caca (Cave Catalogue) ideas and example means to achive them

rev 2, 2013-07-26

Scope of the project

Devise a common framework for keeping track of basic data about caves and cave entrances (coordinates, entrance description, tackle needed, special remarks, history of exploration etc.).

Assumptions (points to necessarily agree upon)

1. Let's NOT try to design a generic catalogue for storing all kind of data about caves of the whole world, inteded for every kind of user (sports, exploration, science). Let's just settle for a generic framework. Let geeks in individual countries or individual communities write their tools operating within this framework.
2. Let's try make it available for the layman, but still well-playable for the geeks.
3. Let's rely on already existing, popular technologies. Let's keep it open source and multiplatform. Let's try not to reinvent the wheel.
4. Let's not assume everyone has an Internet connection while working with their data.
5. Let's version-control as much as possible.
6. Let's support i18n – let's use UTF-8 everywhere and cater for data in many languages (entrance names, cave descriptions, location descriptions etc.)

Proposals (points to discuss)

a) Store things in a file system. Store things remotely in a version control system (say, subversion). Do not use a database to store the original data. Motivation: This way everything can also be used without any special software – you just browse the directories.

b) Use proper XML files for storing cave meta-data. Try to build up on one of the existing formats (eg the HMG format). Motivation: we can use existing XML libraries and facilitate easy import of existing data.

c) Agree upon a small XML specification to store cave meta-data. Cover things such as: cave name, entrances, their names & locations, trip history. Perhaps also: cave descriptions, tackle lists. Leave survey and cave map data out of the scope of the specification. Above all things, devise a good reference system (XML tags for referencing), so that files related to a particular cave (survey notes, survey files, plans, photos, publications) can be listed, categorized and annotated in the XML file.

d) What the Poles would like to have, and what we could commit to do is to write a JavaScript/HTML-based front-end for browsing cave records (XML files), editing them, uploading and annotating files, downloading/uploading/merging caves from/to/with a remote repository. We'd like to make it in a way that it can run both locally and from a remote server.

e) A nice web (on-line) front end could also be prepared, that also indexes the XML files and lets you do nice searches, show cave entrances on top of Google Maps, compile Survex files online and see the cave model rendered through WebGL etc. We'd gladly leave it up to the Islanders, though.

f) There was a discussion whether to group data by caves or by data types (ie. whether the $ROOT data-store directory should contain subdirectories like: cave1/, cave2/, cave3/ or rather surveys/, photos/, descriptions/). Actually I think we could painlessly cover for both approaches. It's just the index (XML) files that have to be well-organized. My proposal:

   1. store XML files under $ROOT/caves/Poland/Tatra/42-G.123.xml (Poland) or $ROOT/caves/49H/12/022.xml (HMG) or $ROOT/caves/1336/049.xml (Austria) – BTW: definitely let the users decide on the depth and general setup of their hierarchies, support anything/anything/anything/cave.xml
   2. store other files either under $ROOT/caves/Poland/Tatra/49-

G.123/surveys/somesurvey.svx or under $ROOT/surveys/Poland/Tatra/49-G.123/somesurvey.svx. Just LINK THEM PROPERLY in the XML files (absolutely to the $ROOT or relatively to the XML file location).

3. put all approaches that we practically use as a configuration option in the front-ends and make it easy for geeks from other countries or caving cultures to add new schemes (selecting a proper scheme is only necessary for entering NEW data; through proper linking, any already entered data will work without the need for proper configuration).

Ideas to work on later or maybe completely throw away

7. Full i18n of the user interface (Chinese, Polish etc.)
8. Devise a clever way of tracking if changes have been propagated to other languages, to show user some warning if their language version of a cave description is behind. Add options to make the life of interpreters easier, eg. search for caves that have modifications not propagated to a particular languages. Or search for caves that lack a certain field in a certain language, despite being defined for other ones.
9. Make sure the fact of connecting a few caves is properly handled by the whole framework.
10. Let the database administrator define any tags that can further be assigned to individual caves, eg. „archaeology/importantsite” or „status/open” or „status/finished” or „access/difficult” or „access/nearroad” or „geology/uppertriasic”. Support searching by tags. Important tags could be assigned icons or just color-coded dots that always show next to a cave name or number. Keep any tagging open, let's not try to negotiate a common set of tags – there are too many differences between the areas, the caving communities etc..
11. Introduce options to also keep area-wide information eg. area description – another XML spec?
12. Options to keep information about expeditions (when the expedition took place, who participated and who when was where) – another XML spec? Automatic generation of expedition chronicles.
13. Automatic drawing of rigging diagrams (maybe just as bitmap files) out of tackle information entered in the XML files :)
14. Exporting GPX files with cave coordinates.