Operation Manager 0.27.0 (console version)

Technical documentation

Katarzyna Wladyszewska, Hadden Sp.J.

Operation Manager 0.27.0 (console version): Technical documentation

by Katarzyna Wladyszewska

Table of Contents

List of Tables

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
Italic	Environment variables.	The name is kept in environmental variable *$DAVIDPRIVDIR*...
Italic	Synopsis options.	[*-l,--log-facility log_facility*]
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 `.damadbudrc`.
`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 / pow( a , a ) * 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 configureable 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

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 speach 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 speach 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 raports on the basis of them. Raports 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 devide 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 rutine work of the system. Such division may be altered many times.

Dependences between the **David system** products are shown on the following chart:.
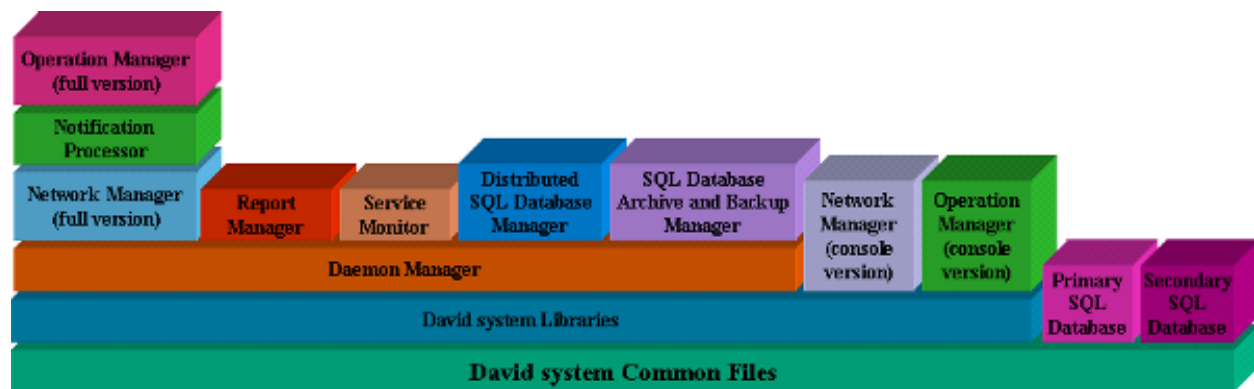
David system functionality can be very large and it depends on particular configuration a lot. The most important features of **David system** are:

- 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:

- `massages (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 in `/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 indicate whether the SQL database of David system has been installed or not.

4.2. Dedicated account for service of David system

There is no needs 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. Operation Manager requirements

The following requirements must be met by a management platform on which **Operation Manager** will work:

- installed, compatible version of **David system Libraries**.

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-om-c-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-om-c-yyy.i386.tar.gz
tar xf david-xxx-om-c-yyy.i386.tar
```

The operations create david-xxx-om-c-yyy.i386 directory in your current directory.

- Change your current directory to david-xxx-om-c-yyy.i386:

```
cd david-xxx-om-c-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 information

7.1. Functionality

Operation Manager makes possible:

- notifing an operator about received massages, pending cases, and recording user's identyfier and his/her reaction time.

7.2. Description

The main **Operation Manager** goal is notifing an operator, through graphic windows and sound signals, about processed events and pending cases (i.e.: power failures, connectivity breaks between devices, device restarts etc.).

7.3. Related articles

Graphic Notifications Presenter (xdgnp)

Chapter 8. Graphic Notifications Presenter (xdgnp)

8.1. General

xdgnp application is **Graphic Notifications Presenter** and it is a part of **Operation Manager**. The application allows to display graphic notifications concerning cases about which full information is received from dgnsd server and react an operator to these notifications. It can play also sound files and read text.

8.2. Synopsis

xdgnp can be run with the following options: [*-l,--log-facility log_facility*] [*-L,--log-level log_level*] [*-v,--version*] [*-h,--help*]

8.3. Options

Table 8.1. xdgnp options

Option name	Description
-l,--log-facility log_facility	Choose log facility: daemon | user | local0 | ... | local7 (default: local6).
-L,--log-level log_level	Choose log level (on stderr and syslog) i.e. messages of selected level and more important levels will be logged: emerg | alert | crit | err | warning | notice | info | debug0 | ... | debug2 (default: warning).
-v,--version	Display version number on stderr and exit.
-h,--help	Display this help and exit.

