

**Major Qualifying Project
Worcester Polytechnic Institute &
Massachusetts Institute of Technology Lincoln Laboratory**

Scalable Light-Weight Integrated Utility Engine (SLIQUE)

**Michael Molignano, Timothy Navien
and Steven Shidlovsky**

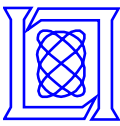
14 October 2009

Group 39

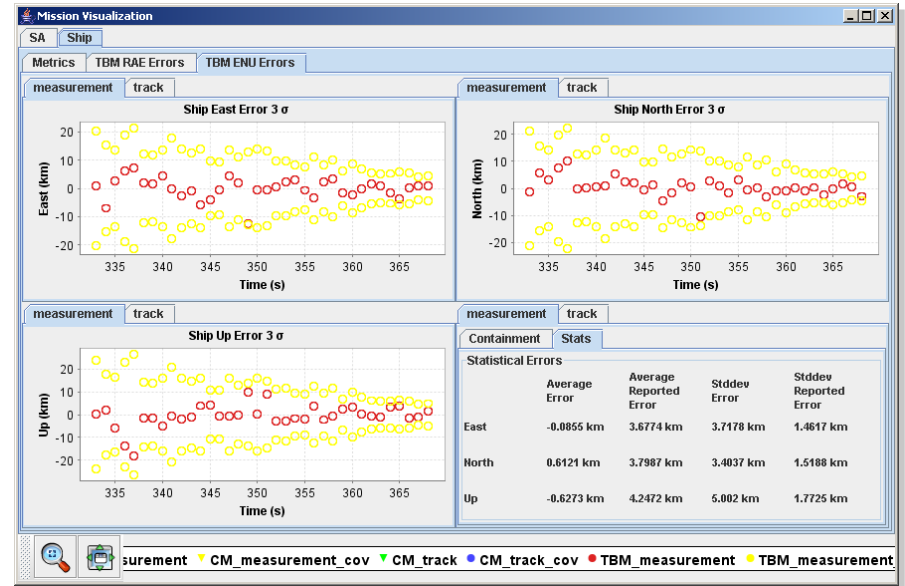
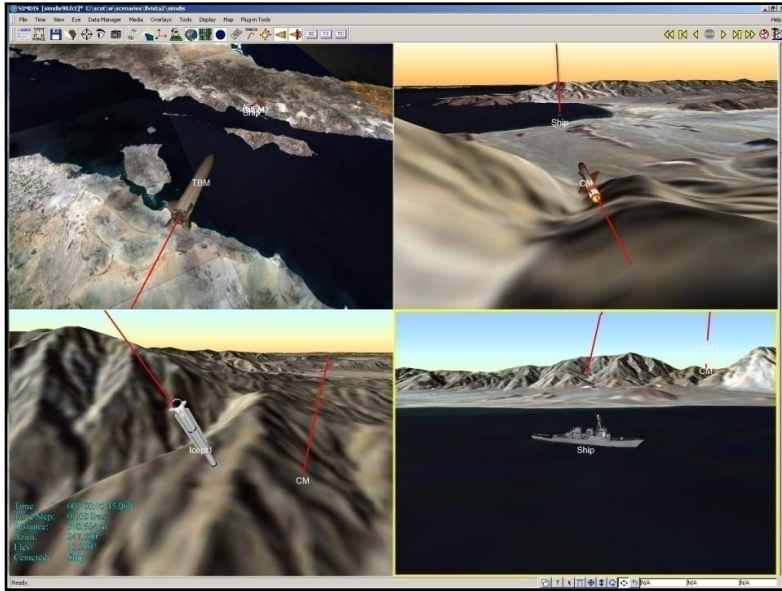
**WPI Advisor:
Professor George Heineman**

