InterTrac Links

Trac supports a convenient way to refer to resources of other Trac servers, from within the Wiki markup, since version 0.10.

Definitions

An InterTrac link can be seen as a scoped TracLinks. It is used for referring to a Trac resource (Wiki page, changeset, ticket, ...) located in another Trac environment.

List of Active InterTrac Prefixes

Prefix Trac Site

Link Syntax

Simply use the name of the other Trac environment as a prefix, followed by a colon, ending with the resource located in the other environment.

```
<target_environment>:<TracLinks>
```

The other resource is specified using a regular TracLinks, of any flavor.

That target environment name is either the real name of the environment, or an alias for it. The aliases are defined in `trac.ini` (see below). The prefix is case insensitive.

When a TracLink? is enclosed in brackets, the alias is expanded. That is, `[th:ticket:1]` is displayed as `trachacks:ticket:1`, since `th` is an alias for `trachacks`. If the TracLink? is not enclosed in brackets, then the alias is displayed as it is typed.

For convenience, there's also some alternative short-hand form, where one can use an alias as an immediate prefix for the identifier of a ticket, changeset or report: (e.g. `#T234`, `[T1508]`, `[trac 1508]`, ...)

Examples

It is necessary to setup a configuration for the InterTrac facility. This configuration has to be done in the TracIni file, `[intertrac]` section.

Example configuration:

```
...
[intertrac]
# -- Example of setting up an alias:
t = trac

# -- Link to an external Trac:
trac.title = Edgewall's Trac for Trac
trac.url = http://trac.edgewall.org
```

The `.url` is mandatory and is used for locating the other Trac. This can be a relative URL in case that Trac environment is located on the same server.

The `.title` information will be used for providing an useful tooltip when moving the cursor over an InterTrac links.

Finally, the `.compat` option can be used to activate or disable a *compatibility* mode:

- If the targeted Trac is running a version below 0.10 (r3526 to be precise), then it doesn't know how to dispatch an InterTrac link, and it's up to the local Trac to prepare the correct link. Not all links will work that way, but the most common do. This is called the compatibility mode, and is `true` by default.
- If you know that the remote Trac knows how to dispatch InterTrac links, you can explicitly disable this compatibility mode and then *any* TracLinks can become an InterTrac link.

Now, given the above configuration, one could create the following links:

- to this InterTrac page:
  - `trac:wiki:InterTrac` trac:wiki:InterTrac
  - `t:wiki:InterTrac` t:wiki:InterTrac
  - Keys are case insensitive: `T:wiki:InterTrac` T:wiki:InterTrac
- to the ticket #234:
  - `trac:ticket:234` trac:ticket:234
  - `trac:#234` trac:#234
  - `#T234` #T234
- to the changeset [1912]:
  - `trac:changeset:1912` trac:changeset:1912
  - `[T1912]` [T1912]
- to the log range [3300:3330]: (**Note: the following ones need `trac.compat=false`**)
  - `trac:log:@3300:3330` trac:log:@3300:3330
  - `[trac 3300:3330]` [trac 3300:3330]
- finally, to link to the start page of a remote trac, simply use its prefix followed by ':', inside an explicit link. Example: `[th: Trac Hacks]` (*since 0.11; note that the* remote *Trac has to run 0.11 for this to work*)

The generic form `intertrac_prefix:module:id` is translated to the corresponding URL `<remote>/module/id`, shorthand links are specific to some modules (e.g. #T234 is processed by the ticket module) and for the rest (`intertrac_prefix:something`), we rely on the TracSearch#quickjump facility of the remote Trac.

See also: TracLinks, InterWiki