

IoT Resource Tree Representation for Massive Data Set

Team member:

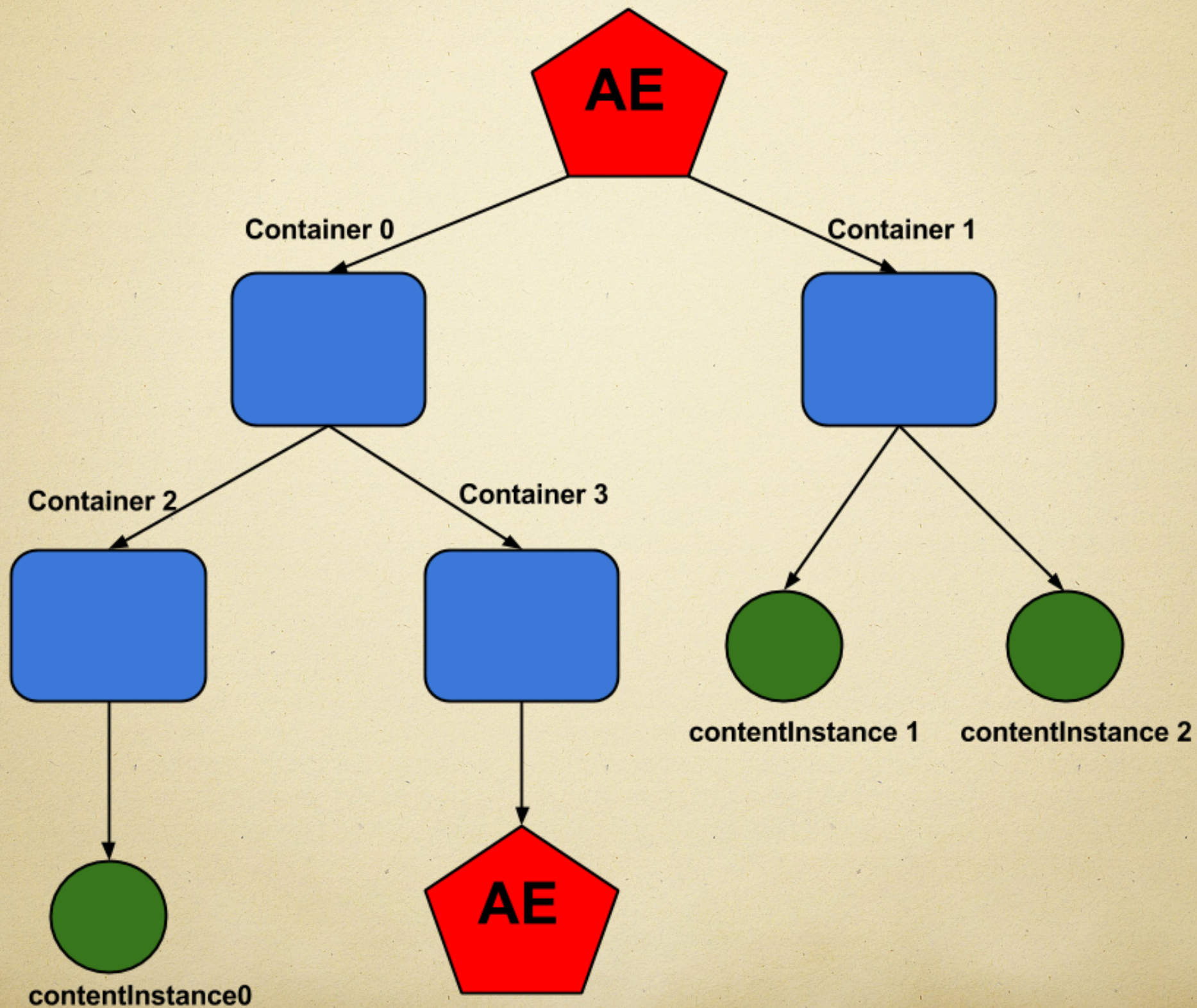
Qian Chen , Yingchao Zhu

Niklas Kunkel , Qingqing Li

Tasks Outlined for Sprint 3



Recursive Method



Reformatting JSON

```
urceName","attributeValue":"contentInstance4"}, {"attributeName":"resourceType","attributeValue":"class org.opendaylight.yang.gen.v1.http.cisco.com.ns.onem2m.com.att.rev140727.ContentInstance"}, {"attributeName":"creationTime","attributeValue":"01-04-15T23:56:112"}, {"attributeName":"lastModifiedTime","attributeValue":"01-04-15T23:56:112"}, {"attributeName":"labels","attributeValue":"Test_For_Python_Code"}, {"attributeName":"parentID","attributeValue":"InCSE1/Team2AEx/container19"}, {"attributeName":"stateTag","attributeValue":"5"}]]]]}
```

```
-----  
{"output":{"responseStatusCode":2002,"ResourceOutput":[{"resourceType":"contentInstance(latest-allAttributes)","resourceID":"contentInstance1","Attributes":[{"attributeName":"resourceID","attributeValue":"1713669515"}, {"attributeName":"resourceName","attributeValue":"contentInstance1"}, {"attributeName":"resourceType","attributeValue":"class org.opendaylight.yang.gen.v1.http.cisco.com.ns.onem2m.com.att.rev140727.ContentInstance"}, {"attributeName":"creationTime","attributeValue":"01-04-15T23:56:620"}, {"attributeName":"lastModifiedTime","attributeValue":"01-04-15T23:56:620"}, {"attributeName":"labels","attributeValue":"Test_For_Python_Code"}, {"attributeName":"parentID","attributeValue":"InCSE1/Team2AEx/container19"}, {"attributeName":"stateTag","attributeValue":"2"}]]]]}
```

