

Charabanc Documentation

Andrew Baxter

March 10, 2008

Contents

1	Introduction	1
1.1	User Interface	1
1.2	Technical introduction	2
2	Package contents	2
3	Making the images	3
3.1	The panoramas and slideshows	3
3.2	The map	4
4	Configuration	4
4.1	Virtual-tour.html	4
4.2	Charabanc.css	4
4.3	Infopage.css	4
4.4	Setup.xml	4
5	Installation	5
5.1	Dojo	5
5.2	Web server config	5

1 Introduction

This document is a rough guide to how to set up Charabanc on your own web server. It doesn't go into step-by-step detail of how to do it, but instead just gives you an overview of how the system works and what you need to do to set it up. I am assuming that if you want to set up a program like this, you know enough to understand these instructions and work out the rest yourself. Also that you already understand the basics of how to set up a website on a server. You are recommended to read through the whole document at least once before you start trying to set anything up. If you have any questions about Charabanc, contact me through the address on the website <http://charabanc.sf.net>.

1.1 User Interface

Charabanc is a javascript program you can use on your website to take people on an online virtual tour of a place. The way it works is that you take panoramic photographs at a series of viewpoints around the site, which are displayed one at a time in a viewport on the page. You can scroll the image in the viewport so it looks as if you are looking around in a full circle.

The images can have hotspots defined, which are active areas that do something when you click on them. There are two kinds of hotspots: path hotspots and info hotspots.

Path hotspots are used for moving from one viewpoint to a neighbouring one. When you click on one it will move you to the viewpoint that the path you have clicked on leads to. While you are moving to the new viewpoint, a slideshow is displayed which makes it look like you are walking along the path.

You can also move to a new viewpoint using a scrollable, zoomable map. Each viewpoint is shown as a small yellow circle on the map, with the one you are currently at shown in red. When you click on a circle, the tour moves to that viewpoint.

Link hotspots bring up a page that tells you more about an object in the picture.

1.2 Technical introduction

Charabanc is written in javascript, an interpreted programming language which runs inside your web browser to allow dynamic effects on web pages. It uses an open source javascript toolkit called dojo <http://www.dojotoolkit.org> to do a lot of the hard work such as making the code cross-browser compatible, and displaying the tabs and zoom bar. It runs entirely client-side, with the only connection to the webserver being plain http requests, so you don't need any special code (e.g. PHP) installed on your web server to use it. You configure it using an xml file which is loaded from the server when the program is initialising. This defines the names of all the viewpoints, the links between them, their location on the map, and so on.

The pictures for the demo, which is from the Middlewood Trust website, were taken on an ordinary digital camera and then stitched together to make panoramas using an open source program called hugin. The charabanc installation files include a few perl scripts which help automate the process of converting the images for the panoramas and slideshows into the right sizes to work with charabanc.

2 Package contents

This section describes the layout of the tar.gz package that you can install from the charabanc website.

Filename	Description
docs/	directory containing the documentation.
scripts/	directory containing scripts which are useful in setting up charabanc.
scripts/cut-map.pl	script to cut a large image of a map into a series of tiles at different zoom levels.
scripts/scale-panoramas.pl	script to scale the raw panorama images into various different resolutions.
scripts/scale-walks.pl	script to scale the raw slideshow images into various different resolutions.
scripts/charabanc-profile.js	used when creating a custom dojo build.
tour/	directory containing the code and other files to run the virtual tour. This is the directory you need to copy to your web server.
tour/css/	directory containing stylesheets.
tour/css/images/	images used by the stylesheets.
tour/css/dojoStyles.css	stylesheet used to customise the dojo toolkit's tab widget.

tour/css/infopage.css	stylesheet which controls the appearance of information pages.
tour/css/tour.css	the main stylesheet for the web page.
tour/images/	directory containing images used by the actual tour program.
tour/js/	directory containing javascript files.
tour/js/pano.js	the main javascript program file for charabanc.
tour/index.html	a very simple index page which displays a short message and a link to open the real tour page.
tour/setup.xml	the xml configuration file. You edit this to tell pano.js what viewpoints, slideshows, and info pages you have defined. The version here is an example from the demo site, with comments included.
tour/virtual-tour.html	the main web page of the site. This lays out the html structure of the charabanc page and loads all the required stylesheets and script files to make the program run.

