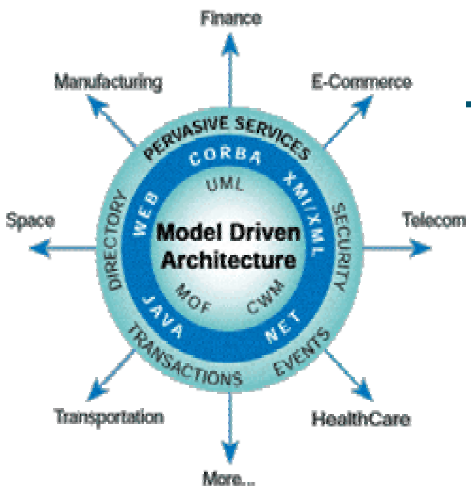
Enterprise-MDA

DataAccessTechnologies

Where Business Meets Technology

EDOC



Enterprise-MDA + SOA

UNIFIED MODELING LANGUAGE™

Applying Model Driven Architecture (MDA) to Services Oriented Architecture (SOA) to enable the Executable Enterprise

DataAccessTechnologies

Where Business Meets Technology

Introductions

Cory Casanave
cory-c@enterprisecomponent.com

Primary author of “CCA” in EDOC

DataAccessTechnologies

Where Business Meets Technology

GSA

Case Study

U.S. General Services Administration (GSA)

Project: Financial Management Line of Business

Customer: GSA OCIO & CFO

Provider: LMI & Data Access Technologies

Tooling: Component-X, Magicdraw UML & Eclipse

Sections reproduced with the permission of the GSA – George Thomas

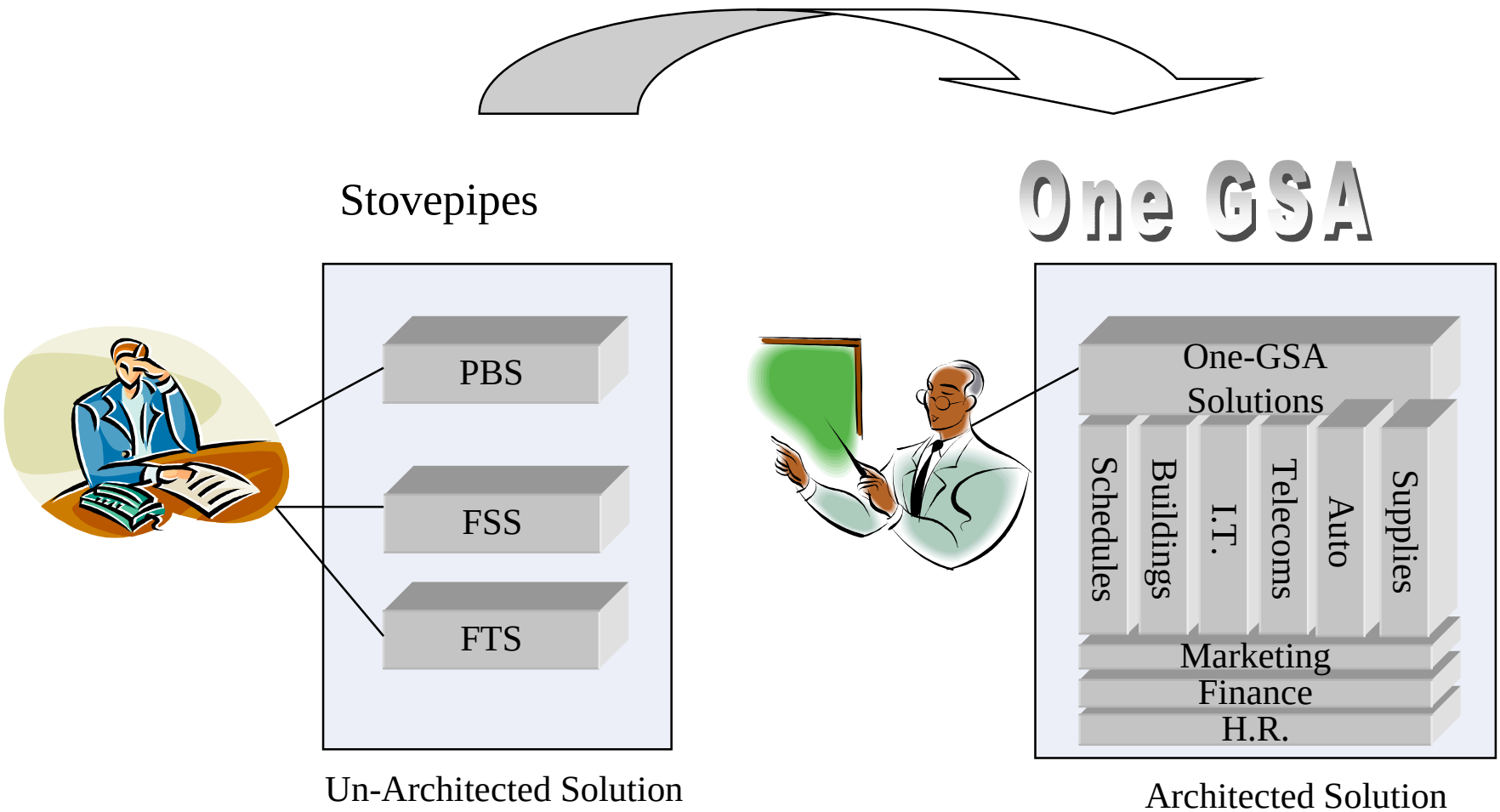
“Sea Change”

- Sea of change
  - Get-it-right (Initiative for better acquisition)
  - Merger of FTS/FSS (Major Internal Organizations)
  - Restructuring to provide a unified face to the customer
  - OMB and Congressional mandates and changes of mission
  - Integrating and modernizing financial management
  - Reduction of redundant processes and systems
- Implications
  - Massive organizational change
  - Massive system changes
  - Retraining staff
  - High cost of change
  - Risky and hard to achieve
  - Change combined with current costs and inefficiencies of redundant stovepipe systems is not practical

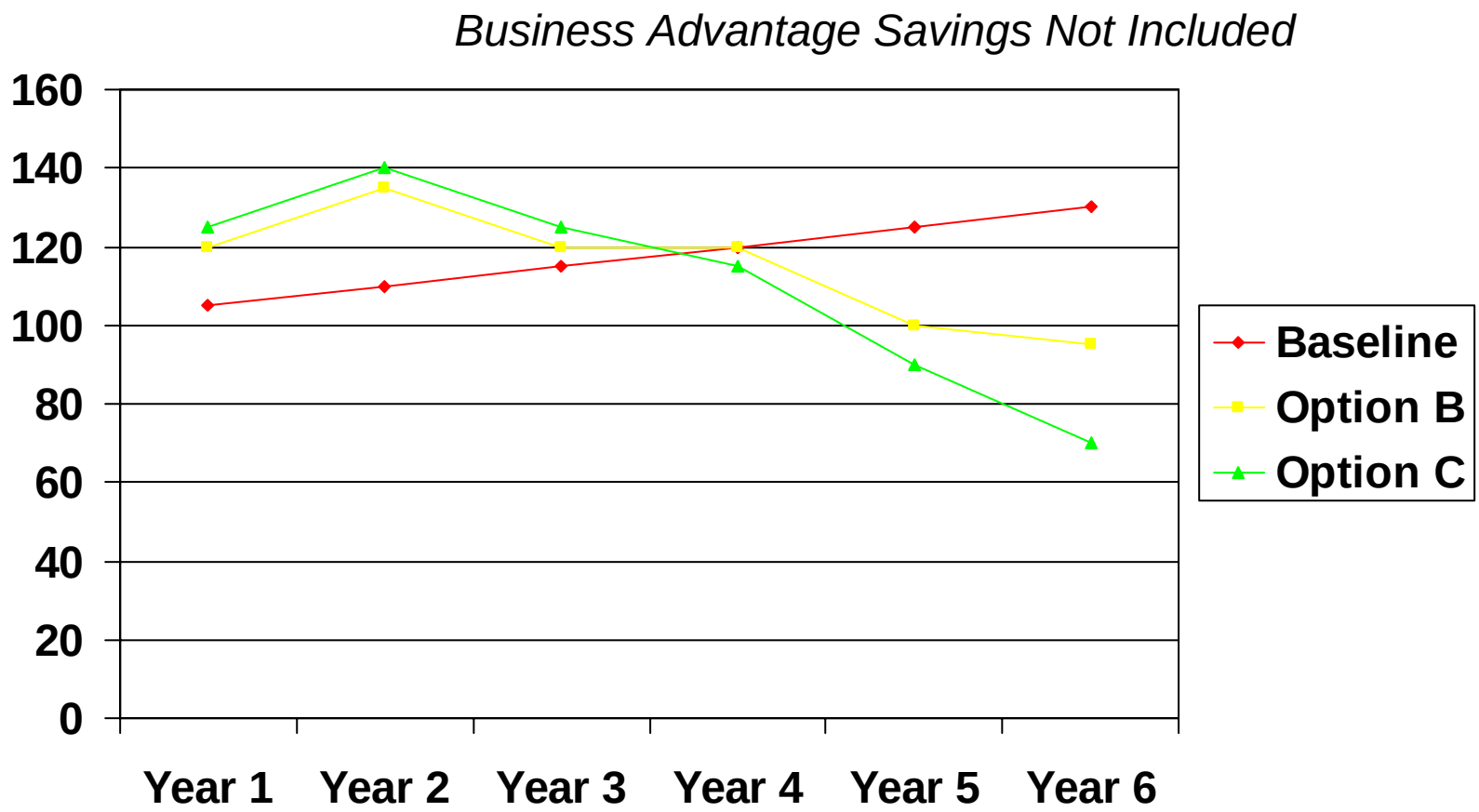
“Sea Change” Enablers & Cost Reduction

- Enterprise-MDA
  - Executable enterprise architecture based on business goals and processes with automated connection to systems layer
- Service Oriented Architecture (SOA)
  - Architecting the business and technology as services
- Value Chain Analysis (VCA)
  - Analyzing and restructuring business processes based on realized customer value
- Enterprise Service Bus
  - Common infrastructure for SOA
- Combined effect of more automated processes

One-GSA Initiative



System + Investment cost over 6 years



Note: Representative Numbers Est. NPV Break Even – About 6 Years

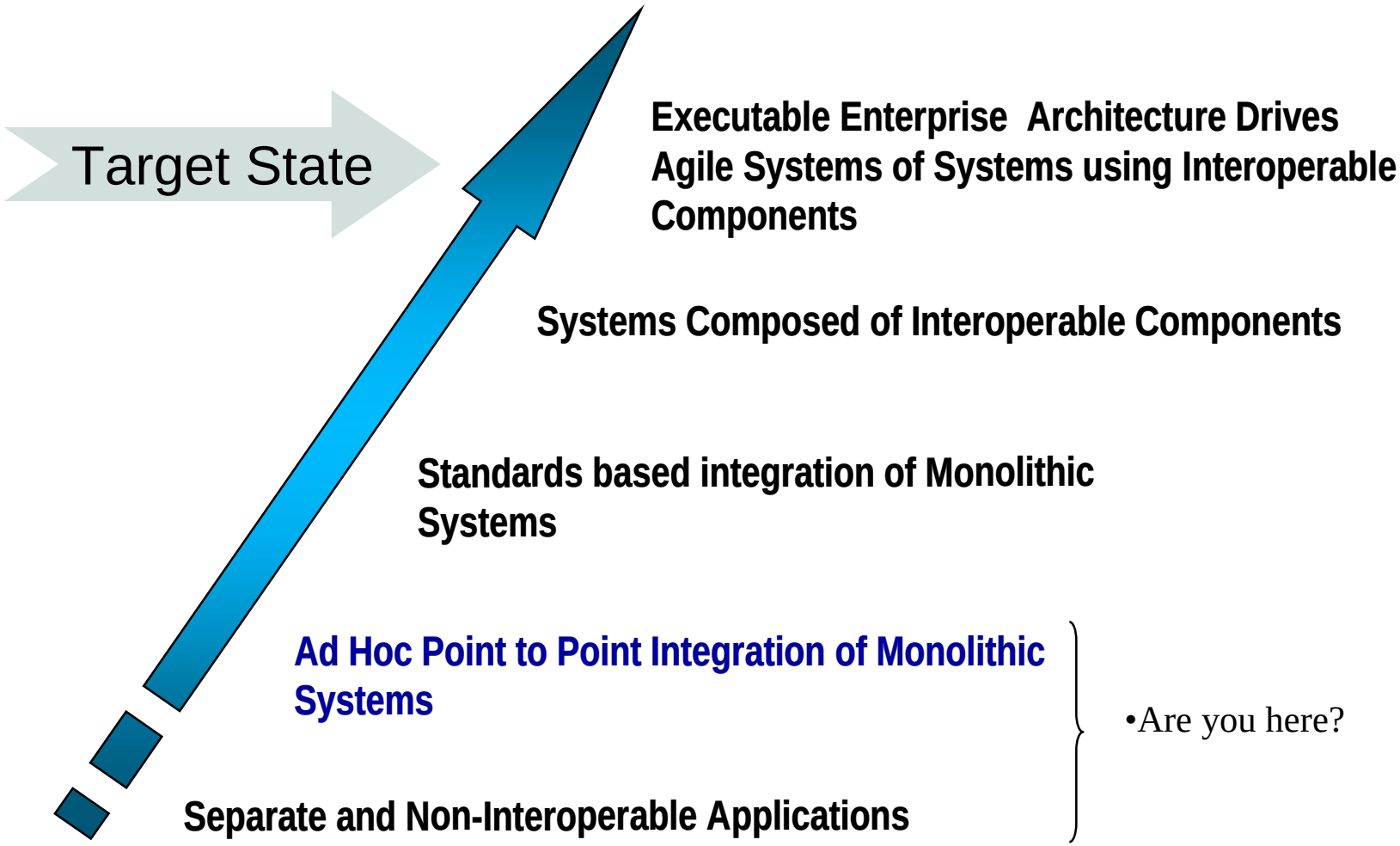
Enterprise Modernization Strategy

- Identify components that will offer greatest ROI
  - Create target executable model
    - We have a baseline with the One-GSA model!
  - Identify system of systems to consider for target
  - Pick an alternative for each;
    - Evolve one or more current systems to support target processes, take on new capabilities and support One-GSA interfaces and/or
    - Harvest one or more systems to build a replacement and/or
    - Integrate functionality into shared services as common components and/or
    - Replace systems or parts of systems that are no longer suitable.
  - Model driven SOA provides the flexibility to mix and match approaches as required. Commonality where possible – diversity where necessary. Evolving over time from integration to common components.
  - End result – architected system of systems

Consolidation into Service Components

- The Good
  - Strategic reduction in operating cost – up to 50%
  - Agile business processes
  - Unification of the enterprise
  - Only way to achieve enterprise transition?
- The Bad
  - Investment in change – As high as 25%
  - Legacy and packaged systems are not componentized
- The Ugly
  - Change is expensive and can be disruptive
  - Current boundaries and ownership change – may require centralized authority and budgeting
  - Requires more “enterprise” agreement – very difficult to get consensus

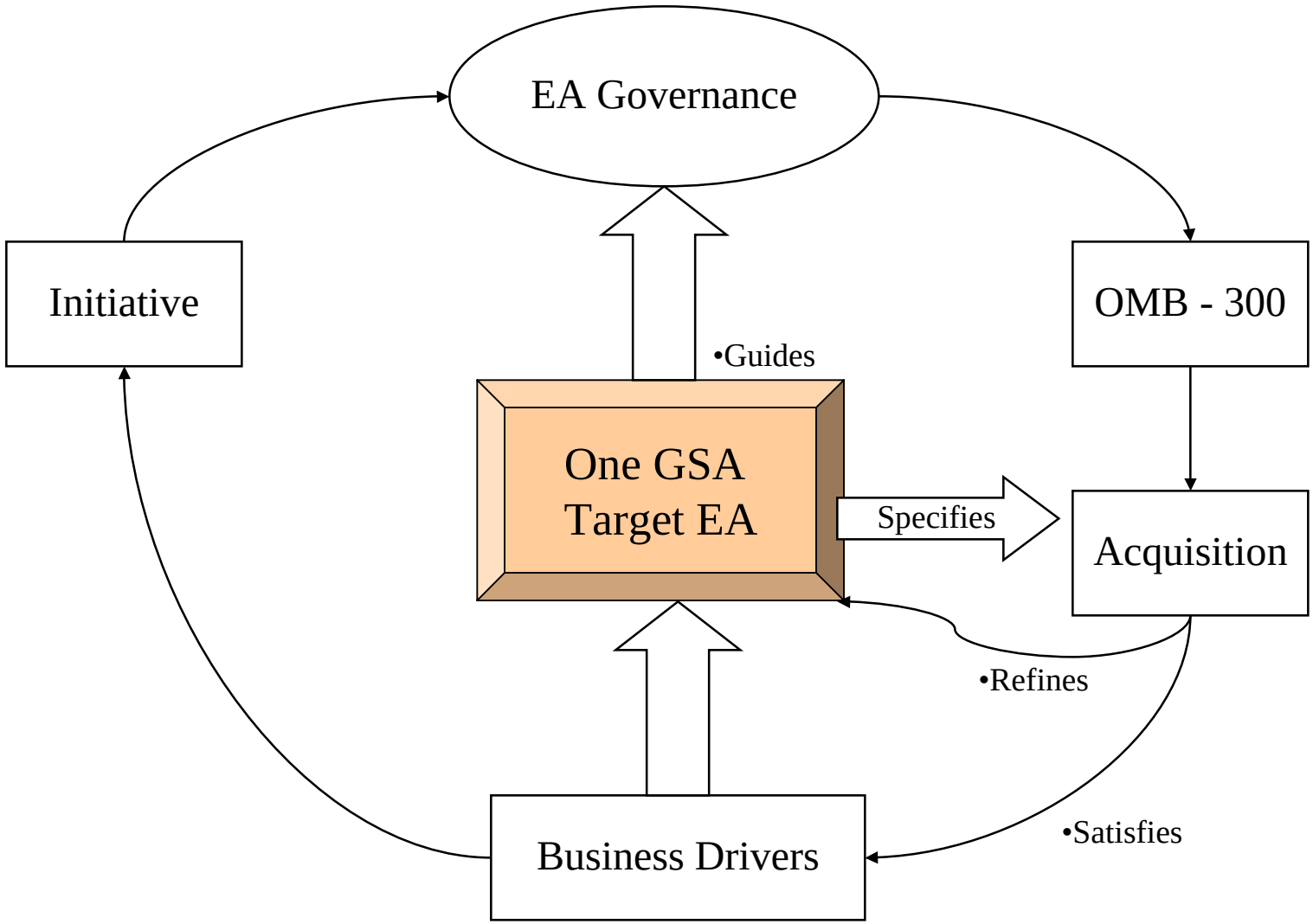
Strategic Migration



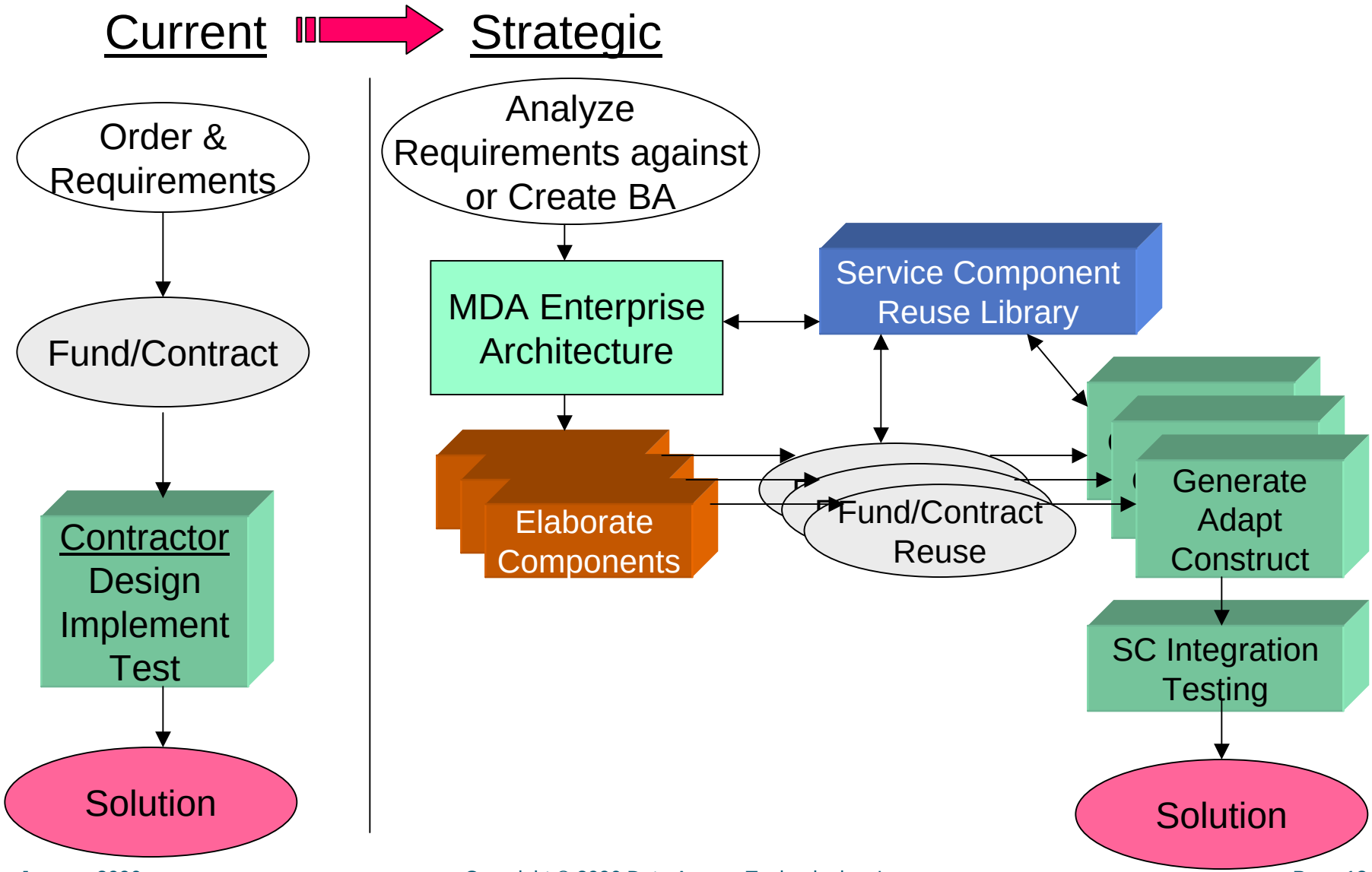
Consolidation into Service Components

- The Good
  - Strategic reduction in operating cost – up to 50%
  - Agile business processes
  - Unification of the enterprise
  - Only way to achieve enterprise transition?
- The Bad
  - Investment in change – As high as 25%
  - Legacy and packaged systems are not componentized
- The Ugly
  - Change is expensive and can be disruptive
  - Current boundaries and ownership change – may require centralized authority and budgeting
  - Requires more “enterprise” agreement – very difficult to get consensus

EA Governance Structure



MDA Enhanced Procurement



DataAccessTechnologies

Where Business Meets Technology

Enterprise MDA

An approach to realizing executable enterprise architecture with MDA and SOA

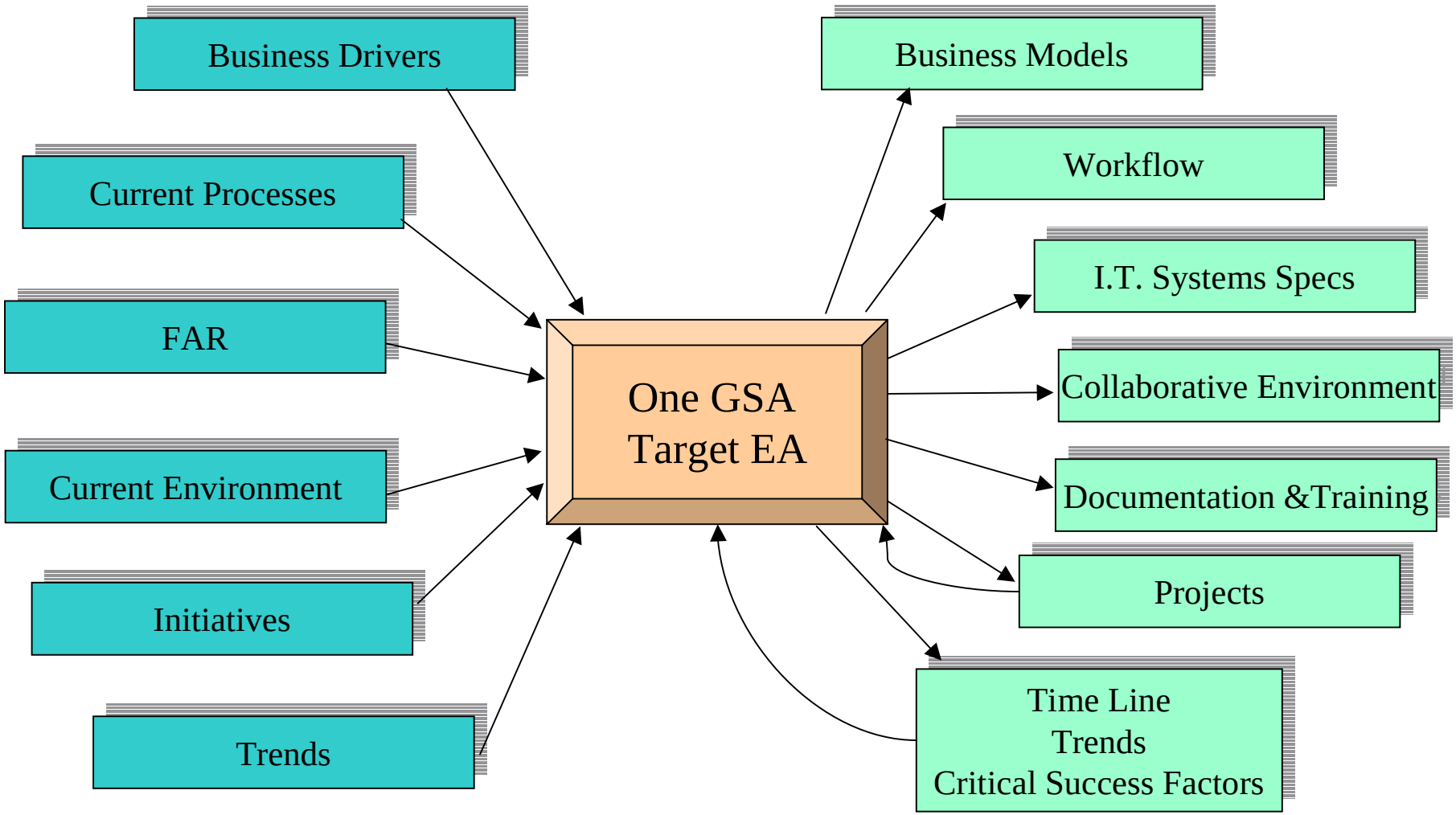
Enterprise MDA

- Architecture at the Enterprise Level
  - Systems of systems
  - Collaboration of organizations, systems & people
  - Wide-scale collaborative processes
    - roles and responsibilities
  - Business Service Oriented Architecture
  - Enterprise Components
  - Componentizing functionality – not creating it
  - Executable processes – smooth transition from model to simulation to solution
- Executable Enterprise Architecture

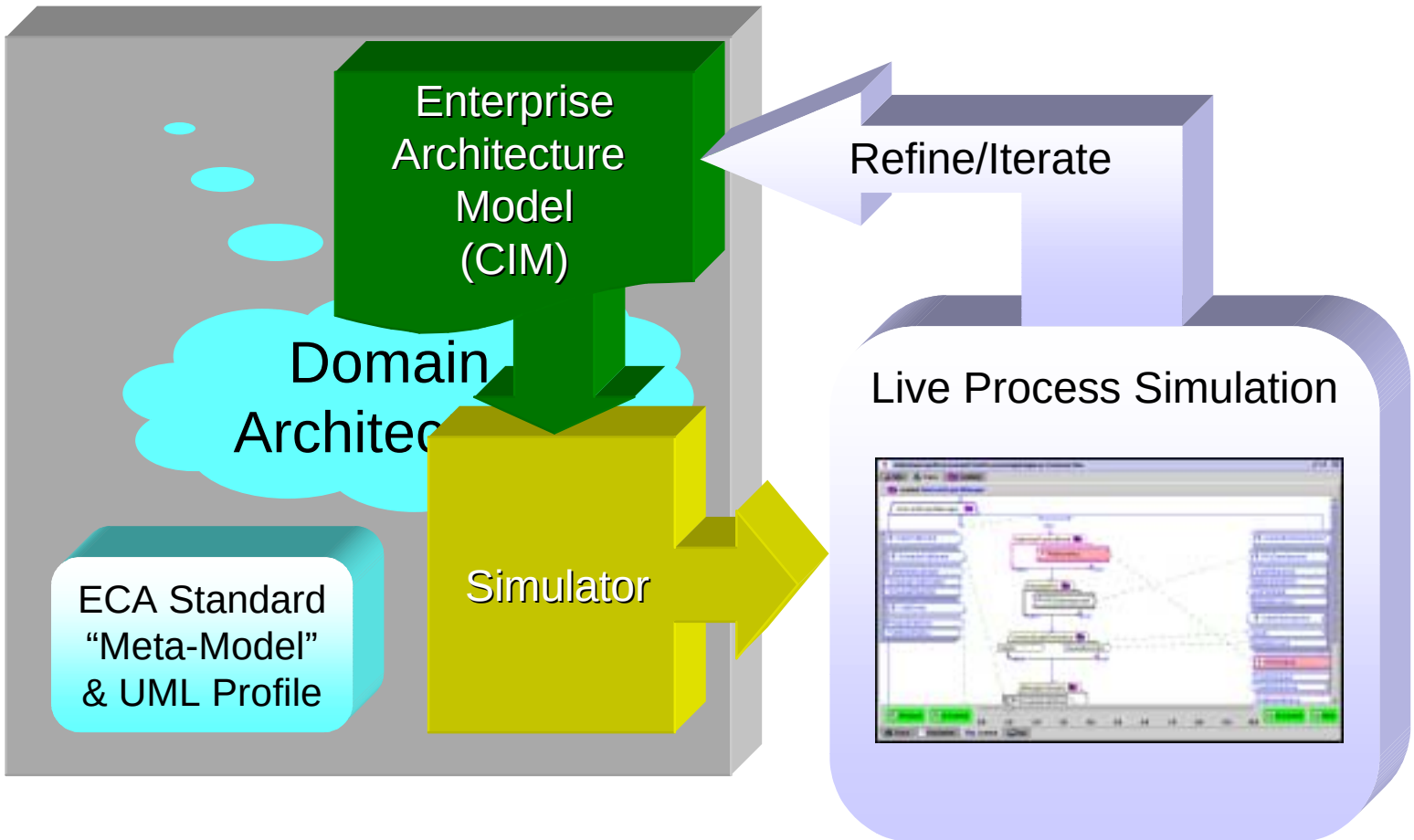
The OMG-Enterprise Collaboration Architect

- ECA is a “profile of UML”, a way to use UML for a specific purpose - it is an OMG standard
  - That purpose is *modeling enterprise systems*.
- You can also think of this as a “modeling framework” for enterprise computing
- ECA is part of the “Model Driven Architecture” (MDA) initiative of the OMG
  - Using precise modeling techniques as part of the development lifecycle to speed development and provide technology independence
- ECA has been adopted by the OMG as part of the EDOC Profile for UML specification.

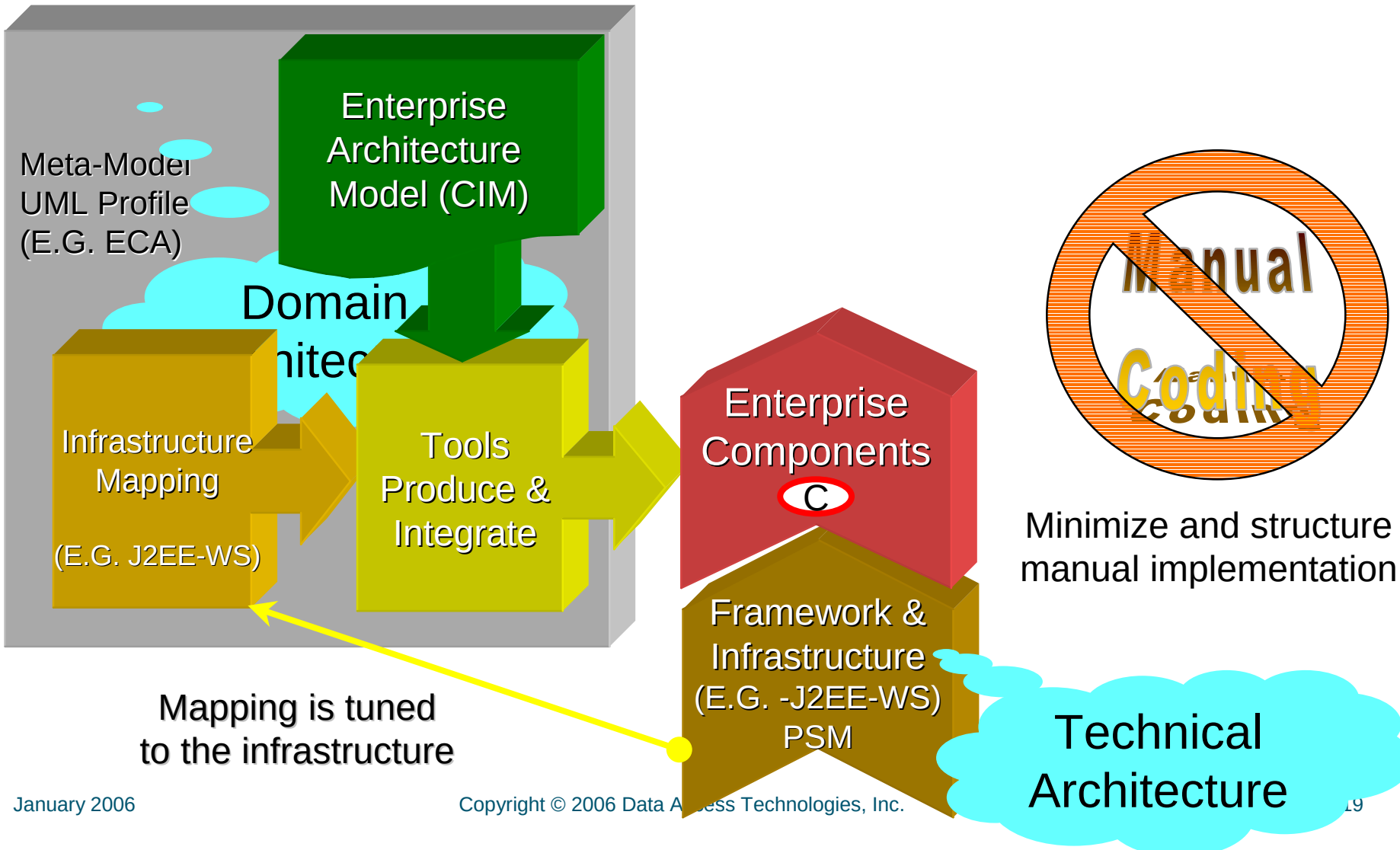
Value Focused Target Architecture



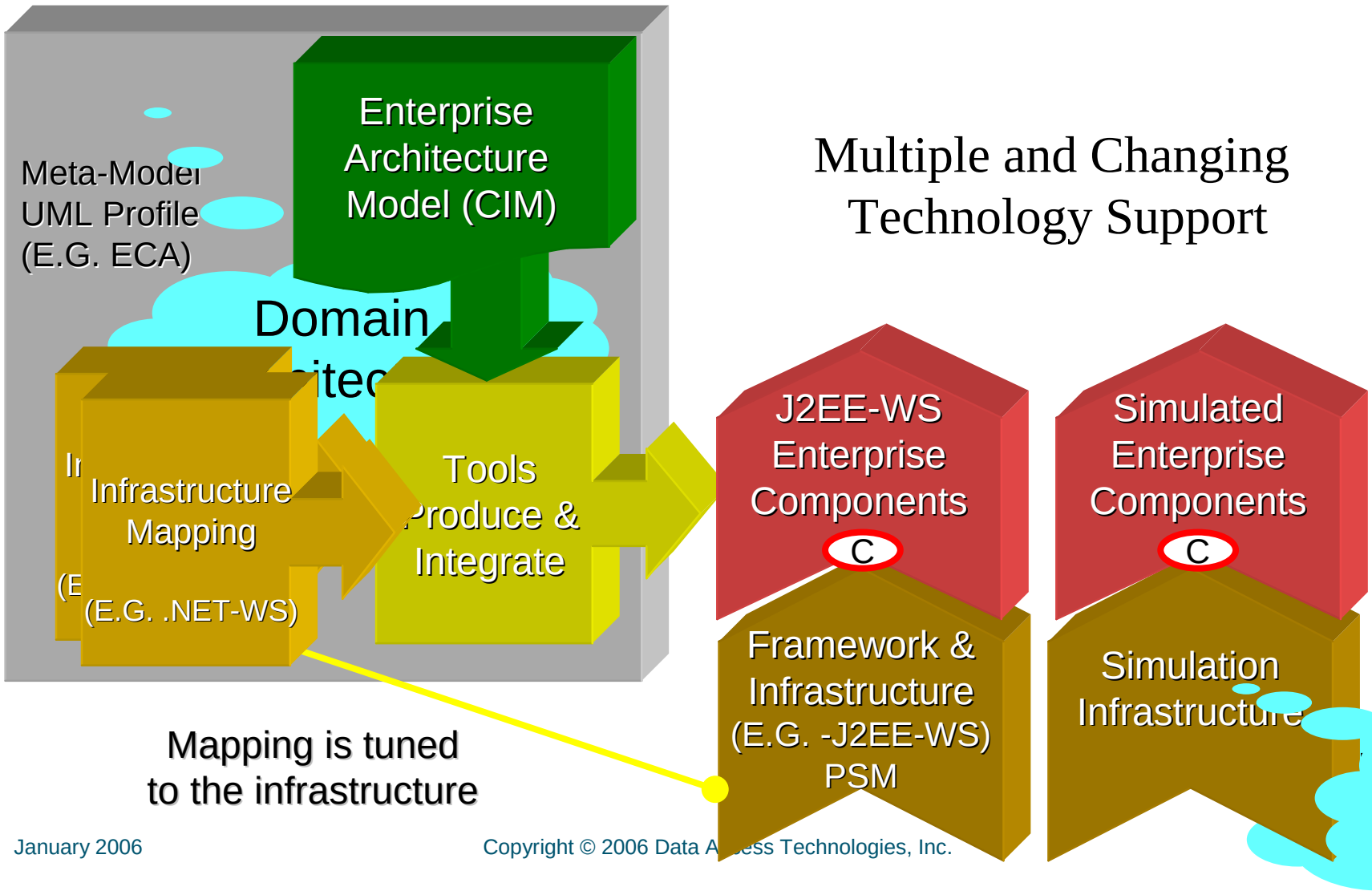
Simulated
Model Driven Architecture



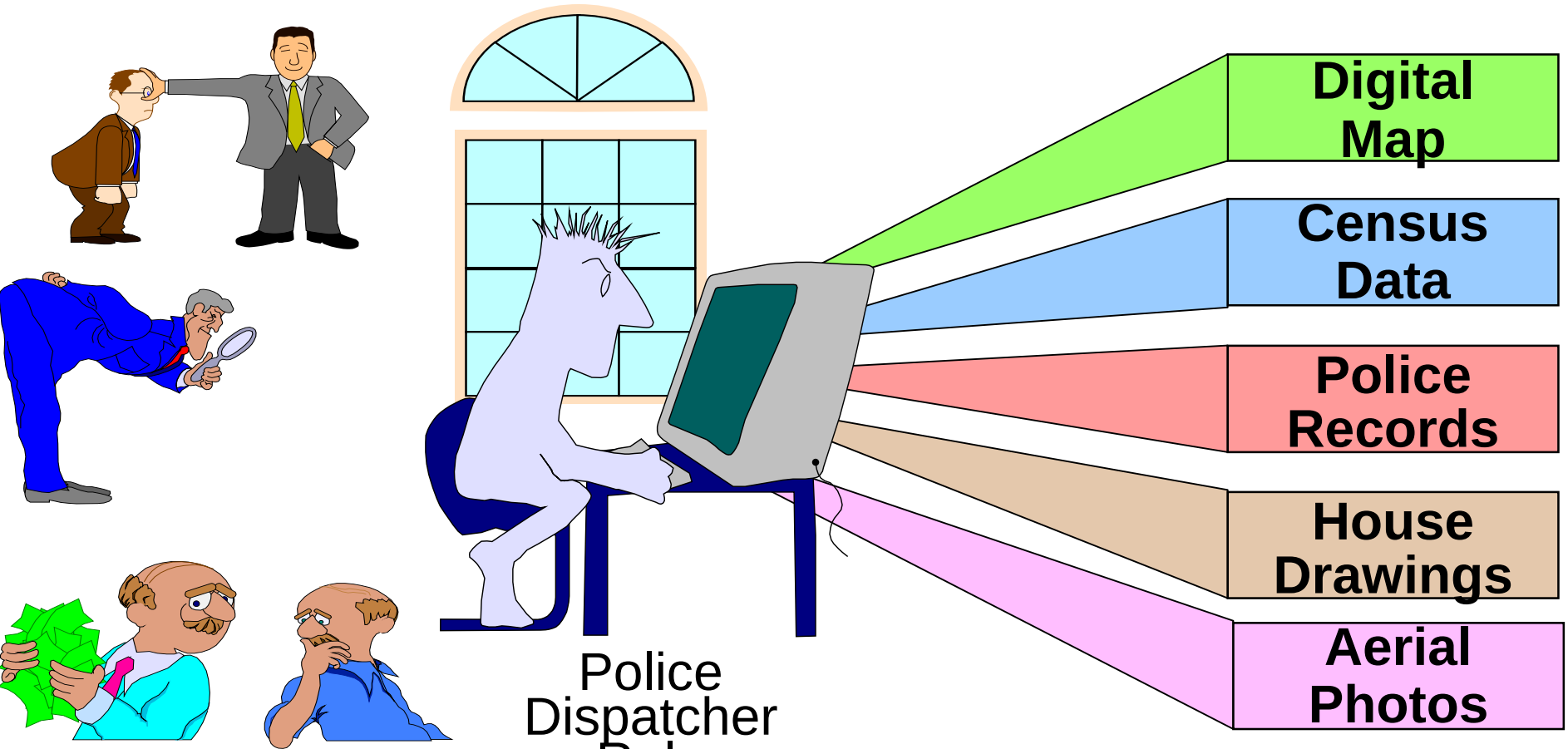
Automated
Model Driven Architecture



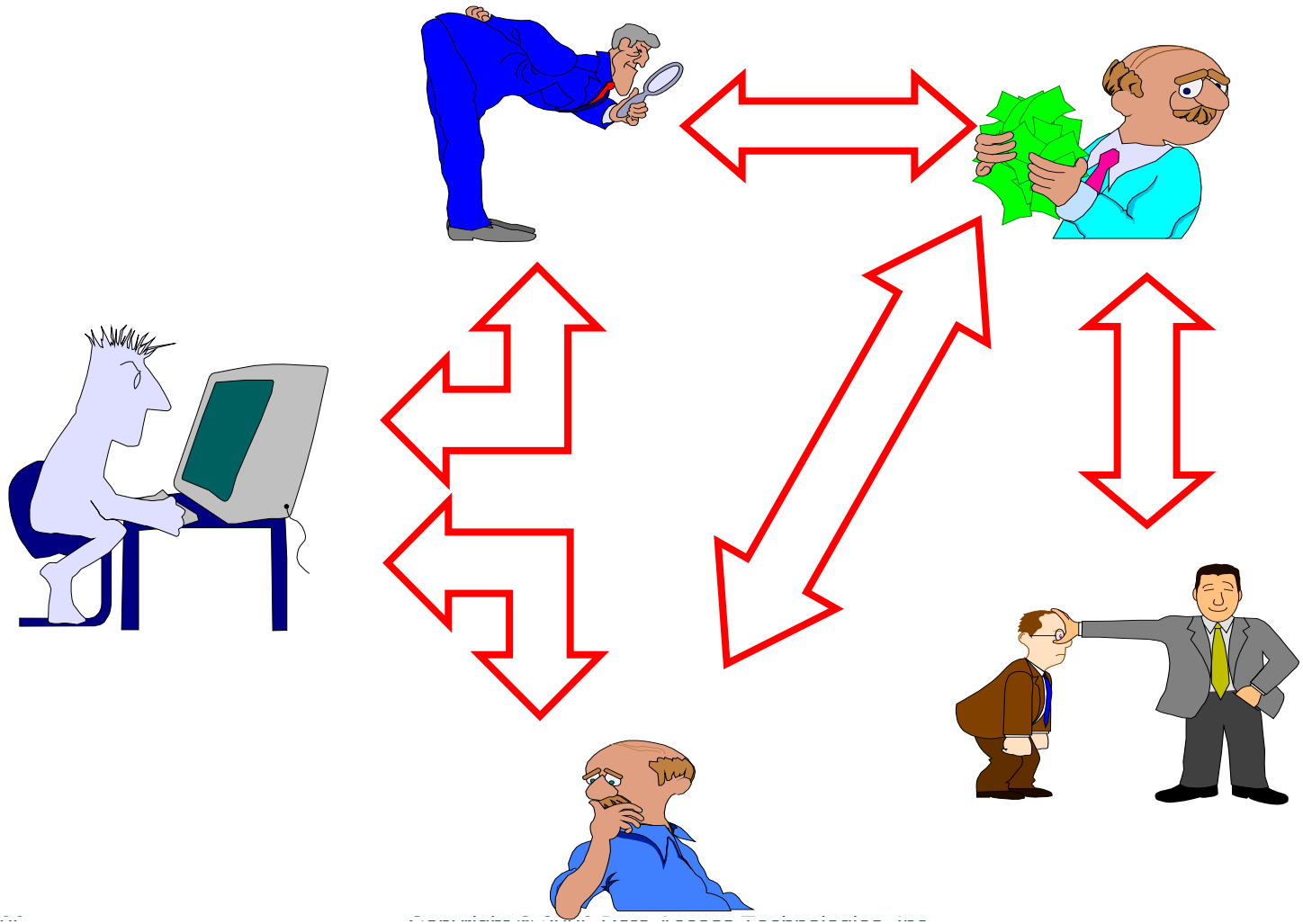
Automated
Model Driven Architecture



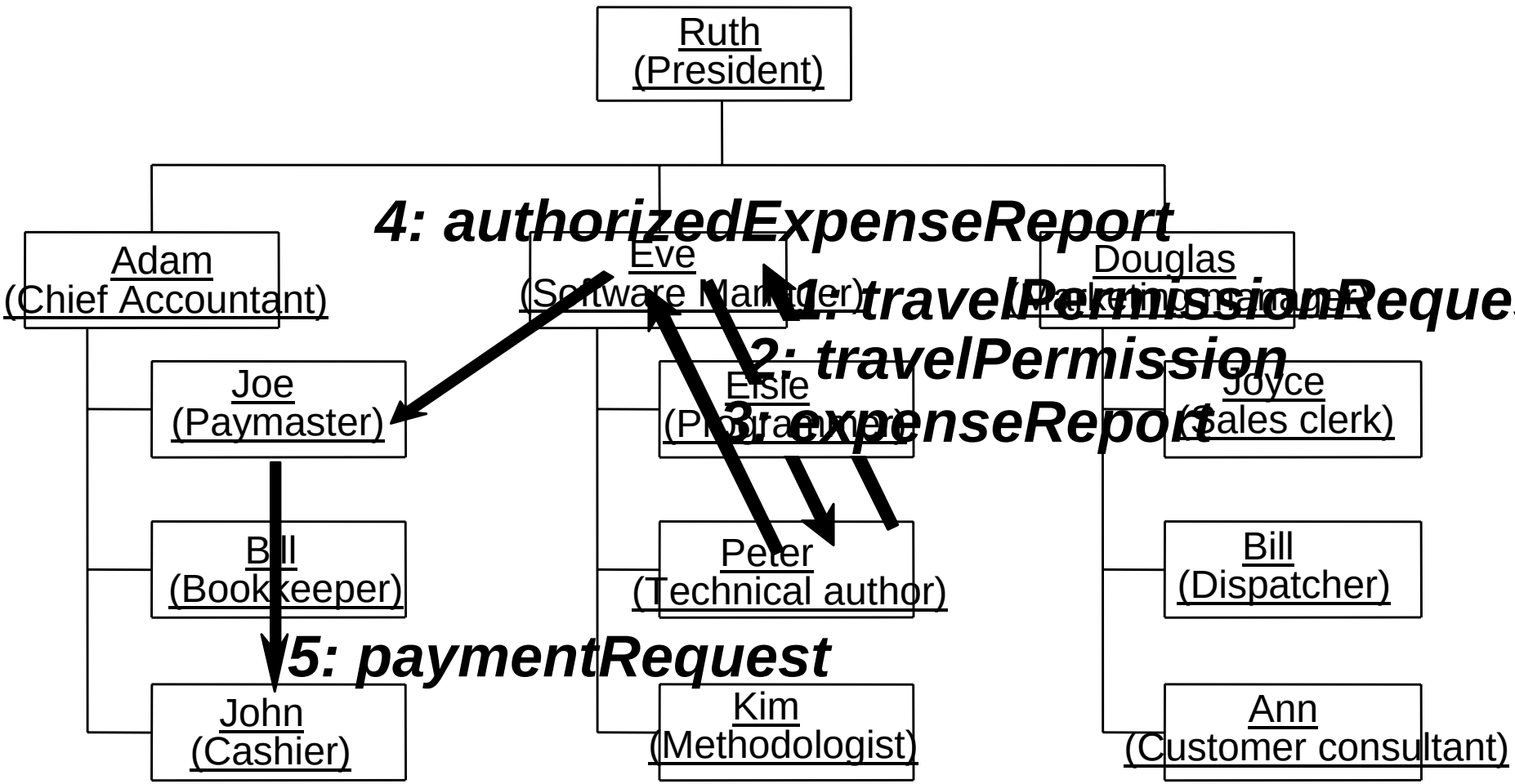
The Connected Enterprise
Content and Communication



Multiple roles in a collaboration

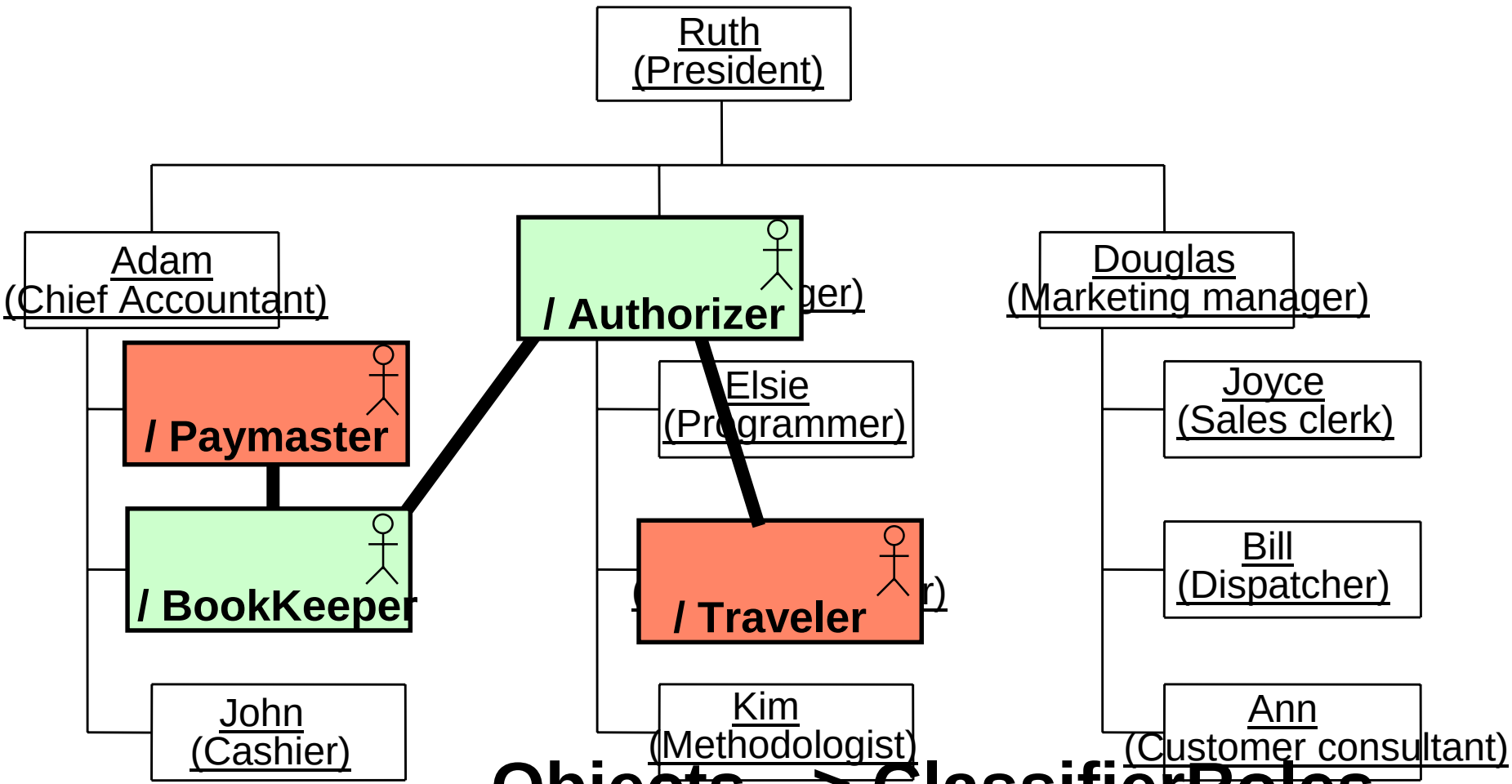


Travel Expense Example



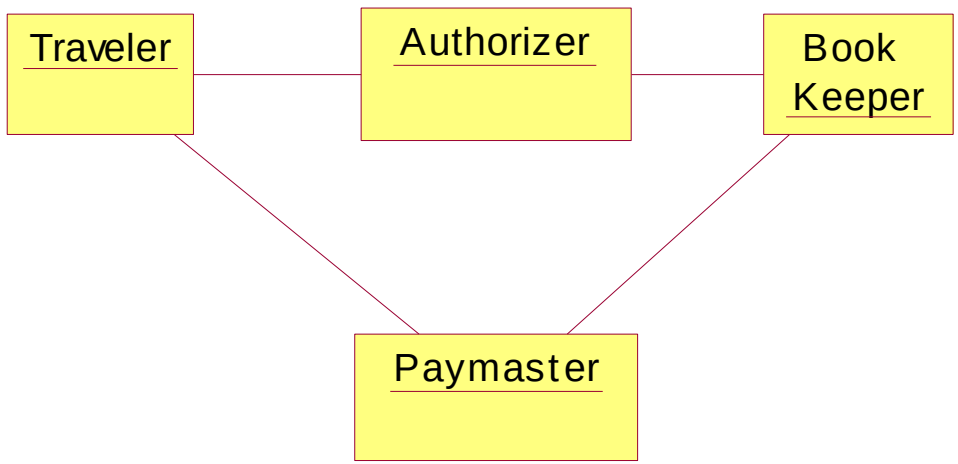
UML Collaboration Diagram

Travel Expense Model



Objects --> ClassifierRoles

Collaboration Diagram



The Marketplace Example

Mechanics Are Us
Buyer

Order

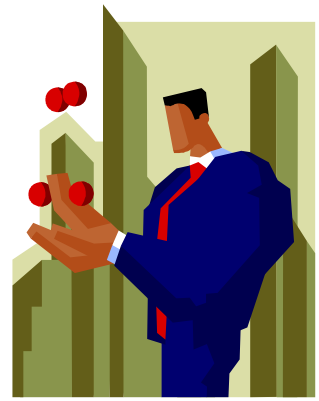
Conformation

Shipped

Process
Complete

Status

Physical
Delivery



Acme Industries
Seller

Ship Req

Shipped

Delivered

GetItThere Freight
Shipper

Where are the services?

