Provide override support for attributes in persistentobject

In 3.4 the 'fields' in a persistentobject definition is checked before the 'attribute_functions'. This makes it impossible to override these 'fields' without rewriting the attribute() method on the class.

Source file: doc/eng-us/features/3.5/persistentobject_attribute.txt Last changed 2026-11-04
https://doc.exponential.earth/exphelp/viewer/features/3.5/persistentobject_attribute.txt

```
*Title: Provide override support for attributes in persistentobject

Incentive:
In 3.4 the 'fields' in a persistentobject definition is checked before
the 'attribute_functions'. This makes it impossible to override these
'fields' without rewriting the attribute() method on the class.

The solution is to make attribute() method to first look trough
'attribute_functions' before checking 'fields' in the peristentobject definition.
This means that 'function_attributes' can override 'fields' to perform
custom code.
Usage: It is currently needed to fix 'remote_id' issue in ezcontentclass

*Documentation:
Add a new entry to 'attribute_functions' which is the same as a 'field'
and make it point to a method that exists in the class. This method should
then be called whenever the attribute is accessed.

However with this it will be impossible to fetch the value of the actual
'field'.
To remedy this a second parameter is added to eZPersistentObject::attribute
which is false by default. If set to true it will only check fields.

Note: eZContentClass:remoteID() has been fixed to use eZPersistentObject::attribute( 'remote_i
d', true );

*Testing:

*SVN Revisions:
8367, 8368, 8379
```

Copyright (C) 1999 - 2014 eZ Systems AS and contributors
This page was inherited from the documentation of the original software and is not covered by the GNU Free Documentation License. It is distributed under the GNU General Public License, version 2 (
https://www.gnu.org/licenses/old-licenses/gpl-2.0.html).