Upgrading to up.time 4

up.time software

because downtime is not an option

Contacting uptime software

By mail:
uptime software inc.
555 Richmond Street West,
PO Box 110
Toronto, Ontario
M5V 3B1

Main telephone line: 416 868 0152
Main fax line: 416 868 4867

Contacting Sales

To contact sales, use the main telephone line: 416 868 0152, and follow the prompts.

Please have the following information available so we may serve you better:

- Operating systems
- Key applications and databases
- Deployment Timeframe
- Project to deploy
- Key problems
- Present tools

Contacting Support

uptime software delivers responsive customer support. Customer support is available to licensed and demonstration users.

uptime software offers user support through the following:

- Documentation
- Application
- Telephone
- E-mail
- Internet site

Before contacting support, consult the up.time Version 4 Manual/Guide, up.time Version 4 Release Notes, up.time Version 4 README, or the help system from the Help button in the application.

To contact sales, use the main telephone line: 416 868 0152, and select option #2.

TABLE OF CONTENTS

Performing a Test Upgrade

Advanced Options

Other Information

CHAPTER 1

Introduction

There are a number of factors to consider if you plan to upgrade from up.time 3.0 to up.time 4. Your primary concern is undoubtedly how to transfer your performance data to the newer version of up.time.

up.time 4 includes a migration utility that will copy the following from your up.time 3.0 DataStore to the up.time 4 DataStore:

- Information about the systems that you are monitoring.
- System groups.
- System configuration information.
- Information about users.
- Scheduled maintenance periods (called *downtime* in version 3.0).

The performance and historical data that is migrated to the up.time 4 DataStore is not deleted from the up.time 3.0 DataStore.

This document explains the process of migrating your data from up.time 3.0 to up.time 4.

You should read this document in its entirety before proceeding with the upgrade.

Why Upgrade to up.time 4?

up.time 4 is far more robust, flexible, and extensible than up.time 3.0, and offers:

- Better performance
  - Consolidated Global Scan with fast, global monitoring for thousands of servers.
- Better scalability
  - Extended database support for large deployments – MySQL, Oracle 10g, SQL Server 2000.
- Better reporting:
  - Enhanced service monitoring.
  - Improved graphing and reporting.
  - Enhanced SNMP monitoring.
  - Custom plug-in monitors.
- Enhanced user interface:
  - New views called My Portal and My Enterprise.
- Easier administration features that you can customize:
  - Powerful new administration features.
  - Support for custom plug-in monitors.
  - Integrated Help and access to the support portal.
- Flexible deployment options:
  - Agent-based or agentless monitoring.

For more information on the features and benefits of up.time 4, visit the uptime software Web site (http://www.uptimesoftware.com/newinuptime4.php).

What You Need to Know About Upgrading

When upgrading to up.time 4, you should be aware of the following:

- The data migration process can take a considerable amount time.

  The process can take several hours or several days, depending on the amount of data that is in your up.time 3.0 DataStore. To determine how long the migration will take, run a pre-migration check. See “Performing a Pre-Migration Check” on page 15 for more information

- You will need at least as much disk space available for up.time 4 as you have for up.time 3.0.

  If you are installing up.time 4 on the same system as up.time 3.0, the total space required for the upgrade will be two times the amount of your current disk space, until you remove up.time 3.0 from the system.

  You can determine how much disk space you need by running a pre-migration check (see “Performing a Pre-Migration Check” on page 15 for more information). If you find that you are running out of disk space during the data migration process, you can opt to migrate smaller amounts of performance data. See page 33 for more information, or contact uptime software Client Support Services for assistance.

- You can only migrate data from up.time 3.0 to a freshly-installed version of up.time 4.

  If you have installed up.time 4 and it has been collecting data for a period of time, the data migration process will fail. If you want to perform an upgrade, you must completely uninstall your current version of up.time 4 and then install up.time 4 Service Pack 1.0

- You must choose the type of database – MySQL, Oracle 10g, or SQL Server 2000 – into which you want to migrate data before performing the upgrade. If you plan to migrate to Oracle 10g or SQL Server 2000, contact uptime software Client Support Services for assistance.

- You can upgrade to and from any supported platform.

  You can, for example, move from using up.time 3.0 on Windows to up.time 4 on Solaris.

- If you have customizations – such as monitors, pager scripts, or customized alerts – you must modify your scripts so that they will work with up.time 4. See “Changes to Scripts” on page 4, or contact uptime software Client Support Services for assistance.

- The migration process will not delete or corrupt the performance or historical data in your up.time 3.0 DataStore.

  You can experiment with the various data migration options until you find one that works best for the amount of performance data that you need to migrate.

Changes in up.time 4

There are a number of differences between up.time 3.0 and up.time 4. The application's infrastructure has been greatly enhanced to make up.time more robust and scalable. Some up.time 3.0 service monitors have been removed, and up.time 4 includes new monitors or enhanced versions of the monitors that are available in version 3.0.

up.time 4 also contains a number of changes to the user interface and to the way in which the application invokes scripts. As well, some of the supported platforms have cahnged.

Interface Changes

The following changes have been made to the user interface in up.time 4:

- The look and feel of the interface has been changed.
- The link that opens the **Syslist** is now in top-right portion of the user interface.
- Reports are scheduled and not on a separate page, but on the definition page for each report.

Changes to Scripts

The way in which scripts are invoked by the application has changed in up.time 4. See page 40 for details about the modifications to custom scripts and alert scripts.

Platform Changes

up.time 4 no longer supports the following platforms (which were supported by version 3.0):

- Red Hat Enterprise Server 2.1 and older.
- Windows 2000.

For more information about the platforms that are supported by up.time 4, visit the product requirements page (http://www.uptimesoftware.com/requirements.php) of the uptime software Web site.

