DomotiGa

Open Source Home Automation for Linux

Manual Date: January 2014

DomotiGa
Open Source Home Automation Software for Linux

Table of Contents

List of Tables

1 Introduction

1.1 Overview

DomotiGa is a Home Automation system designed to control various devices, receive input from various sensors running on the Linux platform.

1.2 Why is it called DomotiGa?

In The Netherlands Home Automation is called Domotica, and since it is written in Gambas, the last part of this word is changed to 'Ga'. This results in one of the few words that don't have zillions of hits in Google, which is handy :-).

1.3 History

DomotiGa is started by rdnzl in 2008, as a replacement for Misterhouse (he even created the now very outdated ia5 web interface for it). Rdnzl initially looked around for another open-source application, but it turned out that most weren't open-source or the application couldn't do that much. Rdznl was busy with a re-write of Misterhouse and then he came across Gambas basic, he tried some things out and this turned out to be easy with the Gambas IDE.

1.4 Features

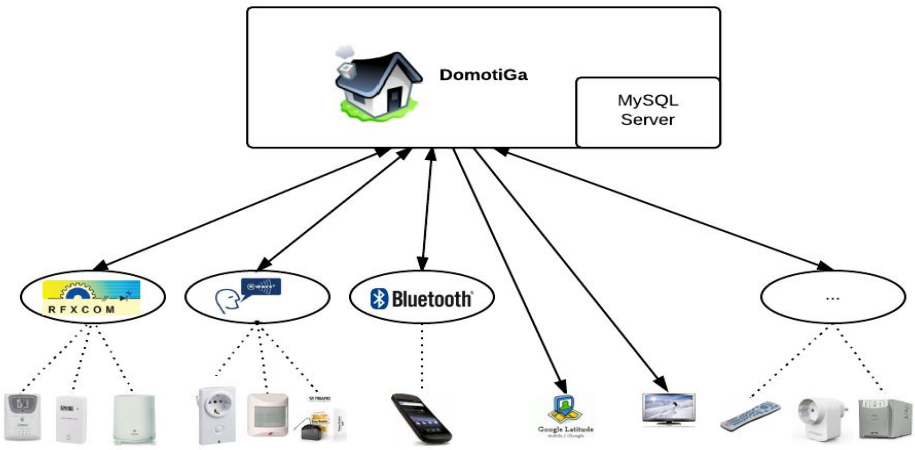
DomotiGa has a long list of features, the following are the major ones:

- Open-source, licenses under GPLv3 (GNU Public License version 3)
- Runs on almost any recent Linux distribution
- Supports a wide range of devices and modules, e.g. 1-Wire, Bluetooth, Cameras, FritzBox, Glatitude, KNX EIB, OpenTherm, Plugwise, RFXCom, X11, Z-Wave. For the complete list of supported devices, see the following link: http://www.domotiga.nl/projects/domotiga/wiki/Modules
- Show statistical web graphs, e.g. power usage, temperature, humity
- Optional Web browser client
- Optional Android client
- Optional iPhone client

1.5 Architecture

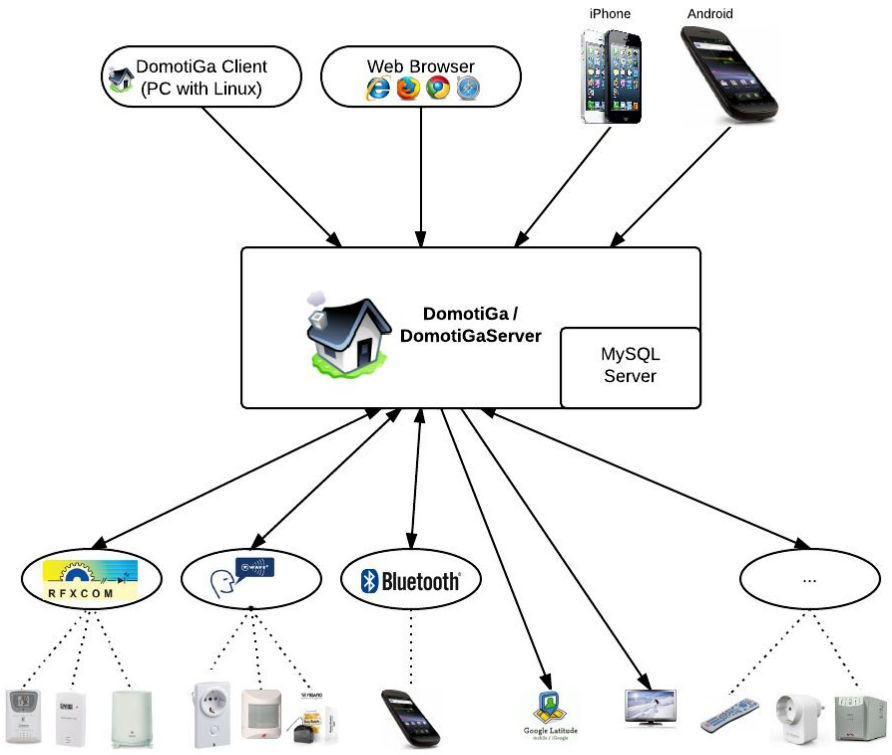
The main component of DomotiGa is the DomotiGa backend, which connects to all devices and modules. The DomotiGa backend can be run with GUI (DomotiGa) or without GUI (DomotiGaServer), the choice is normally based on the hardware platform, the without GUI doesn't need a desktop environment and requires less system resources.

The following overview shows DomotiGa running with a GUI, the GUI is used for setup, configuration and view/control devices and collected information:



It is also possible to run DomotiGa as backend only, the following overview shows DomotiGa/DomotiGaServer as server and clients will connected to it. At this moment the setup and configuration needs to be done via DomotiGa directly, if no GUI is available on the server, DomotiGa needs to be installed as full client on another Linux system. It is also possible to use clients like a web browser, iPhone and Android to view and control devices.

Currently the web browser functionality is fairly limited and is being worked on, to give it (almost) the same functionality as the DomotiGa GUI.



2 How To Start

DomotiGa (or Domotics) can be somewhat overwelling for first-time user. In this chapter the terminology used by DomotiGa is described and the basic work flow how to get started.

2.1 Terminology

The following is the list of terminology used in DomotiGa:

Modules – A module is the part of DomotiGa which can be enabled to do a specific action. This can be e.g. XML-RPC, Web or Telnet service to retrieve information from DomotiGa, but also send e.g. emails, play sounds or twitter depending on a certain action.

Interfaces – An interface is a special type module and can be assigned to one or more DomotiGa device(s). The interface module will connect via e.g. serial or network externally and will process data, the data can be pushed and/or pulled and stored in devices. An example of an interface is Z-Wave, which connects the Aeon Labs Z-Wave USB controller and can communicate with Z-Wave switches, sensors, etc.

Devices – For each switch, dimmer, temperature sensor, smoke detector, remote, etc a device has to be created in DomotiGa. This device stores the information about the device state like e.g. on, off and temperature. This state information can be logged with every change, so that you e.g. know which temperature it was at what time of a certain day. The devices state change are also used to execute events, but also be controlled by events.

Events – Events are one of the most important features of DomotiGa, with them you can execute actions based on conditions. Some examples of events are turning on the light when it is gets dark, send an email if somebody trips a the motion sensor, play a “welcome home” sound if somebody opens the door.

Actions – TBD

Triggers – TBD

Conditions – TBD

2.2 Work Flow

This is the recommended work flow to install and start using DomotiGa. This should get you quickly familiar with it, and a working system. The flow consists of:

- Installation of Operating System
- Installation of required 3rd party software dependencies, Gambas and MySQL
- Installation of DomotiGa
- First Time Setup of DomotiGa – This will setup the required database tables

- Configure localized settings of DomotiGa – Sets your location and timezone information to calculate your sunrise and sunset times.
- Modules – Enable, disable or configure specific module. It is recommended to leave it as default when you setup for the first time, a few standard modules are enabled by default already.
- Interfaces – Adds and configures interfaces. It is recommended to start with 1 interface only, and add devices and events in combination with this interface before adding additional interfaces.
- Devices – Add and configure a device, connected to an interface.
- Events – Add and configure an event, which will execute certain actions based on conditions.

During the First Time setup the default option is to start with a Demo database, which is highly recommended. This gives you the opportunity to play and learn DomotiGa, to get a good sense of the possibilities and options before you start to configure it for your final setup.

2.3 How To Documents

TBD, but the following are thought about/planned:

- RFXCom transceiver + devices
- Z-Wave controller + devices
- Events based on devices, and based on time (globalvars)
- Events – more on actions & advanced conditions

3 Installation

This chapter describes the requirements and how to install DomotiGa on your system.

3.1 System Requirements

DomotiGa will run on any system which can run Gambas, this includes the most widely used Linux distributions.

3.1.1 Linux Distribution

The following list of distributions have been succesfully tested with DomotiGa:

Distribution	Architecture
Ubuntu 10.04	x86 / x86_64
Ubuntu 10.10	x86 / x86_64
Ubuntu 11.04	x86 / x86_64
Ubuntu 12.04	x86 / x86_64
Ubuntu 12.10	x86 / x86_64
Ubuntu 13.04	x86 / x86_64
Ubuntu 13.10	x86_64
Debian 7.0 (Wheezy)	x86 / x86_64 / ARMv6 (Raspberry Pi)
Debian 7.1 (Wheezy)	x86 / x86_64 / ARMv6 (Raspberry Pi)

x86 is a 32-bits and x86_64 is a 64-bits architecture

For a full list of Linux distributions which should work with Gambas, see the following link: http://gambasdoc.org/help/doc/distro

3.1.2 Hardware

DomotiGa will run on any hardware which supports Linux and Gambas3. The following hardware has been succesfully tested:

Hardware
Any Intel / AMD x86 CPU
Any Intel / AMD x64 CPU
Raspberry Pi (Model B)
Cubieboard2

It is recommended to have at least 512MB of memory for DomotiGa and 256MB of memory for DomotiGaServer. Also a minimum 500MB of diskspace is required for the DomotiGa installation.

Virtualization environments like VMWare, VirtualBox and Parallels have been succesfully tested with DomotiGa.

3.2 Installation Overview

The complete installation of DomotiGa consists of the following components:

- Operating System
- Linux Configuration
- Gambas3
- MySQL Server
- DomotiGa

3.3 Operating System

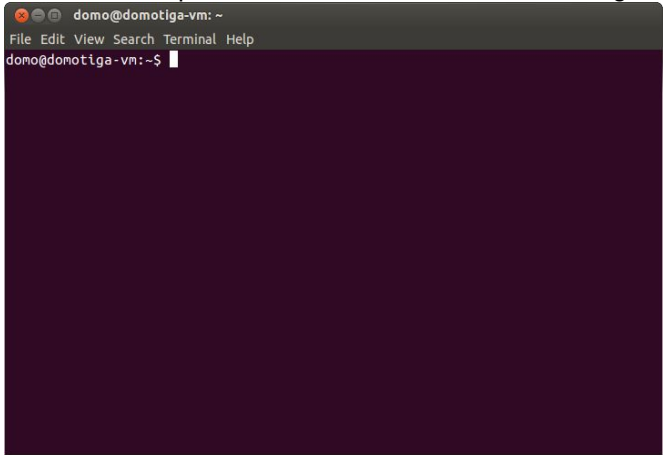
It is recommended to install the Linux Operating System with the standard packages, including the graphical desktop (GNOME, KDE, LXDE, Unity, etc). The graphical desktop is required to run DomotiGa. The DomotiGaServer will run without the graphical desktop installed, but the customized installation is only recommended for advanced users.

