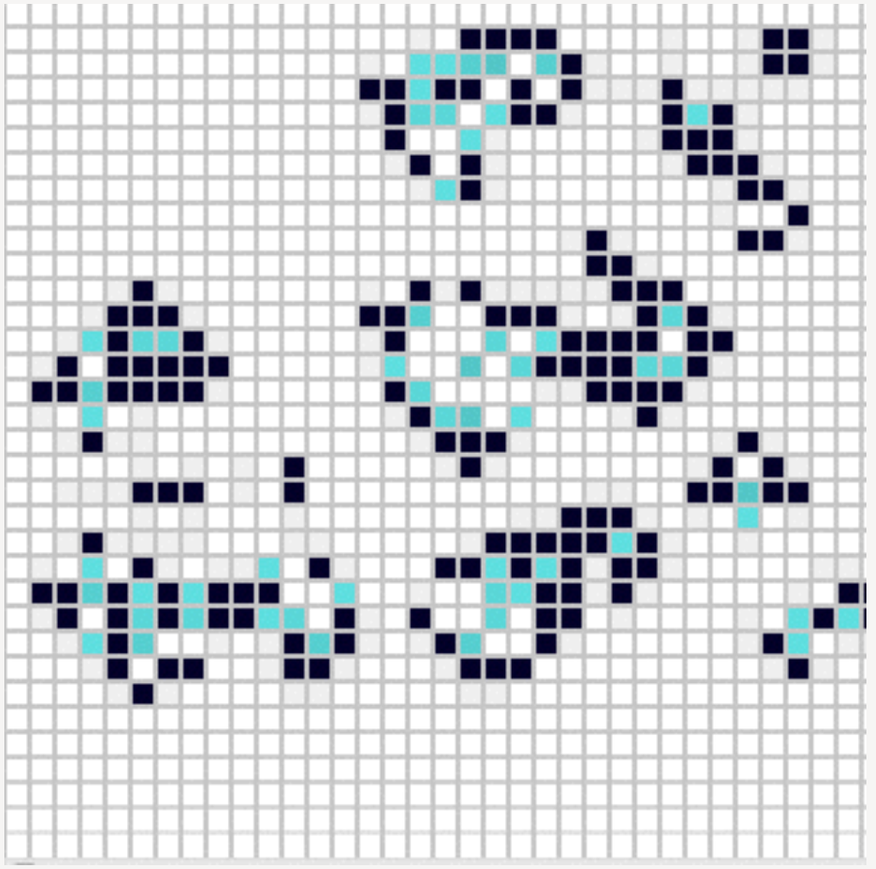
HYPERGRID

by openfin™

	last	first	birth date	income	travel
66	Columbo	Olivia	1993-08-11	$96,645.62	€298,08
67	Soprano	Moe	09/16/19	$81,213.85	€301,25
68	Columbo	Stanley			99,39
69	Luciano	Ethan			19,80
70	Soprano	Mason			88,31
71	Oneil	Boy			39,67
72	Smith	Harpo			73,61
73	Doerre	Sophia			08,15
74	Gotti	Isabella			74,12
75	Doerre	Chico			87,49
76	Luciano	Liam			04,02
77	Doerre	Larry	1981-08-11	$41,887.07	€398,04
78	Gotti	Curly	1980-01-14	$10,724.25	€460,41
79	Gotti	Ava	1959-03-25	$60,095.85	€330,02
80	Barbarosa	Curly	1997-01-04	$82,873.92	€416,23
81	Gotti	Mason	1970-07-08	$30,279.77	€877,33
82	Barbarosa	Ava	1950-03-21	$55,866.98	€939,23
83	Doerre	Stanley	1929-05-07	$98,431.46	€76,13
84	Gotti	Sophia	1960-06-18	$85,623.36	€161,38
85	Gotti	Groucho	1966-06-13	$25,121.39	€957,38
86	Smith	Hardy	1970-12-22	$7,394.16	€156,96
87	Oneil	Hardy	1909-08-17	$78,303.60	€959,92
88	Barbarosa	Moe	1954-09-09	$9,087.22	€34,75
89	Oneil	Stanley	1930-08-11	$62,177.73	€198,80
90	Luciano	Logan	1982-07-24	$95,562.49	€587,59
91	Columbo	Liam	1957-09-08	$27,932.99	€85,61
92	Soprano	Stanley	1903-09-03	$67,128.31	€303,15
93	Doerre	Liam	1999-08-24	$85,873.52	€490,99
94	Oneil	Larry	1963-10-25	$5,275.92	€908,77
95	Doerre	Noah	1949-09-21	$91,970.20	€500,58
96	Barbarosa	Shemp	1956-08-25	$10,951.14	€36,19

