

Data Extractor User Manual



PRTG Data Extractor

© 2015 Paessler AG

All rights reserved. No parts of this work may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems—without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: Januar 2015 in Nuremberg

Table of Contents

Part 1 Introduction	5
Part 2 Download and Getting Started	8
Part 3 Requirements and Setup	10
Part 4 Running the Data Extractor	14
Part 5 Database Setup	19
1 Database Schema Description	21
Part 6 Monitoring the Data Extraction	30
Part 7 Demo Reporting	33
Index	0

Part 1

Introduction

1 Introduction

Welcome to the **PRTG Data Extractor**, a free network tool for PRTG Network Monitor! This tool enables you to retrieve monitoring data and raw data from your PRTG installation. The Data Extractor writes this data into an Microsoft SQL server database. So you as PRTG user have the possibility to process PRTG data further: You can generate your own monitoring data reports exactly adapted for your intended usage.

The present document describes the underlying concepts of the PRTG Data Extractor and shows how to use it. **Note:** In order to use the Data Extractor, you need to have at least basic knowledge about and experience with databases and reporting systems. The present document does not provide instructions about configuring database and report servers. We do not provide any support regarding these basic topics either.

Why Data Extractor?

In the past, PRTG users asked for additional reports and more functions and features for the data reporting system in PRTG. These wishes and suggestions rather vary and so make it a bit inconvenient for a standard solution in PRTG, so our possibilities at Paessler were limited to fulfill all specific user requests. We developed the Data Extractor to meet these manifold user requirements. If you have higher demands on your reporting system than the available standard reports in PRTG, then try out the Data Extractor. The PRTG Data Extractor is free for PRTG users and does not require any licensing costs.

How Does It Work?

PRTG uses its own integrated high performance database to guarantee optimized high speed storage of monitoring data. This data cannot be accessed directly. It is only possible to read the data via PRTG's **Application Programming Interface (API)**. Unfortunately, this approach is neither high-performance nor straightforward. Because of this, we designed the Data Extractor. It can access the data using the PRTG API and stores the data into a relational database. With a corresponding reporting server of your choice you can then create reports suitable to your needs. For best practice, we recommend you to run the data extraction once a day to retrieve new data. You can schedule the run and scale it by using multiple systems.

Please refer to section [Requirements and Setup](#) to see what you need for the Data Extractor and the subsequent sections about how to use it.

Main Advantages in Comparison to the Standard PRTG Reporting System

If you use the PRTG Data Extractor, you will gain the following precious functionalities in addition to the reporting system as it is included in your PRTG installation:

- You are able to customize monitoring data reports in a completely individual way.
- You can use any reporting system you like, already know, or have the developers for.
- A system that has the only purpose "reporting" usually provides more options than every report feature we could integrate into PRTG.

Limitations

There are only minor disadvantages when using the PRTG Data Extractor. These are negligible though if you need richer reports than PRTG itself provides.

- Creating a report will take longer than accessing the same data via the PRTG web interface.
- The latest monitoring data cannot be accessed. Only data that was already retrieved by the Data Extractor can be accessed, of course.
- Due to limitations of the PRTG API, it is not possible to extract sensor data with a scanning interval of 12 hours or more.

Part 2

Download and Getting Started

2 Download and Getting Started

Getting the Data Extractor started is straightforward:

- Download the ZIP file DataExtractor on <http://www.paessler.com/tools/dataextractor>.
- Extract all files into one folder of your choice on a system from where you can reach your PRTG core server and your database server.

No further installation steps are required. You can launch the Data Extractor by opening **DataExtractor.exe**

Part 3

Requirements and Setup

3 Requirements and Setup

In order to extract monitoring data from PRTG and generate corresponding reports, you must have the following systems installed and running:

- [PRTG Core Server](#): Version 14.x.11 or later; optionally an additional PRTG installation
- [Microsoft SQL Server](#): SQL Server 2012 (recommended)
- [Reporting Server](#)

Depending on the size of your installation and the expected traffic of the reports, you can either run everything on the same system or use separate servers. We cannot provide general recommendations regarding this because it is specific to your setup. Please see the sections below for detailed requirements.

PRTG Core Server

You need a running PRTG installation for the Data Extractor (it is explicitly only designed for PRTG Network Monitor and has no other use!). The Data Extractor requires a **PRTG core server version 14.x.11 or later** for full functionality.

Additional PRTG Installation

It is not mandatory to run an additional PRTG installation. However, we strongly recommend to set up a second PRTG installation for the Data Extractor to avoid any influences on your live system. You can use the **PRTG Freeware edition** for this purpose so this approach will not cost you any money for licensing. This additional PRTG installation has only the job to provide the monitoring data via its API. After [downloading](#) and installing the PRTG Freeware edition, deactivate its **PRTG Probe Service** because it will not be needed for the purpose of data extraction.

Important: Use always the same version of PRTG as on your production system! You can find the PRTG software version number under **Setup | PRTG Status | System Status** in the web interface.

Before running the actual data extraction from your additional PRTG installation, you have to accomplish the following steps (please see the [PRTG Manual](#) for details):

1. Stop the PRTG Core Service.
2. Replace the whole local PRTG data directory with a copy of the data directory of your PRTG live system.
3. Restart the PRTG Core Service.
4. Perform API calls to pause all notifications so you will not receive any false alerts.

Note: We strongly recommend you to define scheduled tasks on this system which perform these steps every night before the data extraction.

Microsoft SQL Server

The Data Extractor requires a **Microsoft SQL Server** installation. We recommend using SQL Server 2012.

Database User Permissions for the Data Extractor

There are two ways possible to set up your database for the Data Extractor. The Data Extractor can either create its own database or it can use an already existing database.

