

# Създаване на уеб приложения с Catalyst и Perl

Свилен Иванов  
([svilen@netclime.com](mailto:svilen@netclime.com))

# Какво ще Ви представя днес?

- Rose::DB::Object – модул за обектна абстракция на релационна база данни (Object/Relational mapping)
- Catalyst – средство за разработка на елегантни уеб приложения по шаблона Model/View/Controller
- Създаване на примерно приложение - MyBlog

# Какво е Rose::DB::Object?

- Представяна един ред от таблица като Perl обект
  - Подобни: Class::DBI, DBIx::Class, Tangram, SPOPS
- Възможности
  - Конфигурирането на обекта може да става автоматично – по структурата на базата
    - Може и изрично да се конфигурира от програмиста
  - Поддържани СУБД
    - MySQL, PostgreSQL, Informix, SQLite; експериментално – Oracle
  - Прекрасна документация
  - Много добра поддръжка от разработчика на модула
    - отзивчив пощенски списък, IRC

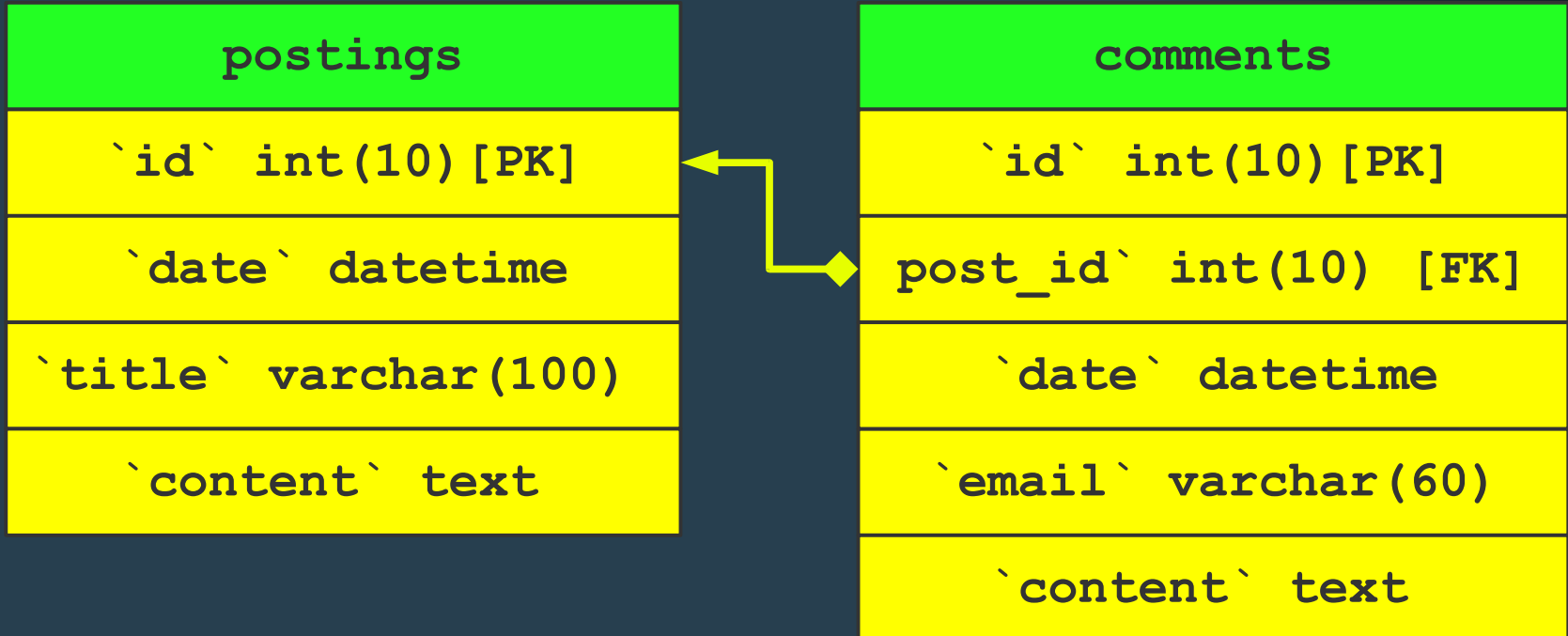
# Какво е Rose::DB::Object?

