



# Hitachi HiCommand® Backup Services Manager

## Notification Callbacks Setup and Installation

November 2006

Version 6.0

Doc ID: MK-95APT0016-01

## 1. Introduction

Hitachi HiCommand® Backup Services Manager version 5.0 and above provides a unique feature that allows you to register a callback notification script to be executed on various threshold or event status changes within the HBSM product. This document provides the information that you will need to install and configure this feature. Here are some examples of alerts or callbacks that can be setup and automated within the HBSM product:

- Change in the status of a tape drive (for example notification of down drives)
- Change in the status of any piece of tape media (for example: notification of a frozen tape)
- Existence of a job that is running slow, stalled, or long-running

If at any time during the install you encounter a problem, or if you simply want to verify a step or have a question about the installation, please contact the Hitachi Data Systems Technical Support Center who will be happy to assist you:

### Technical Support Center Contact Information

Phone:

Nth & Latin America            1-800-348-4357

Europe                                +(44)-175-361-8000

Asia Pacific (call USA GCC)    +(1) 858-547-4765

Email:                                    support@hds.com

## 2. Installation Instructions

The following is a list of installation steps for the HBSM Callbacks feature. You must complete all of these steps for the callback mechanism to be successful.

- 2.1 Verify that you have the external callback runtime library loaded and in the correct oracle directory:
  - 2.1.1 On UNIX platforms confirm that the file /opt/aptare/oracle/lib/extproc.so exists and has the permission bits of “755”. If the file is not present you may need to copy it from /opt/aptare/database/callbacks/*OSDIR* where *OSDIR* is the operating system on which the database is installed.
  - 2.1.2 On WINDOWS platforms, confirm that the file called “extproc.dll” exists in C:\opt\oracle\bin.
- 2.2 The next step is to validate that the existing Oracle TNS Listener process is setup to handle the Oracle “extproc” external interface. To enable this listener, please complete the following steps:

### On a UNIX Portal

- 2.2.1 Logon to the portal as the unix user aptare
- 2.2.2 cd /opt/aptare/oracle/network/admin
- 2.2.3 Verify that the listener.ora file contains the following entries :

```
LISTENER =
```

```
(ADDRESS_LIST =
  (ADDRESS= (PROTOCOL= IPC)(KEY=extproc))
  (ADDRESS= (PROTOCOL= TCP)(PORT=1521))
)

SID_LIST_LISTENER =
(SID_LIST =
  (SID_DESC =
    (SID_NAME = scdb)
    (ORACLE_HOME = /opt/aptare/oracle)
  )
  (SID_DESC =
    (SID_NAME = extproc_sid)
    (ORACLE_HOME = /opt/aptare/oracle)
    (PROGRAM = extproc)
  )
)

STARTUP_WAIT_TIME_LISTENER = 0
INBOUND_CONNECT_TIMEOUT_LISTENER = 10
TRACE_LEVEL_LISTENER = OFF
```

#### 2.2.4 Verify that the tnsnames.ora file contains the following entry:

```
EXTPROC_CONNECTION_DATA =
  (DESCRIPTION =
    (ADDRESS=
      (PROTOCOL=IPC)
      (KEY=extproc)
    )
    (CONNECT_DATA= (SERVICE_NAME=extproc_sid)(SID=extproc_sid))
  )
```

