



Webentwicklung für Cocoa-Entwickler

Andreas Monitzer
2011-02-17



Web 2.0!

Web 2.0–Webseiten sind dynamisch, reagieren sofort auf Input ohne zum Server zu gehen -> JavaScript

JavaScript



Traditionelle Entwickler hören JavaScript.

Aber was ist JavaScript eigentlich?

Wir wollen kein stinkendes JavaScript!

GWT

CoffeeScript

Cappuccino

Warum gibt es so viele Alternativlösungen?



Ohne Kommentar.

JavaScript für Cocoa-Entwickler

Trotzdem finde ich, JavaScript ist eine sehr elegante Programmiersprache.
kurze Einführung, warum das so ist.

NSMutableDictionary

Alle Objekte sind NSMutableDictionarys
jederzeit Änderungen möglich

Zahl String Funktion

Elementare Datentypen:

Zahl sowohl float als auch integer

String auch für Characters (sowohl ‘ als auch “ möglich)

Funktion wie blocks in Objective-C

Variablen sind nicht typisiert!

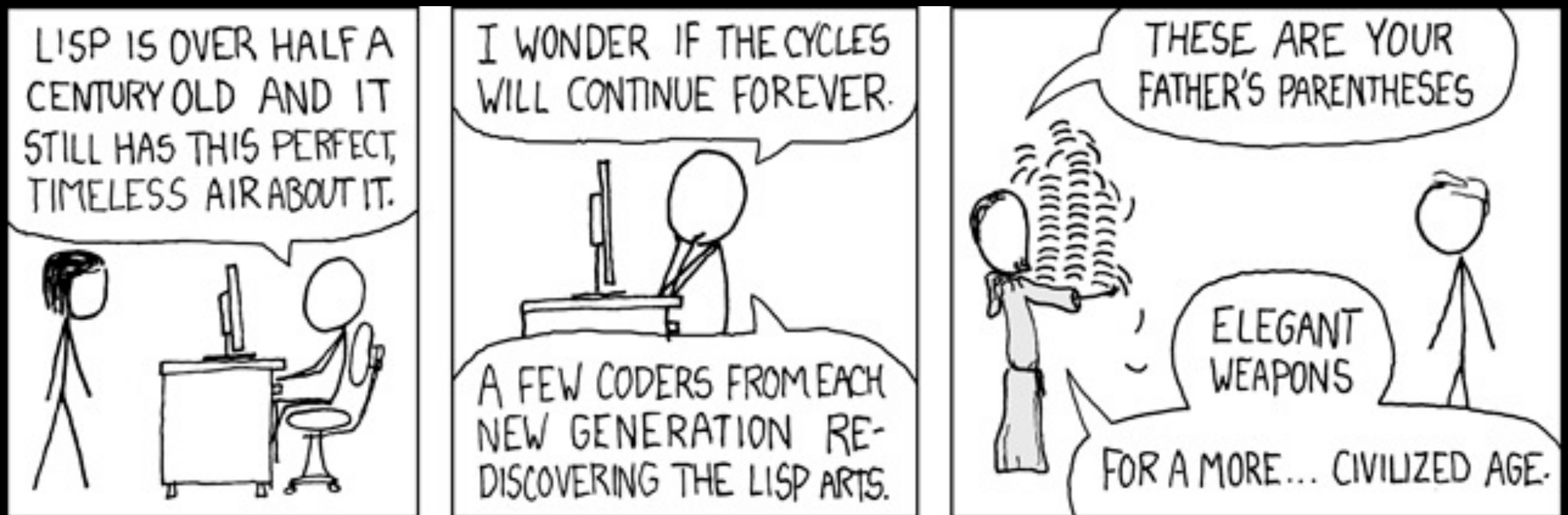
Ursprünge

Ursprünglich hieß JavaScript LiveScript und basiert auf Scheme. Umbenennung wegen Marketing (Netscape/Sun-Kooperation).
eigentlich Scheme-artiger Syntax [klick!] (Scheme ist eine LISP-Variante), Java-artiger Syntax im letzten Moment draufgepickt, hat mit C-Programmiersprachen aber wenig zu tun
erinnert an ein gewisses Comic

Ursprünge

```
;; Tabulation of Hofstadter's male and female sequences
(letrec ((female (lambda(n)
                    (if (= n 0) 1
                        (- n (male (female (- n 1)))))))
         (male (lambda(n)
                  (if (= n 0) 0
                      (- n (female (male (- n 1)))))))
  (display "i male(i) female(i)")(newline)
  (do ((i 0 (+ i 1)))
      ((> i 8) #f)
    (display i) (display "   ")(display (male i))(display "   ")(display (female i))
    (newline)))
```

Ursprünglich hieß JavaScript LiveScript und basiert auf Scheme. Umbenennung wegen Marketing (Netscape/Sun-Kooperation).
eigentlich Scheme-artiger Syntax [klick!] (Scheme ist eine LISP-Variante), Java-artiger Syntax im letzten Moment draufgepickt, hat mit C-Programmiersprachen aber wenig zu tun
erinnert an ein gewisses Comic



```

(defun (female (lambda (n)
  (if (= n 0) 1
      (- n (male (female (- n 1)))))))
  (male (lambda (n)
    (if (= n 0) 0
        (- n (female (male (- n 1)))))))
  (display "i male(i) female(i)")(newline)
  (do ((i 0 (+ i 1)))
      ((> i 8) #f)
      (display i) (display "  ")(display (male i))(display "  ")(display (female i))
      (newline)))

```

Ursprünglich hieß JavaScript LiveScript und basiert auf Scheme. Umbenennung wegen Marketing (Netscape/Sun-Kooperation).
 eigentlich Scheme-artiger Syntax [klick!] (Scheme ist eine LISP-Variante), Java-artiger Syntax im letzten Moment draufgepickt, hat mit C-Programmiersprachen aber wenig zu tun
 erinnert an ein gewisses Comic

Klassen?