- Класовете-наследници наследяват допълнителни методи за:
  - Достъп до стойността всяка колона, както и нейната промяна (accessor/mutators)
  - Записване на обекта в таблица или извличане на обект от таблица по даден критерий (save/load). Базовия клас се грижи дали да създаде или обнови даден ред от таблицата.
  - Ако към текущия обект има връзки с други обекти от типа „1:1“, „\*:1“, те могат да се заредят и инициализират заедно с текущия
- Дава възможност за представяне на различните типове колони (datetime, bitfield) като perl обекти с цел лесното им манипулиране

# Какво е Rose::DB::Object?

- Допълнителен модул RDBO::Manager:
  - Зареждане на множество обекти по произволен критерий или подредба
  - Итериране на множеството обекти като се извличат данните от базата на всяка итерация с цел оптимизация
  - Извличане на множеството обекти заедно с обектите към или от които имат връзка с една заявка (RDBO::Manager се грижи да генерира SQL заявка с правилните JOIN изрази)
  - Преброяване, изтриване или обновяване на обекти по определен критерий
- Примерно приложение с RDBO – MyBlog
  - Избрал съм блог, защото това е модерно. Ако днес беше 1999 година, щях да ви представям Интернет магазин за продажба на лед или друга безумна dotcom идея :)

# MyBlog – Структура на базата данни



# MyBlog - Конфигуриране

- Конфигуриране достъпа до базата данни

```
package MyBlog::Model::DB;  
use base (Rose::DB);
```

```
__PACKAGE__->register_db(  
    domain    => 'development',  
    type      => 'testing',  
    driver    => 'mysql',  
    database  => 'test',  
    host      => 'localhost',  
    username  => 'root',  
    password  => '',  
    server_time_zone => 'UTC',  
);  
__PACKAGE__->default_domain('development');  
__PACKAGE__->default_type('testing');
```

# MyBlog - Конфигуриране

- Създаване на базов обект RDBO

```
package MyBlog::Model::Base;  
use base (Rose::DB::Object);  
use MyBlog::Model::DB;
```

```
sub init_db {  
    MyBlog::Model::DB->new  
}
```

- Конфигуриране на обект Posting

```
package MyBlog::Model::Posting;  
use base (MyBlog::Model::Base);
```

```
__PACKAGE__->meta->table('postings');  
__PACKAGE__->meta->auto_initialize;  
__PACKAGE__->meta->make_manager_class('postings');
```



# MyBlog - добавяне на тема

- Примерно използване

```
use MyBlog::Model::Posting;  
use DateTime;
```

```
my $posting = MyBlog::Model::Posting->new();  
$posting->title('Linux-BG conference!');  
$posting->date(DateTime->now);  
$posting->save();
```

```
mysql> select * from postings\G  
***** 1. row *****  
    id: 1  
   date: 2006-04-11 20:59:54  
  title: Linux-BG conference!
```

# MyBlog – добавяне на коментар

- Конфигуриране на Comment

```
package MyBlog::Model::Comment;  
use base (MyBlog::Model::Base);
```

```
__PACKAGE__->meta->table('comments');  
__PACKAGE__->meta->auto_initialize;
```

- Добавяне на коментар

```
my $posting = MyBlog::Model::Posting->new();  
$posting->title('Linux-BG conference!');
```

```
my $comment = MyBlog::Model::Comment->new();  
$comment->email('mlekonadoi@tkzs.org');  
$comment->content('I liked it');
```

```
$posting->add_comments($comment);  
$posting->save();
```

# MyBlog – добавяне на коментар

```
mysql> select * from postings\G
***** 1. row *****
      id: 1
     date: NULL
    title: Linux-BG conference!
 content:
1 row in set (0.01 sec)
```

```
mysql> select * from comments\G
***** 1. row *****
      id: 1
 post_id: 1
     date: NULL
   email: mlekonadoi@tkzs.org
 content: I liked it
1 row in set (0.00 sec)
```

