

28 November 2012

Semantic Modelling of Dynamic, Multi-Stakeholder Systems

Martin Hall-May, IT Innovation Centre, University of Southampton
2nd SESAR Innovation Days, Braunschweig, Germany

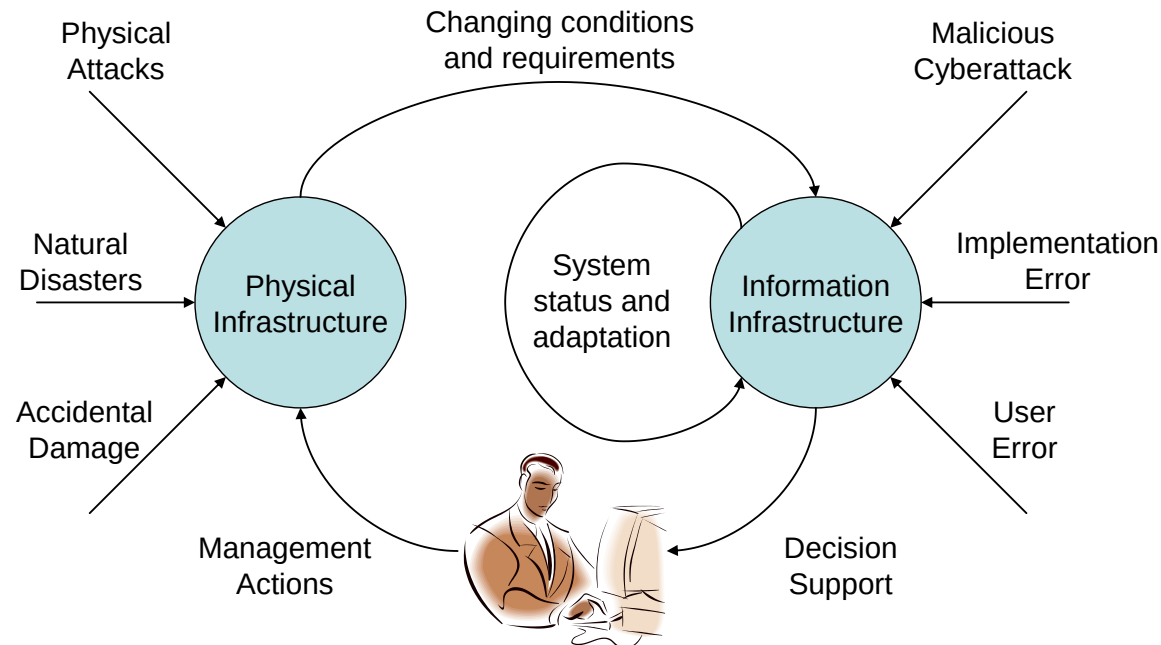


SERSCIS has
received EC
Research Funding

Semantically Enhanced Resilient and Secure Critical Infrastructure Services

Copyright © 2012 University of Southampton IT Innovation Centre and other Members of the SERSCIS Consortium

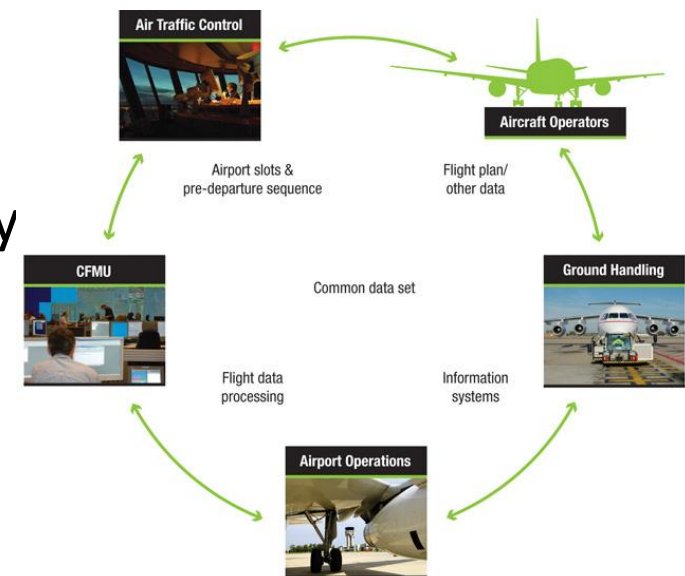
SERSCIS: ICT and Security



- Critical infrastructure ICT is increasingly interconnected
 - information sharing → greater operational efficiency, but also reduced slack and flexibility
 - interconnections → new risks from ICT failure cascade effects
 - overall → more vulnerable to natural, accidental or malicious disruption
- SERSCIS approach: use agile SOA to offset these threats
 - adapt ICT components and networks to meet changing security needs
 - adapt ICT connections to prevent cascades and contain security threats

SERSCIS Objectives

- To exploit agile service oriented architectures and semantic models and reasoning technology
 - to dynamically compose and manage ICT inter-connections related to critical infrastructure
 - to monitor and manage ICT components and connections against well-defined dependability criteria
 - to support human designers and operators to address risks especially by ICT inter-connections
- To validate this approach in case studies from the air traffic sector
 - Airport Collaborative Decision Making (A-CDM)

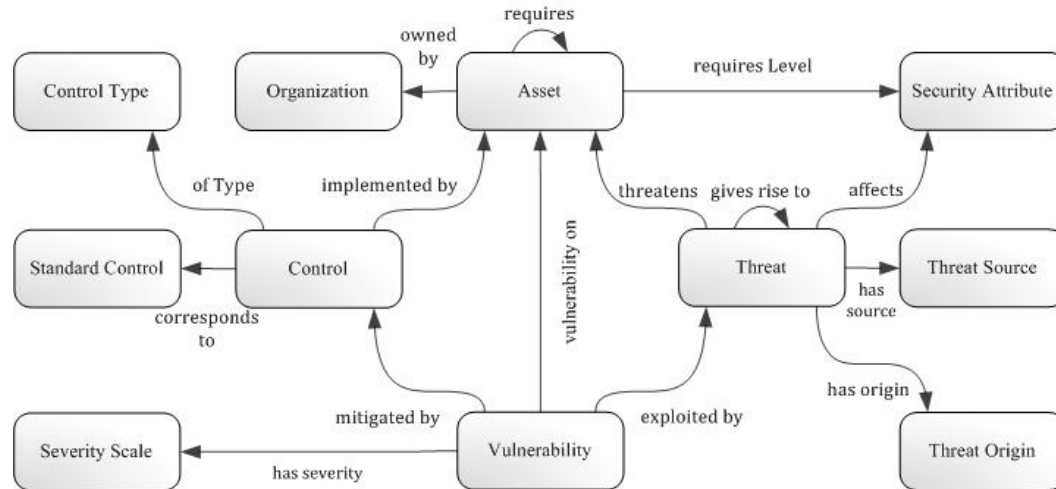


System Modelling Requirements

- To model dynamically composed, multi-stakeholder systems
- To create models at design time, before the actual composition is known
- To help the designer identify system threats, using knowledge from security experts
- To automate any modelling of the run-time system so it can be done 'on the fly'



System Modelling State of the Art



- Security Ontology from SBA
 - includes asset, control, threat and vulnerability classes
 - populated with instances from ISO 27001, etc.
- Can't capture dynamic systems at design time
- Ratio of asserted to inferred facts is quite high

