kb

5986 + 99 + 1 = 6086 kb

Using CSM_tool find the free_sectors in the "page_cache" using the command:

```
<CSM_tool> s 1 long
```

1 sector is equal to 1kb:

If the free_sectors are 4113 then the space available is = 4113 kb

Round 4113 kb to the nearest 1024 which would be 4096 kb

The available page_cache number (4096 kb) is the largest that you can set **MaxSubBatchSize**. Take the size of the batch (6086) and divide by 3. Round up to an integer (6086/3 = 2028.666) in this case 2029. This is the smallest value that the **MaxSubBatchSize** can be, to allow HP11 to break up the batch into 3 pieces (sub-batches).

If the batch size divided by 3, as calculated in the above paragraph is greater than the largest **MaxSubBatchSize** figure allowed, either increase the cache or decrease the size of the batch. An incorrect setting of 2028 would actually cause 4 batches to be created which would cause HP11 to fail because the maximum amount of sub-batches per batch is 3.

One additional variable can be the page count per document. In the above example, if the batch is one document that is 99 pages long, then calculate the page_cache size with the idea that you will only have one batch for the 99-page document because documents cannot span batches. So with this constraint another method would be to take the number of documents, divide by three and round up. Calculate the amount of space needed for the largest document using the previous method and multiply that by the number of documents per batch.

Additional fine-tuning will most likely be necessary. Performance will always be higher with fewer batches being processed. Increasing the batch size in the sample above from 2029 kb to approximately 3046 will cause 2 sub-batches to be created instead of 3. Additionally, fewer documents that contain more pages where possible will also add to better performance.

Workaround for error: 80,1,54 – IS error “No more space in batch”, followed by HP11 error “Can’t write doc header”. This error occurs there is not enough space in cache. The current workaround is to run the batch through using the –D and –S flags. Take note of the space necessary to write the header and space available to write the header in the log file just before the error is received. Figure the difference, round the number up, and add this figure to the **MaxSubBatchSize** value. The increase will most likely be approximately 2 kb.

10.6 Fast Batch Committal

Fast batch committal commits groups or batches of documents together for more efficient committal. It is a performance-tuning item, but only applies to remote committal or Application Servers with no local Storage Service. Therefore, turning the IS – FBC setting ON will not effect HP11 as it is importing to a storage device.

10.7 Document Type setting limitation

HP11 and MR11 can not set the F_DOCTYPE as this is not a value option available during import. The only way to set special document types is to do a post-committal modification using ORACLE or SQL.

10.8 OS path issue

The Operating System contains an environment file that can load paths which origins can be difficult to detect. The file is /etc/profile.

10.9 HP CDROM mounting for ISO9660 CD type

The utilities are released on a CD type that is created for all supported operating systems. HP operating systems require a special mount command to read an ISO9660 CD type.

```
mount -o cdcase /dev/cd0 /cdrom
```

10.10 Error codes that need specific actions

0,2,#### – folder/file path does not exist

0,13,#### – folder/file permissions are incorrect

0,13,1030 – This error can occur when *.*lock* files are not deleted when running on a Windows system. Security attributes during a file copy can cause folders or files to take on read-only permissions. Should this occur you will most likely see the error tuple and following text output “Can’t delete lock file [Permission Denied]” in the journals/imp* file. Check your WorkingDirectory folder and *.*lob* file read permissions.

15,16,17 – This is an IS network error and needs to be resolved by the System Administrator or by contacting FILENET CSS.

80,1,54 – IS error “No more space in batch”, followed by HP11 error “Can’t write doc header”. See the section above titled “How to compute the **MaxSubBatchSize**”.

80,0,44 – IS error “String passed across network exceeds maximum length.”

80,1,46 – IS error “Requested function is not a legal operation.”

This error can be the result of another error that should be caught in the HP/II/MR/II journal log files. "Attempt to create a batch with a batch name which is too long." The batch name input exceeds the 20 character length limitation.

