

# Apache Flex ... ... and the imperfect Web

Presented by  
Sebastian Mohr

July 5th 2012, Berlin

# About me

- HTML / Javascript Developer (2000 - 2006)
- Flex Developer since Summer 2006
- Mate, Robotlegs, Parsley, Spring AS, Swiz
- Study of Cognitive Science, Osnabrueck
- W3C Fan, Semantic Web, Open Web Technologies
- Apache Flex PPMC since November 2011

Homepage: <http://www.masuland.com/>

Twitter: <http://twitter.com/masuland>

LinkedIn: <http://www.linkedin.com/in/masuland>

# The imperfect Web

Gary Oldman explains why "Graffiti" - or the "imperfect Web" respectively - is great for the world ...



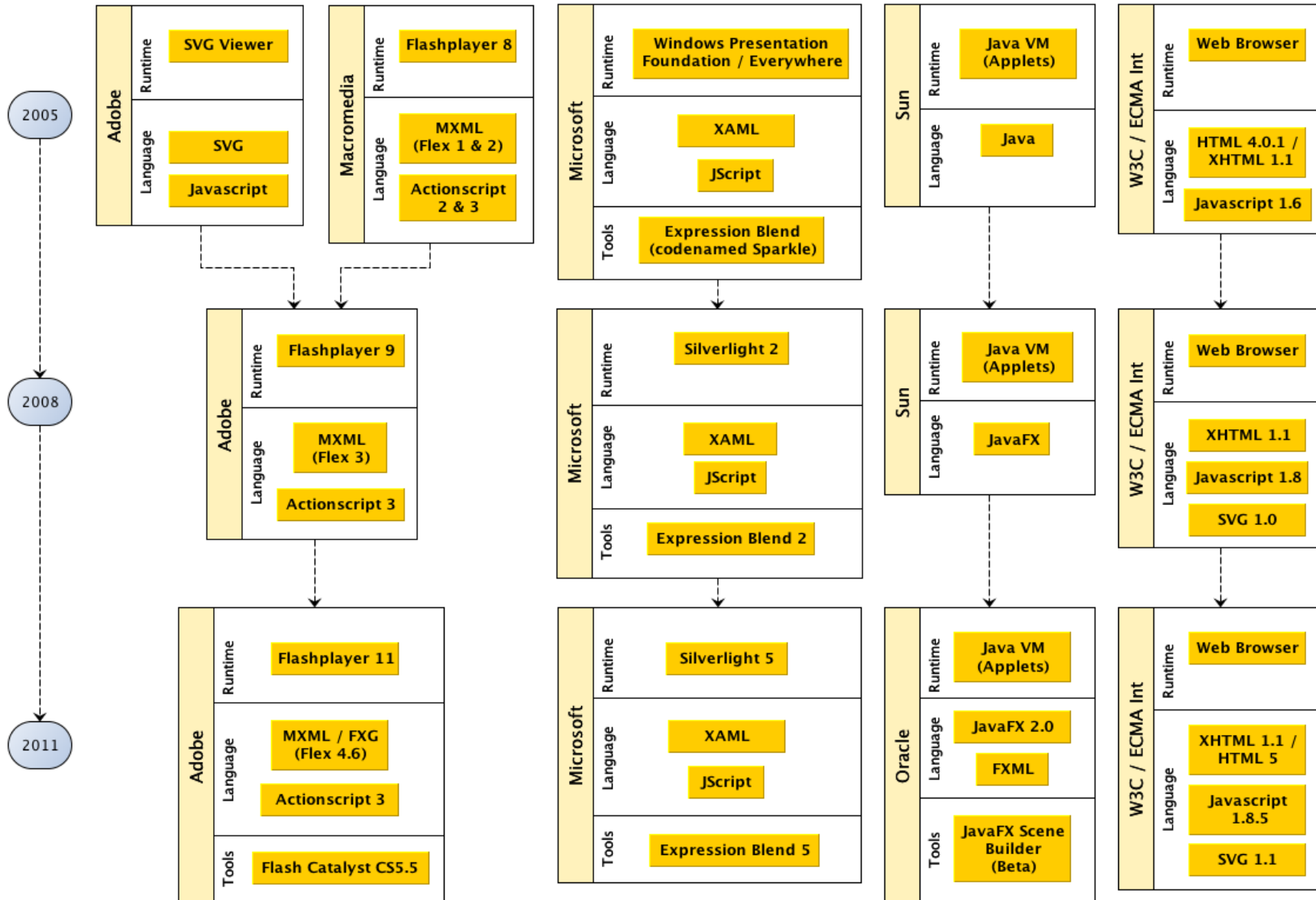
Watch video: <http://www.youtube.com/watch?v=Tt1W0F0yObg>

