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

wholeÂ thing

[+]

IRC #tikiwiki 20050716

[13:47:21] mdavey_ [~chatzilla@82.195.100.25] hat #tikiwiki betreten [13:50:52] q to the developers - is there any tiki code at the moment that provides a callback mechanism? I don't recall seeing anything like that in tikiwiki. [13:50:58] yes, nobody should be able to force you !!! [13:51:12] a callback ? [13:51:45] for what ? and when ? [13:52:14] I'm planning to add a register-callback feature to tiki-registration.php (don't get confused byt he slightly different uses of the word 'register' here) [13:52:49] Hmmm... you mean on given events ? [13:53:00] The idea is that mods will be able to ask tiki-core to be notified when someone new registers with the site, and thus take the appropriate actions. [13:53:07] yes, on events. [13:53:38] as we could say for instance a page change [13:53:50] or some setting in admin ... [13:53:59] yes. or simply logging in. [13:54:13] ah la la [13:54:23] that's future ... [13:55:11] yeah ... a few more db accesses ... [13:55:28] that's becoming a pity on tw.o ... [13:56:08] perhaps something that has been integrated by tiki-integrator needs to provide a seperate login at the moment (such as 3rd party bulletin board) - a callback mechanism would allow a little bit of glue code to be written to also log the user into the bulletin board, when one logs into the site proper. [13:56:54] yes, such mechanism can have many uses [13:57:15] even to communicate with some crons, for instance [13:58:01] My thought was that perhaps some less-used feature was already using its own callbacks - perhaps calendar or blogs? But I don't recall actually seeing such code in my own travels throught the source - so asking other developers for their eyeballs - anything in tiki that does callbacks at the moment? [13:58:04] or simply stack a message ... perhaps ... [13:58:59] so far I remember, not in that "callback" acceptance [13:59:31] something like that could actually save requests too [14:00:38] (db accesses) yea, Id be interested to learn why db accesses on tw.o has jumpted from ~75 to ~300 on homepage recently. [14:00:42] you could imagine to have "your messages", last comments, last changes ... in only one table [14:00:47] *jumped [14:01:04] we are loosing prospects because of that [14:01:14] I hear some noises ... [14:01:36] amette has been thinking of some kind of Tiki Message Bus also... [14:01:38] and never we should need that [14:01:56] that's quite static that whole ! [14:02:20] (if we had events ... perhaps ... right !) [14:02:28] I don't remember, how I came to think about this, though... but e.g. the RSS-generation could subsribe to the

changed_wiki_page message and then generate the RSS - would make the RSS in sync [14:02:40] actually I think just timestamps would help in some places. [14:03:25] but, releasing as is is a suicide [14:03:27] for instance, main menu. If you know the main menu hasn't been changed since march, you don't need to fetch it from the db every time. [14:03:42] voila [14:03:49] yup, same with RSS generation.... you don't need to do anything, if nothing changes... [14:04:20] even if you have lot of connections ... better even normally [14:04:33] should just have been done [14:04:51] but we have some perms set into account [14:05:59] Actually, I think one of the main priorities is to get the mods feature working correctly. Once that is done, every feature could be turned into a mod and then a core profile could install tiki core without any mods at all, and a basic-enabled profile will install the mods that give you an environment like the current basic-enabled profile. [14:05:59] all "events" are re- and re- ... calculated ... perhaps somehow [14:06:45] yes, I'm dreaming of speed.tw.o which would have only a subset of tw.o [14:07:12] ...then we could kick-off a tiki-core project to improve the workings of the core while staying true to the 3rules. [14:07:20] could be I care only of wikis or only of forums [14:07:50] 3rules are no obstacle here [14:08:12] it can be easily made optionnal [14:08:13] toggg yes that would be easy to do with mods and profiles - a tikilite profile that only gives wiki, a tikiforums profile that only gives forums, etc. [14:08:30] for example [14:08:49] but I would perhaps do it direct in code [14:09:05] generating the menu tpl, for example [14:09:31] but, that's bloody, cache should do it [14:09:48] something is wrong here [14:10:02] (optional) true, but if everything has been moved to mods and the profiles install mods as I described, it makes it much easier to make changes without breaking things. [14:10:08] or "ther's something rotten in this kingdom" [14:10:54] why is cache not doing it ? [14:10:55] I think the cache code is probably in the wrong place. If you look outside of tikiwiki, things like Java put the cache at the db abstraction layer. [14:11:48] I mean , should depends on date and perms for example for menu [14:12:40] anyway, you can see that sites without a lot of features are speed already [14:12:47] and tikiwiki is really 2-layer architecture at the moment: frontend (display logic, layout control (smarty), business logic (libs)) and db abstraction, db. [14:12:52] tw.o is quite peculiar [14:13:20] ah la la , only 2 layers ? [14:13:54] moving to a proper 3-layer architecture with the business logic in a seperate layer would help, I think. But then it becomes more heavyweight. [14:14:09] I'd say tikiwiki is 2-layer. Would you disagree? [14:14:57] no, perhaps not ... (if you forget client side) [14:15:09] layers ... [14:15:35] more heavyweight means more temptation to put the different layers on different boxes. That may be a negative from an evaluator's point of view. Makes tikiwiki look more complex than alternatives. [14:16:08] Have you looked in-depth at mambo source? Nice architecture but much slower than tikiwiki 1.8.5 [14:17:09] anyway, I would like to see what are these some 300 requests we have in a page, some trace ... [14:17:59] no, never crawled to deep in mambo, tw and pear me enough

