Furcadia Cache System

Table of Contents

Overview: So what *is* cache?

In a nutshell, cache (in the context of Furcadia and computers in general), is a storage area that contains recently accessed data. This storage area helps to speed up programs by providing a faster means of re-accessing this data.

When it comes to Furcadia, the cache contains files that we've downloaded from the internet and are likely to use them again: custom dreams, custom portraits, dynamic avatars and anything else that doesn't come pre-installed with Furcadia itself.
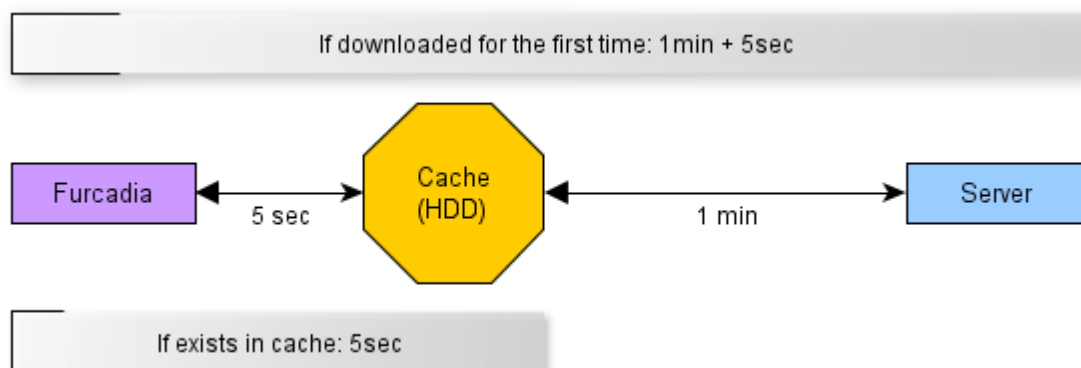
The role of the cache in Furcadia is best described in a diagram above: It takes Furcadia about a minute to download an average dream on an average connection. In addition to that, it takes a few seconds to actually load the dream into memory from the files we've got, so the first time we enter a dream, it takes us over a minute to actually "get there". For the second time and further, since we already have this data on disk, it will only take us a couple of seconds to enter, as downloading this data is no longer necessary!

Cache Limitations: Preventing disk space abuse

On one hand, cache seems like a very useful feature - it speeds Furcadia up! However, this feature comes at a price of disk space: the cache data has to be stored somewhere and that "somewhere" is your hard drive. Every dream you visit, every portrait you look at - they are all saved on your hard-drive and sit there, taking space. It might not be too noticeable, considering that Windows itself and other programs take their share of our 320GB+ hard-disks, but without some kind of limitation, our casual dream-hopping habits can eventually take gigabytes of space, most of which will never be used again! So what do we do about that?
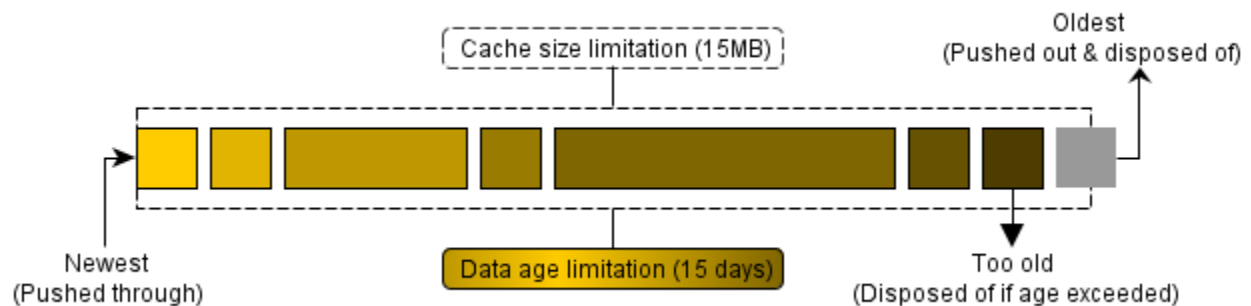
Furcadia imposes several limitations on how big its cache, as well as data age, can get:

