Installation Guide NTF

Installation Instructions

ERICSSON

Disclaimer

The information in this document is subject to change without notice.

Ericsson makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Ericsson shall not be liable for errors contained herein nor for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Trademark List

Solaris	is a trademark of Sun Microsystems, Inc.
Apache Software Foundation	This product includes software developed by the Apache Software Foundation (http://www.apache.org)

1/1531-CRH 109 127 Uen PAB5 2008-05-08

Contents

1 Introduction

1.1 Scope

This document provides step by step instructions for installation, upgrade and uninstallation of the software for the NTF component.

1.2 Audience

This document is primarily written for software installation technicians.

1.2.1 Prerequisite Knowledge

The readers of this document should have a good understanding of Messaging-over-IP and good knowledge and experience of:

- IP based networks
- UNIX
- To install NTF in a HA system, the reader must be familiar with Solaris clusters.

Note: Installation of the software is only to be performed by personnel authorized by having attended the corresponding courses.

1.3 Related Documents

The following documents also contain information related to the component, or are referred to from this document:

- Messaging-over-IP Technical Overview
- Glossary
- Operation and Maintenance NTF
- Operation & Maintenance Deployment Server
- Solaris documentation (http://docs.sun.com) *System Administration Guide, Solaris 10*

1.4 Document Conventions

1.4.1 Notational Conventions

This document uses the following notational conventions:

Bold font style is used for emphasis, to indicate keywords and buttons.

Italic font style is used for references, window/page/menu titles and specific terms.

`Bold Monospace font` is used to describe user input.

`Monospace font` is used for code, paths and on-screen computer output.

Square brackets ‘‘[]’’ are used to enclose parameters that are optional.

Curly brackets ‘‘{}’’ are used to enclose parameter values given as examples.

Less than and greater than characters ‘‘<>’’ are used to enclose variable names.

Backslash ‘‘\’’ at the end of a line means that this line is continued onto the next line.

The dollar sign ‘‘$’’ is the UNIX Korn (and Bourne) shell prompt.

The percent sign ‘‘%’’ is the UNIX C shell prompt.

The number sign ‘‘#’’ is the superuser prompt.

The number sign ‘‘#’’ is also used for comments.

A star ‘‘*’’ matches zero or more characters.

A question mark ‘‘?’’ matches one character.

A tilde ‘‘~’’ means the home directory of the current user.

`<BASEDIR>` is the installation directory for NTF, for example `/apps/ntf`.

`<NTF_HOME>` is the instance directory for NTF, for example `/apps/ntf` or `/global/ms1/ntf`.

2 Installation

2.1 General

The NTF component provides notification services, as described in *Messaging-over-IP Overview*. The component includes software developed by the Apache Software Foundation (http://www.apache.org/).

The instructions in this document covers the installation program for the NTF component.

NTF consists of three parts, an installation an instance and a HA-agent. The **installation** is the NTF package with programs and libraries. The **instance** is the configuration files and templates. The **HA-agent** is the monitoring software needed when NTF runs in a HA system.

If NTF is installed in a standard system, no HA-agent is used and the installation and instance are located in the same directory, so you do not need to think about the different parts. The terms are used in the installation process though, so you should recognize them.

If NTF is installed in a HA system, the HA-agent is used and the installation and one or more instances are created separately, in different directories. Then the different parts are important.

2.1.1 Container File

The NTF software is delivered in a compressed tar file named:

- `ntf_<rel>.crh109127.solaris10.tar.gz`

where "<rel>" gives the release number of the software.

In this document, the container file is called **`<ntf_container_file>`**

2.2 Prerequisites

2.2.1 Software

The NTF component requires that the following software has been installed:

- Solaris 10. Preferably, Solaris 10 shall be installed using the Deployment Server (see document Operation and Maintenance Deployment Server). Alternatively, install Solaris 10 with Sun's latest Recommended Security Patch Cluster for Solaris 10 (http://sunsolve.sun.com)

- Java 1.5. Java must be installed as a package. Ntf checks for package `SUNWj5rt` and `SUNWj5rtx`, both packages must be version 1.5.

2.2.2 Free Memory

Running the NTF component requires that there is more than 256MB virtual memory free.

2.2.3 Disk space

The NTF component requires that there is sufficient disk space in the following partitions:

Table 1 NTF Disk Space Requirements

Partition	Required disk space (minimum)	Comments
<BASEDIR>. e.g. /apps/ntf	20 MB	For NTF installation.
<NTF_HOME>	600 MB per NTF instance	For configuration files, data files and log files.
/etc	2 MB	For `rc` scripts.
/tmp	2 MB	Required as temporary installation space (the partition may be changed by setting the `TEMPDIR` environment variable).

To check disk space, use e.g. **`df -k`** command.

