Part III

Building your analysis on top of WikiTrust

WikiTrust

WikiTrust processes edits in batch or in real-time, and computes author reputation, text trust, text author, and text origin.

WikiTrust -- structure

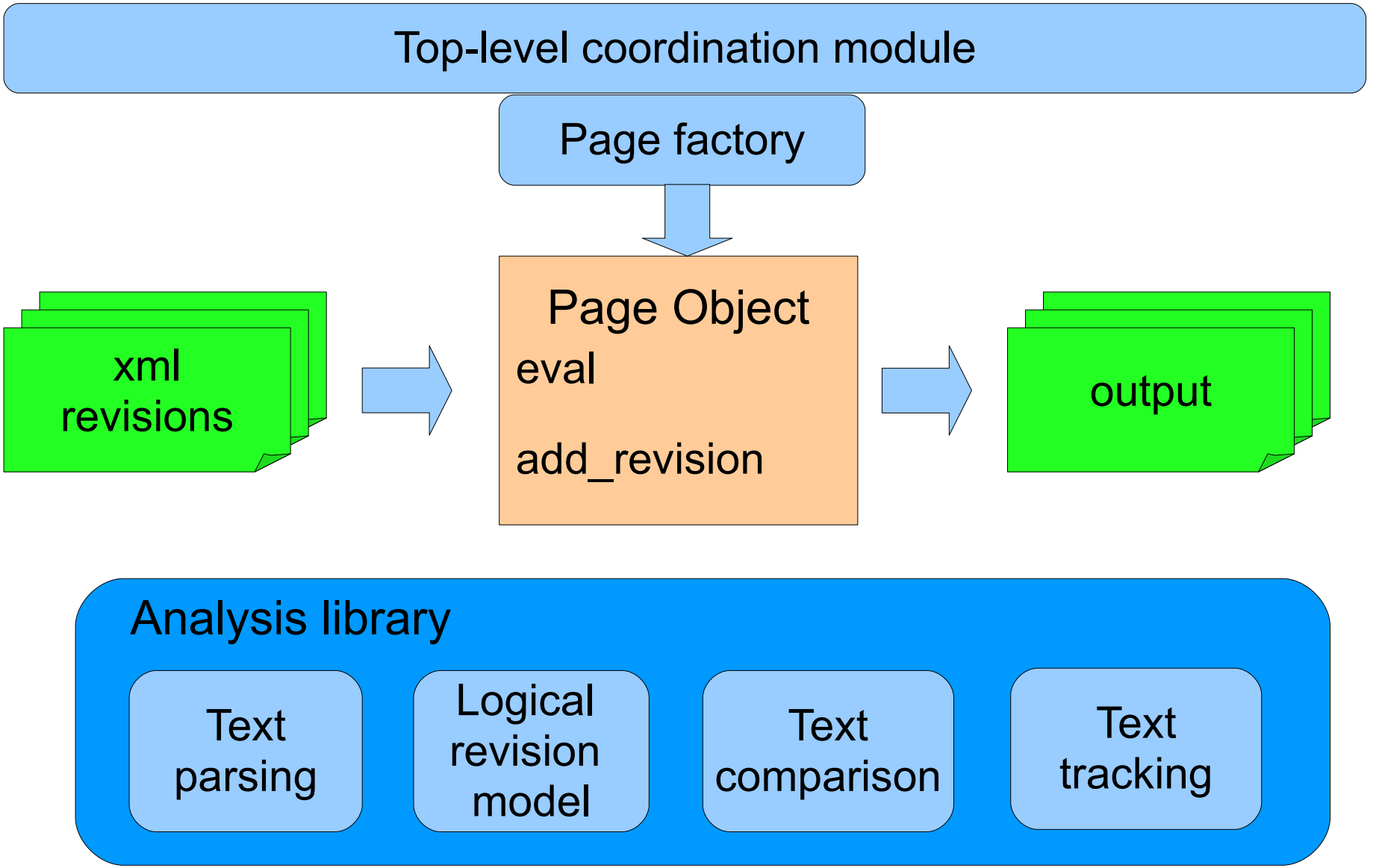
WikiTrust has two modes of working:

- **Online**, as a MediaWiki extension. Whenever a user makes an edit, it updates text trust, origin, and authorship, as well as user reputations. It can then color the text, etc.
- **Batch**, as a series of tools that perform various text tracking and analysis tasks.

General structure of batch mode:

- Input: a compressed xml file containing some pages.
- Output: one, or if you really want, a few, files for each input file.
- Perfectly suited for parallel analysis.

