NAME

ovsdb−tool − Open vSwitch database management utility

SYNOPSIS

ovsdb−tool [*options*] **create** [*db* [*schema*]]
ovsdb−tool [*options*] **compact** [*db* [*target*]]
ovsdb−tool [*options*] **convert** [*db* [*schema* [*target*]]]
ovsdb−tool [*options*] **needs−conversion** [*db* [*schema*]]
ovsdb−tool [*options*] **db−version** [*db*]
ovsdb−tool [*options*] **schema−version** [*schema*]
ovsdb−tool [*options*] **db−cksum** [*db*]
ovsdb−tool [*options*] **schema−cksum** [*schema*]
ovsdb−tool [*options*] **query** [*db*] *transaction*
ovsdb−tool [*options*] **transact** [*db*] *transaction*
ovsdb−tool [*options*] [**−m** | **−−more**]... **show−log** [*db*]
ovsdb−tool help

Logging options:
[**−v**[*module*[**:***destination*[**:***level*]]]]...
[**−−verbose[=***module*[**:***destination*[**:***level*]]]]...
[**−−log−file**[**=***file*]]

Common options:
[**−h** | **−−help**] [**−V** | **−−version**]

DESCRIPTION

The **ovsdb−tool** program is a command-line tool for managing Open vSwitch database (OVSDB) files. It does not interact directly with running Open vSwitch database servers (instead, use **ovsdb−client**).

Basic Commands

create *db schema*

Reads an OVSDB schema from the file named *schema* and creates a new OVSDB database file named *db* using that schema. The new database is initially empty. This command will not overwrite an existing *db*.

schema must contain an OVSDB schema in JSON format. Refer to the OVSDB specification for details.

compact *db* [*target*]

Reads *db* and writes a compacted version. If *target* is specified, the compacted version is written as a new file named *target*, which must not already exist. If *target* is omitted, then the compacted version of the database replaces *db* in-place.

Version Management Commands

An OVSDB schema has a schema version number, and an OVSDB database embeds a particular version of an OVSDB schema. These version numbers take the form *x***.***y***.***z*, e.g. **1.2.3**. The OVSDB implementation does not enforce a particular version numbering scheme, but schemas managed within the Open vSwitch project use the following approach. Whenever the database schema is changed in a non-backward compatible way (e.g. deleting a column or a table), *x* is incremented (and *y* and *z* are reset to 0). When the database schema is changed in a backward compatible way (e.g. adding a new column), *y* is incremented (and *z* is reset to 0). When the database schema is changed cosmetically (e.g. reindenting its syntax), *z* is incremented.

Some OVSDB databases and schemas, especially very old ones, do not have a version number.

These commands work with different versions of OVSDB schemas and databases.

convert *db schema* [*target*]

Reads *db*, translating it into to the schema specified in *schema*, and writes out the new interpretation. If *target* is specified, the translated version is written as a new file named *target*, which must

not already exist. If *target* is omitted, then the translated version of the database replaces *db* in-place.

This command can do simple ‘‘upgrades’’ and ‘‘downgrades’’ on a database’s schema. The data in *db* must be valid when interpreted under *schema*, with only one exception: data in *db* for tables and columns that do not exist in *schema* are ignored. Columns that exist in *schema* but not in *db* are set to their default values. All of *schema*’s constraints apply in full.

needs−conversion *db schema*
Reads the schema embedded in *db* and the standalone schema in *schema* and compares them. If the schemas are the same, prints **no** on stdout; if they differ, print **yes**.

db−version *db*
schema−version *schema*
Prints the version number in the schema embedded within the database *db* or in the standalone schema *schema* on stdout. A schema version number has the form *x*.*y*.*z*. See **ovs−vswitchd.conf.db**(5) for details.

Schema version numbers and Open vSwitch version numbers are independent.

If *schema* or *db* was created before schema versioning was introduced, then it will not have a version number and this command will print a blank line.

db−cksum *db*
schema−cksum *schema*
Prints the checksum in the schema embedded within the database *db* or of the standalone schema *schema* on stdout.

If *schema* or *db* was created before schema checksums were introduced, then it will not have a checksum and this command will print a blank line.

Other Commands

query *db transaction*
Opens *db*, executes *transaction* on it, and prints the results. The *transaction* must be a JSON array in the format of the **params** array for the JSON-RPC **transact** method, as described in the OVSDB specification.

The *db* is opened for read-only access, so this command may safely run concurrently with other database activity, including **ovsdb−server** and other database writers. The *transaction* may specify database modifications, but these will have no effect on *db*.

transact *db transaction*
Opens *db*, executes *transaction* on it, prints the results, and commits any changes to *db*. The *transaction* must be a JSON array in the format of the **params** array for the JSON-RPC **transact** method, as described in the OVSDB specification.

