Pre-requisites

You need Mapserver v4.6 php_mapscript.so and have your .map files in /var/www/html/map/. This is configurable from the admin panel: Maps Config, but first ensure that you have enabled the maps feature in Admin.

Mapserver v4.6 mapscript is now capable to work as an extension to the php apache module

The default map file is pacific.map, it will be loaded if you do not specify anything when clicking on tiki-map.phtml. This is also configurable from the Admin/Features menu.

Compilation of mapscript

I use the following to compile mapserver.so, but now you may find ready made packages for various distributions:

    ./configure --with-proj --with-gdal --with-ogr --with-wms --with-php=/usr/include/php --without-tiff --with-threads make

Installation of mapscript

I then copy the file mapscrip/php3/php_mapscript.so into the php extension directory, usually /usr/lib/php/extensions. I then need to get php to load automatically this extension. I create an ini file for that.

/etc/php/40_mapscript.ini

```
#Loads the MapScript extension
extension = php_mapscript.so
```

To test if the extension is correctly loaded with all library dependencies, install php as command line inteface and run:

```
php -m
```

You should see the module))MapScript((loaded correctly. If a library is missing then you will have to install it too. Check the error messages generated at the beginning by php -m. Most common libraries to install grass, proj, gdal,...

Check in your Apache configuration that phtml files like php files are handled by the Apache php module.

Creation of directories and installation of sample data

I now downlaod the tikiwiki-mapdata package, that you can find in the file section of the tikiwiki project on sourceforge, and I install the contents into the directory /var/ww/html/map. It contains

a world.map file and the data that goes with it in the data/world directory.

Your files that compose the layers must be in the data subdirectory of where are your map files. ie /var/www/html/map/data. They are then manageable with the Layer Manager inside Tiki. You can create subdirectories and upload your geographic files using the Tiki interface.

You need to create a directory /var/www/html/map/images. The mapscript will save the generated maps/images into this directory and Apache will serve them to the browser.

Now ensure that everything under the directory /var/www/html/map is owned by apache.apache, the user and group that runs Apache.

```
chown -R apache.apache /var/www/html/map
```

You should have this directory structure:

```
/var/www/html/map
/var/www/html/map/world.map
/var/www/html/map/images/
/var/www/html/map/data/
/var/www/html/map/data/world/
/var/www/html/map/data/world/* (layers)
```

This directory structure is unsafe, I use the following conf file for Apache to ensure that the maps and data directory are protected against direct download of data. Some of the data you upload should be only visible as rendered as a map.

/etc/httpd/conf.d/41_mapserver.conf

```
 #protect mapserver against downloads
<Directory /var/www/html/map>
    AllowOverride All
    Options -Indexes FollowSymLinks MultiViews ExecCGI
    <IfModule mod_access.c>
        Order allow,deny
        Deny from all
    </IfModule>
</Directory>
<Directory /var/www/html/map/images>
    AllowOverride All
    Options -Indexes FollowSymLinks MultiViews ExecCGI
    <IfModule mod_access.c>
        Order allow,deny
        Allow from all
    </IfModule>
</Directory>
<Directory /var/www/html/map/graphics>
```

```
        AllowOverride All
        Options -Indexes FollowSymLinks MultiViews ExecCGI
        <IfModule mod_access.c>
            Order allow,deny
            Allow from all
        </IfModule>
    </Directory>
    <Location /tiki/files>
        AllowOverride All
        Options -Indexes FollowSymLinks MultiViews ExecCGI
        <IfModule mod_access.c>
            Order allow,deny
            Allow from all
        </IfModule>
        <IfModule mod_mime.c>
            SetHandler default-handler
        </IfModule>
    </Location>
```

Before Mapserver 4.6, php mapscript was obliged to run under a php cgi module. This is why the map rendering page is called tiki-map.phtml instead of the classical php extension. Fortunately since, the mapscript runs inside the php apache module.

When someone wants to download a layer, all the files are copied in the /tiki/files location for download. Unfortunately on my system the file with the extension .MAP are handled as html maps. I created a special directive for http to handle all the files in the /tiki/files directory as normal files as you can see in the apache configuration file above.

Automatic removal of generated images

You also need to add a file to delete generated images, or the /var/www/html/map/images directory will grow indefinitively. I use a file in cron:
/etc/cron.hourly/mapwatch

```
/usr/sbin/tmpwatch -f 2 /var/www/html/map/images/
```

This will check every hours for file older than 2 hours and delete them.

Complete setup in Tiki

Finish the setup on the admin panel with Maps Config. If you have used the tikiwiki-mapdata package ensure the default mapfile is world.map.

Then you need to create a mapfile and set the correct paths see Maps Mapfile Tutorial. If you have used the tikiwiki-mapdata and the paths indicated in this install then you should be set, otherwise you need to edit the world.map file to point to the various paths you have chosen.

Finally ensure that some groups have the correct permissions to view, create and delete mapfiles and their data.