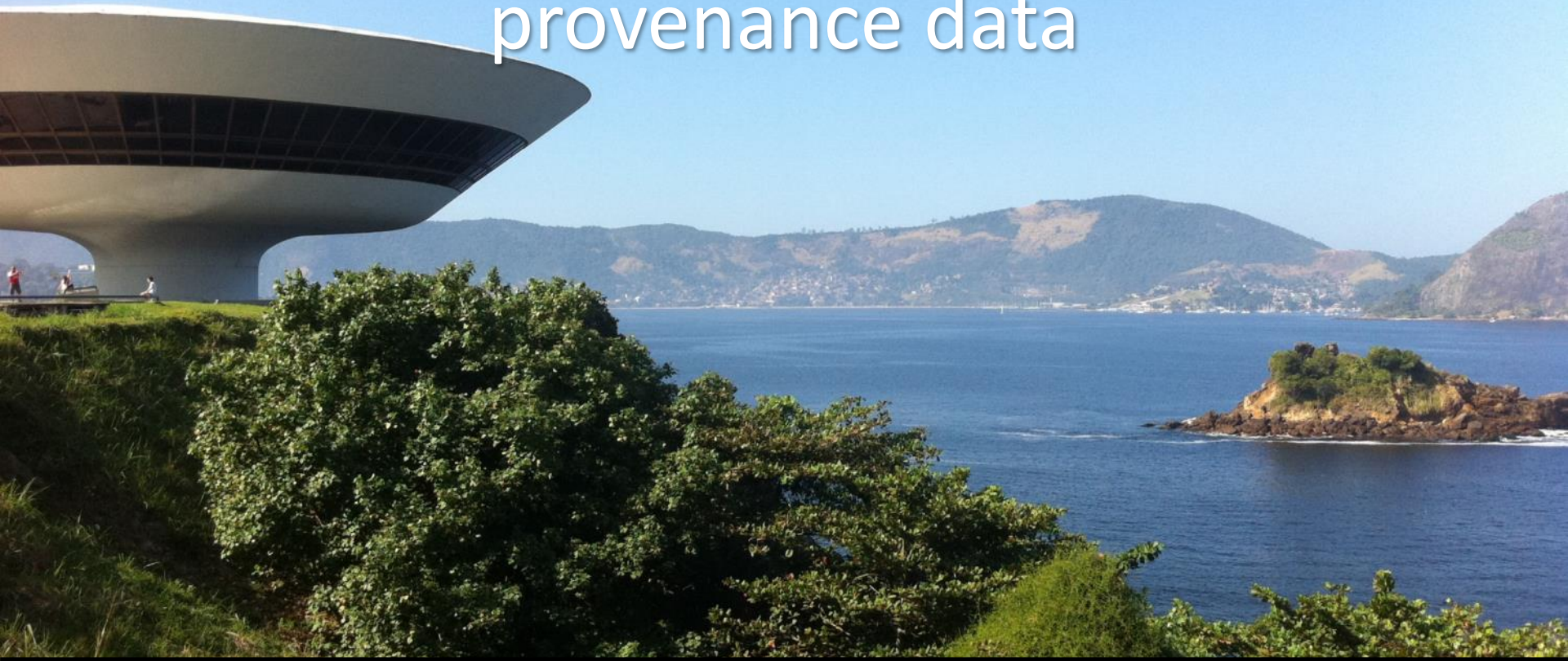


# Prov Viewer: a graph-based visualization tool for interactive exploration of provenance data

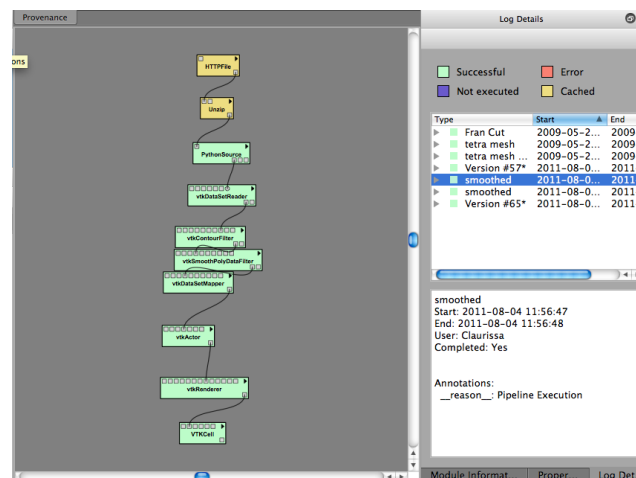
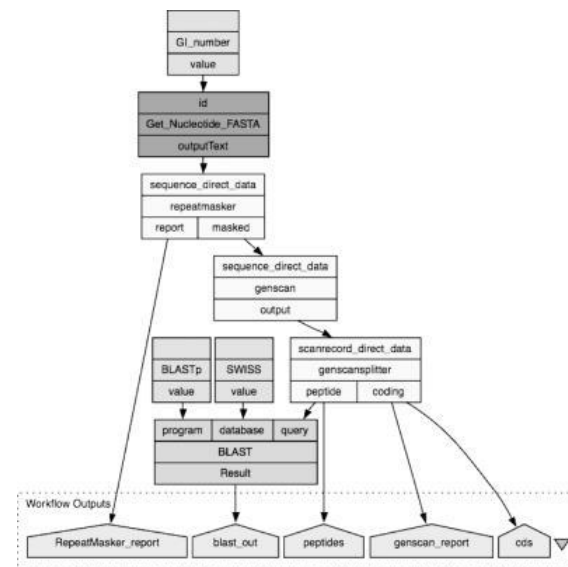


# Introduction

- Recent attention to provenance gathering
- Many tools for capturing provenance data
  - Provenance for each task
- Displaying provenance is an issue in present times
  - Simple node-link diagrams
  - Basic visualization features

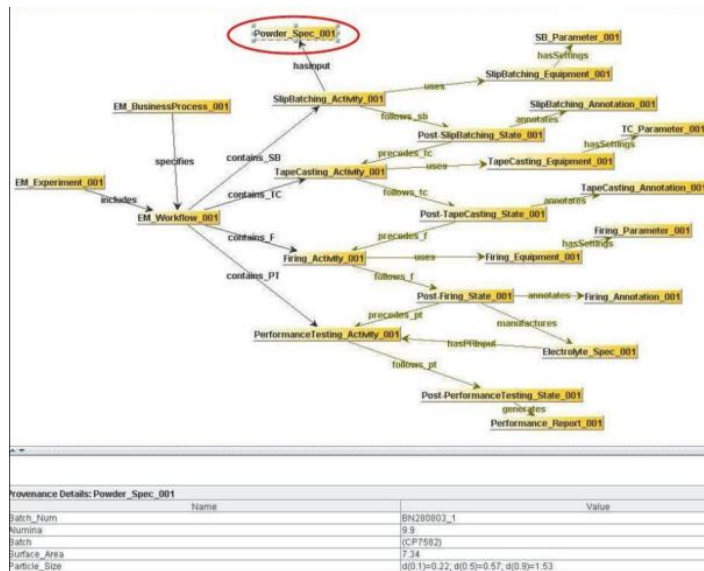
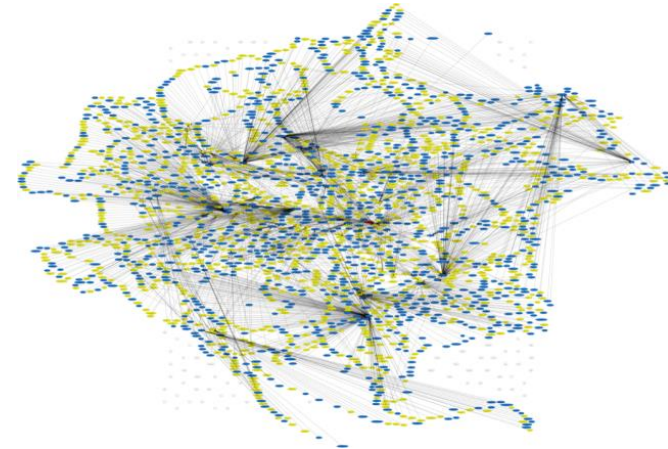
# Motivation

- Workflow management systems
  - Build-in visualization features
  - Lack manipulation features
  - Incompatible with other tools
  - Restricted metadata visualization



# Motivation

- Standalone visualization tools
  - Additional knowledge (e.g., SQL, prolog)
  - Not compatible with PROV
  - Restricted metadata visualization



Definition Simulate runs View runs Provenance

Executions			Users output		User views	
Id	Label	Run on	Data id	Ts	None	
1	exec1	Jun 24, ...	d447	21	Joe	
					Mary	
					<input type="checkbox"/> Compute locally	

