Creating fast, dynamic ACLs in Zend Framework

Wim Godden
Cu.be Solutions

Who am I ?

- Wim Godden (@wimgtr)
- Owner of Cu.be Solutions (http://cu.be)
- PHP developer since 1997
- Developer of OpenX
- Zend Certified Engineer
- Zend Framework Certified Engineer
- MySQL Certified Developer

cu.be
it's in the dot

Talking about...

- ~~Authentication~~
  → Zend_Auth
- ~~Auditing~~
  → Zend_Log
- Authorization
  → Zend_Acl

Authorization

Wikipedia :

"the function of specifying access rights to resources"

What's a resource ?

- Object (Article, Invoice, Document, …)
- Webpage
- Database / table / row
- ...

Standard ACL

- Access to **resources** is defined in **privileges**
- Privileges are grouped together in **roles**
- 2 types of **roles** :
  - Anonymous / Unknown
  - Registered / Known

Within Zend Framework : Zend_Acl

- Flexible
- Uses standard role, resource principles

Zend_Acl : the good

- Recognizable → easy to get started
- No link to specific backend
- Allow + deny
- Proven, tested

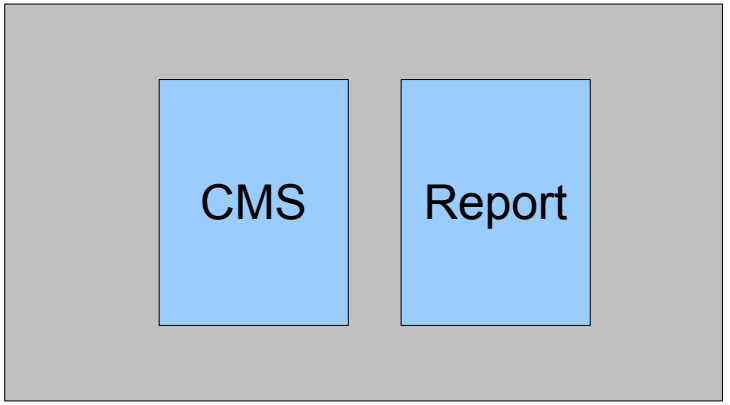
Zend_Acl : the bad & ugly

- Complexity of rules rises quickly
- Performance issues
- All rules are in-code

→ maintainability becomes an issue

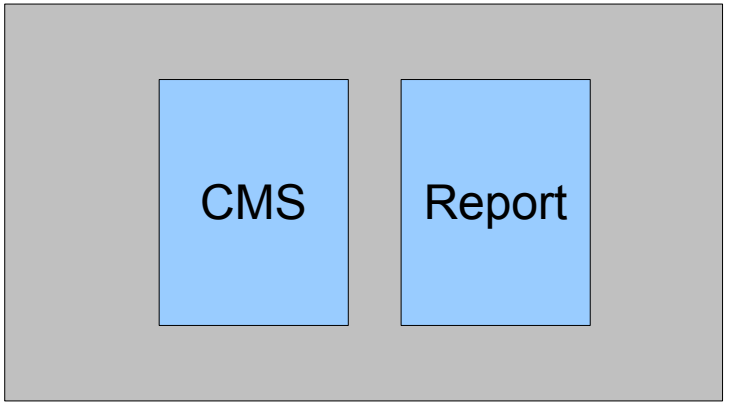
Evolution of a portal

```php
$acl = new Zend_Acl();
$acl->addRole(new Zend_Acl_Role('guest'));
$acl->addRole(new Zend_Acl_Role('member'), 'guest');
$acl->addRole(new Zend_Acl_Role('admin'), 'member');
$acl->addResource(new Zend_Acl_Resource('cms'));
$acl->addResource(new Zend_Acl_Resource('report'));
$acl->allow('guest', 'cms', 'view');
$acl->allow('admin', 'cms', 'edit');
$acl->deny('guest', 'report');
$acl->allow('member', 'report');
```