September 1921

Sun	Mon	Tue	Wed	Thu	Fri	Sat
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1



	X	D	C -	
1			43	0.00377
2			33	0.00377
3			58	0.00377
4			86	0.00377
5				0.00377
6			84	0.00377
7			44	0.00377
8			19	0.00377
9			44	0.00377
10			54	0.00377
11			40	0.00377
12			61	0.00377
13			89	0.00377
14			63	0.00377

openfin™

WHY HYPERGRID?

- People want to move to HTML5, there are no world class open source options
- Finance industry/big data lacks a well supported native feel HTML5/JavaScript grid control

openfin™

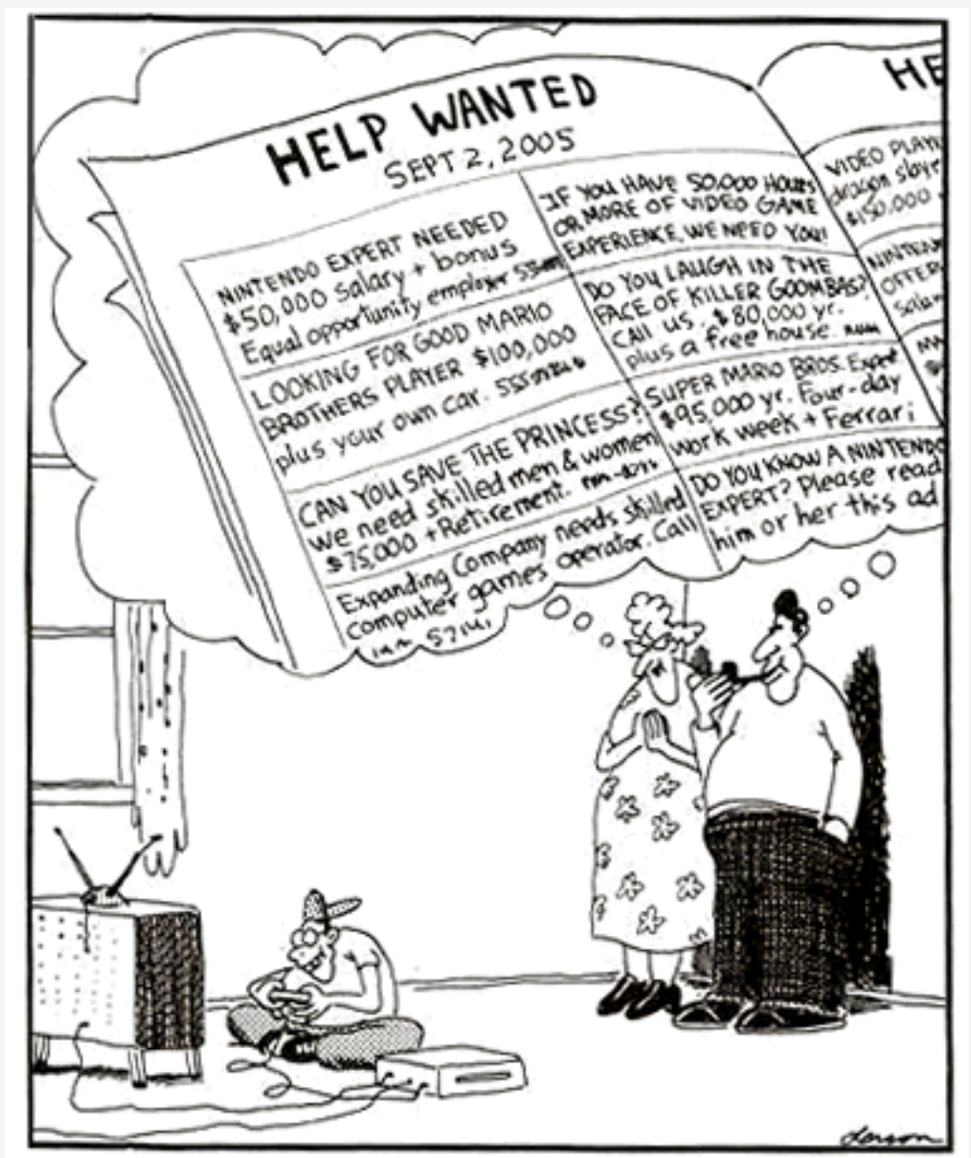
WHEN HYPERGRID?