The following table contains links to the installation procedures for the main Linux distributions:

Distribution	Link to Install manuals
Ubuntu (Desktop)	https://help.ubuntu.com/community/GraphicalInstall https://help.ubuntu.com/community/Installation
Lubuntu (Desktop)	https://help.ubuntu.com/community/Lubuntu/InstallingLubuntu
Debian (Desktop, x86 / x86_64)	http://www.debian.org/releases/stable/installmanual http://wiki.debian.org/LennyIllustratedInstall
Debian (Desktop, Raspberry Pi)	http://elinux.org/RPi_Easy_SD_Card_Setup http://elinux.org/RPi_raspi-config

Installation of Gambas, MySQL Server and DomotiGa are done via the command line. If you are unexperienced with this, please read the section "`Starting a Terminal`" in the following link:
https://help.ubuntu.com/community/UsingTheTerminal

After a terminal window is opened, it should look like the following screen:



3.4 Linux Configuration

After the Operating System is installed a few options are highly recommended to install/configure on the system. These are NTP, SSH and USB Device permissions.

3.4.1 Debian sudo

This section needs to be executed if you are running under Debian, if you use Ubuntu, you can skip this section. By default Debian doesn't allow your user to execute the "`sudo`" command. This command is needed for all kind of administrative tasks. Execute the following command to change to "`root`" first, the "`root`" password is what you configured during installation:

```
$ su -
```

When you have "`root`" privileges you can add your user account to the "`sudo`" group with the following command. In the example the user account is named "`domo`".

```
# adduser domo sudo
```

After you added the "`sudo`" group to your user account, you **need** to logout and login again to activate this change.

3.4.2 NTP

It is important that your system has the right time, you can achieve this by running ntpd (Network Time Protocol daemon). The following steps will install, configure and check it.

First install the required software package:

```
$ sudo apt-get install ntp
```

The next step is optional, but still highly recommended. This will replace the generic "ntp.ubuntu.com" server with the nearby NTP servers:

```
$ sudo nano /etc/ntp.conf
```

Replace the line with the following ones, the example is if you live in the Netherlands. For a complete list for your local NTP servers, check the following link (and go to the active servers): http://www.pool.ntp.org

```
server 0.nl.pool.ntp.org
server 1.nl.pool.ntp.org
server 2.nl.pool.ntp.org
server 3.nl.pool.ntp.org
```

Restart the NTP daemon:

```
$ sudo /etc/init.d/ntp restart
* Stopping NTP server ntpd                                          [ OK ]
* Starting NTP server ntpd                                          [ OK ]
```

Now check if you system is in synchronisation with the NTP servers:

```
$ ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 metronoom.dmz.c PPS(0)           2 u   18   64    1   28.042  36694.5   0.001
 ntp1.hro.nl     193.67.79.202    2 u   17   64    1   30.153  36694.5   0.001
 canopus.no-sens 192.87.106.3     2 u   16   64    1   26.090  36694.6   0.001
 ntp1.trans-ix.n 193.79.237.14    2 u   15   64    1   25.615  36696.2   0.001
```

The offset is the difference between your servers clock and the atom clock time. After a few minutes you will see that the difference is corrected.

```
$ ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 metronoom.dmz.c PPS(0)           2 u   10   64    1   26.920   -0.251   0.001
 ntp1.hro.nl     193.67.79.202    2 u   25   64    1   30.268    0.648   0.001
 canopus.no-sens 192.87.106.3     2 u   25   64    1   26.651    0.273   0.001
 ntp1.trans-ix.n 193.79.237.14    2 u   49   64    1   24.986    2.846   0.001
```

3.4.3 SSH

By default most Linux distributions don't install the ssh daemon (server), which is required to remotely access the system. If you want to access your system remotely via command-line install ssh daemon with the following command:

```
$ sudo apt-get install ssh
```

3.4.4 USB Device Permissions

It is highly recommended to run DomotiGa not under the "`root`" user. By default the standard user account(s) can't read and/or write to the USB serial devices mainly used by DomotiGa. The following command will add the "`dialout`" group to your user account to give the right permissions, in the example the user account is named "`domo`":

```
$ sudo addgroup domo dialout
```

After you added the "`dialout`" group to your user account, you **need** to logout and login again to activate this change.

3.4.5 RRDTool

The Round Robin Database Tool (RRDTool) is the open source industry standard, high performance data logging and graphing system for time series data. With it you can create graphs of the collected data. It is highly recommended to install it, and can be done with the following command:

```
$ sudo apt-get install rrdtool
```

3.5 Gambas

Gambas is a free development and runtime environment based on a Basic interpreter, a bit like Microsoft Visual Basic (but it is not a clone) – http://gambas.sourceforge.net/. DomotiGa is written in Gambas and requires it to run.

When this document was created, the latest version of Gambas is 3.5.2, this is minimum recommended version to use with DomotiGa. It is also recommended to start with the regular releases and not with the TRUNK releases (these are development releases). The TRUNK is only advised if you run into issues with the current release.

There are 3 ways of installing Gambas:

- From PPA (Personal Package Archives) – This archive contains pre-compiled Gambas packages for Ubuntu. This is the fastest and easiest method of installing, but isn't available for all Ubuntu releases and/or architecture (then you need to compile it from source). This is described in section "3.5.1 Gambas From PPA"

- Compile from Source – Latest regular release. This is described in section “3.5.2 Gambas From Source“
- Compile from Source – TRUNK release

Currently there are 2 versions of Gambas available, they are Gambas 2.x and 3.x. The version 2.x isn't supported anymore by DomotiGa. Only use version 3.x with DomotiGa.

3.5.1 Gambas From PPA

Installing Gambas from an Ubuntu PPA is the fastest and easiest way. This will only work on Ubuntu and still maintained release (e.g. *Ubuntu 10.10, 11.04 and 11.10 are End Of Life already*). Normally the x86, x86_64 and ARM architectures are supported. The following commands will install Gambas from the PPA.

```
$ sudo apt-add-repository ppa:nemh/gambas3
$ sudo apt-get update
$ sudo apt-get build-dep gambas3
$ sudo apt-get install gambas3
```

Raspberry Pi can't use the PPA, because it isn't running Ubuntu and needs to be compiled from source.

If your setup can't use the Ubuntu PPA, you need to compile Gambas3 from source, as described in the next section.

3.5.2 Gambas From Source

This section describes the compilation from source and installation of Gambas. To compile from source, we need to first install library dependencies. The specific list of dependencies are dependant on Linux distribution and release.

3.5.2.1 Library Dependencies

The specific list of dependencies are dependant on Linux distribution and release, please select your distribution and release install them. If your release is newer then in the list below, choose the latest one below.

Ubuntu 10.04 / Ubuntu 11.04

```
$ sudo apt-get install subversion build-essential autoconf mysql-client libbz2-dev \
  libfbclient2 libmysqlclient-dev unixodbc-dev libpq-dev libsqlite0-dev libsqlite3-dev \
  libgtk2.0-dev libldap2-dev libcurl4-gnutls-dev libgtkglext1-dev libpcre3-dev \
  libsdl-sound1.2-dev libsdl-mixer1.2-dev libsdl-image1.2-dev libsage-dev libxml2-dev \
  libxslt1-dev libbonobo2-dev libcos4-dev libomniorb4-dev librsvg2-dev libpoppler-dev \
  libpoppler-glib-dev libasound2-dev libesd0-dev libdirectfb-dev libaa1-dev \
  libxtst-dev libffi-dev kdelibs4-dev firebird2.1-dev libqt4-dev libglew1.5-dev \
  libimlib2-dev libv4l-dev libsdl-ttf2.0-dev libgnome-keyring-dev libgdk-pixbuf2.0-dev \
  linux-libc-dev libgmime-2.4-dev libgsl0-dev libgstreamer-plugins-base0.10-dev
```

Ubuntu 12.04 / Ubuntu 12.10

```
$ sudo apt-get install subversion build-essential autoconf mysql-client libbz2-dev \
  libfbclient2 libmysqlclient-dev unixodbc-dev libpq-dev libsqlite0-dev libsqlite3-dev \
  libgtk2.0-dev libldap2-dev libcurl4-gnutls-dev libgtkglext1-dev libpcre3-dev \
  libsdl-sound1.2-dev libsdl-mixer1.2-dev libsdl-image1.2-dev libsage-dev libxml2-dev \
  libxslt1-dev libbonobo2-dev libcos4-dev libomniorb4-dev librsvg2-dev libpoppler-dev \
  libpoppler-glib-dev libasound2-dev libesd0-dev libdirectfb-dev libaa1-dev \
  libxtst-dev libffi-dev kdelibs5-dev firebird2.1-dev libqt4-dev libglew1.5-dev \
  libimlib2-dev libv4l-dev libsdl-ttf2.0-dev libgnome-keyring-dev libgdk-pixbuf2.0-dev \
  linux-libc-dev libgmime-2.6-dev libgsl0-dev libgstreamer-plugins-base0.10-dev
```

Ubuntu 13.04 / Ubuntu 13.10

```
$ sudo apt-get install subversion build-essential autoconf mysql-client libbz2-dev \
  libfbclient2 libmysqlclient-dev unixodbc-dev libpq-dev libsqlite0-dev libsqlite3-dev \
  libgtk2.0-dev libldap2-dev libcurl4-gnutls-dev libgtkglext1-dev libpcre3-dev \
  libsdl-sound1.2-dev libsdl-mixer1.2-dev libsdl-image1.2-dev libsage-dev libxml2-dev \
  libxslt1-dev libbonobo2-dev libcos4-dev libomniorb4-dev librsvg2-dev libpoppler-dev \
  libpoppler-glib-dev libasound2-dev libesd0-dev libdirectfb-dev libaa1-dev \
  libxtst-dev libffi-dev kdelibs5-dev firebird2.1-dev libqt4-dev libglew1.5-dev \
  libimlib2-dev libv4l-dev libsdl-ttf2.0-dev libgnome-keyring-dev libgdk-pixbuf2.0-dev \
  linux-libc-dev libgmime-2.6-dev libgsl0-dev libgstreamer-plugins-base0.10-dev \
  libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev libgmp-dev libopenal-dev \
  libalure-dev
```

Debian 7.0 / Debian 7.1

```
$ sudo apt-get install subversion build-essential autoconf mysql-client libbz2-dev \
  libfbclient2 libmysqlclient-dev unixodbc-dev libpq-dev libsqlite0-dev libsqlite3-dev \
  libgtk2.0-dev libldap2-dev libcurl4-gnutls-dev libgtkglext1-dev libpcre3-dev \
  libsdl-sound1.2-dev libsdl-mixer1.2-dev libsdl-image1.2-dev libsage-dev libxml2-dev \
  libxslt1-dev libbonobo2-dev libcos4-dev libomniorb4-dev librsvg2-dev libpoppler-dev \
  libpoppler-glib-dev libasound2-dev libesd0-dev libdirectfb-dev libaa1-dev \
  libxtst-dev libffi-dev kdelibs5-dev firebird2.1-dev libqt4-dev libglew1.5-dev \
  libimlib2-dev libv4l-dev libsdl-ttf2.0-dev libgnome-keyring-dev libgdk-pixbuf2.0-dev \
  linux-libc-dev libgmime-2.6-dev libgsl0-dev libgstreamer-plugins-base0.10-dev \
  libpoppler-cpp-dev libpoppler-private-dev libpoppler-qt4-dev
```

3.5.2.2 Download Gambas

After the dependencies are installed, we will download the Gambas source, as latest released version or the TRUNK from the SVN. After the download, the source will be compiled and installed on Linux.