156,0,24 - <NCH,0,24> "A network related error was encountered."

The most common reasons for this error are:

- The "nch-server" entry in the /etc/hosts file is incorrect.
- The Domain or Organization name in the **HP/II.cfg/MR/II.cfg** file is incorrect.
- The FILENET IS software is stopped.
- A network problem has occurred.

202,6,3 - ./CheckSSN (10662) WARNING: Current process is not registered

This error usually means that the ISTK libraries are not installed properly and the PS Utility is accessing the IS libraries instead of the ISTK libraries.

202,0,9 – SysV segmentation violation: Problem between IS and ISTK memory management modules. Any error of this type is difficult to detect. The following is a list of know problems that can receive this error:

- 202,0,9 – ISTK install incomplete (Modules not copied down entirely or configuration incomplete.)
- 202,0,9 – ISTK and Utility compile mismatch (Utility is compiled for each ISTK version. See section on Hardware and Software requirements.)
- 202,0,9 – Utility installation incomplete (Not all modules were copied down or permissions on HP/II/MR/II modules are incorrect on UNIX.)
- 202,0,9 – The SysV module from IS is being accessed instead of the SysV module in the ISTK installed directory. Check your PATH to see that the ISTK path is listed before the IS path.

Error output that can be found in the nohup log (/tmp/HP/II_import`yyyymmdd` or /tmp/MR/II_import`yyyymmdd`) file when there is an incorrect path present in the environment:

```
sys_log: fnc_get_process_name error: program is not registered. pid = 15701  
Process aborting due to segmentation violation...
```

0,0,1051 – Does not produce any message output. However, if the *eob* file has changed to *err* and contains one of the following System Errors, and VerifyTiffFrequency is enabled, the journal log file will contain a "Tiff Verification failed" entry.

System Err: 512

System Err: 2618

System Err: 1 (on NT usually)

Journal log entry:

```
--- 2002/05/07 07:56:52 <fnsw> (007934) Err <0,0,1051>  
Tiff Verification failed for  
'/TestEnv/TestBasePS/HP/IItest/wkdir_pshp1/./data/01batch05/asciifile'  
HP/II Err: Image file did not validate
```

Additional Error Codes

This table is a general list of errors that could be encountered when using any of the IS Utility Application tools. Included is a description of their meaning where appropriate. These error code numbers can change in subsequent builds and would not be reflected in this document.

ERROR NUMBER	ERROR TEXT	ERROR INFORMATION
999	lowest error – empty	
1000	Incorrect number of cfg argument	
1001	cfg bad key word	
1002	cfg exceed max	
1003	cfg expect brace	
1004	Cfg file is empty	
1005	cfg get failed	
1006	cfg invalid index type	
1007	cfg invalid keyword	
1008	cfg invalid number	
1009	cfg too many token	
1010	cfg no 2 nd quote	
1011	cfg Required Directory missing from Configuration File	
1012	cfg not digit	
1013	Value for Keyword is not valid	
1014	cfg open fail	
1015	cfg too many index	
1016	cfg unexpected eof	
1017	end of file	
1018	EOB file contains bad information	
1019	cannot convert	
1020	Data Error. Checksum of file is bad	
1021	DocClass info not in configuration file	
1022	Number of Documents in EOB and transact.dat mismatch	
1023	Document ID List File is required (-f option)	
1024	Can't exec command	
1025	Index is not defined in document class	
1026	Invalid document number	
1027	Invalid object id	
ERROR NUMBER	ERROR TYPE	QUALIFIED ERROR
1028	Invalid object name	
1029	Invalid ISR type	

