

XCat は、XOOPS のカテゴリを管理するためのモジュールです。

XCat を使うことで、以下のようなメリットを享受できます。

- モジュール作成時にカテゴリと権限の管理の機能を XCat に任せることができる。

- 運用時に、複数のモジュール間でカテゴリ設定を共有できる。

いずれも、XCat を利用しているモジュールでなければなりません。

XCat では、XOOPS Cube の service 機能を利用して、他のモジュールに対してカテゴリ管理機能を提供しています。

XCat では、カテゴリ — ユーザグループ — アクション の単位で権限を設定(許可／不許可)します。

「アクション」は権限の種類を表し、自由に設定することができます(次ページ参照)。

カテゴリごとにこのような権限設定のテーブルがある

アクション

ユーザ グループ	Group	閲覧	編集	削除	管理
	サイト管理者	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	登録ユーザ	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	ゲスト	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

アクションキー(action key)には、任意の名前をつけることができます。

しかし、利便性を考え、ある程度の統一を図るため、以下のようなアクションキーを標準として提案しています。

- viewer: 閲覧のみの許可。
- poster: 新規の投稿と自分の投稿の修正・削除のみ可能
- editor: 他人の投稿の承認と、修正・削除などが可能
- manager: モジュール全体に関わる設定が可能

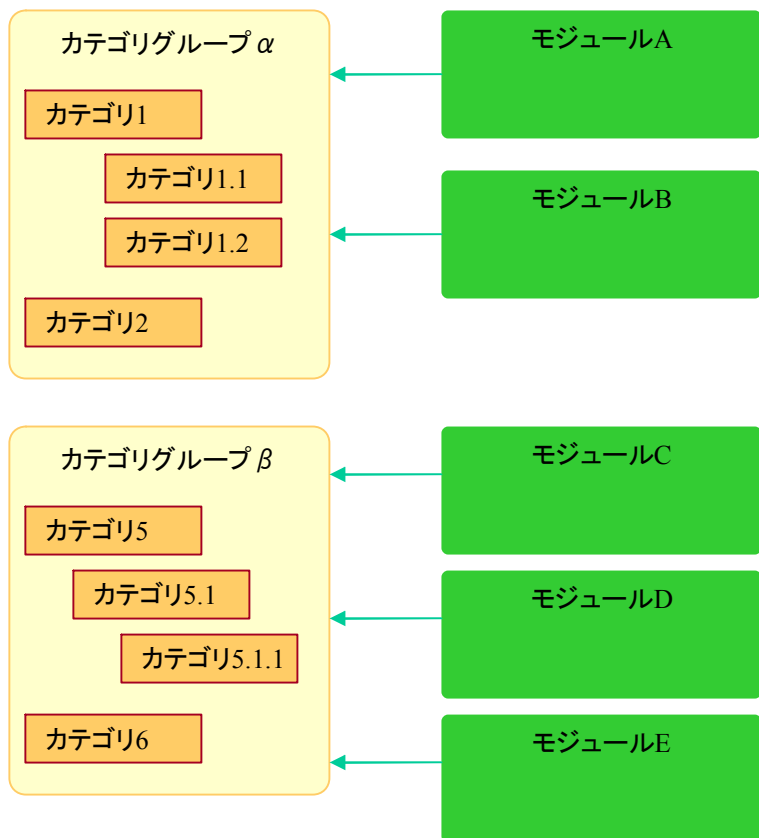
これらのアクションキーは、すべて使う必要はありませんし、独自に定義することもできます。

単純なモジュールであれば、editor 権限のみ使えば良いでしょう。

XCat は複数の「カテゴリグループ」を持つことができます。

これによって、モジュールごとにどのカテゴリグループを使うか、それぞれ指定することができます。

※各カテゴリは、必ず一つのカテゴリグループに属します。どのカテゴリグループにも属さないカテゴリはありませんし、複数のカテゴリグループに属するカテゴリもありません。



