

Operation Manager 0.27.0 (full version)

Dokumentacja techniczna

Katarzyna Władyszewska, Hadden Sp.J.

Operation Manager 0.27.0 (full version): Dokumentacja techniczna

by Katarzyna Władyszewska

Data wydania Kwiecień 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 may be 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.

FIRMA HADDEN NIE PONOSI ŻADNEJ ODPOWIEDZIALNOŚCI ZA SKUTKI WYNIKAJĄCE Z UŻYWANIA NINIEJSZEJ DOKUMENTACJI.

Wszystkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszego dokumentu w jakiegokolwiek postaci jest zabronione.

Wszystkie nazwy handlowe i towarów występujące w niniejszej dokumentacji są znakami towarowymi zastrzeżonymi lub nazwami zastrzeżonymi odpowiednich firm odnośnych właścicieli.

Spis treści

1. Konwencje typograficzne	1
2. Informacje ogólne o systemie David	2
2.1. Charakterystyka ogólna	2
2.2. Architektura systemu David	3
3. Terminologia	6
3.1. Autoryzacja dokonywana przez moduły systemu David	6
3.2. Terminy używane w systemie David i jego dokumentacji	6
4. Instalacja	8
4.1. Główny plik konfiguracyjny systemu David	8
4.2. Dedykowane konto dla obsługi systemu David	8
4.3. Układ katalogów systemu David	9
4.4. Konfiguracja demon-a syslogd	9
5. Wymagania dla Operation Manager-a	10
6. Instalacja	11
6.1. Instalacja z pakietu RPM	11
6.2. Instalacja za pomocą skryptu	11
7. Informacje ogólne	12
7.1. Funkcjonalność	12
7.2. Opis działania	12
7.3. Tematy pokrewne	13
8. Serwer Powiadomień Graficznych (dgnsd)	14
8.1. Opis ogólny	14
8.2. Składnia	14
8.3. Znaczenie opcji w programie dgnsd	14
8.4. Opis działania	15
8.5. Tematy pokrewne	16
9. Klient Dźwięku (sndc)	17
9.1. Opis ogólny	17
9.2. Składnia	17
9.3. Znaczenie opcji w programie sndc	17
9.4. Opis działania	18
9.5. Tematy pokrewne	18
10. Dostęp do Interfejsu Danych SNMP Trap (damsnmpti)	19
10.1. Opis ogólny	19
10.2. Składnia	19
10.3. Znaczenie opcji w programie damsnmpti	19
10.4. Opis działania	20
10.5. Tematy pokrewne	20
11. Analizator Danych SNMP Trap (damsnmptaud)	22
11.1. Opis ogólny	22

11.2. Składnia	22
11.3. Znaczenie opcji w programie damsnmptaud	22
11.4. Format pliku konfiguracyjnego	23
11.5. Opis działania	24
11.6. Tematy pokrewne	25
12. Dostęp do Interfejsu Danych SNMP Data (damsnmpdi)	26
12.1. Opis ogólny	26
12.2. Składnia	26
12.3. Znaczenie opcji w programie damsnmpdi	26
12.4. Opis działania	27
12.5. Tematy pokrewne	27
13. Analizator Danych SNMP Data (damsnmpdaud)	28
13.1. Opis ogólny	28
13.2. Składnia	28
13.3. Znaczenie opcji w programie damsnmpdaud	28
13.4. Format pliku konfiguracyjnego	30
13.5. Format pliku definiującego poziomy minimalne	31
13.6. Opis działania	32
13.7. Tematy pokrewne	35
14. Jednostka Tworząca Bazę Danych Aktywnych Spraw (damadbud)	36
14.1. Opis ogólny	36
14.2. Składnia	36
14.3. Znaczenie opcji w programie damadbud	36
14.4. Format pliku konfiguracyjnego	37
14.5. Opis działania	38
14.6. Tematy pokrewne	39
15. Serwis Aktywnych Spraw (damcsud)	40
15.1. Opis ogólny	40
15.2. Składnia	40
15.3. Znaczenie opcji w programie damcsud	40
15.4. Format pliku konfiguracyjnego	41
15.5. Opis działania	43
15.6. Uruchamianie akcji (programów)	43
15.7. Informacje przekazywane uruchamianym akcjom (programom)	44
15.8. Wyniki działania zakończonych akcji (programów)	46
15.9. Niszczanie spraw	47
15.10. Tematy pokrewne	47
16. Przyciski najczęściej występujące w aplikacjach WWW	48
16.1. Znaczenie przycisków	48
17. Przeglądarka Zarejestrowanych Spraw (Recorded Operation Browser)	50
17.1. Opis ogólny	50
17.2. Opis działania	50
17.2.1. Specyfikacja kryteriów wyszukiwania	50
17.2.2. Wygenerowany raport	52
17.3. Tematy pokrewne	53

18. Przeglądarka Toczących się Spraw (Pending Operation Browser)	55
18.1. Opis ogólny	55
18.2. Opis działania	55
18.2.1. Widok aktywnych spraw	55
18.2.2. Widok gromadzonych danych SNMP Data	58
18.3. Tematy pokrewne	62
19. Prezenter Powiadomień Graficznych (xdgnp)	63
19.1. Opis ogólny	63
19.2. Składnia	63
19.3. Znaczenie opcji w programie xdgnp	63
19.4. Opis działania	63
19.4.1. Rozpoczynanie i kończenie pracy	63
19.4.2. Opis działania głównego okna aplikacji	64
19.4.3. Okienko prezentujące listę zalogowanych użytkowników na serwerze dgnsd ...	68
19.4.4. Konfiguracja aplikacji xdgnp	69
19.5. Tematy pokrewne	72

Spis tabel

1.1. Konwencje typograficzne użyte w dokumencie	1
2.1. Produkty wchodzące w skład systemu David	3
8.1. Znaczenie opcji w programie dgnsd	14
9.1. Znaczenie opcji w programie sndc	17
10.1. Znaczenie opcji w programie damsnmpti	19
11.1. Znaczenie opcji w programie damsnmptaud	22
12.1. Znaczenie opcji w programie damsnmpdi	26
13.1. Znaczenie opcji w programie damsnmpdaud	28
14.1. Znaczenie opcji w programie damadbud	36
15.1. Znaczenie opcji w programie damcsud	40
15.2. Parametry uruchamianych akcji (programów)	44
15.3. Argumenty występujące dla każdego zdarzenia wchodzącego w skład danej sprawy	45
15.4. Argumenty występujące dla zdarzenia typu snmptrap	45
15.5. Argumenty występujące dla zdarzenia typu snmpdata	46
16.1. Przyciski najczęściej występujące w aplikacjach WWW	48
17.1. Przeglądarka Zarejestrowanych Spraw - znaczenie pól widoku szczegółowego grupy Cases	51
18.1. Przeglądarka Toczących się Spraw - znaczenie pól grupy Collected data	59
19.1. Znaczenie opcji w programie xdgnp	63
19.2. Aplikacja xdgnp - tryby pracy	64
19.3. Aplikacja xdgnp - opis poszczególnych kolumn	65
19.4. Aplikacja xdgnp - opis przycisków	66
19.5. Aplikacja xdgnp - opis przycisków funkcyjnych	67
19.6. Opis przycisków aplikacji xdgnp pokrywających się z menu Edit i menu View	67
19.7. Opis przycisków okienka ukazującego listę zalogowanych użytkowników na serwerze dgnsd	68
19.8. Opis kolumn listy użytkowników	68

Rozdział 1. Konwencje typograficzne

Następujące konwencje typograficzne są użyte w niniejszym dokumencie:

Tabela 1.1. Konwencje typograficzne użyte w dokumencie

Czcionka	Znaczenie	Przykład
<i>Kursywa</i>	Nazwy zmiennych środowiskowych	Nazwa pliku przechowywana jest w zmiennej środowiskowej <i>\$DAVIDPRIVDIR...</i>
<i>Kursywa</i>	Opcje składni.	<i>[-l,--log-facility log_facility]</i>
Pogrubiona	Nazwy programów, aplikacji i produktów.	Program damcsud jest częścią Operation Manager-a .
Rozstrzelona	Nazwy opcji i menu.	W menu View znajduje się także opcja Show tool bar.
Rozstrzelona	Nazwy plików i katalogów.	... czyta swój plik konfiguracyjny <code>.damadbudrc</code> .
Rozstrzelona	Nazwy okienek i pól w okienkach dialogowych.	W okienku A sessions property w polu Sticking string podaje się tekst...
Rozstrzelona	Nazwy przycisków.	Przez naciśnięcie przycisku Tab z klawiatury możesz uzyskać focus.
Rozstrzelona pogrubiona	Wzory matematyczne.	$\exp(-x)$, gdy $a = 0$ $\frac{1}{\text{pow}(a, a)} * \text{pow}(x, a) * \exp(-x + a)$, gdy $a > 0$.
Rozstrzelona pogrubiona	Terminy użyte w terminologii systemu David.	SNMP Data - rodzaj danych występujących...
Rozstrzelona pogrubiona	Zawartość plików konfiguracyjnych.	action { ... }

Rozdział 2. Informacje ogólne o systemie David

2.1. Charakterystyka ogólna

System David to system zarządzania siecią komputerową. Jest on pakietem programów (modułów) pozwalającym zdalnie, tzn. poprzez rozległą sieć komputerową (np.: Internet), monitorować i zarządzać, w czasie rzeczywistym, urządzeniami pracującymi w sieciach komputerowych. Jedynym warunkiem, jakie muszą spełniać urządzenia jest, aby pracował na nich agent SNMP (Simple Network Management Protocol). Wobec faktu, że SNMP jest najbardziej rozpowszechnionym protokołem zarządzania na świecie, wymaganie to nie jest szczególnie trudne do spełnienia. Wiele urządzeń posiada oprogramowanie, które pozwala się z nimi komunikować poprzez protokół SNMP. Do urządzeń tych należą m.in.:

- routery IP,
- switche ATM-owe,
- zarządzalne switche ethernetowe,
- UPS-y wyposażone w adaptory SNMP,
- modemy telewizyjnych sieci kablowych pozwalające pracować urządzeniom IP w sieciach telewizji kablowej,
- stacje komputerowe.

Jedną z głównych cech **systemu David** jest fakt, że jest on złożony z wysoce konfigurowalnych i dalece niezależnych od siebie modułów. Staranność o utrzymanie takiego sposobu projektowania systemu jest widoczna od początku jego powstania. W konsekwencji, z tych samych modułów, można zbudować istotnie różniące się w działaniu konfiguracje **systemu David**. Jako jego główne cechy można więc wymienić:

- ogólność w podejściu do sterowania przepływem informacji wynikająca z wysokiej niezależności od siebie modułów systemu,
- wysoka konfigurowalność wszystkich modułów systemu pozwalająca maksymalnie zbliżyć się do oczekiwanego rezultatu podczas konfigurowania pracy systemu,
- skalowalność systemu, tzn. system można łatwo rozbudować dodając kolejne moduły, nawet nie wchodzące w skład **systemu David**, a także bez trudu można poszerzać listę monitorowanych urządzeń,
- wykorzystanie skryptów shell'owych w strumieniu przepływu informacji pozwala w łatwy sposób

formatować i wpływać na przetwarzane informacje,

- wszystkie pliki konfiguracyjne **systemu David**, a także pliki z danymi wejściowymi jak i wyjściowymi, pliki z istotnymi dla systemu komunikatami (log files), są plikami tekstowymi,
- komunikacja z monitorowanymi urządzeniami poprzez protokoły SNMPv1, SNMP 2C i SNMPv3.

2.2. Architektura systemu David

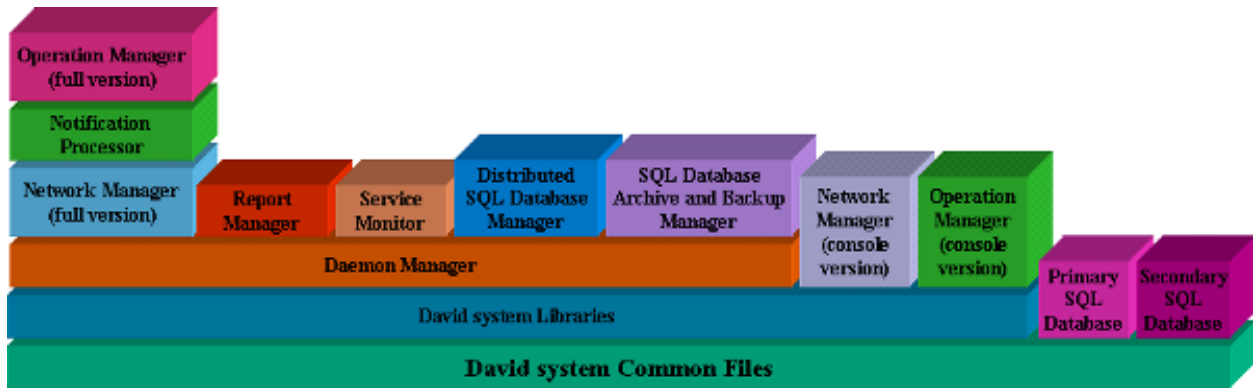
Tabela 2.1. Produkty wchodzące w skład systemu David

Nazwa produktu	Opis
David system Common Files	Produkt podczas instalacji przygotowuje układ katalogów dla innych produktów systemu David . Zawiera też kilka podstawowych plików dla wszystkich innych produktów. Jest to podstawowy produkt systemu David wymagany przez resztę jego produktów.
Primary SQL Database	Produkt instaluje podstawową SQL-ową bazę danych systemu David . Każda instalacja systemu David musi mieć tylko jedną podstawową bazę danych.
Secondary SQL Database	Produkt instaluje dodatkową SQL-ową bazę danych systemu David . Może istnieć wiele dodatkowych baz danych w jednej instalacji systemu David . Pozwala to rozpraszać bazę danych systemu David na wiele serwerów.
David system Libraries	Dostarcza biblioteki systemu David potrzebne aplikacjom systemu. Produkt ten jest wymagany przez wiele innych produktów systemu David .
Daemon Manager	Zajmuje się uruchamianiem i zatrzymywaniem demonów systemu David oraz monitorowaniem ich pracy.
Network Manager (full version)	Produkt poprzez wykorzystanie protokołu SNMP umożliwia wizualizację topologii monitorowanych sieci i automatyczne odkrywanie urządzeń w zarządzanych sieciach. Wizualizacji podlega także stan monitorowanych obiektów. Produkt zajmuje się także zbieraniem danych dotyczących pracy monitorowanych urządzeń, wykorzystując do tego celu protokół SNMP oraz pozwala administrować kontami użytkowników.
Network Manager (console version)	Produkt poprzez graficzną aplikację umożliwia wizualizację topologii monitorowanych sieci oraz stanu monitorowanych obiektów. Pozwala także sterować pracą demonów monitorujących oraz kolekcjonujących dane. Obecnie większość funkcji tej aplikacji jest dostępna także poprzez aplikacje webowe.
Notification Processor	Zajmuje się głównie obsługą komunikatów typu SNMP Trap nadchodzących do stacji zarządzających od monitorowanych urządzeń. Otrzymane komunikaty mogą podlegać dowolnemu formatowaniu do

Informacje ogólne o systemie David

Nazwa produktu	Opis
	postaci czytelnej dla człowieka, a następnie mogą być rejestrowane. Przetworzone w ten sposób komunikaty mogą być również przekazywane do dalszego przetwarzania.
Operation Manager (full version)	Na podstawie przekazywanych mu danych może on uruchamiać wyspecyfikowane akcje. Skomplikowany często sposób oceny sytuacji, dokonywany m.in. na podstawie otrzymywanych sygnałów od innych produktów systemu David wraz z ich korelacją, ma na celu, w sposób nieco bardziej inteligentny niż tylko prosta reakcja na nie, generowanie informacji dla operatora w odpowiedzi na zaistniałe zdarzenia. Graficzna aplikacja wyświetla powiadomienia o zdarzeniach oraz umożliwia odtwarzanie plików dźwiękowych oraz odczytywanie komunikatów przez zewnętrzny syntezytor mowy.
Operation Manager (console version)	Produkt zawiera graficzną aplikację wyświetlającą powiadomienia o zdarzeniach oraz umożliwiającą odtwarzanie plików dźwiękowych oraz odczytywanie komunikatów przez zewnętrzny syntezytor mowy.
Report Manager	Zajmuje się przetwarzaniem zarejestrowanych komunikatów SNMP Trap, wpisów o toczących się sprawach oraz wpisów dotyczących zmiany stanów monitorowanych obiektów (obiektów pingowanych, interfejsów sieciowych i sąsiadów BGP) i generuje na tej podstawie dane do raportów. Wizualizacja wygenerowanych raportów dokonywana jest przez aplikację webową.
Service Monitor	Monitoruje wskazane serwisy sieciowe na poziomie warstwy aplikacyjnej. W tym celu monitoruje wskazane porty TCP wyspecyfikowanych hostów. Sprawdza zarówno dostępność portów jak i poprawność reakcji dla kilku wybranych protokołów sieciowych (HTTP, SMTP, FTP). Może także weryfikować poprawność pracy serwisów poprzez weryfikację danych od nich uzyskanych. Wyniki jego pracy w postaci raportów i wykresów prezentuje aplikacja webowa.
SQL Database Archive and Backup Manager	Zajmuje się archiwizowaniem danych przechowywanych w SQL-owej Bazie Danych, z których korzystają aplikacje systemu David .
Distributed SQL Database Manager	Pozwala podzielić bazę danych systemu David na jedną bazę podstawową oraz wiele baz dodatkowych. Pozwala to zwiększyć szybkość pracy systemu poprzez rozproszenie obciążenia na wiele serwerów. Proces migracji odbywa się podczas normalnej pracy systemu a podział bazy danych może być modyfikowany wielokrotnie.

Zależności pomiędzy poszczególnymi produktami **systemu David** przedstawia poniższy schemat.



Funkcjonalność **systemu David** może być bardzo szeroka i w ogromnym stopniu zależy od konkretnej konfiguracji. Najważniejsze funkcje jakie system może dostarczać to:

- odkrywanie i wizualizacja topologii monitorowanych sieci wraz z wizualizacją stanów poszczególnych węzłów sieci;
- formatowanie i rejestrowanie komunikatów typu SNMP-Trap nadsyłanych przez monitorujące urządzenia;
- automatyczne reagowanie na wybrane komunikaty typu SNMP-Trap nadsyłane przez monitorowane urządzenia;
- możliwość identyfikacji operatora odbierającego od systemu zgłoszenie o problemie (awarii);
- kolekcjonowanie danych dotyczących pracy monitorowanych urządzeń;
- automatyczne reagowanie na wykryte podczas kolekcjonowania danych nieprawidłowe wartości danych;
- prowadzenie ewidencji aktualnych spraw prowadzonych przez system powstałych jako reakcja na zdarzenia mające miejsce w zarządzanej sieci i wykrytych przez system;
- monitorowanie serwisów sieciowych warstwy aplikacji.

Rozdział 3. Terminologia

3.1. Autoryzacja dokonywana przez moduły systemu David

Moduły pracujące w ramach systemu David, które potrzebują dokonywać autoryzacji nadawców wiadomości (np. **damsnmpdaud**, **dnmmsd**, **dgnsd**), korzystają z biblioteki, która sprawdza, czy adres IP nadawcy pasuje do jakiegokolwiek wpisu w pliku `.known.host`. Biblioteka spodziewa się, że plik ten znajduje się w podkatalogu `.sec` katalogu, którego nazwa wskazywana jest przez zmienną `confdir` w pliku `/etc/system-david.conf`.