- Big Data – Millions+ rows
- Dynamic Data – Thousands of rows Thousands of updates/sec
- Maybe not – Small tables, nested tables, lots of css styling – anything HTML tables are already good at…

openfin™

ABOUT ME

- Engineer at OpenFin
- Smalltalk/Java/k-Q-kdb+/javascript/html5



Hopeful parents

openfin™

GOALS FOR THE HYPERGRID

openfin™

HYPERGRID GOALS

- super fast and efficient
- look and behave great
- super easy to use

openfin™

HYPERGRID GOALS

- no deferred rendering
- easy to integrate with your data

openfin™

HYPERGRID GOALS

- good general purpose grid but best for finance / big data
- allow for easy virtualization

openfin™

HYPERGRID GOALS

- use HTML5 / JavaScript best practices
- open-source

openfin™

HYPERGRID GOALS

Initial features to be driven by existing HTML5 grids:

slickgrid:	http://github.com/mleibman/SlickGrid
kendo ui grid:	http://demos.telerik.com/kendo-ui/grid/index
ext-livegrid:	http://www.ext-livegrid.com/
Flexigrid:	http://flexigrid.info/
jQuery Grid:	http://www.trirand.com/blog/
jqGridView:	http://plugins.jquery.com/project/jqGridView
jqxGrid:	http://www.jqwidgets.com/
Ingrid:	http://reconstrukt.com/ingrid/
DataTables:	http://www.datatables.net/index

WHAT IS POLYMER?

- web components polyfill
- best practices helper code
- widget set(s)

openfin™

WHAT IS POLYMER?

- OpenFin believes it's the future
- you do **not** need Polymer to use Hypergrid

openfin™

SOME HYPERGRID FEATURES

THE “PINK” DEMO

openfin™

CURRENTLY MINIMAL DATA MANAGEMENT

INTERNAL AND EXTERNAL

MORE COMING...

```
function swap(arr, i, j) {
    var temp = arr[i];
    arr[i] = arr[j];
    arr[j] = temp;
}

function dualPivotQuicksort(property, arr, lef
    var len = right - left;

    if (len < 27) { // insertion sort for tiny
        for (var i = left + 1; i <= right; i++
            for (var j = i; j > left && arr[j]
                swap(arr, j, j - 1);
            }
        }
        return;
    }
    var third = Math.floor(len / div); //TODO:

    // "medians"
    var m1 = left  + third;
    var m2 = right - third;

    if (m1 <= left) {
        m1 = left + 1;
    }
    if (m2 >= right) {
        m2 = right - 1;
    }
    if (arr[m1][property] < arr[m2][property])
        swap(arr, m1, left);
        swap(arr, m2, right);
    }
```