# The imperfect Web is:

- Webrowsers: Firefox, Chrome, Safari, Opera
- Markups: XHTML, SVG, SMIL, MXML (Flex), XAML (Silverlight), FXML (JavaFX), WebGL, X3D, XForm, XPath, XML, XSLT, XQuery, RDF, OWL, ...
- Script-Engines: ECMAScript, Javascript, Actionscript, JScript
- Ajax & JS-Libs: JQuery, ExtJS, Dojo, Prototype, MochiKit, PhoneGap, Rhomobile, Sencha, ...
- Devices: Desktop, Mobile, TV
- Mobile OS: iOS, Android, Windows Mobile

# Web History (2005 - 2011)

Web History of HTML/SVG/Javascript, Flash, Silverlight & Java (2005-2011)



# The Web Today

	<b>HTML / Javascript</b>	<b>Flashplayer 11</b>	<b>Silverlight 5</b>	<b>Java (Applets)</b>
<b>Runtime Features</b>	Dependent on Web Browser	GPU support	GPU support, Multithreading	GPU support, Multithreading
<b>Markup Language</b>	XHTML 1.1, HTML5, SVG	FXG 2.0, MXML	XAML	FXML
<b>Scripting Language</b>	Javascript	Actionscript	JScript	Java, JRuby, Groovy
<b>Interaction Design Tools</b>	Adobe Muse, Adobe Edge, Adobe Proto	Adobe Flash Catalyst, Adobe Flash Pro	Microsoft Expression Blend	JavaFX Scene Builder
<b>Developer Tools</b>	Eclipse, Adobe Dreamweaver	Adobe Flash Pro, Adobe Flash Builder	Microsoft Visual Studio	Eclipse

# Thank you Tim for ...

+++ HTTP

+++ `www.domain.com`

--- HTML



<http://www.ecma-international.org>



Tim Berners-Lee, [http://en.wikipedia.org/wiki/Tim\\_Berners-Lee](http://en.wikipedia.org/wiki/Tim_Berners-Lee)

# Flash Everywhere?



- Adobe AIR 4+ for Mobile
- Flashplayer 12+ with Multithreading and better GPU-support
- Flashplayer 12+ great for Flash-Games and Apache Flex
- Adobe Open Screen Project?
- Faster Mobile-CPU's each year → Flashplayer 12+ for Mobile again?
- Flash-2-HTML Converter?



# What should we occupy?



Occupy Wallstreet

<http://www.occupywallst.org>



Occupy Flash & Occupy HTML 5

<http://www.occupyflash.org>

<http://www.occupyhtml.org>



Occupy W3C & Occupy ECMA Int.

Not occupied yet:  
<http://www.w3c.org>

Not occupied yet:  
<http://www.emca-international.org>



# Where is Apache Flex in 2050?

- Apache Flex might not be necessary if:
  - Single Webbrowser / Webengine (Webkit?)
  - RIA-HTML (Audio, Video, 3D, Text) (HTML 6?)
  - Binary HTML (like SWF?)
  - Semantic Web interface included into HTML
  - Stronger Javascript / ECMAScript - Engine
    - Typed Objects
    - Packages / OOP

Read article: <http://masuland.wordpress.com/2011/11/28/where-could-flash-coding-be-in-the-year-2050/>



