DEVS Standard for Modeling and Simulation in Web-Centric Environments

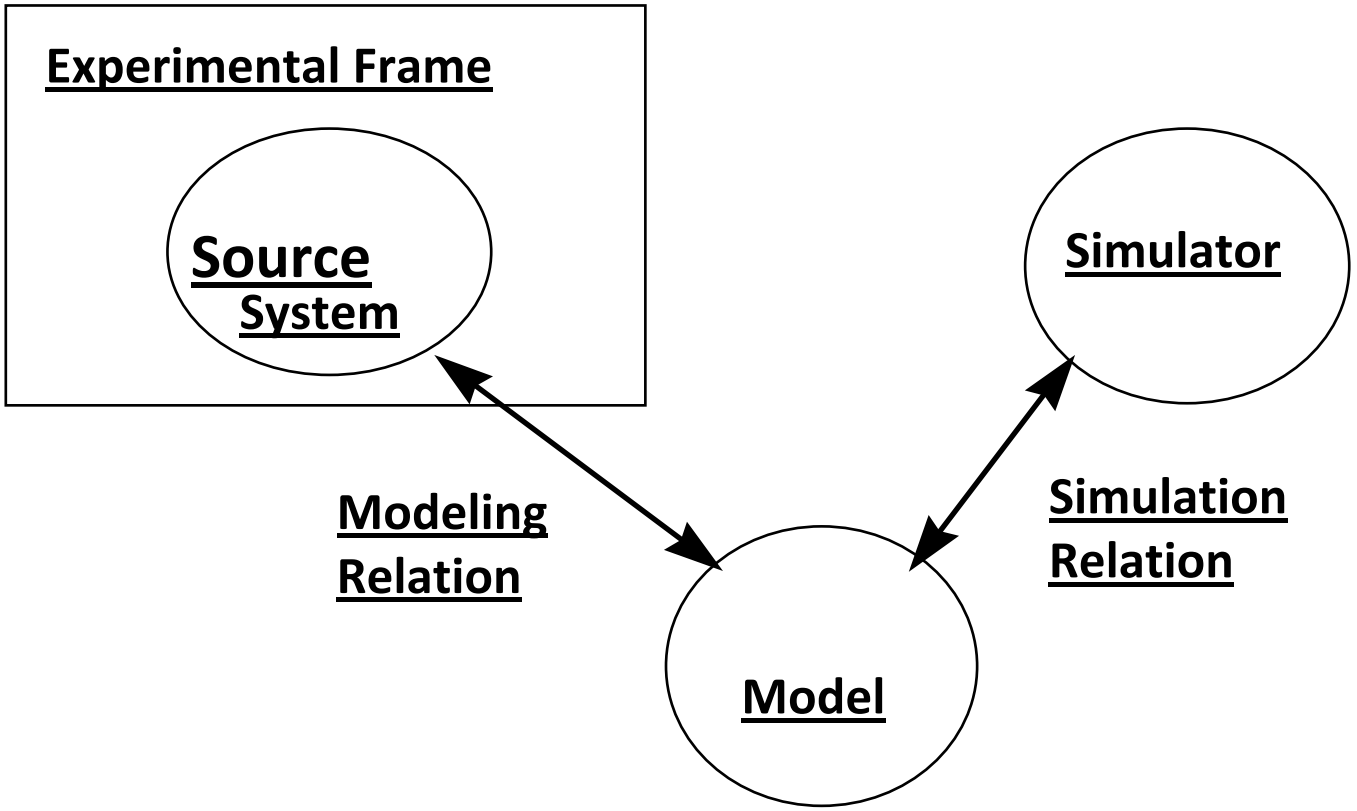
Bernard P. Zeigler
Arizona Center for Integrative Modeling and Simulation
University of Arizona

Presented at

Mosim08: Modélisation, Optimisation et Simulation des Systèmes

March 31-April 2 2008
Paris, France

Modeling and Simulation (M&S) Framework



DEVS – Formal Specification of a System

A discrete event system specification (DEVS) is a structure

$M=<X,S,Y,\delta_{int}, \delta_{ext,} \delta_{con}, \lambda,ta>$

where

X is the set of input values,

S is a set of states,

Y is the set of output values,

δ_{int} :S->S is the internal transition function,

$\delta_{ext,}$:$Q\times X$->S is the external transition function,

$\delta_{con,}$:$Q\times X$->S is the confluent transition function,

ta: S->$R^{+}_{0,\infty}$

Where

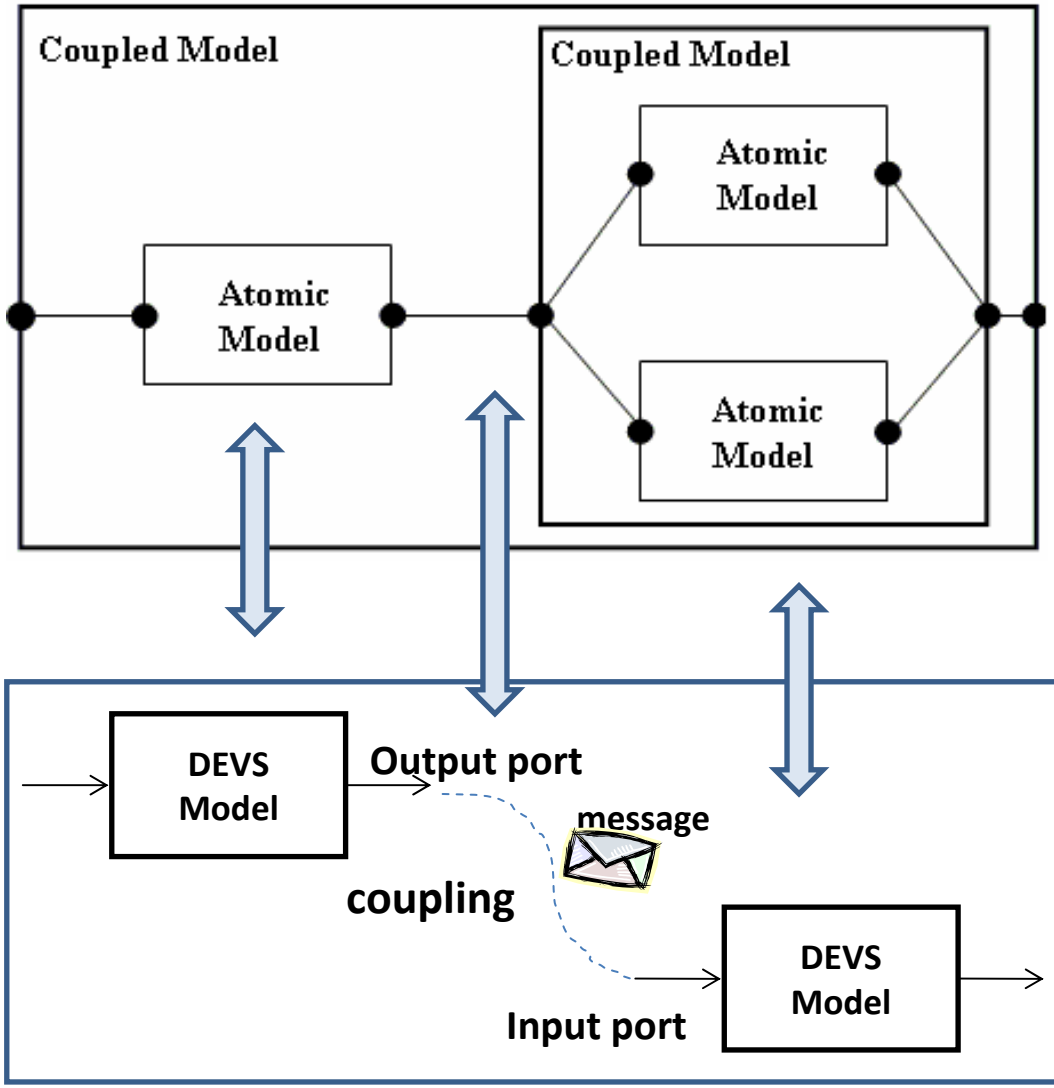
$Q=\{(s,e)\,|\,s\in S, 0 \leq e \leq ta(s)\}$ is the total state set,

