Model-based Design and Implementation of Distributed Real-time Embedded Systems in Cadena

http://www.cis.ksu.edu/cadena

Gurdip Singh
John Hatcliff
Matt Dwyer

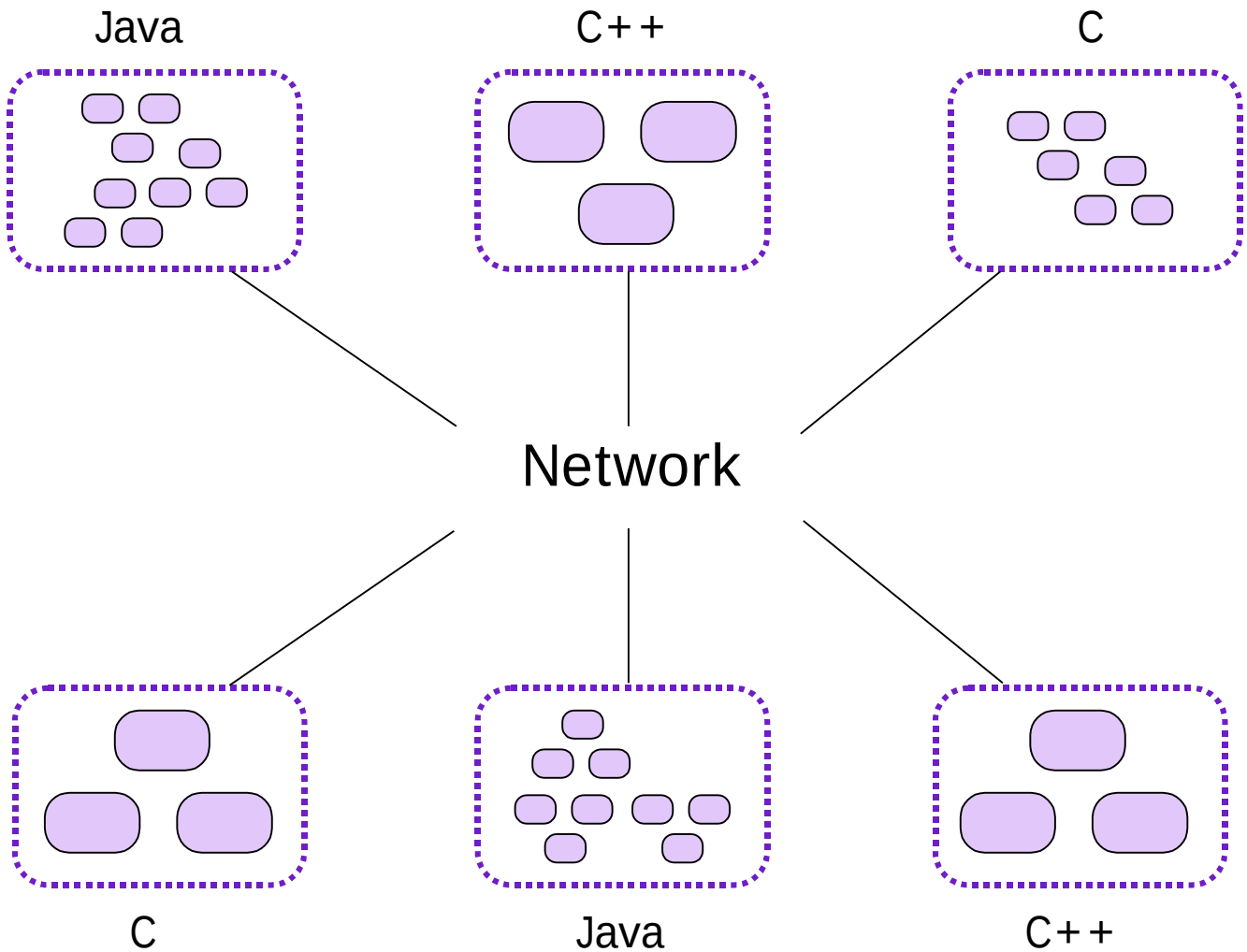
Venkatesh Ranganath
Xianghua Deng
Prashant Shanti Kumar
Qiang Deng

Support

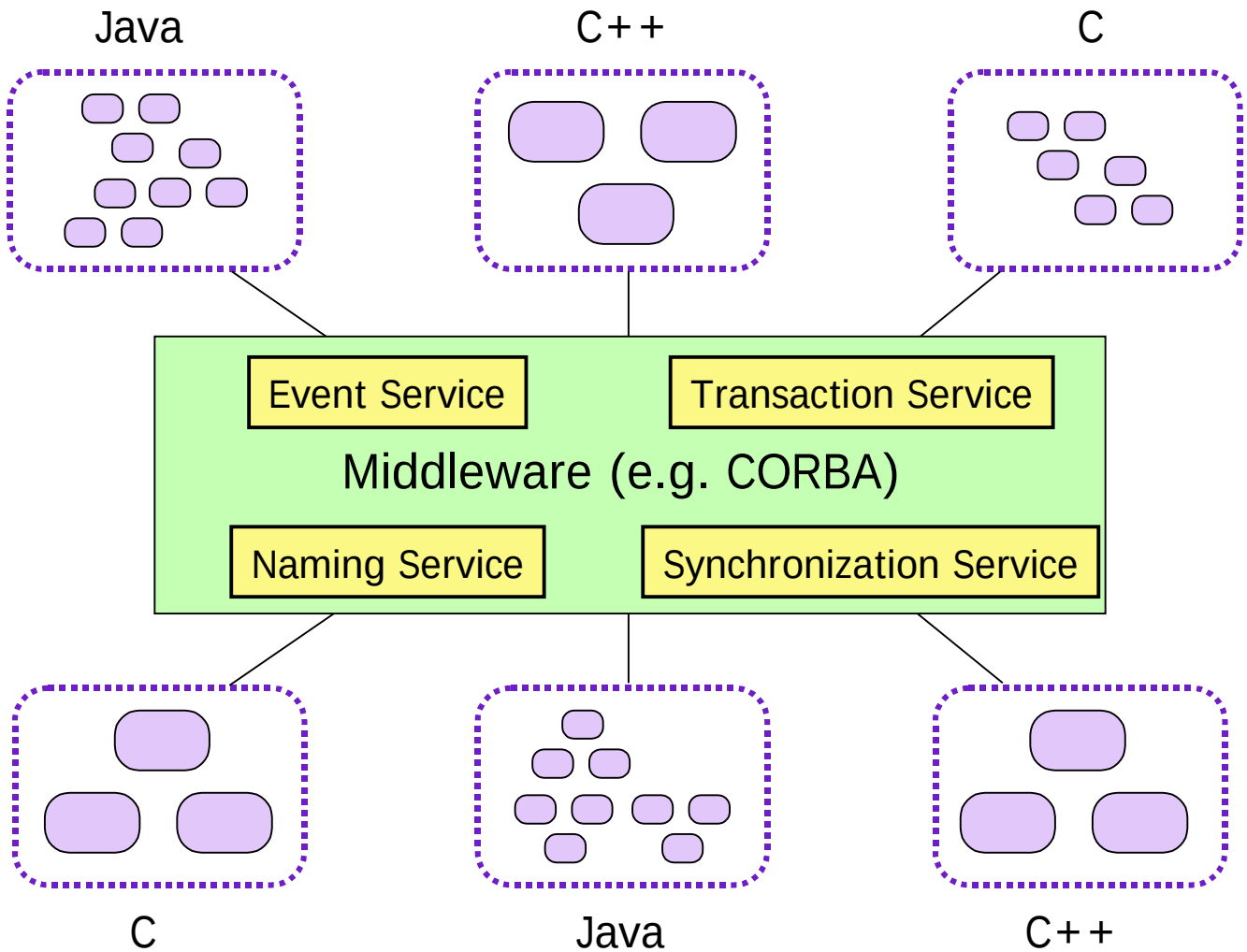
US Army Research Office (ARO)
US National Science Foundation (NSF)
US National Aeronautics and Space Agency (NASA)
US Department of Defense
Advanced Research Projects Agency (DARPA)

Rockwell-Collins ATC
Honeywell Technology Center and NASA Langley
Sun Microsystems
Intel

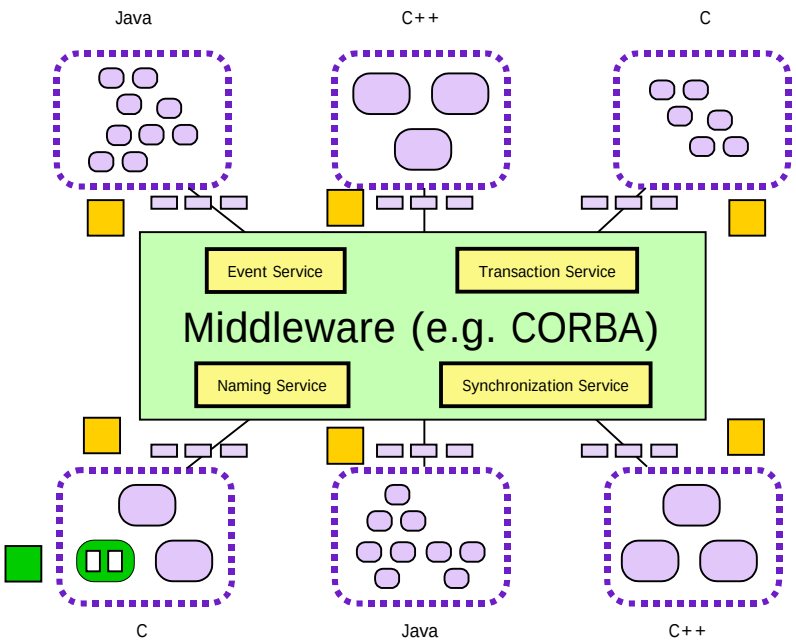
Distributed Components



Distributed Components



Real-time Embedded Systems



Issues

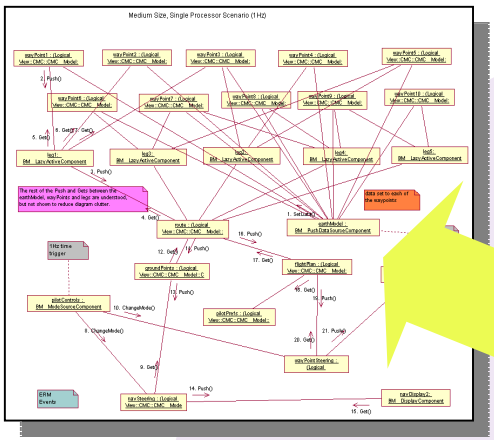
- These systems are huge!
- Extensive use of OO patterns & software layering
- What are appropriate abstractions for formal reasoning?
- How can we help developers write them?
- How to auto-generate aspects?
- How can we configure middleware?