WikiTrust – batch mode logical structure



WikiTrust – Text parsing

Text parsing produces two types of output:

- A list of words, renormalized (lowercased, no punctuation)
- A list of *seps,* which include both words, and syntactic markers (title beginning, title end, bullet, paragraph separator, ...)

The list of seps is a superset of the list of words.

The two lists are cross-linked; you can always go from a word to the corresponding sep, or from the sep to the word (if any).

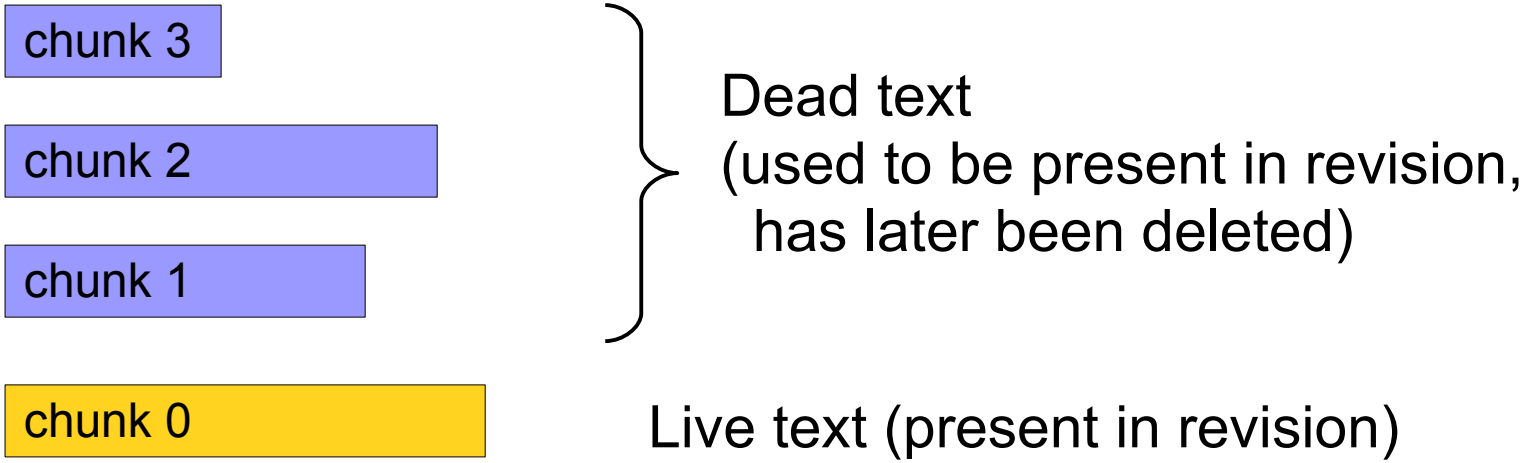
WikiTrust – Text comparison

Chdiff.edit_diff word_array_1 word_array_2 describes the difference between word_array_1 and word_array_2 in terms of a list of:

- Del (k, n) Deleted n words at position k in word_array_1
- Ins (k, n) Inserted n words at position k in word_array_2
- Mov (k, m, n) Moved n words from position k in word_array_1 to position m in word_array_2

