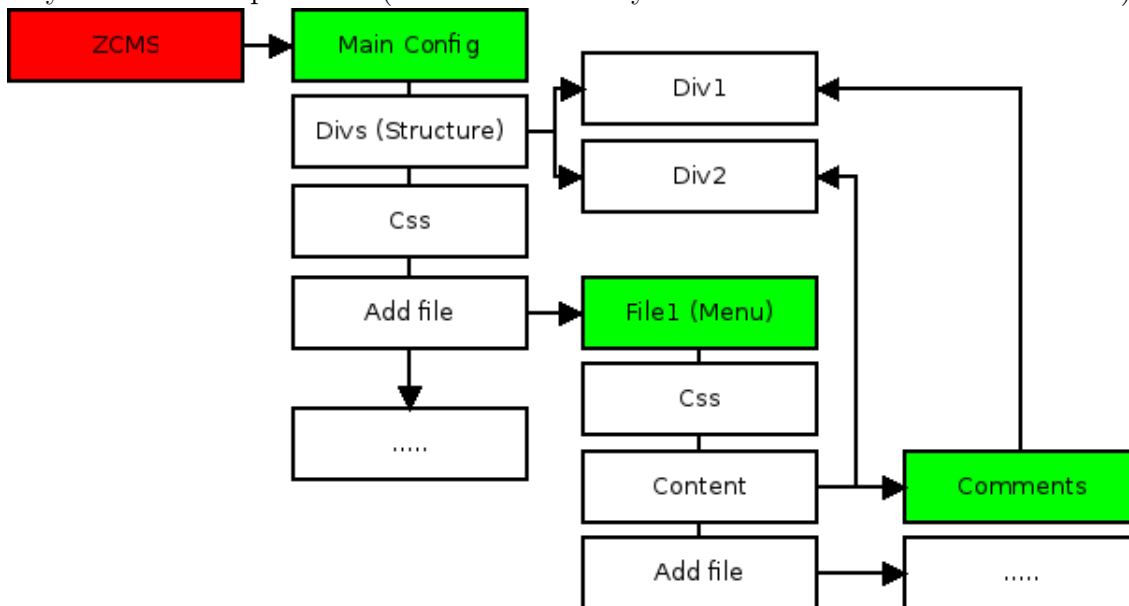# Zcms

1   Introduction

Zcms is a content managent system based almost completely on javascript. It is designed to be lightweight and fast and to use ajax for content loading. Furthermore, it is possible to use mysql, xml or both technologies to store content datas. Building a website with zcms should be as simple as writing a basic xml file.

2   Workflow of Zcms

The first thing zcms does is loading resources/xml/config.xml, which is the main configuration of the site. The main config usually (but not mandatory) loads the css styles, add the layout divs (see Structure section), and sets global variables, like default target or default animation. Thereafter, the main config loads other external files, which can add articles, comments, html, php code or load in turn other files. Every external file must be loaded with a tag parameter that tells zcms which part of the file is to be added. Moreover, a target or a default target previosuly set must be provided (see Files Anatomy - Entries for further informations).

2.1   Menu links

A typical menu link points to index.html#page.xml .Zcms will load automatically page.xml and will add it to the site with ajax.

3   Files Anatomy

Every file is loaded with a tag parameter. Everything outside the tag is ignored and won't be loaded. For php files, the tag must be enclosed in at least one other element. For xml files, the tag can be the ancestor of the document.
There are 2 main sections: config and entries. The content of the site is in the form

of a single or multiple entries added to a target selector.

```
<tag>
  <config>
    ...
  </config>
  <entry>
    ...
  </entry>
  <entry>
    ...
  </entry>
</tag>
```

4   Config

The config tag is useful to load additionals xml,css,php files or to set useful settings.
Available tags are:

- `<css id='css_id'>`

  Load an external css file. The id is used to identify the css in order to remove
  or replace it afterwards.

- `<structure_div>div_name</structure_div>`

  Add a div to the DOM. (See Structure section)

- `<default_target>target</default_target>`

  Set the default target, as a jquery selector, in which entries should be added.
  It is optional, althought an error is returned if no target is defined in external
  files.

- `<animation speed="fast">slide</animation>`

  Set the animation type (slide/hide/fade) and speed(fast/normal/slow or a va-
  lue in ms) when entries are added to the DOM.
  TODO. Animations can't be disabled or enabled globally via config. It should
  be fixed.

- `<home>#home.xml</home>`

  Set the default home page. See links section for informations about links
  meaning.

- `<load_file tag="menu">resources/xml/menu.xml</load_file>`

  Load an external file. Zcms will add the content inside the jquery selector tag.

5   entry

5.1   tab

5.2   Comments

Comments can be enabled or disabled in the single entry using the attribute *comments*={'enabled','disabled'}.

Comments must be located in resources/comments/$entry_id$.xml.

A comment is an entry with *type* attribute set to 'comment'. This ensures that zcms adds author of the comment and date to the entry tab. Additionally, the entry div will have an *id*='$entry_id$_comments' and *class*='comments'.

TODO. ATM configs inside entries are ignored (AFAIK). They can be introduced, allowing to unify tab and comments entries into one more dynamic paradigm.

```
<comments_ $entry_id$  id="$entry_id$" target="#$entry_id$_comments_text" animation="enabled">
        <entry type="comment" id="1" author="m3l7" date="2011-09-07">
                <content>
                        <span>Test comment</span>
                </content>
        </entry>
```

6   Structure

Structure objects can be created using the following methods:

6.1   js methods

- Zcms.Structure.new_div: this will create a simple div. Additional classes can be added.

- Zcms.Structure.new_tab: this will create an entry (**TODO** Rename to new_entry?). An entry is a div with 2 or 3 subdivs: [title], content, comments. Each subdivs has 2 subdivs: _text and _text_right

6.2   xml methods

- $< structure\_div >$: create a div and a subdiv _content. It is created at the end of the current DOM, so it should be used only in the main config. It must be placed inside <config> tag.

- $< entry >$. Create a new tab. See the proper section for further informations.

-