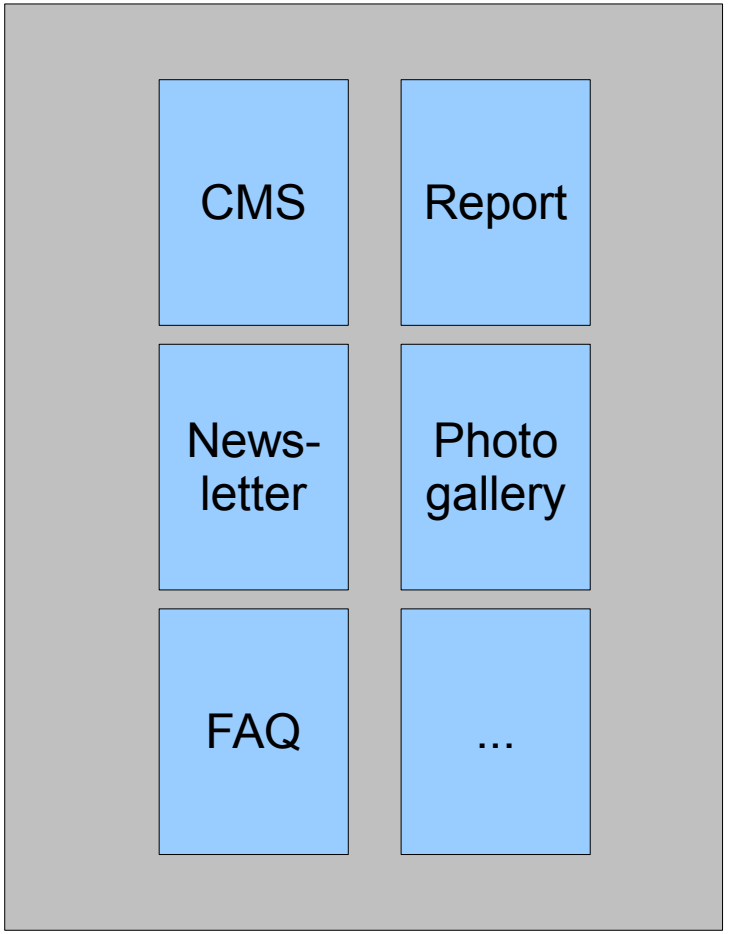


Evolution of a portal

```php
$acl = new Zend_Acl();
$acl->addRole(new Zend_Acl_Role('guest'));
$acl->addRole(new Zend_Acl_Role('departmentA'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentB'), 'guest');
$acl->addRole(new Zend_Acl_Role('admin'), 'member');
$acl->addResource(new Zend_Acl_Resource('cms'));
$acl->addResource(new Zend_Acl_Resource('report'));
$acl->allow('guest', 'cms', 'view');
$acl->allow('admin', 'cms', 'edit');
$acl->deny('guest', 'report');
$acl->allow('departmentA', 'report');
```