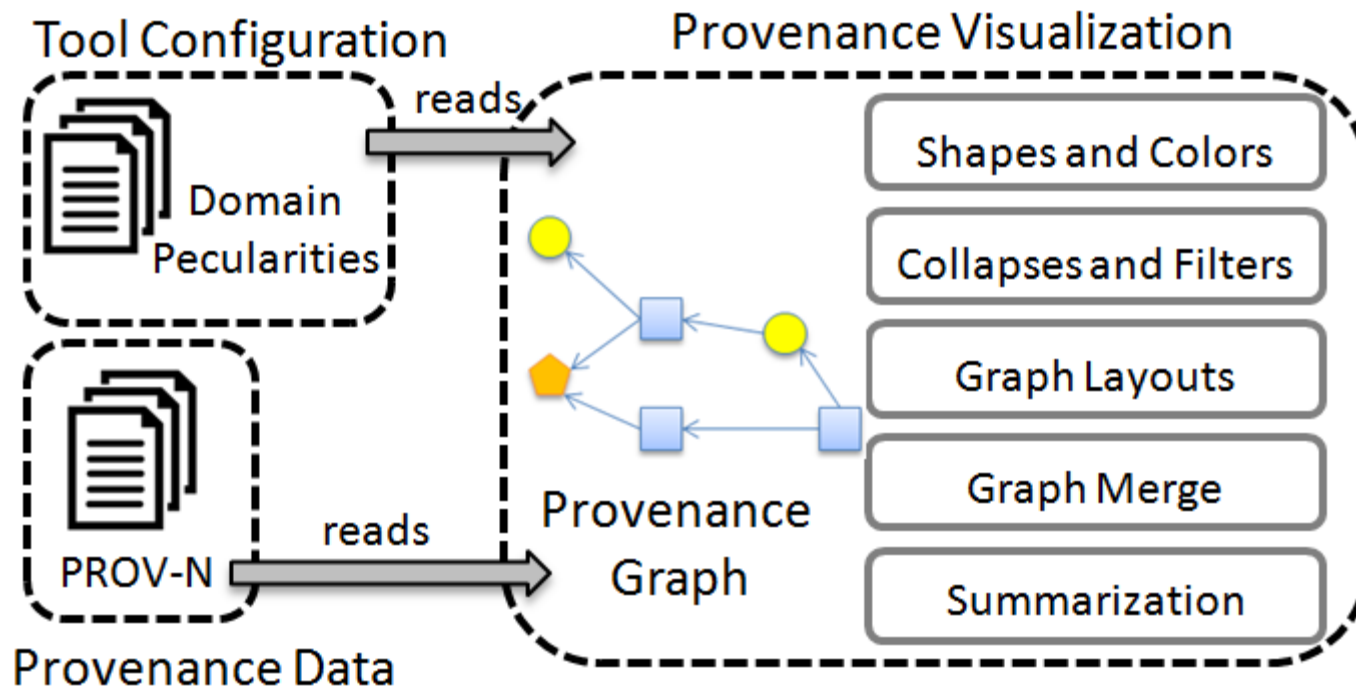
Data Id	Ts
d204	1
d203	1
d206	1
d205	1
d202	1

# Goal

- Specialized provenance visualization tool
- Prov-compatible
- Exploratory and interactive visualization
- Visual analysis of provenance data

# Prov Viewer

- Provenance Graph Visualization Tool



# Configuration

- Layout
- Relationships (Edges)
  - Color scheme
  - Summarization formula for values (if any)
- Vertices
  - Color schemes
- Automatic detection
  - Relationships
  - Vertex attributes for Color schemes

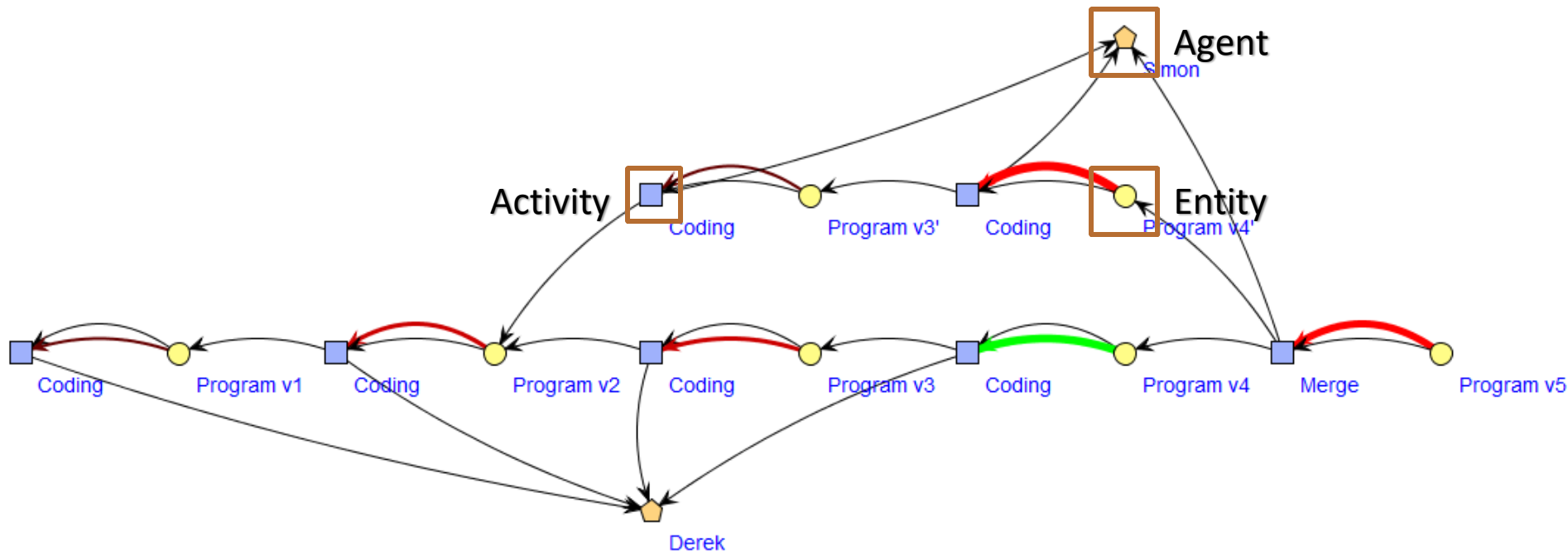
```
<default_layout>SpatialLayout</default_layout>
<layoutAxis_X>ObjectPosition_X</layoutAxis_X>
<layoutAxis_Y>ObjectPosition_Y</layoutAxis_Y>
<imageLocation></imageLocation>
<imageOffset_X>0</imageOffset_X>
<imageOffset_Y>0</imageOffset_Y>
<spatialLayoutPosition>0.0</spatialLayoutPosition>
<zoomLevel>0</zoomLevel>
<temporalLayoutbackbone>Player</temporalLayoutbackbone>
<temporalLayoutscale>1.0</temporalLayoutscale>
```

```
<edgetype>
  <edge>wasGeneratedBy</edge>
  <edgestroke>MAX</edgestroke>
  <collapsefunction>SUM</collapsefunction>
</edgetype>
```

```
<colorscheme>
  <attribute>Duration</attribute>
  <class>ActivityInvertedScheme</class>
  <values></values>
  <goodvalue></goodvalue>
  <badvalue></badvalue>
</colorscheme>
```

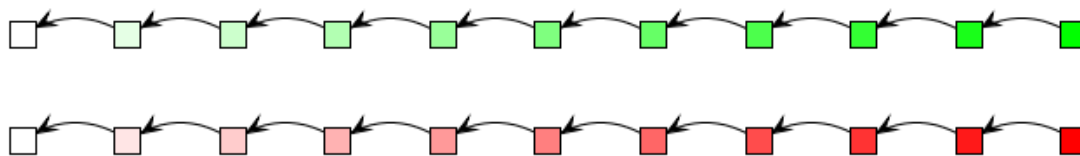
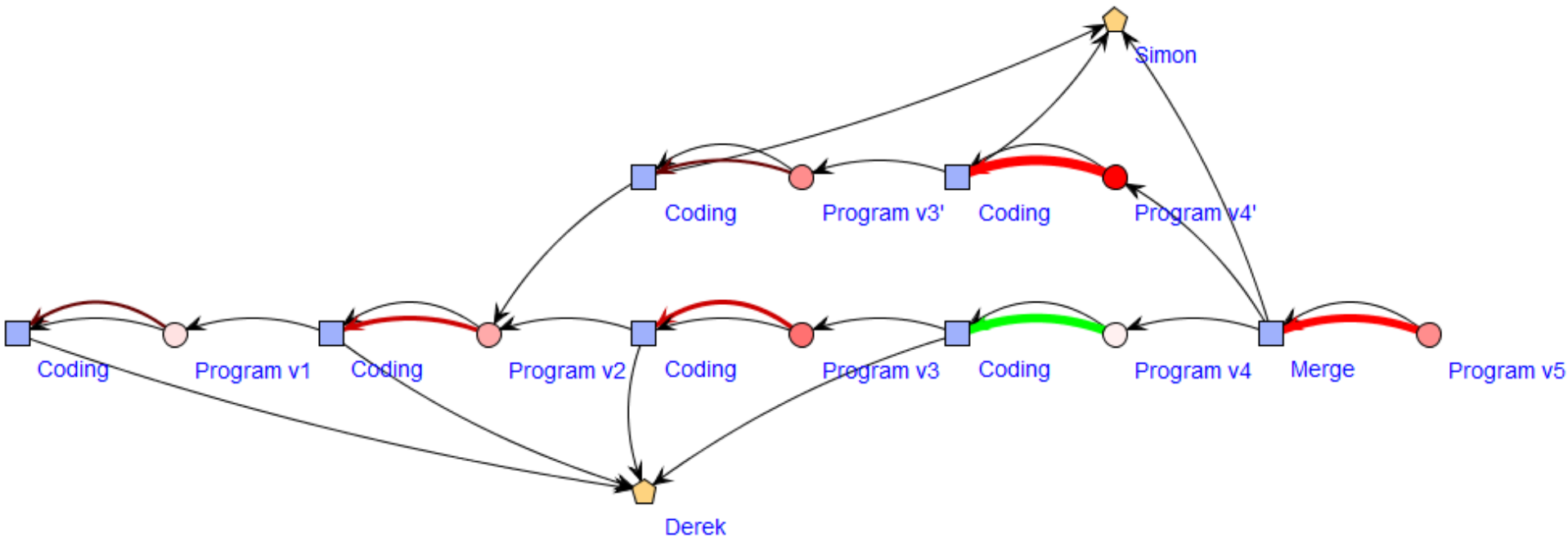
# Shapes and Colors