If the Data Extractor creates its **own new database**, make sure that the corresponding database user has at least the following permissions:

- **dbcreator**
- **sysadmin**

The role **sysadmin** is required because the extractor uses some **alter database** commands.

If the Data Extractor uses an **existing database**, make sure that the corresponding database user has at least the following permissions:

- A connection to the database must be possible.
- The user must be able to create a table. This is only necessary when you create or update the database schema.

Additionally, the user needs to have the following roles:

- **db_datareader**
- **db_datawriter**

Database User Permissions for the Report Generator

In order to retrieve data from the database to generate reports, you can use the same database user as for the Data Extractor. However, we recommend using a dedicated **read-only user** instead. This user only needs the following permission:

- Connection to the database

Additionally, this user needs the following role:

- **db_datareader**

Database Schema

You can automatically add the required database schema to the database management system by running the Data Extractor executable in GUI mode. Using the **Create/Update Database** function, the Data Extractor will create the specified database if it has not existed before and all required tables. Please see section [Database Setup](#) for details.

Reporting Server

You can use any reporting software you like. It just has to be able to retrieve the data that is created by the Data Extractor from the database and process it accordingly. We provide neither any recommendations nor any support for your reporting system. It is specific to your setup and needs.

Part 4

Running the Data Extractor

4 Running the Data Extractor

After you have installed the Data Extractor, run it in the GUI mode. Specify the settings to access your [PRTG server](#), optionally [filter the data](#) which will be retrieved, and optionally specify an HTTP Push Data Advanced sensor to [monitor the result](#) of the data extraction in PRTG. Also, [set up the database](#) accordingly.

Click on the **Run** button to start data extraction. You will see the progress of the extraction in the status bar above this button. You can also run the Data Extractor via the [command line](#). For best practice, we recommend you to run the extraction once a day to extract all new data from PRTG, so schedule the extraction.

The screenshot shows the 'Paessler Data Extractor' window with the following sections:

- PRTG Settings:** Server (empty), Https (unchecked), User (prtgadmin), Pass (empty), Port (80).
- Database Settings:** Server (empty), DB (empty), Win (checked), User (empty), Pass (empty), Buffer (empty), and a 'Create/Update DB' button.
- Sensor Settings:** Server (empty), Port (5050), Token (empty).
- Filter Settings:** From (30.06.2014), To (30.06.2014), Include (checked for Sensor Data and Toplist Data), and Sensor ID (0 - 1000000).

At the bottom, there is a 'Copy Parameters to Clipboard' button and a large 'Run' button.

PRTG Data Extractor GUI

PRTG Settings

Specify the PRTG installation from which you want to extract data in section **PRTG Settings**. We recommend you to use a second PRTG installation for the Data Extractor (see section [Requirements and Setup](#)). Please enter the settings for this additional PRTG server.

PRTG Settings	
Server	Enter the IP address or DNS name of the PRTG server you want to use for data extraction.
Https	Specify the protocol of your PRTG server. Mark the checkbox to connect to your PRTG server via HTTPS . Otherwise, if you use HTTP , leave the checkbox empty.
User	Enter the username for your PRTG server.
Pass	Enter password of the specified user for your PRTG server.
Port	Enter the port to which PRTG is listening. For HTTP connections it is port 80 by default, for HTTPS it is port 443 . The Data Extractor will fill in the corresponding port automatically depending on whether you choose HTTPS above or not.

Database Settings

Specify the database in which the retrieved data will be stored in section **Database Settings**. For details, see also section [Database Setup](#).

Database Settings	
Server	Enter the IP address or DNS name of the SQL server. If it requires an instance, you can specify it as Server\Instance .
DB	Enter the name of the database.
Win	<p>If you use the same user credentials for both running the Data Extractor and for the database, you can mark this checkbox. Your Windows username and password will then be applied to the database. In this case you do not need to specify username and password below.</p> <p>If you want to use a dedicated database user, leave this checkbox unmarked.</p>
User	Enter the username for the database. You need only to do so if you do not use the Windows credentials ("Win" checkbox unmarked).
Pass	Enter the password for the database user. You need only to do so if you do not use the Windows credentials ("Win" checkbox unmarked).

Sensor Settings

You can monitor the data extraction with PRTG by sending the results to an [HTTP Push Data Advanced sensor](#). Add this sensor type to your PRTG server and specify its "address" (server, port, token) in section **Sensor Settings**.

Sensor Settings

Server	Enter the IP address or DNS name of the PRTG server on which the HTTP Push Data Advanced sensor runs. If you leave this field empty, the Data Extractor will not send its results to anywhere.
Port	Enter the number of the port on which the sensor is listening for incoming HTTP calls. You can look up and change the used port in the sensor settings in PRTG.
Token	Enter the Identification Token of the sensor. You can look up and change the token in the sensor settings in PRTG.

Please see section [Monitoring the Data Extraction](#) for details.

Filter Settings

You can define a specific time interval, specific sensor(s), and the kind of data which will be extracted from PRTG in section **Filter Settings**. **Caution:** The more you include here, the longer the data extraction will take.