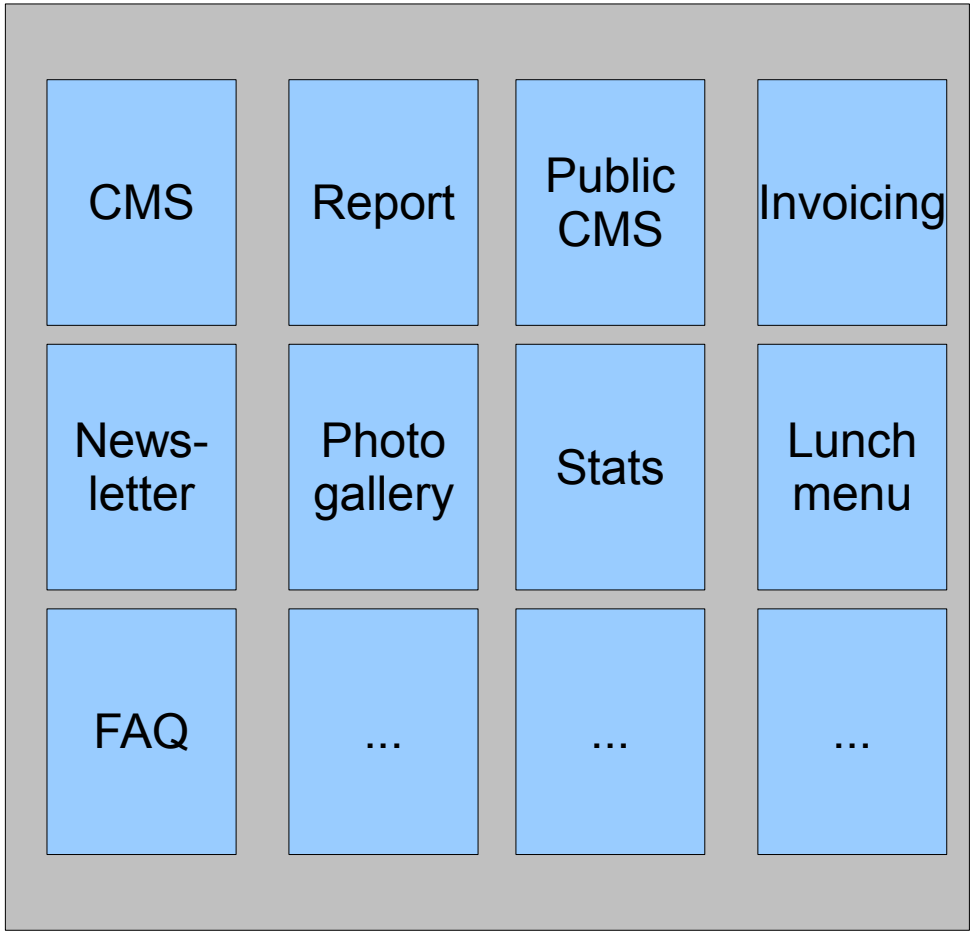


Evolution of a portal



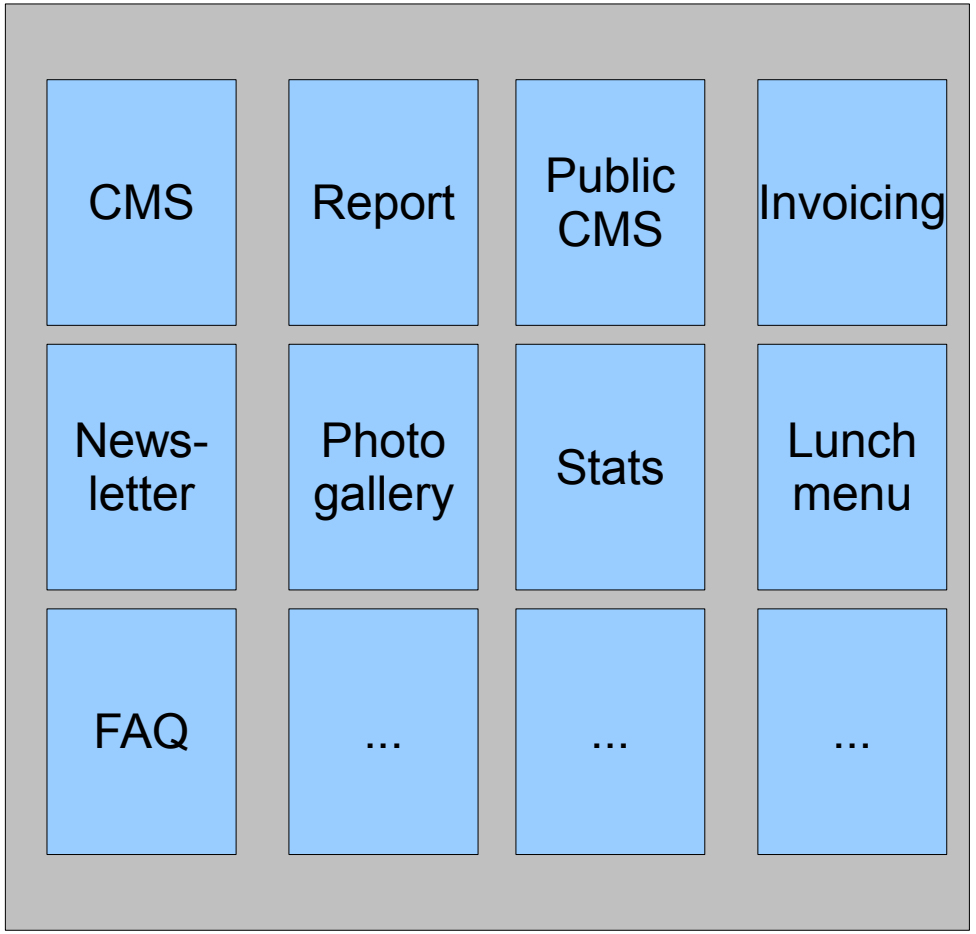

```php
$acl = new Zend_Acl();
$acl->addRole(new Zend_Acl_Role('guest'));
$acl->addRole(new Zend_Acl_Role('departmentA'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentB'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentC_senior_staff'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentC_marketing'), 'guest');
$acl->addRole(new Zend_Acl_Role('admin'), 'member');
$acl->addResource(new Zend_Acl_Resource('cms'));
$acl->addResource(new Zend_Acl_Resource('report'));
$acl->addResource(new Zend_Acl_Resource('newsletter'));
$acl->addResource(new Zend_Acl_Resource('photo'));
$acl->addResource(new Zend_Acl_Resource('faq'));
$acl->allow('guest', 'cms', 'view');
$acl->allow('admin', 'cms', 'edit');
$acl->deny('guest', 'report');
$acl->allow('departmentA', 'report');
$acl->deny('departmentC_senior_staff', 'newsletter');
$acl->allow('departmentC_marketing', 'newsletter');
$acl->allow('member', 'photo', 'view');
$acl->allow('departmentC_marketing', 'photo', 'upload');
$acl->allow('admin', 'photo', 'delete');
$acl->allow('guest', 'faq', 'view');
$acl->allow('member', 'faq', 'comment');
$acl->allow('departmentA', 'faq', 'edit');
$acl->allow('departmentC_senior_staff', 'faq', 'edit');
$acl->allow('admin', 'faq', 'edit');
```

