PluginInclude

Inclusion fragility

Problem: when a page is included via PluginInclude, it's possible for a user to unwillingly break the reference.

We need to have something like Wiki page backlinks

Cases

- When renaming a page (so we need a warning tiki-rename_page.php)
- When deleting a page ([https://dev.tiki.org/tiki-wiki-remove_pages](https://dev.tiki.org/tiki-wiki-remove_pages))
- When editing the content (if start and stop params are used)
- Adding a param for a "read more..." button when displaying a part of another page (using params start/stop)

Using the actual plugin it is possible to display part the content of a page.

To display part of the content from the included page you can:

- Use on a marker in the page (ie: )
- Use a line of text (ie: "this is the last line of part one")
- Use any other element present in the page (seems it should be added in the parameter as the complete line)

It would be nice to have a "more button..." that link to the page itself.

Notes

- PluginInclude can be used not just in wiki pages, but in any content. Notably in Textarea Tracker Field (and it needs to work there too)

Potential solutions

- A report of all usages of PluginInclude and an indication of an error
  - Page doesn't exist
  - start and or stop param doesn't exist in that page
- When a page is edited, renamed or deleted, display a warning that this will break an inclusion

Other work we should do while we are at it

Other issues

- Headings in included pages end collapsible sections in parent page
- Table of contents entries (maketoc) in wrong order when wiki pages are included
- Non-parsed part of heading via inclusion rendered as hexadecimal key like &quot;$3b24424f3be6a8f1f0937355e8f37ee88&quot; in table of contents
- TikiWiki 2.0: Content: Display Last Modification Info on Included Pages

Parsing of the target

In Tiki 11 and anterior, the call to the INCLUDE plugin is substituted with the included target's source, and the resulting augmented source is parsed simultaneously. In versions 12 to 18, the included target's source is parsed independently (by a different call to parse_data()), and the resulting HTML is inserted in the including page's HTML. Starting with version 19, both of these behaviors are possible, and controlled by the INCLUDE plugin's "Parse Included Page" (parse_included_page) parameter.
This parameter was added to let users choose between 2 issues:

1. The included page's `is_html` (WYSIWYG) setting would be ignored until Tiki 12, changing the included source's semantics.
2. Table of contents (maketoc) entries can be in the wrong order when wiki pages are included (versions 12 to 18, and later when `parse_included_page` is enabled), as reported in ticket [Table of contents entries (maketoc) in wrong order when wiki pages are included](https://dev.tiki.org/tiki-index.php?page= likewise).

- I believe it makes sense to keep these 2 behaviors even if one can be completely fixed. The parameter allows to control the semantics of an inclusion. For example, if the included page contains the `page` variable, some may want to evaluate to the including page's name (including source), while others may prefer to evaluate to the included page's name (including the result of an isolated parse).

Currently, inclusion as source also creates another issue, reported in ticket [Headings in included pages end collapsible sections in parent page](https://dev.tiki.org/tiki-index.php?page= likewise). One way this could be solved is by considering that if—for instance—a level 2 section in page `Parent` includes page `Child`, then all included sections are considered below level 2. For example, a level 1 section in `Parent` would be considered as level 3, and a level 3 section in `Parent` would be considered as level 5. Without doing that, sections too high can break the including page's TOC.

Test edit from Marc