2.2.4 Component Dependencies

During installation (when the NTF instance is added), the NTF component tries, once, to register itself in MCR. When NTF is started, it checks its entry in MCR and creates or updates it if necessary. If the MCR entry can not be created or updated, NTF will retry every five minutes until it succeeds or NTF is stopped.

During installation (when the NTF instance is added), the NTF component tries, once, to register its users in MUR. When NTF is started, it checks its users in MUR and creates them if necessary. If the users can not be created, NTF will retry every five minutes until it succeeds or NTF is stopped. Since NTF can not receive any requests until these users are available, NTF will not start with anything else until the registration succeeds.

The NTF interfaces with the following components: MS, MUR, MER, MCR, SMS-C, MMS-C, MVAS/MAS, MCC, WAP-gateway, ESI and MEMA.

2.2.5 Enabler Setup

Usually, NTF shall be able to send notifications with SMS, MWI, WAP and MMS and needs to connect to an SMS-C, WAP-gateway and MMS-C. These enablers must be registered in MCR. This is done with the "Enabler setup" part of MUP. You select the name of for example the SMS-C and set the host name, password and protocol. It is necessary to restart NTF after an enabler has been added. The protocol is set to CIMD2 or SMPP, depending on what is used by the SMS-C. NTF adapts its behavior to the selected protocol automatically.

To see how to add an Enabler, see MUP documentation.

2.3 Prepare Installation

The following steps must be performed before starting the installation of the NTF component.

1. Become superuser (root).

   ```
   % su -
   ```

2. Make sure the port number (e.g. 18001) for NTF management and monitoring connections is not in use. The port number is configurable, so make sure that the port number you are about to select is not already in use.

   Use for instance:

   ```
   netstat -na | grep <portno>
   ```

   where **<portno>** is the port number listed above.

3. If previous, unsuccessful, installation attempts have taken place, uninstall such earlier installations.

   For instructions on how to uninstall the component, see Section 6 Uninstallation on page 31.

2.4 Install NTF in a Standard System

The installation procedure consists of the following steps:

1. Unpack the NTF distribution.

   Get the distribution and unpack it in a temporary location.

2. Configure installation parameters.

3. Install the NTF distribution.

   The NTF package is installed, and NTF and its users are registered in MCR and MUR. Backup and start/stop functionality is set up.

Note: Since each NTF has some configuration parameters that must be individual, NTF can not be automatically installed (using Deployment Server) in a standard system.

4. Verify the installation.

   Check the installation log file. Start NTF and see that it runs as expected.

5. Customize NTF.

6. Start NTF.

   By default NTF has to be started manually, but an installation parameter can be set to start NTF automatically after installation.

7. Backup the installation.

   Backup the new configuration files.

2.4.1 Unpack the NTF Distribution

1. Insert the CD that contains the software into CD-ROM drive on the host.

2. Change to CD-ROM directory.

   ```
   # cd /cdrom/cdrom0
   ```

   (or type: `cd /cdrom/<CD-ROM name>`)

3. Copy the container file for the NTF component to a temporary directory on the host.

   Note: The directory must be writeable.

   ```
   # cp <ntf_container> <tempdir>
   ```

4. Change to the temporary directory.

   ```
   # cd <tempdir>
   ```

5. Extract files from the container file:

   ```
   # gzip -dc <ntf_container_file> | tar xf -
   ```

2.4.2 Configure the Installation Parameters

Note: The parameter names are case insensitive, they are converted to all uppercase by the installation program

Table 2 Installation Parameters

Parameter	Description	Default Value
BASEDIR	Installation directory for the NTF component.	`/apps/ntf`
StartNtf	Start NTF after successfull installation. Yes / No	`No`
McrHost	Name of the host where the messaging component register is located.	`mcrhost`
McrUserName	Username for MCR administrator	`IComponent`
McrPassword	Password for MUR administrator	`abc123`
LogicalZone	Name of the logical zone this NTF belongs to (if any) Set LogicalZone to the empty string if nearness of other components shall be determined from network addresses instead of zone.	`unspecified`.
MurHost	The host where the Messaging User Registry is located	`murhost`
MurUserName	Distinguished name of MUR administrator	`cn=Directory Manager`
MurPassword	Password for MUR administrator	-
SearchBase	Search base for the user registry	`ou=site1,o=userdb`
ImapHost	The host where this NTFs messaging server is located	-
ImapUserName	This is the basename for the notification mailboxes. If several NTFs should be installed in the system, the first one should be named: gnotification1 and the second: gnotification2 etc.	**gnotification1**

The parameters in this table are used in the installation process, and must be correct from the start.

Table 3 Site Dependent Parameters

Parameter
BearingNetwork
ImapRootUserName
ImapRootPassword
SmeSystemId
SmePassword
SmeSystemType

All parameters from NTFs main configuration file `notification.cfg` can be set in the response file `LMENtf.response` prior to installation and will then appear in `notification.cfg` after installation. The table above contains the parameters that must be changed on most sites, and as a convenience they are included in `LMENtf.response`. They are documented in Reference [3] NTF Operation & Maintenance, section Configuration Parameters.

