

# Добавление подгрупп для хлебных крошек

Для решения этой задачи был создан класс Kad\_Subgroups и метод add(). Суть работы метода в том, что он отслеживает события (хуки) добавления групп и структур в хлебные крошки и к каждой группе (структуре) добавляет список дочерних элементов.

## Пример добавления

Выполнение этого метода в коде должно быть между созданием контроллера хлебных крошек и их выводом:

```
// Вывод строки навигации
$Structure_Controller_Breadcrumbs = new
Structure_Controller_Breadcrumbs(
    Core_Entity::factory('Site', CURRENT_SITE)
);

$shop_id = 9;
$oSubgroups = new Kad_Subgroups;
$oSubgroups
    ->shop_id($shop_id)
    ->addSubgroupsToBreadcrumbs();

$Structure_Controller_Breadcrumbs
    ->xsl(
        Core_Entity::factory('Xsl')->getByName('ХлебныеКрошкиСайт7')
    )
    ->show();
```

По умолчанию подгруппы добавляются только для раздела "Интернет-магазин" и групп этого магазина. Однако, установив свойство:

```
$oSubgroups
    ->shop_id($shop_id)
    ->forAllStructures(TRUE)
    ->addSubgroupsToBreadcrumbs();
```

Можно указать, что подгружать подгруппы (дочерние структуры) необходимо для любых элементов хлебных крошек, а не только для интернет-магазина и его групп.

## Структура подгрупп в XML

В XML подгруппы имеют следующую структуру:

```
<shop_group id="923">
    . . . . .
```

```
<subgroups>
  <shop_group>
    .....
  </shop_group>

  <shop_group>
    .....
  </shop_group>
</subgroups>
</shop_group>
```

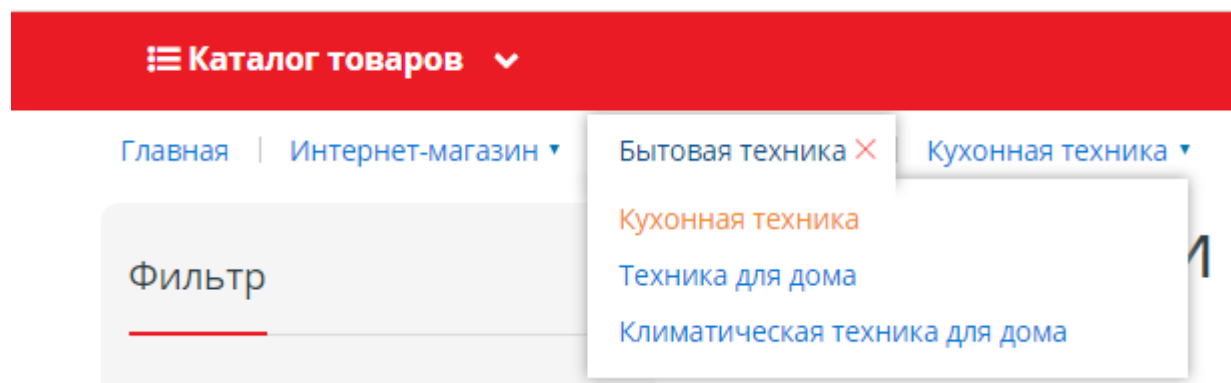
А дочерние структуры (если задано `forAllStructures(TRUE)`) имеют вид:

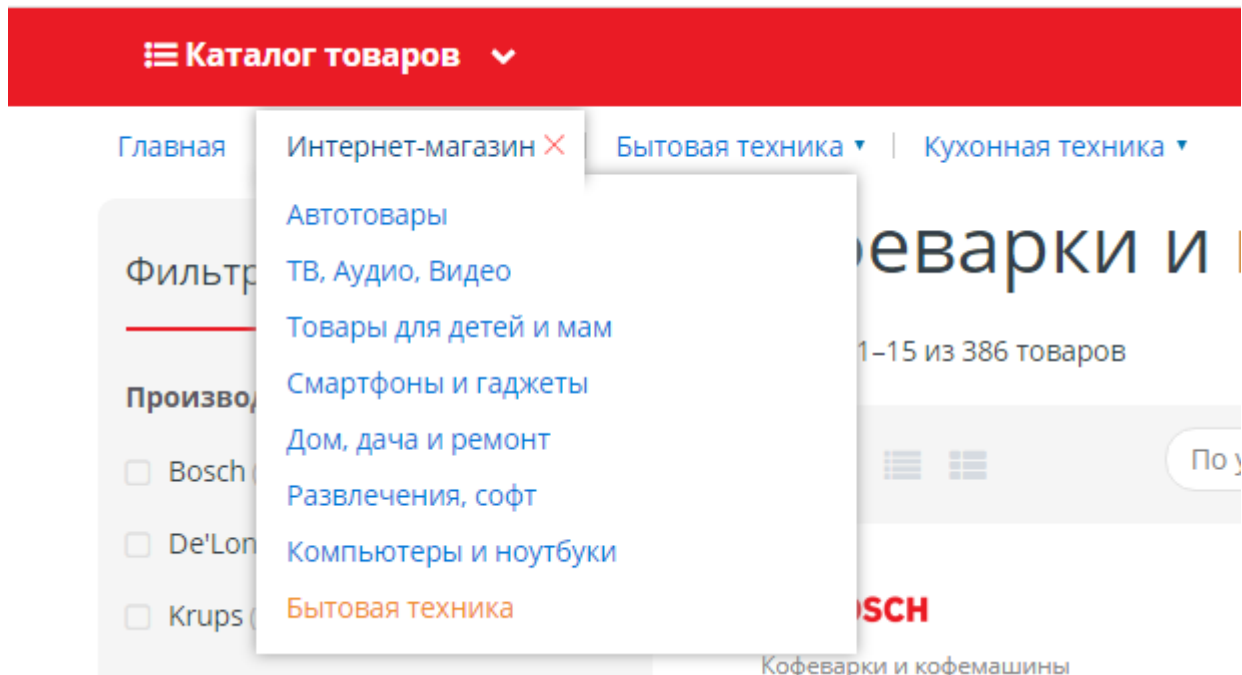
```
<structure id="275">
  .....
  <subgroups>
    <structure>
      .....
    </structure>
    <structure>
      .....
    </structure>
  </subgroups>
  .....
</structure>
```