**MIT LL Advisors:
Mr. Scot DeDeo
Dr. David Holl**

MIT Lincoln Laboratory



Motivation

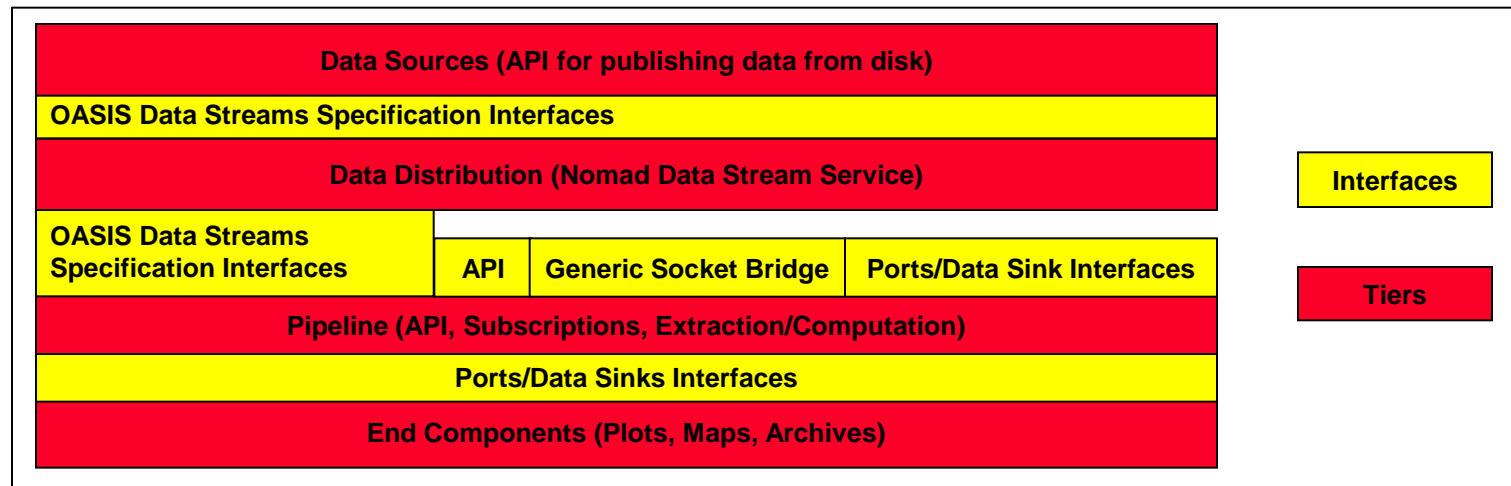


- Need to analyze live and recorded domain-specific mission data
- Produce real-time displays & quick-look/analysis products



Background

- **Lincoln Laboratory Visual Interface and Scalable Transport API (LLVISTA)**
 - Designed to support domain specific mission data
 - Expandable and flexible
 - 4 Tier architecture

