Evergreen Quick Reports Creator Installation Guide

# Prerequisites

1. A working Apache web server, with PHP enabled

**Install PHP and extensions (assumes Debian/Ubuntu based system), as root**  
apt-get install php5 php5-gd php5-pgsql php5-memcache php-pear

pecl install memcache **Note this step may not be required with later Ubuntu versions**  
  
**Restart Apache to activate PHP**  
/etc/init.d/apache2 restart  
  
**Create the directory report-creator. We recommend using the preconfigured Evergreen ILS vhost and placing the files in its root (/openils/var/web)**  
mkdir /openils/var/web/report-creator  
  
**Since Apache runs as opensrf user in most Evergreen installs, we must allow report-creator dir to be writable by the opensrf user (adjust if you are using a different user)**  
chown opensrf:opensrf /openils/var/web/report-creator

1. PHP

This version of Quick Reports was tested on PHP 5.3.10 with the following extensions:

* + **pdo\_pgsql**
  + **memcache**

1. Need to force the /openils/var/web/report-creator/index.php to load by default, utilizing an .htaccess file (included in this source code) or by creating a new file at /openils/var/web/report-creator/.htaccess containing;

DirectoryIndex index.php

# Installation

Note: all paths are relative to the directory where you installed the software

1. Create a new database schema and tables in your pre-existing PostgreSQL instance, installed by Evergreen, by executing the SQL commands in the file entitled sql/quick\_reports\_setup.sql

The default schema that will be created is named quick\_reports. This can be edited if you want to use a different schema. Simply edit the quick\_reports\_setup.sql according to your preferred schema name.

1. If necessary, grant SELECT and UPDATE privileges to the user account that will be used to connect to the database from php. Depending on how you configure the application in the steps below, this is either the user defined in the <reporter> node of the file /openils/conf/openils.xml or a user you specify in the config file below.
2. Extract (or pull from Git) the report-creator application files into the previously created /openils/var/web/report-creator directory so that directory is populated and contains the PHP, CSS, and other code files for Quick Reports.
3. If necessary, change ownership of the files so they can be executed by the web server (typically chown them to the opensrf user).
4. Validate that the web server is able to execute php scripts by executing <install\_location>/tools/testphp.php from your browser. You should see

Hello World! PHP is running.

1. Configure the production.config.php file located in the report-creator/config directory. Refer to the Configuration Parameters section below.
2. Test the connection to PostgreSQL by running <install\_location>/tools/testdb.php from your browser.
   * The tool will attempt to connect to the database and execute a query that returns a count of templates in the reporter.template table. You should see output similar to:

[db:protected] => PDO Object

15 templates were found in reporter.template

* + Check the php error logs for errors if you are not seeing connection information or the number of templates displayed, or if errors are displayed.

1. If you will be using memcache to store sessions (for example, in a clustered “brick” environment), test the connection to the memcache server by running <install\_location>/tools/testmemcache.php from your browser.
   * If there are any error messages then there is a problem connecting to the memcache server. Resolve the issue and retest.
   * If successful, you will see the following message:

A session variable was set. Please refresh the page.

* + Refresh the page by pressing F5. You will then see:

Session variables appear to be operating correctly.   
Array ( [lastAccessTime] => 1429379837 [testMemcache] => ok

* + Note: The value of last AccessTime that you see will be different since it represents the current timestamp. If you refresh the page again, the lastAccessTime should increase. If it does then memcache sessions are working correctly.
  + Check the php error logs for errors if you are not seeing this behavior.

1. A second tool for testing memcache is provided. Execute this from your browser

<install\_location>/tools/testmemcache2.php

This tool is simpler and will display the version of memcache. Check the php error logs for errors if the version is not displayed.

1. Test the software by executing <https://www.yourdomain.tld/report-creator> in your preferred web browser. You should be able to log in with a valid Evergreen user with the proper permissions to access the Quick Reports as shown below.

**Note:** The web server must be able to execute index.php without having to include index.php on the URL (see prerequisite note about .htaccess)

# Evergreen User Permission Requirements

The QR Admin must have the following permissions:

* ADMIN\_SIMPLE\_REPORTS
* CREATE\_REPORT\_TEMPLATE
* RUN\_REPORTS
* SHARE\_REPORT\_FOLDER
* VIEW\_REPORT\_OUTPUT

All library staff who will access Quick Reports must have:

* RUN\_REPORTS
* VIEW\_REPORT\_OUTPUT

# Troubleshooting

If you are still having problems with php you can check your installation by creating a phpinfo page and execute it from your web browser. The contents of the page should be:

<?php

phpinfo();

?>

Save this file with a .php extension and execute it from your browser. Check the configuration for the following sections.

memcache (if you are using memcache to store session information)

PDO

pdo\_pgsql

pgsql

**IMPORTANT:** Be sure to remove this file when you are done. It provides important configuration information that could potentially be used to compromise the security of your environment.