- PROV shapes
  - Pentagon: Agent
  - Circle: Entity
  - Square: Activity
- Default PROV colors
  - Traffic light color based on metadata information

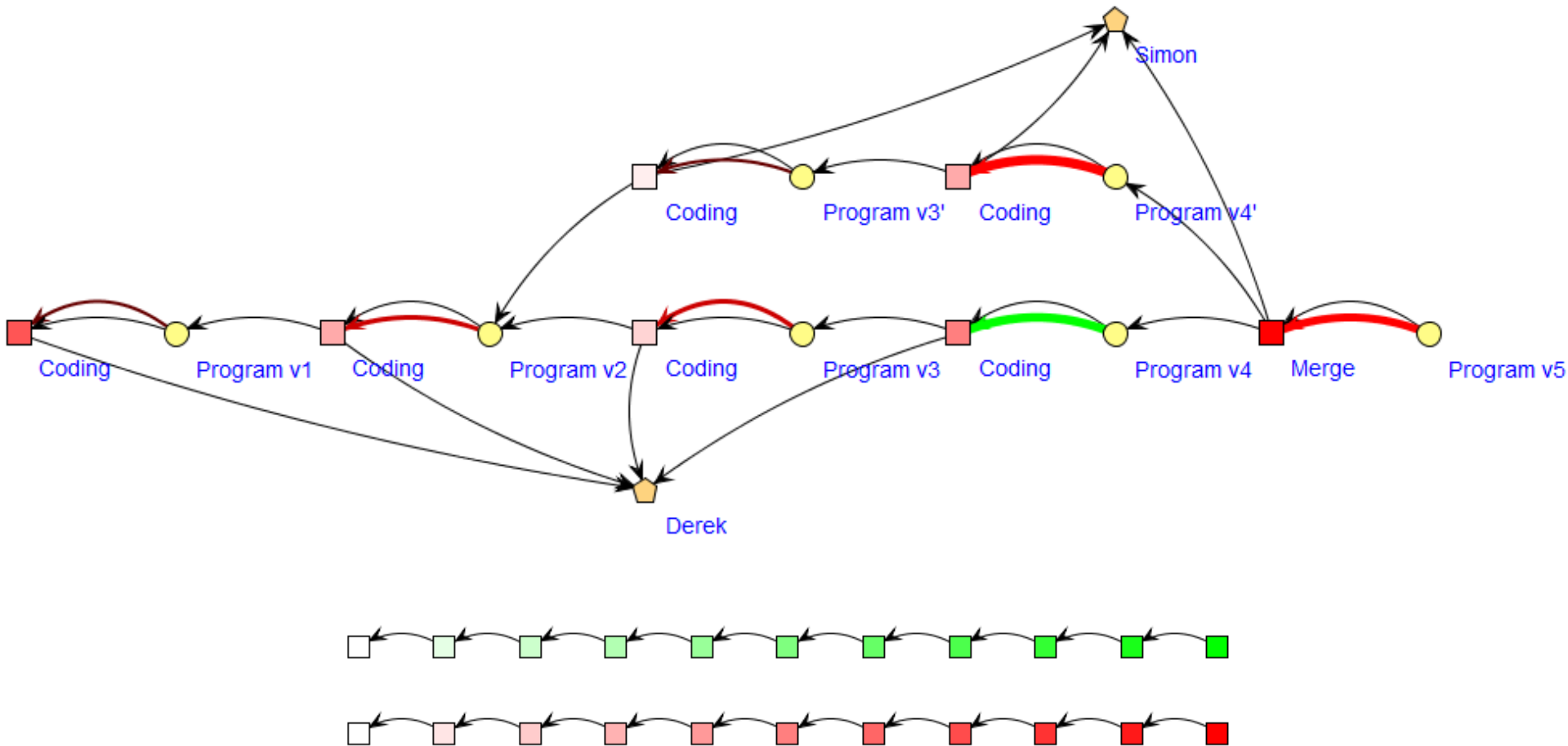




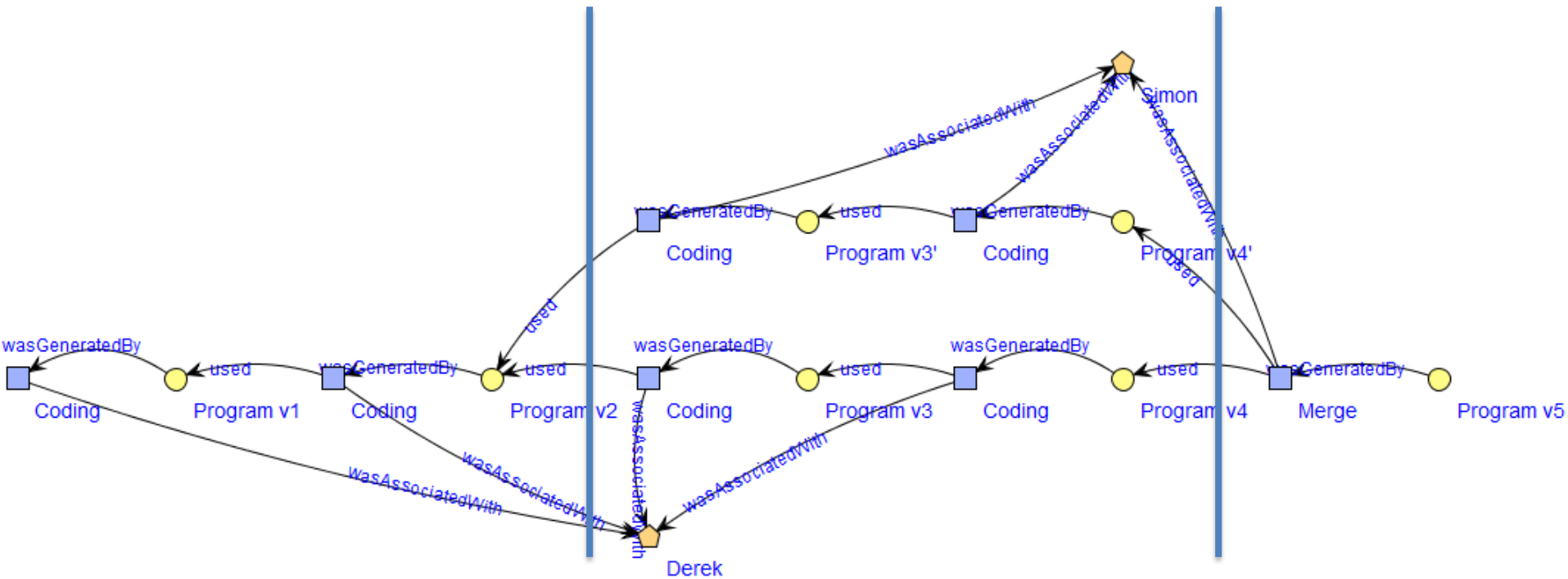
# Shapes and Colors



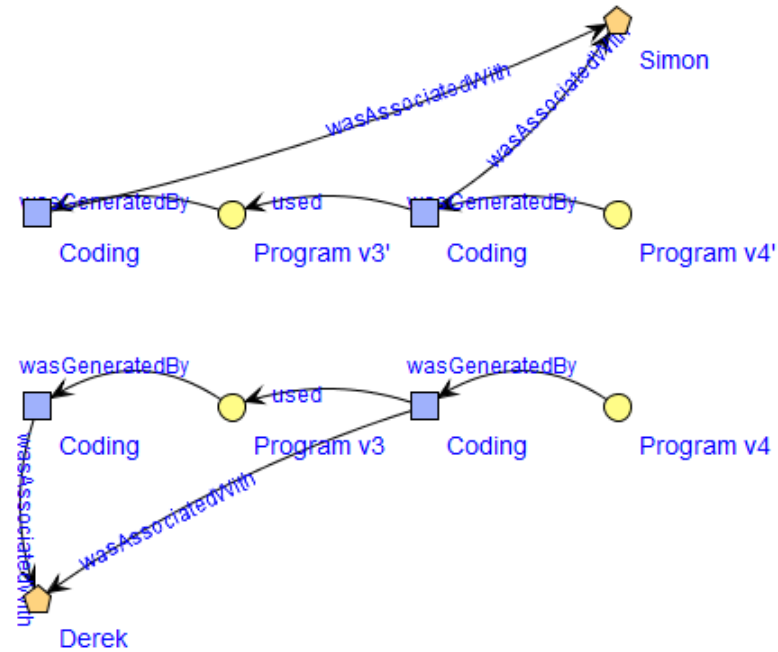