1030	IO error	Could not create open, read or directory. Usually a permissions problem or the file system ran out of space or an NFS mount was lost
1031	Critical IO error – Program Abort	
1032	Can't kill a child process	
1033	Line too long	
1034	malloc failed	Unable to get an allocation of memory from operating system.
1035	Miscellaneous error	
1036	Source domain must be supplied for this application	
1037	Missing required index in configure	
1038	Required command-line option missing	
1039	No annotations were found for this document	ISR error on attempt to complete Activity Logging Event on an annotation.
1040	Document does not contain any page	
1041	Multiple Classes are not supported by this application	ONLY MR11 will receive this error. MR11 does not support multi doc classes.
1042	One or more critical resources are not available	
1043	Not a valid FileNet format. See SC_convert.out for details	
1044	Not enough space	For HP11 Page Cache is full and for MR11 BES Cache is full. Check cache.
1045	open file fail	
1046	Number of Pages in EOB and transact.dat mismatch	
1047	Value for required index is missing	
1048	Can't spawn the external report program	
1049	System Error, see Errno	
1050	Program Terminal ID is Required	
ERROR NUMBER	ERROR TYPE	QUALIFIED ERROR
1051	Image file did not validate	
1052	Too many docs in subbatch. MaxDocPerSubBatch too small	
1053	Too many docs returned by query. Must query on unique index	
1054	Batch contains more than one family	Multiple Families NOT supported.
1055	too many pages in doc	Document exceeds the max page limit

		of %d. Ln='%s'
1056	Too many sub batches. MaxBatchSize too small	HPII ONLY: Maximum number of sub batches has been reached. MaxBatchSize in the configuration file might be too small. This must be big enough to hold 1000 documents or 2000 pages.
1057	Transact file contains incorrect format	
1058	Required input file was not provided	
1059	PS Shared Library Mismatch-Contact FileNet Support	
1060	EMC-Centera Support not yet available-No ClipId Produced	
1061	EMC-Centera device IO error	
1062	ISR Validation Failure	
1063	Unsupported SSN	MSS error can occur if all system SSNs in the MSS NET are not in the license key (PSLSF).
1064	Preliminaries Complete	

11. QUESTIONS AND ANSWERS

This section contains general questions that are often asked about HP II and MR II.

11.1 MR II general product questions

- What is the infrastructure required for MR II implementation?

A standard FILENET Image Services configuration is required because MR II uses FILENET's Batch Entry Services (BES). Since MR II uses the Image Services Toolkit (also known as ISTK) to talk to FILENET Image Services, it can be run on a client/server PC or on the FILENET main server (UNIX or Windows).

- What is the suggested method for scanning and importing the images?

The documents to be imported can be acquired in any number of ways as long as they are placed on a mounted file system in a directory structure identified as the WorkingDirectory in **MR II.cfg**. They can also be stored in any number of formats. The only restriction is the actual viewing/printing formats supported by the client software.

- How is the indexing done for the images?

The index values are stored in a formatted file and read by MR II. The format of the index file and the parameters on how MR II reads them are configured during the implementation.

- How many images could be transferred in a batch? What are the performance factors? How many batches can be run?

The optimum batch size, number of batches, and system impact can all be modeled using FILENET tools. A FILENET Professional Services Implementation Package can determine and tune these settings. Generally MR II can process approximately 50,000 images per day. This number depends on factors like system configuration, size of images, system availability, and so on.

12. APPENDIX A—EPHEMERAL PORT SETTINGS

12.1 Definition

A TCP/IPv4 connection consists of two endpoints, and each endpoint consists of an IP address and a port number. Therefore, when a client user connects to a server computer, an established connection can be thought of as the 4-tuple of: (server IP, server port, client IP, client port). Usually three of the four are readily known -- client machine uses its own IP address and when connecting to a remote service, the server machine's IP address and service port number are required.

When a connection is established, the client side of the connection uses a port number. Unless a client program explicitly requests a specific port number, the port number used is an **ephemeral** port number. Ephemeral ports are temporary ports assigned by a machine's IP stack, and are assigned from a designated range of ports for this purpose. When the connection terminates, the ephemeral port is available for re-use, although most IP stacks won't re-use that port number until the entire pool of ephemeral ports have been used. So, if the client program reconnects, it will be assigned a different ephemeral port number for its side of the new connection.