Evolution of a portal




```php
$acl = new Zend_Acl();
$acl->addRole(new Zend_Acl_Role('guest'));
$acl->addRole(new Zend_Acl_Role('departmentA'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentB'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentC_senior_staff'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentC_marketing'), 'guest');
$acl->addRole(new Zend_Acl_Role('cook'), 'guest');
$acl->addRole(new Zend_Acl_Role('admin'), 'member');
$acl->addResource(new Zend_Acl_Resource('cms'));
$acl->addResource(new Zend_Acl_Resource('report'));
$acl->addResource(new Zend_Acl_Resource('newsletter'));
$acl->addResource(new Zend_Acl_Resource('photo'));
$acl->addResource(new Zend_Acl_Resource('faq'));
$acl->addResource(new Zend_Acl_Resource('invoicing'));
$acl->addResource(new Zend_Acl_Resource('stats'));
$acl->addResource(new Zend_Acl_Resource('lunchmenu'));
$acl->allow('guest', 'cms', 'view');
$acl->allow('admin', 'cms', 'edit');
$acl->deny('guest', 'report');
$acl->allow('departmentA', 'report');
$acl->deny('departmentC_senior_staff', 'newsletter');
$acl->allow('departmentC_marketing', 'newsletter');
$acl->allow('member', 'photo', 'view');
$acl->allow('departmentC_marketing', 'photo', 'upload');
$acl->allow('admin', 'photo', 'delete');
$acl->allow('guest', 'faq', 'view');
$acl->allow('member', 'faq', 'comment');
$acl->allow('departmentA', 'faq', 'edit');
$acl->allow('departmentC_senior_staff', 'faq', 'edit');
$acl->allow('admin', 'faq', 'edit');
$acl->allow('admin', 'photo', 'delete');
$acl->allow('guest', 'faq', 'view');
$acl->allow('member', 'faq', 'comment');
$acl->allow('departmentA', 'faq', 'edit');
$acl->allow('departmentC_senior_staff', 'faq', 'edit');
$acl->allow('admin', 'faq', 'edit');
$acl->allow('cook', 'lunchmenu', 'edit');
$acl->allow('member', 'lunchmenu', 'view');
$acl->allow('accounting', 'invoicing', 'edit');
$acl->allow('admin', 'invoicing', 'edit');
$acl->allow('departmentC_senior_staff', 'invoicing', 'report');
```