2.4.3 Install the NTF Software

1. Run the installation script:

   ```
   # ./install.sh

   Starting install of NTF R12A with the response file
       LMENtf.response.
   Please check the IMAP user name:
           IMAPUserName=gnotification3
   Continue:  [y,n,?]
   ```

   Type **`y`** to continue or **`n`** to abort the installation.

   Note: Since it is easy to forget to change the IMAP user name, the install script reminds you to check it, and displays the value from the response file

2. The installation script will now install the NTF component with the output:

   ```
   Installing NTF package
   Copying instance files and directories...
   ...done. Registering component in MCR...
   ...done. Registering NTF users in MUR...
   ...done.
   Installation of <LMENtf> was successful.
   NTF has been successfully installed.
   View log file
       /apps/logs/ntf/install/ntf.2006_05_30_1411.installog
       for details
   ```

3. Verify the installation by checking the installation log file. The name of the log file is displayed by the installation program.

2.4.4 Customize NTF

Customize NTF as described in Section 2.6 Customize NTF on page 17.

2.4.5 Start NTF

Once NTF is installed and customized, you start it as described in Reference [3] NTF Operation & Maintenance, section Start NTF.

2.5 Install NTF in a HA-System

The installation of NTF in a HA system is a little more complicated. The HA-agent and NTF must be installed on all cluster nodes. Then the NTF instances you want to run are added in the cluster file system.

The installation procedure consists of the following steps:

1. Unpack the NTF distribution.

   Get the distribution and unpack it in a temporary location.

2. Install the NTF HA-Agent

   Install the HA-Agent on all cluster nodes and set up the resource groups.

3. Configure installation parameters.

   Define values for parameters that are different for each installation.

4. Install the NTF software.

   The NTF packages are installed on all cluster nodes.

   Note: NTF can be installed manually or automatically from a Deployment server. The amount of customization that can be done from the Deployment server is limited though.

5. Add NTF instances.

   The individual NTF instances are added in the cluster file system. The instances and their users are registered in MCR and MUR. Backup is set up. Add the wanted NTF instances as described in Section 2.5.5 Add NTF instances on page 15.

6. Set instance parameters.

Some parameters must be unique for each instance. They must be set before an instance is started. Set the parameters as described in Section 2.5.6 Set Instance Parameters on page 16.

7. Customize NTF instances.

8. Start NTF.

   Start NTF using the cluster software.

2.5.1 Unpack the NTF Distribution

1. Insert the CD that contains the software into CD-ROM drive on the host.

2. Change to CD-ROM directory.

   ```
   # cd /cdrom/cdrom0
   ```

   (or type: **cd /cdrom/<CD-ROM name>**)

3. Copy the container file for the NTF component to a temporary directory on the host.

   Note: The directory must be writeable.

   ```
   # cp <ntf_container> <tempdir>
   ```

4. Change to the temporary directory.

   ```
   # cd <tempdir>
   ```

5. Extract files from the container file:

   ```
   # gzip -dc <ntf_container_file> | tar xf -
   ```

2.5.2 Install the NTF HA-Agent

2.5.2.1 Prerequisites

- The cluster software is operational
- Disk management software has been installed and configured
- Quorum devices have been configured
- The cluster installmode property is set to disabled
- The NTF HA Agent package has been unpacked from the NTF delivery file. When the delivery file is unpacked the NTF HA Agent package is located in the HA_Agent directory.

2.5.2.2 Manual installation

Install the LMEAntf package on all cluster nodes (including standby nodes) using the pkgadd command (preferably using the Sun cluster console software):

```
# cd <Unpacked NTF delivery dir.>/HA_Agent/

# pkgadd -d LMEAntf.pkg LMEAntf
```

Register the LMEA.ntf resource type with the cluster software on one cluster node:

```
# /usr/cluster/bin/scrgadm -a -t LMEA.ntf
```

2.5.2.3 Automatic installation (Deployment Server)

- Copy the LMEAntf.pkg package to the /apps/LMEAadmin/install/pkg/common directory on the Deployment server.
- Create a new action object in the Deployment server named “Install NTF HA Agent”, use “action_pkg” in the action field and “LMEAntf” in the argument field.
- Create a new group (if it does not already exists) called “Cluster”. Include the newly created action in the list of actions.
- Make sure that all cluster nodes have the “Cluster” group selected.

2.5.2.4 Configuration

Prerequisites:

The NTF instance is installed in a cluster filesystem, typically this would be in the `/global/<resouce group name>/ntf` directory (e.g. `/global/ms1/ntf`)

Note: In a typical configuration the MS and the corresponding NTF instance should share the same resource group.

