

SQL Database Archive and Backup Manager 0.6.0

Technical documentation

Katarzyna Wladyszewska, Hadden Sp.J.

SQL Database Archive and Backup Manager 0.6.0: Technical documentation

by Katarzyna Wladyszewska

Published April 2010

Copyright © 2003-2010 Hadden Sp.J.

HADDEN MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MANUAL, INCLUDING, BUT NOT LIMITED TO, THE WARRANTY OF DESIGN, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

All rights reserved. No part of this document maybe photocopied, reproduced or translated to another language without the prior written consent of Hadden Sp.J.

All trademarks included in this document are the property of their respective owners.

Table of Contents

1. Conventions	1
2. General information about David system	2
2.1. General	2
2.2. David system architecture	3
3. Terminology	6
3.1. Authorization process made by David system products	6
3.2. David system terminology used in the documentation	6
4. Installation	7
4.1. The main configuration file of David system	7
4.2. Dedicated account for service of David system	7
4.3. Directories of David system	8
4.4. Configuration of syslogd daemon	8
5. SQL Database Archive and Backup Manager requirements	9
6. Installation	10
6.1. Installation from the RPM package	10
6.2. Installation from the script	10
7. General	11
7.1. Functionality	11
7.2. Description	11
7.3. Related articles	11
8. SQL Database Archiver (dsqldb)	12
8.1. General	12
8.2. Synopsis	12
8.3. Options	12
8.4. Description	13
8.5. Related articles	13

List of Tables

- 1.1. The typographical conventions used in this manual 1
- 2.1. David system products 3
- 8.1. dsqldb options 12

Chapter 1. Conventions

The following typographical conventions are used in this manual:

Table 1.1. The typographical conventions used in this manual

Font	What the font represents	Example
<i>Italic</i>	Environment variables.	The name is kept in environmental variable <i>\$DAVIDPRIVDIR...</i>
<i>Italic</i>	Synopsis options.	<i>[-l,--log-facility log_facility]</i>
Bold	Names of programs and products.	damcsud is a part of Operation Manager-a .
Computer	Names of options and menus.	There is Show tool bar option in View menu.
Computer	Names of files and directories.	... reads its configuration file <code>.damadbudrc</code> .
Computer	Names of windows and dialog fields.	In A sessions property window, in Sticking string field, you can write...
Computer	Names of buttons.	Pressing Apply button lets you apply changes.
Computer Bold	Math formulas.	$\exp(-x)$, when $a = 0$ $1 / \text{pow}(a, a) * \text{pow}(x, a) * \exp(-x + a)$, when $a > 0$.
Computer Bold	Terms used in David system terminology.	SNMP Data - a kind of data...
Computer Bold	Contents of configurations files.	action { ... }

Chapter 2. General information about David system

2.1. General

David system is a network management system. It is a packet of applications (modules) that allows computer network to be monitored and managed in real-time through the Internet. There is only one condition that managed devices must meet. Each device must provide SNMP (Simple Network Management Protocol) service. SNMP is the most common management protocol in the Internet so that requirement shouldn't be difficult to meet. Here is the list of typical devices that can be monitored:

- IP routers,
- ATM switches,
- manageable ethernet switches,
- UPSes with a SNMP adapter,
- TV-SAT modems that allow IP devices to work in TV cable networks,
- computers.

One of the most important feature of **David system** is its architecture. It's built of high level configurable and independent from one another modules. This principle is the most essential rule of the project. In consequences, in th metter of speaking, the same modules may build different management system. Here are the main features of **David system**:

- general thinking in information flow controlling that come form high level independence of modules of the system,
- high level configureability of the system modules that allows a special configuration of **David system** to reach end-user expectations so close as it's only possible,
- the system scalability, so you can build up the system adding additional modules in very easy way; note that these modules needn't to be part of **David system** at all; adding another monitored devices to the system is a very easy procedure,
- using shell scripts in information processing is opportunity for modeling information and influence on processing it,
- all configuration files of **David system**, files with input/output data and log files are text files,

- using SNMPv1, SNMPv2C and SNMPv3 to communicate with monitored devices.