What is Not Migrated

The upgrade process migrates configuration information and performance to up.time 4. It also copies System Notes to the up.time 4 DataStore, but the System Notes are not used.

However, the following are not migrated:

- Scheduled and posted reports

  The storage and generation options for reports have changed in up.time 4. If you want to continue running your reports at specific intervals, you must manually configure scheduled reports in up.time 4.

- Archived data

  Any data that is not in the DataStore will not be migrated to up.time 4. If you need to keep your archived data, you should import it into the up.time 3.0 DataStore before performing a migration. For more information, see the uptime software support knowledge base article “Importing Archived Data into up.time 3.0.”

- System membership in multiple groups

  This is not supported in up.time 4. When migrated, systems that appear in multiple groups are assigned to a single up.time 4 group. The other groups to which the system belongs will be placed into the system's custom fields to a maximum of four.

- Monitor and host group escalations

  You must configure monitor and host group escalations in up.time 4. You will be able to quickly configure escalations in up.time 4.

- Custom email formats

  up.time 4 does not support custom email formats. You can, however, write a script to send custom emails and then use that script with a alert profile. For more information, see the section “Creating Alert Profiles” in the *up.time 4 User Guide* and the uptime software support knowledge base article “Creating Custom Alert Scripts in up.time Alert Profiles”.

- DHCP and UDP monitors

  up.time 4 does not include DHCP and UDP monitors. If you need alternative ways in which to perform these checks, contact uptime software Client Support Services.

- Novell systems

  Novell agents are not supported in up.time 4. Any Novell agents are upgraded to nodes. However, agentless Novell NRM systems are migrated properly.

- System visibility

  Users who can view and access all of the functions of up.time 3.0 lose those permissions when they are migrated to up.time 4. System administrators must place these users into up.time 4 groups with appropriate permissions after the data migration process is completed.

  When users are migrated to up.time 4, they are assigned the default role which has basic permissions. You must also assign the users appropriate roles. For more information on how user roles work in up.time 4, and how to assign roles, see the section “Working with User Roles” in the *up.time 4 User Guide*.

- Web server and agent SSL configuration

  The SSL settings for the Web server and up.time agents are not migrated from up.time 3.0. If you need up.time to continue securely communicating with the agents installed on the monitored hosts, you must manually enable SSL for systems in up.time 4.

- Mail delivery on Linux and Solaris

  The Linux and Solaris versions of up.time 3.0 use a local mailer, like Sendmail, to email alerts, reports, and the like. up.time 4 uses an SMTP server, and the installer asks you to specify the name of the default mail host.

Upgrade Steps

uptime software recommends that you upgrade from up.time 3.0 with the following steps:

1. Run a pre-migration check to determine how long the migration process will take and the amount of disk space that is required for the migration. For more information, see “Performing a Pre-Migration Check” on page 15.
2. Select on a migration strategy. For information on common migration strategies, see “Migration Strategies” on page 12.
3. Run the migration utility with the `--detail 0` option. This option is quite fast, and only migrates such configuration data as:

   - Systems and system groups
   - Services
   - Users
   - User groups, which are placed into up.time 4 Notification groups
   - Archive policy information
   - Monitoring period settings

   Once the migration is complete, you can check up.time 4 to determine whether or not the configuration data was properly migrated.
4. Run the migration utility to gather the following performance data for each host in one-hour batches:

   - Service status transitions.
   - Timed monitor data.
   - Performance samples.

   The data migration utility also has options that give you finer control over the amount of data that is converted. For more information, see “Advanced Options” on page 33.

Introduction

CHAPTER 2

Before You Upgrade

uptime software recommends that do some planning before you upgrade to up.time 4. This chapter outlines the factors that you will need to consider when upgrading.

Verify Your Version of up.time 3.0

To perform the upgrade to up.time 4, you must be running up.time 3.0 with service pack 8.0. To determine which service pack you have installed, view the following file:

On Windows, `C:\Program Files\uptime software\up.time 3.0\conf\release.dat`

On Linux or Solaris, `/opt/SPYNuptm/conf/release`

If you need to install an up.time 3.0 service pack, you can download it from the up.time Client Care Portal.

Decisions to Make

When planning your upgrade, you should answer the following questions:

- Are you installing up.time 4 on the same system as up.time 3.0?

  If you are installing up.time 4 on the same system as up.time 3.0, the upgrade process is relatively straightforward. However, if you are installing up.time 4 on a different system, see page 39 for additional information upgrading. Or, contact uptime software Client Support Services.

  As well, you must choose a port other than `9999` for the up.time 4 Web server so that it does not conflict with the default Web server port used by up.time 3.0. Do not use any of the values that are listed in the knowledge base article “Required ports for the up.time monitoring station” (http://support.uptimesoftware.com/article.php?id=021).

- To which database will you be migrating your data?

  up.time 4 supports MySQL, Oracle 10g, and SQL Server 2000. If you are moving performance data to MySQL (the default database for up.time 3.0 and 4), the upgrade process will be relatively straightforward.

  If, however, you plan to migrate performance data to Oracle 10g or SQL Server 2000, you must perform additional setup tasks before performing the upgrade. For more information, see the following uptime software support knowledge base articles (http://support.uptimesoftware.com):

  - "Running up.time on Oracle 10g"
  - "Running up.time on Microsoft SQL Server 2000"

  Contact uptime software Client Support Services before migrating performance data from up.time 3.0 to Oracle 10g or SQL Server 2000.

- How much performance data do you need to migrate?

  The migration process will take longer if you have more performance data to migrate. If you do not need to migrate all of your data, you can opt to covert only one performance or process data sample per hour.

