



Mission Cyber Security Situation Management

Gabriel Jakobson

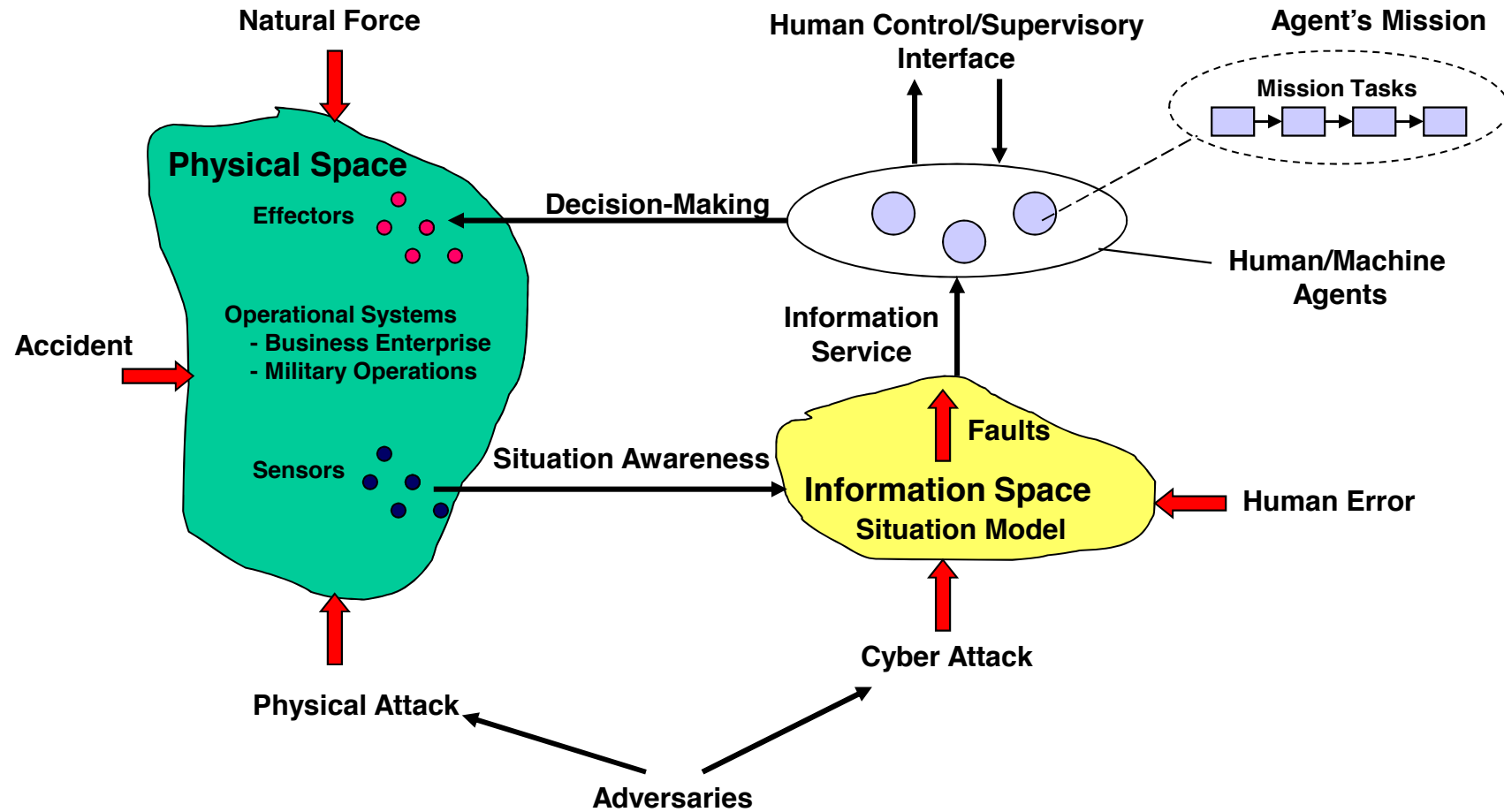
**Altusys Corp., Princeton, NJ, USA
jakobson@altusystems.com**

**2- 4 May, 2011
Tallinn, Estonia**

On Cyber Security Incidents

- **Despite the continuous efforts to secure cyber space of tactical missions, the mission command and control operations inevitably experience security incidents, which result in the loss of confidentiality, integrity and availability of data, and ultimately cause the degradation of the operational objectives of conducted missions, or even bring them to full abort.**
- **Similar security incidents happen in civil applications, including finance, transportation, health care, and enterprise business process management generally**
- **The source of the incidents may be different: cyber attacks, faults in the systems components, human errors, natural or human-made disasters; although all of these incidents may have several similar characteristic in terms of the impact on missions and business processes, we will concentrate in this talk on cyber security incidents**