Create the NTF object:

`# /usr/cluster/bin/scrgadm -a -j <NTF resource> -g <Resource group> -t LMEA.ntf -y Network_resources_used=<Network dependencies> -x Basedir=<NTF installation directory>` where:

<NTF resource> The desired name of the new resource, by convention this should be of the format: <Resource group name>-ntf

Typical example: **`ms1-ntf`**

<Resource group> The Sun Cluster resource group to which this resource should belong.

Typical example: **`ms1`**

<Network dependencies> The name of the logical host object (IP address) for which the NTF component should wait before starting. This resource is typically named:<Resource group>-logical

Typical example: **`ms1-logical`**

<NTF installation directory> The directory where this NTF instance has been installed or will be installed in. This directory is typically called: **`/global/<Resource group name>/ntf`**

Typical example: **`/global/ms1/ntf`**

2.5.3 Configure the Installation Parameters

Note: The parameter names are case insensitive, they are converted to all uppercase by the installation program

Table 4 Installation Parameters

Parameter	Description	Default Value
HAMode	Defines this as an installation in a HA system.	**`no`** Must be changed to yes.
BASEDIR	Installation directory for the NTF component.	**`/apps/ntf`**
McrHost	Name of the host where the messaging component register is located.	**`mcrhost`**
McrUserName	Username for MCR administrator	**`IComponent`**
McrPassword	Password for MUR administrator	**`abc123`**
LogicalZone	Name of the logical zone this NTF belongs to (if any) Set LogicalZone to the empty string if nearness of other components shall be determined from network addresses instead of zone.	**`unspecified`**.
MurHost	The host where the Messaging User Registry is located	**`murhost`**

Table 4 Installation Parameters

Parameter	Description	Default Value
MurUserName	Username for MUR administrator	**admin**
MurPassword	Password for MUR administrator	**admin**
SearchBase	Search base for the user registry	**ou=site1,o=userdb**

The parameters in this table are used in the installation process, and must be correct from the start.

Table 5 Site Dependent Parameters

Parameter
BearingNetwork
ImapRootUserName
ImapRootPassword
SmeSystemId
SmePassword
SmeSystemType

All parameters from NTFs main configuration file `notification.cfg` can be set in the response file `LMENtf.response` prior to installation and will then appear in `notification.cfg` after installation. The table above contains the parameters that must be changed on most sites, and as a convenience they are included in `LMENtf.response`. They are documented in Reference [3] NTF Operation & Maintenance, section Configuration Parameters.

2.5.4 Install the NTF Software

When NTF is installed in a HA system, no script `rc.ntf` is installed. NTF must be started and stopped with the cluster software.

2.5.4.1 Manual Installation

1. Run the installation script:

```
# ./install.sh

Starting install of NTF R12A with the response file \
    LMENtf.response.
Please check the IMAP user name:
        IMAPUserName=gnotification3
Continue:  [y,n,?]
```

Type **y** to continue or **n** to abort the installation.

Note: Since it is easy to forget to change the IMAP user name, the install script reminds you to check it, and displays the value in the response file

2. The installation script will now install the NTF component with the output:

```
Installing NTF package
Installation of <LMENtf> was successful.
NTF has been successfully installed.
View log file /apps/logs/ntf/install/ntf.<date>.installog\
    for details
```

2.5.4.2 Automatic installation (Deployment Server)

If several hosts should be installed with the NTF component, the NTF can be installed using the Deployment Server.

It is recommended to first install a single NTF and verify that it works as intended, for example that the language files are modified as they should be. Then, use the configuration files from the working NTF to configure the Deployment Server installation.

1. Login to the Deployment Server.
2. Copy the NTF container file to the directory `/apps/LMEAadmin/install/dist` and move to this directory.
3. Unzip and extract the files `job_install_ntf` and `LMENtf.response` from the tar file, e.g:

   # gzip -dc <ntf_container_file> | tar xf -

4. Move the script job_install_ntf to the directory /apps/LMEAadmin/install/scripts/jobs

   # mv ntf_files/job_install_ntf /apps/LMEAadmin/install/scripts/jobs/

5. Configure NTF according to: Section 2.4.2 Configure the Installation Parameters on page 6. Be sure to set “StartNtf=no”, so the NTFs do not start until they have been individually configured.
6. Login to the Deployment Server web interface.
7. Add a Group NTF with no actions (if it not already exists).
8. Edit the Clients that shall be installed so that they belong to the Group NTF(if not already done) according to O&M Deployment Server, see Reference [4].

9. Add a Job (according to O&M Deployment Server, see Reference [4]) with the following parameters:

Name	NTF Install
Use certificate	Yes
Status	Scheduled
Run a script (remote)	/apps/LMEAadmin/install/scripts/jobs/job_install_ntf
Arguments	ntf_<rel>.crh109127.solaris10.tar <Deployment_Server_Hostname>
Copy script to remote	Yes
When	Now
Repeat	Never
Groups	NTF

10. Check the result in the log files for Jobs.
11. Verify the installations according to: Section 2.5.9.1 Verify Installation of NTF on page 17.
12. Customize each NTF individually as described below.