e is the time elapsed since last transition,

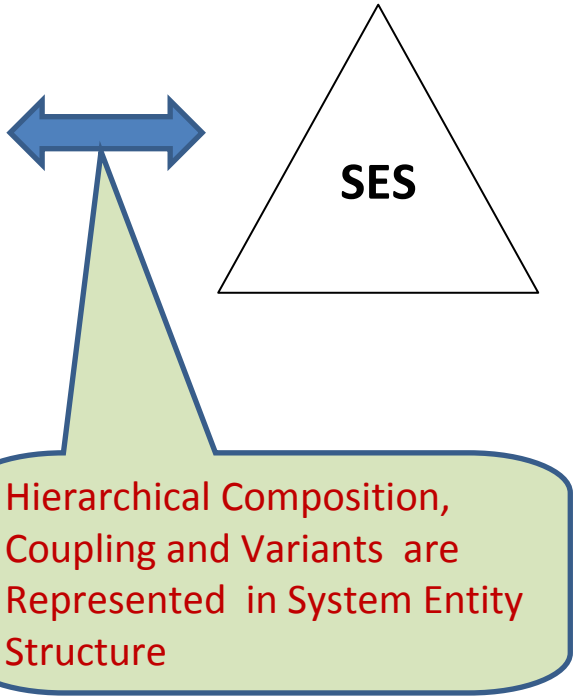
λ:S->Y is the output function and

$R^{+}_{0,\infty}$ is the set of positive reals with 0 and ∞

DEVS Hierarchical Modular Models



System Entity Structure



Advantages of DEVS

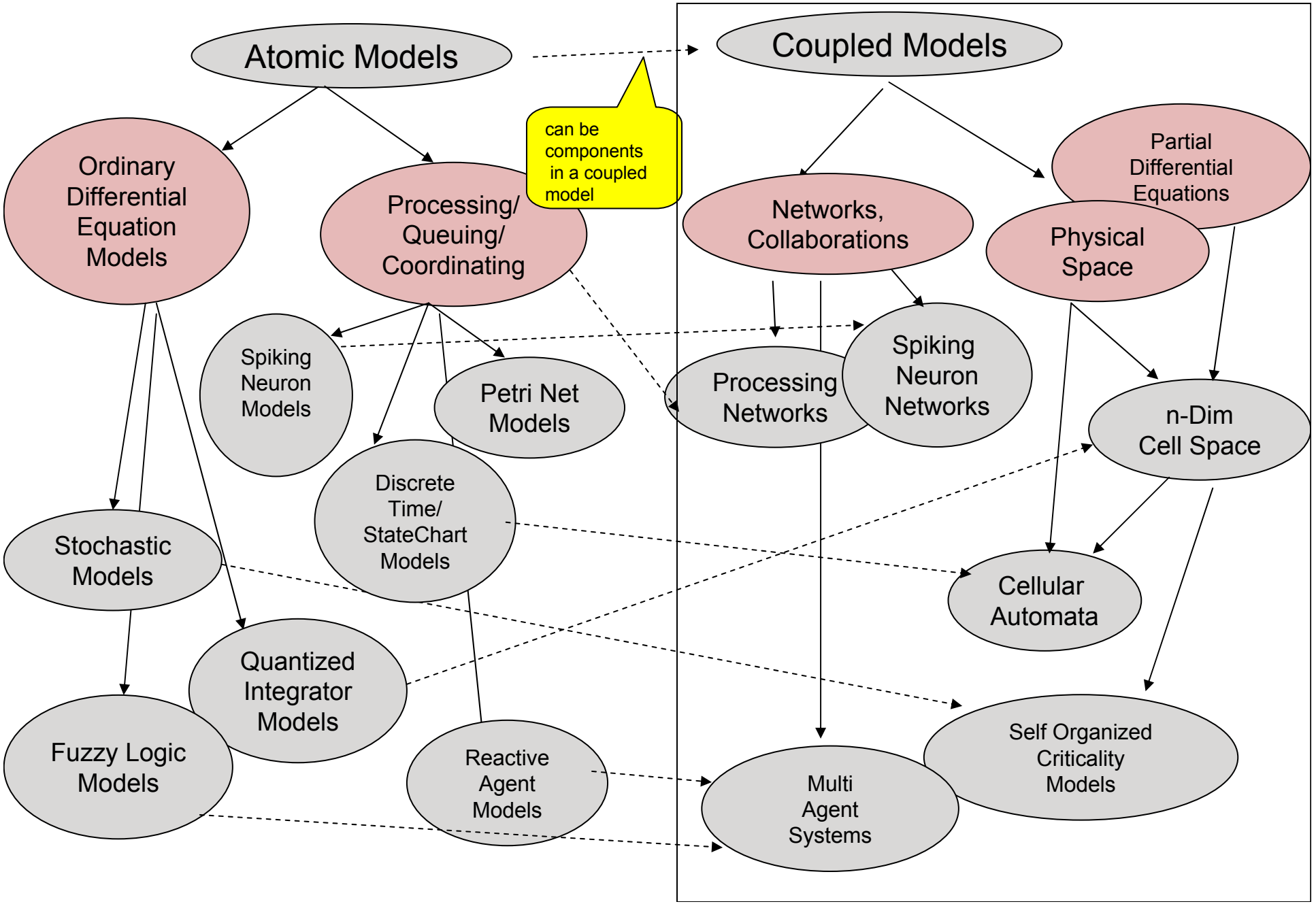
Theory

- Closure under coupling, universality, uniqueness, relation to other formalisms
- Hierarchical Model Construction supports complex systems
- Supporting the correctness of the algorithms and validation of the executing models

Application

- Models, Simulators and Experimental Frames are distinct entities with their own software representations.
- Precise and well-defined mathematical representation
- Models/Experiments are developed systematically for interoperability
- Repositories of models/experiments are created and maintained systematically (and existing components can be easily reused for constructing new models)
- Discrete-event basis improves performance (e.g. no need for have a global clock to control timing)

Some Types of Models Represented in DEVS



DEVS Research Groups/Environments

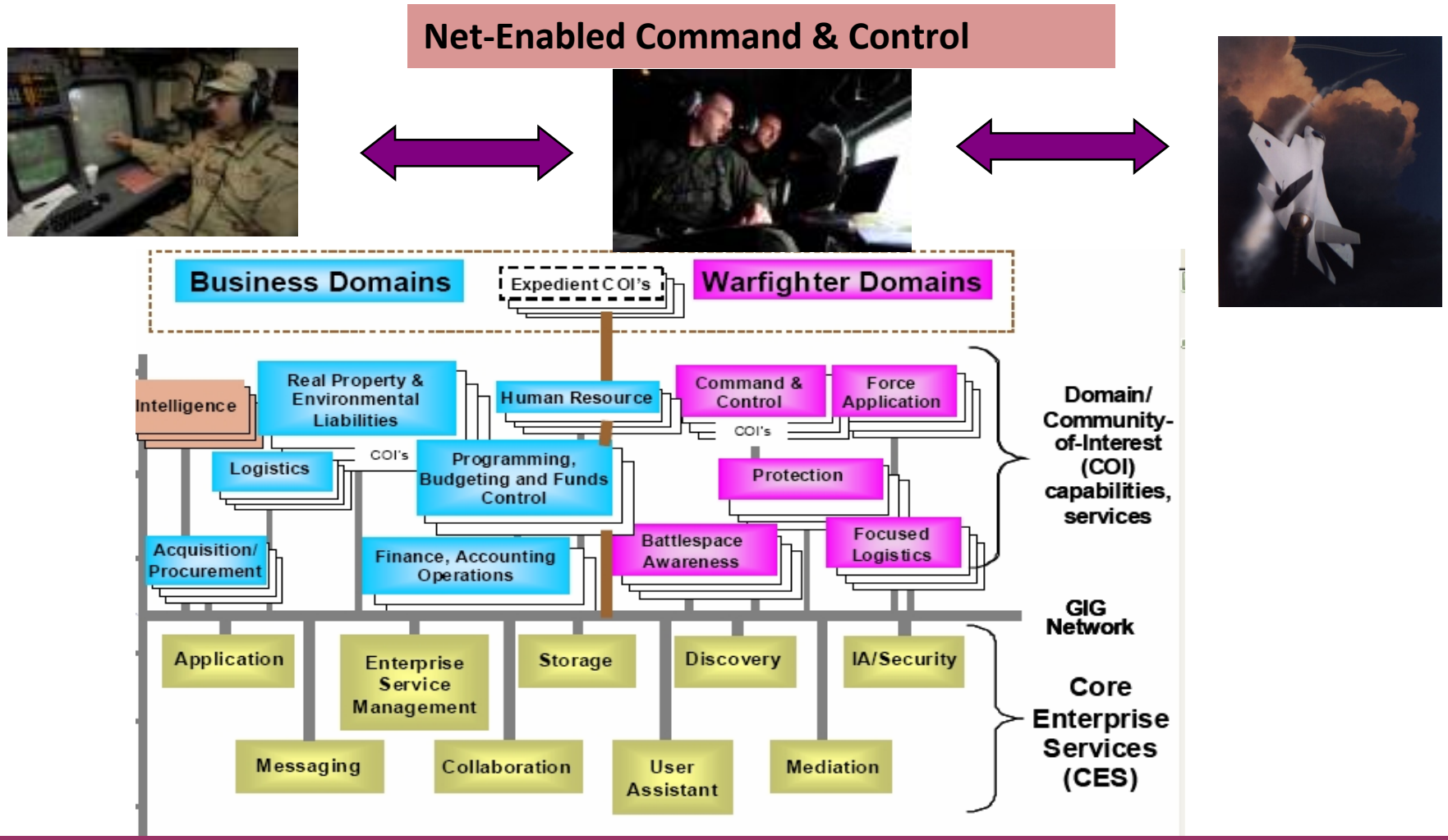
- Carleton's CD++,
- ADEVS (ORNL *),
- DEVS/C#,
- DEVS/HLA,
- DEVSJAVA (ACIMS - University of Arizona *),
- GALATEA (USB – Venezuela),
- LSIS (Aix-Marseille III – France *),
- JDEVS (Université de Corse - France), PyDEVS (McGill),
- PowerDEVS (University of Rosario, Argentina),
- SimBeams (University of Linz – Austria),
- VLE (Université du Litoral -France),
- SmallDEVS (Brno University of Technology, Czech Republic),
- James (University of Rostock,Germany)
- Portugal, Spain, and Russia;;;

- ***W**orkshop on **N**et-**C**entric **M**odeling & **S**imulation*
  March 6–7 2008 - Marseille, France
- http://osa.inria.fr/wiki/NCMS/NCMS

DEVS Adopters

- Joint Interoperability Test Command, USA
- Air Force, Navy / USA, South Korea
- Lockheed Martin Missile Systems
- Usinor – Sachem Expert Control
- Swedish Materials Command
- …

Global Information Grid /Service Oriented Architecture



NCES: Secure, agile, robust, dependable, interoperable data-sharing environment for DOD where warfighter, business, and intelligence users share knowledge on a global network. This, in turn, facilitates information superiority, accelerates decision-making, effective operations and net-centric transformation.

Service Oriented Architecture Basics

Content/Service
Catalogs/Registries

Search
find_xxx

WSDL

Post
save_xxx

(Bind)

Content/Service
Consumer

XML
Schema

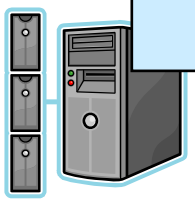
XML
Payload

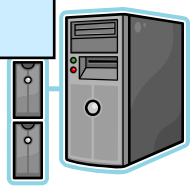
Content/Service
Provider

Client
Access (& Use)