1. **Cache size limitation*** - There is a setting in Furcadia that limits the maximum size of your cache: how much disk space it is allowed to use for this whole purpose. This setting can be found in Furcadia Configuration Utility (System and Hardware, Cache section) and adjusted from there.
2. **Data age limitation** - Each dream and portrait is timestamped so Furcadia knows which dream information hasn't been used in a while and should probably be deleted to free up the disk space. The timestamp is kept in a file called **last_used** in each dream folder. For things such as portraits and files, there are other means of knowing how old they are which we won't cover here.
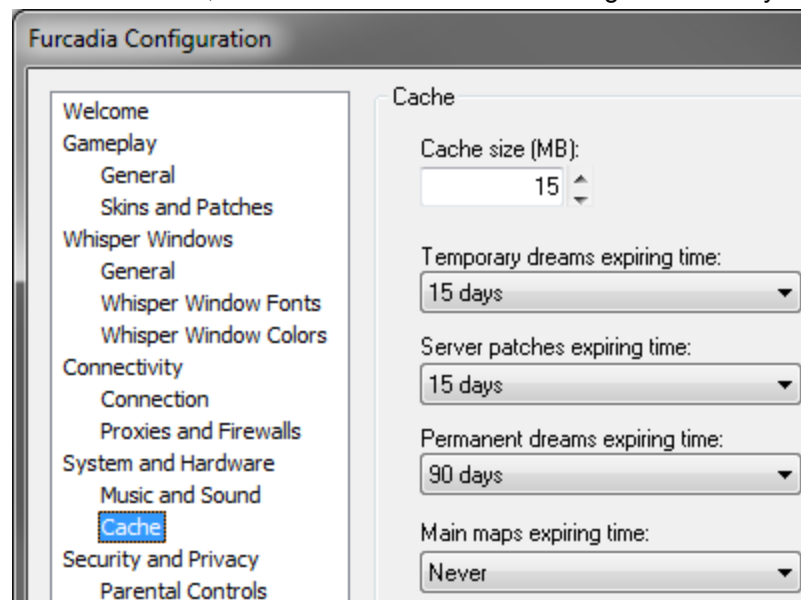
Furcadia Cache Folder: Where is it and what's inside?

When it comes to Furcadia, our cache storage area is a simple folder and depending on which Operating System you have, it is located in different places:

- **Windows 2000/XP:**
  - c:\Documents and Settings\All Users\Application Data\Dragon's Eye Productions\Furcadia\
  - %ALLUSERSPROFILE%\Application Data\Dragon's Eye Productions\Furcadia\
- **Windows Vista/7:**
  - c:\ProgramData\Dragon's Eye Productions\Furcadia\
  - %ALLUSERSPROFILE%\Dragon's Eye Productions\Furcadia\
- **UNIX/Linux (wine):**
  - ~/.wine/drive_c/windows/profiles/All Users/Application Data/Dragon's Eye Productions/Furcadia/
- **MacOS X (wine):**
  - ~/Library/Application Support/Wine/prefixes/Furcadia/drive_c/Users/Public/Application Data/Dragon's Eye Productions/Furcadia/

> **Technical:** Windows sections have the second path which looks similar to the first one. While it doesn't play a big role when it comes to cache, it can be helpful when dealing with irregular systems: systems where the "all users" profile isn't where it usually is, or when dealing with user-specific paths.

Keep in mind that the **Application Data** folder on Windows XP and **ProgramData** folder on Windows Vista/7 are hidden! Therefore, when browsing regularly, the user won't see them and won't be able to continue. They might as well say the folders don't exist when they do!

The best way around it is to use the Run dialog and paste the path in there. Some versions of Windows might throw an error because of the spaces in the path, so if that happens, use double-quotes like this:

"%ALLUSERSPROFILE%\Application Data\Dragon's Eye Productions\Furcadia\"

The Run dialog can be opened from the Start menu, or (more reliably) by pressing WIN+R (where WIN is the Windows key between CTRL and ALT).