# MyBlog – Списък на всички теми

- Автоматично създаден клас за работа с много обекти

```
my $postings = MyBlog::Model::Posting::Manager->
    get_postings(
        with_objects => 'comments',
        sort_by => 'date DESC',
        query => [
            date => { gt => '2006-03-11' },
        ]);

foreach my $posting (@{$postings}) {
    print $posting->title;
    foreach my $comment ($posting->comments) {
        print $comment->content;
    }
}
```

**Нека да направим този модел като уеб приложение!**

# Какво е Catalyst

- Catalyst е среда за разработка на уеб приложения по шаблона MVC:
  - Основни принципи – DRY, KISS, TIMTOWTDI (*eee, tu се оля!*)
  - Представява „скеле“ за създаване на приложение (Controller)
    - не притежава вградени View или Model
    - използват се налични, добре познати и доказали се модули от CPAN – DBIx::Class, Class::DBI, Template Toolikt, Mason
  - Мултиплатформен
    - OS - Linux, Windows, MacOS, FreeBSD, Solaris
    - Apache, lighttpd, IIS, вграден сървър за тестване
    - mod\_perl, FastCGI, plain CGI,

# Catalyst – Възможности

- Лесен за инсталиране – през CPAN
- Удобни инструменти за тестване и разработка
  - Мигновено създаване на ново, работещо приложение с помощта на скрипт
  - Вграден тестов уеб сървър за тестване на приложението, саморестартиращ се при промяна на Perl модул от приложението
  - Ако възникне грешка при изпълнение се показва страница с информация за грешката, както и текущия контекст – HTTP параметри, заредени обекти, cookies, и т.н
  - Средства за създаване на автоматични тестове
- Разширяване чрез plugin като блокчета Lego®
  - Можете да създадете plugin който да се използва в различни проекти
  - Видове - сесии, кеширане, оторизация, контрол на достъпа....

# Дефиниране на действия (actions)

- Контролера е всеки модул наследяващ `Catalyst::Controller`
- Свобода при избирането на URL за всяко действие
  - Задаването става чрез:
    - интерпретиране на името на контролера и името на метода

```
package MyApp::C::Foo;  
  sub hello :Local {  
    my ($self, $c) = @_;  
    $c->res->body("Hello world!");  
  }  
  # Answers /foo/hello
```



# Още действия (за напреднали)

—атрибути на методите на контролера

```
package MyApp::C::Foo;
```

```
sub login : Global {           # /login
sub one   : Path("foo") {      # /foo/foo
sub two   : Path("/bar") {     # /bar
sub three : Path {            # /foo/*
```

```
# Съвпаденията на RE са достъпни в $c->snippets
sub foo :Regex("/foo/(.*)/bar") {
```

```
# /foo/baz/(.*)/quux
sub bar :LocalRegex("baz/(.*)/quux") {
```

# Верига на изпълнение

- Специални действия
  - `index` - изпълнява се само при `/foo`
  - `default` - изпълнява се при всяко `/foo/*`
  - `begin/end` - изпълнява се в началото/края на заявката
  - `auto` – преди изпълнението на действието, което отговаря на заявката
  - .... но Вие имате и свободата да дефинирате в какъв ред да се изпълняват

# MyBlog като Catalyst приложение

```
$ cpan Catalyst
```

```
...
```

```
$ catalyst.pl MyBlog
```

```
...
```

```
$ cd MyBlog
```

```
$ script/myblog_server.pl -r
```

```
...
```

```
[Thu Apr 13 23:58:20 2006] [catalyst] [info] MyBlog
```

```
powered by Catalyst 5.66
```

```
You can connect to your server at http://pink:3000
```

MyBlog on Catalyst 5.66 - Konqueror

Location Edit View Go Bookmarks Tools Settings Window Help

Location: <http://pink:3000/>

## MyBlog on Catalyst 5.66


Welcome to the wonderful world of Catalyst. This [MVC](#) framework will make web development something you had never expected it to be: Fun, rewarding, and quick.

### What to do now?

That really depends on what **you** want to do. We do, however, provide you with a few starting points.

If you want to jump right into web development with Catalyst you might want to check out the documentation.