Example DRE Systems

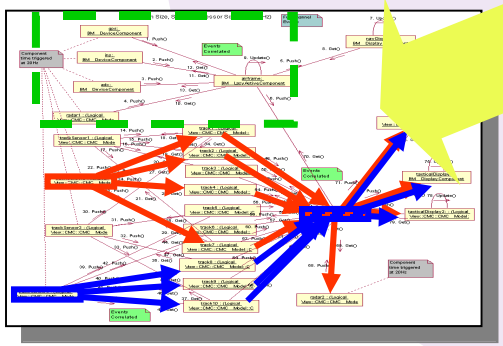
- Mission-control software for Boeing military aircraft
- Boeing's Bold Stroke Avionics Middleware
  - ...built on top of ACE/TAO RT CORBA

- Provided with an Open Experimental Platform (OEP) from Boeing
  - a sanitized version of the real system
  - 100,000+ lines of C++ code (including RT CORBA middleware)
  - 50+ page document that outline development process and describe challenge problems
- Must provide...
  - tool-based solutions that can be applied by Boeing research team to realistic systems
  - solutions that fit within current development process, code base, etc.
  - metrics for that allow Boeing research team to evaluate tool performance and ease of use

Cadena – CCM Development



High-level specification of abstract component behavior



Visualization and design-level reasoning

Cadena

CCM Interface Definition Language

RT Aspect Specs

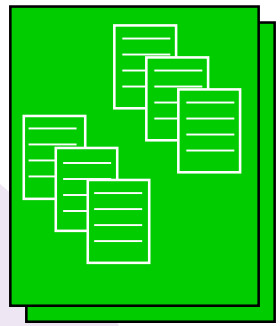
State Transitions

System Configuration

High-level Specification Language

Eclipse Plug-In

Integrated Development Environment

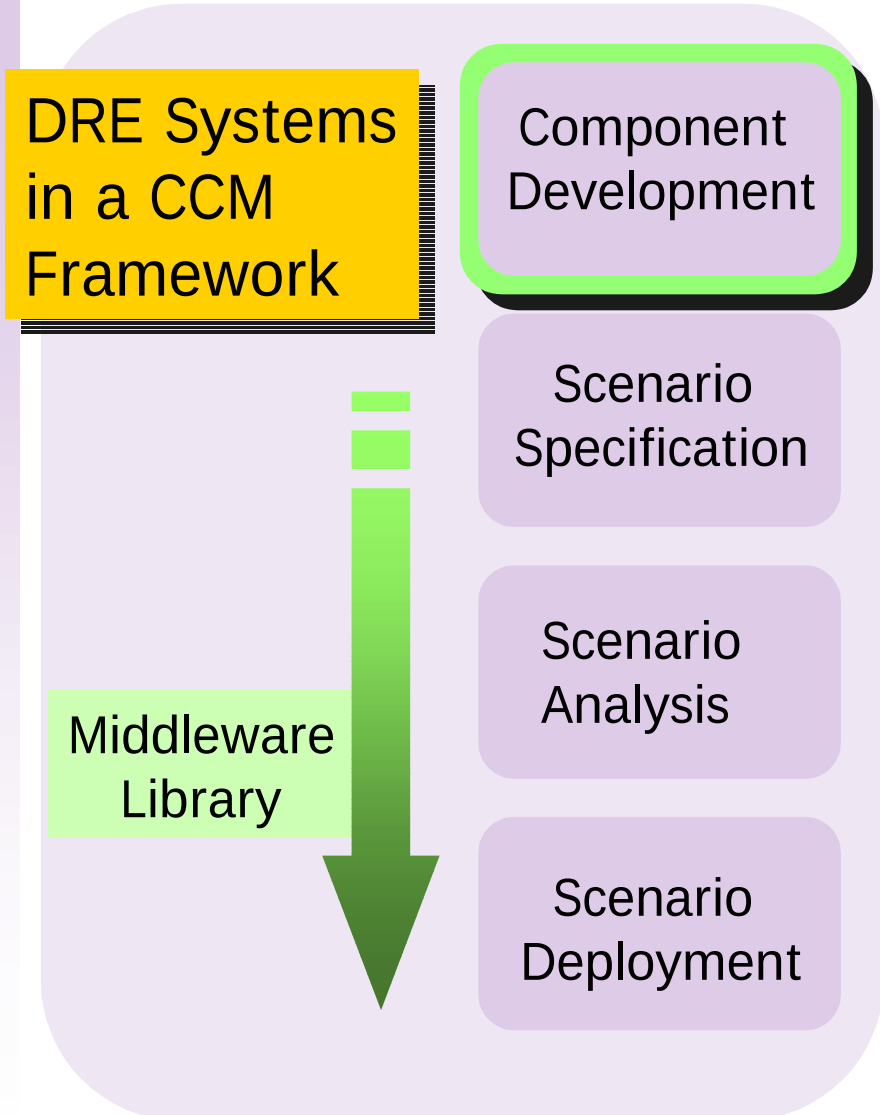


Code generation functions (via OpenCCM) produces code amenable to conformance checking and certification

```
URATION_PASS>
>          <...>
OMPONENT>
D> <...></ID>
<EVENT_SUPPLIER>
<...events this component supplies...>
</EVENT_SUPPLIER>
</COMPONENT>
</HOME>
</CONFIGURATION_PASS>
```

Configuration and Deployment information

Overall Themes

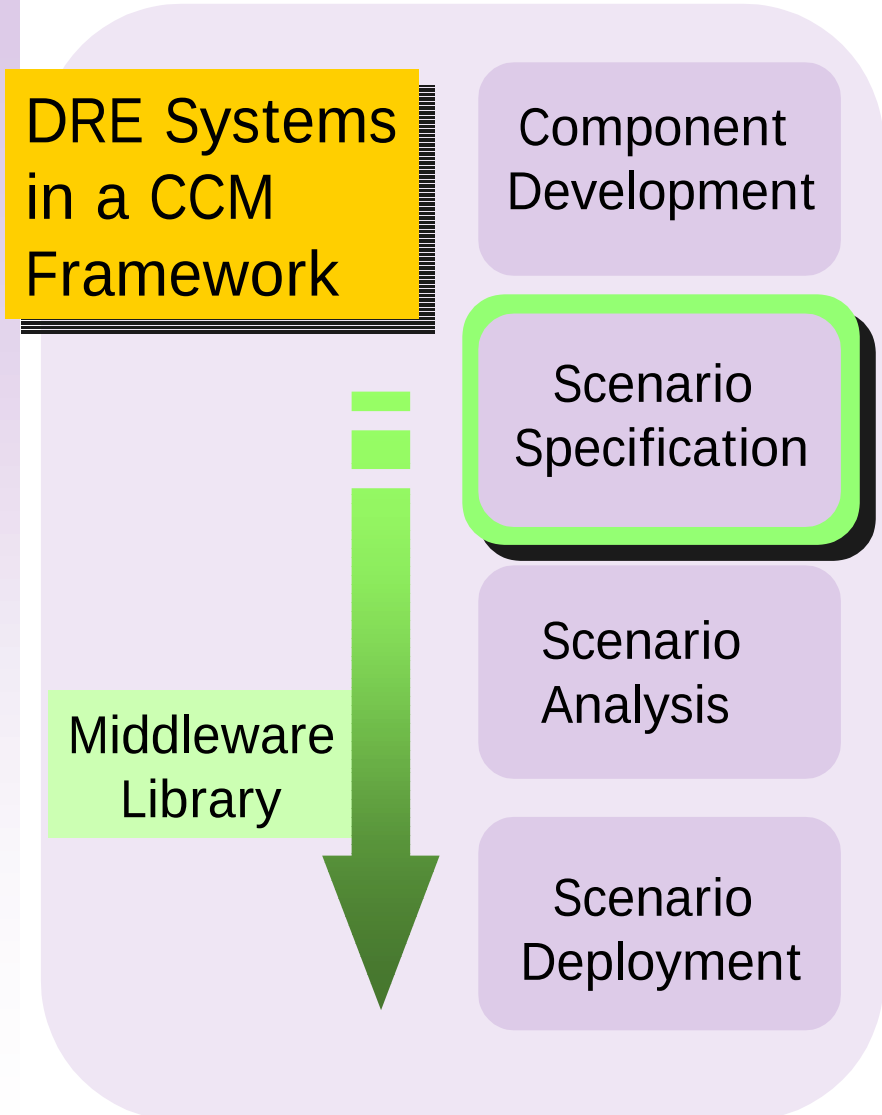


Themes

...COTS/Industry standard component models/frameworks

...Component developer's activities (interface & behavior specs) are centered at the CCM IDL level.

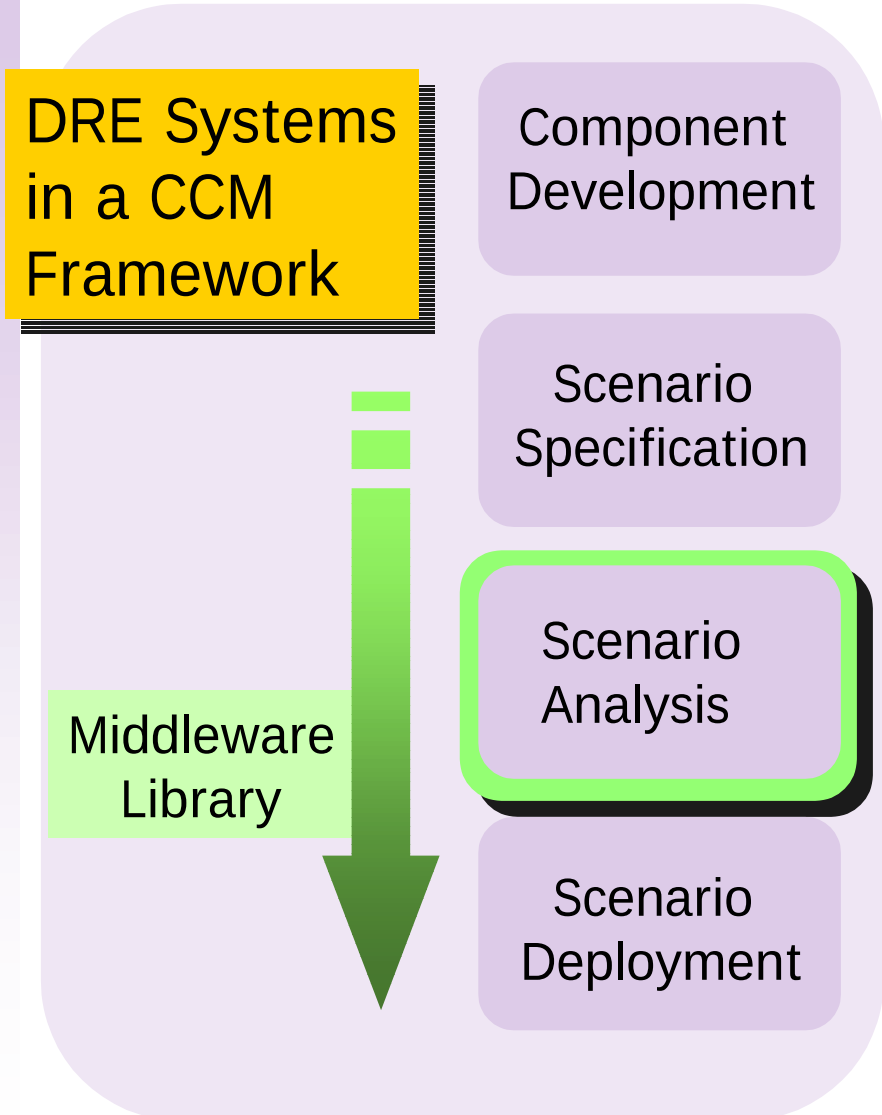
Overall Themes



Themes

...Specify and reason about component connections and dependencies independently of middleware services