# Shapes and Colors



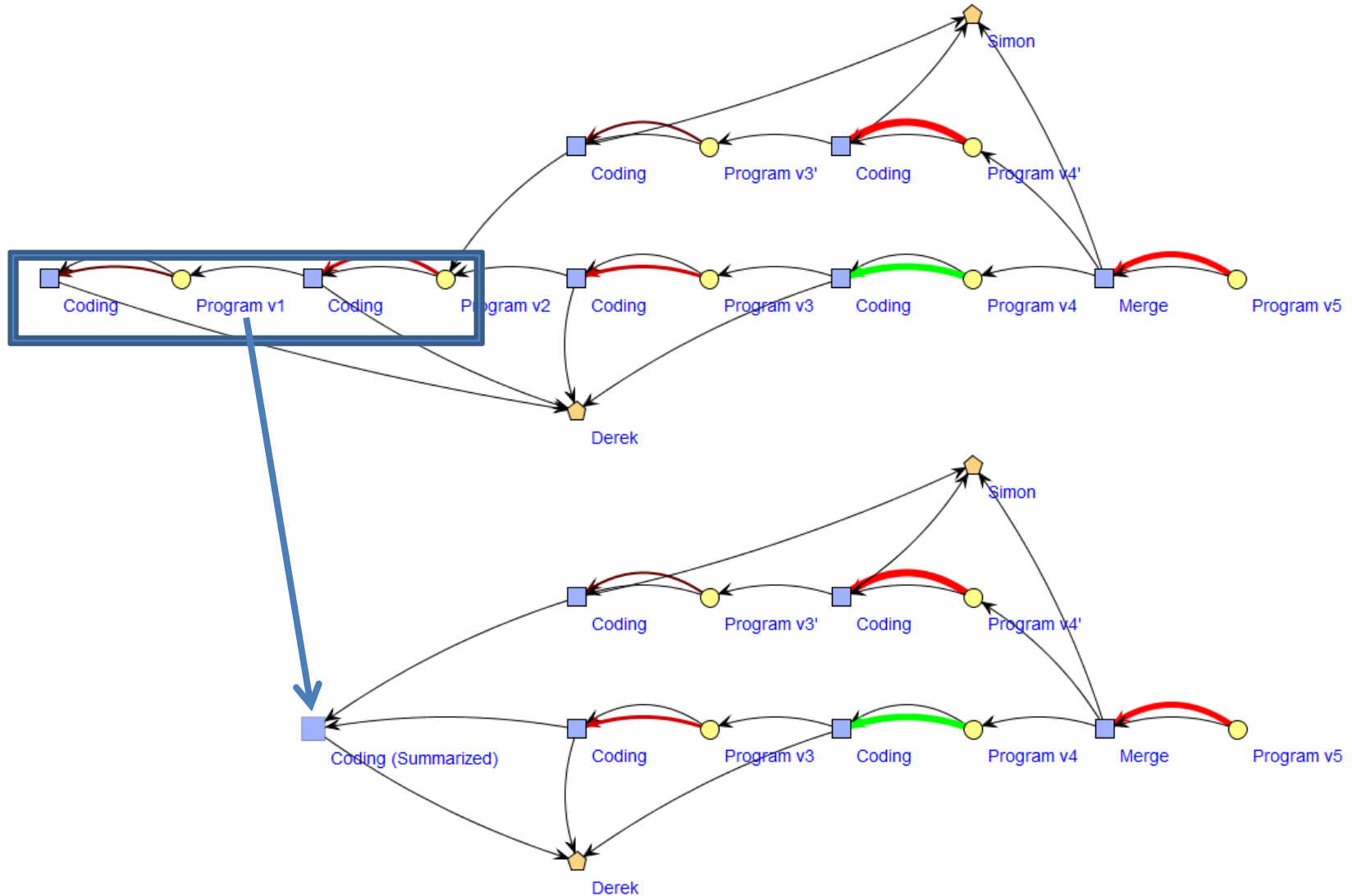
# Temporal Filter



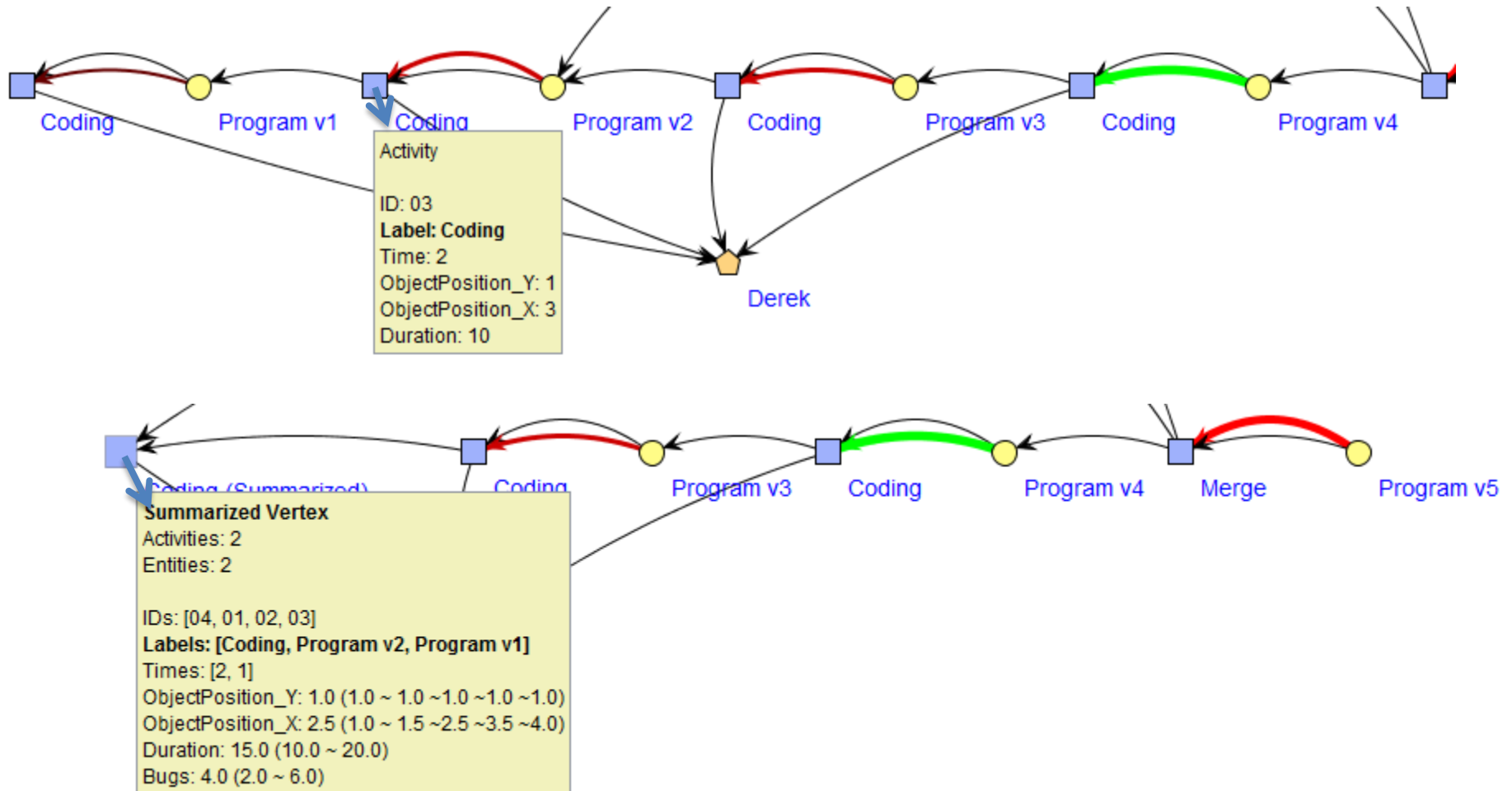
# Temporal Filter



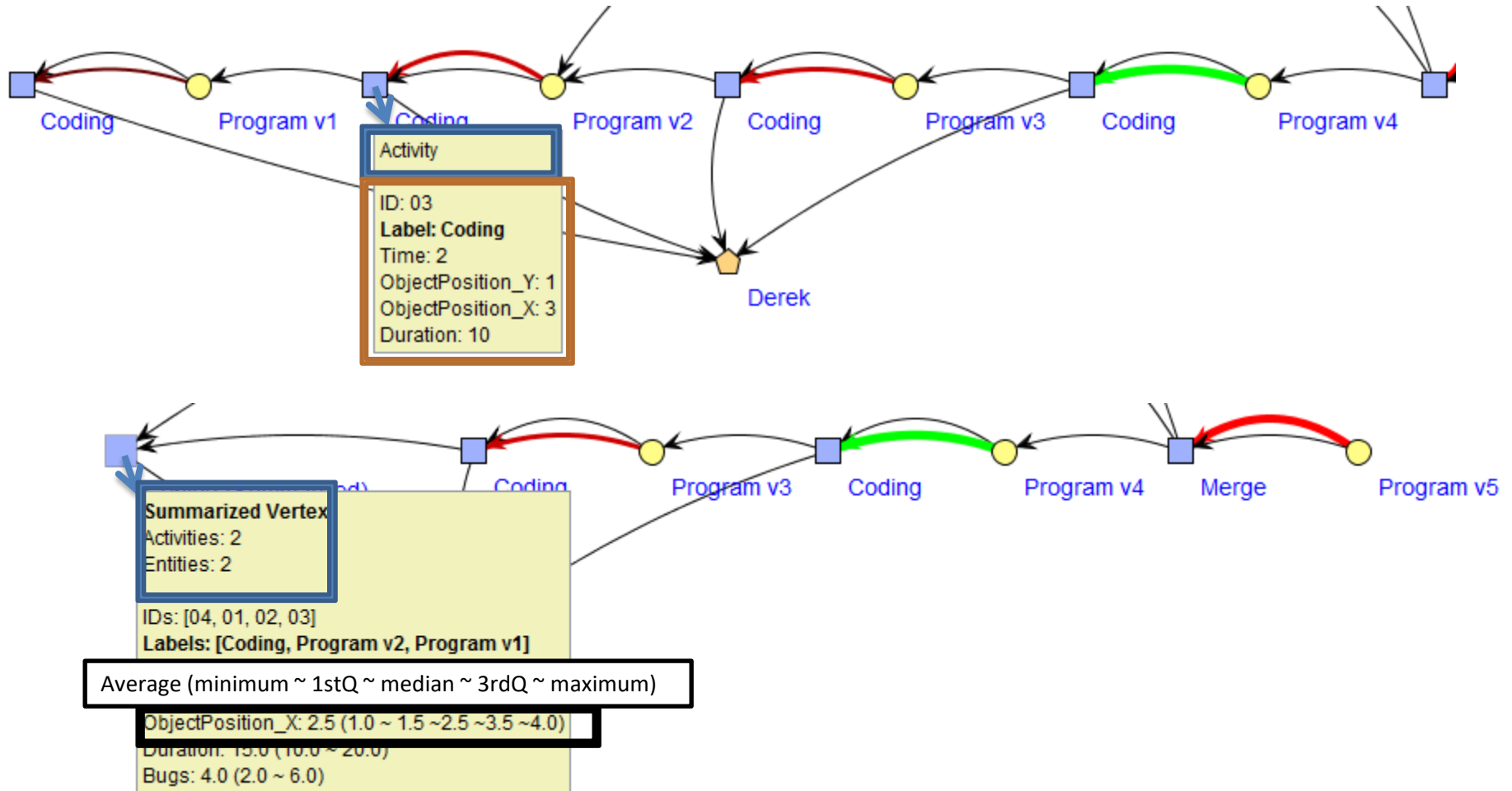
# Summarization



# Vertex Metadata

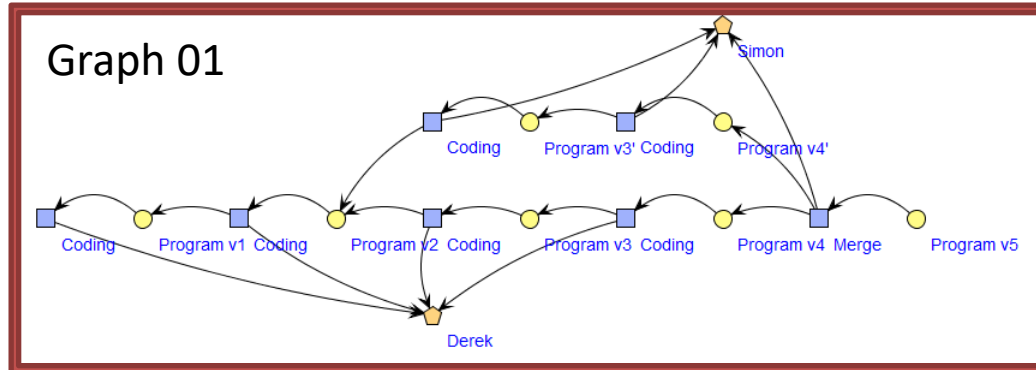


# Vertex Metadata



# Automatic Summarization

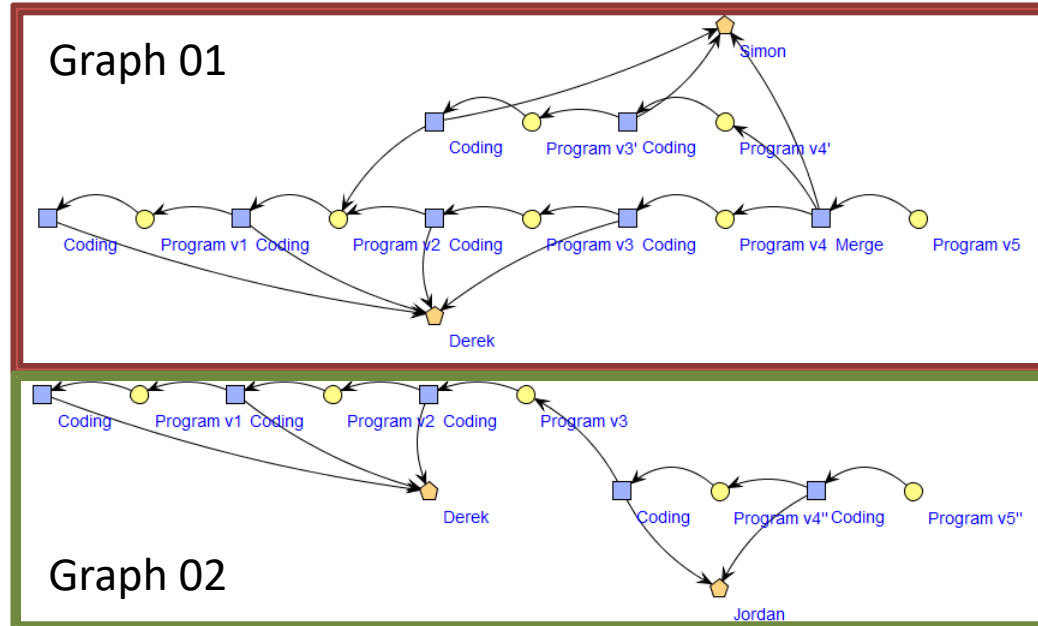
- Combine similar vertices
  - Compare vertices
  - Detect similarity
  - No information loss
- Similarity
  - Vertex type
  - Attributes
  - Values
- Between graphs
  - Graph merge
  - Analyze multiple trials or sessions
- Within same graph
  - Sequential vertices
  - Deduplication