Wpisy w pliku `.known.host` mają postać wyrażeń regularnych specyfikujących adresy IP, które mają być akceptowane.

3.2. Terminy używane w systemie David i jego dokumentacji

Poniżej znajduje się wyjaśnienie części terminów, które są używane przez system David oraz w dokumentach opisujących jego pracę:

- **wiadomości (informacje)** - najczęściej są to dane otrzymywane przez interfejsy **Operation Manager-a**, jego analizatory danych oraz **Jednostkę Tworzącą Bazę Danych Aktywnych Spraw** należącą do tego produktu;
- **komunikaty** - termin ten występuje głównie w produktach: **Notification Processor**, **Operation Manager** i **Report Manager**; najczęściej są to dane, których źródłem są agenci SNMP pracujący na monitorowanych urządzeniach sieciowych;
- **zdarzenia (events)** - termin ten występuje najczęściej w produktach takich jak: **Operation Manager** i **Report Manager**; określa byt, którego źródłem jest pojedyncza dana typu SNMP Trap lub SNMP Data; **zdarzenie** zawsze wchodzi w skład **sprawy**;
- **sprawy (cases)** - termin ten występuje najczęściej w produktach takich jak: **Operation Manager** i **Report Manager**; określa zbiór zdarzeń skojarzonych ze sobą; w skład sprawy musi wchodzić przynajmniej jedno **zdarzenie**;
- **SNMP Trap** - rodzaj danych występujących w produkcie **Operation Manager**, których źródłem są komunikaty otrzymywane od agentów SNMP; komunikaty te nie są odpowiedzią na prośby wysyłane przez stację zarządzającą, lecz są samoistnie wysyłane przez agentów zarządzających urządzeniami sieciowymi i przetwarzane przez produkt **Notification Processor**;

- **SNMP Data** - rodzaj danych występujących w produkcie **Operation Manager**, których źródłem są odpowiedzi otrzymywane od agentów SNMP na prośby, wysyłane do nich przez stację zarządzającą za pomocą produktu **Network Manager**.

Rozdział 4. Instalacja

4.1. Główny plik konfiguracyjny systemu David

Podstawowy plik konfiguracyjny systemu David to `/etc/david-system.conf`. Zawiera on wpisy konfiguracyjne jako pary: klucz = wartość. Poza wpisem `default_email_recipient` w zasadzie żadna inna pozycja nie musi być modyfikowana ręcznie. Wszystkie konieczne modyfikacje dokonywane są podczas instalowania poszczególnych produktów systemu David. Poniżej znajduje się lista możliwych wpisów wraz z ich opisem.

- `user` - nazwa użytkownika z prawami którego pracują demony systemu David;
- `default_email_recipient` - domyślny adres e-mail, gdzie są wysyłane wiadomości od aplikacji systemu David;
- `bindir` - katalog z aplikacjami systemu David (domyślnie: `/usr/bin/david-system`);
- `libdir` - katalog z bibliotekami systemu David (domyślnie: `/usr/lib/david-system`);
- `incdir` - katalog z plikami nagłówkowymi systemu David (domyślnie: `/usr/include/david`);
- `confdir` - katalog z plikami konfiguracyjnymi systemu David (domyślnie: `/etc/david-system`);
- `logdir` - katalog, gdzie są tworzone logi aplikacji systemu David (domyślnie: `/var/log/david-system`);
- `sharedir` - katalog z różnymi plikami (obrazki, pliki audio, serwis webowy itd.) systemu David (domyślnie: `/usr/share/david-system`);
- `docdir` - katalog z dokumentacją systemu David (domyślnie: `/usr/share/doc/david-system`);
- `vardir` - katalog z archiwami bazy SQL-owej systemu David (domyślnie: `/var/lib/david-system`);
- `is_sqldb_installed` - flaga, czy SQL-owa baza systemu David została zainstalowana.

4.2. Dedykowane konto dla obsługi systemu David

Nie ma potrzeby, aby jakikolwiek moduł systemu David pracował z prawami superużytkownika (zazwyczaj konto o nazwie `root` i UID równym 0). Nawet, jeśli dany daemon systemu David wymaga w chwili uruchomienia praw superużytkownika, to zawsze istnieje możliwość wyspecyfikowania jako argumentu uruchomienia demona nazwy użytkownika, którego prawa ma przyjąć.

Najwygodniej jest dodać nowego użytkownika do systemu operacyjnego, pod którego kontrolą ma

pracować system David.

4.3. Układ katalogów systemu David

Układ katalogów i ich zawartość może być zależna od konkretnej konfiguracji systemu David. W standardowej konfiguracji systemu poszczególne katalogi zawierają:

- `/usr/bin/david-system` - pliki binarne i skrypty;
- `/etc/david-system` - pliki konfiguracyjne;
- `/usr/share/doc/david-system` - dokumentację systemu David;
- `/usr/share/david-system` - pliki graficzne, dźwiękowe, portal webowy;
- `/usr/include/david` - pliki nagłówkowe systemu David;
- `/usr/lib/david-system` - biblioteki systemu David;
- `/var/log/david-system` - pliki z logami;
- `/var/lib/david-system` - archiwa bazy SQL-owej systemu David.

4.4. Konfiguracja demon-a syslogd

Moduły systemu David wykorzystują standardowy podsystem `syslog` dostępny na platformach UNIX-owych. Domyślne ustawienia modułów systemu David powodują, że informacje są przesyłane jako typ (`facility`) `local6`. Ustawienia te można oczywiście zmienić w momencie uruchamiania danego modułu. W związku z tym dobrym pomysłem wydaje się takie skonfigurowanie demona `syslogd`, aby wszystkie informacje przesyłane od modułów systemu David znalazły się w jednym miejscu (w jednym lub kilku plikach o charakterystycznej nazwie np.: `david.log`).

Rozdział 5. Wymagania dla Operation Manager-a

Platforma zarządzania, na której ma pracować **Operation Manager** musi spełniać następujące wymagania:

- posiadać zainstalowaną kompatybilną wersję **Notification Processor-a**.

Rozdział 6. Instalacja

6.1. Instalacja z pakietu RPM

Instalacja produktu wymaga uruchomienia przez użytkownika posiadającego prawa `root-a`. Poniżej znajdują się kolejne kroki typowej instalacji:

- Zainstaluj produkt:

```
rpm -i david-xxx-om-f-yyy.rpm
```

6.2. Instalacja za pomocą skryptu

Instalacja produktu wymaga uruchomienia przez użytkownika posiadającego prawa `root-a`. Poniżej znajdują się kolejne kroki typowej instalacji:

- Rozkompresuj i rozpakuj archiwum:

```
gunzip david-xxx-om-f-yyy.i386.tar.gz  
tar xf david-xxx-om-f-yyy.i386.tar
```

Operacje te spowodują utworzenie katalogu `david-xxx-om-f-yyy.i386` w bieżącym katalogu.

- Zmień swój bieżący katalog na `david-xxx-om-f-yyy.i386`:

```
cd david-xxx-om-f-yyy.i386
```

- Przeczytaj plik `LICENSE` z bieżącego katalogu i **KONTYNUUJ INSTALACJĘ TYLKO WTEDY, KIEDY AKCEPTUJESZ WARUNKI TAM ZAWARTE.**
- Uruchom skrypt instalacyjny:

```
./install
```

Rozdział 7. Informacje ogólne

7.1. Funkcjonalność

Operation Manager umożliwia:

- korelacje komunikatów przekazywanych od **Notification Processor** i **Network Manager**;
- grupowanie otrzymywanych komunikatów w aktualne sprawy jednoznacznie identyfikowane w systemie;
- uruchamianie akcji (wyspecyfikowanych programów) dla prowadzonych spraw z uwzględnieniem korelacji zdarzeń (komunikatów) wchodzących w ich skład, wyników działania samych akcji i czasu ich wystąpienia;
- przetwarzanie zdarzeń otrzymywanych od dowolnych procesów nie będących modułami systemu David, co stwarza możliwość reakcji na zdarzenia, których źródłem nie musi być agent SNMP;
- uzyskanie informacji o aktualnych awariach, prowadzonych sprawach i ich obsłudze;
- uczenie się rozpoznawania informacji potencjalnie niebezpiecznych dla monitorowanych urządzeń sieci komputerowej poprzez uczenie się akceptowalnych wartości ich parametrów pracy;
- powiadamianie operatora o otrzymanej wiadomości, toczącej się sprawie oraz rejestrowanie identyfikatora użytkownika i czasu jego reakcji.

7.2. Opis działania

Operation Manager spełnia istotną rolę w całym systemie. Na jego prace wpływają inne produkty, on zaś podejmując decyzje wpływa na ich prace. Dzięki zaimplementowanym mechanizmom stara się on ocenić ważność otrzymywanych informacji, a także odnaleźć powiązania pomiędzy otrzymywanymi wiadomościami. Informacje, które do niego docierają mogą być różnych typów (np.: zwyczajne dane uzyskane poprzez zapytania protokołu SNMP oraz wiadomości typu SNMP-Trap).

Jednym z głównych celów działania produktu jest uruchamianie odpowiednich programów lub skryptów w odpowiedzi na bieżący stan monitorowanych urządzeń oraz powiadamianie operatora, poprzez e-mail, okienko graficzne oraz sygnał dźwiękowy, o zaistniałych zdarzeniach i prowadzonych sprawach (np.: zanik zasilania, brak połączenia pomiędzy urządzeniami, restart urządzenia itp.).

Aktualne wyniki pracy **Operation Manager-a** są dostępne przez **Aplikacje WWW**, poprzez które można także sterować jego pracą.

7.3. Tematy pokrewne

[Dostęp do Interfejsu Danych SNMP Trap \(damsnmpti\)](#)

[Analizator Danych SNMP Trap \(damsnmptaud\)](#)

[Dostęp do Interfejsu Danych SNMP Data \(damsnmpdi\)](#)

[Analizator Danych SNMP Data \(damsnmpdaud\)](#)

[Jednostka Tworząca Bazę Danych Aktywnych Spraw \(damadbud\)](#)

[Serwis Aktywnych Spraw \(damcsud\)](#)

[Przeglądarka Toczących się Spraw \(Pending Operation Browser\)](#)

[Przeglądarka Zarejestrowanych Spraw \(Recorded Operation Browser\)](#)

[Serwer Powiadomień Graficznych \(dgnsd\)](#)

[Prezenter Powiadomień Graficznych \(xdgnp\)](#)

Rozdział 8. Serwer Powiadomień Graficznych (dgnsd)

8.1. Opis ogólny

Program **dgnsd** jest **Serwerem Powiadomień Graficznych** i jest częścią **Operation Manager-a**. Jest to proces daemon, który pracuje bez przerwy w czasie działania systemu. Oczekuje on na zgłoszenia klientów, którzy chcą wyświetlić komunikat o toczącej się sprawie w postaci graficznego komunikatu lub chcą odtworzyć plik dźwiękowy lub odczytać tekst. Wiadomości otrzymane od tych klientów są następnie rozsyłane do zarejestrowanych wcześniej graficznych klientów tzn. takich, którzy potrafią wyświetlić graficzne powiadomienie lub odtworzyć dźwięk. Odpowiedzi od nich są zaś przekazywane tym klientom, którzy wcześniej zgłosili chęć powiadomienia o danej sprawie. Typowym klientem powiadamiającym o sprawie jest program **dsi**, zaś w przypadku próśb o odtworzenie dźwięku typowym klientem zgłaszającym prośbę jest program **sndc**. Klientem graficznym, wyświetlającym okienka z informacjami oraz odtwarzającym dźwięk jest aplikacja **xdgnp**.

8.2. Składnia

Program **dgnsd** można uruchomić z następującymi opcjami: `[-p,--pid-file filename]` `[-l,--log-facility log_facility]` `[-L,--log-level log_level]` `[--nr-of-gc nr]` `[--nr-of-ic nr]` `[--socket-file-4-gc filename]` `[--socket-file-4-ic filename]` `[--port-4-gc port]` `[--port-4-ic port]` `[--ttl seconds]` `[-u,--run-as-user username]` `[--background]` `[-v,--version]` `[-h,--help]`

8.3. Znaczenie opcji w programie dgnsd

Tabela 8.1. Znaczenie opcji w programie dgnsd

Nazwa opcji	Opis
<code>-p,--pid-file filename</code>	Zapisz swój PID od wyspecyfikowanego pliku
<code>-l,--log-facility log_facility</code>	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
<code>-L,--log-level log_level</code>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: notice).
<code>--nr-of-gc nr</code>	Podaj maksymalną liczbę graficznych klientów, która może być obsługiwana w tym samym czasie (domyślnie: 10).
<code>--nr-of-ic nr</code>	Podaj maksymalną liczbę klientów powiadomień, która może być

Nazwa opcji	Opis
	obsługiwana w tym samym czasie (domyślnie: 20).
<i>--socket-file-4-gc filename</i>	Oczekuj połączeń od graficznych klientów poprzez wyspecyfikowany plik typu socket (domyślnie: /tmp/dgns.gc.socket).
<i>--socket-file-4-ic filename</i>	Oczekuj połączeń od klientów powiadomień poprzez wyspecyfikowany plik typu socket (domyślnie: /tmp/dgns.ic.socket).
<i>--port-4-gc port</i>	Nasłuchuj na wyspecyfikowanym porcie TCP, czekając na klientów graficznych (domyślnie: użyj pliku typu socket).
<i>--port-4-ic port</i>	Nasłuchuj na wyspecyfikowanym porcie TCP, czekając na klientów powiadomień (domyślnie: użyj pliku typu socket).
<i>--ttl seconds</i>	Podaj maksymalne TTL dla wiadomości wychodzących (domyślnie: 300).
<i>-u,--run-as-user username</i>	Zrezygnuj z praw root'a i uruchom serwer z prawami podanego użytkownika.
<i>--background</i>	Rozpocznij pracę w tle po uruchomieniu.
<i>-v,--version</i>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<i>-h,--help</i>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

8.4. Opis działania

Program **dgnsd** jest serwerem jednoprotocowym. Klientami jego mogą być zarówno programy zgłaszające chęć powiadomienia operatora o zaistniałym problemie jak i programy potrafiące wyświetlić graficzne powiadomienie w postaci okienka dialogowego. W chwili otrzymania komunikatu od klienta powiadamiającego o sprawie, rozsyłany jest on do zarejestrowanych wcześniej programów, które są aplikacjami graficznymi i potrafią wyświetlić graficzne powiadomienie. Wiadomość zostanie rozesłana tylko tym aplikacjom, które mają zalogowanych użytkowników. Innymi słowy, jeśli aplikacja graficzna chce otrzymywać powiadomienia, to musi się oczywiście połączyć z serwerem oraz dokonać udanej autentykacji użytkownika w systemie David.

Autentykacji dokonuje się na podstawie informacji zawartych w **Bazie Danych Użytkowników** przechowywanej w SQL-owej bazie danych. Klientowi powiadomień, który wysłał dane powiadomienie, zostanie przekazana odpowiedź od klienta graficznego, jeśli taką serwer otrzyma. Z racji, że więcej niż jeden klient graficzny może wysłać odpowiedź, to została wprowadzona prosta reguła. Zgodnie z nią zostanie uwzględniona tylko odpowiedź od tego klienta, który posiada token. W danej chwili może istnieć tylko jeden token dla całego serwera **dgnsd**. W przypadku, gdy nie istnieje żaden token, wtedy liczy się pierwsza odpowiedź. Dany klient graficzny może się logować wielokrotnie i do tego za każdym razem jako inny użytkownik. Zalogowany użytkownik posiada uprawnienia takie, jakie są zapisane w **Bazie Danych Użytkowników**. Użytkownicy mogą dokonywać następujących operacji:

- wylogowywać się;

- wylogowywać użytkowników niższego poziomu;
- pobierać listę zalogowanych użytkowników;
- pobierać token, jeśli nie był nikomu przyznany lub użytkownik, który go posiadał był niższego poziomu;
- oddawać token, jeśli się jest jego właścicielem;
- przekazywać token użytkownikowi z niższego poziomu uprawnień pod warunkiem, że token był wcześniej własnością danego użytkownika lub nie należał do nikogo lub należał do użytkownika niższego poziomu.

Serwer **dgnsd** zajmuje się również odbieraniem od programu [sndc](#) próśb o odtworzenie dźwięku i przesyła je do klientów [xdgnp](#). Mogą być dwa rodzaje takich próśb:

- prośba o odczytanie tekstu;
- prośba o odtworzenie pliku dźwiękowego.

Wiadomości tekstowe są odtwarzane za pomocą syntezy mowy. W przypadku kiedy ma być odtworzony plik dźwiękowy należy podać ścieżkę do niego. Każda ścieżka automatycznie poprzedzana jest katalogiem, w którym lokalnie zainstalowany jest system David (np. /home/david).

Klient [xdgnp](#) może być zainstalowany na innej stacji niż serwer **dgnsd**. Kiedy klient [xdgnp](#) otrzymuje prośbę o odtworzenie pliku dźwiękowego, wówczas sprawdza, czy ma taki plik w swoim lokalnym systemie plików, czy też ma go pobrać ze stacji, na której pracuje serwer **dgnsd**. Jeśli [xdgnp](#) nie znajduje go na swojej stacji to wówczas, jeśli serwer **dgnsd** pracuje na innym komputerze, prosi serwer o przesłanie mu tego pliku. Wszystkie pliki dźwiękowe znajdują się na stacji, na której jest zainstalowany **dgnsd**.

8.5. Tematy pokrewne

[Prezenter Powiadomień Graficznych \(xdgnp\)](#)

[Klient Dźwięku \(sndc\)](#)

Notification Processor: Rejestrator Wiadomości (dsi)

Network Manager: Zarządca Kont (User Manager)

Rozdział 9. Klient Dźwięku (sndc)

9.1. Opis ogólny

Program **sndc** jest **Klientem Dźwięku** i stanowi część **Operation Manager-a**. Jego głównym zadaniem jest wysyłanie do serwera [dgnsd](#) próśb o odtworzenie dźwięku.

9.2. Składnia

Program **sndc** można uruchomić z następującymi opcjami: [[-l,--log-facility log_facility](#)] [[-L,--log-level log_level](#)] [[--socket-file filename](#)] [[--host host](#)] [[--port port](#)] [[--ttl seconds](#)] [[--text text](#)] [[--file filename](#)] [[-v,--version](#)] [[-h,--help](#)]

9.3. Znaczenie opcji w programie sndc

Tabela 9.1. Znaczenie opcji w programie sndc

Nazwa opcji	Opis
-l,--log-facility log_facility	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
-L,--log-level log_level	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: notice).
--socket-file filename	Połącz się z serwerem poprzez wyspecyfikowany plik typu socket (domyślnie: /tmp/dgns.ic.socket).
--host host	Połącz się z serwerem pracującym na wyspecyfikowanym komputerze (domyślnie: połącz się poprzez plik typu socket).
--port port	Połącz się z serwerem poprzez wyspecyfikowany port TCP (domyślnie: użyj pliku typu socket).
--ttl seconds	Podaj maksymalny czas w sekundach dla oczekiwania na zakończenie połączenia (domyślnie: 60).
--text text	Poproś serwer o odczytanie tego tekstu przez jego graficznych klientów.
--file filename	Poproś serwer o odtworzenie tego pliku przez jego graficznych klientów.
-v,--version	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
-h,--help	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

9.4. Opis działania

Program **sndc** zajmuje się wysyłaniem do serwera [dgnsd](#) próśb o odtworzenie dźwięku. Serwer [dgnsd](#) rozsyła następnie tę prośbę do swoich graficznych klientów, którymi są aplikacje [xdgmp](#). Komunikaty, które wysyła **sndc** mogą być dwojakiego rodzaju:

- prośba o odczytanie tekstu;
- prośba o odtworzenie pliku dźwiękowego.