3 Making the images

3.1 The panoramas and slideshows

The way I created the panoramas for the demo site was to go to each viewpoint and take photos on a digital camera in a full circle. Having done this for each viewpoint, you can then stitch the photos together into panoramas using a tool like hugin <http://hugin.sf.net>. This involves defining control points between each pair of images in the panorama which show which points coincide. You need to set hugin to do 'cylindrical panoramas' for it to work with charabanc. There are also various Windows tools which stitch panoramas automatically, but I don't know how good they are. The gimp <http://www.gimp.org> is handy for editing the stitched panoramas - e.g. to remove any curvature in the panorama (use the 'curve bend' filter), or adjust the brightness and contrast.

At this point you should have a set of raw picture files, one for each panorama. However, charabanc expects to find the panorama images in a single directory, with filenames matching 'viewpointname-resolution.jpg', in a series of different resolutions. The reason for the different resolutions is to allow people viewing the site to adjust the image size according to their bandwidth and screen size.

The script `scale-panoramas.pl` can be used to scale the raw images to the required resolutions and give them the right names. To use this script, and the other scripts in the scripts directory, you will need to have the `Imlib2` library and the related perl module installed. This only runs in unix environments as far as I know (linux, BSD and Mac OSX) so if you're using Windows you may have to install cygwin <http://www.cygwin.com>. It expects to find the raw files in a directory which has one raw image per subdirectory, with the filename being the subdirectory name with the extension `.tiff`. It is easiest if you make the subdirectory names the same as the short reference names which will be used when you define the viewpoints and relations between them in the Charabanc config file - e.g. 'gate' or 'carpark'. You may need to edit the first part of the script file to give it the right paths etc for the way you've set things up. See the comments in the script for how to do this.

The slideshow images were also taken using a digital camera. Here the first job is to separate them into subdirectories, each named something like 'gate-carpark' or 'carpark-yard'. Each subdirectory should contain the images, in alphanumeric order, which make up one of the slideshows which run when you move from one viewpoint to another. Then you can run the

script `scale-walks.pl` to scale the images to the right resolutions and put them in the defined output directory. Again, you may need to edit the script to set the right parameters.

3.2 The map

The map was designed in a vector drawing program called inkscape <http://www.inkscape.org>. The image was then exported as a large .png file, and scaled using the script '`cut-map.pl`'. As above, you need to edit the script to make sure it is set up right before you run it. The script will generate a chunk of xml which can be pasted into the configuration file under '`<map>`'. What the script is doing is taking the raw image, cutting it up into tiles whose size is defined at the start of the script (you are recommended to use roughly half the viewport dimensions but any size will work), and outputting the tiles with suitable filenames to a given directory. Then the image is zoomed in by a factor of two and the process is repeated until '`maxZoom`' is reached.

4 Configuration

Having created the images, the next step is to tell Charabanc what they are and where to find them, and to generally configure the system to suit your site. You do this by editing a series of configuration files.

4.1 Virtual-tour.html

This is the html page which describes the basic layout of charabanc on the screen. You may want to edit it slightly to suit your own site, but be careful not to change the basic structure of the document.

4.2 Charabanc.css

This is the stylesheet for virtual-tour.html. You can configure it as you wish to change the visual appearance of the page. To use the built in setup tools, which make it easier for you to configure charabanc, make sure that `tr.config`'s `visibility` property is set to '`visible`'. Remember to set this back again when you have finished.

4.3 Infopage.css

This controls the appearance of the information pages which come up when you click on an info link. For example the colour of the photo border, the margin width and so on.

4.4 Setup.xml

There is an xml config file called `setup.xml` which is loaded by the main charabanc script to tell it what viewpoints you have defined, how they are related, and various other things. The documentation for this file is mostly in comments in the example file itself, so read through it now to get an idea of what's in there.

