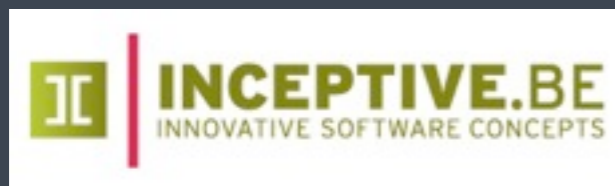


NeXTPLAN*

Enterprise software that rocks!
with Seaside & Smalltalk

Johan Brichau - johan@inceptive.be



NeXTPLAN is...

- A **collaborative** resource planning tool for event organizers of all sizes.
- **Developed for and with** major player of Belgian cultural scene (Vooruit, Gent).
- **Features:** Event planning, resource planning & management, contact management, retro planning, billing, reporting.

NeXTPLAN*



NeXTPLAN Calendar 0.23

http://localhost:8080/NextPlan?_s=9E-id7m25mSR1UZf&_k=eV1QZb31urmy62zP

ucf e-services my web fun news personal research development mac Interesting Links technical Applications Other Bookmarks

Theaterzaal Concertzaal Balzaal Domzaal Minard Majolica Emletrulmte

Mo 30-8-2010 Othello 1 15:00 Shakespeare Surprise Concert 20:00

Tu 31-8-2010 The Room 1 20:00 The Room

We 1-9-2010 Films The Film Festival Othello2 18:00 Shakespeare 1 more events Films The Film Festival

Th 2-9-2010 The Winning Film The Film Festival

Fr 3-9-2010 Ongedecideerd 13:00

Sa 4-9-2010

Su 5-9-2010

Mo 6-9-2010 Othello 1 30 August 2010 - Theaterzaal 15:00 start 23:00 stop 30 August 2010 start description 00:00 Add toggle stop time

Tu 7-9-2010 Films 1 September 2010 Status: Planned Domzaal misc: The Film Festival 5 activities

We 8-9-2010

Th 9-9-2010

Othello2 1 September 2010 Status: Planned Balzaal project: Shakespeare 6 activities Shakespeare Othello Othello 1 Othello2 Othello3 Macbeth Macbeth 1 Macbeth 2 Hamlet Hamlet 1

EVENTS RESOURC... CONTACTS TASKS HELP SEARCH PLANNER TO DO ALERTS UPDATES



NeXTPLAN*

The screenshot displays the NeXTPLAN web application interface. The browser window title is "NextPlan Structure Editor 0.1 X". The address bar shows the URL: <http://localhost:8080/NextPlan?s=dhQj3-2rkh8SwMft&k=6ApFuRXAJ-ppn4sx>. The interface features a calendar view on the left with dates from Monday, August 30, 2010, to Saturday, September 4, 2010. Each day has associated event boxes: "Surprise Concert" (Mon), "Othello 1" (Tue), "Macbeth 1" and "The Room 1" (Tue), "Othello2" and "The Room 2" (Wed), "Hamlet 1" (Thu), "Ongedeeldeerd" and "Othello3" (Fri), and "Films" (Sat). A central resource allocation diagram shows a tree structure starting from "30-8-2010", branching into "Theaterzaal", "Theatertechniek", and "Facility". "Theatertechniek" is linked to "Frank", "Grote PA*", "Kleine PA", "Kleine lichtset", and "Wireless MIC". "Facility" is linked to "Midas 88 Brons", "Soundcraft SM12", and "Speakerset". A bottom navigation bar includes icons for EVENTS, RESOURC..., CONTACTS, TASKS, HELP, SEARCH, PLANNER, TO DO, ALERTS, and UPDATES.



In this talk....

- **Smalltalk** and **Seaside** are an impressive platform to create highly dynamic web applications that blur the traditional distinction with desktop applications
- Some details:
 - Local AJAX updates
 - OODBs
 - SVG + AJAX in Seaside
 - Testing, porting to Gemstone