# Spoon Foundation Ecosystem



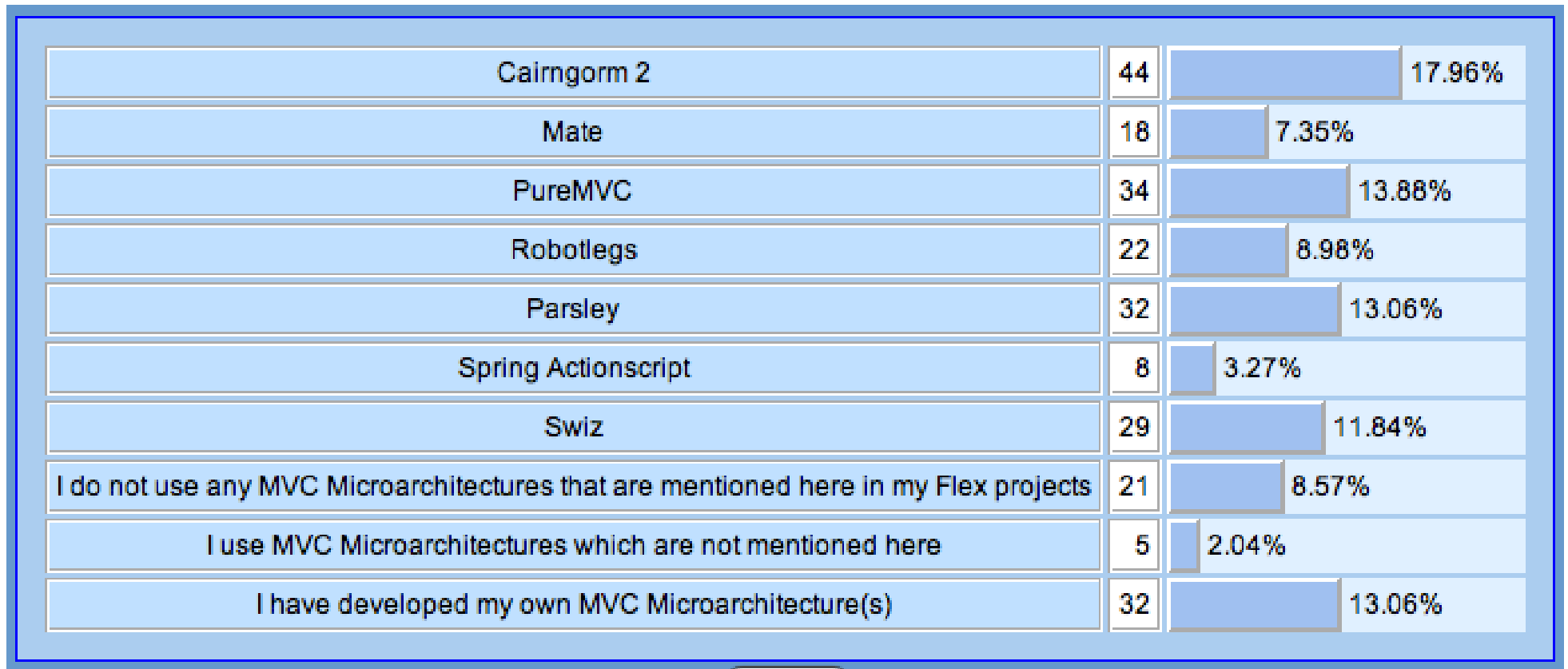
See presentation by Justin Mclean:

<https://docs.google.com/presentation/pub?id=1SsVcr2SLkBJMXk1N6U3-HpxloSUUa4pBX-2f4GrRL5M#slide=id.p57>