Klassen werden mit diesen dictionaries gemacht. Klassen instanzieren heißt, die Klassen zu kopieren (prototyp-basiert)

Methoden?

Methoden sind Einträge im dictionary from Typ Funktion.

Konsequenzen



sehr dynamisches Programmieren, nicht für jeden Programmierer zugänglich
sehrwohl für Objective-C-Entwickler!

“Gotchas”

Probleme beim Umsteigen zu JavaScript

this statt self

“Gotchas”

this statt self

Objektvariablen
immer mit “this.”

“Gotchas”

this statt self

Objektvariablen
immer mit “this.”

“Gotchas”

lokale Variablen mit
“var” deklarieren

this statt self

Objektvariablen
immer mit “this.”

“Gotchas”

lokale Variablen mit
“var” deklarieren

lokale Variablen
gelten immer für
die ganze Funktion

this statt self

Objektvariablen
immer mit “this.”

“Gotchas”

lokale Variablen mit
“var” deklarieren

Arrays sind
Dictionaries mit
Index als Key

lokale Variablen
gelten immer für
die ganze Funktion

this statt self

Objektvariablen
immer mit “this.”

über Objekte
iterieren macht
Probleme

“Gotchas”

lokale Variablen mit
“var” deklarieren

Arrays sind
Dictionaries mit
Index als Key

lokale Variablen
gelten immer für
die ganze Funktion

“=” kopiert keine Objekte
(NSRect!)
this statt self

Objektvariablen
immer mit “this.”

über Objekte
iterieren macht
Probleme

“Gotchas”

lokale Variablen mit
“var” deklarieren

Arrays sind
Dictionaries mit
Index als Key

lokale Variablen
gelten immer für
die ganze Funktion



Web Browser- Integration

soweit zur Programmiersprache, aber wie ist das jetzt mit der Web-Integration?

DOM-Tree

HTML-Seite ist in eine Baum-Datenstruktur gebaut, an der man herummanipulieren kann

Umständlich

nur ist das sehr langweilig, weil die Funktionen dafür sehr lowlevel sind

JavaScript-Libraries?

Es gibt tonnenweise Libraries für JavaScript



jQuery kristallisiert sich als der eindeutige Gewinner heraus
[klick] Beispiel, was damit möglich ist



```
$('.cocoaheads').css('display','none');
```

jQuery kristallisiert sich als der eindeutige Gewinner heraus
[klick] Beispiel, was damit möglich ist

Web 2.0 nur mit jQuery?

Wie ist das, wenn man Web 2.0 nur mit jQuery programmiert?
[klick] in diesem Buch wirds das probiert

Join the discussion @ p2p.wrox.com



Wrox **Programmer to Programmer™**



Professional **XMPP**

Programming with JavaScript® and jQuery

Jack Moffitt

Wie ist das, wenn man Web 2.0 nur mit jQuery programmiert?
[klick] in diesem Buch wirds das probiert

```

on_roster_changed: function (iq) {
    $(iq).find('item').each(function () {
        var sub = $(this).attr('subscription');
        var jid = $(this).attr('jid');
        var name = $(this).attr('name') || jid;
        var jid_id = Gab.jid_to_id(jid);

        if (sub === 'remove') {
            // contact is being removed
            $('# + jid_id).remove();
        } else {
            // contact is being added or modified
            var contact_html = "<li id='" + jid_id + "'" +
                "<div class='" +
                ($('# + jid_id).attr('class') || "roster-contact offline") +
                "'" + "<div class='roster-name'" + name +
                "</div><div class='roster-jid'" + jid + "</div></div></li>";

            if ($('# + jid_id').length > 0) {
                $('# + jid_id').replaceWith(contact_html);
            } else {
                Gab.insert_contact(contact_html);
            }
        }
    });
    return true;
}

```

Funktion grob erklären!

Manipuliert direkt im HTML herum. Zustand des Contacts wird im HTML abgespeichert (und sonst nirgends).

MVC?