Security Situation Management



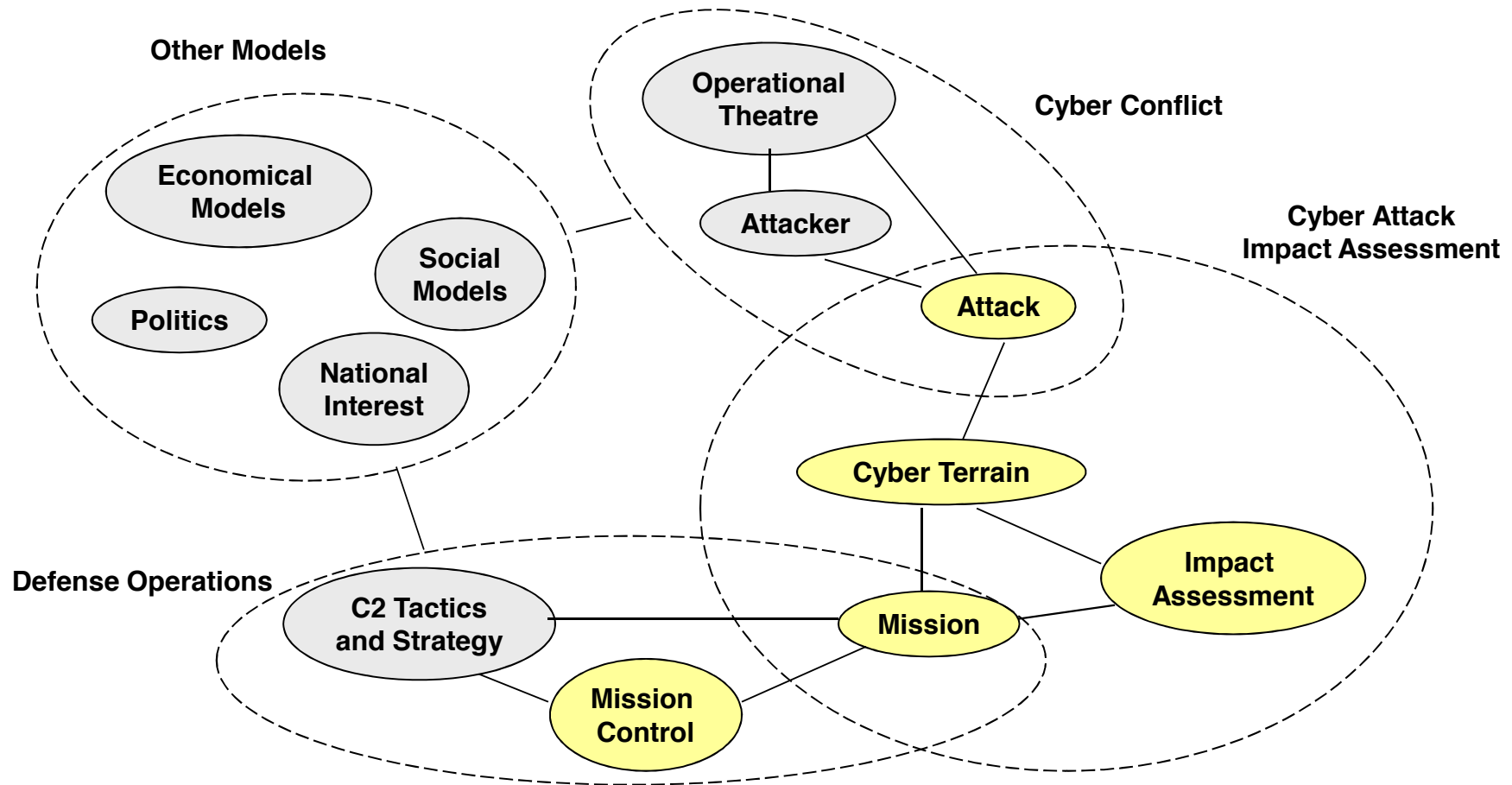
Mission Centricity in Cyber Defense

- **The success of cyber security is traditionally determined by a level of protection provided to information systems (network infrastructure, software assets, and information services) against cyber attackers.**
- **At the same time, the emerging US cyber warfare doctrine states that the ultimate goal of cyber security is protection of current and planned missions from cyber attacks.**
- **Consequently, the answer to the question: “How well did we succeed in protecting our cyber assets?” depends on “How well we succeed in securing the goals set for missions?” We call this mission-centricity in cyber defense.**

Cyber Attack Tolerant Missions

- **The history of intrusion-detection technology of information systems shows that perfect detection and mitigation of cyber attacks remain elusive goals - even information systems that were developed at great cost contain residual vulnerabilities.**
- **We shift emphasis from the hardly possible ``bulletproof'' information systems to self-organized information systems that are capable of self-protection, self-survival and self-recovery.**
- **These information systems being under cyber attacks should support continuity of missions, possibly with pre-defined acceptable degradation of the mission goals. We call such missions as cyber attack-tolerant missions.**