`perldoc Catalyst::Manual::Intro`



The Catalyst logo features a red central circle connected by red lines to three smaller red circles, all enclosed in a white rounded square with a grey border. Below the graphic, the word "catalyst" is written in a bold, lowercase, sans-serif font, and "Web Framework" is written in a smaller, regular, lowercase, sans-serif font below it.

# MyBlog като Catalyst приложение

- Създаване на Controller

```
$ script/myblog_create.pl controller MyBlog
```

```
lib/MyBlog/Controller/MyBlog.pm
```

- Добавяме действие за списък на темите

```
sub list : Local {  
    my ( $self, $c ) = @_  
    $c->stash->{postings} =  
        MyBlog::Model::Posting::Manager->get_postings (  
            with_objects => 'comments',  
            sort_by => 'date DESC' );  
}
```

# MyBlog като Catalyst приложение

- Добавяме действие по подразбиране

```
sub default : Private {  
    my ( $self, $c ) = @_;  
    $c->forward('list');  
}
```

- Създаване на View с Template Toolkit

```
$ script/myblog_create.pl view TT TT
```

- Указваме, че визуализирането на данните ще се извърши чрез Template Toolkit

```
sub end : Private {  
    my ( $self, $c ) = @_;  
    $c->stash->{template} = 'blog.tt';  
    $c->forward('TT');  
}
```

MyBlog on Catalyst 5.66 - Konqueror

Location Edit View Go Bookmarks Tools Settings Window Help

Location: http://pink:3000/myblog

Couldn't render template "file error - blog.tt: not found"

### Request

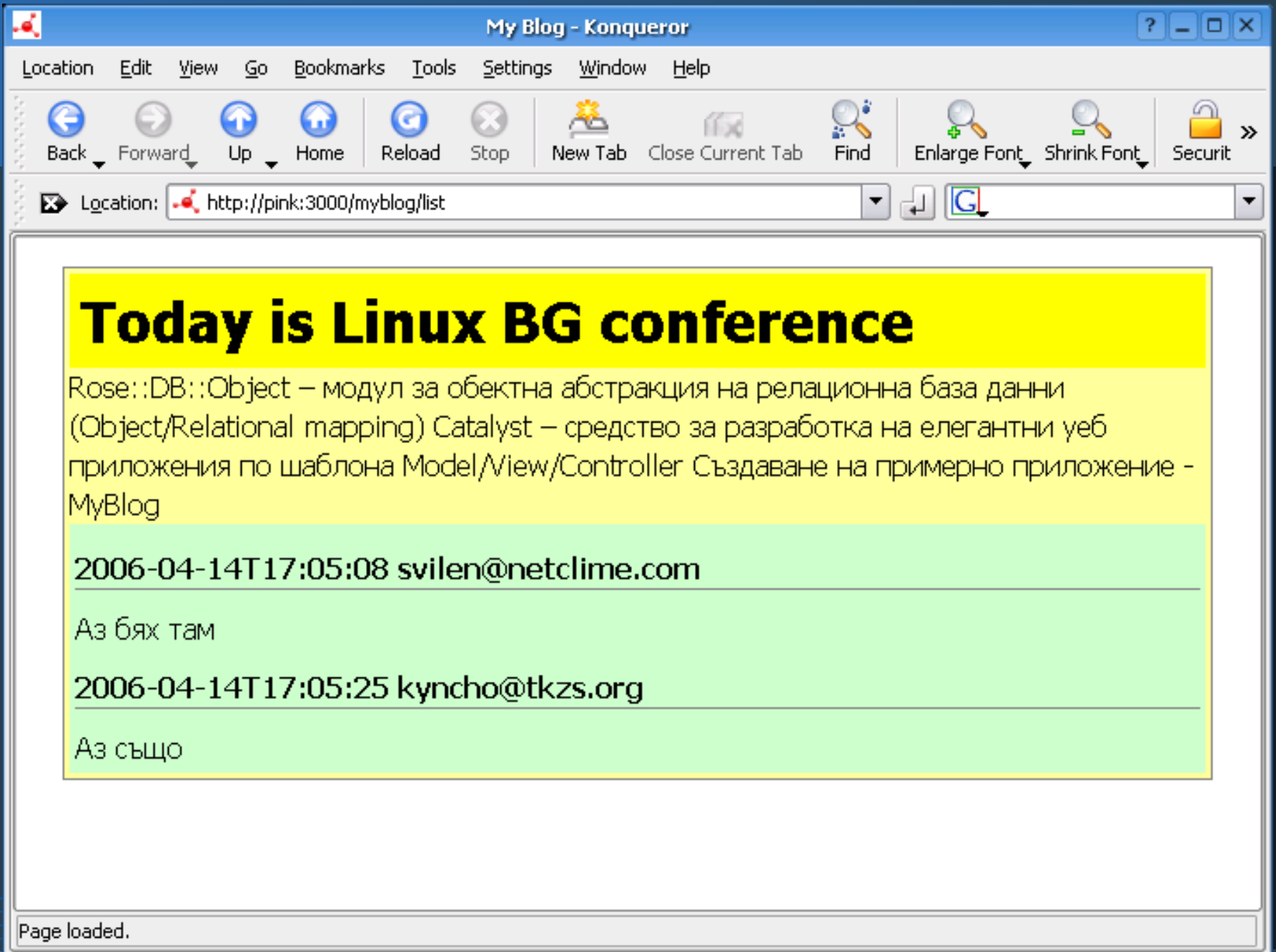
```
bless( {
    'protocol' => 'HTTP/1.0',
    'headers' => bless( {
        'accept-charset' => 'iso-8859-1,
utf-8;q=0.5, *;q=0.5',
        'user-agent' => 'Mozilla/5.0
(compatible; Konqueror/3.4; Linux) KHTML/3.4.2 (like Gecko)',
        'connection' => 'Keep-Alive',
        'accept' => 'text/html,
image/jpeg, image/png, text/*, image/*, */*',
        'accept-language' => 'en',
        'accept-encoding' => 'x-gzip,
x-deflate, gzip, deflate',
        'host' => 'pink:3000'
    }, 'HTTP::Headers' )
}
```