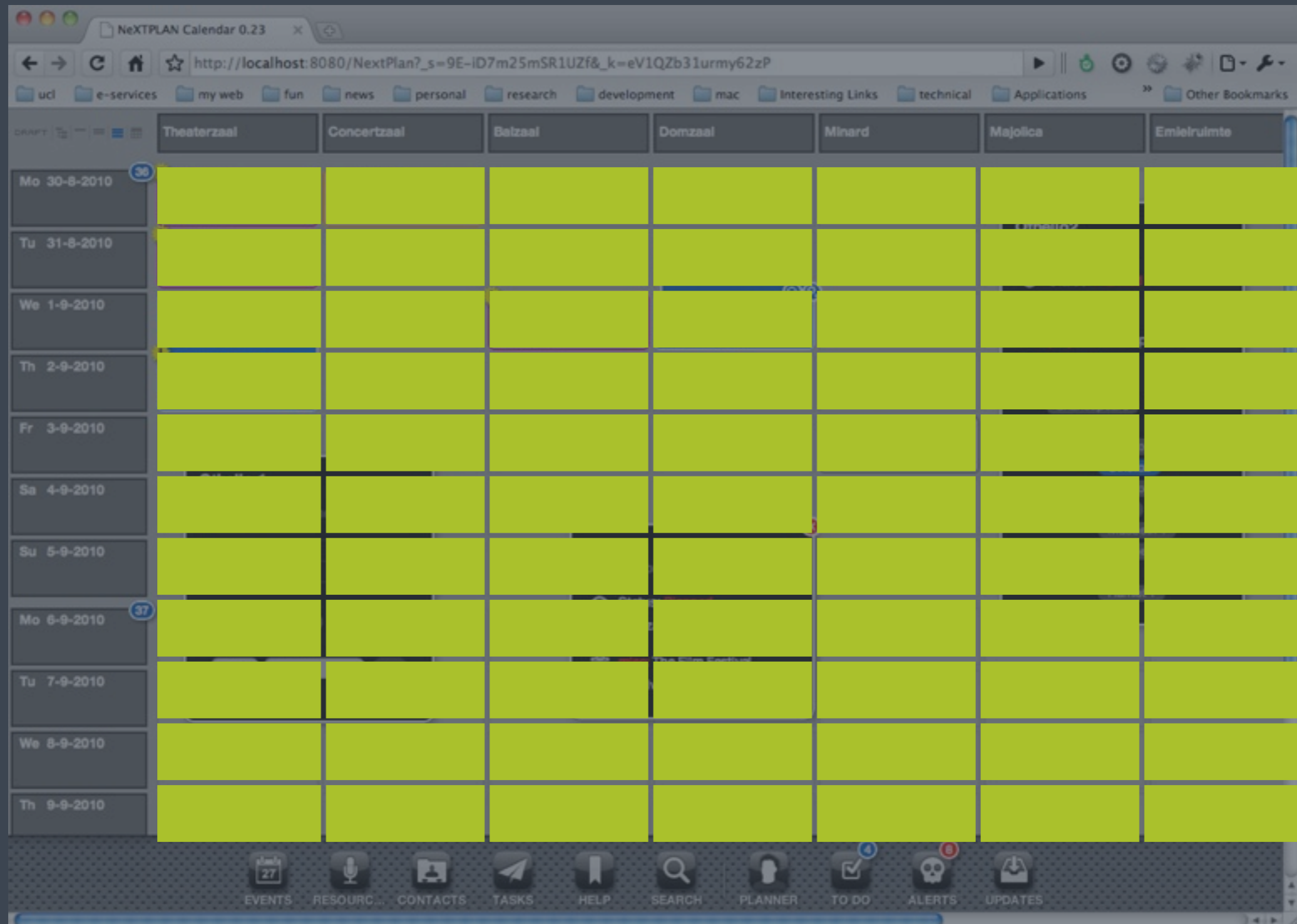
Demo 1

- [demo to show component interactions in calendar: grouped events & bubbles, changing names, decoration]

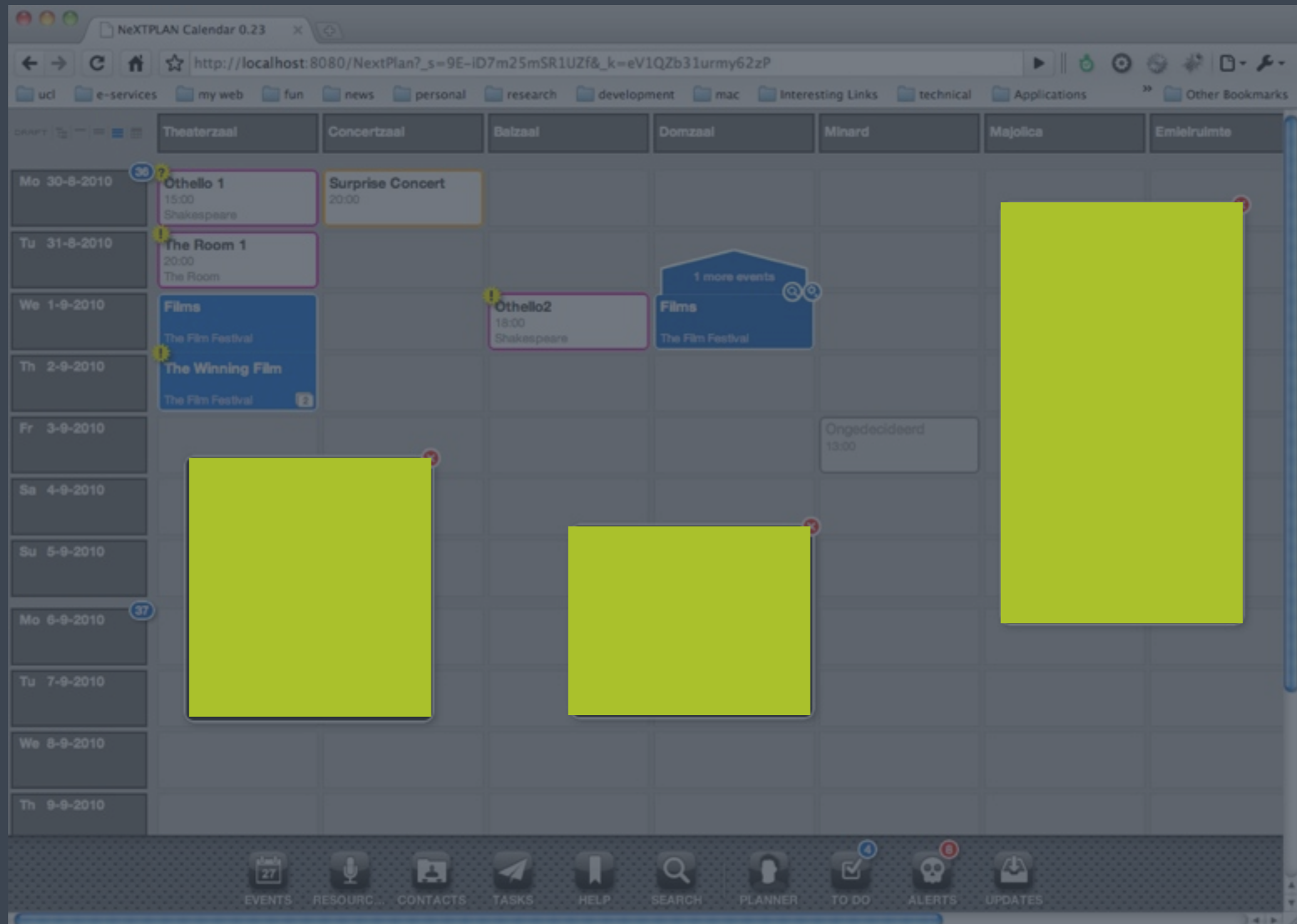
Component-based

The screenshot displays the NeXTPLAN Calendar 0.23 application. The interface is component-based, featuring a grid of event cards and a sidebar for detailed views. The main grid shows events for the week of August 30 to September 9, 2010, across various venues like Theaterzaal, Concertzaal, Balzaal, and Domzaal. Events include 'Othello 1', 'Surprise Concert', 'The Room 1', 'Othello2', 'Films', and 'The Winning Film'. A sidebar on the right provides a detailed view of 'Othello2' on September 1st, showing its status as 'Planned', location as 'Balzaal', project as 'Shakespeare', and a list of 6 activities. A bottom toolbar contains icons for EVENTS, RESOURCE, CONTACTS, TASKS, HELP, SEARCH, PLANNER, TO DO, ALERTS, and UPDATES. The application is running in a browser window at localhost:8080.

Component-based



Component-based



Ajax Updates Architecture

- Every operation eventually invokes the update rendering
- Update rendering
 - Component re-rendering
 - Ajax-jQuery replace operation

Global Ajax Update

The screenshot displays the NeXTPLAN Calendar 0.23 web application. The browser window shows the URL `http://localhost:8080/NextPlan?s=9E-ID7m25mSR1UZf&k=eV1QZb31urmy62zP`. The application interface includes a top navigation bar with tabs for different venues: Theaterzaal, Concertzaal, Balzaal, Domzaal, Minard, Majolica, and Emielruimte. The main calendar grid shows events for the week of August 30 to September 9, 2010. Key events include 'Othello 1' (15:00, Shakespeare) on August 30, 'Surprise Concert' (20:00) on August 30, 'The Room 1' (20:00, The Room) on August 31, 'Othello2' (18:00, Shakespeare) on September 1, and 'Films' (The Film Festival) on September 1. A detailed pop-up window for 'Othello 1' shows the start time (15:00) and stop time (23:00) for August 30, 2010, in Theaterzaal. Another pop-up for 'Films' shows it is planned for September 1, 2010, in Domzaal, with 5 activities. A third pop-up for 'Othello2' shows it is planned for September 1, 2010, in Balzaal, with 6 activities including Othello 1, Othello 2, Othello 3, Macbeth, Macbeth 1, Macbeth 2, Hamlet, and Hamlet 1. The bottom of the application features a toolbar with icons for EVENTS, RESOURCE, CONTACTS, TASKS, HELP, SEARCH, PLANNER, TO DO, ALERTS, and UPDATES.