Similarly, for UDP/IP, when a datagram is sent by a client from an unbound port number, an ephemeral port number is assigned automatically, so the receiver can reply to the sender.

Symptom and Description:

You might need to tune the "ephemeral port" network settings on your server's OS.

The default settings that come pre-configured with the OS are often insufficient for high-volume FileNet activity. As a result, the OS can intermittently run out of free socket ports. This will momentarily prevent the server from opening any new TCP connections, and activity will come to a halt. The system might not be able to recover once free ports become available again.

The system might be susceptible to this problem if your application mix involves a high number of short server connections.

Symptoms of the problem could include:

- * Client intermittently receives "Method of object failed" during Doc.GetCachedFile
- * Image Services Toolkit logs get chronic SPP type errors (e.g. 15,16,17):
COR_Open: connect failed with errno 10048
- * Workflo Queue intensive app gets:

"COR_Open: connect to 30.34.192.237 [32769] failed with errno 227"

There are two methods for tuning the ephemeral ports:

- * Increase the range of ports available to the OS

* Decrease the "TIME_WAIT" period so the OS can re-use a closed port number more quickly. (Reducing TIME_WAIT is NOT CONFIGURABLE on AIX and HPUX 10.20)

Both methods are platform-dependent. Both methods are independent of each other - you can do one, the other, or both.

Increase the ephemeral port range when you install the IS software. Do not modify TIME_WAIT until needed.

The default value for TIME_WAIT is usually 4 minutes. It can often safely be reduced to as little as 2 minutes or even 30 seconds. TIME_WAIT is designed to prevent any "lost packets" from an old connection, from having the same port number as a current connection. If you're in a high-latency environment (for example, WAN or satellite traffic), your TIME_WAIT should be longer (for example, the full 4 minutes). If you're certain that all packets will be received promptly (within milliseconds), then you can safely reduce TIME_WAIT.

12.2 Windows platform

The ephemeral ports must be adjusted for all platforms. The Windows platform is the most affected if the ports are not modified.

The modifications must be made on all servers in the FileNet domain.

Use the Registry Editor (regedt32.exe) to make the modifications.

12.3 *MaxUserPort*

Description:

Determines the highest port number TCP can assign when an application requests an available user port from the system. Typically, ephemeral ports (those used briefly) are allocated to port numbers 1024 through 5000.

NOTE Windows 2000 does not add this entry to the registry. You can add it by editing the registry or by using a program that edits the registry.

Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters

Data Type: REG_DWORD

Range: 5,000-65,534 (port number)

Default Value: 5000

Recommended value: 65534 (65534 DEC)

12.4 *TcpMaxConnectTransmissions*

Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters

Data Type: REG_DWORD

Recommended value: 5 (5 DEC)

12.5 *TcpMaxConnectRetransmissions*

Description:

Determines how many times TCP retransmits an unanswered request for a new connection. TCP retransmits new connection requests until they are answered or until this value expires

Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters

Data Type: REG_DWORD

Range: 0–255 (*retransmission attempts*)

Default Value: 2

Recommended value: 5 (5 DEC)

12.6 *TcpTimedWaitDelay*

Description:

Determines the time that must elapse before TCP can release a closed connection and reuse its resources. This interval between closure and release is known as the TIME_WAIT state or 2MSL state. During this time, the connection can be reopened at much less cost to the client and server than establishing a new connection.

RFC 793 requires that TCP maintain a closed connection for an interval at least equal to twice the maximum segment lifetime (2MSL) of the network. When a connection is released, its socket pair and TCP control block (TCB) can be used to support another connection. By default, the MSL is defined to be 120 seconds, and the value of this entry is equal to two MSLs, or 4 minutes. However, you can use this entry to customize the interval.

Reducing the value of this entry allows TCP to release closed connections faster, providing more resources for new connections. However, if the value is too low, TCP might release connection resources before the connection is complete, requiring the server to use additional resources to re-establish the connection.