2.2. David system architecture

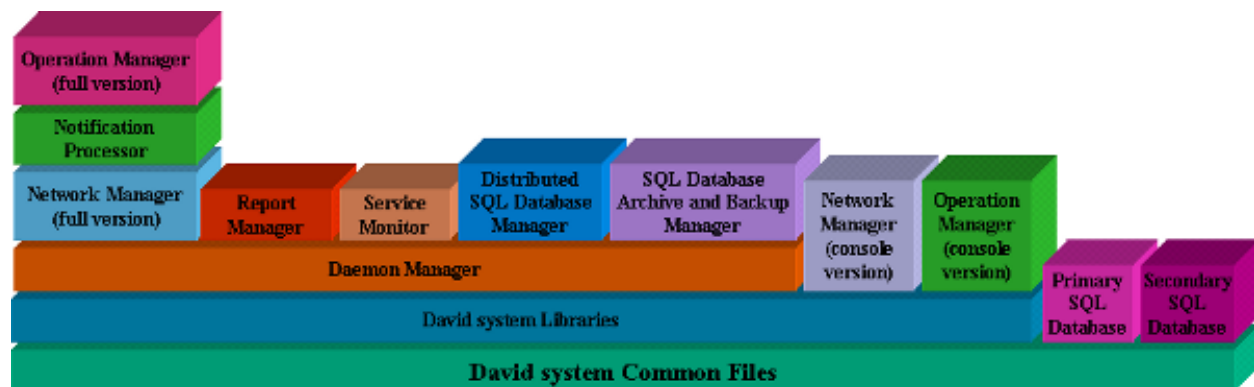
Table 2.1. David system products

Product	Description
David system Common Files	The product, during its installation, prepares the rudimentary directory tree for other products of David system . It also contains some essential and common files for all the products. Thus, this is a fundamental product of David system required by other its products.
Primary SQL Database	The product installs the primary SQL database of David system . Every single installation of David system must have only one the primary database.
Secondary SQL Database	The product installs the secondary SQL database of David system . Each installation of David system may have many secondary databases or none. It allows to distribute the SQL database of David system among many servers.
David system Libraries	This product provides libraries of David system required by its applications. Many other products of David system require that one.
Daemon Manager	It engages in running and terminating daemons of David system as well as monitoring of their work.
Network Manager (full version)	The product using SNMP protocol allows to visualise a topology of monitored networks and auto-discover devices in managed networks. The state of monitored devices also is visualized. The product also collects data from monitored devices using SNMP protocol and allows you to manage user accounts.
Network Manager (console version)	The product, through a graphic application, allows to visualize a topology of monitored networks and shows states of monitored resources. It allows you to control daemons monitoring devices as well as that ones gathering data. Currently, most of functions of that application is obtainable through web applications.
Notification Processor	The product chiefly engages in processing SNMP Trap notifications coming from monitored devices to management stations. The received messages can be formatted to the human readable forms, and then recorded as well. The processed notifications can be passed on to future processing.
Operation Manager (full version)	It can run specified actions on the basis of received data. Sophisticated estimation process depends on information coming from other products of David system and correlation of that information. It tries to build more intelligent and useful notifications then just simple reactions to incoming

General information about David system

Product	Description
	events. The graphic application displays notifications about received events and allows to play audio files as well as reading messages by an outer speech synthesizer.
Operation Manager (console version)	The product contains a graphic application displaying notifications about events and allowing to play audio files as well as reading messages by an outer speech synthesizer.
Report Manager	The product processes recorded SNMP Trap notifications, entries about pending operations and entries about state changes of monitored devices (ping objects, network interfaces and BGP peers), and generates reports on the basis of them. Reports can be viewed using a Web application.
Service Monitor	The product monitors selected network services on application level. In order to do this it monitors selected TCP ports of specified hosts. It checks both availability of ports and a correct reaction for a few selected network protocols (HTTP, SMTP, FTP). It also can verify correctness of work of selected services by verification of received data. Results of its work can be viewed as reports and graphs made available by a Web application.
SQL Database Archive and Backup Manager	It archives the SQL Database used by David system applications.
Distributed SQL Database Manager	It allows to divide the database of David system into one primary database and many secondary ones. Such step boosts performance of the system and decreases load of the servers where daemons of David system work. The migration takes place during the routine work of the system. Such division may be altered many times.

Dependencies between the **David system** products are shown on the following chart..



David system functionality can be very large and it depends on particular configuration a lot. The most important features of **David system** are:

- discovering and visualization of monitored networks topology including visualization of states of

particular nodes;

- possibility of building control panels to monitored devices (they must support SNMP protocol), regardless of device providers;
- formatting and recording SNMP Traps sent by agents working on monitored devices;
- automatic reaction to specified SNMP Traps received from monitored devices;
- possibility of identification of an operator that has received an alert from the system about a problem;
- collecting data concerning parameters of monitored devices;
- automatic reaction to incorrect values of data that were found during data collecting;
- recording pending cases, processed by the system, which have been created as responses for events detected by the system in a monitored network;
- monitoring selected network services on application level.

Chapter 3. Terminology

3.1. Authorization process made by David system products

The modules of David system which need to do an authorization of message senders (i.e. **damsnmpdaud**, **dnmmsd**, **dgnsd**), use the library, that checks whether an IP address of a sender matches with any record found in the file `.known.host`. The library expects to find the file in a directory pointed by a variable `confdir` in the file `/etc/system-david.conf`.