Overall Themes

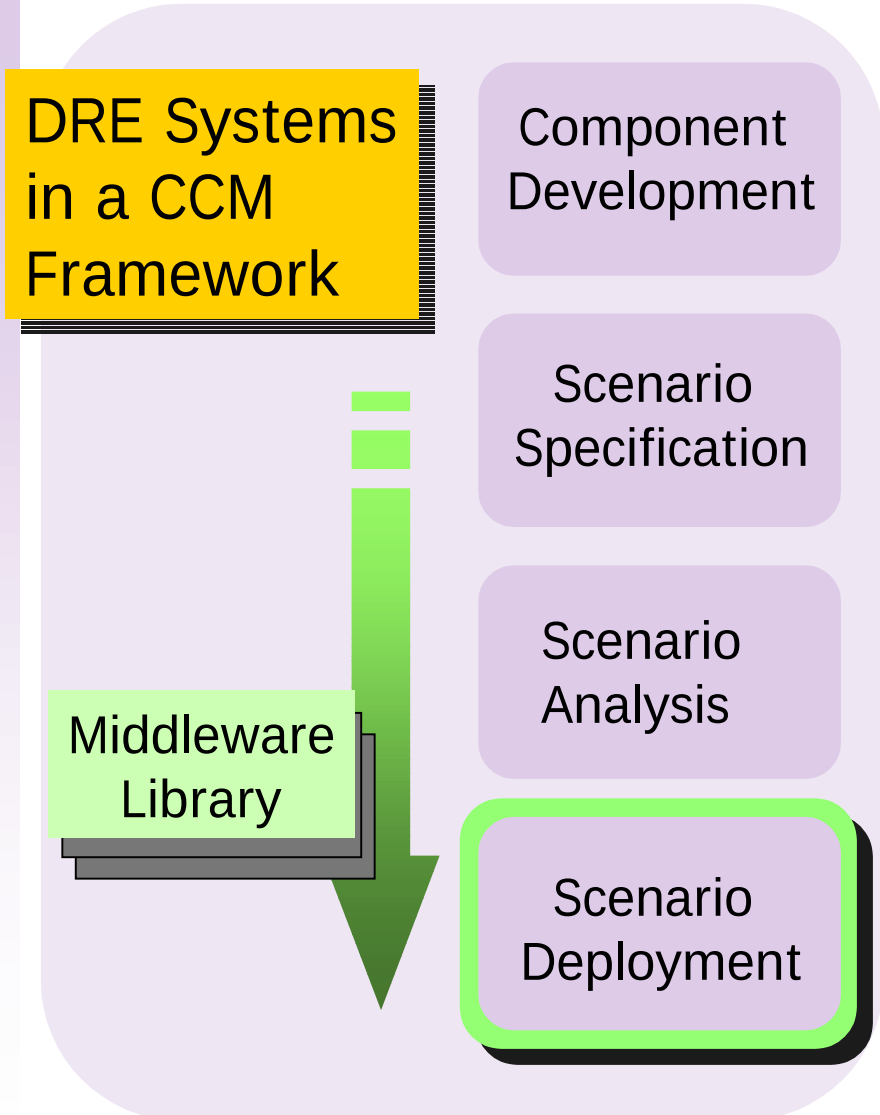


Themes

...Choose particular middleware services and synthesize QoS attributes.

...Check invariants, dependencies, and temporal properties.

Overall Themes

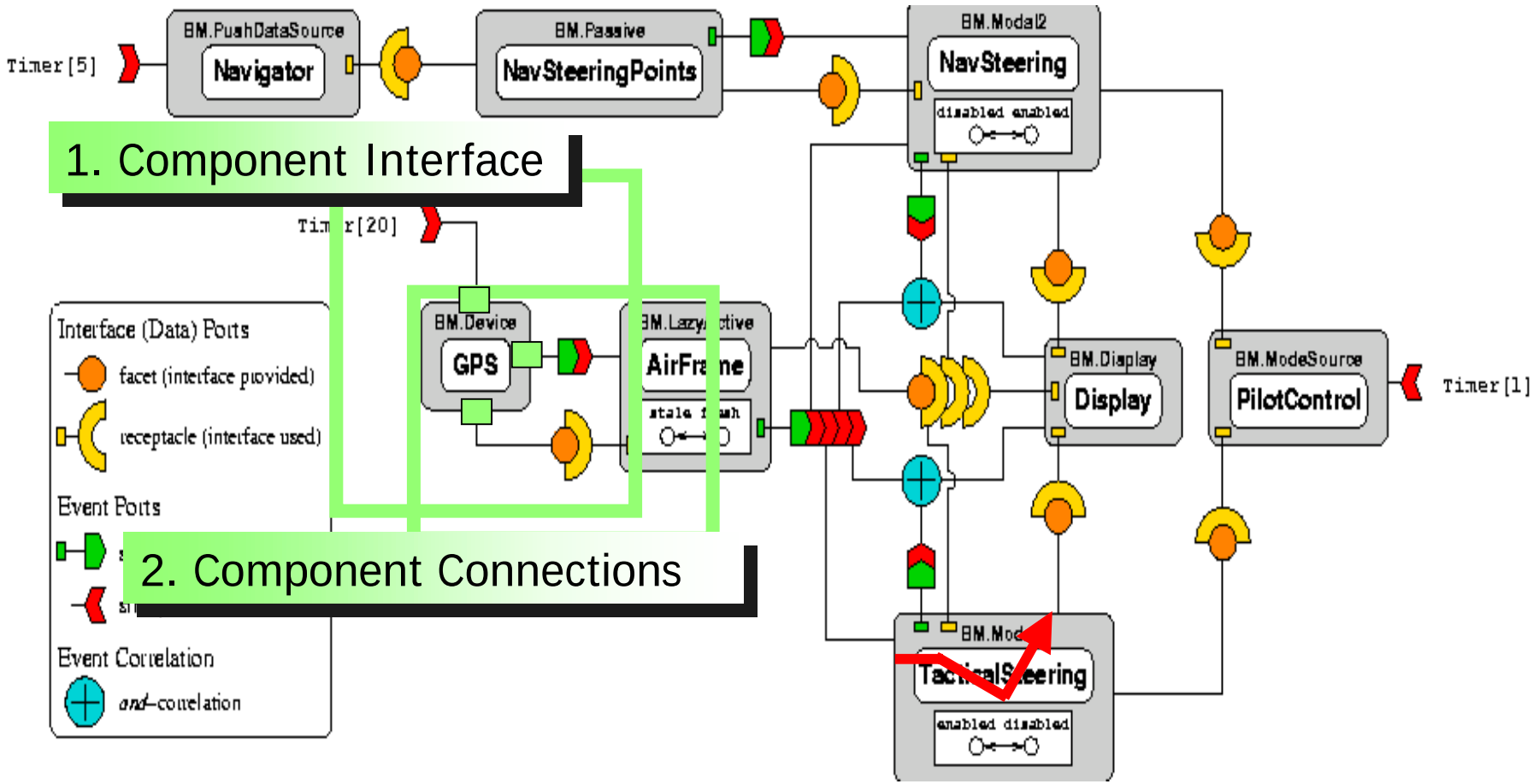


Themes

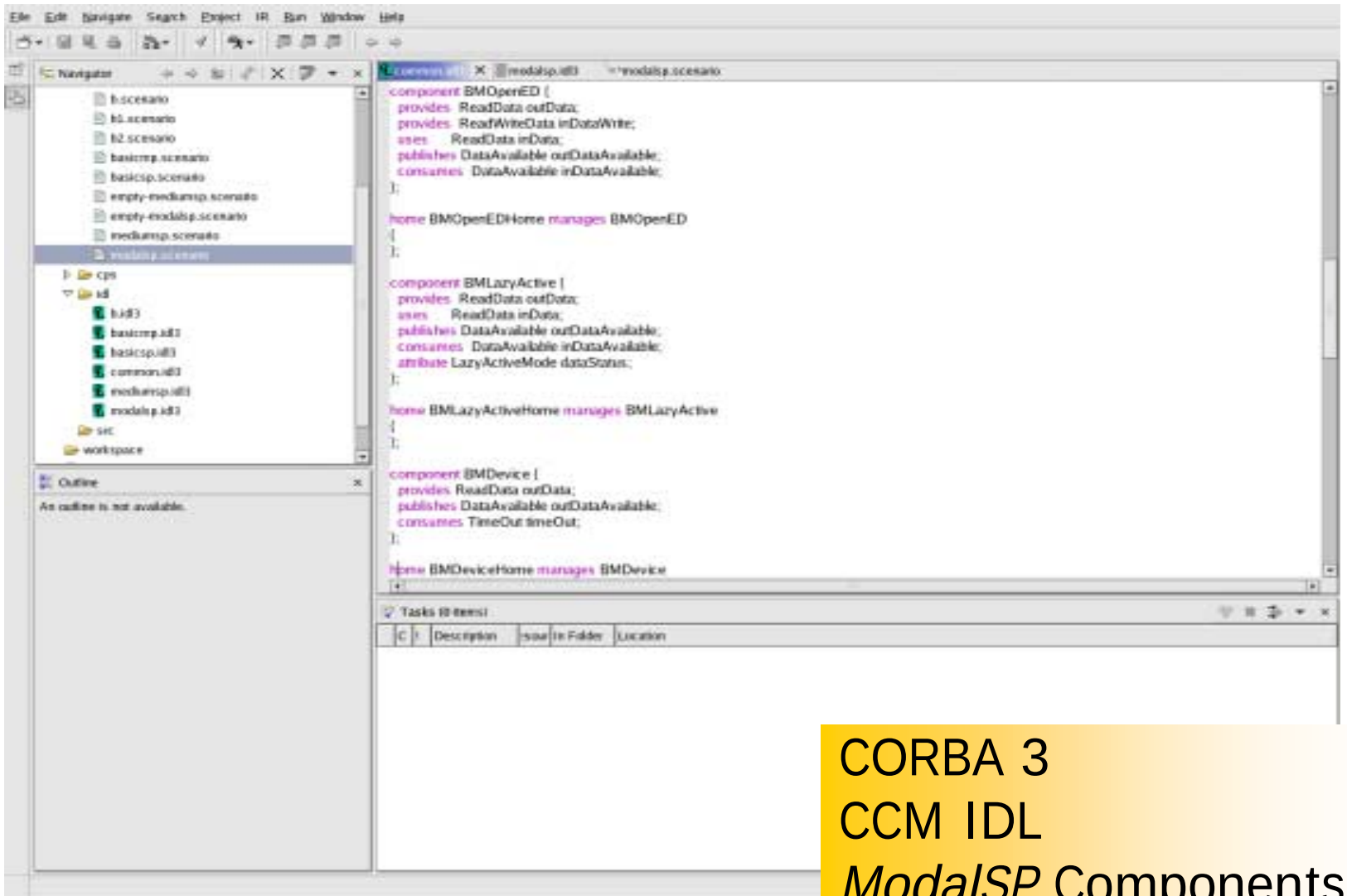
...Autogenerate configuration and component container code for chosen middleware services.

...Deploy based on deployment parameters.