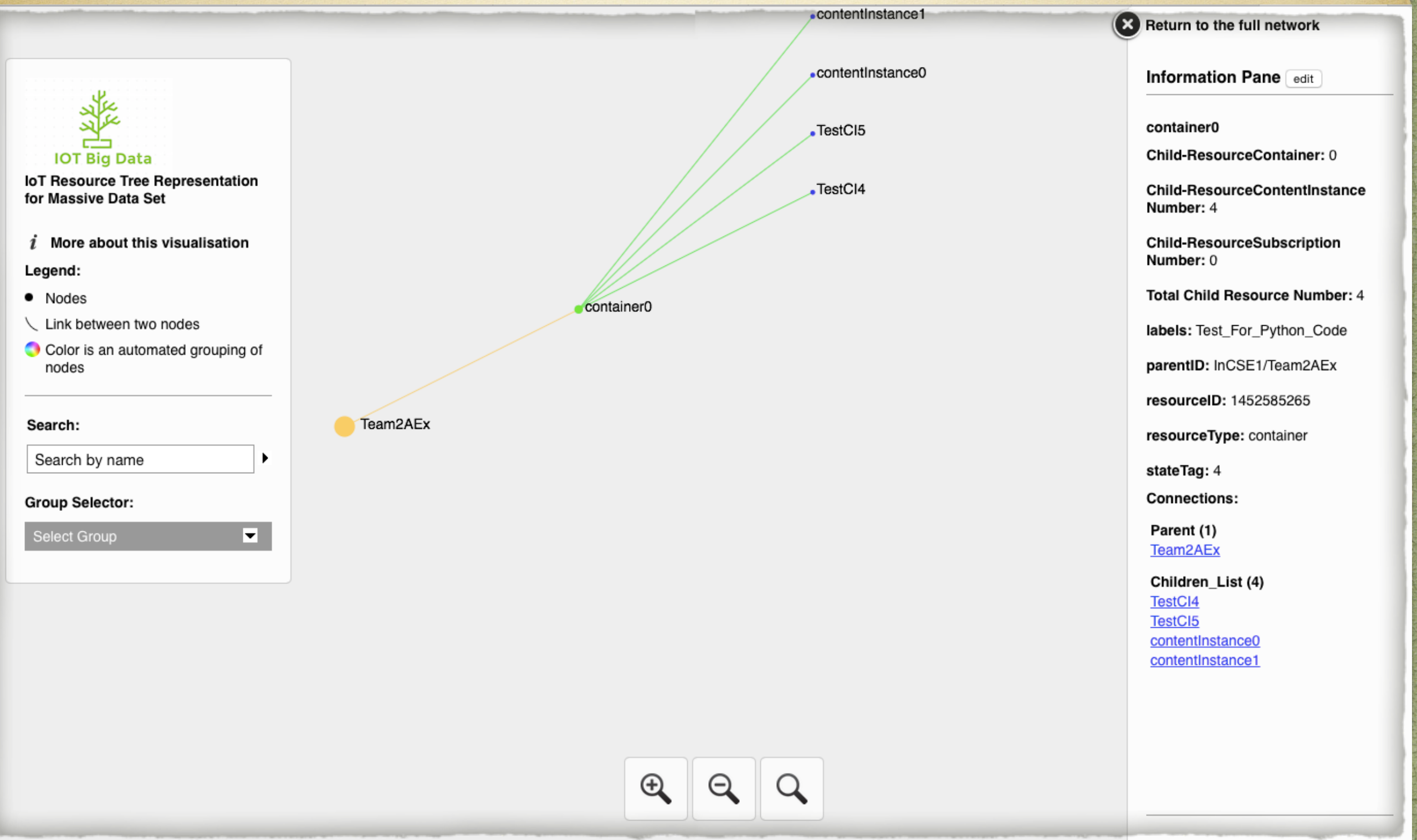
```
-----  
{"output":{"responseStatusCode":2002,"ResourceOutput":[{"resourceType":"contentInstance(latest-allAttributes)","resourceID":"contentInstance2","Attributes":[{"attributeName":"resourceID","attributeValue":"1713669516"}, {"attributeName":"resourceName","attributeValue":"contentInstance2"}, {"attributeName":"resourceType","attributeValue":"class org.opendaylight.yang.gen.v1.http.cisco.com.ns.onem2m.com.att.rev140727.ContentInstance"}, {"attributeName":"creationTime","attributeValue":"01-04-15T23:56:784"}, {"attributeName":"lastModifiedTime","attributeValue":"01-04-15T23:56:784"}, {"attributeName":"labels","attributeValue":"Test_For_Python_Code"}, {"attributeName":"parentID","attributeValue":"InCSE1/Team2AEx/container19"}, {"attributeName":"stateTag","attributeValue":"3"}]]]]}
```

```
-----  
{"output":{"responseStatusCode":2002,"ResourceOutput":[{"resourceType":"contentInstance(latest-allAttributes)","resourceID":"contentInstance0","Attributes":[{"attributeName":"resourceID","attributeValue":"1713669514"}, {"attributeName":"resourceName","attributeValue":"contentInstance0"}, {"attributeName":"resourceType","attributeValue":"class