the last monthes [14:19:47] drupal are lucky they got money (from subscription, 10.000$ I think) and a new server (from sun) [14:26:38] I used to work for Sun ;) ... cut some talk about paper vs e-dictionaries, computer cooling/silencing, etc ... ;) [19:21:10] amette, franck can you think of anywhere in tikiwiki code where you have seen a function-callback-on-event mechanism implemented? [19:21:57] callback pattern or publish-subscribe pattern [19:22:15] mdavey_ : sorry, I don't know about anything like that... [19:23:06] aaahhh ... they exist but not in the schedule you mean, mdavey_ [19:23:37] what is a watch ? [19:23:54] yea I want to know if that generic programming pattern is used anywhere in tikiwik at the moment. [19:24:08] cvs co [19:24:18] find grep [19:24:43] but I doubt in the extend you mean [19:24:50] mdavey_ : I'm not deep in tiki-code, but some kind of extensible-message-bus for tiki would be imho very, very welcome - just like you said: makes Mod-development just a task of pluggin in... [19:24:58] watch is slightly different, I believe. Because it isn't a watch feature that has registered an interest in page changes with the editor feature, but rather something that is hardcoded. [19:25:07] make a game to try it, mdavey [19:25:17] the event catcher game [19:25:18] on the other hand - this throws up some serious design questions... [19:25:47] watch is an event [19:26:15] I mean the wiki update is [19:26:20] sorry [19:27:15] but event means here , the action of mailing subscriber is something else as the update [19:27:33] can be done later, anyway [19:27:46] amette thinks that must be done later [19:27:55] is it ? [19:28:07] I doubt [19:28:19] We would need to extend the code for everything.... [19:28:55] Not only checking of permissions and functions, but sending a message with every action.. [19:29:02] toggg Markov diagrams ... Finite State Machines [19:29:09] lovely [19:29:09] Any PHP-file is affected imho.. [19:29:13] no, not really. There are probably only a few events that happen in tiki. Lets try to list them: [19:29:20] ok [19:29:26] 1. user registers [19:29:34] no [19:29:39] 2. user changes user preferences [19:29:43] 100 sorts of it [19:29:47] yup [19:29:51] 3. wiki changes [19:29:51] 3. user changes user information [19:29:54] 3. .wiki created [19:30:01] 4. article added [19:30:01] ah la la [19:30:06] 5. article type added [19:30:12] 5 is allready too much [19:30:14] no, they are all the same. [19:30:20] ah [19:30:28] 4. content created or modified [19:30:28] mdavey_ : still you gotta put them into the code.... [19:30:32] lot of things are same ... [19:30:52] 5 content deleted [19:30:53] ok, let's listen to mdavey_ - sounds like he has got some plan.... [19:31:08] 6. user banned [19:31:25] I've no plans, only FSM [19:31:52] 7. user logged in [19:32:42] 8. user becomes dormant <-- this one might be difficult to code. needed by things like the active users module. [19:32:49] any others? [19:33:06] mdavey_ : hmm, do we think of other things, here?!? [19:33:16] yes [19:33:24] ok, then I just don't get it... ;) [19:33:40] 9. feature enabled [19:33:46] 10 feature disabled [19:33:47] Why are the ones you just listed sufficient? What are you up to? [19:34:20] Kanzaki_Urumi [deathwolf@AToulon-201-1-27-224.w81-48.abo.wanadoo.fr] hat IRC verlassen: Read error: 60