#### 2.2.5 Stop and restart the Oracle TNS listener using the following commands:

```
lsnrctl stop
lsnrctl start
```

You should see an output similar to the following after starting the Oracle TNS listener

```
LSNRCTL for Linux: Version 9.2.0.7.0 - Production on 29-NOV-2005
10:08:42

Copyright (c) 1991, 2002, Oracle Corporation. All rights reserved.

Starting /opt/oracle/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 9.2.0.7.0 - Production
System parameter file is /opt/oracle/network/admin/listener.ora
Log messages written to /opt/oracle/network/log/listener.log
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=extproc)))
Listening on:
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=aptaredev3.corp)(PORT=1521)))

Connecting to (ADDRESS=(PROTOCOL=IPC)(KEY=extproc))
STATUS of the LISTENER
-----
Alias                LISTENER
Version              TNSLSNR for Linux: Version 9.2.0.7.0 -
Production
```

```
Start Date                29-NOV-2005 10:08:42
Uptime                    0 days 0 hr. 0 min. 0 sec
Trace Level               off
Security                  OFF
SNMP                      OFF
Listener Parameter File   /opt/oracle/network/admin/listener.ora
Listener Log File         /opt/oracle/network/log/listener.log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=extproc)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=aptaredev3.corp)(PORT=1521)))
Services Summary...
Service "extproc_sid" has 1 instance(s).
  Instance "extproc_sid", status UNKNOWN, has 1 handler(s) for this
  service...
Service "scdb" has 1 instance(s).
  Instance "scdb", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

### On a WINDOWS Portal

- 2.2.1 Ensure you are logged on to Windows as a user who is a member of the ORA\_DBA group.
- 2.2.2 Using Windows Explorer, navigate to C:\opt\oracle\network\admin.
- 2.2.3 Verify that the listener.ora file contains the following entries:

```
LISTENER =
  (ADDRESS_LIST =
    (ADDRESS= (PROTOCOL= IPC)(KEY=extproc))
    (ADDRESS= (PROTOCOL= TCP)(PORT=1521))
  )

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = scdb)
      (ORACLE_HOME = C:\opt\oracle)
    )
    (SID_DESC =
      (SID_NAME = extproc_sid)
      (ORACLE_HOME = C:\opt\oracle)
      (PROGRAM = extproc)
    )
  )

STARTUP_WAIT_TIME_LISTENER = 0
INBOUND_CONNECT_TIMEOUT_LISTENER = 10
TRACE_LEVEL_LISTENER = OFF
```

- 2.2.4 Verify that the tnsnames.ora file contains the following entry:

```
EXTPROC_CONNECTION_DATA =
  (DESCRIPTION =
    (ADDRESS =
      (PROTOCOL=IPC)
      (KEY=extproc)
    )
    (CONNECT_DATA= (SERVICE_NAME=extproc_sid)(SID=extproc_sid))
  )
```

- 2.2.5 Stop and restart the Oracle TNS listener by opening a DOS command window and then executing the following commands:

```
lsnrctl stop
lsnrctl start
```

You should see an output similar to the following after starting the Oracle TNS listener

```
LSNRCTL for 32-bit Windows: Version 10.2.0.1.0 - Production on 23-JAN-
2006 19:35:50

Copyright (c) 1991, 2005, Oracle. All rights reserved.
Starting tnslnsr: please wait...
TNSLSNR for 32-bit Windows: Version 10.2.0.1.0 - Production
System parameter file is C:\opt\oracle\network\admin\listener.ora
Log messages written to C:\opt\oracle\network\log\listener.log
Listening on:
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\extprocipc)))
Listening on:
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=127.0.0.1)(PORT=1521)))
Connecting to (ADDRESS=(PROTOCOL=IPC)(KEY=extproc))
STATUS of the LISTENER
-----
Alias                LISTENER
Version              TNSLSNR for 32-bit Windows: Version
10.2.0.1.0 - Production
Start Date           23-JAN-2006 19:35:56
Uptime               0 days 0 hr. 0 min. 3 sec
Trace Level          off
Security              ON: Local OS Authentication
SNMP                 OFF
Listener Parameter File C:\opt\oracle\network\admin\listener.ora
Listener Log File    C:\opt\oracle\network\log\listener.log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\extprocipc)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=127.0.0.1)(PORT=1521)))
Services Summary...
Service "extproc_sid" has 1 instance(s).
  Instance "extproc_sid", status UNKNOWN, has 1 handler(s) for this
service...
Service "scdb" has 1 instance(s).
  Instance "scdb", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