2.5.5 Add NTF instances

After NTF is installed on all cluster nodes, it is not ready for use. First all NTF instances must be added. Repeat the following procedure for each instance:

- Log in to a host where the file system for the NTF instance exists.
- Run the NTF addinstance program

  ```
  #<BASEDIR>/bin/addinstance -d /global/ms1/ntf -i
  ntf1@ms1.ha.domain -m ms1.ha.domain -u gnotification4
  ```

  When NTF is running in a HA system, you must supply **all** the parameters to addinstance:

  - **d** Directory where the NTF instance shall be created
  - **i** The MCR name of the new NTF instance.
  - **m** Name of the host where this NTFs message store is running.
  - **u** Base name for NTFs gnotification users.

Note: The options i, m and u correspond to the parameters Mcr_Instance_Name, ImapHost and ImapUserName in `<NTF_HOME>/cfg/notification.cfg` and `addinstance` will set these values in the configuration file for the new instance

```
Copying instance files and directories...
...done. Registering component in MCR...

MCR Registration
****************

MCR Registration successful

...done. Registering NTF users in MUR...
Gnotification registration
**************************
gnotification4_0 registered.
gnotification4_1 registered.
...
Gnotification registered successful

Notification.off registration
*****************************
notification.off.ms1.ha.domain registered.
Notification.off registration successful
...done.
```

When an NTF instance is added in a HA system, NTFs MCR entry will contain an SNMP context.

The backup script for the instance is located in <NTF_HOME>/moipbackup.

2.5.6 Set Instance Parameters

A few parameters must be set individually for each instance, or the instances will conflict with each other:

SnmpAgentPort Tells the NTF instance which port (and optionally which IP-address) to use for communication between the NTF traffic process and the NTF management agent. SnmpAgentPort must be configered with IP-address in HA-systems. Each NTF instance must have a unique combination of IP-address and port number.

These parameters are described in Reference [3]

2.5.7 Customize NTF Instances

Customize each NTF instance as described in Section 2.6 Customize NTF on page 17.

2.5.8 Start NTF

Once NTF is installed and customized, you start the instances with the cluster software.

2.5.8.1 Enable the NTF agent

```
# /usr/cluster/bin/scswitch -e -j <NTF resource>
```

where <NTF resource> is the same resource as mentioned previously. The command will automatically start-up the NTF component if the resource group is active and enabled.

2.5.8.2 Disable the NTF agent

If you need to bring the NTF instance down for service or other reasons you need to disable the resource.

Note: If the component is brought down outside of the clusters control it will be interpreted as an error and corrective action will be taken (typically involving restarts and eventually a one or more reboots).

```
# /usr/cluster/bin/scswitch -n -j <NTF resource>
```

2.5.9 Verify Installation

2.5.9.1 Verify Installation of NTF

Verify the installation of a NTF by performing the following steps:

1. Check the installation log file `ntf.<date>.installog` to see if there were any errors at the installation of the NTF component.
2. Start NTF and check that it is running and does not write any error messages.
3. Verification done.

2.6 Customize NTF

Once the NTF software is installed, most NTF installations must be adapted to the customer. This is done by editing one or more of the files:

- <NTF_HOME>/cfg/notification.cfg
- <NTF_HOME>/cfg/systemnotification.cfg
- <NTF_HOME/cfg/outdial-default.cfg
- <NTF_HOME/cfg/charconv.cfg

- <NTF_HOME/cfg/charconv.cimd2
- <NTF_HOME>/templates/*.cphr

Note: If only `notification.cfg` is customized, you can do this prior to installation by adding the parameters to the response file. You can then also start NTF automatically after installation by setting the parameter StartNtf=yes in the response file

Note: In a HA-system, where you first install the software and then create instances, you can pre-customize instances by editing the files in `<BASEDIR>/instance_template`. When an instance is created, its files are copied from this directory, so if you edit these files, the change will apply to all instance you create afterwards.

These files are described in Reference [3] NTF Operation & Maintenance, section Change Configuration Parameters in File.

Note: In order for NTF to send any notifications, sieve filters and gnotification mailboxes have to be configured in MS, see Chapter 5.10 Manage Notification Mailboxes in Reference [5] MS Operation & Maintenance.

Note: If you have customized the outdial sequence in NTF R8A (using the parameter OutdialDelayers), the corresponding customization must be done to the new outdial sequence configuration file `outdial-default.cfg`

2.6.1 Upgrading Phrase Files

When upgrading from MoIP 6 FD1 or earlier, a tool for converting phrase files is provided. The tool **phr2cphr.pl** (in <BASEDIR>/bin/) can be used to convert the obsolete phrase files (*.phr) to the new phrase file format (*.cphr). Example usage is:

```
<BASEDIR>/bin/phr2cphr.pl <NTF_HOME>/templates/en.phr >
<NTF_HOME>/templates/en.cphr
```

The resulting cphr follows the syntactical rules of the obsolete count specific template files that were optionally used in earlier versions.