Filter Settings	
From	Specify the day from which on the data will be extracted from PRTG. Enter the date directly (dd.mm.yy) or use the date picker.
To	Specify the last day from which the data will be extracted. Enter the date directly (dd.mm.yy) or use the date picker.
Include	<p>Specify the type of data which will be extracted. Because the data extraction might take very long and might need a lot of disk space, you can exclude the following data by unmarking the corresponding checkbox:</p> <ul style="list-style-type: none"> ▪ Sensor Data: In- or exclude monitoring data which is shown, e.g., in the live data table in PRTG. ▪ Toplist Data: In- or exclude monitoring data which is shown in toplist. <p>Note: All other data will be always extracted, e.g., the device tree ("metadata"). See section Database Schema Description.</p>
Sensor ID	<p>Specify the sensors which will be considered for data extraction. Enter sensor IDs to define a range of sensors.</p> <p>Note: Only monitoring data will be excluded if a sensor's ID is not in the range. "Metadata" will not be affected.</p>

Command Line Mode

You can run the Data Extractor via a command line tool but this console mode is limited in one aspect: It can only extract the data from the day before. So, if you use the command line mode, we recommend you to run the data extraction once a day. You can configure every other setting with command line parameters.

In order to get the required parameters for the command line, open the Data Extractor GUI and configure the settings as described above. Then click on the **Copy Parameters to Clipboard** button (the parameters will be cached automatically) and paste the data into the console (**Ctrl +v**). **Note:** For security reasons, passwords will be copied as placeholders. Replace the placeholders appropriately before executing the command.

The data extraction will start automatically with the given parameters after the Data Extractor has been started and will exit the program as soon as the extraction is finished.

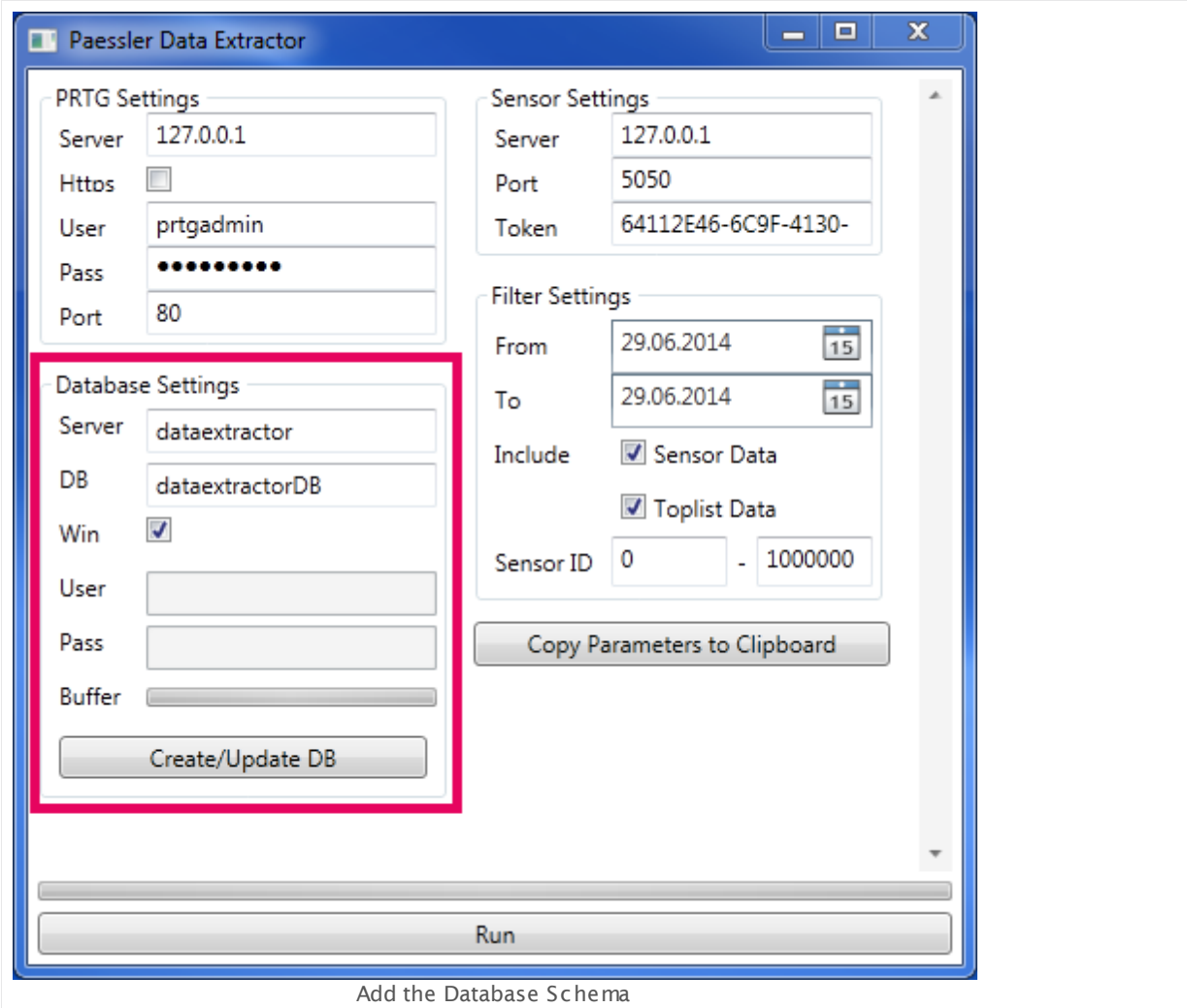
Part 5

Database Setup

5 Database Setup

You can add the database schema to the database management system by starting **dataextractor.exe** in the GUI mode. The executable will run automatically the GUI mode if you do not specify any parameters.

In order add the database schema click on the **Create/Update DB** button.