# Personal Survey (1)

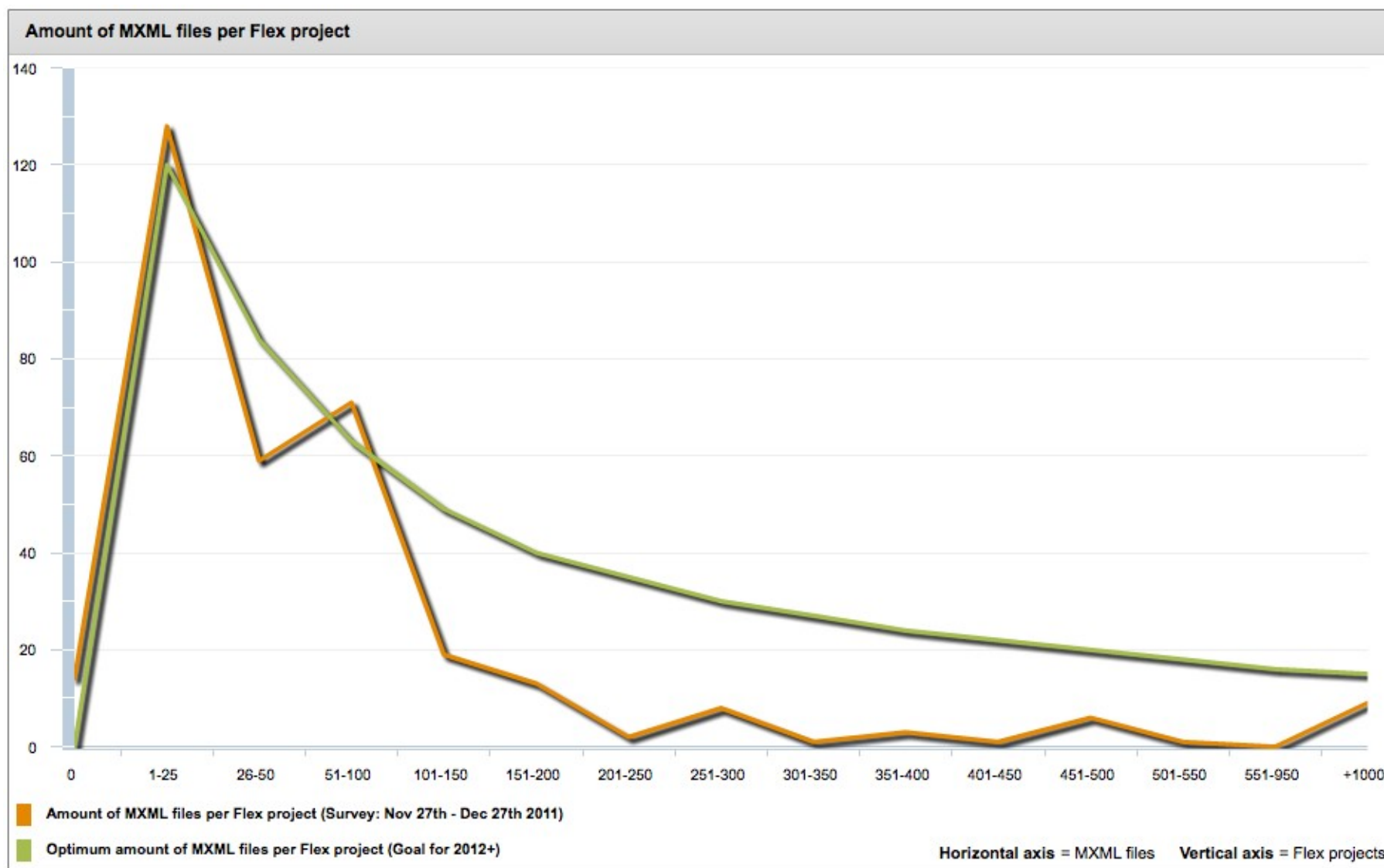
Which Model-View-Controller (MVC) Microarchitectures have you used in your Flex projects?



See personal survey: <http://code.google.com/p/masuland/wiki/Statistics>

# Personal Survey (2)

How many MXML files do you have in your latest Flex projects?