8.4. Description
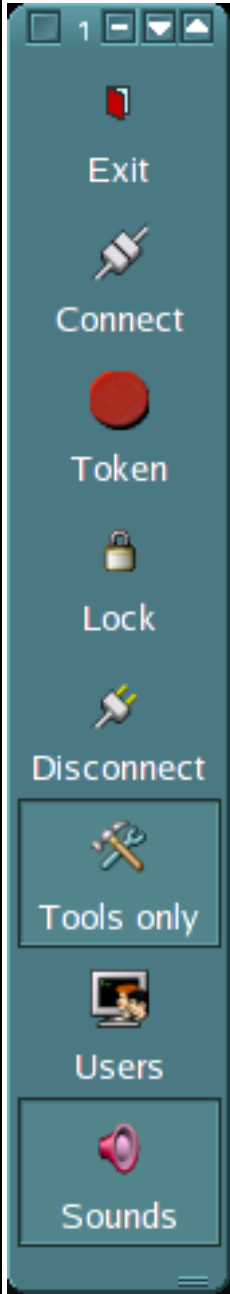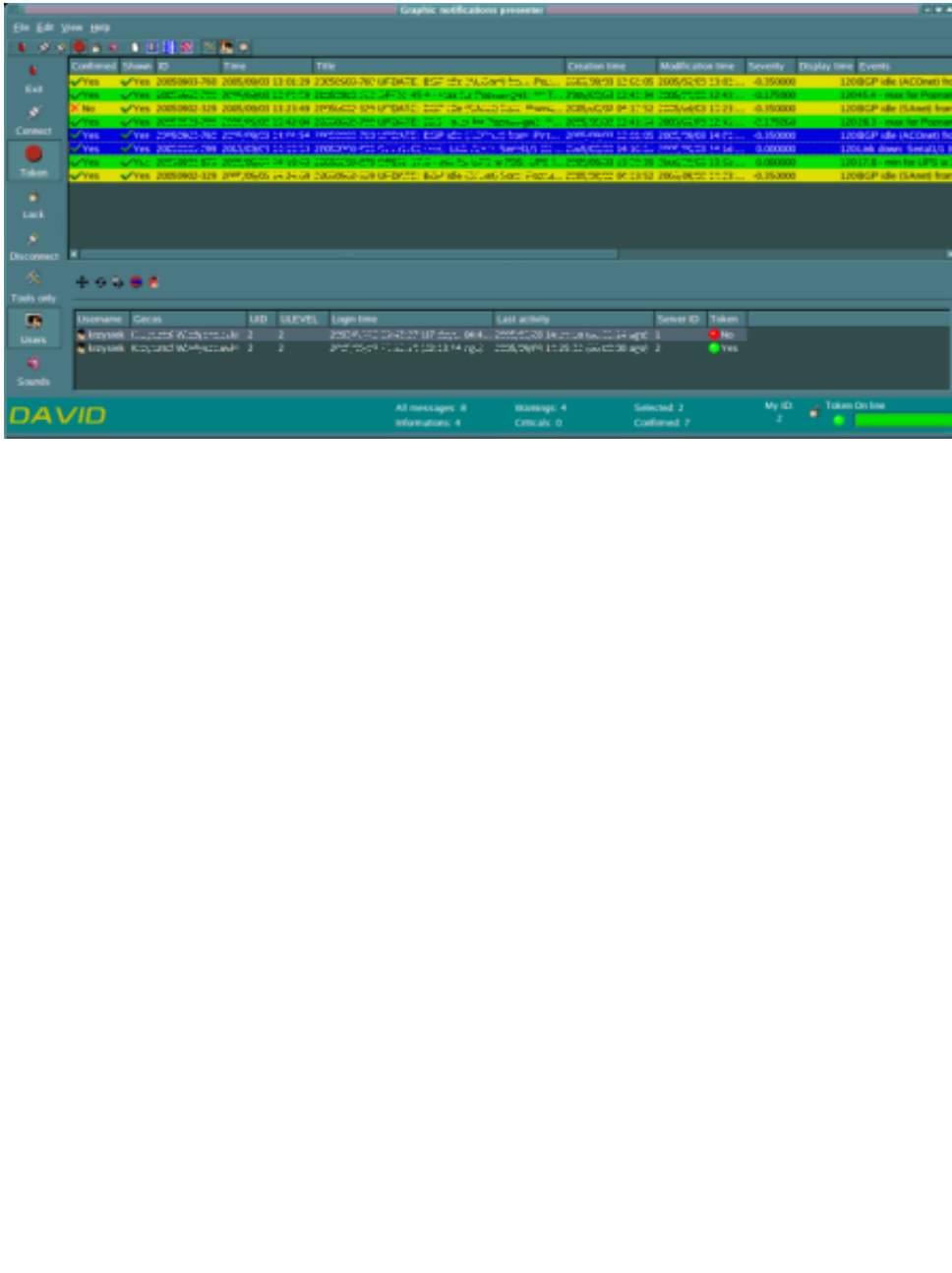
8.4.1. Starting up and terminating the aplication