Tekst jest odtwarzany za pomocą syntezy mowy. W przypadku kiedy ma być odtworzony plik dźwiękowy należy podać ścieżkę do niego. Każda ścieżka automatycznie poprzedzana jest katalogiem, w którym znajdują się pliki dźwiękowe system David (domyślnie `/usr/share/david-system/sounds`). Podanie więc jako nazwy pliku `alarm.wav` jest rozumiane przez graficznego klienta jako plik `/usr/share/david-system/sounds/alarm.wav`.

9.5. Tematy pokrewne

[Serwer Powiadomień Graficznych \(dgnsd\)](#)

[Prezenter Powiadomień Graficznych \(xdgmp\)](#)

Rozdział 10. Dostęp do Interfejsu Danych SNMP Trap (damsnmpti)

10.1. Opis ogólny

Program **damsnmpti** jest **Dostępem do Interfejsu Danych SNMP Trap** i jest częścią **Operation Manager-a**.

10.2. Składnia

Program **damsnmpti** można uruchomić z następującymi opcjami: [\[-f,--server-socket-file socket_file\]](#) [\[-H,--server-host host\]](#) [\[-P,--port-on-host port\]](#) [\[-l,--log-facility log_facility\]](#) [\[-L,--log-level log_level\]](#) [\[-i,--trap-severity level\]](#) [\[-s,--trap-source IP_address\]](#) [\[-a,--trap-agent IP_address\]](#) [\[-c,--trap-community community\]](#) [\[-n,--trap-number information_nr_or_oid\]](#) [\[-e,--trap-enterprise enterprise\]](#) [\[-t,--trap-time-stamp time-stamp\]](#) [\[-g,--trap-generic generic_type\]](#) [\[-p,--trap-specific specific_type\]](#) [\[-d,--trap-id string\]](#) [\[-C,--community community\]](#) [\[-m,--trap-msg message\]](#) [\[--device-id identifier\]](#) [\[--managed-item-type type\]](#) [\[--managed-item-id identifier\]](#) [\[-v,--version\]](#) [\[-h,--help\]](#)

10.3. Znaczenie opcji w programie damsnmpti

Tabela 10.1. Znaczenie opcji w programie damsnmpti

Nazwa opcji	Opis
<i>-f,--server-socket-file socket_file</i>	Połącz się z serwerem poprzez wyspecyfikowany plik typu socket (domyślnie: /tmp/damsnmptid.socket).
<i>-H,--server-host host</i>	Połącz się z serwerem pracującym na wyspecyfikowanym komputerze (domyślnie: połącz się poprzez plik typu socket: /tmp/damsnmptid.socket).
<i>-P,--port-on-host port</i>	Połącz się z serwerem pracującym na wyspecyfikowanym komputerze używając wyspecyfikowanego portu (domyślnie: port 6688 na UDP).
<i>-l,--log-facility log_facility</i>	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
<i>-L,--log-level log_level</i>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: warning).
<i>-i,--trap-severity level</i>	Poziom ważności tej informacji wyrażony jest jako liczba zmiennoprzecinkowa podwójnej precyzji - double - z zakresu <-1.0;1.0> (domyślnie:0.0).

Nazwa opcji	Opis
<i>-s,--trap-source IP_adres</i>	Adres IP nadawcy wiadomości, która ma zostać przekazana Interfejsowi Danych SNMP Trap .
<i>-a,--trap-agent IP_address</i>	Adres IP agenta, który wysłał tę wiadomość.
<i>-c,--trap-community community</i>	Nazwa wspólnoty tej wiadomości.
<i>-n,--trap-number information_nr_or_oid</i>	Identyfikator (OID) tej wiadomości.
<i>-e,--trap-enterprise enterprise</i>	Identyfikator (OID) producenta urządzenia (w bazie MIB), z którego pochodzi ten komunikat.
<i>-t,--trap-time-stamp time-stamp</i>	Liczba sekund jaka upłynęła od inicjalizacji agenta, który wysłał tę wiadomość.
<i>-g,--trap-generic generic_type</i>	Liczba określająca ogólny typ tego komunikatu
<i>-p,--trap-specific specific_type</i>	Liczba określająca specyficzny typ tego komunikatu
<i>-d,--trap-id string</i>	Unikalny ID string identyfikujący tę wiadomość w Operation Manager .
<i>-C,--community community</i>	Community przyznane temu komunikatowi (jeśli jest więcej niż jedno, to muszą być oddzielone znakiem ':').
<i>-m,--trap-msg message</i>	Czytelna dla człowieka wiadomość opisująca ten komunikat.
<i>--device-id identifier</i>	ID urządzenia z Bazy Informacji o Sieci, które jest źródłem tej informacji.
<i>--managed-item-type type</i>	Typ zarządzanego elementu na urządzeniu źródłowym, którego ta informacja dotyczy.
<i>--managed-item-id identifier</i>	ID zarządzanego elementu na urządzeniu źródłowym, którego ta informacja dotyczy.
<i>-v,--version</i>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<i>-h,--help</i>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

10.4. Opis działania

Program **damsnmpti** umożliwia dostęp do [Interfejsu Danych SNMP Trap](#) z poziomu shell'a. Program służy jako pomost do [Interfejsu Danych SNMP Trap](#). Wszystkie informacje o pojedynczym komunikacie otrzymuje w postaci swoich parametrów wywołania i przekazuje je serwerowi, który pełni rolę [Interfejsu Danych SNMP Trap \(damsnmptaud\)](#).

10.5. Tematy pokrewne

[Analizator Danych SNMP Trap \(damsnmptaud\)](#)

Rozdział 11. Analizator Danych SNMP Trap (damsnmptaud)

11.1. Opis ogólny

Program **damsnmptaud** jest **Analizatorem Danych SNMP Trap** i jest częścią **Operation Manager-a**. Pełni także rolę **Interfejsu Danych SNMP Trap**. Jest to proces daemon, który pracuje bez przerwy w czasie działania systemu i oczekuje na dane typu SNMP Trap (nie są to komunikaty TRAP-PDU w sensie protokołu SNMP). Na podstawie danych zawartych w swoim pliku konfiguracyjnym oblicza ważność komunikatu, a następnie przekazuje go dalej. Typowym odbiorcą przetworzonych w ten sposób komunikatów jest **Baza Danych Skojarzeń**.

Program **damsnmptaud** czyta swój plik konfiguracyjny `.damsnmptaudrc` podczas startu. Spodziewa się go znaleźć w katalogu `/etc/david-system`. Demon **damsnmptaud** nie wystartuje jeśli nie będzie mógł przeczytać swojego pliku konfiguracyjnego.

11.2. Składnia

Program **damsnmptaud** można uruchomić z następującymi opcjami: `[-P,--pid-file filename]` `[-l,--log-facility log_facility]` `[-L,--log-level log_level]` `[-f,--in-socketfile filename]` `[-p,--in-port unsignedtype_number]` `[--out-socketfile filename]` `[--out-port unsignedtype_number]` `[--out-host hostname]` `[-u,--run-as-user username]` `[--severity-balance-factor doubletype_number]` `[--background]` `[--ignore-case-4-severity]` `[-v,--version]` `[-h,--help]`

11.3. Znaczenie opcji w programie damsnmptaud

Tabela 11.1. Znaczenie opcji w programie damsnmptaud

Nazwa opcji	Opis
<code>-P,--pid-file filename</code>	Zapisz swój identyfikator procesu (PID) do wyspecyfikowanego pliku
<code>-l,--log-facility log_facility</code>	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
<code>-L,--log-level log_level</code>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: warning).
<code>-f,--in-socketfile filename</code>	Oczekuj połączeń od klientów poprzez wyspecyfikowany pliku typu socket (domyślnie: <code>/tmp/damsnmptaud.socket</code>).
<code>-p,--in-port</code>	Użyj wyspecyfikowanego portu udp do nasłuchiwania zamiast pliku typu

Nazwa opcji	Opis
<i>unsignedtype_numbe</i>	socket (domyślnie użyj pliku typu socket: /tmp/damsnmptaud.socket).
<i>--out-socketfile filename</i>	Użyj wyspecyfikowanego pliku typu socket dla wyjścia (domyślnie: /tmp/damadbud.socket).
<i>--out-port unsignedtype_number</i>	Wysyłaj wychodzące pakiety UDP na ten port (domyślnie: użyj pliku typu socket '/tmp/damadbud.socket').
<i>--out-host hostname</i>	Wysyłaj wychodzące pakiety UDP do tego komputera (domyślnie: 127.0.0.1).
<i>-u,--run-as-user username</i>	Zrezygnuj z praw root'a i uruchom serwer z prawami podanego użytkownika.
<i>--severity-balance-factor doubletype_number</i>	Użyj podanego czynnika ważności (zakres <0.0;1.0>), aby obliczyć ostateczną ważność przetwarzanej wiadomości: nadchodząca_ważność * ten_parametr +- ważność_z_konfiguracji * (1.0-ten_parametr) (domyślnie: 0.5).
<i>--background</i>	Rozpocznij pracę w tle po uruchomieniu
<i>--ignore-case-4-severity</i>	Wylicz ważność przetwarzanej wiadomości ignorując wielkość liter.
<i>-v,--version</i>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<i>-h,--help</i>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

11.4. Format pliku konfiguracyjnego

Plik konfiguracyjny składa się z definicji obiektów typu `group`. Definicja obiektu jest ograniczona klamrami `{ }`. Słowo kluczowe `group` określające typ obiektu musi być jedynym słowem w linii. Ta sama reguła dotyczy klamry rozpoczynającej i kończącej definicję obiektu. Ogólnie więc pojedynczy obiekt wygląda tak:

```
group
{
...
}
```

Obiekt jest zbudowany z dowolnej ilości par elementów. Na każdy element składają się dwie linie pliku. Pierwsza z nich, zaczynająca się od słowa kluczowego `string`, specyfikuje ciąg znaków, zaś druga, rozpoczynająca się od słowa `value`, jego ważność. Przykładowe elementy wchodzące w skład obiektu typu `group` mogą wyglądać następująco:

```
string=222.222.222.222
```

```
value=0.53
```

```
string=A realy bad news
```

```
value=0.99
```

Liczby określające wartości są traktowane jako zmiennoprzecinkowe z zakresu <0.0;1.0>.

11.5. Opis działania

W przypadku, gdy agent oczekujący zgłoszeń klientów poprzez sieć komputerową otrzymuje nowy komunikat UDP, to dokonuje autoryzacji klienta na podstawie jego adresu IP. W tym celu przetwarzane są informacje zawarte w pliku `.known.hosts`. Jeśli nie można otworzyć pliku `.known.hosts` do czytania lub adres IP klienta nie pasuje do żadnego z wpisów w tym pliku, to pakiet UDP jest odrzucany. Dla klientów lokalnych nie jest dokonywana żadna autoryzacja.

W czasie przetwarzania kolejnej zaakceptowanej wiadomości obliczana jest jej ważność, a następnie jest ona przesyłana dalej. Typowym odbiorcą przetworzonych przez demona **damsnmptaud** wiadomości jest **Baza Danych Skojarzeń**.

W procesie oceny ważności komunikatu brany jest pod uwagę tylko identyfikator wiadomości (identyfikator ten dostaje np. program [damsnmpti](#) w postaci argumentu opcji `-d`). Jednoznacznie identyfikuje on wiadomość w całym **Operation Manager**. Początkowa ważność komunikatu wynosi zero. Następnie dla każdego obiektu typu `group`, zdefiniowanego w pliku konfiguracyjnym jest sprawdzane, czy jakkolwiek łańcuch znaków występujący po słowie kluczowym `string=` w tym obiekcie, jest zawarty w identyfikatorze badanego komunikatu. Jeśli tak to brana jest pod uwagę maksymalna wartość przyporządkowana elementowi `string`. Wartość ta następnie jest dodawana do dotychczas obliczonej ważności komunikatu. Na końcu obliczona ważność komunikatu jest dzielona przez ilość obiektów typu `group` zdefiniowanych w pliku konfiguracyjnym. Ostateczna ważność jest liczona według wzoru:

$$s * f - S * (1 - f), \text{ gdy } s < 0$$

lub

$$s * f + S * (1 - f), \text{ gdy } s \geq 0$$

gdzie s jest ważnością komunikatu nadaną w trakcie jego wcześniejszego przetwarzania przez inne moduły systemu David. Parametr S jest dotychczasową ważnością tego komunikatu obliczoną przez program **damsnmptaud** na podstawie swojego pliku konfiguracyjnego. Czynniki f jest argumentem opcji [--severity-balance-factor](#) programu **damsnmptaud**.

11.6. Tematy pokrewne

[Dostęp do Interfejsu Danych SNMP Trap \(damsnmpti\)](#)

[Jednostka Tworząca Bazę Danych Aktywnych Spraw \(damadbud\)](#)

Rozdział 12. Dostęp do Interfejsu Danych SNMP Data (damsnmpdi)

12.1. Opis ogólny

Program **damsnmpdi** jest **Dostępem do Interfejsu Danych SNMP Data** i jest częścią **Operation Manager-a**.

12.2. Składnia

Program **damsnmpdi** można uruchomić z następującymi opcjami: [*-f,--server-socket-file socket_file*] [*-H,--server-host host*] [*-P,--port-on-host port*] [*-l,--log-facility log_facility*] [*-L,--log-level log_level*] [*-a,--data-address IP_address*] [*-o,--data-oid oid*] [*-t,--data-type type*] [*-V,--data-value doubletype_number*] [*-I,--data-id string*] [*-C,--community community*] [*-m,--data-msg message*] [*-p,--data-sender-pid pid*] [*--device-id identifier*] [*--managed-item-type type*] [*--managed-item-id identifier*] [*-v,--version*] [*-h,--help*]

12.3. Znaczenie opcji w programie damsnpdi

Tabela 12.1. Znaczenie opcji w programie damsnpdi

Nazwa opcji	Opis
<i>-f,--server-socket-file socket_file</i>	Połącz się z serwerem poprzez wyspecyfikowany plik typu socket (domyślnie: /tmp/damsnmpdid.socket).
<i>-H,--server-host host</i>	Połącz się z serwerem pracującym na wyspecyfikowanym komputerze (domyślnie: połącz się poprzez plik typu socket: /tmp/damsnmpdid.socket).
<i>-P,--port-on-host port</i>	Połącz się z serwerem pracującym na wyspecyfikowanym komputerze używając wyspecyfikowanego portu (domyślnie: port 6699 na UDP).
<i>-l,--log-facility log_facility</i>	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
<i>-L,--log-level log_level</i>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: warning).
<i>-a,--data-address IP_address</i>	Adres IP urządzenia, z którego pochodzi dana mająca być przekazana Interfejsowi Danych SNMP Data .
<i>-o,--data-oid oid</i>	Identyfikator (OID) MIB danej mającej być przekazanej Interfejsowi Danych SNMP Data .

Nazwa opcji	Opis
<i>-t,--data-type type</i>	Typ danej, która ma być przekazana Interfejsowi Danych SNMP Data . Możliwe są następujące typy: counter - wartość może być tylko rosnąca (32 bity); gauge - wartość może być rosnąca i malejąca (32 bity); error - wartość z błędem; timeout - brak odpowiedzi dla tej zmiennej; other - wartość innego typu (IPADDRESS, OCTET, itd.).
<i>-V,--data-value doubletype_number</i>	Wartość tej zmiennej jako wartość zmiennoprzecinkowa podwójnej precyzji.
<i>-I,--data-id string</i>	Unikalny ID string identyfikujący tą zmienną w Operation Manager .
<i>-C,--community community</i>	Community przyznane temu komunikatowi (jeśli jest więcej niż jedno, to muszą być oddzielone znakiem ':').
<i>-m,--data-msg message</i>	Czytelna dla człowieka wiadomość opisująca znaczenie tej zmiennej.
<i>-p,--data-sender-pid pid</i>	PID (identyfikator) nadawcy wiadomości.
<i>--device-id identifier</i>	ID urządzenia z Bazy Informacji o Sieci, które jest źródłem tej informacji.
<i>--managed-item-type type</i>	Typ zarządzanego elementu na urządzeniu źródłowym, którego ta informacja dotyczy.
<i>--managed-item-id identifier</i>	ID zarządzanego elementu na urządzeniu źródłowym, którego ta informacja dotyczy.
<i>-v,--version</i>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<i>-h,--help</i>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

12.4. Opis działania

Program **damsnmpdi** umożliwia dostęp do [Interfejsu Danych SNMP Data](#) z poziomu shell'a. Program służy jako pomost do [Interfejsu Danych SNMP Data](#). Wszystkie informacje o pojedynczym komunikacie otrzymuje w postaci swoich parametrów wywołania i przekazuje je serwerowi, który pełni rolę [Interfejsu Danych SNMP Data \(damsnmpdaud\)](#).

12.5. Tematy pokrewne

[Analizator Danych SNMP Data \(damsnmpdaud\)](#)

Rozdział 13. Analizator Danych SNMP Data (damsnmpdaud)

13.1. Opis ogólny

Program **damsnmpdaud** jest **Analizatorem Danych SNMP Trap** i jest częścią **Operation Manager-a**. Pełni także rolę **Interfejsu Danych SNMP Data**. Jest to proces daemon, który pracuje bez przerwy w czasie działania systemu i oczekuje na dane typu SNMP Data. Na podstawie analizy danych, które są kolekcjonowane w systemie, a następnie jemu przekazywane, stara się wykryć wartości danych wykraczające poza wartości średnie i przekazać je dalej. Typowym odbiorcą przetworzonych w ten sposób komunikatów jest **Baza Danych Skojarzeń**.

Podczas startu **damsnmpdaud** czyta swój plik konfiguracyjny `.damsnmpdaudrc`. Program spodziewa się go znaleźć w katalogu `/etc/david-system`. Daemon **damsnmpdaud** nie uruchomi się jeśli nie będzie mógł odczytać swojej konfiguracji.

Po przeczytaniu pliku konfiguracyjnego **damsnmpdaud** czyta następnie plik ze zdefiniowanymi minimalnymi poziomami wartości, po przekroczeniu których wysyłane są dopiero powiadomienia do **Bazy Danych Skojarzeń**.