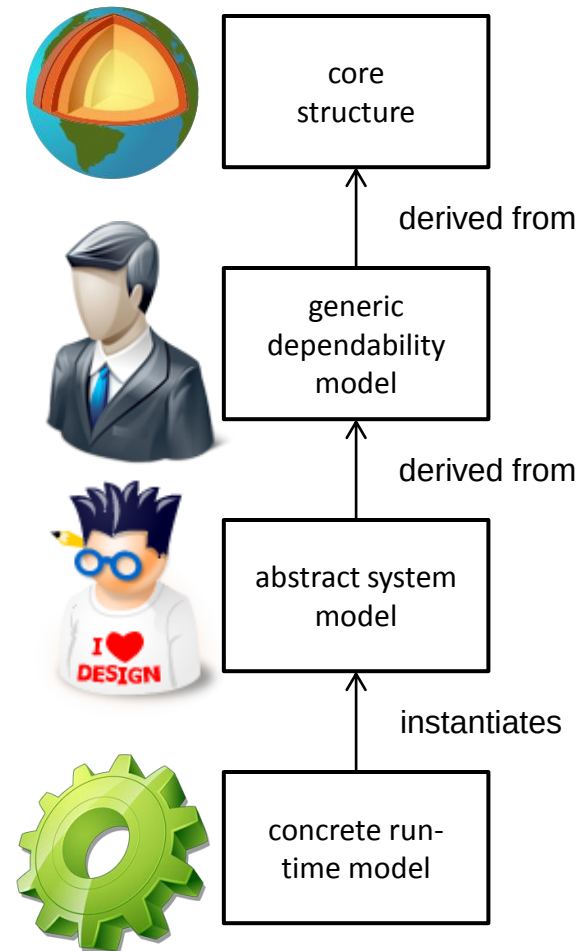
SERSCIS Technical Approach

- Don't start from a model of system topology as this isn't known at design time and changes at run time
- Start from a model of generic security issues
 - system assets, threats and controls
 - system behaviour and threat status
 - threat target, action and consequence
- Base model on current best practice
 - ISO27005, COBIT/ITIL, IETF RFC 2828
- Use OWL to benefit from existing semantic tools

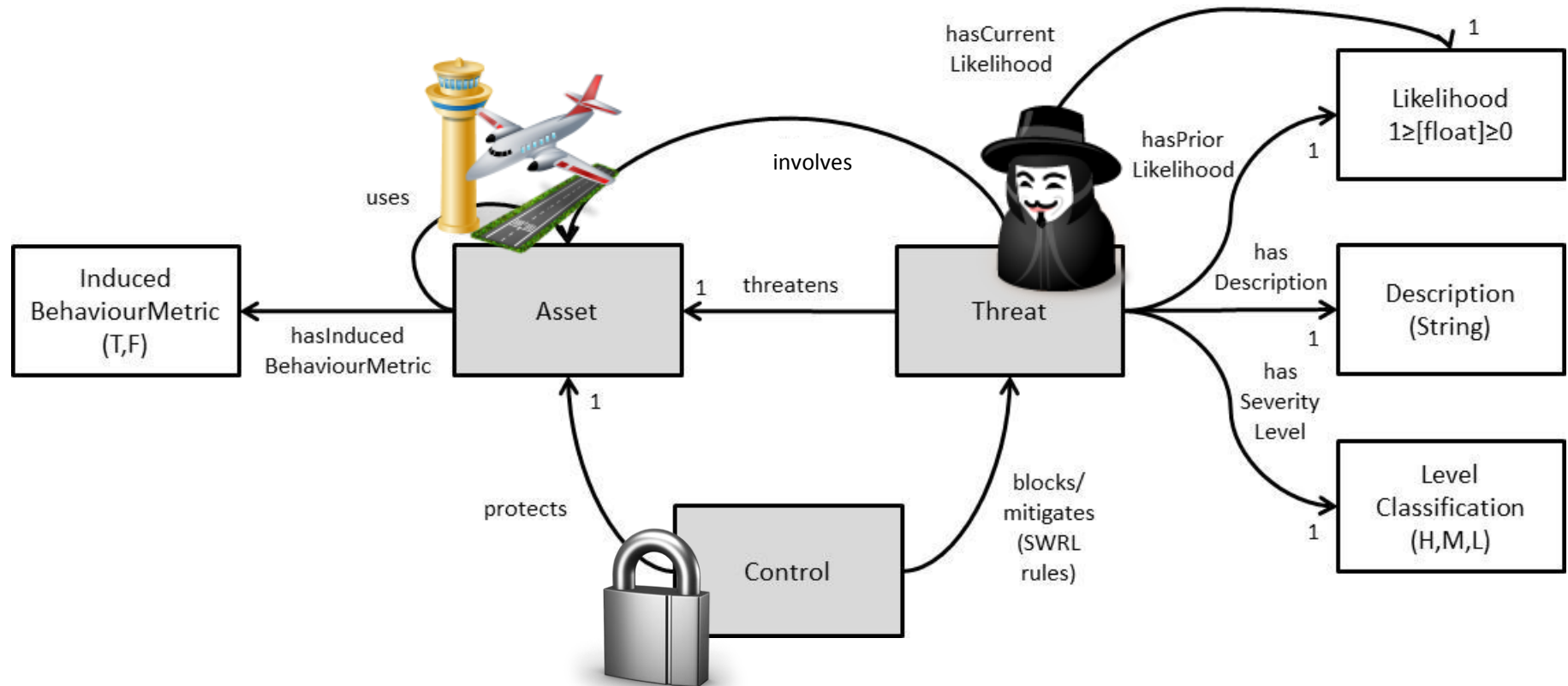


SERSCIS Modelling Abstractions

- Core structure similar to SBA ontology but simplified and 100% class based
- Dependability model classes capturing system-independent assets, threats and controls for multi-stakeholder systems
 - maintained by security experts
- Abstract system model subclasses capturing system-specific assets and threats for a specific application
 - maintained by system designers
- Concrete system model instances capturing current system composition and status (assets, threats and controls)
 - auto-generated by SERSCIS run-time framework