xdgnp reads its configuration parameters from `.xdgnprc` file during its startup. These parameters concern appearance of the program and other working parameters. The application expects to find its configuration file in a directory which name is kept in environmental variable *$DAVIDPRIVDIR*. When such file doesn't exist, the application begins its work with its default settings.

8.4.2. Main view work

The application may work in two view modes. Default mode of view is a full view of the application with the toolbar, the status bar and the list of received notifications. The second mode is only view of the tool panel and in this way the application doesn't occupy so much place on the desktop. It's very easy to switch between these two modes of view during the application work.

Table 8.2. xdgnp - modes of work

A toolbar mode	A full view mode

The full view in its central part shows a list of received notifications. Meaning of particular columns is explained below:

Table 8.3. xdgnp - columns description

Column name	Description
Confirmed	Whether a notification window has been closed by an operator and thereby receiving of the massage has been confirmed.
Shown	Whether a notification window has been shown or reason of failure otherwise.
ID	An unique identifier of a case.
Time	Notification receiving time.
Creation time	Case creation time.
Modification time	Case modification time.
Severity	Severity of a case.
Display time	Maximum time the notification window could be displayed.
Title	A title of a case.
Events	Events included in a case.

The notification list counters are displayed on the status bar. These counters show: a number of all messages, a number of information massages, a number of warnings, a number of critical notifications, a number of selected items and a number of confirmed notifications. Next, an identifier of this application assigned by dgnsd server is displayed. Next there are the buttons 🔒 and 🔓.

Next, there is a control light which signals the application owns the token on dgnsd server. The last item on the status bar is a connection state indicator.

The tool panel which buttons mostly correspond to buttons on the toolbar is displayed on the most left side of the application. `Tools only` button is the only one button located only on the tool panel and not placed on the toolbar. It switches the application between both kinds of its view.

Pressing the right mouse button above the list of received notifications shows the all menus accessible also on the top of the application. Pressing the left mouse button over an item of the list selects or unselects a given item. The list allows you to select many items at the same time.

8.4.2.1. Main view buttons

Buttons on the toolbar allow you to control the application work. The first five buttons from the left side correspond with the options of `File` menu.

Table 8.4. xdgnp - description of the buttons

Button	Description
	It lets you exit the application.
	It lets you connect to **dgnsd** server.
	It lets you close the connection (logout).
	It allows you to take the token.
	It allows you to lock access to the application. Locking the application causes appearing of `Authorization`, dialog in case of any mouse move or any mouse button or any keyboard key pressing. In this dialog you should enter a user name and his/her password to unlock access to the application and login to the server again (you needn't login as the same user).
	Turn on/off a sound service.

The next buttons correspond to `Edit` and `View` menu.

Table 8.5. The buttons correspond to Edit and View menu

Button	Description
	It lets you delete all items of the notification list.
	It lets you delete selected items.
	It lets you reverse selection of the notification list.
	It lets you unselect all previously selected items.
	It allows you to show or hide the tool panel.
	It lets you open the edition panel of comments to received notifications.
	It lets you open the list of logged in users on **dgnsd** server.
	It allows you to configure parameters of the application.