  By default, the up.time service monitors collect performance data every five minutes; approximately 12 samples of data from monitored hosts are collected every hour. You can specify that the migration utility takes the data sample from the top of the hour for each host. By doing this, your data only takes up 1/12 of the space in the DataStore, while retaining the historical integrity of your performance data. For more information on the options that enable you to do this, see page 33.

- How will you manage the transition from up.time 3.0?

  You need to decide whether or not to immediately take up.time 3.0 offline, migrate all of your data, and then put up.time 4 into production. If this is not acceptable, you can opt to gradually move to up.time 4 while running it in parallel with up.time 3.0.

  See "Migration Strategies" on page 12 for more information on ways in which you can transition to using up.time 4.

- How large is your up.time 3.0 DataStore?

  The larger your DataStore, the longer it will take to migrate your data to up.time 4. You can calculate the size of the DataStore using the equation in

the section of the *up.time 3.0 User and Installation Guide* titled “Size of DataStore.”

Contact uptime software Client Support Services for assistance if you have over 100 GB of data, or if your data consumes more than half of your disk space.

- Have your offline archives been moved to location other than the default?

  If you want to migrate archived data, you must first import that data into the DataStore. However, MySQL `.MRG` files – which are used to merge monthly performance tables – do not support full path names; you will not be able to refer to those files if they are anywhere other than the following default locations:

  On Solaris: `/opt/SPYNuptm/DataStore/uptime/archive`

  On Windows: `C:\Program Files\uptime software\up.time 3.0\DataStore\uptime\archive`

- Have your up.time 3.0 scripts been customized?

  Certain script customizations are not carried over to up.time 4. You may need to recreate or modify your scripts. See “Changes to Scripts” on page 4 for more information.

- What is the layout of the file system on the server that is hosting up.time 3.0?

  If you are migrating from up.time 3.0 to up.time 4 on the same system, then you may quickly run out of disk space. As well, up.time 4 is installed in a different directory location than up.time 3.0.

- What is your archive policy?

  If you are archiving your data over short periods and are installing up.time 4 on the system on which up.time 3.0 is running, you may find that the archives will take up disk space. You may need to change your archive policy, or install up.time 4 on another server.

- Do you have any custom software enhancements?

  Custom software enhancements that are designed for up.time 3.0 are not migrated to up.time 4. Contact your account representative for more information.

Working with Support

While uptime software has tried to make the upgrade process as easy as possible, there will be cases in which users will need the assistance of Client Support Services during an upgrade. Contact support if any of the following apply to your installation of up.time 3.0:

- You have replicated the DataStore in order to enhance the performance of up.time.
- Your DataStore contains more than 100 GB of data, or if your data consumes more than half of your disk space.
- You have a large number of offline archives.

Migration Strategies

This section outlines the following three data migration strategies:

- Testing different migration options
- Running up.time 3.0 and up.time 4 in parallel
- Run up.time 3.0 until up.time 4 is ready to put into production

Testing different migration options

Shut down up.time 3.0, install up.time 4 and then immediately migrate your data from up.time 3.0. This strategy enables you to experiment with various data migration options.

You can, for example, migrate specific data – such as configuration information. Once you are confident that the migration process is properly moving your data to the up.time 4 DataStore, you can perform a full migration. See page 33 for more information on migration options.

At any time, you can use the `resetdb` utility to delete all of the data from your up.time 4 DataStore. The `resetdb` utility deletes and then recreates the database structured that is used by up.time 4, but with no data. To use the `resetdb` utility, do one of the following:

- On Windows, `C:\Program Files\uptime software\uptime4\resetdb really`

- On Linux or Solaris, `/usr/local/uptime4/resetdb really`

Running this command will delete all of the data from the DataStore.

The `really` option helps to ensure that you do not inadvertently delete the contents of your DataStore.

Once the database has been reset, you can perform the migration process again.

Running up.time 3.0 and up.time 4 in parallel

Shut down up.time 3.0 and then migrate your data. When the migration is complete, run up.time versions 3.0 and 4 in parallel. By doing this, you can compare reports and graphs, as well as the overall performance of both versions of up.time.

If you use this migration strategy, make sure that the Web servers for up.time 3.0 and up.time 4 are running on different ports.

Run up.time 3.0 until up.time 4 is ready to put into production

You install up.time 4 but have up.time 3.0 running during the migration process. After migrating your data, you take up.time 3.0 offline and switch to using up.time 4. However, doing this will result in a gap in your data for the period during which the migration took place as the migration process will not convert data for that period.

To eliminate any gaps in your performance data, run the migration utility again using the `--catchup` option. The `--catchup` option migrates the data that was gathered after the original snapshot migration and copies it to the up.time 4 DataStore.

See page 36 for more information about the `--catchup` option.

CHAPTER 3

Performing a Pre-Migration Check

uptime software strongly recommends that you run a pre-migration check before migrating data from up.time 3.0. The pre-migration check determines the following:

- How long the data migration process will take.

  The pre-migration check displays estimates for the amount of time that is required to convert:

  - The full database.
  - The following process and disk performance records:
    - Single or multiple CPU statistics.
    - Network statistics.
    - File system information.
    - User and login information.
    - Logical volume manager information for systems using Veritas Volume Manager.
  - One data sample per hour. See page 33 for more information.
  - No performance data.

  These estimates are based on a system that is running Windows 2003 Server Edition with a 2 GHz Pentium 4 processor, 512 MB of memory, and local SCSI storage.

- The amount of disk space that is required for the migrated data.

  The pre-migration check will display the disk space requirements for each of the data migration options that are listed above.

The results of the check are printed to screen, and are written to the following file:

- On Windows: `C:\Program Files\uptime software\uptime4\tmp\preflight.log`.
- On Linux or Solaris: `/usr/local/uptime4/tmp/preflight.log`.

Using the results of the pre-migration check, you can determine how much data to migrate as well as the proper migration strategy to use.