## Дополнительные возможности

Так же этот класс обладает ещё одним методом: `getAllChildrenGroups()`, который в качестве параметра принимает группу интернет-магазина и возвращает массив всех подгрупп любой степени вложенности, включая родительскую (ту, что была передана в качестве параметра).

## Пример использования данного решения на сайте z97.ru





## Исходный код

```
<?php
/**
 * Класс для работы с подгруппами
 *
 * @author Kozyrev Egor, KAD Systems (©) 2017
 * @date 30-11-2017
 * Начало >>
 */
class Kad_Subgroups {

    private $shop_id; //Идентификатор интернет-магазина
    private $bForAllStructures; //Указатель на то, будут ли добавляться подгруппы
    //первого уровня только для интернет-магазина или для для всех разделов

    public function shop_id($id)
    {
        $this->shop_id = $id;
        return $this;
    }

    public function forAllStructures($arg)
    {
        $this->bForAllStructures = $arg;
        return $this;
    }

    /**
     * Добавление подгрупп первого уровня в контроллер хлебных крошек
```

```

*/
public function addSubgroupsToBreadcrumbs()
{
Core_Event::attach('Structure_Controller_Breadcrumbs.onAfterAddStructure',
    function($controller, $args) {

        $controller->clearEntities();
        $oStructure = $args[];
        $xml = Core::factory("Core_Xml_Entity")
            ->name('subgroups');

        $oShop = Core_Entity::factory('Shop', $this->shop_id);

        //Для интернет-магазина добавляем группы товаров
        if($oStructure->id == $oShop->structure_id)
        {
            $group = Core_Entity::factory("Shop_Group");
            $group->queryBuilder()
                ->where("parent_id", "=", )
                ->where("active", "=", "1")
                ->where("shop_id", "=", $this->shop_id);
            $aoChildren = $group->findAll();

            foreach (array_reverse($aoChildren) as $oChild)
            {
                $cloned = clone $oChild;
                $xml
                    ->addEntity(
                        $cloned
                    );
            }
        }
        //Для любой другой структуры добавляем дочерние структуры первого уровня
        elseif($this->bForAllStructures)
        {
            $aoChildren = $oStructure->Structures->findAll();

            foreach($aoChildren as $oChild)
            {
                $cloned = clone $oChild;
                $xml
                    ->addEntity(
                        $cloned
                    );
            }
        }

        $oStructure->addEntity($xml);
    });

Core_Event::attach('Structure_Controller_Breadcrumbs.onAfterAddShopGroups',

```

```

function($controller, $args) {

    $controller->clearEntities();
    $aShop_Groups = $args[];

    //Добавление подгрупп в группы товаров
    foreach (array_reverse($aShop_Groups) as $oGroup)
    {
        $group = Core_Entity::factory("Shop_Group");
        $group->queryBuilder()
            ->where("parent_id", "=", $oGroup->id)
            ->where("active", "=", "1");
        $aoChildren = $group->findAll();

        $xml = Core::factory("Core_Xml_Entity")
            ->name('subgroups');

        foreach ($aoChildren as $oChild)
        {
            $cloned = clone $oChild;
            $xml
                ->addEntity(
                    $cloned
                );
        }
        $oGroup->addEntity($xml);
    }
});
}

/**
 * Принимает объект группы и возвращает массив с этой самой группой
 * и всеми её дочерними подгруппами всех уровней вложенности
 */
public function getAllChildrenGroups($oGroup)
{
    $aResult = array($oGroup);

    $oGroup->queryBuilder()
        ->where("parent_id", "=", $oGroup->id);
    $aGroups = $oGroup->findAll();

    if(count($aGroups))
    {
        foreach ($aGroups as $group)
        {
            $aResult = array_merge($aResult,
$this->getAllChildrenGroups($group));
        }
        return $aResult;
    }
}

```

```
    }  
    else  
    {  
        return array($oGroup);  
    }  
}  
  
}  
/**  
* << Конец  
* @author Kozyrev Egor, KAD Systems (©) 2017  
* @date 30-11-2017  
*/
```

Страницу создал Егор Козырев 17.11.17 в 15:36