SOAP

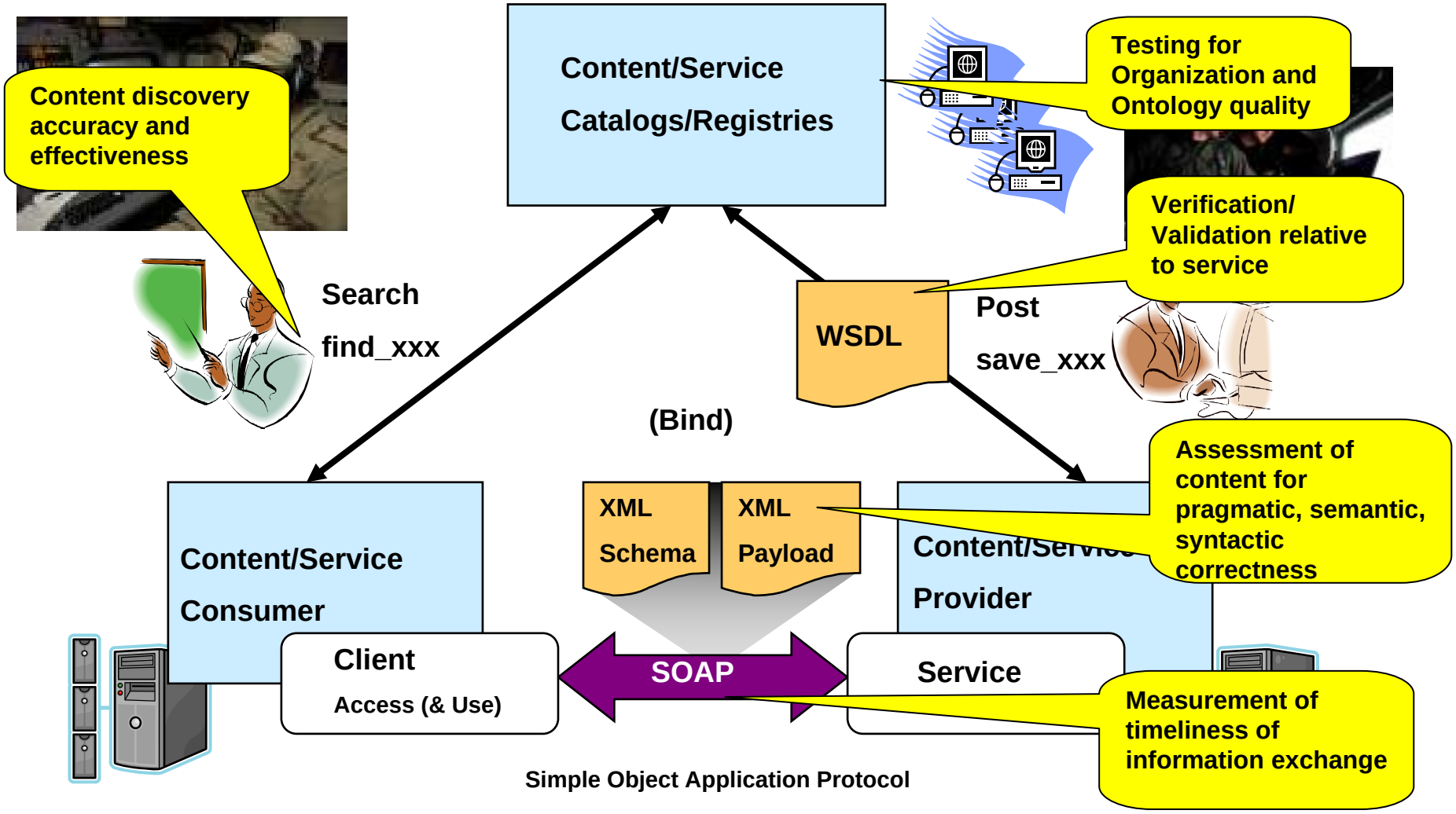
Service



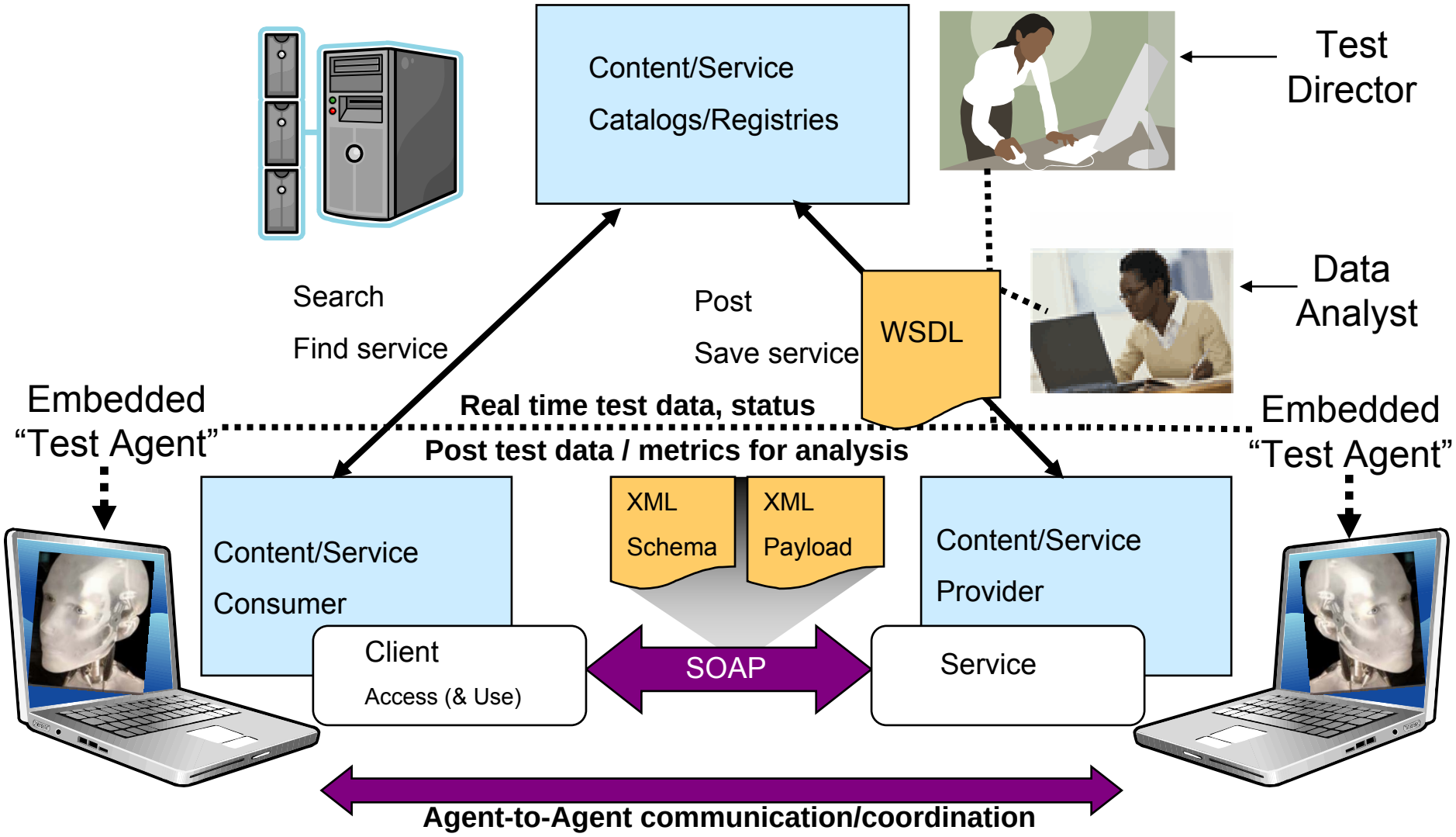


Simple Object Application Protocol

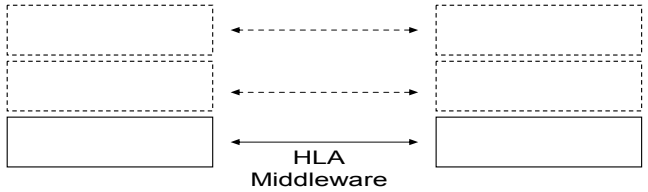
Requirements for Testing and Data Collection



Net-Centric Test Agent Capability (NTAC)