Running the Pre-Migration Check

To run a pre-migration check, do the following on the system that is running up.time 4:

1. **At the command line, navigate to the `scripts` folder.**

   On Windows, the default location for the folder is `C:\Program Files\uptime software\uptime4\scripts`.

   On Linux or Solaris, the default location for the folder is `/usr/local/uptime4/scripts`.

2. **Do one of the following:**

   - On Windows, run the command `upgradev3 <--host [hostname]> --check`.
   - On Linux or Solaris, run the command `./upgradev3 --check`.

   If up.time 3.0 is installed on a different system, you can add the `--host <hostname>` option to the command, where `<hostname>` is the name of the system on which the up.time 3.0 DataStore is running.

   If you do not specify a host, the default is `localhost`.

   Output similar to the following is displayed on screen and is written to the file `preflight.log` (found in the `tmp` folder where up.time 4 is installed):

   ```
   HOW LONG WILL THIS TAKE?

   Your database has a total of 225,824,061 rows. The time
   to convert it will depend on your hardware and network
   configurations, but expect it to take about 31.36 hours.
   for a full conversion.
   ```

You can also give options to reduce the amount of detailed data that is converted, as follows:

--detail 3: full conversion (225,824,061 rows): 31.36 hours.

--detail 2: selectively convert process and detailed disk performance records (98,890,616 rows): 13.73 hours.

--detail 1: only convert one data sample per hour (18,818,668 rows): 2.61 hours.

--detail 0: do not convert performance data: about 1 minute.

HOW MUCH DISK SPACE WILL I NEED?

The amount of space depends on the detail level you choose. See above for the descriptions of the options.

--detail 3: about 21.88GB.

--detail 2: about 9.4GB.

--detail 1: about 1.84GB.

--detail 0: about 20MB.

WHAT ELSE DO I NEED TO KNOW?

The following users will need to be placed into up.time 4 user groups with appropriate visibility after the conversion.

* tslothrop
* rfripp
* bbruford
* tcrabin

If you have any questions about the information above, or about the upgrade process in general, please do not hesitate to call uptime support at +1-416-868-0152.

CHAPTER 4

Migrating Your Data

Once you have planned the upgrade (see page 9) and performed a pre-migration check (see page 15), you can run the upgrade utility. As discussed in the section “Upgrade Steps” on page 7, you should first migrate configuration information from up.time 3.0 and then migrate your performance data

The data that is migrated to the up.time 4 DataStore is not deleted from the up.time 3.0 DataStore.

This chapter discusses an upgrade in general terms. For an example of an upgrade, see page 23. For information on the recommended steps for the upgrade, see page 7.

You can also perform an upgrade when up.time 3.0 and 4 are installed on different systems on your network. For more information, see page 40.

If you want to carry out a test upgrade, follow the steps on page 31.

Migrating Data

To migrate data from up.time 3.0, do the following:

1. **If you have not already done so, install up.time 3.0 Service Pack 7.2.**

   You can download the service pack from the up.time Client Care Portal.

2. **Install up.time 4 and the license key for the application on the up.time 4 system.**

   See the *up.time 4 User Guide* for more information.

If you plan to use Oracle 10g or SQL Server 2000 as your DataStore, read the following uptime software support knowledge base articles (http://support.uptimesoftware.com):

- "Running up.time on Oracle 10g"
- "Running up.time on Microsoft SQL Server 2000"

Contact uptime software Client Support Services ***before*** migrating your data to Oracle 10g or SQL Server 2000.

3 **At the command line, navigate to the `scripts` folder under the root installation directory.**

On Windows, the default location for the folder is `C:\Program Files\uptime software\uptime4\scripts`.

On Linux or Solaris, the default location for the folder is `/usr/local/uptime4/scripts`.

4 **Do one of the following:**

- On Windows, run the command `upgradev3 --detail [n] <--host [hostname]>`.
- On Linux or Solaris, run the command `./upgradev3 --detail [n] <--host [hostname]>`.

Where:

- `--detail [n]` specifies the amount of data that will be converted. `[n]` is one of the following:
  - `3` – All of the data in the DataStore is migrated.
  - `2` – Only one process and one detailed disk performance record per hour is migrated.
  - `1` – Only one data sample per hour is migrated.
  - `0` – Only configuration information is migrated (see page 7 for details). No performance data is migrated.

For more information on samples per hour, see page 33.

- `--host <hostname>` is the name of the system on which the up.time 3.0 DataStore is found. Use this option if up.time 3.0 is running on another system. Otherwise, you do not need specify this option; the default is `localhost`.

The data migration utility also has several other options that give you finer control over the amount of data that is migrated. See “Advanced Options” on page 33 or contact Client Support Services.

5 **A message outlining the pre-upgrade steps appears.**

Press Enter to continue, or type `quit` to stop the upgrade.

When the migration process is complete, the following messages appear:

```
Upgrade finished
Database is up to date
```

APPENDIX A

A Sample Upgrade

This section outlines upgrade from an up.time 3.0 installation called `uptime30` to a new installation of up.time 4, which is installed on a separate host and which is using MySQL as the DataStore.

The upgrade outlined in this section is involves the following steps:

- Install up.time 4
- Run a Pre-Migration Check
- Migrate your up.time 3.0 Configuration Settings
- Confirm your Migration in up.time 4
- Migrate Samples of your Performance Data

Install up.time 4

To install up.time 4, download the installation package for your operating system from the uptime software Web site or use the installation CD. For details on how to install up.time 4, see the *up.time 4 User Guide*.

Note that while the Linux and Solaris versions of up.time 3.0 use Sendmail for emailing alerts, reports, and the like, up.time 4 uses an SMTP server. The installation program asks you to specify a name for the default mail host. This should be your corporate mail server, for example `mail.mycompany.com`.