All phr files found in <NTF_HOME>/templates during upgrade is converted to .cphr files if one does not exist.

Note: Please review the resulting cphr-files and verify the expected output using the tool **generatetext** (See Reference [3] NTF Operation & Maintenance, section Configuring Notification Templates for Different Languages)

2.7 Backup

A backup should be taken:

- After each installation.

- Prior to each Upgrade or Rollback procedure.
- Any time the configuration has been changed.

For information about which files that are recommended to back up, see *Operation and Maintenance NTF*, section *Backup*.

3 Start and Stop

The NTF component is automatically started at reboot and if specified during installation. To manually start/restart/stop/check status of the component, see *Operation and Maintenance NTF*, section *Operation*

4 Upgrade

This chapter describes how to upgrade NTF.

Note: When you upgrade from NTF R8A, outdial notifications that were started but not completed, will be lost.

4.1 Upgrade NTF, in a Standard System from Solaris 8

Upgrading NTF from version R11B or earlier uses the following procedure:

1. Backup the NTF data and configuration files.

2. Install Solaris 10 using the deployment server.

3. Unpack the NTF distribution.

   Get the distribution and unpack it in a temporary location.

4. Install the new NTF software.

   Run the installation script.

5. Restore the old backup by running the script `<BASEDIR>/bin/restoreOldBackup`. This script reads an old backup file and extracts it to correct locations for the new NTF. Run this script with arguments to the old directories for template and data. For a typical installation upgrading from R11B where the location hasn't been modified this will be:

   ```
   # <BASEDIR>/bin/restoreOldBackup -d /apps/ntf \
      -t /opt/ntf/templates -D/var/ntf/journal \
      -l /opt/ntf/logs/NotificationTrace.log \
      -f <full path to backupfile>
   ```

6. Start NTF. If upgrading from R11B patch 5 or any earlier version this can take some time of the slamdown_journal file has many entries. This is because NTF must add cosname to every entry in the file at startup, this will normally take 100 ms per entry. If it takes to long it is possible to delete the slamdown_journal.current and slamdown_journal.last file and restart NTF.

7. Customize NTF with any new functionality that should be used according to Section 2.6 Customize NTF on page 17. All phrase files must be updated to the new format according to Section 2.6.1 Upgrading Phrase Files on page 18.

8. Restart NTF if NTF has been customized.

4.2 Upgrade NTF, in a Standard System from Solaris 10

Upgrading NTF from version R12A or later uses the following procedure:

1. Backup the NTF data and configuration files.
2. Unpack the NTF distribution.

   Get the distribution and unpack it in a temporary location.
3. Install the new NTF software.

   Run the installation script. The new NTF software is installed in the same location as the old version. The old configuration is kept.
4. Start NTF if NTF is not configured to start after the installation.
5. Customize NTF with any new functionality that should be used according to Section 2.6 Customize NTF on page 17. All phrase files must be updated to the new format according to Section 2.6.1 Upgrading Phrase Files on page 18.
6. Restart NTF if NTF has been customized.

Note: When upgrading, all installation parameters and all configuration is taken from the old installation. The parameters in the response file, `LMENtf.response`, are ignored, with one exception; you may set `StartNtf` to control whether NTF shall start after the upgrade or not.

4.3 Upgrade from a Standard System to a High Availibility System.

1. Backup the NTF data and configuration files.
2. Uninstall the old NTF using the procedure documented for that release.
3. If upgrading from R11A the following step is needed:

   Install Solaris 10 using Deployment Server.
4. Install HA Agent and NTF software as described in Step 4 on page 9
5. Add the wanted NTF instances as described in Section 2.5.5 Add NTF instances on page 15.
6. Restore the old backup by using the script `<BASEDIR>/bin/restoreOldBackup` if the old version is version R11B or lower. This script reads an old backup file and extracts it to correct locations for this version. Run this script with arguments to the old directories for template and data. For a typical installation upgrading from R11B where the location hasn't been modified this will be:

   ```
   # <BASEDIR>/bin/restoreOldBackup -d <NTFHOME> \
   ```

```
-t /opt/ntf/templates -D /var/ntf/journal \
-l /opt/ntf/logs/NotificationTrace.log \
-f <full path to backupfile>
```

The configuration can be restored according to *Operation and Maintenance NTF*, section *Backup* if the old version is R12 or higher

7. Customize NTF with any new functionality that should be used according to Section 2.6 Customize NTF on page 17. All phrase files must be updated to the new format according to Section 2.6.1 Upgrading Phrase Files on page 18.

8. Start NTF using the cluster software. If upgrading from R11B patch 5 or any earlier version this can take some time of the slamdown_journal file has many entries. This is because NTF must add cosname to every entry in the file at startup, this will normally take 100 ms per entry. If it takes to long it is possible to delete the slamdown_journal.current and slamdown_journal.last file and restart NTF.