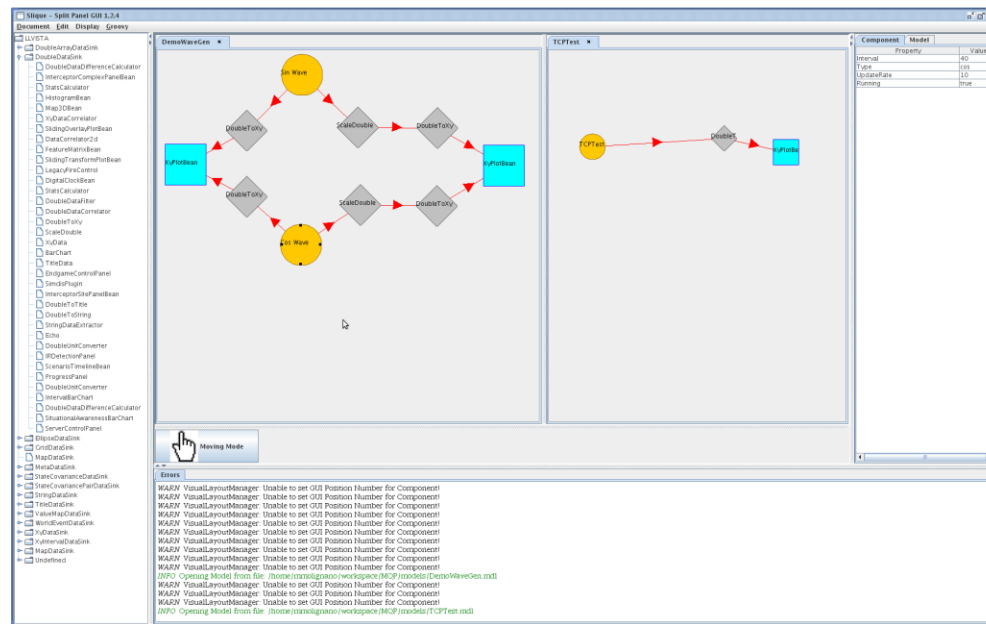


- **Constraints**
 - Requires extensive knowledge of Java and Groovy scripting engine
 - Requires understanding of underlying framework



