

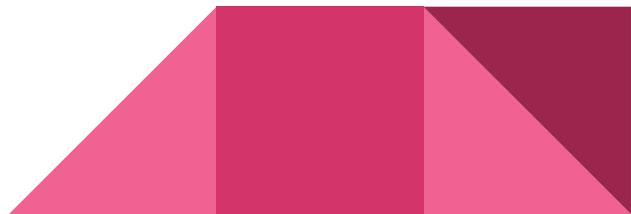
Not Your High School Geometry Class

How to Develop for the Browser Client with AngularJS

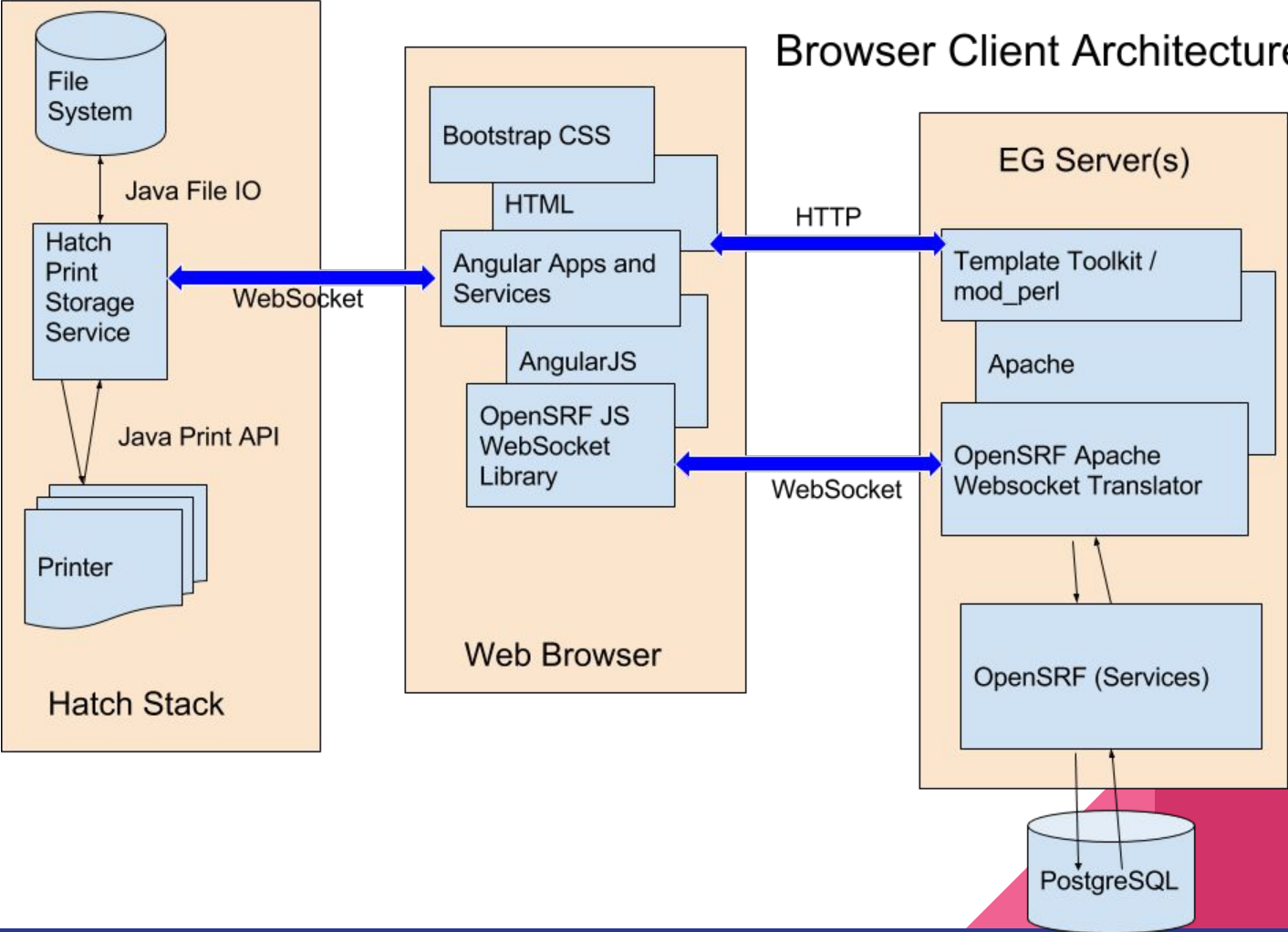
2016 Evergreen Conference – 22 April 2016

The three amigos

- Bill Erickson, KCLS – architecture of the browser client
- Mike Rylander, ESI – what is AngularJS anyway?
- Galen Charlton, ESI – widgets, widgets, and more widgets



Browser Client Architecture



Templates with Style

- Template Toolkit
- Bootstrap CSS
- Local CSS



Scripting

AngularJS

```
<div>{{hold_count}}</div>
```

```
$scope.hold_count = array_of_holds.length;
```



new WebSocket(path);

Persistent, full-duplex data streams.