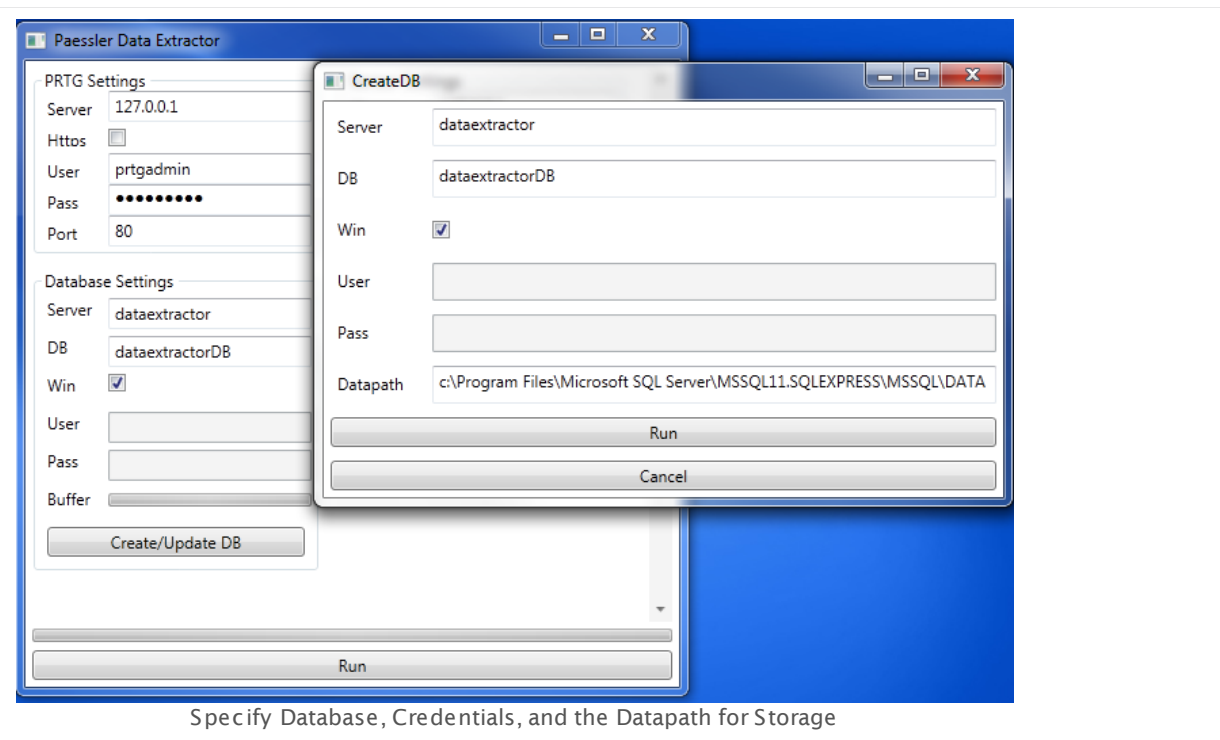


In the appearing **Create DB** window, specify your [database settings](#).

- Enter the SQL server name into the field **Server**.
- Enter the database name into the field **DB**.
- Enter the credentials for the database: username into the field **User**, password into the field **Pass**. If you use your Windows credentials, mark the checkbox for **Win** and leave the other credential fields empty.

- Enter a path on a disk on the system of your SQL server into the field **Datapath**. The data files will be stored there.
- Click on the **Run** button and confirm the following popup window.

The Data Extractor will then create the defined database if it has not existed yet (and update otherwise). It will add all required tables. Please wait until the Data Extractor has finished the setup.



Note: After updating the Data Extractor, it might be necessary to update your database to the latest version (click **Create/Update DB**). The import will return an error if the database schema is too old. The database will not update automatically to allow using an account with stricter permissions for the data extraction. We also strongly recommend you to create a backup of your database before applying any updates.

Please see section [Database Schema Description](#) for details about the database schema.

5.1 Database Schema Description

This section describes the available database tables for the Data Extractor and shows [use cases](#) of database schemes.

Tables	Description
System	<p>The system table contains a list of all installation from which the Data Extractor has retrieved data. The key is the PRTG Server field from the extractor so try use always the same value there. We recommend using the DNS name of your PRTG server so that you can identify it more easily.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ ID: the IP address of the PRTG server ▪ Name: the name of the PRTG server ▪ Port: the number of the port which is used for the PRTG server ▪ Protocol: the protocol which is used for the PRTG server
Group	<p>The group table contains a list of all probes and groups from which the Data Extractor has retrieved data.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server which contains this probe or group ▪ ID: the ID of this probe or group on the PRTG server ▪ Name: the display name of this probe or group ▪ Parent: the ID of the parent group. Note: The Root group has a NULL-reference because it does not have a parent in the PRTG device tree either.
Device	<p>The device table contains a list of all devices from which the Data Extractor has retrieved data.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server which contains this device ▪ ID: the ID of this device on the PRTG server ▪ Name: the display name of this device ▪ Parent: the ID of the parent group (i.e., the group which contains this device)
Sensor	<p>The sensor table contains a list of all sensors from which the Data Extractor has retrieved data.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server on which this sensor runs ▪ ID: the ID of this sensor on the PRTG server

Tables	Description
	<ul style="list-style-type: none"> ▪ Name: the display name of this sensor ▪ Device: the ID of the parent device of this sensor (i.e., the device which this sensor monitors) ▪ SensorKind: the type of this sensor. Note: The PRTG internal type names are used here. ▪ Interval: the scanning interval of this sensor as defined in PRTG
Channel	<p>The channel table contains all channel settings of the target PRTG servers. It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server on which this channel runs ▪ Sensor: the ID of the sensor which includes this channel ▪ ID: the ID of this channel ▪ Name: the display name of this channel <p>The channel settings are given in additional columns with the current specifications. Most of these columns are self-explanatory for PRTG users. See also the PRTG API (Application Programming Interface) documentation for custom sensors in the PRTG web interface (section Advanced EXE/Script, Advanced SSH Script, and Advanced HTTP Push Data sensors: Elements). There you can find descriptions of possible values for particular channel settings.</p> <ul style="list-style-type: none"> ▪ mode ▪ kind ▪ decimalmode ▪ decimaldigits ▪ speedsize ▪ speedtime ▪ volumesize ▪ isPrimary ▪ customunit ▪ valuelookup ▪ Limit LowerError ▪ Limit LowerWarning ▪ Limit UpperWarning ▪ Limit UpperError

