MVC Revival on the Web

Janko Mivšek

janko.mivsek@eranova.si

@mivsek @aidaweb

AIDAweb
Smalltalk Web Application Server

Motivation

- 30 years of Smalltalk, 30 years of MVC
  - 34 years exactly, since 1979
- Hot in JavaScript MVC frameworks
- For Single-Page and Realtime web apps
- We Smalltalkers should respect and revive our pearls better
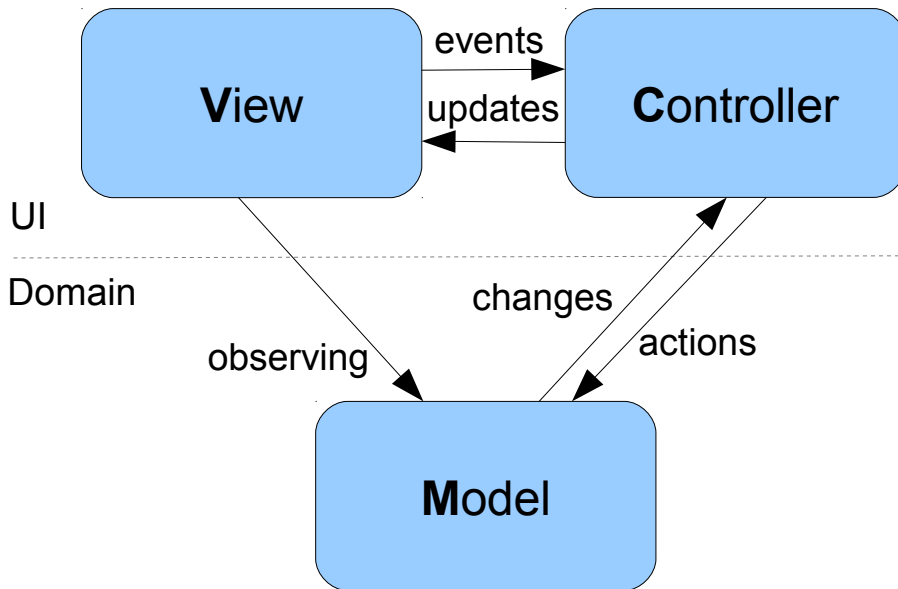
AIDAweb
Smalltalk Web Application Server

Contents

- MVC explained
- History
- Usage in current JavaScript frameworks
- Usage in Smalltalk
- Current and future MVC in Aida/Web

AIDAweb
Smalltalk Web Application Server

MVC Explained

Architectural design pattern
- Model for domain specific data and logic
- View for presentation to the user
- Controller for interactions with the user
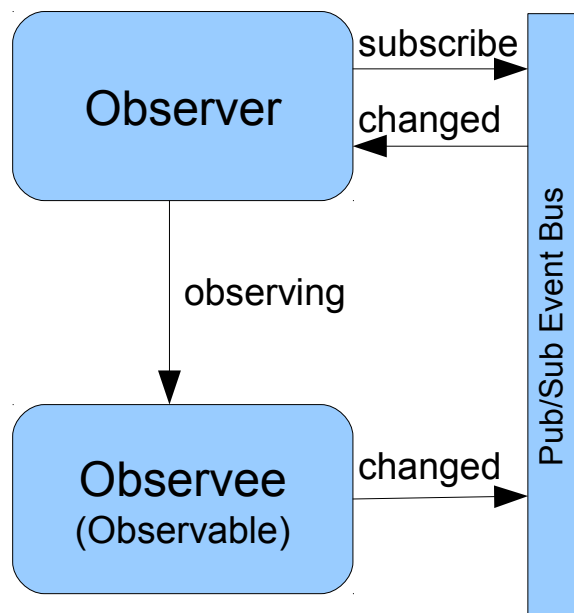 and updating domain model

Main benefit:
- separation of concerns

add: self documentTitleElement; add: self documentActionsElement.
bucketElement := self bucketsElement. "detailed info, references..."
add: self infoLineElement. "brief info about document"
add: bucketElement. "infoLine needs buckets before"

MVC based on Observer pattern

Observer ⟶ subscribe ⟶ Pub/Sub Event Bus

Pub/Sub Event Bus ⟶ changed ⟶ Observer

Observer ⟶ observing ⟶ Observee (Observable)