Installing a License

Once up.time 4 is installed, you must install a valid license for up.time 4. To obtain a license key, visit the uptime software support upgrade portal (http://support.uptimesoftware.com/upgrade.php).

To install a license, do the following:

1. **In the up.time 4 Tree panel, click License Info.**

2 **Paste the new or updated license into the License Key text box.**

3 **Click Update.**

Run a Pre-Migration Check

The pre-migration check estimates the amount of time and disk space required to migrate all of your data, or just a portion of it. To run a pre-migration check, do the following:

1 **At the command line, navigate to the scripts folder.**

On Windows, the default location for the folder is `C:\Program Files\uptime software\uptime4\scripts`.

On Linux or Solaris, the default location for the folder is `/usr/local/uptime4/scripts`.

2 **Run the following command:**

- On Windows, run `upgradev3 --host uptime30 --check`.
- On Linux or Solaris, run `./upgradev3 --host uptime30 --check`.

The pre-migration check returns the following output:

```
HOW LONG WILL THIS TAKE?

Your database has a total of 40,081,579 rows. The time to
convert it will depend on your hardware and network
configurations, but expect it to take about 5.57 hours for a
full conversion.

You can also give options to reduce the amount of detailed
data that is converted, as follows:

--detail 3: full conversion (40,081,579 rows): 5.57 hours.

--detail 2: only converts one process and detailed disk
performance record per hour (18,219,711 rows): 2.53 hours.

--detail 1: only converts one data sample per hour
(3,340,127 rows): 0.46 hours.

--detail 0: does not convert performance data: about 1
minute.

HOW MUCH DISK SPACE WILL I NEED?

The amount of space depends on the detail level you choose.
See above for the descriptions of the options.
```

```
--detail 3: about 3.66GB.
--detail 2: about 1.3GB.
--detail 1: about 330.7MB.
--detail 0: about 20MB.
```

As well, the pre-migration check warns that:

- Several System Notes will be copied to the up.time 4 DataStore but not used.
- A number of systems that are in multiple system groups will be assigned to a single group in up.time 4.
- The DHCP monitor and several alert profiles will not be migrated to up.time 4.
- Scripts for several custom monitors must be changed, since the way in which up.time invokes scripts has changed.

For more information on the pre-migration check and its output, see “Performing a Pre-Migration Check” on page 15.

Migrate your up.time 3.0 Configuration Settings

Next, migrate the configuration setting from up.time 3.0. To do this, run one of the following commands on the server that is hosting up.time 4:

- On Windows, run the command `upgradev3 --host uptime30 --detail 0`.
- On Linux or Solaris, run the command `./upgradev3 --host uptime30 --detail 0`.

The option `--detail 0` only migrates the configuration information from up.time 3.0. No performance data is converted.

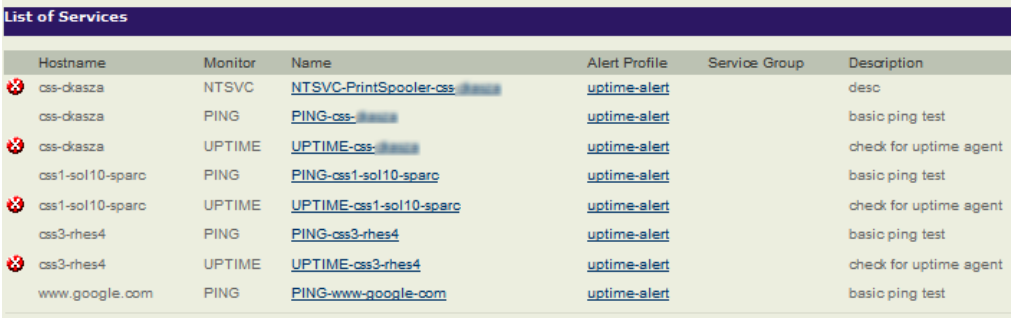
Confirm your Migration in up.time 4

You can confirm that the configuration data was migrated properly by comparing the following in both versions of up.time:

- The contents of **Global Scan**.
- The list of service instances.

- The list of users.
- Notification Groups (in up.time 4) and User Groups (in up.time 3).
- The **Archive Policy** and **Monitoring Period** settings in the **Config** panel.

The following image illustrates the service instances on the up.time 3.0 system:



The following image illustrates the service instances on the up.time 4 system after configuration data was migrated:

The lists contain the same services, indicating that the migration was successful.

Migrate Samples of your Performance Data

Once you are satisfied that the configuration data has been properly migrated, you can begin migrating performance data. You should start off by migrating one data sample per hour. For more information on data samples, see page 33.

Do one of the following:

- On Windows, run the command `upgradev3 --host uptime30 --detail 1`.
- On Linux or Solaris, run the command `./upgradev3 --host uptime30 --detail 1`.

If you use any other `--detail` option after migrating configuration data using `--detail 0`, the migration utility re-migrates the configuration data and then migrates performance data.

Before running the migration utility at a detail level different from `--detail 0`, you must first run the following command:

```
cd <uptime_dir>
./resetdb really --nodata
```

Where is `<uptime_dir>` the directory in which up.time 4 is installed. The resetdb command erases all of the up.time 4 data in the DataStore, leaving you with an empty database. It will not, however, delete the data in the up.time 3.0 DataStore.