openfin™

EASY CUSTOM CELL RENDERING

```
//simple implementation of a sparkline.  see [Edward Tufte spark
edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=0001OR)
paintSparkbar: function(ctx, x, y, width, height) {
    var val = this.config.value;
    if (!val || !val.length) {
        return;
    }
    var count = val.length;
    var eWidth = width / count;
    var bgColor = this.config.isSelected ? this.config.bgSelColo
    ctx.fillStyle = bgColor;
    ctx.fillRect(x, y, width, height);
    ctx.fillStyle = '#010167';
    for (var i = 0; i < val.length; i++) {
        var barheight = val[i] / 110 * height;
        ctx.fillRect(x + 5, y + height - barheight, eWidth * 0.6
        x = x + eWidth;
    }
},

//simple implementation of a sparkline, because it's a barchart
).  see [Edward Tufte sparkline](http://www.edwardtufte.com/bboa
msg?msg_id=0001OR)
paintSparkline: function(ctx, x, y, width, height) {
    var val = this.config.value;
    if (!val || !val.length) {
        return;
    }
    var count = val.length;
    var eWidth = width / count;
    var bgColor = this.config.isSelected ? this.config.bgSelColo
    ctx.fillStyle = bgColor;
    ctx.fillRect(x, y, width, height);
    ctx.strokeStyle = '#010167';
    ctx.fillStyle = '#010167';
    ctx.beginPath();
    var prev;
    for (var i = 0; i < val.length; i++) {