Records in the file `.known.host` are regular expressions specifying acceptable IP addresses.

3.2. David system terminology used in the documentation

There is an explanation of some terms, that are used in David system and its documentation:

- **messages (information)** - data received by interfaces of **Operation Manager**, its data analysers and **Cases Database Unit** of the product.
- **notifications** - the term often is used in the products: **Notification Processor**, **Operation Manager** and **Report Manager**; There are mostly data, that a source are SNMP agents working on network monitored devices.
- **events** - the term often is used in the products: **Operation Manager** and **Report Manager**; and it describes a being, that a source is SNMP Trap or SNMP Data; an **event** is always a part of a **case**;
- **cases** - the term often is used in the products: **Operation Manager** and **Report Manager**; and it describes a group of events connected one another; one **event** at last must be included in a **case**;
- **SNMP Trap** - a kind of data of **Operation Manager** product, which a source are received responses from SNMP agents; SNMP Traps aren't answers on the requests sent by a management station, but they are sent by agents managing network interfaces and processed by **Notification Processor** product;
- **SNMP Data** - a kind of data of **Operation Manager** product, which a source are received responses from SNMP agents on request which a management station sent to them by **Network Manager**.

Chapter 4. Installation

4.1. The main configuration file of David system

The essential configuration file of David system is `/etc/david-system.conf`. It contains entries as pairs: `key = value`. Basically, except the entry `default_email_recipient`, there is no such need to modify any record in that file. All necessary modifications are made during installation processes of particular David system products. Below, there is a list of all entries along with their descriptions that may occur in this basic configuration file.

- `user` - a name of the user with which rights all daemons of David system works;
- `default_email_recipient` - the default e-mail address where messages from David system applications are sent;
- `bindir` - the directory containing David system applications (default: `/usr/bin/david-system`);
- `libdir` - the directory containing David system libraries (default: `/usr/lib/david-system`);
- `incdir` - the directory containing David system headers (default: `/usr/include/david`);
- `confdir` - the directory containing David system configuration files (default: `/etc/david-system`);
- `logdir` - the directory containing log files of David system applications (default: `/var/log/david-system`);
- `sharedir` - the directory containing various files (images, audio files, web files) of David system (default: `/usr/share/david-system`);
- `docdir` - the directory containing various files (images, audio files, web files) of David system (default: `/usr/share/david-system`);
- `vardir` - the directory containing archive files of David system SQL database (default: `/var/lib/david-system`);
- `is_sqldb_installed` - the flag that indicates whether the SQL database of David system has been installed or not.

4.2. Dedicated account for service of David system

There is no need to run any David system module as superuser (usually an account `root` with UID equals 0). Even if some David system daemon requires root rights when starting, there is always possibility to specify, as one of the daemons starting arguments, a user that rights should be taken.

It is a good idea to add a new user to an operating system, under which control David system will work.

4.3. Directories of David system

This hierarchy depends on a particular configuration of David system. In the default system configuration, David system contains the following directories:

- `/usr/bin/david-system` - binaries and shell scripts;
- `/etc/david-system` - configuration files;
- `/usr/share/doc/david-system` - the documentation;
- `/usr/share/david-system` - graphic and audio files, web portal;
- `/usr/include/david` - David system header files;
- `/usr/lib/david-system` - David system libraries;
- `/var/log/david-system` - log files;
- `/var/lib/david-system` - archive files of the David system SQL database;

4.4. Configuration of syslogd daemon

David system modules use `syslog` subsystem available on UNIX platforms. Default configuration of the system modules causes that log messages are sent with `local6` facility. It may be changed for every module during its startup. Its recommended to configure `syslogd` daemon to write all messages from David system modules into one place (one or more files with characteristic name i.e.: `david.log`).

Chapter 5. SQL Database Archive and Backup Manager requirements

The following requirements must be met by a management platform on which **SQL Database Archive and Backup Manager** will work:

- installed, compatible version of **Daemon Manager**.

Chapter 6. Installation

6.1. Installation from the RPM package

You must be `root` to install the product. The typical installation looks as this one following below:

- Install the product:

```
rpm -i david-xxx-sqldbabm-yyy.rpm
```

6.2. Installation from the script

You must be `root` to install the product. The typical installation looks as this one following below:

- Uncompress and unpack the archive:

```
gunzip david-xxx-sqldbabm-yyy.i386.tar.gz  
tar xf david-xxx-sqldbabm-yyy.i386.tar
```

The operations create `david-xxx-sqldbabm-yyy.i386` directory in your current directory.