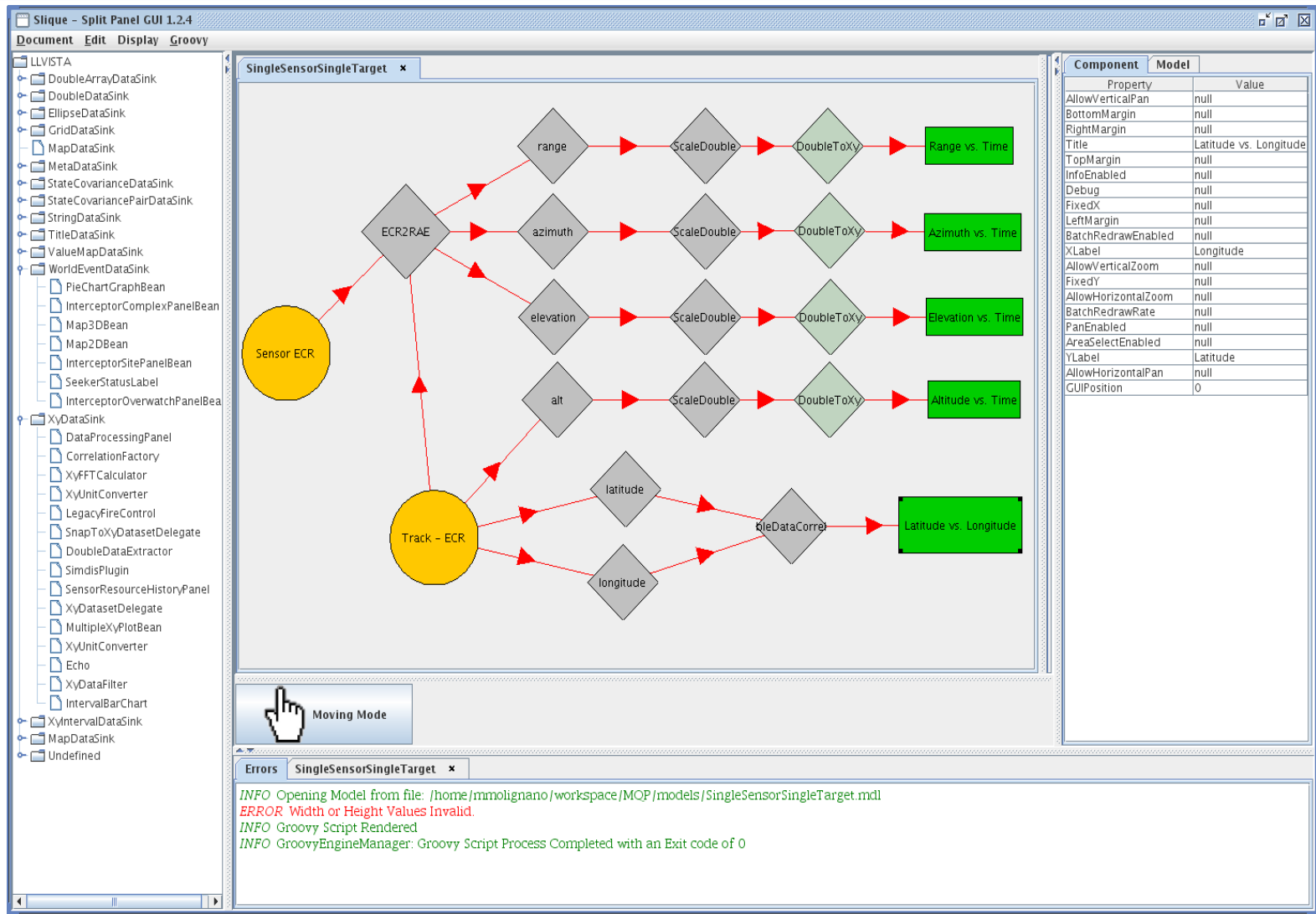
Project

- **SLIQUE**
 - Developed a visual design tool
 - Create and execute mission data processing pipelines
 - Uses drag & drop interface with LLVISTA components and connections
 - Provides in-line validation
 - Auto-discovery of components to grow with LLVISTA
 - Reduces required knowledge of Java and LLVISTA