- <https://github.com/disconnect/apache-websocket>
- `OpenSRF/src/gateway/osrf_websocket_translator.c`



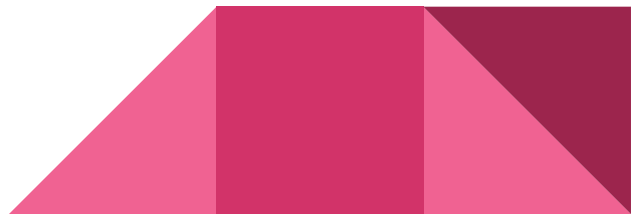
Network API : Typical Use

```
egCore.net.request('open-ils.foo', 'open-ils.foo.give.me.things', abc)
  .then(
    function(thing) { console.log('got a thing: ' + thing) }
  );
```



Network API : Streaming

```
egCore.net.request('open-ils.foo', 'open-ils.foo.give.me.things', abc).then(  
  function() { console.log('all done') },  
  null, // exception  
  function(a_thing) { console.log('got one thing: ' + a_thing) }  
  });
```



I18N / L10N

- Static Strings

```
<div>[% I('Renew Items') %]</div>
```

- Dynamic Template Strings

```
<div>[% I('Welcome To [_1]', '{{lib_name}}') %]</div>
```

```
$scope.lib_name = some_org.name();
```



Scripting: Modules

- angular-ui-bootstrap
- Hotkeys
- Ngtoast
- Angular-file-saver
- ...



Build and Deploy

- Bower - Manage Dependencies
- Jasmine - Unit Test Framework
- PhantomJS - Headless JS engine
- Grunt - Coordination (also minification)
- Node - Required by the above



Browsers Can Do Most of the Things

“Hatch” for...

- Seamless printing to multiple printers
- External file storage
- 3rd-party software integration



What is this ANGU-LAR?

AngularJS is a structural framework for dynamic web apps. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application's components clearly and succinctly. Angular's data binding and dependency injection eliminate much of the code you would otherwise have to write. And it all happens within the browser, making it an ideal partner with any server technology.

Angular is what HTML would have been, had it been designed for applications. HTML is a great declarative language for static documents. It does not contain much in the way of creating applications, and as a result building web applications is an exercise in what do I have to do to trick the browser into doing what I want?

The impedance mismatch between dynamic applications and static documents is often solved with:

- a library - a collection of functions which are useful when writing web apps. Your code is in charge and it calls into the library when it sees fit. E.g., jQuery.
- frameworks - a particular implementation of a web application, where your code fills in the details. The framework is in charge and it calls into your code when it needs something app specific. E.g., durandal, ember, etc.

Thanks for the wall of text, buddy

- Data binding
 - `{{important_value}}`
 - `$scopes` feed bound data, and forms update `$scopes`
- Directives to invent new HTML elements
 - `<eg-draw-the-org-tree-as-a-dropdown selected="the_library"/>`
 - We use this for most repeated behaviors.
- Markup based control structures -- loops, conditionals
 - `<div ng-show="itIsTrue()">Not False</div>`
 - `<ul ng-repeat="thing in pile">Thing is {{thing}}`



Good stuff

- HTML Templates
 - Inline and separate
- Support for forms and form validation
 - Bind form to data, make sure it fits your object shape
- Attaching new behavior to DOM elements, such as DOM event handling
 - Notice clicks, of course
 - Throw special events at the UI



MOAR GOOD

- Promises
 - `promise.then(success_handler, fail_handler, notify_handler).finally(handler, notify_handler)`
 - ERMAHGERD, promises are my favorite things!
- Services and service factories
 - Singleton data providers
- Filters
 - Transform and filter data the same way, everywhere



Where to find stuff

- Open-ILS/web/js/ui/default/staff
 - `services/` ⇒ shared services and directives
 - `module/page/app.js` ⇒ controller for page
 - `test/` ⇒ unit test goodness
- Open-ILS/src/templates/staff
 - `share/` ⇒ shared templates
 - `module/page/index.tt2` ⇒ index page
 - `t_foo.tt2` ⇒ template include, directive template, etc.
 - `css/` ⇒ local CSS



Evergreen's custom directives

- egGrid
- egOrgSelector



egGrid

```
<eg-grid  
  ng-hide="forbidden"  
  features="allowAll,-display"  
  id-field="id"  
  idl-class="acp"  
  auto-fields="true"  
  grid-controls="gridControls"  
  menu-label="[% l('Buckets') %]"  
  persist-key="cat.bucket.copy.view">
```

...