Outline

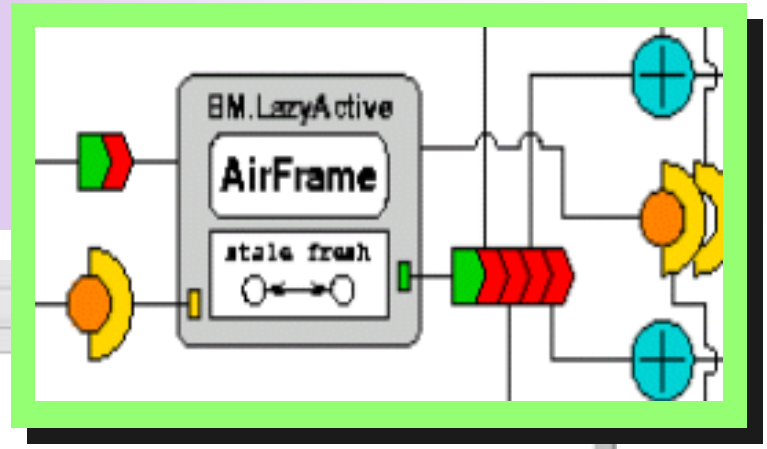

Timer[5]
BM.PushDataSource
Navigator
BM.Passive
NavSteeringPoints
BM.Modal2
NavSteering
disabled enabled
1. Component Interface
Timer[20]
Interface (Data) Ports
facet (interface provided)
receptacle (interface used)
Event Ports
Event Correlation
and-correlation
BM.Device
GPS
AirFrame
stale flush
2. Component Connections
BM.Display
Display
BM.ModeSource
PilotControl
Timer[1]
enabled disabled
3. Dependence Information

Component IDL



CORBA 3
CCM IDL
ModalSP Components

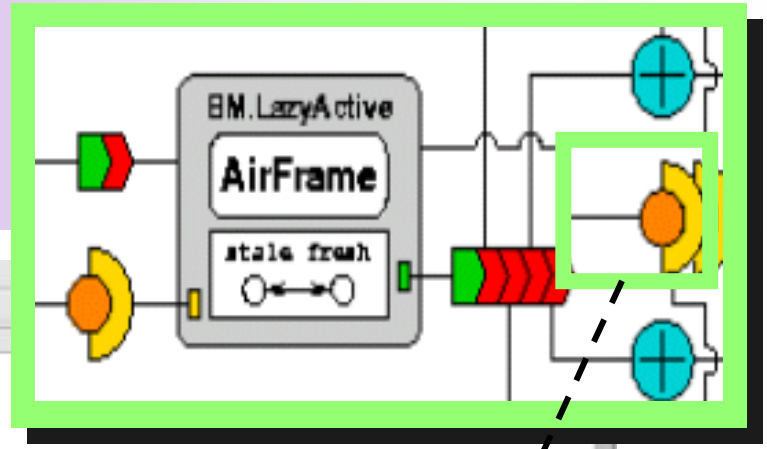
Component IDL




```
component BMLazyActive {
  provides  ReadData outData;
  uses      ReadData inData;
  publishes DataAvailable outDataAvailable;
  consumes  DataAvailable inDataAvailable;
  attribute LazyActiveMode dataStatus;
};
```

CORBA 3
CCM IDL
ModalSP Components

Component IDL

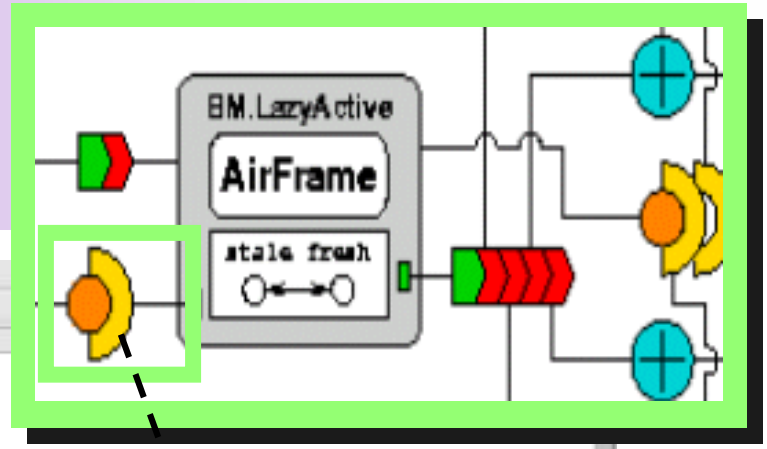



```
component BMLazyActive {
  provides  ReadData outData;
  uses      ReadData inData;
  publishes DataAvailable outDataAvailable;
  consumes  DataAvailable inDataAvailable;
  attribute LazyActiveMode dataStatus;
};
```

output data port (facet)

CORBA 3
CCM IDL
ModalSP Components

Component IDL



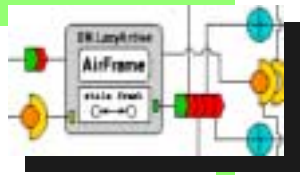

```
component BMLazyActive {
  provides  ReadData outData;
  uses      ReadData inData;
  publishes DataAvailable outDataAvailable;
  consumes  DataAvailable inDataAvailable;
  attribute LazyActiveMode dataStatus;
};
```

input data port (receptacle)

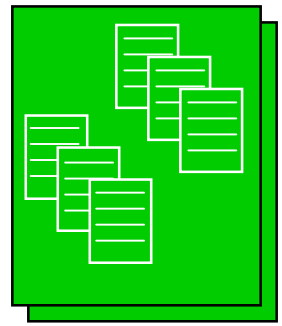
CORBA 3
CCM IDL
ModalSP Components

Leverage CORBA IDL

```
component BMLazyActive {
 provides  ReadData outData;
 uses      ReadData inData;
 publishes DataAvailable outDataAvailable;
 consumes  DataAvailable inDataAvailable;
 attribute LazyActiveMode dataStatus;
};
```



IDL Compiler



Component Implementation Stubs & Skeletons

+

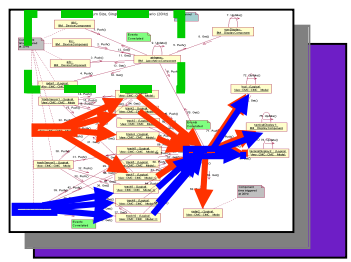
```
dependencydefault
   == none;

dependencies {
 inDataAvailable
   ->
  outDataAvailable;
}
```

Dependency Annotations

```
behavior {
 if (mode==enabled) {
  push outDataAvailable;
 else
  …
 }
```

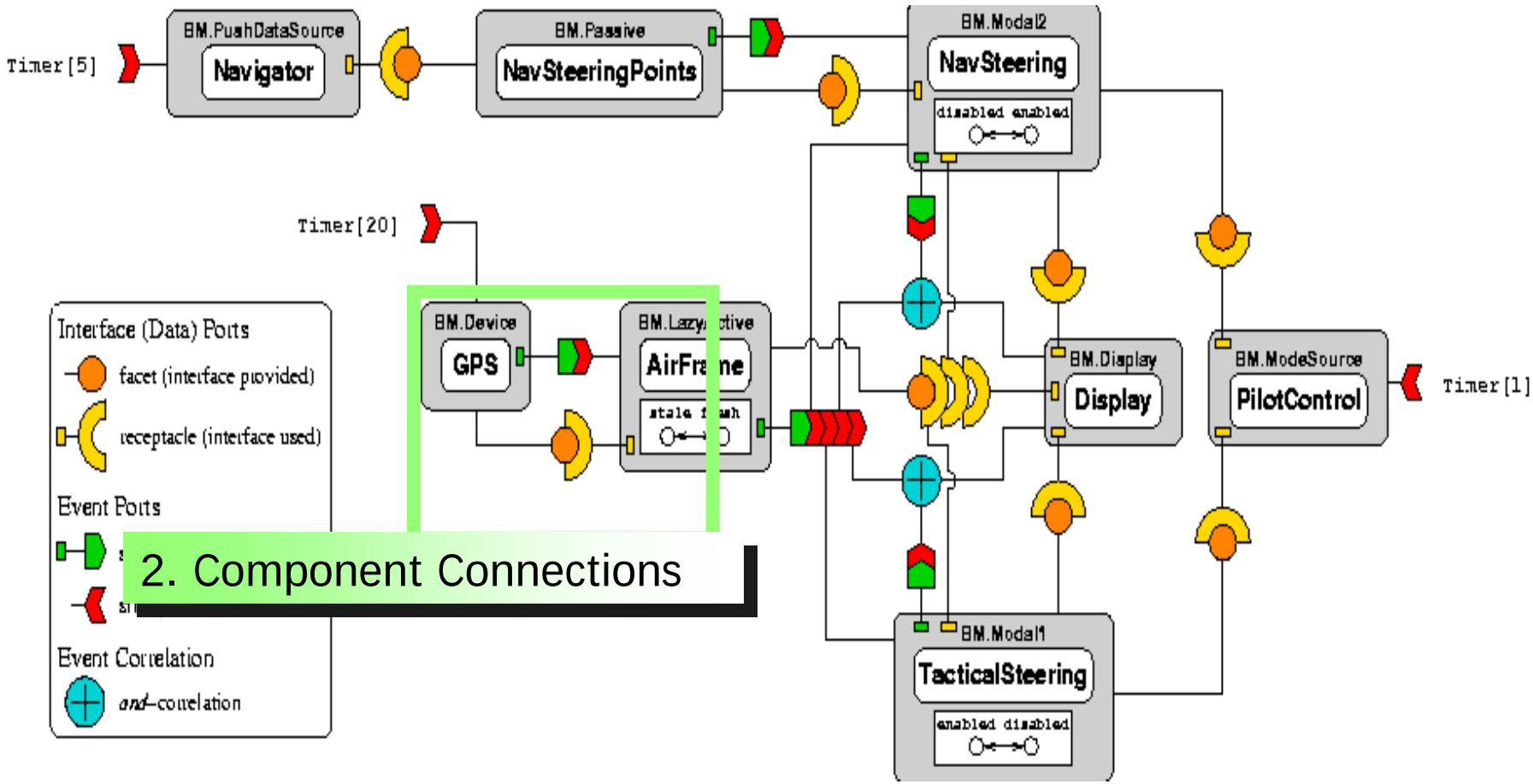
Transition System Semantics

Model Builder



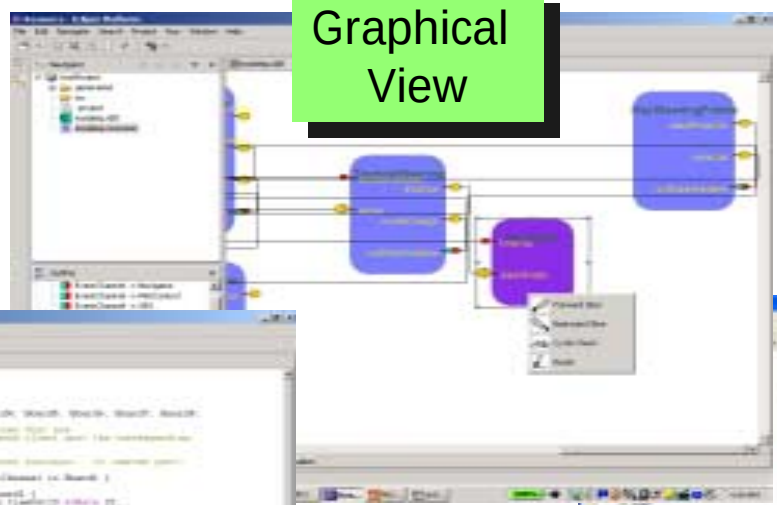
Dependency Analysis and Model-checking Engine