Evolution of a portal



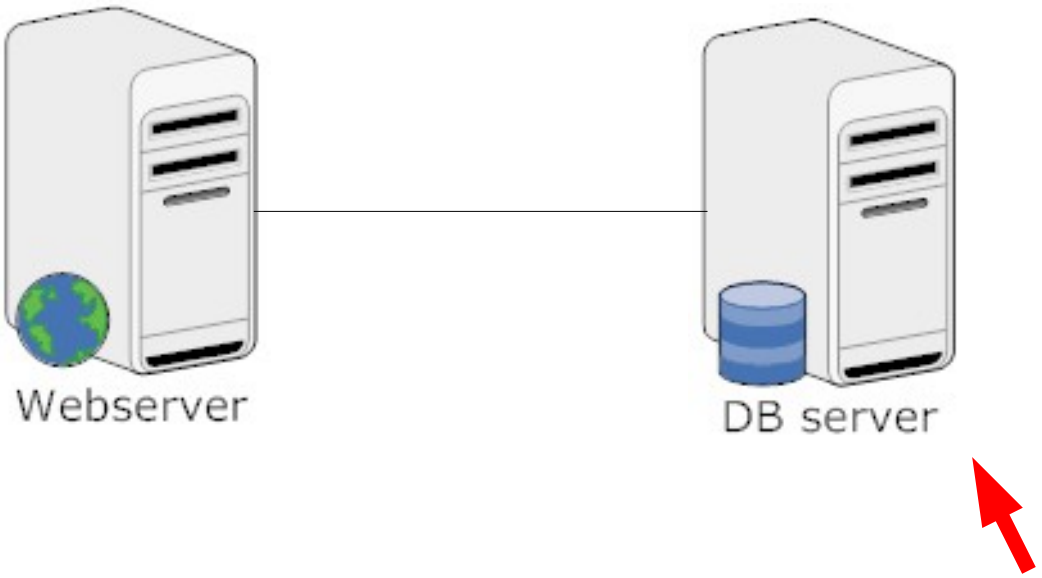

```php
$acl = new Zend_Acl();
$acl->addRole(new Zend_Acl_Role('guest'));
$acl->addRole(new Zend_Acl_Role('departmentA'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentB'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentC_senior_staff'), 'guest');
$acl->addRole(new Zend_Acl_Role('departmentC_marketing'), 'guest');
$acl->addRole(new Zend_Acl_Role('cook'), 'guest');
$acl->addRole(new Zend_Acl_Role('admin'), 'member');
$acl->addResource(new Zend_Acl_Resource('cms'));
$acl->addResource(new Zend_Acl_Resource('report'));
$acl->addResource(new Zend_Acl_Resource('newsletter'));
$acl->addResource(new Zend_Acl_Resource('photo'));
$acl->addResource(new Zend_Acl_Resource('faq'));
$acl->addResource(new Zend_Acl_Resource('invoicing'));
$acl->addResource(new Zend_Acl_Resource('stats'));
$acl->addResource(new Zend_Acl_Resource('lunchmenu'));
$acl->allow('guest', 'cms', 'view');
$acl->allow('admin', 'cms', 'edit');
$acl->deny('guest', 'report');
$acl->allow('departmentA', 'report');
$acl->deny('departmentC_senior_staff', 'newsletter');
$acl->allow('departmentC_marketing', 'newsletter');
$acl->allow('member', 'photo', 'view');
$acl->allow('departmentC_marketing', 'photo', 'upload');
$acl->allow('admin', 'photo', 'delete');
$acl->allow('guest', 'faq', 'view');
$acl->allow('member', 'faq', 'comment');
$acl->allow('departmentA', 'faq', 'edit');
$acl->allow('departmentC_senior_staff', 'faq', 'edit');
$acl->allow('admin', 'faq', 'edit');
$acl->allow('admin', 'photo', 'delete');
$acl->allow('guest', 'faq', 'view');
$acl->allow('member', 'faq', 'comment');
$acl->allow('departmentA', 'faq', 'edit');
$acl->allow('departmentC_senior_staff', 'faq', 'edit');
$acl->allow('admin', 'faq', 'edit');
$acl->allow('cook', 'lunchmenu', 'edit');
$acl->allow('member', 'lunchmenu', 'view');
$acl->allow('accounting', 'invoicing', 'edit');
$acl->allow('admin', 'invoicing', 'edit');
$acl->allow('departmentC_senior_staff', 'invoicing', 'report');
```

Hard to ...

- maintain all rules
- keep track of the rules
- debug the rules

Possible solution : database

- Extend Zend_Acl to database driven design
- Good : no code changes required
- Bad : more load on DB



A different approach

- Not **THE** solution, merely **A** solution
- Uses database, but...
- Additional caching layer
- ZF Conventional Modular Directory Structure
- Backend interface for ~~easy~~ management

```
application/
    default/
        controllers/
            IndexController.php
            FooController.php
        models/
        views/
            scripts/
                index/
                foo/
            helpers/
            filters/
    blog/
        controllers/
            IndexController.php
        models/
        views/
            scripts/
                index/
            helpers/
            filters/
    news/
        controllers/
            IndexController.php
            ListController.php
        models/
        views/
            scripts/
                index/
                list/
            helpers/
            filters/
```

