

## Base Distribution - Work Package # 45088: Improved REST support

<b>Status:</b>	Accepted	<b>Priority:</b>	Should have
<b>Author:</b>	Bastian Waidelich	<b>Category:</b>	General / Project
<b>Created:</b>	2013-02-01	<b>Assignee:</b>	Bastian Waidelich
<b>Updated:</b>	2013-02-16	<b>Due date:</b>	2013-03-29
<b>Subject:</b>	Improved REST support		
<b>Description:</b>	<p>h1. Improved REST support</p> <ul style="list-style-type: none"><li>* Target Audience: *developers* who want to use Neos</li><li>* Responsible: "Bastian Waidelich (click to contact)":<a href="http://forge.typo3.org/users/61">http://forge.typo3.org/users/61</a></li><li>* Implemented by: "Bastian Waidelich (click to contact)":<a href="http://forge.typo3.org/users/61">http://forge.typo3.org/users/61</a>, "Andreas Förthner (?) (click to contact)":<a href="http://forge.typo3.org/users/19">http://forge.typo3.org/users/19</a></li><li>* Amount: 18 work days</li><li>* Version: *must have* for 1.0</li><li>* Planned Implementation Timeframe: week 7 to 14, 2013</li></ul> <p>h2. Motivation</p> <p>A solid webservice interface provides countless possibilities. With a proper foundation it would easily be possible to expose the TYPO3 Content Repository as service for instance. This would allow other systems to interact with arbitrary nodes (e.g. mobile clients creating pages, a simple plugin creating content nodes, ...).</p> <p>Another use case are custom service APIs for the clients business logic. Flow already provides basic support for that, but there are some important parts missing.</p> <p>h2. Goal</p> <p>The goal is to have a solid foundation to easily provide custom REST APIs. What has been started at #37604 will be used as base for this.</p> <p>h2. Deliverables</p> <ul style="list-style-type: none"><li>* *MUST*: more flexible request handling</li><li>** Reworked argument merging #45290 - this is required in order to fix argument mapping with non-GET requests</li><li>** More flexible parsing of body arguments #45293 - this is required in order to support custom request formats</li><li>* *MUST*: Routing improvements</li><li>** Global route constraints #27117 - this is required in order to match requests based on the request method and/or arbitrary request headers</li><li>** Nested sub routes #43966 - this is required in order to easily allow developers to include "RESTful" routes</li><li>* *MUST*: session-less authentication #45282 - currently Flow creates a session for every authentication, this is incompatible with REST</li><li>* *MUST*: solid test coverage</li><li>* *MUST*: documentation</li><li>* *COULD*: improved error handling - currently exceptions are rendered independently from the requested format (HTML for all web requests)</li><li>* *COULD*: exemplary (mobile) client that interacts with the TYPO3CR</li></ul>		

### History

#### 2013-02-07 11:02 - Andreas Förthner

Just a short comment regarding the session-less authentication: This is already possible as long as none of your configured authentication providers starts a session (@Flow\Session(autoStart=true) at the authenticate() method). We simply need a provider without this annotation and no session

should be started/needed.

**2013-02-07 12:05 - Bastian Waidelich**

Andreas Förthner wrote:

> Just a short comment regarding the session-less authentication: This is already possible as long as none of your configured authentication providers starts a session [...]

I don't think so, I already found 3 places where Flow relies on a session to be active ("AuthenticationProviderManager::emitAuthenticatedToken()", "AuthenticationProviderManager::isAuthenticated()" and "RequestDispatchingAspect::setInterceptedRequest()"). For the latter Christopher already came up with a solution: <https://review.typo3.org/#/c/17967/>

**2013-02-08 17:29 - Bastian Waidelich**

- *Subject changed from [WIP] Improved REST support to Improved REST support*

**2013-02-12 14:25 - Sebastian Kurfuerst**

Discussion during EAB / Neos meeting:

\* Bastian estimates that 25% of this whole package is documentation.

\* the documentation also should include easy-to-run examples (f.e. how to create pages using the REST API); f.e. with curl or guzzle

\* proper REST support would also be the basis for implementing e.g. CMIS standard later on

**2013-02-16 14:59 - Sebastian Kurfuerst**

- *Status changed from New to Accepted*

**2013-02-16 15:11 - Sebastian Kurfuerst**

- *Due date set to 2013-03-29*

- *Start date changed from 2013-01-31 to 2013-02-16*

- *Estimated time set to 108.00*

Has been discussed with Jan-Hendrik from the EAB on 12.02.2013; and has been \*accepted\* on 15.02.2013 by the EAB.

(The attached PDF is a snapshot of this work package at the time of acceptance.)

So, Bastian and Andi, you can schedule this work package in your day-to-day calendar and implement it then -- can't wait to see it in Neos 1.0 :-)

Another side-note: Please do not forget to add comments to this work package and update "% done" while you are working on this package.

Greetings, Sebastian