LA Aqueduct Filtration Plant Web App

Technical overview

CJ Cenizal

Department of Water Quality

Home | Locations | Help

LA Aqueduct Filtration Plant

view earlier samples

Time	$CHCl_3$	$CHBr_3$	$CHBrCl_2$	CBr_2Cl_2	
18:00	0.00995	0.00995	0.00995	0.00995	
18:20	0.00995	0.00995	0.00995	0.00995	
18:40	0.00995	0.00995	0.00995	0.00995	
19:00	0.00995	0.00995	0.00995	0.00995	
19:20	0.00995	0.00995	0.00995	0.00995	
19:40	0.00995	0.00995	0.00995	0.00995	
20:00	0.00995	0.00995	0.00995	0.00995	
20:20	0.00995	0.00995	0.00995	0.00995	
20:40	0.00995	0.00995	0.00995	0.00995	
21:00	0.00995	0.00995	0.00995	0.00995	

view later samples

ALERT TECHNICIANS

Department of Water Quality

Home | Locations | Help

LA Aqueduct Filtration Plant

Technicians at this location are currently on alert

view earlier samples

Time	$CHCl_3$	$CHBr_3$	$CHBrCl_2$	CBr_2Cl_2	
04:00	0.00995	0.00995	0.00995	0.00995	
04:20	0.00995	0.00995	0.00995	0.00995	
04:40	0.00995	0.00995	0.00995	0.00995	
05:00	0.00995	0.00995	0.00995	0.00995	
05:20	0.00995	0.00995	0.00995	0.00995	
05:40	0.00995	0.00995	0.00995	0.00995	
06:00	0.01167	0.01167	0.01167	0.01167	
06:20	0.01167	0.01167	0.01167	0.01167	
06:40	0.00995	0.00995	0.00995	0.00995	⚠

view later samples

ALERT TECHNICIANS

Sections

Product requirements
Tech stack
Domain definition
Application layer
View layer
Retrospective

Product requirements

- Single-page app
- Samples table, alert feature
- Table updates in real-time

LA Aqueduct Filtration Plant

1 ⚠ Technicians at this location are currently on alert

2 ▲ view earlier samples ▲

Time	$CHCl_3$	$CHBr_3$	$CHBrCl_2$	CBr_2Cl_2	
01:00	0.00995	0.00995	0.00995	0.00995	
01:20	0.00995	0.00995	0.00995	0.00995	
01:40	0.00995	0.00995	0.00995	0.00995	
02:00	0.01167	0.01167	0.01167	0.01167	
02:20	0.02705	0.02705	0.02705	0.02705	
02:40	0.02705	0.02705	0.02705	0.02705	⚠
03:00	0.02705	0.02705	0.02705	0.02705	⚠
03:20	0.00995	0.00995	0.00995	0.00995	⚠
03:40	0.00995	0.00995	0.00995	0.00995	⚠

Dibromochloromethane

3

4

5

6

▼ view later samples ▼

7 Alert Technicians

Which requirement to defer?

1. Message shows when location is under alert.
2. Table is limited to 10 records at a time, ordered by time, with pagination at top and bottom. Pagination in either direction is disabled when the first or last record is reached.
3. Molecular formulae are expanded to common name in a tooltip on hover: $CHCl_3$ - Chloroform, $CHBr_3$ - Bromoform, $CHBrCl_2$ - Bromodichloromethane, CBr_2Cl_2 - Dibromochloromethane.
4. Rows are highlighted according to the total amount of all contaminants in the record. Yellow - over 0.0400, red - over 0.0800.
5. The first row below 0.0400 after a yellow or red should be colored green.
6. An alert icon is shown for each record at a time when the facility was under alert.
7. An “Alert Technicians” action button is enabled only when the facility is not currently under alert and the latest record is in a yellow or red state.

Alerting is important = high value, high cost.
Molecular formulae expansion = low value, low cost.
Green row = low value, high cost.

Tech stack

- React - Low-risk evaluation opportunity.
- Flux - Clear data flow.
- SCSS - Familiar, widespread.
- NPM scripts - Build process w/ fewer dependencies.
- PostCSS - Vendor prefixes w/o mixins.
- Browserify - Modular code (CommonJS).