Global Ajax Update

The screenshot displays the NeXTPLAN Calendar 0.23 application. The browser address bar shows the URL: <http://localhost:8080/NextPlan?s=9E-ID7m25mSR1UZf&k=eV1QZb31urmy62zP>. The interface features a top navigation bar with tabs for Theaterzaal, Concertzaal, Balzaal, Domzaal, Minard, Majolica, and Emielruimte. A vertical sidebar on the left lists dates from Monday, August 30, 2010, to Thursday, September 9, 2010. The main calendar area is a grid where events are represented as colored blocks. Several event detail panels are open, including 'Othello 1' (30 August 2010, Theaterzaal, 15:00 start, 23:00 stop), 'Othello 2' (1 September 2010, Balzaal, 18:00 start), and 'Films' (1 September 2010, Domzaal). A bottom navigation bar contains icons for EVENTS, RESOURCE, CONTACTS, TASKS, HELP, SEARCH, PLANNER, TO DO, ALERTS, and UPDATES. The entire application window is framed by a decorative border of green squares.

Global Ajax Update

```
scriptToUpdatebodyOn: html
  ^ (html jQuery: #cells)
    load: [:r | self renderCellsOn: r]
```

- + Easy and crisp
- + Universally applicable
- Re-renders too much (too long!)
- Wastes in-place editing advantage
- Most of the time updates are local

Local updates

The screenshot displays the NeXTPLAN Calendar 0.23 interface. The main area is a calendar grid for August and September 2010, organized by room: Theaterzaal, Concertzaal, Balzaal, Domzaal, Minard, Majolica, and Emlerulmte. Events are represented by colored cards with details like start times and titles. Three event cards are expanded into detailed views:

- Othello 1:** 30 August 2010 - Theaterzaal, 15:00 start, 23:00 stop.
- Fिल्ms:** 1 September 2010, Domzaal, misc: The Film Festival, 5 activities.
- Othello2:** 1 September 2010, Balzaal, project: Shakespeare, 6 activities. The activity list includes: Shakespeare, Othello, Othello 1, Othello2, Othello3, Macbeth, Macbeth 1, Macbeth 2, Hamlet, Hamlet 1.

The bottom navigation bar contains icons and labels for: EVENTS, RESOURCE, CONTACTS, TASKS, HELP, SEARCH, PLANNER, TO DO, ALERTS, and UPDATES.

Local updates

The screenshot displays the NeXTPLAN Calendar 0.23 web application. The browser address bar shows the URL: `http://localhost:8080/NextPlan?s=9E-ID7m25mSR1UZf&k=eV1QZb31urmy62zP`. The application interface includes a top navigation bar with tabs for Theaterzaal, Concertzaal, Balzaal, Domzaal, Minard, Majolica, and Emiekrumte. The main calendar grid shows events for the week of August 30 to September 9, 2010. Key events include 'Othello 1' (15:00, Shakespeare) on August 30, 'The Room 1' (20:00, The Room) on August 31, 'Films' (The Film Festival) on September 1, and 'Othello2' (18:00, Shakespeare) on September 1. A yellow highlight is placed on the 'Othello2' event. Three pop-up windows are visible: 'Othello 1' (30 August 2010 - Theaterzaal) with start and stop times, 'Films' (1 September 2010) with status 'Planned' and location 'Domzaal', and 'Othello2' (1 September 2010) with status 'Planned' and location 'Balzaal'. The bottom navigation bar contains icons for EVENTS, RESOURCE, CONTACTS, TASKS, HELP, SEARCH, PLANNER, TO DO, ALERTS, and UPDATES.

Local updates

The screenshot displays the NeXTPLAN Calendar 0.23 interface. The main calendar grid shows events for the week of August 30 to September 9, 2010. Key events include 'Othello 1' (30 Aug, Theaterzaal), 'The Room 1' (31 Aug, Theaterzaal), 'Films' (1 Sep, Domzaal), and 'Othello 2' (1 Sep, Balzaal). A yellow highlight is placed on the 'Films' event card on September 1st, with a red arrow pointing to it from the left. Another red arrow points from the yellow highlight to a detailed event pop-up for 'Films' on September 1st, 2010. This pop-up shows the status as 'Planned', location as 'Domzaal', and project as 'misc: The Film Festival'. A third detailed pop-up for 'Othello 2' is visible on the right side of the screen, showing its status as 'Planned', location as 'Balzaal', and project as 'Shakespeare'. The interface includes a top navigation bar with room names (Theaterzaal, Concertzaal, Balzaal, Domzaal, Minard, Majolica, Emiekrumte) and a bottom toolbar with icons for Events, Resources, Contacts, Tasks, Help, Search, Planner, To Do, Alerts, and Updates.