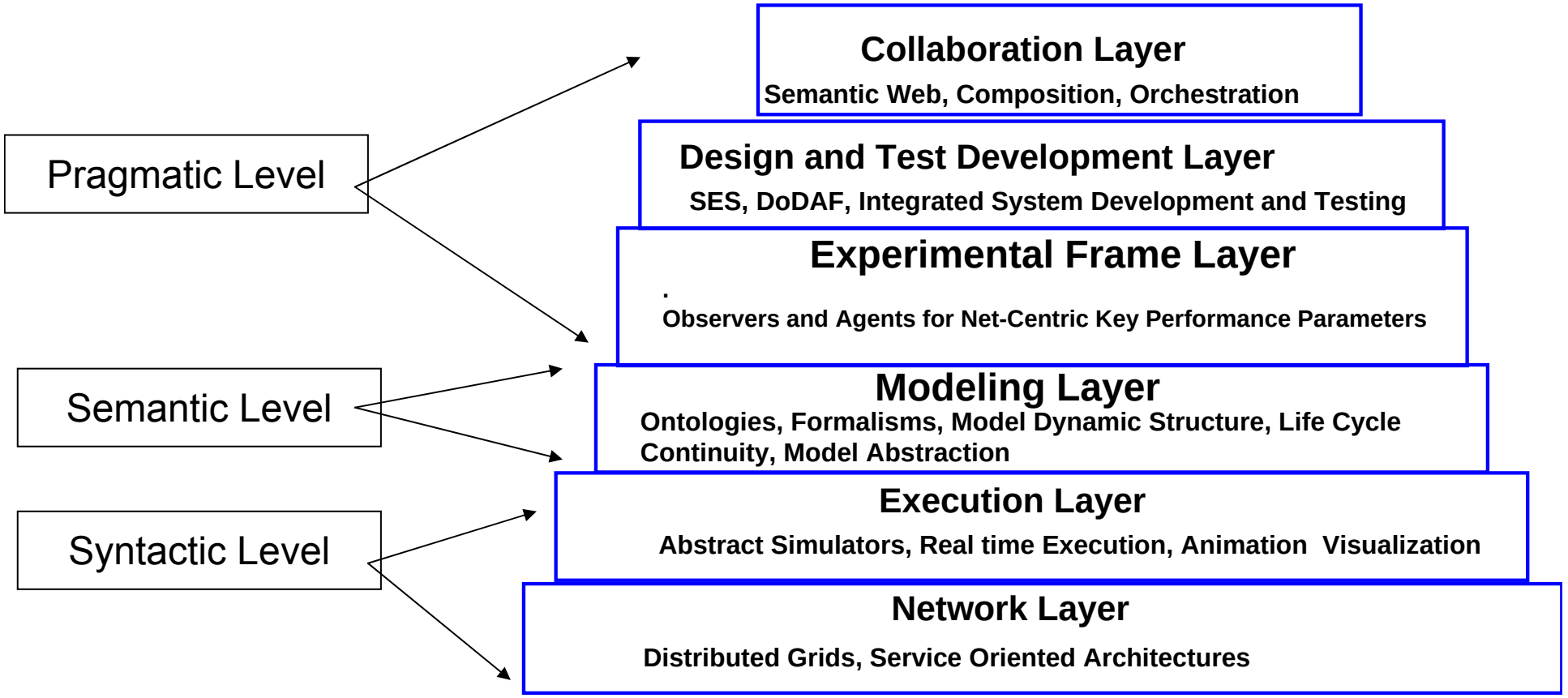


Levels of System of System Interoperability

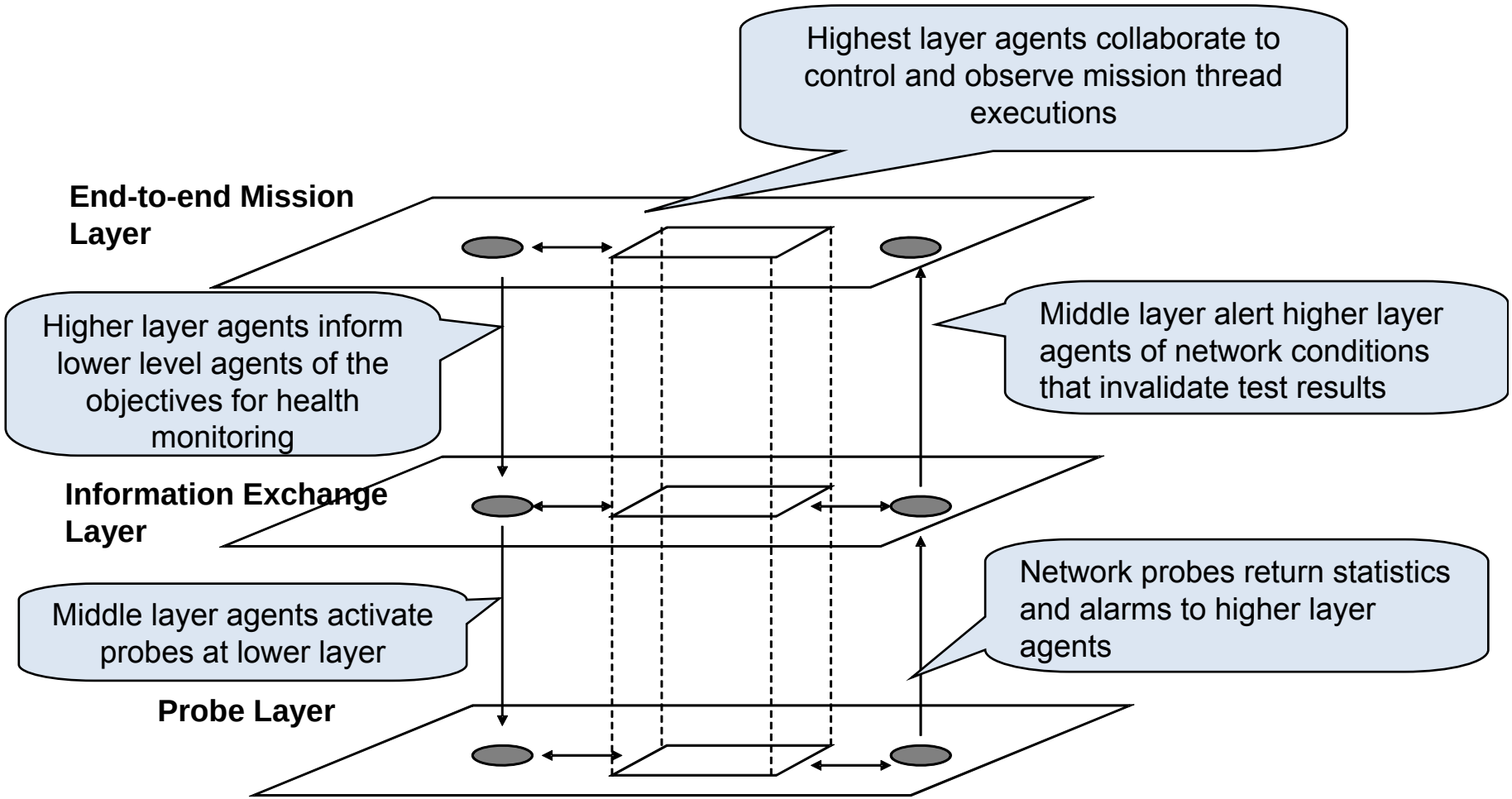


Linguistic Level	Interoperability Demonstrated if:	Example
Pragmatic – How information in message is used	The receiver reacts to the message in a manner that the sender intends	A commander's order is obeyed by the troops in the field as the commander intended. (This assumes semantic interoperability.)
Semantic – Shared understanding of meaning of messages	The receiver assigns the same meaning as the sender did to the message.	An order from a commander to multi-national participants in a coalition operation is understood in the same manner despite translation into different languages.
Syntactic – Common rules governing composition and transmitting of messages	The consumer is able to receive and parse the sender's message	A common network protocol (e.g., IPv4) ensures that all nodes on the network can send and receive data bit arrays while adhering to a prescribed format.

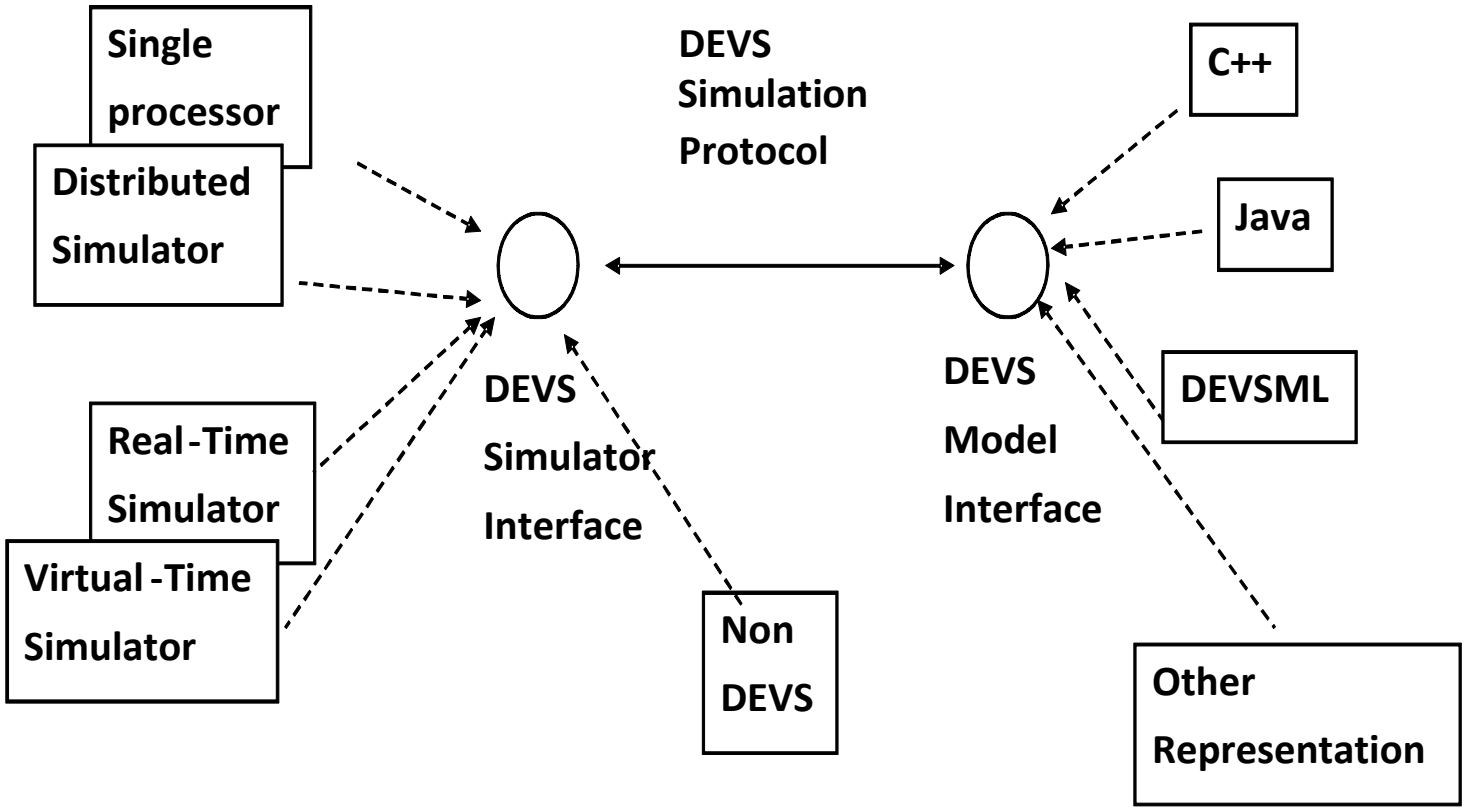
Mapping M&S Layers to Linguistic Levels



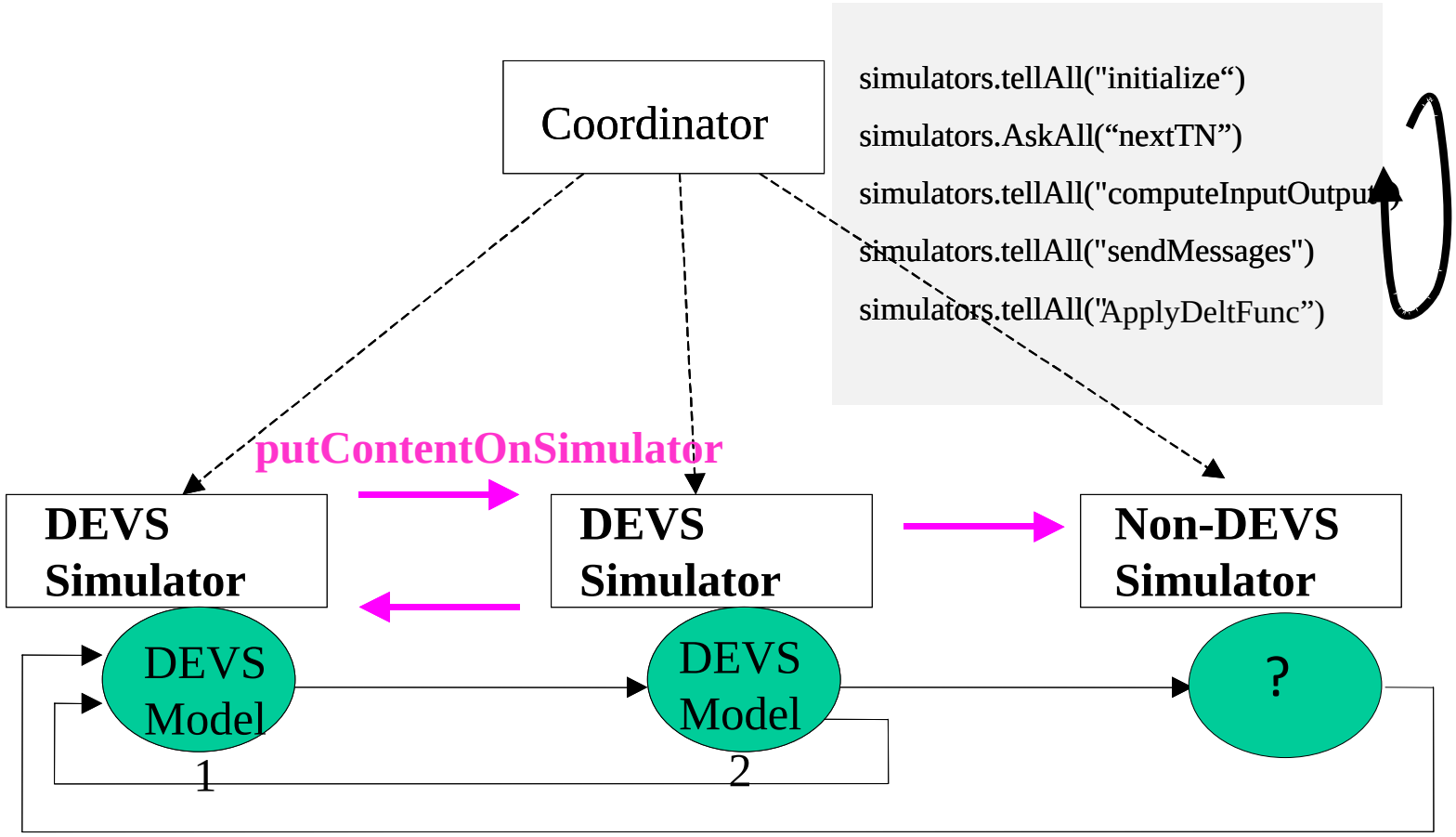
NTAC Agent-based Test Instrumentation Infrastructure supports simultaneous testing at multiple levels