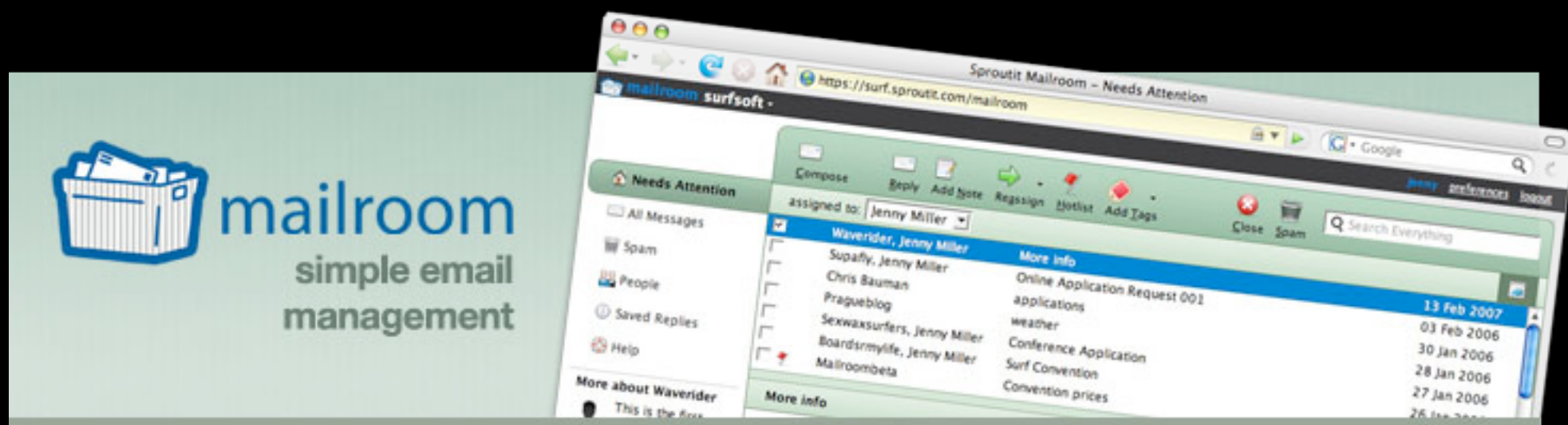


Noch nie davon gehört
Das Cocoa-Entwicklerherz blutet

Neue Library muss her!

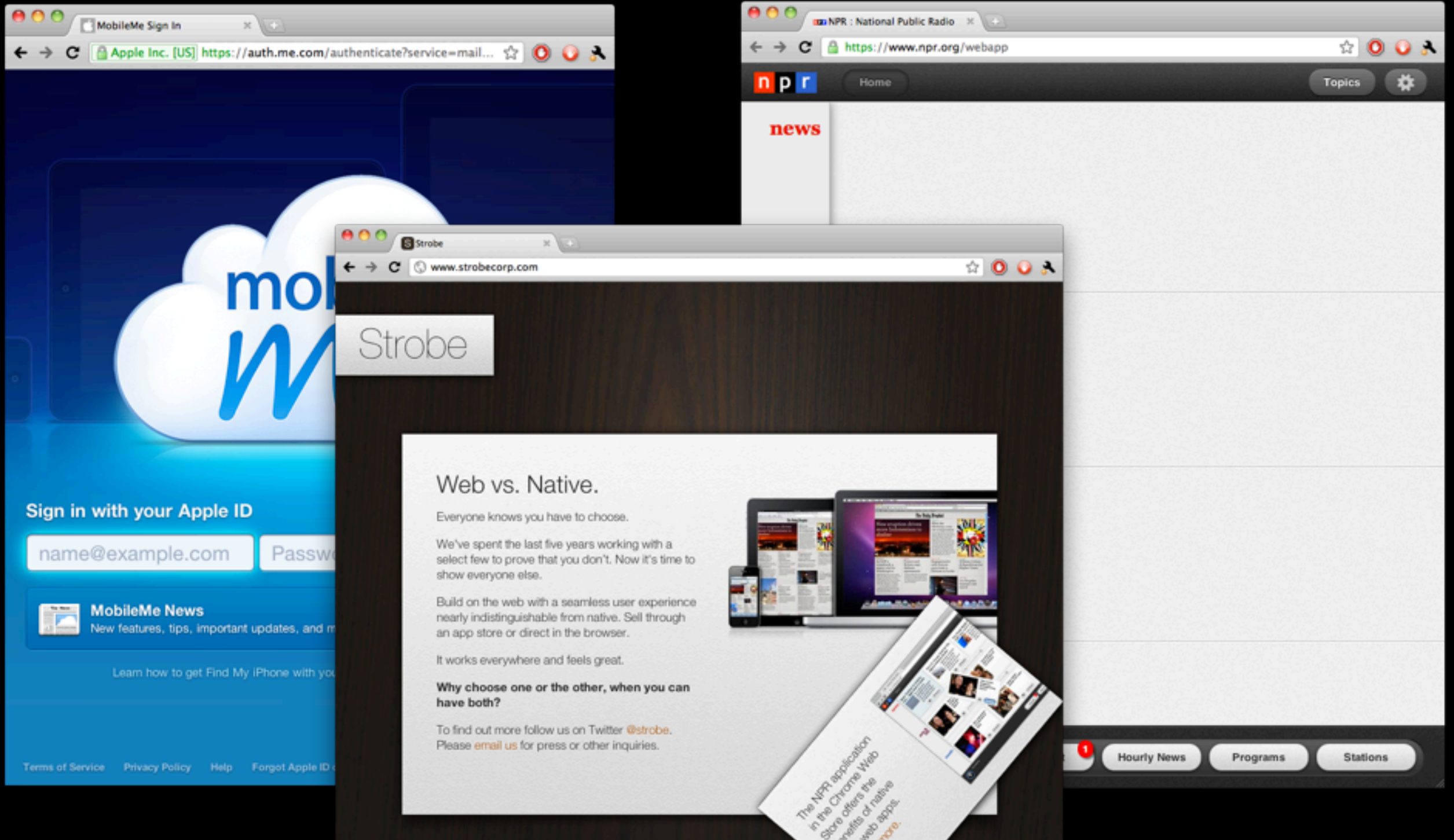


Geschichte



Sproutit für Mailroom entwickelt
Apple adaptiert für MobileMe und iWork.com, Programmierer angestellt dort
Heute: Spinoff in Strobe, Inc.

Wer verwendet?



MobileMe/iWork.com, NPR, Strobe

But SproutCore isn't your mama's JavaScript library.

