

Web pods

Accessing Control Data Through Web Standards (WebSockets, JSON, HTML, CSS)

WebPODS:

Web Protocol Oriented Distribution Service

Broadcasts real-time data aggregated by diirt, using WebSockets and JSON.

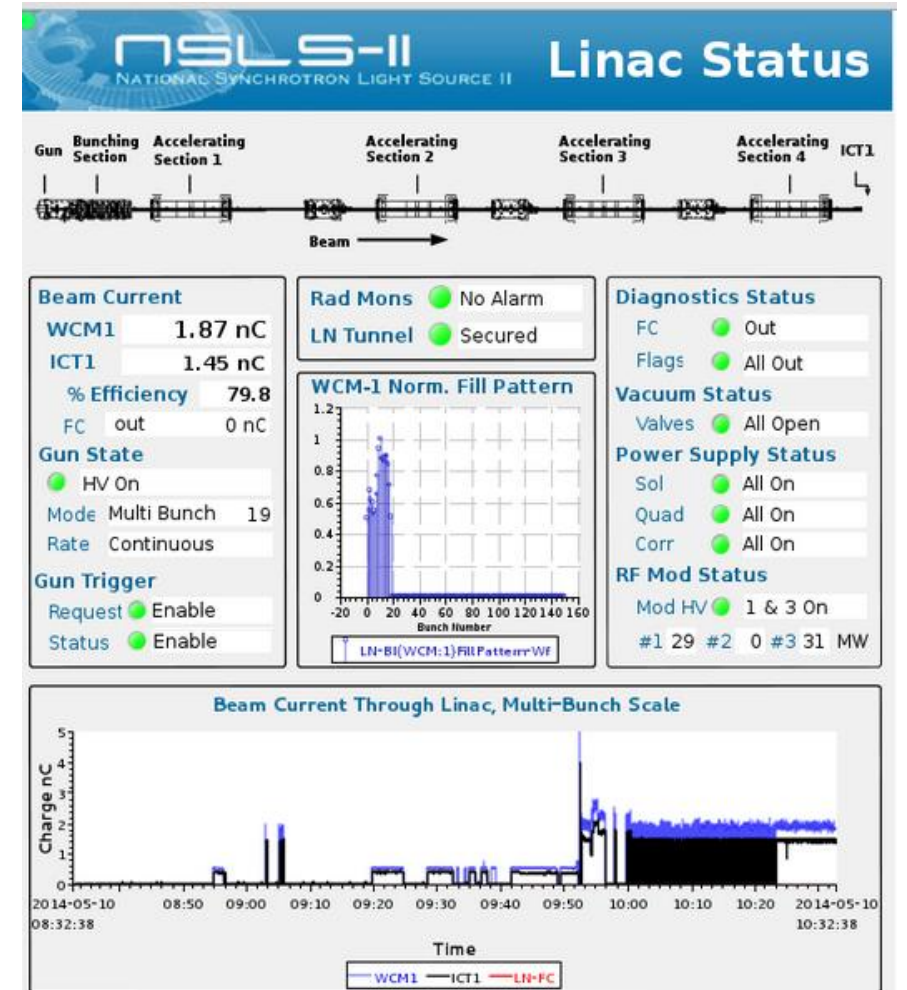
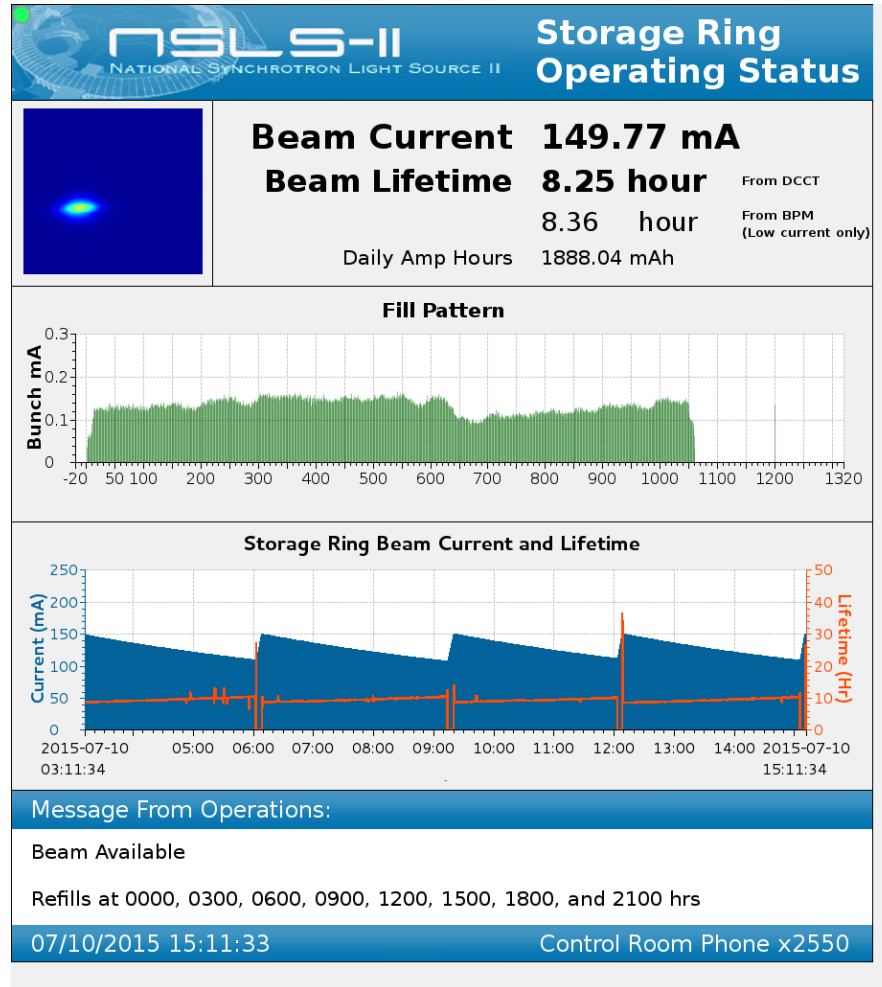
[Test with probe](#)[Widget examples](#)[User documentation](#)[Developer documentation](#)