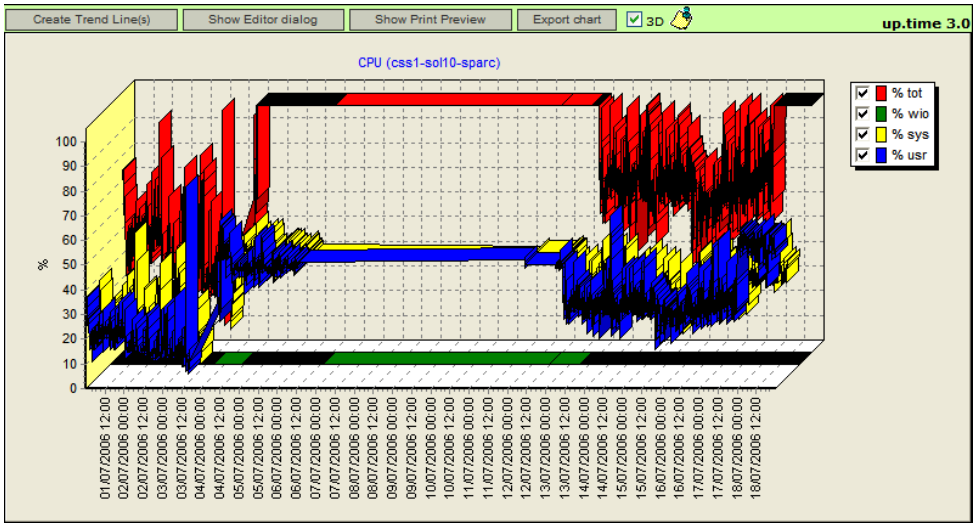
Alternatively, you can use one of the data migration utility's advanced options (see page 33 for more information) to convert an even more granular data sample. For example, running the following command:

```
./upgradev3 --host uptime30
--oneFsStatsSamplePerHour --earliestDate 2006-03-21
```

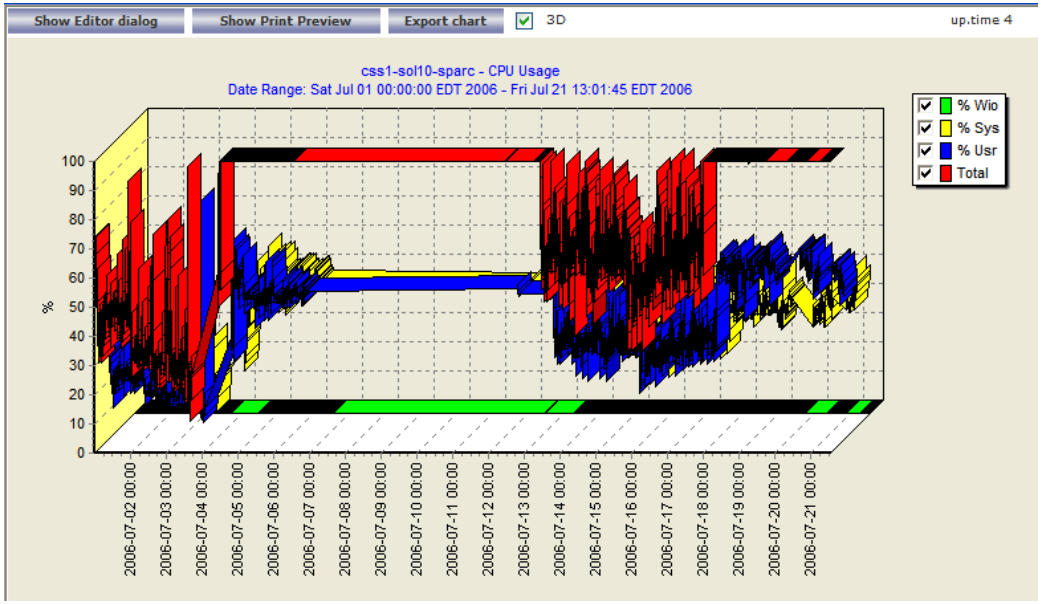
will only migrate one file system capacity sample per hour for each host configured in up.time 3.0, starting from March 21, 2006 to the current date.

A good way in which to determine whether or not the performance data was properly migrated is to generate and compare graphs using both versions of

up.time. The following image illustrates a CPU Usage graph that was generated on the up.time 3.0 system:



The following image illustrates the same graph generated on the up.time 4 system:



Once you are confident that data has been properly converted, you can continue migrating more data. For example, using the following command:

```
./upgradev3 --host uptime30 --onePSInfoSamplePerHour
```

will only migrate one process information data sample per hour for each host.

Or, you can run the upgrade utility with the `--detail 3` option to convert all of the data in the DataStore to the up.time 4 format. However, doing this will take a considerable amount of time and disk space. In this example, a full conversion will take approximately 5.57 hours, and will require 3.66 GB of disk space.

APPENDIX A

Performing a Test Upgrade

This section outlines how to perform a test upgrade using a copy of your up.time 3.0 DataStore. Performing a test upgrade enables you to run the migration from a different server than the one on which your production version of up.time 3.0 is running. This gives you the opportunity to test various data migration options while not impacting upon the performance of your production database.

Steps Involved in the Test Upgrade

To perform a test upgrade, carry out the following steps:

1. **Install up.time 3.0 on your testing system.**

   The version of up.time 3.0 as well as the operating system on the testing system must be the same as on the production system.

2. **Stop all up.time processes on the testing and production servers.**

3. **Do one of the following:**

   - On Windows, copy the directory `UptimeDataStore/uptime` from the up.time 3.0 production system to the folder `C:\Program Files\uptime software\up.time 3.0\` on the testing system:
   - On Solaris, copy the directory `DataStore/uptime` from the up.time 3.0 production system to the directory `/opt/SPYNuptm/` on the testing system.

4. **Restart all up.time processes on the production system.**

   The production system can resume normal operations.

5. **Restart only the database on the testing system.**

   You will not need to install a license on the testing system as only the database is required for the up.time 3.0 to up.time 4 data migration

6 **Run the test upgrade on your new up.time 4 installation.**

up.time 4 can be installed on the same host as the up.time 3.0 test version. At the command line, specify the hostname and database of the test system. See “Migrating Your Data” on page 19 for more information.