“Big Picture” Of Cyber Situation Management



Course Outline

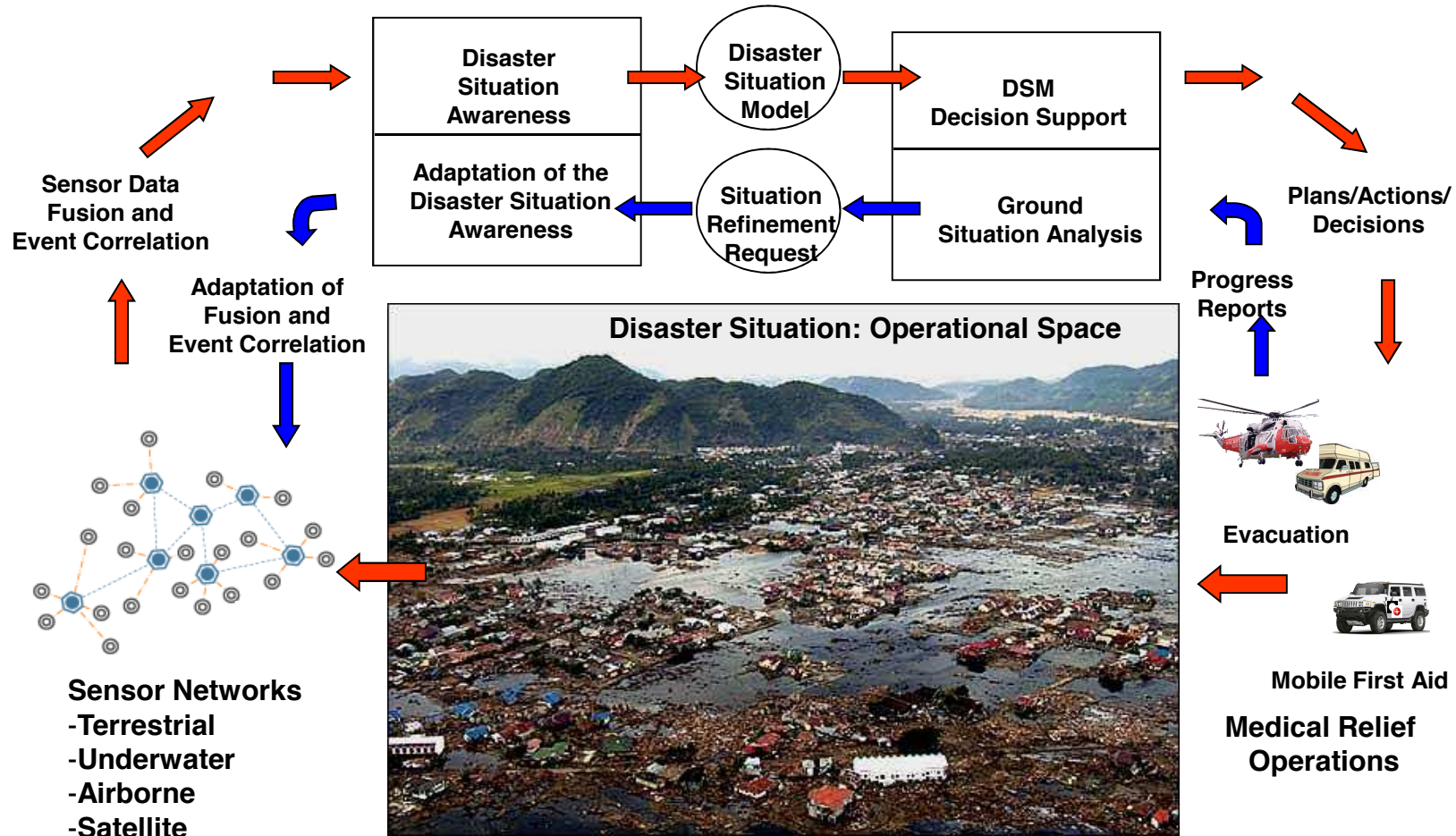
1. Introduction
2. Brief Overview of Situation Management
3. Mission Cyber Security Modeling Framework
4. Real-Time Cyber Attack Impact Assessment
5. Assessment of Plausible Future Cyber Situations
6. SAIA – Situation Awareness and Impact Assessment System
7. Sample Application
8. Conclusions and Future Research

Brief Overview of Situation Management

What is Situation Management?

