

TYPO3 Core - Feature #89015

Lazy Loading for TCA / Backend View to improve performance

2019-08-26 15:45 - Alex Nostadt

Status:	New	Start date:	2019-08-26
Priority:	Should have	Due date:	
Assignee:		% Done:	0%
Category:	Performance	Estimated time:	0.00 hour
Target version:	Candidate for Major Version	Complexity:	
PHP Version:		Sprint Focus:	
Tags:	tca, performance, lazy loading, large-site		
Description			
Hi guys, especially when having records with several fields and tons of records TCA / Backend View can become extremely slow, even more on dev systems which does not have the power of the production system. When there's the need to edit such a record (in dev mode) load time can be very long.			
Related issues:			
Related to TYPO3 Core - Epic #93547: Collection of problems with large sites Accepted 2021-02-19			

History

#1 - 2019-08-26 15:52 - Alex Nostadt

- Description updated

#2 - 2019-08-26 15:52 - Alex Nostadt

- Subject changed from Lazy Loading for TCA to improve performance to Lazy Loading for TCA / Backend View to improve performance

#3 - 2019-08-26 15:58 - Alex Nostadt

- Target version set to Candidate for Major Version

#4 - 2020-02-20 09:02 - DANIEL Rémy

Hello

Thanks for taking care of creating an issue.

Could you provide more context to the performance issue that you have?
Especially TYPO3, php and database version.

Also, an example extension, with the mentioned TCA would be helpful.

And an screenshot of the browser's devtools showing the rendering time of the page (to see if it is a server issue or a browser rendering issue).

Cheers.

#5 - 2020-02-27 12:51 - Alex Nostadt

Providing TCA examples etc. would require more time as I do not want to provide the TCA which is used in a customer project. As I have time, I provide a demo ext. along the Dev-Tool Screenshot. I keep that in mind and hope to be able to provide that somewhen soon. Thus I guess it will not happen within the next months. I am very much up for this feature, yet have other higher priorities atm.

In the meantime I provide a more in depth description:

We have a Model "Campaign" which has 5 tabs. In the general tab are a few simple fields **and** relation to an other Model "CampaignValue". Editors can CRUD "CampaignValue"-Records to schedule Record-Updates. It also includes a "CampaignValue"-History-Table. As manual changes of "Campaign" leads to a new "CampaignValue"-Entry this list can become quite long over the time. Further, each Campaign is associated to a "Product" and in the Campaign selected "Product"-Data can be overwritten. Nowadays a lot of fields can be overwritten. These fields are RTE, IRRE, 1:n relations and simple fields such as input. The Model Product has a bunch of tabs and relations either, yet I guess the Campaign example is fine.

I hope so far you get what I am talking about.

All the data needs to be loaded, yet not every may be required when we want to edit a record. Let's say we only want to change a few properties, only these would require being loaded. So, for example all of these are on the first tab, then only the first tab is loaded upon opening the record. Then we do our changes, hit save and are good to go.

Of course, if there're displayConds, these needs to be considered by a lazyload feature. I assume there's more to consider. I rate the complexity to medium / hard.

We're also making quite heavy usage of the bookmark-feature to access our test records quickly. So, we end up in the regular TCA Edit-View always. Hence, I am aware of the option to edit selected properties via TCA-List-View.

Here are a few ideas:

- Lazyload needs to be enabled per table.
 - Lazyload can be enabled for certain fields
 - Lazyload can be enabled for tabs
-

An other reason for this feature:

T3 BE allows editor to work on their mobile phones already. Loaded content can be reduced and therefore editors can work more fast via mobile phones.

#6 - 2020-03-03 08:48 - Christian EBI

Do you have more info about the structure of the TCA and their inline elements?

For example, one known performance bottleneck happens when you have a TCA table with 1:n IRRE relations to another TCA table and that child tables "label" in the TCA isn't a simple db field of the same table, but for example the relation field to its parent. Something like:

```
// TCA config of the 1:n irre child
return [
    'ctrl' => [
        'label'                => 'title',
        'label_alt'            => 'tt_content', // the relation field to its parent table
        'label_alt_force'.    => true,
        'sortby'               => 'sorting',
    ],
];
```

TYPO3 would then automatically look up the label of the parent table from the "tt_content" relation field and show it in the TCEForm for the children elements. But this could easily turn into a performance bottleneck in several cases. For example, the "l10n_parent" selects would then all need create many similar db calls to fetch these labels from the foreign table and this would easily blow up the number of db calls in the backend for each new 1:n irre record that is added.

#7 - 2020-11-24 13:56 - Sybille Peters

- *Tags changed from tca, performance, lazy loading to tca, performance, lazy loading, large-site*

#8 - 2021-02-19 07:00 - Sybille Peters

- *Related to Epic #93547: Collection of problems with large sites added*