Zitat von der Webseite

SproutCore ist nicht etwas, das man einer Webseite hinzufügt
SproutCore generiert die Webseite selber
Man programmiert nur mehr JavaScript, so gut wie kein HTML
Ähnlicher zu Cocoa als zu Webseitenprogrammierung

Webseite



Webapplikation



Sudoku ist mit Sproutcore gemacht, gratis im Google Chrome Store

MVC!

SproutCore hat eine strikte MVC-Trennung

MVCSD = Model View Controller Store Datasource
dazu später

MVC!

Naja, MVCSD

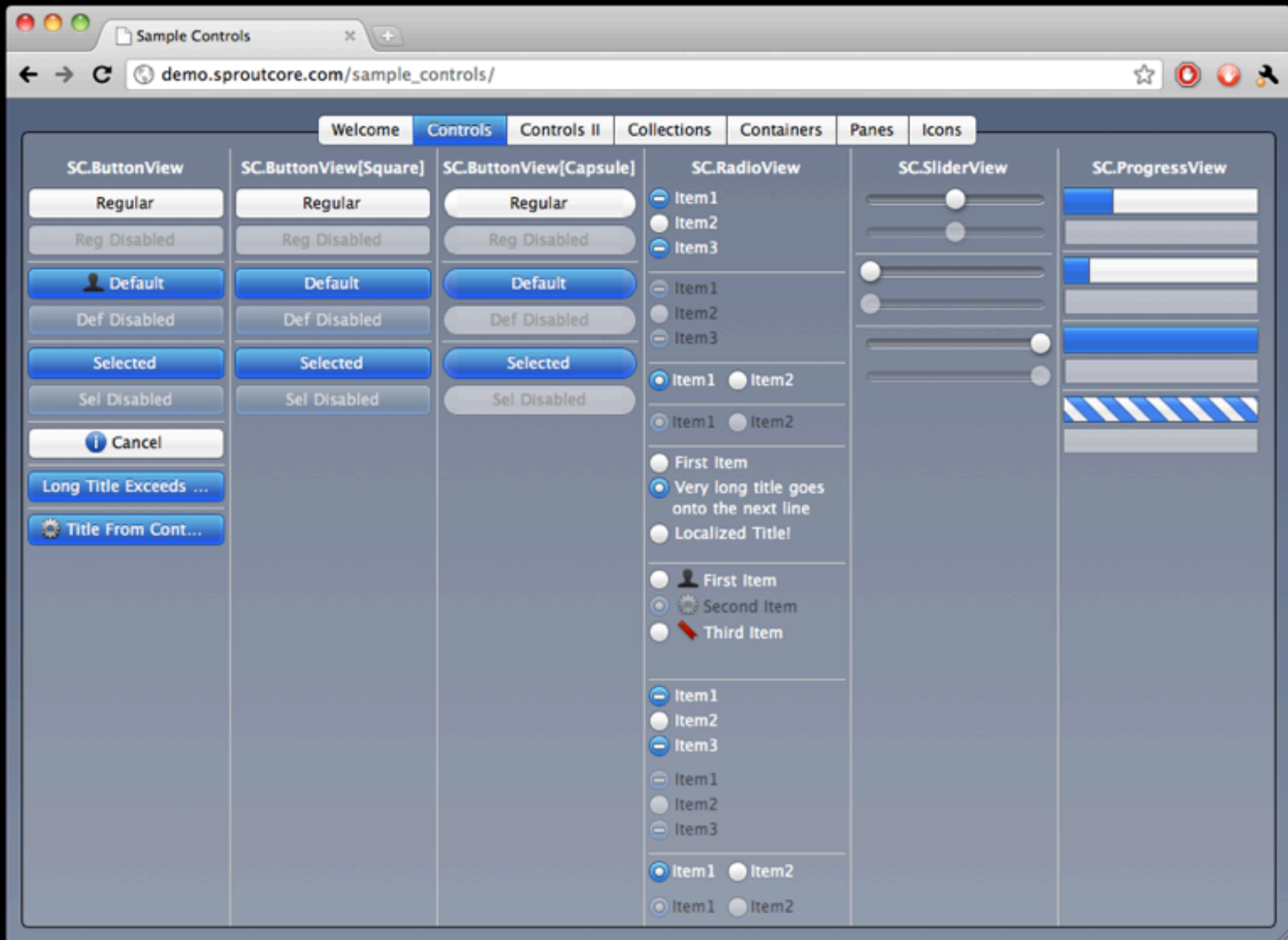
SproutCore hat eine strikte MVC-Trennung

MVCSD = Model View Controller Store Datasource
dazu später

```
MyApp.mainPage = SC.Page.design({
  mainPane: SC.MainPane.design({
    childViews: 'labelView'.w(),
    labelView: SC.LabelView.design({
      layout: { centerX: 0, centerY: 0, width: 200, height: 18 },
      textAlign: SC.ALIGN_CENTER,
      tagName: "h1",
      value: "Welcome to SproutCore!"
    })
  })
});
```

view-beispiel

layout erklären (left, right, top, bottom, centerX, centerY, width, height)



http://demo.sproutcore.com/sample_controls/

Custom Views?

sehr einfach, man generiert in einer Klasse das HTML und kanns mit jQuery modifizieren
Event handling fast identisch zu Cocoa

jQuery ist erst ab 1.5 (der nächsten Version) integrierter Teil von SproutCore, in 1.4 kann
mans aber als Framework einbinden und es wird automatisch verwendet
CoreQuery erklären

Controller?

Wie sieht jetzt der Controller-Layer aus?