Outline

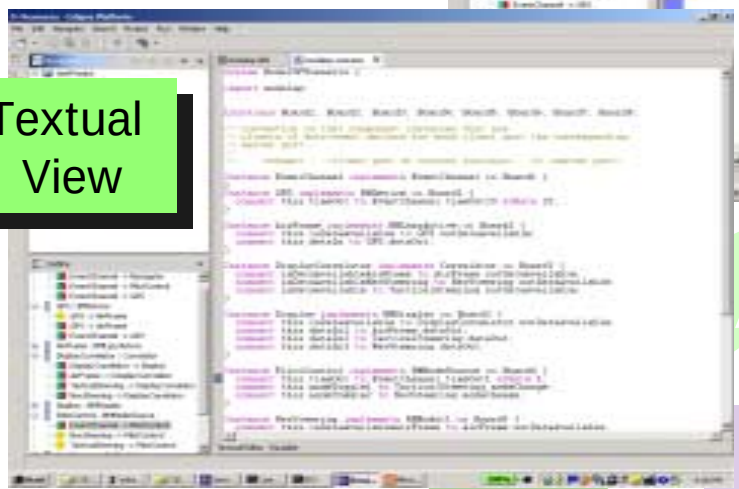

Timer[5]
BM.PushDataSource
Navigator
BM.Passive
NavSteeringPoints
BM.Modal2
NavSteering
disabled enabled
Timer[20]
BM.Device
GPS
BM.Lazy
AirFrame
stale flash
BM.Display
Display
BM.ModeSource
PilotControl
Timer[1]
BM.Modal1
TacticalSteering
enabled disabled
Interface (Data) Ports
facet (interface provided)
receptacle (interface used)
Event Ports
Event Correlation
and-correlation
2. Component Connections