Observee (Observable) ⟶ changed ⟶ Pub/Sub Event Bus

- Observer looks at observee
- Observee is not aware of that

Dependency mechanism:
 - Observer is dependent on Observee state
 - Observee must report state changes to Observer
 - Pub/Sub Event Bus decouples Observee from
   Observer to preserve its unawarnes of observation

Main benefit:
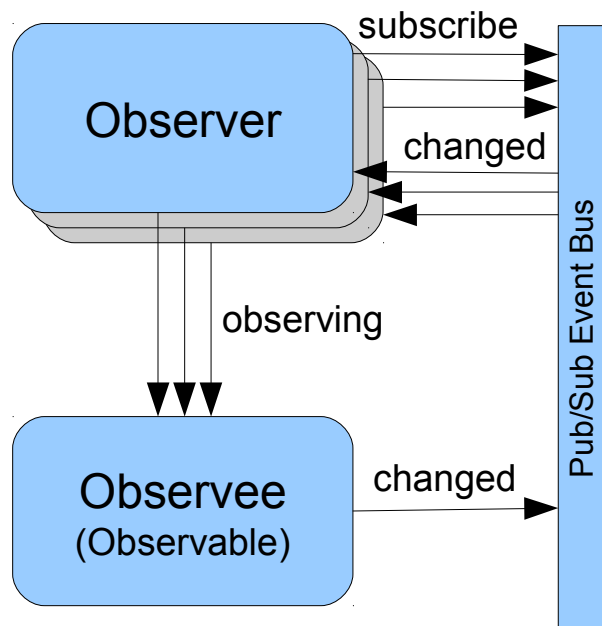- separation of concerns

Multiple observers

- Multiple observers of the same Observee
- In MVC an Observee is View and Observee is domain Model, therefore:

  - many views of the same model
  - many apps
  - many users
  - mix of all three cases

AIDAweb
Smalltalk Web Application Server

Example: Counter demo in Aida/Web

▷ Multiuser realtime web counter example

▷ http://demo.aidaweb.si

    – click **Realtime** on the left

    – click Decrease or Increase to change counter

    – counter is changed on all other's browsers

History of MVC (1)

Invented by Trygve Reenskaug when he worked in 1978/79 with Alan Kay's group on Xerox Parc on Smalltalk and Dynabook