NOTE Normally, TCP does not release closed connections until the value of this entry expires. However, TCP can release connections before this value expires if it is running out of TCP control blocks (TCBs). The number of TCBs the system creates is specified by the value of the [MaxFreeTcbs <58770.asp>](#) entry.

Windows 2000 does not add this entry to the registry. You can add it by editing the registry or by using a program that edits the registry.

Location: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters

Data Type: REG_DWORD

Range: 30-300 seconds

Default Value: 240 seconds = 4 minutes

Recommended value: 45 seconds (45 DEC)

12.7 FN_COR_QLEN

15,16,17 errors – WSAECONNREFUSED 10061 – indicate that the NLS_Archiver is not able to connect to the IS COR_Listen because there is no COR queue space.

The FN_COR_QLEN variable needs to be explicitly set in Windows, the default of 5 is not enough. Go to **My Computer > Advance > Environment Variables** to add the variable FN_COR_QLEN with a value of 20-25 (this is the default used on UNIX systems). This should resolve the 15,16,17 errors.

12.8 UNIX Platform

Possible issues with ephemeral ports on UNIX platforms

On the IS server, verify that the UDP low ephemeral port is not set to 32768. We've only seen this on AIX so far, but it could also happen on Solaris.

On AIX enter 'no -a | grep udp_ephemeral'

On Solaris enter ' ndd /dev/udp udp_smallest_anon_port' , then ' ndd /dev/udp udp_largest_anon_port'

On HP 10.30/11.x enter ' ndd /dev/udp udp_smallest_anon_port' , then ' ndd /dev/udp udp_largest_anon_port'

On HP 10.20, this shouldn't be a problem, but check to see if the high_port_enable flag has been set by entering '/usr/contrib/bin/nettune -l | grep high' (the switch is a lower case L)

In all cases the number for the low end of the range must be higher than 32771 or other programs could come in and step on NCH_daemon. By default most UNIX implementations start this value at 32768.

12.9 IBM AIX

On AIX, check the /etc/rc.net file (or on AIX 5.2 in the /etc/tunables/nextboot file) for the following lines:

```
/usr/sbin/no -o udp_ephemeral_high=65535
```

```
/usr/sbin/no -o udp_ephemeral_low=42767
```

For more details on the steps to modify the /etc/rc.net file, see the IS 4.0 Installation and Configuration Procedures for AIX/6000.

Perform the steps in this section on all servers. Backup the /etc/rc.net file. As root user, make sure you have write permission on the above files by entering:

```
chmod 754 /etc/rc.net
```

Use your preferred text editor (such as vi) to modify the /etc/rc.net file.

Locate the following 'if' statement, near the end of the file:

```
#####
# The socket default buffer size (initial advertised TCP window) is being
# set to a default value of 16k (16384). This improves the performance
# for ethernet and token ring networks. Networks with lower bandwidth
# such as SLIP (Serial Line Internet Protocol) and X. 25 or higher bandwidth
# such as Serial Optical Link and FDDI would have a different optimum
# buffer size.
# (OPTIMUM WINDOW = Bandwidth * Round Trip Time)
#####
if [ -f /usr/sbin/no ] ; then
    /usr/sbin/no -o tcp_sendspace= 16384
    /usr/sbin/no -o tcp_recvspace= 16384
fi
```

Add the following lines:

```
/usr/sbin/no -o tcp_keepidle= 80
/usr/sbin/no -o tcp_keepintvl= 20
/usr/sbin/no -o tcp_ephemeral_high= 65535
/usr/sbin/no -o tcp_ephemeral_low= 42767
/usr/sbin/no -o udp_ephemeral_high= 65535
/usr/sbin/no -o udp_ephemeral_low= 42767
```

so the "if" statement looks like this:

```
if [ -f /usr/sbin/no ] ; then
    /usr/sbin/no -o tcp_sendspace= 16384
    /usr/sbin/no -o tcp_recvspace= 16384
    /usr/sbin/no -o tcp_keepidle= 80
    /usr/sbin/no -o tcp_keepintvl= 20
    /usr/sbin/no -o tcp_ephemeral_high= 65535
    /usr/sbin/no -o tcp_ephemeral_low= 42767
    /usr/sbin/no -o udp_ephemeral_high= 65535
    /usr/sbin/no -o udp_ephemeral_low= 42767
fi
```

Save your changes and exit from the file.