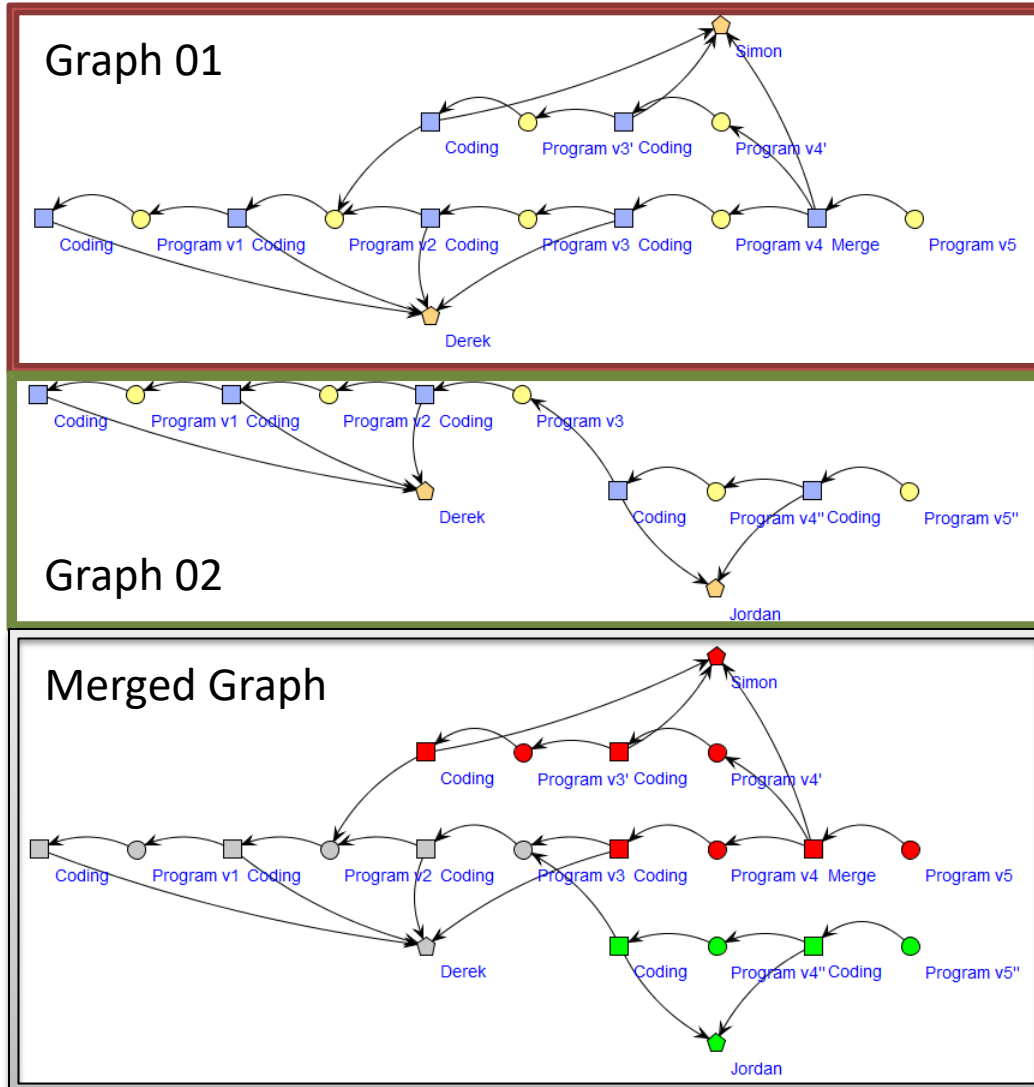
# Automatic Summarization

- Combine similar vertices
  - Compare vertices
  - Detect similarity
  - No information loss
- Similarity
  - Vertex type
  - Attributes
  - Values
- Between graphs
  - Graph merge
  - Analyze multiple trials or sessions
- Within same graph
  - Sequential vertices
  - Deduplication



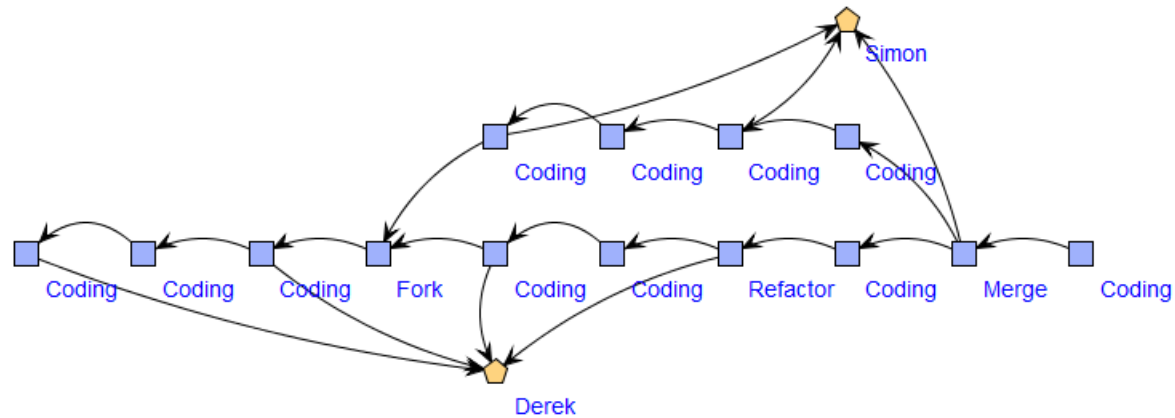
# Automatic Summarization

- Combine similar vertices
  - Compare vertices
  - Detect similarity
  - No information loss
- Similarity
  - Vertex type
  - Attributes
  - Values
- Between graphs
  - Graph merge
  - Analyze multiple trials or sessions
- Within same graph
  - Sequential vertices
  - Deduplication



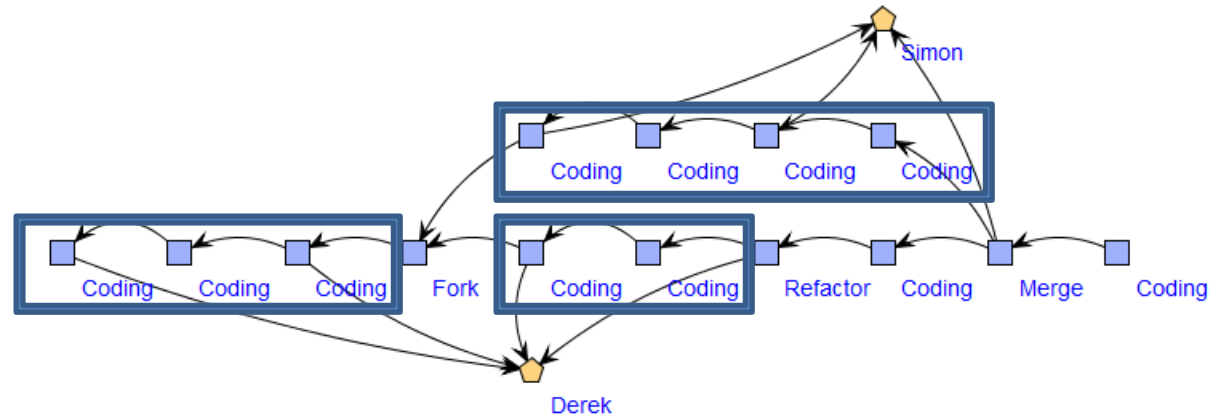
# Automatic Summarization

- Combine similar vertices
  - Compare vertices
  - Detect similarity
  - No information loss
- Similarity
  - Vertex type
  - Attributes
  - Values
- Between graphs
  - Graph merge
  - Analyze multiple trials or sessions
- Within same graph
  - Sequential vertices
  - Deduplication



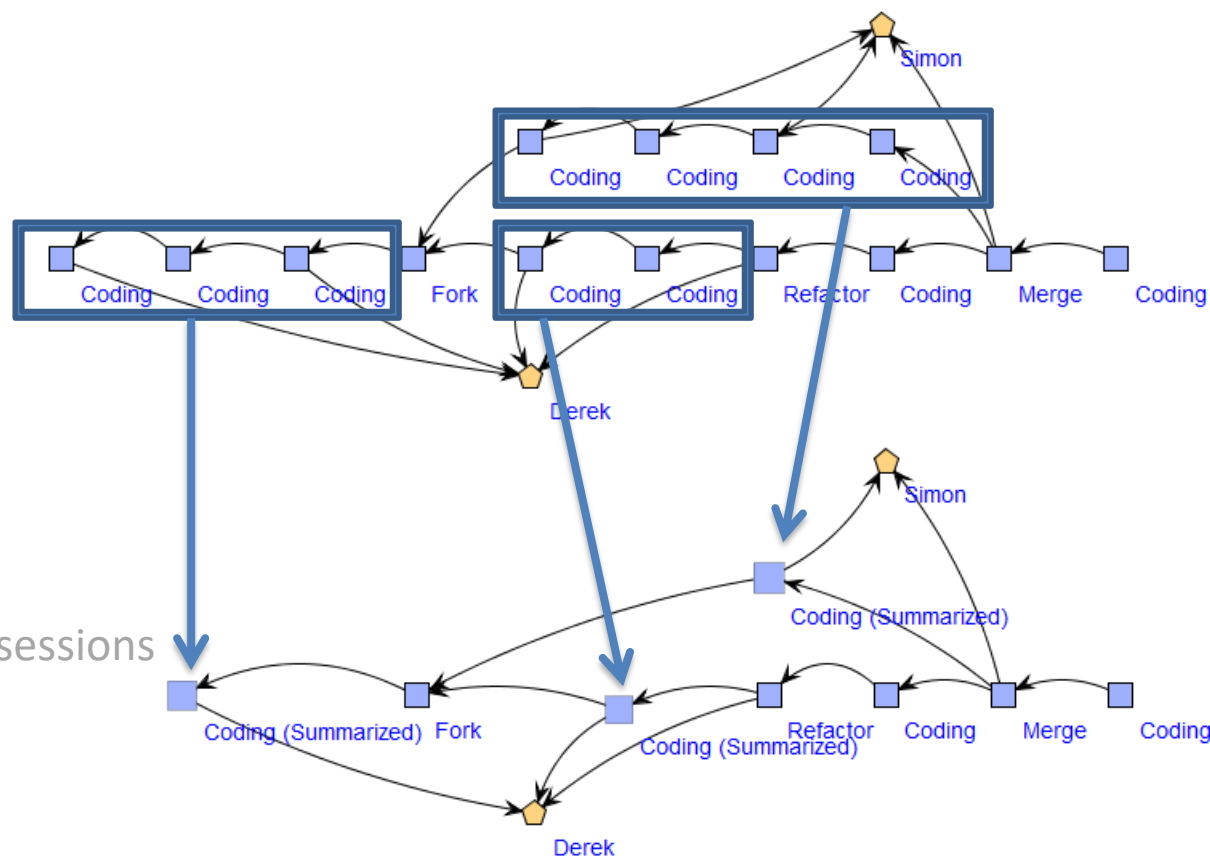
# Automatic Summarization

- Combine similar vertices
  - Compare vertices
  - Detect similarity
  - No information loss
- Similarity
  - Vertex type
  - Attributes
  - Values
- Between graphs
  - Graph merge
  - Analyze multiple trials or sessions
- Within same graph
  - Sequential vertices
  - Deduplication