Domain: Sample model

06:00	0.01167	0.01167	0.01167	0.01167	⚠

Persisted: ID, time, activeAlert, readings

Computed:

- Human-readable time value
- Total (sum of all readings) value
- Alert level (default, warning, danger)

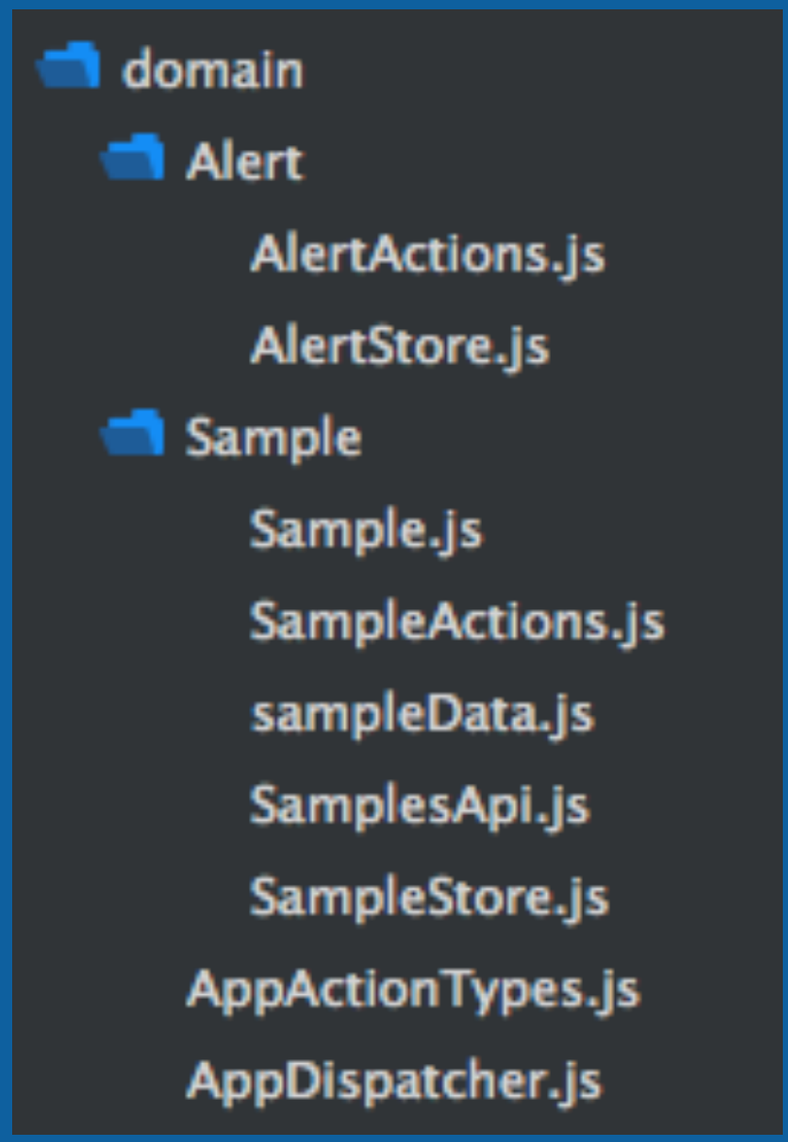
Domain: Alert state

- “isAlert” boolean state
- “isAlertable” boolean state

Application layer

Flux makes it easy to model this domain w/ stores, actions, and services.

Encourages SRP.



Actions mediate changes to the application layer, made by the view layer.

```
fetchSamples: function() {
  SamplesApi.fetch().then(function(response) {
    AppDispatcher.dispatch({
      type: AppActionTypes.SAMPLE_ADDED,
      payload: {
        samples: response.samples
      }
    });
  }, function(error) {
  });
},

nextPage: function() {
  AppDispatcher.dispatch({
    type: AppActionTypes.NEXT_PAGED
  });
},

previousPage: function() {
  AppDispatcher.dispatch({
    type: AppActionTypes.PREVIOUS_PAGED
  });
}
```