Create temporary directory

We will create a temporary directory for the Gambas installation. The Gambas source will be downloaded and compiled here.

```
$ mkdir -p ~/install
```

Download Gambas Latest Release

After the temporary directory is created, we will download the latest Gambas release. The following command will do that.

```
$ cd ~/install
$ wget http://sourceforge.net/projects/gambas/files/gambas3/gambas3-3.5.2.tar.bz2
$ tar xvjf gambas3-3.5.2.tar.bz2
```

Download Gambas TRUNK Release

If you want to download the TRUNK release, the development release, this can be done as follows from the official SVN (subversion) repository.

```
$ cd ~/install
$ svn checkout svn://svn.code.sf.net/p/gambas/code/gambas/trunk gambas
```

3.5.2.3 Configure Gambas

After we succesfully downloaded the Gambas source, the build environment needs to be configured. Depending on the hardware, this can take some time, please be patient.

```
$ cd ~/install/gambas*
$ ./reconf-all
$ ./configure -C
```

The configuration should run succesfully and the following message can be safely ignored (normally the disabled components aren't used/needed by DomotiGa):

```
||
|| THESE COMPONENTS ARE DISABLED:
|| - gb.jit
|| - gb.media
||
```

3.5.2.4 Compile Gambas

After the build environment is prepared, we can compile the Gambas source to binary format. Depending on the hardware, this can take some time, please be patient.

```
$ make
```

3.5.2.5 Install Gambas

The last step will install the Gambas binaries into the right directories. Depending on the hardware, this can take some time, please be patient.

```
$ sudo make install
```

3.5.2.6 Test Gambas

As final step we will test if Gambas can be started succesfully. This should launch the Gambas IDE (Integrated Development Environment).

```
$ gambas3
```

When it shows the following screen, Gambas is successfully installed and working:

3.6 MySQL Server

DomotiGa uses the MySQL database to store its data in. The MySQL server hasn't been installed yet, this can be done with the following command.

```
$ sudo apt-get install mysql-client mysql-server
```

At the installation of MySQL the new password for the MySQL “root” user is asked, choose a new none-standard password for optimal security. This password will be used later to setup the DomotiGa tables into MySQL.

3.6.1 Increase MySQL Connection Time-Out

DomotiGa Client and backend require a persistent connection to the MySQL database. Normally DomotiGa generate enough database queries and updates to the connection open, but it is recommended to increase the connection time-out to a higher value. The MySQL configuration file is usually found in “`/etc/mysql/my.cnf`”. Execute the following command on the DomotiGa backend system to edit the MySQL configuration file:

```
$ sudo nano /etc/mysql/my.cnf
```

Check if the following line already exists, else add it to the file:

```
wait_timeout            = 2147483
```

If unfamiliar with the "nano" editor, please check the following link for a cheat sheet: http://www.tuxradar.com/content/text-editing-nano-made-easy

Restart the MySQL Server to use the new configuration with the following command:

```
$ sudo /etc/init.d/mysql restart
```

3.7 DomotiGa

After gambas and MySQL have been installed and configured, we finally can download DomotiGa.

Library Dependencies

These dependencies should already be installed, but just to be sure, they are checked/installed again.

```
$ sudo apt-get install subversion
```

Download DomotiGa

```
$ cd ~
$ svn checkout http://svn.domotiga.nl/domotiga/trunk/ domotiga
```

Never run DomotiGa as "root" user.

4 First Time Startup

This chapter describes the first time startup of DomotiGa, to initialize the required MySQL database tables. Currently DomotiGa can only be setup via DomotiGa GUI (running as backend or as a client connecting to DomotiGaServer). Both methods are described in this chapter, but the DomotiGa GUI as backend is recommended for first time users.

4.1 DomotiGa

This is the recommended option for the first time users of DomotiGa, to setup and run DomotiGa on the Linux graphical desktop. DomotiGa will start the wizard to setup the MySQL database tables, also it will create log directories and configuration files.

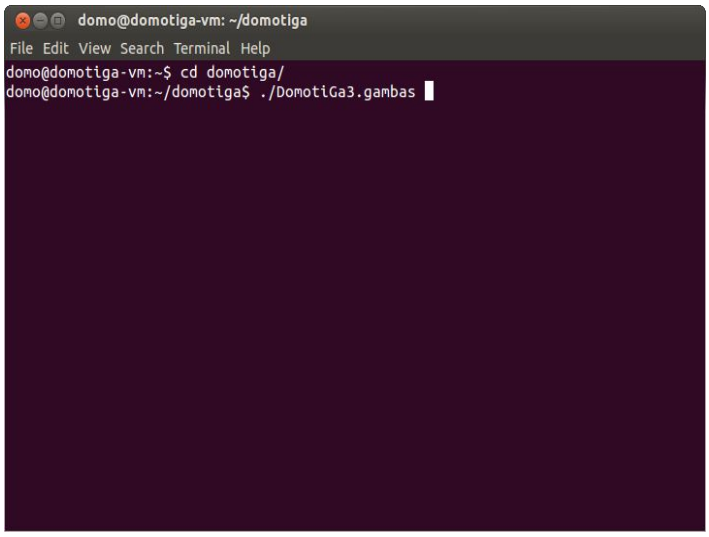
After DomotiGa has been setup, it is still possible to switch to DomotiGaServer and DomotiGa GUI client.

4.1.1 Start DomotiGa

Open a terminal window and change to the DomotiGa directory and start-up the application.

```
$ cd ~/domotiga
$ ./DomotiGa3.gambas
```

The terminal window will be as the following screen:



4.1.2 Database Install Wizard

At first time use the required DomotiGa database in the MySQL server hasn't been setup, DomotiGa will show the following install wizard screen when it detects the database doesn't exist:

4.1.3 Accept License

Before you can continue, you need to accept the GPLv3 license and disclaimer of DomotiGa. If you disagree with the GPLv3 (http://www.gnu.org/licenses/gpl-3.0.html) license and disclaimer, you can still exit DomotiGa by clicking on the "*Cancel*" button. If you agree with the license and disclaimer check the box "`I agree and want to use this program`" and on the "`Next`" button as the following screen shows:

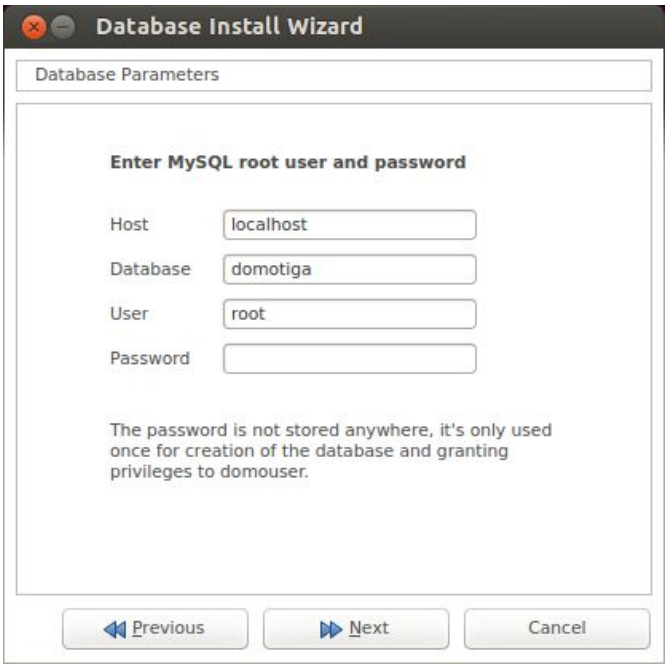
4.1.4 Demo Database

At this moment there isn't a DomotiGa database yet, the Database Install Wizard gives you the option to setup a Demo database or an empty database. For first time user it is recommend (and default option) to setup the Demo database. Then the database will be filled with example devices, events and other data – this will give you a good overview what is possible with DomotiGa. If you want to setup from scratch, click on the "`Empty database`" option and press on "`Next`" button as the following screen shows:

After selected the "Demo database" it is still easy to remove these examples from the database. This is documented further in this document.

4.1.5 Database Parameters

The database isn't setup yet, we only selected to setup the Demo or empty database. The following screen shows the MySQL database parameters. If it is a default installation, you can leave the "`Host`", "`Database`" and "`User`" parameters as shown. Only the "`password`" need to be supplied, this is the password which has been configured in previous section "`2.5 MySQL Server`". Click on the "`Next`" button to execute the real data import.

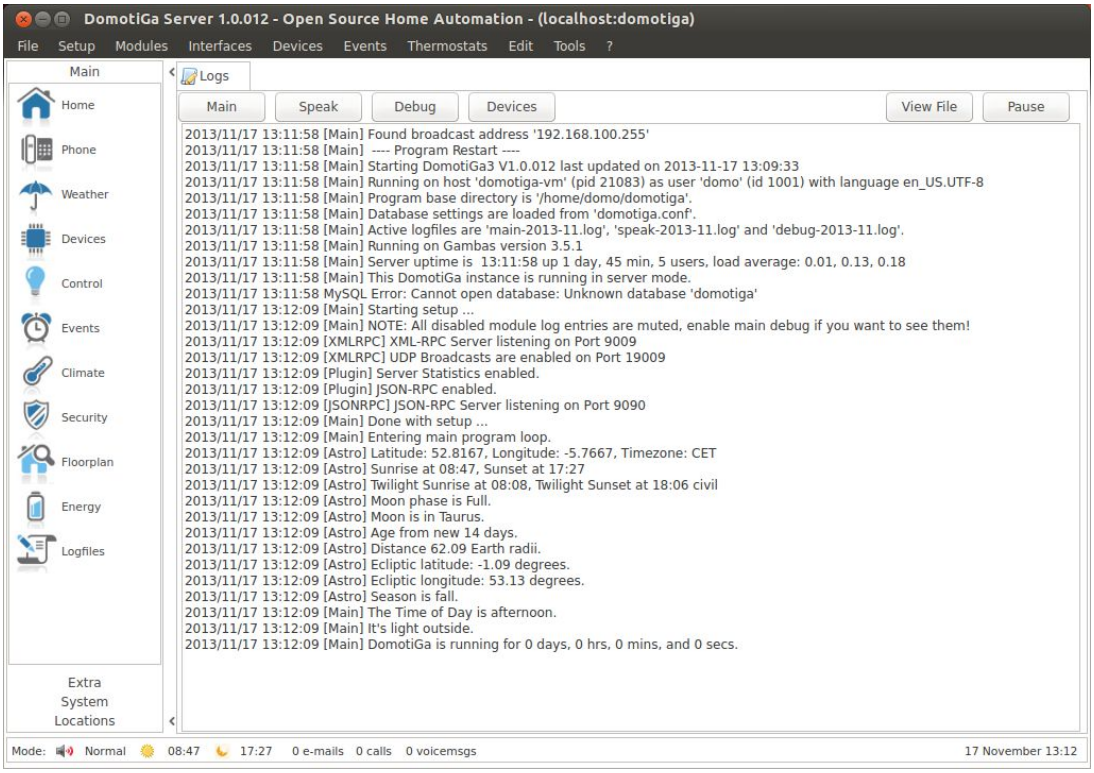


4.1.6 Database Configured

The data import has been executed now. If all went well, the following screen shows the output what you should have. If you typed in a wrong password, you can click on "`Previous`" button and try again. Click on the "`OK`" button to continue.

4.1.7 DomotiGa Started

If all previous steps are successful, DomotiGa will start and show the following screen.



The MySQL error message can be safely ignored, because this always happens during the first time setup.

4.1.8 Next Steps

Now DomotiGa is up-and-running we can continue with the first time setup of DomotiGa, because some localized things like timezone need to be configured. Also interfaces, devices, events, triggers, conditions and actions need to be added to make it a complete domotica system.

Now go to section “4.5 Localized settings“ to continue with the next steps.

4.2 DomotiGaServer

This section describes the first time startup if you want to run DomotiGa backend without a GUI, the so-called DomotiGaServer. This can be an option if your hardware isn't powerful (e.g. Raspberry Pi and other ARM processor based systems) enough or you don't want run a graphical desktop on your server with the DomotiGa backend. The DomotiGa backend can only be configured via the GUI and this needs to be installed as a full client on another Linux system.

If you already have setup DomotiGa in the previous section, you can skip this section.

4.2.1 Create DomotiGa Database

The DomotiGaServer doesn't have a wizard to install the example or empty database, thus this need to be done manually. Execute the following commands to install the example database on the DomotiGa backend:

```
$ mysqladmin -u root -p create domotiga
$ mysql -u root -p domotiga <~/domotiga/install/domotiga.sql
```

If you want to install the empty database, replace "`domotiga.sql`" with "`domotiga-empty.sql`".