Different resources

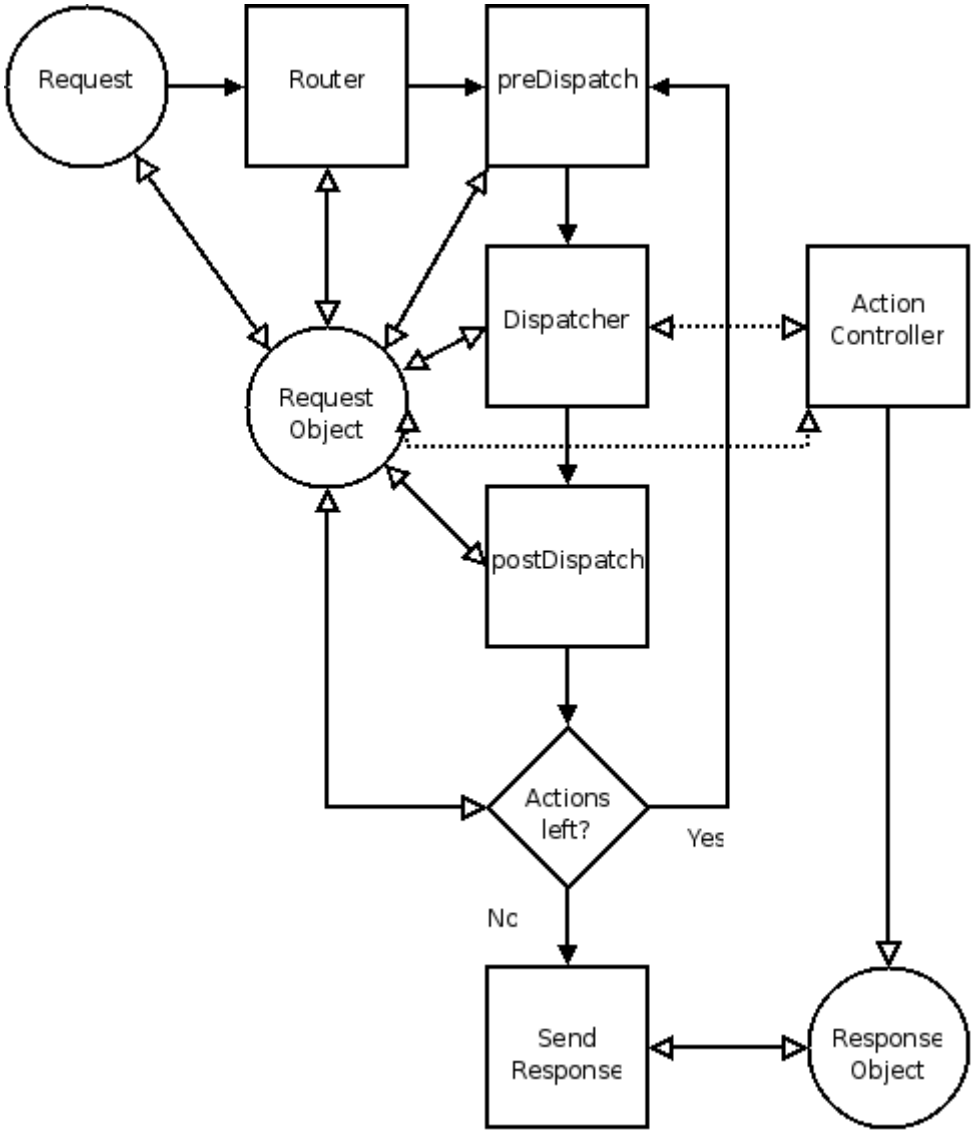
- Zend_ACL :

```
$acl->addResource(new Zend_Acl_Resource('cms'));
$acl->allow('guest', 'cms', 'view');
$acl->allow('admin', 'cms', 'edit');
```

- Access to :
  - Controller : cms
  - Action : view / edit

- Why not integrate with the request itself ?

Controller plugins



Zend_Acl as a controller plugin

```php
<?php
class My_Plugin_Acl extends Zend_Controller_Plugin_Abstract {
  private $_acl = null;

  public function __construct(Zend_Acl $acl) {
    $this->_acl = $acl;
  }

  public function preDispatch(Zend_Controller_Request_Abstract $request) {
    $role = (Zend_Auth::getInstance()->hasIdentity()) ? 'user' : 'guest';

    //For this example, we will use the controller as the resource:
    $resource = $request->getControllerName();

    if(!$this->_acl->isAllowed($role, $resource, 'view')) {
      //If the user has no access we send him elsewhere by changing the request
      $request->setModuleName('auth')
              ->setControllerName('auth')
              ->setActionName('login')
              ->setDispatched(false);
      return false;
    }
  }
}
```

Initializing the ACL

Let's have a look

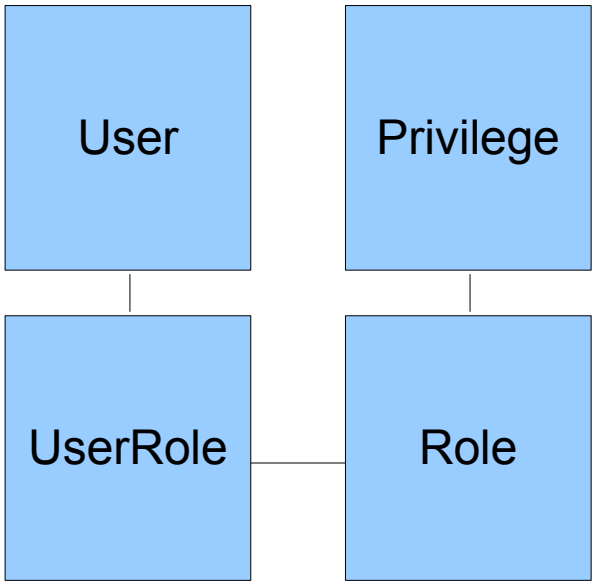
Zend_Acl manual rules

```php
<?php
class My_Acl extends Zend_Acl {
  public function __construct() {
    //Add a new role called "guest"
    $this->addRole(new Zend_Acl_Role('guest'));

    //Add a role called user, which inherits from guest
    $this->addRole(new Zend_Acl_Role('user'), 'guest');

    //Add a resource called page
    $this->add(new Zend_Acl_Resource('page'));

    //Add a resource called news, which inherits page
    $this->add(new Zend_Acl_Resource('news'), 'page');

    //Finally, we want to allow guests to view pages
    $this->allow('guest', 'page', 'view');

    //and users can comment news
    $this->allow('user', 'news', 'comment');
  }
}
```