- Change your current directory to `david-xxx-sqldbabm-yyy.i386`:

```
cd david-xxx-sqldbabm-yyy.i386
```

- Read `LICENSE` file from the current directory and `CONTINUE THE INSTALLATION, ONLY WHEN YOU ACCEPT ALL CONDITIONS INCLUDED IN THE LICENSE.`
- Run the installation script:

```
./install
```

Chapter 7. General

7.1. Functionality

SQL Database Archive and Backup Manager makes possible:

- archiving old data created by David system modules and stored in SQL tables;
- notifying the administrator that a new archive is available.

7.2. Description

SQL Database Archive and Backup Manager archives old data from SQL tables that store historical data. The data allows to draw graphs, report logs etc. **SQL Database Archive and Backup Manager** prepares archives and next informs the administrator that the new archive is ready. The administrator may record such archive on CD or DVD.

7.3. Related articles

[SQL Database Archiver \(dsqldbaid\)](#)

Chapter 8. SQL Database Archiver (dsqldbdbad)

8.1. General

dsqldbdbad program is **SQL Database Archiver** and it is a part of **SQL Database Archive and Backup Manager**. It is a daemon process which works all the time the system is running. It archives old data from SQL tables.

8.2. Synopsis

dsqldbdbad can be run with the following options: [-P,--pid-file *filename*] [-l,--log-facility *log_facility*] [-L,--log-level *log_level*] [-u,--run-as-user *username*] [--background] [--max-os-load *number*] [--max-num-days *number*] [--max-num-days-4-weekly *number*] [--max-num-days-4-monthly *number*] [--max-num-days-4-yearly *number*] [--min-rows-2-move *number*] [--max-rows-2-move *number*] [--min-rows-2-dump-table *number*] [--archive-dir *dirname*] [--max-archive-dir-mb *size*] [--compress-cmd *command*] [--compressed-file-suffix *string*] [--notification-cmd *command*] [--min-free-space-mb *size*] [-v,--version] [-h,--help]

8.3. Options

Table 8.1. dsqldbdbad options

Option name	Description
-P,--pid-file <i>filename</i>	Write PID to the specified file.
-l,--log-facility <i>log_facility</i>	Choose log facility: daemon user local0 ... local7 (default: local6).
-L,--log-level <i>log_level</i>	Choose logging level (on stderr and syslog): emerg alert crit err warning notice info debug0 ... debug2 (default: warning).
-u,--run-as-user <i>username</i>	Drop root privileges and run the server as the specified user.
--background	Go to background after startup.
--max-os-load <i>number</i>	Set maximum OS load that the normal work is allowed to be continued (default: 44).
--max-num-days <i>number</i>	Set maximum number of days when the data are still treated as current and not archived (default: 180).
--max-num-days-4-weekly <i>number</i>	Set maximum number of days when the data in weekly tables are still treated as current and not archived (default:365*2).
--max-num-days-4-monthly <i>number</i>	Set maximum number of days when the data in monthly tables are still treated as current and not archived (default:365*5).
--max-num-days-4-yearly	Set maximum number of days when the data in yearly tables are still

Option name	Description
<i>number</i>	treated as current and not archived (default:365*10).
<i>--min-rows-2-move number</i>	Set minimum number of rows to be moved from an original table to a temporary one (default:10000).
<i>--max-rows-2-move number</i>	Set maximum number of rows that can be moved from an original table to a temporary one as a single operation (default: 100000).
<i>--min-rows-2-dump-table number</i>	Set minimum number of rows in a temporary table when it is allowed to be dumped into the file (default: 1000000).
<i>--archive-dir dirname</i>	Set name of a top directory where all dump files will be created (default: dump).
<i>--max-archive-dir-mb size</i>	Set maximum size in MB (MB=1048576 bytes) of all files archived in a single archive directory (default:700MB).
<i>--compress-cmd command</i>	Use this command to compress SQL table dump files (default: gzip).
<i>--compressed-file-suffix string</i>	Expect files with such suffix as compressed after running the compress command (default: .gz).
<i>--notification-cmd command</i>	Use this command to notify the user that a new archive is completed (default: /usr/bin/david-system/sqldbdbm/new-archive-notification.sh).
<i>--min-free-space-mb size</i>	Set minimum of required free space in MB (on filesystem that contains the archive directory; default: 2048).
<i>-v,--version</i>	Display version number and exit.
<i>-h,--help</i>	Display these informations and exit.

8.4. Description

dsqldbdb program archives old data from the SQL tables, that store historical data. The data allows to draw graphs, report logs, etc. **dsqldbdb** program moves old data to temporary tables, that it creates automatically. If a number of rows in a temporary table exceeds a fixed value, the table is dumped to a file. Then, the file is compressed. If a sum of dumped files exceeds a fixed value, the all dumped files are moved to a new directory, that performs the function of archive. The user is noticed, that a new archive is ready. He can record the whole directory on CDs or DVDs and delete it.

8.5. Related articles

Network Manager: Slow Query Executor (dsqed)