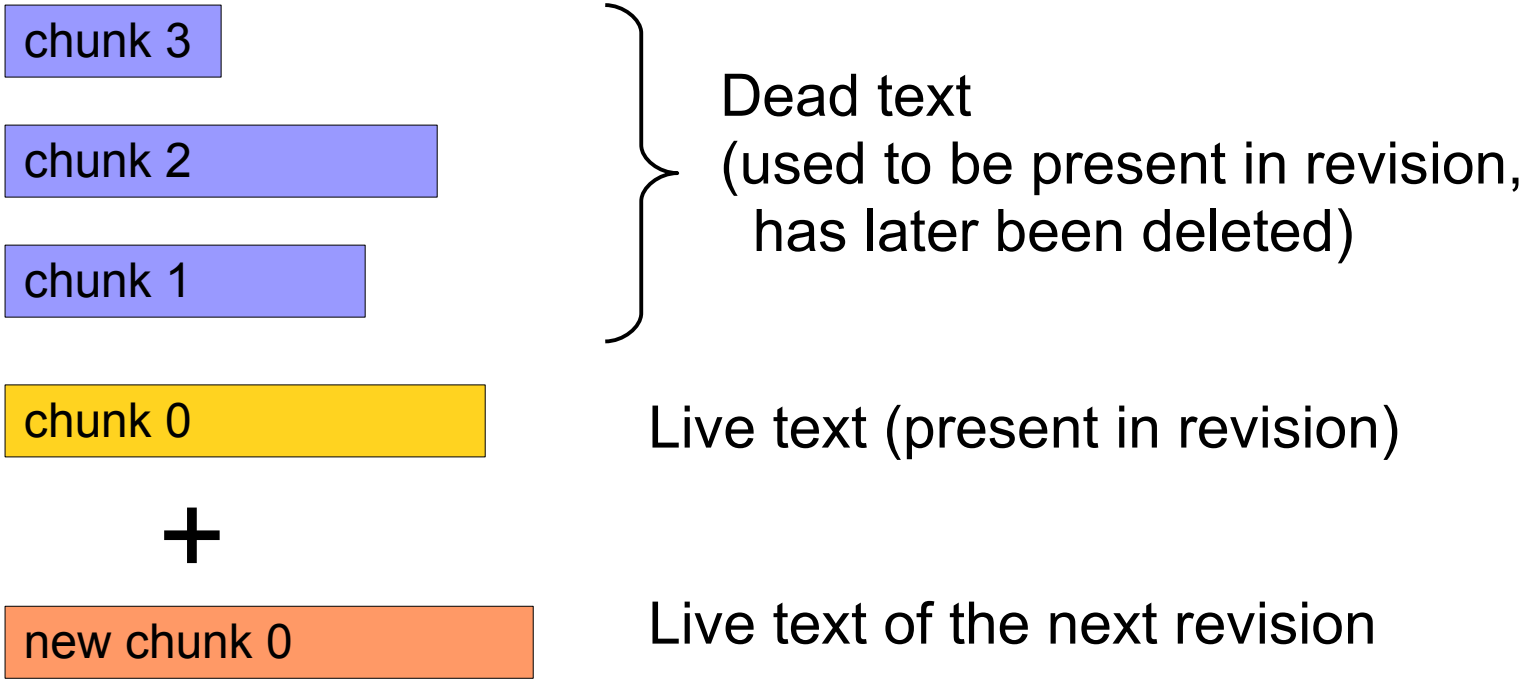
This information enables the computation of edit distances, and also, to find the previous revision most similar to the current one.