Three Synchronized Views

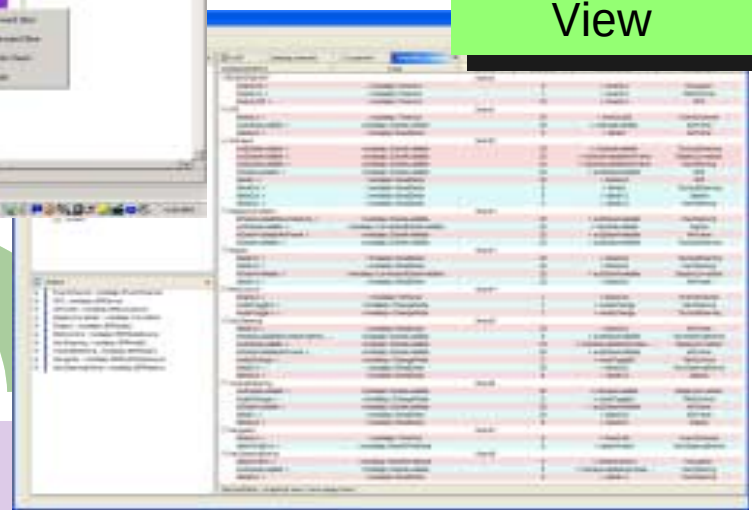
Scenario Description



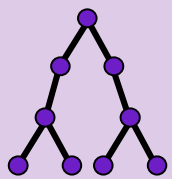
Graphical View



Textual View

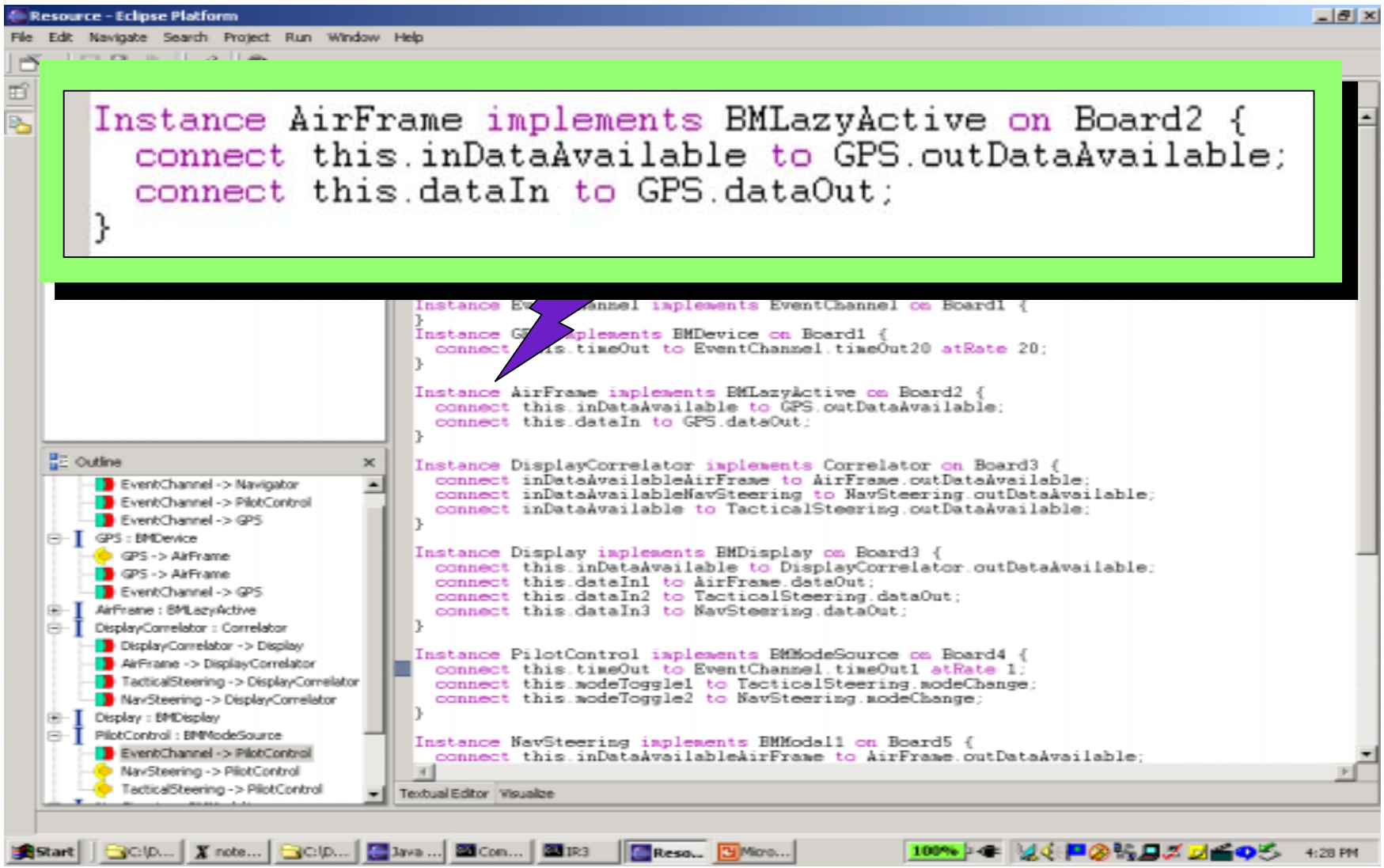


Spreadsheet View

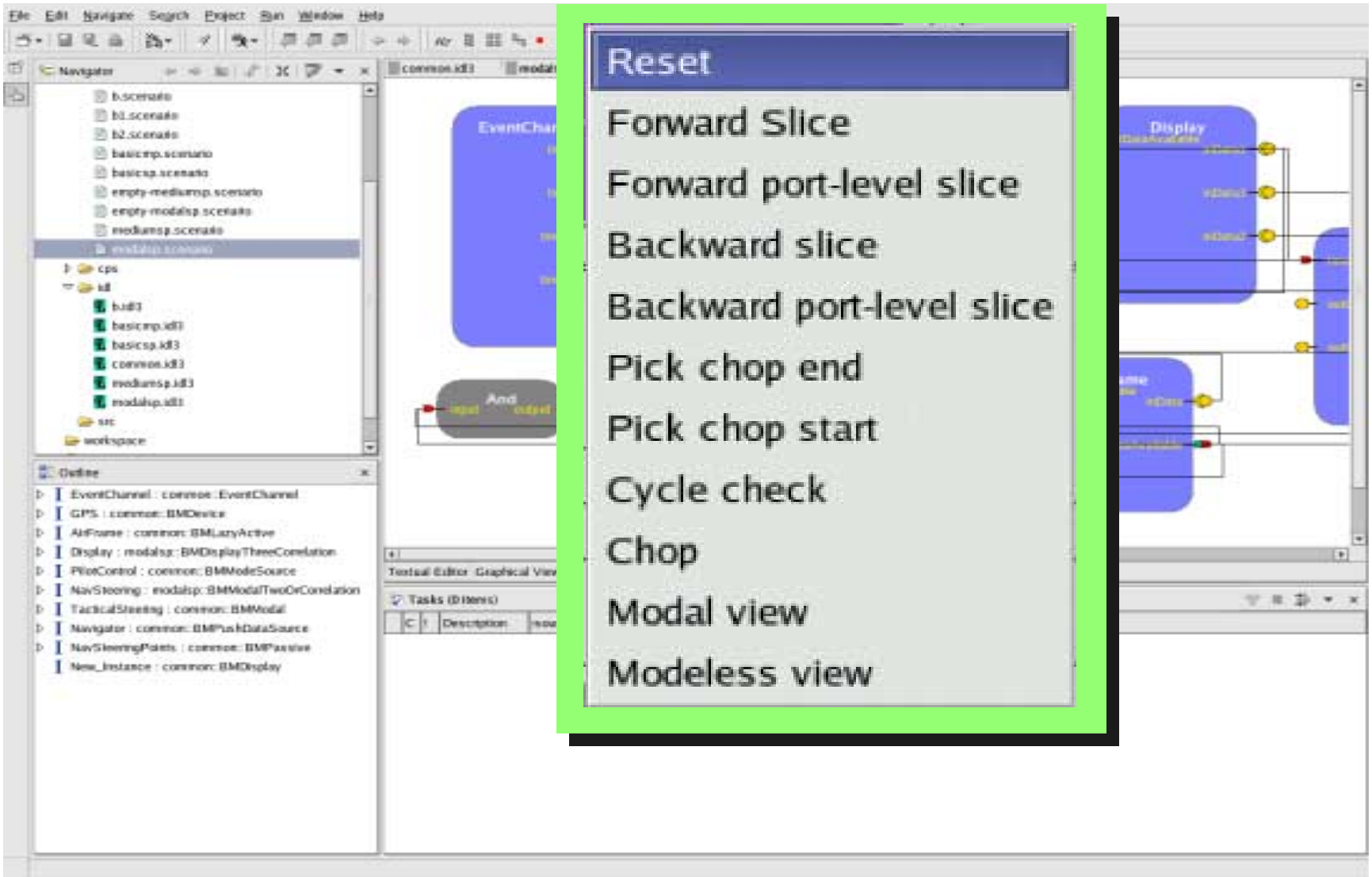


Single Internal Representation

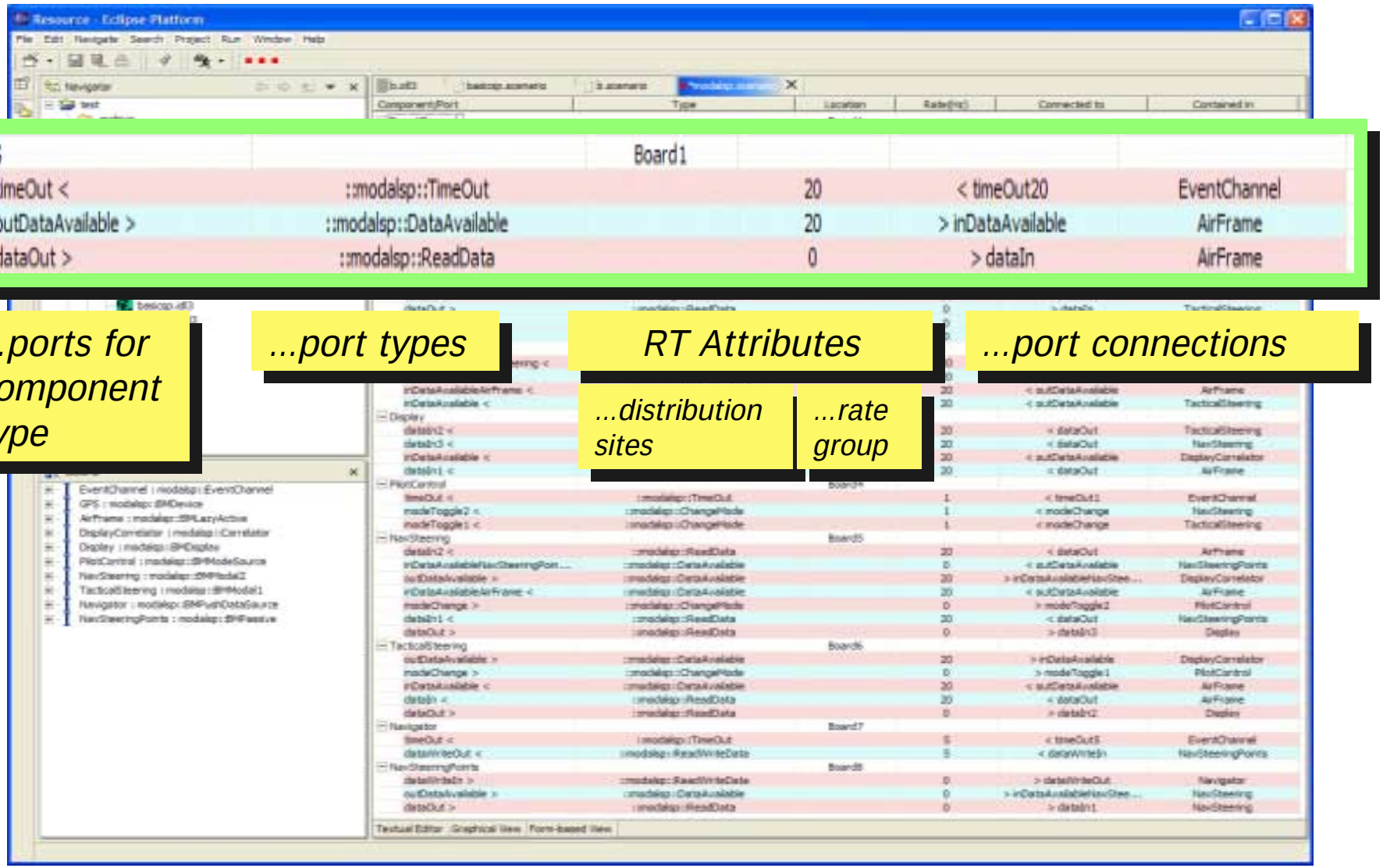
Textual View



Graphical View



Spreadsheet View



...ports for component type

...port types

RT Attributes

...distribution sites

...rate group

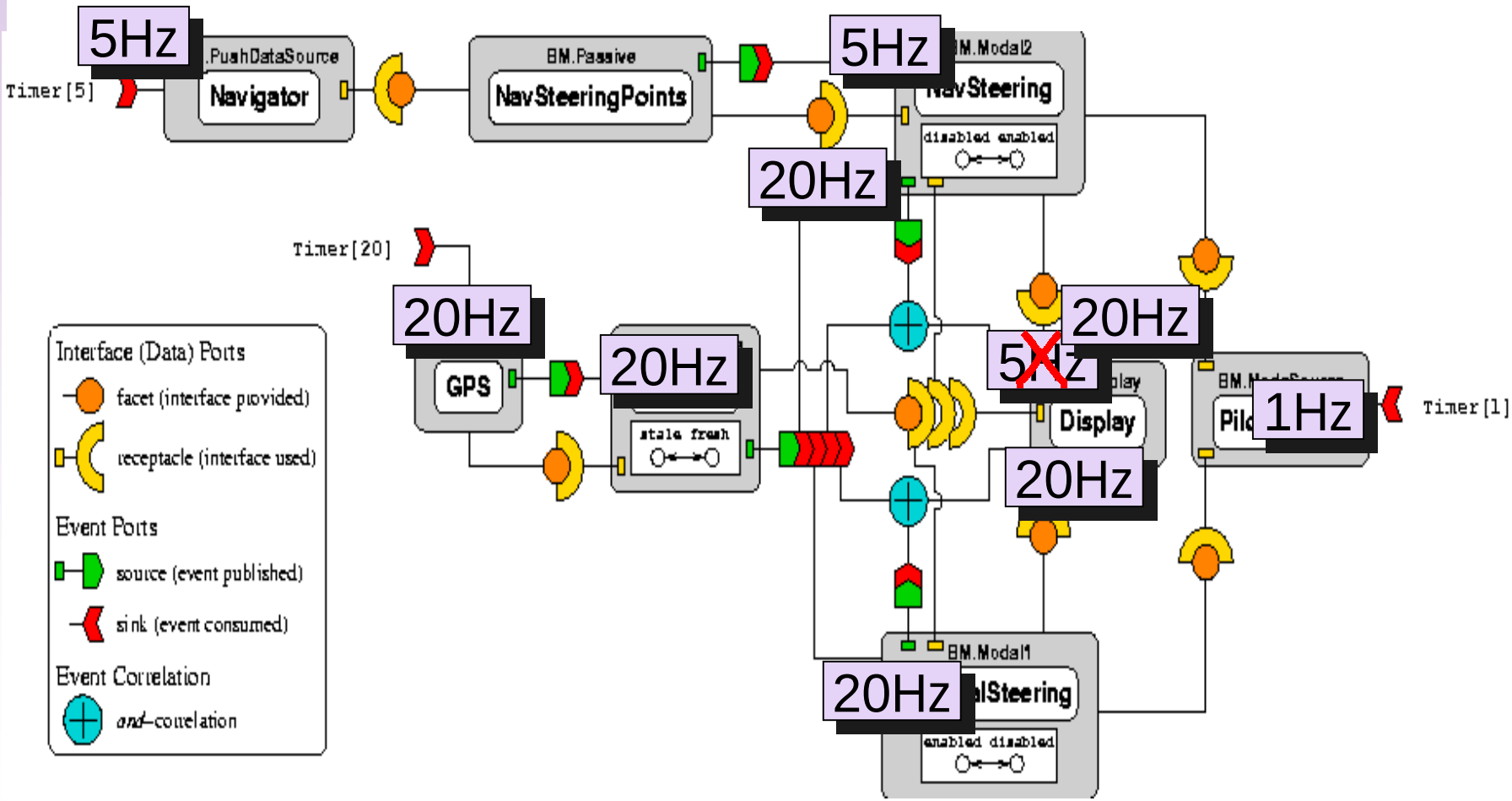
...port connections

Aspect Synthesis

- Analysis algorithms
- Synthesis of QoS and real-time aspects

Aspect Synthesis

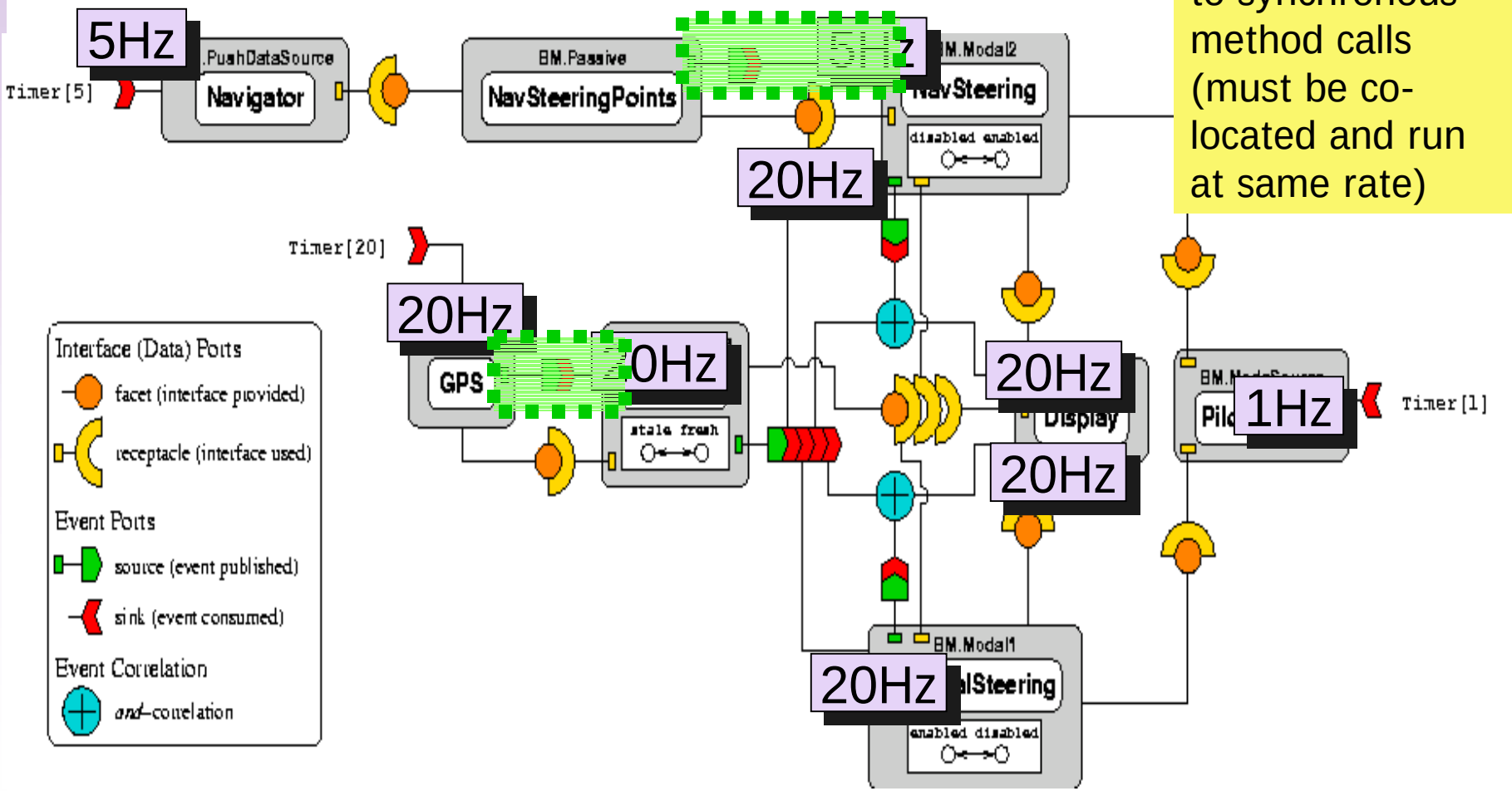
Dependency-driven rate assignment to event consumers