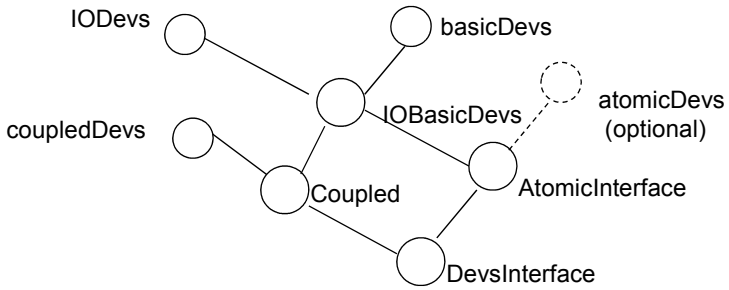
Concept of DEVS Standard

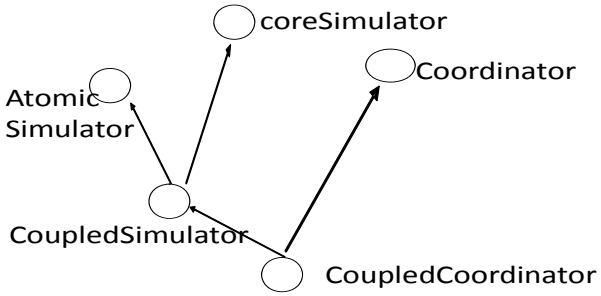


DEVS Simulation Protocol



DEVS Standard Interfaces






```
interface coreSimulatorInterface{
void setSimulators
                (Collection<CoreSimulatorInterface>);
void initialize();
Double nextTN();
void computeInputOutput(Double t);
void applyDeltFunc(Double t);
void putContentOnSimulator
      (CoreSimulatorInterface sim, ContentInterface c);
 void sendMessages();
}
```

Finite Deterministic DEVS : FD-DEVS

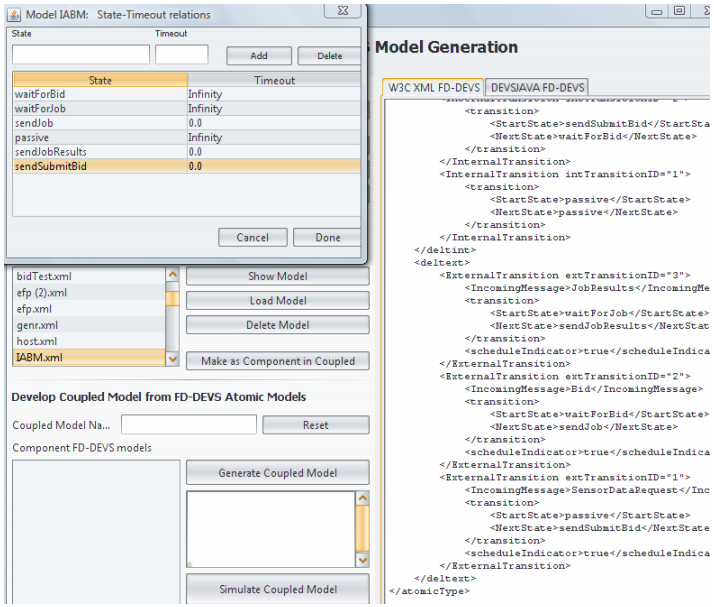
FDDEVS = <incomingMessageSet, outgoingMessageSet, StateSet, TimeAdvanceTable, InternalTransitionTable, ExternalTransitionTable,OutputTable>
where
incomingMessageSet, outgoingMessageSet, StateSet are finite sets
TimeAdvanceTable: StateSet $\rightarrow R_{0,\infty}^{+}$ (the positive reals with zero and infinity)
InternalTransitionTable: StateSet $\rightarrow$ StateSet
ExternalTransitionTable: StateSet $\times$ incomingMessageSet $\rightarrow$ StateSet,
OutputTable: StateSet $\rightarrow$ the set of subsets of outgoingMsgSet

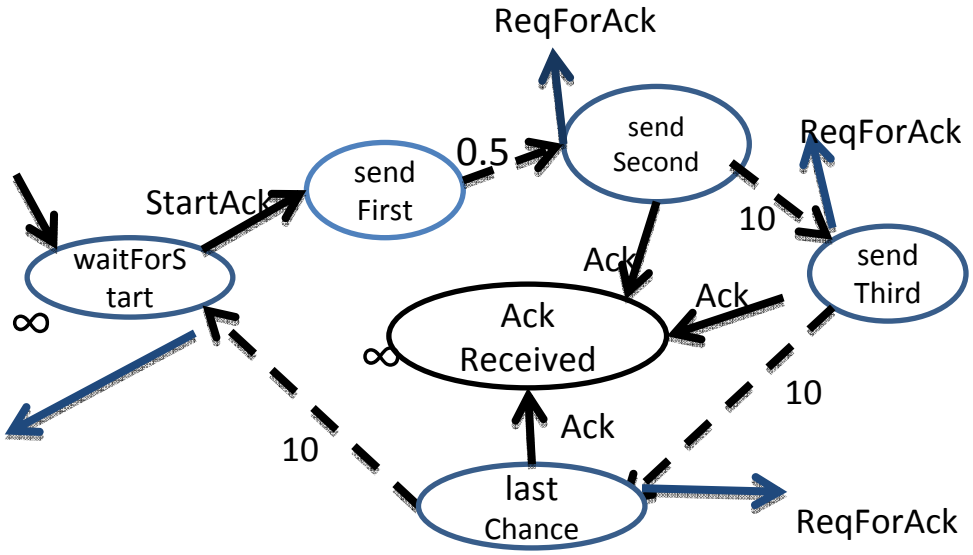
Natural Language For FDDEVS
- *to start hold in PHASE for time SIGMA*
- *when in PHASE and receive MSG go to PHASE' <eventually>*
- *hold in PHASE for time SIGMA*
- *after PHASE then output MSG*
- *from PHASE go to PHASE'*

Semantics defined by mapping into DEVS

FD-DEVS

- The “right” abstraction of DEVS – retains important timing properties
- Amenable to analysis
- Supports automation
- Maps to DEVSJAVA
- Supplies a skelton that can be extended to full DEVS
- Simple XML expression





Example of Natural Language Spec:

1.to start passivate in waitForStart

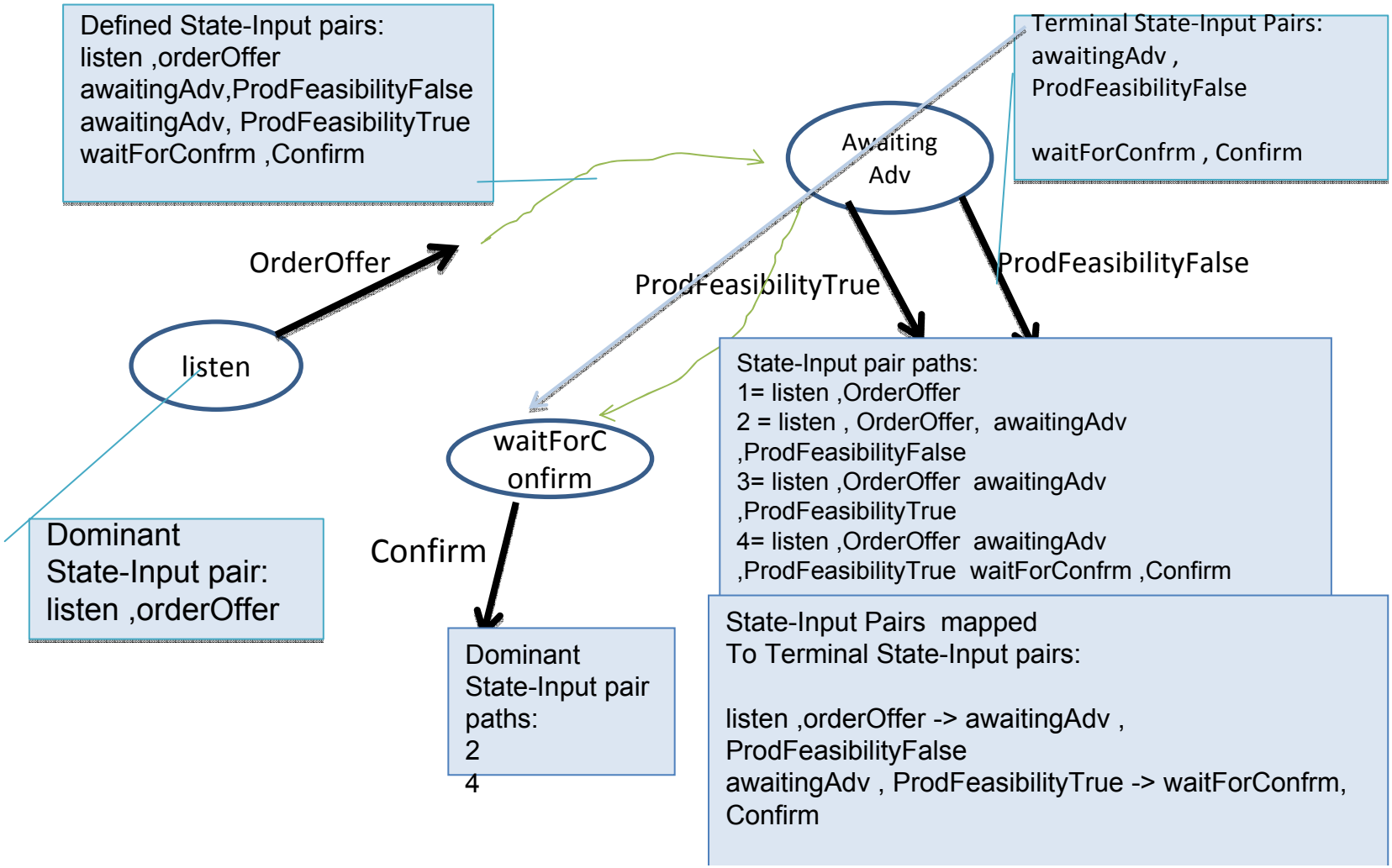
2.when in waitForStart and receive StartAck go to sendFirst

