TYPO3.Flow - Feature # 459

Status:	Closed	**Priority:**	Should have
Author:	Robert Lemke	**Category:**	Object
Created:	2008-04-16	**Assigned To:**	Robert Lemke
Updated:	2010-10-20	**Due date:**	
PHP Version:			
Has patch:			
Complexity:			
Subject:	Implement dependency injection via abstract factory methods		

Description

Dependency injection works fine so far as long as components in a singleton scope are requested. Protoype components can't be injected as soon as more than one instance is needed or the requesting class / method needs to have control over the constructor arguments (which is very often the case).

Therefore consumer code currently calls $this->componentManager->getComponent('NameOfProtoypeComponent') in order to get a fresh instance.

This can be optimized by moving the instantiation into an (abstract) factory method:

```
    abstract protected function createSomeComponent($argument1, $argument2);
```

This method should be detected by the component manager and automatically be implemented by creating a proxy class.

History

#1 - 2008-07-01 15:35 - Robert Lemke

- Target version deleted (45)

#2 - 2008-07-23 14:16 - Robert Lemke

- Status changed from Accepted to Closed

- Estimated time set to 0.00

Dropped the idea for now (see mailing list discussion about component factory).