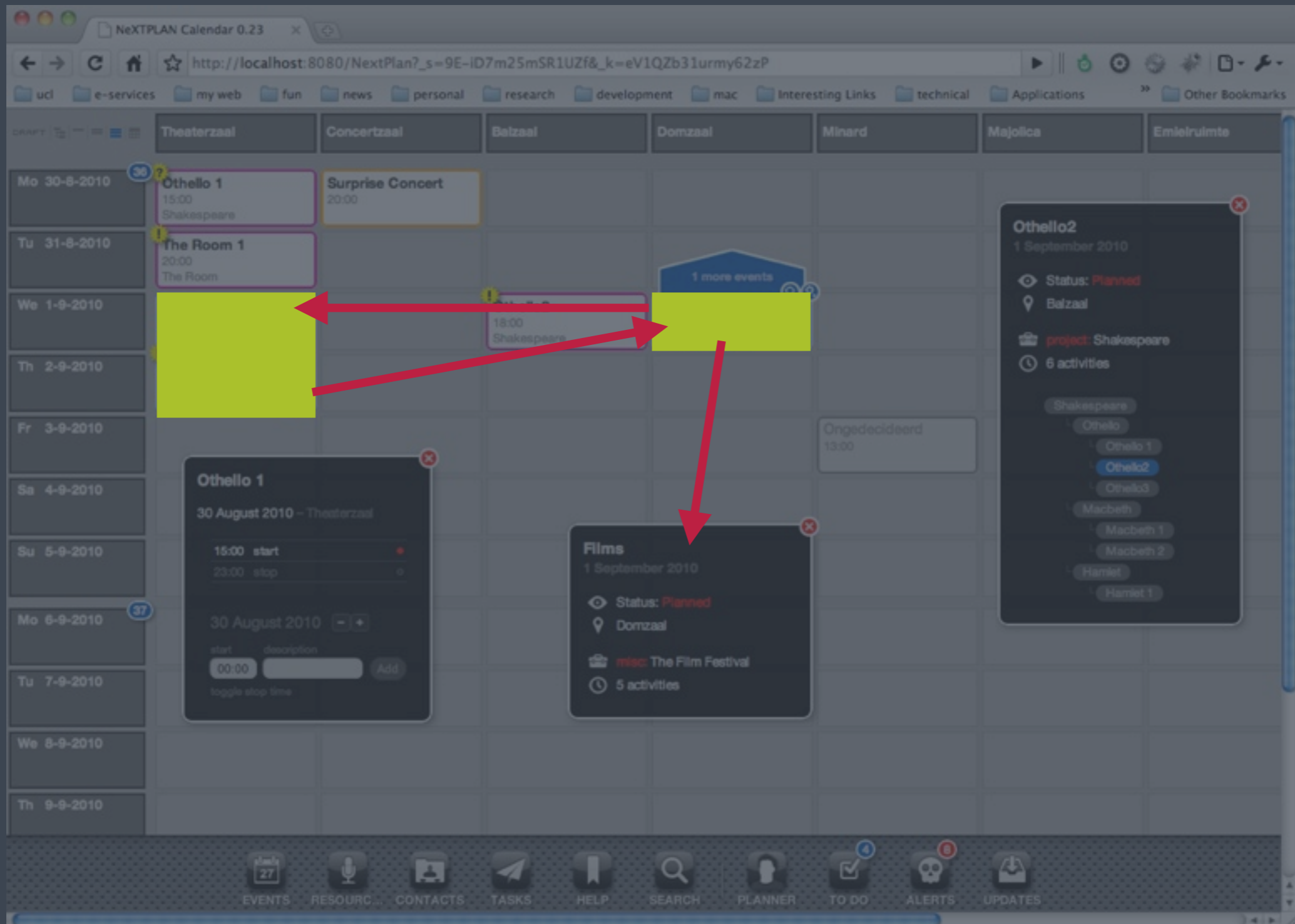
Local updates

The screenshot displays the NeXTPLAN Calendar 0.23 interface. The main calendar grid shows events for August and September 2010. A yellow highlight is placed on the calendar grid for the week of September 1st, with red arrows pointing to it from the event details panels. The event details panels include:

- Othello 1**: 30 August 2010 - Theaterzaal, 15:00 start, 23:00 stop.
- Othello 2**: 1 September 2010, Balzaal, project: Shakespeare, 6 activities.
- Films**: 1 September 2010, Domzaal, misc: The Film Festival, 5 activities.

The interface also shows a navigation bar at the bottom with icons for EVENTS, RESOURCE, CONTACTS, TASKS, HELP, SEARCH, PLANNER, TO DO, ALERTS, and UPDATES.

Local updates



Updates of other users (via database changes)!

Pitfalls

- Instance variable values
- Capture concurrent changes to calendar
- Client side behavior (Javascript)
 - Invoke again or not?
- For example:
 - Drag & drop of cells
 - Context menu on cells
 - Dynamic calendar extension on page scroll

Seaside Design Pattern?

UpdateableComponent

htmlId

renderContentOn:

Complete render

renderInternalOn:

Internal render

needsRefresh

Wants to render?

scriptToUpdateOn:

Script to render

Local updates

```
CalendarComponent>>updateScriptOn: canvas  
^ canvas jQuery ajax script: [:s |  
    self cells do: [:c |  
        c needsRefresh ifTrue:[  
            s << (c updateScriptOn: s)]]]
```

```
Cell>>updateScriptOn: canvas  
^ (canvas jQuery id: self htmlId)  
    load: [:r | self renderInternalOn: r]
```

Local updates

```
CalendarComponent>>updateScriptOn: canvas
^ canvas jQuery ajax script: [:s |
    self children do: [:c |
        c needsRefresh ifTrue:[
            s << (c updateScriptOn: s)]]]
```

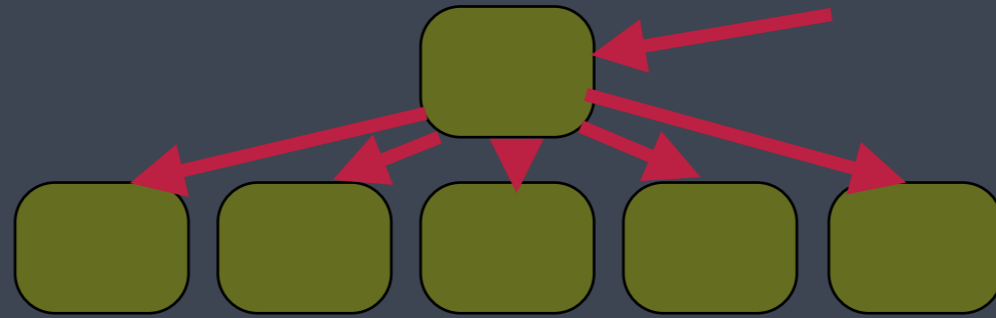
```
Cell>>updateScriptOn: canvas
^(canvas jQuery id: self htmlId)
    load: [:r | self renderInternalOn: r]
```

Local updates

```
CalendarComponent>>updateScriptOn: canvas  
^ canvas jQuery ajax script: [:s |  
    self children do: [:c |  
        c needsRefresh ifTrue:[  
            s << (c updateScriptOn: s)]]]
```

```
Cell>>updateScriptOn: canvas  
^ (canvas jQuery id: self htmlId)  
    replaceWith: [:r | self renderContentOn: r]
```

Local updates



- Own session changes
 - True object-based change propagation
- Concurrent session changes
 - Manual handling

SVG in Seaside

- **SeasideDynamicSVG** package
 - SVG canvas & brushes
 - SVG + jQuery
- SVG + **AJAX**
 - Dynamic load of svg content
 - **Problem:** Painter's algorithm for "z-index"
 - **Solution:** svgLoad

Demo 2

- [show Tree view animations driven by Ajax updates]