See personal survey: <http://code.google.com/p/masuland/wiki/Statistics>



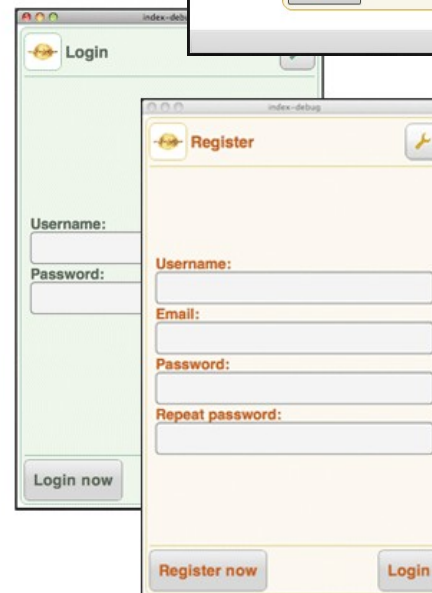
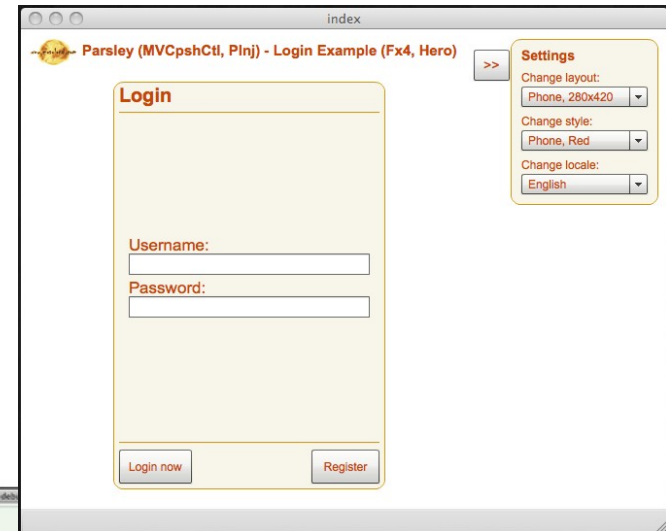
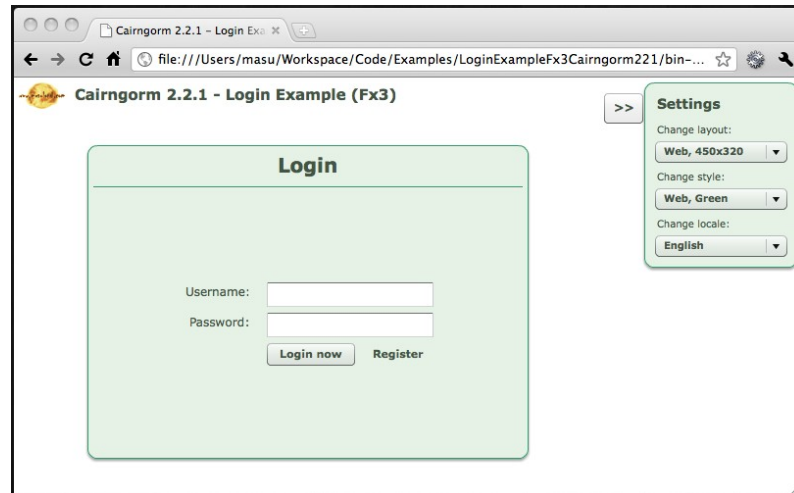
# Goals for Apache Flex

- Strengthening large-scale Flex applications
- Highly productive and fast development environment
- Flex code should be easily mergeable across different flex projects worldwide
- Flex code should be based on coding standards
- Flex code should be sustainable and easily maintainable
- There should be more highly skilled Flex Developers around
- Developer tools should be easy to be used and should help the designer/developer to be more productive
- The Flashplayer - "Actionscript Virtual Machine" (AVM) respectively - should be as powerful as the "Java Virtual Machine" (JVM)
- The Adobe Flash Catalyst should be as powerful as Microsoft Expression Blend