13.2. Składnia

Program **damsnmpdaud** można uruchomić z następującymi opcjami: `[-f,--socketfile filename]` `[-p,--port unsignedtype number]` `[--snmpdf-average-len doubletype number]` `[--snmpdf-time-between-balances unsignedtype number]` `[--snmpdf-balance-factor doubletype number]` `[--snmpdf-max-severity-factor doubletype number]` `[--snmpdf-min-severity-factor doubletype number]` `[--snmpdf-min-balances4info unsignedtype number]` `[--snmpdf-ttl seconds]` `[--snmpdf-timeout-text-prefix text]` `[--snmpdf-sender-delay-max unsignedtype number]` `[--snmpdf-severity-growing-factor-4min unsignedtype number]` `[--snmpdf-severity-growing-factor-4max unsignedtype number]` `[--snmpdf-ignored-community text]` `[--out-udp-socketfile filename]` `[--out-udp-port unsignedtype number]` `[--out-host hostname]` `[-P,--pid-file filename]` `[-l,--log-facility log_facility]` `[-L,--log-level log_level]` `[-u,--run-as-user username]` `[--background]` `[--ignore-case-4-severity]` `[-v,--version]` `[-h,--help]`

13.3. Znaczenie opcji w programie damsnpdaud

Tabela 13.1. Znaczenie opcji w programie damsnpdaud

Nazwa opcji	Opis
<code>-f,--socketfile filename</code>	Oczekuj połączeń od klientów poprzez wyspecyfikowany pliku typu socket (domyślnie: <code>/tmp/damsnmpdaud.socket</code>).

Analizator Danych SNMP Data (damsnmpdaud)

Nazwa opcji	Opis
<i>-p,--port unsignedtype_number</i>	Użyj wyspecyfikowanego portu udp do nasłuchiwania zamiast pliku typu socket (domyślnie: użyj pliku typu socket: /tmp/damsnmpdaud.socket).
<i>--snmpdf-average-len doubletype_number</i>	Użyj podanej długości okna przeciętnych wartości każdej danej; długość okien dla wartości maksymalnych i minimalnych jest taka sama i wynosi odpowiednio $(1.0 - \text{długość_okna_przeciętnych_wartości})/2$; ten parametr musi być większy niż 0.0 (0%) i mniejszy niż 1.0 (100%) (domyślnie: 0.9).
<i>--snmpdf-time-between-balances unsignedtype_number</i>	Użyj wyspecyfikowanej liczby sekund, która musi wystąpić pomiędzy balansowaniem każdej danej (domyślnie: 10800).
<i>--snmpdf-balance-factor doubletype_number</i>	Użyj wyspecyfikowanego czynnika balansowania do uczenia się o granicach okien przeciętnych wartości (domyślnie: 0.3).
<i>--snmpdf-max-severity-factor doubletype_number</i>	Użyj wyspecyfikowanego czynnika ważności, który pozwala decydować, czy otrzymana dana powinna być przekazana dalej, kiedy jej wartość jest poniżej lub powyżej długości okna jej przeciętnych wartości pomnożonej przez ten czynnik (jeśli automatyczne balansowanie jest włączone dla niej, domyślnie: 2.0).
<i>--snmpdf-min-severity-factor doubletype_number</i>	Użyj wyspecyfikowanego czynnika ważności, który pozwala decydować, czy otrzymana dana powinna być przekazana dalej, kiedy jej wartość jest poniżej lub powyżej długości okna jej przeciętnych wartości pomnożonej przez ten czynnik (jeśli automatyczne balansowanie jest włączone dla niej, domyślnie 0.5).
<i>--snmpdf-min-balances4info unsignedtype_number</i>	Wysyłaj wyjściowe pakiety UDP, kiedy liczba balansowań każdej danej jest równa lub większa niż podana wartość (domyślnie: 36).
<i>--snmpdf-ttl seconds</i>	Usuwać dane, których czas modyfikacji jest starszy od obecnego o podaną liczbę sekund (domyślnie: 172800).
<i>--snmpdf-timeout-text-prefix text</i>	Podaj prefix tekstu, który będzie poprzedzał (w ID wiadomości) adres IP komputera, który nie odpowiada na zapytania SNMP (domyślnie: 'Timeout for: ').
<i>--snmpdf-sender-delay-max unsignedtype_number</i>	Podaj maksymalną wartość, po której nowy nadawca będzie przypisany jako nowy, godny zaufania nadawca tych danych (domyślnie: 10).
<i>--snmpdf-severity-growing-factor-4min unsignedtype_number</i>	Podaj czynnik wzrostu ważności dla wartości poniżej okien średnich wartości (domyślnie: 4).
<i>--snmpdf-severity-growing-factor-4max unsignedtype_number</i>	Podaj czynnik wzrostu ważności dla wartości powyżej okien średnich wartości (domyślnie: 4).

Nazwa opcji	Opis
<i>--snmpdf-ignored-community text</i>	Podaj community, aby zatrzymać przekazywanie pewnego typu danych: dane zawierające (wielkość liter nie ma znaczenia) to community nie będą przekazywane dalej (domyślnie: ignore).
<i>--out-udp-socketfile filename</i>	Użyj wyspecyfikowanego pliku typu socket dla wyjścia (domyślnie: /tmp/damadbud.socket).
<i>--out-udp-port unsignedtype_number</i>	Wysyłaj wychodzące pakiety UDP na ten port (domyślnie: użyj pliku typu socket '/tmp/damadbud.socket').
<i>--out-host hostname</i>	Wysyłaj wychodzące pakiety UDP do tego komputera (domyślnie: 127.0.0.1).
<i>-P,--pid-file filename</i>	Zapisz swój identyfikator procesu (PID) do wyspecyfikowanego pliku.
<i>-l,--log-facility log_facility</i>	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
<i>-L,--log-level log_level</i>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: warning).
<i>-u,--run-as-user username</i>	Zrezygnuj z praw root'a i uruchom serwer z prawami podanego użytkownika.
<i>--background</i>	Rozpocznij pracę w tle po uruchomieniu.
<i>--ignore-case-4-severity</i>	Wylicz ważność przetwarzanej wiadomości ignorując wielkość liter.
<i>-v,--version</i>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<i>-h,--help</i>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

13.4. Format pliku konfiguracyjnego

Plik konfiguracyjny składa się z definicji obiektów typu `group`. Definicja obiektu jest ograniczona klamrami `{ }`. Słowo kluczowe `group` określające typ obiektu musi być jedynym słowem w linii. Ta sama reguła dotyczy klamry rozpoczynającej i kończącej definicję obiektu. Ogólnie więc pojedynczy obiekt wygląda tak:

```
group
```

```
{
```

```
...  
}
```

Obiekt jest zbudowany z dowolnej ilości par elementów. Na każdy element składają się dwie linie pliku. Pierwsza z nich, zaczynająca się od słowa kluczowego `string`, specyfikuje ciąg znaków, zaś druga, rozpoczynająca się od słowa `value`, jego ważność. Przykładowe elementy wchodzące w skład obiektu typu `group` mogą wyglądać następująco:

```
string=222.222.222.222
```

```
value=0.53
```

```
string=A realy bad news
```

```
value=0.99
```

Liczby określające wartości są traktowane jako zmiennoprzecinkowe z zakresu $\langle 0.0; 1.0 \rangle$.

13.5. Format pliku definiującego poziomy minimalne

Plik konfiguracyjny składa się z definicji obiektów typu `level`. Definicja obiektu jest ograniczona klamrami `{}`. Słowo kluczowe `level` określające typ obiektu musi być jedynym słowem w linii. Ta sama reguła dotyczy klamry rozpoczynającej i kończącej definicję obiektu. Ogólnie więc pojedynczy obiekt wygląda tak:

```
level
```

```
{
```

```
...  
}
```

Pola występujące pomiędzy klamrami są ściśle określone, podobnie jak ich kolejność. Każdy wpis w definicji obiektu zajmuje sam całą linię i ma postać:

```
słowo_kluczowe=
```

Pierwszym polem jest `minvalue4info` określające za pomocą liczby typu `double` minimalną wartość, która musi zostać przekroczona, aby powiadomienie mogło zostać wysłane. Kolejnym i zarazem ostatnim polem jest `name`. Może ono wystąpić jeden lub więcej razy i definiuje tekst, który musi

zawierać się w identyfikatorze badanej danej, aby ta definicja mogła być do niej zastosowana.

13.6. Opis działania

W przypadku, gdy agent oczekujący zgłoszeń klientów poprzez sieć komputerową otrzymuje nowy komunikat UDP, to dokonuje autoryzacji klienta na podstawie jego adresu IP. W tym celu przetwarzane są informacje zawarte w pliku `.known.hosts`. Jeśli nie można otworzyć pliku `.known.hosts` do czytania lub adres IP klienta nie pasuje do żadnego z wpisów w tym pliku, to pakiet UDP jest odrzucany. Dla klientów lokalnych nie jest dokonywana żadna autoryzacja.

Przetwarzanie kolejnego zaakceptowanego komunikatu rozpoczyna się od sprawdzenia, czy przetwarzana dana już istnieje w Bazie Danych **Analizatora Danych SNMP Data**. Zakłada się, że dwie dowolne dane są takie same wtedy i tylko wtedy, gdy ich identyfikatory (identyfikator taki dostaje np. program [damsnmpdi](#) w postaci argumentu opcji `-I`) są sobie równe. Jeżeli nie istnieje to zostaje wpisana do Bazy Danych ze startowymi wartościami takimi jak: czas utworzenia i modyfikacji, liczby wystąpień itp. Jeśli dana występowała już wcześniej w Bazie Danych to uaktualniany jest czas modyfikacji i licznik wystąpień. Jeśli przetwarzana dana jest typu `error` lub `timeout` to zwiększany jest odpowiedni licznik wystąpień, a następnie dana jest przekazywana procedurze, która wysyła dane na wyjście **Analizatora Danych SNMP Data**. W przypadku, gdy dana jest typu `counter` to obliczany jest przyrost wartości od momentu ostatniej aktualizacji tej danej. Przyrost ten staje się wartością badaną w dalszym procesie obróbki tej wiadomości. Kiedy zaś dana jest typu `gauge` to wartością badaną jest rzeczywista wielkość tej danej. Następnie jeśli wystąpią określone warunki informacja o tej danej może zostać przesłana na wyjście **Analizatora Danych SNMP Data**.

Należy zauważyć, że komunikat jeszcze przed aktualizacją danej istniejącej w Bazie może zostać odrzucony, kiedy wystąpią pewne, specyficzne warunki. Kiedy mianowicie identyfikator nadawcy komunikatu jest różny od dotychczasowego nadawcy, to komunikat nie zostanie uwzględniony. Dopiero po wystąpieniu pewnej (argument opcji `--snmpdf-sender-delay-max`) ilości takich komunikatów, podczas gdy dotychczasowy nadawca nic nie przesyła, zostanie przyznany tej danej identyfikator nowego nadawcy.

W następnym etapie przetwarzania danej wartość badana jest porównywana z oknem przeciętnych wartości dla tej danej. Jeśli wartość badana jest mniejsza od dolnej krawędzi tego okna lub większa od jego górnej krawędzi to jest przekazywana procedurze, która wysyła dane na wyjście **Analizatora Danych SNMP Data**.

Procedura balansująca uruchamiana jest niezależnie od otrzymywanych i przetwarzanych danych. Sprawdza ona najpierw, czy dla wskazanej danej jest dozwolony proces balansowania (dla dowolnej danej można bowiem ustawić sztywne granice okna przeciętnych, tzn. akceptowalnych wartości, które nie będą ulegać zmianie). Jeśli minął ustawiony wcześniej czas od momentu ostatniego balansowania dla tej danej to proces ten może zostać znów powtórzony. Przy pierwszym balansowaniu granice (`window_min, window_max`) okna przeciętnych wartości są obliczane według wzoru:

```
delta = (max - min) * (1.0 - len) / 2.0 * factor
```

```
window_min = min + delta
```

```
window_max = max - delta
```

gdzie `max` oznacza maksymalną wartość z dotychczas występujących dla badanej danej, zaś `min` wartość minimalną. Parametr `len` odpowiada argumentowi opcji `--snmpdf-average-len` programu **damsnmpdaud**, zaś `factor` argumentowi opcji `--snmpdf-balance-factor` programu **damsnmpdaud**. Przy kolejnych balansowaniach granice okna przeciętnych wartości są wyliczane następująco:

```
delta1 = (window_max - window_min) * ((1.0 - len) / 2.0 - hits_min / hits_after_balance) * factor
```

```
delta2 = (window_max - window_min) * ((1.0 - len) / 2.0 - hits_max / hits_after_balance) * factor
```

```
window_min = window_min + delta1
```

```
window_max = window_max + delta2
```

gdzie `hits_min` oznacza ilość wystąpień wartości poniżej dolnej granicy okna od czasu ostatniego balansowania, zaś `hits_max` oznacza analogicznie ilość wystąpień powyżej górnej granicy okna. Parametr `hits_after_balance` oznacza ilość wszystkich wystąpień tej danej od czasu ostatniego balansowania. Po aktualizacji granic okna zerowane są liczniki trafień w poszczególne przedziały, a także licznik wszystkich trafień po balansowaniu.

Procedura, która wysyła dane na wyjście **Analizatora Danych SNMP Data**, sprawdza najpierw (w przypadku, gdy wysyłana dana jest innego typu niż `timeout` lub `error`), czy dla tej danej wystąpiła już odpowiednią ilość razy procedura balansowania (jeśli dana ta podlega automatycznemu balansowaniu). Jeżeli ta wstępna weryfikacja przebiegła pomyślnie to w przypadku, gdy dana ta podlega automatycznemu balansowaniu i nie jest typu `timeout` ani `error` to sprawdzane jest, czy przekroczenie którejs z granicy okna przeciętnych wartości jest "istotnie duże". Przez zwrot, że jej wartości są poniżej wyznaczonego dla niej poziomu, rozumie się, że identyfikator tej danej pasuje do pewnego zdefiniowanego obiektu typu `level` (tzn. wszystkie łańcuchy znaków pól `name` zdefiniowanych w tym obiekcie są zawarte w identyfikatorze tej danej) oraz średnia arytmetyczna granic okna przeciętnych wartości tej danej i wartość bezwzględna jej wartości są mniejsze od wartości tego obiektu. W celu stwierdzenia, czy nastąpiło "istotnie duże" przekroczenie granicy okna przeciętnych wartości obliczana jest wartość według wzoru:

```
window_min * min_severity_factor
```

gdzy wartość danej jest poniżej dolnej granicy okna oraz jest ona typu `counter` lub:


```
window_min - (window_min - min) * min_severity_factor
```

gdy wartość danej jest poniżej dolnej granicy okna oraz jest ona typu gauge, zaś według wzoru:

```
window_max + (window_max - window_min) * max_severity_factor
```

gdy wartość danej jest powyżej górnej granicy okna. Parametr `min_severity_factor` odpowiada argumentowi opcji `--snmpdf-min-severity-factor` programu **damsnmpdaud**, zaś `max_severity_factor` odpowiada argumentowi `--snmpdf-max-severity-factor`. W pierwszych dwóch przypadkach, gdy wartość jest większa lub równa obliczonej wielkości, dana nie jest wysyłana, zaś w ostatnim przypadku dana nie jest wysyłana, gdy jej wartość jest mniejsza bądź równa obliczonej wielkości.

W przypadku, gdy dana nie została odrzucona przez procedurę wysyłającą obliczana jest jej ważność, a następnie jest ona przesyłana na wyjście **Analizatora Danych SNMP Data**. Typowym odbiorcą tych wiadomości jest **Baza Danych Skojarzeń**.

W procesie oceny ważności komunikatu brany jest pod uwagę tylko identyfikator wiadomości (identyfikator ten dostaje np. program `damsnmpdi` w postaci argumentu opcji `-l`). Jednoznacznie identyfikuje on wiadomość w **Operation Manager**. Początkowa ważność komunikatu wynosi zero. Następnie dla każdego elementu z pliku konfiguracyjnego jest sprawdzane, czy łańcuch znaków występujący po słowie kluczowym `string=` jest zawarty w identyfikatorze badanego komunikatu. Jeśli tak to brana jest pod uwagę maksymalna wartość przyporządkowana elementowi `string`. Wartość ta następnie jest dodawana do dotychczas obliczonej ważności komunikatu. Na końcu obliczona ważność komunikatu jest dzielona przez ilość obiektów typu `group` zdefiniowanych w pliku konfiguracyjnym. W przypadku gdy komunikat nie jest ani typu `timeout` ani `error` to obliczane jest o ile procent został przekroczony poziom krawędzi okna średnich wartości w stosunku do wartości krańcowych. Dolny kraniec stanowi minimum dwóch liczb: najmniejszej wartości jaka wystąpiła dla tej danej lub zero.

Dla wartości poniżej okna ważność jest liczona następująco:

```
factor * (1.0 - (x - min) / (window_min - min))
```

lub

```
factor * (1.0 - (x - min) / (max - min))
```

gdzie `x` jest przetwarzaną wartością, `window_min` jest dolną krawędzią okna średnich wartości, zaś `min` jest dolnym, a `max` górnym krańcem wartości. Parametr `factor` jest argumentem opcji `--snmpdf-severity-growing-factor-4min`. Wartość jest liczona według drugiego wzoru, gdy odległość między `window_max` a `window_min` jest mniejsza od 1% odległości pomiędzy `max` a `min`.

Dla wartości powyżej okna ważność jest liczona następująco:

```
factor * (1.0 - (max - x) / (max - window_max))
```

lub

```
factor * (1.0 - (x - min) / (max - min))
```

gdzie x jest przetwarzaną wartością, $window_max$ jest górną krawędzią okna średnich wartości, zaś max jest górnym, a min dolnym krańcem wartości. Parametr $factor$ jest argumentem opcji [--snmpdf-severity-growing-factor-4max](#). Wartość jest liczona według drugiego wzoru w przypadku analogicznym jak dla wartości poniżej okna. Pilnowany jest także fakt, aby ważność nie przekroczyła jedynki. Na końcu ważność przyjmuje znak ujemny.

Procedura serwisowania Bazy Danych uruchamiana jest co pewien okres czasu i polega na usuwaniu danych, które już od pewnego czasu (argument opcji [--snmpdf-ttl seconds](#)) nie zostały zaktualizowane (zmodyfikowane). Tylko te dane zostaną usunięte z Bazy Danych, które posiadają atrybut pozwalający na tą operację.

13.7. Tematy pokrewne

[Dostęp do Interfejsu Danych SNMP Data \(damsnmpdi\)](#)

[Jednostka Tworząca Bazę Danych Aktywnych Spraw \(damadbud\)](#)

Rozdział 14. Jednostka Tworząca Bazę Danych Aktywnych Spraw (damadbud)

14.1. Opis ogólny

Program **damadbud** jest **Jednostką Tworzącą Bazę Danych Aktywnych Spraw** i jest częścią **Operation Manager-a**. Jest to proces daemon, który pracuje bez przerwy w czasie działania systemu i oczekuje na dane wysyłane przez [Analizatory Danych SNMP Data](#) i [SNMP Trap](#). Na podstawie swojej konfiguracji buduje powiązania pomiędzy otrzymywanymi komunikatami. Każdy otrzymany komunikat przekazuje dalej. Ich dedykowanym odbiorcą jest [Serwis Aktywnych Spraw](#).

Podczas startu **damadbud** czyta swój plik konfiguracyjny `.damadbudrc`. Program spodziewa się go znaleźć w katalogu `/etc/david-system`.