- 2.3 The next step is to setup the callback database triggers and register the external callback process within the Oracle database. To do this you need to do the following:
- 2.3.1 On UNIX cd to /opt/aptare/database/callbacks
  - 2.3.2 On WINDOWS open a DOS command window and cd to C:\opt\oracle\database\callbacks
  - 2.3.3 sqlplus portal/<portal\_password<sup>1</sup>> @setup\_callbacks
- 2.4 To customize the callbacks registered in the Oracle database, you need to make copies of the template callback scripts and place them in the appropriate directory for your operating system. The template scripts included with the product illustrate how to perform a simple action with the parameters passed into the callback script. You can then edit your copy of the template scripts and customize them for your specific needs. For example, you can interface to your email process to send email alerts for particular job failures. To copy the scripts follow the instructions below for your particular operating system:
- 2.4.1 On UNIX, copy the job completion script from  
“/opt/aptare/database/callbacks/samples/aptare\_job\_completed.sh” to  
“/opt/aptare/database/callbacks/bin/aptare\_job\_completed.sh”  
and the tape drive script from  
“/opt/aptare/database/callbacks/samples/aptare\_drive\_status\_change.sh” to  
“/opt/aptare/database/callbacks/bin/aptare\_drive\_status\_change.sh”  
Make sure the shell scripts that are invoked by the HBSM Callbacks are executable (i.e. permissions should be 755).
  - 2.4.2 On Windows, copy the job completion script from  
“C:\opt\oracle\database\callbacks\samples\aptare\_job\_completed.cmd” to  
“C:\opt\oracle\database\callbacks\bin\aptare\_job\_completed.cmd”  
and the tape drive script from  
“C:\opt\oracle\database\callbacks\samples\aptare\_drive\_status\_change.cmd” to  
“C:\opt\oracle\database\callbacks\bin\aptare\_drive\_status\_change.cmd”
- 2.5 To test the installation of the HBSM Callbacks, please execute the following steps:
- 2.5.1 For UNIX:  
cd /opt/aptare/database/callbacks  
Run the following command to test the above callback:  
sqlplus portal/<portal\_password> @tst  
You should see a response similar to the following:  
Invoking the script:  
/opt/aptare/database/callbacks/bin/aptare\_drive\_status\_change.sh  
Script returned: 0  
PL/SQL procedure successfully completed.
  - 2.5.2 For Windows:

---

<sup>1</sup> Default password is portal

```
cd \opt\oracle\database\callbacks
```

```
sqlplus portal/<portal_password> @tst
```

You should see a response similar to the following:

```
Invoking the script:
```

```
C:\opt\oracle\database\callbacks\bin\aptare_drive_status_change.  
cmd
```

```
Script returned: 0
```

```
PL/SQL procedure successfully completed.
```

### 3. Advanced Customization

To customize the parameters passed to the callback scripts, their name, location or the conditions under which the callback scripts are invoked, you can make copies of, then edit and re-apply the following files:

For UNIX:

```
/opt/aptare/database/stored_procedures/user_package.sql
```

```
/opt/aptare/database/stored_procedures/nbu/nbu_callbacks_pkg.sql.
```

For Windows:

```
C:\opt\oracle\database\stored_procedures\user_package.sql
```

```
C:\opt\oracle\database\stored_procedures\nbu\nbu_callbacks_pkg.sql
```

Hitachi Data Systems provides the source for these files so you can modify and personalize for your environment. Assuming your copied files are named `my_user_package.sql` and `my_nbu_callbacks_pkg.sql`, here's how to apply your modified files:

- 3.1 On UNIX cd to `/opt/aptare/database/stored_procedures`
- 3.2 On WINDOWS open a DOS command window and cd to `C:\opt\oracle\database\stored_procedures`
- 3.3 `sqlplus portal/<portal_password> @my_user_package.sql`  
`sqlplus portal/<portal_password> @my_nbu/nbu_callbacks_pkg.sql`

Note: You may have to reapply your modified files if you upgrade to the product to a newer release.