Read article: <http://code.google.com/p/masuland/wiki/WhatsWrongWithFlex>



# Masuland „Login Example“ for Web, Desktop and Mobile



See code examples:  
<http://code.google.com/p/masuland/wiki/LoginExample>

# Is coding „fun“?

## At the beginning ... probably true!

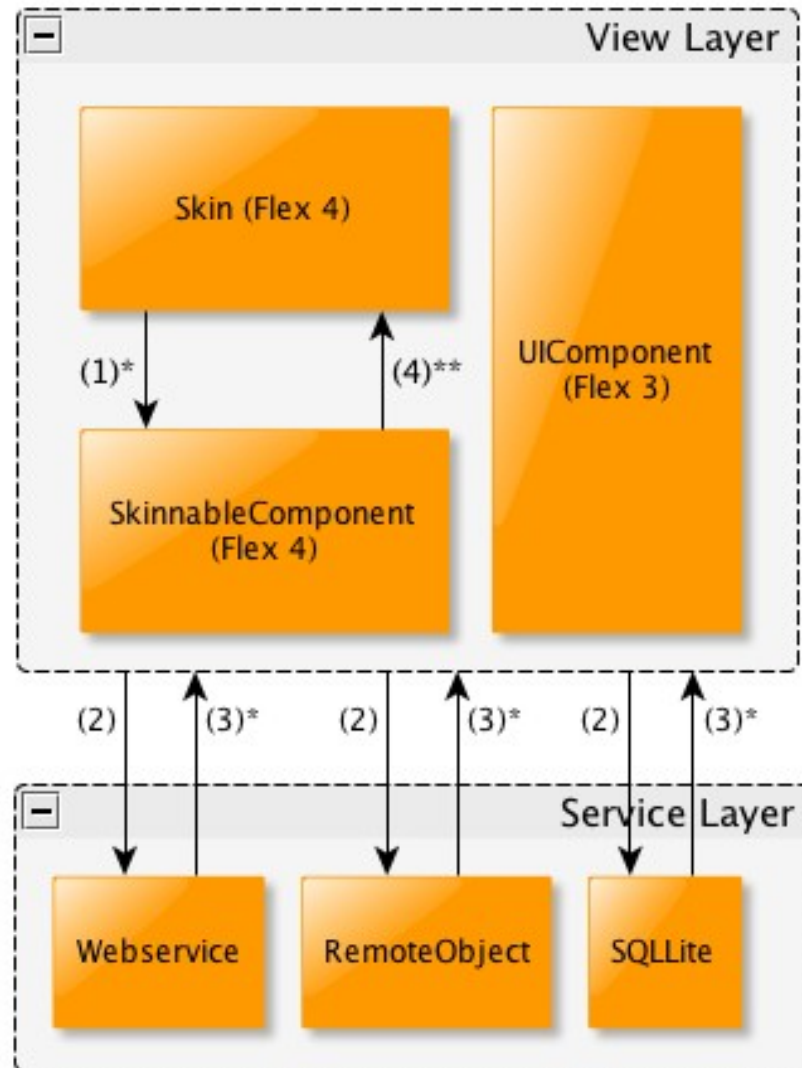
- Coding is work
- Goal-Driven Development ... work, work, work, work, work, work -> HAPPY
- Social-Driven Development – mental hygiene & social hygiene
- An architecture often proves its quality at the end of the project
- Desirable: Less frustrations and high success/motivation while working on bugs and features of your app
- The more features in your app, the more dependencies you have to deal with
- Testability of your code
- Optimal development environment and configuration
- Anarchy vs. coding conventions / design patterns
- Artistic freedom of your software engineers



# Apache Flex Layers



## Flex Layer – Application Flow

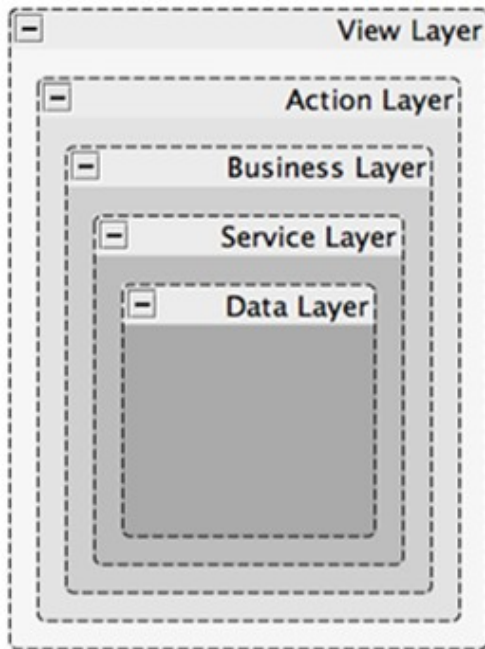


- (1)\* User Interface dispatches Event to SkinnableComponent (Flex 4) class
- (2) UIComponent (Flex 3) / SkinnableComponent (Flex 4) calls Service
- (3)\* Service returns data
- (4)\*\* Databinding updates User Interface