Fake API simulates requests and responses w/ promises.

```
var _isInitial = true;
var _sampleIndex = 34;
var _startTime = new Date();

var THRESHOLD_SECONDS = 3;

function _getNextSample() {
  // Bootstrap app with some data.
  if (_isInitial) {
    _isInitial = false;
    var data = [];
    for (var i = 0; i < _sampleIndex; i++) {
      data.push(sampleData[i]);
    }
    return data;
  }
  // Map 20 minutes in recorded time to a few seconds in real time.
  var currentTime = new Date();
  var secondsElapsed = (currentTime - _startTime) / 1000;
  if (secondsElapsed >= THRESHOLD_SECONDS) {
    // Reset the timer and return the next sample.
    _startTime = new Date();
    if (sampleData[_sampleIndex]) {
      return [sampleData[_sampleIndex++]];
    }
  }
}

var SamplesApi = {

  fetch: function() {
    var deferred = Q.defer();
    var nextSample = _getNextSample();

    if (nextSample) {
      deferred.resolve({
        samples: nextSample
      });
    } else {
      deferred.reject(new Error('No sample available'));
    }

    return deferred.promise;
  }

};
```

Sample data is precomputed vs. computed on the fly (e.g. $filter).

```
function Sample(id, time, activeAlert, readings) {
  this.id = id;
  this.time = new Date(time);
  var hours = this.time.getHours().toString();
  if (hours.length == 1) {
    hours = '0' + hours;
  }
  var minutes = this.time.getMinutes().toString();
  if (minutes.length == 1) {
    minutes = '0' + minutes;
  }
  this.clockTime = [hours, minutes].join(':');
  this.activeAlert = activeAlert;
  this.readings = readings;
  this.total = _.reduce(this.readings, function(memo, value) {
    return memo + value;
  });
  this.state = 'default';
  if (this.total > WARNING_THRESHOLD && this.total <= ALERT_THRESHOLD) {
    this.state = 'warning';
  } else if (this.total > ALERT_THRESHOLD) {
    this.state = 'danger';
  }
}
```

SampleStore provides an interface that the view layer can use to access sample-related application state.

```
hasNextPage: _hasNextPage,

hasPreviousPage: _hasPreviousPage,

getSamples: function() {
  var index = _page * SAMPLES_PER_PAGE;
  var endIndex = (_samples.length > index + SAMPLES_PER_PAGE) ? index + SAMPLES_
  return _samples.slice(index, endIndex);
},

getLatestSample: _getLatestSample
```

AlertStore data is SampleStore's data,
represented with new meaning.