14.2. Składnia

Program **damadbud** można uruchomić z następującymi opcjami: `[-f,--in-socketfile]` `[-p,--in-port unsignedtype_number]` `[-P,--pid-file filename]` `[-l,--log-facility log_facility]` `[-L,--log-level log_level]` `[-u,--run-as-user username]` `[--background]` `[-v,--version]` `[-h,--help]`

14.3. Znaczenie opcji w programie damadbud

Tabela 14.1. Znaczenie opcji w programie damadbud

Nazwa opcji	Opis
<code>-f,--in-socketfile</code>	Oczekuj połączeń od klientów poprzez wyspecyfikowany pliku typu socket (domyślnie: <code>/tmp/damadbud.socket</code>).
<code>-p,--in-port unsignedtype_number</code>	Użyj wyspecyfikowanego portu udp do nasłuchiwanie zamiast pliku typu socket (domyślnie użyj pliku typu socket: <code>/tmp/damadbud.socket</code>).
<code>-P,--pid-file filename</code>	Zapisz swój identyfikator procesu (PID) do wyspecyfikowanego pliku
<code>-l,--log-facility log_facility</code>	Wybierz typ logowania (log facility) do syslogd: <code>daemon</code> <code>user</code> <code>local0</code> ... <code>local7</code> (domyślnie: <code>local6</code>).
<code>-L,--log-level log_level</code>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: <code>emerg</code> <code>alert</code> <code>crit</code> <code>err</code> <code>warning</code> <code>notice</code> <code>info</code> <code>debug0</code> ... <code>debug2</code> (domyślnie: <code>notice</code>).
<code>-u,--run-as-user username</code>	Zrezygnuj z praw root'a i uruchom serwer z prawami podanego użytkownika.

Nazwa opcji	Opis
<code>--background</code>	Rozpocznij pracę w tle po uruchomieniu.
<code>-v,--version</code>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<code>-h,--help</code>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

14.4. Format pliku konfiguracyjnego

Plik konfiguracyjny składa się z obiektów, którego typ wskazuje słowo kluczowe użyte w jego nazwie, zaś jego specyfikacja jest ograniczona klamrami `{ }`. Obecnie jedynym dozwolonym typem obiektu jest `association`. Słowo kluczowe określające typ obiektu musi być jedynym słowem w linii. Ta sama reguła dotyczy klamry rozpoczynającej i kończącej definicję obiektu. Ogólnie więc pojedynczy obiekt wygląda tak:

```
association
```

```
{
...
}
```

Pola występujące pomiędzy klamrami są ściśle określone, podobnie jak ich kolejność. Każdy wpis w definicji obiektu zajmuje sam całą linię i ma postać:

```
słowo_kluczowe=
```

Pierwszym polem jest `var`, za którym musi być podany łańcuch znaków określający nazwę zmiennej, na którą można się będzie powoływać w dalszej części definicji obiektu. Pole `var` może wystąpić zero lub więcej razy i w ten sposób definiować wiele zmiennych. Dwa ostatnie pola występujące w definicji obiektu to `string1` oraz `string2`. Odnoszą się one do identyfikatorów komunikatów, które próbują być ze sobą skojarzone. Przykładowa definicja obiektu wygląda następująco:

```
association
```

```
{
var=$1
string1=link $1 up
string2=link $1 down
}
```

i zgodnie z tą definicją dwie wiadomości o identyfikatorach odpowiednio:

```
information: link serial1/1 up in router
```

```
information: link serial1/1 down in router
```

pasują do siebie, zaś o identyfikatorach:

```
information: link serial1/2 up in router
```

```
information: link serial1/1 down in router
```

już do siebie nie pasują.

14.5. Opis działania

W przypadku, gdy agent oczekujący zgłoszeń klientów poprzez sieć komputerową otrzymuje nowy komunikat UDP, to dokonuje autoryzacji klienta na podstawie jego adresu IP. W tym celu przetwarzane są informacje zawarte w pliku `.known.hosts`. Jeśli nie można otworzyć pliku `.known.hosts` do czytania lub adres IP klienta nie pasuje do żadnego z wpisów w tym pliku, to pakiet UDP jest odrzucony. Dla klientów lokalnych nie jest dokonywana żadna autoryzacja.

Dla każdego przetwarzanego komunikatu przeszukiwana jest lista aktywnych wiadomości, w celu sprawdzenia, czy dany komunikat już się na niej nie znajduje. Dwa komunikaty uznawane są za takie same wtedy i tylko wtedy, gdy są tego samego typu (SNMP Data lub SNMP Trap) i spełniają warunki specyficzne dla danego typu. W przypadku typu SNMP Trap muszą być równe ich identyfikatory, zaś w przypadku typu SNMP Data dodatkowo muszą mieć ten sam powód wysłania komunikatu przez [Analizatora Danych SNMP Data](#) (powodem może być: wartość poniżej dolnej granicy okna przeciętnych wartości, wartość powyżej górnej granicy okna, błąd lub nie otrzymanie odpowiedzi). Jeśli komunikat znajduje się na liście aktywnych wiadomości to następuje jego aktualizacja, a następnie informacja o jego nadejściu wysyłana jest na wyjście programu **damadbud**.

W przypadku, gdy komunikat nie został znaleziony na liście aktywnych wiadomości, to są przeszukiwane wszystkie komunikaty. Jeśli zostanie on odnaleziony, to zostanie zaktualizowany i stanie się aktywny. Jeśli komunikat nie zostanie odnaleziony, to zostanie dodany do listy wiadomości jako aktywny komunikat. Ostatnią operacją dokonywaną na przetwarzanym komunikacie jest wysłanie informacji o jego nadejściu na wyjście programu **damadbud**.

W przypadku dodawania nowego elementu do Bazy Danych sprawdzane jest, czy nie powinien on być skojarzony z jakimkolwiek innym komunikatem z Bazy Danych zgodnie z wczytaną podczas startu konfiguracją z pliku `.damadbudrc`. W tym celu przetwarzany jest każdy komunikat z Bazy bez względu na to, czy znajduje się na liście aktywnych wiadomości, czy też nie. Dla każdej takiej pary sprawdzane jest, czy ich identyfikatory nie pasują do definicji któregoś z obiektów `association{...}` znalezionych w pliku konfiguracyjnym. W tym celu badany jest każdy ze zdefiniowanych obiektów. Jeśli `string1` występuje w pierwszym komunikacie, zaś `string2` w

drugim, przy warunku potraktowania wskazanych części identyfikatorów jako zmiennych zgodnie z definicją pól `string1` i `string2`, to komunikaty te zostaną ze sobą skojarzone.

Program **damadbud** przegląda także aktywne komunikaty. Te z nich, które nie były aktualizowane przez ostatnie 15 min. są ustawiane w stan pasywny. Co kilka minut natomiast, uruchamiana jest procedura usuwania pasywnych komunikatów.

14.6. Tematy pokrewne

[Analizator Danych SNMP Trap \(damsnmptaud\)](#)

[Analizator Danych SNMP Data \(damsnmpdaud\)](#)

[Serwis Aktywnych Spraw \(damcsud\)](#)

Rozdział 15. Serwis Aktywnych Spraw (damcsud)

15.1. Opis ogólny

Program **damcsud** jest **Serwisem Aktywnych Spraw** i jest częścią **Operation Manager-a**. Jest to proces daemon, który pracuje bez przerwy w czasie działania systemu i oczekuje na dane wysyłane przez [Jednostkę Tworzącą Bazę Danych Aktywnych Spraw](#) oraz serwisuje Aktywne Sprawy. Na podstawie wiadomości zgromadzonych w **Bazie Danych Skojarzeń** stara się konsolidować otrzymywane komunikaty tworząc w ten sposób sprawy obsługiwane przez system. Jego praca polega na usuwaniu przestarzałych spraw z **Bazy Danych Aktywnych Spraw** oraz na ewentualnym uruchamianiu dla nich wyspecyfikowanych programów. Podczas startu **damcsud** czyta swój plik konfiguracyjny `.damcsudrc`. Program spodziewa się go znaleźć w katalogu `/etc/david-system`. Daemon **damcsud** nie uruchomi się jeśli nie będzie mógł odczytać swojej konfiguracji.

15.2. Składnia

Program **damcsud** można uruchomić z następującymi opcjami: `[-P,--pid-file filename]` `[-l,--log-facility log_facility]` `[-L,--log-level log_level]` `[--max-load-level double_number]` `[--max-period-4-load-checking seconds]` `[--max-lock seconds]` `[--case-ttl seconds]` `[-u,--run-as-user username]` `[--background]` `[-v,--version]` `[-h,--help]`

15.3. Znaczenie opcji w programie damcsud

Tabela 15.1. Znaczenie opcji w programie damcsud

Nazwa opcji	Opis
<code>-P,--pid-file filename</code>	Zapisz swój identyfikator procesu (PID) do wyspecyfikowanego pliku.
<code>-l,--log-facility log_facility</code>	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
<code>-L,--log-level log_level</code>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: warning).
<code>--max-load-level double_number</code>	Zawieś wykonywania akcji kiedy średnie obciążenie systemu operacyjnego jest wyższe niż ta wartość (domyślnie: 20.0). Pośmiertne akcje wciąż będą wykonywane, podobnie jak te, które mają być wykonane tylko raz oraz te, których okres powtarzania jest większy

Nazwa opcji	Opis
	niż argument opcji <code>--max-period-4-load-checking</code> .
<code>--max-period-4-load-checking seconds</code>	Wykonaj akcję ignorując obciążenie systemu operacyjnego jeśli jej czas powtarzania (w sekundach) jest większy niż ta wartość (domyślnie: 60).
<code>--max-lock seconds</code>	Nie blokuj Bazy Danych dla obsługiwanych spraw (to service cases) dłużej niż podany odcinek czasu (domyślnie: 2).
<code>--case-ttl seconds</code>	Ustaw Time to Live (w sekundach) dla każdej sprawy od czasu jej modyfikacji (domyślnie: 180).
<code>-u,--run-as-user username</code>	Zrezygnuj z praw root'a i uruchom serwer z prawami podanego użytkownika.
<code>--background</code>	Rozpocznij pracę w tle po uruchomieniu
<code>-v,--version</code>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<code>-h,--help</code>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

15.4. Format pliku konfiguracyjnego

Plik konfiguracyjny składa się z obiektów, których typ wskazuje słowo kluczowe użyte w ich nazwie, zaś ich specyfikacja jest ograniczona klamrami `{}`. Obecnie jedynym dozwolonym typem obiektu jest `action`. Słowo kluczowe określające typ obiektu musi być jedynym słowem w linii. Ta sama reguła dotyczy klamry rozpoczynającej i kończącej definicję obiektu. Ogólnie więc pojedynczy obiekt wygląda tak:

```
action
```

```
{
```

```
...
```

```
}
```

Pola występujące pomiędzy klamrami są ściśle określone, podobnie jak ich kolejność. Każdy z pierwszych 6 wpisów w definicji obiektu zajmuje sam całą linię i ma postać:

```
słowo_kluczowe=
```

Pierwszym polem jest `id`, za którym musi być podany identyfikator (liczba typu `unsigned long`) obiektu. Na drugim miejscu występuje pole `delaytime`, które wyznacza liczbę sekund jaka musi

upłynąć od chwili utworzenia sprawy, aby dana akcja mogła być uruchomiona. Kolejne pole `repeatingtime` wskazuje, co jaką liczbę sekund dana akcja może zostać powtórzona. Powtarzanie i występowanie akcji pomaga uściślić kolejne pole `period`. Jego wartości są typu wyliczeniowego. Dozwolone są następujące wartości:

`after_death`

`after_join`

`after_split`

`once`

Każda inna wartość tego pola (w tym pusta) jest rozumiana jako wartość `always`.

Ostatnim wpisem mogącym wystąpić więcej niż jeden obowiązkowy raz jest obiekt `program_spec`. Jego definicja jest ograniczona klamrami `{}`, które muszą, podobnie jak słowo kluczowe `program_spec`, wystąpić same w osobnych liniach. Pierwszym wpisem tego obiektu jest `program`, który wyznacza ścieżkę do programu jaki winien być uruchomiony dla tej akcji jeśli zostaną spełnione określone warunki. ostatnim polem mogącym wystąpić więcej niż jeden obowiązkowy raz jest pole `name`. Specyfikuje ono łańcuch znaków, który musi wystąpić chociaż w jednym zdarzeniu zawartym w sprawie, aby dana akcja była dla niej uruchomiona. W przypadku, gdy pól `name` jest więcej niż jedno, to wszystkie muszą wystąpić w pojedynczym zdarzeniu, aby akcja mogła być uruchomiona.

Schemat definicji każdego obiektu wygląda więc następująco:

```
action
{
id=
delaytime=
repeatingtime=
period=
program_spec
{
program=
name=
...
}
```

```
}  
...  
}
```

Znaczenie wartości poszczególnych pól obiektu akcji jest omawiane przy okazji przedstawiania działania programu **damcsud**.

15.5. Opis działania

W momencie otrzymania nowego komunikatu sprawdzane jest, czy nie występuje już jako zdarzenie w jakiejś z istniejących spraw. Jeśli występuje to zwiększany jest licznik wystąpień zdarzenia, a stan zdarzenia zmienia się na aktywny (może nie być żadnej zmiany, gdyż wcześniej stan zdarzenia mógł być także aktywny). Jeśli nie zostało znalezione żadne zdarzenie odpowiadające komunikatowi, to tworzone jest nowe zdarzenie. Licznik wystąpień zdarzenia jest ustawiony na jeden, stan zdarzenia zostaje ustalony jako aktywny. Następnie sprawdzane są wszystkie zdarzenia w kolejnych sprawach, czy któreś z nich nie jest powiązane z przetwarzaną wiadomością, tzn. czy zostało utworzone takie powiązanie przez program [damadbud](#). W przypadku, gdy zostało znalezione takie skojarzenie, to wówczas przetwarzane zdarzenie jest dołączane do danej sprawy. Jeśli nie to tworzona jest nowa sprawa zawierająca na razie jedno zdarzenie.

Pierwszą czynnością serwisującą jest sprawdzenie, czy dana sprawa nie powinna zostać zniszczona, tzn. usunięta z Bazy Danych. Jeśli nie to sprawdzane jest, czy nie powinny dla danej sprawy zostać uruchomione jakieś akcje (programy) i ewentualnie są one uruchamiane.

Poniżej zostały bliżej przedstawione poszczególne procedury.

15.6. Uruchamianie akcji (programów)

Sprawdzenie, czy dla danej sprawy należy uruchomić jakąś akcję, jest procedurą zależną od wielu parametrów. Dla danej sprawy przeglądane są wszystkie zdefiniowane akcje (zbudowane w oparciu o plik konfiguracyjny wczytywany podczas uruchamiania programu **damcsud**) i sprawdzane są kryteria, czy dana akcja może zostać uruchomiona dla tej sprawy. Akcja nie zostanie uruchomiona, jeśli ilość sekund jaka minęła od czasu utworzenia sprawy jest mniejsza niż wartość parametru `delaytime` akcji. Kolejną procedurą sprawdzającą, czy dana akcja może zostać uruchomiona dla tej sprawy jest sprawdzenie, czy wszystkie wartości parametrów `name` dla jakiegokolwiek obiektu `program_spec` występują w ID chociaż jednego, aktywnego zdarzenia wchodzącego w skład sprawy. W przypadku, gdy dana akcja ma być uruchomiona jako ostatnia dla tej sprawy, wtedy występowanie wartości parametrów `name` danej akcji jest przeprowadzane dla każdego zdarzenia (także nieaktywnego) wchodzącego w skład sprawy. Jeśli nie wystąpią wszystkie wartości parametrów `name`, dla żadnego obiektu `program_spec`, wtedy dana akcja nie jest uruchamiana. Następnie sprawdzane jest, czy upłynął określony przez parametry `repeatingtime` czas od momentu utworzenia sprawy. Jeśli nie, to akcja nie jest uruchamiana.

Następnie sprawdzana jest zgodność parametru `period` danej akcji z obecnym stanem sprawy. Jeśli nie ma to być ostatnia akcja dla tej sprawy, to sprawdzane jest, czy dana akcja już w tej chwili nie przebiega dla tej sprawy. Jeśli tak to nie jest uruchamiana kolejny raz. Nie zostanie również uruchomiona, gdy wartość parametru `period` danej akcji została zdefiniowana jako `once`, a już była wcześniej dla tej sprawy uruchomiona. Jeśli od czasu ostatniego uruchomienia danej akcji dla tej sprawy minął czas określony parametrem `repeatingtime`, to zostanie ona uruchomiona ponownie. Dana akcja zostanie również uruchomiona, jeśli będzie uruchomiona po raz pierwszy, lub jako ostatnia akcja dla tej sprawy.

15.7. Informacje przekazywane uruchamianym akcjom (programom)

Program, który ma zostać uruchomiony, jako swoje parametry uruchomienia (poza tymi wyspecyfikowanymi w polu `program` danej akcji), otrzymuje informacje dotyczące całej sprawy i każdego zdarzenia, które wchodzi w jego skład. Rodzaje parametrów dla każdego zdarzenia wchodzącego w skład sprawy zależą od jego rodzaju, tzn. czy jego źródłową wiadomością jest dana typu SNMP Trap, czy SNMP Data. Wszystkie jednak parametry są poprzedzone słowami kluczowymi zaczynającymi się podwójnym lub pojedynczym znakiem minus. Wszystkie programy otrzymują poniższe parametry:

Tabela 15.2. Parametry uruchamianych akcji (programów)

Nazwa parametru	Opis
--flow-severity severity	Wewnętrzna ważność sprawy.
--flow-id ID	Numer sprawy w danym dniu
--action-id ID	Numer akcji (wartość pola id danej akcji).
--opened	Informacja o tym, że sprawa została otworzona (uruchamiane programy decydują, co to znaczy; zazwyczaj jest to powiadomienie operatora). Parametr ten nie wystąpi, jeśli sprawa nie została zaznaczona jako otwarta.
--community community	Suma wszystkich communities wchodzących w skład, każdego zdarzenia wchodzącego w skład sprawy.
--closing-reason description	Informacja o przyczynie zamknięcia sprawy. Parametr ten nie wystąpi, jeśli nie został podczas wcześniejszych akcji podany. Jednak dla akcji uruchamianej po śmierci sprawy, nigdy nie będzie pusty, gdyż domyślną wartością jest tekst "Not specified".
--closed-by uid	Identyfikator osoby zamykającej sprawę. Parametr ten nie wystąpi, jeśli nie został podczas wcześniejszych akcji podany. Jednak dla akcji uruchamianej po śmierci sprawy, nigdy nie będzie pusty, gdyż domyślną wartością jest liczba "-1".
--action-done numer_akcji:kod_powrotu	Numer akcji oraz kod ostatniego jej powrotu (argument może wystąpić zero lub więcej razy).
--flow-ctime seconds	Czas utworzenia sprawy (w sekundach od początku ery UNIX-owej).

Nazwa parametru	Opis
--flow-mtime seconds	Czas modyfikacji sprawy (w sekundach od początku ery UNIX-owej).