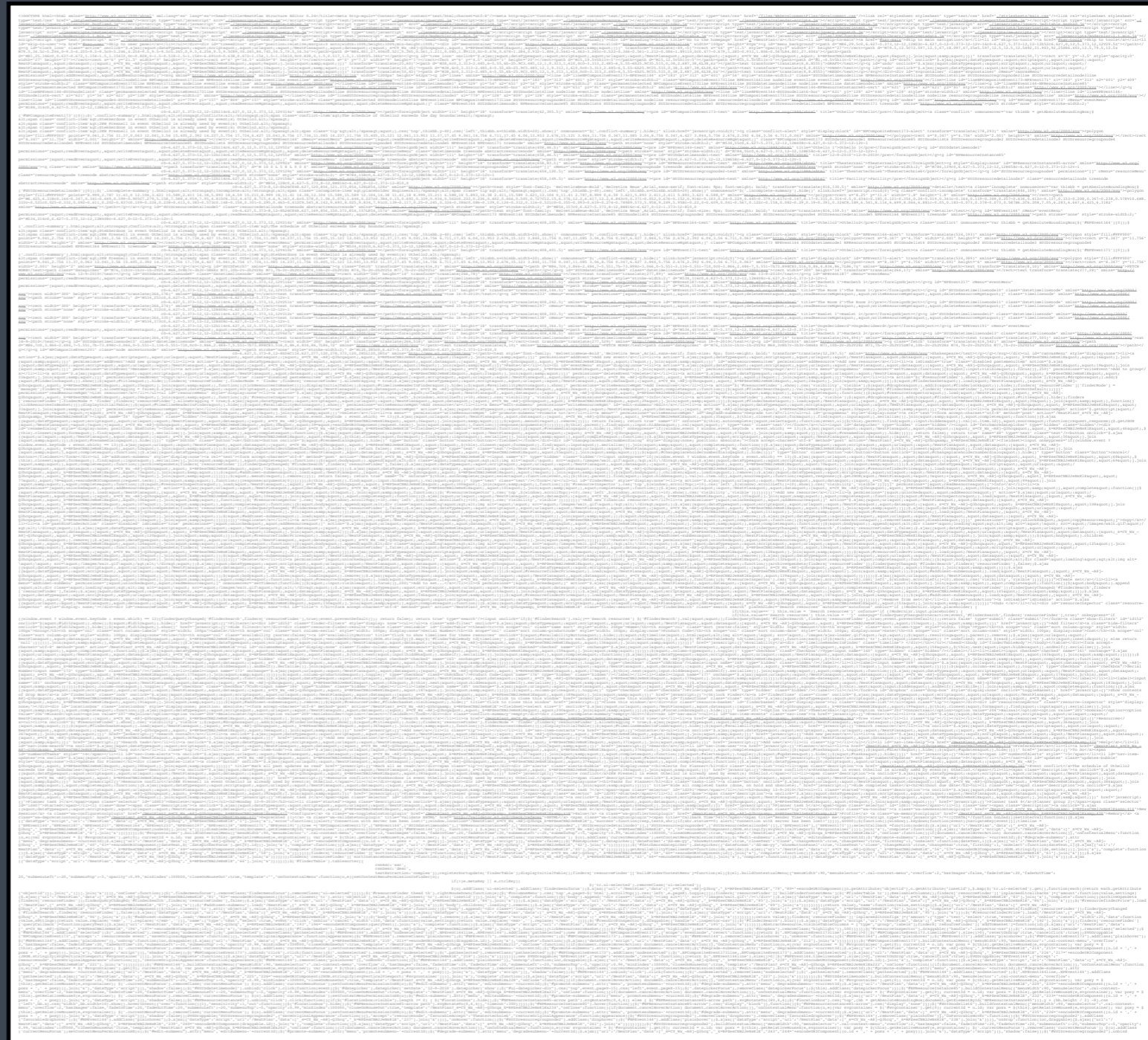
JQuery svgLoad

z-index 1

z-index 2

z-index 3

⋮



JQuery-SVG

- jQuery **load** for SVG content
 - Load **new content** in SVG canvas
 - Add content **at end** of SVG Canvas
- + add content **to existing** SVG element
 - Control "z-index" by adding to SVG groups
 - Not yet present on [squeaksource](#) / jQuery plugin

