bacon.js+AngularJs

~~bacon.js+AngularJs~~

Selfish + FRP

AngularJs Routes

AngularJs routes:

- initialization on configuration level (good)
- route declaration IS declaration (best)
- declaration of link route -> controller -> tpl (oka-ay)
- resolve method (wow)
- resolve can inject (several) promises into controller (!!!)

```
app.config( function ($routeProvider) {
    …
    $routeProvider
            .when('/', {
                        templateUrl: '/index.html',
                        controller: 'Index',
                        resolve:{
                              post:getPost,
                              story:getStory
                        }
            )
    …
});
```

← Example

Bad things:

- no template can be attached to rejected case

- reject just sends the event $routeChangeError (can be threated somehow ‘good’)

- manual promises nesting
- no parallel/nested views (only one ng-view)
- less number even of bad things

AngularUI Routes

AngularUI Router

- state machine - (god bless you!)
- states are bound to named, nested and parallel views (exellent!!)
- ui-sref state transition directive (wah!)
- $state.go state transition function
- dependencies are inherited into nested views (cool!)

Example

```
.config(function($stateProvider, $urlRouterProvider) {
  $stateProvider
    .state('list', {  url: '/list', templateUrl: ‘list.html', controller: 'ListCtrl'    })
    .state('list.item', {
       url: '/:item',
       templateUrl: ‘list.item.html',
       controller: function($scope, $stateParams) {
           $scope.item = $stateParams.item;
       }
    })
})
```

index.html

```
<!DOCTYPE html>
…
<div class="container">
    <h1> Title   </h1>
    <div ui-view></div>
 </div>
```

list.item.h

```
<img ng-src="/assets/images/{{item}}.jpg" />
```

list.html

```
<div>
         Choose the picture:
         <a ng-repeat="item in stories" ng-href="/list/{{item.id}}"</a>
         <div ui-view></div>
</div>
```

Bad things:

- requires Angular Routes…

- partial ‘url’ param of nested states are controversial

- mess up with ‘ui-view’...

- no reject templates (can be workarounded with $state.go in promise reject)

- manual promises nesting

Stream Routes

Stream Routes:

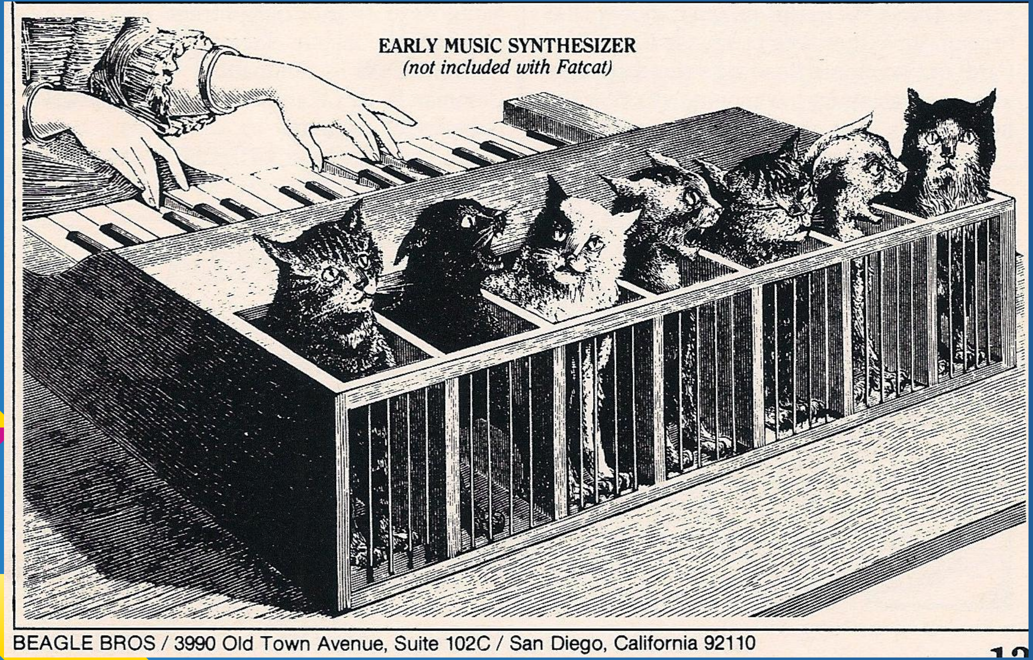
- nested, parallel views! (I’ve dreamed about it)
- parallel/nested dependencies (in shoke)
- template can be attached to any endpoint
- filtering endpoints
- view is declared by unique attribute. No ‘ui-view’ mess up.

- caching data dependencies
- intellectual caching (ovations)

Even More!

