TYPO3.Flow - Major Feature # 58184

Status:	New	**Priority:**	Should have
Author:	Carsten Bleicker	**Category:**	Http
Created:	2014-04-25	**Assigned To:**	
Updated:	2014-04-25	**Due date:**	
PHP Version:			
Has patch:	No		
Complexity:			
Subject:	HTTP request argument building for different use cases		

Description

HTTP Request

The concept of http request handling is imho currently not realy http compatible.
First of all, there is no body possible for PATCH request types. Its simply skipped.
Body is currently only avalable for POST and PUT.

But there are additional issues.
Lets say f.e. you want to update the sourcecode of you logo.gif.
So an http request arrives with $_GET[filename]=logo.gif
and a Post with the sourcecode. request type will be image/gif.
request method will be PUT.

The body will be handled with: parse_str($body, $arguments);
@see Packages/Framework/TYPO3.Flow/Classes/TYPO3/Flow/Http/Request.php:560
wich will fail.

Another example wich will be not possible:
Rename the logo.gif to logo_old.gif.
What will arrive is a request type PATCH (if above patch support enabled)
you run in other problems. $_GET[filename]=logo.gif will be an identifier in this case.
body arguments of this PATCH request containing also „&filename=logo_old.gif“.
So we end up in a request just holding request->getArgument('filename') === logo_old.gif.
The get information is lost :(

Maybe you say now „thats not a patch case“.
So lets take a look at it in case of a COPY request type:
get: filename=logo.gif
post: filename=logo_copy.gif
requestmethod: COPY
what you want to copy (the resource logo.gif) is lost also in this case because of the merging of
get and post. @see Packages/Framework/TYPO3.Flow/Classes/TYPO3/Flow/Http/Request.php:513

I want to discuss to make a difference like this:
$this->request->getBody()->getArguments('filename');
$this->request->getResource()->getArguments()->getArgument('filename');

Example 3:
Receiving http request with multipart body.
This could end up in:
$this->request->getBody()->getArguments('part1');

```
$this->request->getBody()->getArguments('part2');
$this->request->getBody()->getArguments('part3');
```

Validation:
Different kinds of requests types should invoke different kind of validations:
1. POST/PUT validates the whole entity (resource)
2. PATCH validates only those properties wich realy should be patched (present in the body arguments)
3. COPY does not invoke validation because an existing resource is valid
4. DELETE should also not invoke validation because an existing resoure is valid.

kind regards
carsten

History