Web Service

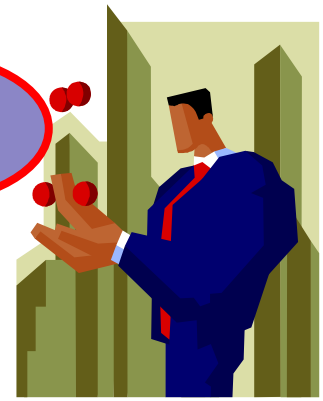
Mechanics Are Us
Buyer

Order

Conformation

Shipped

Web Service



Acme Industries
Seller

Web Service

Status

Physical
Delivery

Web Service

GetItThere Freight
Shipper

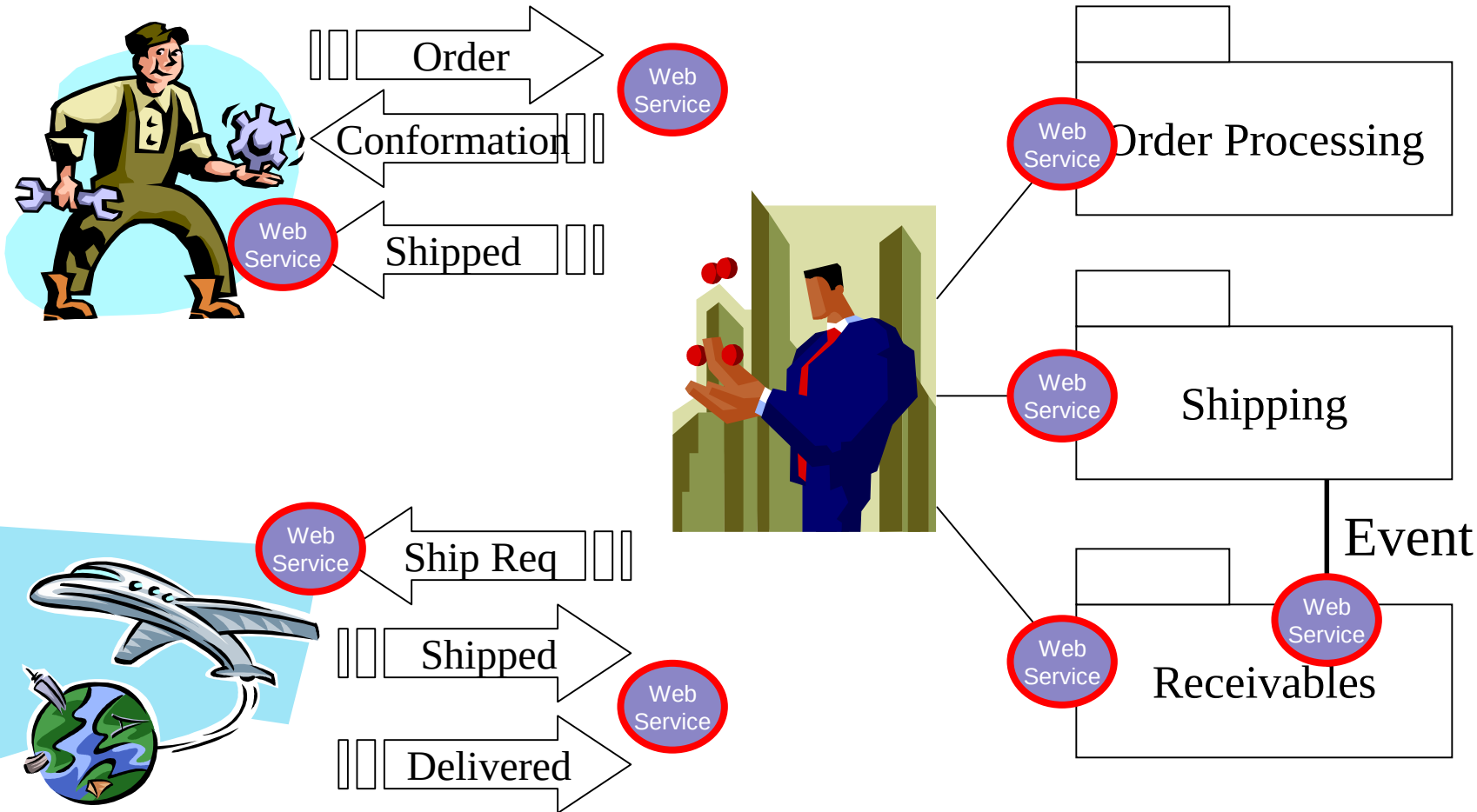
Ship Req

Shipped

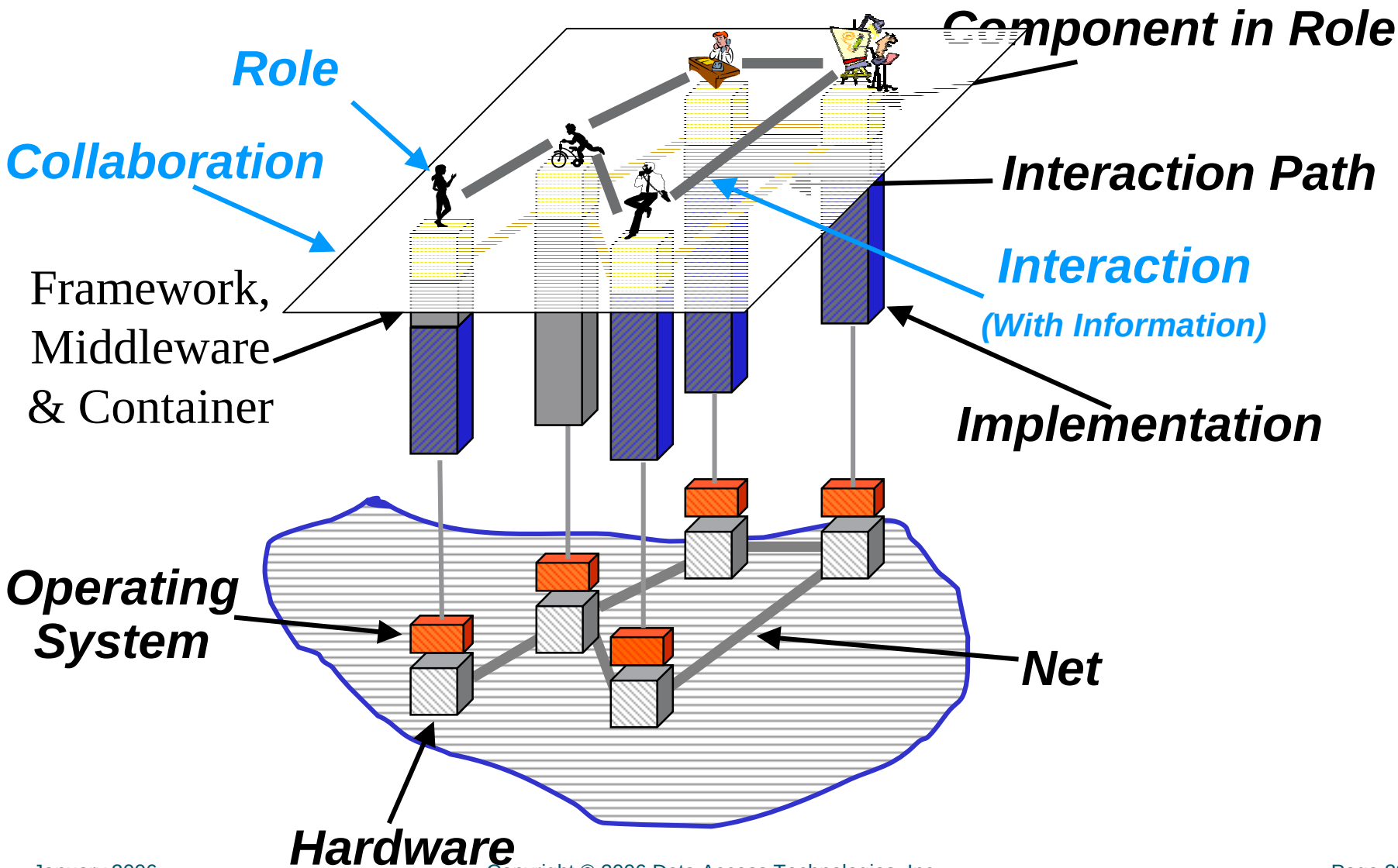
Delivered

Web Service

Inside the Seller

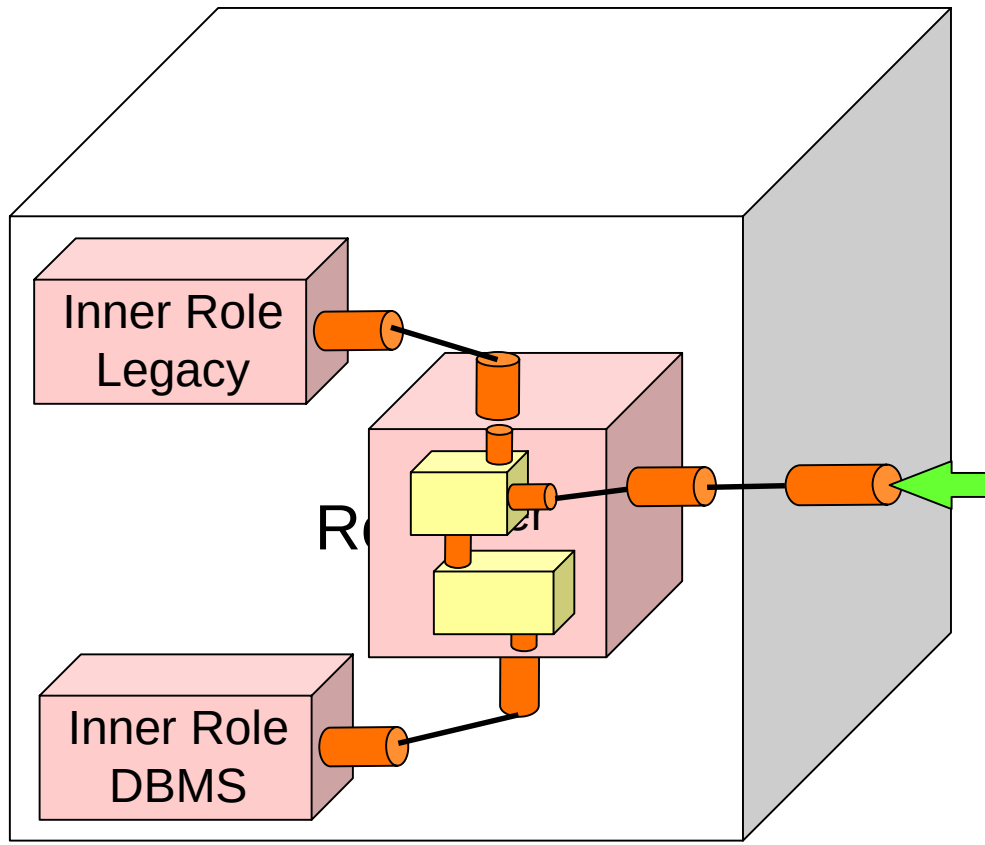


Roles to Systems

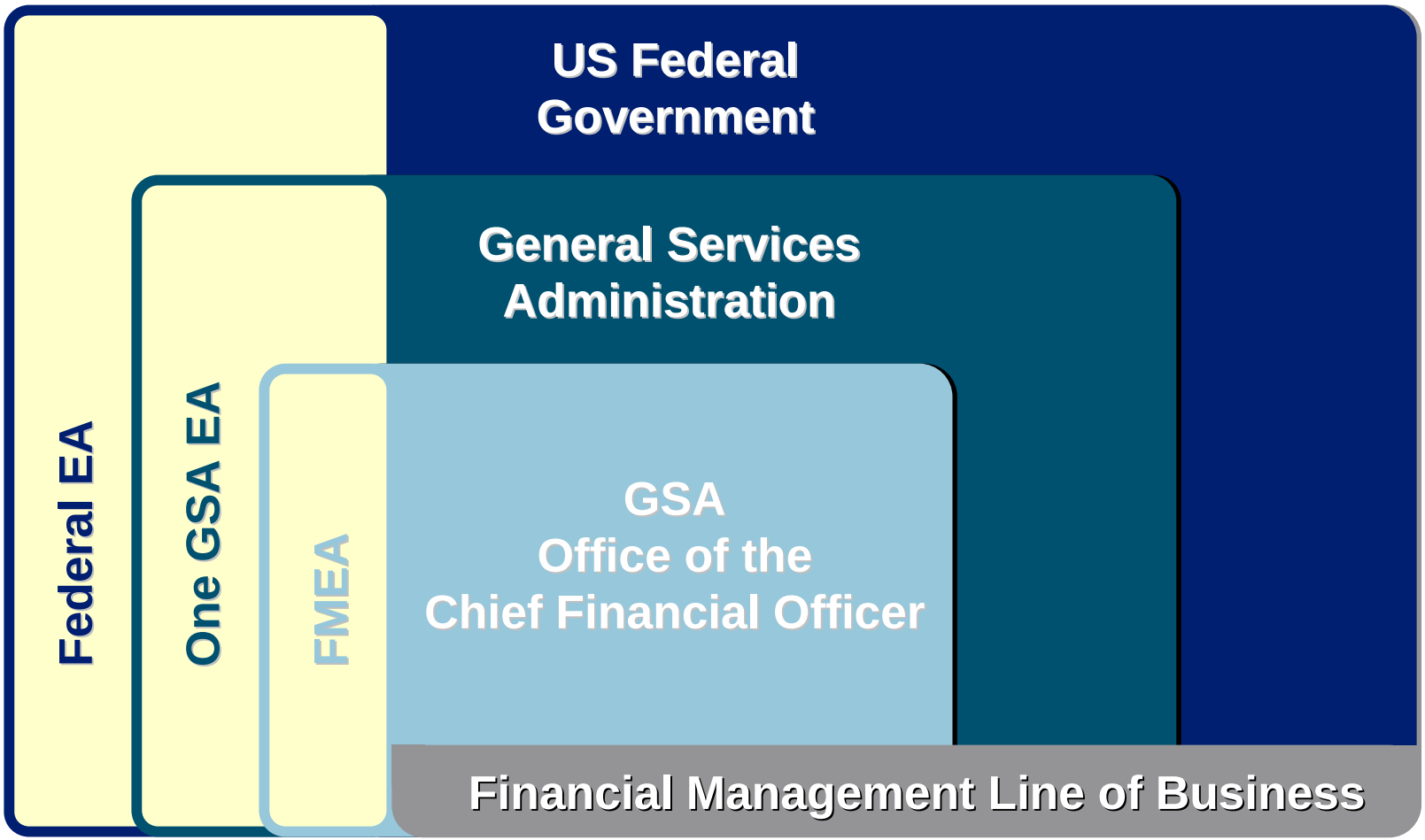


Drilling down – inside a role

- The open domain should make no assumptions about the “inside” of a role.
- Inside one role you *frequently* find more collaborating “parts” of the enterprise - the same model *may* be used
- Until you get to system inside a managed domain
  - Shared resources (DBMS)
  - Common Management
  - Frequently a legacy system
  - Code



FMEA Example in Context



FMEA

- The Financial Management Enterprise Architecture (FMEA) project was sponsored by the US General Services Administration (GSA) Offices of the Chief Information Officer and Chief Financial Officer.
- The project was carried out under the umbrella of the “One GSA” Enterprise Architecture program.
- Project deliverables included:
  - A target business architecture for consistent and comprehensive financial management supporting all GSA services and staff offices.
  - A logical system architecture for a cohesive financial management suite supporting the business architecture, particularly in areas in which a transition needed to be made off legacy systems.
  - A set of interface definitions to act as the basis for a standard GSA financial management service-oriented architecture.

Model Driven Architecture

MDA as defined for GSA Enterprise Architecture work

- Computation Independent Model (CIM)
  - The business model
- Platform Independent Model (PIM)
  - Technology independent system specification
  - Conforms to the business model (CIM)
- Platform Specific Model (PSM)
  - Technology specific (e.g., middleware, application platform, etc.) system design
  - Conforms to the system specification (PIM)

Computation Independent Model

- Process Model (EDOC)
  - Business Environment
  - Discipline Roles
  - Enterprise Roles
  - Work Roles
  - Activities and Subactivities
- Information Model (UML)
  - Business Transactions
  - Business Entities

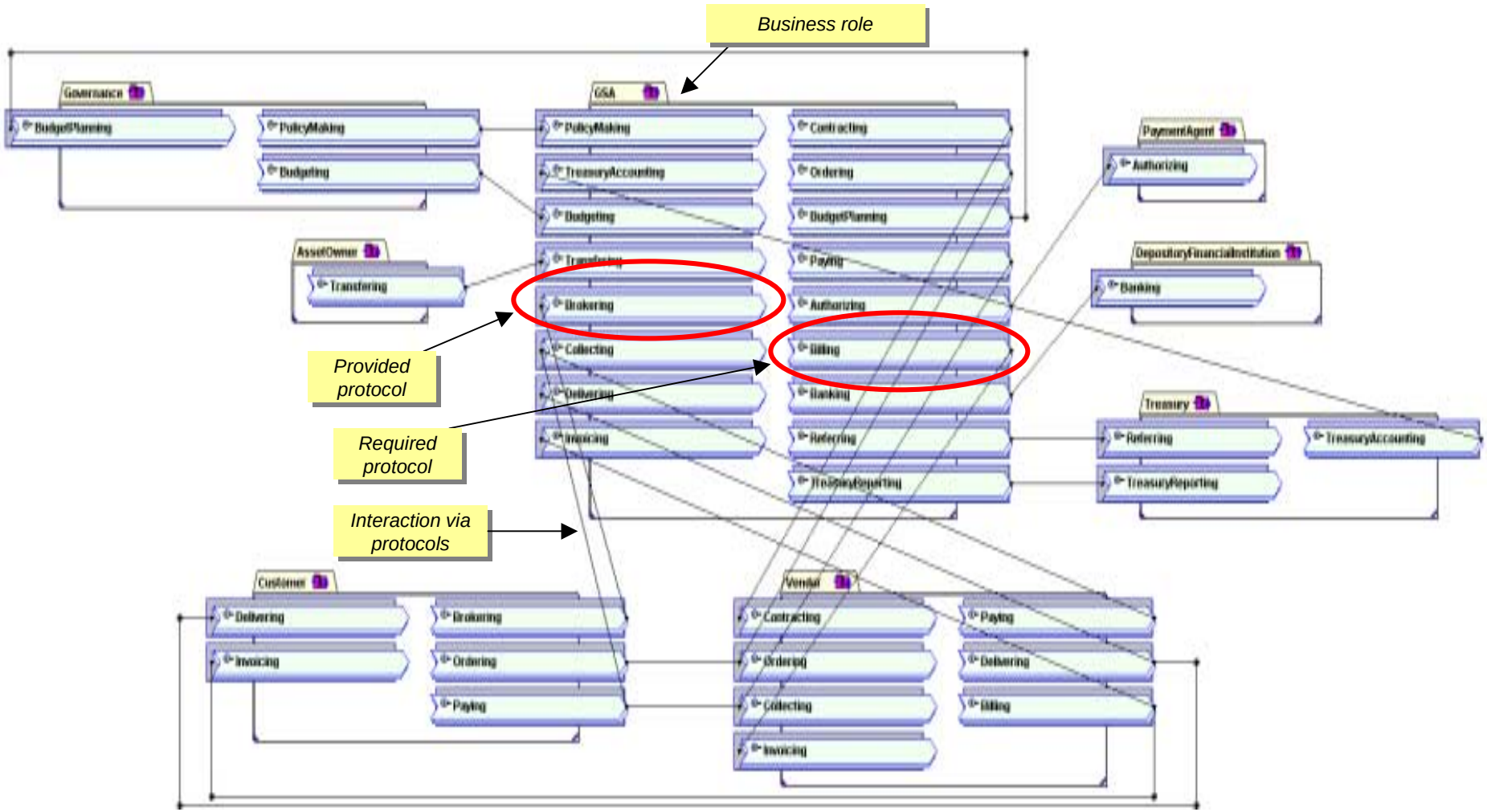
Roles and Collaborations

- **Role:** A specification of the responsibility to perform specific functions in the context of a business process.
- **Collaboration:** A closed set of roles interacting to carry out a business process to achieve some joint purpose.
- **Protocol:** A defined conversation between two roles that may be extended over time (i.e., responses of one party to the other may not be immediate).

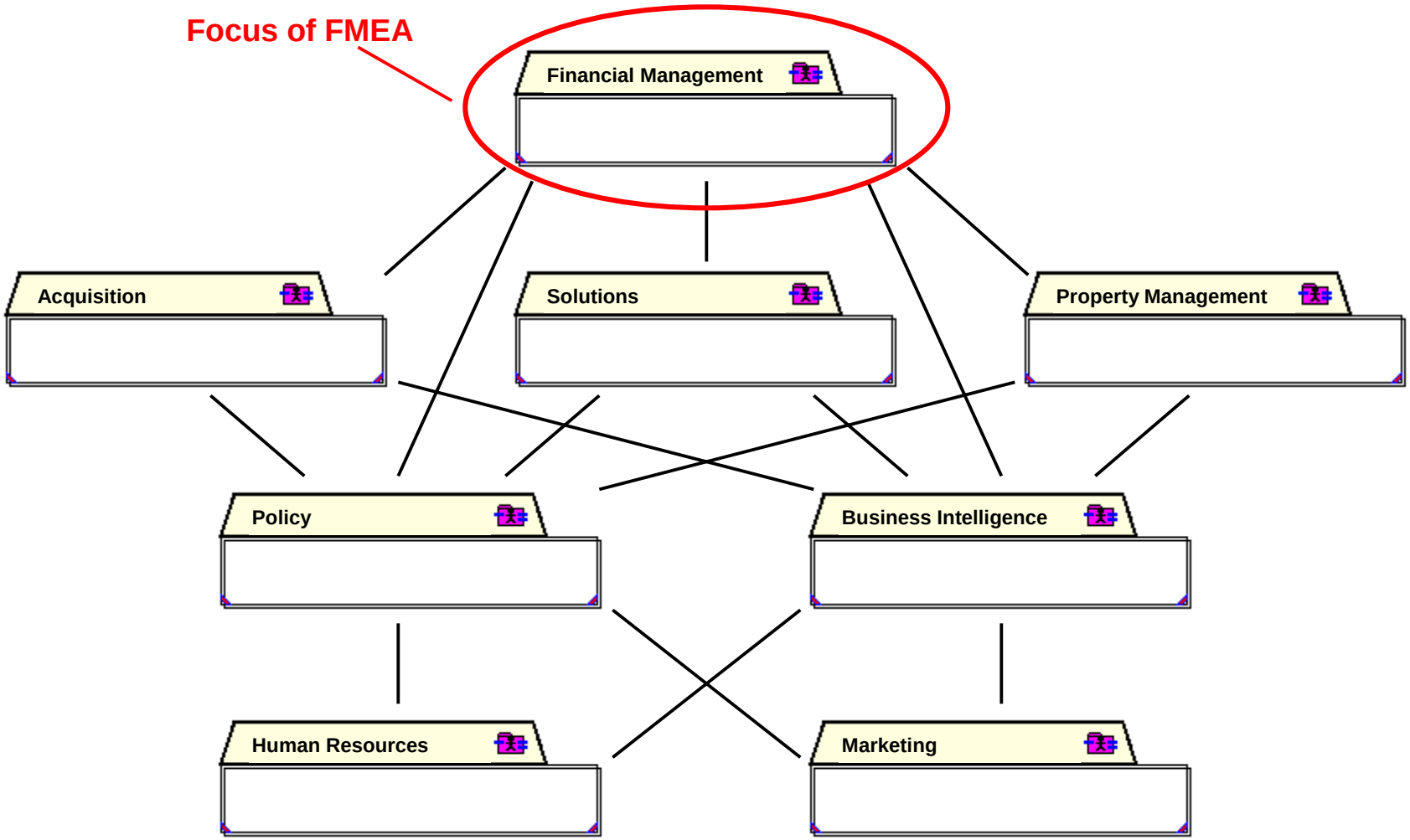
Role Decomposition

- **Business Role:** A role played by a business organization in a specific business environment in which business processes emerge collaboratively, rather than by being managed by any single composite entity.
- **Discipline Role:** A role with responsibility for a major business “discipline” within a business organization, independently of the current physical organizational structure of that business.
- **Enterprise Role:** A role with top-level enterprise responsibility for some set of related business services provided within a certain discipline.
- **Work Role:** A role responsible for carrying out one or more business services within an enterprise role, at roughly the level that could be assigned to an individual worker or supported by a specific function in an information system.

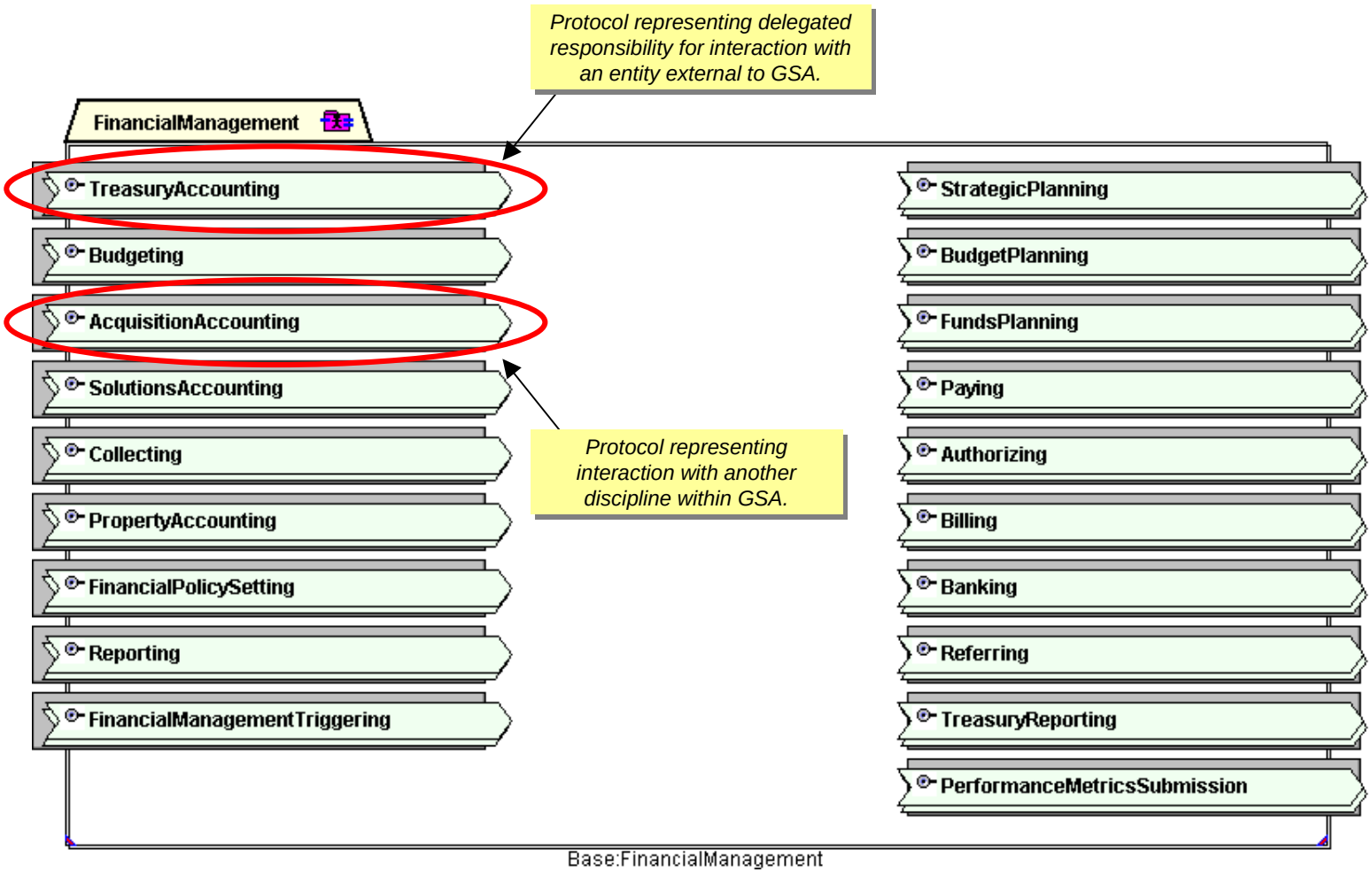
GSA Business Environment



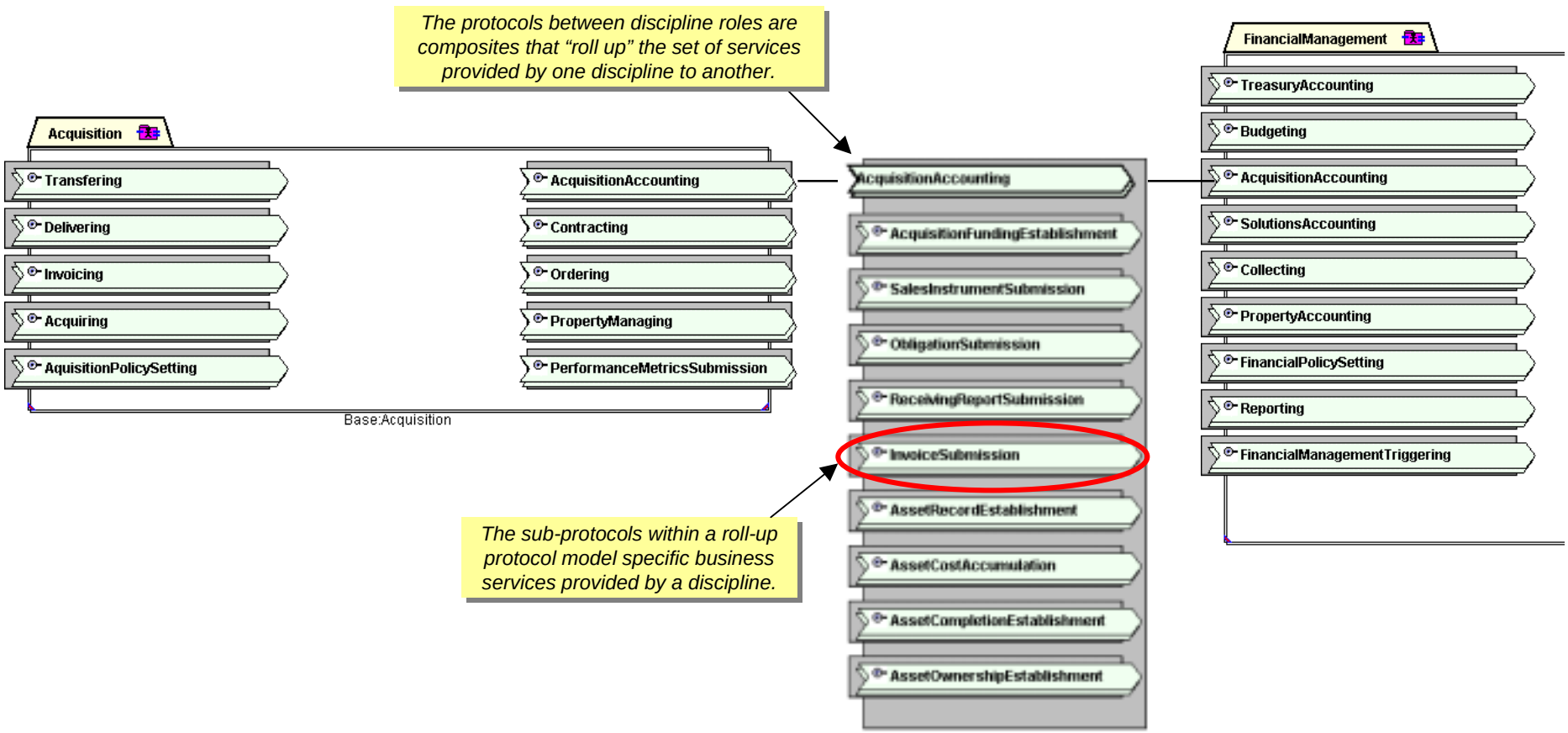
“One GSA” Disciplines (Simplified View)



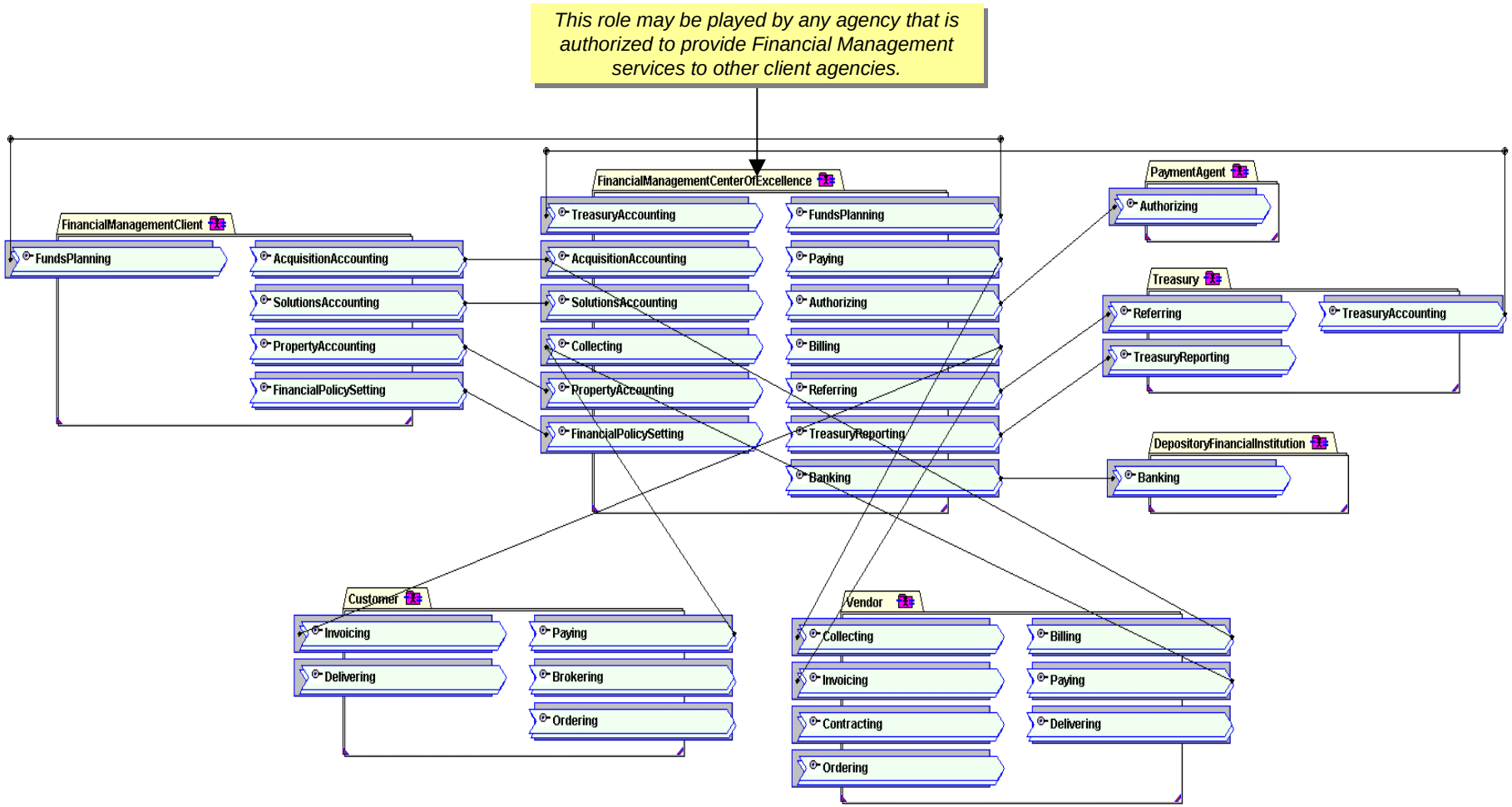
Financial Management Discipline Role



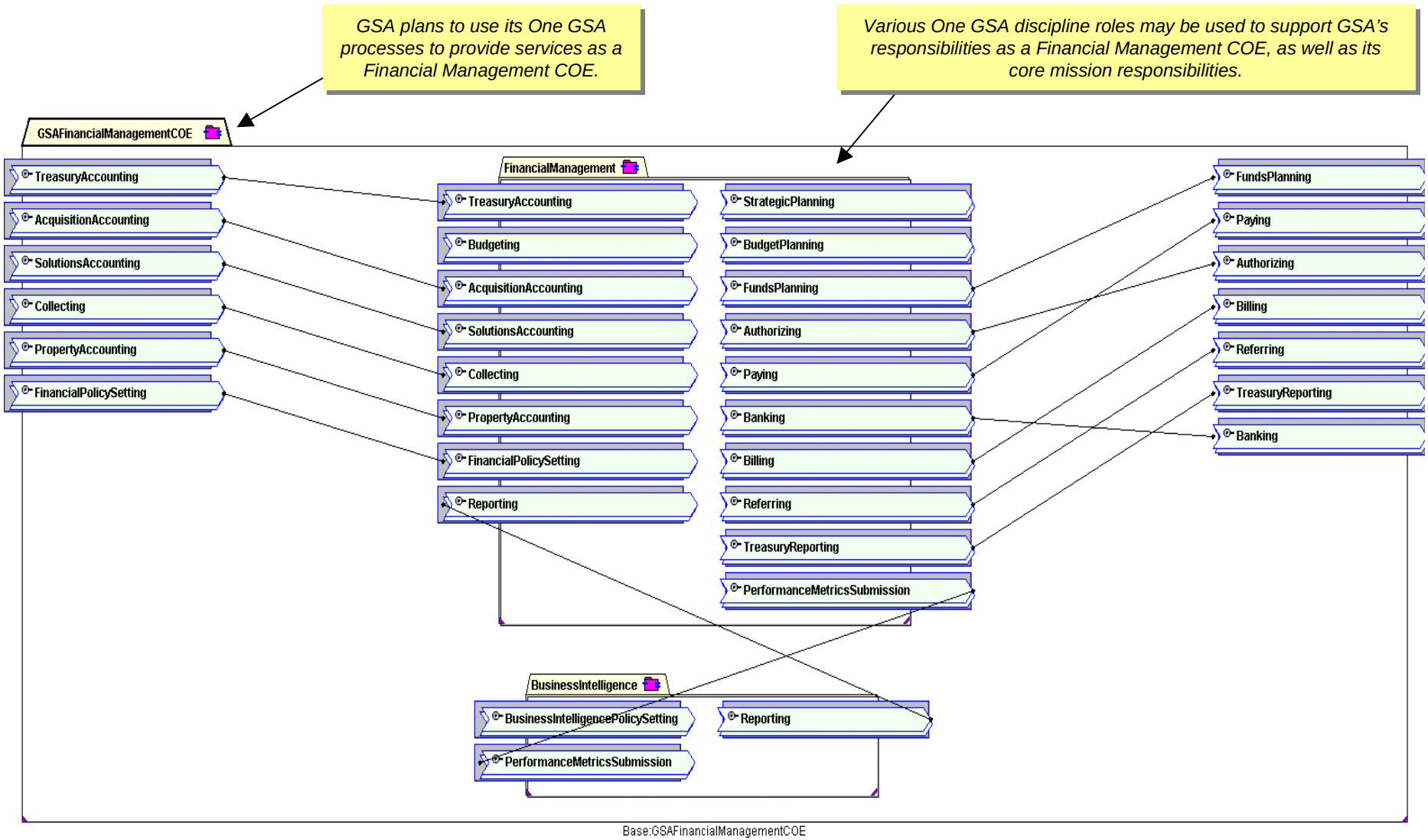
“Roll-Up” Protocol



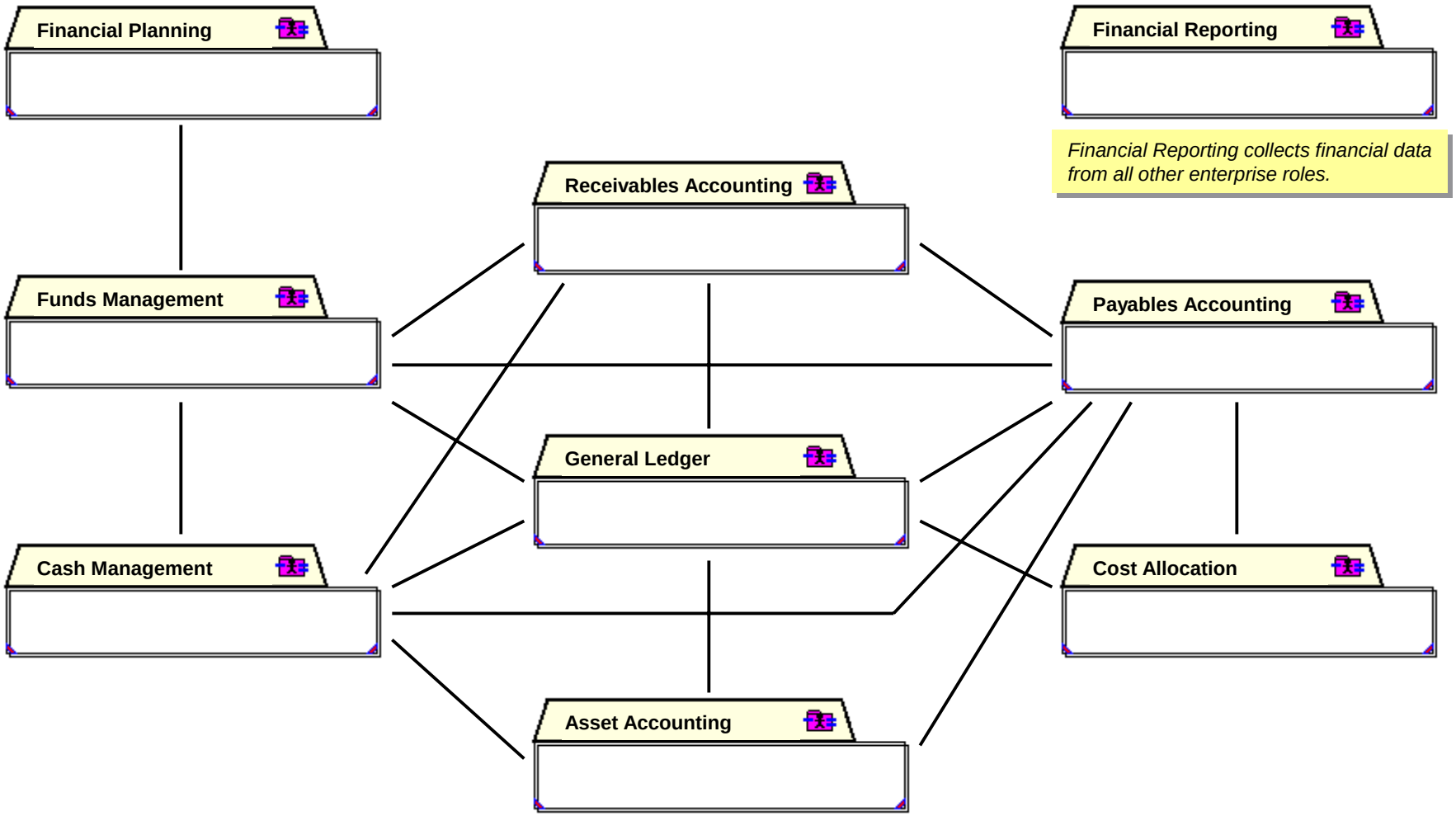
“FM LOB” Business Environment



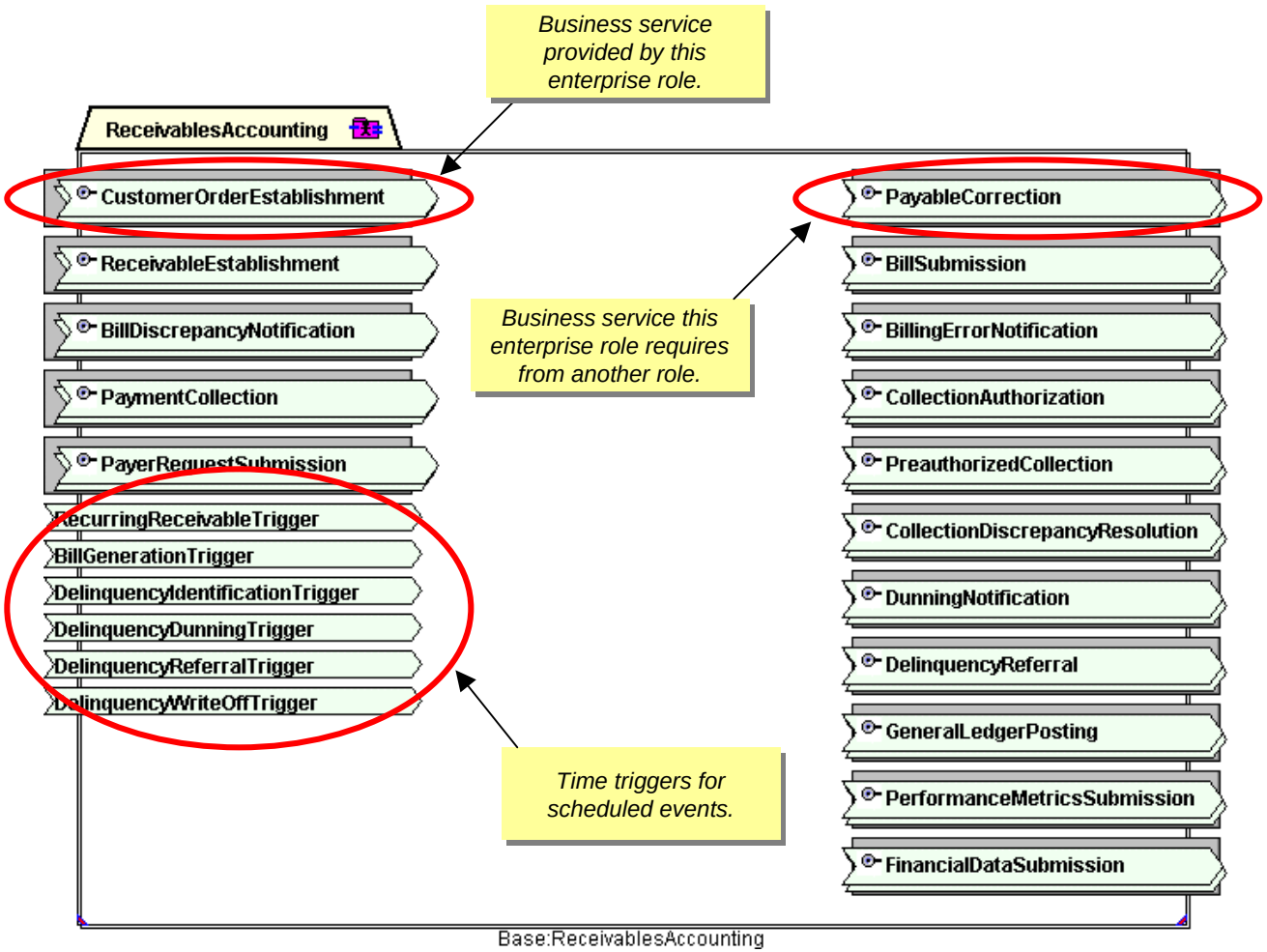
GSA Financial Management COE



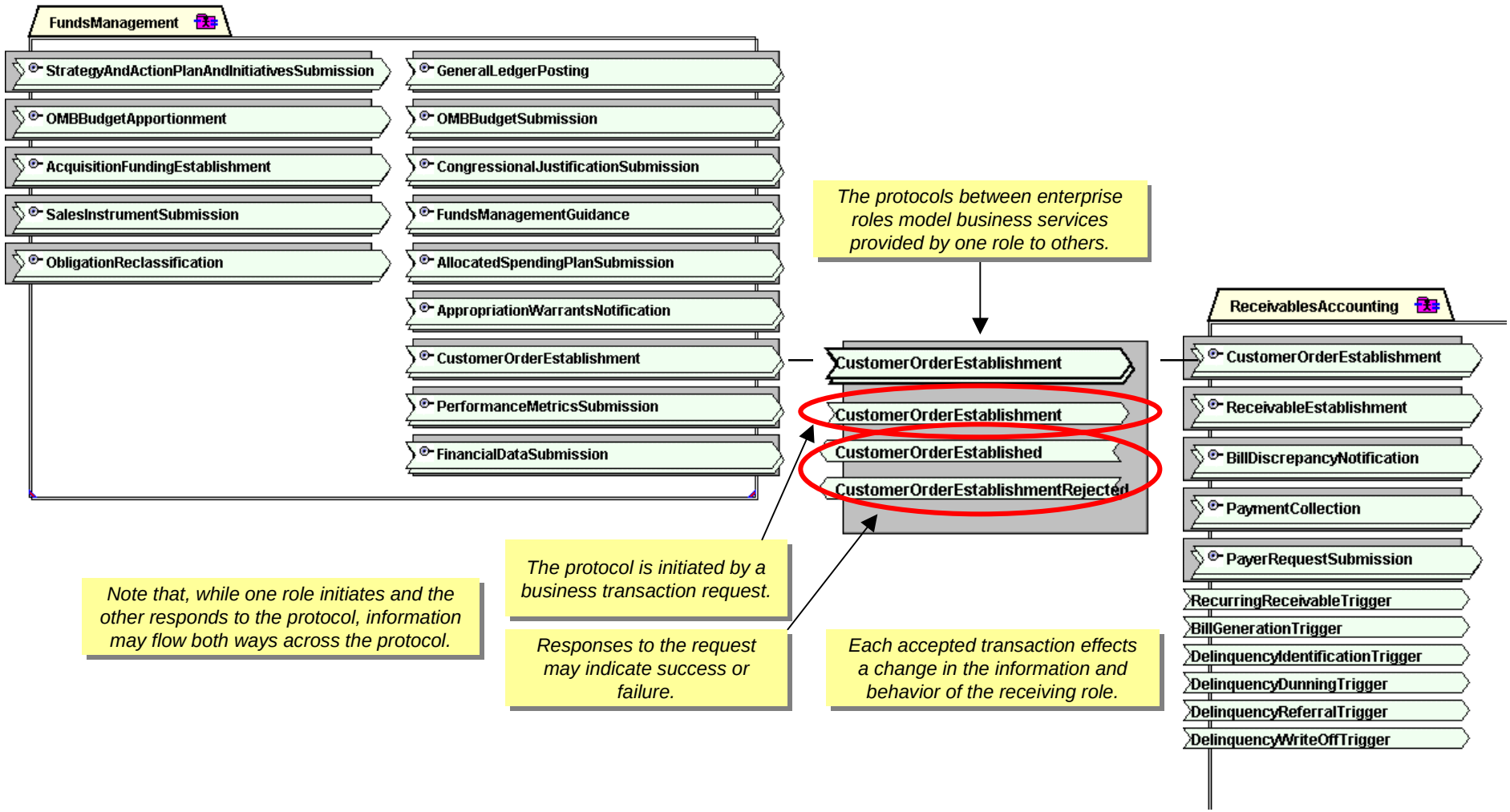
Financial Management Enterprise Roles (Simplified)