        var barheight = val[i] / 110 * height;
        if (!prev) {
            prev = barheight;
        }
        ctx.lineTo(x + 5, y + height - barheight);
        ctx.arc(x + 5, y + height - barheight, 1, 0, 2 * Math.PI
        x = x + eWidth;
```

openfin™

POLYMER PLUGIN PATTERN

- hypergrid developer
- application developer
- installOn(hypergrid)

"install_InTo(hypergrid)"

```
<fin-hypergrid class="abs">
    <fin-hypergrid-behavior-in-memory></fin-hypergrid-behavior-in-memory>
    <fin-hypergrid-excel></fin-hypergrid-excel>
</fin-hypergrid>
```

openfin™

SINGLE FILE DEPLOYMENT

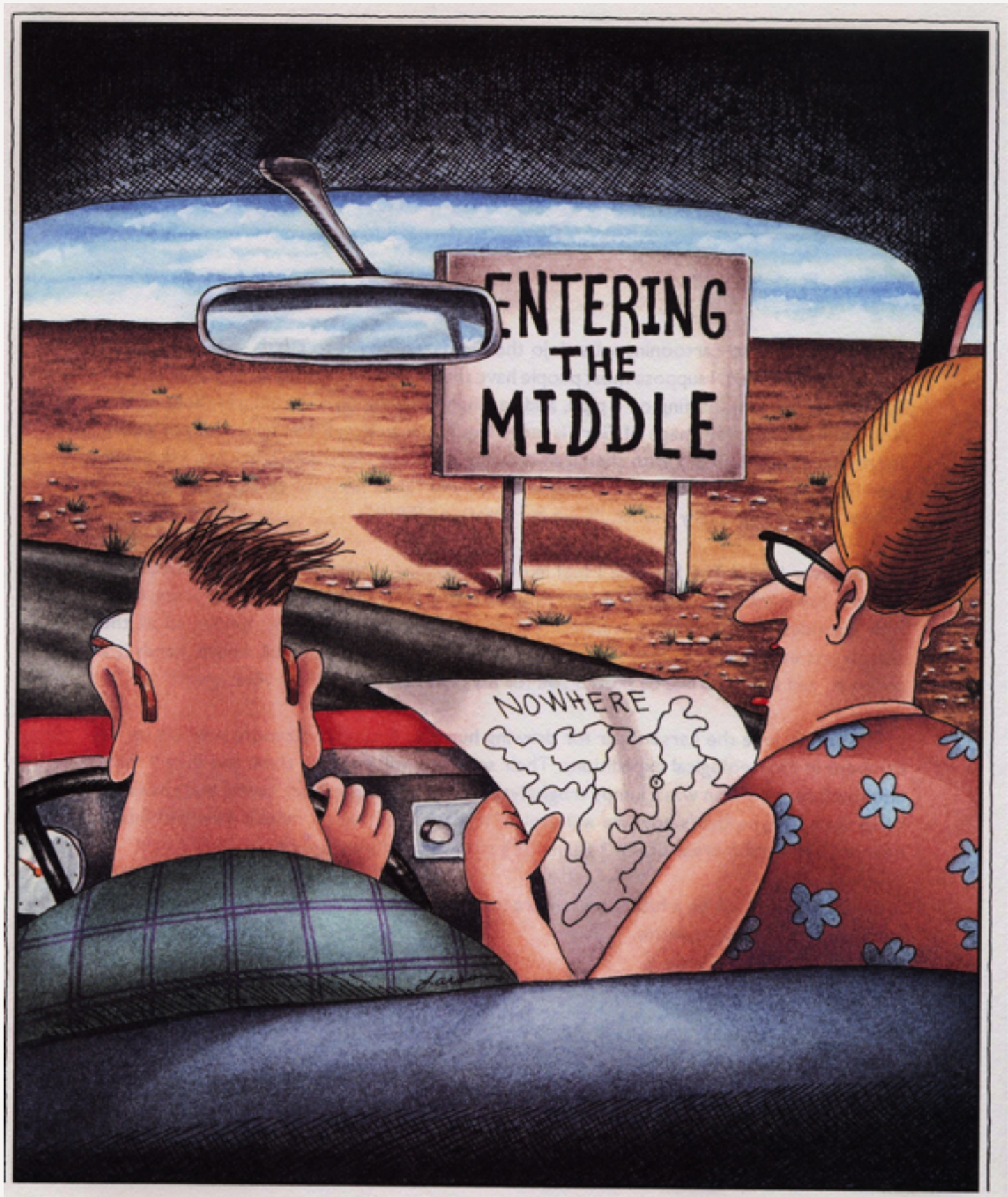
```
<!doctype html>
<html>
<head>
  <meta name="viewport" content="width=device-width, minimum-scale=1.0, initial-scal
  <title>fin-hypergrid JSON Standalone Demo</title>
  <script src="bower_components/accountingjs/accounting.min.js"></script>
  <script src="bower_components/webcomponentsjs/webcomponents.js"></script>
  <link rel="import" href="fin-hypergrid.min.html">

  <style>
    fin-hypergrid {
      position: absolute;
      top: 0;
      right: 0;
      bottom: 0;
      left: 0;
      overflow: hidden;
    }
  </style>
</head>
<body unresolved fullbleed>


      <fin-hypergrid class="abs" id="json-example">
        <fin-hypergrid-behavior-json></fin-hypergrid-behavior-json>
      </fin-hypergrid>
```