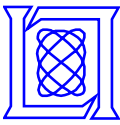




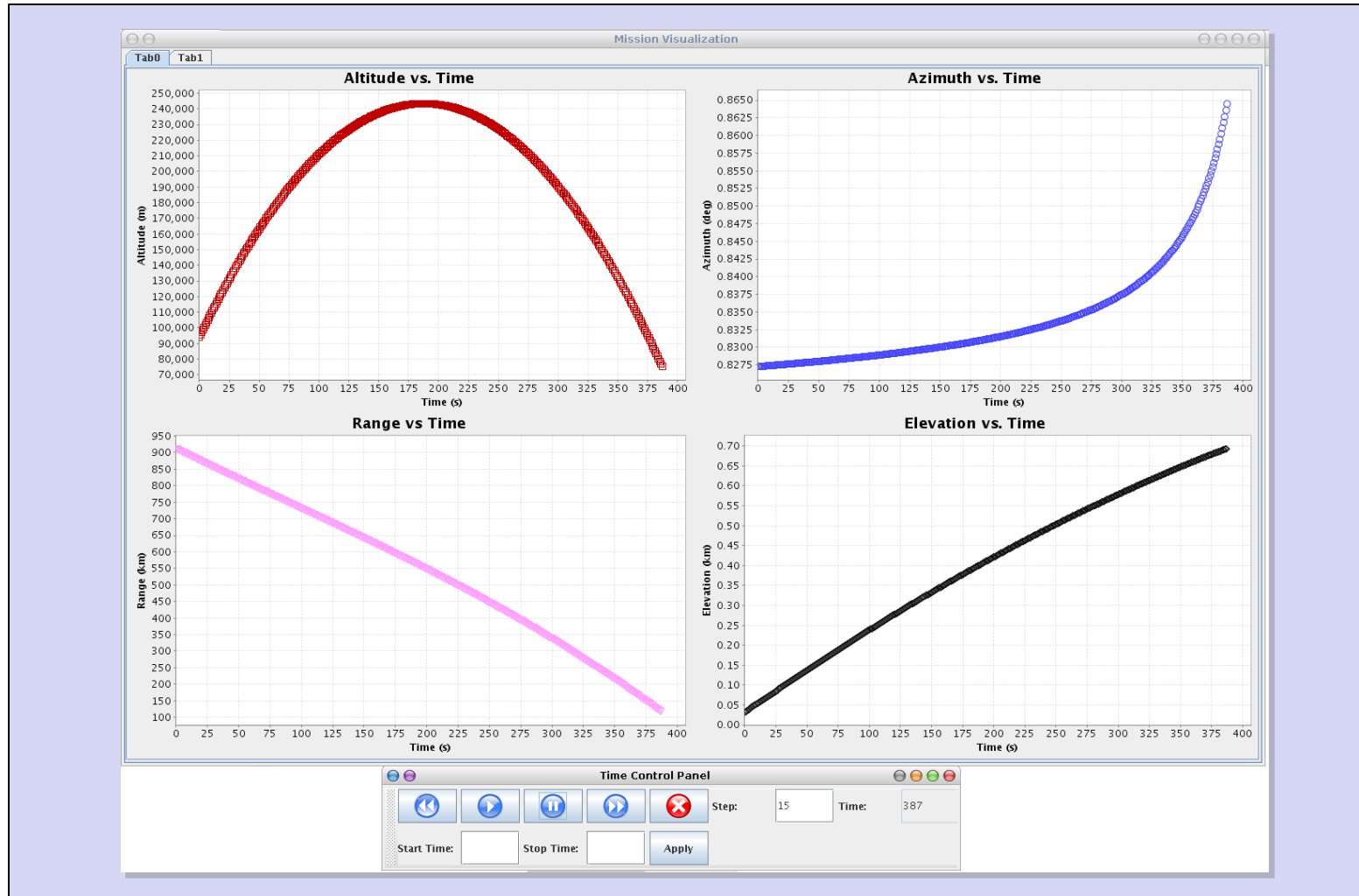
System in Use



Example Mission Data Processing Pipeline Using SLIQUE



System in Use



Rendered Configuration from SLIQUE



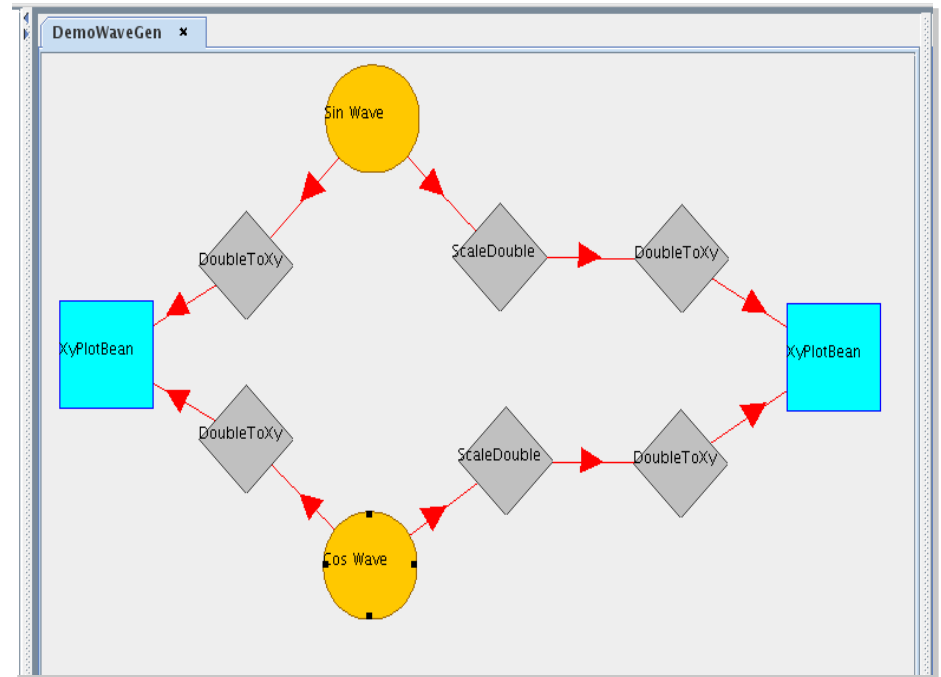
Methodology

- **Software engineering principles**
 - **Unified Modeling Language**
 - **Iterative development**
 - **Source Code Control**
 - **Documentation**
 - **Code reviews**
 - **Feature/Code freeze dates**
 - **Validation of code through testing**
- **Conformance reporting of LLVISTA components**



Design Decisions

- **Model-View-Controller**
 - Allows modeling pipeline internally
 - Allows for graphical connection to display model
 - Easy rendering of model through groovy
 - Supports testing of model and controllers
 - Allows checking conformance of LLVISTA components