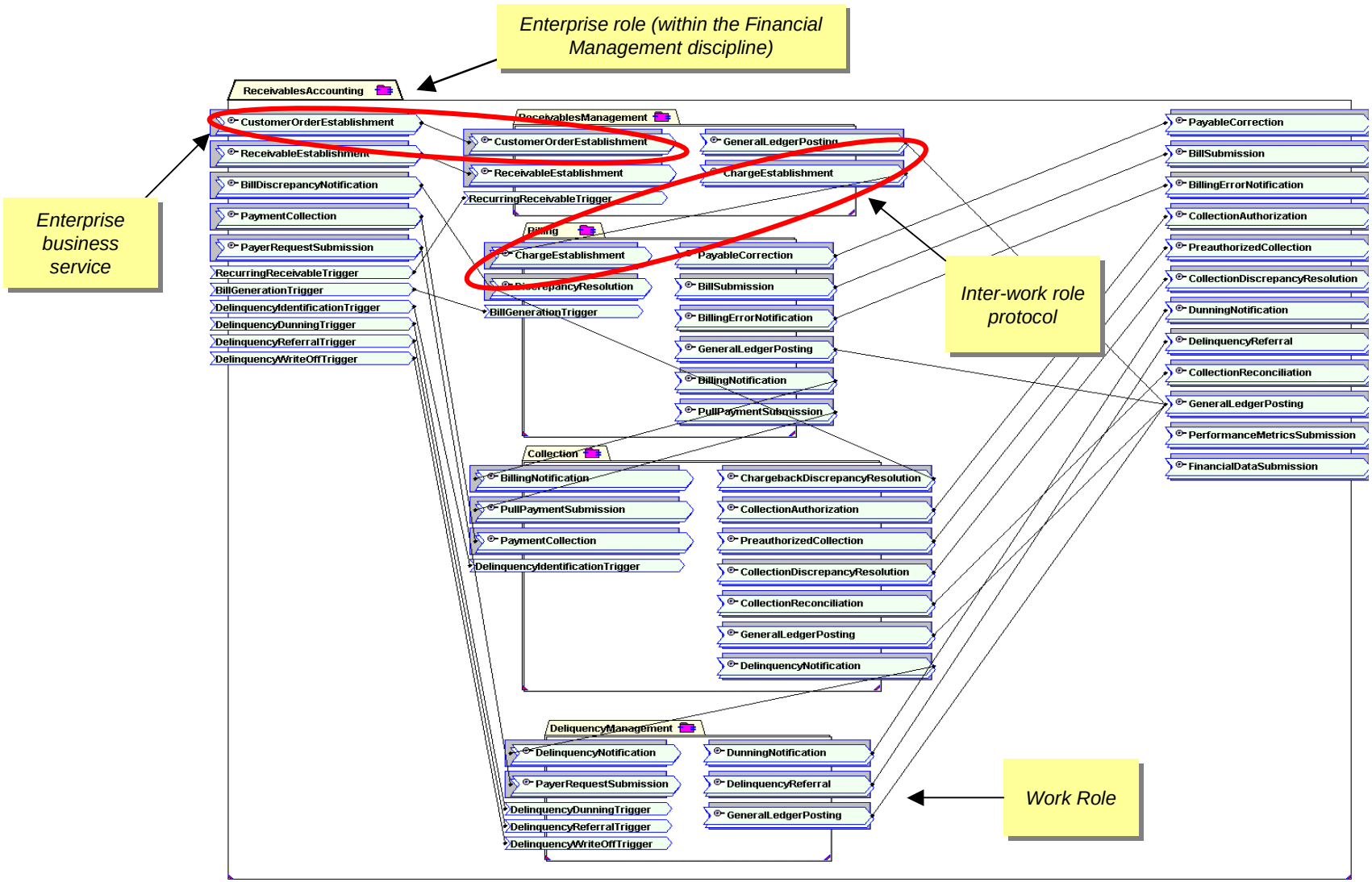
Example Enterprise Role



Example Business Service Protocol



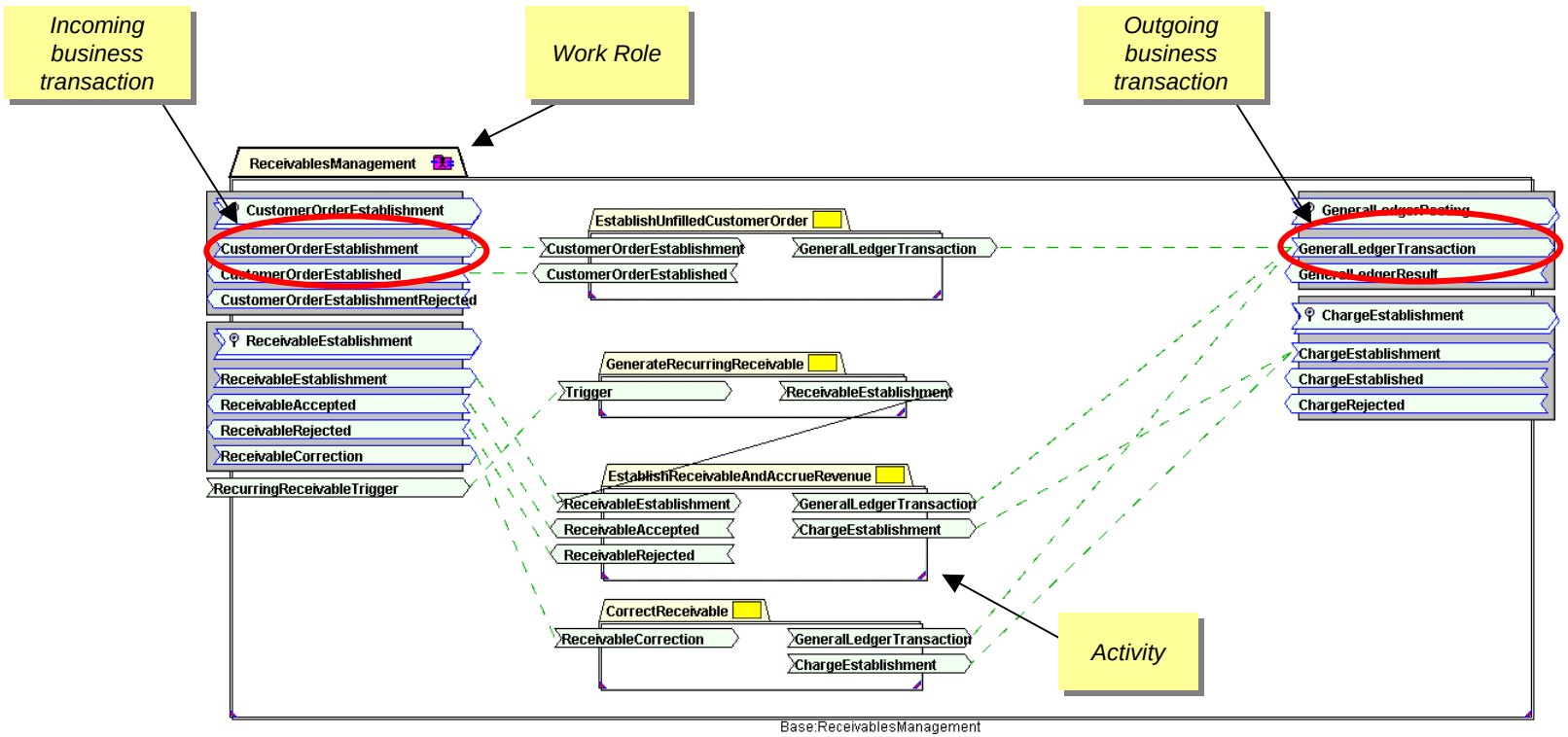
Example Work Roles



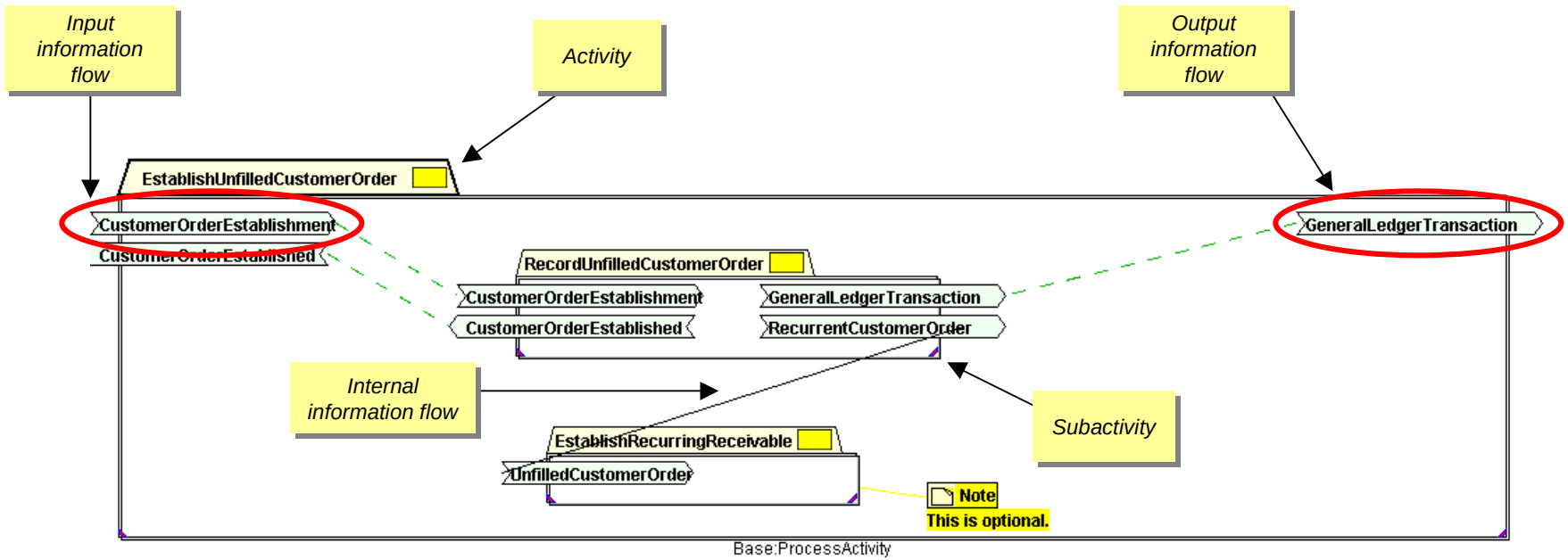
Activities and Choreographies

- **Activity:** A specification of a business function in the context of a role.
- **Choreography**: A specification of the sequencing of external interactions required in order to carry out given business responsibilities.
  - A work role is choreographed in terms of the activities required to perform the business services provided by the work role.
  - A complicated activity may be choreographed in terms of subactivities.
  - A subactivity (or simple activity without subactivity decomposition) is choreographed directly in terms of the event-triggered sequencing of its acceptance of inputs and sending of outputs.

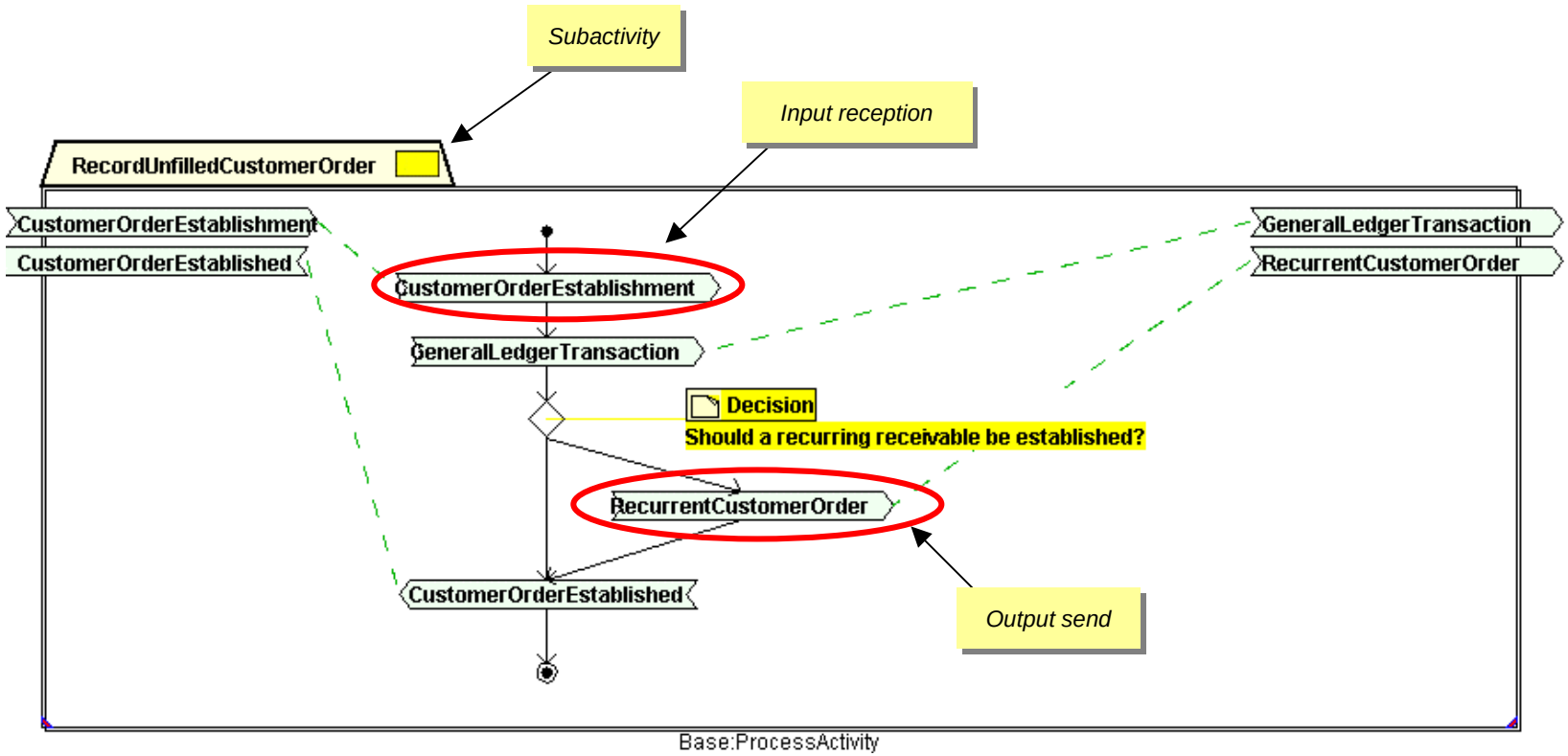
Example Activities



Example Subactivities



Example Subactivity Choreography



Example Subactivity Requirements

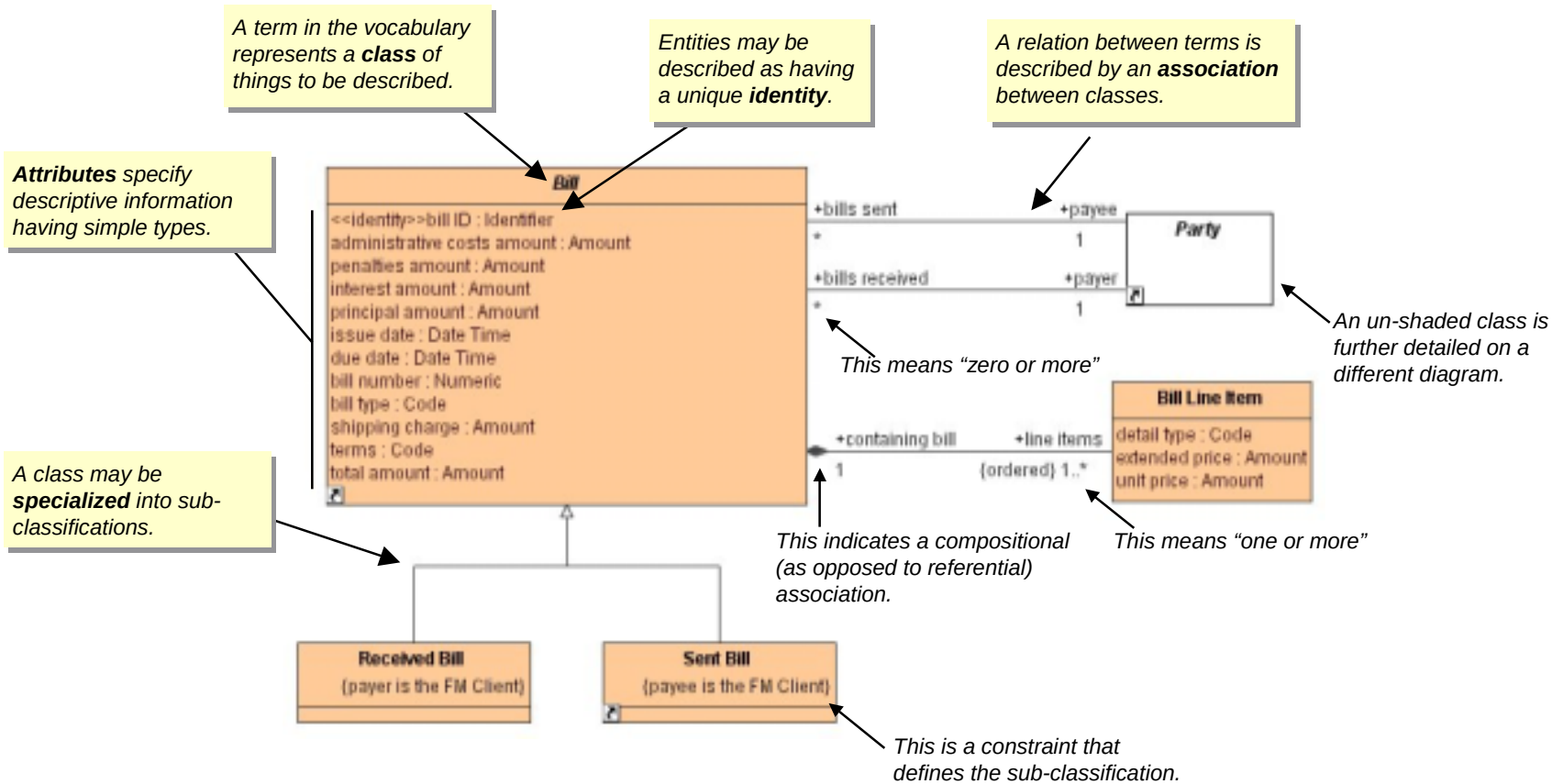
Description: Record a new unfilled customer order, as established via a specific sales instrument.

Generate general ledger transactions to increase Unfilled Customer Orders and decrease Anticipated Reimbursements.

Requirement

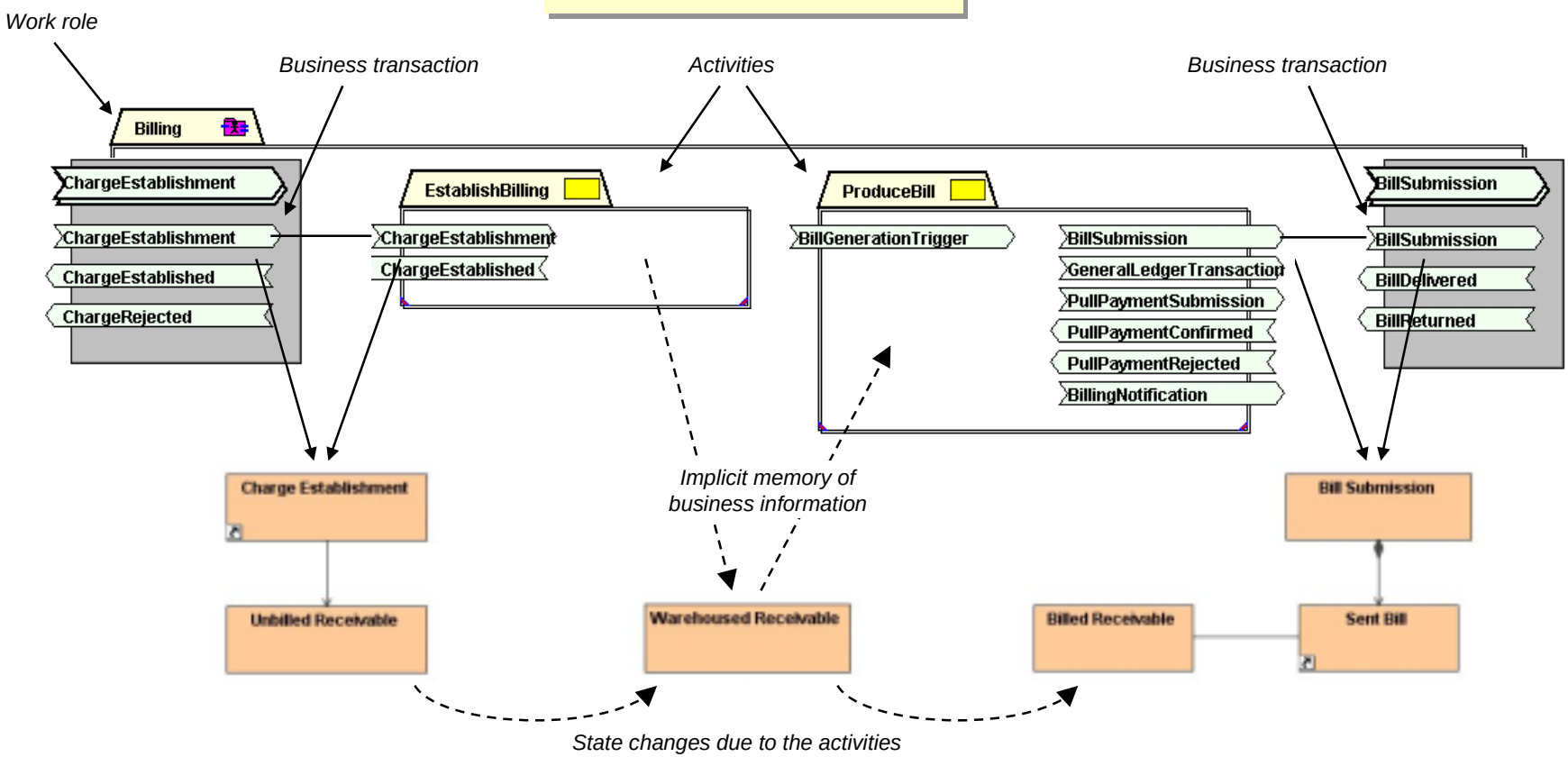
RMA-03	Reimbursable agreement information. Capture and accumulate reimbursable agreement information that includes the following: * Billing limit * Billing terms * Customer order amount * Amount obligated * Amount expended * Advances collected * Advances applied to earned revenue * Remaining balance on advances * Amount earned * Amount billed * Accounts receivable * Collections on receivables. Enable access to reimbursable agreement information by customer ID number, reimbursable agreement number, project, or fund.	JFMIP Core Requirements 2005

Information Model



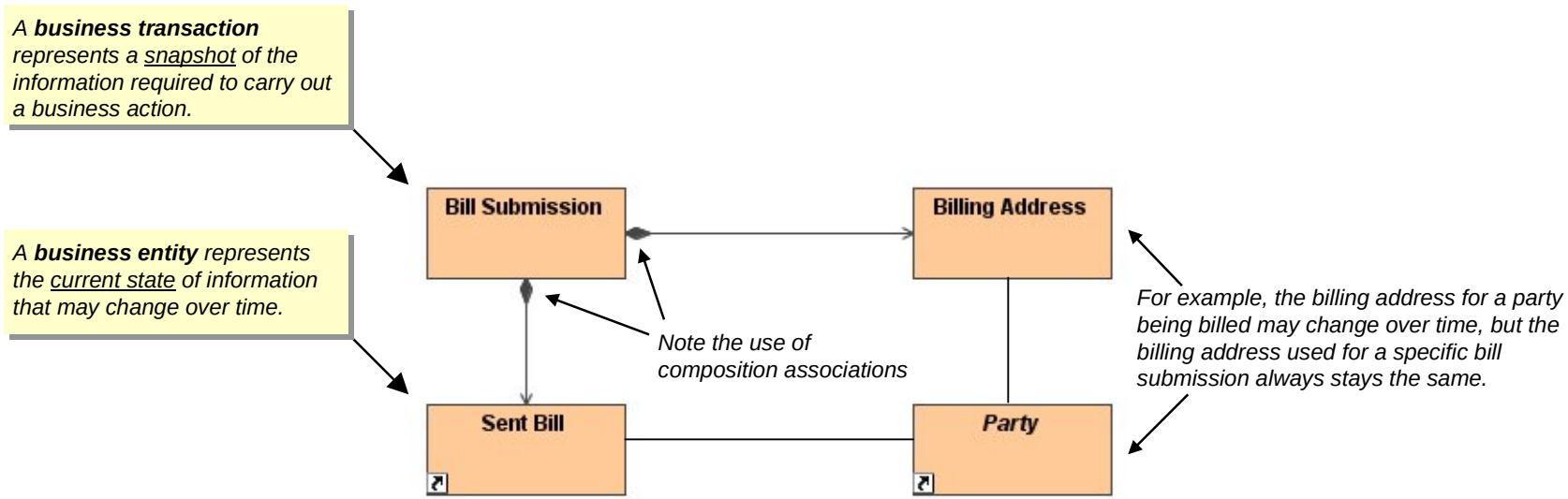
Information Model: What Is It For?

*The **process model** describes how business activities are (or are to be) carried out.*

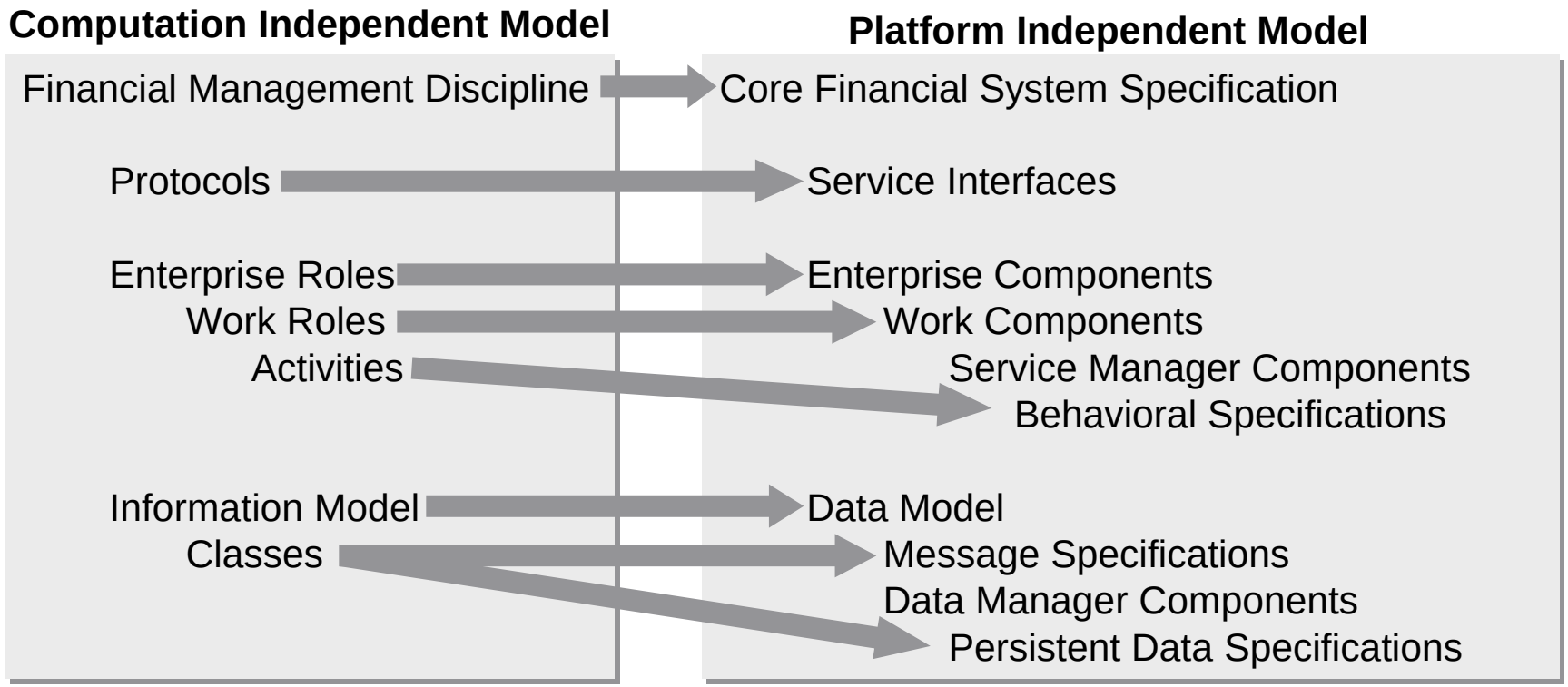


*The **information model** details the vocabulary of the business entities and transactions used in the process model.*

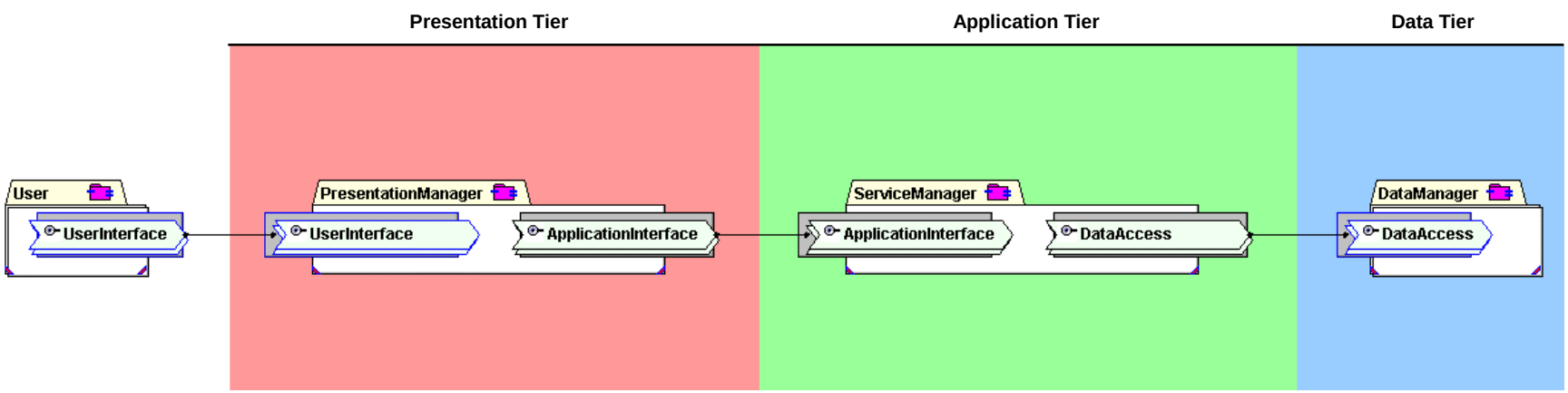
Information Model: Entities and Transactions



Platform Independent Model



Three-Tier Component Architecture

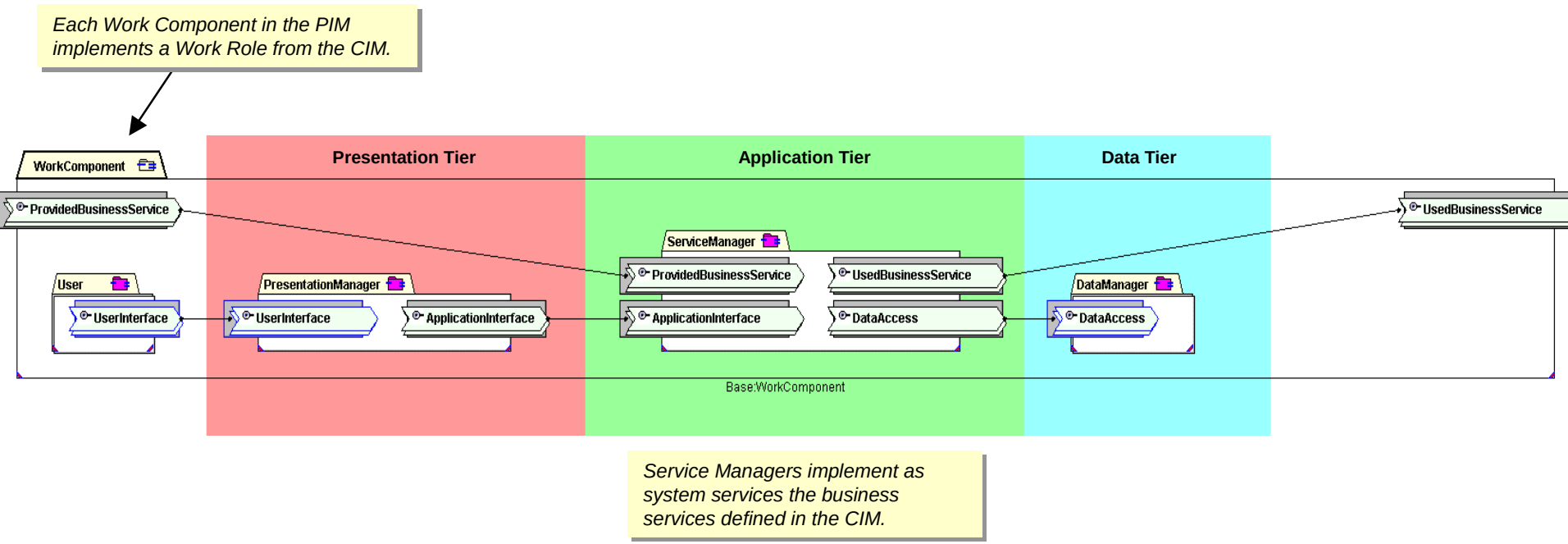


Presentation Manager components provide user access to application services.

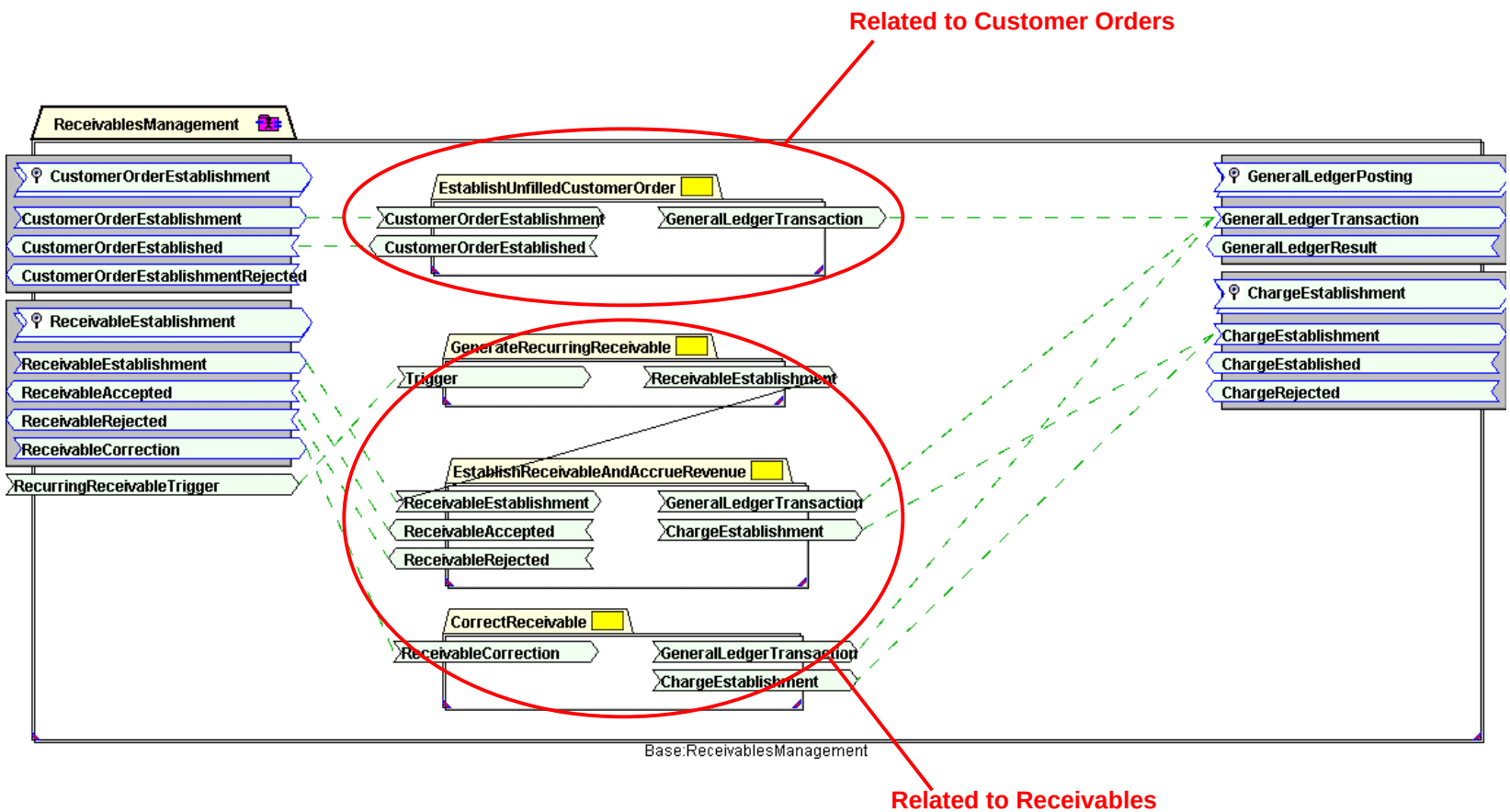
Service Manager components provide transactional implementation of application services.

Data Manager components persist data between application transactions.