4.4 Upgrade from NTF in a High Availability System from Solaris 8.

1. Backup the NTF data and configuration files for all NTF instances.

2. Uninstall old NTF using the procedure documented for that release.

3. Install Solaris 10 using Deployment Server.

4. Install HA Agent and NTF software as described in Section 2.5 Install NTF in a HA-System on page 9

5. Add the wanted NTF instances as described in Section 2.5.5 Add NTF instances on page 15, probably in the locations where old NTF was installed.

6. Restore the old backup by using the script `<BASEDIR>/bin/restoreOldBackup`. This script reads an old backup file and extracts it to correct locations for the current version. Run this script with arguments to the old template and datadirectory and to the old log file..

```
# <BASEDIR>/bin/restoreOldBackup -d <NTFHOME> \
   -t <old template directory> \
   -D <old datadirectory> -l <oldlogfile> \
   -f <full path to backupfile>
```

7. Customize NTF according to Section 2.6 Customize NTF on page 17. All phrase files must be updated to the new format according to Section 2.6.1 Upgrading Phrase Files on page 18.

8. Start NTF. If upgrading from R11B patch 5 or any earlier version this can take some time of the slamdown_journal file has many entries. This is because NTF must add cosname to every entry in the file at startup, this will

normally take 100 ms per entry. If it takes to long it is possible to delete the slamdown_journal.current and slamdown_journal.last file and restart NTF.

4.5 Upgrade from NTF in a High Availability System from Solaris 10.

Note: By migrating NTF instances and upgrading one node at a time, it is possible to make a more clever upgrade. It is recommended to design such an upgrade procedure for the configuration at hand. An example is shown in Section 4.6 Upgrade from NTF in a High Availability System with extra node from Solaris 10. on page 26.

1. Backup the NTF data and configuration files for all NTF instances.
2. Install the new NTF software

   Run the installation script on all nodes. The new NTF software is installed in the same location as the old version. The old configuration is kept.
3. Upgrade the NTF configuration.

   Upgrade the NTF configuration with the script <BASEDIR>/bin/upgradeinst ance. The -d option tells upgradeinstance where the instance is located.

   #<BASEDIR>/bin/upgradeinstance -d /global/ms1/ntf
4. Customize NTF according to Section 2.6 Customize NTF on page 17. All phrase files must be updated to the new format according to Section 2.6.1 Upgrading Phrase Files on page 18.
5. Start NTF.

4.6 Upgrade from NTF in a High Availability System with extra node from Solaris 10.

This section describes an alternative way to upgrade NTF with less downtime. It can be used when an unused cluster node is available for the upgrade.

This upgrade procedure uses an unused node, called new and the node where NTF is running, called old.

1. Backup the old NTF data and configuration files for all NTF instances.
2. Install HA Agent and NTF software on the new node, as described in the steps Step 1 on page 9 to Step 4 on page 9.
3. Stop NTF on the old node using the cluster software.
4. Move NTF.

   Move NTF from the old node to the new node using the cluster software. After the move, both the old and new NTF software will be on the new

node. Since the solaris package information is for the new software, the old software will not be used.

5. Upgrade NTF instance

   Upgrade the NTF configuration with the script <BASEDIR>/bin/upgradein stance.

   the -d option tells upgradeinstance where the instance is located.

   ```
   #<BASEDIR>/bin/upgradeinstance -d /global/ms1/ntf
   ```

   This adds any new configuration files. Existing configuration files are kept and are not updated or replaced.

6. Customize NTF according to Section 2.6 Customize NTF on page 17.

7. Start NTF with the cluster software.

8. Install HA Agent and NTF software on the old node.

9. Done.

Note: If several NTF instances shall be upgraded, you repeat the steps from ‘‘Stop old NTF.’’ to ‘‘Remove obsolete files.’’ for each instance.

4.7 Procedure for Solaris 10 Live upgrade

Installation of the NTF adds some files located outside the /apps file system on the operating system disk. When an upgrade of the operating system (Solaris 10) is performed, all NTF related files residing outside of /apps will no longer be visible. The files affected are:

/etc/init.d/rc.ntf

/etc/rc3.d/S99ntf (symbolic link)

These files will normally be backed up and restored into the new upgraded operating system by the LiveUpgrade framework. Should the framework fail the files can be retrieved from the old Operating System version using the lumount command.

5 Rollback

To roll back to the previous installation of NTF:

1. Make sure that you have superuser (root) privileges.
2. Back up the configuration data for the NTF you want to replace.

   See *Operation and Maintenance NTF*, section *Backup*.
3. Uninstall the NTF component. See Section 6 Uninstallation on page 31.
4. Reinstall the NTF release you want to roll back to and use the backup made from that release as input.
5. Verify that the NTF has been installed correctly and that the NTF has started.