Our ACL

id	name	email	pw
1	Chris	chris@acl.com	*******
2	Jake	jake@places.com	*******
3	Jeniffer	jenny@ho.me	*******

id	role_id	module	controller	action
1	2	newsletter	send	index
2	1	cms	article	edit
3	3	%	%	%



user_id	role_id
1	2
2	3
3	1

id	name
1	webmaster
2	marketeer
3	admin

cube
it's in the dot

Application_Acl

```
class Application_Acl
{
    public function isAllowed($user = null, $request = null, $privilege = null)
    {
        if (is_null($user) === false && $user !== false && $user instanceof User) {
            $userId = $user->id;
        } else {
            $userId = 0;
        }
        $db = Zend_Db_Table::getDefaultAdapter();
        $stmt = $db->query('
            select
               module_name,
               controller_name,
               action_name
            from
                privilege
                      join role
                              on role.id = privilege.role_id
                      join userRole
                              on userRole.role_id = role.role_id
            where
                userRole.user_id = ?
                and
                (
                    module_name = "%"
                    or
                    (
                        module_name = ?
                        and
                        (
                            controller_name = "%"
                            or
                            (
                                controller_name = ?
                                and
                                (
                                    action_name = "%"
                                    or
                                    action_name = ?
                                )
                            )
                        )
                    )
                )
            ',
            array(
                $userId,
                $request->getModuleName(),
                $request->getControllerName(),
                $request->getActionName()
            )
        );
                $stmt->execute();
                $row = $stmt->fetch(); // Returns a row or false
                if ($row !== false) {
                    return true;
                } else {
                    return false;
                }
            }
        }
```

To cache or not to cache - Option 1 (no cache)

- 1 query per ACL request
  = 1 query per pageview / ajax request
- 99.99% will be identical
  → "Just rely on MySQL query cache !"

FAIL !

- Even for cached queries, MySQL connections use memory, I/O, CPU, ...

Caching - Option 2 (cache the main query)

- 1 DB query per user for each unique ACL request
- User with 20 privileges → 20 possible requests