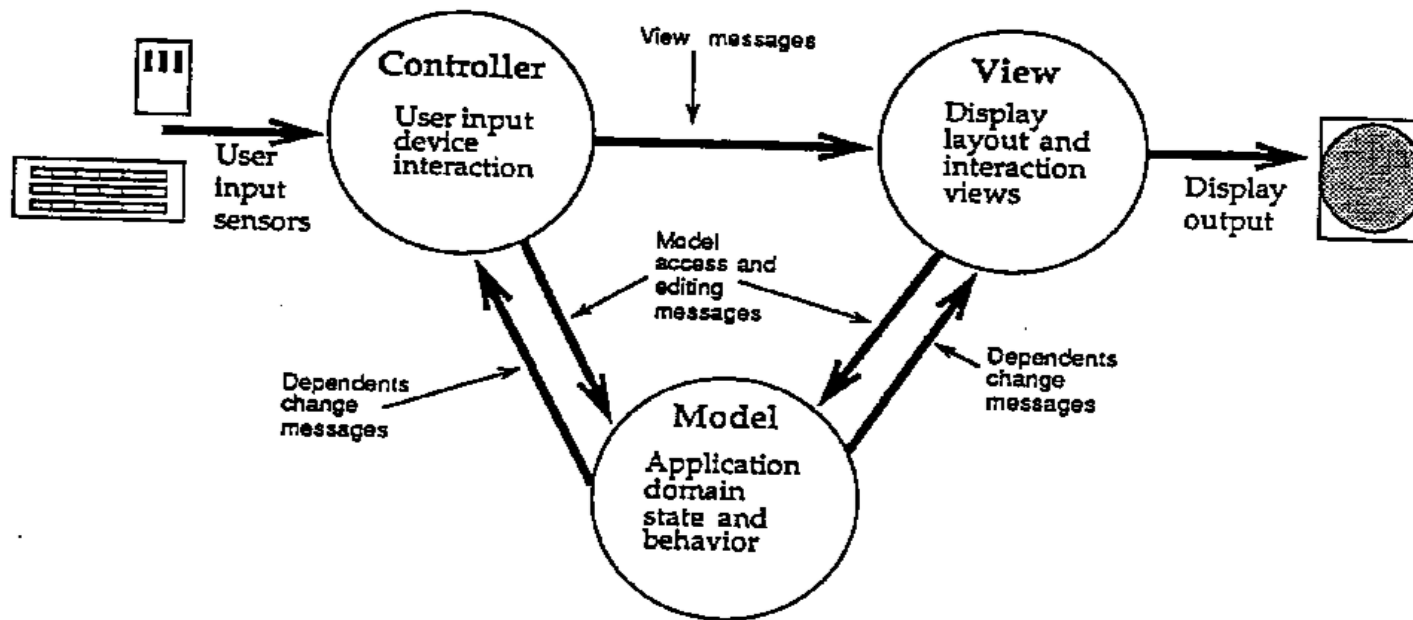– Final term Model-View-Controller coined 10. December 1979.
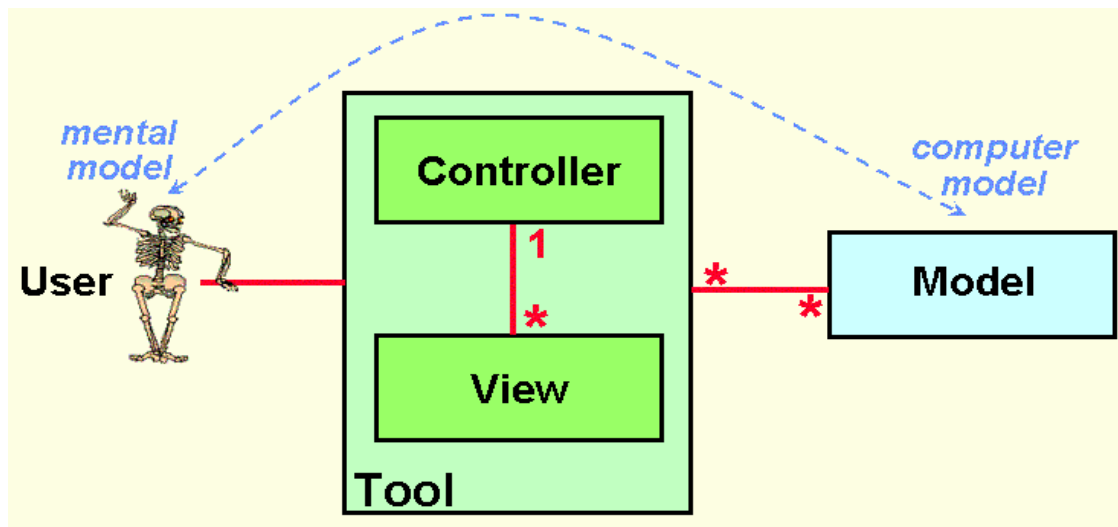
Reenskaug's diagram of MVC basic idea from 2006

History of MVC (2)

MVC State and Message Sending
A Description of the Mode-View-Controller User
Interface Paradigm in the Smalltalk-80 System
Glen E. Krasner and Stephen T. Pope
Parcplace, 1988

History of MVC (3)

> **80's**
  - 1979: invented by Trygve Reenskaug
  - Smalltalk-80, VisualWorks by Parcplace

> **90's**
  - NeXTSTEP, OpenStep, Cocoa
  - 1996: Aida/Web
  - Squeak MVC

> **00's**
  - JSP, Apache Struts, Spring,
  - ASP.NET
  - Ruby on Rails

> **10's**
  - JavaScript MVC frameworks: Backbone, Ember, Angular, ...

add: self documentTitleElement; add: self documentActionsElement.
bucketElement := self bucketsElement. "detailed info, references..."
add: self infoLineElement. "brief info about document"
add: bucketElement. "infoLine needs buckets before"

AIDAweb
Smalltalk Web Application Server

MVC in JavaScript

KnockoutJS

Backbone.js

Ember.js

- First JS MVC framework

AngularJS

- Supported by Google
- Influences new HTML5 standarts
- Dart
- Many ex-Smalltalkers there
- **W3C Web Components**

☑ TodoMVC

- JS MVC »playground«
- simple web app to test
  and compare frameworks
- http://todomvc.com

add: self documentTitleElement; add: self documentActionsElement.
bucketElement := self bucketsElement. "detailed info, references..."
add: self infoLineElement. "brief info about document"
add: bucketElement. "infoLine needs buckets before"