4.2.2 Setup MySQL IP Address

The DomotiGa client communicates directly with the DomotiGa backend and MySQL database. By default, MySQL only allows connections from the localhost address, and the DomotiGa client can't connect. This procedure will re-configure MySQL to listen on all the network interfaces. The configuration file is usually found in "`/etc/mysql/my.cnf`". Execute the following command on the DomotiGa backend system to edit the MySQL configuration file:

```
$ sudo nano /etc/mysql/my.cnf
```

Locate the following line in the file:

```
Bind-address            = 127.0.0.1
```

Replace the line with the following content (remove any "`#`" in front of the line):

```
Bind-address            = 0.0.0.0
```

If unfamiliar with the "`nano`" editor, please check the following link for a cheat sheet:
http://www.tuxradar.com/content/text-editing-nano-made-easy

Restart the MySQL Server to use the new configuration with the following command:

```
$ sudo /etc/init.d/mysql restart
```

4.2.3 Setup MySQL Authorisation

By default, MySQL only allows local connections, and the DomotiGa client can't connect from another Linux system in the network.

Start the MySQL command-line tool as follows:

```
$ mysql -u root -p
```

It will ask for the password of the MySQL "`root`" user, this has been configured in previous section "`2.5 MySQL Server`". After the password has been supplied, it will start the editor and you type in the following MySQL commands to give the right authorisation:

```
mysql> GRANT ALL ON *.* TO domouser@'%' IDENTIFIED BY 'kung-fu';
mysql> FLUSH PRIVILEGES;
```

To exit the MySQL command-line tool, just type in the "quit" command as follows:

```
mysql> quit
```

4.2.4 Start DomotiGaServer

Change to the DomotiGa directory and start-up the DomotiGaServer application. The DomotiGaServer can run in a terminal of the Linux system with and without a graphical desktop.

```
$ cd ~/domotiga
$ ./DomotiGaServer3.gambas
```

DomotiGaServer will start and writes the main logging to the standard output (on your screen).

```
2013/11/17 13:34:33 [Main] Found broadcast address '192.168.100.255'
2013/11/17 13:34:33 [Main]  ---- Program Restart ----
2013/11/17 13:34:33 [Main] Starting DomotiGaServer3 V1.0.012 last updated on 2013-11-17 13:12:21
2013/11/17 13:34:33 [Main] Running on host 'domotiga-vm' (pid 21517) as user 'domo' (id 1001) with language en_US.UTF-8
2013/11/17 13:34:33 [Main] Program base directory is '/home/domo/domotiga'.
2013/11/17 13:34:33 [Main] Database settings are loaded from 'server-domotiga.conf'.
2013/11/17 13:34:33 [Main] Active logfiles are 'server-main-2013-11.log', 'server-speak-2013-11.log' and 'server-debug-2013-
11.log'.
2013/11/17 13:34:33 [Main] Running on Gambas version 3.5.1
2013/11/17 13:34:33 [Main] Server uptime is  13:34:33 up 1 day, 1:08, 5 users, load average: 0.00, 0.01, 0.05
2013/11/17 13:34:33 [Main] Checking directory structure ...
2013/11/17 13:34:33 [Main] Connecting to database ...
2013/11/17 13:34:33 [Main] Loading modules ...
2013/11/17 13:34:33 [Main] Starting setup ...
2013/11/17 13:34:33 [Main] NOTE: All disabled module log entries are muted, enable main debug if you want to see them!
2013/11/17 13:34:33 [XMLRPC] XML-RPC Server listening on Port 9009
2013/11/17 13:34:33 [XMLRPC] UDP Broadcasts are enabled on Port 19009
2013/11/17 13:34:33 [Plugin] Server Statistics enabled.
2013/11/17 13:34:33 [Plugin] JSON-RPC enabled.
2013/11/17 13:34:33 [JSONRPC] JSON-RPC Server listening on Port 9090
2013/11/17 13:34:33 [Main] Done with setup …
2013/11/17 13:34:33 [Main] Starting main program …
2013/11/17 13:34:33 [Main] House mode is set to 'Normal'.
2013/11/17 13:34:33 [Main] Mute mode is set to 'Audio Enabled'.
2013/11/17 13:34:33 [Main] Entering main program loop …
2013/11/17 13:34:33 [Astro] Latitude: 52.8167, Longitude: -5.7667, Timezone: CET
2013/11/17 13:34:33 [Astro] Sunrise at 08:47, Sunset at 17:27
2013/11/17 13:34:33 [Astro] Twilight Sunrise at 08:08, Twilight Sunset at 18:06 civil
2013/11/17 13:34:33 [Astro] Moon phase is Full.
2013/11/17 13:34:33 [Astro] Moon is in Taurus.
2013/11/17 13:34:33 [Astro] Age from new 14 days.
2013/11/17 13:34:33 [Astro] Distance 62.09 Earth radii.
2013/11/17 13:34:33 [Astro] Ecliptic latitude: -1.09 degrees.
2013/11/17 13:34:33 [Astro] Ecliptic longitude: 53.13 degrees.
2013/11/17 13:34:33 [Astro] Season is fall.
2013/11/17 13:34:33 [Main] The Time of Day is afternoon.
2013/11/17 13:34:33 [Main] It's light outside.
2013/11/17 13:34:33 [Main] DomotiGa is running for 0 days, 0 hrs, 0 mins, and 0 secs.
2013/11/17 13:34:33 [Main] The current tagline is 'Do well, you hear it never. Do ill, hear it forever.'
```

4.2.5 Start DomotiGa Client

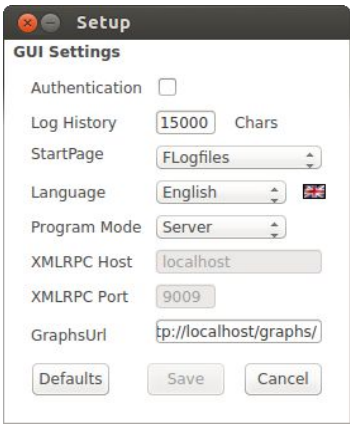
The MySQL and DomotiGaServer steps are executed on the Domotiga backend. Now we are switching to the other Linux system which also has gambas and DomotiGa application installed. Open a terminal window on the graphical desktop and change to the DomotiGa directory and start-up the application.

```
$ cd ~/domotiga
$ ./DomotiGa3.gambas
```

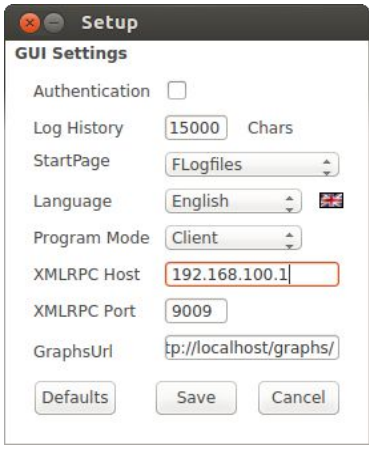
DomotiGa will shown the same screen as seen in section “3.1.7 DomotiGa Started”.

4.2.6 Configure Client/Server mode

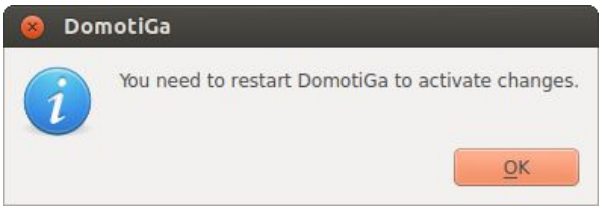
The default mode DomotiGa is running in, is the server mode. To connect to the DomotiGa backend, DomotiGa needs to be reconfigured to run in client mode. Go to the top menu bar and click on “Setup”, this will show a big drop-down list. In this list click on “GUI” and the will give you the following screen:



The program mode is on "`Server`", click on this and select "`Client`" option. After selection of the "`Client`" option the "`XMLRPC Host`" and "`XMLRPC Port`" can be configured. Configure the "`XMLRPC Host`" to the IP addresses of your DomotiGa backend. The "`XMLRPC Port`" can be left as is, unless you changed this on your DomotiGa backend.



After the configuration is done, click on the "`Save`" button. This will show you the message that DomotiGa needs to be restarted to activate the changes, as the following screen shows:



Click on the "`OK`" button to continue, but we aren't restarting DomotiGa at this point yet.

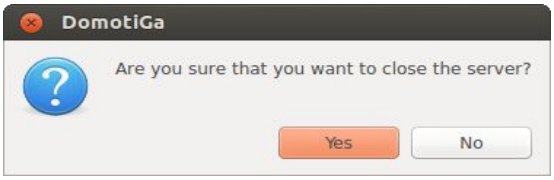
4.2.7 Configure Database

The DomotiGa client also needs to connect remotely to the MySQL database and for that to function, the IP address of the database connection needs to be changed. Go to the top menu bar and click on "`Setup`", this will show a big drop-down list. In this list click on "`Database`" and this will give you the following screen:

Configure the "`Server`" from "`localhost`" to the DomotiGa backend IP address and click on the "`Save`" button.

4.2.8 Restart DomotiGa

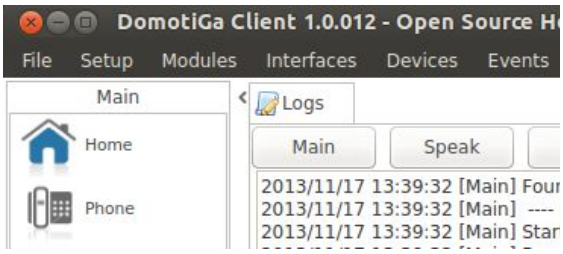
After Domotiga has been set to "`client`" mode and the database reconfigured, we need restart DomotiGa to make the changes active and start it as DomotiGa client. Go to the top menu bar and click on "`File`", this will a short drop-down list. Click on the "Quit" button and then show a big drop-down list. In this list click on "`Database`" and this will give you the following screen:



Click on the "`Yes`" button to exit DomotiGa. Restart the DomotiGa client with the following command:

```
$ ./DomotiGa3.gambas
```

After the DomotiGa client has started, the window should show "`DomotiGa Client`" instead of the earlier "`DomotiGa Server`". This is also shown in the following screen:



4.2.9 Next Steps

Now DomotiGa is up-and-running we can continue with the first time setup of DomotiGa, because some localized things like timezone need to be configured. Also interfaces, devices, events, triggers, conditions and actions need to be added to make it a complete domotica system.

Now go to chapter "4.9" to continue with the next steps.

4.3 Start at Reboot

By default DomotiGa or DomotiGaServer will not automatically start-up when the system is rebooted. This is normally a preferred option for a domotica system. Please note it depends on your system if it supports automatic resume on power faillure.

Automatic resume after a power faillure depends on your used system, please check your manuals if it is supported and how to enable it.

The next sections will describe how to automatically start DomotiGa and DomotiGaserver at reboot, but you should only enable one of them. Depending on your choose backend configuration, choose DomotiGa or DomotiGaServer. The DomotiGa requires a desktop environment, thus it can only start when the desktop has started for an user. The DomotiGaServer doesn't require a desktop, thus we can start it directly at boot-up with the regular init-scripts of Linux.

4.3.1 DomotiGa

TBD

http://askubuntu.com/questions/30931/how-do-i-make-a-program-auto-start-everytime-i-log-in

https://help.ubuntu.com/community/AddingProgramToSessionStartup

http://askubuntu.com/questions/48321/how-do-i-start-applications-automatically-on-login

4.3.2 DomotiGaServer

For the DomotiGaServer a start-up script is supplied for Ubuntu and Debian distributions. These scripts are installed in the "`~/tools/scripts`" directory.

4.3.2.1 Ubuntu / Debian

Ubuntu is based on Debian and the same procedure applies to both to automatically start DomotiGaServer.

Copy the script

First we will copy the script to the "`/etc/init.d`" directory:

```
$ cd ~/domotiga/tools
$ sudo cp domotigaserver3.debian /etc/init.d/domotigaserver3
```

Modify the script

The script needs to be modified, because it contains the username under which it will be started at boot time. Execute the following command to edit the script:

```
$ sudo nano /etc/init.d/domotigaserver3
```

Locate the following line in the file:

```
############################################
USER=domo
############################################
```

In the example the user account is named "`domo`", replace this with your user account to run DomotiGaServer.

