Graph Maintenance

Cytoscape Integration
Key Deliveries

● Extending graph support adding during independent study
  ○ New file input formats HIPPIE and MatSim
  ○ add and remove vertices and edges
  ○ addition of novel file format supporting group in Spring

● Writing a plugin for Cytoscape to import MASS graph for visualization
  ○ Retrieve MASS in-memory graph from Cytoscape
What is “graph” support?

Multi-dimensional model
- Data is chunked to each node in stripes
- Each Place owns an upper and lower boundary of data within the entire dataset
- Outside of this must be exchanged or retrieved between places

Graph model
- Added an additional data structure within GraphPlaces to hold additional layers of data
- Each node holds a slice of all of the vertices
- Communication is direct between vertices (no sense of a ‘boundary’)
- ExchangeAll is performed on logical neighbors instead of data ‘boundaries’
MASS Data Models

Existing multi-dimensional model

- MASS Layer
  - Existing in PlacesBase

New model for graphs

- Graph Layer 1
  - Added in GraphPlaces

- Graph Layer 2

Node 1 - Controller
Node 2 - Worker 1
Node 3 - Worker 2
Node 4 - Worker 3
MatSim XML

- XML format
- Technically multiple files. Using the network only
  - Additional file for vehicle types
- Includes edge list and vertex definitions
- Ids are 1 based sequential
- Used a combination of built-in Java XML Reader and XPath queries