egGrid

```
[% INCLUDE 'staff/cat/bucket/copy/t_grid_menu.tt2' %]
```

```
<eg-grid-action label="[% l('Request Selected Copies') %]"  
  handler="requestItems"></eg-grid-action>
```

```
<eg-grid-field path="id" required hidden></eg-grid-field>
```

```
<eg-grid-field path="call_number.record.id" required hidden></eg-grid-field>
```

```
<eg-grid-field label="[% l('Barcode') %]" path='barcode' visible>
```

```
  <a target="_self" href="[% ctx.base_path %]/staff/cat/item/{{item['id']}}">  
    {{item['barcode']}}  
  </a>
```

```
</eg-grid-field>
```

```
</eg-grid>
```



egGrid menus

```
<!-- global grid menu displayed on every Bucket page -->  
<eg-grid-menu-item label="[% l('New Bucket') %]"  
  handler="openCreateBucketDialog"></eg-grid-menu-item>  
  
<eg-grid-menu-item label="[% l('Edit Bucket') %]"  
  handler="openEditBucketDialog"></eg-grid-menu-item>  
  
<eg-grid-menu-item label="[% l('Delete Bucket') %]"  
  handler="openDeleteBucketDialog"></eg-grid-menu-item>  
...
```



egGrid

```
function drawBucket() {
  return bucketSvc.fetchBucket($scope.bucketId).then(
    function(bucket) {
      var ids = bucket.items().map(
        function(i){return i.target_copy()}
      );
      if (ids.length) {
        $scope.gridControls.setQuery({id : ids});
      } else {
        $scope.gridControls.setQuery({});
      }
    }
  );
}
```



egGrid

```
$scope.requestItems = function() {  
  var copy_list = $scope.gridControls.selectedItems().map(  
    function (i) {  
      i.id;  
    }  
  );  
};
```

...



... and custom services

- egCore
- egStrings
- egIDL
- egNet
- egPCRUD
- egAlertDialog / egConfirmDialog / egPromptDialog



egCore


- Bucket of stuff
 - egCore.net (egNet)
 - egCore.auth
 - egCore.pcrud
 - egCore.perm
 - egCore.hatch
 - egCore.idl
 - egCore.strings
 - egCore.env



egStrings

```
<script>
angular.module('egCoreMod').run(['egStrings', function(s) {
s.PATRON_CARD_INACTIVE =
  "[% l('The card used to retrieve this account is inactive and may not be used to
circulate items.') %]";
s.PATRON_INACTIVE =
  "[% l('This account is inactive and may not circulate items.') %]";
s.PATRON_ACCOUNT_EXPIRED =
  "[% l('This account has expired and may not circulate items.') %]";
s.CIRC_CLAIMS_RETURNED =
  "[% l('Item "[_1]" is marked as Claims Returned', '{{barcode}}') %]";
...

```



egStrings in action...

```
case 'CIRC_CLAIMS_RETURNED':  
    return service.exit_alert(  
        egCore.strings[evt[0].textcode],  
        {barcode : params.copy_barcode}  
    );
```




egIDL

```
var bucket = new egCore.idl.ccb();  
bucket.owner(egCore.auth.user().id());  
bucket.name(name);  
bucket.description(desc || '');  
bucket.btype('staff_client');
```



egNet

```
egCore.net.request(  
  'open-ils.actor',  
  'open-ils.actor.container.create',  
  egCore.auth.token(), 'copy', bucket  
)  
.then(function(resp) {  
  if (resp) {  
    if (typeof resp == 'object') {  
      console.error('bucket create error: ' + js2JSON(resp));  
      deferred.reject();  
    } else {  
      deferred.resolve(resp);  
    }  
  }  
});
```



egPCRUD

```
egCore.pcrud.retrieve('bre', $scope.recordId)
```

```
...
```

```
egCore.pcrud.retrieveAll('cbs', {}, {atomic : true})
```

```
.then(function(list) { $scope.bib_sources = list; });
```



Dialogs

```
egConfirmDialog.open(  
  egCore.strings.CONFIRM_REFUND_PAYMENT, '',  
  {  xactIds : ''+ids,  
    ok : function() {  
      // reset the received payment amount.  this ensures  
      // we're not mingling payments with refunds.  
      $scope.payment_amount = 0;  
    }  
  }  
);
```



... and borrowed widgets

- ngToast
- FileSaver ⇒ egJsonExporter



Toast



"Toast" by Ozzy Delaney CC-BY on Flickr: <https://www.flickr.com/photos/24931020@N02/sets/72157632906852850/>

No really, toast

```
.config(['ngToastProvider', function(ngToastProvider) {
  ngToastProvider.configure({
    verticalPosition: 'bottom',
    animation: 'fade'
  });
}])
...
ngToast.create(egCore.strings.MARK_OVERLAY_TARGET);
```



Piles of little widgets make big widgets

- egMarcEditRecord

```
<eg-marc-edit-record fast-add="true"  
on-save="refresh_record_callback"  
dirty-flag="stop_unload" record-id="record_id">  
</eg-marc-edit-record>
```



Thanks!