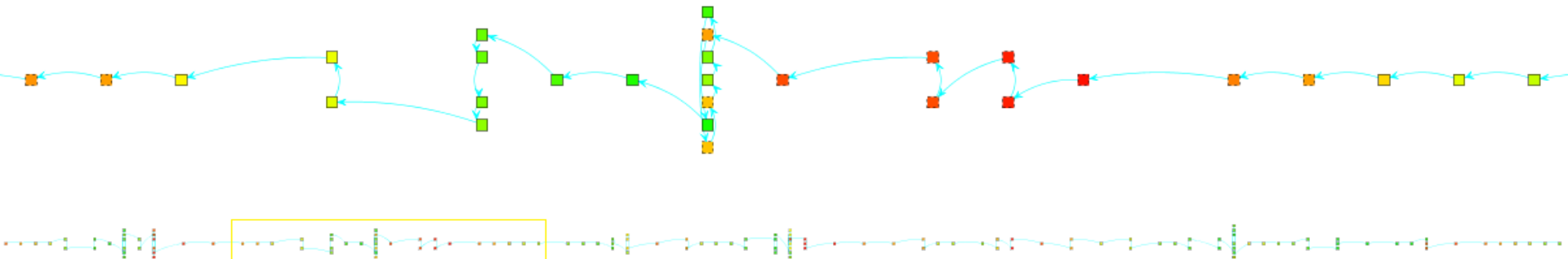


# Automatic Summarization

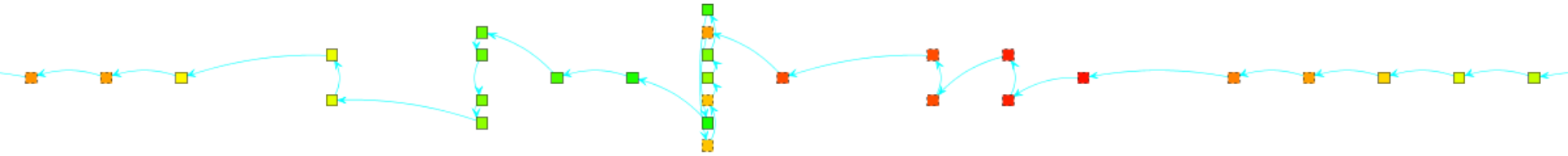
- Combine similar vertices
  - Compare vertices
  - Detect similarity
  - No information loss
- Similarity
  - Vertex type
  - Attributes
  - Values
- Between graphs
  - Graph merge
  - Analyze multiple trials or sessions
- Within same graph
  - Sequential vertices
  - Deduplication