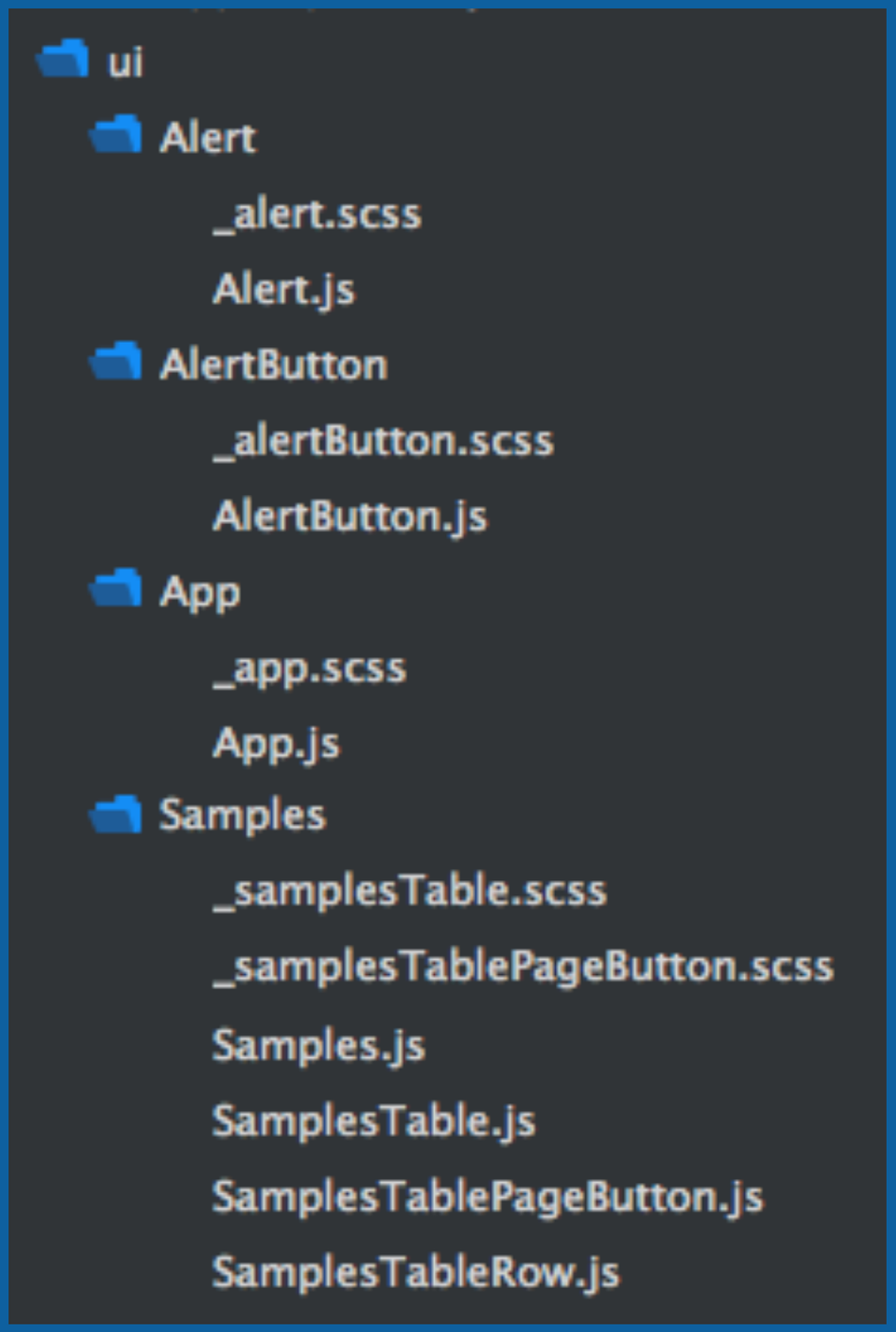
isAlert and isAlertable state is derived from the
latest sample.

```
isAlert: function() {
  return SampleStore.getLatestSample().activeAlert;
},

isAlertable: function() {
  if (this.isAlert()) {
    return false;
  }
  var latestSample = SampleStore.getLatestSample();
  if (!latestSample.state || latestSample.state == 'default') {
    return false;
  }
  return true;
}
```

View layer

Styles, “markup”, and logic grouped per component.

Components are project-specific. (We can generalize when the need arises.)



App.js as entry point and hub.

- Polls API
- Links view state to application state

```
componentDidMount: function() {
  AlertStore.addChangeListener(this.onChange);
  SampleStore.addChangeListener(this.onChange);
  // Poll for new samples.
  pollToken = setInterval(function() {
    SampleActions.fetchSamples();
  }, 1000);
  // Get first batch of samples immediately.
  SampleActions.fetchSamples();
},

componentWillUnmount: function() {
  SampleStore.removeChangeListener(this.onChange);
  clearInterval(pollToken);
},

render: function() {
  return (
    <div>
      <Alert
        isVisible={this.state.isAlert}
      />
      <Samples
        samples={this.state.samples}
        hasNextPage={this.state.hasNextPage}
        hasPreviousPage={this.state.hasPreviousPage}
      />
      <AlertButton
        isActive={this.state.isAlertable}
        onClick={this.onAlertButtonClick}
      />
    </div>
  );
},

onAlertButtonClick: function() {
  AlertActions.activateAlert();
}
```

UI components are composed of other components.