Bindings



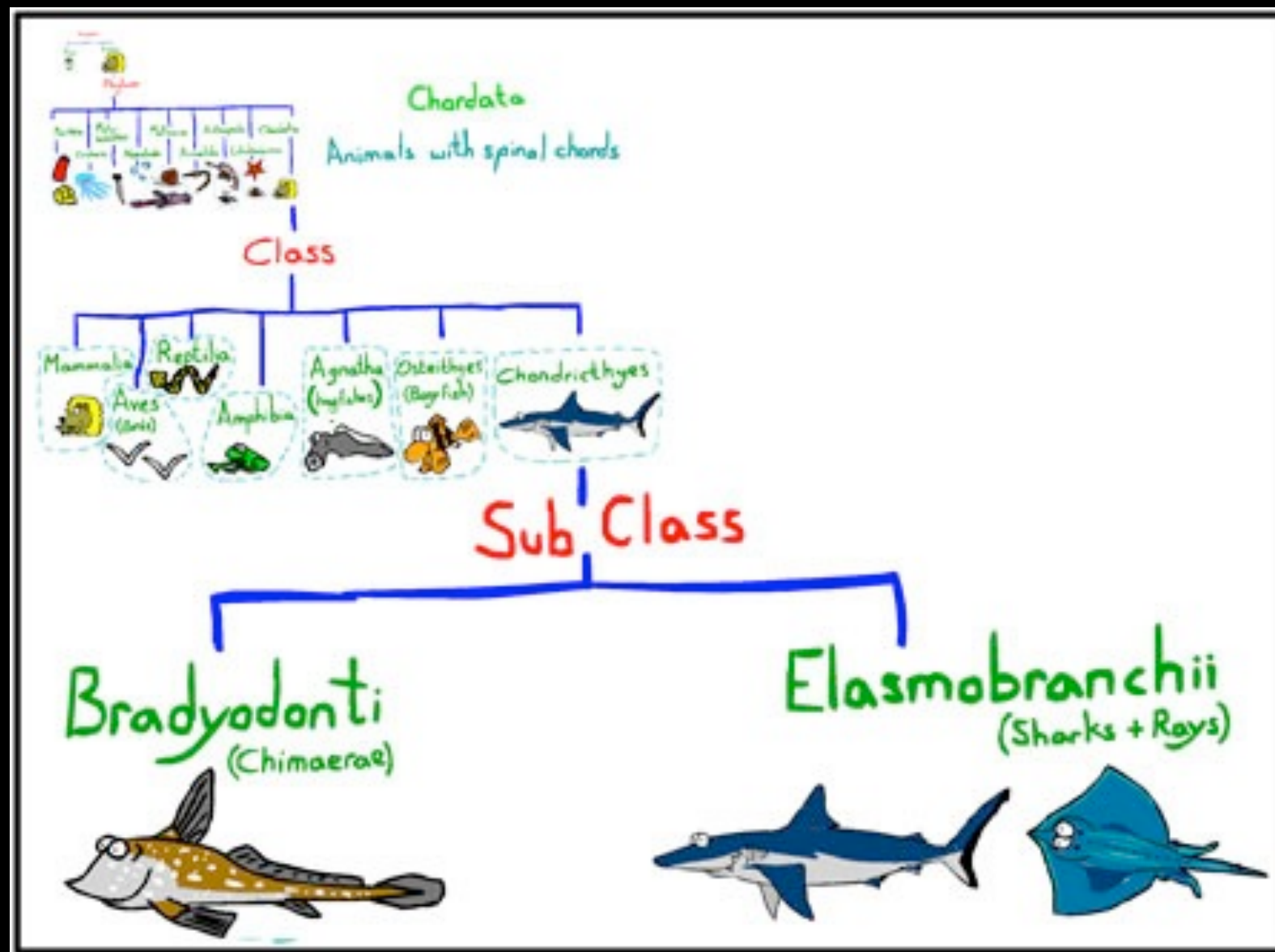
Bindings für iOS-devs erklären

SC.ObjectController

SC.ArrayController

Funktionieren wie in Cocoa, beide erklären
“Subklassen” davon anlegen

Subklassen?



Wie geht denn das in JavaScript?
SproutCore hat sein eigenes System dafür

```
MyApp.UserArrayController = SC.ArrayController.extend({
    selectionChanged: function() {
        var selection = this.get('selection');

        if(selection.length() == 1) {
            var user = selection.firstObject();
            var chat = this.chats[user];
            chat.show();
        }
    }.observes('selection')
});
```

mit extend erstellt man eine subklasse
accessor erklären (Key Value Coding)
observes erklären! (Key Value Observing)

```
projectTreeView: SC.ListView.design({  
  layout: { top: 5, width: 300, bottom: 5, left: 5 },  
  contentValueKey: "name",  
  contentBinding:  
    "Todos.projectsTreeController.arrangedObjects",  
  selectionBinding :  
    "Todos.projectsTreeController.selection"  
}),
```

Beispiel für ein Binding


```
MyApp.GameObject = SC.Object.extend({
  position: { x: 0, y: 0 },
  width: 0,
  height: 0,

  isoPosition: function() {
    var x = this.get('position').x;
    var y = this.get('position').y;

    return {
      x: 40 * (x - y) - (this.get('width') - 80) / 2,
      y: 40 * (x + y) * 0.75
    };
  }.property('position', 'width').cacheable()
});
```

SC.Object ist wie NSObject
property() erklären, cacheable() erklären



AJAX

Auch SproutCore-Apps müssen ab und zu mit dem Server kommunizieren. Das passiert via AJAX.