Aspect Synthesis

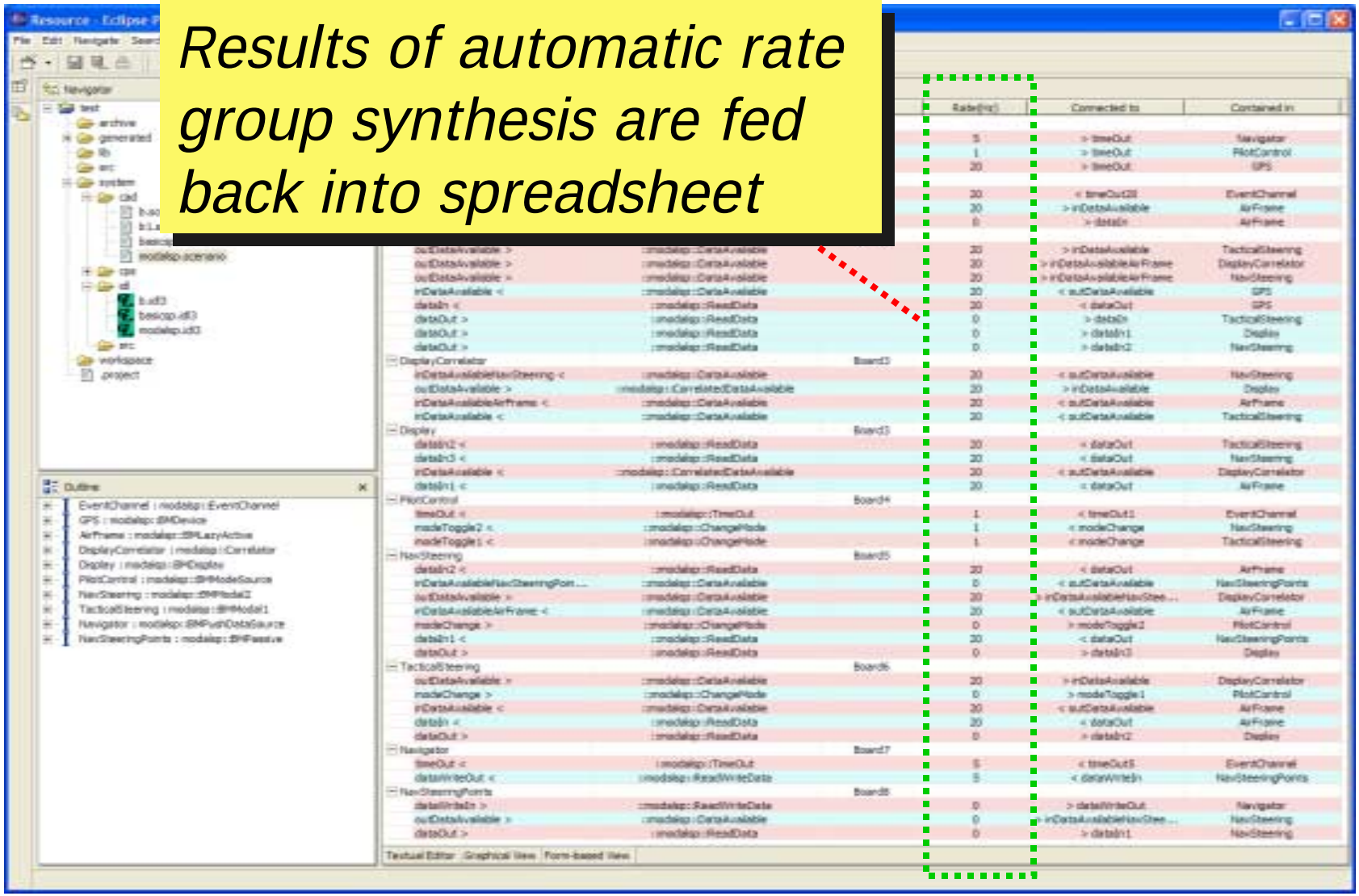
Automatic detection of optimization opportunities

Asynchronous message delivery to synchronous method calls (must be co-located and run at same rate)



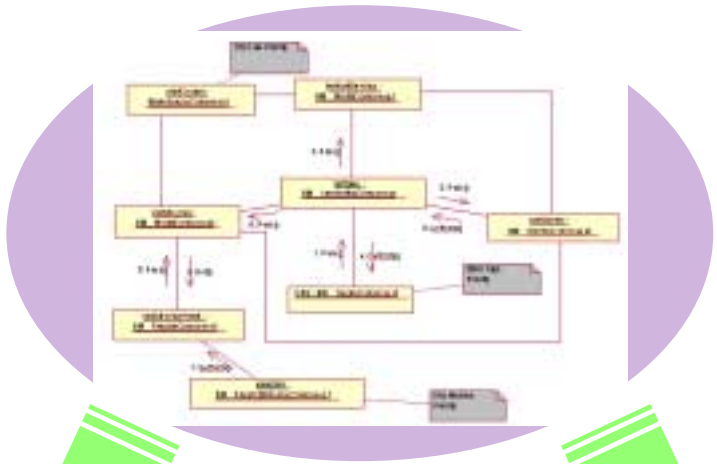
Spreadsheet View

Results of automatic rate group synthesis are fed back into spreadsheet

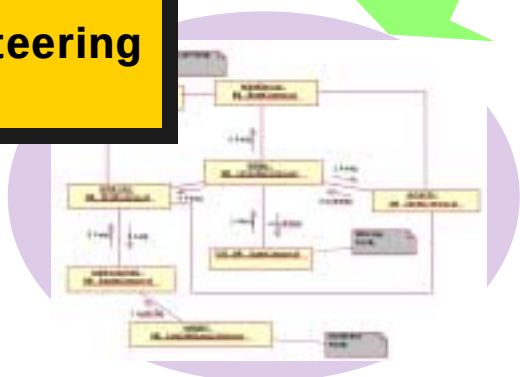


Mode-based Projections

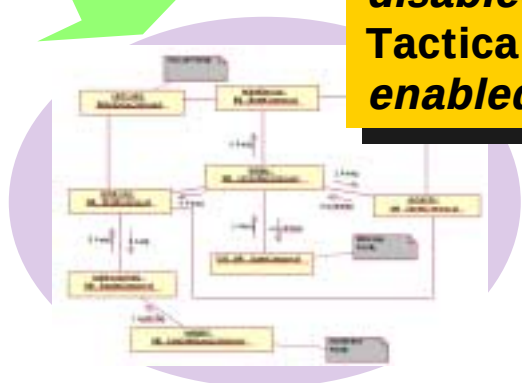
Scenario Diagram w/
Complete Connectivity



NavSteering
enabled
TacticalSteering
disabled



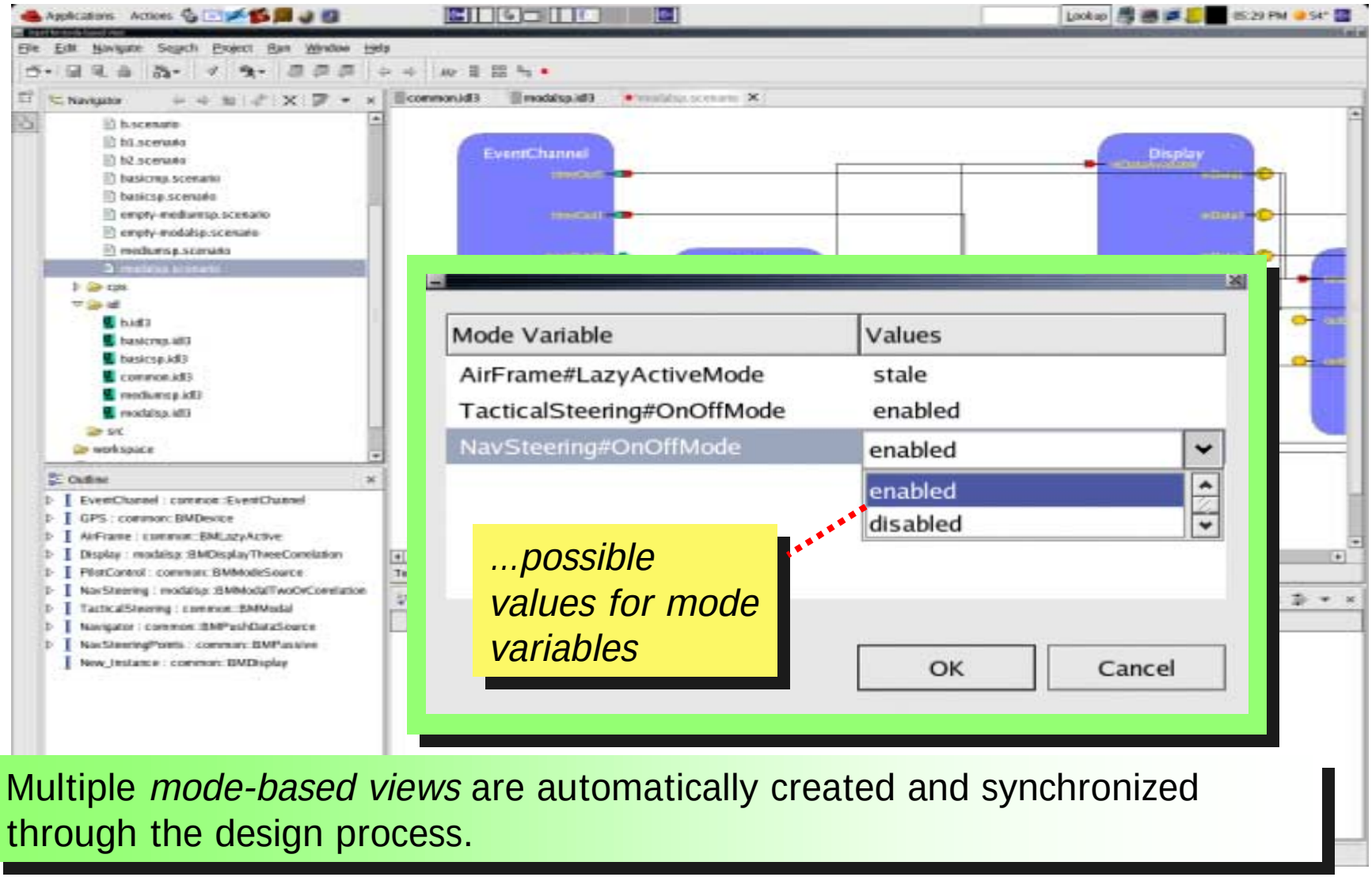
. . .



NavSteering
disabled
TacticalSteering
enabled

Enabled Connectivity for Different Modes

Mode-based Projections

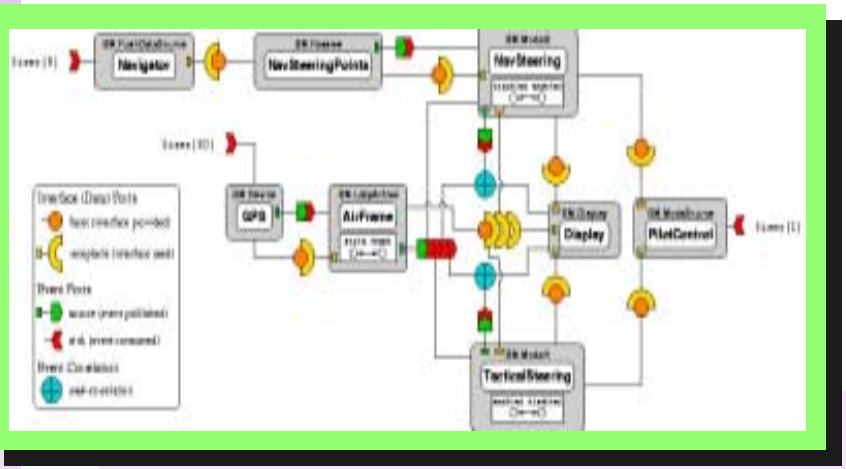


Multiple *mode-based views* are automatically created and synchronized through the design process.