Poniższe argumenty występują dla każdego zdarzenia wchodzącego w skład danej sprawy. Może ich więc być więcej niż jeden pełen zbiór podany poniżej:

Tabela 15.3. Argumenty występujące dla każdego zdarzenia wchodzącego w skład danej sprawy

Nazwa argumentu	Opis
--event-hitsnumber number	Ilość wystąpień tego zdarzenia podczas trwania tej sprawy.
--event-state state	Stan zdarzenia, gdzie parametr state może przybrać jedną z poniższych wartości: active - aktywna, passive - pasywna.
--event-id number	Numer jednoznacznie identyfikujący zdarzenie.
--event-ctime seconds	Czas utworzenia zdarzenia (w sekundach od początku ery UNIX-owej).
--event-mtime seconds	Czas modyfikacji zdarzenia (w sekundach od początku ery UNIX-owej).
--event-type type	Rodzaj komunikatu, do którego odnosi się zdarzenie. Możliwe są dwa poniższe typy wartości: snmptrap, snmpdata.

W przypadku, kiedy typem jest wartość `snmptrap`, wówczas występują poniższe parametry:

Tabela 15.4. Argumenty występujące dla zdarzenia typu snmptrap

Nazwa argumentu	Opis
-T seconds	Czas otrzymania komunikatu (w sekundach od początku ery UNIX-owej).
-s IP	Źródłowy adres komunikatu
-a IP	Adres IP agenta SNMP, który wysłał wiadomość.
-c community	Nawa wspólnoty wiadomości.
-n OID	Identyfikator komunikatu w bazie MIB.
-e OID	Identyfikator producenta w bazie MIB.
-t timestamp	Znacznik czasowy nadawcy komunikatu.
-g generic	Ogólny numer komunikatu
-p specific	Specyficzny numer komunikatu
-d string	Identyfikator komunikatu w Operation Manager .
-m message	Czytelna dla człowieka wiadomość wyjaśniająca znaczenie komunikatu.

W przypadku, kiedy typem jest wartość `snmpdata`, wówczas występują poniższe parametry:

Tabela 15.5. Argumenty występujące dla zdarzenia typu `snmpdata`

Nazwa argumentu	Opis
-f type	Powód przesłania wiadomości do Bazy Danych Skojarzeń . Może przybrać jedną z następujących wartości: min - wartość poniżej dolnej krawędzi okna średnich wartości tej danej, max - wartość powyżej górnej krawędzi okna średnich wartości tej danej, error - błąd dla tej danej, timeout - przekroczony czas oczekiwania na odpowiedź dla tej danej.
-T seconds	Czas otrzymania wiadomości (w sekundach od początku ery UNIX-owej).
-s IP	Adres odpytywanego urządzenia.
-o OID	Identyfikator danej w bazie MIB.
-t type	Typ danej będący jedną z poniższych wartości: counter - licznik, gauge - dowolna wartość, error - błąd, timeout - przekroczony czas oczekiwania na odpowiedź, other - żaden z powyższych typów.
-v value	Ostatnia wartość danej.
-a value	Dolna krawędź okna średnich wartości tej danej.
-A value	Górna krawędź okna średnich wartości tej danej.
-d string	Identyfikator komunikatu w Operation Manager .
-m message	Czytelna dla człowieka wiadomość wyjaśniająca znaczenie komunikatu.

15.8. Wyniki działania zakończonych akcji (programów)

Program **damcsud** spodziewa się, że każda z uruchomionych akcji (program) może wypisać rezultat działania na swoje standardowe wyjście. Jeśli nic nie zostanie wypisane na standardowe wyjście, wtedy taka akcja nie będzie mogła wpłynąć na stan sprawy, dla której została uruchomiona.

Rezultat działania akcji jest przekazywany w postaci linii tekstu. Każda linia odpowiada pojedynczemu zdarzeniu wchodzącemu w skład danej sprawy, lub jeśli składa się tylko ze słowa `opened`, to znaczy, że sprawa powinna zostać zaznaczona jako otwarta (następne akcje uruchamiane dla tej sprawy będą otrzymywać argument `--opened`). Innymi dwoma wyjątkami są linie rozpoczynające się od słów `closing-reason` oraz `closed-by`. W pierwszym przypadku zapamiętany zostanie tekst występujący po słowie kluczowym, który następnie będzie podawany kolejnym akcjom jako argument opcji `--closing-reason`. Analogiczna sytuacja nastąpi w drugim przypadku, gdzie zapamiętany identyfikator stanie się argumentem opcji `--closed-by` kolejnych akcji. Każda linia odpowiadająca pojedynczemu zdarzeniu składa się z dwóch elementów: identyfikatora zdarzenia (argument każdej opcji `--event-id`) oraz kodu będącego wartością typu `long` (32 bitowa liczba ze znakiem). Jeśli kod jest wartością dodatnią lub zerem, to stan tego zdarzenia jest ustawiany na `active`. W przeciwnym wypadku stan zdarzenia jest

ustawiany na [passive](#). Ze wszystkich kodów dla każdego zdarzenia, jakie wygenerowała akcja wybierany jest kod maksymalny nie mniejszy jednak niż -1 i jest on uznawany za kod powrotu całej akcji.

Jeśli kod powrotu jest liczbą dodatnią lub zerem, to czas modyfikacji danej sprawy jest ustawiany na chwilę bieżącą, od której dodawana jest liczba sekund równa kodowi powrotu akcji (tzn. czas modyfikacji danej sprawy jest młodszy od chwili bieżącej o liczbę sekund równą kodowi powrotu akcji).

Na końcu operacji zapamiętywany jest kod powrotu tej akcji dla danej sprawy.

15.9. Niszczenie spraw

Procedura sprawdzająca, czy dana sprawa może zostać już zniszczona, oblicza czas, jaki upłynął od ostatniej modyfikacji sprawy. Sprawa nie zostanie zniszczona, jeśli wartość ta będzie mniejsza niż argument opcji [--case-ttl](#) programu **damcsud** lub jakakolwiek akcja będzie wciąż uruchomiona dla danej sprawy. Sprawa nie zostanie także zniszczona, gdy dla jakiegokolwiek zdarzenia typu SNMP Data wchodzącego w jej skład, jego ostatnia wartość, dostarczona [Analizatorowi Danych SNMP Data](#), będzie poza zakresem średnich wartości i nie będzie starsza niż 30 minut.

Tuż przed ostatecznym zniszczeniem danej sprawy może zostać uruchomiona akcja spełniająca określone warunki tej sprawy. Może zostać uruchomiona co najwyżej jedna taka akcja.

15.10. Tematy pokrewne











[Jednostka Tworząca Bazę Danych Aktywnych Spraw \(damadbud\)](#)

Rozdział 16. Przyciski najczęściej występujące w aplikacjach WWW





16.1. Znaczenie przycisków

W poniższej tabeli zostały zebrane przyciski, które w aplikacjach WWW występują najczęściej. Ich funkcja w poszczególnych aplikacjach jest zbliżona, a czasem nawet identyczna. Niektóre z nich mogą jednak pełnić dodatkowe funkcje, które przy okazji omawiania poszczególnych aplikacji.

Tabela 16.1. Przyciski najczęściej występujące w aplikacjach WWW

Przycisk	Opis
	Powoduje powrót do widoku poprzedniej strony.
	Generalnie powoduje usunięcie elementu tzn. np: powoduje zamknięcie sprawy (case), ustawienie zdarzenia (event) w stan pasywny itp.
	Podwołuje przejście do edycji danego elementu.
	Najczęściej oznacza zatwierdzenie operacji i przejście do jej wykonania (np.: wygenerowanie raportu używając wybranych kryteriów).
	Powoduje przejście do widoku szczegółowego.
	Pozwala przejść do wyższego poziomu w hierarchii elementów.
	Otwiera nowe okno z danymi przygotowanymi do wydruku.
	Powoduje przejście do prezentacji wykresu z danymi dla danego elementu (Przeglądarka Kolekcji).
	Odświeża widok strony.
	Powoduje akceptację zmienionych wartości jako aktualnych.

Przyciski najczęściej występujące w aplikacjach WWW

Przycisk	Opis
	Powoduje przejście do raportu dla danego elementu (Przeglądarka Raportów o Węzłach).
	Powoduje przejście do przeglądarki raportów o Trapach dla danego elementu (Przeglądarka Trapów).
	Powoduje przejście do przeglądarki raportów o sprawach (cases) dla danego elementu (Przeglądarka Zarejestrowanych Spraw).
	Zachowuje zmiany dokonane przez użytkownika.

Rozdział 17. Przeglądarka Zarejestrowanych Spraw (Recorded Operation Browser)

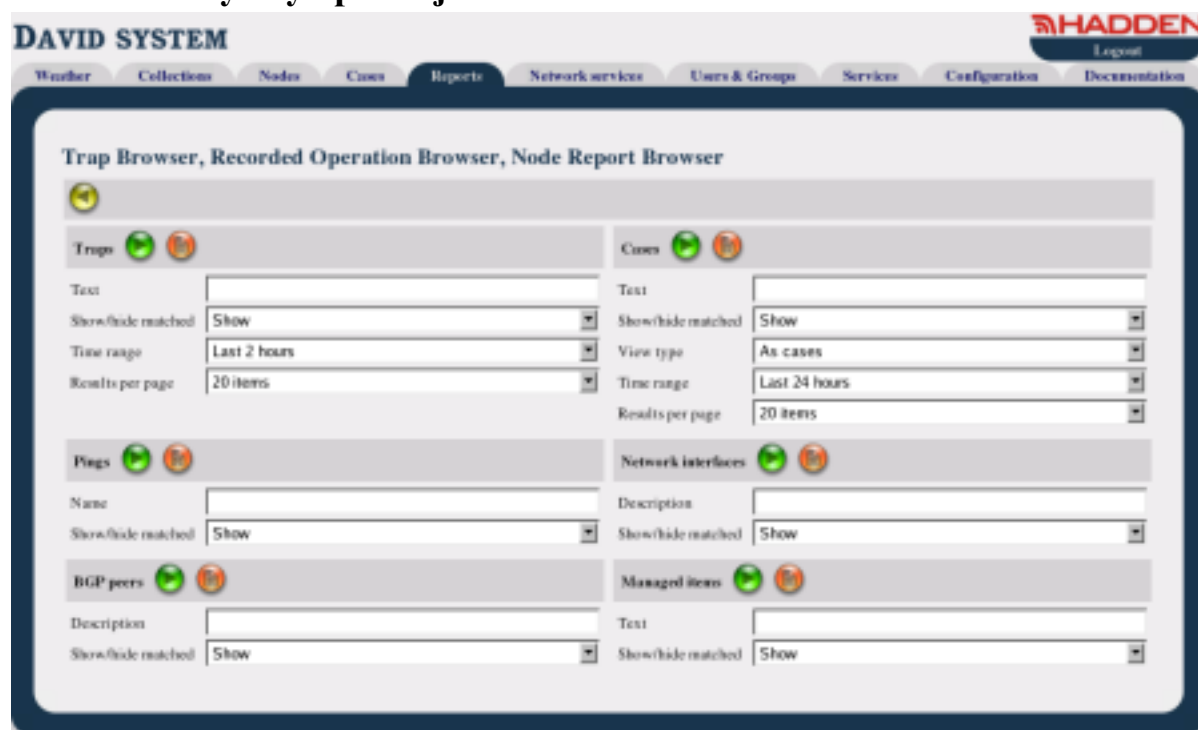
17.1. Opis ogólny

Przeglądarka Zarejestrowanych Spraw jest aplikacją WWW oraz częścią **Operation Manager-a**. Pozwala ona przeglądać zarejestrowane informacje o sprawach (cases) prowadzonych przez **Operation Manager-a**.

17.2. Opis działania


17.2.1. Specyfikacja kryteriów wyszukiwania

17.2.1.1. Widok domyślny aplikacji



Przeglądarka Zarejestrowanych Spraw jest dostępna poprzez zakładkę Reports. Jest to grupa Cases tej zakładki. Podając kryteria wyboru w grupie Cases dostajemy listę zarejestrowanych informacji na temat przetwarzanych spraw.

17.2.1.2. Grupa Cases

Cases 

Text

Community

Source device

Show/hide matched

View type

Time range

Date and period

Results per page

Each bar occupies

Graph's width

Graph's height

Limit graph results (case view only)

Pola zawarte w grupie oraz ich znaczenie zostały przedstawione w poniższej tabeli.

Tabela 17.1. Przeglądarka Zarejestrowanych Spraw - znaczenie pól widoku szczegółowego grupy Cases

Pole	Znaczenie
Text	Czytelny dla człowieka opis sprawy.
Community	Etykieta sprawy złożona z etykiet zdarzeń wchodzących w jej skład (pojedyncze etykiety oddzielone są od siebie znakiem '!').
Source device	Urządzenie będące źródłem sprawy.
Show/hide matched	Pole to oznacza, czy sprawy spełniające kryteria wyszukiwania mają zostać pokazane, czy ukryte.
View type	Możliwe są dwa sposoby pokazywania listy zarejestrowanych spraw: wybranie <i>As cases</i> powoduje grupowanie wpisów na temat tej samej sprawy w jeden ciąg informacji; wybranie <i>As log entries</i> powoduje traktowanie każdego wpisu niezależnie i jest raczej mniej naturalnym sposobem przeglądania informacji o występujących sprawach.
Time range	Specyfikacja zakresu czasowego poprzez wybranie odcinka czasu trwającego do chwili obecnej.
Date and period	Specyfikacja zakresu czasowego poprzez wybranie konkretnej daty i czasu w połączeniu z określonym odcinkiem czasu (np.: 1 dzień, 2 dni, 1 tydzień

Pole	Znaczenie
	itp.).
Results per page	Określa maksymalną ilość wpisów pokazanych za jednym razem.
Each bar occupies	Wskazanie ile czasu ma obejmować pojedynczy odcinek czasu na wykresie rozkładu występowania spraw w czasie.
Graph's width	Szerokość wykresu.
Graph's height	Wysokość wykresu.
Limit graph results	Dla widoku As cases ustawia, czy generowany wykres długości trwania poszczególnych spraw ma obejmować wszystkie wybrane sprawy, czy aktualnie prezentowane (ograniczone parametrem Results per page).

17.2.2. Wygenerowany raport

17.2.2.1. Widok typu As cases

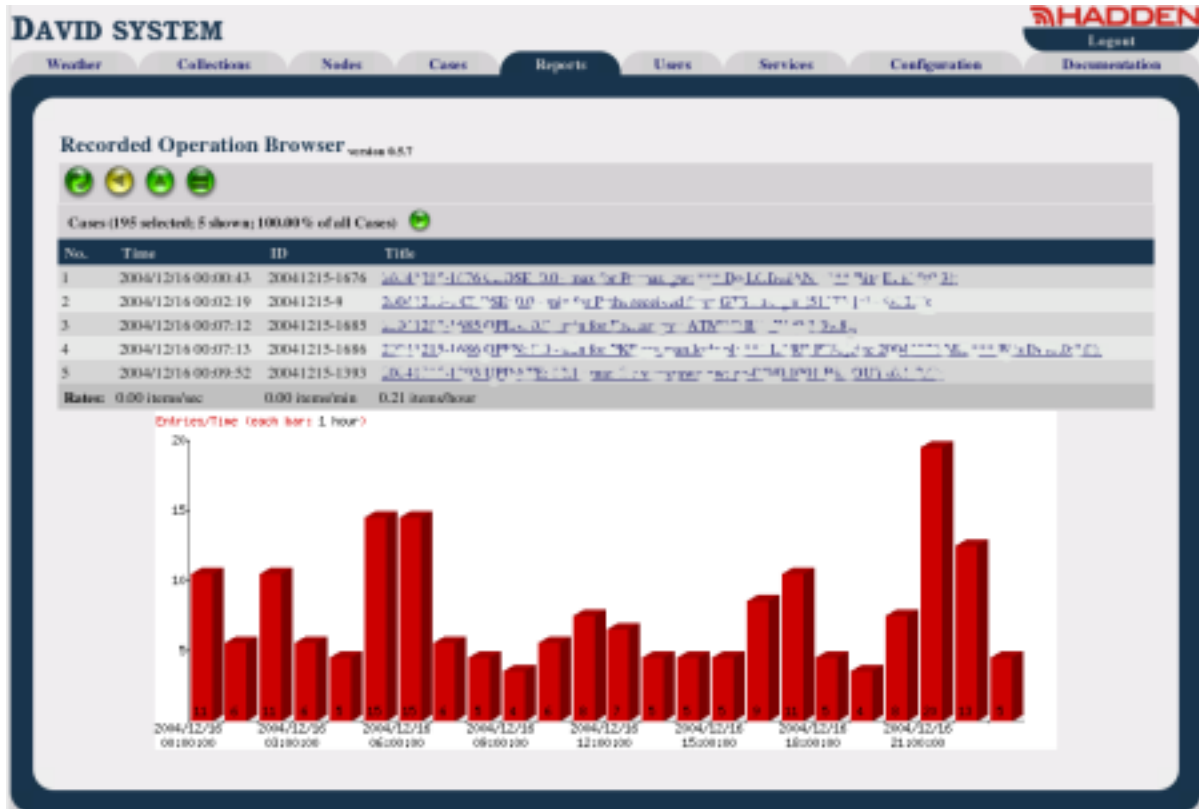


Widok typu As cases jest naturalnym sposobem przeglądania informacji o występujących sprawach. Główne wiersze zawierające numery porządkowe podają informacje o pierwszym wpisie na temat danej sprawy. Jeśli wpisów jest więcej niż jeden, to wówczas pojawi się wiersz dodatkowy zawierający informację o ostatnim wpisie na temat danej sprawy. Kolumna ID zawiera identyfikatory spraw. Linki w kolumnie Title pozwalają dotrzeć do bardziej szczegółowych informacji na temat danej sprawy

zaczynając od listy wszystkich wpisów na jej temat. Kolumna Duration podaje czas począwszy od pierwszego, widocznego wpisu na temat danej sprawy do ostatniego (często jest to czas trwania sprawy). Kolumna Percent of period obrazuje procentowy udział wielkości pola Duration danej sprawy do wybranego, podczas generowania raportu, odcinka czasu.

Poniżej listy znajduje się wykres obrazujący rozpiętość czasową pomiędzy pierwszym a ostatnim wpisem na temat każdej z prezentowanych spraw.

17.2.2.2. Widok typu As log entries



Widok typu As log entries traktuje każdy wpis niezależnie i jest mniej naturalnym sposobem przeglądania informacji o występujących sprawach. W pewnych wypadkach może być jednak przydatnym sposobem spojrzenia na dany problem. Kolumna ID zawiera identyfikatory spraw. Linki w kolumnie Title pozwalają dotrzeć do bardziej szczegółowych informacji na temat danej sprawy zaczynając od listy wszystkich wpisów na jej temat.

Poniżej wyświetlanej listy znajduje się wykres pokazujący ilość wpisów o sprawach, spełniających kryteria wyboru, w wybranym odcinku czasu.

17.3. Tematy pokrewne

[Dostęp do Interfejsu Danych SNMP Trap \(damsnmpti\)](#)

[Analizator Danych SNMP Trap \(damsnmptaud\)](#)

[Dostęp do Interfejsu Danych SNMP Data \(damsnmpdi\)](#)

[Analizator Danych SNMP Data \(damsnmpdaud\)](#)

[Jednostka Tworząca Bazę Danych Aktywnych Spraw \(damadbud\)](#)

[Serwis Aktywnych Spraw \(damcsud\)](#)

Rozdział 18. Przeglądarka Toczących się Spraw (Pending Operation Browser)

18.1. Opis ogólny


Przeglądarka Toczących się Spraw jest aplikacją WWW oraz częścią **Operation Manager-a**. Zapewnia ona dostęp do aktualnie toczących się spraw oraz do innych danych działającego produktu.