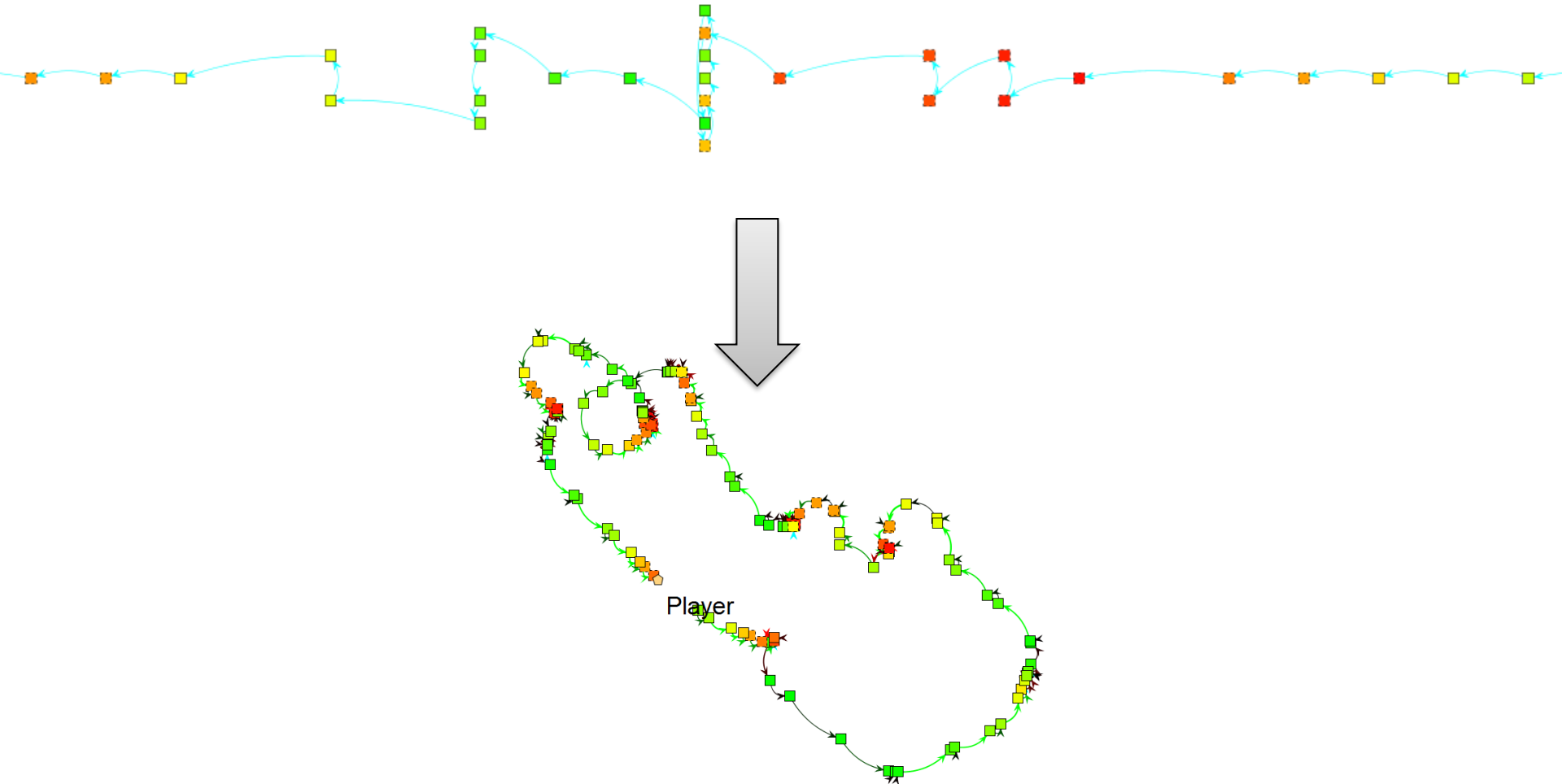
# Spatial referencing data



# Spatial referencing data

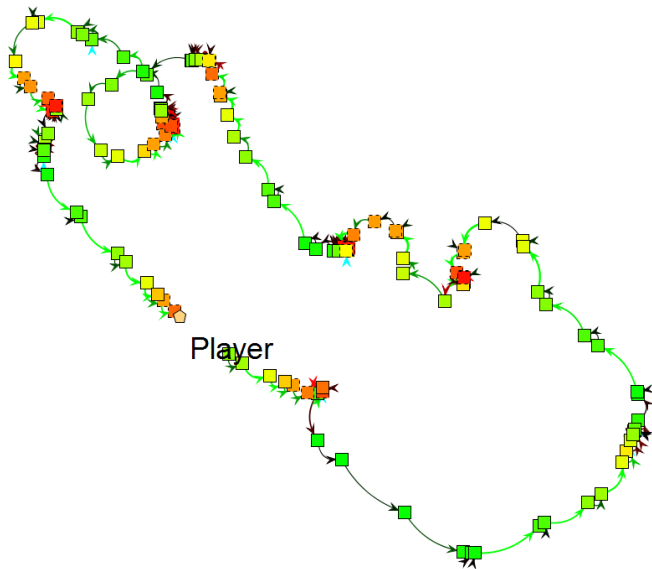


# Spatial referencing data





# Spatial referencing data



# Spatial referencing data

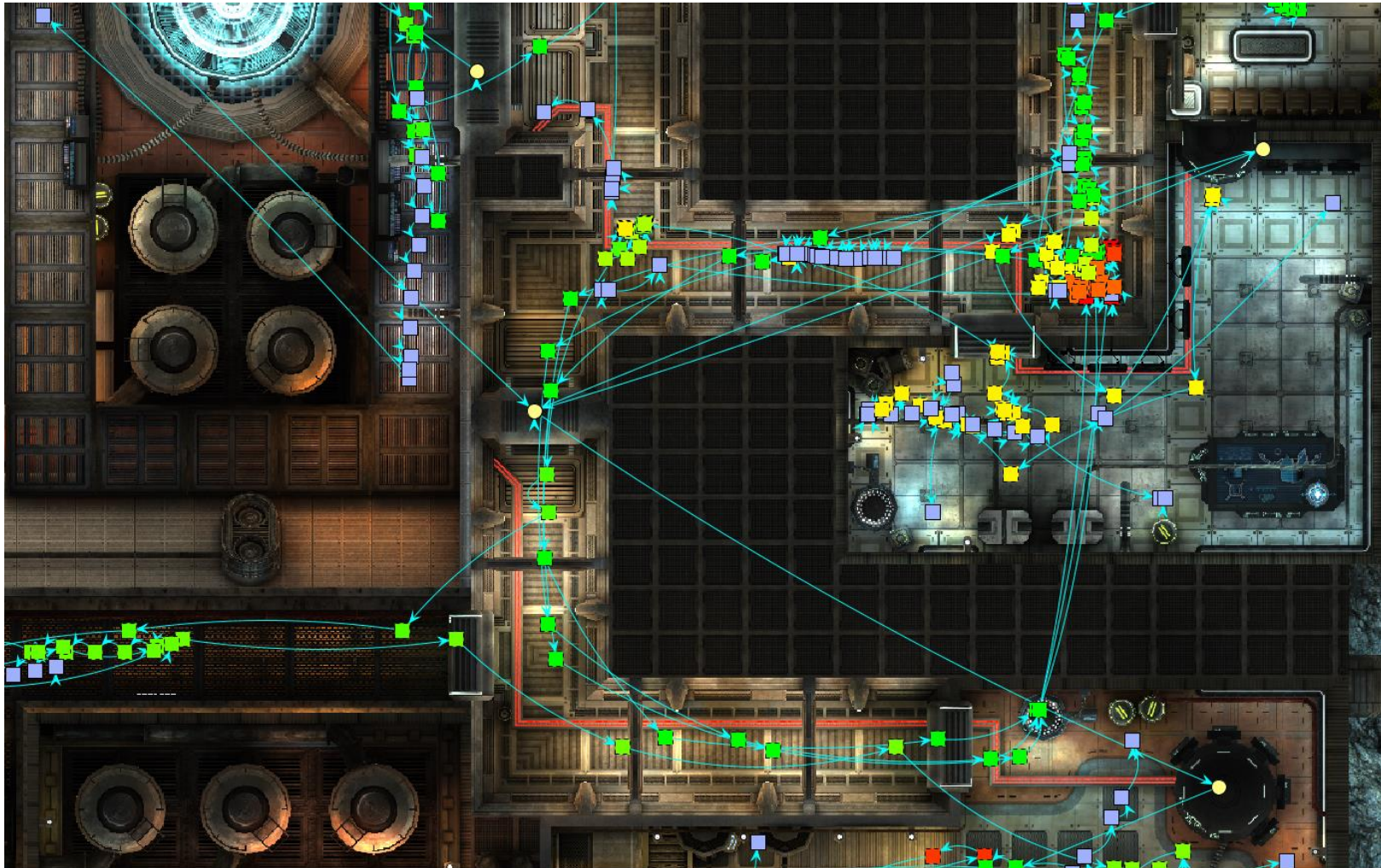


# Case Study

- Game session
  - Angry Bots from Unity
- Graph
  - Vertices: 1275
  - Edges: 2976
- Vertices represent actions (Activity), objects (Entity), and agents



# Zoom



# 1st Combat



# 2nd Combat



# 3rd Combat

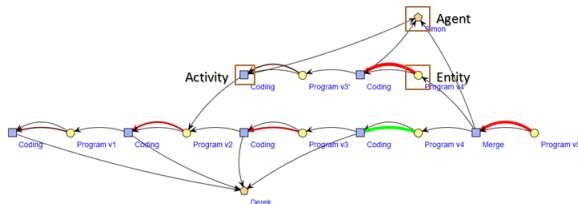




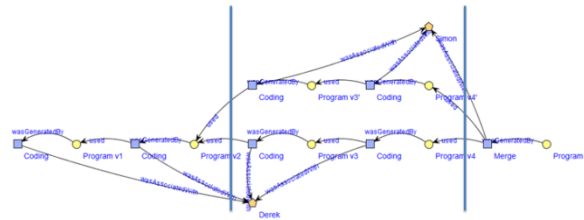
# Conclusion

## Shapes and Colors

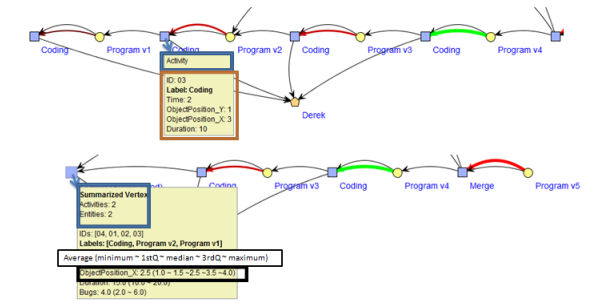
- PROV shapes
  - Pentagon: Agent
  - Circle: Entity
  - Square: Activity
- Default PROV colors
  - Color Schema
  - Traffic light color based on metadata information



## Temporal Filter

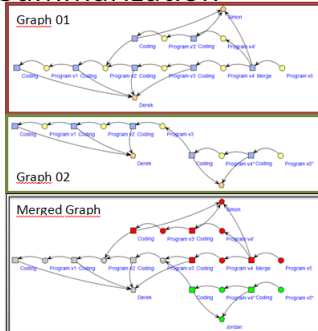


## Vertex Metadata

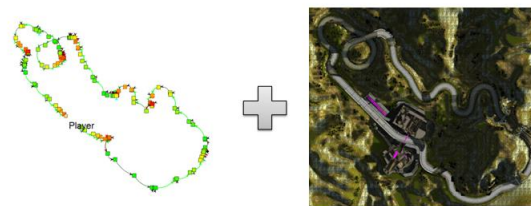


## Automatic Summarization

- Combine similar vertices
  - Compare vertices
  - Detect similarity
  - No information loss
- Similarity
  - Vertex type
  - Attributes
  - Values
- Between graphs
  - Graph merge
  - Analyze multiple trials or sessions
- Within same graph
  - Sequential vertices
  - Deduplication



## Spatial referencing data

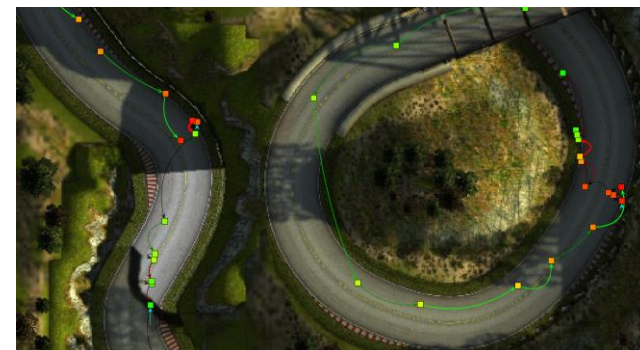
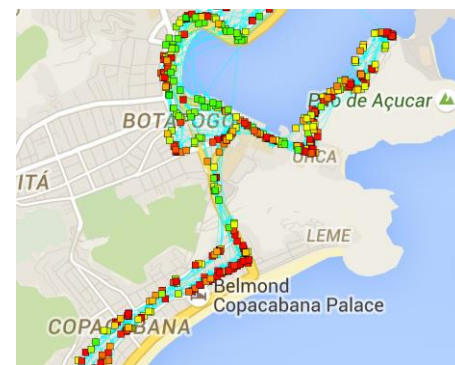


## Zoom



# Conclusion

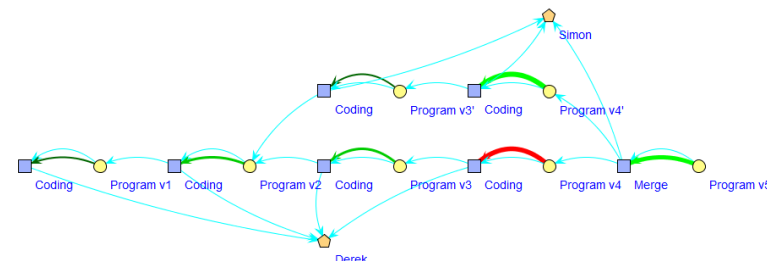
- Future work
  - Improved summarization techniques
  - Explore more visualization features and layouts
  - Tool optimization
  - Improved graph merge heuristic



- Acknowledgements



The authors thank CAPES, CNPq, and FAPERJ for the financial support.



# Prov Viewer: a graph-based visualization tool for interactive exploration of provenance data

<https://github.com/gems-uff/prov-viewer/wiki>



Troy C. Kohwalter<sup>1</sup>  
Thiago Oliveira<sup>1</sup>, Juliana Freire<sup>2</sup>, Esteban W. G. Clua<sup>1</sup>, Leonardo G. P. Murta<sup>1</sup>

<sup>1</sup>Instituto de Computação – Universidade Federal Fluminense (UFF)

<sup>2</sup>New York University