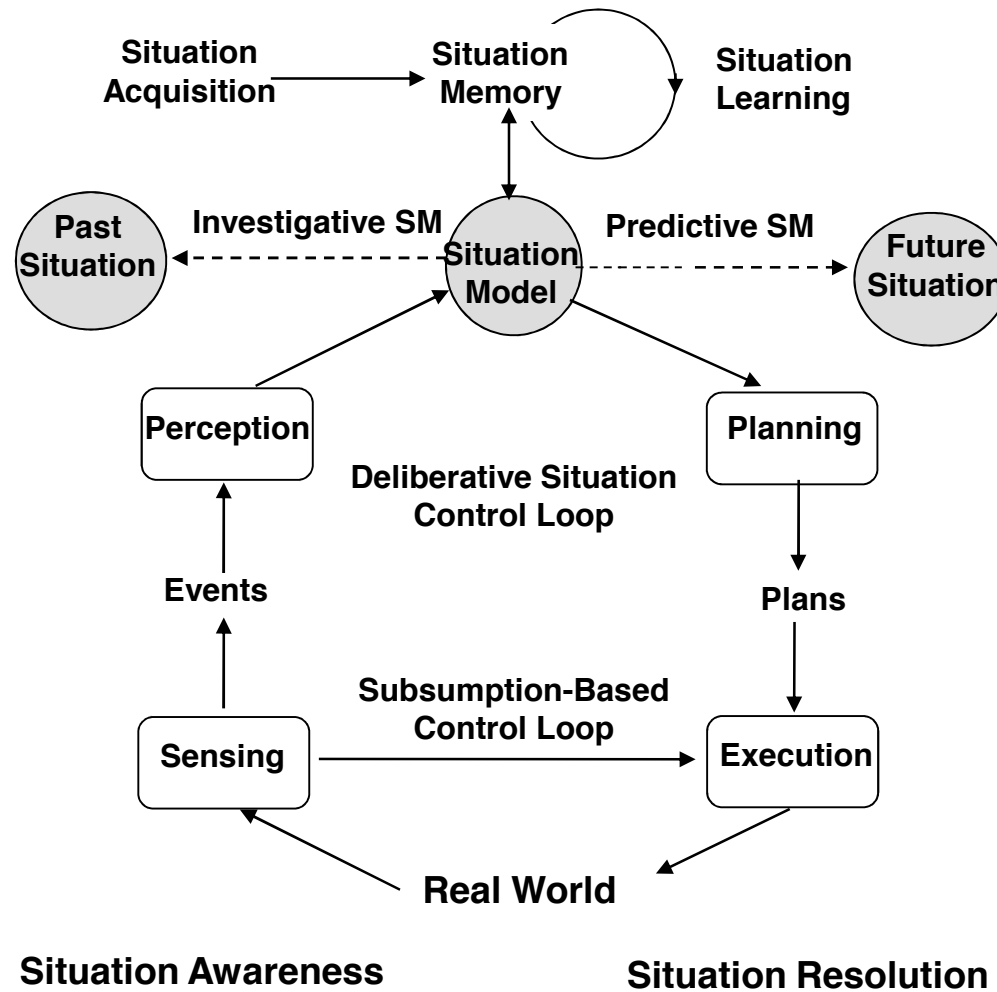
- **Situation Management (SM) is a synergistic goal-directed process of recognition, control, and prediction of situations happening in dynamic systems;**
- **The tasks of instrumentation of the dynamic system, modeling of situations, reasoning about the situations, action planning, situation prediction, domain and situation knowledge acquisition, and situation learning are essential technology ingredients of SM;**
- **Informally, situations are seen as states of a dynamic system observed at particular time;**
- **Complexity of the situations may range from a single attribute value of an object, or a single relation among the attribute values, to complex collections of objects interlinked by various class, structural, spatial, temporal, and other (domain-specific) relations;**

Disaster Rescue and Recovery



Gabriel Jakobson, "Situation Sensing, Fusion and Management for Collaborative Emergency Operations", Presentation at Pacific Telecommunication Conference PTC 2008, Honolulu, Hawaii, January 13-15, 2008.

Situation Management : A High-Level View



The Types of Situation Management

- We identify three basic types of situation management :
 - diagnostic,
 - control, and
 - predictive type of situation management
- The investigative SM is concerned with a retrospective analysis of causal situations which determine why a certain situation happened. The control type of SM aims to change or keep the current situation, while the predictive type of SM aims to project possible future situations.
- For example, finding a root of a packet transmission failure in a telecommunication network is an example of an investigative SM; moving a tank unit from the area of direct hostile fire is a control type SM; and a projection of a potential terrorist attack on a critical infrastructure element is an example of a predictive SM.

Deliberative and Subsumption-Based Situation Control Loop

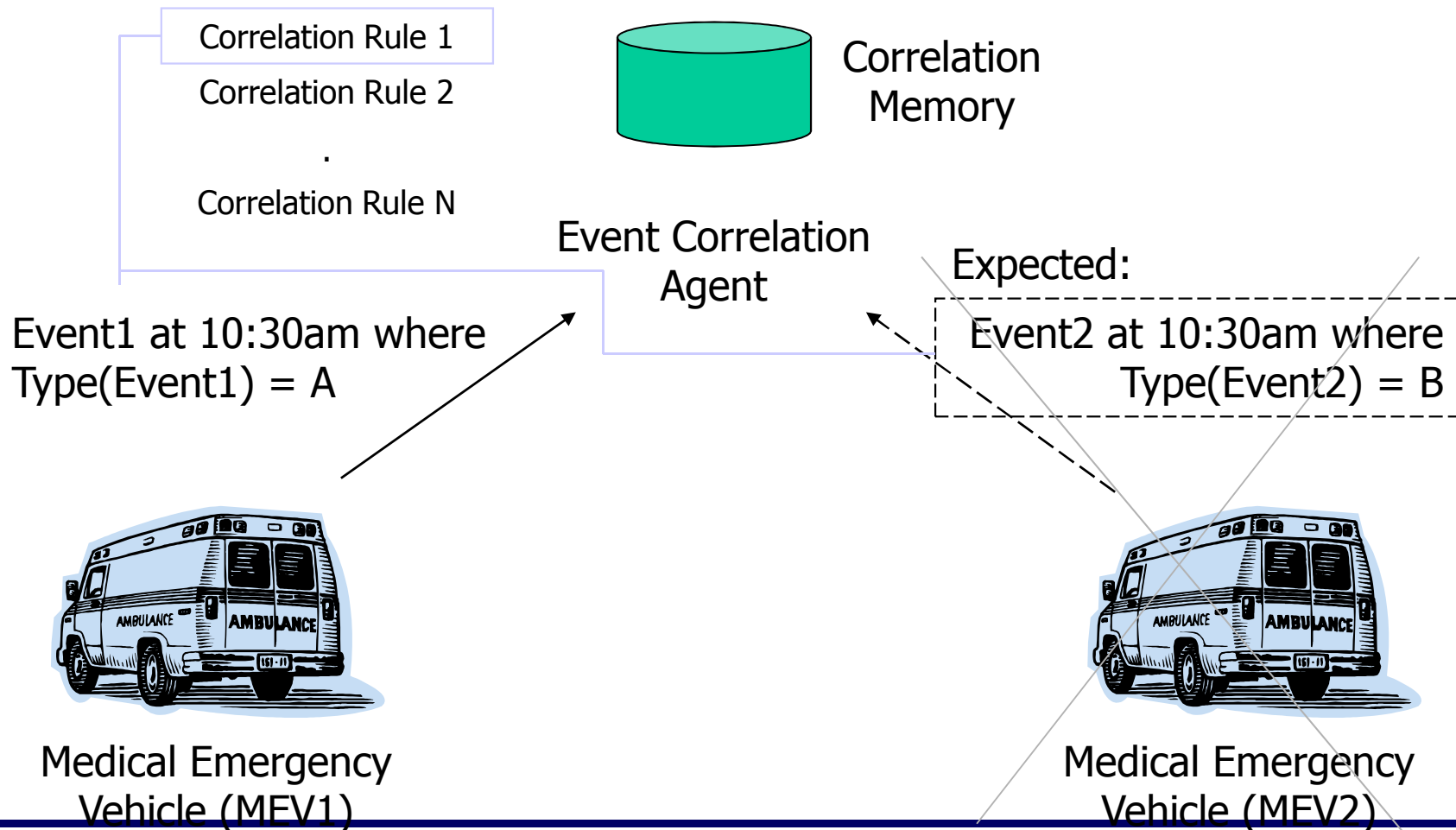
- **Deliberative Situation Control Loop:**
 - Sensing
 - Perception
 - Planning
 - Execution.
- **Subsumption-Based Control (from sensing to execution)**
- **Major Processes of Situation Management:**
 - Situation Awareness (sensing + perception + prediction))
 - Situation Resolution (planning + execution)
 - Situation Knowledge Management (acquisition + learning)

