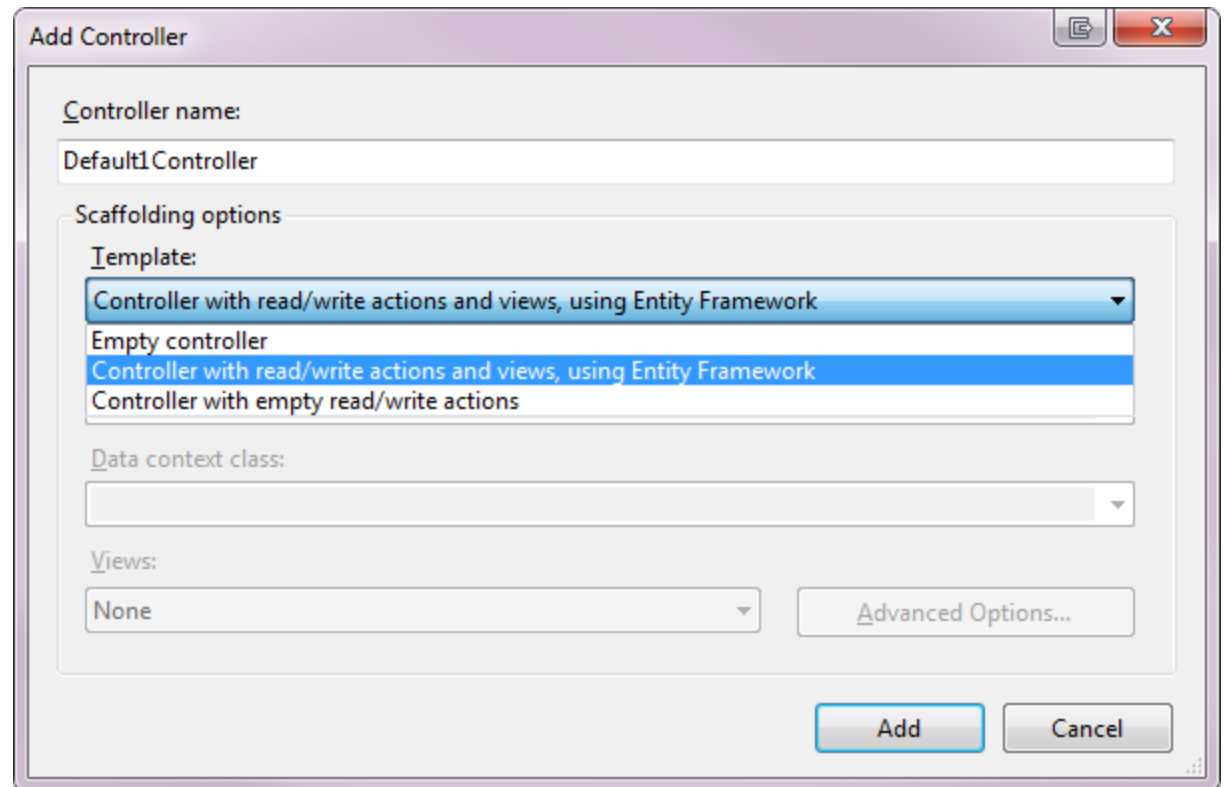


Rails



Scaffolding

- Manual setup is no fun



The image shows a 'Add Controller' dialog box with the following fields and options:

- Controller name:** A text box containing 'Default1Controller'.
- Scaffolding options:**
 - Template:** A dropdown menu with four options: 'Controller with read/write actions and views, using Entity Framework' (selected), 'Empty controller', 'Controller with read/write actions and views, using Entity Framework', and 'Controller with empty read/write actions'.
 - Data context class:** An empty dropdown menu.
 - Views:** A dropdown menu with 'None' selected.
- Buttons:** 'Add' and 'Cancel' buttons at the bottom right, and an 'Advanced Options...' button next to the Views dropdown.