# Какво стана?

- Не сме създали TT шаблон - blog.tt

```
[% FOREACH posting = c.stash.postings() %]
  <div class="posting">
    <h1> [% posting.title() | html %]</h1>
    [% posting.content() | html %]
    <div class="comment">
      [% FOREACH comment = posting.comments() %]
        [% comment.date() %] [% comment.email() %]
        [% comment.content() %]
      [% END %]
    </div>
  </div>
[% END %]
```





# Да добавим коментари

- Създаваме ново действие в контролера

```
sub add_comment : Local {  
  my ( $self, $c ) = @_;  
  MyBlog::Model::Comment->new (  
    'email'    => $c->req->param('comment_email'),  
    'post_id' => $c->req->param('post_id'),  
    'content' => $c->req->param('comment_content'),  
    'date'    => DateTime->now() )->save();  
  $c->res->redirect('list');  
}
```

# Да добавим коментари

- Създаване на форма

```
<form method="POST"
  action="[ % c.uri_for('add_comment') %]">
  <input type="hidden" name="post_id"
    value="[ % posting.id() %]">
  <input type="text" name="comment_email">
  <textarea name="comment_content"></textarea>
  <input type="Submit" value="Add Comment">
</form>
```

My Blog - Konqueror

Location Edit View Go Bookmarks Tools Settings Window Help

Back Forward Up Home Reload Stop New Tab Close Current Tab Find Enlarge Font Shrink Font Security Dov

Location: http://pink:3000/myblog/list

## Today is Linux BG conference

Rose::DB::Object – модул за обектна абстракция на релационна база данни (Object/Relational mapping) Catalyst – средство за разработка на елегантни уеб приложения по шаблона Model/View/Controller Създаване на примерно приложение - MyBlog

2006-04-14T17:05:08 svilen@netclime.com

Аз бях там

2006-04-14T17:05:25 kyncho@tkzs.org

Аз също

Page loaded.

# Заклучение

- Обектната абстракция на релационна база данни значително улеснява ежедневни, рутини задачи на програмиста, същевременно остава свободата за сложни заявки
- Catalyst предлага изключително лесен начин за създаване на уеб приложения като се базира на съществуващи SRAN модули
- Създаването на уеб приложения е забавно с помощта на Catalyst :)