Situation Management Applications

Interest in SM is motivated by the increasing complexity and scale of real-time applications, including such applications as:

- Military applications in imagery, sensor, radar, sonar and intelligence information processing for target identification and tracking; asymmetric and network-centric battlefield management;
- Emergency and crisis management applications of post-disaster relief and recovery operations during natural, technological and terrorist caused disasters;
- Industrial applications related to real-time surveillance, fault diagnostics, and predicting the behavior of complex networks and systems;
- Security applications in the area of threat prediction, vulnerability analysis, and intrusion detection associated with the protection of human, cyber and physical assets;
- Medical applications of human body sensing, real-time health monitoring and medical situation recognition

Example: DSM Situation Recognition



Example: DSM Recognition Rule

Suppose an event of type *A* issued at time *t1* from a some medical emergency vehicle *?mev1*, but during the following 1-minute (60 second) interval an expected event of type *B* was not issued from medical emergency vehicle *?mev2*. It is also noted that medical emergency vehicles *?mev1* and *?mev2* form a group. The prefix '?' refers to a variable.

Correlation Rule 2: EXPECTED-EVENT-RULE

Conditions:

MSG: EVENT-TYPE-A ?msg1

TIME ?t1

VEHICLE: VEHICLE-TYPE-MEV ?mev1

Not MSG: EVENT-TYPE-B ?msg2

TIME ?t2

VEHICLE: VEHICLE-TYPE-MEV ?mev2

GROUP: GROUP-TYPE-MEV ?mev1 ?mev2

AFTER:?t1 ?t2 60

Actions:

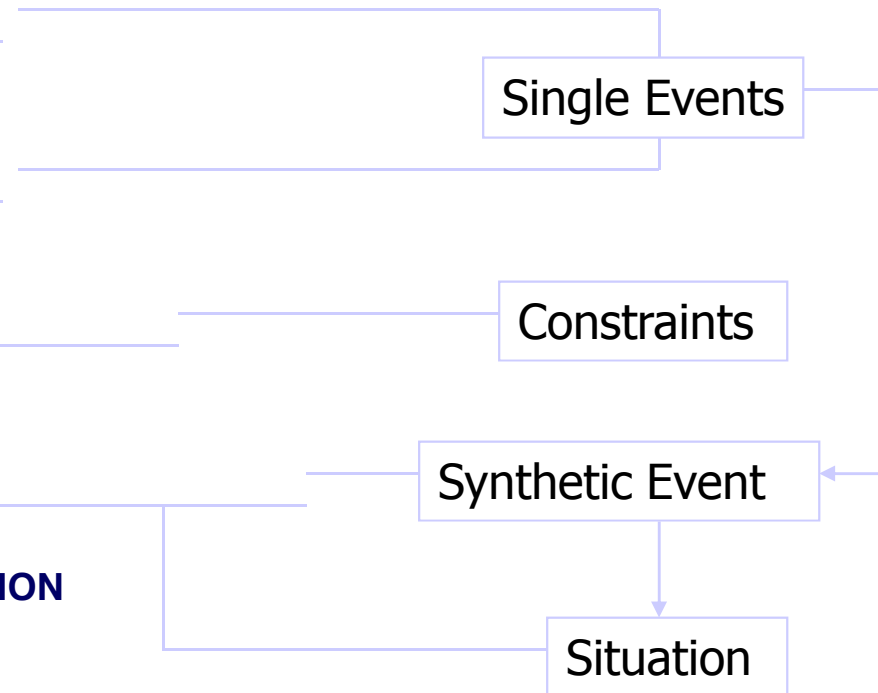
AssertSituation: LOST-MEV-CONTACT-SITUATION

VEHICLE1 ?mev1

VEHICLE2 ?mev2

EVENT1 ?msg1

EVENT2 ?msg2



Example: DSM Recognition Rule

SituationName LOST-MEV-CONTACT-SITUATION

SituationClass MEV-SITUATION

Parameters

VEHICLE1

VEHICLE2

EVENT1

EVENT2

.....