Known Issues: with glassfish 4.0, the error handling is not up to spec due to some bugs in the bundled version of tyrus. Either update the WebSocket implementation or update glassfish to 4.1.

WebPODS when installed

Motivation

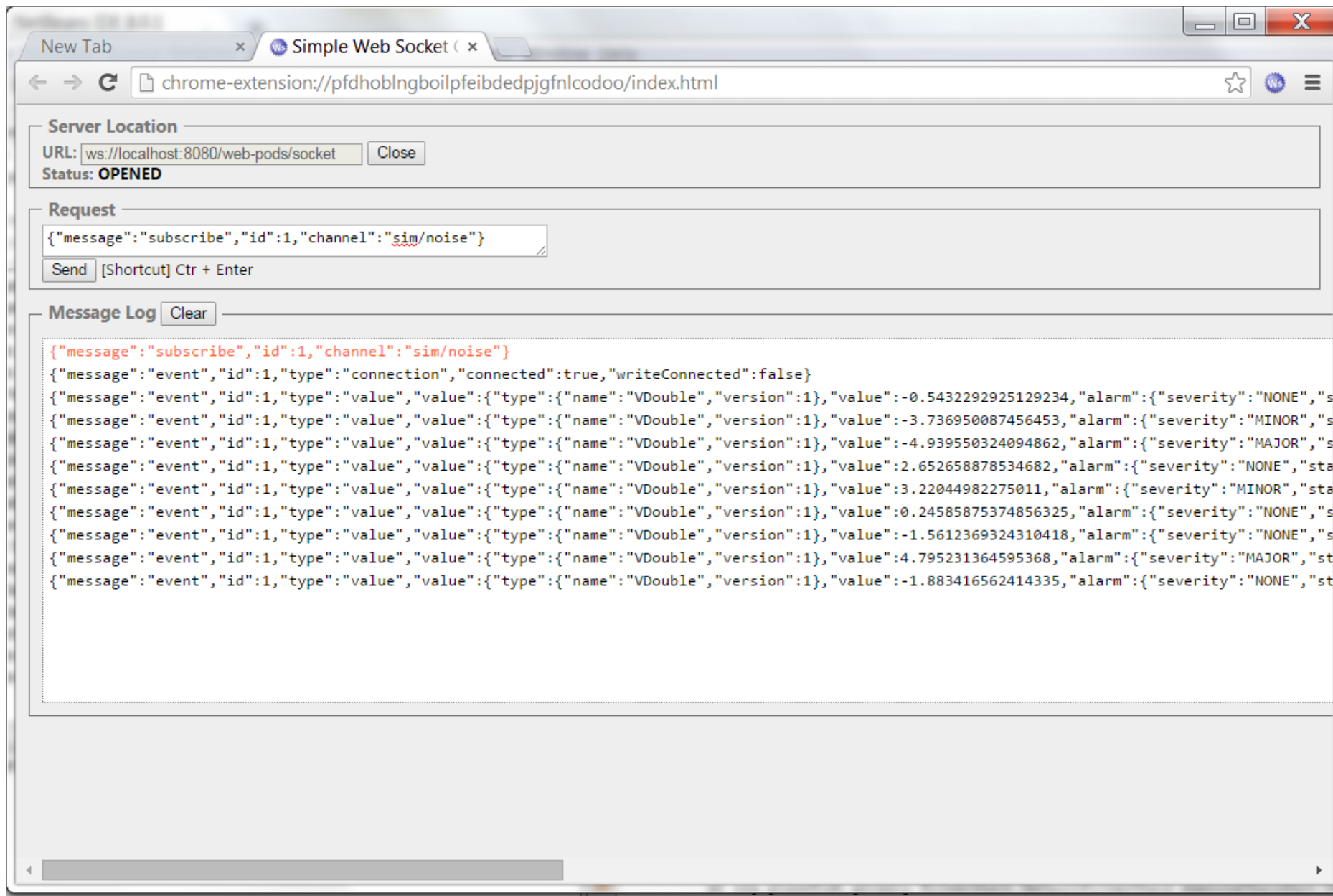
Create web pages to display control system data