Scaffolding

- Ruby has scaffolding out-of-the-box
- Scaffolding is a quick way to generate major pieces of an application
- Create models, view, controllers and migrations from a single action
- Less work that we have to do

Environments

- **Environments**
 - Dev
 - Test/QA
 - Staging
 - Production
- **Rails commands let you specify the target environment**
 - `Rails server -e production`
 - `Rake db:migrate RAILS_ENV=test`
- **Default environment is development**

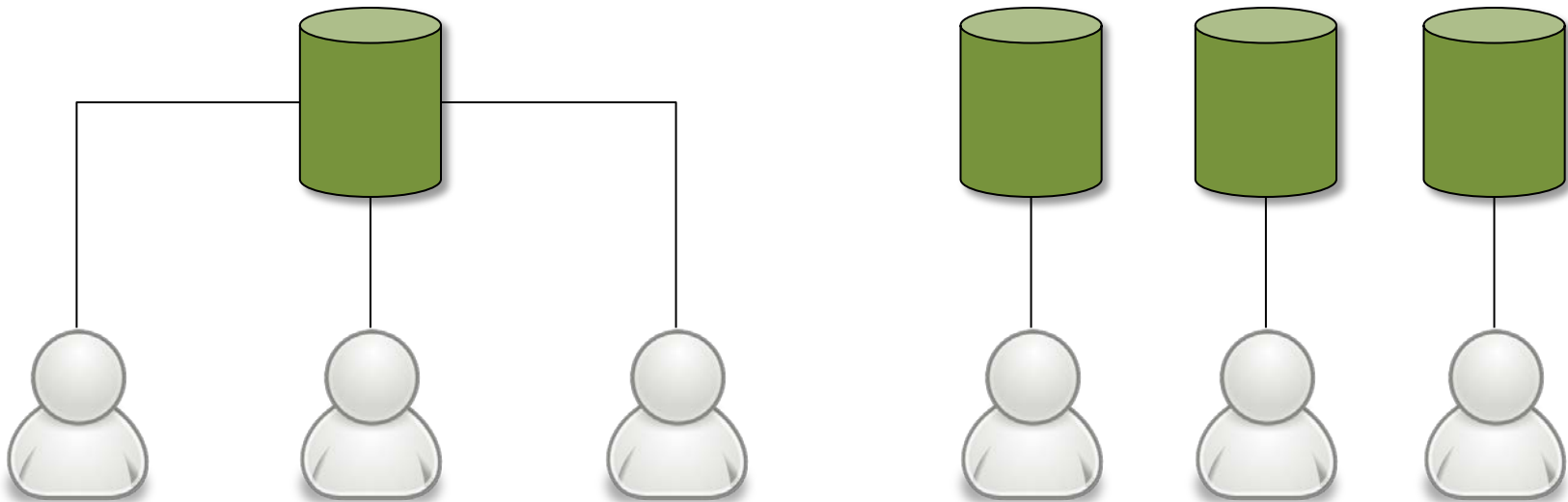
```
development:
  adapter: sqlserver
  database: MyBlog
  dataserver: '127.0.0.1\SQLEXPRESS'
  username: bloguser
  password: '1234'

test:
  adapter: sqlserver
  database: Rails4NET
  dataserver: localhost
  instance: SQLEXPRESS
  mode: dblib
  port: 1433
  username: sa
  password: "1234"

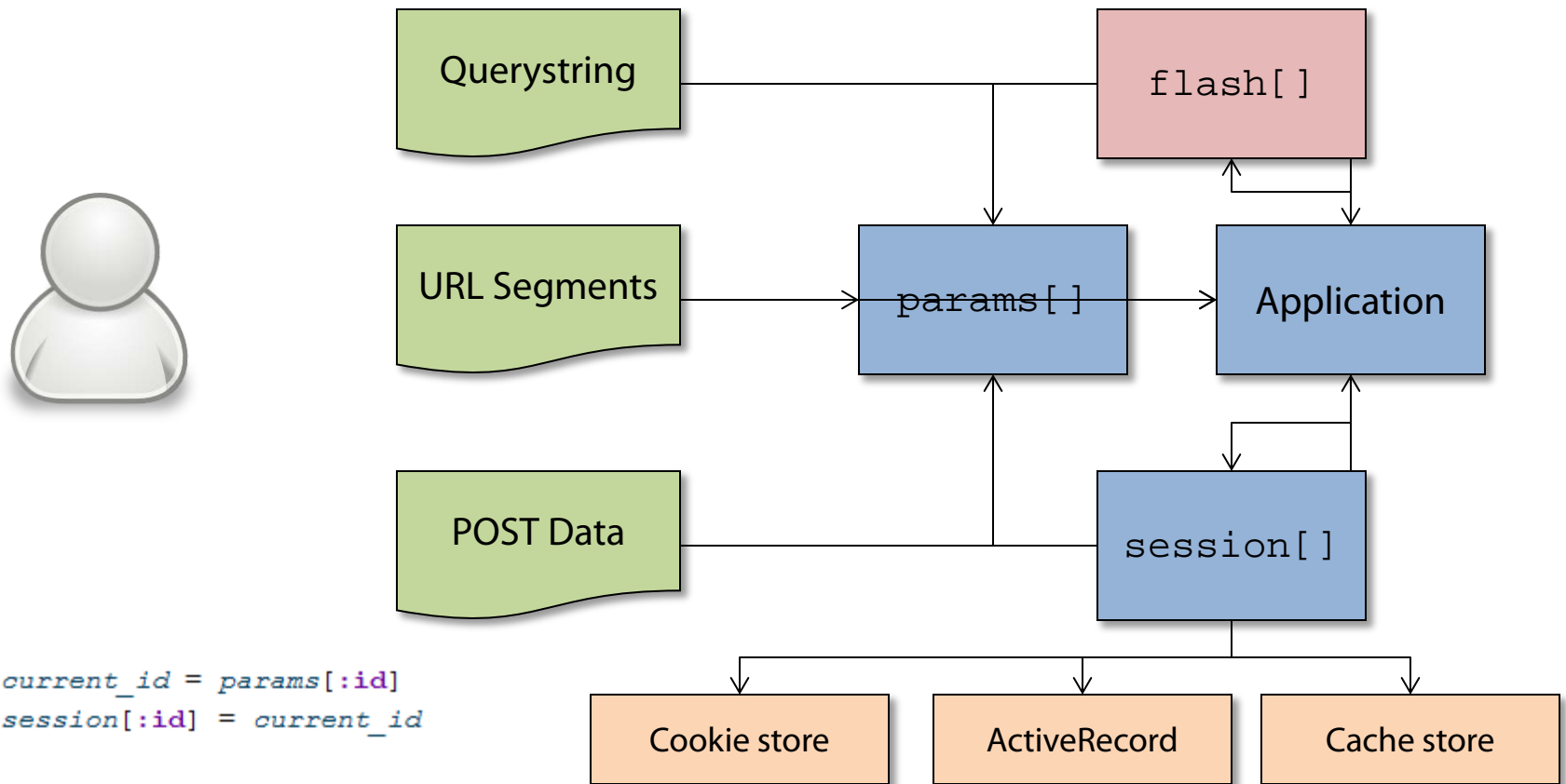
production:
  adapter: sqlserver
  database: Rails4NET
  dataserver: localhost
  instance: SQLEXPRESS
  mode: dblib
  port: 1433
  username: sa
  password: "1234"
```