Model-Checking Infrastructure

- Infrastructure dedicated to event-based middleware
- Complete model derived from CCM IDL, component-assembly description and annotations
- Model-checking for global system properties

Complete Model



Configuration Info

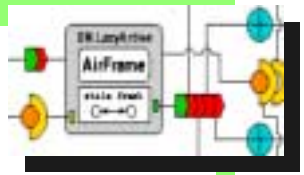
Transition System Specs

RT CORBA Event Channel

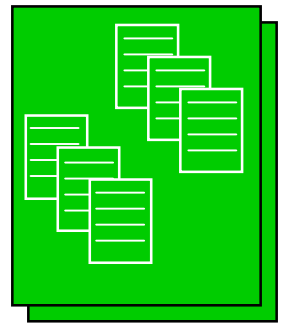
System Components

Leverage CORBA IDL

```
component BMLazyActive {
 provides  ReadData outData;
 uses      ReadData inData;
 publishes DataAvailable outDataAvailable;
 consumes  DataAvailable inDataAvailable;
 attribute LazyActiveMode dataStatus;
};
```



IDL Compiler



Component Implementation Stubs & Skeletons

+

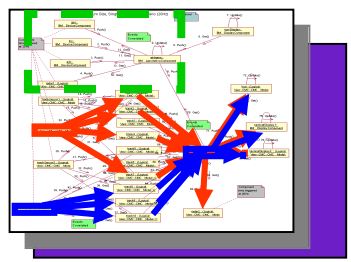
```
dependencydefault
   == none;

dependencies {
 inDataAvailable
   ->
  outDataAvailable;
}
```

Dependency Annotations

```
behavior {
 if (mode==enabled) {
  push outDataAvailable;
 else
  …
 }
```

Transition System Semantics

Model Builder



Dependency Analysis and Model-checking Engine

Incremental Specification

Increasing Effort & Strength of Verification

Specifications

port action dependencies

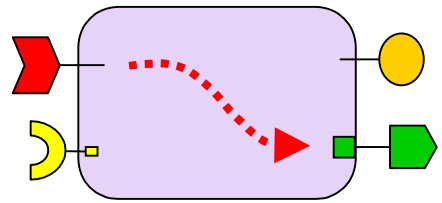
refinement

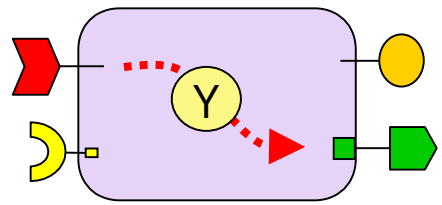
state-based dependencies

refinement

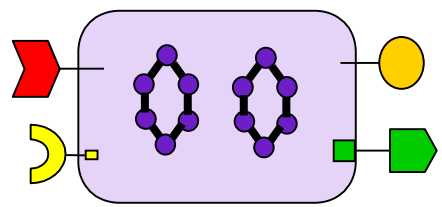
component transition semantics

Component Structure





...only in mode Y

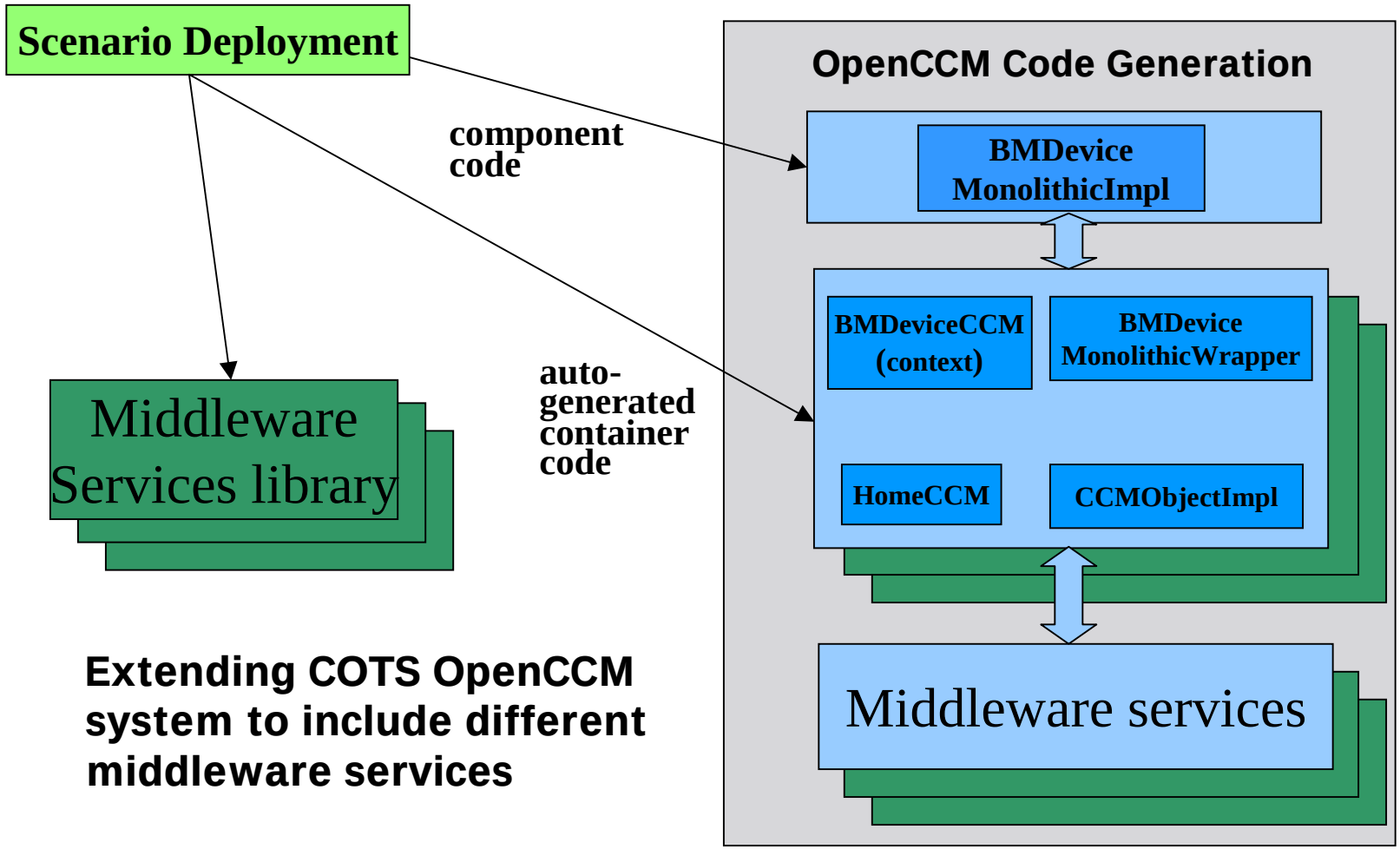


...state machines give abstract behavior

Code generation and Configuration

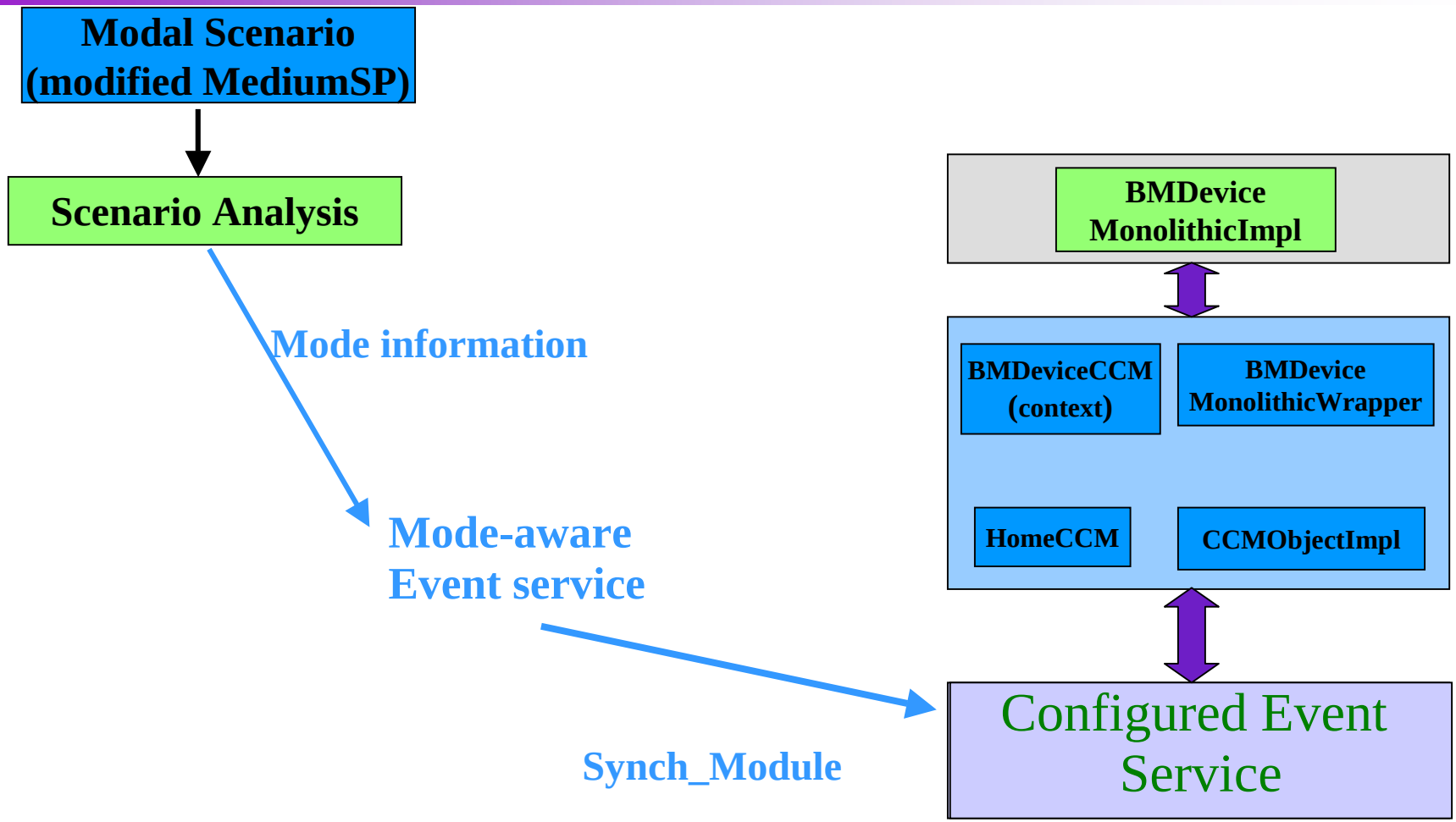
- OpenCCM CCM IDL to Java compiler to component stub and skeleton code generation
- XML files to container and middleware configuration

Container Adaptation



Extending COTS OpenCCM system to include different middleware services

Experiments: Mode-aware Event Service



Conclusions

- Cadena: model-driven design and implementation of DRE systems
- Light-weight specifications
- Component assembly description forms
- Dependency analysis tools
- Model-check infrastructure
- Code generation and configuration

Project Web Site

http://www.cis.ksu.edu/cadena