Additionally, in `View` menu two options are placed - `Show tool bar` and `Show status bar`. They allow you to show or hide the toolbar and the status bar.

Help menu may inquire about a version and a creation time of the application.

8.4.3. List of logged in users on dgnsd server
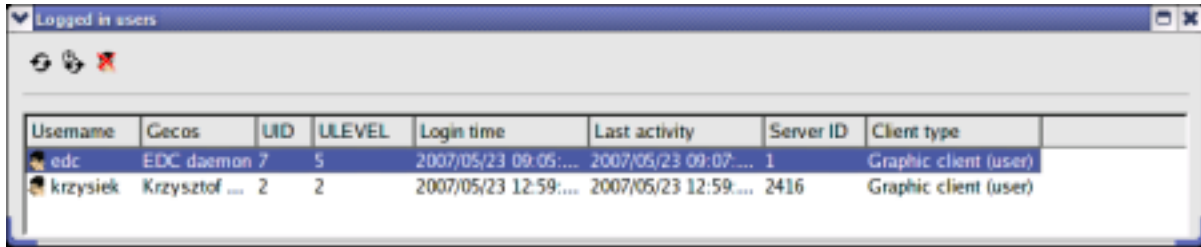
The dialog with a list of logged in users on **dgnsd** server allows to make some actions on these users. A set of buttons available on this dialog is assigned to that. The buttons are described in the chart below:

Table 8.6. dgnsd server - description of the dialog buttons

Button	Description
✛	It allows you to change the dialog into the window docked at the main view of the application and vice versa.
↻	It refreshes the list of logged in users.
⟳	It turns on/off auto-refreshing of the list.
⬤	It lets you grant the token a selected user of the list.
✗	It lets you logout the selected user.

The rules of granting the token and logging out the users are described in the chapter `Description` of dgnsd server.

Meaning of particular columns of the list is explained below:

Table 8.7. Columns description

Column	Description
Username	User's name.
Gecos	Account description.
UID	User's identifier (number).
ULEVEL	Level of user's rights.
Login time	Login time as a specified user (it needn't be equal the connection time in case the authentication has been done more then once).
Last activity	Time of last activity of a user (this instance of the application) on dgnsd server understood as sending any message to the server.
Server ID	An identifier of a given instance of the application on dgnsd server (the same user may be logged in running more then one instance of **xdgnp**).

| Column	Description
Token	Indicates whether a given user (exactly: that instance of the application) has the token.

8.4.4. xdgnp configuration

Pressing the button ![icon] opens the configuration dialog. The window is divided into three configuration areas.

8.4.4.1. dgnsd parameters configuration
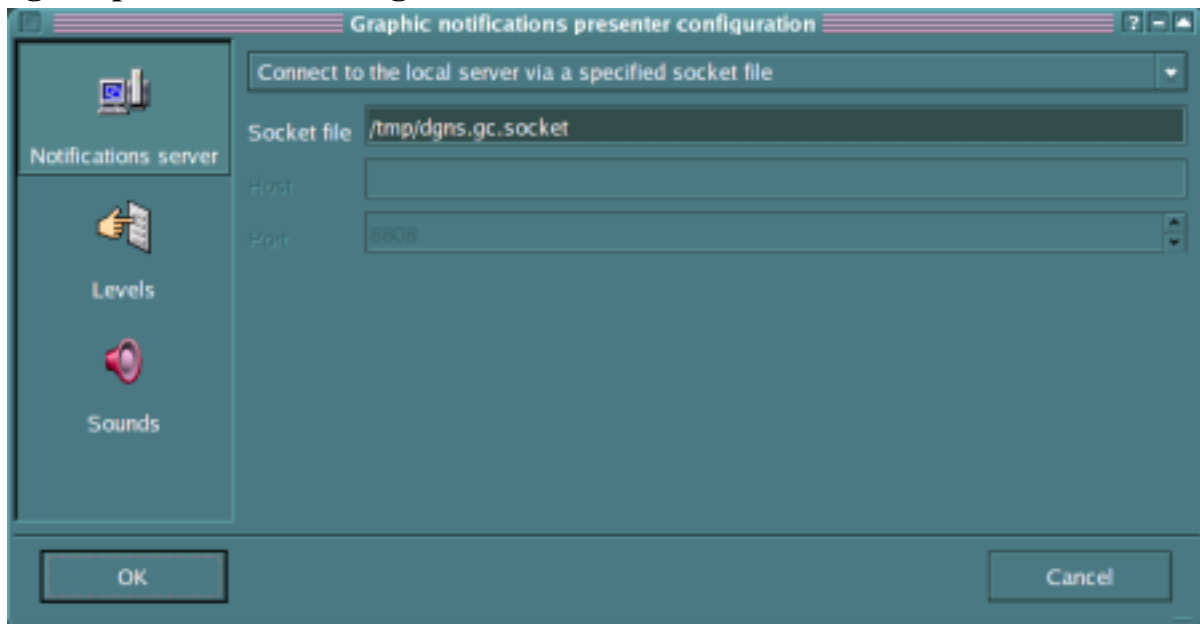
There's a list of two items at the top of the window which lets you choose a type of connection with dgnsd server. The first option lets you choose the local connection via a socket file that name you may enter below. The second one indicates that you want to make the connection via the network. In this case you should enter a host name or its IP address and a number of TCP port, in which the server listens for.