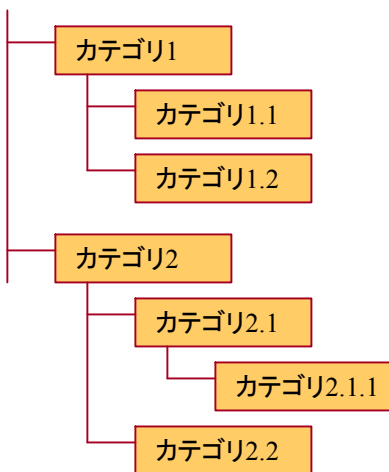
カテゴリグループでは、どのようなアクションキーを持つかと、権限設定が行われていない場合に適用されるデフォルトの権限を設定します。

カテゴリグループの編集		
カテゴリグループ名	XOOPS	
カテゴリ階層制限	0	
権限の種類	権限の名前と表示名を入れてください(権限の名前は半角英数)	
キー	表示名	権限初期設定
view	閲覧	<input checked="" type="checkbox"/>
edit	編集	<input type="checkbox"/>
delete	削除	<input checked="" type="checkbox"/>
manage	管理	<input checked="" type="checkbox"/>
<input type="button" value="権限の行を追加"/>		
<input type="button" value="送信"/>		

カテゴリグループでは、カテゴリ階層の深さも制限することができます。

制限を「1」にすれば、カテゴリ階層を持ちません。

制限「0」はカテゴリ階層に制限を持たせません。



カテゴリは親子関係の階層構造を持ちます。

カテゴリ一つ一つに、権限を設定することができます。

あるカテゴリに権限が設定されていない場合、その親の権限設定を継承します。その親も権限設定がなければさらにその親を参照し、というように、トップレベルのカテゴリまで遡って権限を継承します。

トップレベルのカテゴリにも権限が設定されていなければ、カテゴリグループで設定されている初期権限が適用されます。

権限の継承発生はカテゴリ単位に判断されます。それ以上の細かい単位、ユーザグループや権限ごとではありません。

たとえば、"カテゴリ:A" - "groupid:1" - "VIEW" という権限の設定があれば、"カテゴリ:A" - groupid:2" - "VIEW" や "カテゴリA" - groupid:1" - "CREATE" について、権限の継承は発生しません。もしこれらの権限が設定されていなければ、「不可」として扱われます。

XCat のserviceを使ってできる画面周りの機能のサンプルです。XCat では、serviceで取得したデータを丸ごと渡すことで適切に整形して表示する smarty プラグインを幾つか用意しています。

- 1▶ service getTree でツリーを取得し、それを Smarty の function.xcatsvc_catselect に渡す

カテゴリ	<div>- Legacy ▼</div> <div>- Legacy</div>
名前	<div>--trust_path</div> <div>admin</div>

カテゴリの一覧を select の形で表示。

権限を持たない場合は選択肢に表示されません。

- 2▶ service getTree でツリーを取得し、それを Smarty の function.xcatsvc_tree に渡す



カテゴリの一覧をツリーの形で表示。

カテゴリをクリックすると、カテゴリのページに移動します。

※カテゴリのページのURLは、“固定文字列”+“カテゴリid”の形に限定されます。

例: `http://dummy.jp/modules/dummy/index.php?action=ItemList&cat_id=3`

カテゴリに対して権限がない場合には、リンクが張られません。

- 3▶ service getTree でツリーを取得し、それとモジュール側のコンテンツ一覧を Smarty の function.xcatsvc_tree に渡す

ヘルプ

カテゴリと、そのカテゴリに属するコンテンツをツリーの形で表示。

このサイトについて

このサイトは何のサイトですか

ユーザ登録した方が良いですか？

XCat を利用する場合、必ず以下のコードを最初に実行します。
次ページ以降のコードを実行する前にも必要になります。

```
1: $root =& XCube_Root::getSingleton();  
2: $service = $root->mServiceManager->getService("Xcat_CatService");  
3: $client = $root->mServiceManager->createClient($service);
```

service のファンクションを呼び出す

```
1: $tree = $client->call('getTree',  
    array('gr_id'=>$grId, 'cat_id'=>0, 'p_id'=>0, 'action'=>'viewer', 'uid'=>$uid));
```

service 自体あまり知られていないと思いますので、service で提供されるファンクションについて簡単に説明します。

service で提供されているファンクションを使う場合は、上で作った \$client を使い、\$client->call(ファンクション名, 引数の配列); を実行します。