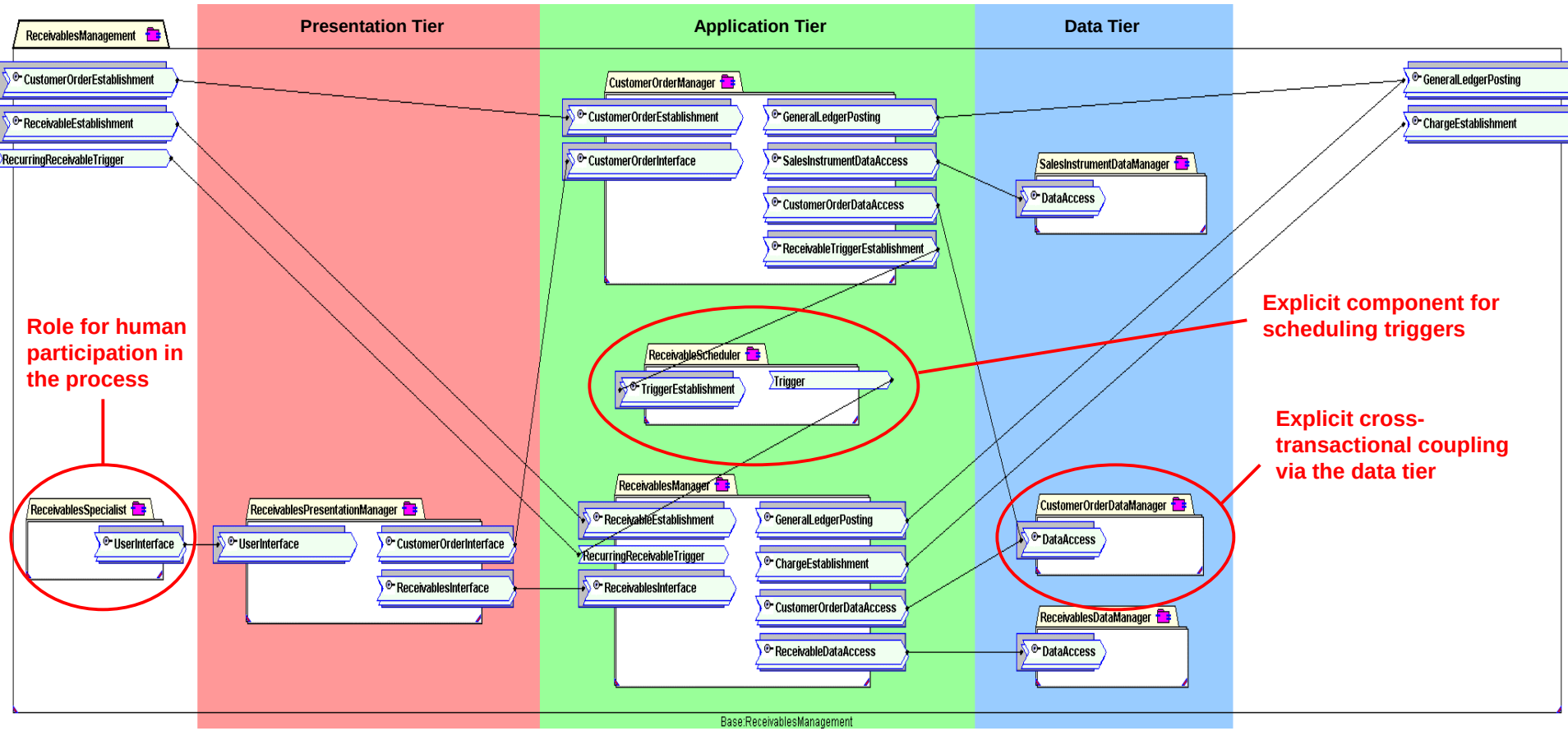
Service-Oriented Component Architecture



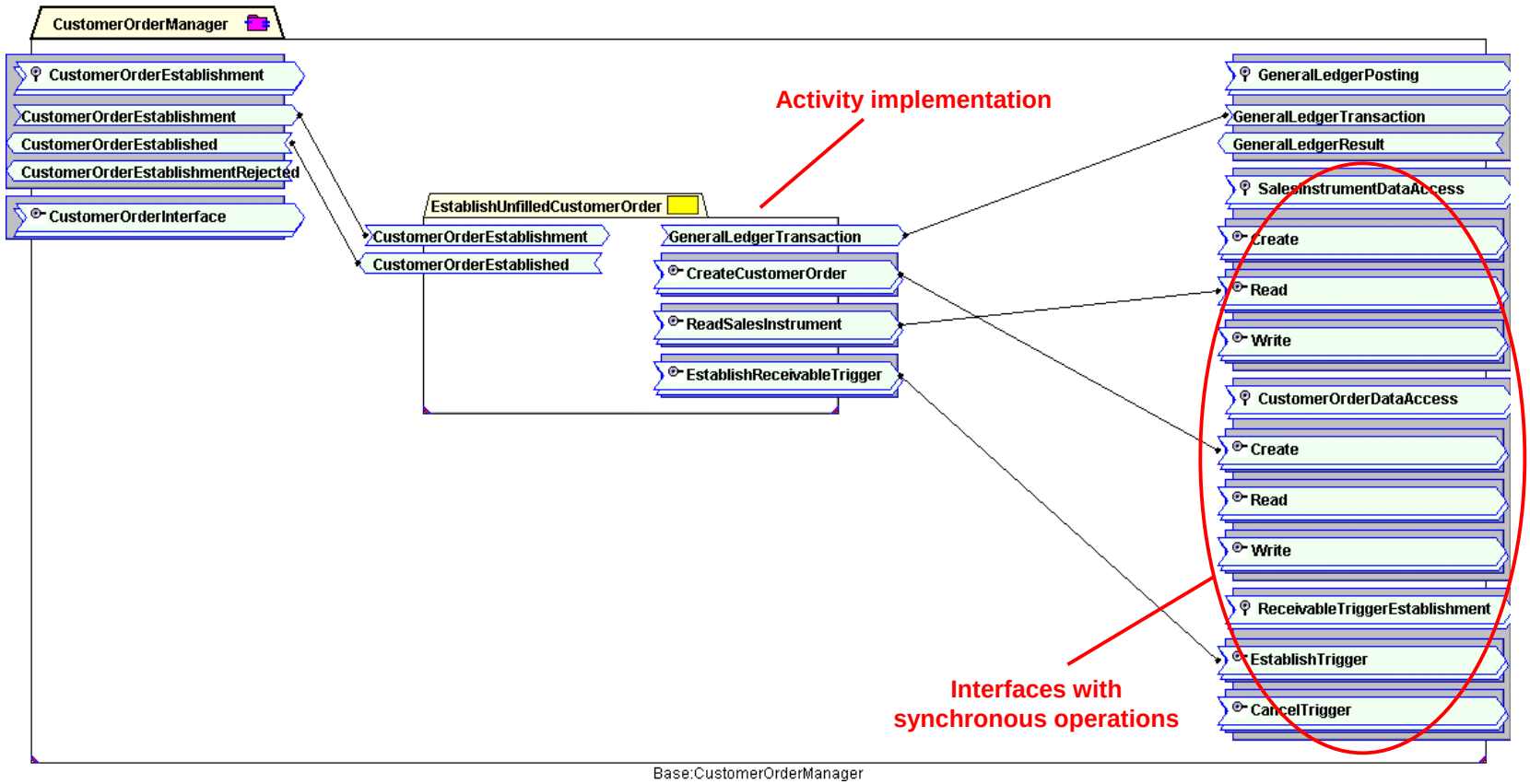
Example Work Role (from CIM)



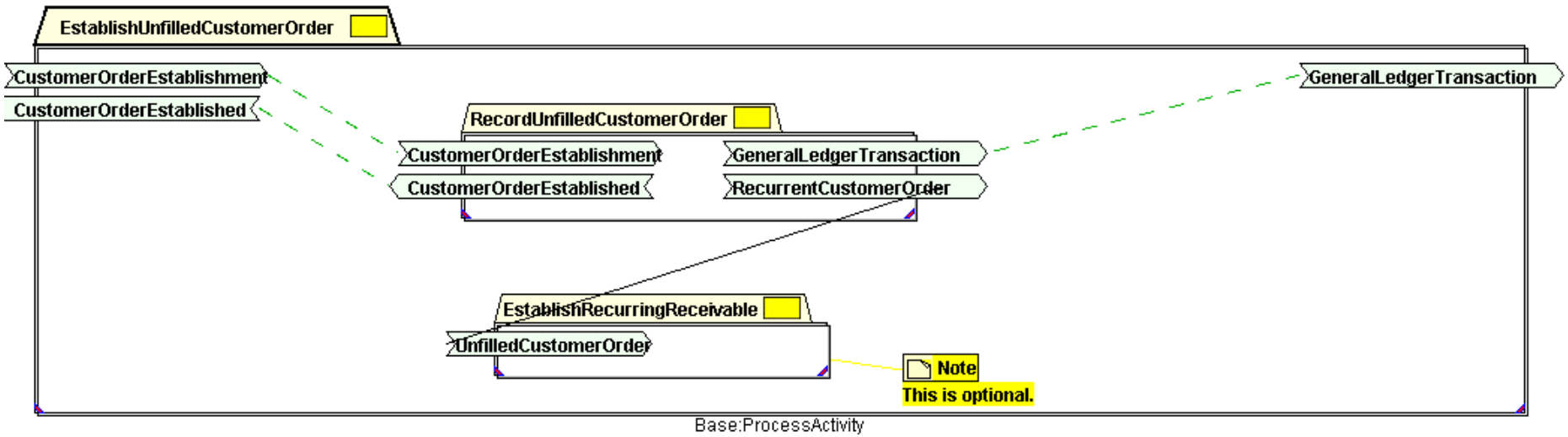
Corresponding Work Component



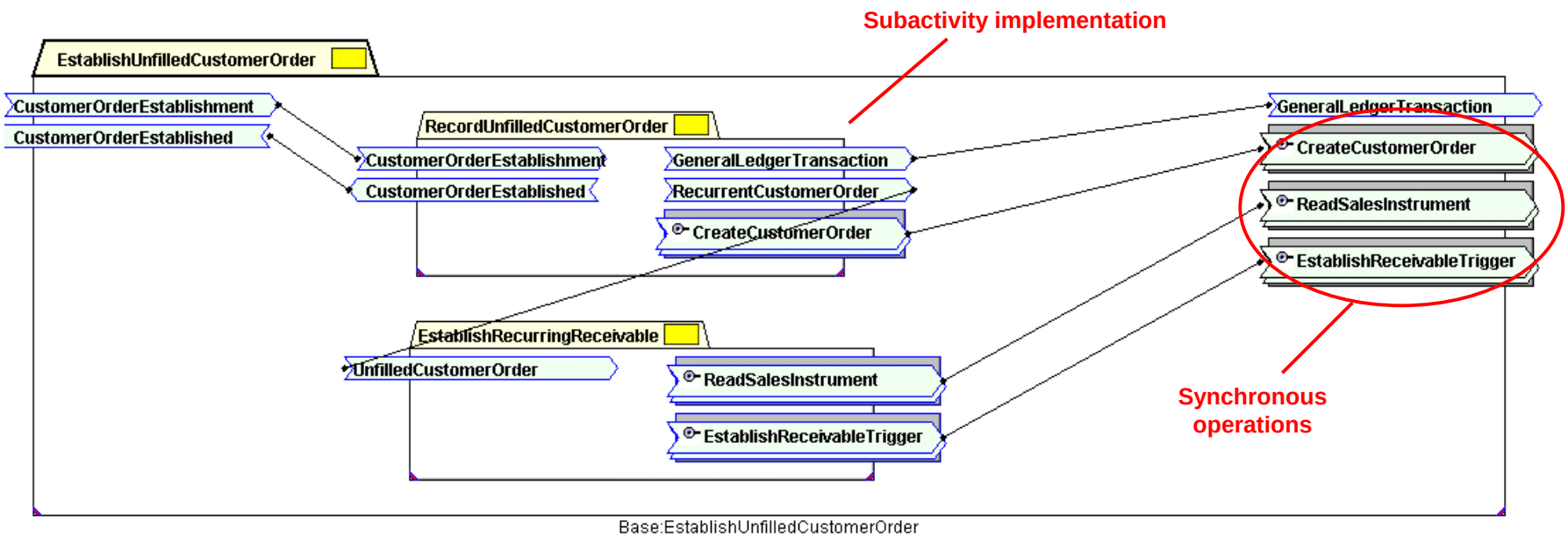
Example Service Manager Implementation



Example Activity (from CIM)

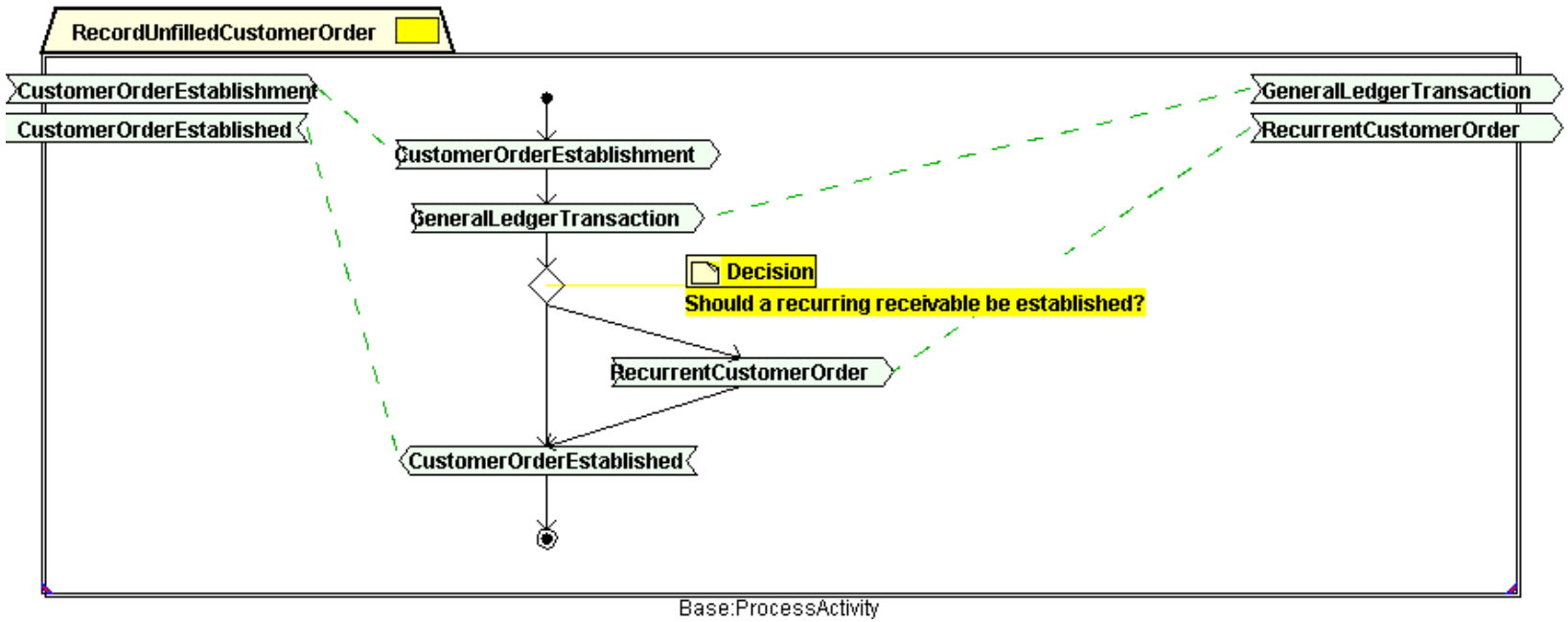


Example Activity Implementation

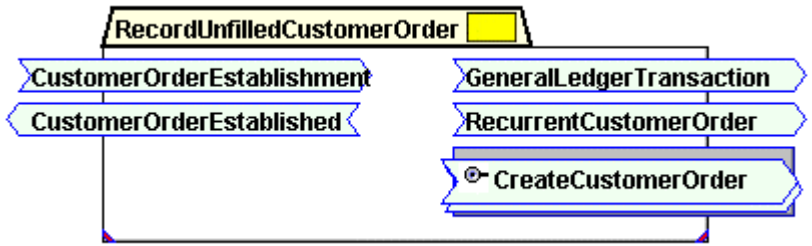


An activity implementation must conform to the choreography of the activity it implements. Directly implementing the subactivities makes this manifest.

Example Subactivity Choreography (from CIM)

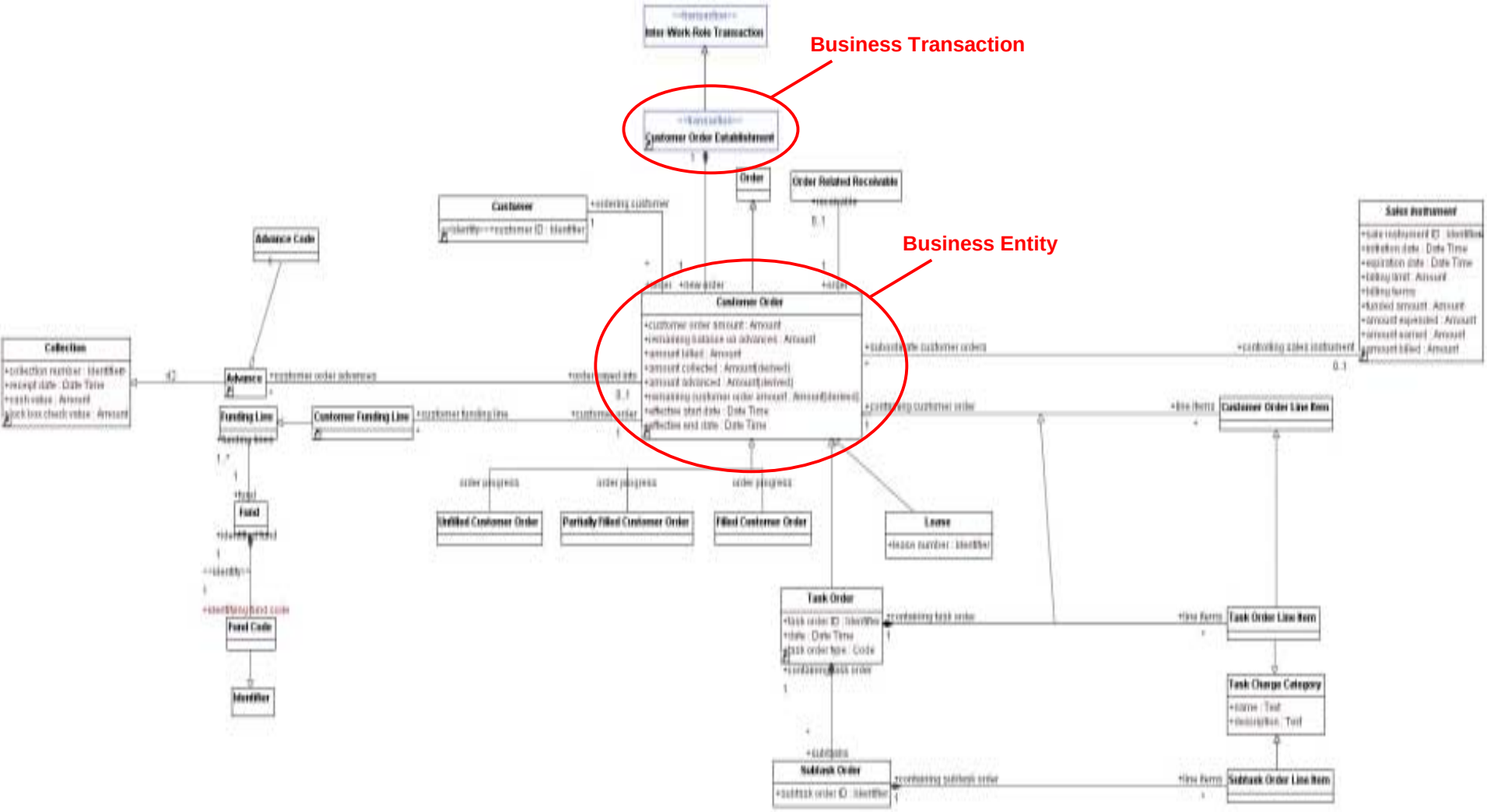


Example Subactivity Functional Specification

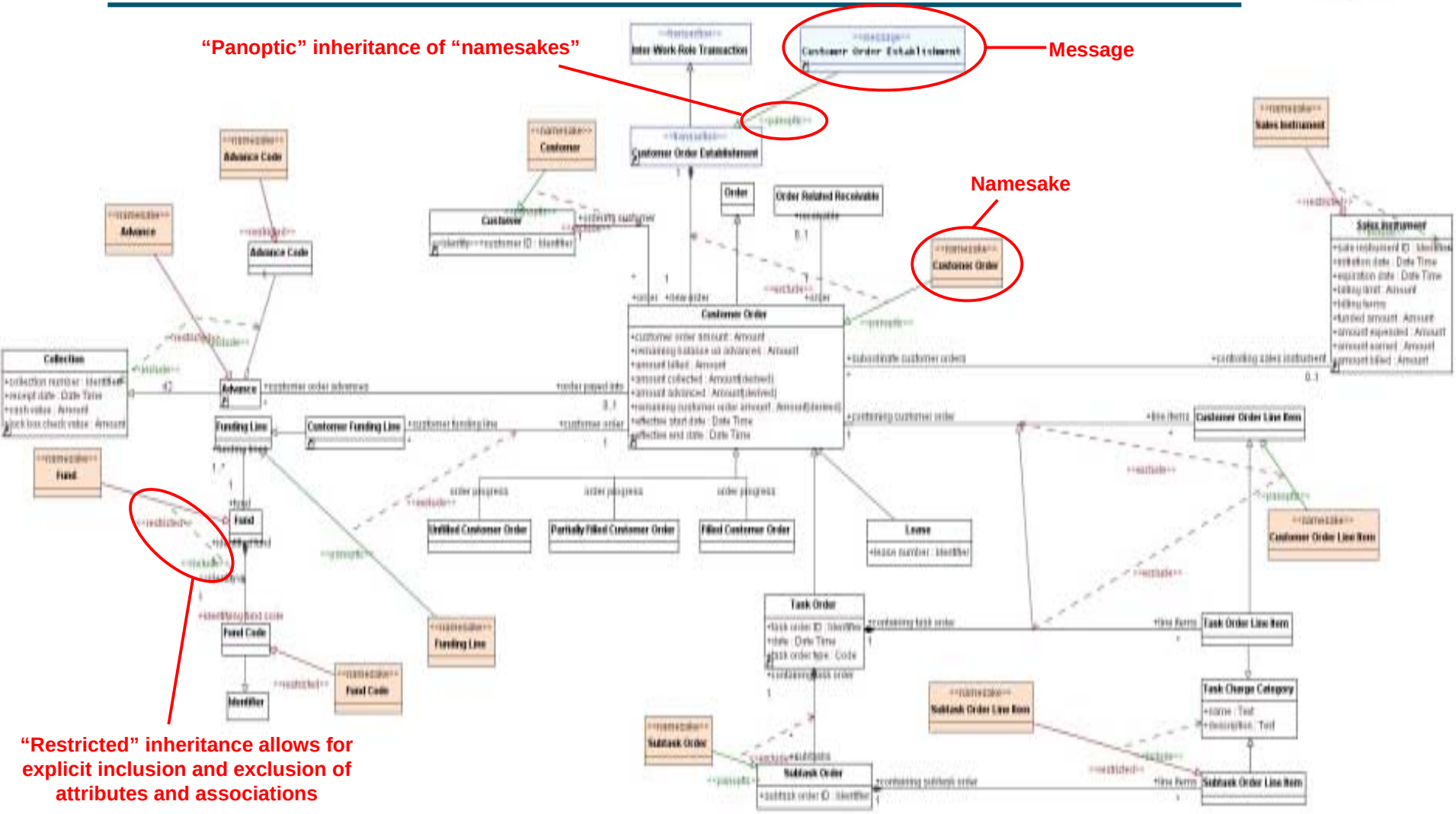


1. **Receive** CustomerOrderEstablishment
2. **Let** newOrder = CreateCustomerOrder(CustomerOrderEstablishment.newOrder).data
3. **Send** GeneralLedgerTransaction to increase Unfilled Customer Orders and decrease Anticipated Reimbursements
4. **Send** newOrder as RecurrentCustomerOrder
   (**Note**: EstablishRecurringReceivables will check if there are actually any creation triggers.)
5. **Send** CustomerOrderEstablished

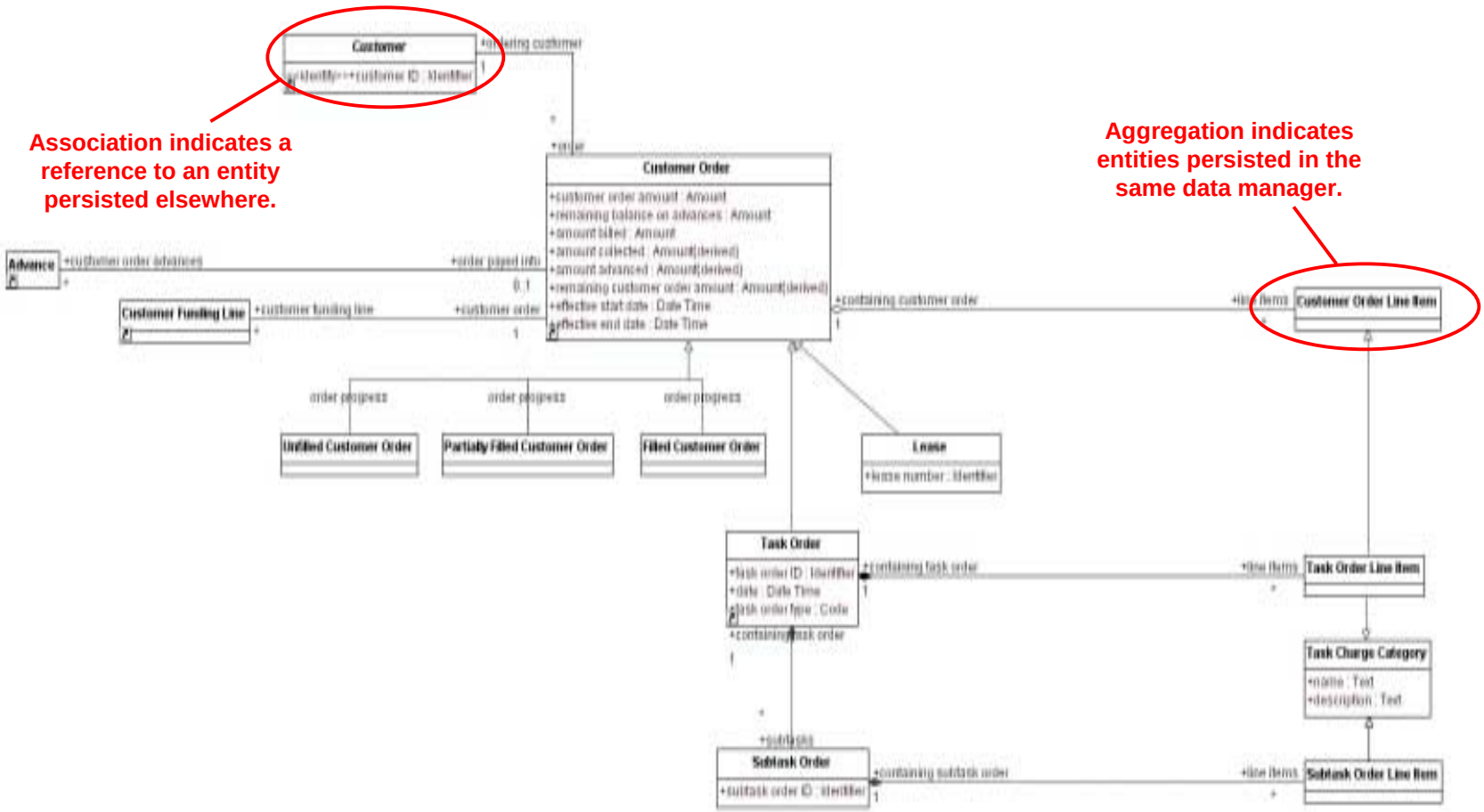
Example Transaction Information Model (from CIM)



Example Transaction Message Model



Example Persistence Model



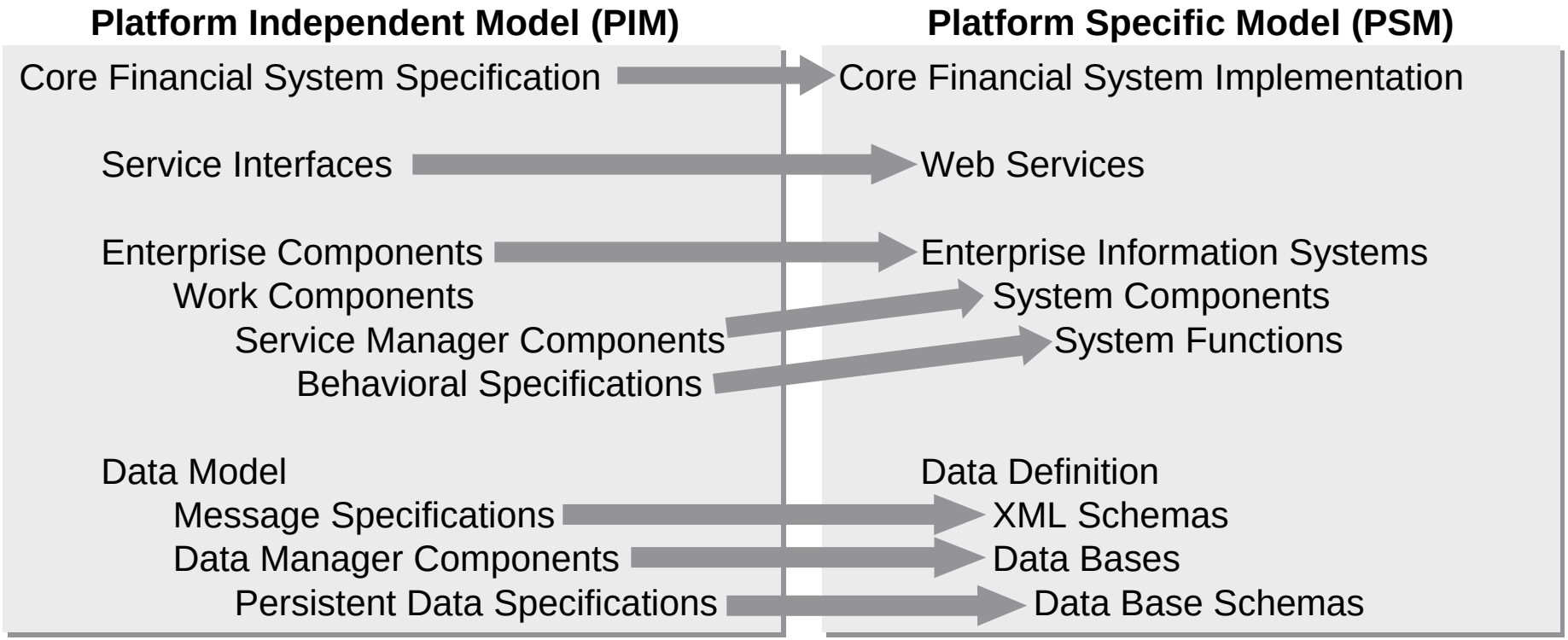
DataAccessTechnologies

Where Business Meets Technology

Realizing a Business Architecture With SOA Technologies

Roles to Enterprise Components & Services

Platform Specific Model



Example Web Services Definition

```
<wsdl:portType
  name="CustomerOrderEstablishment.CustomerOrderEstablishment">
    <wsdl:operation name="CustomerOrderEstablishment">
      <wsdl:input
  message="tns:CustomerOrderEstablishmentPanc
                    name="CustomerOrderEstablishment">
      </wsdl:input>
    </wsdl:operation>
 </wsdl:portType>


 <wsdl:portType
  name="CustomerOrderEstablishment.CustomerOrderEstablishmentCallbac
  k">
    <wsdl:operation name="CustomerOrderEstabli
      <wsdl:input
  message="tns:CustomerOrderEstablishedPanopticInheritanceCluster“
                    name="CustomerOrderEstablished">
      </wsdl:input>
    </wsdl:operation>
    <wsdl:operation name="CustomerOrderEstablishmentRejected">
```

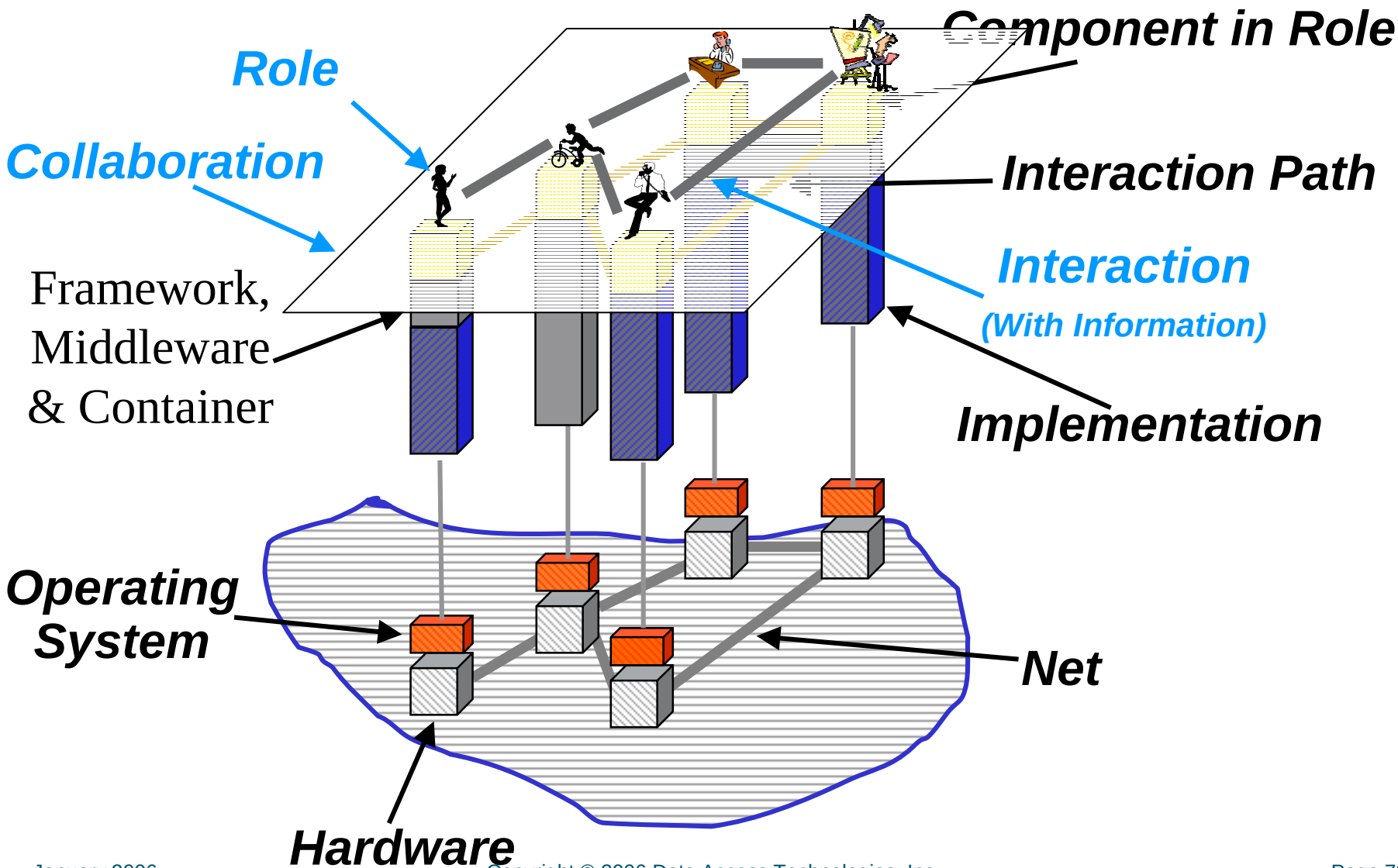
The primary port type has operations corresponding to the request flows in the protocol.

The callback port type has operations corresponding to the response flows in the protocol.

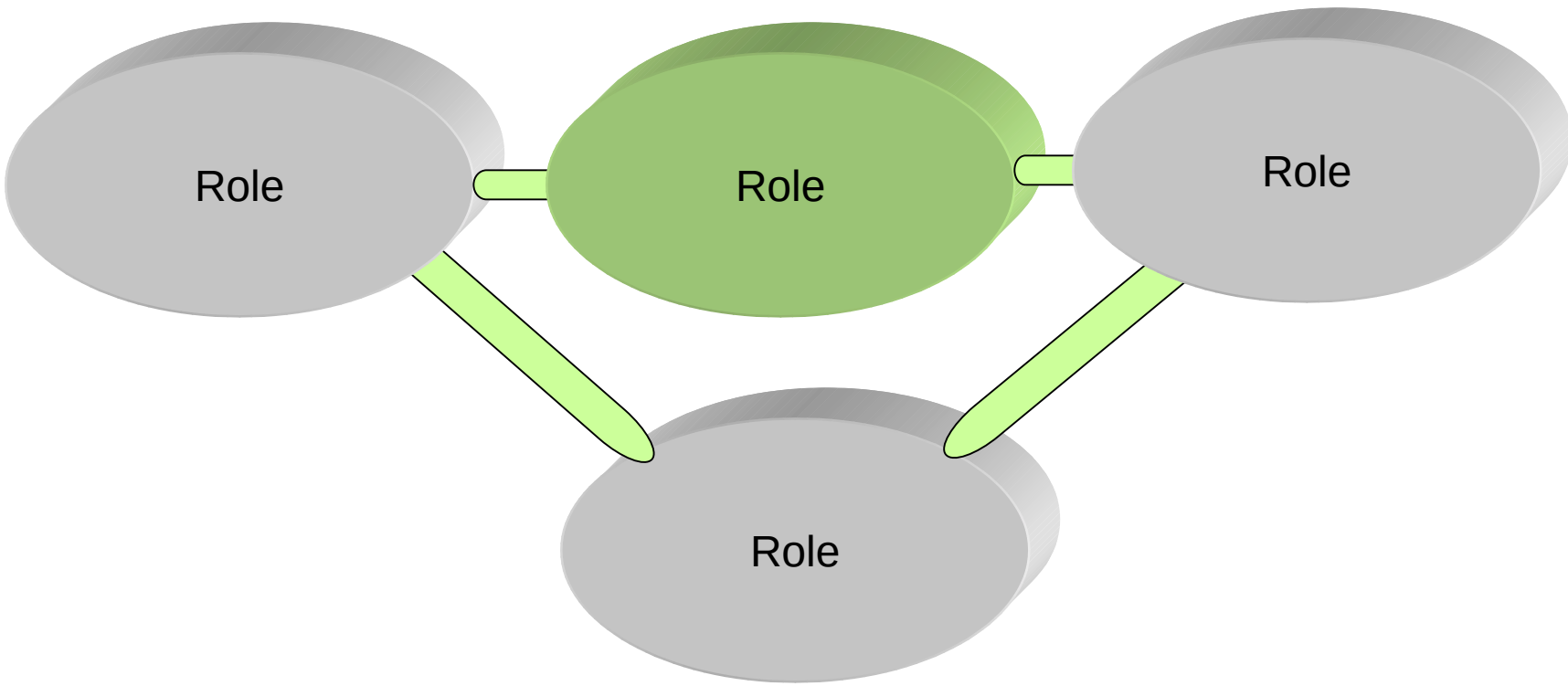
Example Transaction Message XML Document

```
<CustomerOrderEstablishment>
    <customerOrderEstablishment>
        <newOrder>
            <customerOrder>
                <customerOrderID> … </customerOrderID>
                <customerOrderAmount> … </customerOrderAmount>
                    <orderingCustomer>
                    <customer>
                            <customerID> … </customerID>
                    </customer>
                    <party>
                        <name> … </name>
                    </party>
                    </orderingCustomer>
                <controllingSalesInstrument>
                        <salesInstrumentID> … </salesInstrumentID>
                    </controllingSalesInstrument>
                …
                    <lineItems>
                        …
                    </lineItems>
            </customerOrder>
            </newOrder>
```

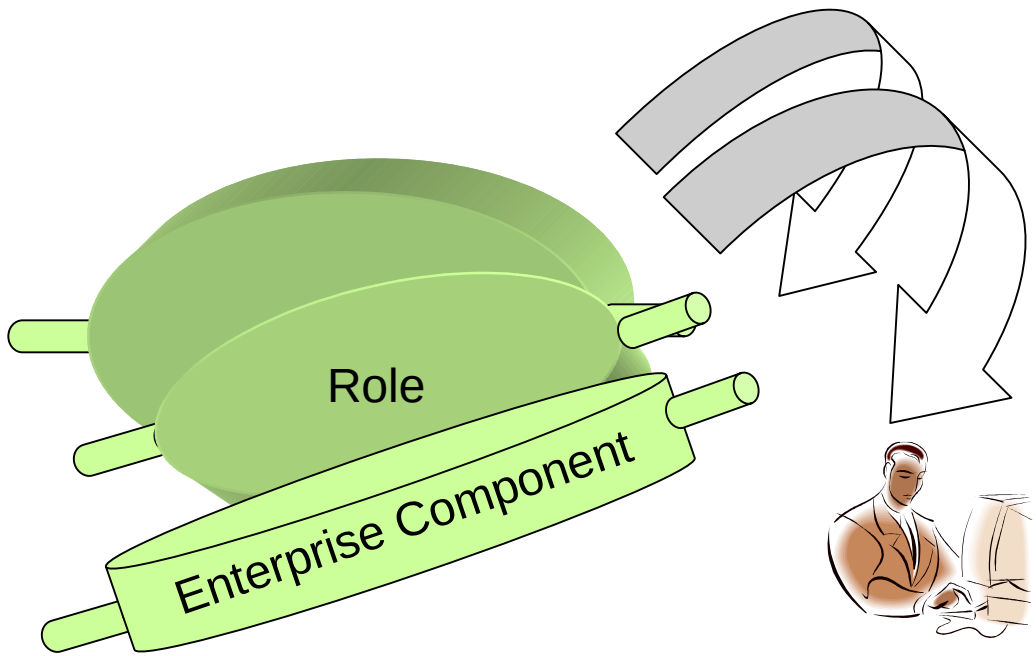
Roles to Systems



Business (CIM) view -Collaborating Roles



“Upper” PIM View - Enterprise Component



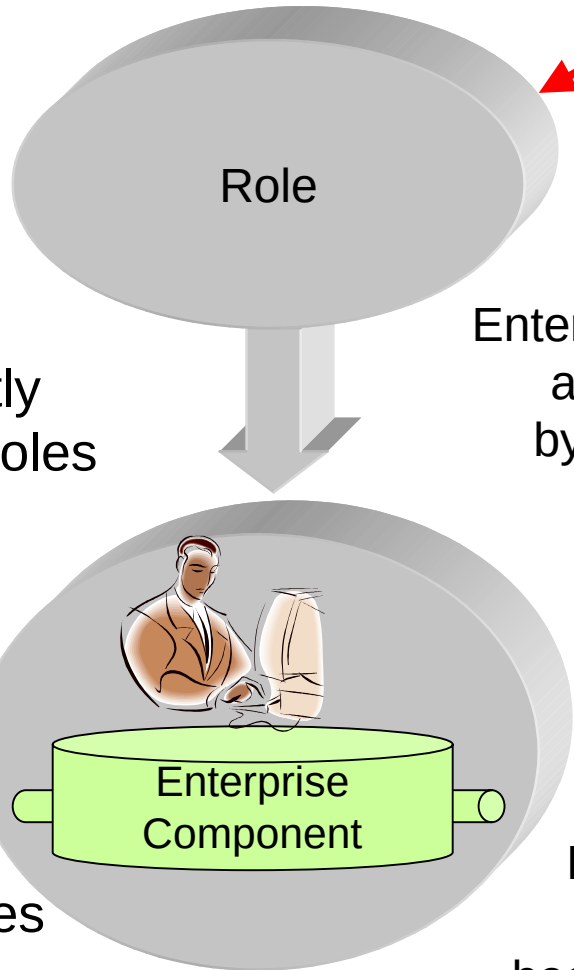
“Rotate” to look
At other aspects
of the component

People, organizations
And/or enterprise components
play roles in Business
Processes.

The “Enterprise Digital Assistant”

Business Process

People, Organizations
And systems play roles



Enterprise components help people and organizations play roles by automating and monitoring The business process

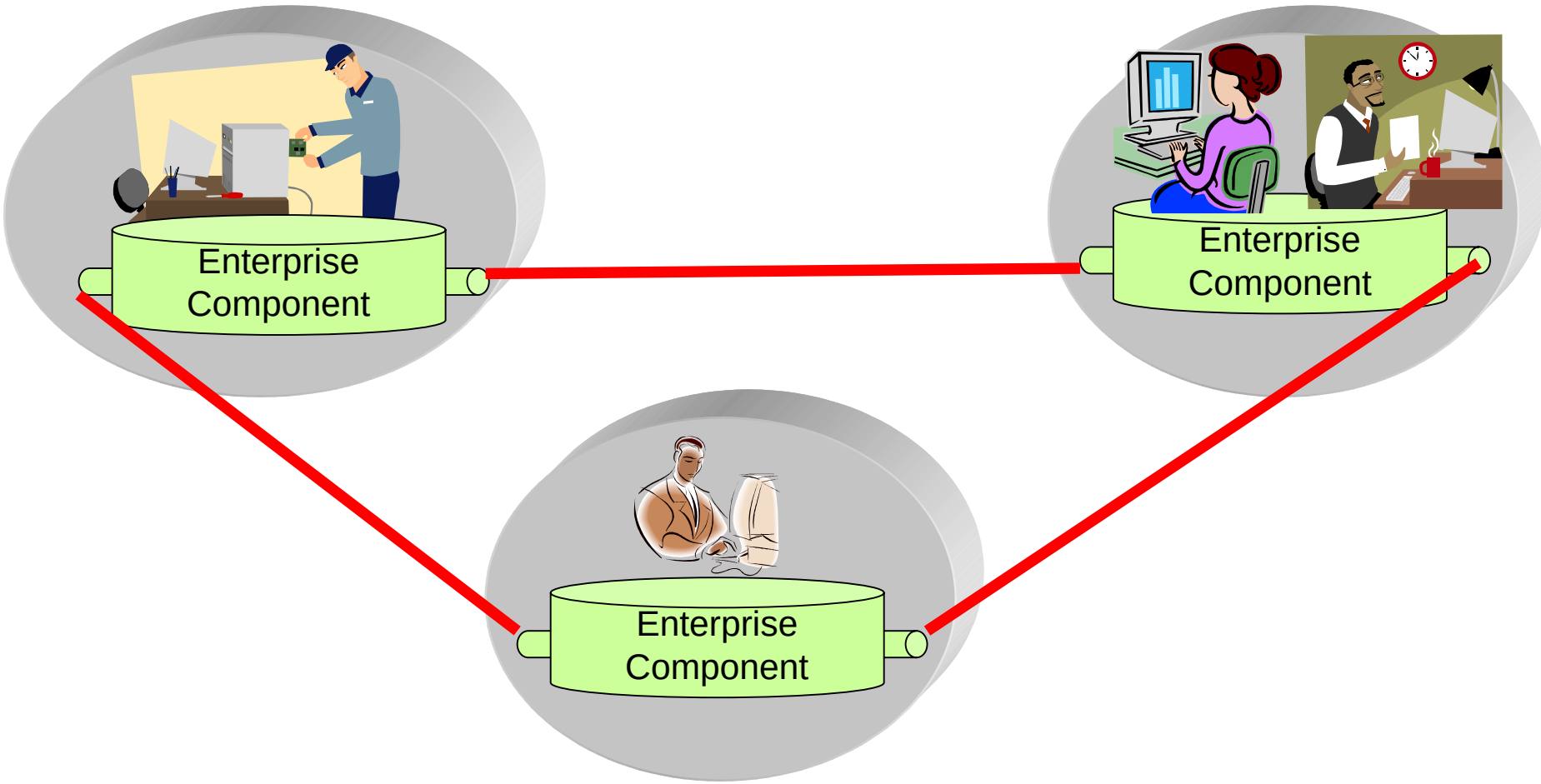
Components frequently help people play these roles

People, organizations and systems components work together to realize roles

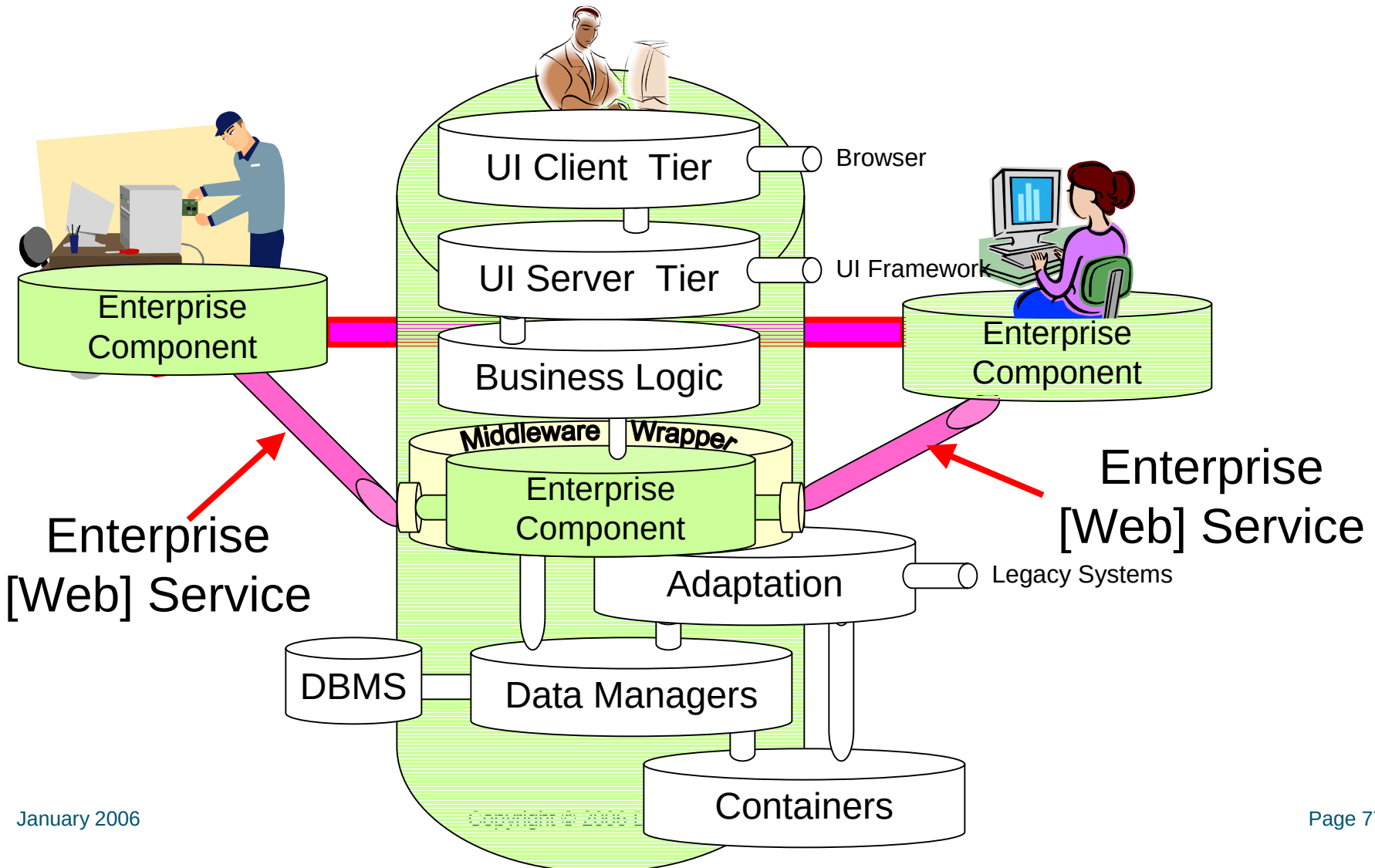
Components are the peoples Automated assistant

From the system perspective. People and organizations become part of the implementation Of the role

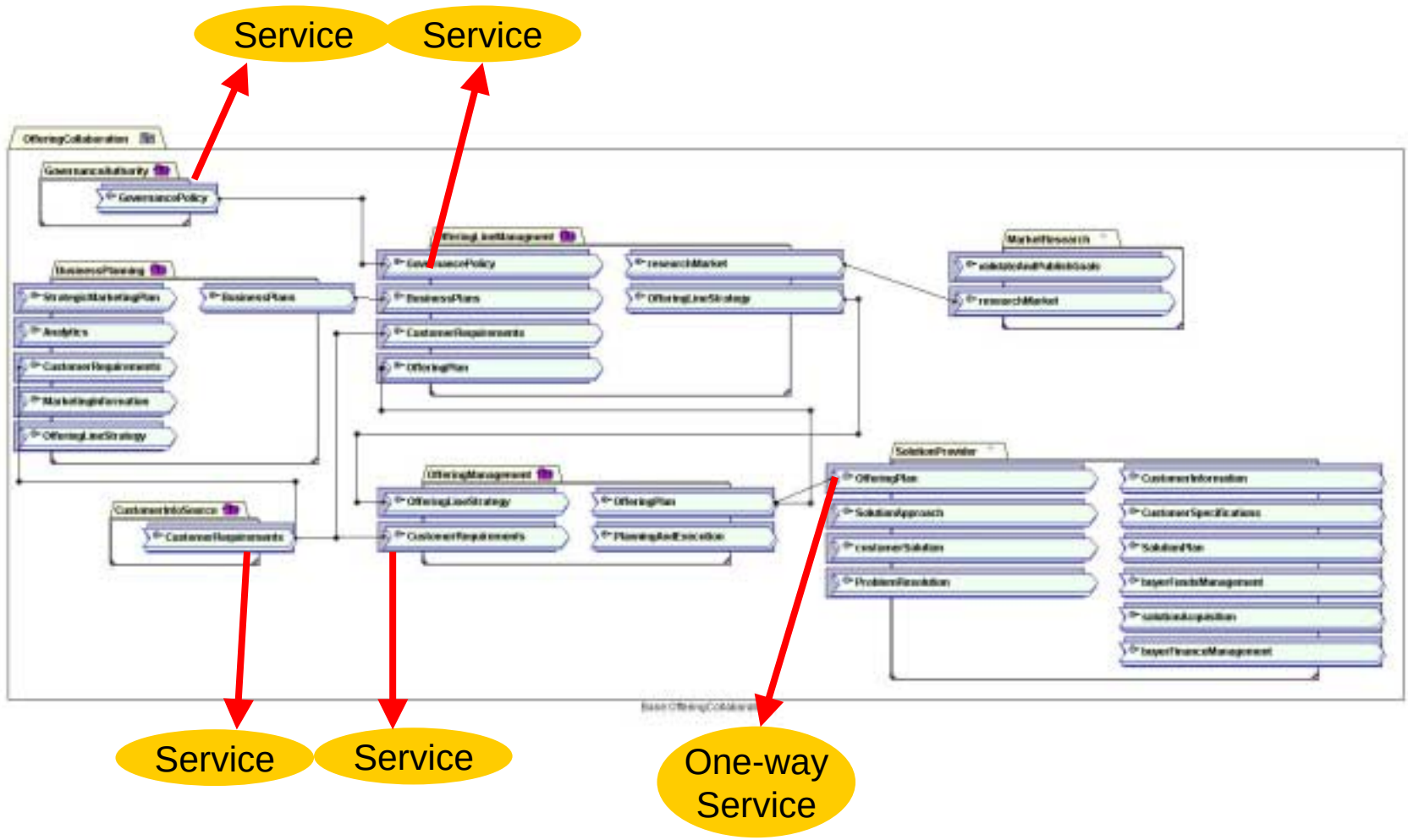
People, Components & Organizations Collaborating



