

## TYPO3 Core - Feature #82619

### Equivalent of old DatabaseConnection hooks in the new DBAL

2017-09-29 11:23 - Saskia Schreiber

<b>Status:</b>	New	<b>Start date:</b>	2017-09-29
<b>Priority:</b>	Should have	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>	Database API (Doctrine DBAL)	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>Complexity:</b>	
<b>PHP Version:</b>		<b>Sprint Focus:</b>	
<b>Tags:</b>			
<b>Description</b>			
Is there a chance to get an equivalent to the old postProcessAction hooks in the new Connection class?			
I'm aware of the ConnectionPool wrapperClass feature, however this is not a solution extensions could use (think of search indexers, third-party interfaces etc.).			
Doctrine itself also does not provide an entry point - only via Loggers, quite uncomfortable as the query is already parsed there.			

#### History

##### #1 - 2017-09-29 11:24 - Mathias Brodala

- Subject changed from Equivalent of old DatabaseConnection hooks for the new Connection class to Equivalent of old DatabaseConnection hooks in the new DBAL

##### #2 - 2017-09-29 11:25 - Saskia Schreiber

- Description updated

##### #3 - 2017-09-29 11:26 - Mathias Schreiber

Can you describe the usecase in a bit more detail?

##### #4 - 2017-09-29 11:56 - Saskia Schreiber

Mathias Schreiber wrote:

Can you describe the usecase in a bit more detail?

In my case, I need to track inserts/updates to the database so I can later update the corresponding records in Elasticsearch (it's a search extension). Updates could be made through DataHandler, Extbase or even a custom QueryBuilder query, so the Connection class is the only point where I can be sure to track everything.

As I mentioned, I previously used the DatabaseConnection hooks in TYPO3 7 and was quite surprised that there is no way to do the same thing in V8.

##### #5 - 2018-04-05 20:02 - Franz Koch

I'm facing the very same issue as Saskia. I need to migrate a 7.6 based website to 8.7 and heavily depend on the old hooks in order to apply custom permission checks for which I have to alter the Query itself.

Here is a description to my scenario:

I have like 10 different extensions that provide content. This content is provided in several ways to customers of my client, so that they can enrich their website with it. The content is grouped into various packages (multiple packages by content type) by my client, and his customers can book/rent those packages. The underlying booking logic is generic and can be applied to any extension (regardless of custom ones or ones from TER). Until now, I've been using the hooks in DatabaseConnection to manipulate the queries on the fly, when booking restrictions had to apply. For this, I added JOINS and additional constraints to only return allowed content that has been booked by the customer. And since queries could happen from either Extbase or the Typo3 backend (flexforms, ...), the lowest common entry point have been the hooks in DatabaseConnection and this worked just fine. Right now I have no way to achieve any of it.

To get this functionality back, I think it would be sufficient to add a "pre" and "post" hook to QueryBuilder->execute() and pass the QueryBuilder object along for manipulation. For now, I guess I have to XCLASS and add those hooks/signals myself. If you guys think those hooks would also be suitable for the core, I could create a PR.