

Hvrcontrol

Contents

- [Name](#)
- [Synopsis](#)
- [Description](#)
- [Options](#)
- [Examples](#)
- [Files](#)

Name

hvrcontrol - Send and manage internal control files.

Synopsis

hvrcontrol [-options] *hubdb chn*

Description

Command **hvrcontrol** either sends HVR 'controls' to replication jobs, or removes them. A 'control' is a message file which can serve two functions;

1. To tell a job to do something else when it is already running. For example, wakeup or change its default behavior.
2. To instruct a job to treat certain rows in a special way, e.g. skip an old or 'bad' row, send a certain change straight to a 'fail table', or be resilient for some rows during an online refresh.

Correct use of command **hvrcontrol** requires understanding of undocumented internals of HVR. For this reason this command should only be used after consultation with HVR Technical Support or when its use is recommend by an HVR error message.

HVR sends control files internally in these areas;

- Command **Hvrstart** tells the **Hvrscheduler** to send a **trigger** control file. Jobs which are in a 'cycle loop' will detect this file and do an extra cycle even if they are still running. When this cycle is done they will delete this control file, so **Hvrstart -w** commands will terminate (otherwise they would keep hanging).
- Online refresh jobs (**Hvrrefresh -q**) sends *refresh taskname_online* (default is **refr_online**) control files to instruct capture and integrate jobs to skip changes made to the base tables before the refresh and to treat changes made while the refresh is running with resilience.

The argument *hubdb* specifies the connection to the hub database. For more information about supported hub databases and the syntax for using this argument, see [Calling HVR on the Command Line](#).

Options

This section describes the options available for command **hvrcontrol**.

Parameter	Description
-c	Only send control to capture jobs. By default , the control is sent to both capture and integrate jobs.

-d	Delete older control files while creating the new control, so that the new control replaces any old controls. The older control is deleted if it was for the same job and it had the same control name (see option -n).
-D	Delete control files and do not create a new control. All control files for the channel are deleted unless options -c , -i , -l or -n are supplied.
-Ename=value	Set environment variable <i>name</i> to <i>value</i> in affected job.
-f	Affected changes should be sent directly to the 'fail table' instead of trying to integrate them. All changes are failed unless options -w or -t are supplied. This option can only be used on an integrate job and cannot be combined with options -m , -r or -s .
-F	Affected jobs should finish at the end of the next replication cycle.
-hclass	Location <i>class</i> of the hub database. Valid values for <i>class</i> are db2 , db2i , ingres , mysql , oracle , postgresql , sqlserver , or teradata . For more information, see Calling HVR on the Command Line .
-i	Only send control to integrate jobs. By default , the control is send to both capture and integrate jobs.
-lx	Only send controls to jobs for locations specified by <i>x</i> . Values of <i>x</i> may be one of the following: Values of <i>x</i> maybe one of the following: <ul style="list-style-type: none"> • <i>loc</i>: Only location <i>loc</i>. • <i>/1-/2</i>: All locations that fall alphabetically between <i>/1</i> and <i>/2</i> inclusive. • <i>!/loc</i>: All locations except <i>loc</i>. • <i>!/1-/2</i>: All locations except for those that fall alphabetically between <i>/1</i> and <i>/2</i> inclusive. Several -lx instructions can be supplied together.
-mcol	For affected changes value of column <i>col</i> should be set to missing. Setting the value to missing means that the change will not have data for <i>col</i> anymore. The column value is set to missing for all changes unless options -w or -t are supplied. It is not recommended to use this option on a key column. This option cannot be combined with options -f , -r or -s .
-nctrlname	Name of control. This is part of the file name of the control created for each job and it also affects which old control files are deleted if option -d or -D are supplied. The default is adhoc .
-r	Affected changes should be treated with resilience (as if action /Resilient=SILENT is defined) during integration. All changes are resilient unless options -w or -t are supplied. This option can only be used on an integrate job and cannot be combined with options -f , -m or -s .
-s	Affected changes should be skipped. All changes are skipped unless options -w or -t are supplied. This option cannot be combined with options -f , -m or -r .