Actions

PLAN SEND-EMERGENCY-HELICOPTER

Situation

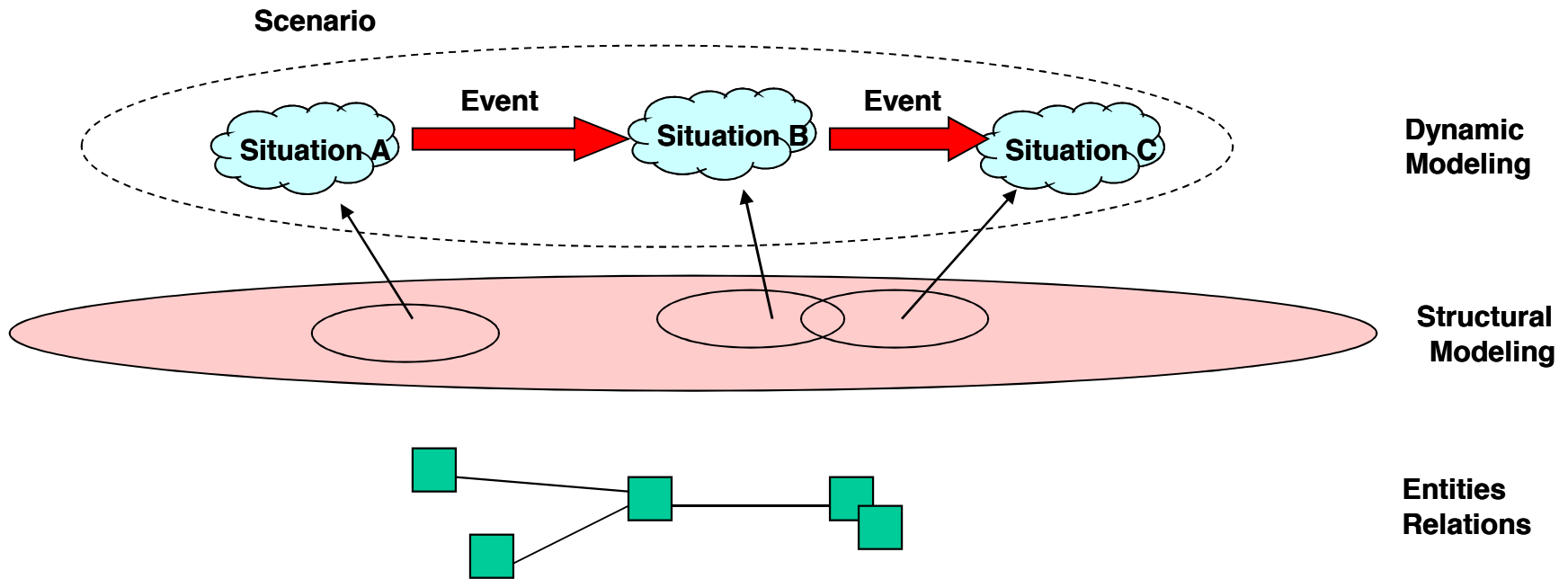
Simple Deliberative
Plan Reasoning

Plan

Situation Modeling - Overview

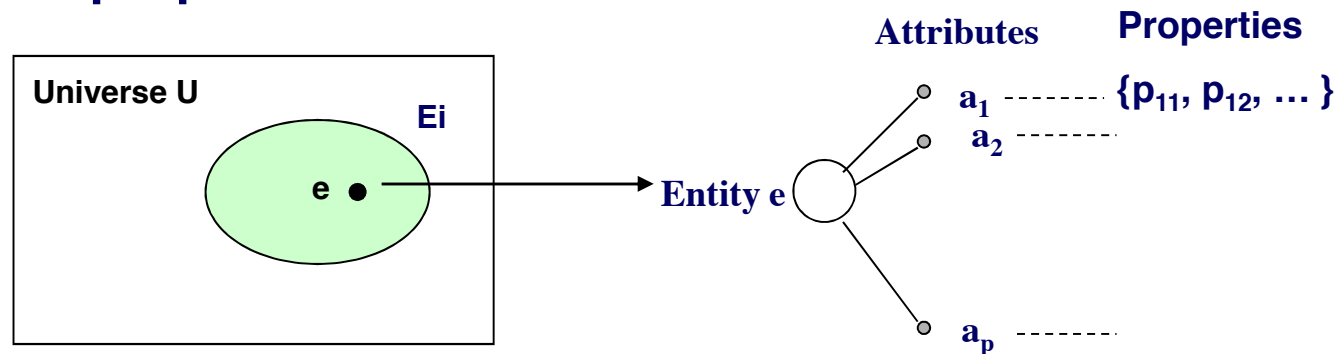
- **Structural Modeling**
 - **Entities, attributes, attribute domains, constraints**
 - **Entity classes, class ontologies, core ontologies**
 - **Relations**
- **Dynamic Modeling**
 - **Situations**
 - **Events**
 - **Actions**
 - **Time**
- **Representation**
 - **Primary concept specification languages (set-theoretical, FSM)**
 - **Graphical modeling languages (e.g. UML)**
 - **Programming languages (e.g. SGL, GOLOG)**