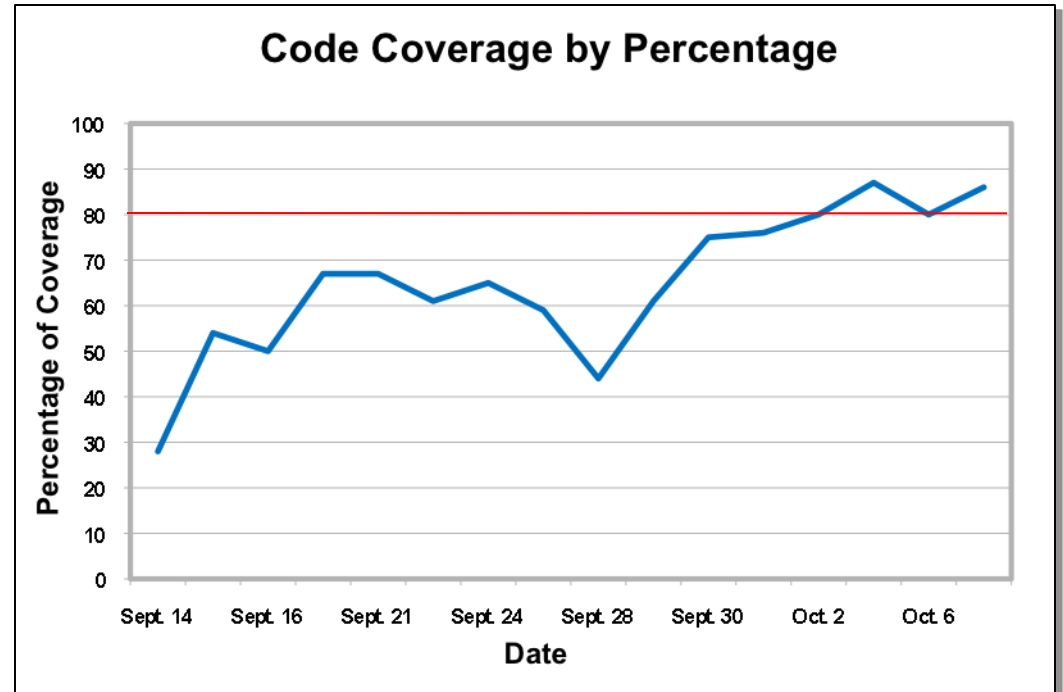
Challenges

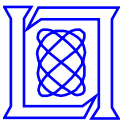
- **Restricted development environment**
- **Java Swing development**
- **Merging LLVISTA details**
- **LLVISTA component standardization**
- **Groovy engine difficulties**



Evaluation

- **Code coverage through JUnit testing**
- **Code review**
 - Singular and Team
- **User testing from multiple group**
39 members
 - Incorporated Feedback throughout Project
- **Comparison of manual vs SLIQUE-generated scripts**