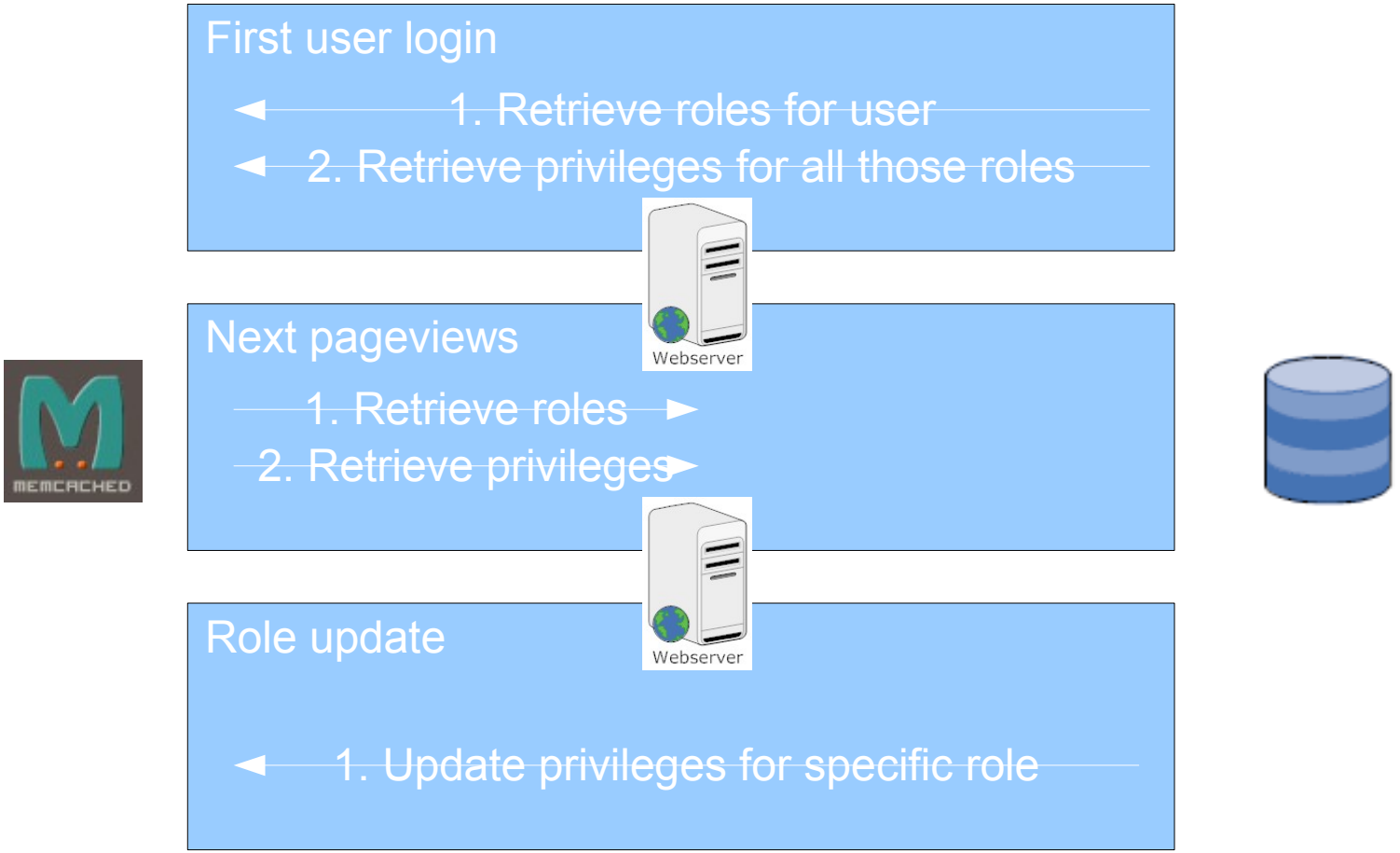
- All subsequent pageviews : 1 cache request

What's in the cache ?

Entry	Data
acl_user_3_%_%_%	1
acl_user_1_cms_article_edit	1
acl_user_1_admin_destroy_planet	0

Problem : what if we add a privilege to a role ?
→ All cached entries for all users should be refreshed (ouch !)

Caching - Option 3 - denormalize in cache



What's in the cache ?

User's roles

Entry	Data
acl_user_3	3, 1, 4
acl_user_1	3, 1, 2
acl_role_1	a:3:{i:0;s:45:"a:2:{i:0;s:5:"%_%_ %";}";i:1;i:1308106740;i:2;s:6:"604800";}

Privileges listed in a role

Caching - Option 3 - denormalize in cache

- 1 DB query per user at login (retrieve user's roles)
- User with 20 privileges → just 1 DB query
- 1 cache query per pageview + 1 per role

- Good :
  - Less queries on DB
  - Less data in the cache (only roles, not full privileges of each user)
  - Add a privilege to a role → update the role only
- Bad :
  - More queries on cache

Choice : depends on where your highest load is
(but memory is cheap and Memcache is fast !)

Caching - let's have a look

Managing the roles / privileges

- Zend_Acl : manual typ(o)ing
- Goals :
  - automation
  - easy management

→ Reflection

Reflection ?

- Used to inspect objects during run-time
- Available since PHP 5.0
- Can be applied to :
  - Classes
  - Objects
  - Methods
  - Functions
  - Properties
  - Extensions

Reflection - example

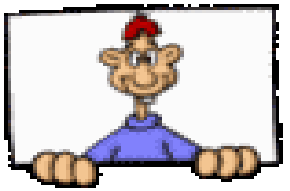
```php
<?php
class Test
{
    static public function testMe($reason)
    {
        echo 'I have a reason : ' . $reason;
    }
}

$reflector = new ReflectionClass('Test');
echo 'Class name : ' . $reflector->getName() . "\n";
echo "Methods : \n";
var_dump($reflector->getMethods());
```

Outputs :

```
Class name : Test
Methods :
array(1) {
  [0]=>
  &object(ReflectionMethod)#2 (2) {
    ["name"]=>
    string(6) "testMe"
    ["class"]=>
    string(4) "Test"
  }
}
```

Backend interface with reflection

Questions ?



cube
it's in the dot

Contact

- Web http://techblog.wimgodden.be
- Slides http://www.slideshare.net/wimg
- Twitter @wimgtr
- E-mail wim.godden@cu.be

Thanks !

Feel free to rate my webinar at http://tinyurl.com/acltalk

cube
it's in the dot