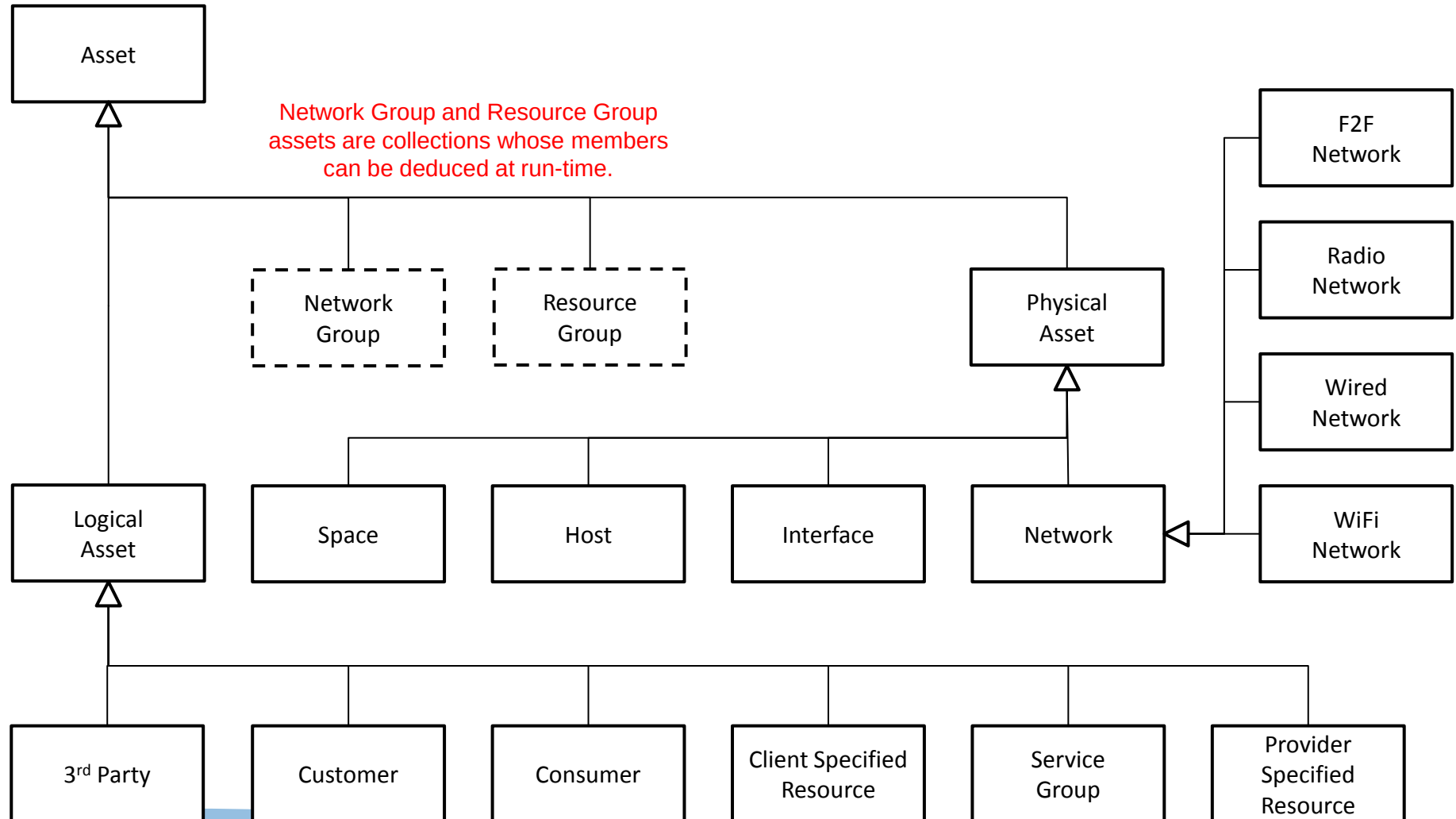


Core Model

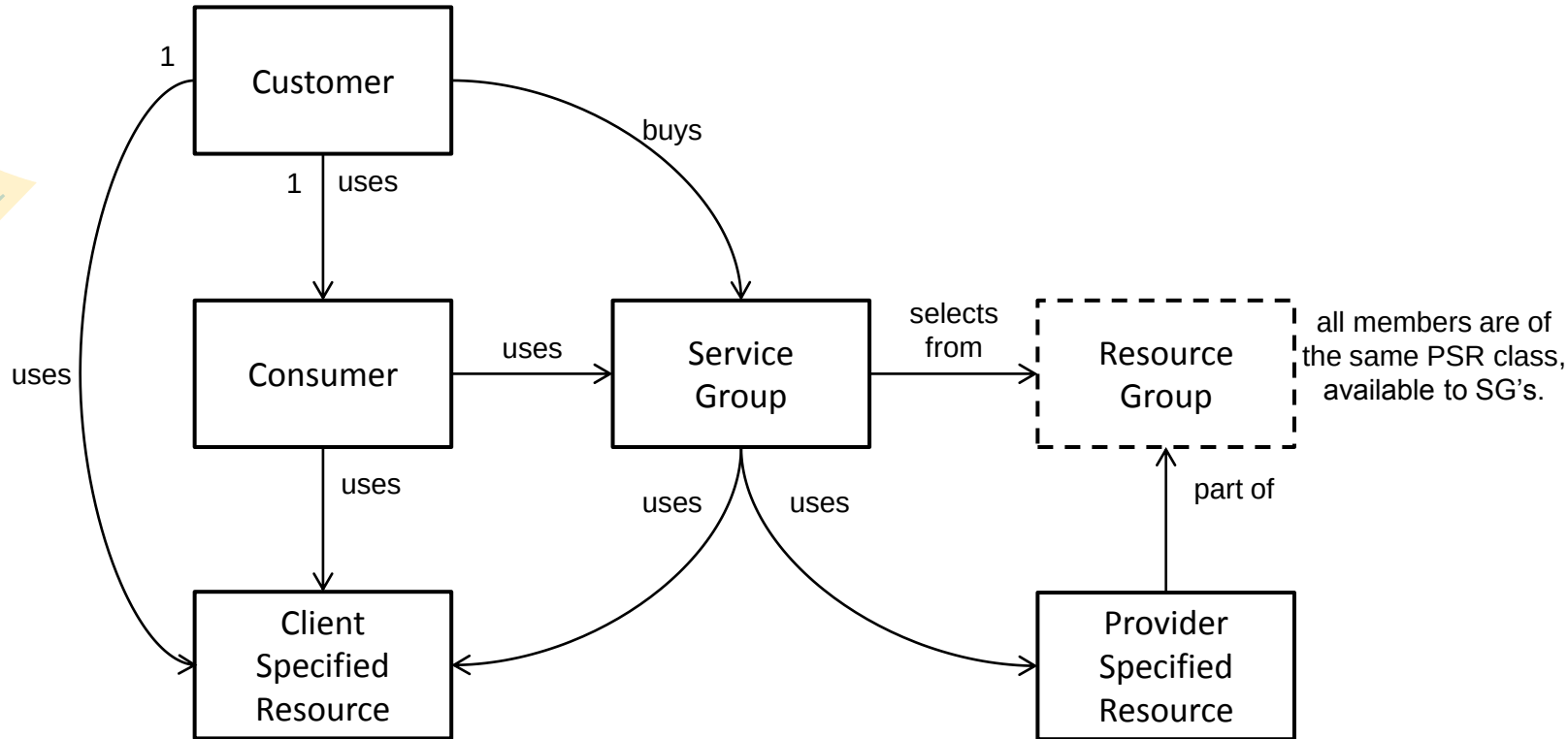


- The basic structure, determines what reasoning can be used
- Behaviours expressed using the existing SERSCIS metrics ontology
- Core model is created and maintained by SERSCIS technology developers