Add script to init-scripts

The following command will add the script to the init-scripts of Ubuntu/Debian:

```
$ sudo update-rc.d domotigaserver3 defaults

 Adding system startup for /etc/init.d/domotigaserver3 …
   /etc/rc0.d/K20domotigaserver3 -> ../init.d/domotigaserver3
   /etc/rc1.d/K20domotigaserver3 -> ../init.d/domotigaserver3
   /etc/rc6.d/K20domotigaserver3 -> ../init.d/domotigaserver3
   /etc/rc2.d/S20domotigaserver3 -> ../init.d/domotigaserver3
   /etc/rc3.d/S20domotigaserver3 -> ../init.d/domotigaserver3
   /etc/rc4.d/S20domotigaserver3 -> ../init.d/domotigaserver3
   /etc/rc5.d/S20domotigaserver3 -> ../init.d/domotigaserver3
```

Test script

Before we reboot the system, we want to test if the script really can work. We will first start and check if the process has started with the following commands:

```
$ sudo /etc/init.d/domotigaserver3 start
$ sudo /etc/init.d/domotigaserver3 status
 * DomotiGaServer3.gambas is running
```

Now stop the process again and check if it is really stopped:

```
$ sudo /etc/init.d/domotigaserver3 stop
$ sudo /etc/init.d/domotigaserver3 status
 * DomotiGaServer3.gambas is not running
```

Reboot

Now reboot your system to verify the DomotiGaServer will automatically start after the reboot. After the reboot this can be checked with the following command:

```
$ sudo /etc/init.d/domotigaserver3 status
 * DomotiGaServer3.gambas is running
```

4.4 First Time Configuration

Now DomotiGa is up-and-running we can start with the setup/configuration. The configuration can be overwelling for the first time. The suggested order of configuration is as follows:

- Localized settings
- Modules
- Interfaces
- Devices
- Events

If you started with the Demo database it is highly recommend to play with DomotiGa to get a good sense of the possibilities and options before you start to configure it for your final setup.

4.5 Localized settings

DomotiGa needs some local settings like e.g. latitute/longtitude, timezone, seasons, daylight saving and temperature as input for calculation like sunset/sunrise, display temperature in Celsius or Fahrenheit, etc. Go to section 5.8 Astro and Location to configure this.

4.6 Modules

XML-RPC
etc

4.7 Interfaces

RFXCom
Z-Wave
etc

4.8 Devices

Locaties
floorplan
Groups

4.9 Events

Events
Triggers
Conditions
Actions

5 Setup

Setup Menu

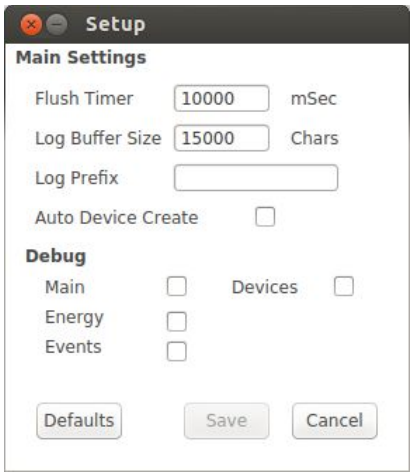
5.1 Overview

TBD

Option	Description
Main	Global settings of DomotiGa
CallerID	TBD
Database	TBD
Dictionary	TBD
GUI	TBD
Home Screen	TBD
Astro and Location	TBD

5.2 Main

Go to the top menu bar and click on "`Setup`", this will show a big drop-down list. In this list click on "`Main`" and this will give you the following screen with the default values:



5.2.1 Flush Timer

The time interval when **all** buffered writes are written to disk. Normally this only affects the logfiles written by DomotiGa and for optimal performance it is recommended to leave it on 10000 milliseconds (10 seconds).

5.2.2 Log Buffer Size

The number of characters shown in the logfile window in the DomotiGa GUI. If the maximum is reached, the logfile window is cleared and starts from the top again. This parameter applies to the "`Main`", "`Speak`" and "`Debug`" logfile windows/tabs.

5.2.3 Log Prefix

The log prefix used for broadcasting messages from the DomotiGa backend to the DomotiGa GUI. You can leave this one empty, because it isn't a very usefull option. This option needs to some redesign.

5.2.4 Auto Create Device

If an unknown device sends a status update to DomotiGa, it can automatically create a device for you. This can be usefull option to (temporary) enable to allow creation of your new devices .

The "`Auto Create Device`" doesn't work yet for all interface and device combinations. Please report your device to support@domotiga.nl and it will be added in the next version.

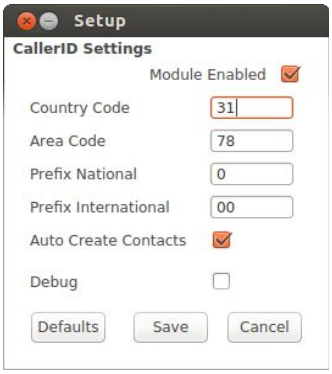
5.2.5 Debug

For troubleshooting purpose DomotiGa has the option to enable debugging. DomotiGa has 2 levels for debugging, on global and module/interface level. The following 4 options are debugging on global level:

- Main – Normally only information about enabled modules/interfaces are written to the logfile. If this is enabled, it will also log information about disabled modules/ interfaces in the main module.
- Energy – Enable logging for devices of the energy type, like P1 and RFXMeter in the energy module.
- Events – Enable logging of events. This is very usefull when troubleshooting when an event is not working as expected in the events module.
- Devices – Enable logging of devices in the devices module. When data comes into a device, it will be first transformed in the right format and send to the common devices module.

5.3 CallerID

TBD



5.3.1 Enabled

Remove from GUI

5.3.2 Country Code

TBD

5.3.3 Area Code

TBD

5.3.4 Prefix National

TBD

5.3.5 Prefix International

TBD

5.3.6 Auto Create Contacts

TBD

5.3.7 Debug

TBD

5.4 Database

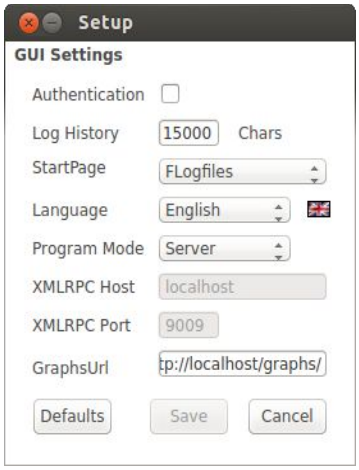
TBD

5.5 Dictionary

TBD

5.6 GUI

Go to the top menu bar and click on "`Setup`", this will show a big drop-down list. In this list click on "`GUI`" and this will give you the following screen with the default values:

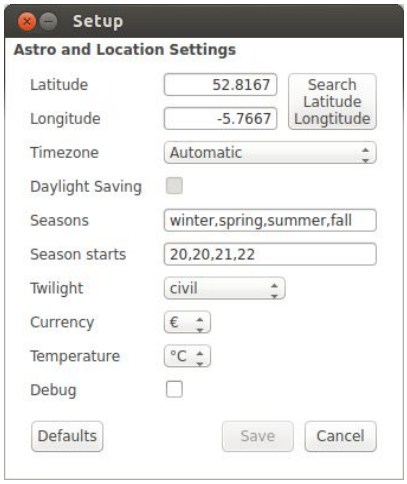


5.7 Home Screen

TBD

5.8 Astro and Location

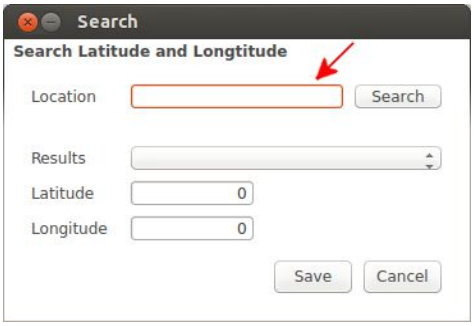
Go to the top menu bar and click on "`Setup`", this will show a big drop-down list. In this list click on "`Astro and Location`" and this will give you the following screen with the default values:



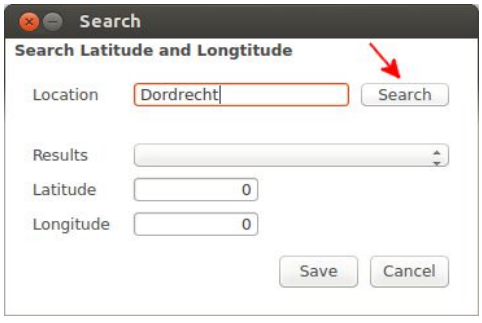
5.8.1 Latitude/Longitude

The latitude and longitude are the coordinates of a location, this should be changed to your residence. They are used to calculate the sunrise/sunset times. These can be used to e.g. turn a light on/off when it gets dark.

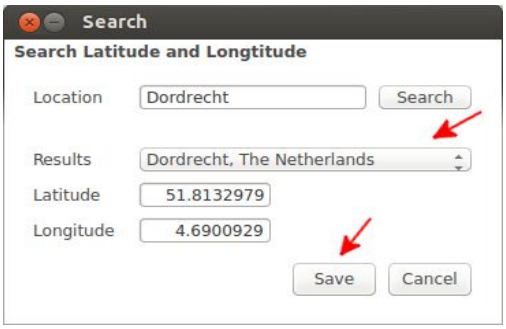
There are 2 ways of finding your latitude and longitude, this can be done via the search option in DomotiGa or manually. To use the DomotiGa option, click on the "`Search Latitude Longtitude`" button. This will show the following search screen and fill in your town/city into the "`Location`" box.



After you typed in your town/city, the following screen will be shown and click on the "`Search`" button.



After you clicked on the "Search" button, it can take a moment to retrieve this information from the internet (the Google API is used). Then the following screen is shown. If nothing can be found, please specify a larger town/city in your neighbourhood. It is possible the search responds with more then 1 result, then you can choose from the drop-down list your choice. After you selected your location, click on "Save" button.

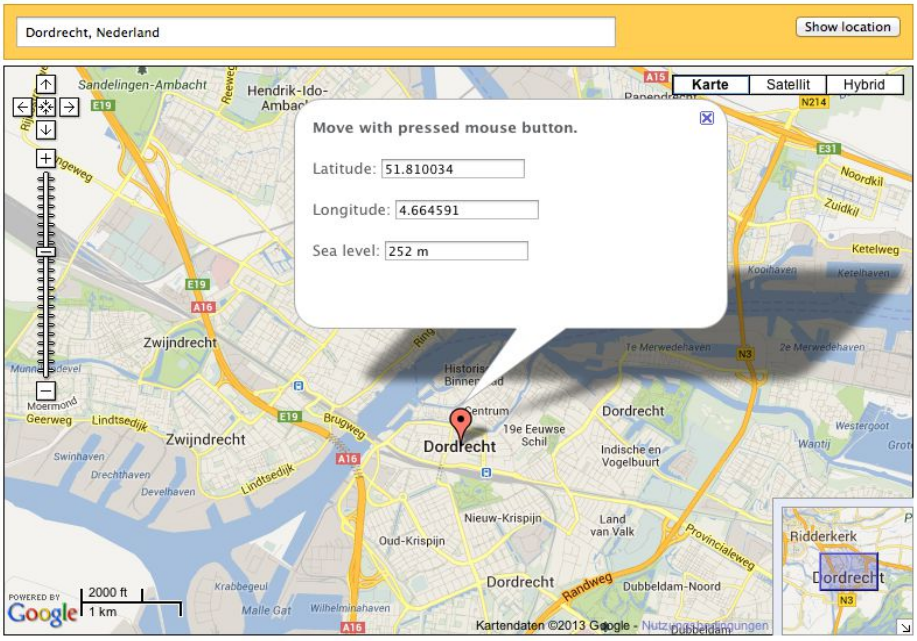


If you clicked on "Save", the latitude and longitude values are used in the previous "Astro and Location" screen. If you clicked on "Cancel", the values aren't overwritten in the previous screen.

