

# Urchin Software from Google: Install Guide for FreeBSD/ Linux

Software Version: Urchin 6.5

Revision: December 12th, 2008

## Before You Start - Important Note for FreeBSD Users

FreeBSD has a hard-coded process datasize limit of 500MB, which is too small to allow Urchin to run due to the size of the geodata that has to be memory resident during log processing. The error you will see during runtime is:

```
ERROR: (8011-323-1057) Failed to allocate memory
```

To address this, you'll need to increase the FreeBSD system's default process datasize, which you can do by editing the `/boot/loader.conf` file and adding these lines:

```
# Increase max process data segment size to 1GB
kern.maxdsiz="1073741824"
```

You'll need to reboot afterwards.

**WARNING!!!** Be very careful when changing system limits like this. If you set `kern.maxdsiz` to be too large, your system may become unbootable. We **strongly** recommend that you read the FreeBSD documentation before making this modification, and assess the potential risks as they apply to your site-specific configuration.

## Overview of Installation Steps

- Install the desired database server on the Urchin system or add Urchin's database to an existing server. Currently supported database engines:
  - MySQL 4.1.22 or later
  - PostgreSQL 8.2.5 or later
- Create the Urchin database and user on the database server and grant access permissions
- Unpack the Urchin distribution in a temporary directory
- Run the `install.sh` script
- License the Urchin installation

## Option 1: Using MySQL with Urchin

- If you don't already have a MySQL server set up on your system, you can obtain a distribution from <http://dev.mysql.com/downloads/>
- Install the MySQL server per the instructions on the MySQL site
- Make sure that you know the MySQL database administrator login (e.g. `root`) and password

- Choose the database name, local MySQL user and password you will use with Urchin. Save this information, as you'll need it later during the Urchin installation. The example below creates a database named `urchin` and a MySQL user name `urchin` with password `urchinpassword` (but you should obviously choose your own password!)
- Create the Urchin database, user and permissions with the following commands:

```
root@yourserver# mysql -u root -p
mysql> CREATE DATABASE urchin CHARACTER SET UTF8;
mysql> GRANT ALL ON urchin.* TO 'urchin'@'localhost' IDENTIFIED BY
'urchinpassword';
```

- Determine the pathname of the MySQL socket for communicating with your MySQL server. You can usually find this in the `my.cnf` file for your MySQL installation, look for the line that looks like `socket=/var/lib/mysql/mysql.sock`

## Option 2: Using PostgreSQL with Urchin

- If you don't already have a PostgreSQL server set up on your system, you can obtain a distribution from <http://www.postgresql.org>
- Install the PostgreSQL server per the instructions on the PostgreSQL site
- Make sure that the `psql` command is in your path and you know the PostgreSQL database administrator login (e.g. `pgsql`) and password
- Choose the database name, local PostgreSQL user and password you will use with Urchin. Save this information, as you'll need it later during the Urchin installation. The example below creates a database named `urchin` and a PostgreSQL user name `urchin` with password `urchinpassword` (but you should obviously choose your own password!)
- Create the Urchin database, user and permissions with the following commands:

```
root@yourserver# psql -U pgsql -d postgres
postgres=# CREATE DATABASE urchin WITH ENCODING 'UTF8';
postgres=# CREATE USER urchin WITH PASSWORD 'urchinpassword';
postgres=# GRANT ALL ON DATABASE urchin TO urchin;
postgres=# \connect urchin;
postgres=# CREATE LANGUAGE 'PLPGSQL';
```

## Installing Urchin

- Make sure that you have the Urchin configuration database name, user and password that you selected above.
- Create a temporary directory and unpack the Urchin distribution into it:

```
mkdir /var/tmp/urchintemp
cd /var/tmp/urchintemp
tar xzvf urchin6500_<platform>_installer.tar.gz
```

Create a temporary directory and unpack the Urchin distribution into it.