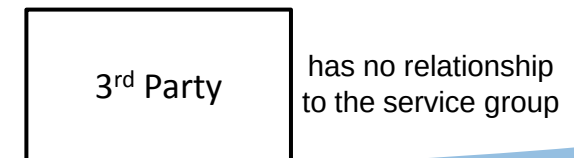
Dependability Model: Asset Classes



Dependability Model: Logical Assets

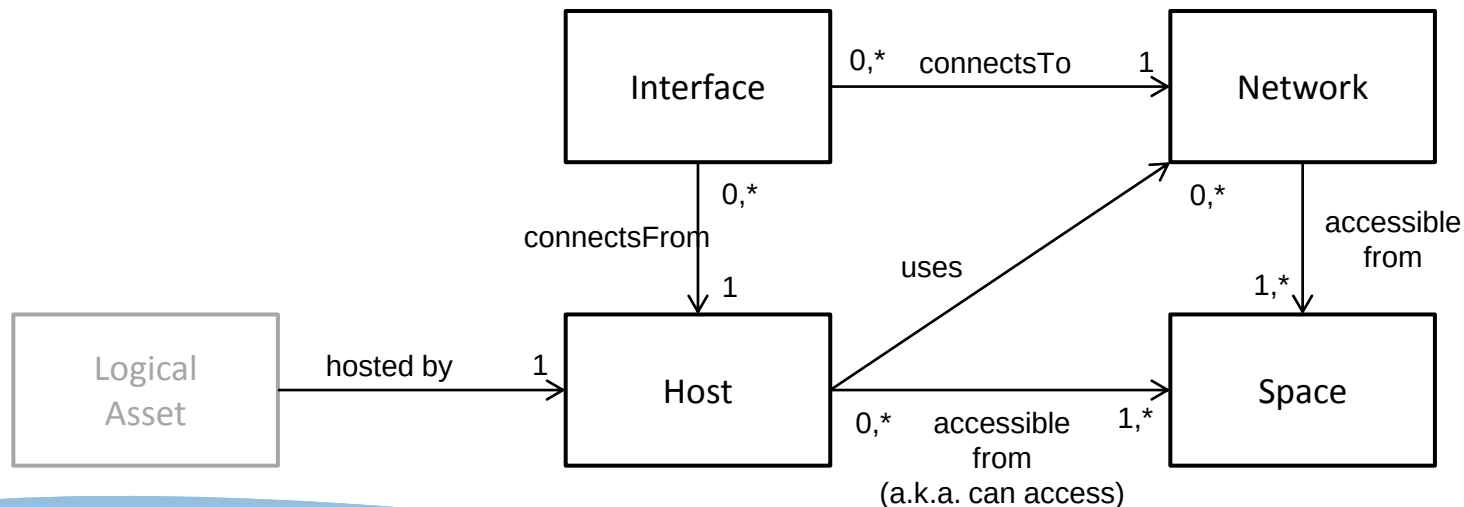
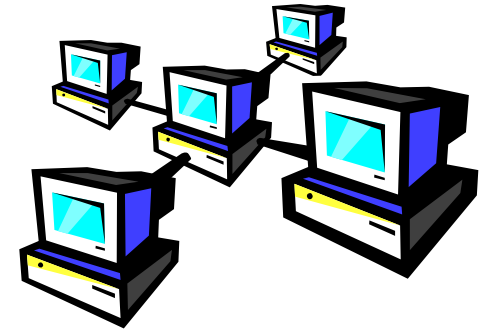


A resource group instance exists for each type of PSR that can be used by service groups hosted by the service provider.

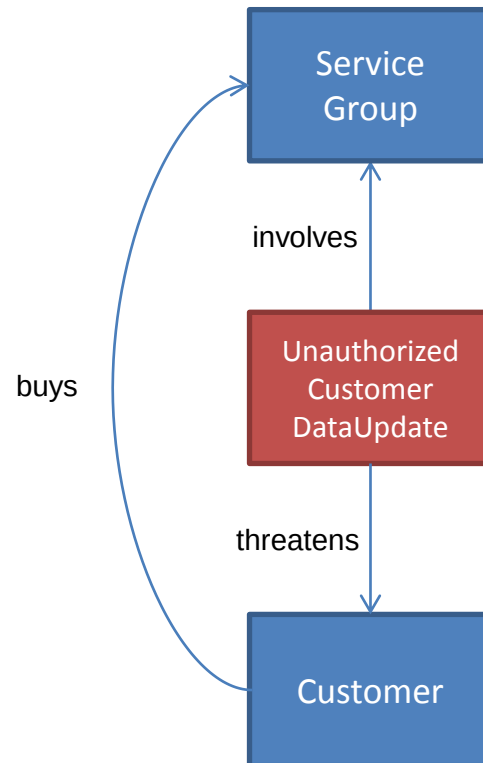


Dependability Model: Physical Assets

- Logical assets are hosted
- Hosts communicate over networks
 - Interfaces are host/network pairs
- Hosts and networks are accessible from spaces