To manually configure the latitude and longtitude, use the following link to find your coordinates:

http://www.mapcoordinates.net/en

After opening the link the web page opens and shows (custom) Google Maps. Click on the input field and type in your location, in the screen example below, the location is "Dordrecht, Nederland". To find the coordinates, click on the "Show location" button – this will show the latitude and longitude.



Use the latitude and longitude in the "Astro and Location" screen of DomotiGa. Please note the values can be negative and mind the "dot" character.

5.8.2 Timezone

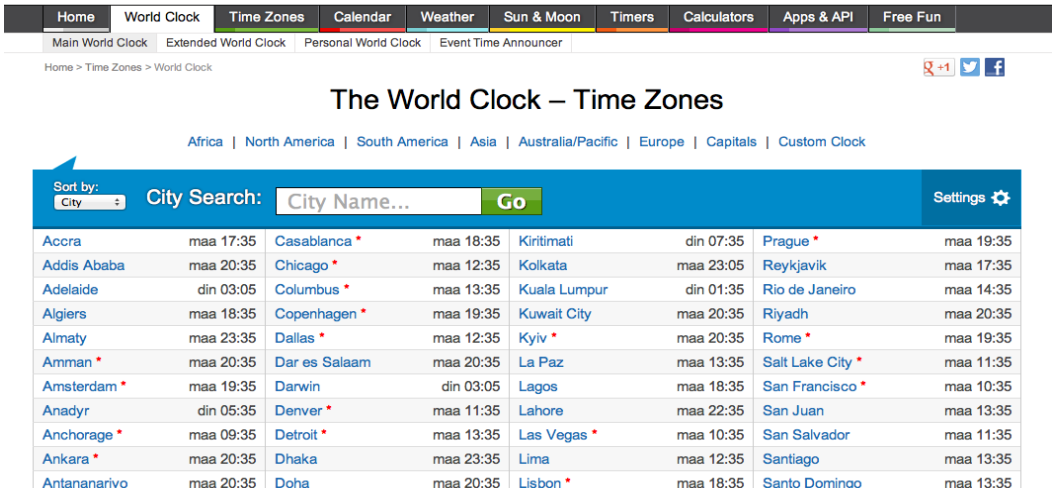
A time zone is a region on Earth that has a uniform standard time for legal, commercial, and social purposes. It is convenient for areas in close commercial or other communication to keep the same time, so time zones tend to follow the boundaries of countries and their subdivisions. Most of the time zones on land are offset from Coordinated Universal Time (UTC) by a whole number of hours (UTC−12 to UTC+14), but a few are offset by 30 or 45 minutes.

Normally you can leave the timezone setting on "`Automatic`" and DomotiGa will retrieve this information from your Operating System. Offset by 30 or 45 minutes is supported in "`Automatic`" mode and not when manually configured.

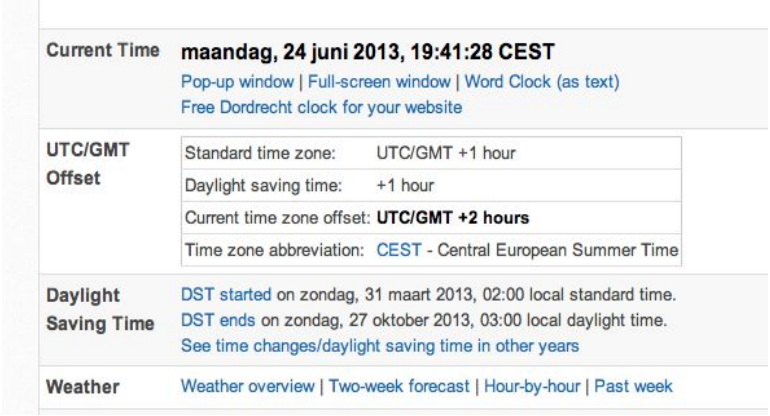
Use the following link to find your timezone, if you aren't familiar with it or want to manually configure it:

http://www.timeanddate.com/worldclock/

After opening the link the web page opens and the example screen is shown below. Click on the input field "`City Name...`" and type in your location. To Find the timezone information, click on the "`Go`" button. If you city isn't found, use the closest largest city.



When you found the city, the following screen shows you the relevant information. In the example below we use the city "`Dordrecht`" and the "`Standard time zone`" is "`UTC/GMT +1 hour`". This "`+1`" is very important, this means that the "`timezone`" value in the configuration panel has to be set to "`1`".



5.8.3 Seasons

In this field the 4 seasons "`winter,spring,summer,fall`" are defined and are used in the variables of DomotiGa. These season names can be customized to your local language.

5.8.4 Season starts

In this field the season start "`20,20,21,22`" days are defined. With the default configuration it will change seasons on the following days of the year:

- Winter to Spring on 20-March
- Spring to Summer on 20-June
- Summer to Fall on 21-September
- Fall to Winter on 22-December

The change of seasons of the current year can be different (plus or minus a day) then the default configuration, this is a known limitation of DomotiGa.

5.8.5 Twilight

You can define twilight simply as the time of day between daylight and darkness, whether that's after sunset, or before sunrise. Astronomers, the experts on nighttime, recognize three kinds of twilight. DomotiGa uses the "`Civil`" twilight as recommended default.

- Civil twilight. It starts as soon as the sun dips below the western horizon. There's enough light to see, but people turn on their lights to drive a car, and the streetlights are starting to come on. Civil twilight officially ends when the sun is 6 degrees below the horizon.
- Nautical twilight. It begins when it's fairly dark outside. By definition, nautical twilight ends when a distant line of a sea horizon stops being visible against the background of the sky – about when the sun is 12 degrees below the horizon. And even then some people still call it twilight.
- Astronomical twilight. It ends when all traces of sky glow are gone. By definition, astronomical twilight ends when the sun is 18 degrees below the horizon. Then astronomers can begin to observe the stars, assuming no clouds are in the way!

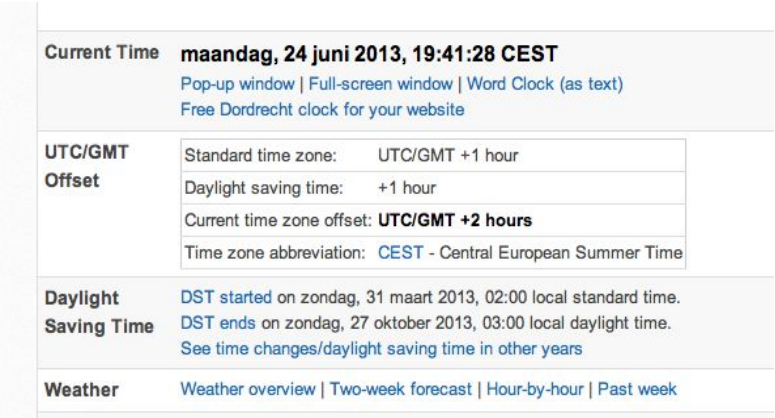
In the following table is an example shown of the possible twilight start and end times for the 3 different settings:

Civil		Nautical		Astronomical	
Start	End	Start	End	Start	End
06:30	16:59	05:56	17:33	05:23	18:06

Use the following link if you are interesting in more information about twilight:
http://en.wikipedia.org/wiki/Twilight

5.8.6 Daylight Saving

Some higher latitude countries use daylight saving time for part of the year, typically by changing clocks by an hour. Many land time zones are skewed toward the west of the corresponding nautical time zones. This also creates a permanent daylight saving time effect. The daylight saving is also called summertime/wintertime in some countries. The following screen shows that for this location the “Daylight saving time” is “+1 hour”, this means they have daylight saving at that moment.



The daylight saving option needs to manually be enabled/disabled every time when it happens on your location.

5.8.7 Currency

The currency to be used in DomotiGa. The choices are € (EURO) and $ (US DOLLAR).

5.8.8 Temperature

The temperature to be used in DomotiGa. The choices are °C (Celcius) and °F (Fahrenheit).

5.8.9 Debug

Enable debugging of “`Astro and Location`” feature in case of issues. The output will be written to the debug logfile.

6 Modules

This chapter describes the available modules in DomotiGa. A module is the part of DomotiGa which can be enabled to do a specific action. This can be e.g. XML-RPC, Web or Telnet service to retrieve information from DomotiGa, but also send e.g. emails, play sounds or twitter depending on a certain action.

6.1 Overview

The following table gives you an overview of the available modules, with a brief description:

Module	Description
Email	Send email messages via SMTP
Google | GMail	TBD
Publish Data | Bwired Map	
Publish Data | MQTT	
Published Data | Pachube	
Published Data | PVoutput	
Published Data | TemperaturNu	
Notifiers | Notify My Android	
Notifiers | Prowl	
Notifiers | Pushover	
P2000 Scanner	
RRDTool	Tool to create graphs
Servers | Telnet	Legacy MisterHouse support
Servers | SmartVISU	
Servers | XML-RPC	Interface for DomotiGa and Web Clients
Server Stats	
Sound	
Thermostat	Internal thermostat
TV Guide	
Twitter	Post twitter messages (tweets)
VoiceText	
Weather | WeatherBug	

Each module has the following 2 configuration options:

- Module Enabled – If the module is enabled or disabled
- Debug – Enable or disable debugging of this module

6.2 Email

Output module for sending email messages, triggered by an action. The email messages are send by the commonly used SMTP protocol. The SMTP server, SMTP port, from and to addresses have to be configured. On this moment SMTP authentication and/or SSL aren't supported yet.

6.3 Google / GMail

Input module for Google's GMail to checks your Google email account for new email(s) and displays the counter on your home screen.

6.4 Twitter

Output module for posting tweets on twitter, triggered by an action. A valid user and password have to be configured.

6.5 VoiceText

Output module for Text-to-Speech (TTS) functionality. This module makes converts text to speech and plays it via the speakers. To use Text-to-Speech an TTS engine needs to be installed, the following 3 are supported:

- Cepstral
- Espeak
- Picotts

For detailed installation instructions, please visit the Wiki:

http://www.domotiga.nl/projects/domotiga/wiki/VoiceText

6.6 Telnet Server

A telnet alike input module to send very limit commands to. This is legacy support for the MisterHouse "`mhsend`" command.

6.7 Weather

Input module for weather information, this can be displayed on the home screen. Currently only the "`WeatherBug`" service is supported. To use the "`WeatherBug`" you require a WeatherBug ID (also called API license key), use the following steps to request it:

- Open the following link:
  http://weather.weatherbug.com/desktop-weather/api-register.html
- Register yourself and you will receive an email for registration validation. Click on the link in this email and you will get your API license key.

You need to wait 30 minutes after registration, before you can use it (this is also stated on page of the link. From experience we noticed that it can take longer, please try again after 1 hour if the first attempt fails.

After you have the WeatherBug ID, you need execute the following steps to use it:

- Enter your ID into the "`WeatherBug ID`" field
- Go to the field "`City`" and type in your city
- Click on "`Search`" button
- The city/cities found are shown in the "`Result`" drop-down box. It is possible then more then one city is field and you need to select the right city
- After you made a selection, the "`City`" and "`City Code`" will shown the WeatherBug value and will be used when you view the weather in the Home screen.

6.8 Server Stats

Module which will create RRD graphs of the CPU load and memory usage of your Linux server. These graphs are available in the home screen under "`System`" => "`Statistics`". The RRDTool needs to be installed and enabled to use this module.

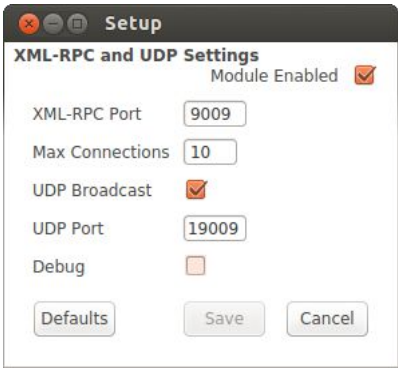
6.9 TV Guide

TBD – Need a HowTo!