Situation Modeling



Entities

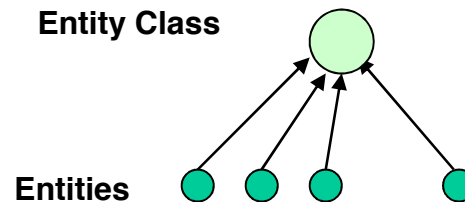
- Let's assume that there exists a universe U , real or abstract that could be sensed, perceived, reasoned and affected, and which is populated with entities $e \in E_i$, $E_i \subseteq U$
- An entity e is a thing of significance that has distinctive existence and is represented by set of attributes $\{a_1, a_2, \dots, a_p\}$
- Each attribute is a collection of attribute properties, such as attribute name, type, value, default value, and other application-specific properties



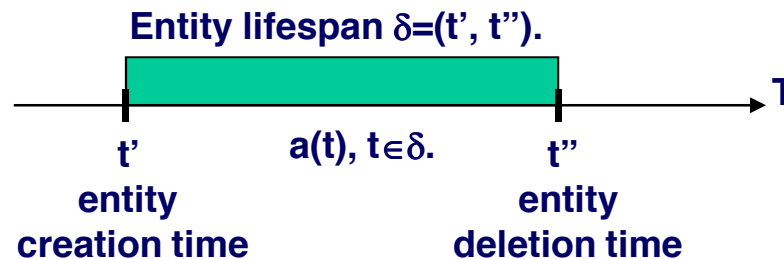
- Attribute value is a triplet containing an actual attribute value, certainty estimation, and time, either a point or interval time during which the attribute holds its value.

Entity Classes

- A set of entities with certain common attributes defines an abstract entity class



- Some entities are active, they change their attributes and properties in time; Some of the entities can interact with other entities forming multi-entity systems
- We will consider entities as dynamic time-dependent objects with their time of creation t' , time of clear t'' , and corresponding lifespan $\delta=(t', t'')$. Any attribute value of an entity is defined only during the existence of the entity, i.e. $a(t), t \in \delta$



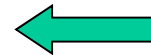
Example: A Simplified Entity Class Specification of a M1-Abrams Tank

```

<DomainClass Name="M1-Abrams" Documentation="A class describing US Army M1 Abrams Main Battle Tank">
  <DomainClassParent>
    <DomainClassLink Name="Main-Battle-Tank"/>
  </DomainClassParent>
  <DomainClassLocation>
    <DCLocSlot Name="Tank-Location">
  </DomainClassLocation>
  <DomainClassTime>
    <DCUnivTimeSlot Name="Unit-Time">
  </DomainClassTime>
  <DomainClassSlots>
    <DCIntegerSlot Name="Combat-Weight" 54.5/>
    <DCIntegerSlot Name="Maximum-Speed" 45/>
    <DCIntegerSlot Name="Power-to-Weight-Ratio" 27/>
    <DCIntegerSlot Name="Total-Crew" 4/>
    <DCIntegerSlot Name="Length-of-Hull" 24.49/>
  </DomainClassSlots>
  <DomainClassMethods>
    DCDatabaseMethod SetValue "Tank-Location"
    DCDatabaseMethod GetValue "Tank-Location"
  </DomainClassMethods>
</DomainClass>

```

Entity Model



Entity



Entity Specification

US Army M1 Abrams Main Battle Tank
 Combat Weight: 54.5 tons
 Maximum Speed: 45 mph
 Power to weight ratio: 27 HP/ton
 Length of Hull: 24.49 feet
 Height: 8.68 feet
 Total Crew: 4 soldiers
 Weapons: 120mm Howitzer,
 .50 Caliber Heavy Machine Gun,
 and two 7.62mm M60 GPMGs

Relations

Relation is a mental abstraction of linking a certain number, very often two, entities together. Mathematically, relation $R \subseteq E_1 \times \dots \times E_m = \{(e_1, \dots, e_m) / e_1 \in E_1, \dots, e_m \in E_m\}$, where $E_1, \dots, E_m \subseteq U$.