openfin™

JSON SUPPORT

```
','textfield','textfield'];

document.addEventListener('polymer-ready', function() {

    //get ahold of our json grid example
    var jsonGrid = document.querySelector('#json-example');

    //get it's table model
    var jsonModel = jsonGrid.getBehavior();

    //get the cell cellProvider for altering cell renderers
    var cellProvider = jsonModel.getCellProvider();

    //set the header labels
    jsonModel.setHeaders(headers);

    //set the fields found on the row objects
    jsonModel.setFields(fields);

    //set the actual json row objects
    jsonModel.setData(data);
```

HALFWAY!



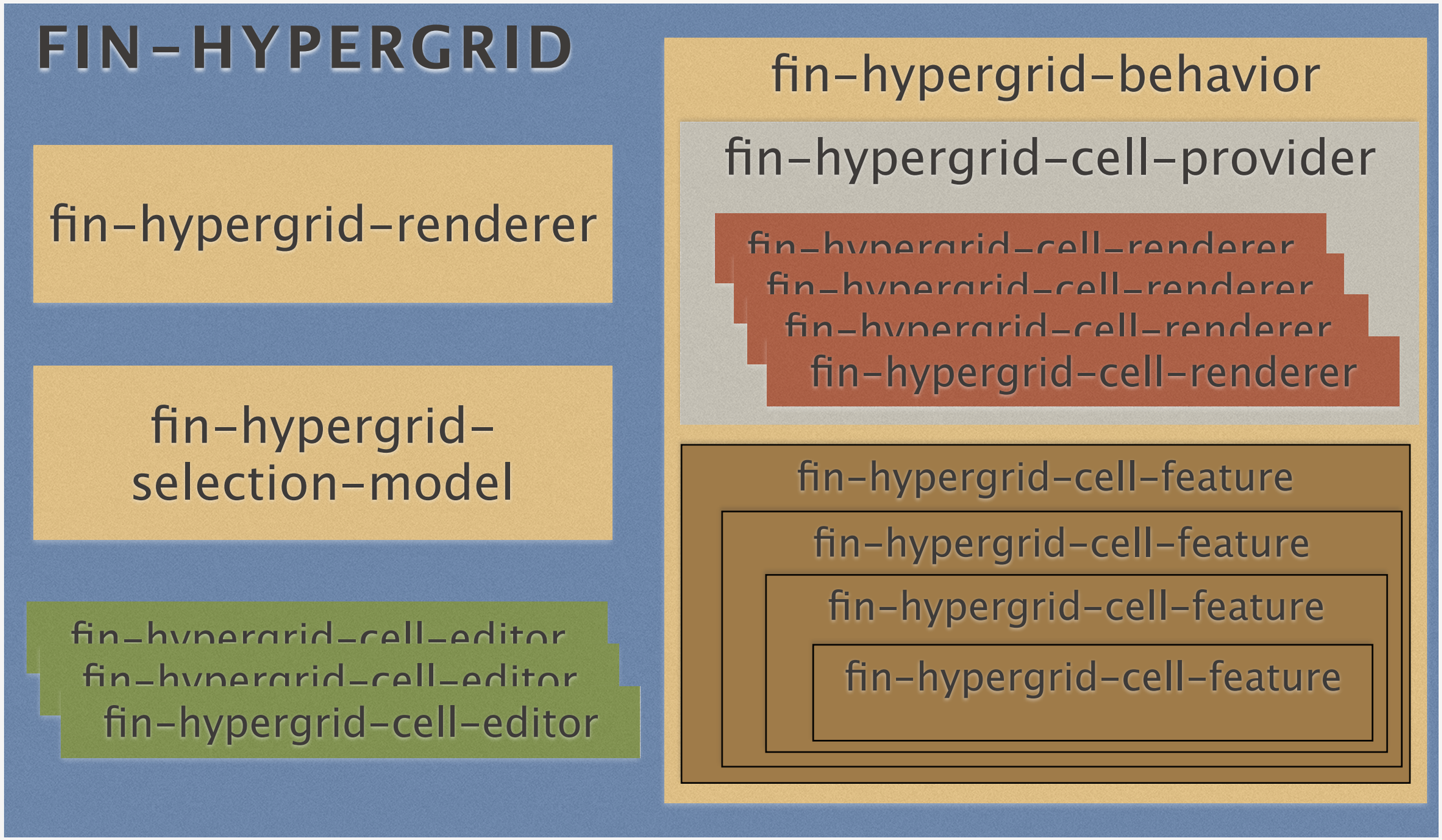
"Well, this is just going from bad to worse."