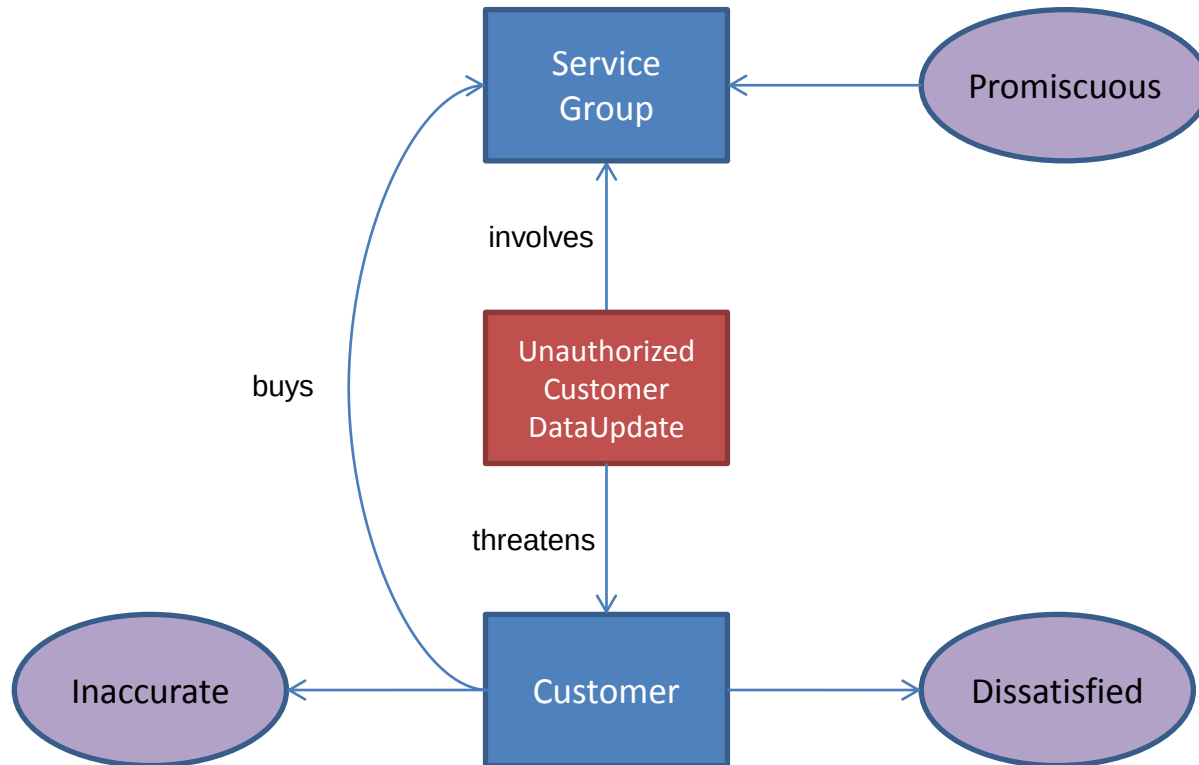


Dependability Model: Example Threat



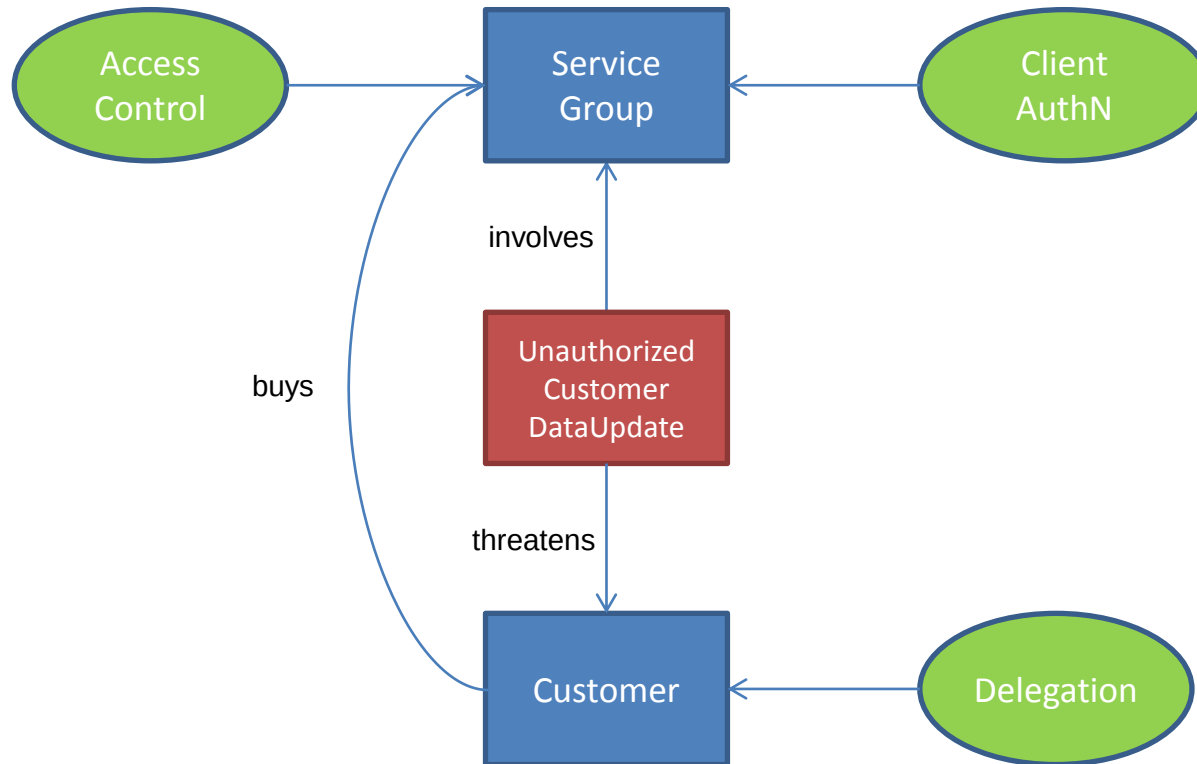
Attacker threatens the Customer by accessing the Service in the context of their SLA without authorization.

Dependability Model: Example Threat



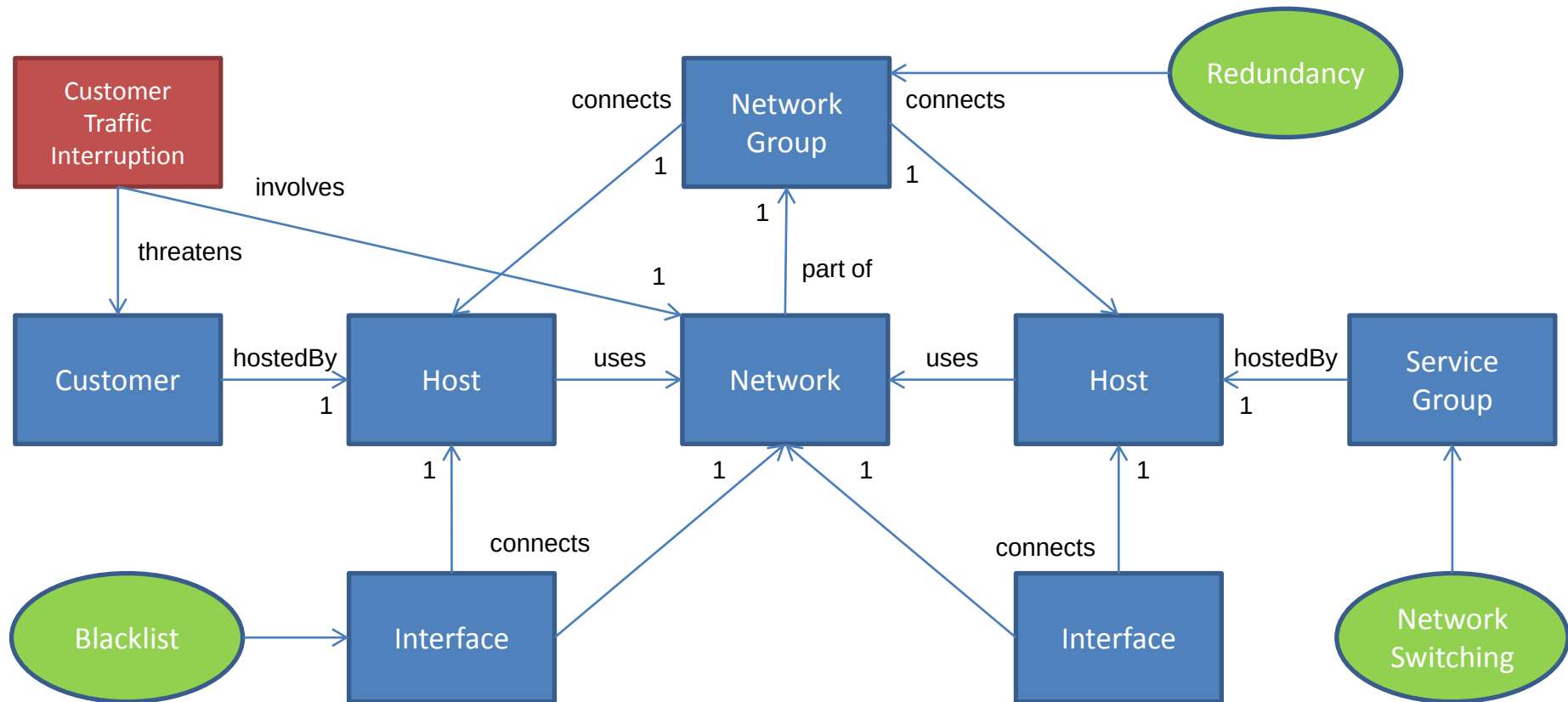
Threat consequence is that the customer's data is changed and becomes inaccurate, and the customer is unhappy. The attack action involves exploiting service promiscuity (i.e. faulty or missing access control).

Dependability Model: Example Threat



Fix by authenticating client credentials and enforcing access control at the service, and by delegating rights at the customer (by issuing tokens or managing the access control policy).

A More Complicated Example



If we have more than one network (redundancy) and the SG is capable of switching to use another network, then we can blacklist this network path. (Done by blacklisting the remote interface, as if we blacklist the network we'll be unable to use it for other assets).

Dependability Model Details

- The dependability model is stakeholder centric
 - represents a dynamic service-oriented system from the perspective of one (primary) stakeholder
- Assets represent value to the primary stakeholder
 - Customers represent the value of the relationship and interactions with an accountable client
 - ServiceGroups represent the value of a working service implementation
 - Resources represent the value of upstream services, etc.



Behaviours

- Asset values are compromised if they (are induced to) misbehave
 - overcommitted: compromises credibility
 - overloaded: compromises performance (due to external load)
 - unavailable: compromises availability
 - unreliable: compromises (process) integrity
 - underperforming: compromises performance (due to internal problems)
 - inaccurate: compromises (data) integrity
 - inaccessible: compromises accessibility
 - unauthentic: compromises authenticity
 - unauthorized: compromises authority
 - unaccountable: compromises (business) responsibility
 - indiscreet: compromises confidentiality
 - dissatisfied: compromises business value