Neighbors: 
```
"/network/links/link[@from='" + (index + 1) + "]"
```
<table>
<thead>
<tr>
<th>Protein complex string and non-sequential integer id</th>
<th>Interaction value/weight</th>
<th>Free form metadata field</th>
<th>Required map from logical id to global index</th>
</tr>
</thead>
<tbody>
<tr>
<td>AL1A1_HUMAN 216</td>
<td>AL1A1_HUMAN 216 0.76</td>
<td>experiments in vivo, Two-hybrid; pmids: 12081471, 16189514, 25416956; sources: HPRD, BioGRID, IntAct, MINT, J2D, Rual05</td>
<td></td>
</tr>
<tr>
<td>ITA7_HUMAN 3679</td>
<td>ACHA_HUMAN 1134 0.73</td>
<td>experiments in vivo, Affinity Capture-Western, affinity chromatography technology; pmids: 10910772; sources: HPRD, BioGRID, J2D</td>
<td></td>
</tr>
<tr>
<td>NEB1_HUMAN 55607</td>
<td>ACTG_HUMAN 71 0.65</td>
<td>experiments in vitro, in vivo; pmids: 9362313, 12052877; sources: HPRD</td>
<td></td>
</tr>
<tr>
<td>SRGN_HUMAN 5552</td>
<td>CD44_HUMAN 960 0.63</td>
<td>experiments in vivo; pmids: 9334256, 16189514, 16713569; sources: HPRD, J2D, Rual05, Lim06</td>
<td></td>
</tr>
<tr>
<td>PAK1_HUMAN 5058</td>
<td>ERBB2_HUMAN 2064 0.73</td>
<td>experiments in vivo, Affinity Capture-Western, affinity chromatography technology; pmids: 9774445; sources: HPRD, BioGRID, J2D, STRING</td>
<td></td>
</tr>
<tr>
<td>DLG4_HUMAN 1742</td>
<td>ERBB2_HUMAN 2064 0.67</td>
<td>experiments in vivo, Two-hybrid, Affinity Capture-Western, Co-fractionation, affinity chromatography technology; pmids: 10839362, 16713569; sources: HPRD, BioGRID, J2D, Lim06</td>
<td></td>
</tr>
<tr>
<td>P85L_HUMAN 5296</td>
<td>ERBB2_HUMAN 2064 0.89</td>
<td>experiments in vivo, Reconstituted Complex, Biochemical Activity, protein array, pull down, enzymatic study; pmids: 1334406, 16273903, 16729043; sources: HPRD, BioGRID, MINT, J2D, IntAct, KEGG, STRING</td>
<td></td>
</tr>
<tr>
<td>PTN18_HUMAN 26469</td>
<td>ERI2_HUMAN 2064 0.88</td>
<td>experiments in vitro, pull down, anti tag coimmunoprecipitation, x-ray crystallography, phosphatase assay; pmids: 14660651, 25081058; sources: HPRD, J2D, IntAct</td>
<td></td>
</tr>
<tr>
<td>SMURF2_HUMAN 64730</td>
<td>RHG05_HUMAN 394 0.88</td>
<td>experiments: Two-hybrid, Affinity Capture-Western, pmids: 15231748, 28514442; species: Mus musculus (Mouse); sources: HPRD, MINT, J2D, Collard04, IntAct, BioGRID</td>
<td></td>
</tr>
<tr>
<td>UBX11_HUMAN 91544</td>
<td>ZFYV9_HUMAN 9372 0.73</td>
<td>experiments: Two-hybrid, pmids: 16189514, 15231748; sources: HPRD, MINT, J2D, Rual05, Collard04, IntAct, BioGRID</td>
<td></td>
</tr>
<tr>
<td>NCTR1_HUMAN 9437</td>
<td>ZFYV9_HUMAN 9372 0.73</td>
<td>experiments: Two-hybrid, pmids: 16189514, 15231748; sources: HPRD, MINT, J2D, Rual05, Collard04, IntAct, BioGRID</td>
<td></td>
</tr>
<tr>
<td>NCTR1_HUMAN 9437</td>
<td>ZFYV9_HUMAN 9372 0.73</td>
<td>experiments: Two-hybrid, pmids: 16189514, 15231748; sources: HPRD, MINT, J2D, Rual05, Collard04, IntAct, BioGRID</td>
<td></td>
</tr>
<tr>
<td>LYN_HUMAN 4067</td>
<td>PPR18_HUMAN 5511 0.67</td>
<td>experiments in vitro, Biochemical Activity, enzymatic study; pmids: 11104670; sources: HPRD, BioGRID, J2D</td>
<td></td>
</tr>
<tr>
<td>NPHN_HUMAN 4868</td>
<td>LYN_HUMAN 4067 0.52</td>
<td>experiments in vivo; pmids: 12848735; sources: HPRD, J2D</td>
<td></td>
</tr>
<tr>
<td>DLG4_HUMAN 1742</td>
<td>LYN_HUMAN 4067 0.59</td>
<td>experiments in vivo; pmids: 981808, 9202138; species: Mus musculus (Mouse); sources: HPRD, BioGRID, HomoMINT, J2D</td>
<td></td>
</tr>
<tr>
<td>BCAR1_HUMAN 9564</td>
<td>LYN_HUMAN 4067 0.9</td>
<td>experiments in vitro, in vivo, Affinity Capture-Western, pmids: 981808, 9202138; species: Mus musculus (Mouse); sources: HPRD, BioGRID, HomoMINT, J2D</td>
<td></td>
</tr>
<tr>
<td>U119A_HUMAN 9994</td>
<td>LYN_HUMAN 4067 0.79</td>
<td>experiments in vitro, in vivo, Reconstituted Complex, Affinity Capture-Western, pmids: 9687533, 10790433; sources: HPRD, BioGRID, J2D</td>
<td></td>
</tr>
<tr>
<td>TRAT1_HUMAN 50852</td>
<td>LYN_HUMAN 4067 0.85</td>
<td>experiments in vitro, Reconstituted Complex, Affinity Capture-Western, pmids: 9687533, 10790433; sources: HPRD, BioGRID, J2D</td>
<td></td>
</tr>
<tr>
<td>SKAPI_HUMAN 8631</td>
<td>LYN_HUMAN 4067 0.75</td>
<td>experiments in vitro, Reconstituted Complex, Affinity Capture-Western, pmids: 9195895; sources: HPRD, BioGRID, J2D</td>
<td></td>
</tr>
<tr>
<td>SKAP3_HUMAN 9935</td>
<td>LYN_HUMAN 4067 0.8</td>
<td>experiments in vivo, Affinity Capture-Western, pmids: 981808, 9202138; species: Mus musculus (Mouse); sources: HPRD, BioGRID, HomoMINT, J2D</td>
<td></td>
</tr>
<tr>
<td>LYN_HUMAN 4067</td>
<td>TRPV4_HUMAN 59341 0.77</td>
<td>experiments in vitro, in vivo, Biochemical Activity, Affinity Capture-Western, pmids: 14635045; sources: HPRD, BioGRID, J2D</td>
<td></td>
</tr>
<tr>
<td>NCTR3_HUMAN 259197</td>
<td>CD59_HUMAN 966 0.74</td>
<td>experiments in vitro, in vivo, Affinity Capture-Western, pmids: 14635045; sources: HPRD, BioGRID, J2D</td>
<td></td>
</tr>
</tbody>
</table>
Graph addressing

Vertex attribute: “Hello”
Global index: 362
MASS Size: 100
Chunk size: 25

Layer Index: 3 (362 / 100)
Node index: 2 (362 % 100 / 25)
Place index: 12 (362 % 25)
Graph Maintenance Tasks

