JavaScript for IoTivity

Sakari Poussa, Intel
@spoussa
Topics

OPEN INTERCONNECT CONSORTIUM

IoTivity

API

JS
Application developer can use only web technologies to create end-to-end solutions.

Not the only option but focus of this talk.
• Specifications
• Data models
• Certification
• Marketing

• Open Source implementation
• QA
• API bindings

No JavaScript
- JavaScript API for OIC
- Web IDL specification
- Single API – Multiple Implementations
- All open source
- Travis, sample code, demos
- IoTivity 1.0
## API Highlights

### Client

<table>
<thead>
<tr>
<th>Promise</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Promise</td>
<td>findResources()</td>
</tr>
</tbody>
</table>

**Events:**
- resourcefound

### Resource

<table>
<thead>
<tr>
<th>Promise</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Promise</td>
<td>retrieve()</td>
</tr>
<tr>
<td>Promise</td>
<td>update(data)</td>
</tr>
<tr>
<td>Promise</td>
<td>observe()</td>
</tr>
</tbody>
</table>

**Events:**
- update

### Server

<table>
<thead>
<tr>
<th>Promise&lt;resource&gt;</th>
<th>Method</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>register</td>
<td>register(data)</td>
<td></td>
</tr>
</tbody>
</table>

**Events:**
- retrieverrequest
- updaterequest
- observerrequest
Discovery

OIC Client

Request: DISCOVER

Response: URI, Type, ID

OIC Server
Retrieve, Update

**OIC Client**

Request: RETRIEVE

Response: DATA

Request: UPDATE

Response: OK

**OIC Server**
Cloud

OIC and Cloud

REST API

CoAP over TCP

XMPP
Remote Access with REST APIs

OIC Device Network (local) → API SERVER → HTTP(S) and JSON

Cloud

HTTP(S) and JSON

Remote Access with REST APIs
A diagram illustrates the interaction between a Sensor (OIC server), API Server (REST), and Cloud/Mobile entity. The processes include:

- **Discovery**: A Sensor (OIC server) makes a **GET** request to /oic/res, and the API Server responds with a URI, Type, ID:
  
  ```json
  {"href":"/a/light","rt":"core.light","if":"oc.mi.def"}
  ```

- **Read**: The Cloud/Mobile requests a **GET** for /a/light, and the API Server returns:
  
  ```json
  {"href":"/a/light","properties":{"on":false}}
  ```

- **Update**: The Cloud/Mobile updates the Sensor by making a **PUT** request for /a/light, changing the property "on" to true. The response is an **OK** status code:
  
  ```json
  {"href":"/a/light","properties":{"on":true}}
  ```
REST API SERVER

- All JavaScript
- Node.js, Express, and iotivity-node
- Open source https://github.com/01org/iot-rest-api-server
- RAML API documentation
- Features
  - OIC discovery (/oic/res and /oic/d)
  - OIC RETRIEVE and UPDATE
  - Observing
Next Steps

Security – Sandbox

Small devices – V8 is big

JS Engine API ?
Demo

Visit Intel Booth for Demo

IoTivity-node in Action

No Native Application Code

JavaScript, HTML, WebGL and CSS
Questions ?