18.2. Opis działania

Przeglądarka Toczących się Spraw jest dostępna poprzez zakładkę Cases. Jest ona podzielona na grupy dające dostęp do poszczególnych rodzajów danych. Grupa Statistics pozwala obejrzeć dane statystyczne poszczególnych modułów wchodzących w skład produktu i ma raczej znaczenie diagnostyczne. Grupa Cases daje dostęp do aktualnie prowadzonych spraw, zaś grupa Collected data umożliwia dostęp do danych, kolekcjonowanych przez [Analizator Danych SNMP Data](#).

18.2.1. Widok aktywnych spraw


18.2.1.1. Specyfikowanie kryteriów wyszukiwania

Cases 

Text

Community

State

Grupa zawiera pole Text pozwalające podać szukany tekst jaki występuje w opisie toczących się spraw. Pole Community pozwala przeszukiwać sprawy pod kątem zawartości tekstu w ich polach Community. Pole State pozwala wybrać te sprawy, które aktualnie znajdują się w wybranym stanie. Kliknięcie na przycisk  uruchamia proces wyszukiwania spraw.

18.2.1.2. Lista aktywnych spraw



The screenshot shows the 'Pending Operation Browser' interface within the 'DAVID SYSTEM' application. The interface includes a navigation menu with options like 'Weather', 'Collections', 'Nodes', 'Cases', 'Reports', 'Network services', 'Users & Groups', 'Services', 'Configuration', and 'Documentation'. The 'Cases' tab is selected. Below the navigation, there are three status icons (green, yellow, red) and a table of pending operations.

No.	State	ID	Updates	Description
1	🟡	20070716-59	🟢	IP filter violation on interface...
2	🟡	20070717-337	🟢	IP filter violation on interface...
3	🟡	20070717-338	🟢	IP filter violation on interface...
4	🟡	20070717-340	🟢	Connection (eth0)...
5	🟡	20070717-341	🟢	Link down...

Lista jest podzielona na kolumny charakteryzujące poszczególne sprawy. Kolumna State sygnalizuje stan w jakim dana sprawa się aktualnie znajduje, zaś ID podaje jej identyfikator. Kolumna Updates pokazuje, czy wysyłanie update-ów dla danej sprawy jest zablokowane, czy nie. Kolumna Description pokazuje opis sprawy. Kliknięcie na pole ID lub pole Description powoduje wyświetlenie szczegółowych danych o wybranej sprawie.

DAVID SYSTEM **HADDEN**
Logout

Weather Collections Nodes **Cases** Reports Network services Users & Groups Services Configuration Documentation

Pending Operation Browser version 0.9.3


ID:	20070717-345
State:	ACTIVE
Creation time:	2007/07/17 09:09:58
Modification time:	2007/07/17 09:09:58
Touch time:	2007/07/17 09:09:58
Service time:	2007/07/17 09:11:24
Severity:	-0.300
Community:	
Send updates:	Yes
Closing reason:	
Closed by:	

Events			Actions		
State	Hits	Description	ID	State	Calls
!	1	IP filter violation on interface...			

Poza wartościami poszczególnych pól opisujących daną sprawę, znajduje się tu lista zdarzeń wchodzących w skład tej sprawy oraz lista akcji, które były lub są uruchomione dla tej sprawy.

Lista zdarzeń jest podzielona na kolumny opisujące stan zdarzenia (kolumna *State*), liczbę wystąpień danego zdarzenia od początku istnienia sprawy (kolumna *Hits*) oraz jego znaczenie (kolumna *Description*).

Lista akcji jest podzielona na kolumny: *ID* - identyfikator akcji pochodzący z pliku konfiguracyjnego modułu [damcsud](#), *State* - aktualny stan akcji (DONE lub RUNNING) i *Calls* - liczba uruchomień danej akcji.

Przycisk  umieszczony na pasku narzędziowym, powyżej opisu danej sprawy, pozwala ją zamknąć.

Kliknięcie na pole *Description* pokazuje widok szczegółowy wybranego zdarzenia wchodzącego w skład danej sprawy.


The screenshot shows the 'Pending Operation Browser' interface. At the top, there is a navigation menu with tabs: Weather, Collections, Nodes, Cases, Reports, Users, Services, Configuration, and Documentation. The 'Cases' tab is selected. Below the navigation, the title 'Pending Operation Browser version 0.5.1' is displayed. There are four status icons: a green circle with a checkmark, a green circle with a play button, a red circle with a stop button, and a yellow circle with a warning sign. The 'Case:' field contains a long alphanumeric string. Below this, there are three sections: 'Running event part', 'Event part', and 'SNMP Trap part'. Each section contains a table of properties and values.

Running event part	
State:	PASSIVE
Hits:	1307
Creation time:	2005/01/07 16:06:42
Modification time:	2005/01/09 21:51:04

Event part	
Severity:	-0.350
Event state:	ACTIVE
Event hits:	12981
Event creation time:	2004/12/29 16:12:26
Event modification time:	2005/01/09 21:51:04


SNMP Trap part	
Receiving time:	2005/01/09 21:51:04

Pole Case opisuje skrótowo sprawę, do której należy wybrane zdarzenie. Kolejne pola opisują właściwości poszczególnych części zdarzenia powstałych w procesie jego tworzenia zaczynając od komunikatu SNMP Trap lub SNMP Data.

Przycisk  umieszczony na pasku narzędziowym, powyżej opisu danego zdarzenia, pozwala ustawić stan zdarzenia na pasywny.

18.2.2. Widok gromadzonych danych SNMP Data

18.2.2.1. Specyfikowanie kryteriów wyszukiwania

Collected data 

Text

Community

Device type

Source device

Last modified within


Last value

Significant values only

Pola zawarte w grupie oraz ich znaczenie zostały przedstawione w poniższej tabeli.

Tabela 18.1. Przeglądarka Toczących się Spraw - znaczenie pól grupy Collected data

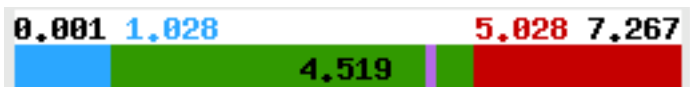
Pole	Znaczenie
Text	Czytelny dla człowieka opis danej.
Community	Etykieta przypisana danej.
Device type	Typ urządzenia źródłowego, z którego pochodzi dana.
Source device	Urządzenie źródłowe, z którego pochodzi dana.
Last modified within	Odcinek czasu, w którym dana była ostatnio modyfikowana.
Last value	Przedział wartości, do którego dana może być zaliczona (w przedziale średnich wartości, poza tym przedziałem, poniżej przedziału, powyżej przedziału).
Significant values only	Wskazuje, czy wyświetlane mają być dane o zbiorze wartości poniżej wielkości ignorowanej.

Kliknięcie na przycisk  uruchamia proces wyszukiwania kolekcjonowanych danych, których poprawnych wartości uczy się [Analizator Danych SNMP Data](#).

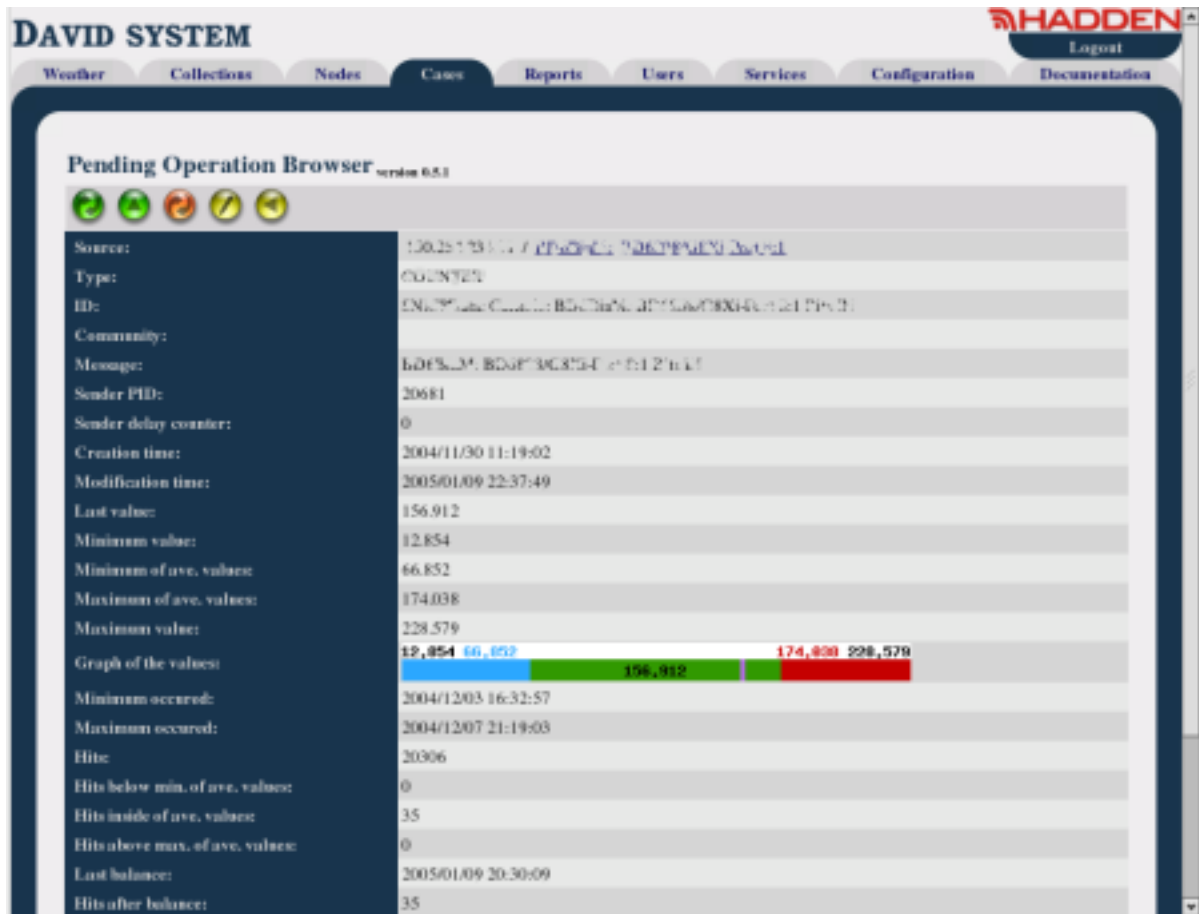
18.2.2.2. Lista danych SNMP Data spełniająca kryteria wyszukiwania




Lista jest podzielona na kolumny. Kolumna State pokazuje, czy ostatnia wartość danej znajduje się poniżej zakresu średnich wartości (kolor fioletowy), czy mieści się w zakresie (kolor zielony), czy też powyżej tego zakresu (kolor czerwony). Kolumna Source pokazuje element źródłowy danej (np.: urządzenie i interfejs sieciowy, którego dana dotyczy). Kolumna Modification time podaje czas ostatniej modyfikacji danej. Kolumna Last value pokazuje położenie ostatniej wartości danej w zakresie jej średnich wartości. Wykres pokazuje wartości: minimalną i maksymalną w kolorze czarnym, minimalną spośród średnich wartości w kolorze niebieskim i maksymalną w kolorze czerwonym oraz ostatnią (aktualną) w kolorze czarnym, umieszczonym w środkowej części wykresu, i dodatkowo narysowaną jako fioletową kreskę.



Kolumna Description opisuje w sposób czytelny dla człowieka znaczenie danej oraz zawiera link, którego kliknięcie powoduje wyświetlenie szczegółowych informacji na temat wybranego elementu



Poza dokładnymi wartościami poszczególnych pól danej typu SNMP Data znajduje się tu również wizualizacja okna przeciętnych wartości, których progów nauczył się system w trakcie swojej pracy. Konfiguracja systemu pozwala, aby dla wskazanych danych system nie uczył się ich przeciętnych wartości, lecz aby te wartości mogły być określone przez użytkownika.

Kliknięcie na przycisk  umożliwia przejście do trybu edycji wybranej danej typu SNMP Data.

Użytkownik może wtedy określić granice okna przeciętnych wartości.

Rozdział 19. Prezenter Powiadomień Graficznych (xdgnp)

19.1. Opis ogólny

Program **xdgnp** jest **Prezenterem Powiadomień Graficznych** i jest częścią **Operation Manager-a**. Pozwala ona wyświetlać okienka graficzne o sprawach, o których informacje dostaje od serwera [dgnsd](#) i reagować na te powiadomienia operatorowi. Służy również do odtwarzania plików dźwiękowych i wiadomości tekstowych.

19.2. Składnia

Program **xdgnp** można uruchomić z następującymi opcjami: [[-l,--log-facility log_facility](#)] [[-L,--log-level log_level](#)] [[-v,--version](#)] [[-h,--help](#)]

19.3. Znaczenie opcji w programie xdgnp

Tabela 19.1. Znaczenie opcji w programie xdgnp

Nazwa opcji	Opis
<i>-l,--log-facility log_facility</i>	Wybierz typ logowania (log facility) do syslogd: daemon user local0 ... local7 (domyślnie: local6).
<i>-L,--log-level log_level</i>	Wybierz poziom logowania (na standardowe wyjście błędów i do daemona syslogd), tzn. będą wypisywane komunikaty wybranego poziomu oraz komunikaty poziomów ważniejszych: emerg alert crit err warning notice info debug0 ... debug2 (domyślnie: warning).
<i>-v,--version</i>	Wypisz numer wersji na swoje standardowe wyjście błędów i zakończ.
<i>-h,--help</i>	Wypisz komunikat o użyciu na swoje standardowe wyjście błędów i zakończ.

19.4. Opis działania

19.4.1. Rozpoczynanie i kończenie pracy

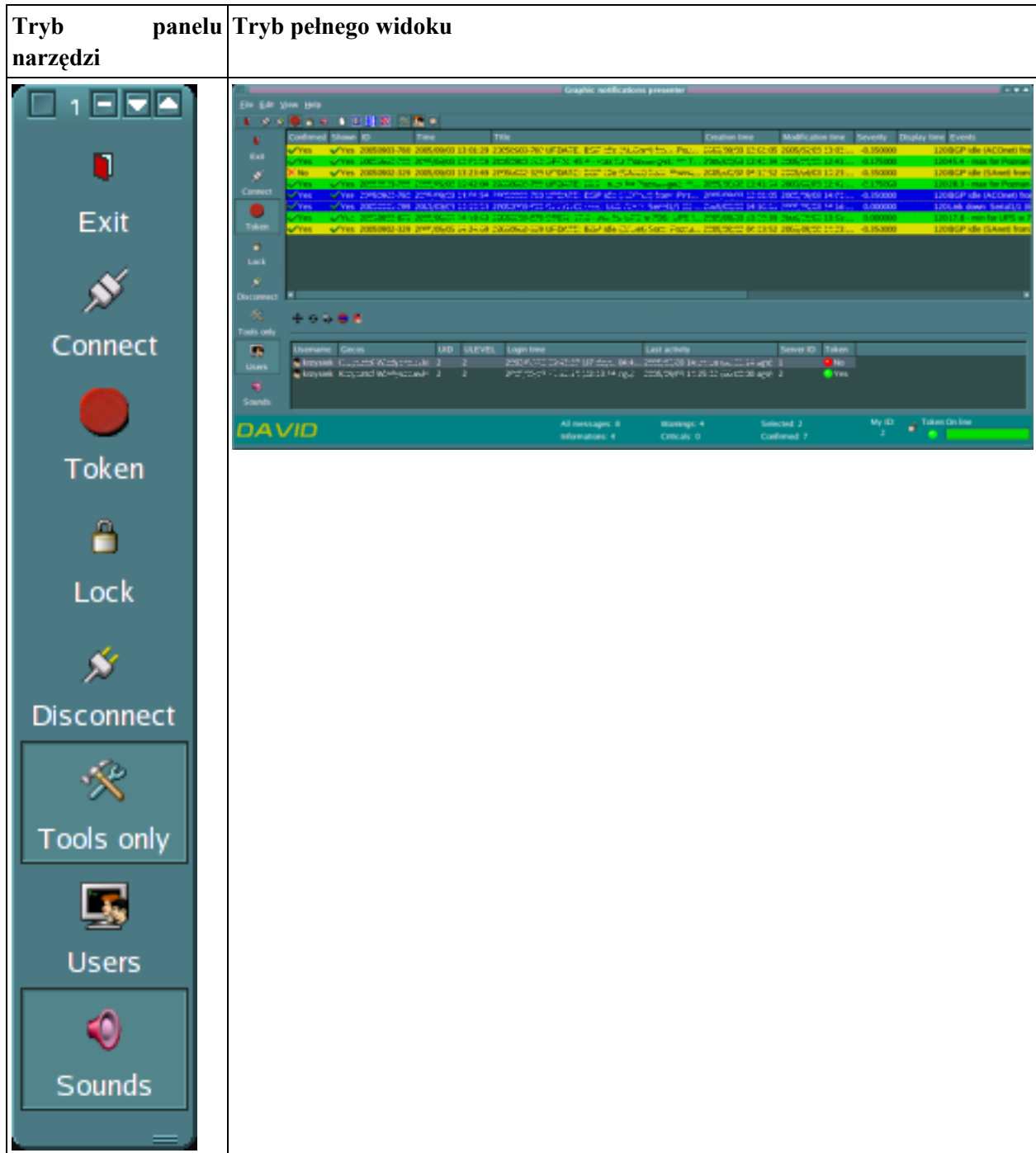
Aplikacja **xdgnp** podczas startu odczytuje swoje parametry konfiguracyjne, dotyczące swojego wyglądu jak również niektórych parametrów pracy z pliku `.xdgnprc`. Aplikacja spodziewa się znaleźć ten plik w katalogu, którego nazwa przechowywana jest w zmiennej środowiskowej `$DAVIDPRIVDIR`. W

przypadku kiedy taki plik nie istnieje aplikacja rozpoczyna pracę z domyślnymi ustawieniami.

19.4.2. Opis działania głównego okna aplikacji

Aplikacja może pracować w dwóch trybach prezentacji głównego widoku. Domyślnym jest prezentacja aplikacji w pełnym widoku z paskami narzędzi, stanu i listą otrzymanych powiadomień. Drugim możliwym trybem prezentacji jest widok tylko panelu narzędzi dzięki czemu aplikacja nie zajmuje dużo miejsca na desktopie. Pomiedzy oboma widokami bardzo łatwo można przełączać się podczas pracy aplikacji.

Tabela 19.2. Aplikacja xdgnp - tryby pracy



Widok pełny w swojej centralnej części pokazuje listę otrzymanych powiadomień. Poszczególne kolumny opisują:



Tabela 19.3. Aplikacja xdgnp - opis poszczególnych kolumn

Prezenter Powiadomień Graficznych (xdgnp)

Nazwa kolumny	Opis
Confirmed	Fakt, czy okienko z powiadomieniem zostało zamknięte przez operatora i tym samym przyjęcie wiadomości zostało potwierdzone.
Shown	Fakt, czy okienko z powiadomieniem zostało pokazane, a jeśli nie to opisany jest powód odmowy jego wyświetlenia.
ID	Unikalny identyfikator sprawy.
Time	Czas otrzymania powiadomienia.
Creation time	Czas utworzenia sprawy.
Modification time	Czas modyfikacji sprawy.
Severity	Ważność sprawy.
Display time	Maksymalny czas jaki mogło być wyświetlane dane okienko informujące o sprawie.
Title	Tytuł sprawy.
Events	Zdarzenia wchodzące w skład sprawy.