6.10 XML-RPC

Input module for external applications like Web and DomotiGa Clients can do procedure call on the DomotiGa backend, to push and pull information. It is using HTTP as transport protocol and XML as the encoding.This option is enabled by default and has to be enabled to use clients.

Go to the top menu bar and click on "`Modules`", this will show a drop-down list. In this list click on "`XML-RPC`" and this will give you the following screen with the default values:



6.11 Publish Data

TBD

6.12 Thermostat

Module which enables the built-in thermostat functionality.

6.13 RRDTool

Module which enables the Round Robin Database Tool (RRDTool) functionality. RRDTool is the open source industry standard, high performance data logging and graphing system for

time series data. This option has to be enabled to allow DomotiGa to create graphs of the collected data.

7 Interfaces

An interface is a special type module and can be assigned to one or more DomotiGa device(s). The interface module will connect via e.g. serial or network externally and will process data, the data can be pushed and/or pulled and stored in devices. An example of an interface is Z-Wave, which connects the Aeon Labs Z-Wave USB controller and can communicate with Z-Wave switches, sensors, etc.

7.1 Overview

The following table gives an overview of the available interfaces, with a brief description:

Interface	Description
1-Wire	
Anel PwrCtrl	
Audio Video	Submenu
...	
...	
...	
...	

7.2 Serial USB Device Naming

Most of the interfaces you can connect to DomotiGa are either serial USB devices, serial RS232 devices or through the IP network. The serial USB devices are mostly used, because e.g. RFXCom, Z-Wave, ZigBee and PlugWise use them. If you connect a serial USB device to your Linux system it will be a device name like "`/dev/ttyUSB0`", when you connect multiple devices it will continue numbering and it can be "`/dev/ttyUSB1`", "`/dev/ttyUSB2`", "`/dev/ttyUSB9`", etc. This numbering is done by Linux and is somewhat confusing to remember which-device-is-connected-where. Only it gets even worse if you remove one of your devices and the number can change after reboot, and you need to reconfigure DomotiGa to get it working again.

Luckily Linux gives you with udev rules the ability to create a symlink (alias) to a serial USB device, which overcomes (most) of the previous problem. There are 3 ways of doing it, both have their pro-and-con and they are the following ones:

- Symlink based on USB port location
- Symlink based on USB device serial number
- Manual symlink

Before you start with any of the options, you need to determine to which "`/dev/ttyUSB*`" it will be currently connected. We will use that device to retrieve the information to configure the udev rules. Just plug in you serial USB device and afterwards type in "`dmesg`" and that will give you the following output (the last lines are the relevant ones):

```
$ dmesg
…
[  498.955742] usb 2-2.2: new full-speed USB device number 5 using uhci_hcd
[  499.125902] usb 2-2.2: New USB device found, idVendor=0403, idProduct=6001
[  499.125907] usb 2-2.2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[  499.125911] usb 2-2.2: Product: USB Serial Converter
[  499.125914] usb 2-2.2: Manufacturer: FTDI
[  499.125916] usb 2-2.2: SerialNumber: FTFVMOV5
[  499.187080] usbcore: registered new interface driver usbserial
[  499.187775] usbcore: registered new interface driver usbserial_generic
[  499.188395] usbserial: USB Serial support registered for generic
[  499.194816] usbcore: registered new interface driver ftdi_sio
[  499.195651] usbserial: USB Serial support registered for FTDI USB Serial Device
[  499.196206] ftdi_sio 2-2.2:1.0: FTDI USB Serial Device converter detected
[  499.196265] usb 2-2.2: Detected FT232RL
[  499.196267] usb 2-2.2: Number of endpoints 2
[  499.196269] usb 2-2.2: Endpoint 1 MaxPacketSize 64
[  499.196270] usb 2-2.2: Endpoint 2 MaxPacketSize 64
[  499.196271] usb 2-2.2: Setting MaxPacketSize 64
[  499.199230] usb 2-2.2: FTDI USB Serial Device converter now attached to ttyUSB0
```

Notice that this specific serial USB device got the name "`ttyUSB0`", that will be used in the examples below.

Do **not** run the "`udevadm`" command on a Raspberry Pi, this will cause a kernel panic and require a hard reboot (remove power). Use the "`usb-devices`" command to get the same information on a Raspberry Pi.

7.2.1 Symlink based on USB port location

To make a udev rule to create the symlink, we first need find the required port information. This can be done with the "`udevadm`" command as shown in the example below. In this example the port information is '`KERNELS=="2-2.2:1.0"`' and is used in the next step.

```
$ udevadm info --attribute-walk -p /class/tty/ttyUSB0
…
  looking at device '/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-2/2-
2.2/2-2.2:1.0/ttyUSB0/tty/ttyUSB0':
    KERNEL=="ttyUSB0"
    SUBSYSTEM=="tty"
    DRIVER==""

  looking at parent device '/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-
2/2-2.2/2-2.2:1.0/ttyUSB0':
    KERNELS=="ttyUSB0"
    SUBSYSTEMS=="usb-serial"
    DRIVERS=="ftdi_sio"
    ATTRS{port_number}=="0"
    ATTRS{latency_timer}=="1"

  looking at parent device '/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-
2/2-2.2/2-2.2:1.0':
    KERNELS=="2-2.2:1.0"
    SUBSYSTEMS=="usb"
    DRIVERS=="ftdi_sio"
    ATTRS{bInterfaceClass}=="ff"
    ATTRS{bInterfaceSubClass}=="ff"
    ATTRS{bInterfaceProtocol}=="ff"
    ATTRS{bNumEndpoints}=="02"
    ATTRS{supports_autosuspend}=="1"
    ATTRS{bAlternateSetting}==" 0"
    ATTRS{bInterfaceNumber}=="00"
    ATTRS{interface}=="USB Serial Converter"
…
```

When you determined your "KERNELS==" parameter, you can create or modify the udev rule. For DomotiGa we use the filename "/etc/udev/rules.d/100-usbadapter.rules", so execute the following command:

```
$ sudo nano /etc/udev/rules.d/100-usbadapter.rules
```

Add the following line to the file, replace the "2-2.2:1.0" value with the your own and also the "ttyUSBzwave" with the name you want to give your device.

```
SUBSYSTEMS=="usb",KERNELS=="2-2.2:1.0",NAME="ttyUSBzwave"
```

Repeat the previous steps for all you serial USB devices.

There are 2 ways of making your new configuration active, the first is to restart udev with "/etc/init.d/udev restart" and to unplug and replug all your devices or reboot your system. The following command will do it easy – reboot your system:

```
$ sudo reboot
```

After we have rebooted, check if all devices are as expected:

```
$ ls -l /dev/ttyUSB*
crw------- 1 root dialout 188, 0 jul 13 10:18 /dev/ttyUSB0
crw------- 1 root dialout 188, 1 jul 13 10:33 /dev/ttyUSB1
crw------- 1 root dialout 188, 2 jul 13 10:16 /dev/ttyUSB2
crw------- 1 root dialout 188, 3 jul 13 10:13 /dev/ttyUSB3
crw------- 1 root dialout 188, 4 jul 13 10:23 /dev/ttyUSB4
crw------- 1 root dialout 188, 5 jul 13 10:23 /dev/ttyUSB5
lrwxrwxrwx 1 root root         1 jul 13 10:33 /dev/ttyUSBdigitemp
lrwxrwxrwx 1 root root         5 jul 13 10:30 /dev/ttyUSBled
lrwxrwxrwx 1 root root         0 jul 13 10:18 /dev/ttyUSBonkyo
lrwxrwxrwx 1 root root         2 jul 13 10:31 /dev/ttyUSBplugwise
lrwxrwxrwx 1 root root         4 jul 13 10:23 /dev/ttyUSBspare
lrwxrwxrwx 1 root root         3 jul 13 10:31 /dev/ttyUSBzwave
```

Now you can use the e.g. "/dev/ttyUSBzwave" symbolic name in DomotiGa and that will not change between reboots. Please don't move your serial USB devices to different ports, else you need to repeat the above steps again.

7.2.2 Symlink based on USB device serial number

To make a udev rule to create the symlink, we first need find the required port information. This can be also done with the "udevadm" command as shown in the example below. In the example below the vendor identifier, product identifier and serial number are explicity shown. Please note that these 3 values aren't in a specific order, check the whole output to find them.

Some chip vendors use the same vendor identifier, product identifier and serial number for all their devices. In such can you can't use this method anymore, because udev will find 2 devices with the same information. The following devices/adapters are known to have that defect:

- Adapters with a Prolific PL2303 chip

```
$ udevadm info -a -n /dev/ttyUSB0
…
   looking at device '/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-2/2-
2.2/2-2.2:1.0/ttyUSB0/tty/ttyUSB0':
    KERNEL=="ttyUSB0"
    SUBSYSTEM=="tty"
    DRIVER==""

  looking at parent device '/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-
2/2-2.2/2-2.2:1.0/ttyUSB0':
    KERNELS=="ttyUSB0"
    SUBSYSTEMS=="usb-serial"
    DRIVERS=="ftdi_sio"
    ATTRS{port_number}=="0"
    ATTRS{latency_timer}=="1"

  looking at parent device '/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-
2/2-2.2/2-2.2:1.0':
    KERNELS=="2-2.2:1.0"
    SUBSYSTEMS=="usb"
    DRIVERS=="ftdi_sio"
    ATTRS{bInterfaceClass}=="ff"
    ATTRS{bInterfaceSubClass}=="ff"
    ATTRS{bInterfaceProtocol}=="ff"
    ATTRS{bNumEndpoints}=="02"
    ATTRS{supports_autosuspend}=="1"
    ATTRS{bAlternateSetting}==" 0"
    ATTRS{bInterfaceNumber}=="00"
    ATTRS{interface}=="USB Serial Converter"

  looking at parent device '/devices/pci0000:00/0000:00:11.0/0000:02:00.0/usb2/2-
2/2-2.2':
    KERNELS=="2-2.2"
    SUBSYSTEMS=="usb"
    DRIVERS=="usb"
    ATTRS{bDeviceSubClass}=="00"
    ATTRS{bDeviceProtocol}=="00"
    ATTRS{devpath}=="2.2"
    ATTRS{idVendor}=="0403"                    <= This line is the Vendor ID
    ATTRS{speed}=="12"
    ATTRS{bNumInterfaces}==" 1"
    ATTRS{bConfigurationValue}=="1"
    ATTRS{bMaxPacketSize0}=="8"
    ATTRS{busnum}=="2"
    ATTRS{devnum}=="5"
    ATTRS{configuration}==""
    ATTRS{bMaxPower}=="44mA"
    ATTRS{authorized}=="1"
    ATTRS{bmAttributes}=="a0"
    ATTRS{bNumConfigurations}=="1"
    ATTRS{maxchild}=="0"
    ATTRS{bcdDevice}=="0600"
    ATTRS{avoid_reset_quirk}=="0"
    ATTRS{quirks}=="0x0"
    ATTRS{serial}=="FTFVMOV5"                  <= This line is the serial number
    ATTRS{version}==" 2.00"
    ATTRS{urbnum}=="16"
    ATTRS{ltm_capable}=="no"
    ATTRS{manufacturer}=="FTDI"
    ATTRS{removable}=="unknown"
    ATTRS{idProduct}=="6001"                   <= This line is the Product ID
    ATTRS{bDeviceClass}=="00"
    ATTRS{product}=="USB Serial Converter"
…
```

When you determined your vendor identifier, product identifier and serial number, you can create or modify the udev rule. For DomotiGa we use the filename "`/etc/udev/rules.d/100-usbadapter.rules`", so execute the following command:

```
$ sudo nano /etc/udev/rules.d/100-usbadapter.rules
```

Add the following line to the file, 3 values with the your own and also the "ttyUSBzwave" with the name you want to give your device.

```
SUBSYSTEM=="usb", ATTRS{idVendor}=="0403", ATTRS{idProduct}=="6001",
ATTRS{serial}=="FTFVMOV5", SYMLINK+="ttyUSBzwave"
```

Repeat the previous steps for all you serial USB devices.