Tables	Description
AbsoluteData	<p>The absolute data table contains all measurements within the defined time span from channels with absolute values.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server from which the value is extracted ▪ Sensor: the ID of the sensor from which the value is extracted ▪ Channel: the ID of the channel from which the value is extracted ▪ Time: the timestamp of the extracted value (i.e., the time of the corresponding sensor scan) ▪ Value: the measurement of this channel at the given time ▪ Date: the day of the measurement
DifferenceData	<p>The table for difference data is similar to the AbsoluteData table but contains two measurement values for every entry (just like in the PRTG web interface for "delta" sensors which have always two value columns):</p> <ul style="list-style-type: none"> ▪ Volume: the volume value of this channel ▪ Speed: the speed value of this channel
SensorStatus	<p>The sensor status table contains a log with all status changes of every sensor.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server from which the value is extracted ▪ Sensor: the ID of the sensor from which the value is extracted ▪ Time: the timestamp of the extracted value (i.e., the time of the corresponding sensor scan) ▪ Status: the code of the status that the sensor turned to at the given time. See the Constants table for a description of the status codes.
Toplist	<p>The toplist table contains a list of all existing toplist.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server on which this toplist exists ▪ Sensor: the ID of the sensor which includes this toplist ▪ ID: the ID of this toplist ▪ Name: the display name of this toplist
ToplistIDX	<p>The toplist index table contains a list of all toplist indexes.</p> <p>It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server on which this toplist exists

Tables	Description
	<ul style="list-style-type: none"> ▪ Sensor: the ID of the sensor which includes this toplist ▪ Toplist: the ID of this toplist ▪ ID: the index of the toplist; for each interval, this ID will be increased by one ▪ Start and End: the start point and end point of the toplist interval ▪ Interval: the time span of the toplist interval in seconds
ToplistData	The toplist data table contains all columns which are available for toplist. The particular column names are self-explanatory for PRTG users. Note : If a specific column is not selected in the toplist settings in PRTG, it will be referenced as NULL .
Notification	<p>The notification table contains a list of all configured notifications. It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server on which this notification is configured ▪ ID: the ID of this notification ▪ Name: the display name of this notification ▪ Active: the activity status of this notification. 0 means inactive, 1 means active.
Trigger	<p>The trigger table contains a list of all defined notification triggers. If a trigger is inherited, it will be listed multiple times. It has the following columns:</p> <ul style="list-style-type: none"> ▪ System: the PRTG server on which this trigger is set up ▪ Parent: the ID of the object from which this trigger is inherited (or the ID of the sensor itself if there is no inheritance) ▪ Sensor: the ID of the sensor for which this trigger is set up (also if is inherited) ▪ ID: the ID of the trigger regarding one sensor; if one sensor has more than one trigger, each trigger of this sensor will have a unique ID ▪ TriggerType: the kind of trigger ▪ Notification: the ID of the notification which is triggered when the first trigger condition is met ▪ Escalation: the ID of the notification which is triggered when the escalation condition is met ▪ Clearing: the ID of the notification which is triggered when the condition clears ▪ Source: the source of the trigger (self, parent, library); see the Constants table for the meaning of a particular source ID

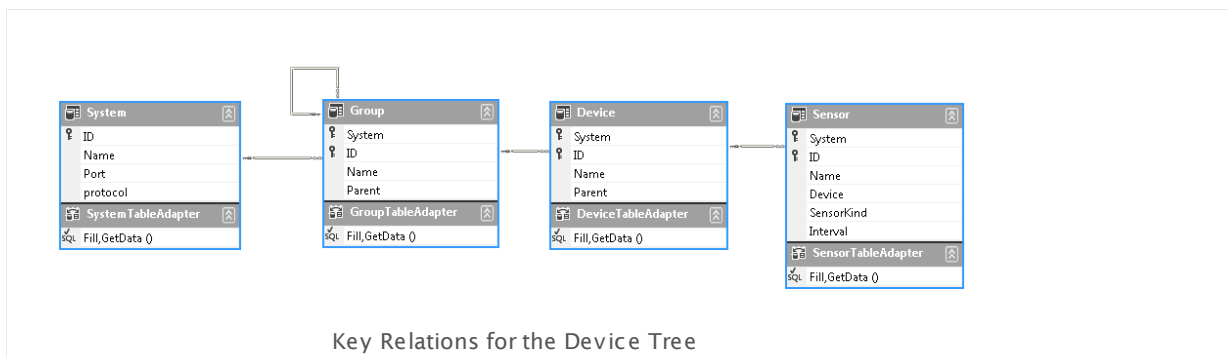
Tables	Description
	<ul style="list-style-type: none"> ▪ Condition: the status of the sensor which provokes the trigger ▪ ValidUntil: This field indicates when the trigger was found and extracted from PRTG the last time. A trigger will not be removed from the table if it was removed in PRTG or its inheritance changed but the ValidUntil field will not be updated anymore. So you can see if a trigger still exists or not.
Logs	The logs table contains information about problems during the data import into the database.
Constants	The constants table shows descriptions of values in some columns of some other tables (e.g., sensor states, toplist protocols, trigger sources). Every constant has always a fix reference to the table and column in which it can appear. Some constants are inserted during database creation, some are added on the fly when they are needed. Constants are stored independent from the system.
ChangeLog	The change log table contains version related information about changes in the database structure. When you install an update of the Data Extractor which causes changes to the database, all applied changes are listed in this table. You can also use the change log to see which version is installed if a report requires a specific version of the database.