(Operation timed out) [19:34:43] ikkys [~chatzilla@218.208.223.68] hat #tikiwiki betreten [19:34:49] I think that is about it. Some of the wording might need changing a bit. For instance, 4 should probably be content published/changed, 5 should be content unpublished. [19:34:52] I thought of any event creating a message - best even passive ones, like reading a page - so we got all options open for the future.... [19:35:07] helloss [19:35:13] higuys [19:36:11] amette okay, why would a feature need to know if you were reading a page? Not saying you are wrong, just trying to see how you are thinking. [19:37:40] mdavey_ : As a matter of fact, I don't know (that's why I thought: best - probably 'eventually' would have been better). But perhaps/eventually you want to gather some kind of statistics - if we implement such a service, then we should take care, that it can handle anything... [19:38:11] ... I thought of making it optional - pretty much like permission system: generate messages for.... [19:38:23] My plan would be to provide some kind of generic mechanism that would allow tiki mods to ask tiki-core to be notified when an event occurs, allow any code to tell tiki-core about an event, and allow tiki mods to unregister for event notification when done. [19:39:40] tiki-core would maintain a list of events and functions to call when that event happens, and would be responsible for actually performing the callbacks. [19:39:42] Yes, pretty much that was my thought, too - only I thought of making every action capable of generate a message to, say Tiki-core [19:40:14] yup, that's what I thought of, why don't you want to make it as granular as Tiki is at the moment? [19:41:06] okay, same kind of thinking. My initial thought was to use php, but xmlrpc or soap or some kind of message service would work just as well. [19:41:32] On the other hand: we don't have to make that immediately with the first try - when this Tiki-Core is established, it should be only one line of code in every function to be added later on... [19:42:00] Yes, I thought of php, too. But using xmlrpc or soap I think is actually much better [19:42:50] okay, statistics yea I hadn't thought of that. That would be nice - different people could write modules to do different types of graphs, of hits, or browser distribution, most popular pages, etc. [19:43:27] but start with the simple stuff. registration is simple because AFAIK, there is only one route, through tiki-registration.php [19:43:44] login is pretty simple too, a login screen and a login module. [19:45:02] franck [~franck@adsl-67-124-190-154.dsl.lsan03.pacbell.net] hat #tikiwiki verlassen [19:45:04] Yes, why not start with that as a proof of concept! This Tiki-core would have to be modular anyways... [19:46:20] oh nice :) TNT-devs ? [19:46:39] when I say tiki-core, I mean the ill-defined set of files that includes tiki-setup.php, smarty, tikilib, ... [19:46:45] And Tiki-Mods shouldn't only be able to subscribe to Tiki-core, but also register their own message-functions there - and now we really get into some serious design-trouble.... [19:47:19] mdavey_ : yes, we are just ripping it apart to make it a Tiki-Micro-Kernel and add all the other functions as daemons... [19:47:20] TNT is the onliest way [19:47:49] Tri-Nitro-Tuluol? [19:48:13] toluol [19:48:18] something like that, yes, dynamite [19:48:27] ahahah [19:48:47] soft dynamite

[19:48:59] toggg : please explain yourself.... [19:49:30] oh ... that was another nick ... [19:49:53] amette (micro-kernel) yea kinda a web os or web desktop [19:51:08] mdavey_ : yup, that would be cool - and without being a real coder - I think most of the code of Tiki could be reused.... eventually... ;) [19:53:40] but *this* change will be very lightweight - this is still Tiki1 afterall and there are 3rules. Hmm, starting to think that this mythical 3rules is like the Cable that controlls the Internet. Maybe there is no 3rules either? [19:54:53] true :) [19:55:30] but versioning is in praxis, and "renaming"... lets put kernel stuff into tiki2 [19:56:08] http://en.wikipedia.org/wiki/There_is_no_Cabal [19:56:11] toggg [~bertrand@lns-th2-2-lyo-82-249-206-146.adsl.proxad.net] heiÃft nun redadair [19:56:23] Pris [~FU@200.69.24.3] hat #tikiwiki betreten [19:56:23] redadair [~bertrand@lns-th2-2-lyo-82-249-206-146.adsl.proxad.net] heiÃft nun red_adair [19:56:27] WebWeasel [~WebWeasel@CPE-24-208-125-172.kc.res.rr.com] hat #tikiwiki betreten [19:56:42] yes, this all sounds like real tiki2 stuff - but what is tiki 2 anyways....?!? (rhetoric question) [19:57:11] vapourware [19:57:15] lol [19:57:19] this whole talking and maybe acting upon common consensus? or another luis ;) [19:57:28] oh rhetorical, damn ;) [19:57:32] hehe ;) [19:58:11] implementing a demo using login for example would be nice for core-starters ;) [19:58:57] ahahah registered !!! Red Adair :) [20:00:46] lorinc : yes, that's true... still it's some work to do... [20:02:06] toggg : we are not putting out fire - we are starting one [20:07:34] I'm game. I'll start looking into this properly towards the end of next week. [20:08:21] He got game! And I'm in game, too!

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