openfin™

QUICK HYPERGRID DESIGN

openfin™

HYPERGRID DESIGN

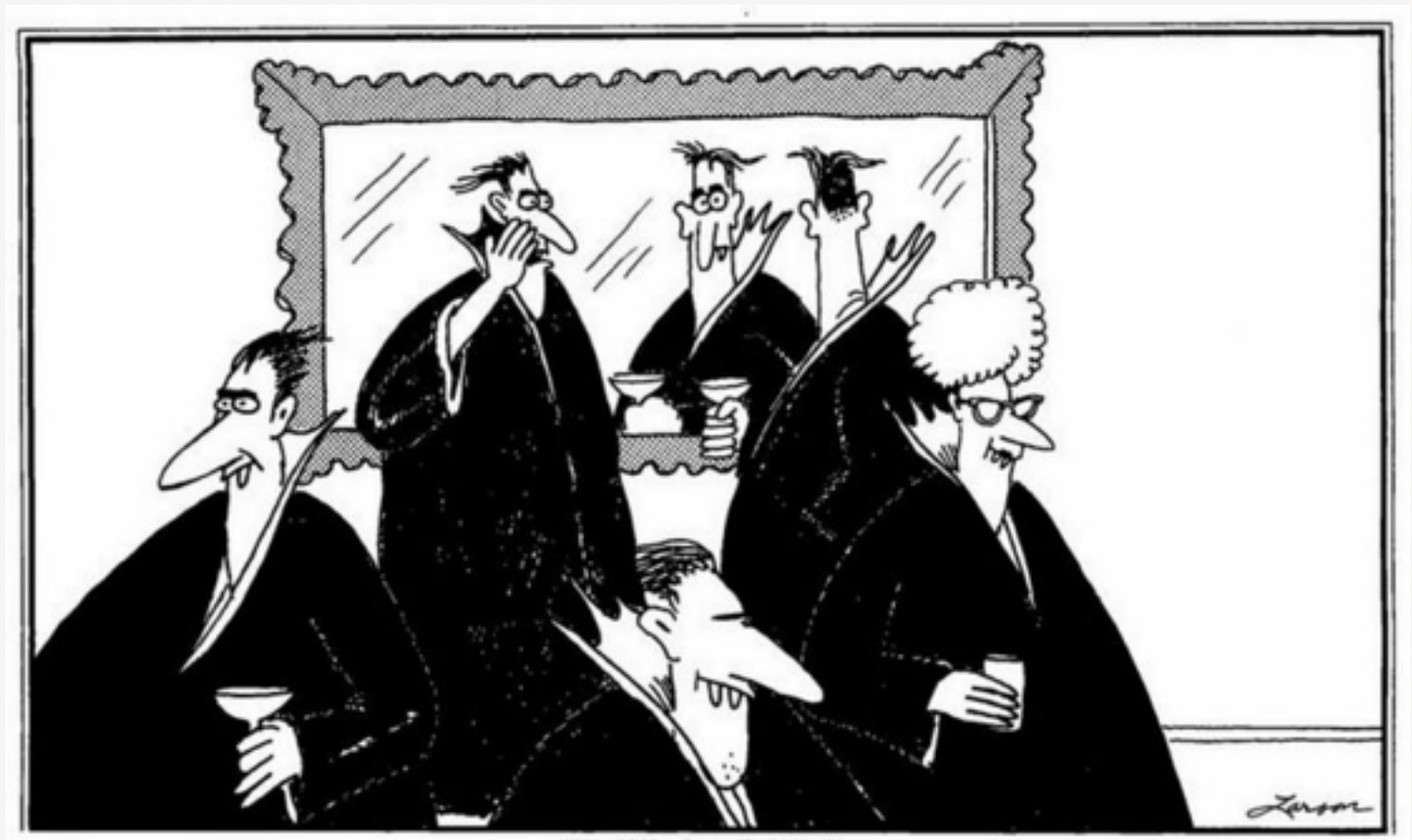


openfin™

HYPERGRID DESIGN

DEPENDENCIES

- fin-vampire-bar (scrollbars)
- fin-canvas
- fin-rectangles


"Hank! You're reflecting!"

openfin™

TECHNIQUES

(LESSONS LEARNED)

openfin™

TECHNIQUES

- common sense
- use techs found in grid/table controls from other languages
- row/column based scrolling

TECHNIQUES

- talking to graphics context is expensive; know the precedence
- only iterate/render what is visible

openfin™

TECHNIQUES

- great book “HTML5 game design insights”
- requestAnimationFrame
- know about device pixel density and backing store

openfin™

TECHNIQUES

- rectangle based selection model
- stable sorts for sort accumulation
- polymer plugin pattern

openfin™

TECHNIQUES (GOTCHAS)

- polymer inheritance issues
- border rendering, background fill per cell extremely expensive

PROCESS AND BEST PRACTICES

openfin™

PROCESS AND BEST PRACTICES

- environment was tricky
- restarted three times
- browserify, webcomponents, polymer

openfin™

PROCESS AND BEST PRACTICES

- grunt based
- livereload (added)
- vulcanize (added)
- wct-testing (modified)

PROCESS AND BEST PRACTICES

- aggressive validation
- polymer stack solutions

1. documentation
2. demos

openfin™

PROCESS AND BEST PRACTICES

- yeoman generator for polymer
- yeoman generator of my process
- https://github.com/stevewirts/generator-fin-polymer