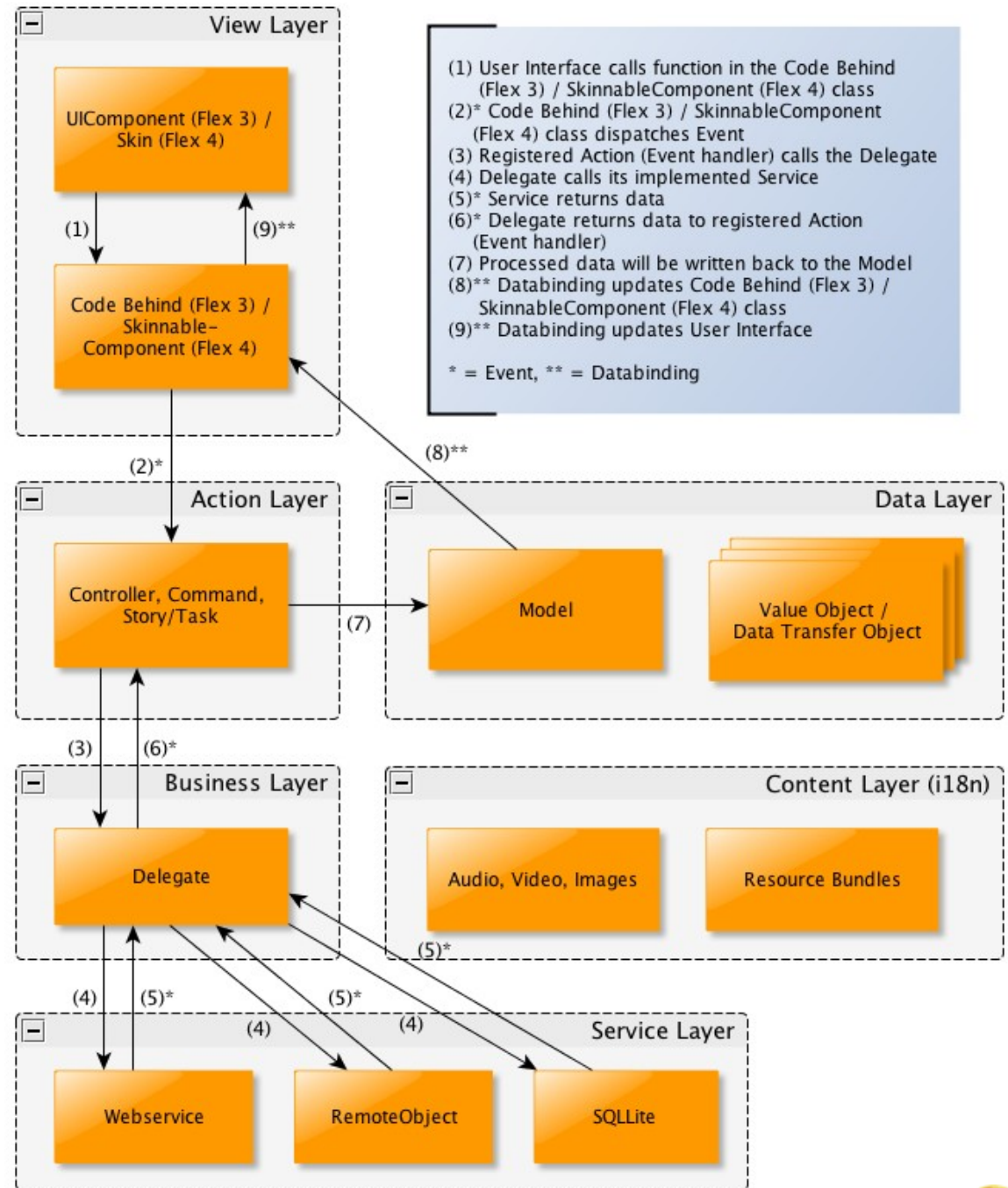
\* = Event, \*\* = Databinding

# Architectural Pattern / Nanoarchitectures

Application Layer Dependency - No Event (NoEvt)