JQuery svgLoad

z-index 1

<g> ... </g>

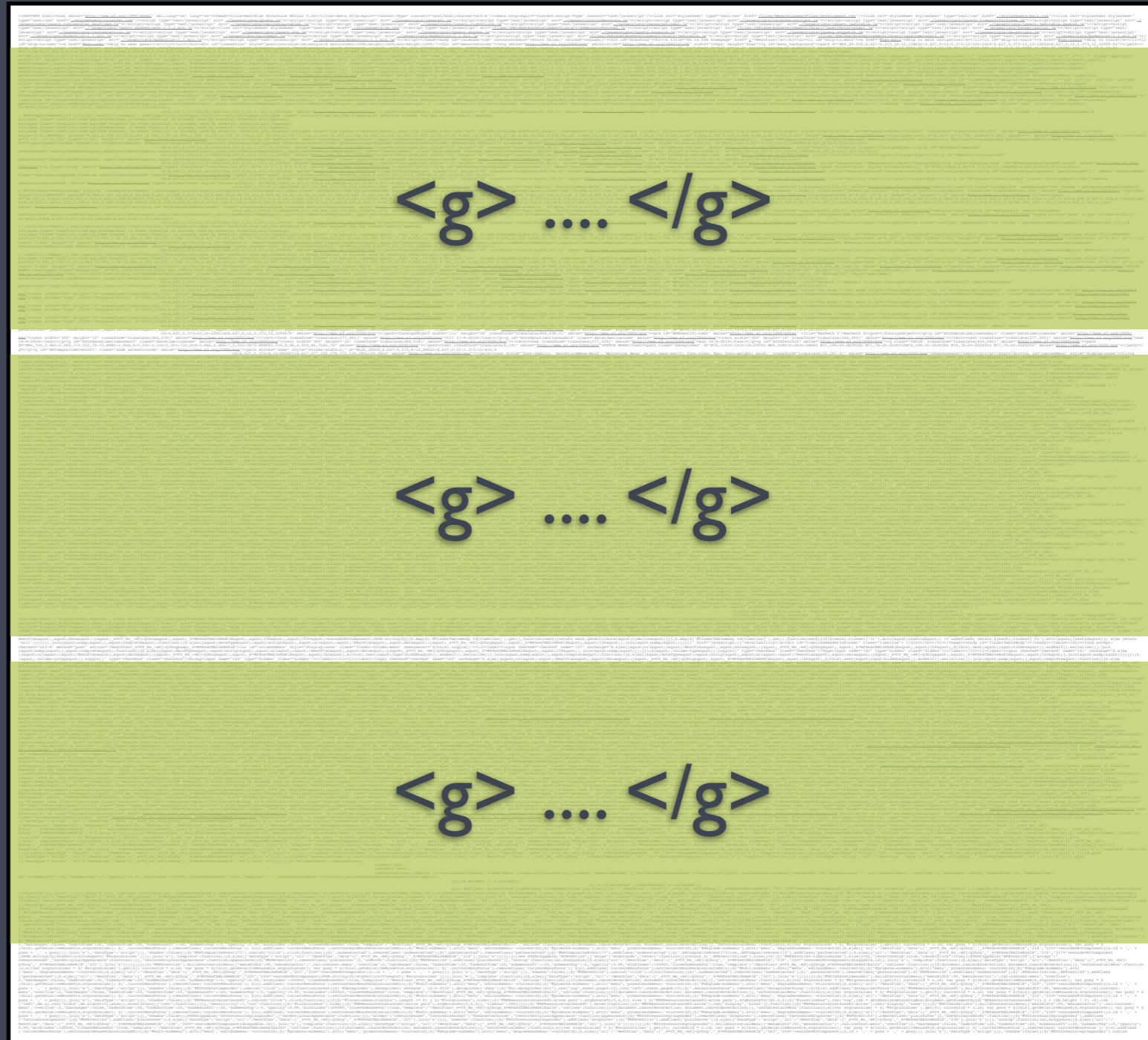
z-index 2

<g> ... </g>

z-index 3

<g> ... </g>

⋮



OODBs

	indexes	nested Tx	write barrier
Magma	✓	✓	✓
GOODS			
Gemstone	✓		✓*

OODBs

	indexes	nested Tx	write barrier
Magma	✓	✓	✓
GOODS			
Gemstone	✓		✓*

Solution: General database access layer

DAO	✓***	✓	✓
-----	------	---	---

GOODS Manual Write Barrier

Smalltalk runtime

Loaded objects



Database



GOODS Manual Write Barrier

Smalltalk runtime

Loaded objects



Write buffer



Database