# CONFIGURATION PARAMETERS

|  |  |  |  |
| --- | --- | --- | --- |
| PARAMETER | DESCRIPTION | DEFAULT | ADDITIONAL REFERENCE |
| ERROR\_REPORTING\_LEVEL | php error reporting level | E\_ALL | <http://php.net/manual/en/errorfunc.constants.php> |
| DEFAULT\_TIME\_ZONE |  |  | <http://php.net/manual/en/timezones.america.php> |
| SITE\_ROOT | The application directory | / | This is the <install\_location> used in the instructions above. |
| SESSION\_TIMEOUT | Sessions expire after this many minutes | 120 | <http://php.net/manual/en/session.configuration.php#ini.session.gc-maxlifetime>  Note: the application will multiply this setting by 60 to convert to seconds. |
| BROWSER\_PAGE\_CACHE\_TIMEOUT | Browser page cache time in seconds | 20 | Seconds  <http://php.net/manual/en/function.session-cache-expire.php> |
| MAX\_LOGIN\_ATTEMPTS\_ALLOWED | Max login attempts | 3 | Users receive a different error message after this many unsuccessful login attempts. |
| USERS\_ALLOWED\_PERMISSIONS | Comma separated string of integers | 141,143 | Users with these permission (defined in the table permission.usr\_perm\_map) will be allowed to login and use the application. |
| ADMINS\_ALLOWED\_PERMISSIONS | Comma separated string of integers | 1200 | Users with these permission (defined in the table permission.usr\_perm\_map) will be allowed to login and use the application and will be given access to the admin functions. |
| ALLOW\_GROUP\_PERMISSIONS | Boolean | false | If true, grants permissions to users who belong to a group (defined in permission.grp\_perm\_map) that has permissions for: USERS\_ALLOWED\_PERMISSIONS ADMINS\_ALLOWED\_PERMISSIONS |
| ADMIN\_PERMISSION\_REQUIRED\_MESSAGE | Text |  | Error message that is displayed if a user attempts to access admin functions without sufficient permissions. |
| DB\_SCHEMA | Text | quick\_reports | The name of the database schema that was created during the setup procedure. |
| PARENT\_FOLDER\_NAME | Text | Quick Reports | The name of the folder where reports are saved in the original reporting tool. |
| REPORT\_FOLDER\_NAME | Text | Reporter.report\_folder | The database table that contains the report folder info. |
| OUTPUT\_FOLDER\_NAME | Text | Reporter.output\_folder | The database table that contains the report output folder info. |
| ADMIN\_SCHEDULED\_REPORTS\_QUERY\_DAYS | Integer | 30 | Number of days in the past to include when an admin gets a list of scheduled reports. Currently running and future reports will always be included. |
| SHOW\_DOC\_URL | Boolean | True | For v4 templates, indicates whether to show the DocURL along with the description on the Quick Report Templates page. |
| SHOW\_FIELD\_DOC | Boolean | True | For v4 templates, indicates whether to show the additional Field Doc information for report filter fields when creating or editing a report. |
| PGSQL\_USE\_OPENSRF\_XML\_CONFIG | Boolean | True | If true, obtain PostgreSQL login information from the opensrf config fle. This will be slower but more secure.  If false, define the PostgreSQLl login information explicitly in this config file. |
| OPENSRF\_XML\_PATH | Text | /openils/conf/opensrf.xml | This is the file used to obtain config information when either PGSQL\_USE\_OPENSRF\_XML\_CONFIG or MEMCACHE\_USE\_OPENSRF\_XML\_CONFIG is set to true. |
| PGSQL\_HOST | PostgreSQL host |  | Required when PGSQL\_USE\_OPENSRF\_XML\_CONFIG is false |
| PGSQL\_PORT | PostgreSQL port |  | Required when PGSQL\_USE\_OPENSRF\_XML\_CONFIG is false |
| PGSQL\_DBNAME | PostgreSQL database name |  | Required when PGSQL\_USE\_OPENSRF\_XML\_CONFIG is false |
| PGSQL\_USER | PostgreSQL user name |  | Required when PGSQL\_USE\_OPENSRF\_XML\_CONFIG is false. Ignored when true. |
| PGSQL\_PASSWORD | PostgreSQL passwrord |  | Required when PGSQL\_USE\_OPENSRF\_XML\_CONFIG is false. Ignored when true. |
| SESSIONS\_IN\_MEMCACHE | Boolean | True | If true, store session data in memcache. This is required if there is more than one server running this application and sticky sessions are not being implemented by a load balancer to ensure a user always hits the same server.  If false, standard php sessions are used and stored on the local drive of the server. This is best when there is only one server running the application or if there are multiple servers, the load balancer is using sticky sessions to ensure a user always accesses the same sever. |
| MEMCACHE\_USE\_OPENSRF\_XML\_CONFIG | Boolean | True | If SESSIONS\_IN\_MEMCACHE is true then this setting determines where to get the memcache configuration information. If true, the memcache configuration is obtained from the opensrf config file specified in OPENSRF\_XML\_PATH |
| MEMCACHE\_PROTOCOL\_1 | Protocol | Tcp | Protocol to use to access the primary memcache server. |
| MEMCACHE\_PROTOCOL\_2 | Protocol | Tcp | Protocol to use to access the secondary memcache server. |
| MEMCACHE\_PARAMS\_1 | Text | ?persistent=1&weight=2&timeout=2&retry\_interval=10 | Parameter string to use when accessing the primary memcache server |
| MEMCACHE\_PARAMS\_2 | Text | ?persistent=1&weight=2&timeout=2&retry\_interval=10 | Parameter string to use when accessing the secondary memcache server |
| MEMCACHE\_HOST\_1 | Primary memcache host name or IP address |  | Required when MEMCACHE\_USE\_OPENSRF\_XML\_CONFIG is false. Ignored when true. |
| MEMCACHE\_HOST\_2 | Secondary memcache host name or IP address |  | Required when MEMCACHE\_USE\_OPENSRF\_XML\_CONFIG is false. Ignored when true.  Comment out this setting if there is only one memcache server or set it to the same value as MEMCACHE\_HOST\_1. |
| MEMCACHE\_PORT\_1 | Port to use to access the primary memcache server | 11211 | Required when MEMCACHE\_USE\_OPENSRF\_XML\_CONFIG is false. Ignored when true. |
| MEMCACHE\_PORT\_2 | Port to use to access the secondary memcache server | 11211 | Required when MEMCACHE\_USE\_OPENSRF\_XML\_CONFIG is false. Ignored when true. |