引数は連想配列として、ファンクションごとに規定されている引数を渡します。上の例では、gr_id, cat_id, p_id, action, uid を連想配列にして渡しています。

▶ カテゴリ一覧をツリー形式で取得し、権限のあるカテゴリのみをツリー形式で表示する。

```
1: $catTree = $client->call('getTree',  
    array('gr_id'=>$grId, 'cat_id'=>0, 'p_id'=>0, 'action'=>'viewer', 'uid'=>$uid));
```

「call('getTree', 」の部分では、getTree という名前の service のメソッドを呼び出しています。このメソッドにより、カテゴリの配列を取得します。

引数

gr_id: グループID / cat_id: カテゴリID。gr_id か cat_id のどちらかを渡します。gr_id が分からない場合は、cat_id を渡せば service 側でその cat_id が属しているグループを見つけて gr_id に変換します。両方渡した場合は、gr_id のみが利用されます。

p_id: 親カテゴリID。特定のカテゴリ配下のカテゴリツリーのみを取得したい場合は、この引数を指定します。戻ってくるカテゴリツリーには、p_id の配列自体は含まれません。全ツリーを取得したい場合は 0 を渡します。

action: パーミッションのアクションの種類 / uid: ユーザID。権限を持っているかどうかをチェックした状態でツリーを取得したい場合に action を指定します。uid のみを指定しても無視されます。action のみを指定した場合は、ゲストグループの権限をチェックした結果を返します('uid=>0 を指定した場合も同様)。

返り値

返り値は、以下の項目からなるカテゴリの配列です。配列は、カテゴリの親子関係に沿った順番で戻ります。同一階層のカテゴリは weight 順に並びます。

int cat_id	カテゴリID
string cat_title	カテゴリ名
int gr_id	カテゴリグループID
int p_id	親カテゴリID
text cat_desc	カテゴリの説明
int weight	表示順
text options	カテゴリオプション
int cat_depth	カテゴリの階層(最上階は1)
string permit	引数 uid で指定されたユーザが引数action についての権限を持っている(=1)か、もっていない(=0)か。action を指定しなかった場合はすべて持っている(=1)で返ります。

▶ セレクトボックスにツリーを表示する (「画面表示に関わる機能」の「1」)

template ファイル内の記述

```
1: <select name="cat_id">
2: <{xcatsvc_catselect tree=$catTree selectedValue=$object->get('cat_id')}>
3: </select>
```

1行目と3行目は通常のhtmlのselect フォームタグです。

2行目は、XCat で提供している Smarty プラグイン function.xcatsvc_catselect.php を使って option タグを表示させるところです。

引数 tree には、前頁の getTree で取得した \$catTree 配列をそのまま渡します。template ファイルに渡すには、通常通り \$render->setAttribute('catTree', \$catTree); や、\$tpl->assign('catTree', \$catTree); などを使います。

引数 selectedValue には <option> タグの selected="selected" 属性を指定したいカテゴリIDを指定します。

カテゴリ	<input type="text" value="- Legacy"/>
名前	<input type="text" value="-- trust_path admin"/>

▶ リンクつきでツリーを表示する (前頁の「2」)

template ファイル内の記述

```
1: <{xcatsvc_tree tree=$catTree url="./index.php?action=CatView&cat_id=%d"}>
```

1行目は、XCat で提供している Smarty プラグイン function.xcatsvc_tree.php を使ってカテゴリツリーを表示する部分です。

引数 tree には、前頁の getTree で取得した \$catTree 配列をそのまま渡します。template ファイルに渡すには、通常通り \$render->setAttribute('catTree', \$catTree); や、\$tpl->assign('catTree', \$catTree); などを使います。