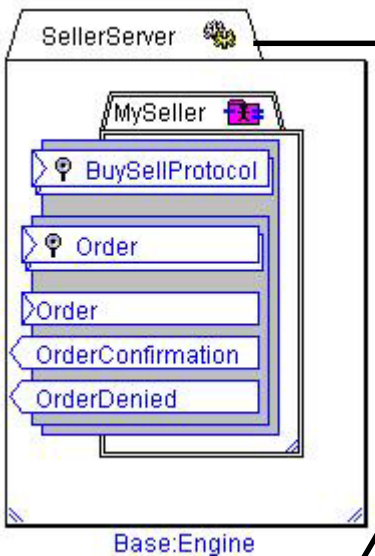
“Lower” PIM View - Enterprise Component Internals



PSM View - Mapping to [web] Services



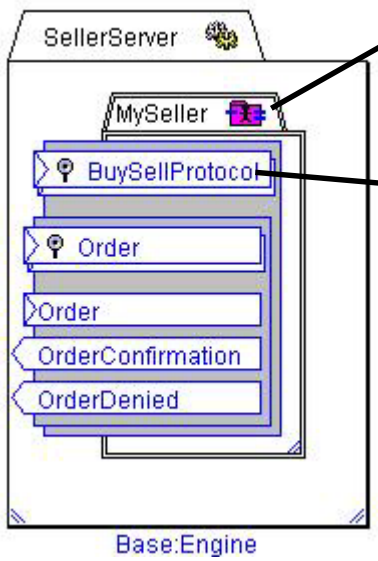
Mapping of a WSDL Engine



- <definitions xmlns="http://schemas.xmlsoap.org/ws
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soa
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mi
xmlns:http="http://schemas.xmlsoap.org/wsdl/http
ENC="http://schemas.xmlsoap.org/soap/encoding/
xmlns:xs2000="http://www.w3.org/1999/XMLSchem
xmlns:xs2001="http://www.w3.org/2001/XMLSchem
targetNamespace="urn:SellerServer" xmlns:tns="urn:S
xmlns:CoreTypes="urn:CoreTypes" xmlns:Ordering="ur
- <!--
definitions obtained from component /BuySell/Deployment/Selle

Aspects
WSDL
WSDL-SOAP

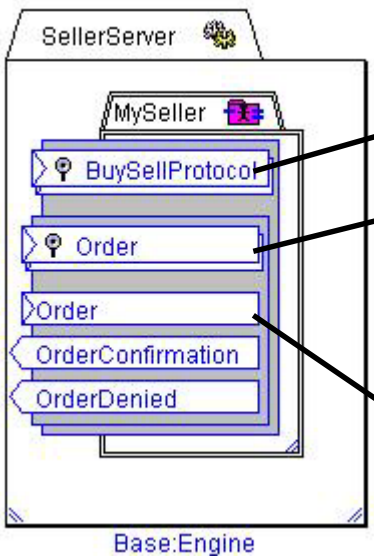
Mapping of an Enterprise Component



Aspects
WSDL
WSDL-SOAP

```
- <service name="MySeller">
- <!--
    implemented service role
/BuySell/Deployment/SellerServer/MySeller    -->
    <documentation><p> </p></documentation>
  - <port name="BuySellProtocol"
binding="tns:BuySellProtocol">
  - <!--
    original service port was
/BuySell/Deployment/SellerServer/MySeller/BuySellProtoc
ol (extending Component
&lt;/BuySell/SellerImplementation/MySeller/BuySellProtoc
ol&gt; )    -->
    <soap:address
location="http://localhost:8080/cx/app/BuyS
ell/Deployment/SellerServer/MySeller/BuyS
ellProtocol" />
  </port>
</service>
```

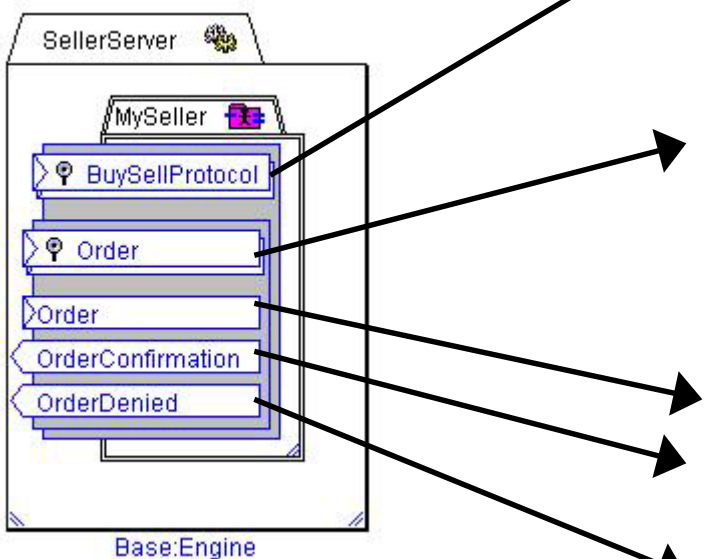
Mapping of a protocol binding



Aspects
WSDL
WSDL-SOAP

```
<binding name="BuySellProtocol"
 type="tns:BuySellProtocol">
<soap:binding
      transport="http://schemas.xmlsoap.org/soap/h
      style="rpc" />
<operation name="Order">
<soap:operation
      soapAction="urn:/BuySell/Community/BuySellP
      col/Order" style="rpc" />
<input name="Order">
<soap:body use="encoded" namespace="urn:SellerServ
      encodingStyle="http://schemas.xmlsoap.org/so
      encoding/" />
```

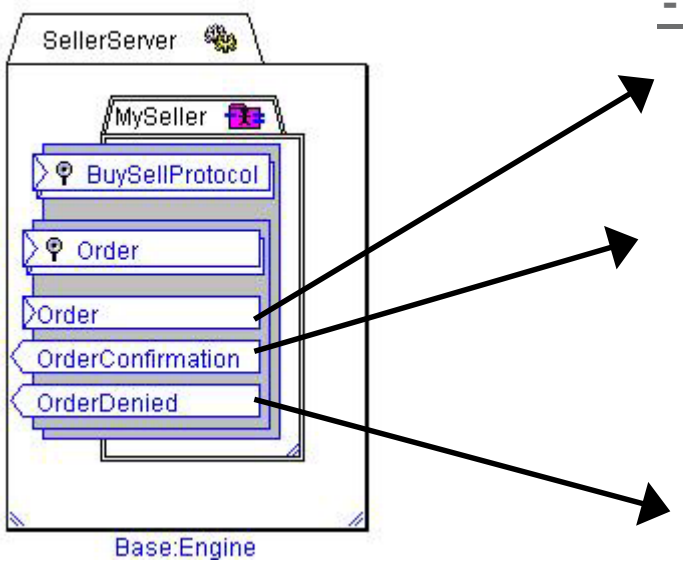
Mapping of a protocol




```
- <portType name="BuySellProtocol">
- <!--
  original cx operation =
/BuySell/Community/BuySellProtocol/Order    -->
- <operation name="Order">
- <!--
  original cx flow port =
/BuySell/Community/BuySellProtocol/Order/Order    -->
  <input name="Order" message="tns:Order" />
  <output name="OrderConfirmation"
message="tns:OrderConfirmation" />
  <fault name="OrderDenied"
message="tns:OrderDenied" />
  </operation>
  </portType>
```

Aspects
WSDL
WSDL-SOAP

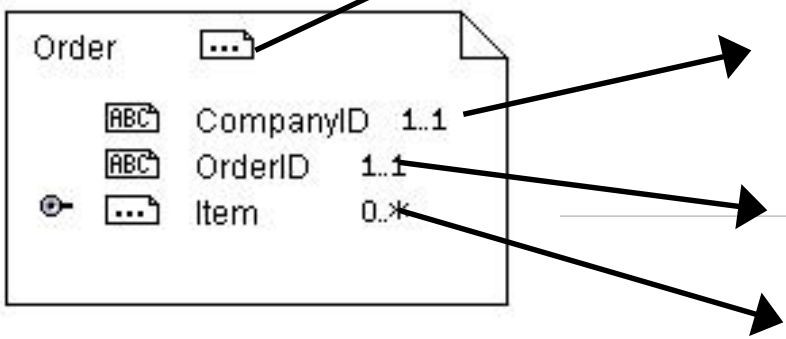
Mapping of message types




```
- <message name="Order">
    <part name="Order" type="Ordering:Order"
<message name="OrderConfirmation">
    <part name="OrderConfirmation"
  type="Ordering:OrderConfirmation" />
    </message>    </message>
- <message name="OrderDenied">
    <part name="OrderDenied"
  type="Ordering:OrderDenied" />
    </message>
```

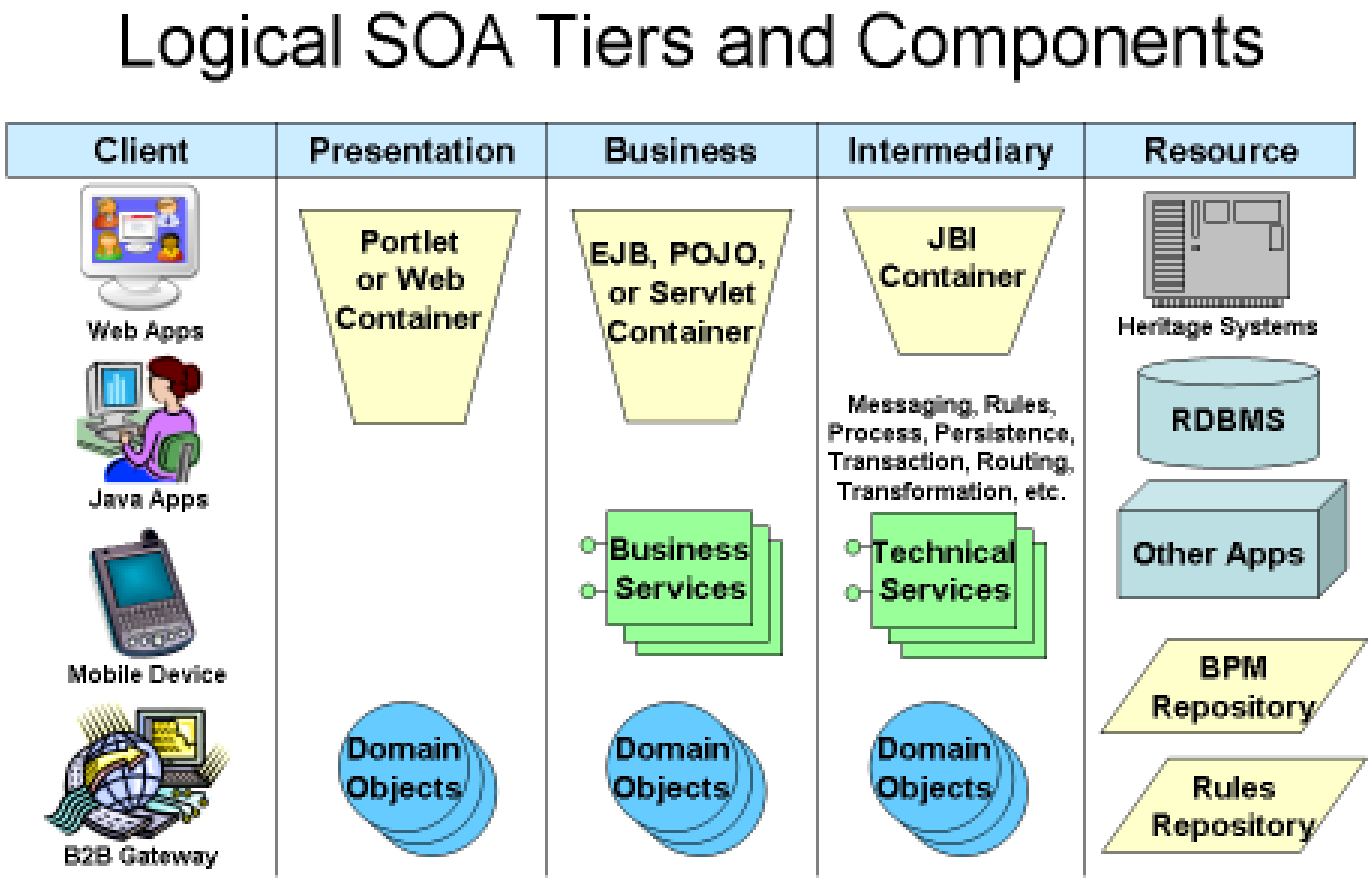
Aspects
WSDL
WSDL-SOAP

Mapping of data types



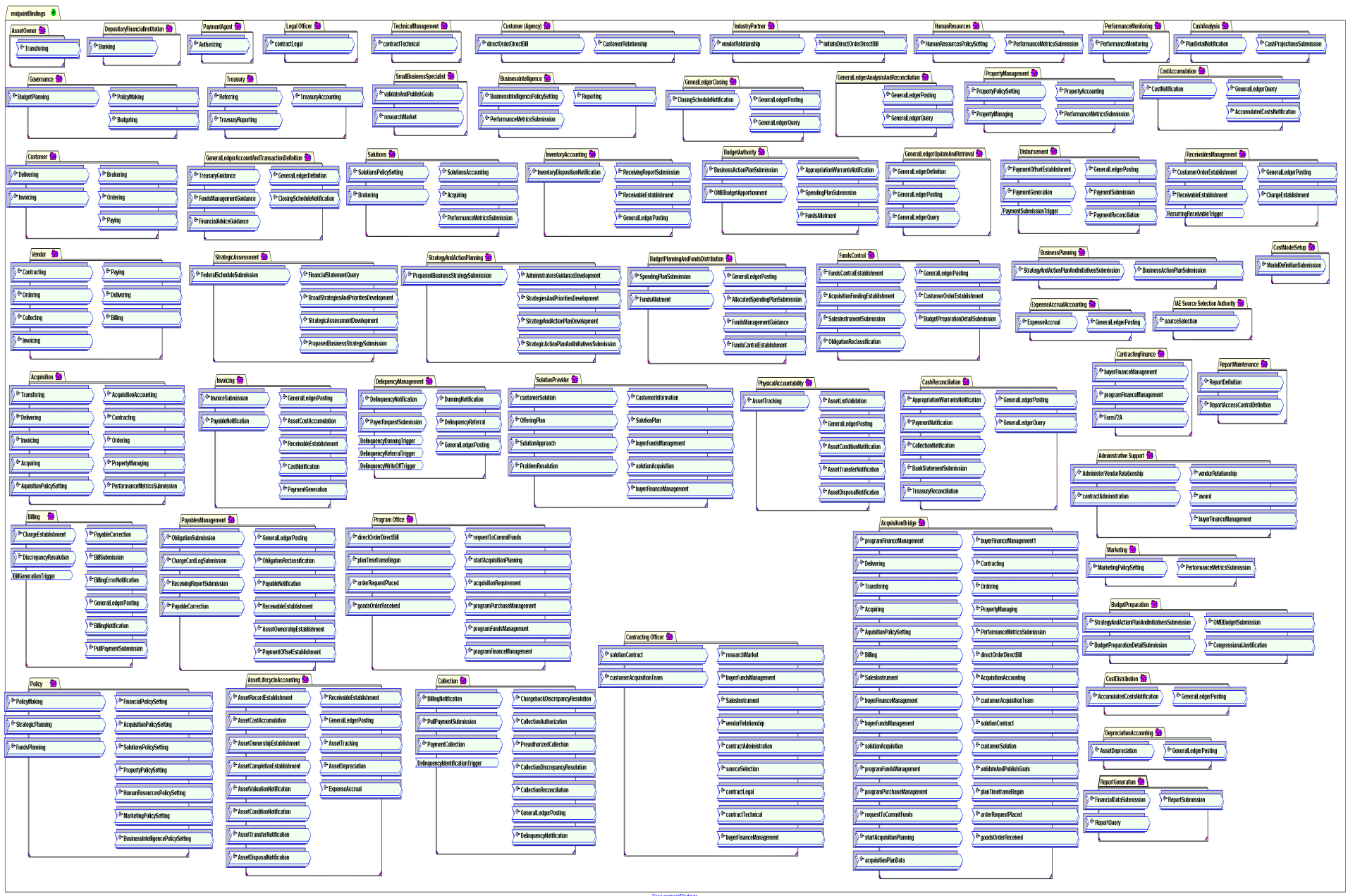

```
- <xs2001:complexType name="Order">
-
- <xs2001:sequence>
    <xs2001:element minOccurs="1"
maxOccurs="1" name="CompanyID"
type="CoreTypes:CompanyID" />
    <xs2001:element minOccurs="1"
maxOccurs="1" name="OrderID"
type="Ordering:OrderID" />
    <xs2001:element minOccurs="0"
maxOccurs="unbounded" name="Item"
type="Ordering:Item" />
    </xs2001:sequence>
    </xs2001:complexType>
```

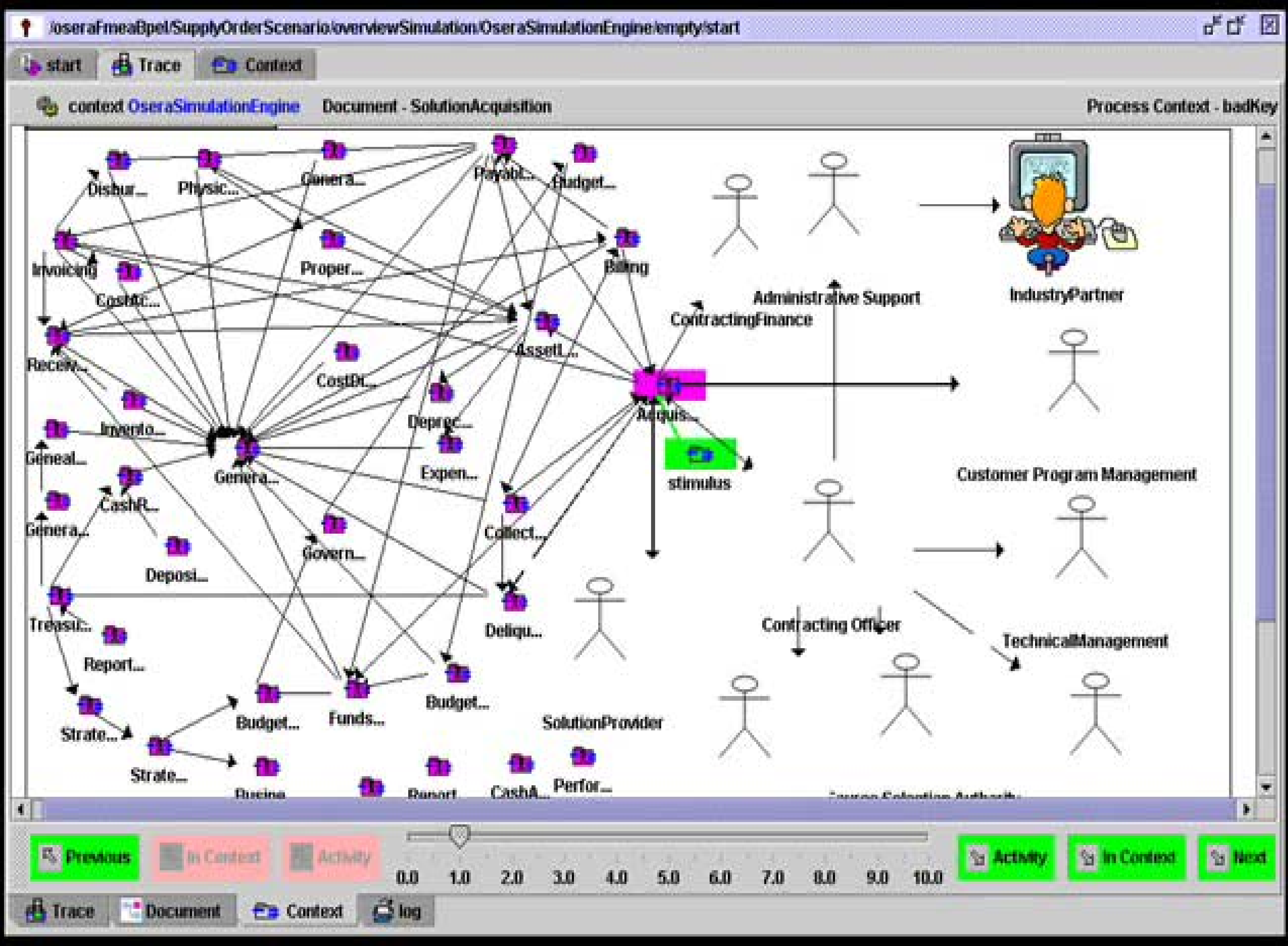
Enterprise Service Bus



* Complements of jBoss

Many BPEL Processes support the CIM




/oseraFmeaBpel/SupplyOrderScenario/overviewSimulation/OseraSimulationEngine/empty/start
start
Trace
Context
context OseraSimulationEngine
Document - SolutionAcquisition
Process Context - badKey
Disbur...
Physic...
Genera...
Payabl...
Budget...
Invoicing
Proper...
Billing
Costbc...
Administrative Support
IndustryPartner
ContractingFinance
AssetI...
Receiv...
CostBr...
Acquis...
Invento...
Deprec...
Geneal...
Genera...
Expen...
stimulus
Customer Program Management
CashR...
Collect...
Genera...
Govern...
Deposi...
Treasu...
Deliqu...
Contracting Officer
TechnicalManagement
Report...
Budget...
Funds...
Budget...
SolutionProvider
Strate...
Strate...
Perfor...
Previous
In Context
Activity
0.0
1.0
2.0
3.0
4.0
5.0
6.0
7.0
8.0
9.0
10.0
Activity
In Context
Next
Trace
Document
Context
log

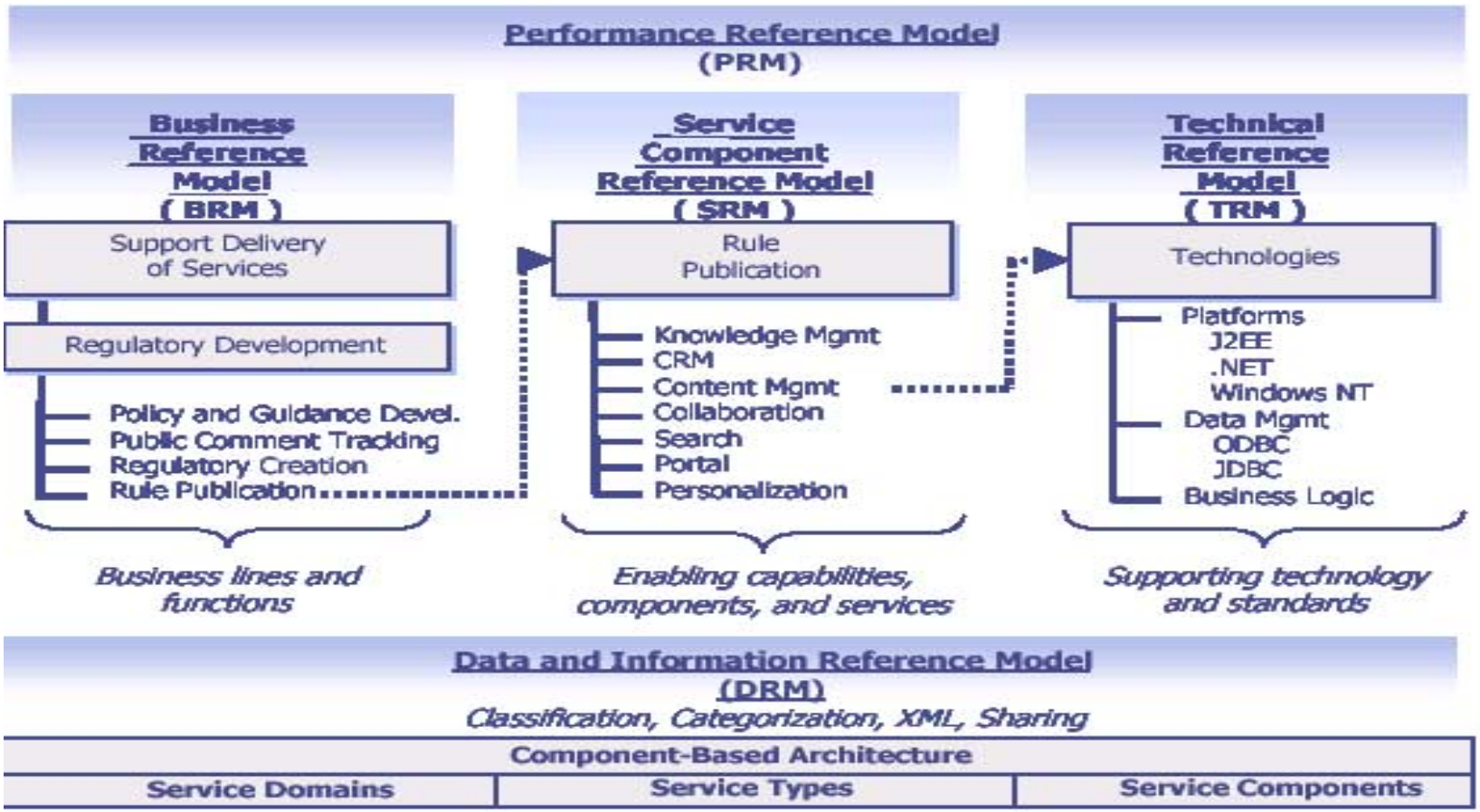
DataAccessTechnologies

Where Business Meets Technology

Federal Enterprise Architecture

Support for the FEA as a view of the enterprise architecture

FEA (from reference)



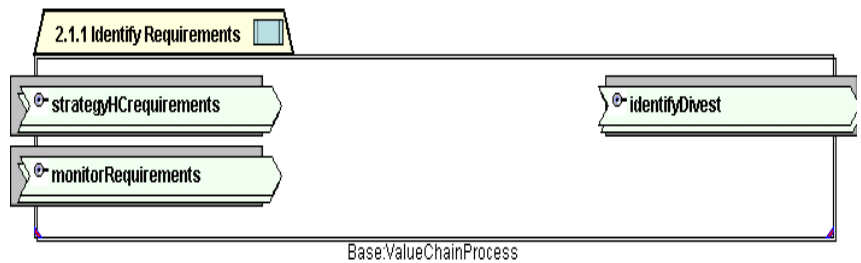
FEA/ComponentX

Business Reference Model (BRM)

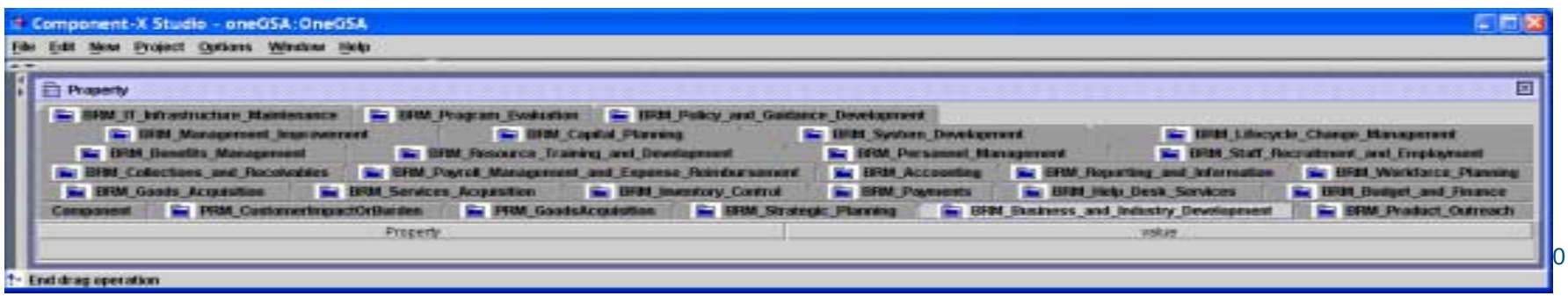
Community Process

Service Component Reference Model (SRM)

Roles, processes, activities



Reference model associations via aspect/properties



name	baseline	planned	actual	achieved
Financial Avoidance				
Productivity QuantityPerTime	0	10	2	
Productivity ProductsPerFTE				
Productivity PercentResourcesUsed				
Productivity PercentImprovement				
Productivity PercentElectronic	0	100	0.9	
CycleTime ProductionTime	5	0	2.5	
CycleTime CycleToWaitTimeRatio	1	0	0.27999999999	
CycleTime PlannedVersusActual	.8	1	1.1	
Quality EffectivityRate	.8	1	0.99	
Quality ComplaintsPerCustomer	.1	0	0.01	
Management Involvement				
Management PolicyCoverage				
Management ApplicationsRequired	5	1	0	
Management RequirementComplianceExtent				
Management PolicyComplianceExtent				
Management UnidentifiedRiskEvents				

Net effect

- Using these open standards and automated techniques we can;
  - 80% Reduction in complexity (Conservative)
  - Achieve the strategic advantage of an open and flexible enterprise
  - Produce and/or integrate these systems FASTER and CHEAPER than could be done with legacy techniques
  - Provide a lasting software asset that will outlive the technology of the day

Sample Applications

- One-GSA Executable Enterprise Architecture for the General Services Administration
- Enterprise Component Architecture for U.S. Army PEO-STRI
- Intelligence application for Raytheon & DARPA
- Collaboration Architecture for Kaiser Permanente

Contact

Cory Casanave

Data Access Technologies

www.enterprisecomponent.com

cory-c@EnterpriseComponent.com

DataAccessTechnologies

Where Business Meets Technology

Military Example

Linking Tactical C2 systems with “Live”, “Virtual” and “Constructive” simulations (SIMCI)

Call for Fire - MLRS

Bde - FSE

Bde- Artillery

Bn-FSE

FA Bn - FSE

Call For Fire

Fire Ready

Rounds Complete

Fire Shot

Fire Splash

Mission Fire Rpt

End Of Mission

Co-FIST

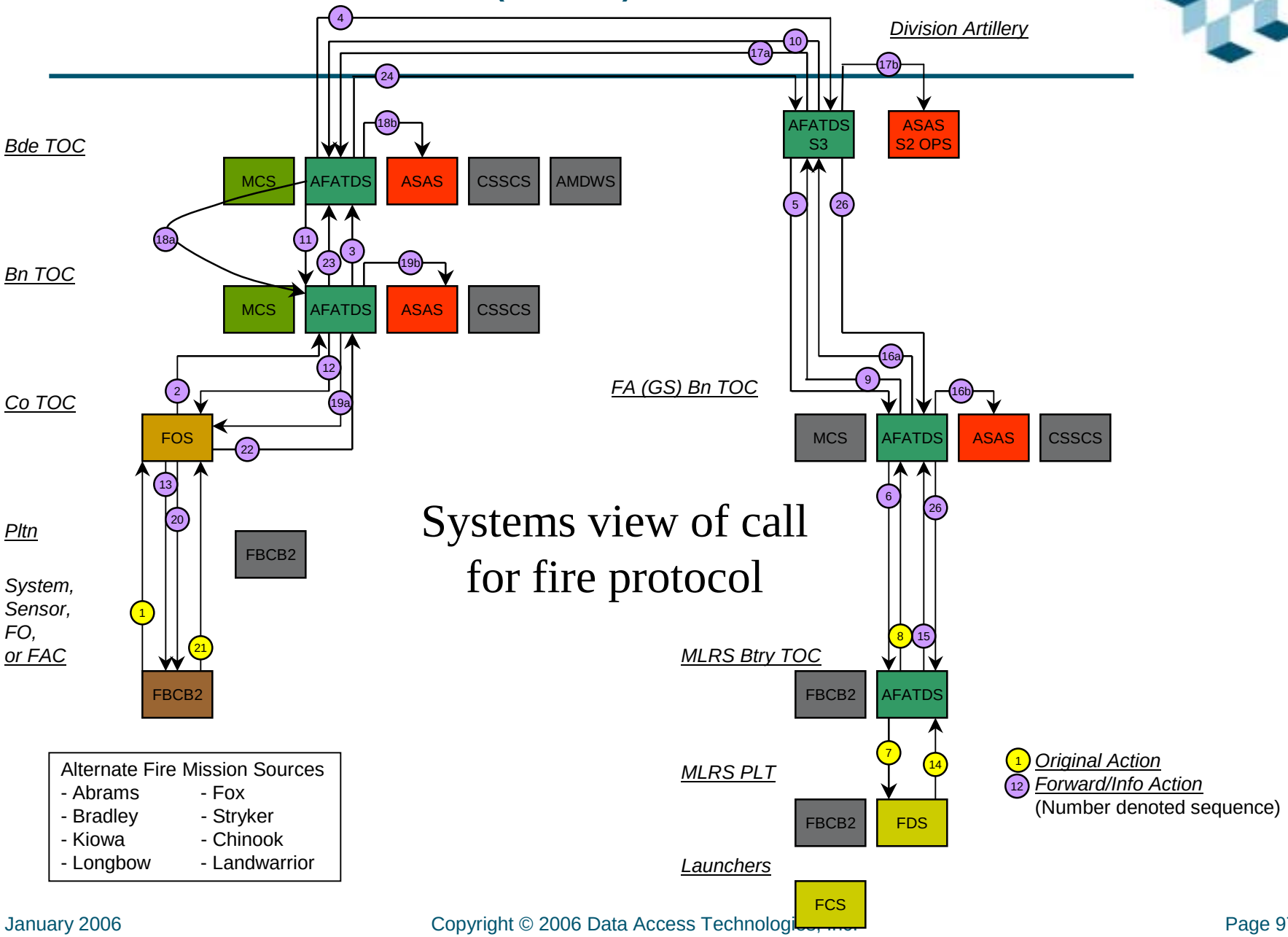
MLRS Battery

Forward Observer

MLRS

Call for fire is a “protocol” of action and information exchange between parties

Thread 2b - FS: CFF To GS MLRS (AFATDS)



Thread Detail (Spread Sheet)

Thread 2b - FS: CFF to GS MLRS (FDS)								
CFF mission (immediate suppression) initiated by System, Sensor, FO or FAC and fired by GS MLRS								
Step	Activity/Task	Originating OPFAC	System	Information	Type	Destination OPFAC	System	Info Action
1	System, Sensor, FO or FAC detects target; analzes situation and makes call for fire request	OBS	FBCB2	Call For Fire (CFF)	K02.4	Co FIST	FOS	A
2	Co FIST receives CFF request from platform, does a quick analysis (checking for dual targeting, etc.), and passes CFF to the Bn FSE	Co FIST	FOS	CFF		Bn TOC	AFATDS	A
3	Bn FSE receives CFF, makes determination that Bn assets cannot handle the mission, and passes the request to the Bde FSE	Bn TOC	AFATDS	CFF	Pkg 10/11 Fmt	Bde TOC	AFATDS	A
4	Bde FSE receives CFF, makes determination that Bde assets cannot handle the mission, and passes the request to the FA Bde (Div Arty) FDC	Bde TOC	AFATDS	CFF	Pkg 10/11 Fmt	FA Bde TOC	AFATDS	A
5	FA Bde FDC receives CFF, makes determination that GS FA assets can handle the mission, and passes the request to the FA (GS) Bn FDC	FA Bde TOC	AFATDS	CFF		FA Bn TOC	AFATDS	A
6	FA Bn FDC receives Fire request/order to fire, assigns mission to MLRS battery	FA Bn TOC	AFATDS	CFF	Pkg 10/11 Fmt	MLRS Btry CP	AFATDS	A
7	Btry FDS receives OTF and transmits this OTF to the MLRS platoon FDS	MLRS Btry CP	ATADS	OTF	BCS Fmt	MLRS Platoon	FDS	A
8	4 different messages transmitted back to the observer and as info to other echelons, depending upon the specific fire mission requested, some may not be necessary; "Ready" states that the battery is preparing to fire the mission, "Shot" stastes that the i	MLRS Btry CP	AFATDS	MTO, Shot, Spalsh, Rounds Complete	Pkg 10/11 Fmt	FA Bn TOC	AFATDS	I

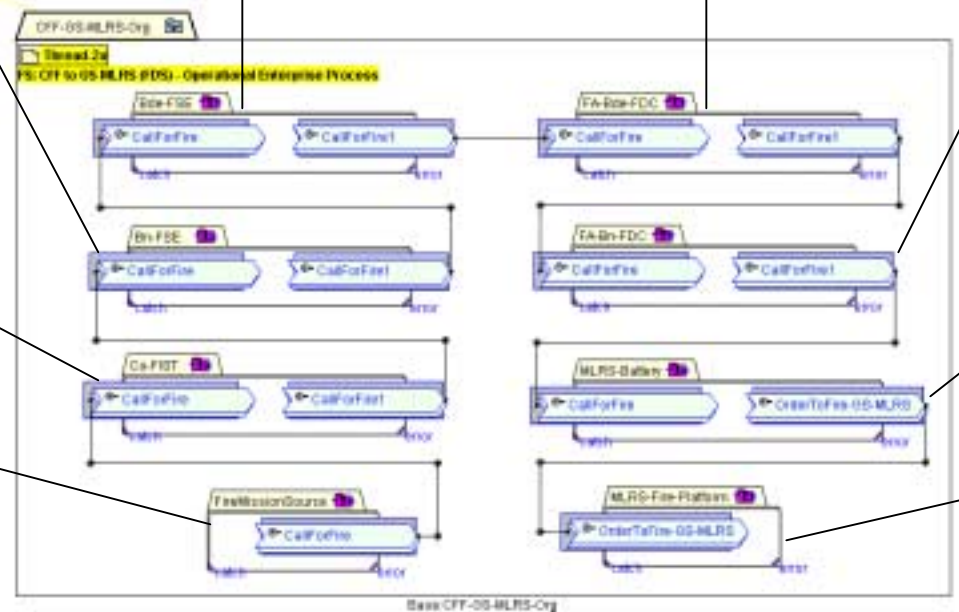
Call for Fire - MLRS

Bde - FSE

Bde- Artillery

Bn-FSE

FA Bn - FSE



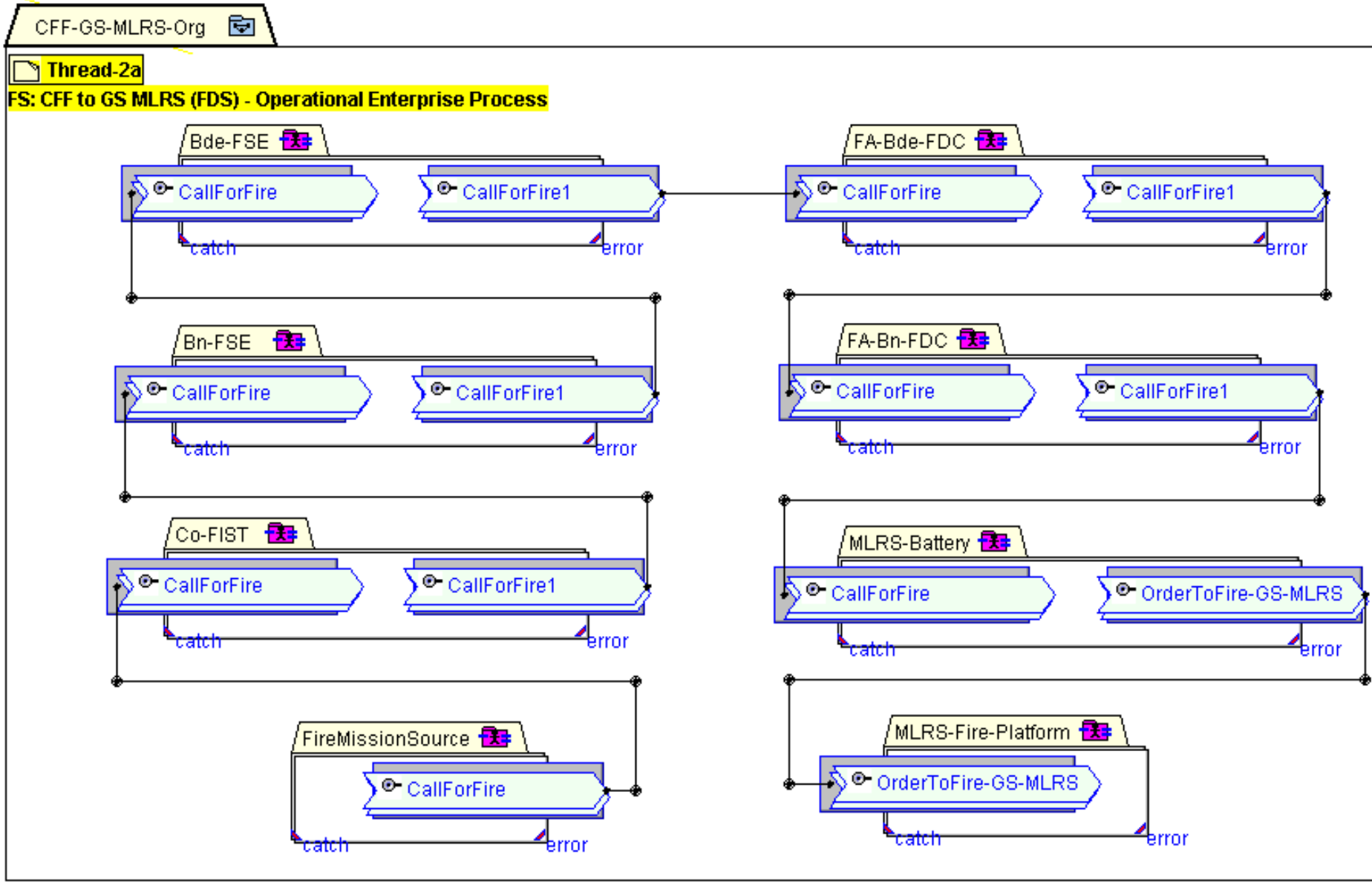
Co-FIST

MLRS Battery

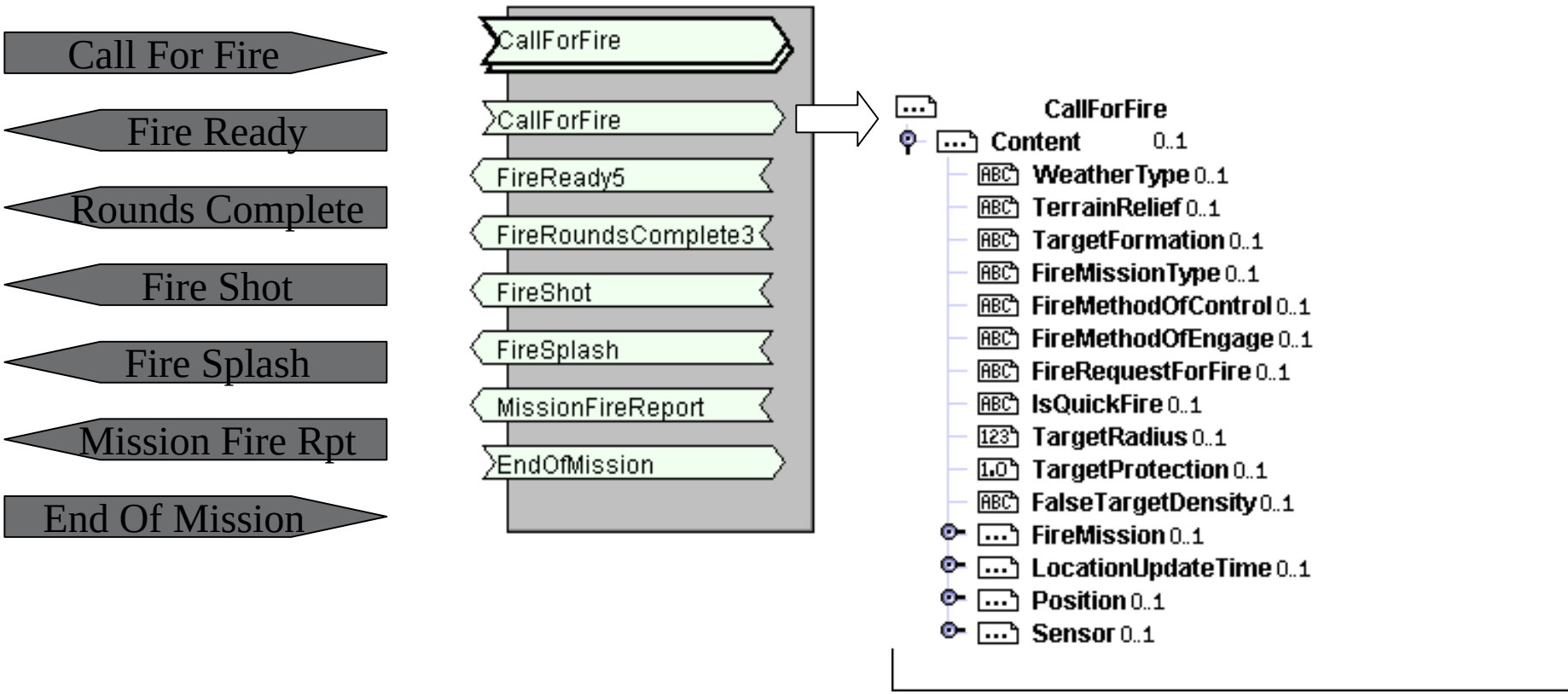
Forward Observer

MLRS

Model Of CFF Thread

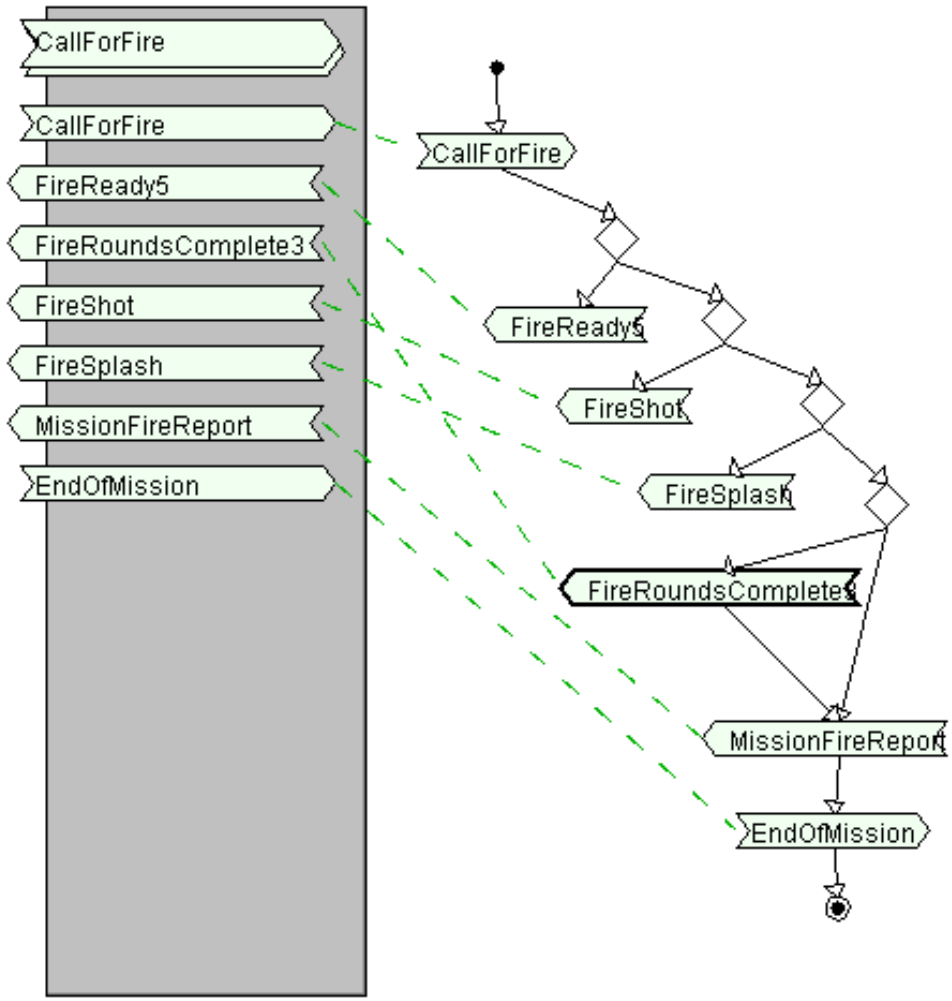


Model Information Flows

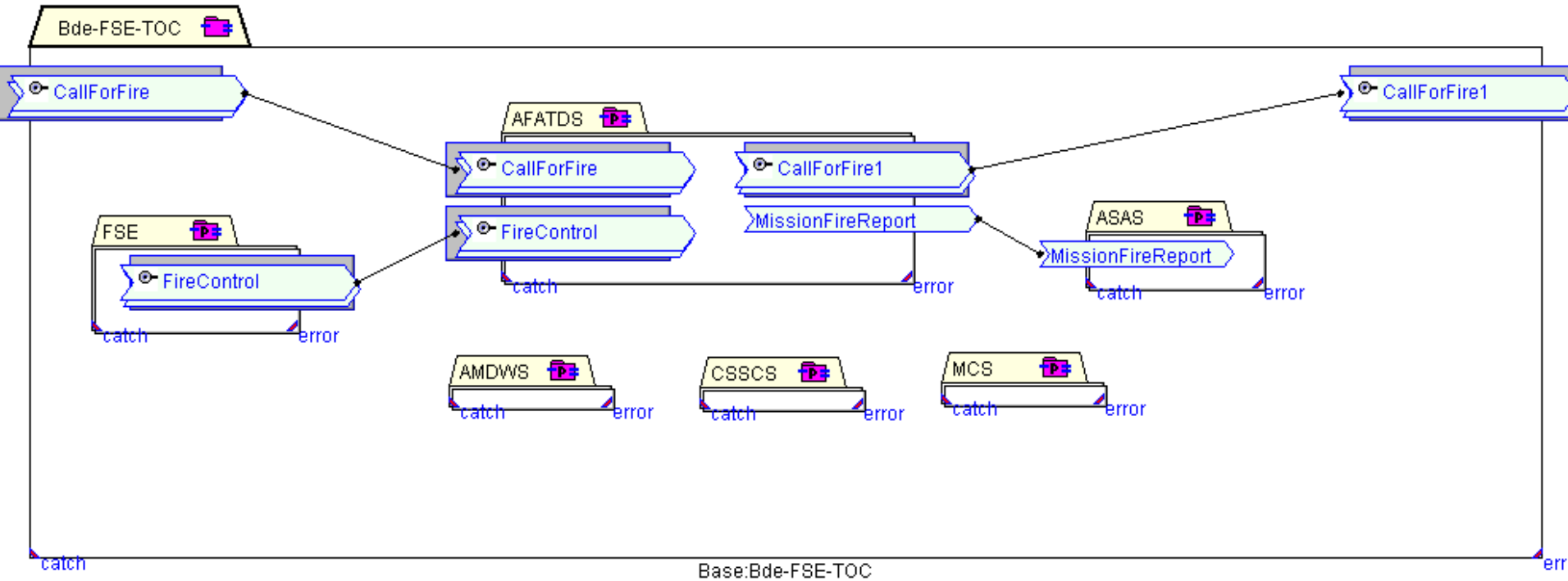


* Not technology details!