<p>-ty</p>	<p>Only filter rows for tables specified by <i>y</i>. Values of <i>y</i> may be one of the following:</p> <ul style="list-style-type: none"> • <i>tbl</i>: Only table <i>tbl</i>. • <i>t1-t2</i>: All tables that fall alphabetically between <i>t1</i> and <i>t2</i> inclusive. • <i>!tbl</i>: All tables except <i>tbl</i>. • <i>!t1-t2</i>: All tables except for those that fall alphabetically between <i>t1</i> and <i>t2</i> inclusive. <p>Several -ty instructions can be supplied together to hvrcontrol. This option must be used with either options -f, -m, -r or -s.</p>
<p>-u<i>user</i>[/<i>pwd</i>]</p>	<p>Connect to hub database using DBMS account <i>user</i>. For some databases (e.g. SQL Server) a password <i>pwd</i> must also be supplied.</p>
<p>-w<i>where</i></p>	<p>Where condition which must have form <i>columnname operator value</i>.</p> <p>The <i>operator</i> can be either = != <> > < >= or <=. The <i>value</i> can be a number, '<i>str</i>', X'<i>hex</i>, or a date. Valid date formats are <i>YYYY-MM-DD [HH:MM:SS]</i> in local time or <i>YYY Y-MM-DDTHH:MM:SS+TZD</i> or <i>YYYY-MM-DDTHH:MM:SSZ</i> or today or now [[±] <i>SEC S</i>] or an integer (seconds since 1970-01-01 00:00:00 UTC).</p> <p>For some operators (= != <>) the value can be a list separated by ' '. If multiple -w options are supplied then they are AND-ed together. For an OR condition multiple control files may be used. This option must be used with either options -f, -m, -r or -s.</p>
<p>-x<i>expire</i></p>	<p>Expiry. The affected job should expire the control file and delete it when this time is reached. Valid date formats are <i>YYYY-MM-DD [HH:MM:SS]</i> in local time or <i>YYYY-MM-DDTHH:MM:SS+TZD</i> or <i>YYYY-MM-DDTHH:MM:SSZ</i> or today or now [[±] <i>SECS</i>] or an integer (seconds since 1970-01-01 00:00:00 UTC). Option -x0 therefore means that the control will be removed by the job after its first cycle.</p>
<p>-X<i>expire</i></p>	<p>Receive expiry. The affected job should expire the control file and delete it after it has processed all changes that occurred before this time. Valid date formats are <i>YYYY-MM-DD [HH:MM:SS]</i> in local time or <i>YYYY-MM-DDTHH:MM:SS+TZD</i> or <i>YYYY-MM-DDTHH:MM:SSZ</i> or today or now [[±] <i>SECS</i>] or an integer (seconds since 1970-01-01 00:00:00 UTC).</p>

Examples

- **Example 1:** Instruct all jobs in channel **sales** to skip rows for table **x** with **prod_id<5** use:

```
hvrcontrol -s -tx "-wprod_id<5" hubdb/pwd sales
```

- **Example 2:** Instruct all jobs in channel **sales** to send everything before the transaction with **hvr_tx_seq=000014013DF50001** into the fail tables.

```
hvrcontrol -f "-whvr_tx_seq<'000014013DF50001'" hubdb/pwd sales
```

- **Example 3:** Instruct all jobs in channel **sales** to send everything before the change with **hvr_tx_seq=000014013DF50001** and **hvr_countdown=3** into the fail tables. In HVR, each change has a unique **hvr_tx_seq** and **hvr_tx_countdown** combination, with these values acting as major and minor numbers respectively. Note that **hvr_tx_countdown** has reverse ordering (i.e. for a big transaction the first change has countdown 100 and the last has countdown 1). Note the use of comparison operator **<<** for major/minor ordering.

```
hvrcontrol -f "-whvr_tx_seq<<'000014013DF50001'" "-whvr_tx_countdown>3" hubdb/pwd sales
```

- **Example 4:** Instruct an integrate job for location **q** to be resilient for all changes where **(prod_id=1 and prod_price=10) or (prod_id=2 and (prod_price=20 or prod_price=21))** use two HVR controls:

```
hvrcontrol -i -lq -r -wprod_id=1 -wprod_price=10 hubdb/pwd sales
hvrcontrol -i -lq -r -wprod_id=2 "-wprod_price=20|21" hubdb/pwd sales
```

- **Example 5:** Make a running log-based capture job write a dump of its state (including all open transactions) into its log file (**\$HVR_CONFIG/log/hubdb/chn-cap-loc.out**), use the following command:

```
hvrcontrol -c hubdb/pwd sales TxDump
```

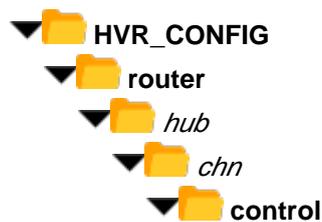
- **Example 6:** View the contents of all control files affecting a channel, use the following command that converts the internal format into a readable XML format:

```
hvrrouterview -s hubdb/pwd sales
```

- **Example 7:** Delete all controls affecting a channel use:

```
hvrcontrol -D hubdb/pwd sales
```

Files



`tstamp.ctrl-jobname` ~~Control file~~ contains instructions for a replication job. The contents of the file can be inspected using command **hvrrouterview**.