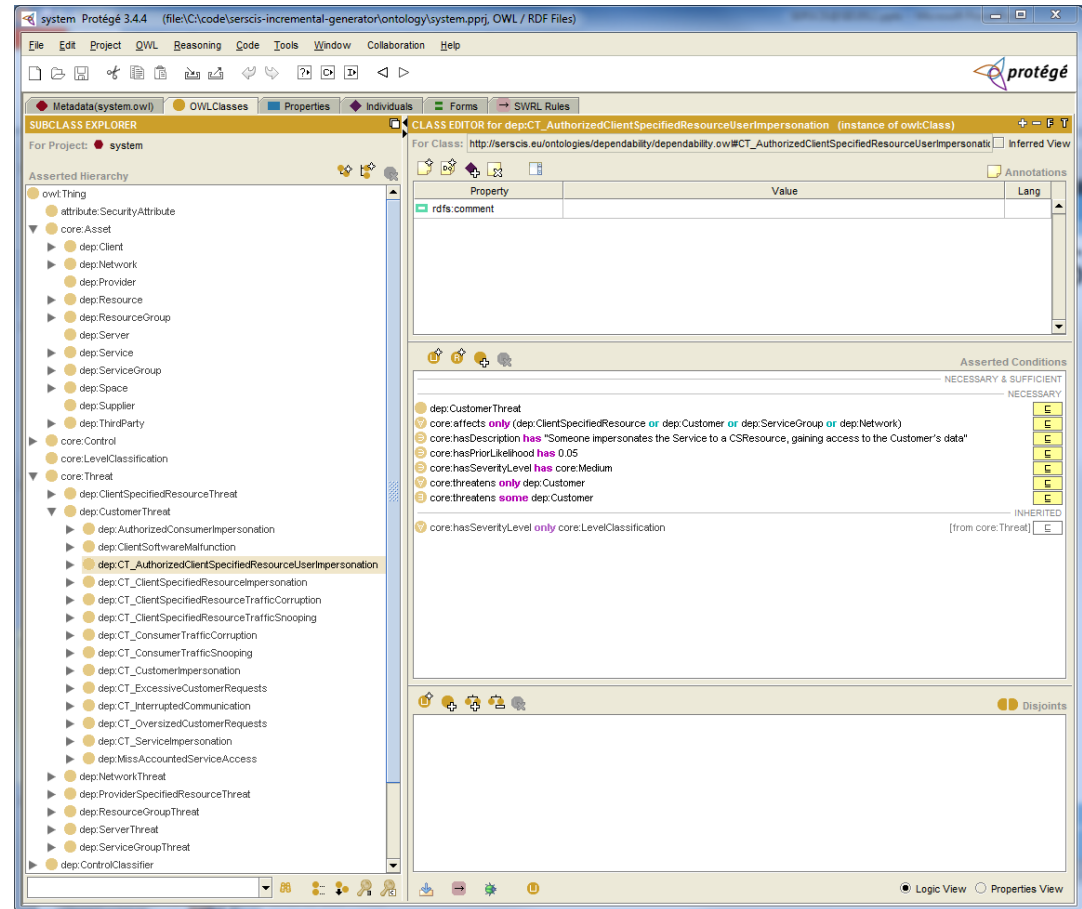
Threats and Controls

- Threats represent both a cause and its effect on a targeted asset
 - other assets may also be affected: their behaviour may change or their controls may be related to the threat
- Controls represent the means to block or mitigate threats involving a protected asset
 - controls are defined by their effects, not their mechanism
 - the effect is defined by SWRL threat classification rules
- Threat classification rules define which controls are needed at threatened and involved assets to block or mitigate each threat

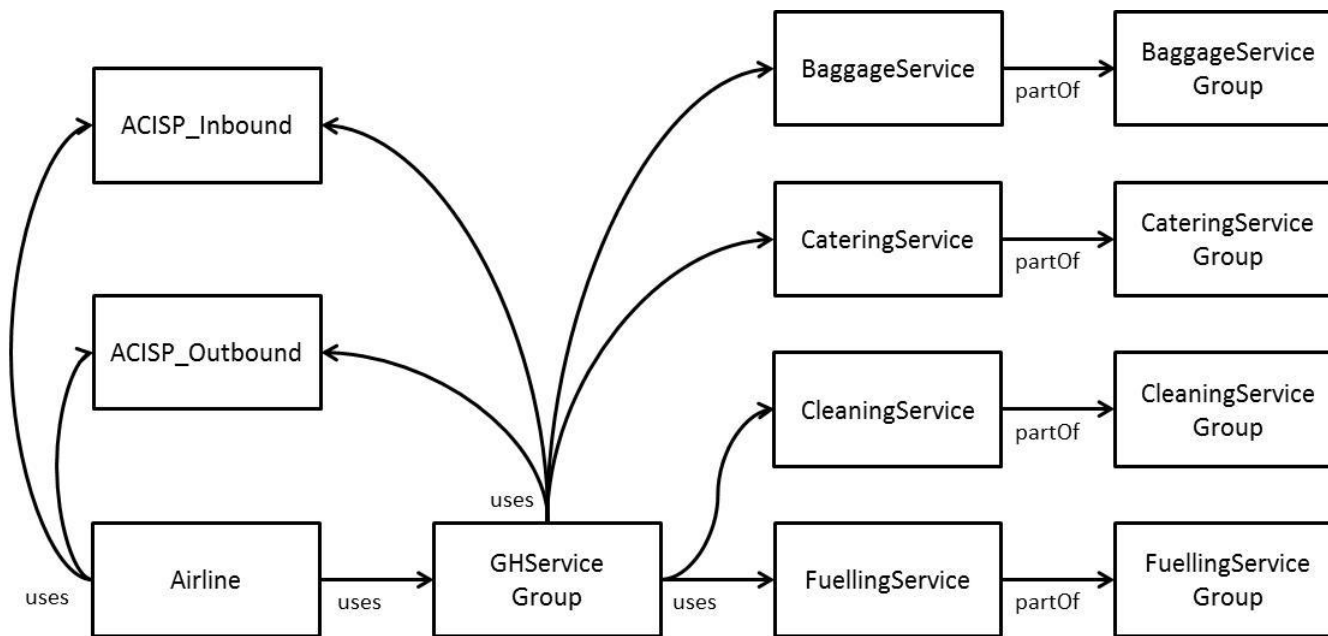


Dependability Model: Statistics

- 12 asset types
- 82 threat types
- 21 control types
- 15 behaviours
- 133 SWRL rules