DataStore

Der SproutCore DataStore ist eine lokale OO-Datenbank, die sich mit einem Server synchronisieren kann (entspricht in etwa CoreData).
Ein Eintrag darin nennt sich "Record"

```
Todos.Task = SC.Record.extend({  
    isDone: SC.Record.attr(Boolean),  
    description: SC.Record.attr(String)  
});
```

ist eine Subklasse

die Felder müssen nicht angegeben werden hier (ist ja nur ein dictionary), aber wenn man es so macht gibt es typchecks

“Fixtures” werden verwendet, um Beispieldaten für die Entwicklung einzufügen in die datenbank

wird hier nicht näher eingegangen

```
Todos.Task = SC.Record.extend({
  isDone: SC.Record.attr(Boolean),
  description: SC.Record.attr(String),

  project: SC.Record.toOne("Todos.Project", {
    inverse: "tasks", isMaster: NO
  }),
});
```

Relationen kann man auch darstellen

```
var query = SC.Query.local(MyApp.User,  
    '(subscription = "both") OR (subscription = "to")');  
var users = MyApp.store.find(query);
```

users ist danach ein array, das sich dynamisch updatet, wenn Objekte dazukommen oder entfernt werden!



Interface zum Server?

SproutCore ist serveragnostisch, da kann irgendeine Software drauf laufen
Konsequenz: Man muss auch die Clientseite komplett selber schreiben



DataSource

Dieses Interface nennt sich DataSource
DataSource bekommt mit wenn neue Objekte angelegt/gelöscht werden im store, und kann selber Objekte anlegen/löschen



Achtung, Bug mit Relationen!

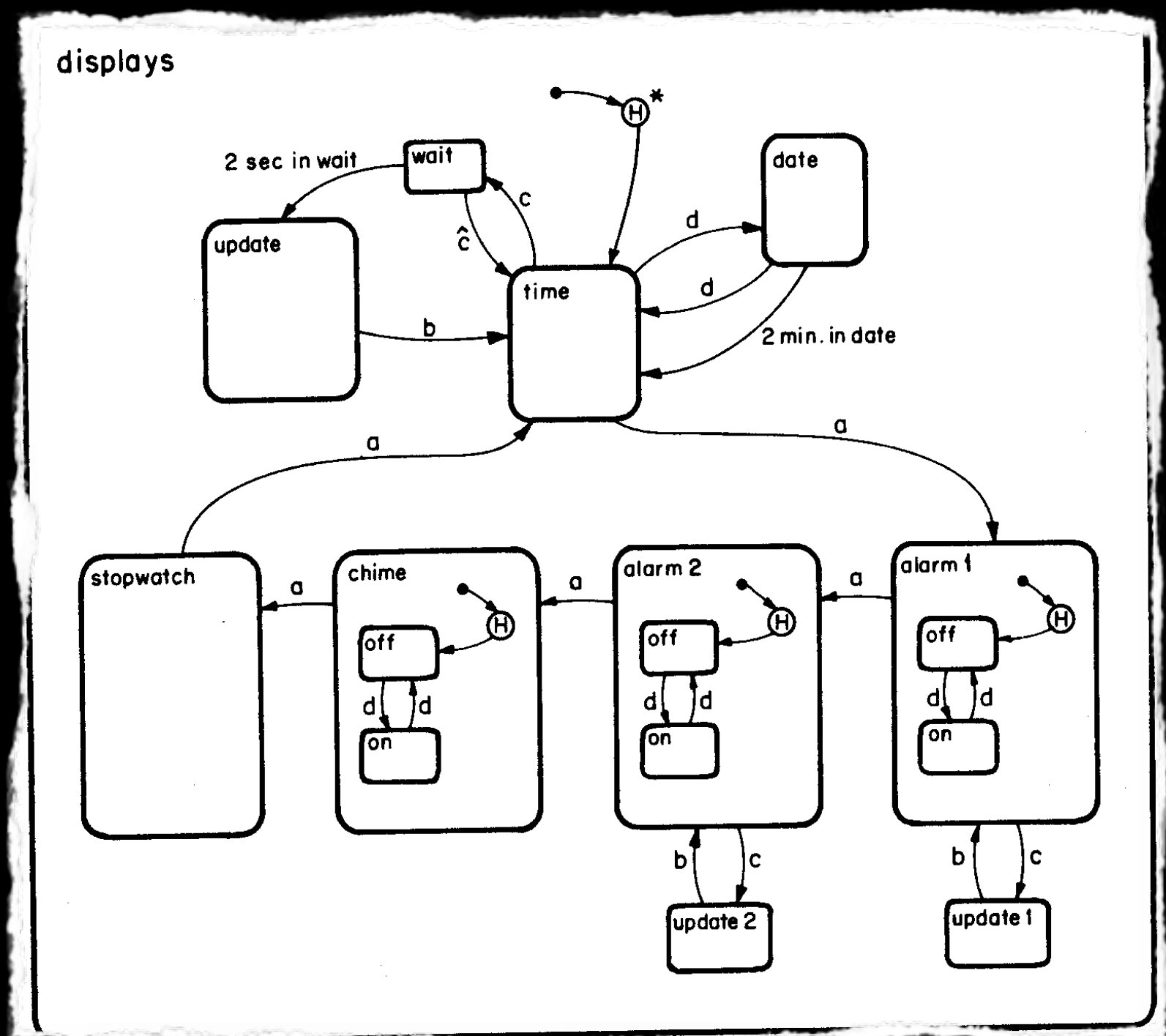
Relationen werden in der DataSource nicht richtig gehandhabt, die Rückreferenzen werden nicht aktualisiert.