WebPODS

(Protocol Oriented Distribution Service)

- The Web PODS aims to provide real-time publish/subscribe communication using
 - WebSockets
 - JSON
 - Pure HTML/CSS widgets
- Main benefits:
 - Uses web standards (any web client in any language)
 - Widgets can be configured through standard CSS
 - Data access outside of control network (possibly WAN)
 - Web pages do not need to live on the same server where the gateway is



Chrome extension: Simple Web Socket Client

Socket Info

CONNECT

DISCONNECT

Subscription Info

SUBSCRIBE

Current Subscriptions

id: 0, channel: sim/noise

id: 1, channel: sim/ramp

PAUSE

RESUME

UNSUBSCRIBE

FILTER

SHOW ALL

Events

CLEAR

-4.674640598376064

0

-0.3762008265139336

-1

3.7166769654591096

-2

3.6610547793450507

?

```
{
  "message": "event",
  "id": 0,
  "type": "value",
  "value": {
    "type": {
      "name": "VDouble",
      "version": 1
    },
    "value": -4.674640598376064,
    "alarm": {
      "severity": "MAJOR",
```

Connected to: ws://localhost:8080/web-pods/socket

ws://localhost:8080/web-pods/socket sim/noise

CONNECT PAUSE CLOSE

```
{
  "message": "event",
  "id": 0,
  "type": "value",
  "value": {
    "type": {
      "name": "VDouble",
      "version": 1
    },
    "value": -1.5941866134048608,
    "alarm": {
      "severity": "NONE",
      "status": "NONE"
    },
    "time": {
      "unixSec": 1413834500,
      "nanoSec": 265582126,
      "userTag": null
    },
    "display": {
      "lowAlarm": -4,
      "highAlarm": 4,
      "lowDisplay": -5,
      "highDisplay": 5,
      "lowWarning": -3,
      "highWarning": 5,
      "units": "x"
    }
  }
}
```

Test javascript client (Danielle Connolly, UMich)

Read-only text field for a channel (wp-text-monitor)

See the source of this page for examples on how to use the widget. For more details, see [the WebPODS widgets reference documentation](#).

Simple monitoring

These simple examples just add the channel, using the data-channel property.

sim/ramp:

sim/noise:

sim/strings:

sim/table:

Error conditions

Behavior under different error conditions.

Slow connection:

Error messages:

Wrong formula:

Changing size

Using CSS style width and height to change size.

width: 50px; height: 18px -

width: 100px; height: 22px -

width: 150px; height: 26px -

width: 200px; height: 30px -

Changing fonts

Using CSS style font to change the font size or type. Note that, by default, the font is inherited.

font-family: monospace -

font-family: serif -

font: italic bold 18px Georgia, serif -

Changing text alignment

Using CSS style text-align to change text alignment in the field.

text-align: left -

text-align: center -

text-align: right -

Positioning

Using CSS style margin or float to adjust widget position.

margin-left: 50px -

float: right -

Text monitor examples

Read-only LED widget (wp-led)

See the source of this page for examples on how to use the widget. For more details, see [the WebPODS widgets reference documentation](#).