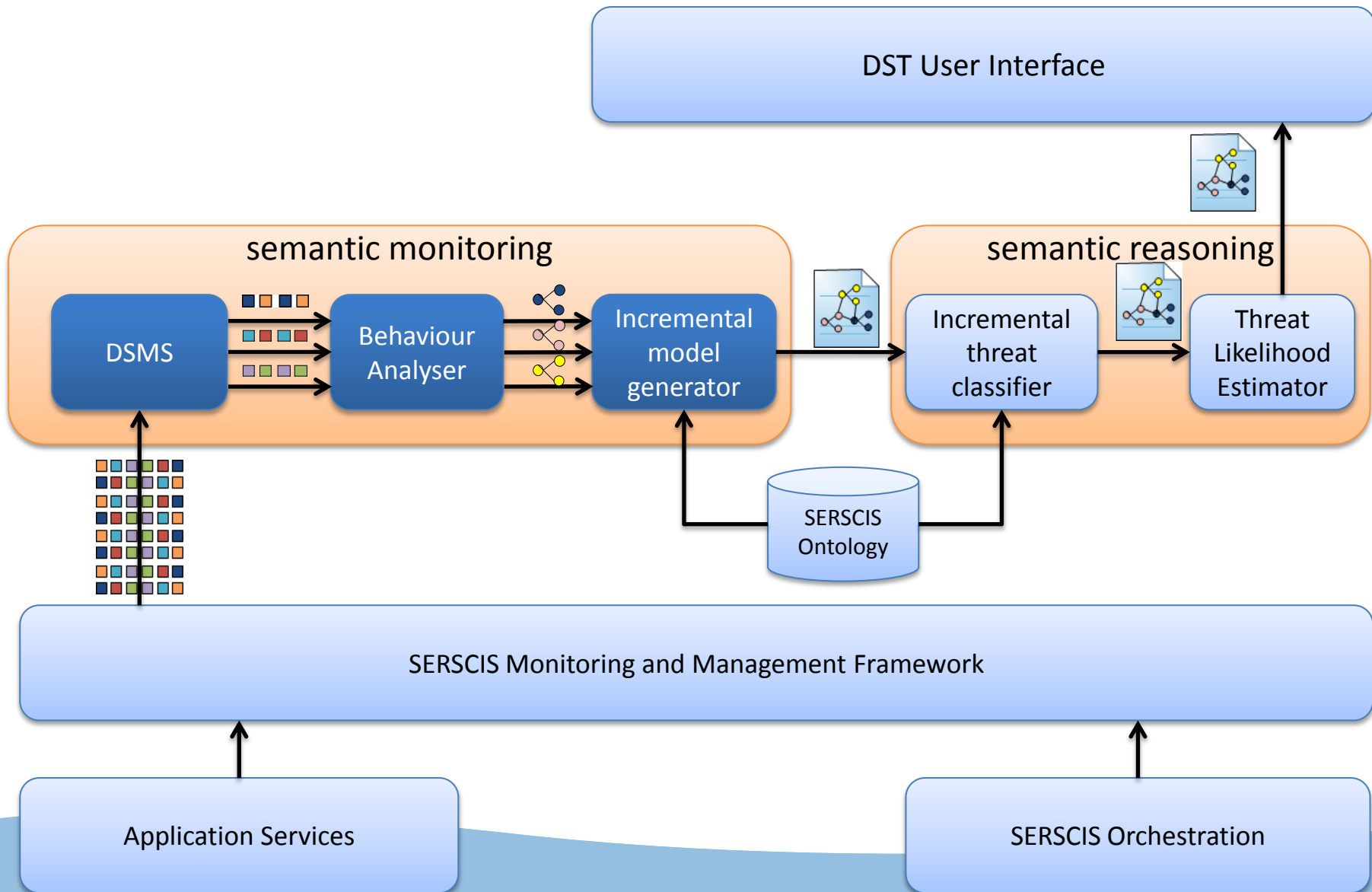
Abstract System Model



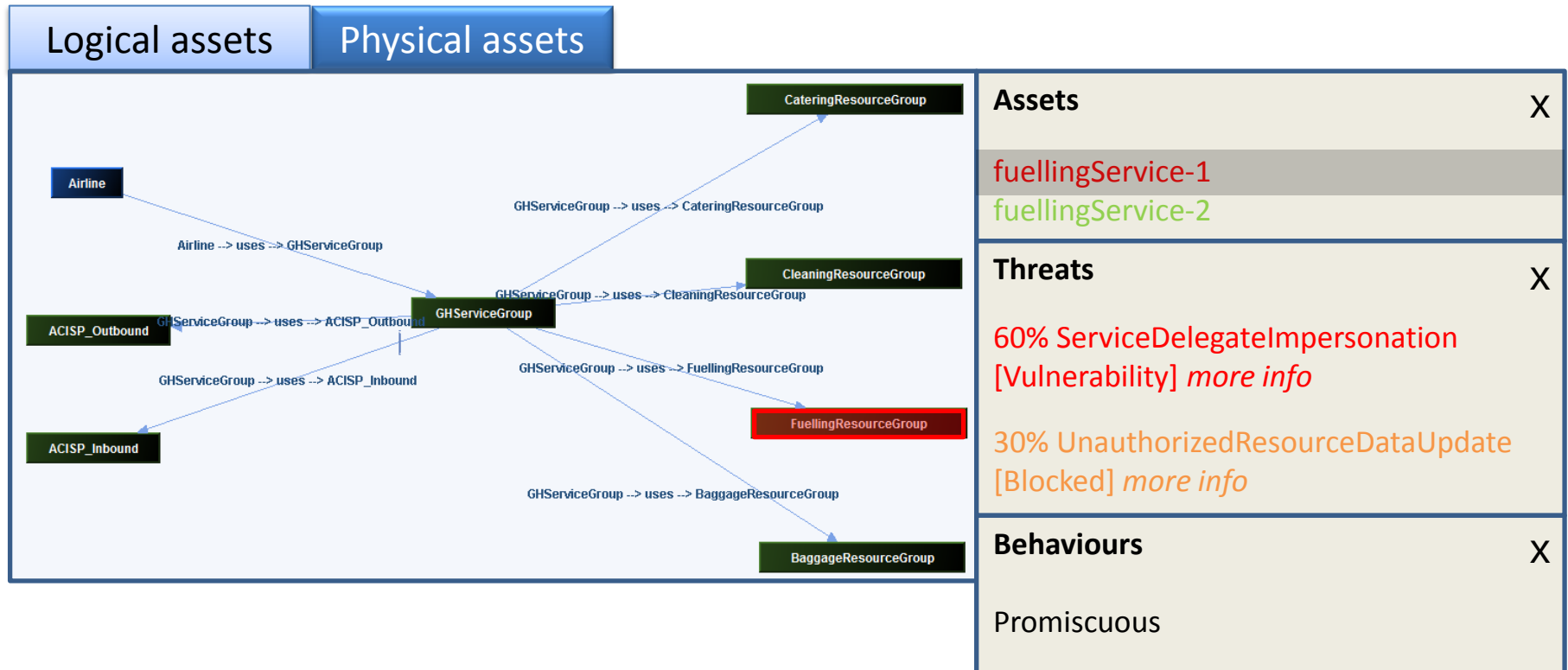
- A design-time model of the structure of a specific dynamic, multi-stakeholder service-oriented system
- Created and maintained by the system designer