- **Important!** To ensure correct installation, make sure that there is no 'active' instance of Urchin 6 on the target machine. So disable any existing version of Urchin 6 by manually typing following command:

```
/path/to/urchin6/bin/urchinctl stop
```

- If multiple instances of Urchin 6 are required to run on the same machine, then they should be installed in different locations, should have different web server ports and connect to different databases.
- Urchin copies all necessary files to the target folders and overwrites any existing files. In order to keep any customization intact, user should install Urchin 6 in a new directory, or rename the existing installation directory.
- Urchin 6.5 can either be installed from scratch or upgraded from an existing version of Urchin 6.
  - During 'New' installation, Urchin prompts user for all configuration parameters, like web server port, db configuration settings etc. In case of 'upgrade', Urchin reuses the configuration of existing Urchin 6 instance.
  - Urchin 6.5 can only be used for upgrading some earlier version of Urchin 6. The supported versions for upgrade are:
    - Urchin 6 - English Release : Version 6.400
    - Urchin 6 - International Release : Version 6.401
    - Urchin 6 - Service Pack 1 : Version 6.402
  - During 'New' installation, Urchin prompts user to "Initialize database tables". If user want to keep existing DB configuration, then '**No**' should be selected. Although, in this case, Urchin will keep existing configuration by not recreating the existing tables, yet it may create '*additional tables*' or '*columns in existing tables*' for proper functioning of Urchin 6.5.

- Now run the Urchin installer. You can either run it interactively and follow the prompts by typing:

```
./install.sh
```

or you can do an complete install directly from the command line, supplying all the necessary parameters for the install. You can obtain a list of the command line parameters by typing:

```
./install.sh -h
```

- Once the installer completes, you should now have a running version of Urchin on your system. To finish the setup, point your favorite web browser at this URL:

```
http://yourserver:<port>
```

where <port> is the network port you chose during the installation (default is 9999).

- If you wish to restore your profile reporting data from an older version of Urchin 6, copy the `history`, `reports` and `cpc` (if exists) directories from `/path/to/old-urchin6/data/` to your new `/path/to/old-urchin6/data/` directory.
- Click on the **Obtain Demo License** link and follow the licensing steps to enable Urchin and perform the initial setup steps.

## Configuring Tracking

Once Urchin is installed, there may be additional steps required to set up tracking on your website depending on your use case below.

### Case 1: Using Urchin with IP-UserAgent Tracking

No additional configuration is necessary. When adding profiles to Urchin, be sure to specify IP+UserAgent as the Visitor Tracking Method.

### Case 2: Using Urchin with UTM Tracking

To use Urchin with UTM tracking, it is necessary to install a small piece of javascript tracking code on each of your website's pages. Here are the steps:

- Copy the `urchin.js` and `__utm.gif` files from the `util/utm` directory of the Urchin distribution to the document root of your website (e.g. the top level directory of your content)
- On each page of your website, place the following tracking code right after the any META tags in the HEAD section:

```
<script src="/urchin.js" type="text/javascript">
</script>
<script type="text/javascript">
  _userv=0;
  urchinTracker();
</script>
```

- When adding profiles to Urchin, be sure to specify Urchin Traffic Monitor (UTM) as the Visitor Tracking Method

### Case 3: Using Urchin concurrently with Google Analytics

Please note that the new tracking features in the Google Analytics "ga.js" tracking code are not backward compatible with Urchin 6. If you wish to track your website with both Google Analytics and Urchin Software, we recommend using the ga.js-based tracking code for best results. If you choose to use `urchin.js`, you will not be able to use the new ga.js tracking features in Google Analytics.

To use Urchin with a website that is already being tracked with Google Analytics through the "**ga.js**" tracking code, you just need to install a small piece of javascript tracking code on each of your website's pages. Here are the steps:

- Copy the `urchin.js` and `__utm.gif` files from the `util\utm` directory of the Urchin distribution to the document root of your website (e.g. the top level directory of your content)
- On each page of your website, place the following tracking code right after the any META tags in the HEAD section:

```
<script src="/urchin.js" type="text/javascript">
</script>
<script type="text/javascript">
  _userv=0;
  urchinTracker();
</script>
```

To use Urchin with a website that is already being tracked with Google Analytics through the old version of the tracking code "**urchin.js**", you just need to make a slight modification to the existing tracking code for Google Analytics. Here are the steps:

- Copy only the \_\_utm.gif file from the util\utm directory of the Urchin distribution to the document root of your website (e.g. the top level directory of your content).
- Add the line shown in **bold** below to the existing tracking code on your website pages. Be sure not to modify any other parts of the existing tracking code!

```
<script src="http://www.google-analytics.com/urchin.js"
type="text/javascript">
<script type="text/javascript">
  _userv=2;
  _uacct="UA-XXXXXXX-Y";
  urchinTracker();
</script>
```

- When adding profiles to Urchin, be sure to specify Urchin Traffic Monitor (UTM) as the Visitor Tracking Method