Simple monitoring

These simple examples just add the channel, using the data-channel property.

sim/ramp:

sim/flipflop(1.0):

=sim/noise!>0.0:

sim/table:

Multi-state

For enum, the LED can display all states.

=alarmOf('sim/ramp'):

=enumOf('sim/noise', arrayOf(-1.0, 1.0), arrayOf("LOW", "MIDDLE", "HIGH")):

Error conditions

Behavior under different error conditions.

Slow connection:

Error messages:

Wrong formula:

Changing size

Using CSS style width and height to change size.

width: 5px; height: 18px -

width: 10px; height: 10px -

width: 15px; height: 15px -

width: 20px; height: 20px -

Changing fonts

By default, the size of the LED changes with the font.

font-family: monospace -

font-family: serif -

font: italic bold 18px Georgia, serif -

font: 14px monospace -

Positioning

Using CSS style margin or float to adjust widget position.

margin-left: 50px -

float: right -

vertical-align: bottom -

vertical-align: top -

WebPODS

Specify server location

```
<script>
  var wsUri = "ws://diirt.aglt2.org/web-pods/socket";
  var wp = new Client(wsUri);
</script>
```

Adding text-monitor

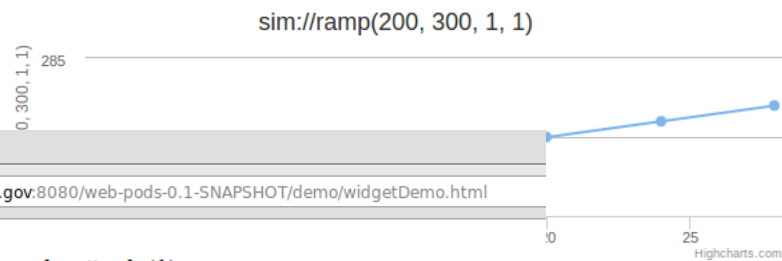
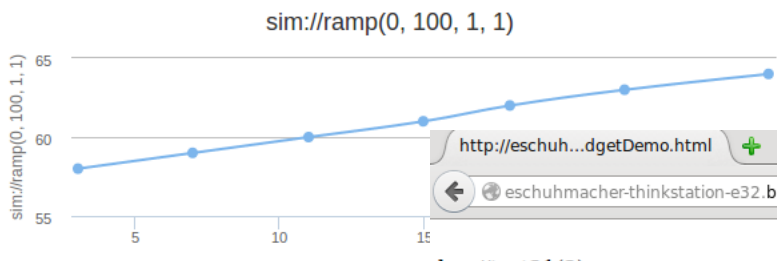
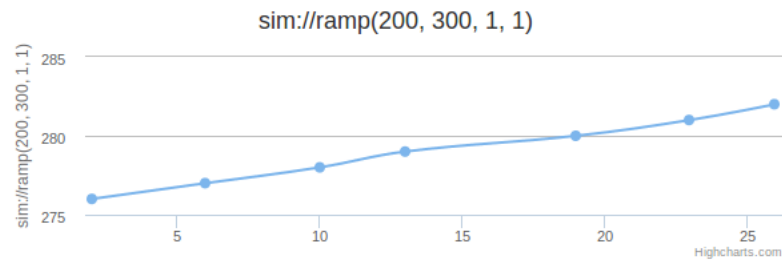
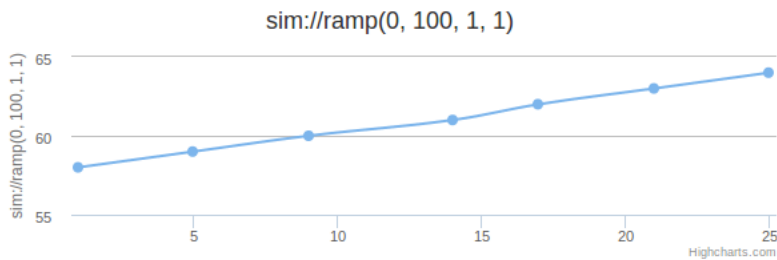
```
<p>sim/ramp: <span class="wp-text-monitor" data-channel="sim/ramp"></span></p>
<p>sim/noise: <span class="wp-text-monitor" data-channel="sim/noise"></span></p>
<p>sim/string: <span class="wp-text-monitor" data-channel="sim/string"></span></p>
```