GOODS Manual Write Barrier

Smalltalk runtime

Loaded objects



Write buffer

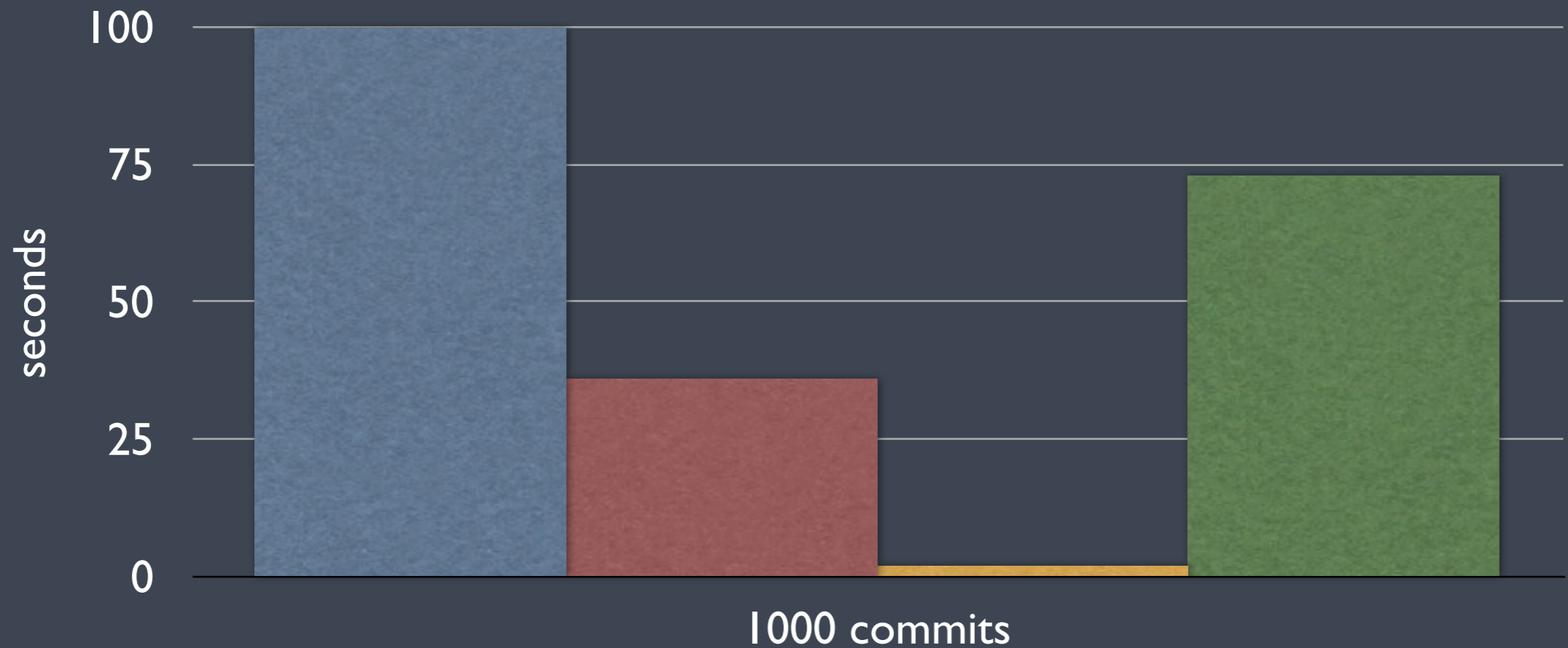


Database

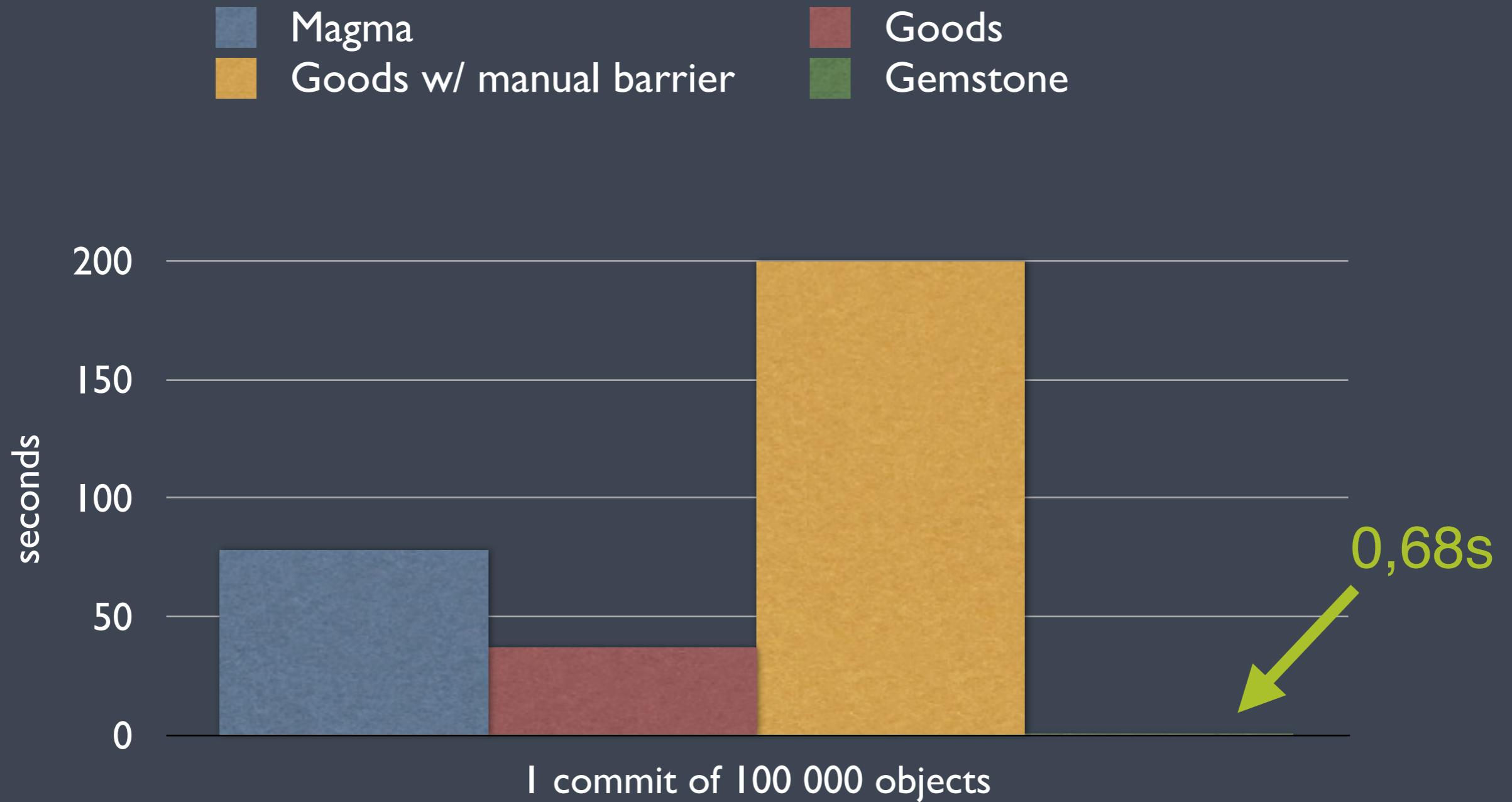
GOODS Manual Write Barrier

- **Track changes** to database objects
 - Notify on change (e.g. Announcements)
 - Use customized Collection classes
- GOODS uses the write-buffer on commit

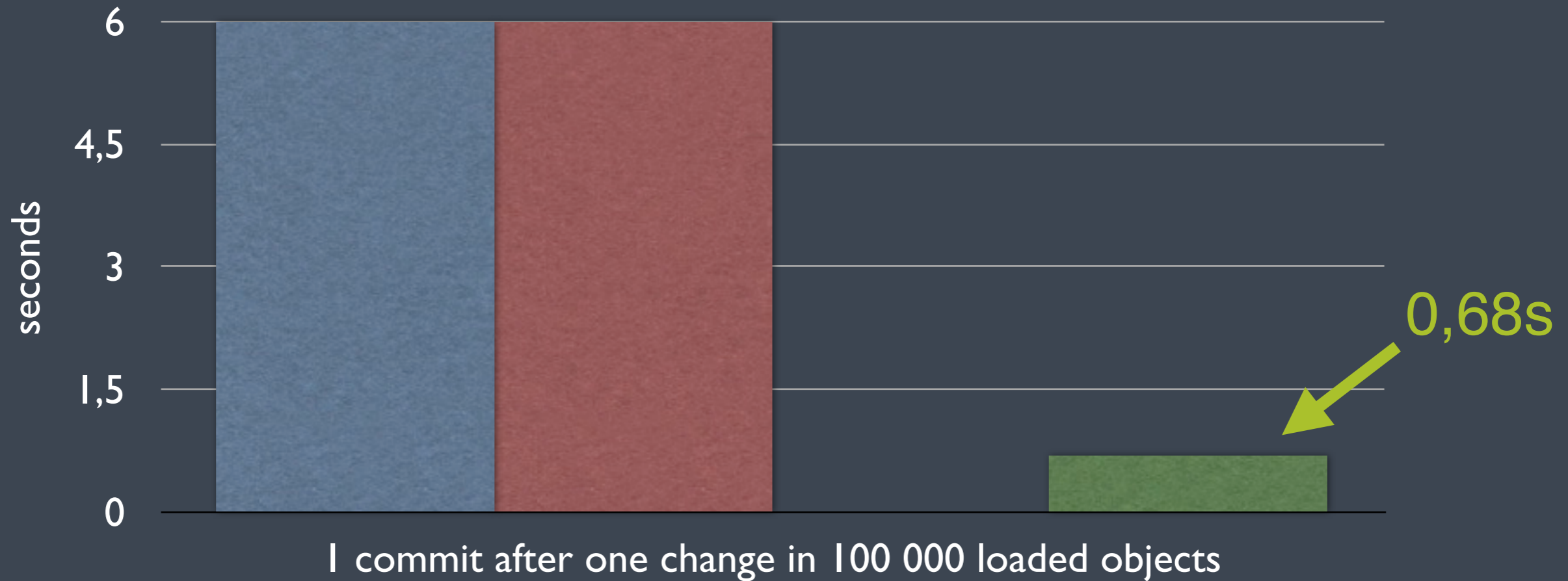
Performance (1)



Performance (2)



Performance (3)



Demo 3

- [demo of resource Finder & inspector to underline database performance issue]