引数 url には、カテゴリツリーのリンクをクリックしたときに表示したいページのURLを指定します。 %d はカテゴリIDに置き換えられます。

カテゴリ

- インストール
 - Legacy
 - trust_path
- ブロック

▶ カテゴリIDからカテゴリ名を取得する

```
1: $catTitle = $client->call('getTitle', array('cat_id'=>$catId));
```

カテゴリIDをカテゴリ名に変換します。主に詳細表示画面で使います。返り値にはstring のカテゴリ名がそのまま来ます。
一覧表示画面などで複数のカテゴリIDをカテゴリ名に変換する場合は、次の getTitleList もご利用ください。

▶ カテゴリIDからカテゴリ名を取得する

```
1: $catTitleArr = $client->call('getTitleList', array('gr_id'=>$grId));
```

キーがカテゴリIDで値がカテゴリ名の配列を取得します。
一覧表示画面などで複数のカテゴリIDをカテゴリ名に変換する場合に使います。
template で、以下のように書くことでカテゴリIDからカテゴリ名を表示することができます。

```
1: <{$catTitleArr[$catId]}>
```

▶ ユーザIDによる権限のチェック

```
1: if($client->call('checkPermitByUid',  
    array('action'=>'poster', 'uid'=>$uid, 'cat_id'=>$catId))){  
2:         権限を持っている場合の処理  
3: }else{  
4:         権限を持っていない場合の処理  
5: }
```

`cat_id` で指定したカテゴリについて、`uid` で指定したユーザが `action` で指定した処理を行う権限を持っているかどうかをチェックします。
指定された `uid` のユーザが所属しているユーザグループのいずれかが権限を持っている場合は `true` を、いずれも権限を持っていない場合は `false` を返します。また、ユーザを直接して権限を振っている場合にも `true` を返します(未実装)。

▶ ユーザグループIDによる権限のチェック

```
1: if($client->call('checkPermitByGroupid',  
    array('action'=>'poster', 'groupid'=>$uid, 'cat_id'=>$catId))){  
2:         権限を持っている場合の処理  
3: }else{  
4:         権限を持っていない場合の処理  
5: }
```

`cat_id` で指定したカテゴリについて、`groupid` で指定したユーザグループが `action` で指定した処理を行う権限を持っているかどうかをチェックします。
指定された `groupid` のユーザグループが権限を持っている場合は `true` を、権限を持っていない場合は `false` を返します。

▶ 子カテゴリ一覧の取得

```
1: $children = $client->call('getChildren',  
    array('cat_id'=>$catId, 'action'=>'viewer', 'uid'=>$uid));
```

引数 `cat_id` の子カテゴリの一覧を配列で返します。子カテゴリは、`weight` の順の配列になります。配列の中の項目は、カテゴリツリーと同じです。また、`action`、`uid` もカテゴリツリーと同様で、`action` を指定した場合は権限をチェックした結果を `permit` キーの値につけて返します。`uid` のみを指定しても無視される点、`uid` が無指定および `0` 指定の場合はゲストグループ扱いで権限チェックをして結果を返す点も同じです。

▶ カテゴリパスの取得

```
1: $catPath = $client->call('getCatPath', array('cat_id'=>$catId, 'order'=>'ASC'));
```

パンくずリストのような、カテゴリのリストを取得したいときに使います。

引数 `cat_id` から親カテゴリを遡っていき、その結果を配列で返します。配列の項目は、`cat_id`、`cat_title` です。

返ってくる配列には、引数として渡した `cat_id` は含まれません。その親から上位の配列になります。

引数 `order` に `'ASC'` が指定されている場合は、親から子へという順番の配列になります。それ以外の場合は、子から親へという順番の配列になります。

▶ カテゴリグループ一覧の取得

```
1: $grArr = $client->call('getGrList', array());
```

カテゴリグループの一覧を返します。モジュール側で、どのカテゴリグループを使うかを選ばせるときなどに使います。

返り値は、`gr_id`:カテゴリグループID、`gr_title`:カテゴリグループ名、`level`:カテゴリの深さの制限、`action`:設定される権限の種類からなる配列です。