org.opendaylight.yang.gen.v1.http.cisco.com.ns.onem2m.com.att.rev140727.ContentInstance"}, {"attributeName":"creationTime","attributeValue":"01-04-15T23:56:457"}, {"attributeName":"lastModifiedTime","attributeValue":"01-04-15T23:56:457"}, {"attributeName":"labels","attributeValue":"Test_For_Python_Code"}, {"attributeName":"parentID","attributeValue":"InCSE1/Team2AEx/container19"}, {"attributeName":"stateTag","attributeValue":"1"}]]]]}
```

```
{  
  "id": "14",  
  "source": "1452585265",  
  "target": "235222053"  
},  
{  
  "id": "15",  
  "source": "1452585265",  
  "target": "235222052"  
},  
{  
  "id": "16",  
  "source": "1452585265",  
  "target": "261661984"  
},  
{  
  "id": "17",  
  "source": "1452585265",  
  "target": "261661983"  
}  
],  
"nodes": [  
  {  
    "attributes": {  
      "Child-ResourceContainer": "3",  
      "Child-ResourceSubscription Number": "0",  
      "Total Child Resource Number": "3",  
      "labels": "Test_Only_For_Yingchao",  
      "parentID": "InCSE1",  
      "resourceID": "1415208429",  
      "resourceType": "AE"  
    },  
    "color": "rgb(255,204,102)",  
    "id": "1415208429",  
    "label": "Team2AEx",  
    "size": 15,  
    "x": 0,  
    "y": 0  
  },  
  {  
    "id": "14",  
    "source": "1452585265",  
    "target": "235222053"  
  },  
  {  
    "id": "15",  
    "source": "1452585265",  
    "target": "235222052"  
  },  
  {  
    "id": "16",  
    "source": "1452585265",  
    "target": "261661984"  
  },  
  {  
    "id": "17",  
    "source": "1452585265",  
    "target": "261661983"  
  }  
]
```