3.hold in sendFirst for time 0.5 then output ReqForAck and go to sendSecond

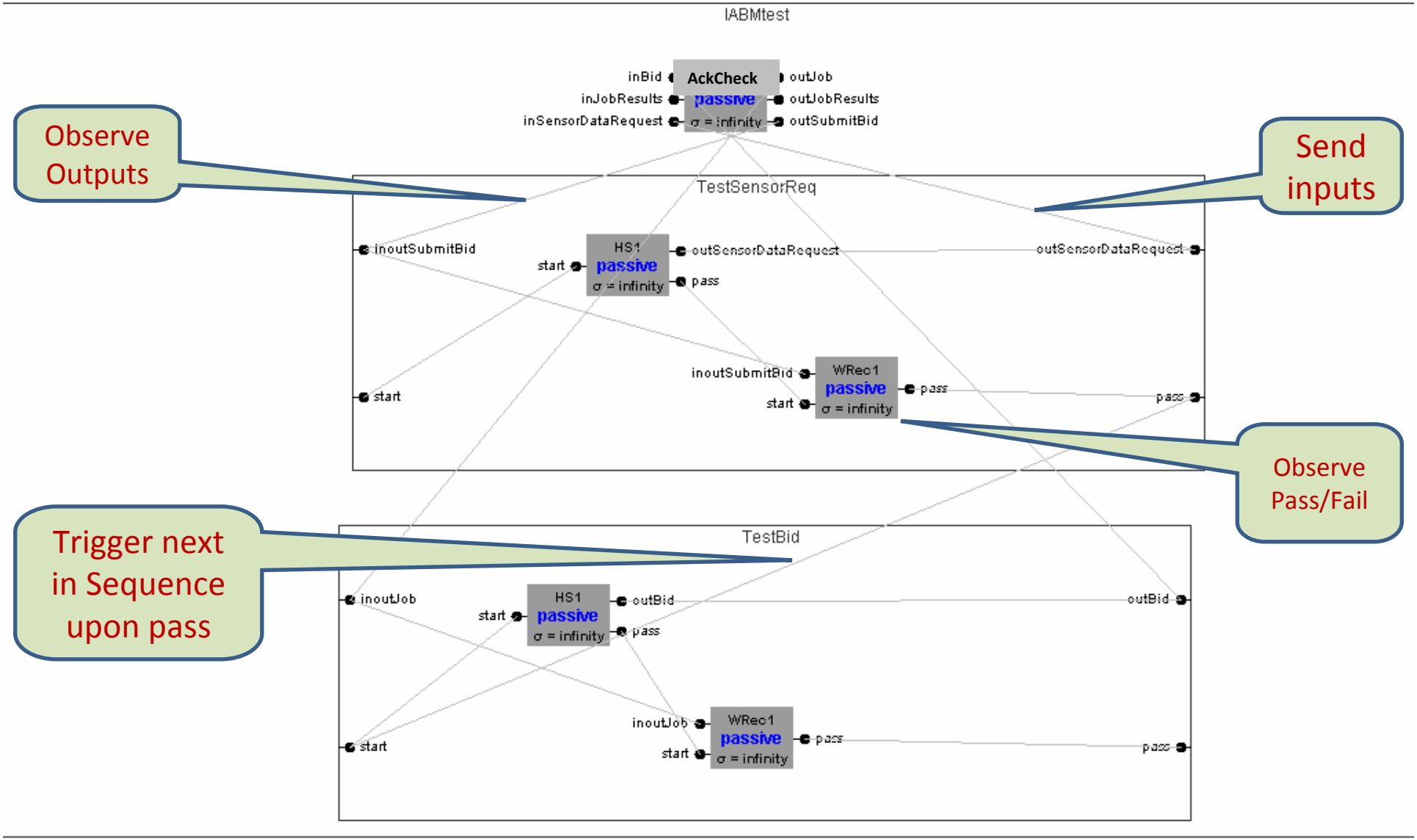
4.hold in ackReceived for time Infinity

5….

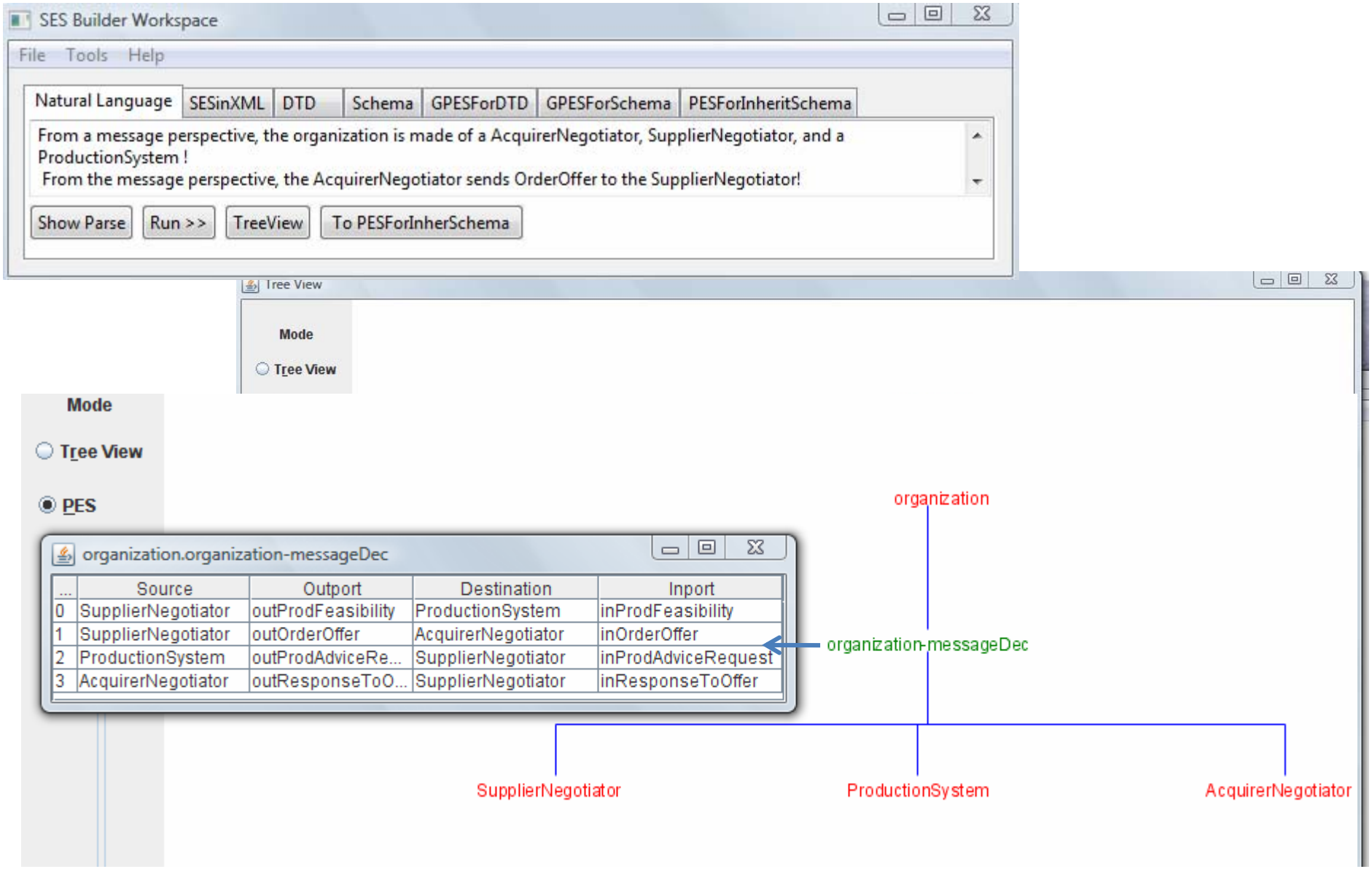
Automated Analysis: Based on State Input Pairs Enables Automated Test Model Generation



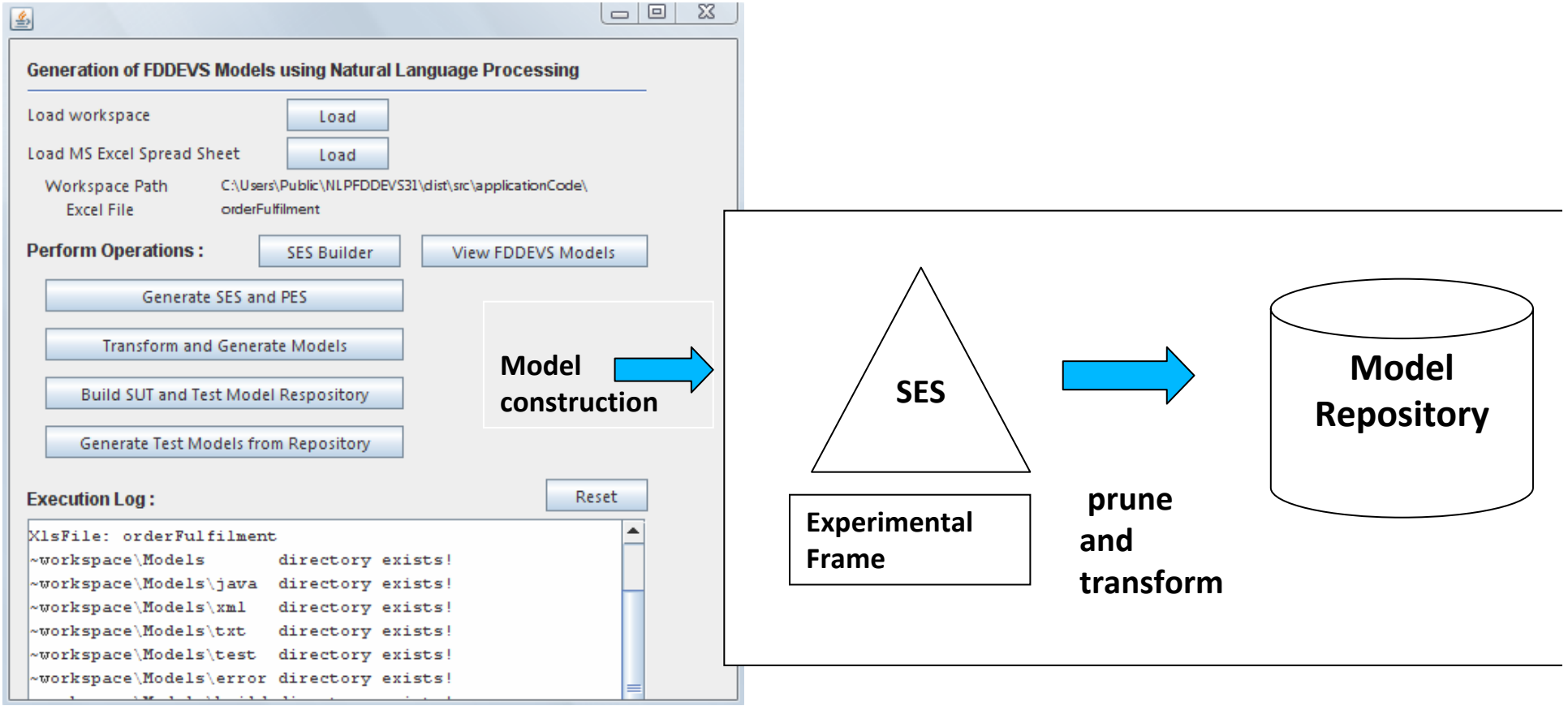
Generated Test Models in DEVSJAVA SimView