## 8.4.4.2. Work levels configuration
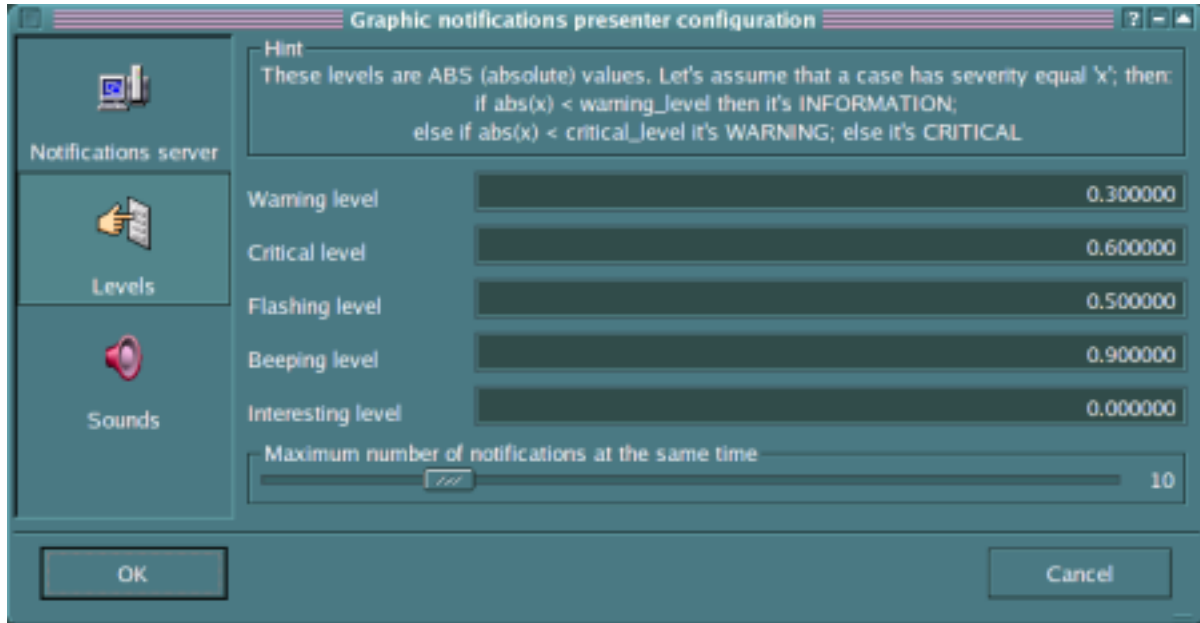
This tab lets you set levels of severities for incoming notifications. All levels are treated as absolute values i.e.: if severity of a given notification equals x, and if

```
abs(x) < warning_level
```

then the notification is recognized as information. Parameters `Warning level` and `Critical level` determine severity of notifications. Parameter `Flashing level` indicates level from which the notification title fleshes during displaying a notification window while `Beeping level` value tells you from which severity value the window beeps. Parameter `Interesting level` shows a minimum level of severity below that notifications are not displayed as a notification window. Even then the notification is added as a new item to the received notification list in the main view of application. `Maximum number of notifications at the same time` group lets you limit number of notification dialogs shown at the same time.