Result: small, focused components.

```
render: function() {
  return (
    <div>
      <SamplesTablePageButton
        label="view earlier samples"
        isActive={this.props.hasPreviousPage}
        onClick={this.onClickPreviousPage}
      />
      <SamplesTable
        samples={this.props.samples}
      />
      <SamplesTablePageButton
        label="view later samples"
        isActive={this.props.hasNextPage}
        onClick={this.onClickNextPage}
        isPointedDown={true}
      />
    </div>
  );
},

onClickNextPage: function() {
  SampleActions.nextPage();
},

onClickPreviousPage: function() {
  SampleActions.previousPage();
}
```

Samples.js

```
render: function() {
  var rows = this.props.samples.map(function(sample) {
    return (
      <SamplesTableRow
        sample={sample}
        key={sample.id}
      />
    )
  })
  return (
    <table className="wq-samples-table">
      <thead>
        <tr>
          <th classNam="wq-samples-table-header wq-samples-table-header--prim
          <th classNam="wq-samples-table-header" title="chloroform">CHCl<sub>
          <th classNam="wq-samples-table-header" title="bromoform">CHBr<sub>3
          <th classNam="wq-samples-table-header" title="bromodichloromethane"
          <th classNam="wq-samples-table-header" title="dibromochloromethane"
          <th classNam="wq-samples-table-header"></th>
        </tr>
      </thead>
      <tbody>
        {rows}
      </tbody>
    </table>
  );
}
```

SamplesTable.js

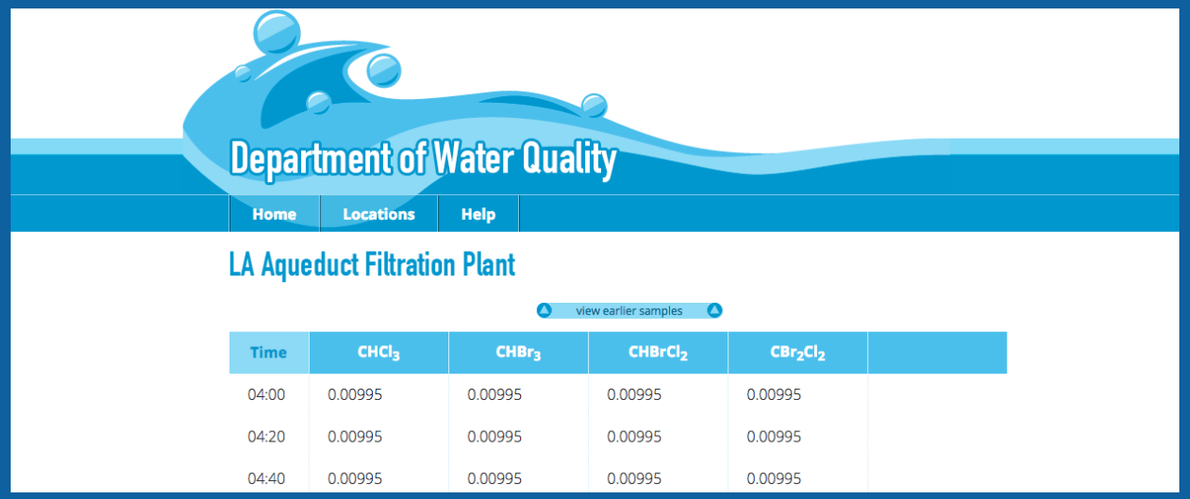
Styles:

- Namespaced classes
- BEMish syntax
- Avoid specificity wars
- Use flexbox (IE10+)

```scss
.wq-alert {
  display: none;
  align-items: center;
  background-color: #fbb9b9;
  border-radius: 8px;
  padding: 20px;
  margin-bottom: 24px;

  &.is-alert-visible {
    display: flex;
  }
}

.wq-alert__icon {
  @include bg-image('alert', 64px, 58px);
}

.wq-alert__label {
  color: #9a3333;
  font-size: 25px;
  font-weight: 700;
  margin-left: 50px;
}
```

Retro: UI tradeoffs

Centered + fixed width instead of responsive



Images w/ alt attribute instead of DIN webfont

Retro

- More domain modeling up-front (accidentally modeled after “Readings” not “Samples”).
- Rename Sample “state” property to “level”.
- DIN or alternate webfont instead of images.
- Tests + more comments!

LA Aqueduct Filtration Plant Web App

Technical overview

Product requirements
Tech stack
Domain definition
Application layer
View layer
Retrospective

Thank you!

CJ Cenizal