System Entity Structure (SES) : SESBuilder



System Entity Structure/Model Base Repository: Support Automated DEVS Generation and Reuse



Overall Processing

A	B	C	D	E	F	G
(U) DSRL ID (UNCLASSIFIED)	(U) Requirement	(U) Requirement Description (UNCLASSIFIED)	SESMicroRepresentation	FDDEVSRepresentation	Comments and Questions	Standards
1306		The IABM shall be capable of inhibiting (selectable via Global Static C2 Rule Set) automatic correlation of a composite and TDL remote track where either track has the strength field set to a value greater than 1.,	From the mult perspective, tracks is made of more than one track ! A track can be composite or TDLremote in origin ! A track has a strengthField ! The range of track's strengthField is double with values[0,1] !	IABM1306: to start passivate in passive ! IABM1306: when in passive and receive EnableAutoCor go to AutoCorMode ! IABM1306: passivate in AutoCorMode !	StrengthFieldTestPositive =	Global Static C2 Rule Set

Augmented Test Models

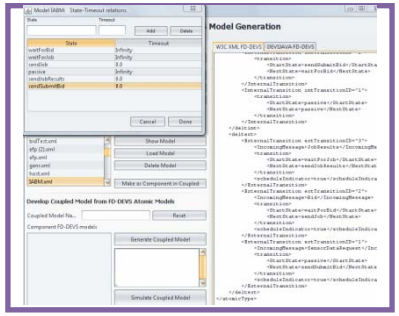
Automated

manual

FDDEVS atomic model

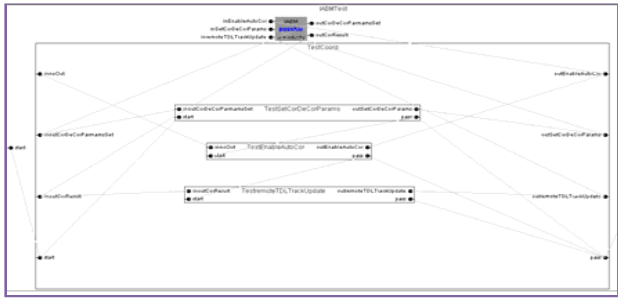
SES

SES/XML Of DSR texts

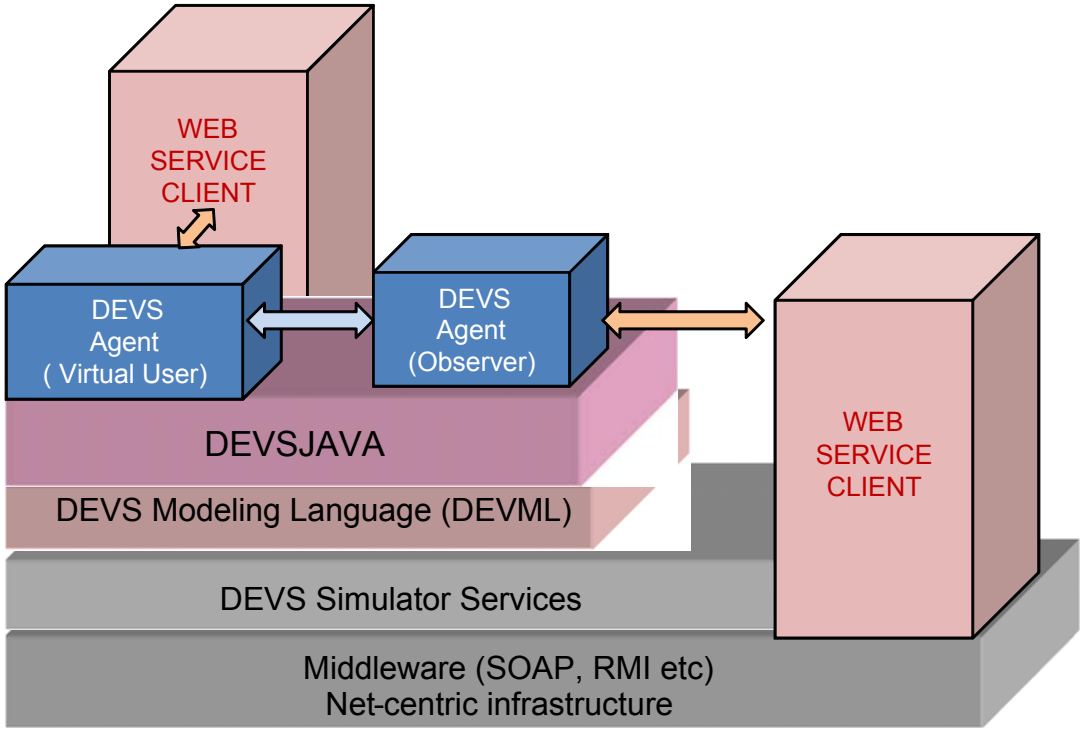


FDDEVS test models

Pruned ES



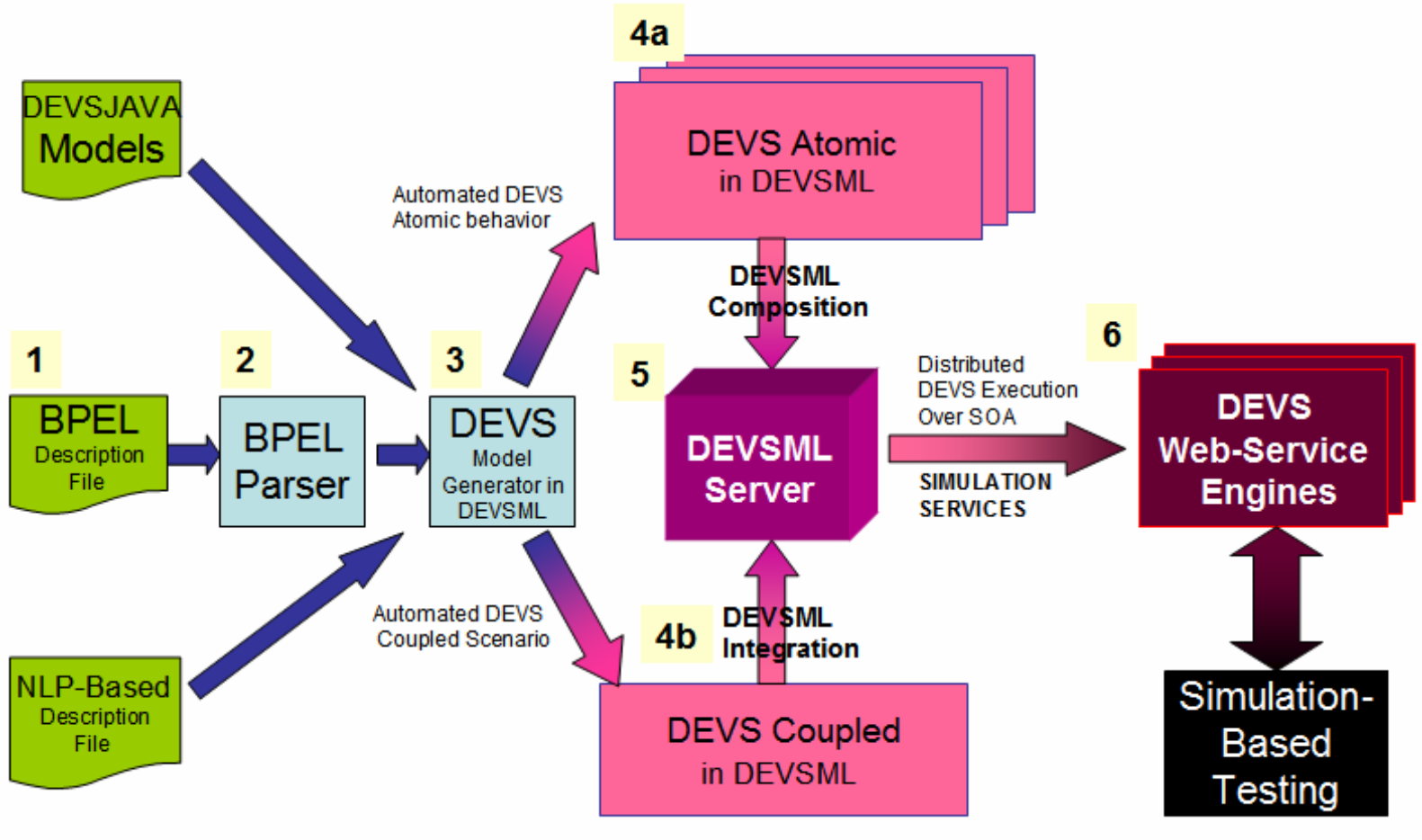
DEVS/SOA Infrastructure: Supports Deployment and Execution of DEVS Models on the Web



