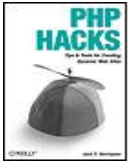


Free Open Book



PHP Hacks

Origami Paper Planes



◀ PREV

NEXT ▶

Hack 50. Create a Message Queue



Use a MySQL table to create a simple message queue for delayed notifications.

Background processing is always a problem in web applications. Users like to get snappy responses from their web pages, but sometimes processing can take a while. A classic example is a web notification mail-out. The user initiates some process that requires mailing a notice to 100 people, but mailing to 100 people takes a while. Making the server wait for all the mail to go out isn't a good idea. It looks like the application is hung or about to crash. Ideally we could return a page to the user, and then in the background, send out the 100 messages. But how do we do that?

One method is to create a simple database-driven message queue. The web page puts some data into the queue, which is executed by another process later (usually fired off by the `cron` task-scheduling system). The message queue in this system takes two parameters: the function to be run and the arguments to the function. Therefore, you can delay almost any processing you want.

Figure 5-27 shows the simple schema for the message queue. There are really only two fields: `func`, which holds the name of the function, and `args`, which holds the XML version of the arguments.

Figure 5-27. The schema for the queue

queue	
id	
func	
args	

Figure 5-28 shows the status of the queue table. It starts empty, with no messages. Then a couple of messages are added. Messages are removed as they are processed.

Figure 5-28. The queue after a few operations

Main Menu

- PHP Hacks
- Table of Contents
- Copyright
- Credits
- Preface
- Chapter 1. Installation and Basics
- Chapter 2. Web Design
- Chapter 3. DHTML
- Chapter 4. Graphics
- Chapter 5. Databases and XML
 - Section 5.1.
 - Hacks 3450: Introduction
 - Hack 34. Design Better SQL Schemas
 - Hack 35. Create Bulletproof Database Access
 - Hack 36. Create Dynamic Database Access Objects
 - Hack 37. Generate CRUD Database Code
 - Hack 38. Read XML on the Cheap with Regular Expressions
 - Hack 39. Export Database Schema as XML
 - Hack 40. Create a Simple XML Query Handler for Database Access
 - Hack 41. Generate Database SQL
 - Hack 42. Generate Database Select Code
 - Hack 43. Convert CSV to PHP
 - Hack 44. Scrape Web Pages for Data
 - Hack 45. Suck Data from Excel Uploads
 - Hack 46. Load Your Database from Excel
 - Hack 47. Search Microsoft Word Documents
 - Hack 48. Create RTF Documents Dynamically
 - Hack 49. Create Excel Spreadsheets Dynamically
 - Hack 50. Create a Message Queue
- Chapter 6. Application Design

Empty message queue

id	func	args
----	------	------

After two message adds

id	func	args
1	mail_notification	<xml>...</xml>
2	mail_notification	<xml>...</xml>

After one message removal

id	func	args
2	mail_notification	<xml>...</xml>

One of the great things about a message queue is that since it's in the database, it persists even if the process that is reading or writing the queue table crashes.

5.18.1. The Code

Save the SQL in [Example 5-41](#) as queue.sql.







Example 5-41. SQL that sets up the queue in the database

```
DROP TABLE IF EXISTS queue;
CREATE TABLE queue (
    id MEDIUMINT NOT NULL AUTO_INCREMENT,
    func TEXT,
    args TEXT,
    PRIMARY KEY ( id )
);
```




The queue itself is handled by the code in [Example 5-42](#) (saved as queue.php).


Example 5-42. Script that handles queue management

```
<?php
require( "DB.php" );
$db =& DB::connect("mysql://root:password@localhost/queue", array( ));
if (PEAR::isError($db)) { die($db->getMessage( )); }
function add_to_queue( $func, $args )
{
    global $db;
    $dom = new DomDocument( );
    $root = $dom->createElement( "arguments" );
    foreach( $args as $argtext )
    {
```

-  [Chapter 7. Patterns](#)
-  [Chapter 8. Testing](#)
-  [Chapter 9. Alternative UIs](#)
-  [Chapter 10. Fun Stuff](#)
-  [Colophon](#)
-  [Index](#)