Cache Folder Contents

The Furcadia cache folder is split into several subfolders that store specific files:

- **Dynamic Avatars** - avatar files that didn't come pre-installed with Furcadia - these don't expire!
  - **Avatar#.zip** - contains all the files for a single dynamic avatar, extracted when done downloading.
  - **DPlayer#.fox** - contains 20 images of a given avatar.
  - **DPort#.fox** - contains 3 portrait images for a given avatar (unspecified, female, male).
  - **DSpeci#.fox** - contains 3 specitag images for a given avatar (male, female, unspecified).
  - **dynamicavatars.dat** - contains assignment information about each dynamic avatar.
- **Permanent Maps** - dream packages for the main maps, uploaded by Dragon's Eye Productions.
  - **pm*pkgname*** - contains all the necessary files to load a dream.
- **Portrait Cache** - contains downloaded images for later re-use.
  - **images.howl** - an SQLite3 database with images downloaded through the <img/> tags.
  - **portraits.howl** - an SQLite3 database with non-animated custom portraits.
- **Temporary Dreams** - similar to Permanent Maps, this one stores dream data uploaded by other players.
  - **td###########** - contains all the necessary files to load a dream.
- **Temporary Files** - contains files downloaded from the file server or about to be uploaded there.
  - **rca######.rch** - an RCH package for uploading to a the file server.
  - **mpa######.map** - a temporary version of the map we're about to upload.
  - **portrait_######.fox** - an animated custom portrait in its encrypted form.
- **Temporary Patches** - a folder that's never in use, possibly for individual downloaded patches - a feature that's not implemented right now.

Dream Package Contents

When it comes to dream packages (files that represent a single dream), the contents of each package is no different than the contents of your original dream folder, except for several things:

1. No subfolders: if you had a separate patch folder, all these files will be stored together in the cache folder, be it **td*something*** or **pm*something***.
2. The map filename is renamed to **default**.
3. The DragonSpeak file is translated into two files: **default.dsb** (Binary DragonSpeak file that contains all the numbers in the original file) and **default.txb** (DragonSpeak string index file). The second file will stay on the file server and NOT be sent to other people: messages and other text are handled solely by the server.
4. Map, DragonSpeak and patch files that have the correct names can be in an encrypted form! Patch files that have an unrelated name (i.e.: bubbles.fox) will not be encrypted, thus one should avoid leaving them in the patch folder! Any other file (music, text, ini, ...) are not encrypted even if you set it so.

All relevant patch files are translated into FOX files, so if you have **items.fsh** or **items.fs2** in your patch folder, people will have it as **items.fox**.

What can we, as technicians, do with the cache?

So far, we've learned about cache and how Furcadia uses it, but there is more to it when it comes to helping other people or looking for problems in certain areas. So what can we do with the cache?

Clearing the cache

The most common thing you could do with cache is clear it. There are several issues out there that involve corrupt data being stored in cache or Windows file permissions set incorrectly. There is a simple enough solution to most of them: getting rid of the affected files or folders.

Here is a list of common symptoms and the specific files or folders you should clear to get rid of them:

Symptoms	Cache Elements
Custom portraits show up and then revert back to the "Loading..." image shortly afterwards	Portrait Cache\portraits.howl
A specific **main** dream is crashing Furcadia upon entry, or fills up with objects that shouldn't be there (i.e. pillows or switches everywhere)	Permanent Maps (either find the specific folder within, or delete the entire Permanent Maps)
A specific **player-uploaded** dream is crashing Furcadia upon entry, or fills up with objects that shouldn't be there (like the previous one)	Temporary Dreams (either find the specific folder within, or delete the entire Temporary Dreams)
Dynamic avatars don't show up and players look like rodents instead	Dynamic Avatars

One of the good things in Furcadia cache, is the fact that you can delete any block of information (i.e.: a dream package, an animated portrait, the portrait cache file, etc..) and it will be restored as necessary later on. Be it necessary, you can delete the entire Dragon's Eye Productions folder responsible for the cache and it will be restored anew. This makes troubleshooting cache-related issues a breeze and clearing the entire cache is simple enough for a regular user to do, based on your guidance.

