ConversionFromPhpNuke

PhpNuke conversion

Introduction

I have been using))PhpNuke((for my website for two years now, but I would like to move on to Tiki. To help others I'll describe this process for my own website.

Update: I have decided NOT to use Tikiwiki for my own website. This has nothing to do with Tikiwiki, it's just that I prefer to use a Java based tool (I'll use MMBase, a professional open source CMS). I hope that if someone ever wants to migrate from PhpNuke to Tikiwiki, this page will give a starting point.

Assumptions

This document describes the conversion from PhpNuke 6.0. It does not attempt to be a step-by-step conversion manual but I hope it will make conversions easier. Note: this type of conversions is not trivial.

Goals

I would like to convert the PhpNuke stories to Tikiwiki articles. This includes the topics and comments. I want to do a one-time conversion. It would also be possible to re-use the PhpNuke code and database tables for stories, and integrate them into the Tiki program structure, but then what would be the point of conversion?

Stories to Articles

The first step is to convert PhpNuke's stories to articles. These are similar structured in both CMS tools, so this shouldn't be extremely hard. Let's have a look at the table structures first.

PhpNuke stories

mysql> describe nuke_stories ;

Field	Type	Null	Key	Default	Extra
sid	int(11)		PRI	NULL	auto_increment
catid	int(11)		MUL	0	
aid	varchar(30)				
title	varchar(80)	YES		NULL	
time	datetime	YES		NULL	
hometext	text	YES		NULL	
bodytext	text				
comments	int(11)	YES		0	
counter	mediumint(8) unsigned	YES		NULL	
topic	int(3)			1	
informant	varchar(20)				
notes	text				
ihome	int(1)			0	
alanguage	varchar(30)				
acomm	int(1)			0	
haspoll	int(1)			0	

pollID	int(10)		0
score	int(10)		0
ratings	int(10)		0

19 rows in set (0.00 sec)

Tiki articles

mysql> describe tiki_articles ;

Field	Type	Null	Key	Default	Extra
articleId	int(8)		PRI	NULL	auto_increment
title	varchar(80)	YES	MUL	NULL	
authorName	varchar(60)	YES		NULL	
topicId	int(14)	YES		NULL	
topicName	varchar(40)	YES		NULL	
size	int(12)	YES		NULL	
useImage	char(1)	YES		NULL	
image_name	varchar(80)	YES		NULL	
image_type	varchar(80)	YES		NULL	
image_size	int(14)	YES		NULL	
image_x	int(4)	YES		NULL	
image_y	int(4)	YES		NULL	
image_data	longblob	YES		NULL	
publishDate	int(14)	YES		NULL	
created	int(14)	YES		NULL	
heading	text	YES	MUL	NULL	
body	text	YES	MUL	NULL	
hash	varchar(32)	YES		NULL	
author	varchar(200)	YES		NULL	
reads	int(14)	YES	MUL	NULL	
votes	int(8)	YES		NULL	
points	int(14)	YES		NULL	
type	varchar(50)	YES		NULL	
rating	decimal(3,2)	YES		NULL	
isfloat	char(1)	YES		NULL	

25 rows in set (0.00 sec)

Observations

- PhpNuke's *rating* is different in concept than Tiki's. In PhpNuke it indicates the opinion of readers about a page, in Tiki it is a rating given by the author of the article about the subject of the article.
- The unique identifier of a story (*sid*) is an integer with length 11, whereas Tiki's *articleId* has 8 as

length. This indicates the display size only; the data-type is still integer so conversion is possible.
- For many websites the ID of an article is known outside the database, it is often used in links for example. Therefore it's important to keep the articleId and sid in sync.
- The date-type used in PhpNuke is *datetime*, Tiki uses an *int* for this.

Conversion

The easiest way to convert between two different databases is to write a small script. I used PHP for this. As stated above it is important to keep the articleId and sid the same. These are unique keys. When experimenting with this script, it can be useful to remove all articles in Tiki sometimes. Syntax for that is:

Removing Tiki articles: delete from tiki_articles ;

Date datatype

The conversion from the PhpNuke date format (a textual representation, yyyy-mm-dd hh:mm) to the timestamps used by Wiki can be done using mktime(). PhpNuke doesn't make a difference between publish- and creation date (although programmed articles are supported). Therefore the script will consider the publish- and creation date to be the same.

Bodytext and heading contents

An important difference in the rendering of PhpNuke and Tiki is the treatment of linebreaks. PhpNuke stores the contents of a story as HTML in the database. In Tikiwiki the contents of an article is stored as text; when rendering an article linebreaks will be converted to <br /> tags. The obviously makes more sense when taking into account different outputs (that do not recognize HTML tags), but it requires a text conversion in this script.

Scripts

This is a **very preliminary version**! It gives an indication of how the script will look like, it's far from usable. It converts only the 5 first articles.

The script:

```
<?php
error_reporting (E_ALL);
Function textConvert($original) {
$string = preg_replace("(\n)", "", $original);
$string = mysql_escape_string($string);
return $string;
}
Function Transfer () { global $id, $authorName, $title, $hometext, $bodytext, $time, $counter;

mysql_connect("localhost","root","");

$database="tiki";
@mysql_select_db($database) or die( "Unable to select database");

$publishTimeStamp = strtotime("$time");
$query = "insert into tiki_articles set articleId='$id', title='$title', reads='$counter',
authorName='$authorName', created='$publishTimeStamp', type = 0, useImage='n',isfloat='n', heading=
'$hometext', body= '$bodytext', hash= '', publishDate = '$publishTimeStamp' ";

$result=mysql_query($query) or die ("Query error: " . mysql_error());;
echo "Result: $result <br />";
}
echo "<html><head><title>Conversion</title></head><body>Conversion...";
mysql_connect("localhost","root",""); $database="nuke";
```

```
@mysql_select_db("$database") or die( "Unable to select database nuke");

$table="nuke_stories";

$query="select * from $table order by time desc";

$result=mysql_query($query);


$num=mysql_numrows($result);

echo "Number of rows: $num<br /><br />";
$i=0;
while ($i < $num):

$authorName=mysql_result($result,$i,"aid");
$title=textConvert(mysql_result($result,$i,"title"));
$hometext=textConvert(mysql_result($result,$i,"hometext"));
$bodytext=textConvert(mysql_result($result,$i,"bodytext"));
$id=mysql_result($result,$i,"id");
$time =mysql_result($result,$i,"time");
$counter = mysql_result($result, $i, "counter");
echo "$id $title, ";
Transfer ();

$i++;
endwhile;

?>
```