Doctrine.ORM - Bug # 33921

Status:	Closed	**Priority:**	Must have
Author:	Martin Lipp	**Category:**	
Created:	2012-02-14	**Assigned To:**	Karsten Dambekalns
Updated:	2012-08-31	**Due date:**	
Subject:	Doctrine's removeElement() deletes OneToMany objects without repository update		

Description

The Doctrine ArrayCollection removeElement($object) function forces the deletion of $object and all of its childs out of the database without calling any repository functions. This only occurs if the property i want to remove is defined as an OneToMany assiocation.

It seems to happen because the orphanRemoval attribute is set to true somewhere (default value should be false, i could not track down where the attribute is set to true) or cause the condition in the Doctrine\ORM\PersistentCollection removeElement() function is wrong:

```
if ($this->association !== null && $this->association['type'] == ClassMetadata::ONE_TO_MANY &&
$this->association['orphanRemoval'])
    $this->em->getUnitOfWork()->scheduleOrphanRemoval($element);
```

Should OneToMany associations (as any other) not only be scheduled for deleting after an explicit update/remove command (whatever the orphanRemoval attribute says)?

Annotating the property with @ORM\OneToMany(mappedBy="xyz", orphanRemoval=false) does not solve the issue.

The newsgroup entry for some other details and example models:
http://lists.typo3.org/pipermail/flow3-general/2012-February/001962.html

Related issues:			
related to TYPO3.Flow - Bug # 40389: Orphan Removal in ManyToMany Associations		**Resolved**	**2012-08-29**

History

#1 - 2012-06-21 12:29 - Karsten Dambekalns

- Project changed from Doctrine 2 to Doctrine.ORM

#2 - 2012-08-30 19:27 - Christian Müller

Hi, even if late some explanation of what happened before I close this ticket.

First of all in the spirit of Domain Driven Design we automatically enable orphanRemoval in case an entity hasn't got a repository.
Two points from your example came together, first this automatic orphanRemoval shouldn't happen for ManyToMany relations, this was fixed in master now and will come with 1.2 because it is a breaking change if you rely on this wrong behaviour.
Second probably your Person should have a repository and image should be related to only one of the two. I cannot define a different domain model without knowing your exact intentions, but generally a non aggregate root (so an entity without repository) should have relation only to one aggregate root and all other related entities should be fine for orphanRemoval (besides the ManyToMany bug you experienced).

#3 - 2012-08-30 19:27 - Christian Müller

- Assigned To set to Karsten Dambekalns

@Karsten please close/reject, I have no rights for this project. Thanks.

#4 - 2012-08-31 15:22 - Karsten Dambekalns

- Status changed from New to Closed