> **Note:** By deleting cache elements, you will make it necessary for Furcadia to re-download the content next time it needs it. In itself, it's not a big deal **unless you have dialup**! Dialup connections will likely operate at 5KB/sec or less, and an intense download can render other internet activity of the user (browsing the web, messengers, etc) near-impossible! As such, it is a good idea to delete as less files as you can in order to resolve the problem: if you can delete just **portraits.howl** to solve a portrait issue, it will be a lot better than deleting absolutely everything. Try being as unobtrusive as you can, and you will seem a lot more professional than those who suggest to reinstall Furcadia for every single problem they encounter.

Analyzing unencrypted dreams for errors

When analyzing dreams of other users, we often have to rely on their word and deal with very little information available to us. One of the things that can greatly help us in figuring out the problem, is knowing where to find the files for that dream in cache. There are several advantages to knowing your way around the cache:

1. You don't need to waste time requesting the user's dream - this is the next best thing (if not better) and you can access it from your own hard-drive.
2. The files in cache represent the files on the server itself, so you can actually see what the Furcadia server and the Furcadia client sees.
3. Having such solid information in your hands, you won't have to rely solely on the user to know what they talk about, or to explain themselves properly - you will be able to answer a lot of your own questions and find the problem faster.

There are two basic ways to find the cache folder of a specific dream:

1. If you have an extended `which command (if you get three lines of text instead of one), you can find the dream package name (and thus, the folder name) in the tribble-related line, after the text "This tribble feels like: ". Just keep in mind that if the package name starts with **pm**, then the cache folder is located in **Permanent Maps**. If the package name starts with **td**, then the cache folder is in **Temporary Dreams**. If all you see is "::", then the dream has no package: it is either a map loaded from your Furcadia folder itself, or a broken dream that no longer has a package associated with it.
2. The most recently downloaded dream will create a cache folder with the most recent "modification date". All you need to do to find the most recent cache folder is right-click an empty area in the folder that contains them, pick **Sort By** and then **Modification Date**. Depending on your sorting (ascending/descending), the most recent cache folder will either be the first or the last one in the list. If you are looking for a dream you have already accessed once which is no longer "recent", you will have to delete all the cache folders and re-download the dream to *make* it recent.

While we won't discuss how to actually analyze and debug dreams through the cache (it's a long subject in itself), being able to do so can make you more successful in finding dream issues and knowing, in general, how Furcadia works.

Semi-live patch modifications

Finally for the least of tricks you can do with cache to make lives easier: did you know that once downloaded and extracted, the dream files in cache don't get validated by Furcadia? This allows us to delete or change files in cache and see the effects of our actions next time we re-enter the dream!

There are several benefits to this ability:

1. People who have a slow upload speed and a big dream (like 4-6MB big) could preview all sorts of minor graphical/music changes without having to re-upload the dream every time! All you need to do is make the changes in the patch files inside the cache and re-enter the dream. This trick has a lot of caveats, so it's definitely not something an inexperienced dreamweaver would want to do, but for those who know, it does give a little bit of power...

Technicians who are trying to find a bad file that crashes Furcadia can simply take files out one-by-one from the cache folder, and see when the dream actually loads for them without crashes - and that without any re-uploads! This can prove to be a faster method than having the (sometimes clueless) owner do it the long way and re-upload several times just to check.

Credits

I wish to extend my thanks to the following people for their contributions (direct or otherwise):
- **Treeki** - for providing path information on Windows 7.
- **Dream Dancer** - for providing path information on Windows 2000.
- **Nijûmi** - for providing path information on MacOS X.

This document was compiled and written by Artex / IceDragon and more information about Furcadia is available at http://ftr.icerealm.org/

Footnotes

There are several things I'd like to point out that are a bit different than what they *should* be in practice. I'll add them as I go:

1. The Cache Size Limitation is not working as of right now - Furcadia might be set to 15MB, but expect the actual size of the cache folder to be much bigger.