

## TYPO3.Flow - Bug # 34404

<b>Status:</b>	New	<b>Priority:</b>	Should have
<b>Author:</b>	Alexander Berl	<b>Category:</b>	
<b>Created:</b>	2012-02-29	<b>Assigned To:</b>	
<b>Updated:</b>	2012-02-29	<b>Due date:</b>	

### PHP Version:

**Has patch:** No

**Complexity:** easy

**Affected Flow version:** Git master

**Subject:** JsonView transformObject does not respect `_descendAll` configuration

### Description

When configuring a jsonView for a domain object, the `'_descendAll'` configuration is currently not taken into account, leading to the need to configure all properties for traversal.

The part of `TYPO3\FLOW3\MVC\View\JsonView::transformObject` line 199+

```
if (lis_array($propertyValue) && !lis_object($propertyValue)) {
    $propertiesToRender[$propertyName] = $propertyValue;
} elseif (isset($configuration['_descend']) && array_key_exists($propertyName, $configuration['_descend'])) {
    $propertiesToRender[$propertyName] = $this->transformValue($propertyValue,
$configuration['_descend'][$propertyName]);
}
```

should be

```
if (lis_array($propertyValue) && !lis_object($propertyValue)) {
    $propertiesToRender[$propertyName] = $propertyValue;
} elseif (isset($configuration['_descendAll'])) {
    $propertiesToRender[$propertyName] = $this->transformValue($propertyValue, $configuration['_descendAll']);
} elseif (isset($configuration['_descend']) && array_key_exists($propertyName, $configuration['_descend'])) {
    $propertiesToRender[$propertyName] = $this->transformValue($propertyValue,
$configuration['_descend'][$propertyName]);
}
```

A further **nice-to-have feature** would be a **real** `_descendAll` configuration that just carelessly descends all objects, though it probably should still deal with recursion of domain objects or stop at a specific recursion depth.

Something like an int configuration `_recursionDepth` would work:

```
} elseif (isset($configuration['_recursionDepth']) && $configuration['_recursionDepth'] > 0) {
    $configuration['_recursionDepth']--;
    $propertiesToRender[$propertyName] = $this->transformValue($propertyValue, $configuration);
}
```

*On a side note:* The current implementation for configuration settings behaves differently for arrays / `ArrayAccess` objects and other objects.

While for arrays, `_descendAll` overrules `_only` and `_exclude` settings, for an object `_only` and `_exclude` overrule `_descend` (and with

the patch above also `_descendAll()`).

## History

---