Use Cases of the Database Schema

This section shows key relations in the database. In general, the database schemes represent the object hierarchy in PRTG.

Device Tree

This is the "basic use case" that you can combine with most of the other use cases. Everytime a sensor is referenced it might be useful to know the tree path of this sensor.

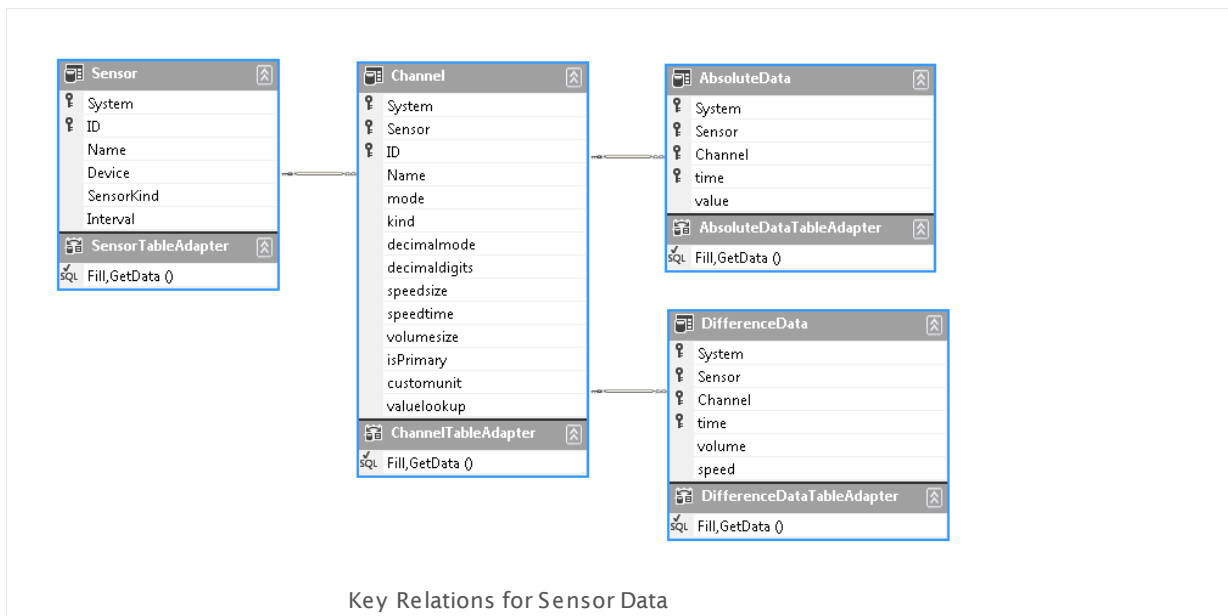


This schema basically shows:

- **System** can reference an unlimited number of **Group**, **Group** can reference only one **System**
- **Group** can contain an unlimited number of **Group**, **Group** can be in only one **Group**
- **Group** can reference an unlimited number of **Device**, **Device** can reference only one **Group**
- **Device** can reference an unlimited number of **Sensor**, **Sensor** can reference only one **Device**

Sensor Data

This schema shows the key relations for sensor data.

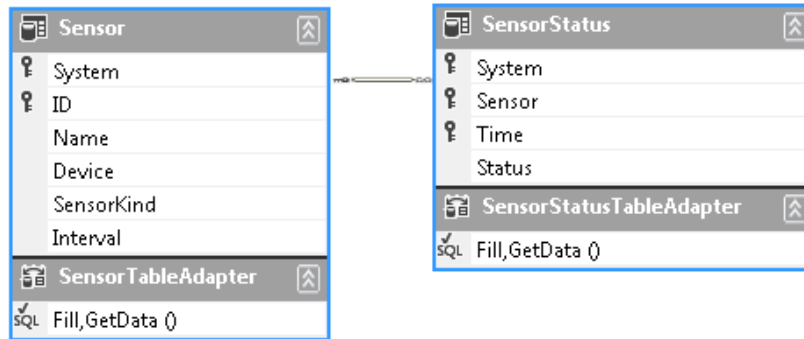


This schema basically shows:

- **Sensor** can reference an unlimited number of **Channel**, **Channel** can reference only one **Sensor**
- **Channel** can reference an unlimited number of **AbsoluteData** and **DifferenceData**
- **AbsoluteData** and **DifferenceData** can reference only one **Channel** respectively

Sensor Status

This schema shows the key relations for sensor states.



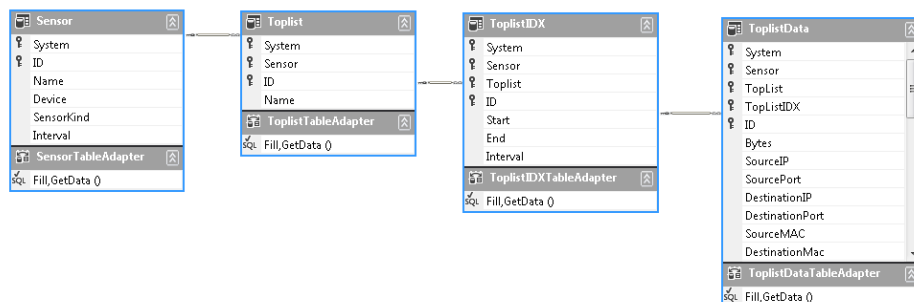
Key Relations for Sensor Status

This schema basically shows that **Sensor** can reference an unlimited number of **SensorStatus**, **SensorStatus** can reference only one **Sensor**. If you want to know the status of a sensor at a specific time, you can retrieve the latest entry from the **SensorStatus** table before that time.

