

Explanation

Observers register a callback on an event. Two specialised forms of callback allow an observer to be notified immediately before the main list so they can prepare, or after the main list so they can tidy up.

An observable raises an event notification by calling `tikiinternaleventlib->raise_event(...)`. The library then calls all the functions in the `pre_event`, `event` and `post_event` lists in turn:

pseudocode:

```
$result = $this->query("select * from `tiki_events` where event like $event order by `callback_type`,
`order`"); $continue = true; while ($continue && $res = $result->fetchRow()) { $class =
$res['object']; $method = $res['method']; if ( is_callable(array(get_class($class), $method)) ) {
$continue = $class->$method($raisedBy, &$data); } }
```

When an observer no longer needs to be notified of an event, it calls `notificationslib->unregister_callback(...)` with the same arguments it used to request a callback.

Discussion/participation

Where ideas can be exchanged, debated, etc. Interested people can subscribe to the wiki page and/or to these forums as they would a mailing list.

IRC log of July 16th 2005 that started the whole thing

[\[+\]](#)

More discussion

- the "user becomes dormant". In 1.10, there is the timeout and the "I am back" for a user in the tiki logs. The delay is still in the code 5mn. [sylvie](#)