Reboot the server(s)

12.10 HP-UX

HPUX 10.20 ONLY

Perform the steps in this section on all servers. Your FileNet system will run more efficiently if you change the following in the `/etc/rc.inittfns` file. The modification expands the number of available ephemeral ports. This modification is not required, but has been found to be optimal when running FileNet software.

Ephemeral ports are temporary ports assigned by a server's IP stack, and are assigned from a designated range of ports for this purpose. When network traffic is extremely heavy, it's possible to run out of ephemeral ports unless you specify the `high_port_enable` option in `/etc/rc.inittfns`. Backup the `/etc/rc.inittfns` file. As root user, make sure you have write permission on the this file by entering:

```
chmod 754 /etc/rc.inittfns
```

Use your preferred text editor (such as `vi`) to modify the `/etc/rc.inittfns` file.

Locate the following statement near the end of the file: **# Set up network options HPUX (parameters in half- seconds)**

Add the following line after that statement:

```
/usr/contrib/bin/nettune -s tcp high_port_enable 1
```

Save your change and exit from the file.

Reboot the server(s)

HPUX 10.30/HPUX 11.x

As root user, `vi /sbin/rc2.d/S340net =>`

```
ndd -set /dev/udp udp_smallest_anon_port 42767  
ndd -set /dev/udp udp_largest_anon_port 65535  
ndd -set /dev/tcp tcp_smallest_anon_port 42767  
ndd -set /dev/tcp tcp_largest_anon_port 65535  
ndd -set /dev/tcp tcp_time_wait_interval 30000 (for HPUX 10.30/11.x)
```

Save your changes and exit from the file.

Reboot the server(s)

12.11 Sun Solaris

Perform the steps in this section on all servers. On Solaris, set these parameters in `/etc/rc2.d/S69inet` - double check the install documentation for syntax, or contact CSS Network Support for assistance.

Your FileNet system will run more efficiently if you make the following changes to the `/etc/rc2.d/S69inet` file. The modification expands the number of available ephemeral ports and reduces the time-out delay. These modifications are not required, but have been found to be optimal when running FileNet software.

Ephemeral ports are temporary ports assigned by a server's IP stack, and are assigned from a designated range of ports for this purpose. When network traffic is extremely heavy, it's possible to run out of ephemeral ports unless you specify a wider range of port numbers in `/etc/rc2.d/S69inet`.

The `tcp_close_wait_interval` parameter determines the length of time the server waits before reusing a closed ID socket. Although the default value is typically around 240000 milliseconds (4 minutes), this parameter can safely be reduced to as little as 30000 milliseconds (30 seconds) on high-speed networks.

Make a backup copy of the `/etc/rc2.d/S69inet` file before you modify it.

As root user, make sure you have write permission on the file by entering:

```
chmod 754 /etc/rc2.d/S69
```

Use your preferred text editor (such as vi) to modify the `/etc/rc2.d/S69inet` file.

Add the following lines near the end of the file:

```
ndd -set /dev/udp udp_smallest_anon_port 42767  
ndd -set /dev/udp udp_largest_anon_port 65535  
ndd -set /dev/tcp tcp_smallest_anon_port 42767  
ndd -set /dev/tcp tcp_largest_anon_port 65535  
ndd -set /dev/tcp tcp_close_wait_interval 30000 (for Solaris 2.x only)  
ndd -set /dev/tcp tcp_time_wait_interval 30000 (for Solaris 8 and above)
```

Save your change and exit from the file.

Reboot the server(s)