ROADMAP

openfin™

ROADMAP

- bug fixes
- more tests
- more/better documentation
- column autosizing
- column picker
- scrollbar fixes
- grouping
- filters
- pagination
- hierarchical capabilities
- navigation
- row dnd/resizing

openfin™

ROADMAP

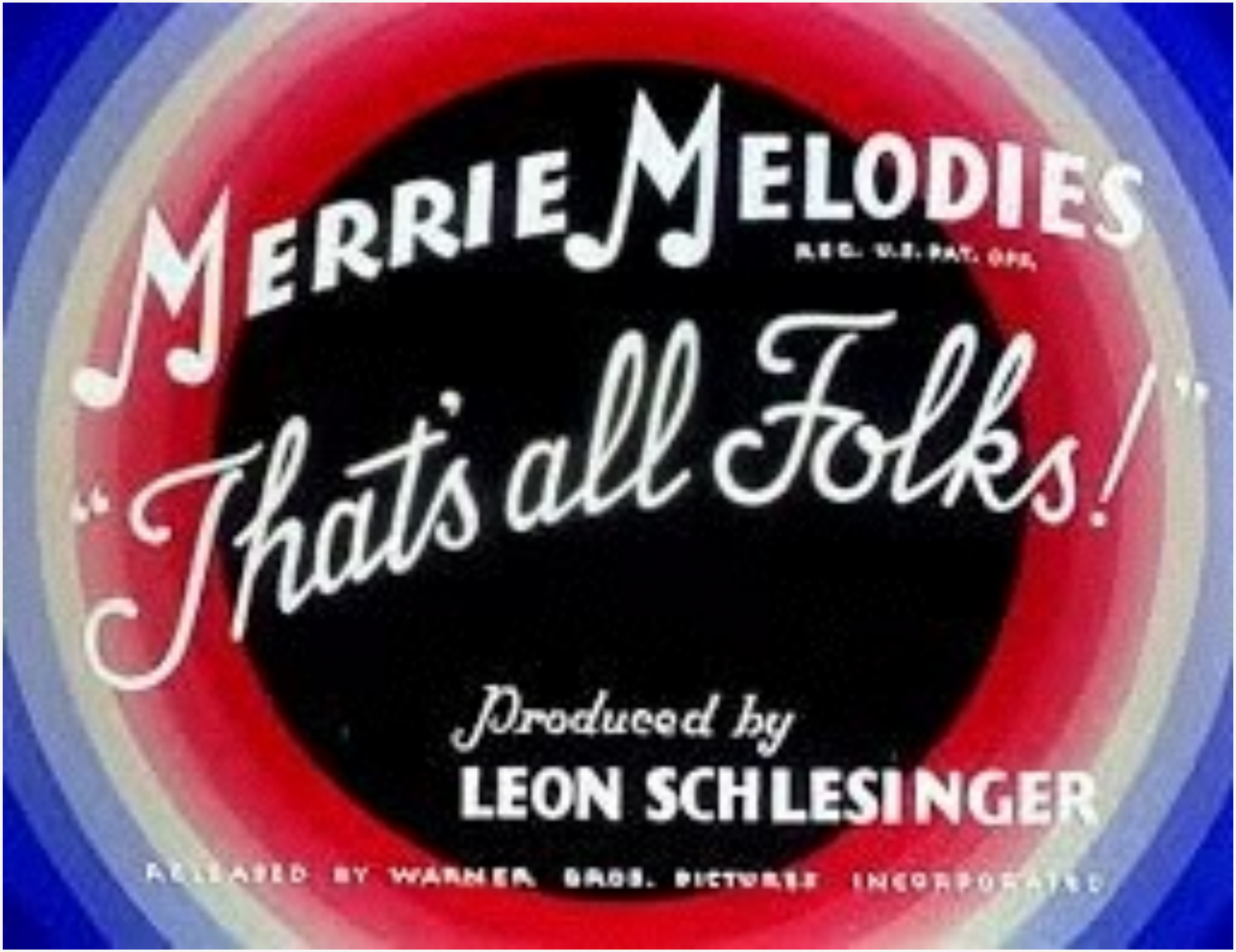
- more cell editors
- performance tuning
- better mobile experience
- better build process
- better dev process
- more examples
- enum columns
- damage rectangles
- more renderers
- excel-esque features
- better internal data management behavior
- ???????

openfin™

SUMMARY

- We have already put an enormous amount of effort into Hypergrid
- OpenFin is committed to seeing Hypergrid become the best html5 grid control available

openfin™


MERRIE MELODIES
REG. U.S. PAT. OFF.
"That's all Folks!"
Produced by
LEON SCHLESINGER
RELEASED BY WARNER BROS. PICTURES INCORPORATED