Ki



Statecharts-Implementation für SproutCore
integrierter Teil von 1.5+
auch als Framework in 1.4

statechart diagram



[https://frozenscanuck.wordpress.com/
2011/02/15/ki-just-got-better/](https://frozenscanuck.wordpress.com/2011/02/15/ki-just-got-better/)

Eigentlich könnte man für dieses Thema einen eigenen Vortrag halten, daher hier nur mal ein Link mit mehr Informationen.

Wie fange ich an?

www.sproutcore.com

#sproutcore auf FreeNode
@sproutcore

Dokumentation ist sehr schlecht, es wird aber grade intensiv dran gearbeitet
[klick] neue guides im Stil von Ruby on Rails werden gerade geschrieben
Buch ist auch im Entstehen

IRC-Channel ist am gleichen Netzwerk wie #cocoaheads_at

Wie fange ich an?

www.sproutcore.com

#sproutcore auf FreeNode
@sproutcore

Dokumentation ist sehr schlecht, es wird aber grade intensiv dran gearbeitet
[klick] neue guides im Stil von Ruby on Rails werden gerade geschrieben
Buch ist auch im Entstehen

IRC-Channel ist am gleichen Netzwerk wie #cocoaheads_at

SproutCore: Writing Code for a Multi-Device World

- O'Reilly Webcast
- Yehuda Katz, Strobe Inc. (@wycats)
- Dienstag, 22.2.2011, 19:00-20:00 CET
- Anmeldung:
<http://oreillyn.net.com/pub/e/1853>

Entwicklungsumgebung?

Xcode ungeeignet
Dashcode als Texteditor geht
TextMate soll gut gehen
vim/emacs natürlich
eclipse

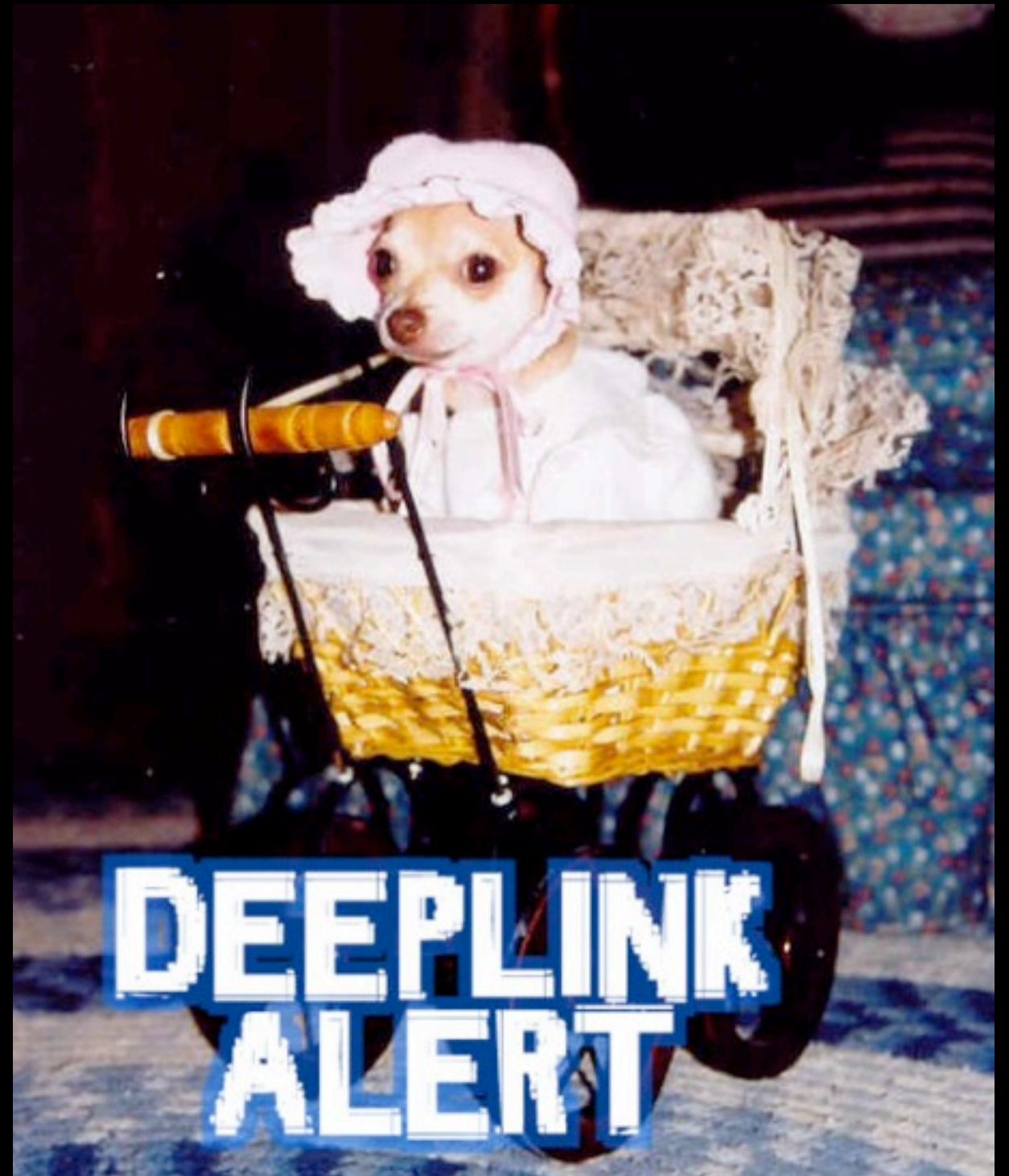
Amber



<http://blog.sproutcore.com/post/3075780393>

Optimierung von Sproutcore für mobile Geräte
man kann auswählen, welche Teile man für sein App haben will
zB die ganzen Controls sind meistens eher unnötig, will eh der Designer komplett anders haben

Kontroverse Deep Links



SproutCore-Apps sind für den Browser nur ein File, daher keine History und kein direktes verlinken von Seiten.