Flex with MVC Layers - Application Flow



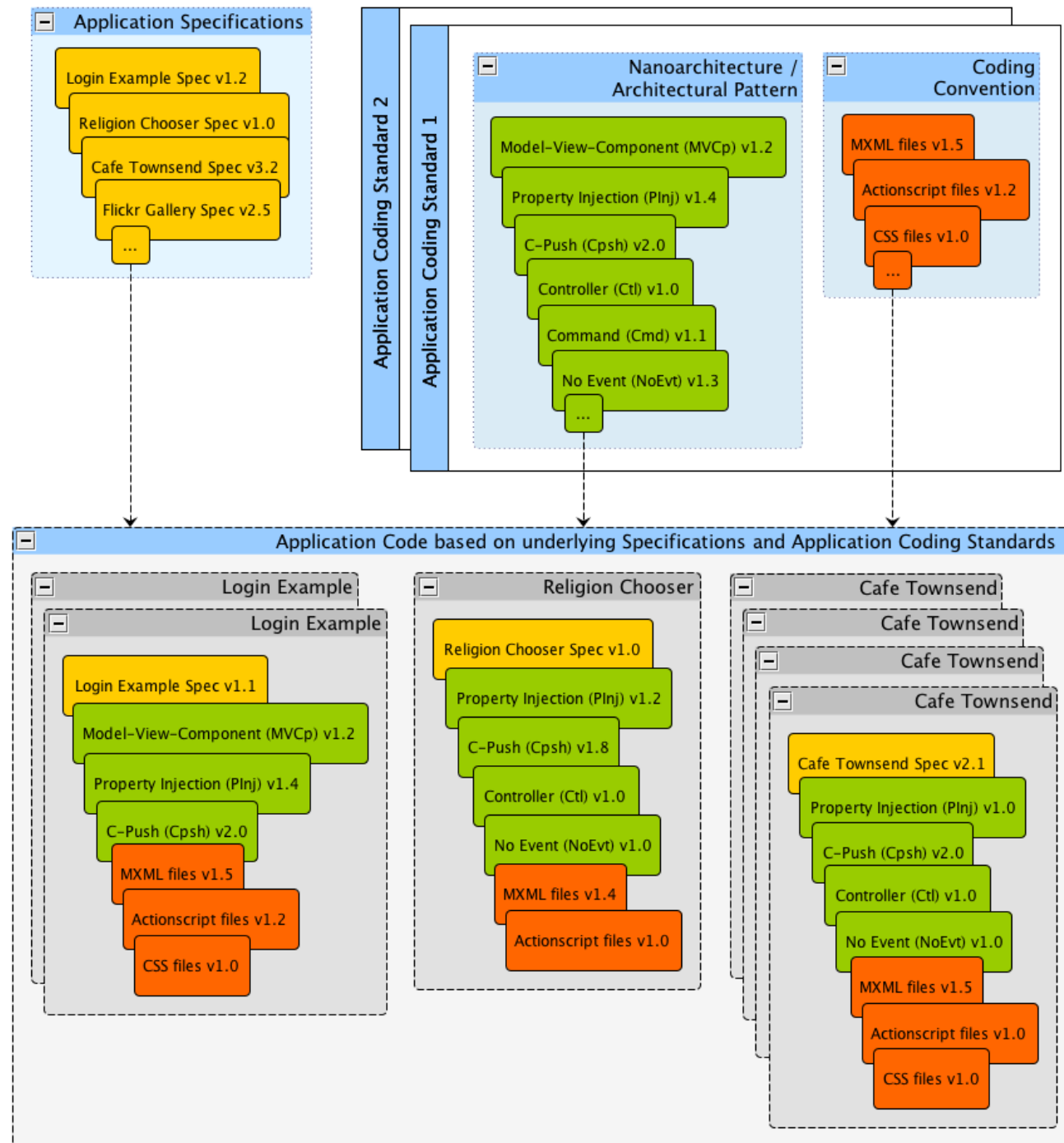


# Application Development Procedure

1. **Application Code** that shows the working application with source code available
2. **Specification Document** which explains the purpose of the application and collects all its requirements
3. **Application Coding Standard** which consists of a collection of Architectural Patterns and underlying Coding Conventions in the code
  - 3.1. **Architectural Patterns** just like "Model-View-Controller" (MVC), "Model-View-Presenter" (MVP), "Presentation Model" (PM) ... etc.
  - 3.2. **Coding Conventions** on the basis of Flex 3 or Flex 4

Read definition page: <http://code.google.com/p/masuland/wiki/ApplicationDevelopmentProcedure>





Thank you!  
Questions?

# References

- Apache Flex Community Revue July 2012,  
<http://www.spoon.as/2012/apache-revue-july-2012/>
- Apache Flex 4.8 Release Candidate,  
<http://people.apache.org/~cframpton/ApacheFlexRC/current/>
- Apache Flex News, <http://incubator.apache.org/flex/news.html>
- Apache Flex Wiki,  
[https://cwiki.apache.org/confluence/display/FLEX/Apache+Flex+\(incubating\)+Wiki](https://cwiki.apache.org/confluence/display/FLEX/Apache+Flex+(incubating)+Wiki)
- Apache Flex Mailing Lists, <http://incubator.apache.org/flex/mailling-lists.html>
- Apache Flex JIRA, <https://issues.apache.org/jira/browse/FLEX>
- Apache Flex Blog, <http://blogs.apache.org/flex/>
- Adobe Flex, <http://sourceforge.net/adobe/flexsdk/wiki/About/>
- Adobe Flex Roadmap, <http://www.adobe.com/devnet/flex/whitepapers/roadmap.html>
- Adobe Flashplayer Roadmap,  
<http://www.adobe.com/devnet/flashplatform/whitepapers/roadmap.html>
- Masuland „Login Example“, <http://code.google.com/p/masuland/wiki/LoginExample>