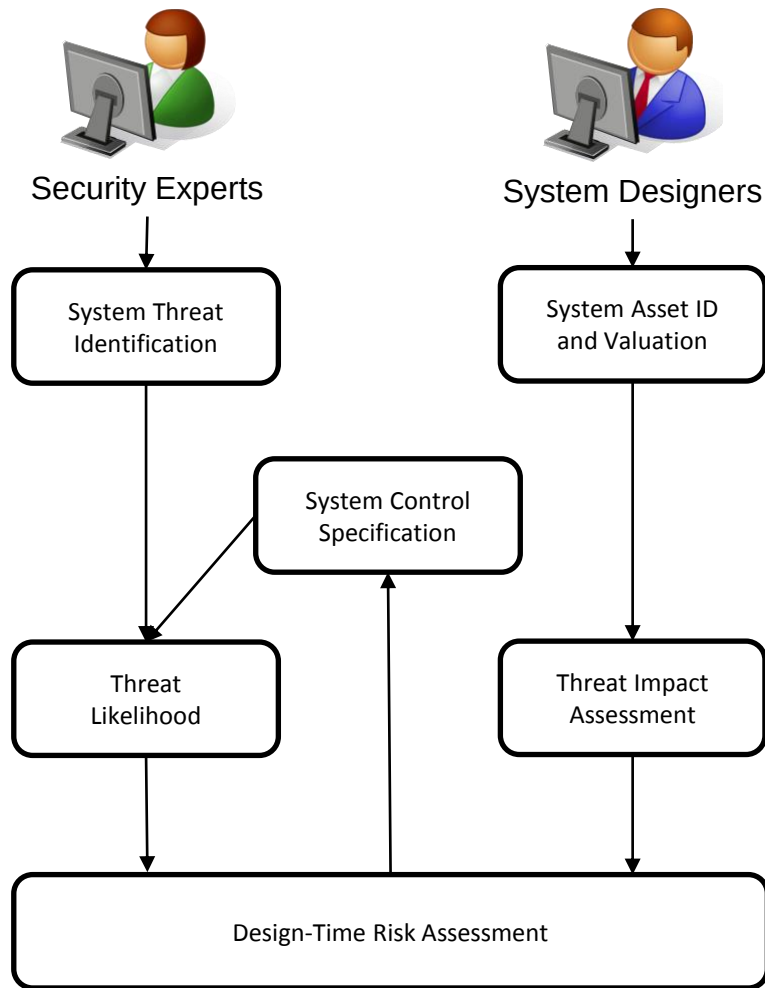
SERSCIS Run-time



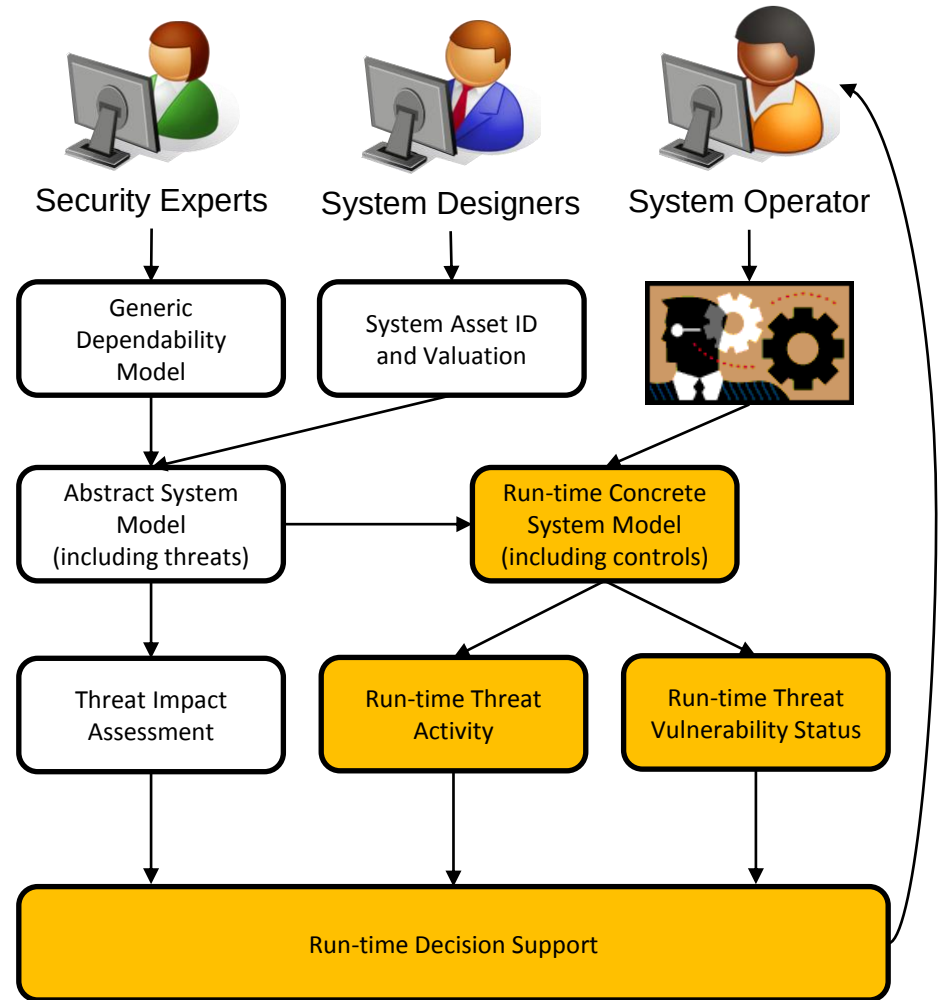
SERSCIS Decision Support



SERSCIS Risk Management



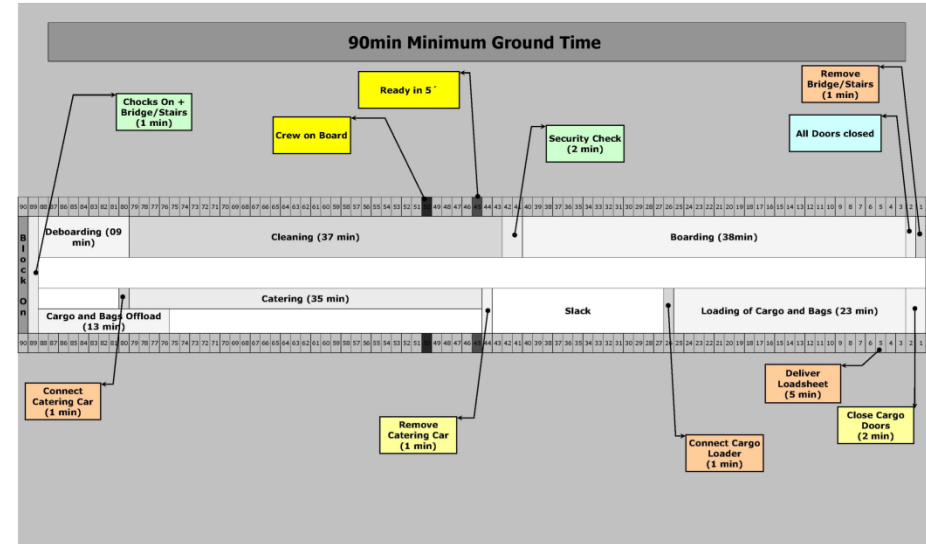
Conventional Risk Management



SERSCIS Risk Management

Validation Case Study

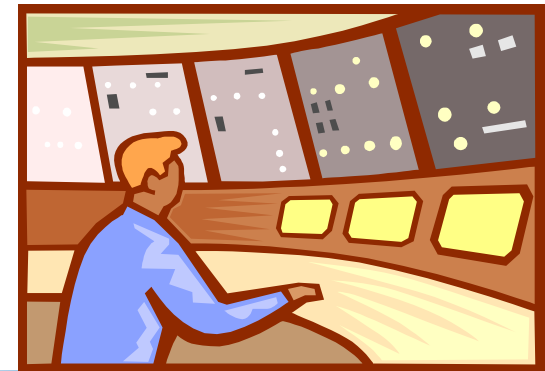
- Airport Collaborative Decision Making (A-CDM)
 - focus on air-side aircraft turn-around
 - complex multi-actor workflow composition and management
 - heavily dependent on ICT data exchange for process management



- Quality of information is key
 - accuracy of service scheduling information
 - trustworthiness of information sources
- Source of aircraft ready times needed by Eurocontrol

Summary

- Proof-of-concept case studies completed
 - using an emulated A-CDM network (Airlines, Ground Handler, Ramp Services, CFMU) loosely based on Vienna Airport
 - simulates attacks and faults that can't be injected in a real airport
- SERSCIS methodology and concepts proven
 - design-time modelling of A-CDM network
 - run-time detection and diagnosis of simulated threats: unauthorized data tampering, software faults and supply-side resource shortages
 - machine reasoning for threat detection
- <http://www.serscis.eu>
 - models and tools release open-source
- Future work: scale up to wider ATM system



Credits



- IT Innovation Centre, UK
 - project coordination, semantic modelling, autonomic system management and composition



- Joanneum Research Centre, Austria
 - run-time system composition, risk management methodology



- Austro Control, Austria
 - A-CDM validation case studies, air-traffic sector dissemination and exploitation



- KEMEA, Greece
 - run-time decision support methodology and tools, validation case studies



- Port of Gijon, Spain
 - multi-sector dissemination and exploitation

Autoridad Portuaria de Gijón