8.4.4.3. Sound configuration in xdgnp application
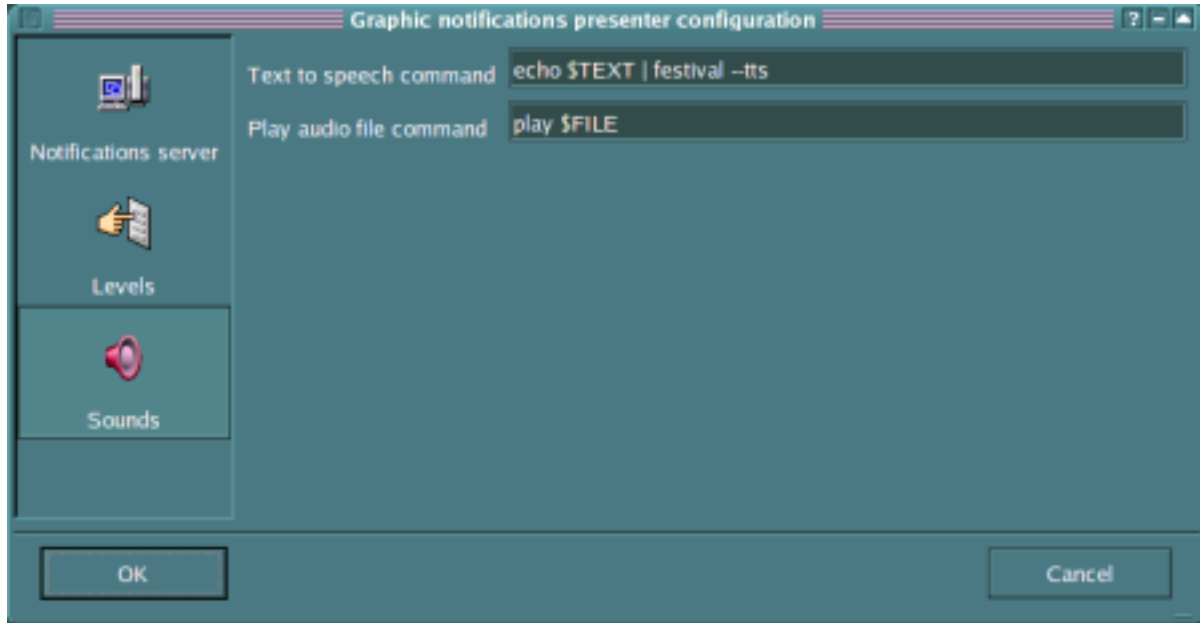
The tab allows you to specify adequate commands, that are run to play sound files and text messages. In `Text to speech command` field a command specification is taken place, that runs the speech synthesizer. In `Play audio file command` a command is specified, that helps to play sound files.

8.4.4.4. A notification window

The graphic window displaying a notification shows on its title bar number of a case that generated the notification. At the top of the window there is a text describing severity of the message (INFORMATION, WARNING, CRITICAL). The severity depends on two parameters: severity of the message set by the client and levels of severities set by the user of**xdgnp** application. Below the text there is a title of the message and next there is a table that displays all `events` included in the `case`. Below the table there is `Close` button that lets you close the window which is equivalent to a confirmation of

receiving of a given notification. At the bottom of the window the time remaining to closing up the window is displayed.

Each row of the table presents a particular event included in the case. Particular columns describe:

- `Status` - state of an event (three possibilities: `Active` (A), `Passive` (H), `Not managed here` (NM)). You may change the state from active to passive and vice versa;

- `Message` - human readable description of an event;

- `Creation time` - event creation time (in this case);

- `Modification time` - last appearing time of an event;

- `Hits numbers` - number of hits of an event since its creation time;

- `Successors` - a list of successors (another events) of that event placed one by one.

8.5. Related articles

Notification Processor: Information Recorder (dsi)

Graphic Notifications Server (dgnsd)