There are 2 ways of making your new configuration active, the first is to restart udev with "/etc/init.d/udev restart" and to unplug and replug all your devices or reboot your system. The following command will do it easy – reboot your system:

```
$ sudo reboot
```

After we have rebooted, check if all devices are as expected:

```
$ ls -l /dev/ttyUSB*

crw------- 1 root root    188, 0 jul 13 10:18 /dev/ttyUSB0
crw------- 1 root root    188, 1 jul 13 10:33 /dev/ttyUSB1
crw------- 1 root root    188, 2 jul 13 10:16 /dev/ttyUSB2
crw------- 1 root root    188, 3 jul 13 10:13 /dev/ttyUSB3
crw------- 1 root root    188, 4 jul 13 10:23 /dev/ttyUSB4
crw------- 1 root root    188, 5 jul 13 10:23 /dev/ttyUSB5
crw-rw---- 1 root dialout 188, 1 jul 13 10:33 /dev/ttyUSBdigitemp
crw-rw---- 1 root dialout 188, 5 jul 13 10:30 /dev/ttyUSBled
crw-rw---- 1 root dialout 188, 0 jul 13 10:18 /dev/ttyUSBonkyo
crw-rw---- 1 root dialout 188, 2 jul 13 10:31 /dev/ttyUSBplugwise
crw-rw---- 1 root dialout 188, 4 jul 13 10:23 /dev/ttyUSBspare
crw-rw---- 1 root dialout 188, 3 jul 13 10:31 /dev/ttyUSBzwave
```

Now you can use the e.g. "/dev/ttyUSBzwave" symbolic name in DomotiGa and that will not change between reboots.

7.2.3 Manual Symlink

The third option is to create a manual symlink to the device in the "/dev/serial/by-id" directory. After you plugged into your device, check this directory for the unique device name as follows:

```
$ ls -l /dev/serial/by-id/
total 0
lrwxrwxrwx 1 root root 13 jul 13 19:55 usb-FTDI_USB_Serial_Converter_FTFVMOV5-if00-
port0 -> ../../ttyUSB0
```

The output shows you the unique device name and also the "ttyUSB0" name you found earlier when you pulled it in. Create a symlink to this one as follows:

```
$ sudo ln -s /dev/serial/by-id/usb-FTDI_USB_Serial_Converter_FTFVMOV5-if00-port0
/dev/ttyUSBzwave
```

Repeat the previous steps for all you serial USB devices.

7.3 Interfaces

1-Wire

Digitemp

Midon TEMP08

OWFS

OWW

- Anel PwrCtrl
- Audio Video
  - Denon AV
  - iPort Dock
  - LG TV
  - Onkyo/Integra AV
  - Sharp TV
  - Squeeze Server
  - Pioneer AV
  - XBMC xPL
- Bluetooth
- CallerID
  - Asterisk
  - Fritz!Box
  - Ncid
- Cameras
  - Cameras
  - IP Videoserver
  - Sony Visca
- CUL
- DSC Security Panel
- Visonic Security Panel
- Energy Sensors
  - Current Cost
  - SmartMeter
  - YouLess
- EZcontrol
- ELV MAX!
- HDDTemp
- HomeMatic
- Input/Output
  - Domotica
  - Velleman K8055
  - Weeder I/O
- Remote Control
  - CF iViewer
  - UIR/IRMan
  - IRTrans

- LIRC
- JeeLabs
- KNX/EIB
- LED Matrix
- Meteohub
- Network Ping
- NMEA GPS
- OpenTherm
- PLCBUS
- Plugwise
- RFXCom
  - RFXCom Receiver LAN/USB
  - RFXCom Transmitter LAN/USB
  - RFXCom Transceiver RFXTrx
  - RFXCom xPL
- Shell
- SMS Modem
- UPS Monitor
- X10
  - X10Cmd
  - Xanura CTX35
  - Mochad
- xPL
- Z-Wave

8 Devices

Devices

8.1 Introduction

Add Device

List Devices

Modules, Groups …

- Modules
- Groups
- Locations
- Floorplans
- Blacklist

9 Events

Interfaces

9.1 Introduction

Edit Macro

Add Event

Edit Categories

List Events

Add Trigger

Add Condition

Add Action

10 Upgrade

This chapter describes the optional steps to upgrade Gambas and/or DomotiGa.

10.1 Gambas

Upgrading Gambas is normally a manual action. The TRUNK release is regulary updated fixes and improvements and is recommended to be used/updated if you run into issues with DomotiGa. Following the next steps to upgrade (=re-install) Gambas on your system.

If running Gambas with the DomotiGaServer backend and DomotiGa client, both systems need to be upgraded.

10.1.1 Stop DomotiGa/DomotiGaServer

All applications running Gambas need to be stopped, before Gambas is upgraded.

10.1.2 Remove the installed Gambas files

With most software it isn't needed to remove the current installed released first, but it is possible that some locations of binaries or libraries have changed. So first we will first remove Gambas completely from the system with the following commands:

```
$ sudo /bin/bash
# rm -f /usr/local/bin/gbx3 /usr/local/bin/gbc3
# rm -f /usr/local/bin/gba3 /usr/local/bin/gbi3
# rm -rf /usr/local/lib/gambas3
# rm -rf /usr/local/share/gambas3
# rm -f /usr/local/bin/gambas3
# rm -f /usr/bin/gambas3
# rm -f /usr/local/bin/gambas3.gambas
# rm -f /usr/bin/gbx3 /usr/bin/gbc3 /usr/bin/gba3 /usr/bin/gbi3
# rm -rf /usr/lib/gambas3
# rm -rf /usr/share/gambas3
# rm -f /usr/bin/gambas3
# rm -f /usr/bin/gambas3.gambas
# rm -f /usr/bin/gbs3.gambas
# exit
```

10.1.3 Backup Gambas

First we will move the locally download old Gambas source to another directory follows:

```
$ cd ~/install
$ mv gambas gambas.bak
```

10.1.4 Upgrade Gambas

The upgrade of Gambas is really a reinstallation of Gambas. Please repeat the following steps (section):

- 3.5 Gambas

10.2 DomotiGa

Regularly fixes and features are added to DomotiGa and is recommended to upgrade if you have issues or want the latest features. Upgrading DomotiGa is easy to do with a few simple commands, as described below.

10.2.1 Stop DomotiGa/DomotiGaServer

All running DomotiGa applications need to be stopped, before it can be upgraded.

10.2.2 Make Backup

First we will make a backup of DomotiGa with the following command:

```
$ cp -rp ~/domotiga ~/domotiga.<date>
```

Next we will make a backup of the MySQL database. This can be done with the following command:

```
$ mysqldump -u root -p domotiga >~/domotiga.backup/domotiga-backup.sql
```

10.2.3 Update DomotiGa

DomotiGa can easily be updated with the following commands, it will retrieve the changed files from the DomotiGa SVN.

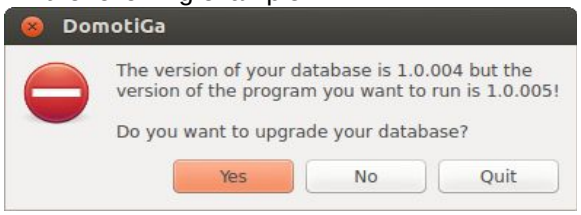
```
$ rm -rf ~/install/domotiga
$ cd ~/install
$ svn checkout http://svn.domotiga.nl/domotiga/trunk/ domotiga
$ cp -rp ~/install/domotiga ~
```

10.2.4 Start DomotiGa

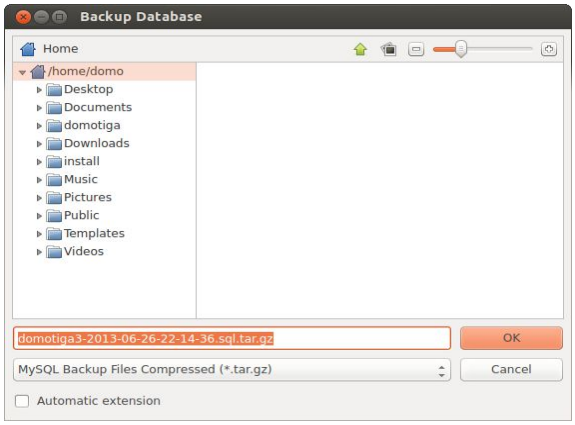
Restart the DomotiGa with the following command:

```
$ ./DomotiGa3.gambas
```

If the database structure has been updated, the DomotiGa GUI will prompt you with the screen as shown in the following example.

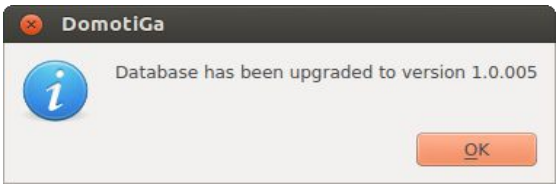


Before it applies the changes to the database, DomotiGa GUI will ask if it should a backup of the existing database, as shown in the following screen. With this backup you can rollback if you encounter issues.



After the backup is successfully created, it shows the folllowing screen.

After clicking on "OK" it will apply the changes to the database and will shown you a screen as in the example below.



DomotiGa will upgrade from you previous version to the latest version. If upgrading across multiple versions e.g. `1.0.004 => 1.0.005 => 1.0.006` then DomotiGa will prompt you multiple times to upgrade.

11 Troubleshooting

This chapter describes common errors and their solution.

11.1 <Device> Error: Cannot open serial port (#5)

If DomotiGa is used with USB serial devices it is possible it can't access the serial device and you get e.g. "`RFXCom transceiver Error: Cannot open serial port (#5)`" error message in the main logfile. This can be for 2 reasons, serial port doesn't exist or the permissions aren't correct. First check if they configured serial device is available on the system:

```
$ ls -l /dev/ttyUSB*
crw-rw---- 1 root dialout 188, 0 jun 30 21:57 /dev/ttyUSB0
```

This example shows a device with "`ttyUSB0`" exists, depending on the number of devices it can be "`ttyUSB1`", "`ttyUSB2`", etc.

If the serial device exists on the system, possible the "`dialout`" group isn't added to your user account. This is required to allow your user access to the USB serial devices. The following command will add the "`dialout`" group to your user account to give the right permissions, in the example the user account is named "`domo`":

```
$ sudo addgroup domo dialout
```

After you added the "`dialout`" group to your user account, you **need** to logout and login again to activate this change.

11.2 Lost connection to MySQL server

If you see the following "`Error parsing <device> query failed: Lost connection to MySQL server during query at Devices.ValueUpdate`" error message, it is possible your MySQL persistent connection time-out is set too low. Normally DomotiGa generate enough database queries and updates to the connection open, but it is recommended to increase the connection time-out to a higher value. The MySQL configuration file is usually found in "`/etc/mysql/my.cnf`". Execute the following command on the DomotiGa backend system to edit the MySQL configuration file:

```
$ sudo nano /etc/mysql/my.cnf
```

Check if the following line already exists, else add it to the file:

```
wait_timeout            = 2147483
```

Restart the MySQL Server to use the new configuration with the following command:

```
$ sudo /etc/init.d/mysql restart
```

Version History

Version	Status	Date	Details of Changes	Author(s)
0.01	DRAFT	28-Jun-2013	Initial version	Alexie
0.02	DRAFT	13-Jul-2013	Major update	Alexie
0.03	DRAFT	28-Jul-2013	Updated Gambas to version 3.4.2	Alexie
0.04	DRAFT	17-Sep-2013	Included Ubuntu 13.10, updated screenshots	Alexie
0.05	DRAFT	18-Nov-2013	Updated to Gambas 3.5.1 and DomotiGa 1.0.012	Alexie
0.06	DRAFT	19-Jan-2014	Updated to Gambas 3.5.2 and correct wrong start scripts	Alexie