APPENDIX B

Advanced Options

If you need more control over the amount of data that you want to migrate, you can add any of the options that are listed in this appendix to the migration command. You can use these options if you:

- Only need to migrate a small portion of your performance data.
- Want to test the migration process by converting, for example, a portion of your performance data. From there, you generate graphs or reports in both up.time 3.0 and up.time 4 and compare the results to determine whether or not the data was properly migrated.
- You do not have enough disk space to migrate all of your up.time 3.0 data to up.time 4. By migrating only a subset of the data in the DataStore, you can ensure that you will have enough disk space while at the same time migrating your important historical performance data.

Many of the options listed in this appendix take one data sample per hour. By default, up.time service monitors collect data every five minutes; approximately 12 samples of data from monitored host are collected every hour. You can specify that the migration utility takes the data sample from the top of the hour for each host – essentially collecting only one data sample each hour. By doing this, your data only takes up 1/12 of the space in the DataStore, while retaining the historical integrity of your performance data.

Options Reference

You can add the following options to the data migration command:

Do not use the `--detail [n]` option with any of the options that are listed below. For more information on the `--detail` option see page 20.

- `--host <host_name>`

  The name of the remote system on which up.time 3.0 is running. Do not use this option if you installed up.time 4 on the same system as up.time 3.0.

- `--dbname <database_name>`

  The name of the up.time 3.0 DataStore. Use this option if up.time 3.0 is running on another system. Otherwise, you do not need specify this option; the default is `uptime`.

- `--oneSamplePerHour`

  Converts only one sample of data sample for each one hour period.

- `--earliestDate <yyyy-mm-dd>`

  Converts only data samples from the specified date onwards.

- `--onePsInfoSamplePerHour`

  Converts only one sample of process information for each one hour period.

- `--oneCpuStatsSamplePerHour`

  Converts only one sample of CPU statistics information for each one hour period.

- `--oneMpuStatsSamplePerHour`

  Converts only one sample of multiple CPU statistics information for each one hour period.

- `--oneFsStatsSamplePerHour`

  Converts only one sample of file system statistics information for each one hour period.

- `--oneNetStatsSamplePerHour`

  Converts only one sample of network statistics information for each one hour period.

- `--oneDiskStatsSamplePerHour`

  Converts only one sample of disk statistics information for each one hour period.

- `--oneWhoInfoSamplePerHour`

  Converts only one sample of user information (user name and number of logins) for each one hour period.

- `--oneLvmStatsSamplePerHour`

  Converts only one sample of logical volume manager (LVM) information for each one hour period. This option is only valid for systems using the Veritas Volume Manger.

- `--verifyTableIntegrity`

  Runs the MySQL `check table` command on the up.time 3.0 DataStore check for errors in the database. The data migration utility exits if the `check table` command encounters errors.

- `--v3ConfFile`

  The full path to up.time 3.0 configuration file. This is used to migrate up.time 3.0 Global Scan settings and the default agent port to up.time 4.

  If you have installed up.time 4 on the same server as up.time 3.0, you specify the path to the file – for example, `/opt/SPYNuptm/conf/uptime.conf`.

  If you have installed up.time 4 on a different server, you must copy the `uptime.conf` file from the system that is hosting up.time 3.0 to the server on which up.time 4 is running. Then, specify the path to the file.

- `--retryFailedBatches`

  If an attempt to migrate performance data failed, this option enables you to restart the migration from the point at which the previous attempt failed. The data migration utility reads this information from the following log file:

  On Windows, `C:\Program Files\uptime software\uptime4\tmp\failedBatches.log`

  On Linux or Solaris, `/usr/local/uptime4/tmp/failedBatches.log`

- `--catchup`

  If you are migrating data to up.time 4 while up.time 3.0 is still running, then there will be a gap in your data for the period during which the

migration took place. The `--catchup` option, which is run *after* the main migration, takes the data that was collected while the migration was taking place from the up.time 3.0 DataStore and copies it to the up.time 4 DataStore.

Examples

The examples in this section illustrate how to use some of the options that are listed above.

Migrating Process Information

In this example, one sample of process information per hour is being migrated from the system `uptime30` to up.time 4 running on Windows. Only data from May 5, 2005 onward is being migrated.

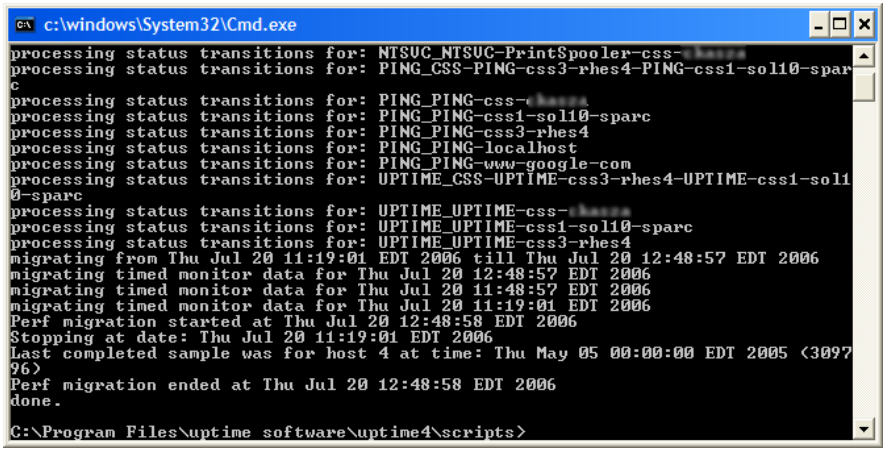
```
C:\Program Files\uptime software\uptime4\scripts\upgradev3
--host uptime30 --onePsInfoSamplePerHour --earliestDate
2005-05-05
```

Using the --catchup Option

You would use the `--catchup` option if you ran up.time 3.0 while migrating data to up.time 4. The `--catchup` option fills in the gap in your data for the period during which the migration took place.

```
C:\Program Files\uptime software\uptime4\scripts\upgradev3
--host uptime30 --catchup
```

The migration utility processed the service status transitions for each service in up.time 3.0, and then migrates monitor data, as shown below:



What to Do If the Migration Fails

If the data migration process fails at any point, you should run the migration utility with the `--retryFailed Batches` option (see page 35). This option restarts the migration at the point at which the previous attempt failed.