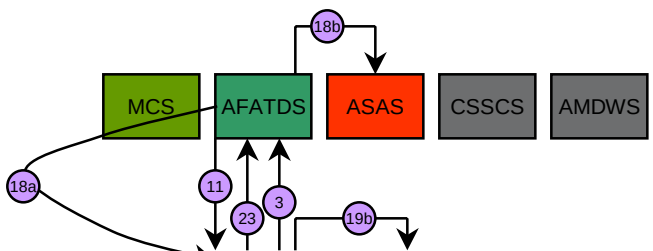
Choreography – Understanding When



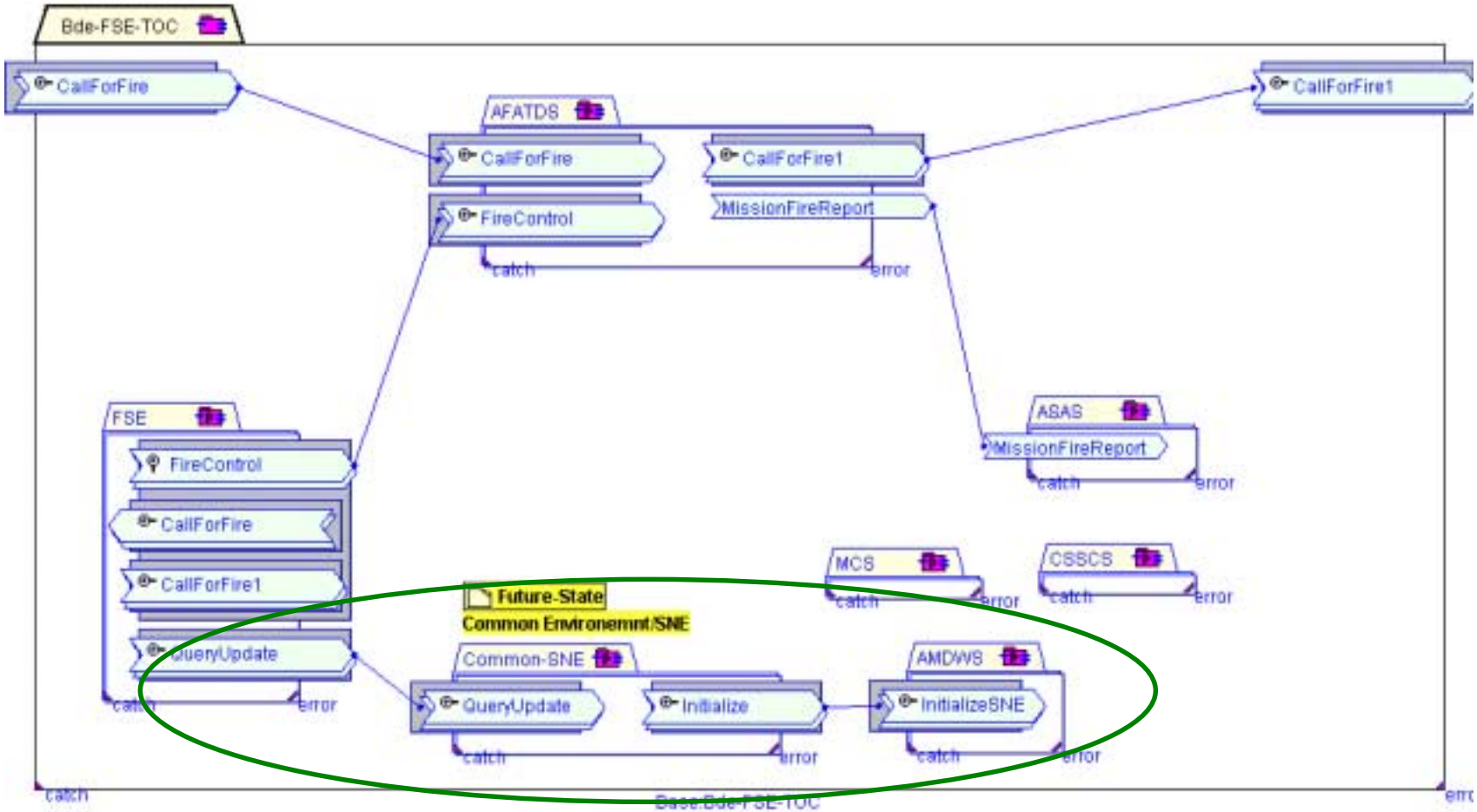
Drill Down - Inside of a TOC



From “threads”

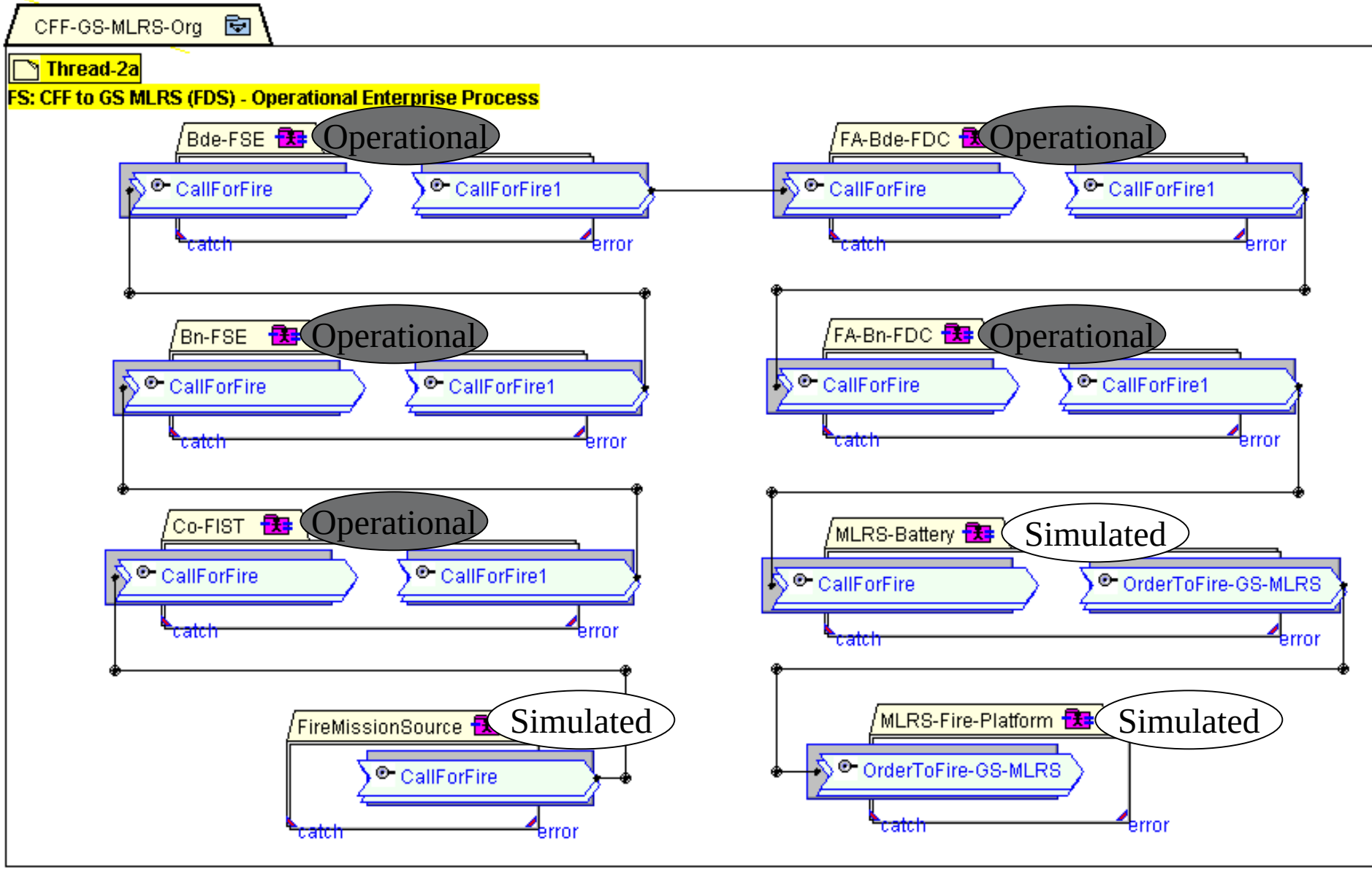


Inside of a TOC

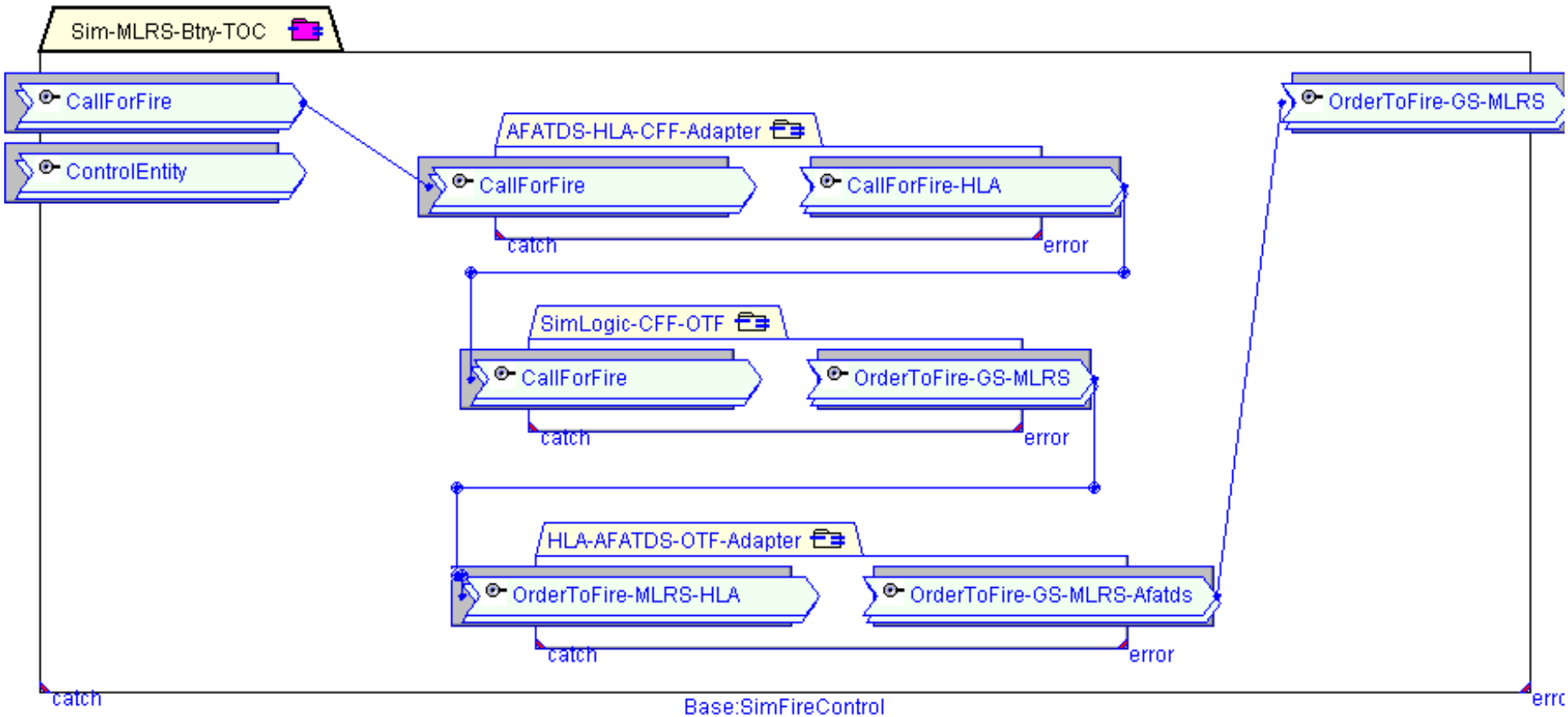


Example Future State

Simulated or Real?



Inside of a Simulated "TOC"

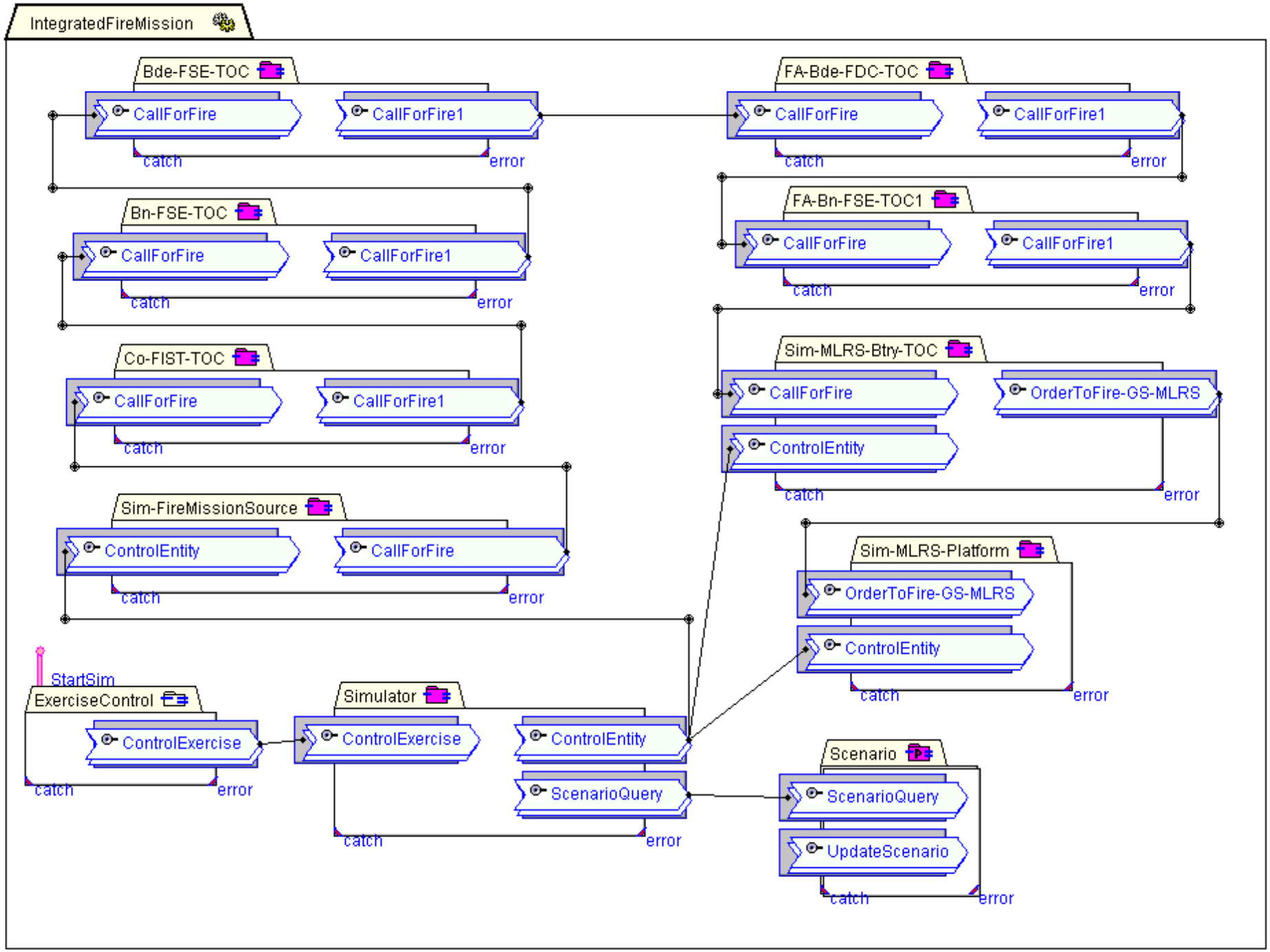


DataAccessTechnologies

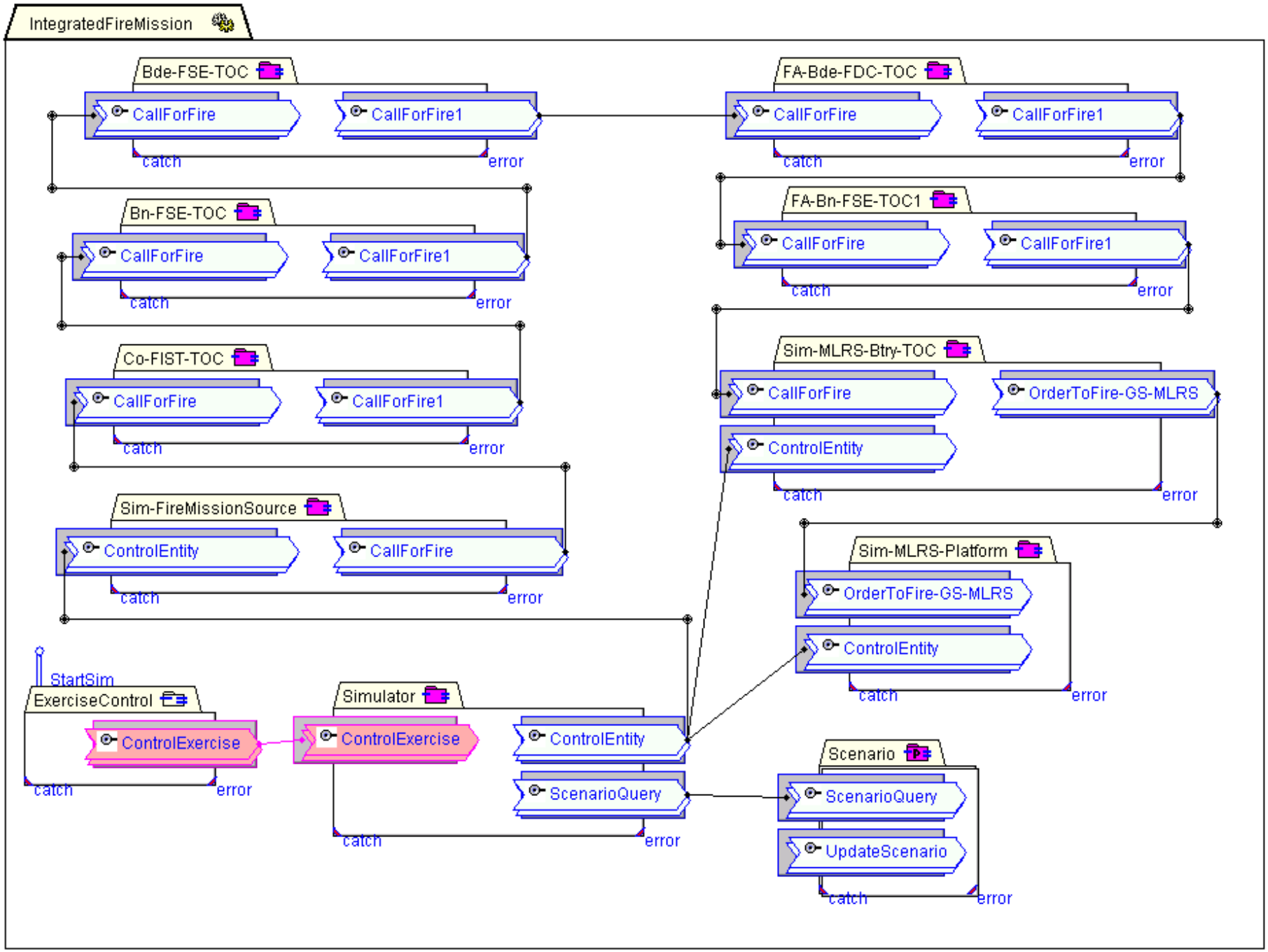
Where Business Meets Technology

Simulating the Process

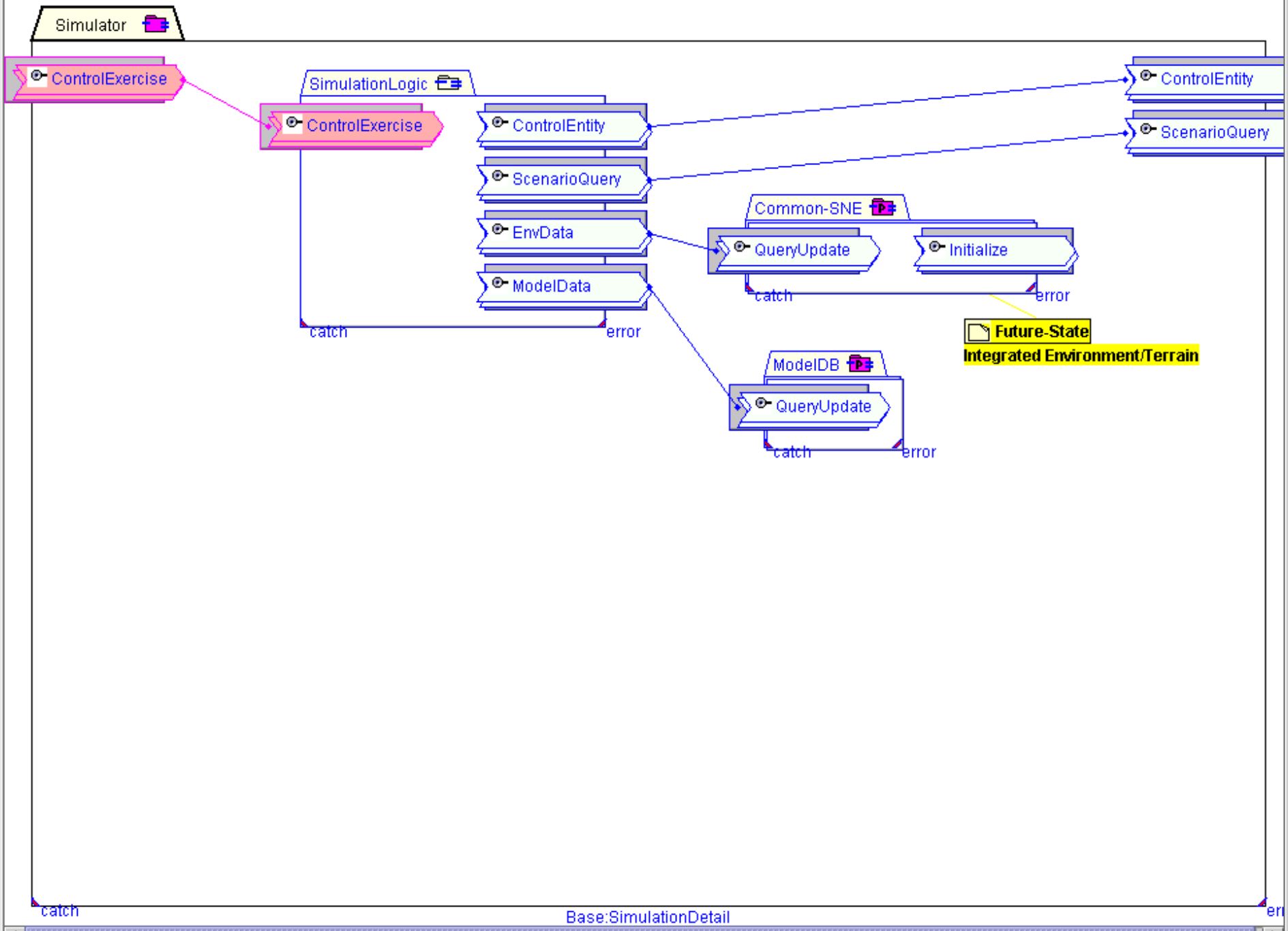
Tactical and Simulated components interacting

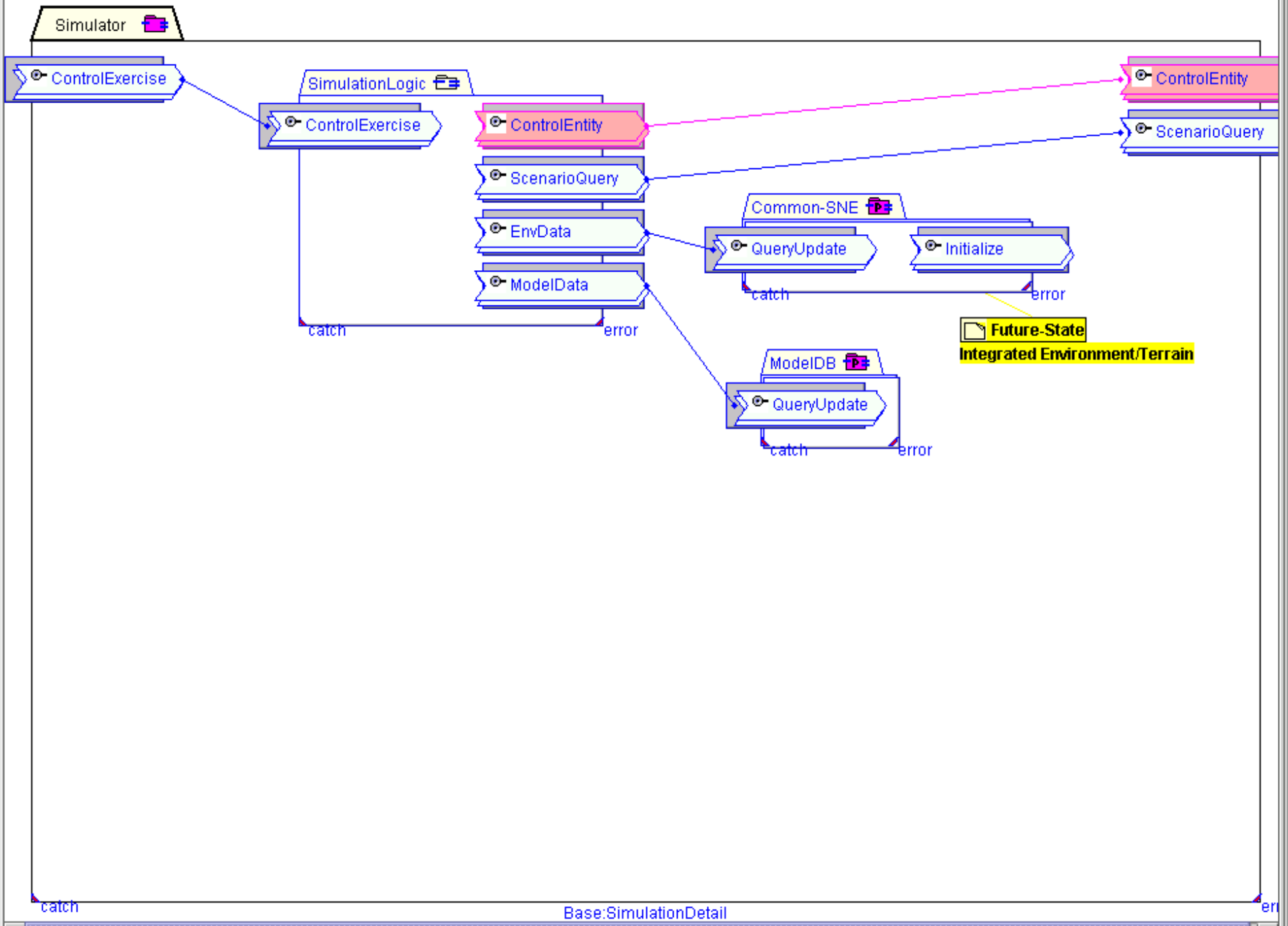


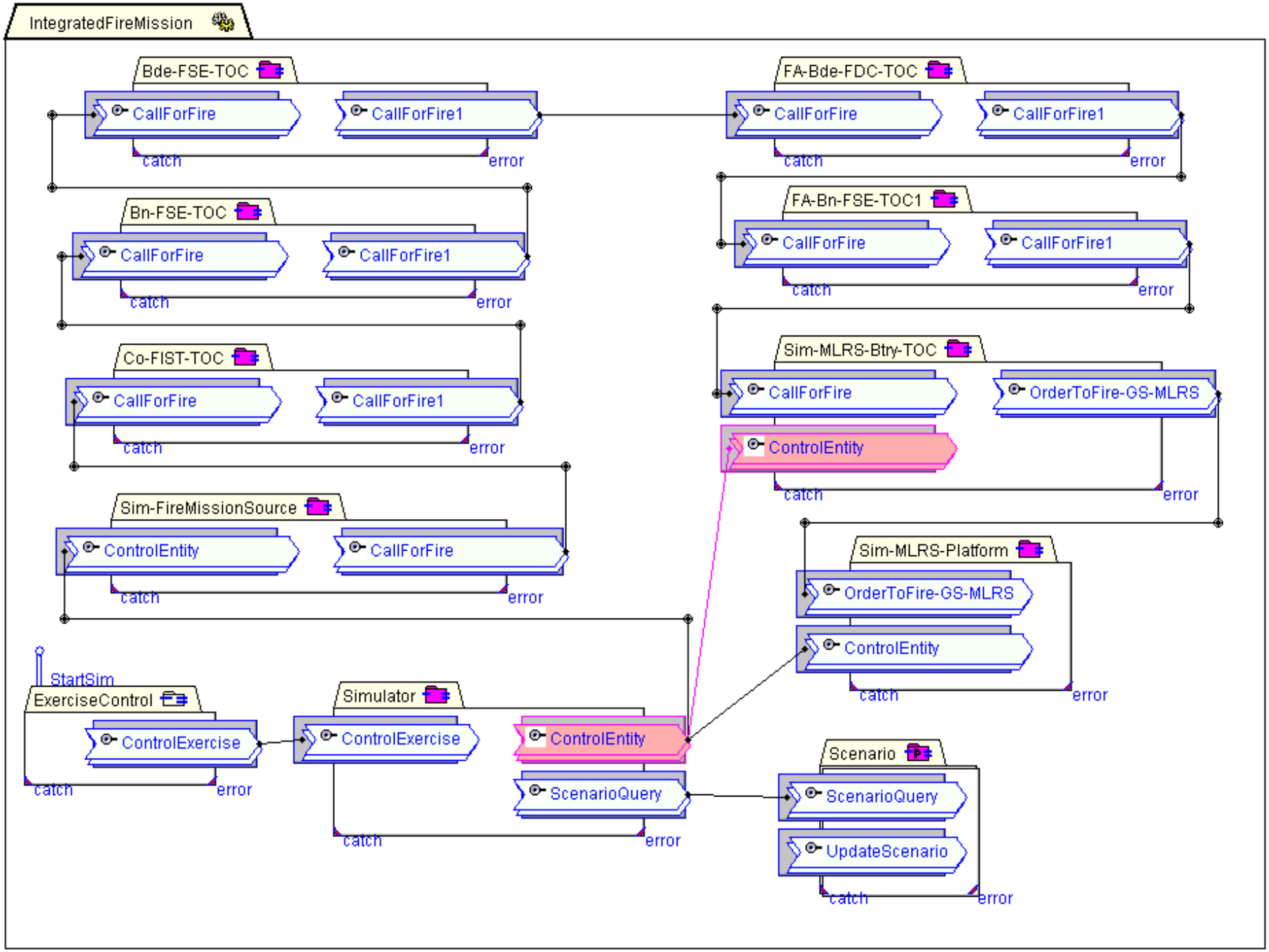
Base:IntegratedFireMission



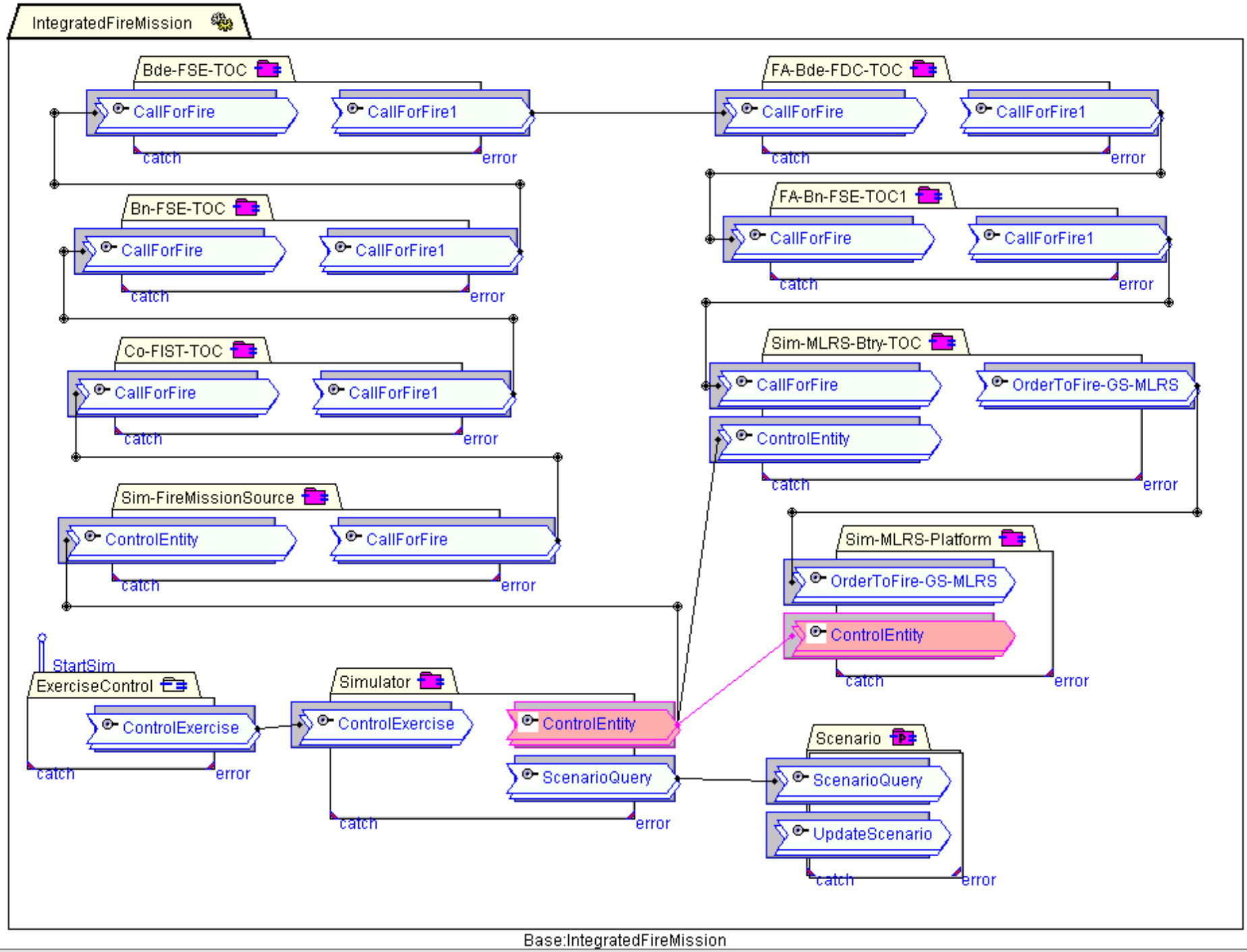
Base:IntegratedFireMission


Simulator
ControlExercise
SimulationLogic
ControlExercise
ControlEntity
ScenarioQuery
EnvData
ModelData
catch
error
ControlEntity
ScenarioQuery
Common-SNE
QueryUpdate
Initialize
catch
error
Future-State
Integrated Environment/Terrain
ModelDB
QueryUpdate
catch
error
catch
Base:SimulationDetail

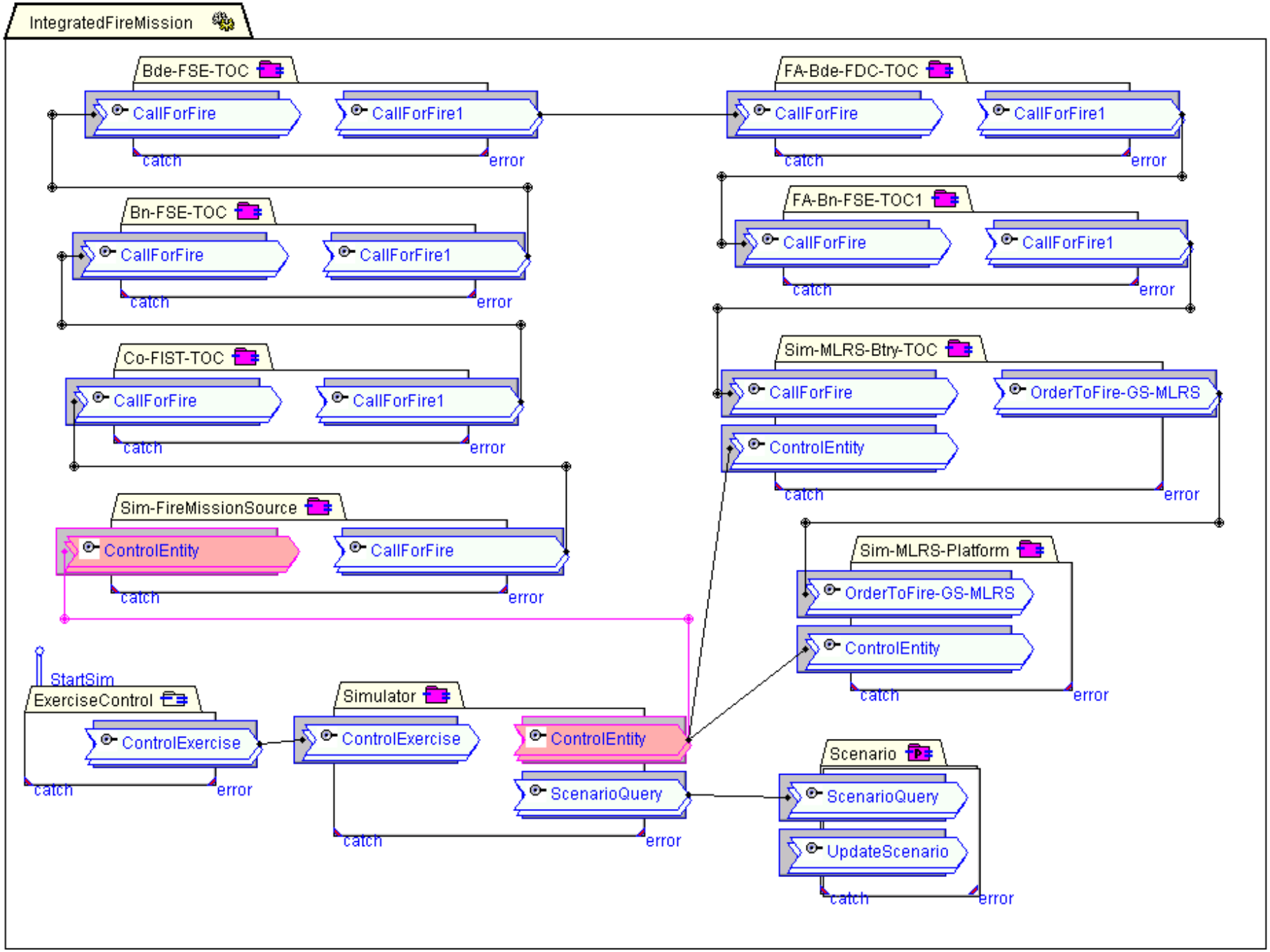

Simulator
ControlExercise
SimulationLogic
ControlExercise
ControlEntity
ScenarioQuery
EnvData
ModelData
catch
error
ControlEntity
ScenarioQuery
Common-SNE
QueryUpdate
Initialize
catch
error
Future-State
Integrated Environment/Terrain
ModelDB
QueryUpdate
catch
error
catch
Base:SimulationDetail



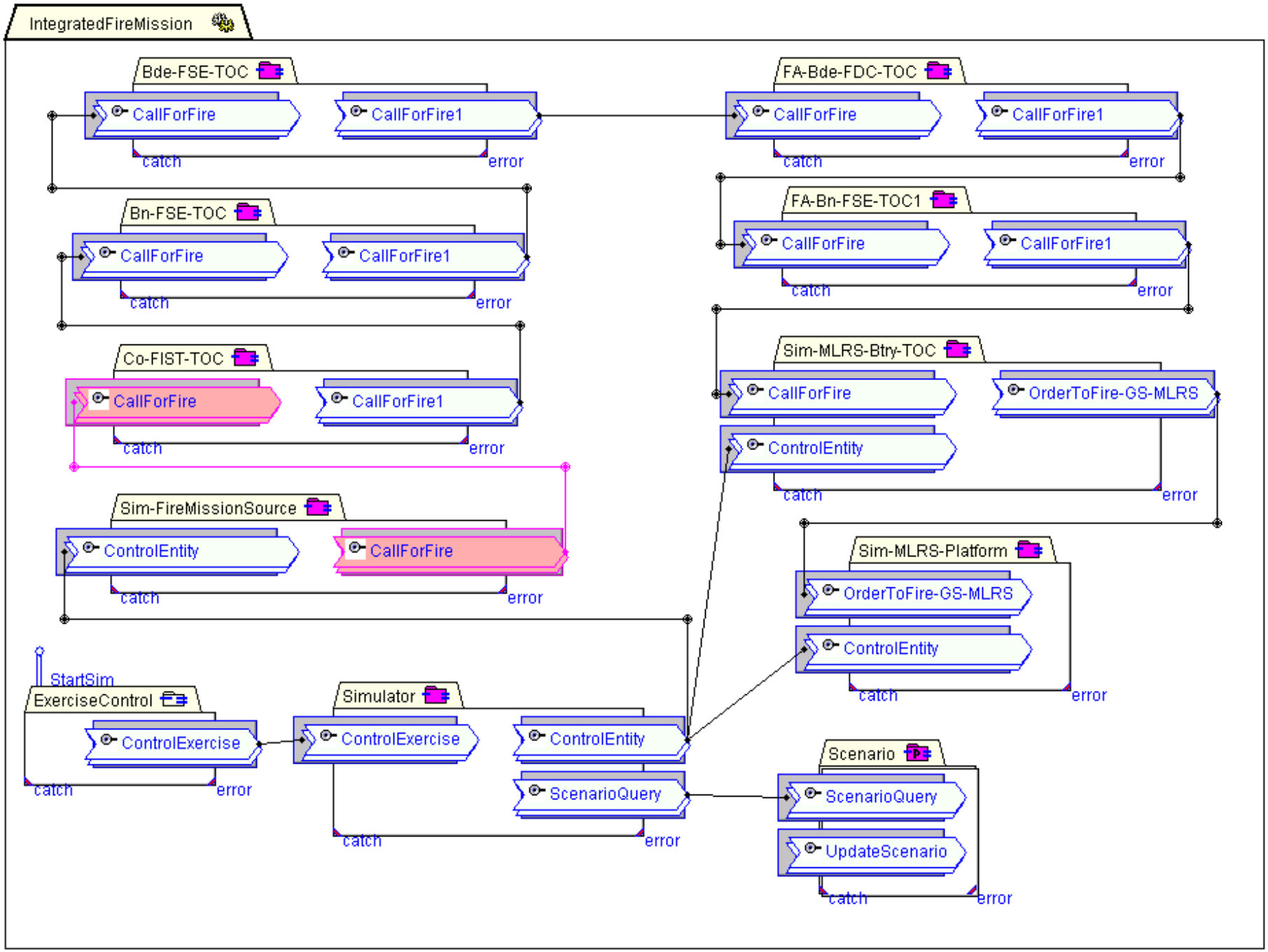
Base:IntegratedFireMission



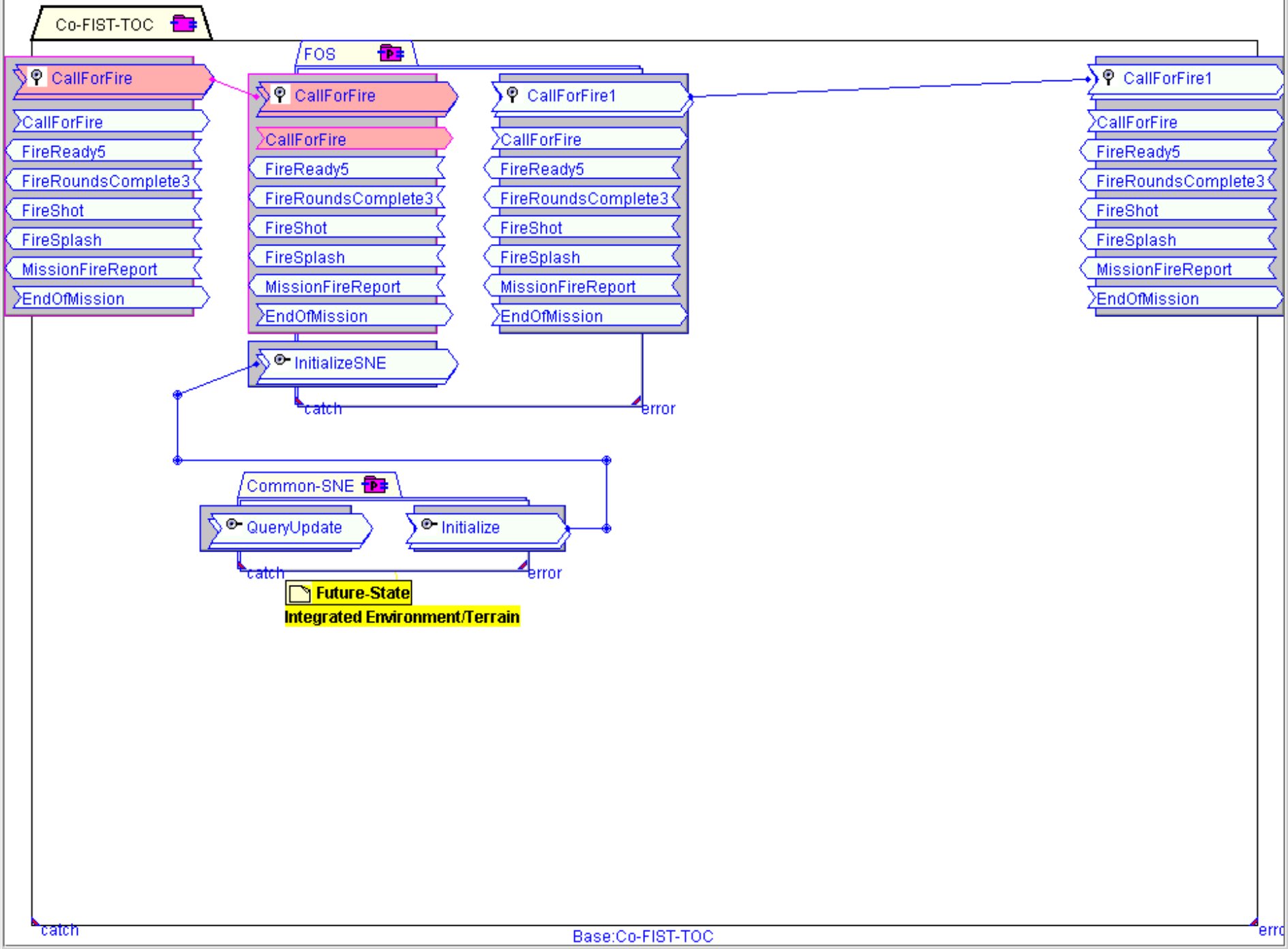
Base:IntegratedFireMission

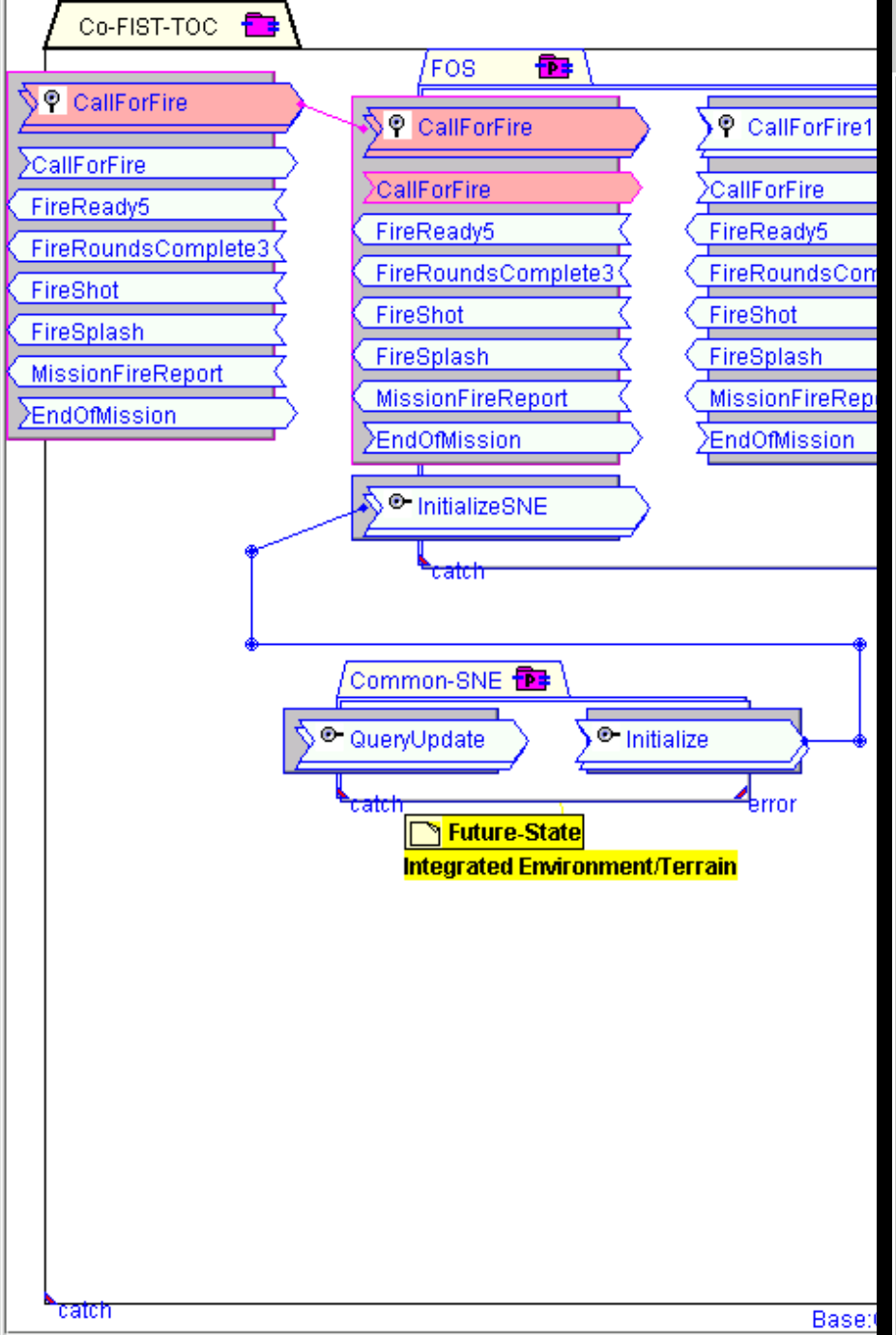


Base:IntegratedFireMission



Base:IntegratedFireMission


Co-FIST-TOC
FOS
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
InitializeSNE
catch
error
Common-SNE
QueryUpdate
Initialize
catch
error
Future-State
Integrated Environment/Terrain
catch
Base:Co-FIST-TOC