Migrations

- Manage the evolution of database schema
- Schema is described in self-contained classes
- Automatic synchronization, even if many versions behind
- Everyone can stay/get up-to-date in an organized way



User data



```
current_id = params[:id]
session[:id] = current_id
```

config/initializers/session_store.rb

User data

- **Value**

- Cookie's value or list of values

- **Path**

- The path for which the cookie applies

- **Domain**

- The domain for which the cookie applies

- **Expires**

- The time at which the cookie will expire

- **Secure**

- Whether the cookie is only transmitted over HTTPS

- **HTTP Only**

- Where the cookie is accessible via scripting or only HTTP

```
cookies[:remember_me] = true
remember_me = cookies[:remember_me]
cookies.delete :remember_me
```

```
cookies[:login] = {
  :value => 'johndoe555',
  :path => '/',
  :domain => 'localhost',
  :expires => Time.now + 3600,
  :secure => true,
  :httponly => false
}
```

Summary

- **Rails installs from a Gem**
- **Build a working application fast**
 - `Rails new [app_name]`
- **WEBrick is a web server that comes with Ruby**
 - Requires no setup or configuration
- **Scaffolding generates all of what we need with a single command**
- **Use TinyTDS and ActiveRecord-SQLServer-Adapter to with MSSQL**
- **Routing is a central part of an MVC application**
- **Store and retrieve user data using `params[]`, `session[]`, `cookies[]`**
- **Session data is stored to the cookie by default, but can be changed**