parsers

parsers

In hydrus, a parser is an object that takes a single block of HTML or JSON data and returns many kinds of hydrus-level metadata.

Parsers are flexible and potentially quite complicated. You might like to open *network->manage parsers* and explore the UI as you read these pages. Check out how the default parsers already in the client work, and if you want to write a new one, see if there is something already in there that is similar--it is usually easier to duplicate an existing parser and then alter it than to create a new one from scratch every time.

There are three main components in the parsing system (click to open each component's help page):

- **Formulae:** Take parsable data, search it in some manner, and return 0 to n strings.
- **Content Parsers:** Take parsable data, apply a formula to it to get some strings, and apply a single metadata 'type' and perhaps some additional modifiers.
- **Page Parsers:** Take parsable data, apply content parsers to it, and return all the metadata in an appropriate structure.

Once you are comfortable with these objects, you might like to check out these walkthroughs, which create full parsers from nothing:

- e621 HTML gallery page
- Gelbooru HTML file page
- Artstation JSON file page API

Once you are comfortable with parsers, and if you are feeling brave, check out how the default imageboard and pixiv parsers work. These are complicated and use more experimental areas of the code to get their job done. If you are trying to get a new imageboard parser going and can't figure out subsidiary page parsers, send me a mail or something and I'll try to help you out!

When you are making a parser, consider this checklist (you might want to copy/have your own version of this somewhere):

- Do you get good URLs with good priority? Do you ever accidentally get favourite/popular/advert results you didn't mean to?

- If you need a next gallery page URL, is it ever not available (and hence needs a URL Class fix)? Does it change for search tags with unicode or http-restricted characters?
- Do you get nice namespaced tags? Are any unwanted single characters like -/+/? getting through?
- Is the file hash available anywhere?
- Is a source/post time available?
- Is a source URL available? Is it good quality, or does it often just point to an artist's base twitter profile? If you pull it from text or a tooltip, is it clipped for longer URLs?

Revision #1
Created 12 March 2021 16:52:16 by CuddleBear
Updated 12 March 2021 18:24:07 by CuddleBear