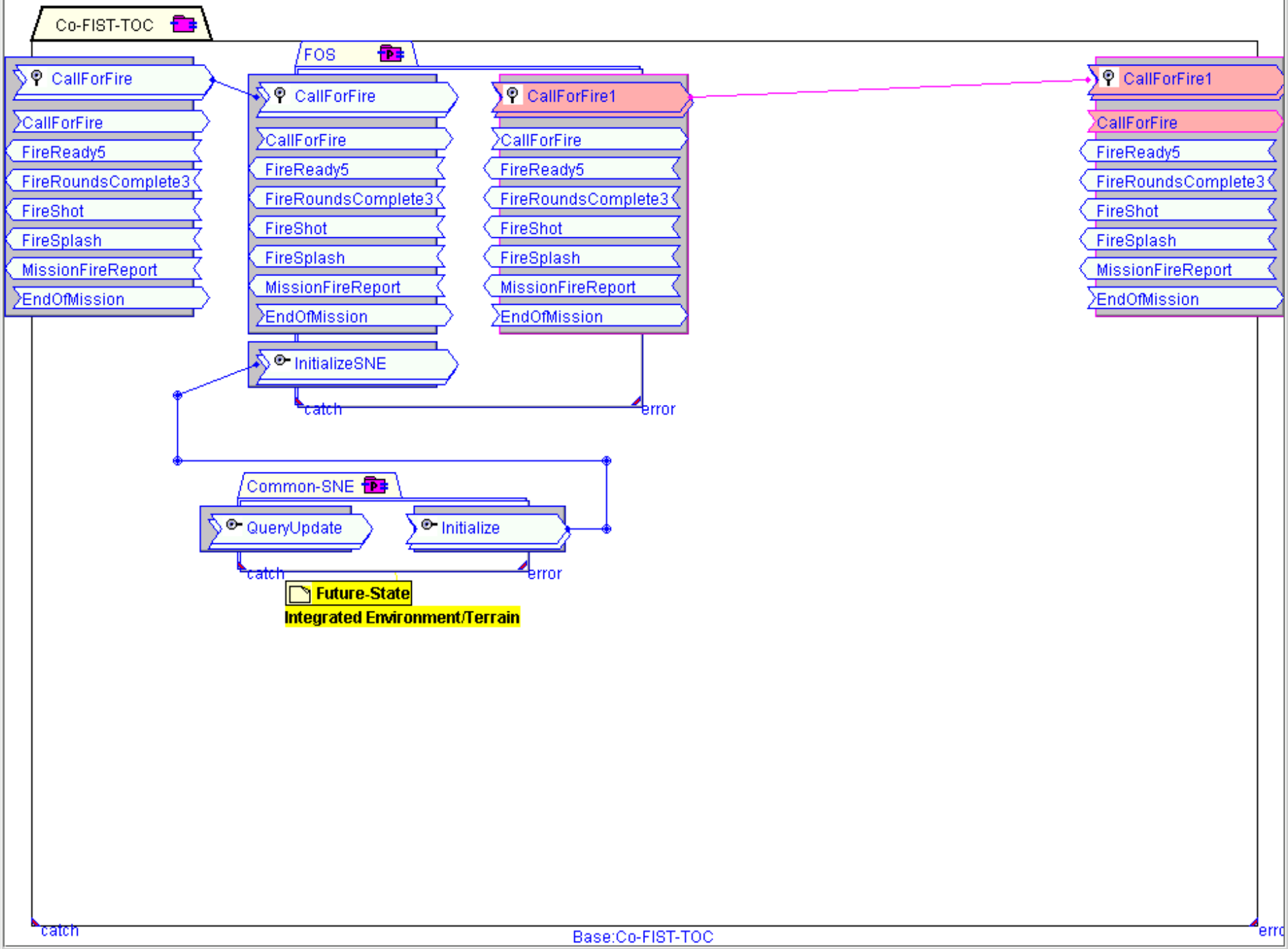


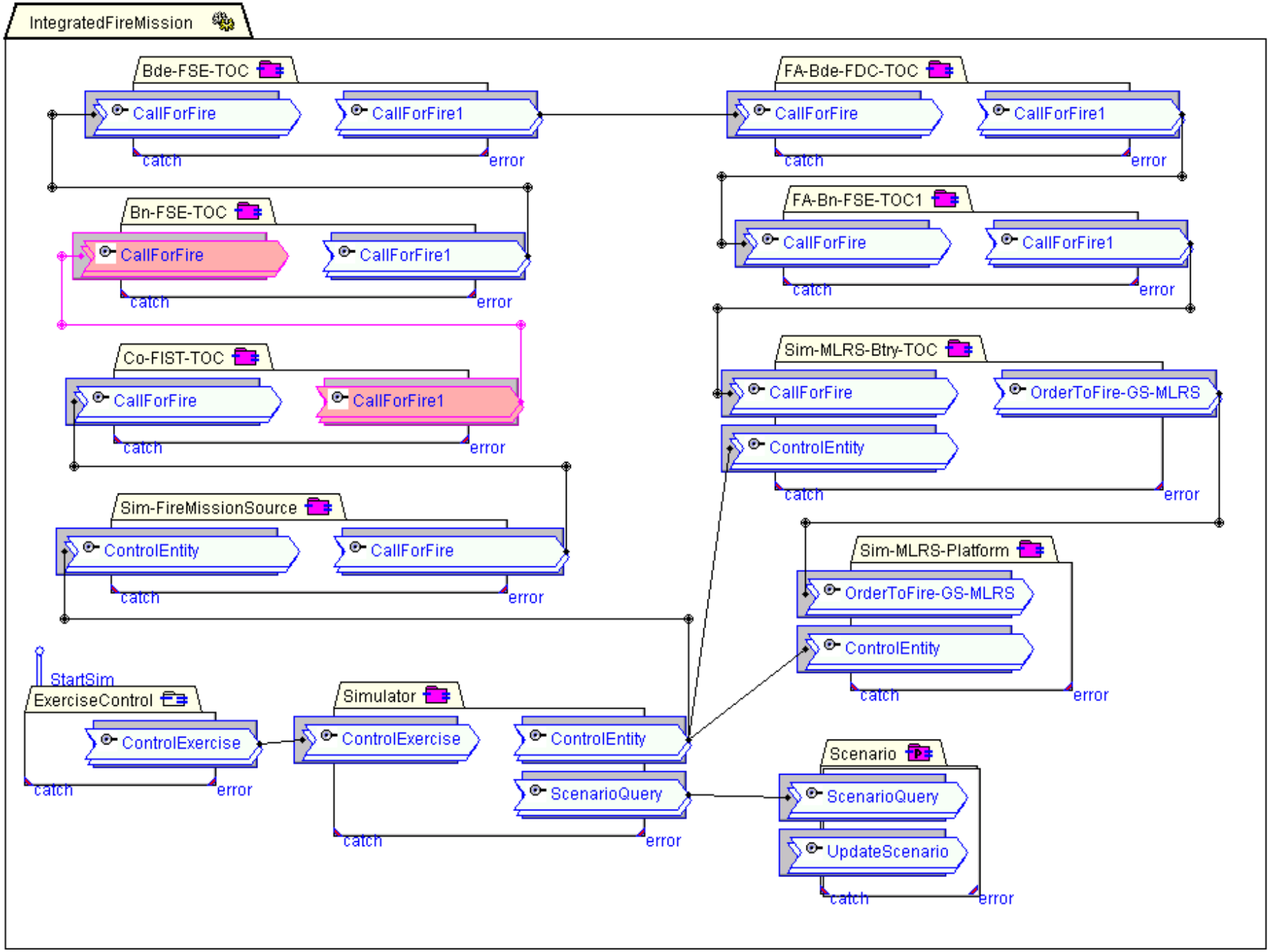
StartSim | Trace | Context

XML Event - CallForFire

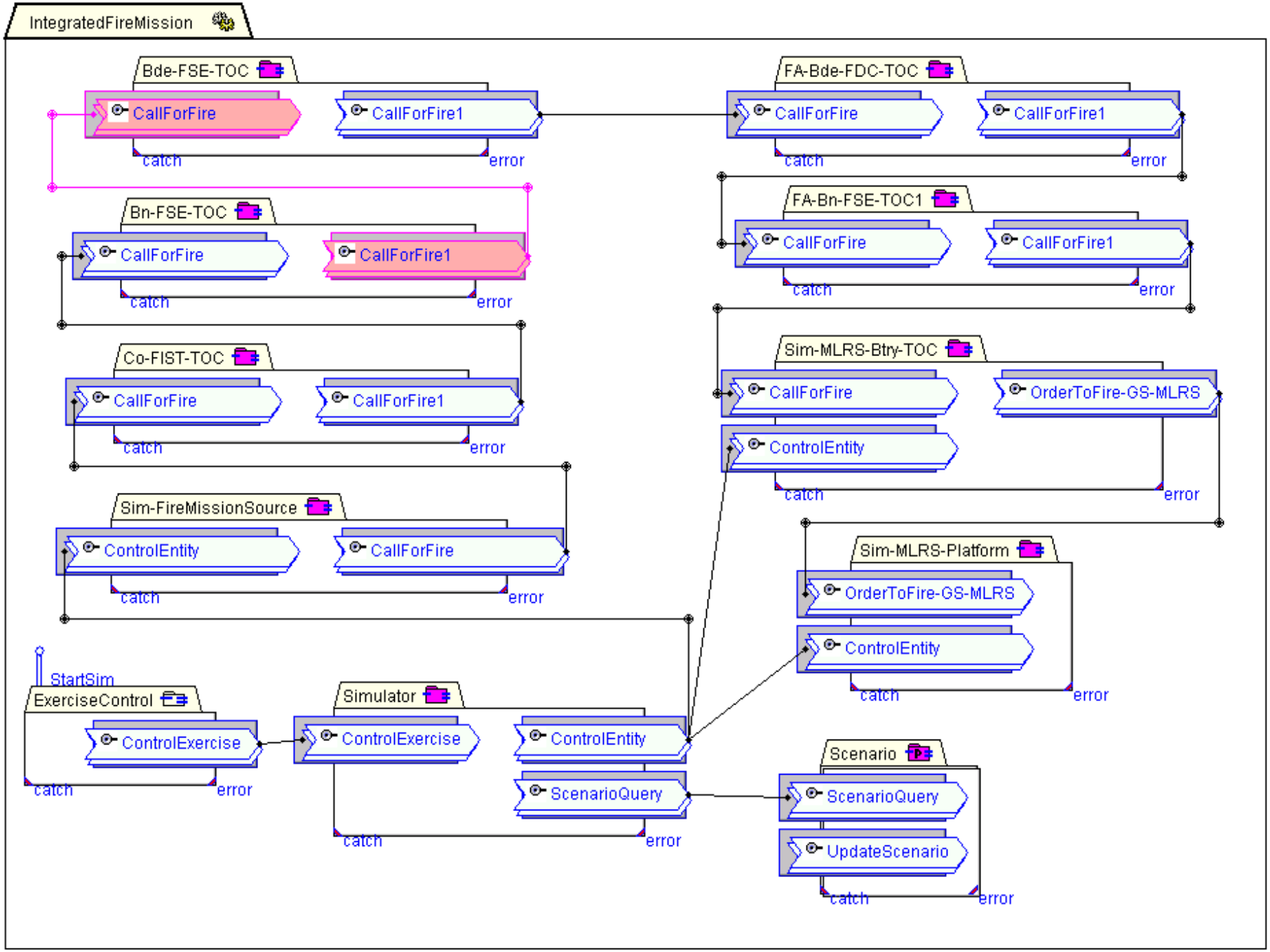
text | form

```
<CallForFire>
 <Content>
  <WeatherType/>
  <TerrainRelief/>
  <TargetFormation/>
  <FireMissionType/>
  <FireMethodOfControl/>
  <FireMethodOfEngage/>
  <FireRequestForFire/>
  <IsQuickFire/>
  <TargetRadius>0</TargetRadius>
  <TargetProtection>0</TargetProtection>
  <FalseTargetDensity/>
  <FireMission>
   <ProjectileType/>
   <FiringUnits/>
   <NumVolleys/>
   <NumRounds/>
  </FireMission>
  <LocationUpdateTime>
   <IsValid/>
```

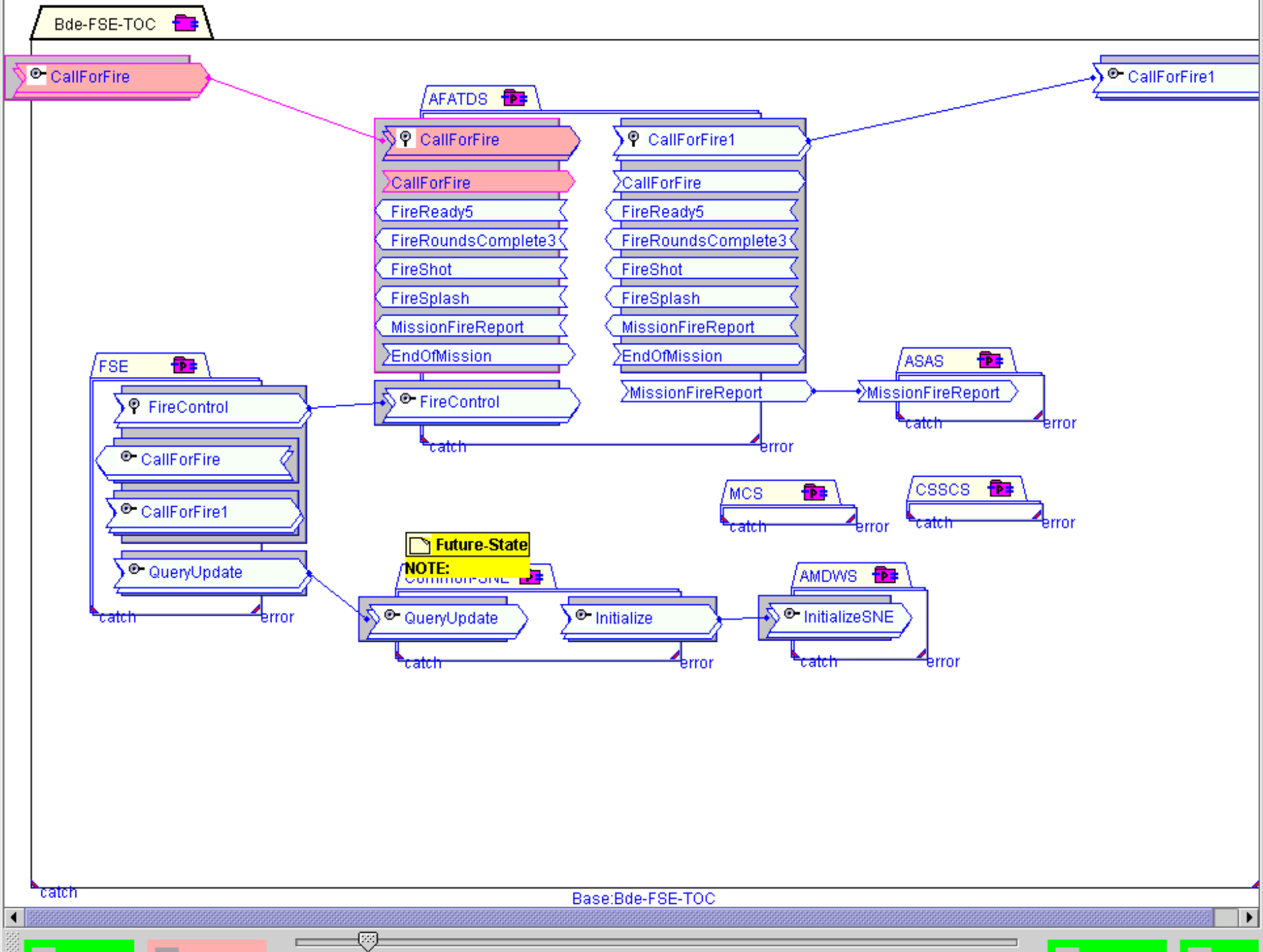

Co-FIST-TOC
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
FOS
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
InitializeSNE
catch
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
error
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
Common-SNE
QueryUpdate
Initialize
catch
error
Future-State
Integrated Environment/Terrain
catch
Base:Co-FIST-TOC
erro

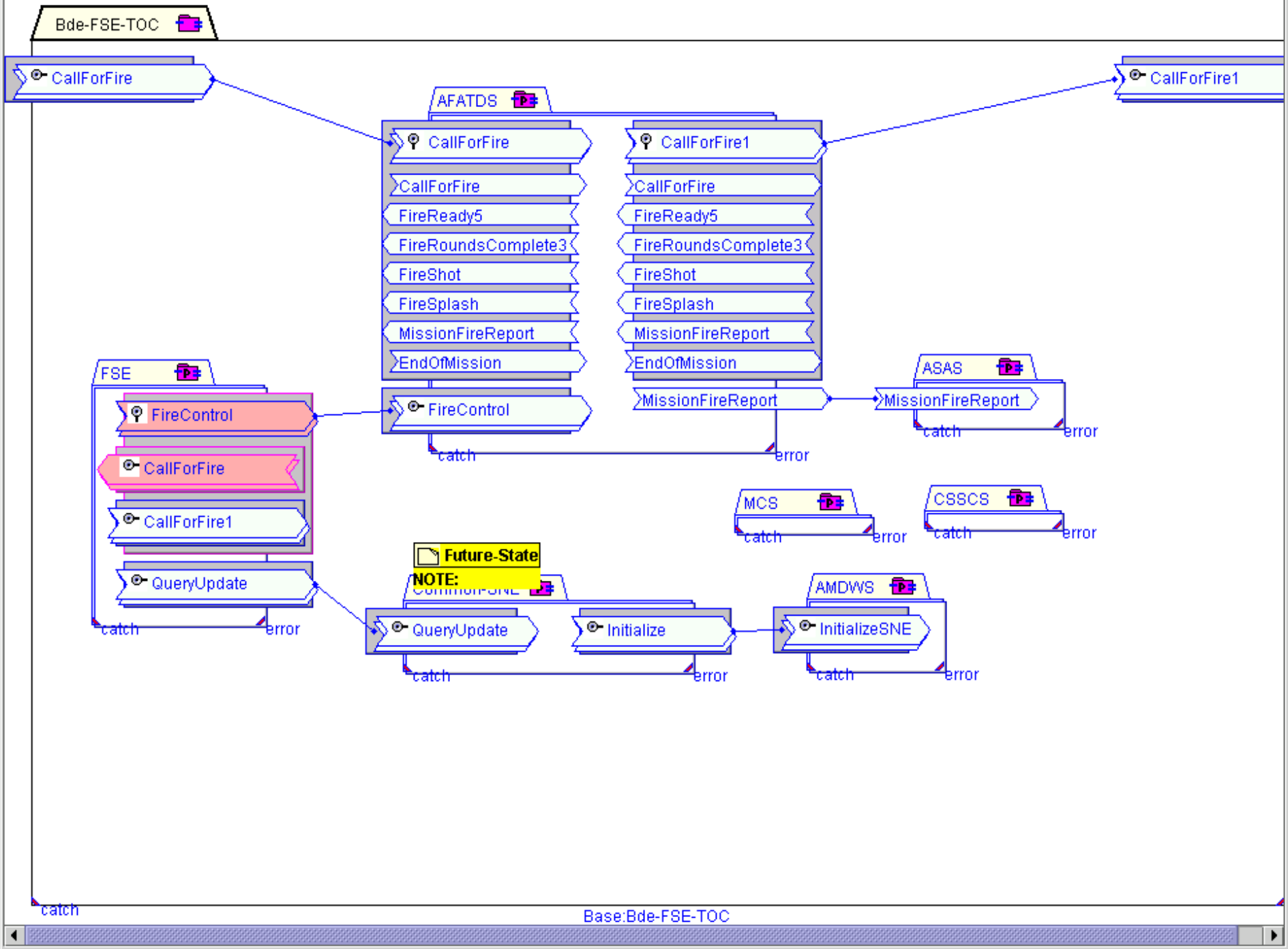


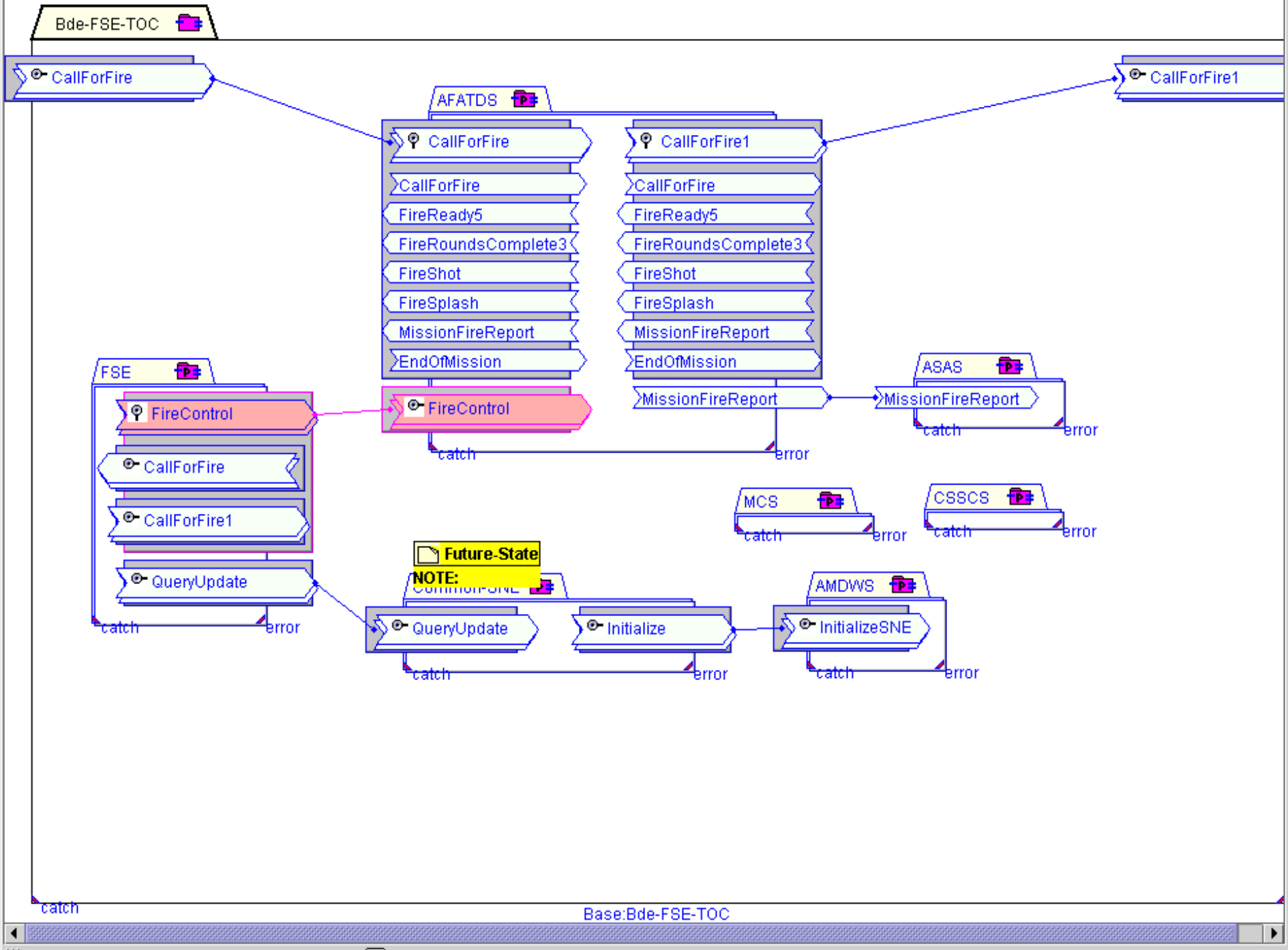
Base:IntegratedFireMission

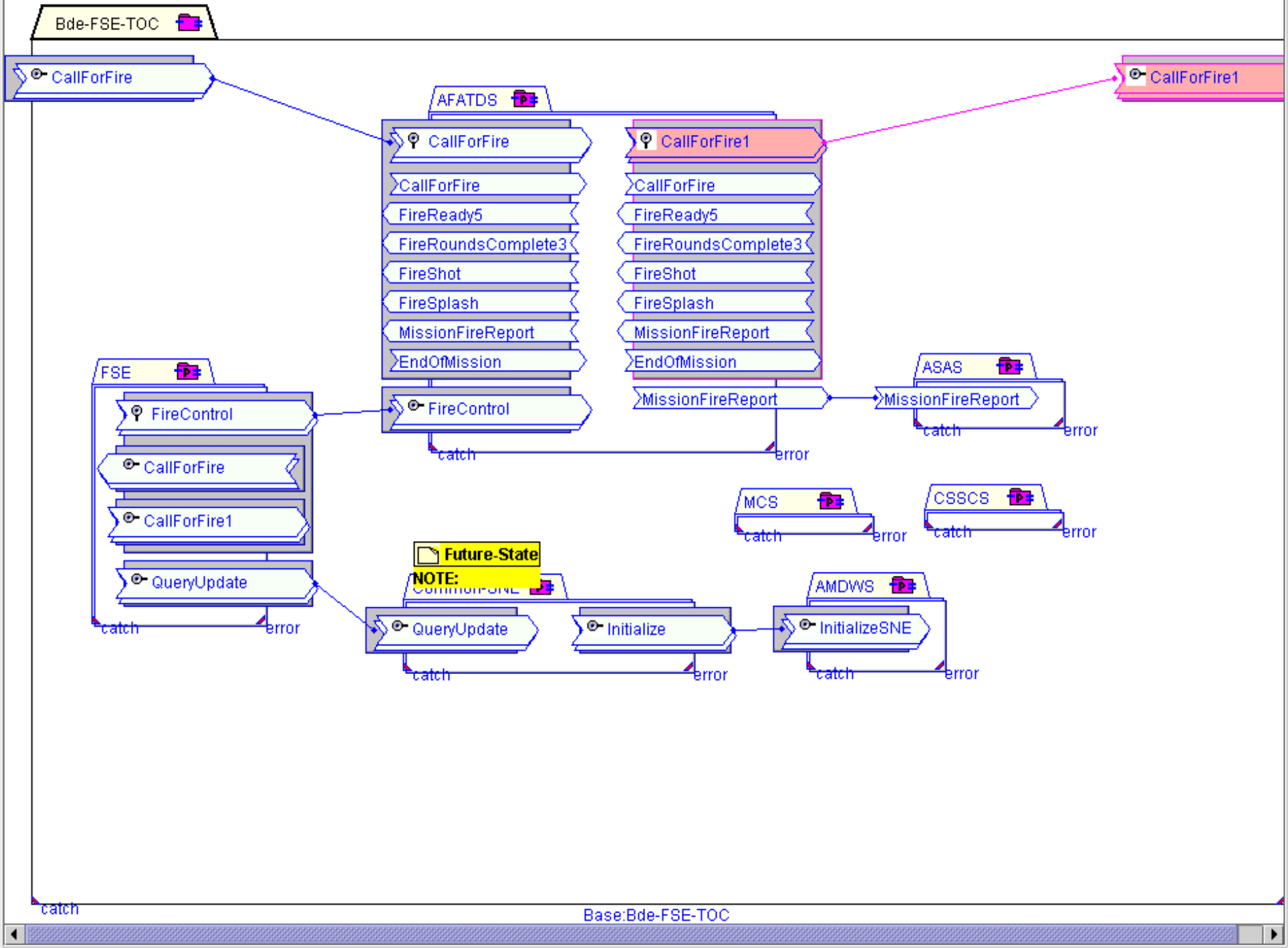


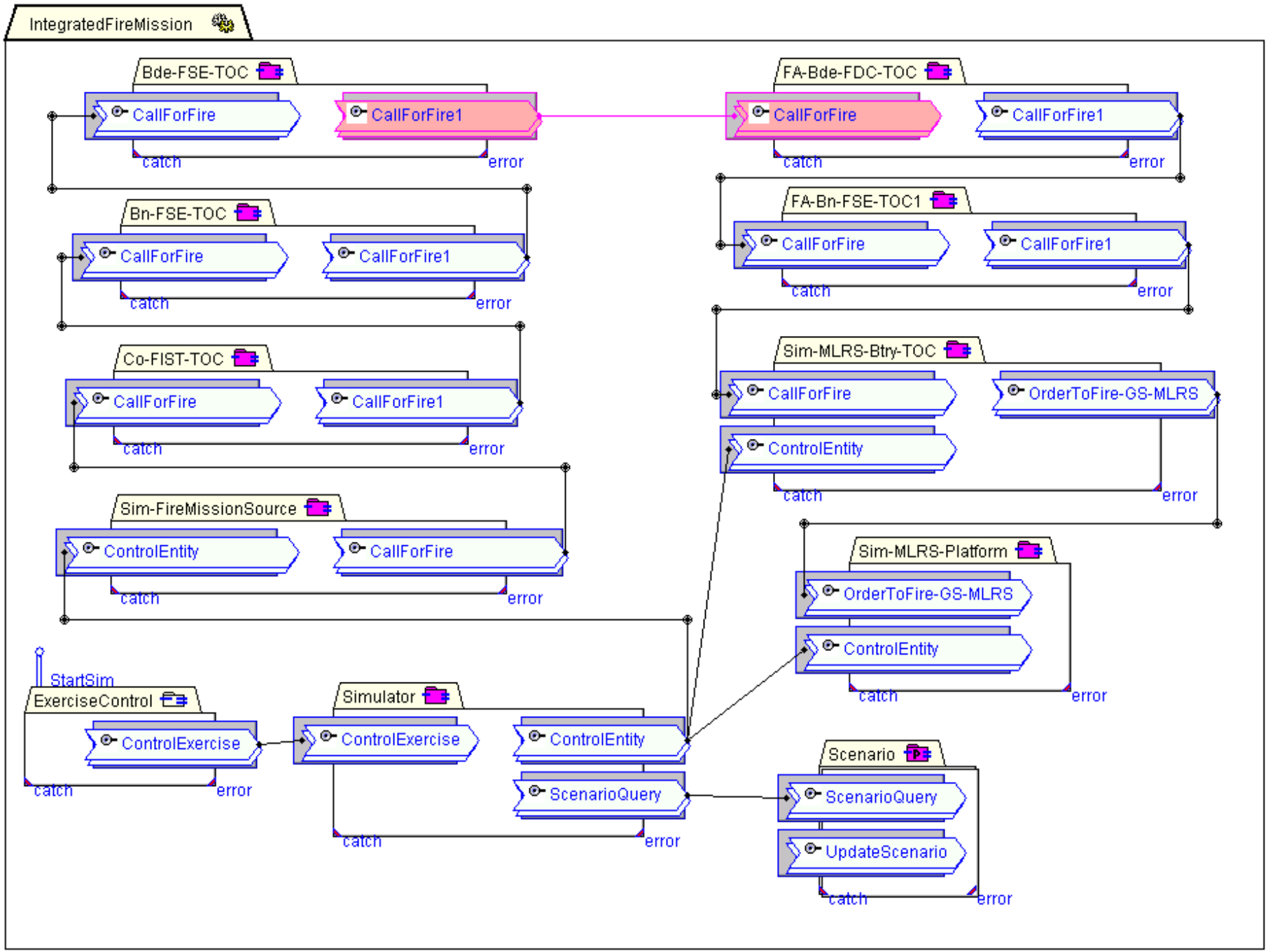
Base:IntegratedFireMission


Bde-FSE-TOC
CallForFire
AFATDS
CallForFire
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire1
FSE
FireControl
CallForFire
CallForFire1
QueryUpdate
catch
error
FireControl
catch
error
MissionFireReport
ASAS
MissionFireReport
catch
error
MCS
catch
error
CSSCS
catch
error
Future-State
NOTE:
QueryUpdate
Initialize
catch
error
AMDWS
InitializeSNE
catch
error
catch
Base:Bde-FSE-TOC

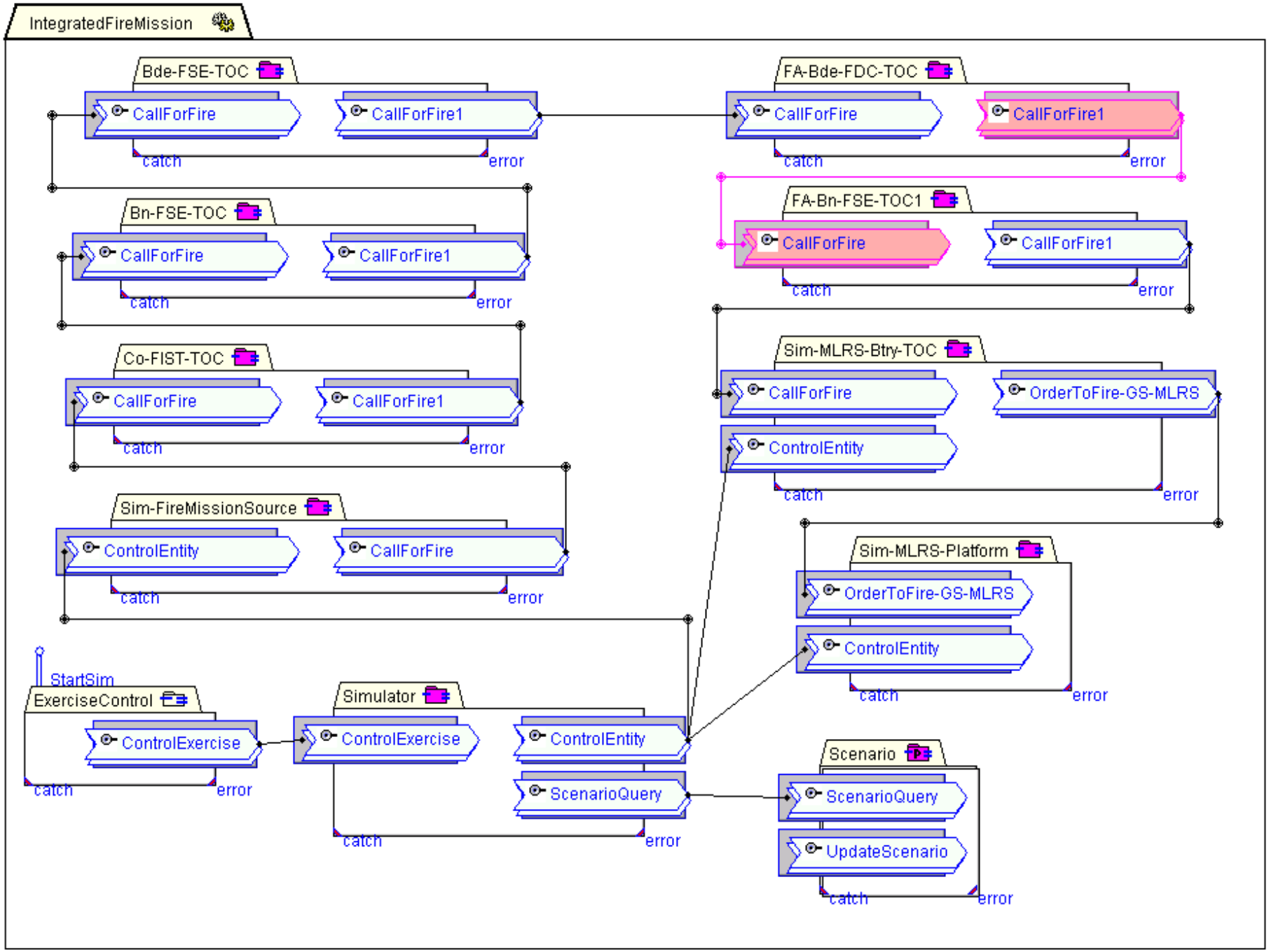

Bde-FSE-TOC
CallForFire
AFATDS
CallForFire
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire1
FSE
FireControl
CallForFire
CallForFire1
QueryUpdate
catch
error
FireControl
catch
error
MissionFireReport
ASAS
MissionFireReport
catch
error
MCS
catch
error
CSSCS
catch
error
Future-State
NOTE:
QueryUpdate
Initialize
catch
error
AMDWS
InitializeSNE
catch
error
catch
Base:Bde-FSE-TOC


Bde-FSE-TOC
CallForFire
CallForFire1
AFATDS
CallForFire
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
FSE
FireControl
CallForFire
CallForFire1
QueryUpdate
catch
error
ASAS
MissionFireReport
MCS
CSSCS
Future-State
NOTE:
QueryUpdate
Initialize
AMDWS
InitializeSNE
Base:Bde-FSE-TOC

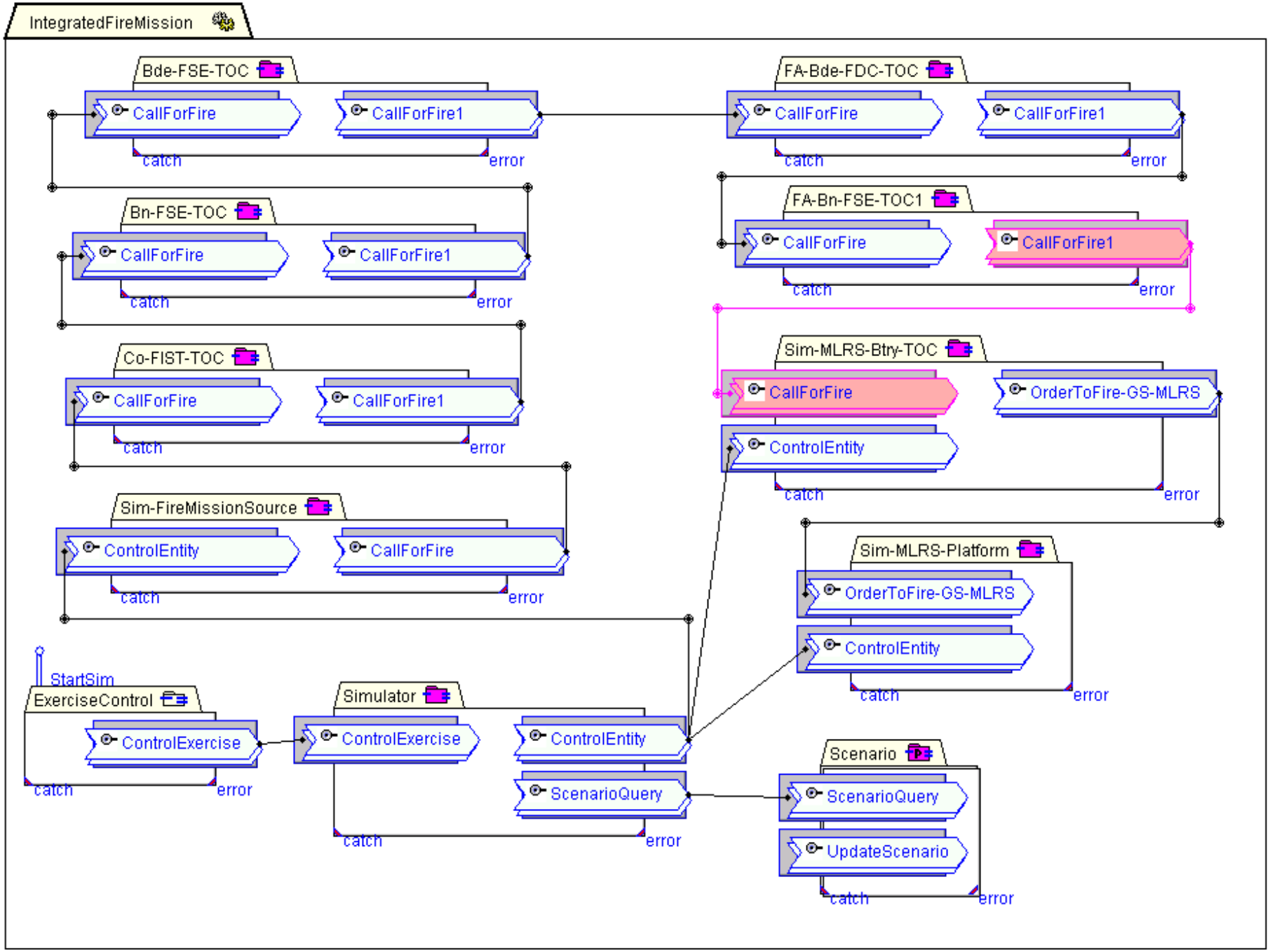

Bde-FSE-TOC
CallForFire
AFATDS
CallForFire
CallForFire1
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
FSE
FireControl
CallForFire
CallForFire1
QueryUpdate
FireControl
MissionFireReport
ASAS
MissionFireReport
catch
error
catch
error
MCS
CSSCS
catch
error
catch
error
Future-State
NOTE:
QueryUpdate
Initialize
AMDWS
InitializeSNE
catch
error
catch
error
catch
error
catch
Base:Bde-FSE-TOC