If data migration fails again, contact uptime software Client Support Services for assistance. As well, email the following log files:

- `failedBatches.log`

  On Windows systems this file is found in the folder `C:\Program Files\uptime software\uptime4\tmp\`

  On Linux or Solaris systems this file is found in the folder `/usr/local/uptime4/tmp/`

- `uptime.log`

  On Windows systems this file is found in the folder `C:\Program Files\uptime software\uptime4\logs\`

  On Linux or Solaris systems this file is found in the folder `/usr/local/uptime4/logs/`

APPENDIX C

Other Information

This appendix contains information that you may find useful both during and after the upgrade process.

Upgrading When up.time 3.0 and 4 are Installed on Different Systems

You can perform an upgrade when up.time 3.0 and 4 are installed on different systems. If you are upgrading under these circumstances, note the following:

- You must specify the `--host <host_name>` option when running the migration utility. For example:

  ```
  upgradev3 --host uptime30 --detail 2
  ```

  See “Migrating Data” on page 19 for more information on running the migration utility.

- You can migrate the Global Scan and agent port settings from up.time 3.0. To do this, you must first copy the file `uptime.conf` from the system that is running up.time 3.0 to the system that is running up.time 4. This file is located in the following folder:

  On Windows: `C:\Program Files\uptime software\up.time 3.0\conf\`

  On Linux or Solaris: `/opt/SPYNuptm/conf/`

  Then, use the `--v3ConfFile` option with the upgrade utility. For example, if you copied `uptime.conf` to the `/tmp` folder on a Solaris system use the following command:

  ```
  ./upgradev3 --detail 2 --v3ConfFile /tmp/uptime.conf
  ```

  For more information on the `--v3ConfFile` option, see page 35.

Modifications to Custom Scripts and Alert Scripts

The way in which scripts are invoked by the application has changed in up.time 4. This section outlines the changes required for custom monitor scripts and alert scripts.

Custom Monitor Scripts

It is assumed that you will take up.time 3.0 out of service when you have successfully completed the migration. In that case, you will need to modify your up.time 3.0 custom monitor scripts so that they will work with up.time 4.0. Many custom scripts use the `perfget` utility, which is not included in up.time 4, to communicate with an agent. You will need to modify your scripts to use the `netcat` utility (which reads and writes data across a network connection), which comes with up.time 4.

The syntax for `netcat` is:

```
echo “rexec PASS PATH ARGS” | netcat <agent_system_name>
<agent_system_port>
```

In this case, `rexec` is an agent keyword and not the `rexec` system utility.

For detailed information on the arguments listed above, see the uptime software support knowledge base article “Creating Custom Service Monitors in up.time 4”.

The syntax for the `netcat` command varies depending on the operating systems that are communicating, as shown below:

- UNIX to Windows: `echo ‘rexec PASS “PATH ARGS”’`
- UNIX to UNIX: `echo ‘rexec PASS PATH ARGS’`
- Windows to Windows: `echo rexec PASS “PATH ARGS”`
- Windows to UNIX: `echo rexec PASS PATH ARGS`

Alert Scripts

You will need to change your alert scripts. Alert scripts in up.time 4 use environment variables rather than command line parameters to collect information about a service outage. For more information on alert scripts in up.time 4, see the support knowledge base article “Creating Custom Alert Scripts in up.time Alert Profiles”.

Upgrade Process Quick Reference

What You Need to Know Before Upgrading

- The migration process will not alter your up.time 3.0 installation in any way. Your up.time 3.0 installation can remain in production until up.time 4 is up and running.
- The data migration process can take a considerable amount time.
- You can upgrade to and from any supported platform.
- You will need at least as much disk space available for up.time 4 as you have for up.time 3.0. For example, if your up.time 3.0 installation consumes 30 GB of hard disk and want to install up.time 4 on the same system you will need at least 60GB of space for both up.time 3.0 and 4.
- You can only migrate data from up.time 3.0 to a freshly-installed version of up.time 4.
- You must choose the type of database (MySQL, Oracle, MSSQL) into which you want to migrate data *before* doing a migration.
- If you have custom alerts or service monitors you must modify your scripts so that they will work with up.time 4.

What Won't be Migrated

- Scheduled and posted reports
- Archived data
- System membership in multiple groups
- Monitor and host group escalations
- Custom email formats
- DHCP and UDP monitors
- Novell agents (agentless Novell NRM systems are migrated)
- System visibility permissions
- Web server and agent SSL configuration
- Mail delivery settings on Linux and Solaris

Upgrade Steps

	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6	Step 7
up.time 3.0 system	Perform a database check (optional)				Stop up.time 3.0	Restart up.time 3.0	Take up.time 3.0 offline
up.time 4 system (can be the same system as up.time 3.0)	Request a license key for up.time 4	▪ Install up.time 4 ▪ Install the license key	Perform a pre-migration check. Consult with support on items that will not be converted or which may require updates.	Perform a test migration: ▪ Migrate configuration data ▪ Migrate samples of performance data	Perform either a full migration, or migrate only the performance data that you need	Compare configuration settings and performance data with that in up.time 3.0	Put up.time 4 into production

For assistance, contact uptime software Client Support Services at (416) 868-0152 (select option 2) or email support@uptimesoftware.com.