HTML5 bringt eine Funktion für Browser History (nächster slide)

pushState

```
window.history.pushState(data, "Title", "/new-url");
```

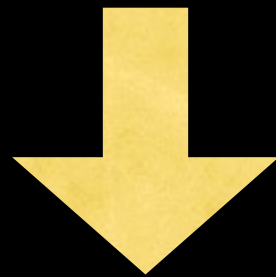
kommt aber nicht in IE9!

Mehr Infos: <http://badassjs.com/post/840846392>

Hashbangs



http://twitter.com/cocoaheads_at



http://twitter.com/#!/cocoaheads_at

Wird von Google unterstützt, wandelt die URL automatisch in einen speziellen Request um, mit der der Server die reinen Daten ohne JavaScript schicken kann. SproutCore hat dafür keine Funktionalität, kann man aber problemlos selber dazubauen.

Aber: Wenn man eine Webseite hat, wo man sehr viele solche Deep Links braucht sollte man sich fragen, ob SproutCore überhaupt die richtige Lösung ist. Niemand würde eine Newsseite in Cocoa programmieren!

Meine Anwendung

ein rundenbasiertes Web-MMO
Interface zum Server ist via XMPP

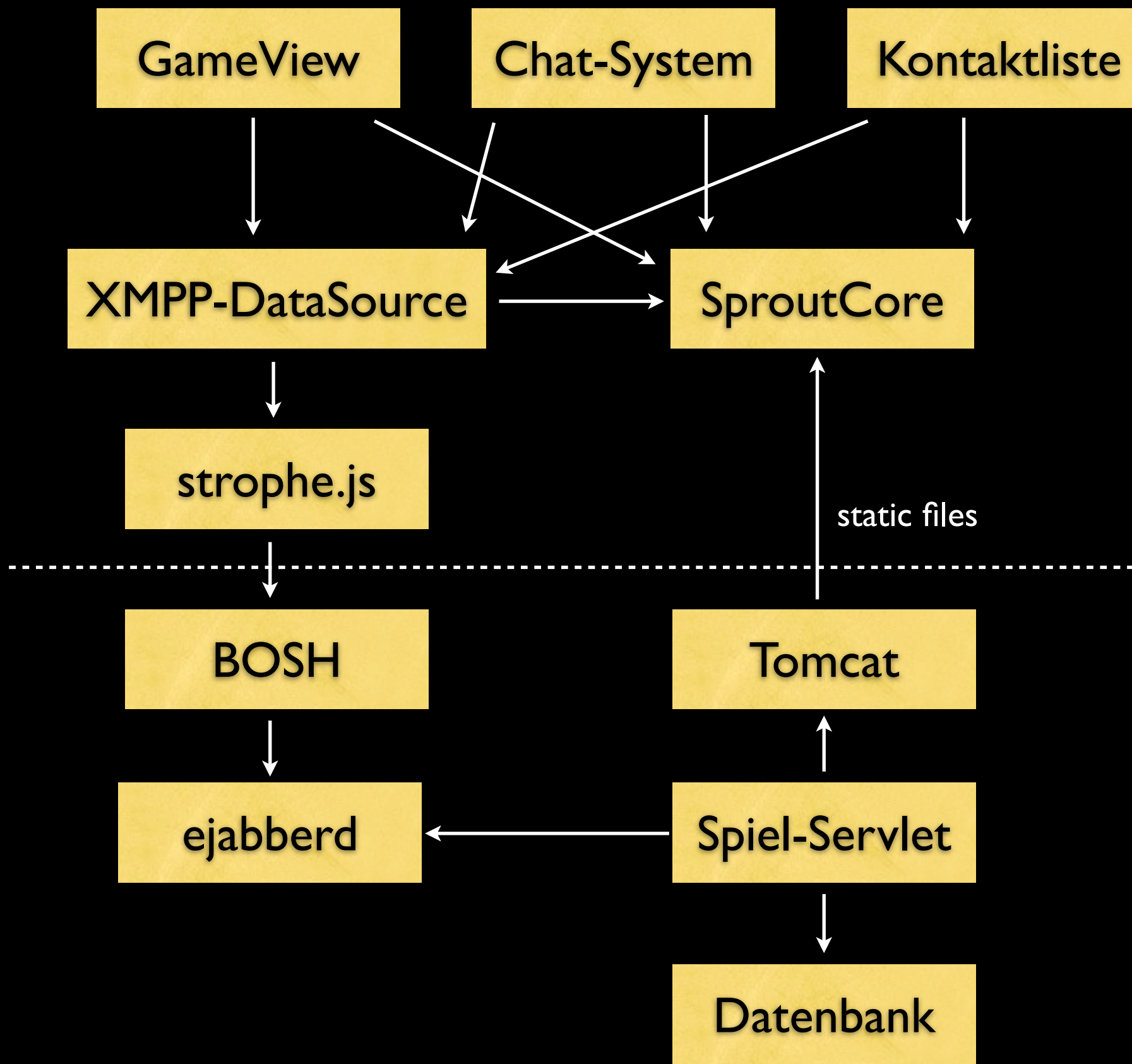


Diagramm erklären

dann Demo (hoffentlich!)
sc-server und sc-gen erklären dabei!