WikiTrust – Text tracking

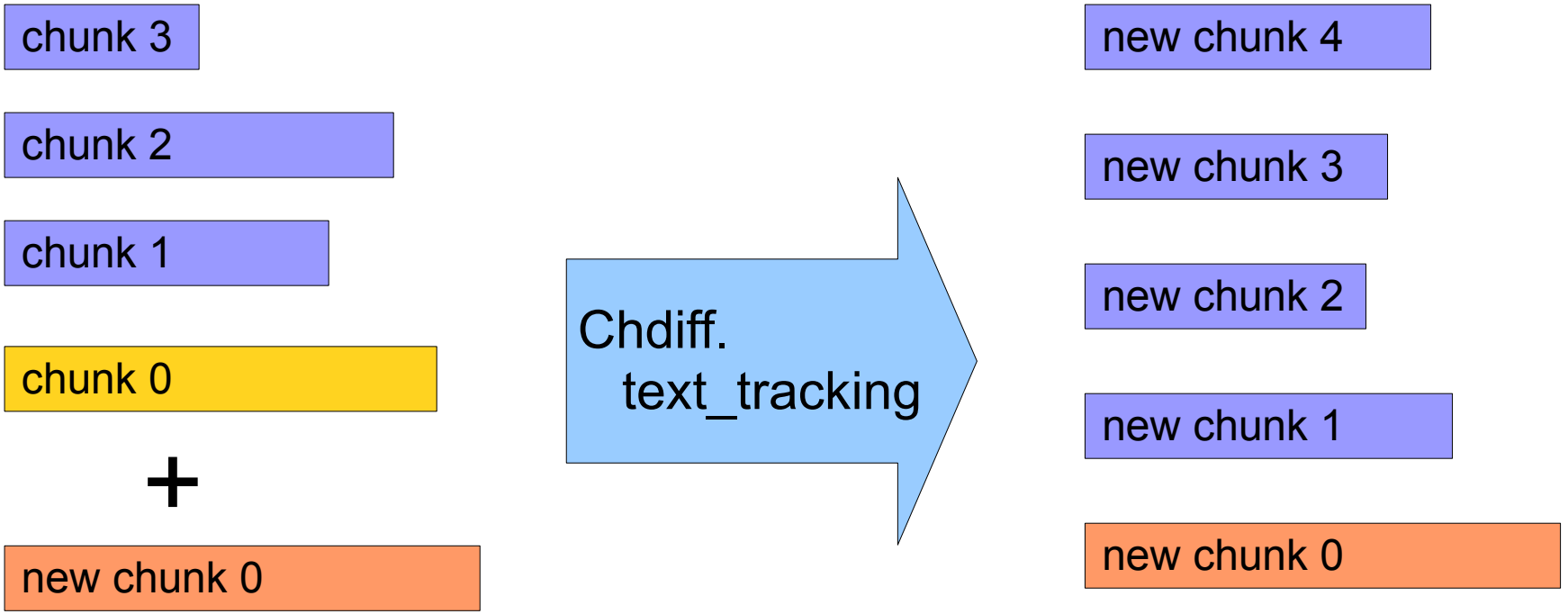


Wikitrust tracks both the text present in a revision, and the text that used to be present in the revision history, but has been subsequently deleted.

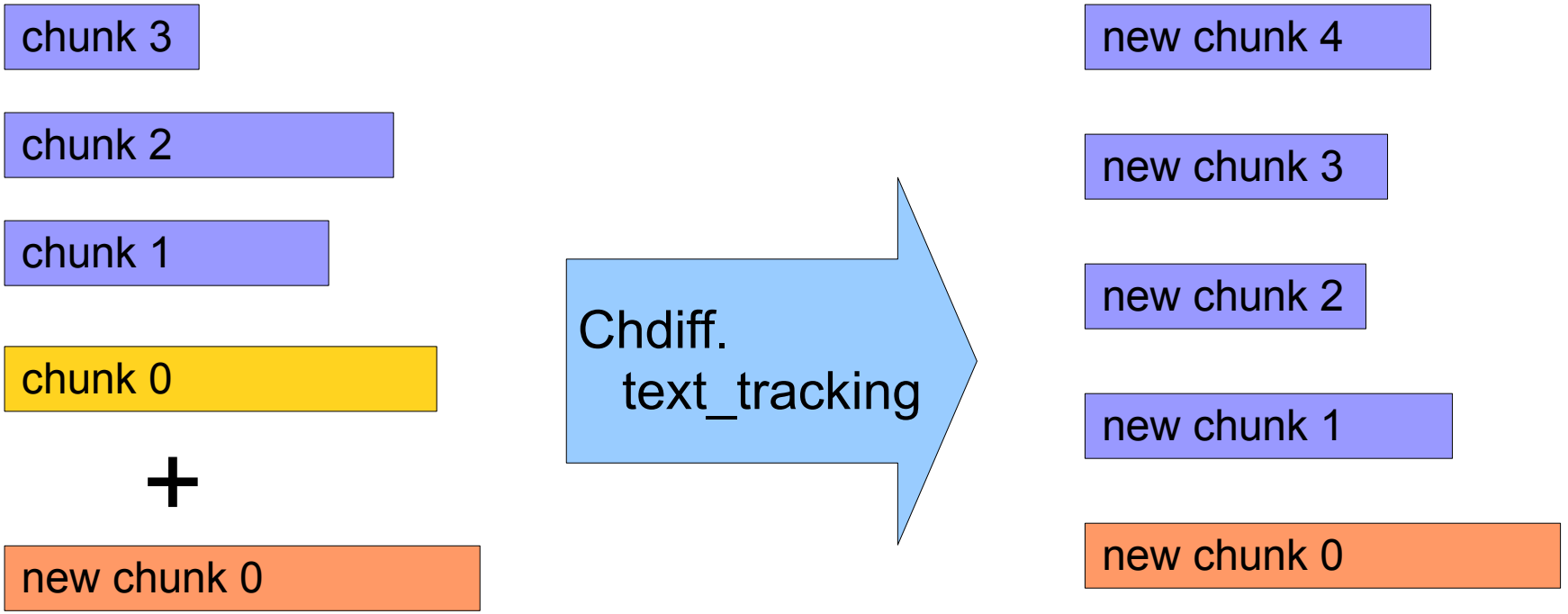
WikiTrust – Text tracking



WikiTrust – Text tracking



WikiTrust – Text tracking



- Mins (k, n) Insert n words at position k of chunk_0
- Mdel (k, c, n) Deletes n words from position k of chunk_n
- Mmov (k, c, m, d, n) Moves n words from position k of chunk_c to position m of chunk_d

Recipe to add a new analysis

To make a new type of analysis, you need to do only two things:

- Create a new subclass of Page, where:
  - method add_revision adds a revision
  - method eval signifies there are no more revisions, and does any last processing.
- Modify page_factory.ml, adding an option to produce objects of the new subclass, whenever a page is encountered.

Example: Let's see how to add an analysis that computes, for each user, the sum of the “live time” of all the words introduced by the user. (The “live time” is the time for which a word is shown).