- Implement add & remove vertices
- Implement add & remove edges
- Serialize the data model to send to cytoscape
Maintenance - Vertices

Add

- Determine the new owner
  - controller maintains serial id
  - add we are the owner
  - or send new message type ‘MAINTENANCE_ADD_PLACE’ to the owner

Remove

- call remove locally
- send new message type ‘MAINTENANCE_REMOVE_PLACE’ to all nodes
Maintenance - Edge

Add

- Determine if source and neighbor vertices exist
- Find the owner of the source vertex
- Send new message type ‘MAINTENANCE_ADD_EDGE’

Remove

- Find owner of source vertex
- Send new message type ‘MAINTENANCE_REMOVE_EDGE’
Maintenance Example

- Basic constructor for empty graph
- Create vertices and edges
- CytoscapeListener for communication with Cytoscape
- pause to allow connection via Cytoscape

```kotlin
fun main() {
    MASS.setLoggingLevel(LogLevel.DEBUG)
    MASS.init(10000000)
    val places = GraphPlaces(0, Node::class.qualifiedName, 100)
    places.addVertex(1001)
    places.addVertex(10001)
    places.addEdge(1001, 10001, 0.9)
    places.addEdge(10001, 1001, 0.5)
    CytoscapeListener(places)
    MASS.pause()
    MASS.finish()
}
```
Maintenance - Serialization

- Java Object Serialization with ObjectOutputStream
- Lightweight GraphModel for storing the high level structure of the graph
- Lightweight VertexModel to store the vertex and neighbor information
- Converge all onto the controller node
- Send the full model across the wire to Cytoscape
Initial Cytoscape Plugin
Cytoscape Integration - Part 1

- Read graph from MASS
- Node & edge styling
- Create new nodes for non-existent neighbors
CytoscapeListener - MASS to Cytoscape

- A new Thread with a socket listening for requests from Cytoscape
- Dispatches requests to a Graph interface such as getGraph
- getGraph sends a ‘MAINTENANCE_GET_PLACES’ request to all nodes
- Populates a GraphModel of the local nodes vertices
- Merges response GraphModel from each node into a single GraphModel
- Send the complete model to Cytoscape
Data flow of MASS -> Import Network
// Create an empty network
CyNetwork cyNetwork = cnf.createNetwork();

List<VertexModel> vertices = graph.getVertices();

CyNode[] nodes = new CyNode[vertices.size()];

// Add vertices to the network
for (int v = 0; v < vertices.size(); v++) {
    VertexModel vertex = vertices.get(v);
    nodes[v] = cyNetwork.addNode();
    // set name for new vertex
    cyNetwork.getDefaultNodeTable().getRow(nodes[v].getSUID()).set("name", vertex.id);
}

// Add edges after all vertices are created
for (int v = 0; v < vertices.size(); v++) {
    VertexModel vertex = vertices.get(v);
    for (Object neighbor : vertex.neighbors.stream().filter(n -> !n.equals(vertex.id)).collect(Collectors.toList())) {
        CyNode neighborNode = nodesMap.get(neighbor);
        CyEdge edge = cyNetwork.addEdge(nodes[v], neighborNode, true);
        CyRow edgeRow = cyNetwork.getDefaultEdgeTable().getRow(edge.getSUID());
        edgeRow.set("name", v + ":" + neighbor);
        edgeRow.set("interaction", vertex.id + " --> " + neighbor);
    }
}
Limitations

- Layout code is not intuitive
  - User must layout manually (F5 is quickest)
- Graph is collected on master
- Large graph such as HIPPIE_CURRENT is very sluggish
  - Appears to be on Cytoscape side
- Client library configuration not obvious after the move to Java 11
  - Cytoscape is 1.8

Future Work and Enhancements

- Streaming/partial loading of a graph in Cytoscape could improve intractability
- Make MASS host and port configurable from within Cytoscape
- Send jobs from Cytoscape to run on MASS
- Configurable styling e.g:
  - Weight based coloring
  - Attribute based node style
- Move to JSON serialization
HIPPIE CURRENT data set with neighbors in RED
References for Cytoscape and others

- [http://code.cytoscape.org/javadoc/3.7.2/](http://code.cytoscape.org/javadoc/3.7.2/)
  - Actually useful javadoc style Cytoscape api reference

- [https://github.com/cytoscape/cytoscape-app-samples](https://github.com/cytoscape/cytoscape-app-samples)
  - These samples + the “app ladder” serve as the bulk of what I needed to get started

  - ‘App Ladder’