Adding led

```
<p>sim/ramp: <span class="wp-led" data-channel="sim/ramp"></span></p>
<p>sim/flipflop(1.0): <span class="wp-led" data-channel="sim/flipflop(1.0)"></span></p>
<p>sim/noise: <span class="wp-led" data-channel="sim/noise"></span></p>
```

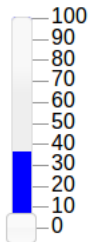
Specify color for "LOW" enum value

```
.wp-led.customvalues .value-low {
  color: LightCyan;
  content: "\f175";
}
```



loc://test31(3)

loc://asda(1)



sim://ramp(0, 100, 1, 1)

12

sim://ramp(0, 100, 1, 1)

13

CS-Studio

File Edit Search CSS Window Help

100%

CSStudio

Navigator

- CSS
 - opi-regression-tests
 - um-opi
 - .git
 - raw-data
 - condor-MSUsummary.opi
 - condor-UMsummary.opi
 - ganglia-UMWorkerNodes.opi
 - README
 - condor-correlation.opi
 - ganglia-condor-2dHistogram.opi
 - ganglia-condor-correlation.opi
 - ganglia-correlation.opi
 - ganglia-histograms.opi
 - ganglia-selectable-histograms.opi
 - test.opi
 - .project

ganglia-condor-corre... ganglia-selectable-... condor-correlation... ganglia-histograms.o...

wp://sim/noise(0,1,0.1)

PV Formula: wp://sim/noise(0,1,0.1)

Value: 0.9527329992386852 [MAJOR - HIHI]

