

Framework Goals & Benefits

Build a More Robust, Scalable and Maintainable RIA

Framework at a Glance

PureMVC is a lightweight framework for creating applications based upon the classic Model-View-Controller design *meta*-pattern.

This free, open-source framework is implemented in ActionScript 2 and 3, Java, C# and a number of other popular programming languages. This allows development on a wide variety of platforms including:

- **Mobile Environments:** FlashLite, .NET Compact Framework, J2ME
- **Server Environments:** ColdFusion, J2EE, PHP, Python
- **Browser Environments:** Flash/Flex, JavaFX, Silverlight
- **Desktop Environments:** .NET, AIR, FLASH, J2SE

Goals

The PureMVC framework has a very narrow main goal: to help you separate your application's coding concerns into three discrete tiers; Model, View and Controller.

To do this well and push the framework to a stable state is single guiding intent.

Extending the usefulness of the framework by providing utilities and demos that illustrate the overall best practices for use of the framework and utilities is the next.

Other goals driving the initial AS3 reference implementation were:

Goals

- Simple but appropriate framework scope.
- Reduce confusion over application layers, class roles, responsibilities and collaborations.
- Balance speedy implementation with scalability and maintainability.
- Provide clear and adequate documentation.
- Avoid platform dependencies.
- Hide complexity from developer.
- A well-planned framework that doesn't require continual upgrade and maintenance to keep up with new features, deprecations and refactorings.

Benefits

The choice of framework for your application heavily influences your architecture, thereby affecting the total cost of ownership by impacting future maintainability. Choose wisely.

Implementing your application with PureMVC brings the following benefits:

Pattern-based

- Supported by time-proven design patterns that can be implemented in any OOP environment.
- Easy to understand.

PureMVC is a free, open source framework created and maintained by Futurescale, Inc. Copyright © 2006-08. Some rights reserved.
Reuse is governed by the Creative Commons 3.0 Attribution Unported License. PureMVC, as well as this documentation and any training materials or demonstration source code downloaded from Futurescale's websites is provided 'as is' without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of fitness for a purpose, or the warranty of non-infringement.

Benefits

Useful implementation classes

- Framework can be used 'out-of-box' with minimal implementation requirements.
- Façade class provides a single collaborator for communication with the core framework.

Loosely-coupled architecture

- Promotes reusability of view components and model data objects and services.
- Incorporates publish / subscribe-style notifications.
- Creates clear separation of client-tier coding concerns.

Programmed to interfaces

- Framework supports extensibility by sub- classing or interface implementation.
- Interfaces defined for all framework classes.
- All classes built with extension in mind; protected methods and variables and interface parameters in method signatures.
- Framework provides for future extension via other libraries by grouping supporting patterns into a separate package, and making no core dependencies on the pattern packages.

PureMVC is a free, open source framework created and maintained by Futurescale, Inc. Copyright © 2006-08. Some rights reserved.
Reuse is governed by the Creative Commons 3.0 Attribution Unported License. PureMVC, as well as this documentation and any training materials or demonstration source code downloaded from Futurescale's websites is provided 'as is' without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of fitness for a purpose, or the warranty of non-infringement.

Benefits

Well documented

- Source code freely available.
- Complete API documentation.
- Conceptual and UML Diagrams.
- Unit tests for all classes and methods.
- Plenty of demos show the basics and demonstrate best practices
- *Implementation Idioms & Best Practices* document by the architect.
- Professional courseware under development has already been beta-tested by hundreds of students.

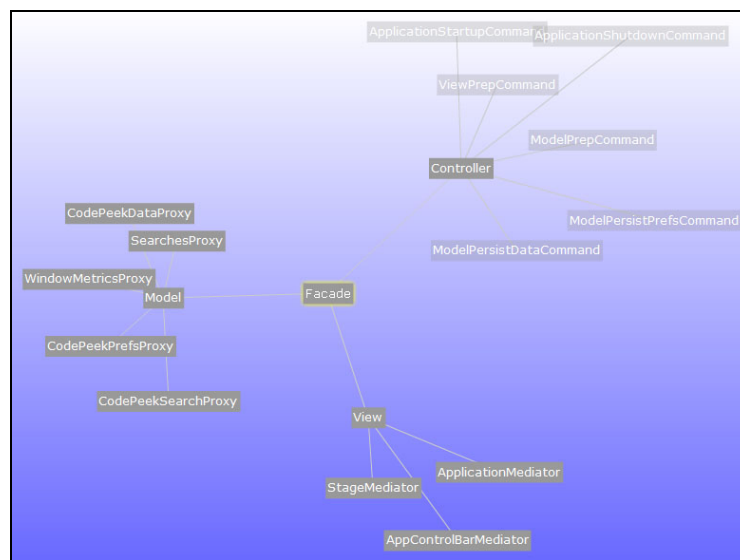


Figure 1 Making the separation of Model, View and Controller interests eliminates the most harmful responsibility misplacement issues that hamper scalability and maintainability.

PureMVC is a free, open source framework created and maintained by Futurescale, Inc. Copyright © 2006-08. Some rights reserved. Reuse is governed by the Creative Commons 3.0 Attribution Unported License. PureMVC, as well as this documentation and any training materials or demonstration source code downloaded from Futurescale's websites is provided 'as is' without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of fitness for a purpose, or the warranty of non-infringement.