Getting Sigma.JS Work

InteractiveVis



Thought for Efficiency

MemCache

GET requests = expensive operation

Transferring a JSON file = much quicker

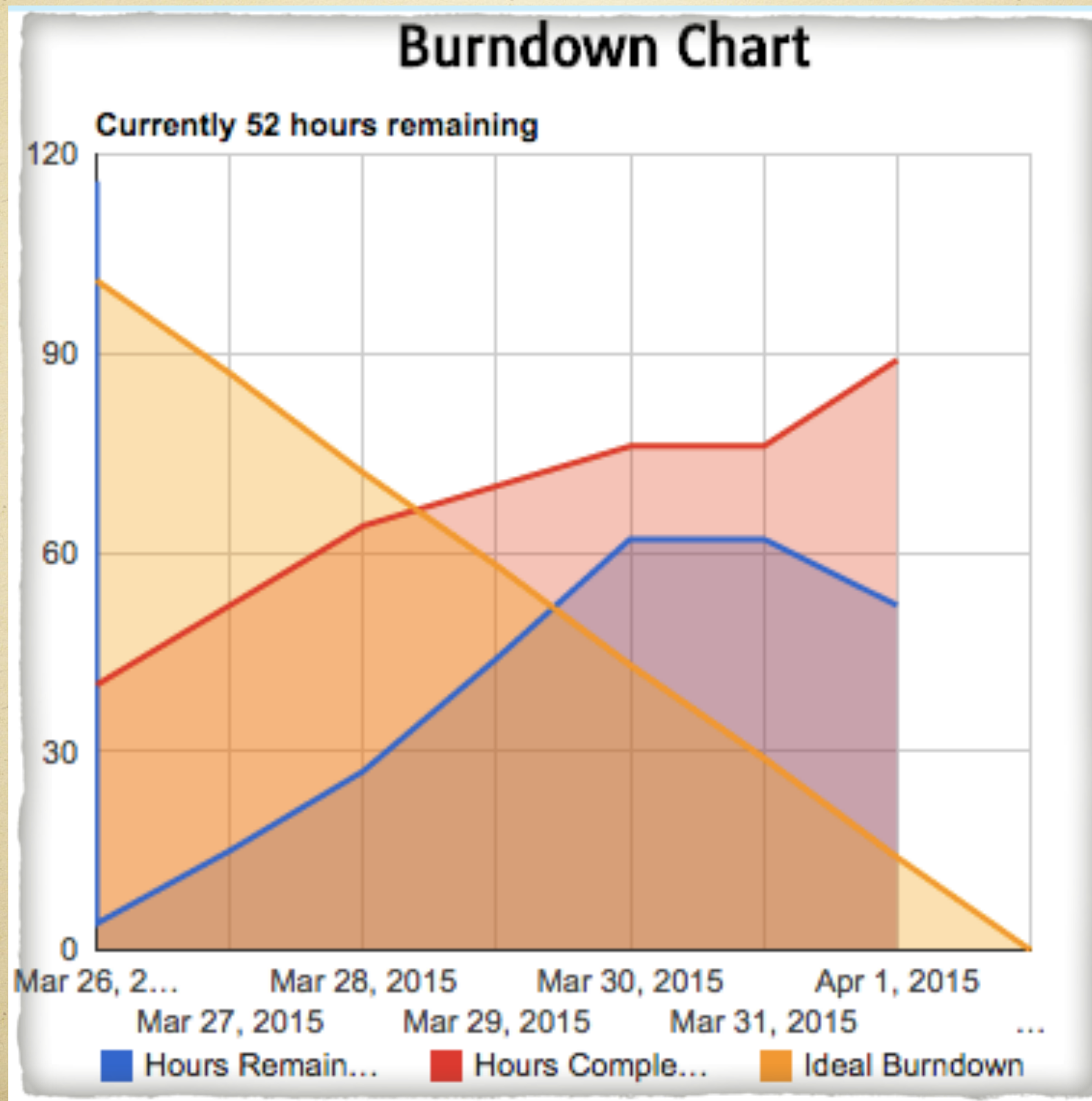
Generate JSON file on interval -> “Near Real-Time

Keep it cached in memory server-side

Compression

Utilize compression to fit JSON into memory

Burndown Chart



“Demo”

Work for the next Sprint

Sprint 7 (eow 11) [NoBurn] **84**


Utilize advanced visualization applications in sigma.js
👁 ⌚ Apr 6 🧠 21   

Implement Tree Management Functionality
✅ 0/3 🧠 21

Server-side memcache and compression
🧠 10

Server-side cron job for generating JSON on interval
🧠 8

REST calls from Client Visual should trigger generating new JSON and reset cron interval
🧠 3

(EXTRA) Learn AJAX to dynamically generate smaller JSON files for lower UI latency
👁 ✅ 0/3 ⌚ Apr 8 🧠 21   

Add a card...