TYPO3.Flow - Feature # 3312

Status:	Needs Feedback	**Priority:**	Should have
Author:	Robert Lemke	**Category:**	Log
Created:	2009-05-15	**Assigned To:**	Robert Lemke
Updated:	2010-10-20	**Due date:**	
PHP Version:			
Has patch:			
Complexity:			
Subject:	Allow for easy logging by annotations		

Description

For debugging but also other purposes, it would be nice to be able to log a method call by just adding an annotation to it.

This logs any call of the method into the SystemLogger with severity level "DEBUG" and stores the argument values in the additionalData arg:

```
/**
 * A method
 *
 * @param string $someArgument
 * @return string
 * @log before
public function myMethod($someArgument) {
   ...
}
```

This is identical:

```
/**
 * A method
 *
 * @param string $someArgument
 * @return string
 * @log before (logger = SystemLogger)
public function myMethod($someArgument) {
   ...
}
```

Here the method invokation is logged to a logger with identifier "MyLog", with severity WARNING

```
/**
 * A method
 *
 * @param string $someArgument
 * @return string
 * @log before (logger = MyLog, severity = WARNING)
public function myMethod($someArgument) {
```

```
    ...
}
```

This logs the method call with a custom message:

```
/**
 * A method
 *
 * @param string $someArgument
 * @return string
 * @log before (message = "Called my method with $someArgument")
public function myMethod($someArgument) {
    ...
}
```

This logs the method call and the returning:

```
/**
 * A method
 *
 * @param string $someArgument
 * @return string
 * @log around
public function myMethod($someArgument) {
    ...
}
```

So basically you have the same types like for AOP advices:

- @log before
- @log after
- @log afterReturning
- @log afterThrowing
- @log around

History

#1 - 2009-05-15 15:17 - Karsten Dambekalns

In what way (if any) will arguments and/or return values be logged? The example with the custom message includes a variable, but what happens in other cases? And how will objects / arrays be formatted for logging?

#2 - 2009-05-15 15:28 - Bastian Waidelich

Maybe method arguments and return values could be passed as $additionalData to the logger by default. Than the logger would decide how to represent objects/arrays.

#3 - 2009-05-25 09:59 - Christopher Hlubek

I think objects could be formatted with __toString() for logging. Arrays should be displayed like [foo, bar], where each argument would be formatted as a string.

Regarding the variables: we will need special ones for the return value (around, after and afterReturning) and the thrown exception (if any).

I would like an additional log type @log time to log the time spent in this method. This could be used to do live profiling of certain methods (e.g. controller actions) in production.

#4 - 2009-07-13 11:25 - Robert Lemke

- Target version deleted (283)