- data refreshes on route params change
- route control flow (several matches, last matches)
- any animation on route change
- ‘when’ can attach endpoint to stream (!)
- each endpoint is stream!
- possibility to create non-angular render into view and run application on server.

It is FRP, baby


EARLY MUSIC SYNTHESIZER
(not included with Fatcat)
BEAGLE BROS / 3990 Old Town Avenue, Suite 102C / San Diego, California 92110

(using bacon.js)

Example

```
$streamRoute
        .defaultView('base-view')
                   .animate(slideUp)
        .when( '/list’, ‘/stories’)
            .depends( getStories, 'stories', 'storyId' )
           .and( getSomething, ‘something’, ‘somethingId’)
                .if( notSuccess )
                    .render(void 0, '/story/tpl-access-denied.html')
                            .animate(FadeInOut)
            .render('view','/tpl-list.html')
         .when( ‘/list/:itemId’)
                   render(void 0, ‘/tpl-list.item.html’, ‘base-view.item’)
```

index.html

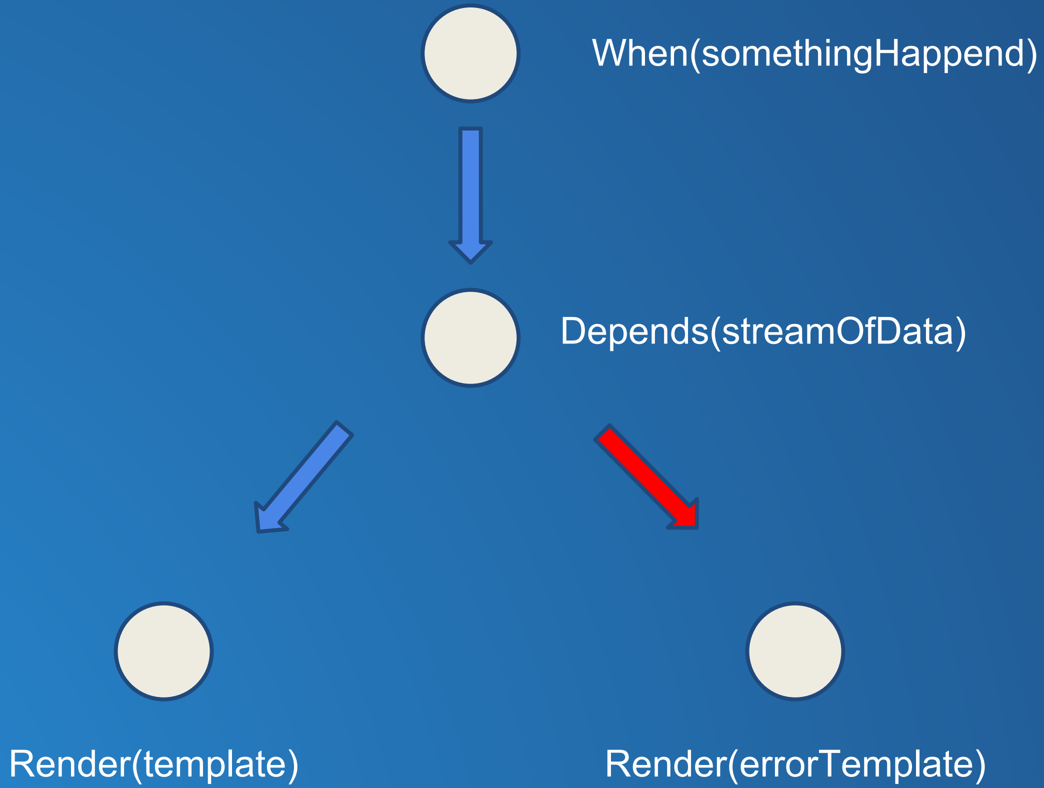
```
<!DOCTYPE html>
…
<div class="container">
    <h1> Title   </h1>
    <div base-view></div>
 </div>
```

list.item.h

```
<img ng-src="/assets/images/{{stories[itemId]}}.jpg" />
```

list.html

```
<div>
         Choose the picture:
        <a ng-repeat="item in stories" ng-href="/list/{{item.id}}”</a>
</div>
<div item></div>
```


When(somethingHappend)
Depends(streamOfData)
Render(template)
Render(errorTemplate)

Outputs

- When:

{ params: { itemId: 10}, "/list/10", route: “/list/:itemId” }

- Depends:

{ …, stories: {...} }

- If: the same, just filtered.

- Render: stops execution.

Links

- http://angularjs.org/
- https://github.com/angular-ui/ui-router
- https://github.com/baconjs/bacon.js/tree/master