Note: This method is an approximation because any reboot or failure of the PRTG core or probe service will cause a delay of the status log.

Toplists

This schema shows the key relations for toplist.



Key Relations for Toplists

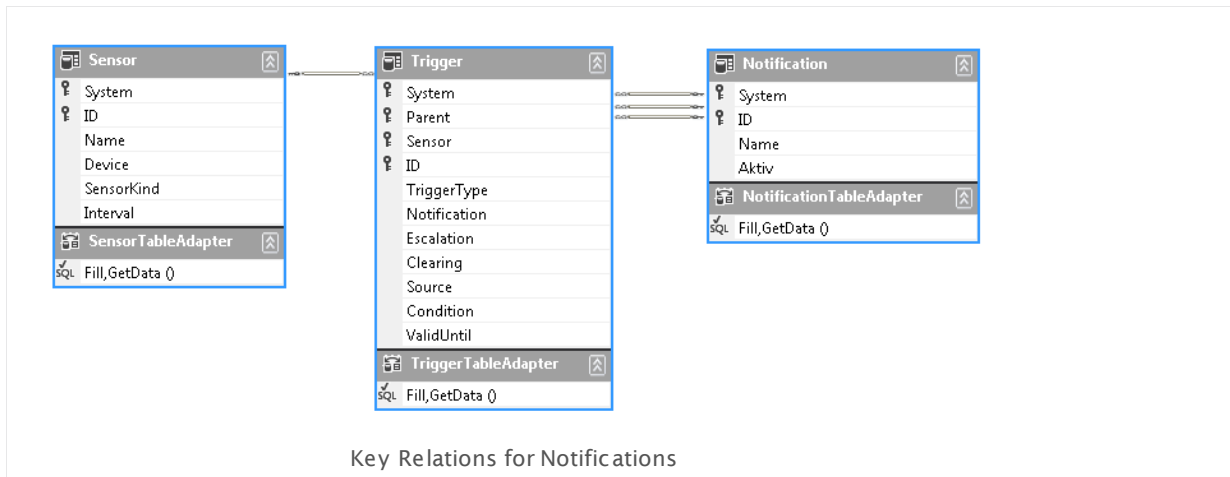
This schema basically shows:

- **Sensor** can reference an unlimited number of **Toplist**, **Toplist** can reference only one **Sensor**
- **Toplist** can reference an unlimited number of **ToplistIDX**, **ToplistIDX** can reference only one **Toplist**

- **ToplistIDX** can reference an unlimited number of **ToplistData**, **ToplistData** can reference only one **ToplistIDX**

Notifications

This schema shows the key relations for notifications.



This schema basically shows:

- **Sensor** can reference an unlimited number of **Trigger**, **Trigger** can reference only one **Sensor**
- **Trigger** can reference three **Notifications** (provoking condition, escalation, clearing), **Notification** can reference an unlimited number of **Trigger**