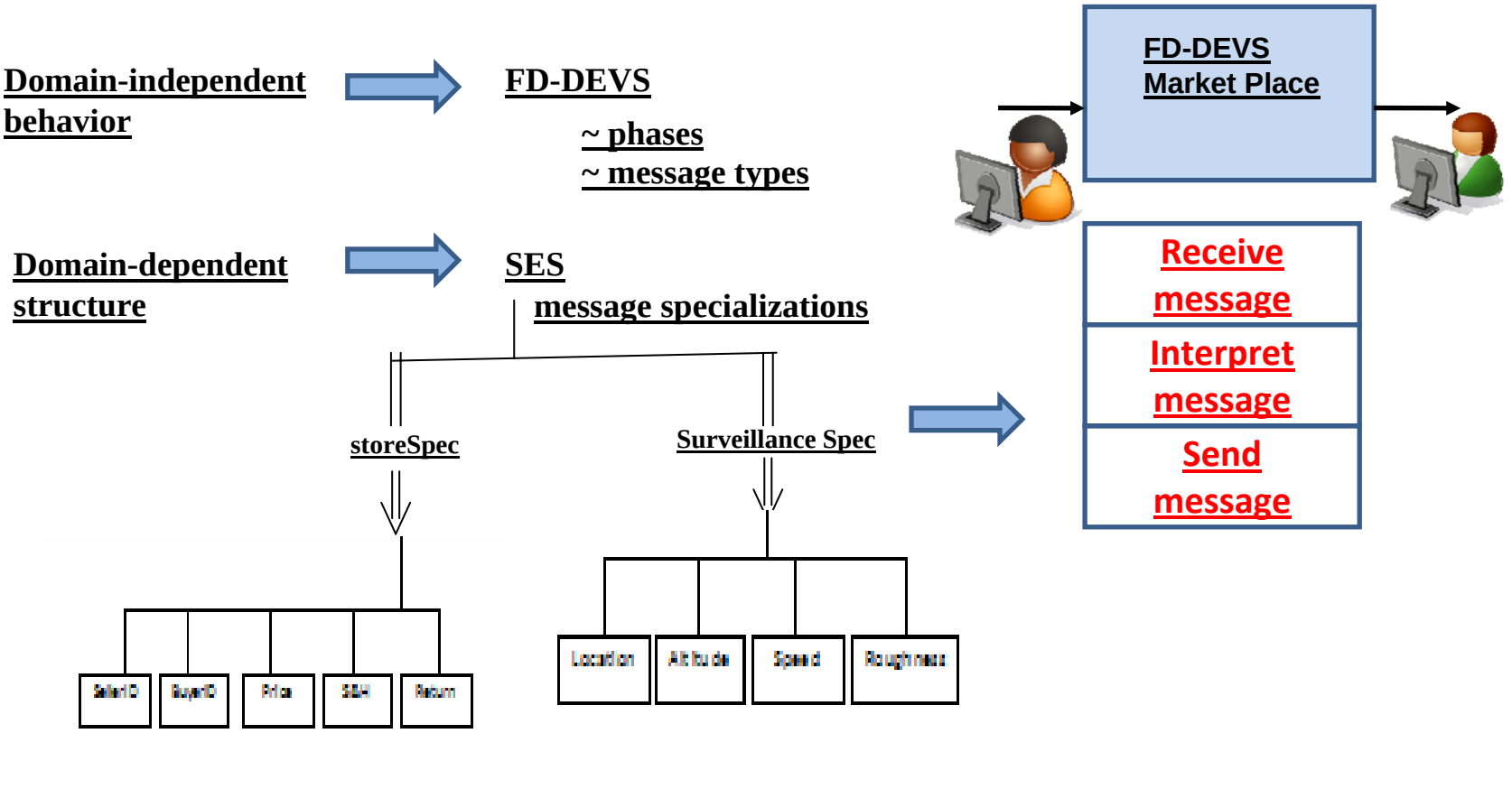
- **Service Oriented Architecture (SOA) consists of various W3C standards**
- **Client server framework**
- **XML Message encapsulated in SOAP wrapper**
- **Machine-to-machine interoperable interaction over the network based on WSDL interface descriptions**

Run Example

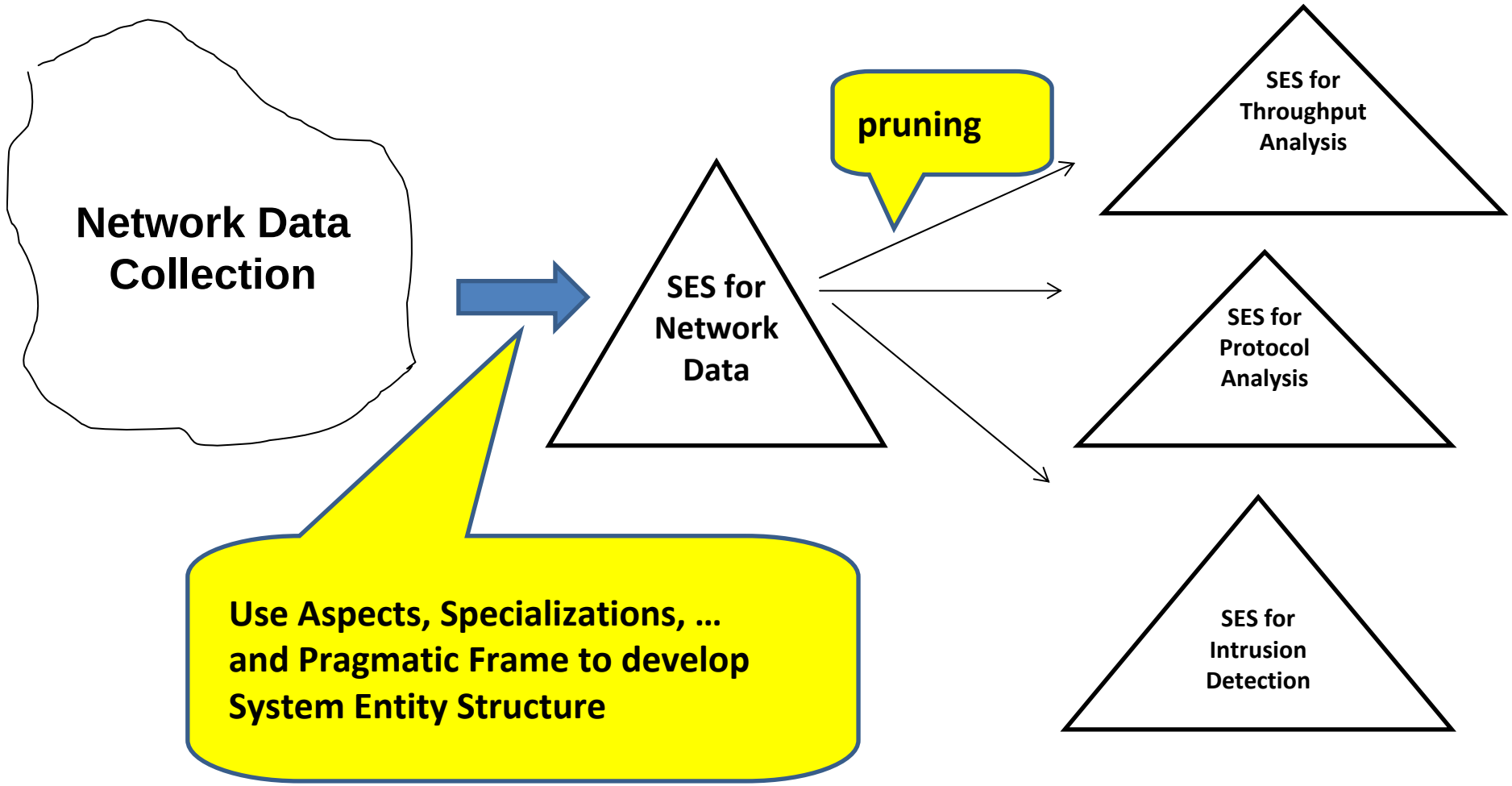
Deploying Models: DEVSML and DEVS/SOA



Automated Negotiation Support in Multi-Agent Web Environments



Analysis-Based Network Data Extraction



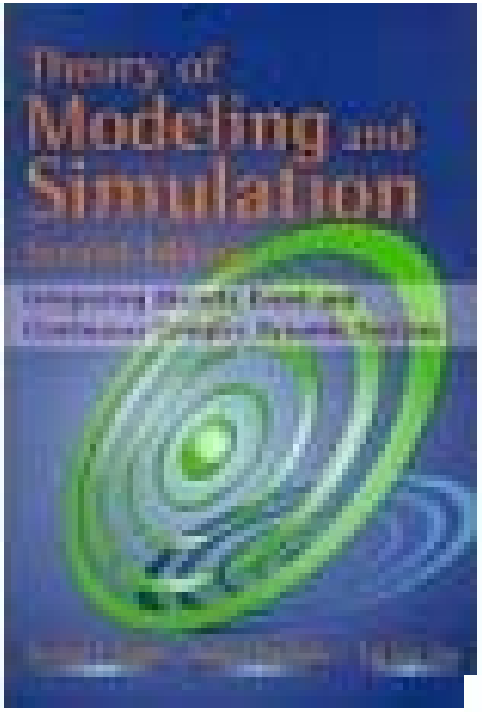
Applications

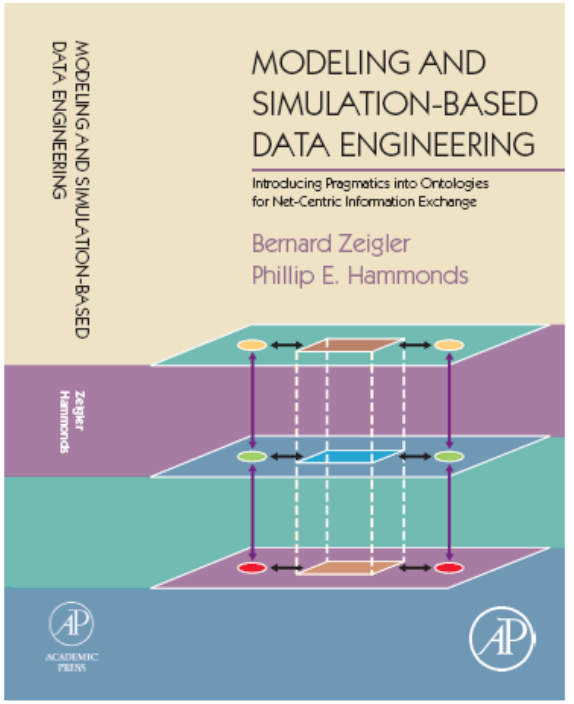
- Natural language capture of high level information technology systems requirements
- Automated generation of FDDEVS kernel DEVSJAVA/C++ models for distributed real-time net-centric IT systems testing
- Development of web service workflows using DEVS/SOA
- Network Traffic data capture, focused extraction, and model generation for exercising IT systems e.g., intrusion detection

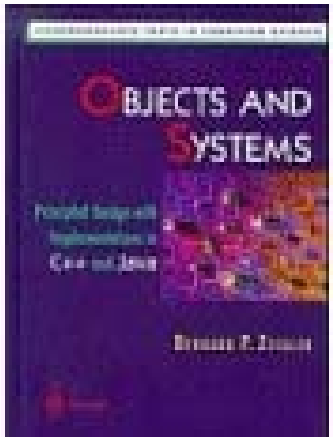
Conclusions

- DEVS and SES provide a framework based on Systems Theory for Web-Centric M&S environments
- Supports integrated development and testing
- DEVS standard supports sharable models and repository reuse on the Service Oriented Architecture
- Provides a basis for achieving higher levels of interoperability – can work with HLA or not: DEVS/SOA provides a SOA implementation independent of HLA
- The framework supports development of generic tools which in turn support a wide array of web service domain specific specializations and applications

Books and Web Links







devsworld.org

acims.arizona.edu

Rtsync.com