AIDAweb
Smalltalk Web Application Server

MVC in Smalltalk

> Nowadays in use:

> GUI Frameworks

- VisualWorks

- Dolphin (MVP)

> Web Frameworks

- Aida/Web

add: self documentTitleElement; add: self documentActionsElement.
bucketElement := self bucketsElement. "detailed info, references..."
add: self infoLineElement. "brief info about document"
add: bucketElement. "infoLine needs buckets before"

AIDAweb
Smalltalk Web Application Server

Future MVC in Aida/Web
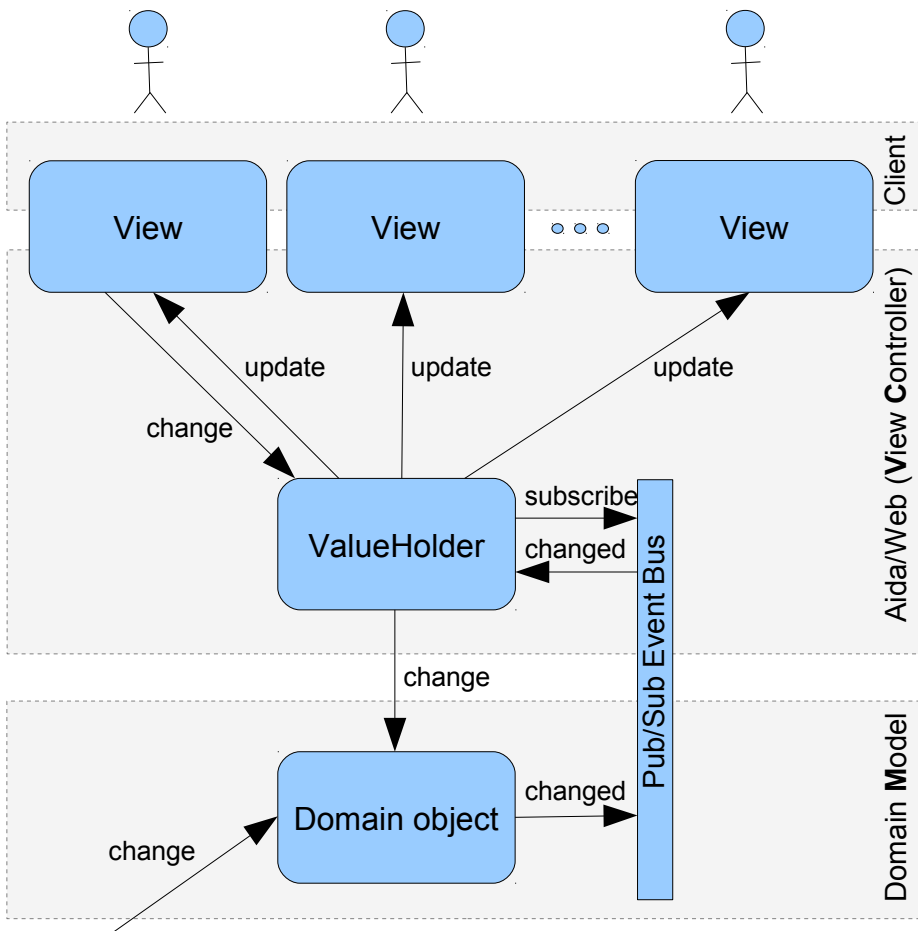
- Extend dependency mechanism up to the web clients
- auto update when domain object changes
- realtime
- decoupled
- no need for extra code

Change detection

counter: anInteger
   counter := anInteger.
   ~~self changed: #counter.~~

- Automatic change detection is needed

- Possible solution:
  - Immutability flag as in VisualWorks
  - last remaining bit in object header used
  - flags an object as immutable (read-only)
  - any write attempt raise and exception

self beImmutable.

...

counter: anInteger
   counter := anInteger.

exception

continue

Exception handler

self beMutable
counter:= anInteger
self beImmutable
self changed: #counter

AIDAweb
Smalltalk Web Application Server

Conclusion

- MVC is well alive
- Very usefull for nowadays Single Page Realtime web apps
- Smalltalk community encouraged to find new ways of using MVC in our apps

AIDAweb
Smalltalk Web Application Server

Thanks for your attention!

www.aidaweb.si

@aidaweb

AIDAweb
Smalltalk Web Application Server