Part 6

Monitoring the Data Extraction

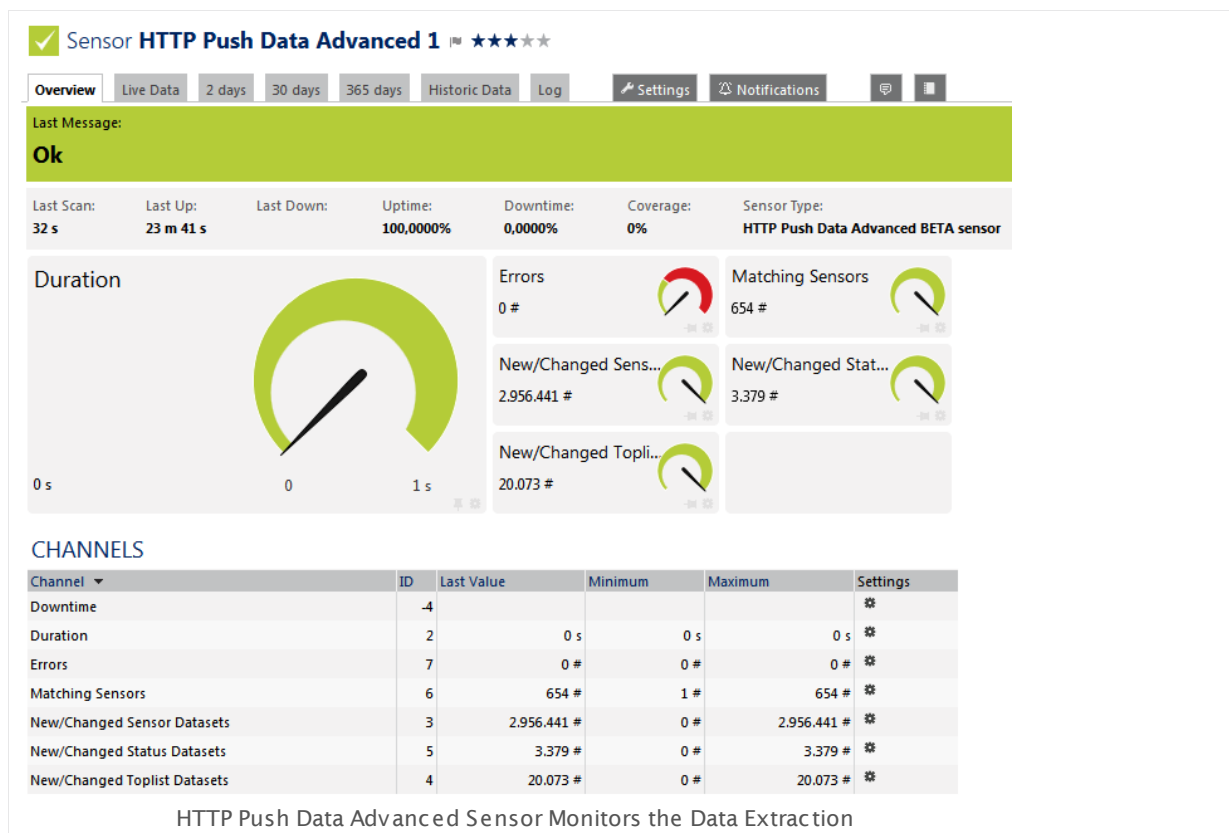
6 Monitoring the Data Extraction

PRTG can monitor the data extraction with an [HTTP Push Data Advanced sensor](#). Add this sensor type to PRTG and configure its "address" in section [Sensor Settings](#) of the Data Extractor. The Data Extractor will send a push message to your specified PRTG server each time the data extraction is finished or has been canceled.

The HTTP Push Data Advanced sensor will show you the following information in dedicated channels:

- **Duration:** the time the data extraction took
- **New/Changed Sensor Datasets:** the number of sensor datasets which were extracted
- **New/Changed Toplist Datasets:** the number of toplist datasets which were extracted
- **New/Changed Status Datasets:** the number of status datasets which were extracted
- **Matching Sensors:** the number of sensors which matched your specifications in [Filter Settings](#)
- **Errors:** the number of errors which occurred during the data extraction; see the [Logs table in the database](#) or the [LogViewer demo report](#) for details about any errors

Note: If you run the extraction twice for the same timespan, the older sensor values will be deleted from the sensor data before the new values are added.



Part 7

Demo Reporting

7 Demo Reporting

In order to demonstrate the usage of the Data Extractor to generate your own customizable reports with it, we provide you some demo reports. This manual section shows the software you need to run them and how to start reporting.

Note: Please understand that we do not provide any support for any issues with custom reports!

Requirements

You need the following software for the demo report generation:

- **Internet Information Server (IIS)**

The **Internet Information Server (IIS)** is part of Windows and is either already installed or you can add it as role/feature. It is required for the **Microsoft SQL Server Reporting Services (SSRS)**.

- **SQL Server Reporting Services (SSRS)**

The **Microsoft SQL Server Reporting Services (SSRS)** are part of your SQL server installation. You can add them by using the SQL server installer. When you run the configuration manager for SSRS, it will ask for the specification of a database. This is a dedicated database for storing metadata about the reports; it is **not** the same database where the Data Extractor stores its data! We recommend you to use a dedicated database for this purpose.

You need also to configure a **web service URL** as well as a **manager URL**. The web service URL will be used to access the reports after generation; you will need the management URL to add reports or change permissions. We recommend you to use **SSL** with a valid certificate for both URLs.

- **Business Intelligence Development Studio**

You can install the **Business Intelligence Development Studio** also using the SQL server installer. You need it to make changes to demo reports. With this software, you will be able open the demo project and edit the reports as you like.

Note: You will need to add the **TargetServerURL** in the report project properties to management URL of your report server before you can use the **Deploy** function. You will also need to change the settings of **DataSource1** according to your database.

Adding Your Data Source

After you have installed the required software, you have to add a data source. The data source has to address the database which is used by the Data Extractor. See section [Requirements and Setup](#) for information about the needed permissions.

In order to add the data source, follow the steps below:

- Browse to the **management URL** on the report server.
- Click on **Data Sources**.

- Choose **New Data Source** from the menu.
- Fill in all required information.
- In the **Connection String** field, we recommend you to enter the following string (replace the placeholders accordingly):

```
Data Source=<your_server>;Initial Catalog=<your_database>
```

Adding the Demo Reports

After you have defined your data source, add the demo reports which use the data source.

- Browse to the **management URL** on the report server.
- Click on **Upload File**. You can create a folder for the reports first and then upload the report files to it.
- After the upload is finished, click on a report name to edit its settings. Either a blank page or the report will be shown. Open the settings by clicking the report name on top of the page.
- Open the category **Data Source** and select the data source which you have just created.

After saving the settings, you can open the report:

- Browse to the **management URL** or to the **report service URL**.
- Click on the report name.
- Reports require at least one parameter so adjust it at the top of the page.
- Click on **Show Report** to generate the report.

You can also define a default value for one of the parameters, for example, the **system** parameter. If you want to use defaults, configure them on the report's details page (right next to the data source setting).

Available Demo Reports

We provide the following demo reports which you can use to test the capabilities of the Data Extractor:

Demo Reports	
DataSize	Reports the size of a specific device/sensor in the database.
DataSizeTime	Reports the number of datasets which the database contains for a specific day.
DeviceStatus	Reports the summarized status history of all sensors on a specific device for a defined time span.
LogViewer	Reports the logs of the Data Extractor.

Notifications	Reports the notifications usage.
SensorData	Reports the data of a specific sensor.
SensorGraph	Shows the historic data graph of a specific sensor.
SensorStatus	Reports the status history of a specific sensor for a defined time span.
SystemHealth	This report is a demo for gauges . It reports the last value of all System and Probe Health channels whose data were extracted.
ToplistData	Reports the data of a specific toplist.