6 Uninstallation

6.1 Uninstall NTF in a Standard System

Uninstallation of an NTF component is performed by using the `<NTF_HOME>/uninstall/uninstall.sh` program. It will remove the NTF instance, uninstall the NTF software and perform the necessary unregistrations.

To uninstall the NTF component:

1. Become superuser (root).

   `% su -`

2. Back up required files.

   For information about recommended files to back up, see *Operation and Maintenance NTF*, section *Backup*.

3. Change directory to: `<NTF_HOME>/uninstall/`

   `# cd <NTF_HOME>/uninstall/`

4. Run the `uninstall.sh` script.

   `# ./uninstall.sh`

   ```
   Do you want to uninstall the NTF component?
   Choice:  [y,n,?]
   ```

   Type **`y`** to continue or **`n`** to abort the uninstallation.

5. The uninstallation script will now uninstall the NTF component with the output:

   ```
   NTF has been successfully uninstalled.
   View /apps/logs/ntf/uninstall/ntf.<date>.uninstallog
   for details
   ```

6. Manually delete the data in NTFs data directory, unless you will immediately install NTF again, and want to keep pending notifications.

6.2 Uninstall NTF in a HA-System

Uninstallation in a HA system consists of removing NTF instances with the program `<BASEDIR>/bin/rminstance` and uninstalling NTF software with the program `<BASEDIR>/uninstall/uninstall.sh`. You can remove

some instances and keep some instances and the NTF software, or you can remove all instances and remove the NTF software.

To uninstall the NTF component:

6.2.1 Remove NTF instances

1. Become superuser (root).

   ```
   % su -
   ```

2. Back up the NTF instances you will remove.

   For information about recommended files to back up, see *Operation and Maintenance NTF*, section *Backup*.

3. Remove NTF instances that shall be uninstalled, from the cluster file system. This will remove the instance files and unregister the instance and its NTF users.

   ```
   #/apps/ntf/bin/rminstance -d /global/ms2/ntf
   MCR unregistration
   ******************
   MCR unregistration successful
   Gnotification unregistration
   ****************************
   gnotification2_0 removed.
   gnotification2_1 removed.
   ...
   Gnotification unregistered successful
   Notification.off unregistration
   ******************************
   notification.off.<mailhost> removed.
   Notification.off unregistration successful
   Data files in /global/ms2/ntf/data are not removed \
       automatically.
   Remove them manually if you do not want to keep\
       pending notifications.
   ```

4. Remove data files from the instances data directory, if you do not want to keep them.

6.2.2 Remove NTF resource

As all the NTF resources are associated to the same resource type (LMEAntf), every NTF resource has to be removed.

Remove NTF resource

a Become superuser (root).

   ```
   % su -
   ```

b Stop NTF.

```
# scswitch –n –j <ntf resource>
```

For instance, # scswitch –n –j ms1-ntf.

c Remove the NTF resource.

```
# scrgadm –r –j <ntf resource>
```

For instance, # scrgadm –r -j ms1-ntf.

Repeat step two and three until all NTF resources are removed.

6.2.3 Remove NTF resource type

1. Become superuser (root).

   % su -

2. Remove the NTF resource type, unless you want to reuse it later.
   # scrgadm -r -t LMEA.ntf

6.2.4 Uninstall the NTF Software

If you have removed all NTF instances that may run on a node, you can uninstall NTF from that node:

1. Become superuser (root).

   % su -

2. Change directory to: `<BASEDIR>/uninstall/`

   # cd /apps/ntf/uninstall/

3. Run the `uninstall.sh` script.

   # ./uninstall.sh

   ```
   After NTF is uninstalled, the tools to remove instances\
       are gone.
   Remember to remove the instances first if you do not wish\
       to keep them.
   Do you want to uninstall the NTF component?
   Choice:  [y,n,?]
   ```

   Type **y** to continue or **n** to abort the uninstallation.

4. The uninstallation script will now uninstall the NTF component with the output:

```
NTF has been successfully uninstalled.
View /apps/logs/ntf/uninstall/ntf.<date>.uninstallog\
    for details
#
```

7 Logging

NTFs runtime logfiles are described in Reference [3] NTF Operation & Maintenance, section Log Files

7.1 Installation Log

The installation of the NTF component produces the following log file:

- `/apps/logs/ntf/install/ntf.<date>.installog`

7.2 Uninstallation Log

The uninstallation of the NTF component produces the following log file:

- `/apps/logs/ntf/uninstall/ntf.<date>.uninstallog`

Reference List

[1] *Messaging-over-IP Overview*, 5/1551-HDB 101 02 Uen

[2] *Glossary*, 1/0033-1/HDB 101 02 Uen

[3] *Operation & Maintenance NTF*, 1/1543-CRH 109 127 Uen

[4] *Operation & Maintenance Deployment Server*, 2/1553-CRH 109 266 Uen

[5] *Operation & Maintenance MS*, 1/1543-CRH 109 084 Uen