Results

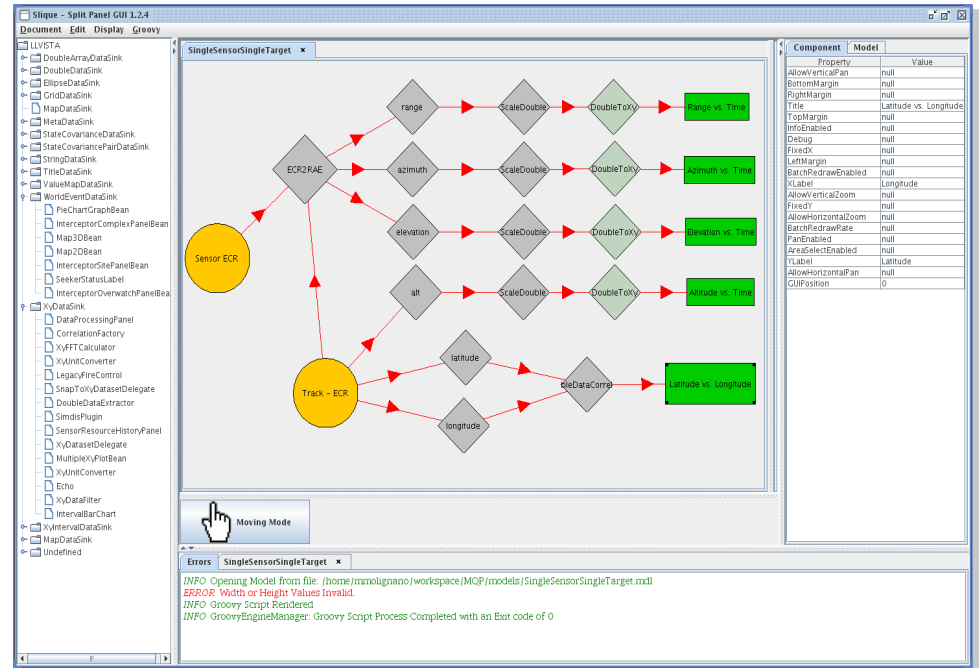
Script Using Current Method

```

116 ad = AdvertisementUtilities.createStateAdvertisement("track", "CM", "truth", "ecr");
117
118 // build the event messages
119 firstTimeMessage = ["type":"truth", "event_type":"new_track", "track":"CM",
120                    "coordinate-frame":"ecr", "object_name":"CM_truth",
121                    "mark_name":"black_triangle-up"];
122
123 lastTimeMessage = ["type":"truth", "event_type":"lost_track", "track":"CM"];
124 def cmSource = new ASCIIStateDataStreamSource(CM_Truth_File_Name, ad, 9, 0);
125 cmSource.setFirstTimeEventMessages([firstTimeMessage]);
126 cmSource.setLastTimeEventMessages([lastTimeMessage]);
127 cmSource.interpolator = new LinearDataRateInterpolator();
128 cmSource.interpolator.interpolateRate = 1;
129 cmSource.interpolator.startOnFirstRoundNumber = true;
130
131 /*** Truth Source for Interceptor 1 ***/
132 ad = AdvertisementUtilities.createStateAdvertisement("interceptor", "Icept1", "truth", "ecr");
133
134 // build the event messages
135 firstTimeMessage = ["type":"truth", "event_type":"new_interceptor", "interceptor":"Icept1",
136                    "coordinate-frame":"ecr", "object_name":"Icept1_truth",
137                    "mark_name":"black_triangle-down"];
138 lastTimeMessage = ["type":"truth", "event_type":"lost_track", "interceptor":"Icept1"];
139 def icept1Source = new ASCIIStateDataStreamSource(ICEPT1_Truth_File_Name, ad, 9, 0);
140 icept1Source.setFirstTimeEventMessages([firstTimeMessage]);
141 icept1Source.setLastTimeEventMessages([lastTimeMessage]);
142
143 /*** Truth Source for Interceptor 2 ***/
144 ad = AdvertisementUtilities.createStateAdvertisement("interceptor", "Icept2", "truth", "ecr");
145
146 // build the event messages
147 firstTimeMessage = ["type":"truth", "event_type":"new_interceptor", "interceptor":"Icept2",
148                    "coordinate-frame":"ecr", "object_name":"Icept2_truth",
149                    "mark_name":"black_cross"];
150 lastTimeMessage = ["type":"truth", "event_type":"lost_track", "interceptor":"Icept2"];
151 def icept2Source = new ASCIIStateDataStreamSource(ICEPT2_Truth_File_Name, ad, 9, 0);
152 icept2Source.setFirstTimeEventMessages([firstTimeMessage]);
153 icept2Source.setLastTimeEventMessages([lastTimeMessage]);
154
155 /*** Ship Measurements on TBM ***/
156 ad = AdvertisementUtilities.createStateCovarianceAdvertisement(
157     "Ship", "TBM", "measurement", "rae");
158
159 //build the event messages
160 firstTimeMessage = ["type":"measurement", "event_type":"new_object_data", "sensor":"Ship", "track":"TBM"];

```

Script Creation Using Design-Tool



- Enables use of LLVISTA by more people



Future Work

- **Future Improvements**
 - **Improve interaction with non-standardized components**
 - **On-the-fly components**
 - **Advanced GUI layout with additional GUI components**
 - **Per-component documentation**



Acknowledgments

We would like to thank the following for their guidance and assistance throughout the project:

Mr. Scot DeDeo

Professor George Heineman

Dr. David Holl

Mr. Joe Georger

Mr. Dave Oostdyk

Professor Ted Clancy

Mr. Vivek Varshney

Group 39



Questions?