Transactions & Concurrency



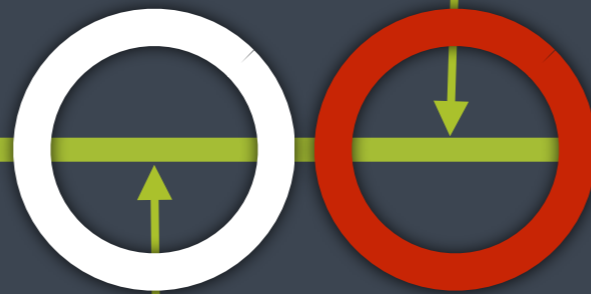
Transactions & Concurrency



User 2 changes objects shared in User 1's view

Transactions & Concurrency

User 1 changes
same objects

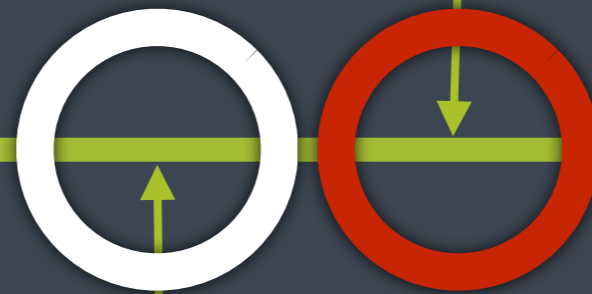


User 2 changes objects
shared in User 1's view

Transactions & Concurrency

User 1 changes
same objects

User 1 is
notified



User 2 changes objects
shared in User 1's view

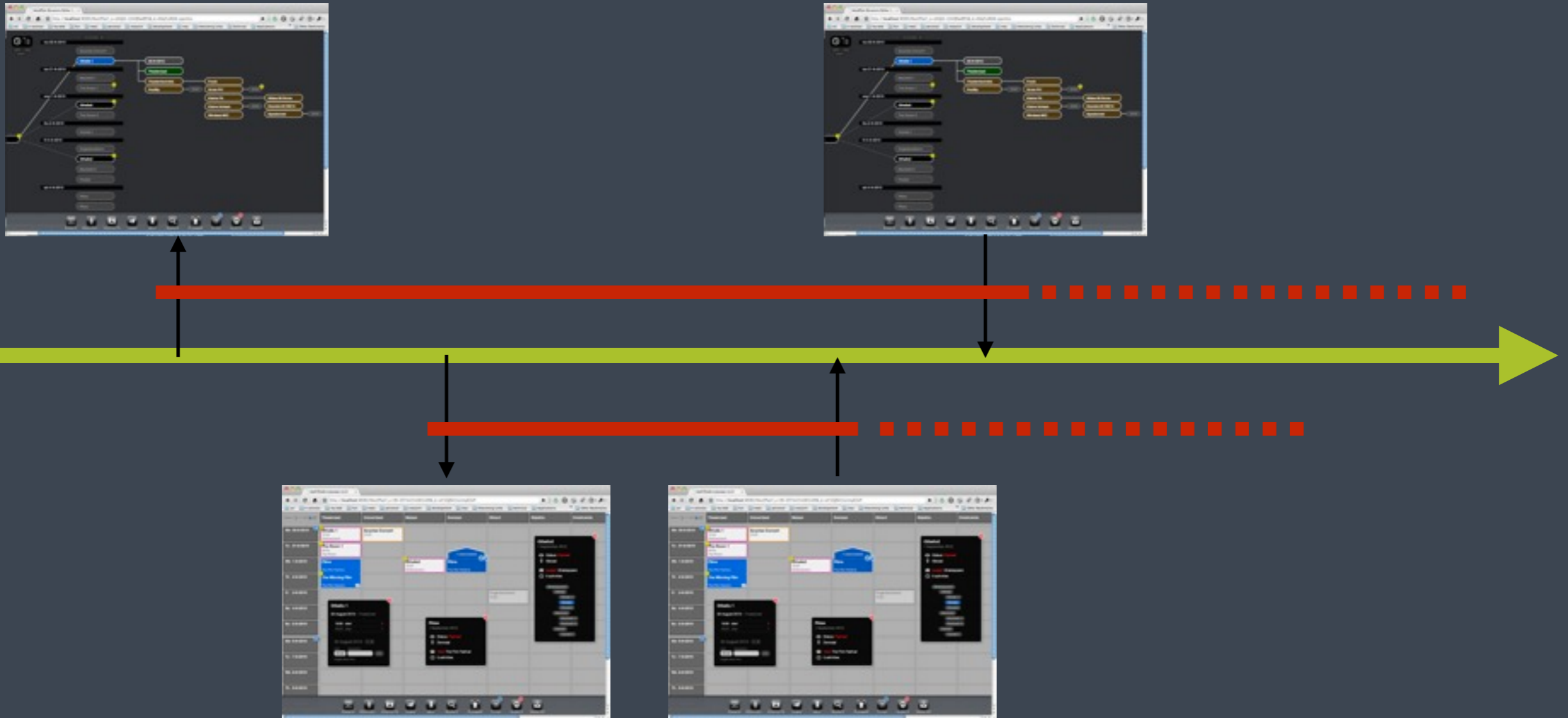
Web-based Concurrency

- Conceptually, this is a transaction conflict!
- But....

Transactions & Concurrency

- Database session scopes
 - Db session per Seaside session [GOODS]
 - Single Db session [Seaside+Gemstone]
- Transaction boundaries
 - Request [Seaside+Gemstone]
 - Conversation

Transactions & Concurrency



Application is not designed
around conversations

GOODS

- Loading/refreshing loaded objects takes a loooooong time
- **Refresh DB objects manually** when user will also see the refresh on the screen (i.e. no refresh in an action callback)
- **Concurrent changes automatically detected** as transaction conflicts w.r.t. concurrent application sessions

Transactions & Concurrency



Seaside + Gemstone

Gemstone + Seaside

- Loaded objects are always up-to-date when request is handled (**refresh is automatic**)
- Concurrent modifications only detected as transaction conflicts w.r.t. concurrently handled requests
- Use **object change tracking** to detect concurrent changes w.r.t. application sessions.

Object Change Tracking

Smalltalk

Loaded objects



Write buffer



Database

“X-db” Tx Framework

```
self session db
  executeAsTransaction: [ ... ]
  forObjects: # (...)
  onSuccess: [ ... ]
  onFailure: [ ... ]
  retryIf: [ ... ]
```

Hybrid components

- JavaScript and Smalltalk
- Client-side interactivity
- Seaside's Javascript generation
- Scalability problems:
 - Passenger / callback registry size
 - Client-side performance

Demo 4

- Show in-place editing of Large amount of data: focus in client-Side functionality

Callbacks

```
assignCallbackScriptOn: html  
  ^ (html jQuery ajax callback: [...])  
    asFunction  
      assignTo: 'someJsVar'
```

```
assignCallbackOn: html  
  ^ (html urlForAction:[...])  
    assignTo: 'someJsVar'
```

Testing

- Unit testing
- SeasideTesting component testing
 - Added callback simulation
- WebTester testing

Callback Simulation

```
testDeleteGroup
| theCell group theNewCell1 theNewCell2 |

theCell := self cellAtRow: 3 forLocationWithName: 'Theaterzaal'.
group := theCell menuFocusEvent.

self executeAsCallback: [
    self component currentMenuFocusEvent: group.
    self component removeSelectedEvents].

theNewCell1 := self cellAtRow: 3 forLocationWithName: 'Theaterzaal'.
theNewCell2 := self cellAtRow: 4 forLocationWithName: 'Theaterzaal'.
self assert: theNewCell1 isEmpty.
self assert: theNewCell2 isEmpty
```


Porting to Gemstone

- OODB transaction model
- Grease + Lint rules
- Pharo <-> Gemstone
 - Strict compiler (no empty statements, shadowing,...)
 - Exception -> ExceptionA
 - Blockclosure -> ExecutableBlock
 - Date/Duration/Timespan protocol
 - Assignments to temporaries (GS <3.0)

NeXTPLAN*

Thank you!

Questions?
Want a demo offline?

<http://www.inceptive.be>