Timestamp: 2014/10/14 09:57:54.828278260

New Value: 0.9527329992386852

Status: Connected

carcassi

Correlations from current condor and ganglia inform

X wall Y cpu

Size bytes_in Color queue

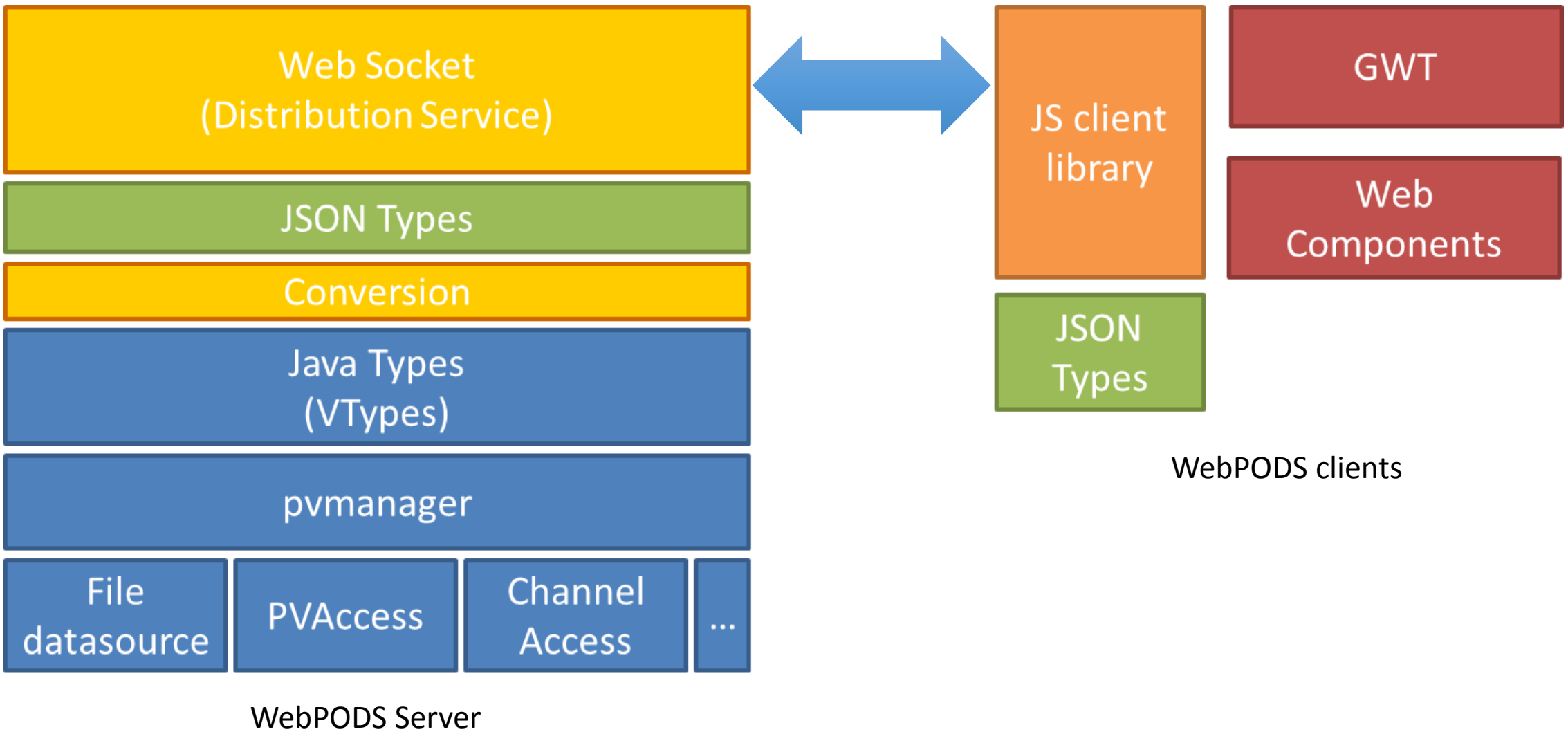
Filter

host

| | | | | | | | | | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| h... | s... | j... | G... | e... | q... | P... | t... | o... | c... | w... | f... | A... | b... | b... | b... | c... | c... | c... | c... | c... | c... | c... | d... |
| b... | 2 | 2... | u... | d... | D... | n... | n... | b... | 2... | 2... | 0... | 6 | 1... | 1... | 2... | 0... | 8... | 0... | 8... | 8... | 1... | 2... | 0... |

Using CS-Studio off site through Web Pods

WebPODS Architecture



Web Pods server configuration

```
[xxx@diirt ~]# more .diirt/pods/web/mappings.xml
<?xml version='1.0' encoding='UTF-8'?>
<mappings version="1">
  <mapping channel="calibration/constants"
substitution="file:///path/to/file/table.csv"
permission="READ_ONLY"/>
  <mapping channel="public-.*" permission="READ_WRITE"/>
  <mapping channel="cf-tag-(.\w)" substitution="=cfQuery($1)"
permission="READ_ONLY"/>
  <mapping channel="sim/(.*)" substitution="sim://$1"
permission="READ_ONLY"/>
</mappings>
```

Access security planned, not yet implemented

- Use wss (like https) for authentication
- Use username/role/unix group/host for authorization

Be careful not to expose too much

Web Pods

- Rebroadcast data using Web sockets and JSON
 - Play nice with firewalls, get WAN notifications, data available to web tools, server is one class (no logic, a pass-through to pvmanager), should scale (different sockets on different servers, not tested)
 - Not a substitute to CA/PVA. JSON serialization problems (verbose, does not follow IEEE standards for double, no “delta” updates). Not intended for heavyweight applications.
- Uses
 - Active web pages
 - WAN gateway for CS-Studio and other tools
 - Easy way to make data available to CS-Studio and other tools
 - Setup a directory shared through web-pods
 - Write csv or jvtype files with scripts or text editors (updates are sent)
- Something similar could be done with PVA (Pva Pods)
 - Already designed pieces (e.g. mapping logic) to be reusable

Web Pods

- What's done:
 - Version 1 specification done and implemented
 - Preliminary Javascript API
 - Finalized HTML/CSS compliant widgets (text-monitor, led, table, bubble-graph)
 - Being used in production at UMich for 6 months under light load. No downtime.
 - Client-side Java API
 - Client-side and server-side formulas
- What needs to be done:
 - More widgets
 - Client-side security (server side is implemented)
 - Load tests
 - Review Javascript API
 - Client-side reconnected and ping/pong
- What could be done:
 - Investigate binary protocols (for large data or pre-calculated graphs)
 - Make single widget spec for HTML and JavaFX widgets

WebPODS

- Website
 - <http://diirt.org/>
- Wiki
 - <https://github.com/diirt/diirt/wiki/WebPODS>
- Specs available on the WIKI:
 - <https://github.com/diirt/diirt/wiki/WebPods-Protocol-Specification-v1>
 - <https://github.com/diirt/diirt/wiki/JSON-serialization-for-vTypes-v1>