There is a setup tool built into charabanc which you can use to write the xml data describing the location of hotspots in the panoramas, and the position of the panoramas on the map. See later in section 5 for how to configure your web server to run charabanc. See above in section 4.2 for how to turn on the setup tool. The best way of getting started is to set up the map first using `cut-map.pl`, and paste the xml data generated by this script into the config file

under '<map>'. Once you have the map working, then you can generate the files for one of the panoramas, and write the xml for this under '<views>'. The subsections '<mappoint>' and '<links>' under '<view>' don't need to be filled in yet. Then look at the panorama in charabanc to make sure it's OK. Then make sure the map is zoomed to the area where the viewpoint is, turn on 'record mouse clicks' under the 'Setup' tab, go back to the map, and click on the point where you want the viewpoint symbol to appear. This will generate a line of xml in the 'Setup' tab which you can paste into the xml file.

Now you can start defining regions as hotspots. For each hotspot, you need to start by turning 'record mouse clicks' off, then go to the panorama and make sure it is panned to the right area which shows all of the region you want to define. It may help if you use the highest possible resolution for the panorama image. Then turn 'record mouse clicks' back on, and click on a series of points round in a polygon which define the region you want to be a hotspot. Once you have done this, go back to the 'Setup' tab and copy the series of '<point ... >' xml entries from there into the right place in setup.xml (see the example entries for reference).

Having defined the hotspots, you also need to put in other entries which give more information about each link. These are under '<walks>' and '<infos>' in setup.xml - see the example config file for more help on this.

5 Installation

5.1 Dojo

Charabanc uses a toolkit called dojo <http://dojotoolkit.org> to do a lot of the donkey work - e.g. making the code reasonably cross-browser compatible, displaying the tabbed windows, etc. To run charabanc, you will need to download a 0.4.x version of dojo from their website, and install it in an appropriate place on the server. The location of the dojo release is set in virtual-tour.html - currently it is './dojo/dojo.js'. I would recommend making a compressed build of dojo.js according to the instructions in the dojo 0.4 book - see the dojo website. There is a profile file suitable for this in the scripts directory of the charabanc release. You can also use the dojo compressor to compress pano.js, but leave doing this until you have everything set up the way you want it and are ready to publish the site.

5.2 Web server config

This section describes how to set up charabanc to run with your web server. There are no special requirements for the server, as the whole system works using plain http requests with no server side scripting. You will probably need to set up a local web server to test and configure charabanc before uploading it. Apache <http://httpd.apache.org> is free and works well, but again, any decent web server would do.

The first thing is to decide what kind of directory structure to use. There is some flexibility in this - the config file has options to choose what the paths to the images and info pages are from the main html page. The way I have done this is to have one directory 'tour' with everything in, and four subdirectories of this. These are 'code', 'tour-images', 'info-pages', and 'dojo'. 'Code' is a copy of the 'tour' subdirectory in the distribution, including the edited config file. 'Tour-images' is the directory containing the images for the tour - panoramas and slideshows. 'Info-pages' is all the html pages which come up when you click on an info link, and the relevant images for these. (Note that if you do it this way, the info pages must have their images linked in as './tour-images/image-name.jpg'). 'Dojo' contains version 0.4.3 of the dojo javascript toolkit. See here <http://www.dojotoolkit.org> for details of how to download, configure, and install dojo.

Having decided on the directory structure, you can set the relative paths to the various directories in the first section of the config file. You may also need to edit `virtual-tour.html` to get the right path to `dojo.js`.

The easiest way to upload the files to your web server is to create an image on your own hard disk of how you want the site laid out, then maybe test this against a local installation of apache `http://httpd.apache.org` before using a tool like `lftp` to mirror the local installation to the remote server.

If you need any more help with the installation, refer to the documentation for your web server. If you have any charabanc-specific queries, email me at the address on the website `http://charabanc.sf.net`.

Good luck, and I hope you enjoy setting up your tour.