Na pasku statusu pokazywane są liczniki elementów prezentowanej listy powiadomień. Poszczególne liczniki pokazują: liczbę wszystkich wiadomości, liczbę informacji, liczbę ostrzeżeń, liczbę ostrzeżeń krytycznych, liczbę elementów zaznaczonych oraz liczbę powiadomień potwierdzonych. W dalszej części paska statusu znajduje się identyfikator tej aplikacji jaki został przydzielony jej przez serwer [dgnsd](#) oraz kilka opcji opisanych poniżej.

Tabela 19.4. Aplikacja xdgnp - opis przycisków

Przycisk	Opis
	Ikona wskazująca, czy aplikacja jest obecnie zablokowana.
	Ikona wskazująca, czy aplikacja jest obecnie odblokowana.

Kolejnym elementem umieszczonym na pasku statusu jest lampka sygnalizująca, czy ta aplikacja jest właścicielem tokena na serwerze [dgnsd](#). Ostatnim elementem jest wskaźnik obecnego statusu połączenia aplikacji z serwerem.

Lewy, skrajny obszar aplikacji zajmuje panel narzędzi, którego przyciski w większości pokrywają się z przyciskami paska narzędziowego oraz opcjami menu pełnego widoku aplikacji. Przyciskiem, który występuje tylko na panelu narzędzi jest przycisk `Tools only`, który przełącza aplikację między dwoma rodzajami jej widoku.







Naciśnięcie prawego przycisku myszy nad listą otrzymanych powiadomień powoduje ukazanie się listy wszystkich menu dostępnych także w górnej części aplikacji. Naciśnięcie lewego przycisku myszy nad elementem listy otrzymanych powiadomień powoduje zaznaczenie lub odznaczenie danego elementu.

Lista ta bowiem umożliwia zaznaczenie wielu elementów naraz.

19.4.2.1. Znaczenie przycisków funkcyjnych w głównym oknie aplikacji








W głównym widoku aplikacji na pasku narzędziowym znajdują się przyciski, za pomocą których można sterować pracą aplikacji. Pierwsze pięć przycisków znajdujących się w lewej części paska narzędziowego odpowiadają opcjom z menu `File`.


Tabela 19.5. Aplikacja xdgnp - opis przycisków funkcyjnych

Przycisk	Opis
	Umożliwia wyjście z programu
	Umożliwia połączenie się z serwerem dgnsd w celu zalogowania się.
	Umożliwia przerwanie połączenia z serwerem (wylogowanie się).
	Pozwala przejąć token.
	Pozwala zablokować dostęp do aplikacji. Wówczas przy jakimkolwiek naciśnięciu klawisza myszy lub klawiatury, lub poruszeniu myszy pojawi się okienko <code>Authorization</code> , w którym należy wpisać nazwę użytkownika i hasło, aby odblokować dostęp do aplikacji i zalogować się tym samym ponownie do serwera (nie trzeba się logować jako ten sam użytkownik).
	Włącza/wyłącza obsługę dźwięku.

Kolejne przyciski pokrywają się z menu `Edit` i menu `View`.

Tabela 19.6. Opis przycisków aplikacji xdgnp pokrywających się z menu Edit i menu View

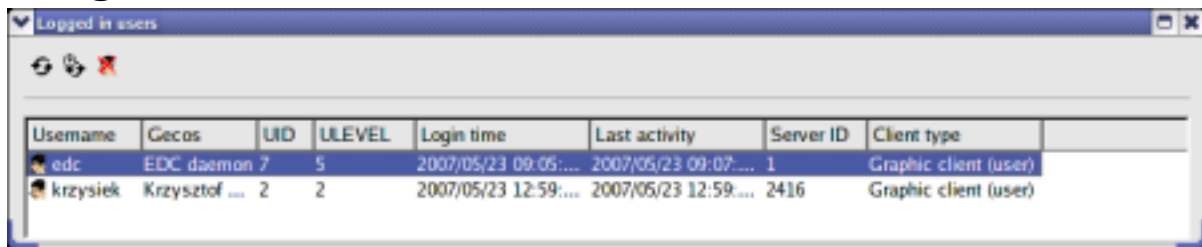
Przycisk	Opis
	Pozwala usunąć całą listę powiadomień.
	Pozwala na usunięcie wybranych wierszy z powiadomieniami.
	Pozwala na odwrócenie zaznaczenia wierszy listy komunikatów.
	Umożliwia odznaczenie wszystkich wcześniej zaznaczonych komunikatów.
	Umożliwia wyświetlenie lub schowanie panela narzędzi, znajdującego się z lewej strony głównego widoku aplikacji.
	Pozwala otworzyć panel edycyjny komentarzy do otrzymanych powiadomień.
	Umożliwia otworenie okienka prezentującego listę zalogowanych użytkowników na serwerze dgnsd .

Przycisk	Opis
	Pozwala skonfigurować parametry pracy aplikacji.

W menu View znajdują się jeszcze dwie opcje - Show tool bar oraz Show status bar - za pomocą których można wyświetlić lub schować pasek narzędziowy oraz pasek statusu.

Poprzez menu Help można dowiedzieć się o wersji i czasie powstania aplikacji.






19.4.3. Okienko prezentujące listę zalogowanych użytkowników na serwerze dgnsd



Username	Gecos	UID	ULEVEL	Login time	Last activity	Server ID	Client type
edc	EDC daemon 7	5	5	2007/05/23 09:05:...	2007/05/23 09:07:...	1	Graphic client (user)
krzysiek	Krzysztof ...	2	2	2007/05/23 12:59:...	2007/05/23 12:59:...	2416	Graphic client (user)

Okienko ukazujące listę zalogowanych użytkowników na serwerze [dgnsd](#), za pomocą przycisków, opisanych poniżej, pozwala na kilka dodatkowych operacji.

Tabela 19.7. Opis przycisków okienka ukazującego listę zalogowanych użytkowników na serwerze dgnsd

Przycisk	Opis
	Umożliwia zmianę okienka dialogowego w panel w głównym oknie aplikacji i na odwrót.
	Pozwala odświeżyć listę zalogowanych użytkowników.
	Pozwala włączać lub wyłączać automatyczne odświeżanie listy użytkowników co 10 sekund.
	Umożliwia przekazywanie tokena wskazanemu użytkownikowi z listy.
	Umożliwia wylogowanie wskazanego użytkownika.


Zasady przekazywania tokena i wylogowywania użytkowników zostały opisane przy okazji opisanie pracy serwera [dgnsd](#).

Poszczególne kolumny listy użytkowników opisują:

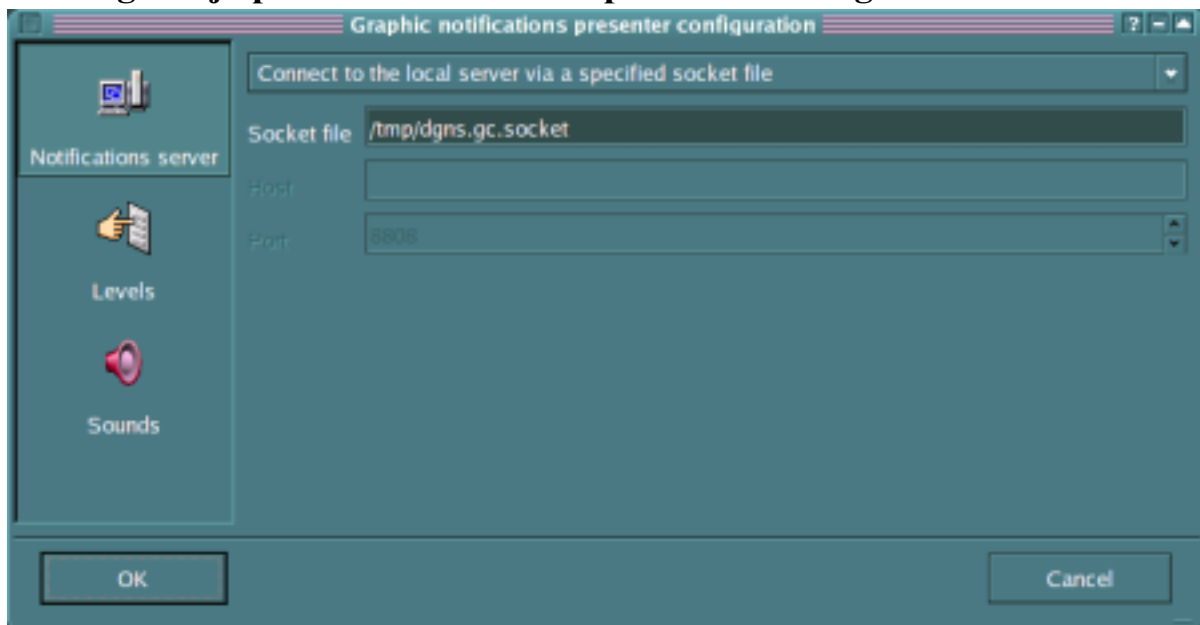
Tabela 19.8. Opis kolumn listy użytkowników

Nazwa kolumny	Opis
Username	Nazwa użytkownika.
Gecos	Opis konta.
UID	Identyfikator (numer) użytkownika.
ULEVEL	Poziom uprawnień użytkownika.
Login time	Czas zalogowania się jako dany użytkownik (nie musi być równy czasowi połączenia się z serwerem, jeśli autentykacja odbywała się więcej niż jeden raz).
Last activity	Czas ostatniej aktywności użytkownika (danej instancji aplikacji) na serwerze dgnsd rozumianej jako przesłanie jakiegokolwiek komunikatu do serwera.
Server ID	Identyfikator danej instancji aplikacji na serwerze dgnsd (ten sam użytkownik może być zalogowany za pomocą więcej niż jednej instancji aplikacji xdgnp).
Token	Wskazuje, czy dany użytkownik (konkretnie: dana instancja aplikacji) posiada token.

19.4.4. Konfiguracja aplikacji xdgnp

Weiskając przycisk  można uruchomić okienko konfiguracyjne, które jest podzielone na trzy obszary konfiguracji.

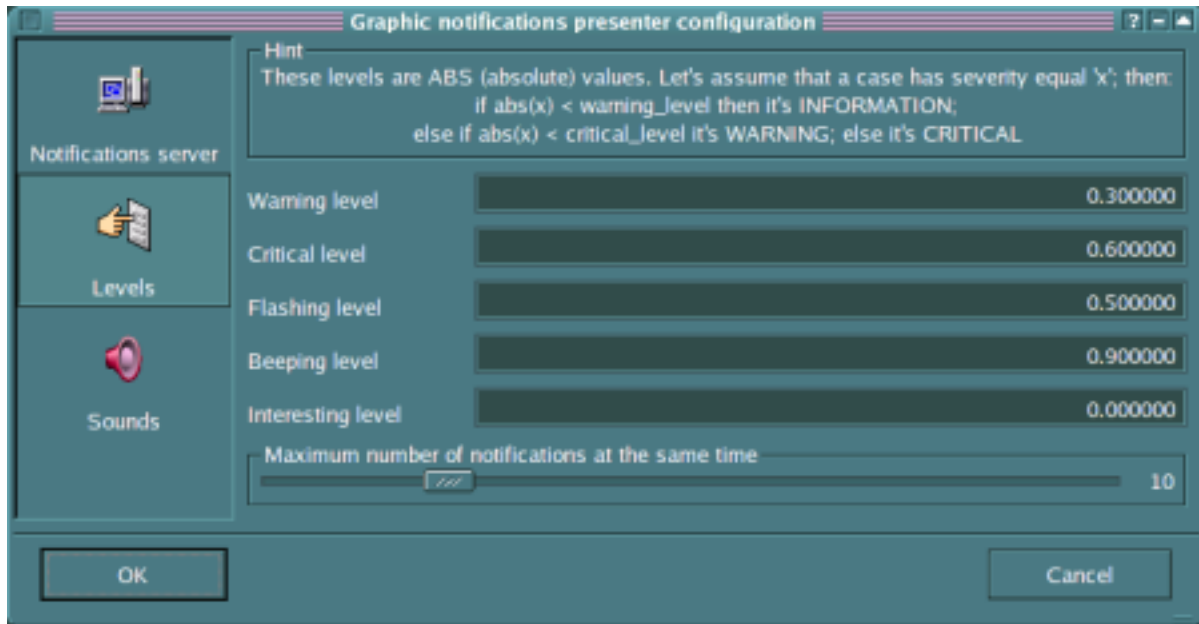
19.4.4.1. Konfiguracja parametrów serwera powiadomień dgnsd



U góry okienka znajduje się lista z wyborem dwóch możliwości specyfikacji parametrów połączenia z

serwerem [dgnsd](#). Jedna opcja umożliwia połączenie się z serwerem poprzez plik typu socket, którego nazwę można wpisać poniżej. Druga zaś wskazuje, że chcemy połączyć się z serwerem poprzez sieć komputerową. Należy wtedy podać nazwę komputera lub jego adres IP oraz numer portu TCP, na którym pracuje serwer na podanym komputerze.

19.4.4.2. Konfiguracja poziomów pracy aplikacji xdgnp

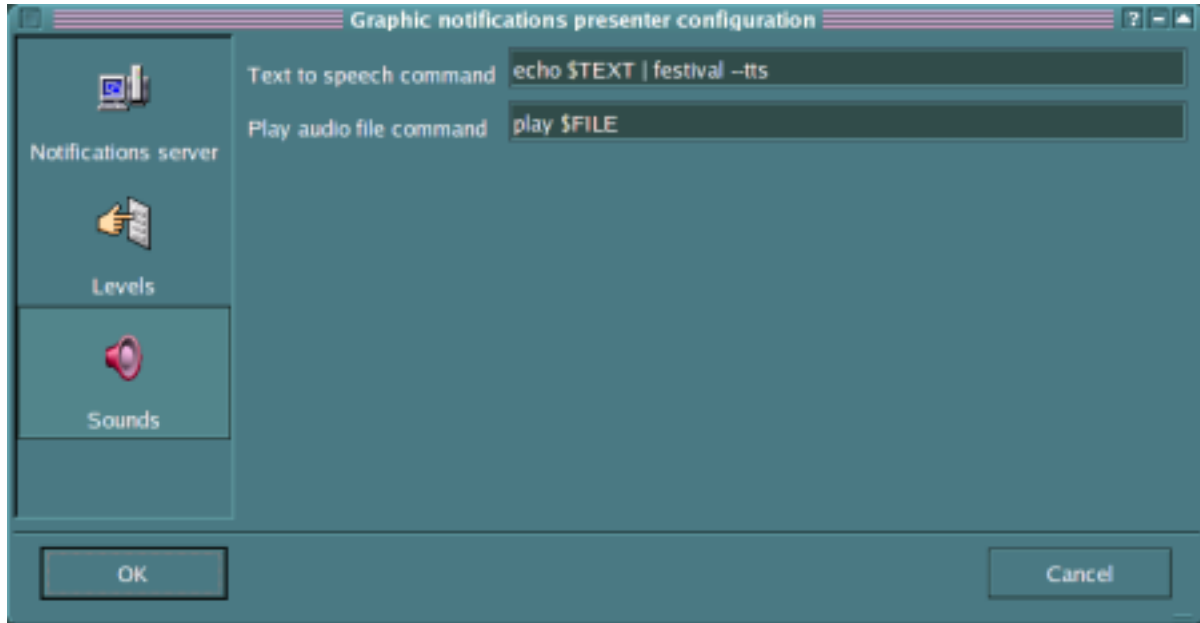


Zakładka ta pozwala ustawić wartości poziomów ważności otrzymywanych powiadomień. Wszystkie poziomy są rozumiane jako wartości bezwzględne to np.: jeśli ważność powiadomienia wynosi x to jeśli

$\text{abs}(x) < \text{warning_level}$

to ważność wiadomości jest uznawana jako informacyjna. Parametry `Warning level` oraz `Critical level` wyznaczają ważność powiadomień. Parametr `Flashing level` mówi od jakiej wartości będzie migał napis określający ważność komunikatu podczas wyświetlania okienka z powiadomieniem, zaś parametr `Beeping level` mówi od jakiej ważności będzie wydawany krótki sygnał dźwiękowy podczas wyświetlania tego okienka. Parametr `Interesting level` wskazuje minimalny poziom ważności, poniżej którego komunikat nie będzie wyświetlany w postaci okienka powiadamiającego. Zawsze jednak zostanie dodany nowy element listy otrzymanych powiadomień w głównym okienku aplikacji. Grupa `Maximum number of notifications at the same time` pozwala ustawić maksymalną liczbę okienek powiadomień w tym samym czasie.

19.4.4.3. Konfiguracja dźwięków w aplikacji xdgnp



Zakładka ta pozwala wyspecyfikować odpowiednie polecenia, które są uruchamiane w celu odtworzenia plików dźwiękowych i wiadomości tekstowych. W polu `Text to speech command` dokonuje się specyfikacja polecenia, które służy do uruchomienia syntezy mowy, zaś w polu `Play audio file command` specyfikuje się polecenie, przy pomocy którego odtwarzane są pliki dźwiękowe.

19.4.4.4. Obsługa okienka graficznego z powiadomieniem



Okienko graficzne prezentujące treść wiadomości o jakiej winien być poinformowany operator posiada na pasku tytułowym datę pojawienia się zgłoszenia klienta. U góry okienka pojawia się napis określający ważność komunikatu (INFORMATION, WARNING, CRITICAL). Jest on zależny od dwóch parametrów: od ważności komunikatu ustawionej przez klienta oraz od zakresów wartości dla poszczególnych ważności ustawianych przez użytkownika aplikacji **xdgnp**. Poniżej znajduje się tytuł

wiadomości, a pod nim tabelka z poszczególnymi zdarzeniami wchodzącymi w skład sprawy. Pod tabelką znajduje się przycisk `Close` pozwalający zamknąć okienko, co jest równoznaczne z potwierdzeniem przyjęcia wiadomości. Na samym dole okienka widoczny jest czas pozostały do jego automatycznego zamknięcia.

Każdy wiersz tabelki prezentuje poszczególne zdarzenie wchodzące w skład sprawy. Poszczególne kolumny opisują:

- `Status` - stan zdarzenia (do wyboru są trzy możliwości: `Active (A)`, `Passive (H)`, `Not managed here (NM)`). Status można zmieniać z aktywnego na pasywny i na odwrót, natomiast stan `Not managed here` nie podlega modyfikacji;
- `Message` - czytelna dla człowieka treść zdarzenia;
- `Creation time` - czas utworzenia zdarzenia (w tej sprawie);
- `Modification time` - czas ostatniego pojawienia się zdarzenia;
- `Hits numbers` - ilość wystąpień (powtórzeń) od czasu utworzenia zdarzenia;
- `Successors` - lista najbardziej prawdopodobnych następców (następnych zdarzeń) tego zdarzenia umieszczonych jeden pod drugim.

19.5. Tematy pokrewne

Notification Processor: Rejestrator Wiadomości (dsi)

[Serwer Powiadomień Graficznych \(dgnsd\)](#)