pt_extlist - Bug #50819

Fetching of properties from childelements of type ObjectStorage not possible

2013-08-06 13:34 - Michael Bakonyi

Status:	Needs Feedback	**Start date:**	2013-08-06
Priority:	-- undefined --	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			

Description

With T3 6.1.2 and pt_extlist 1.2.0-dev it is not possible to fetch properties of type ObjectStorage if extbase is chosen as databackend.

E.g. if:

```
    backendConfig < plugin.tx_ptextlist.prototype.backend.extbase
    backendConfig {

      dataBackendClass = Tx_PtExtlist_Domain_DataBackend_ExtBaseDataBackend_ExtBaseDataBackend
      dataMapperClass = Tx_PtExtlist_Domain_DataBackend_Mapper_DomainObjectMapper
      queryInterpreterClass = Tx_PtExtlist_Domain_DataBackend_ExtBaseDataBackend_ExtBaseInterpreter_ExtBaseInterpreter
      repositoryClassName = Vendor\Medialibrary\Domain\Repository\EntryRepository

      dataSource {
      }

      tables (
      )

      baseFromClause (
      )

      baseWhereClause (
      )

      baseGroupByClause (
      )
    }

    fields {
      title {
        # reference to object created by "repositoryClassName" 
        table = __self__
        # property of model
        field = title
      }

      categories{
        table = sys_category
        field = title
      }
    }
```

is set the error

```
Uncaught exception 'Exception' with message 'Trying to call non-existing method getSys_categories on Vendor\Medialibrary\Domain\Model\Entry ' in /bla/typo3conf/ext/pt_extlist/Classes/Domain/DataBackend/Mapper/DomainObjectMapper.php:172
```

appears. So it tries to use the table-name for the getter-method – I guess because it can't fetch the extbase-domain. If the TS-config "table" is set to "categories" (a property from my own Domain referencing the ObjectStorage) pt_extlist complains:

```
Uncaught exception 'Exception' with message 'Trying to get a property title on a domain object of type TYPO3\CMS\Extbase\Persistence\ObjectStorage that does not implement a getter for this property.
```

You can reproduce the behavior by using the default template "Demolist_ExtBase_02" – at which I'm a little bit irritated additionally regarding

```
            groupuid {
                 table = usergroups
                 field = uid
            }

            grouptitle {
                table = usergroups
                field = title
            }
```

as the tables for the groups is "fe_groups" ..?

History

#1 - 2013-08-13 17:14 - Michael Knoll

- Status changed from New to Needs Feedback

- Priority changed from Must have to -- undefined --

Your problem is in the definition of fields:

```
    fields {
      title {
        # reference to object created by "repositoryClassName"
        table = __self__
        # property of model
        field = title
      }

      categories{
        table = __self__
        field = title
      }
    }
```

You can only access properties of the domain model returned by the repository in use. If you need other properties, you have to implement getters therefore, e.g. getTitle on the EntryDomainModel.

Hope this helps!
should be

```
    fields {
      title {
        # reference to object created by "repositoryClassName"
        table = __self__
        # property of model
        field = title
      }

      categories{
        table = __self__
        field = title
      }
    }
```

#2 - 2013-09-04 10:06 - Michael Bakonyi

Ok, thx for your feedback, I think now I understand.

What I still don't understand then the TS I posted already:

```
groupuid {
  table = usergroups
  field = uid
}
```

If it's not possible to reference other tables/ repositories how come this line is included in "Demolist_ExtBase_02"?

Additionally the table "usergroups" doesn't exist by system-default and there's also no mapping (like usergroups = fe_groups) somewhere in the TS.