The *db* is opened and locked for read/write access, so this command will fail if the database is opened for writing by any other process, including **ovsdb−server**(1). Use **ovsdb−client**(1), instead, to write to a database that is served by **ovsdb−server**(1).

show−log *db*
Prints a summary of the records in *db*’s log, including the time and date at which each database change occurred and any associated comment. This may be useful for debugging.

To increase the verbosity of output, add **−m** (or **−−more**) one or more times to the command line. With one **−m**, **show−log** prints a summary of the records added, deleted, or modified by each transaction. With two **−m**s, **show−log** also prints the values of the columns modified by each change to a record.

OPTIONS

Logging Options

−v[*spec*]
−−verbose=[*spec*]
Sets logging levels. Without any *spec*, sets the log level for every module and destination to **dbg**. Otherwise, *spec* is a list of words separated by spaces or commas or colons, up to one from each category below:

- A valid module name, as displayed by the **vlog/list** command on **ovs−appctl**(8), limits the log level change to the specified module.
- **syslog**, **console**, or **file**, to limit the log level change to only to the system log, to the console, or to a file, respectively. (If **−−detach** is specified, **ovsdb−tool** closes its standard file descriptors, so logging to the console will have no effect.)

  On Windows platform, **syslog** is accepted as a word and is only useful along with the **−−syslog−target** option (the word has no effect otherwise).
- **off**, **emer**, **err**, **warn**, **info**, or **dbg**, to control the log level. Messages of the given severity or higher will be logged, and messages of lower severity will be filtered out. **off** filters out all messages. See **ovs−appctl**(8) for a definition of each log level.

Case is not significant within *spec*.

Regardless of the log levels set for **file**, logging to a file will not take place unless **−−log−file** is also specified (see below).

For compatibility with older versions of OVS, **any** is accepted as a word but has no effect.

−v
−−verbose
Sets the maximum logging verbosity level, equivalent to **−−verbose=dbg**.

−vPATTERN:*destination***:***pattern*
−−verbose=PATTERN:*destination***:***pattern*
Sets the log pattern for *destination* to *pattern*. Refer to **ovs−appctl**(8) for a description of the valid syntax for *pattern*.

−vFACILITY:*facility*
−−verbose=FACILITY:*facility*
Sets the RFC5424 facility of the log message. *facility* can be one of **kern**, **user**, **mail**, **daemon**, **auth**, **syslog**, **lpr**, **news**, **uucp**, **clock**, **ftp**, **ntp**, **audit**, **alert**, **clock2**, **local0**, **local1**, **local2**, **local3**, **local4**, **local5**, **local6** or **local7**. If this option is not specified, **daemon** is used as the default for the local system syslog and **local0** is used while sending a message to the target provided via the **−−syslog−target** option.

−−log−file[=*file*]
Enables logging to a file. If *file* is specified, then it is used as the exact name for the log file. The default log file name used if *file* is omitted is **/home/joe/git/openvswitch/_run/log/ovsdb−tool.log**.

−−syslog−target=*host***:***port*
Send syslog messages to UDP *port* on *host*, in addition to the system syslog. The *host* must be a numerical IP address, not a hostname.

−−syslog−method=*method*
Specify *method* how syslog messages should be sent to syslog daemon. Following forms are supported:

- **libc**, use libc **syslog()** function. This is the default behavior. Downside of using this options is that libc adds fixed prefix to every message before it is actually sent to the syslog daemon over **/dev/log** UNIX domain socket.
- **unix:***file*, use UNIX domain socket directly. It is possible to specify arbitrary message format with this option. However, **rsyslogd 8.9** and older versions use hard coded parser

function anyway that limits UNIX domain socket use. If you want to use arbitrary message format with older **rsyslogd** versions, then use UDP socket to localhost IP address instead.

- **udp:***ip*:*port*, use UDP socket. With this method it is possible to use arbitrary message format also with older **rsyslogd**. When sending syslog messages over UDP socket extra precaution needs to be taken into account, for example, syslog daemon needs to be configured to listen on the specified UDP port, accidental iptables rules could be interfering with local syslog traffic and there are some security considerations that apply to UDP sockets, but do not apply to UNIX domain sockets.

Other Options

−h

−−help Prints a brief help message to the console.

−V

−−version

Prints version information to the console.

FILES

The default *db* is **/home/joe/git/openvswitch/_run/conf.db**. The default *schema* is **/home/joe/git/openvswitch/_run/share/openvswitch/vswitch.ovsschema**. The **help** command also displays these defaults.

SEE ALSO

ovsdb−server(1), **ovsdb−client**(1), and the OVSDB specification.