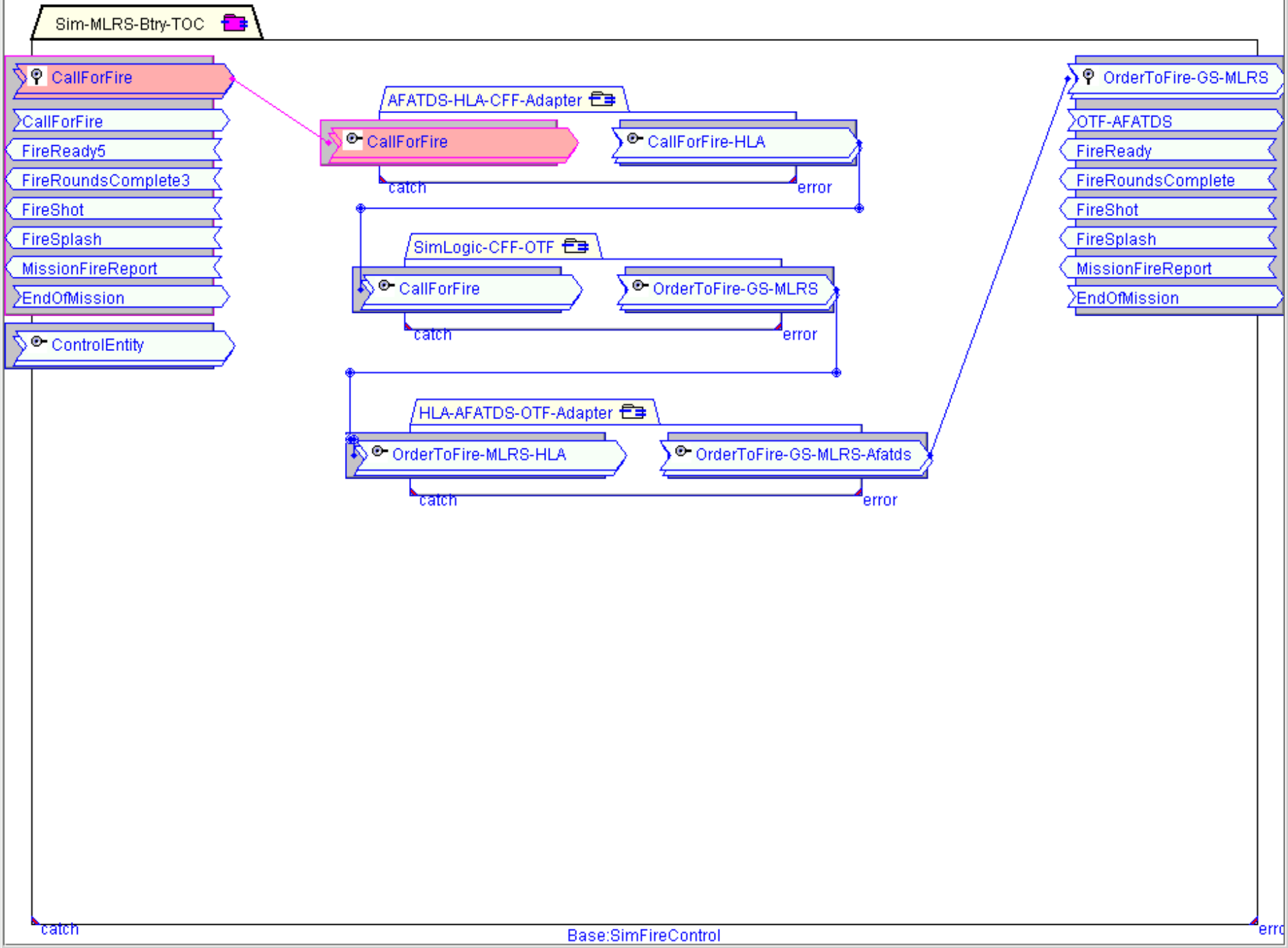
Base:IntegratedFireMission

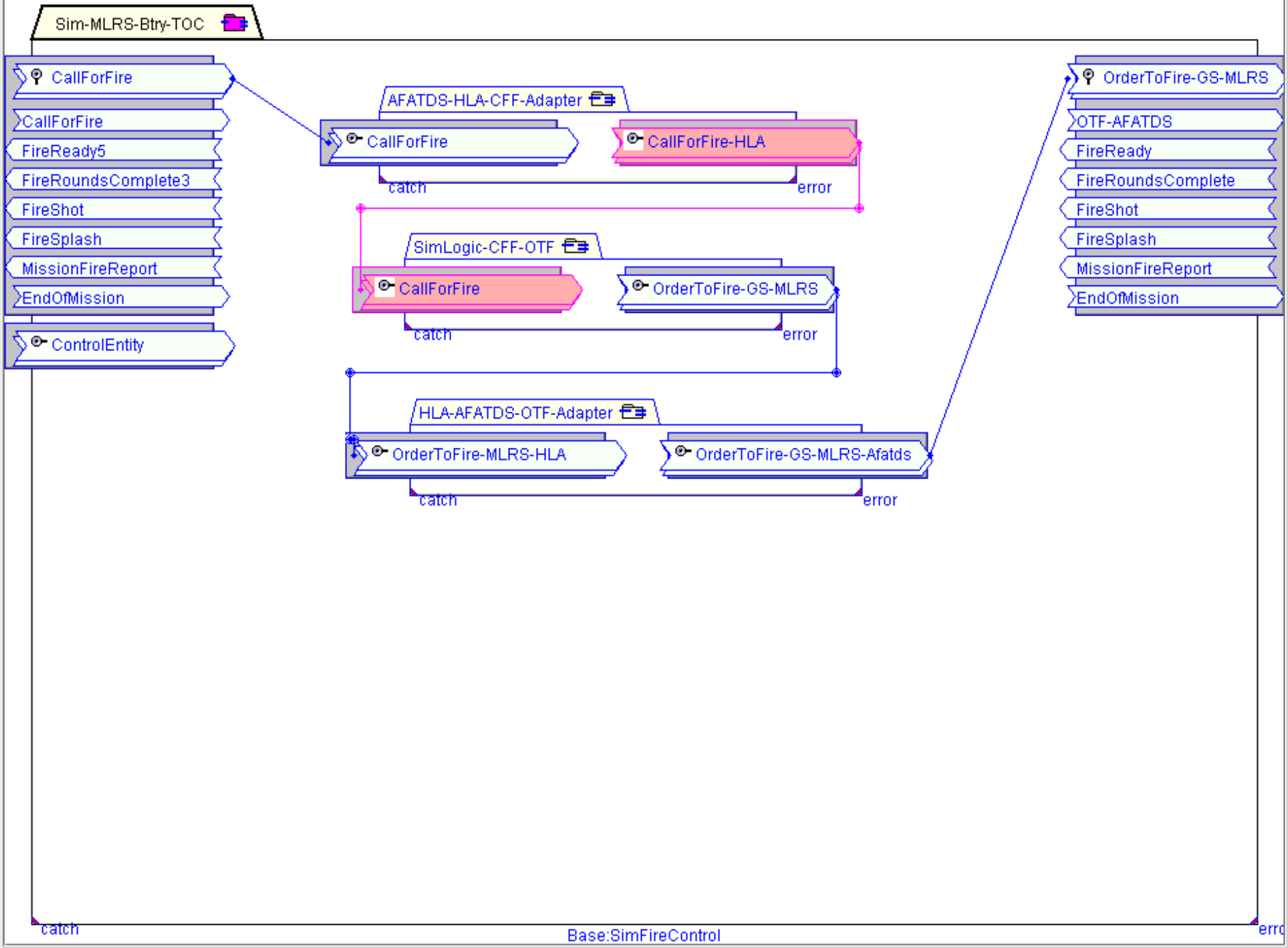


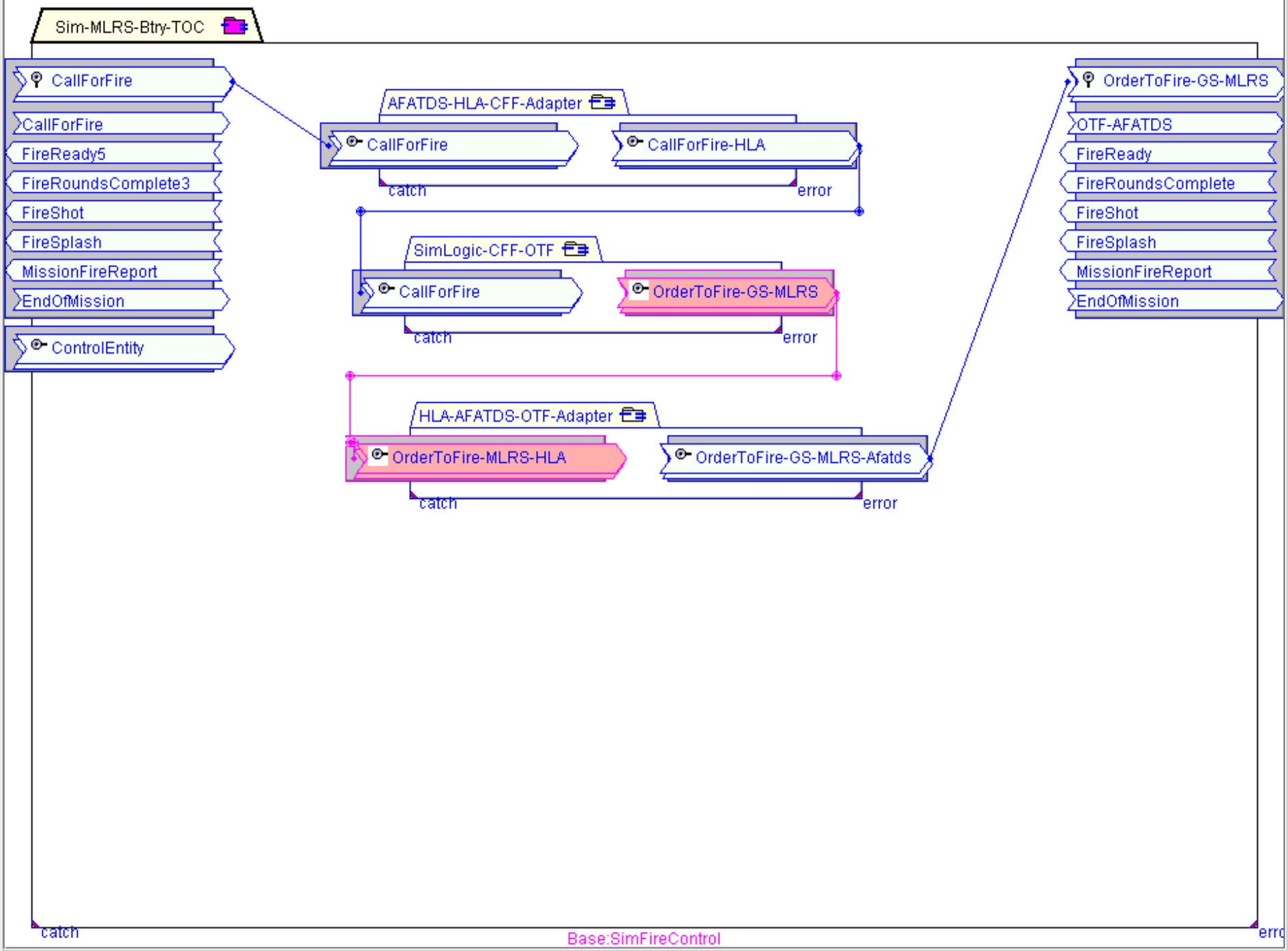
Base:IntegratedFireMission

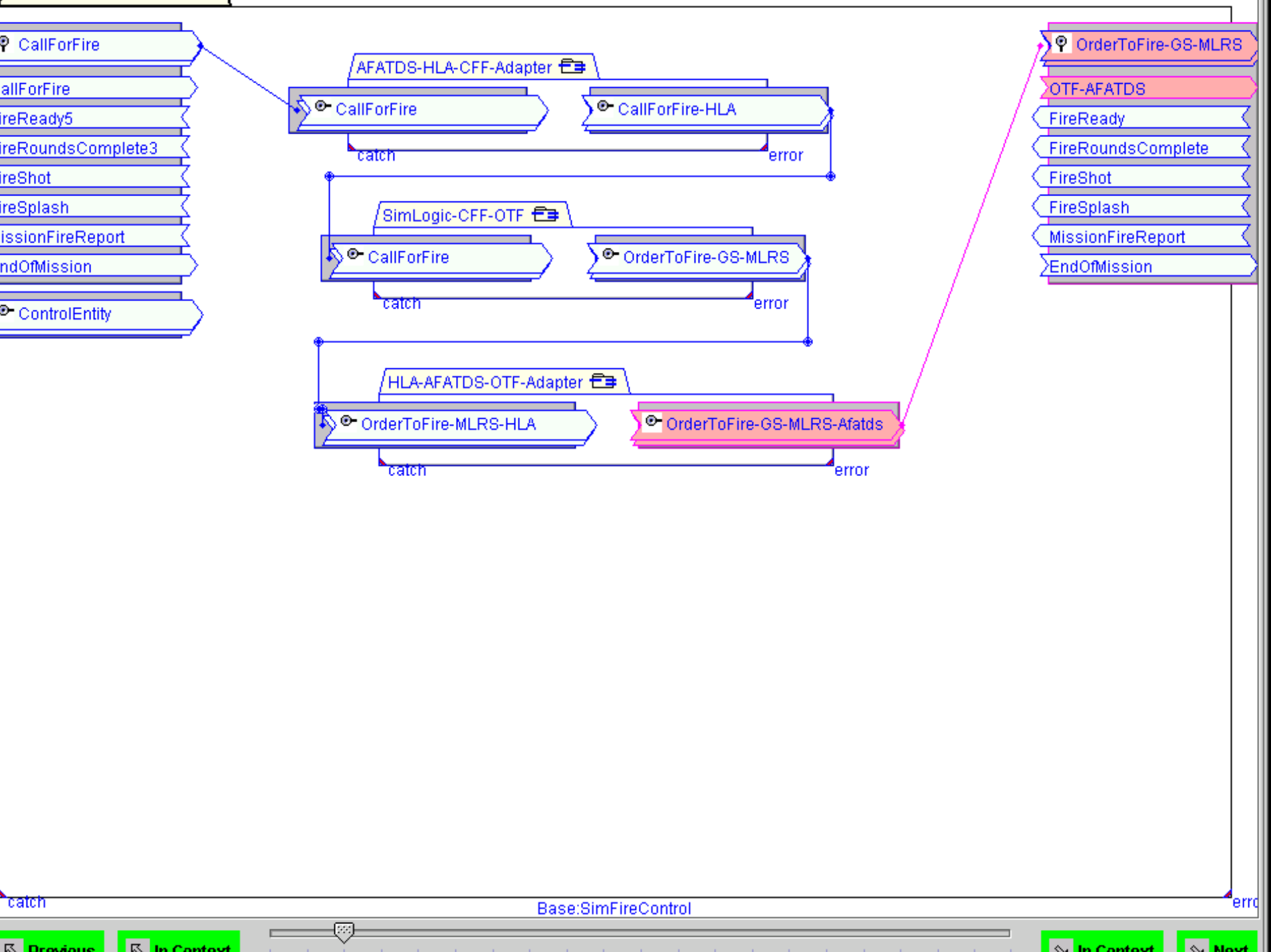


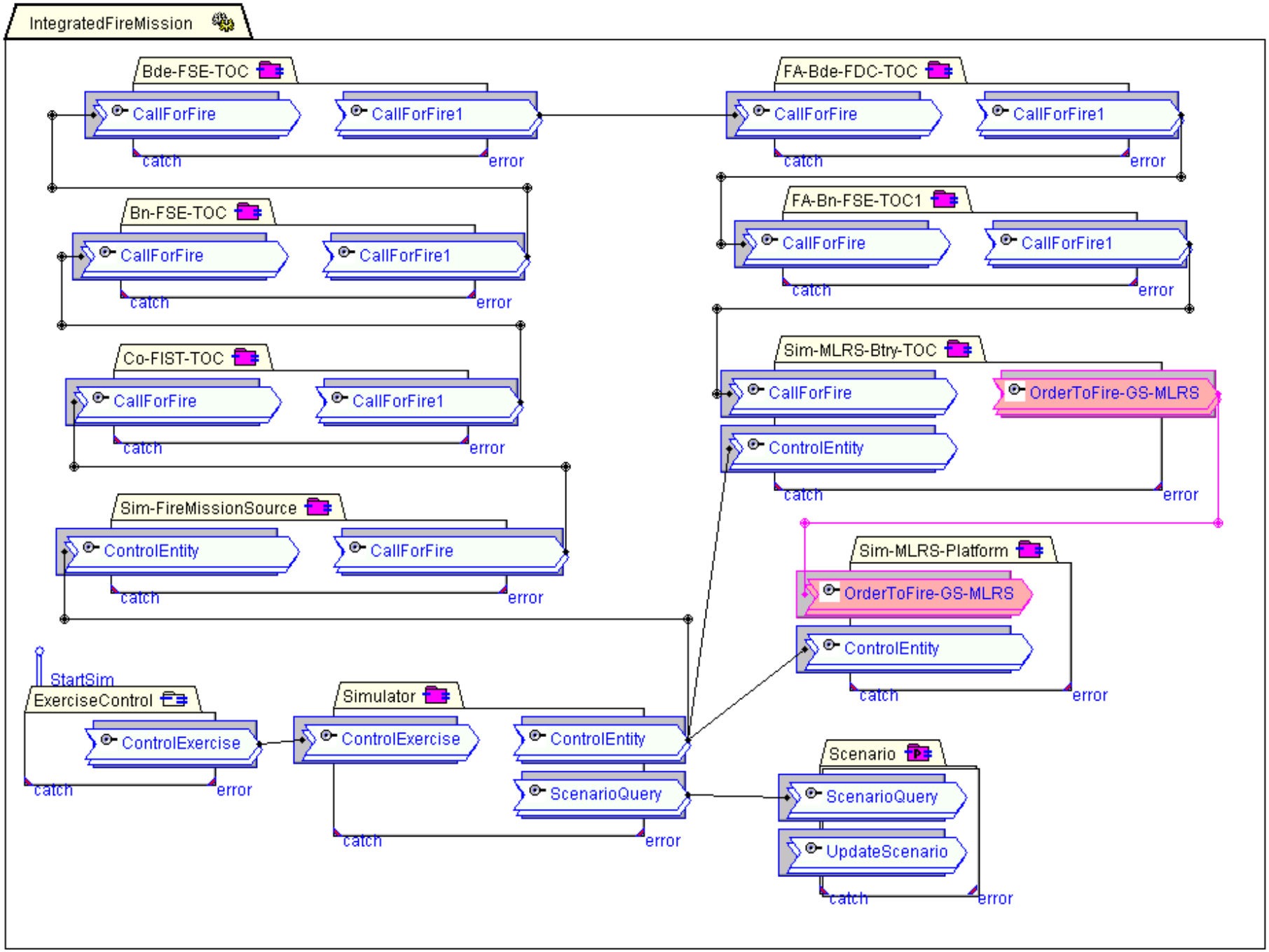
Base:IntegratedFireMission


Sim-MLRS-Btry-TOC
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
ControlEntity
AFATDS-HLA-CFF-Adapter
CallForFire
CallForFire-HLA
catch
error
SimLogic-CFF-OTF
CallForFire
OrderToFire-GS-MLRS
catch
error
HLA-AFATDS-OTF-Adapter
OrderToFire-MLRS-HLA
OrderToFire-GS-MLRS-Afatds
catch
error
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
Base:SimFireControl

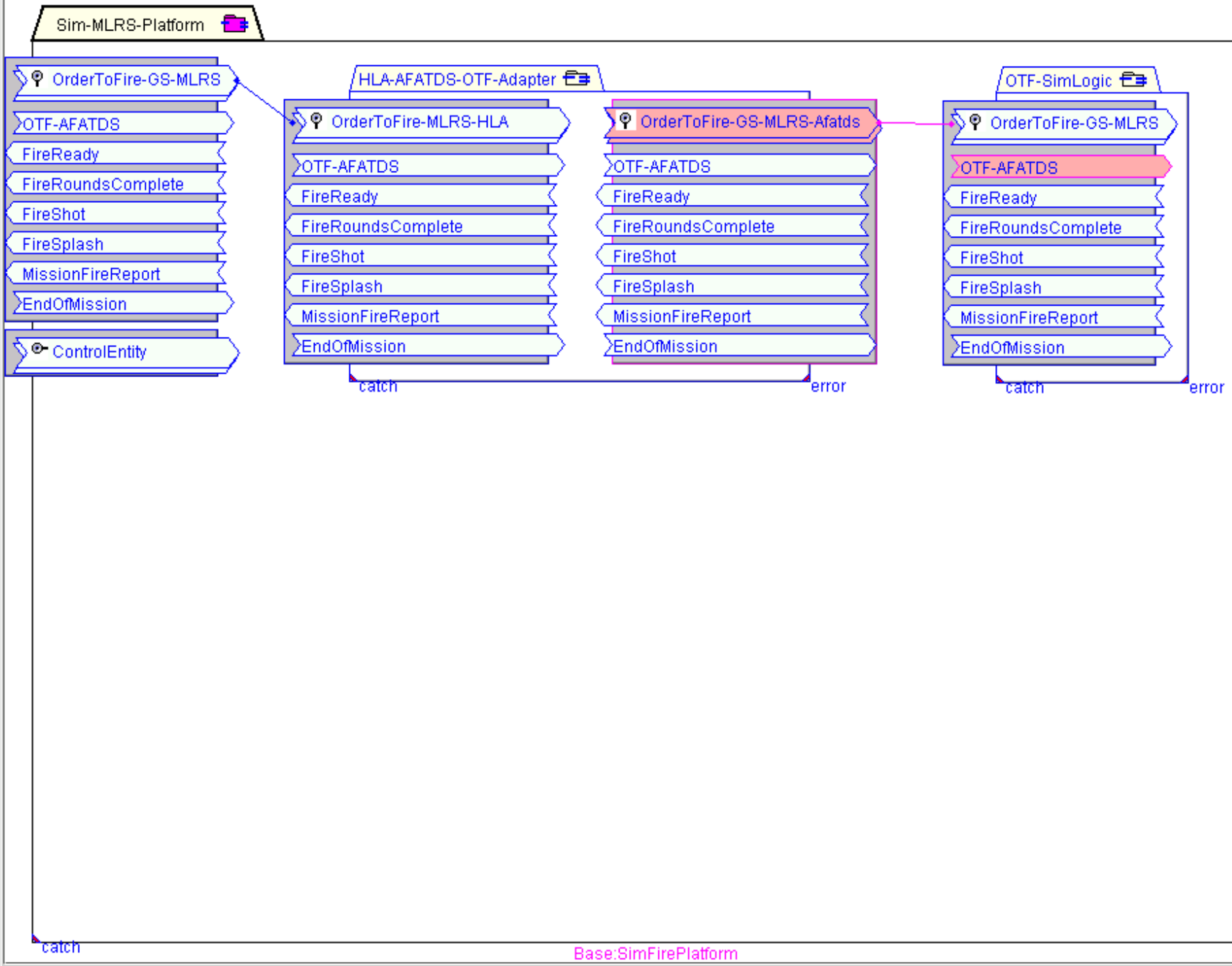

Sim-MLRS-Btry-TOC
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
ControlEntity
AFATDS-HLA-CFF-Adapter
CallForFire
CallForFire-HLA
catch
error
SimLogic-CFF-OTF
CallForFire
OrderToFire-GS-MLRS
catch
error
HLA-AFATDS-OTF-Adapter
OrderToFire-MLRS-HLA
OrderToFire-GS-MLRS-Afatds
catch
error
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
Base:SimFireControl

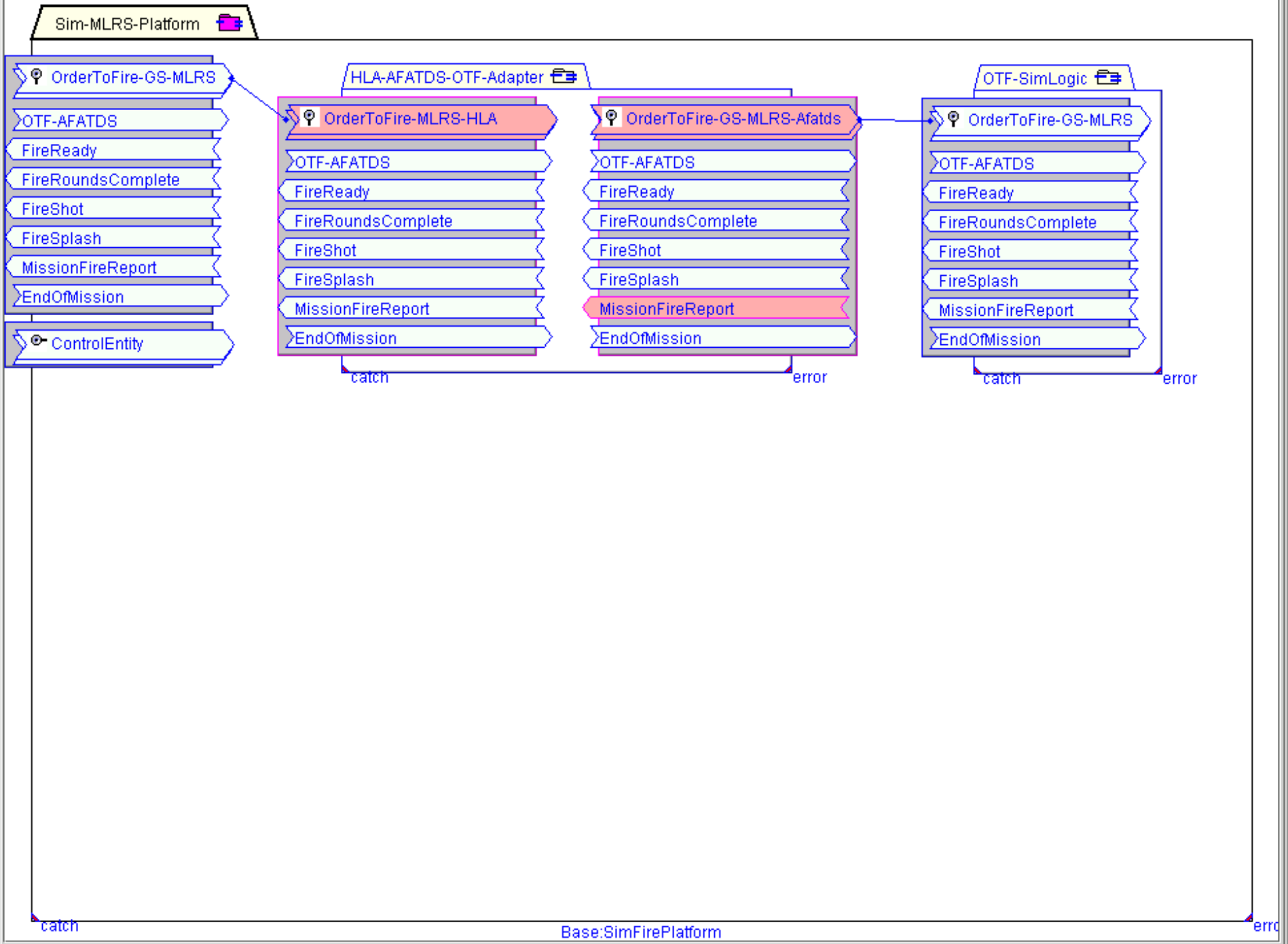

Sim-MLRS-Btry-TOC
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
ControlEntity
AFATDS-HLA-CFF-Adapter
CallForFire
CallForFire-HLA
catch
error
SimLogic-CFF-OTF
CallForFire
OrderToFire-GS-MLRS
catch
error
HLA-AFATDS-OTF-Adapter
OrderToFire-MLRS-HLA
OrderToFire-GS-MLRS-Afatds
catch
error
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
Base:SimFireControl

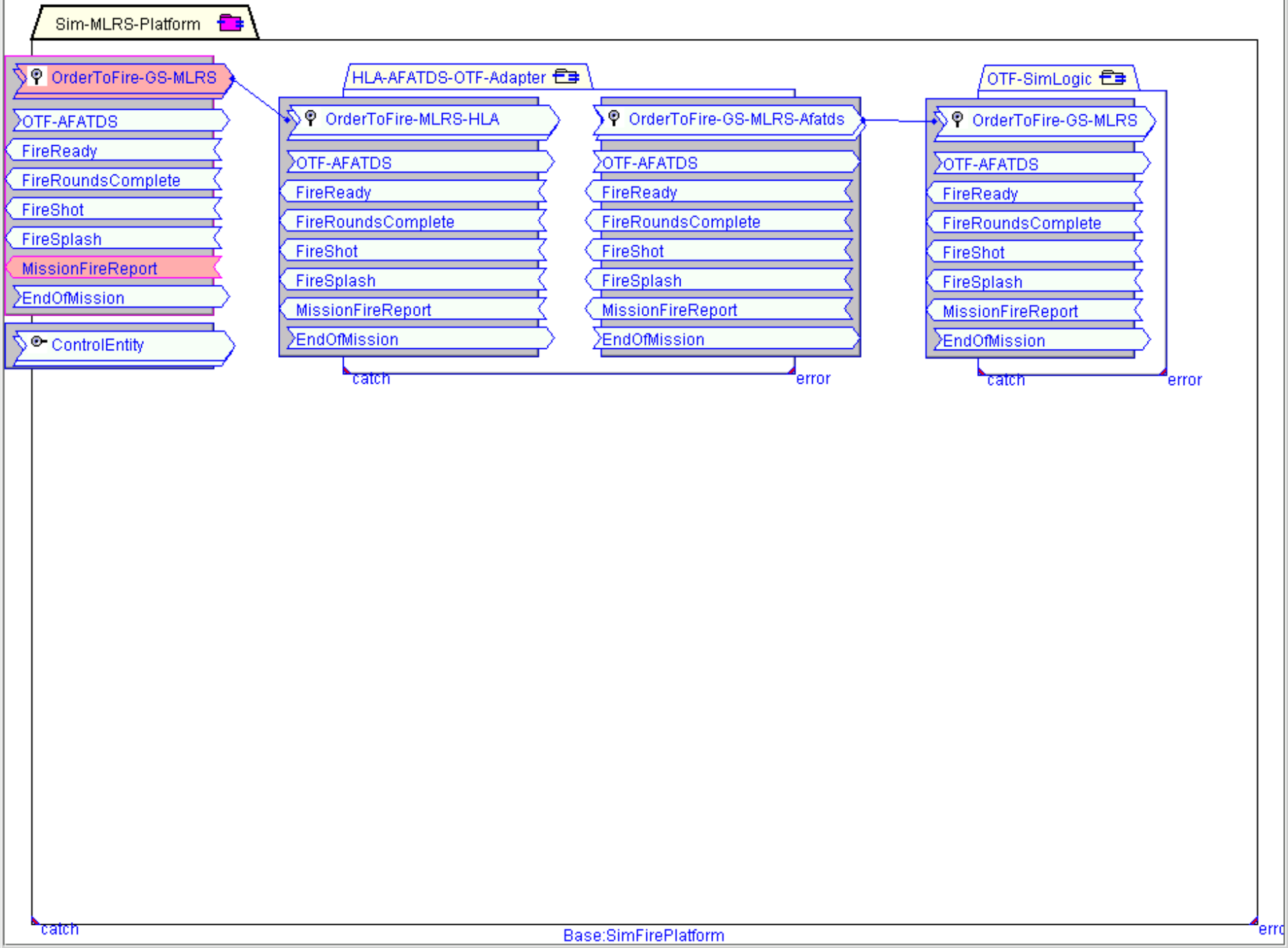

CallForFire
allForFire
ireReady5
ireRoundsComplete3
ireShot
ireSplash
issionFireReport
ndOfMission
ControlEntity
AFATDS-HLA-CFF-Adapter
CallForFire
CallForFire-HLA
catch
error
SimLogic-CFF-OTF
CallForFire
OrderToFire-GS-MLRS
catch
error
HLA-AFATDS-OTF-Adapter
OrderToFire-MLRS-HLA
OrderToFire-GS-MLRS-Afatds
catch
error
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
Base:SimFireControl
erro

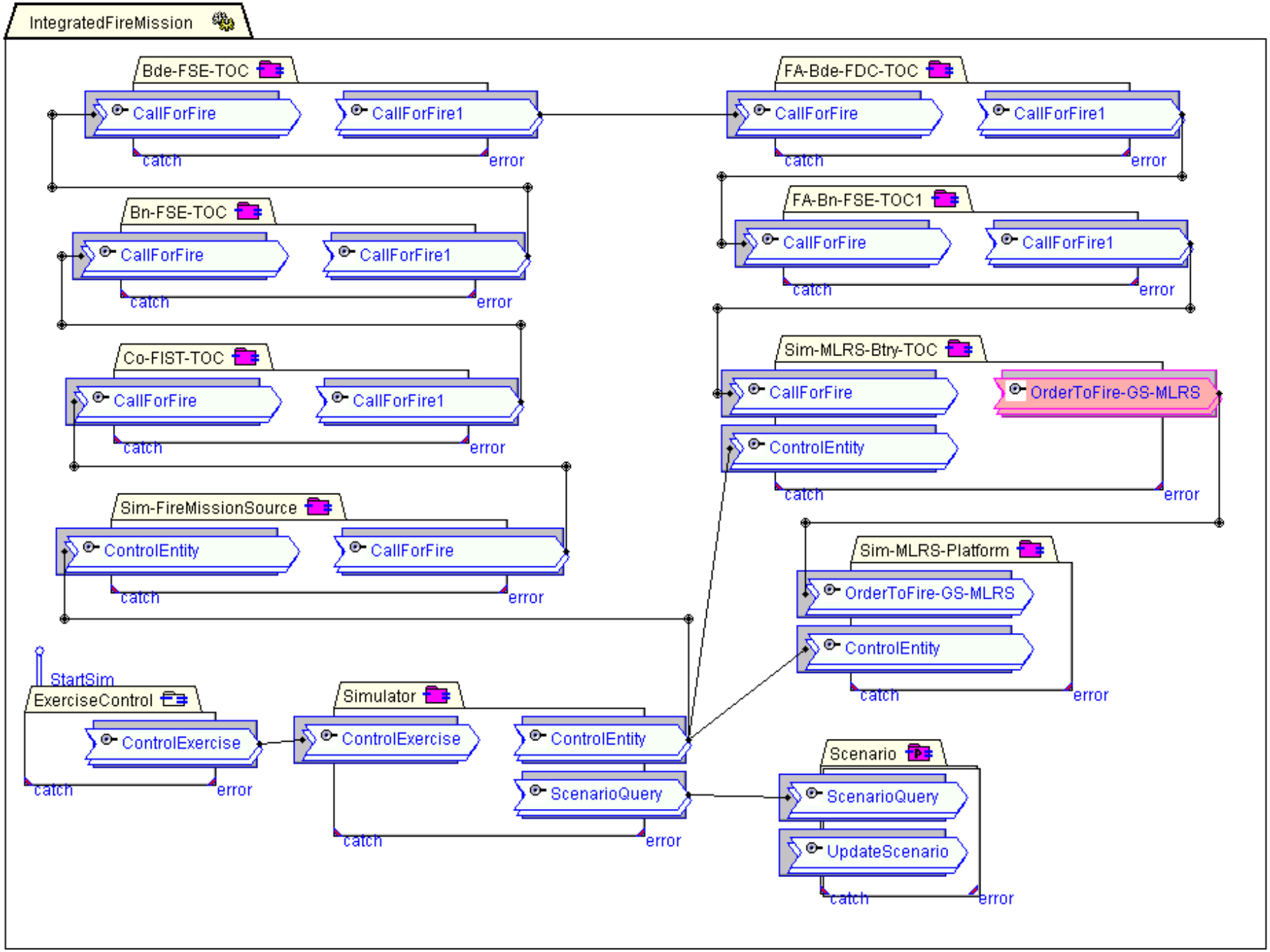


Base:IntegratedFireMission

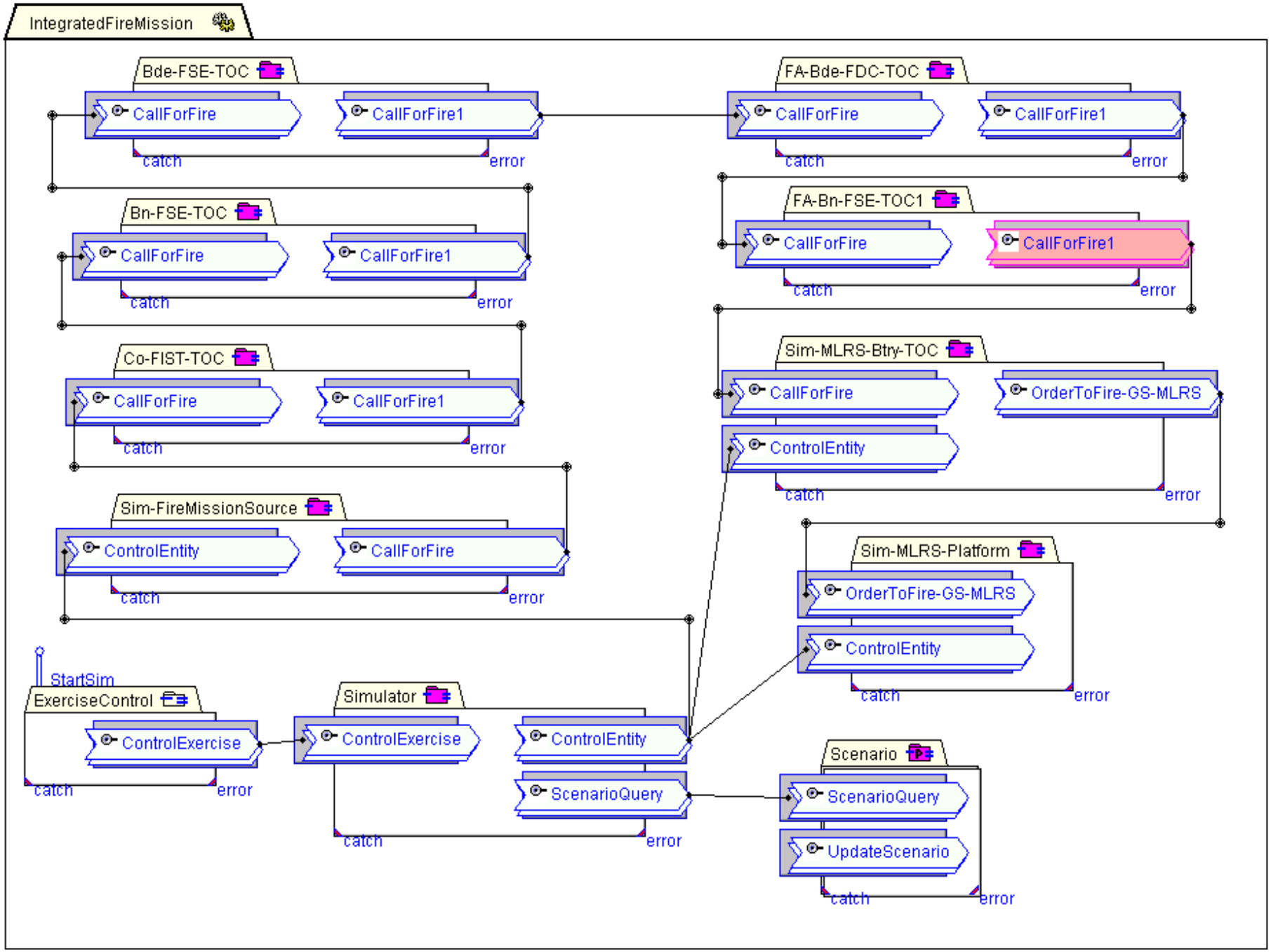

Sim-MLRS-Platform
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
ControlEntity
HLA-AFATDS-OTF-Adapter
OrderToFire-MLRS-HLA
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
error
OrderToFire-GS-MLRS-Afatds
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
OTF-SimLogic
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
error
catch
Base:SimFirePlatform


Sim-MLRS-Platform
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
ControlEntity
HLA-AFATDS-OTF-Adapter
OrderToFire-MLRS-HLA
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
error
OrderToFire-GS-MLRS-Afatds
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
OTF-SimLogic
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
error
catch
Base:SimFirePlatform
erro

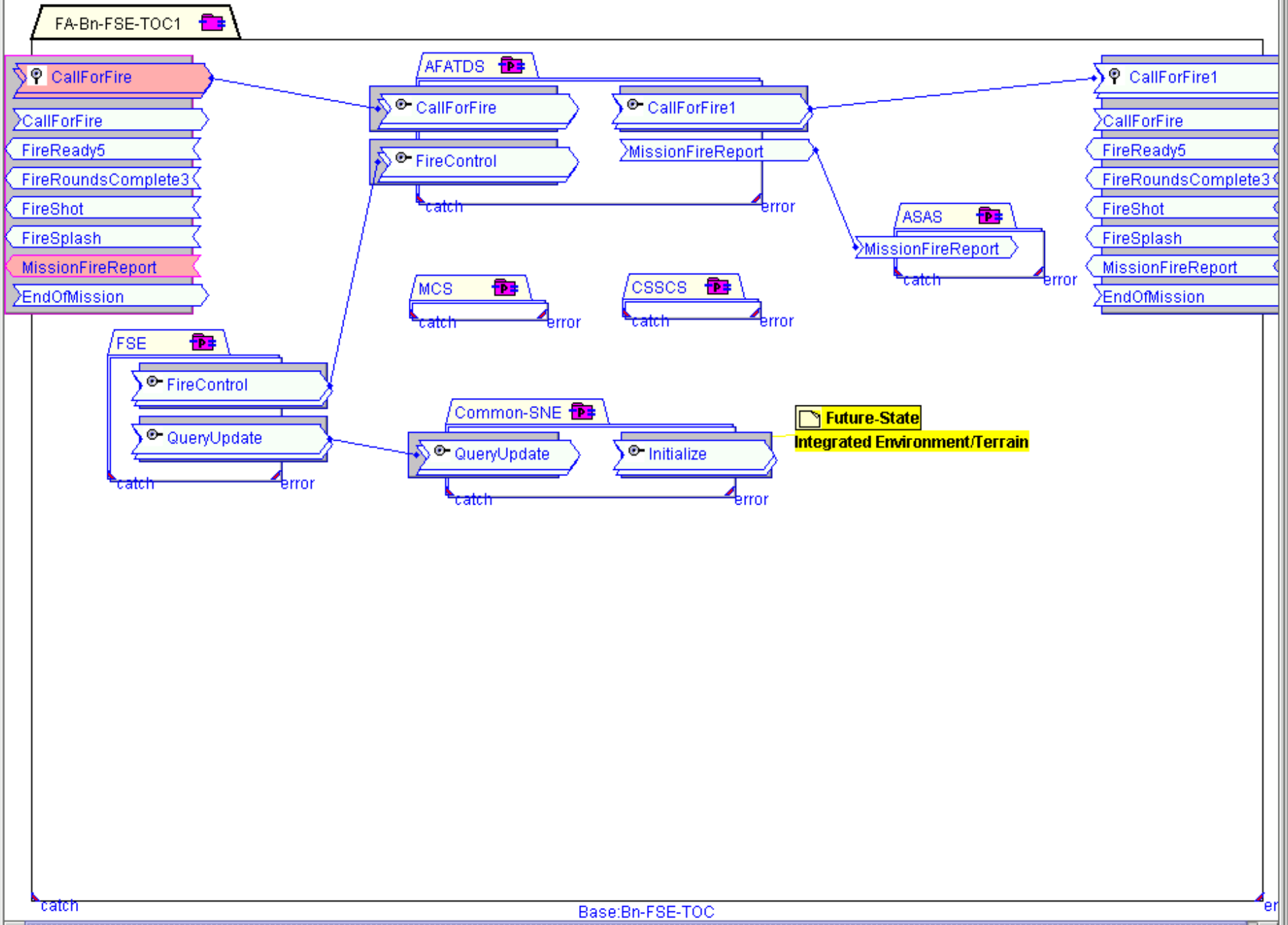

Sim-MLRS-Platform
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
ControlEntity
HLA-AFATDS-OTF-Adapter
OrderToFire-MLRS-HLA
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
error
OrderToFire-GS-MLRS-Afatds
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
OTF-SimLogic
OrderToFire-GS-MLRS
OTF-AFATDS
FireReady
FireRoundsComplete
FireShot
FireSplash
MissionFireReport
EndOfMission
catch
error
catch
Base:SimFirePlatform

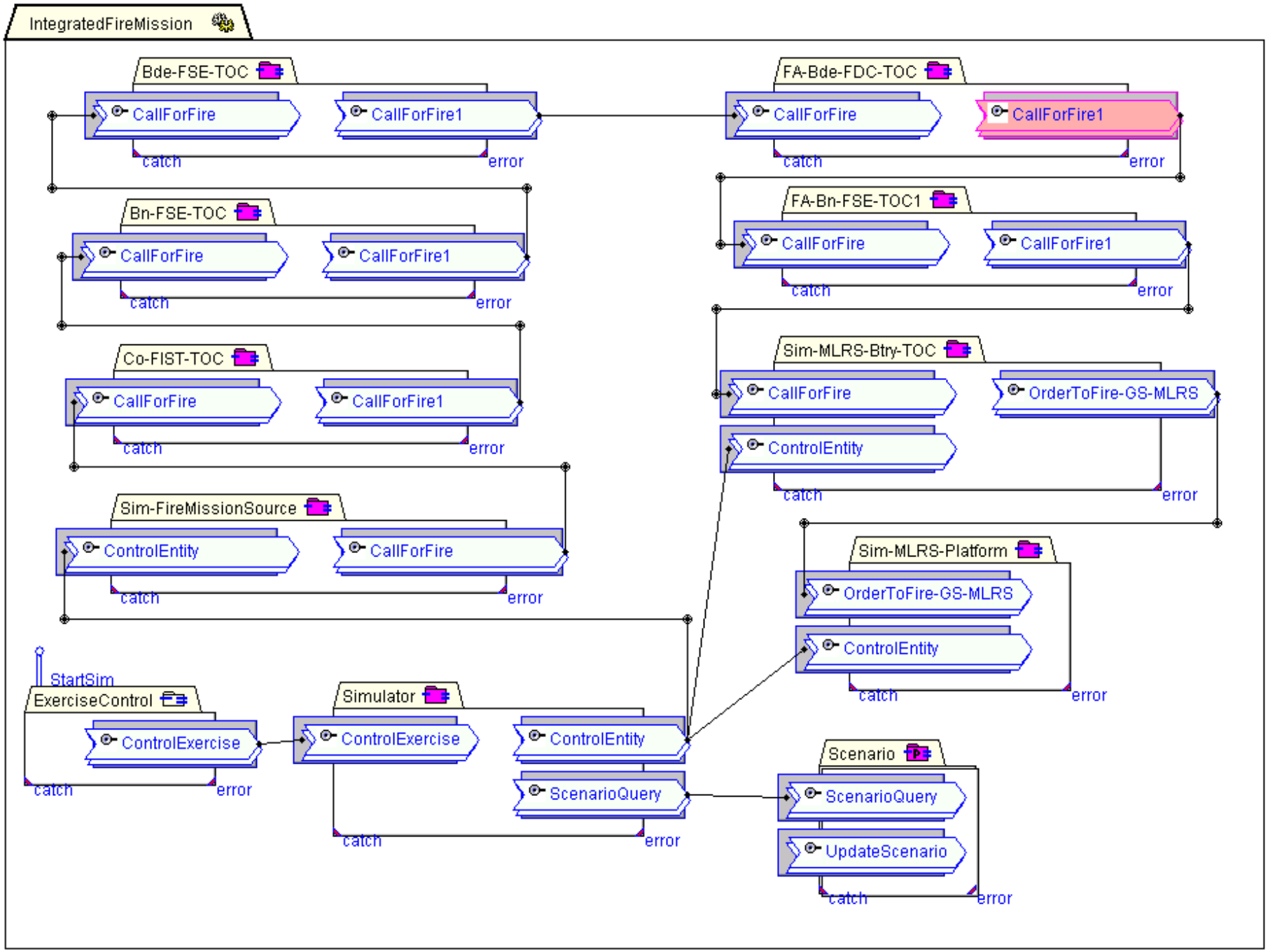


Base:IntegratedFireMission

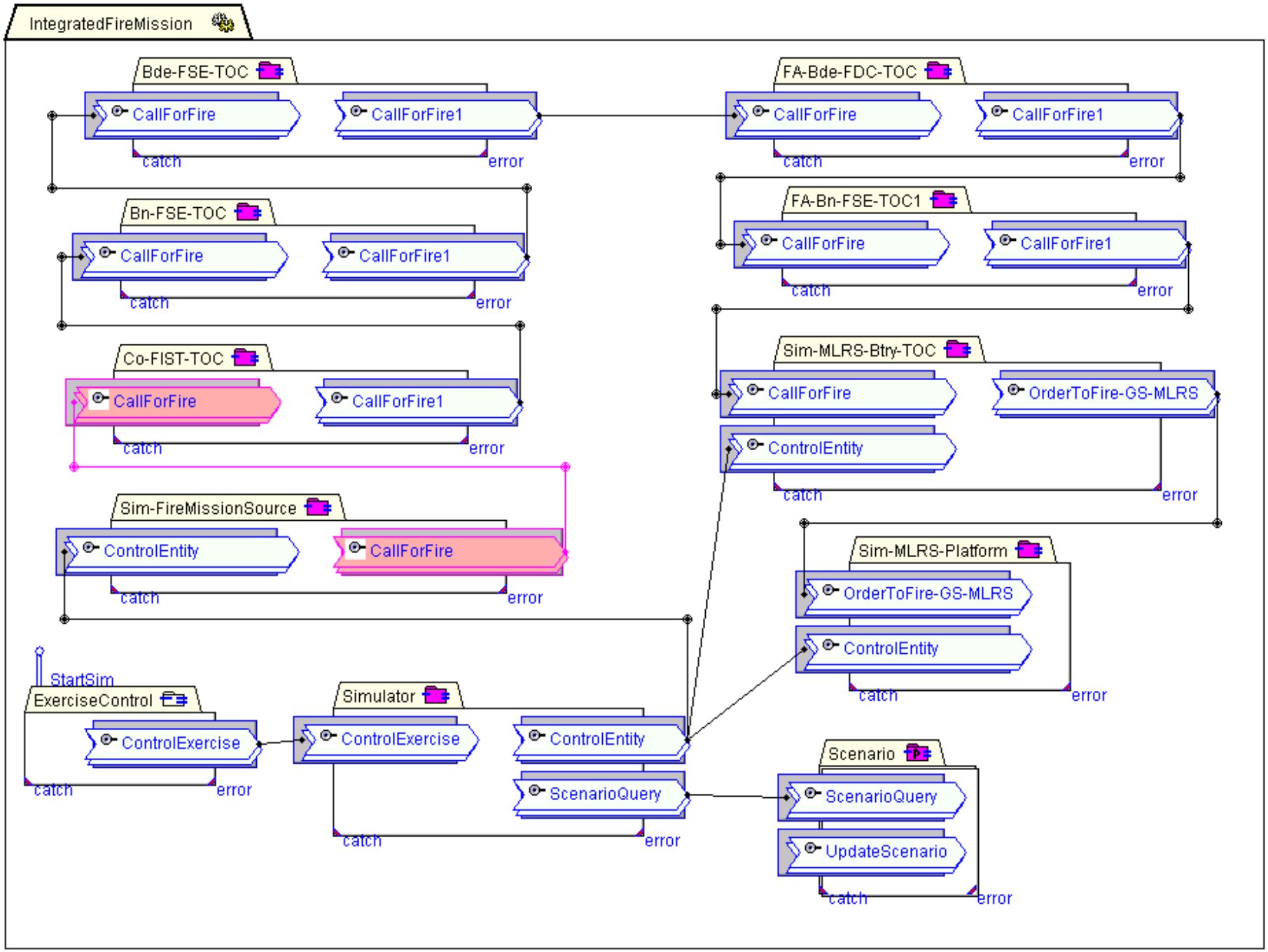


Base:IntegratedFireMission


FA-Bn-FSE-TOC1
CallForFire
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
AFATDS
CallForFire
FireControl
catch
error
CallForFire1
MissionFireReport
ASAS
MissionFireReport
catch
error
CallForFire1
CallForFire
FireReady5
FireRoundsComplete3
FireShot
FireSplash
MissionFireReport
EndOfMission
MCS
catch
error
CSSCS
catch
error
FSE
FireControl
QueryUpdate
catch
error
Common-SNE
QueryUpdate
Initialize
catch
error
Future-State
Integrated Environment/Terrain
catch
Base:Bn-FSE-TOC



Base:IntegratedFireMission

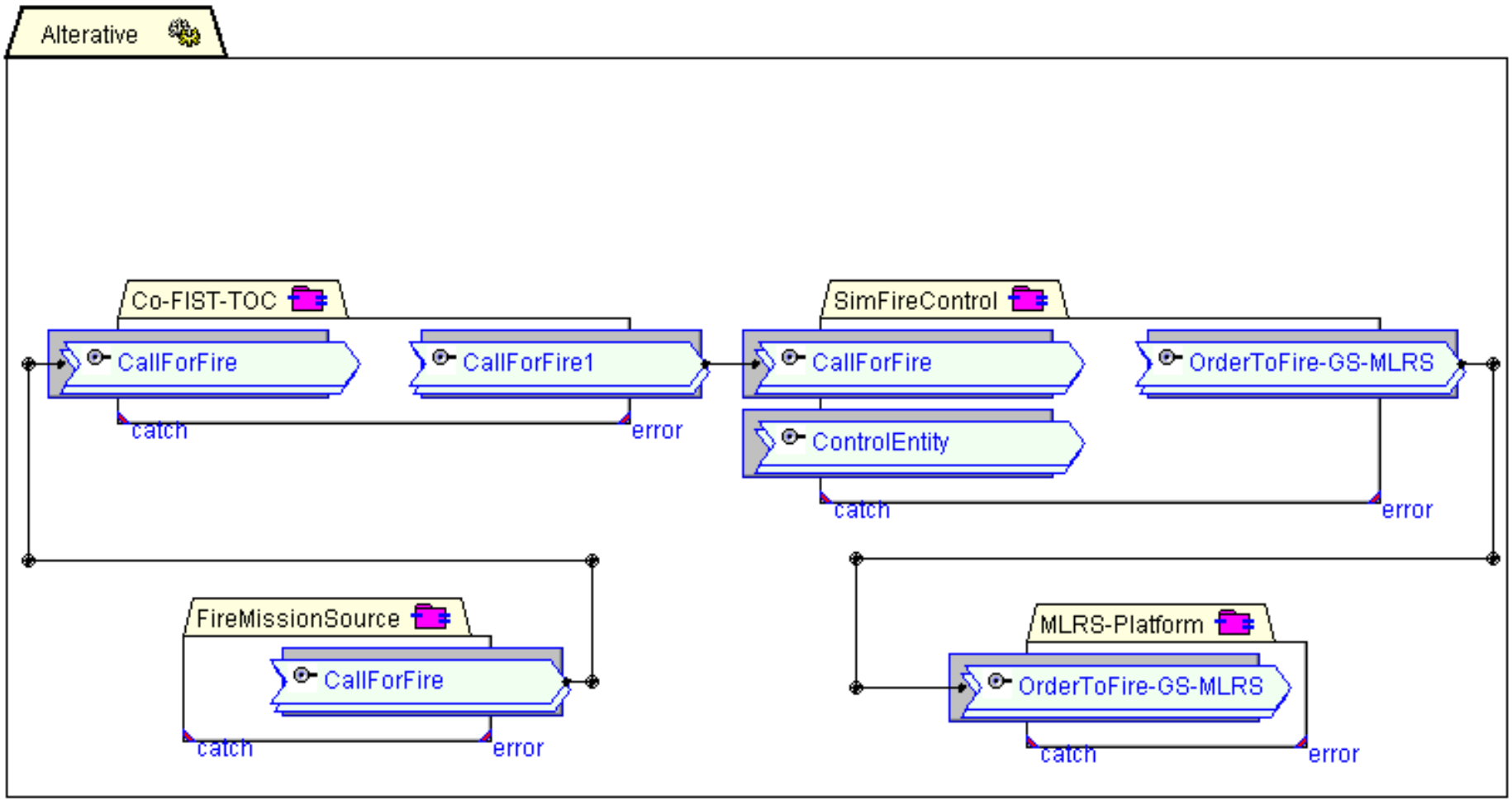


Base:IntegratedFireMission

Simulation Summary

- We can simulate a process
- Integrating real and simulated components
- Understanding the interactions
- At any level of detail

Trying Alternatives



Base:Alterative