More Books

-  [PHP Hacks](#)
-  [Processing Xml With Java - A Guide To Sax, Dom, Jdom, Jaxp, And Trax](#)
-  [The Koran \(Holy Qur'an\)](#)
-  [Macromedia Flash 8 Bible](#)
-  [Search Engine Optimization for Dummies](#)
-  [YouTube Traffic](#)
-  [PHP 5 for Dummies](#)
-  [Harry Potter and The Chamber of Secrets](#)
-  [Harry Potter and the Sorcerer's Stone](#)
-  [The Pilgrim's Progress](#)
-  [Wireless Hacks](#)
-  [Flash Hacks. 100 Industrial-Strength Tips & Tools](#)
-  [PayPal Hacks. 100 Industrial-Strength Tips and Tools](#)
-  [Amazon Hacks](#)
-  [Pdf Hacks](#)
-  [The Da Vinci Code](#)
-  [Google Hacks](#)
-  [The Holy Bible](#)
-  [Windows XP For Dummies](#)
-  [Harry Potter and the Half-Blood Prince](#)
-  [Seo Book](#)
-  [Upgrading and Repairing Networks](#)
-  [Macromedia Dreamweaver 8 UNLEASHED](#)
-  [Windows XP Annoyances](#)
-  [Windows XP Hacks](#)
-  [Microsoft Windows XP Power Toolkit](#)
-  [Teach Yourself MS Office In 24Hours](#)
-  [iPod & iTunes Missing Manual](#)
-  [PC Hacks 100 Industrial-Strength Tips and Tools](#)
-  [PC Overclocking, Optimization, and Tuning - 2th Edition](#)
-  [PC Hardware In A Nutshell 3rd Edition](#)
-  [PC Hardware in a Nutshell, 2nd Edition](#)
-  [Upgrading and Repairing PCs](#)
-  [Google for Dummies](#)
-  [MySQL Cookbook](#)
-  [Teach Yourself Macromedia Flash 8 In 24 Hours](#)
-  [PHP Cookbook](#)
-  [Sams Teach Yourself JavaScript in 24 Hours](#)



```

        $arg = $dom->createElement( "argument" );
        $arg->appendChild( $dom->createTextNode( $argtext ) );
        $root->appendChild( $arg );
    }
    $dom->appendChild( $root );
    $sth = $db->prepare( "INSERT INTO queue VALUES ( 0, ?, ? )" );
    $db->execute( $sth, array( $func, $dom->saveXML( ) ) );
}
function run_queue( )
{
    global $db;
    $delsth = $db->prepare( "DELETE FROM queue WHERE id = ?" );
    $res = $db->query( "SELECT id, func, args FROM queue" );
    while( $res->fetchInto( $row ) )
    {
        $id = $row[0];
        $func = $row[1];

        $argxml = $row[2];
        $dom = new DomDocument( );
        $dom->loadXML( $argxml );
        $args = array( );
        foreach( $dom->getElementsByTagName( "argument" ) as $node )
        {
            $args []= $node->nodeValue;
        }
        call_user_func_array( $func, $args );
        $db->execute( $delsth, array( $id ) );
    }
}
?>

```

add.php, shown in [Example 5-43](#), tests addition of mail to the queue.

Example 5-43. A simple script that fires off a mail notification

run.php ([Example 5-44](#)) gives a quick status notice.

Example 5-44. A script that shows what's going on (sort of) to the user

```

<?php
require_once( "queue.php" );
function mail_notification( $user, $text )
{
    print "Mailing $user:\n$text\n\n";
}
run_queue( );
?>

```

The primary code in this example is in the queue.php library, used by both the add.php script, which adds elements to the queue, and the run.php script, which runs the elements currently stored in the queue.

The queue library defines two functions: `add_to_queue()`, which inserts a record into the queue, and `run_queue()`, which selects all of the rows from the table and runs them one by one. The `run_queue()` function is fun because it never actually knows what it's going to do! It simply reads the function name and the arguments to the function out of the database. Then it invokes the `call_user_func_array()` PHP

function, which takes a function name as a parameter, as well as the arguments to that function as a second parameter. That means you can add functionality to the queue without ever changing these basic add and run functions.

5.18.2. Running the Hack

Running the hack starts by setting up the queue table in the database:

```
% mysqladmin --user=root --password=password create queue
% mysql --user=root --password=password queue < queue.sql
```

Running the add.php script using the PHP command-line interpreter adds a message to the message queue (this kind of code would run on the web page). In this case, it adds an example email item to the queue. You can run the script several times if you want to create several example email messages:

```
% php add.php
```

Then, running the run.php script executes the messages in the event queue:

```
% php run.php
Mailing jack@oreilly.com:
You owe us some money. Pay up.
```

Each message is printed to the console but can be mailed out to someone easily.

5.18.3. See Also

- "Observe Your Objects" [Hack #67]

◀ PREV

NEXT ▶

Index: [SYMBOL][A][B][C][D][E][F][G][H][I][J][K][L][M][N][O][P][Q][R][S][T][U][V][W][X][Y][Z]

Origami Paper AirPlane



Best sites ▼

Free Games

Paper Airplane

Paper Airplane - Paper Airplane - Paper Airplane - Paper Airplane - Paper Airplane - Paper Airplane -
Paper Airplane - Paper Airplane -