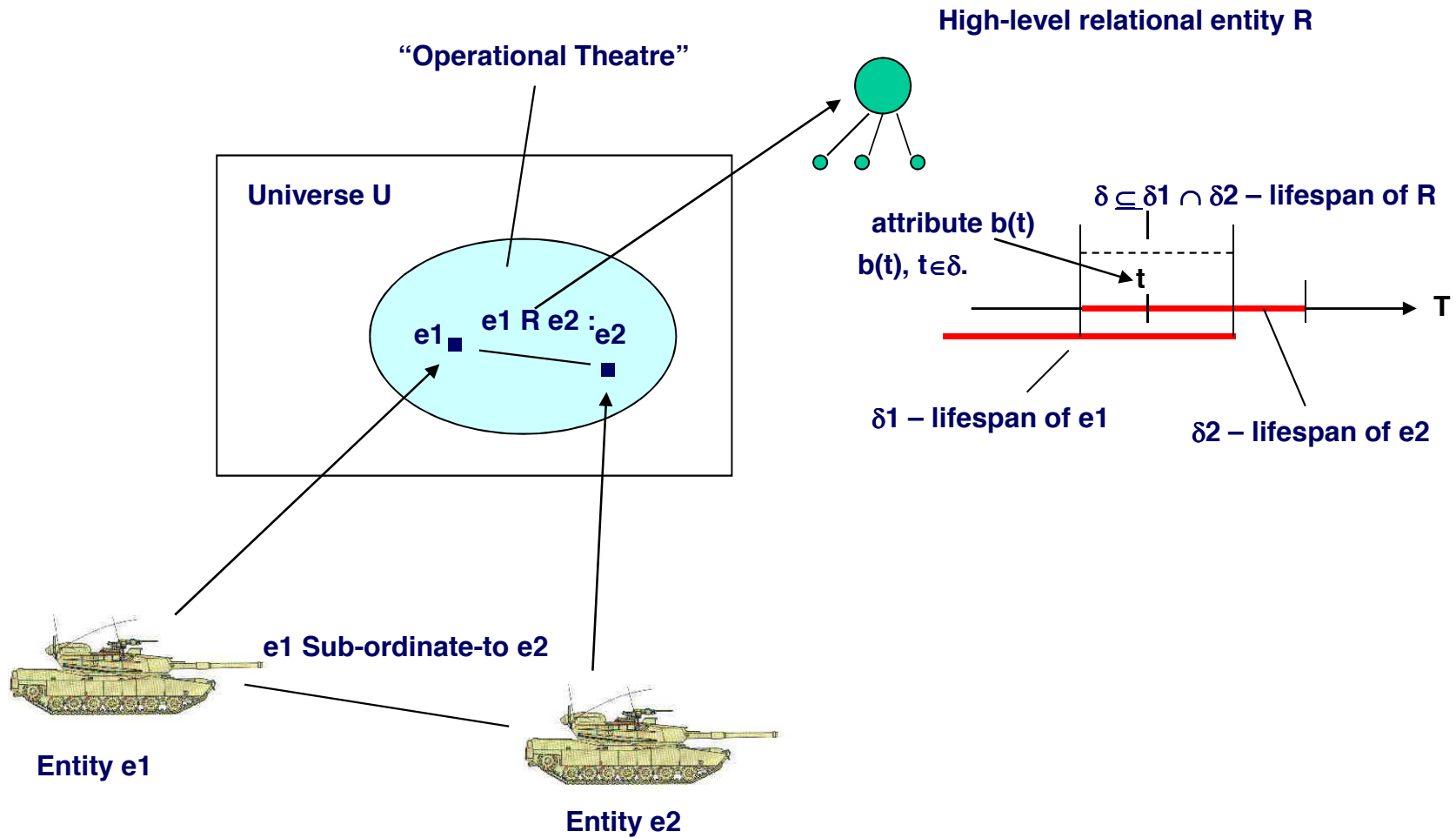
In most practical applications it is enough to consider only binary relations $R \subseteq E_i \times E_j$.

Relation R could be considered as a set of instant relationships $R = \{r_1, \dots, r_q\}$. In case of binary relations the commonly used notation for $r \in R$ is $r = e_i R e_j$, where $r = (e_i, e_j) \in R$.

In several practical applications it is required to consider relations as entities, in sense that they are characterized by set of attributes $\{b_1, b_2, \dots, b_h\}$, and all the features that were attached to the attributes of entities.

In the same way as entities, we will consider relations as dynamic time dependent objects with their time of creation t' , time of clear t'' , and corresponding lifespan $\delta = (t', t'')$. The following time dependency should hold for a relationship: if $e_i R e_j$ and δ_i, δ_j are lifespans of e_i, e_j , accordingly, then for the relationship $e_i R e_j$ the lifespan $\delta \subseteq \delta_i \cap \delta_j$. Any attribute value $b(t)$ of a relation is defined only during the existence of the relation, i.e. $b(t), t \in \delta$.

Time-Dependent Relational Entity Model



Types of Relations

- **For our further discussion, it is important to consider the following types of relations between entities:**
 - **Class relations**
 - **Structural relations**
 - **Spatial relations**
 - **Domain-specific relations**
- **Class relation establishes a link between an entity and abstract entity class or between entity classes. Class relation is the major tool of conceptualization of entities and building conceptual frameworks of abstract concepts (ontologies).**
- **Structural relations Part-Of , Overlaps-With and Similar-With are the basic construction primitives of the universe.**
- **Spatial relations Inside, Near, Above, etc. are used to express topological (spatial) links between the entities.**
- **There is large number of various domain specific relations, which semantics depends on the particular domain. For example, Service x Supported-by Network y, Unit x Under-Fire-of unit y, Element x Connected- by Trunk-T1-to Element y.**

Base Situations

1. Entity-based situation. Let $\{a_1, \dots, a_p\}$ be set of situational attributes of entity e . Situation $S_e(d)$ on entity e during a time interval d , $d \subseteq \delta$, where δ is the lifespan of entity e is defined as

$$S_e(d) = \langle a_1(t), \dots, a_p(t) \rangle \in v_1 \times \dots \times v_p / \forall (t, t') \in d [\langle a_1(t), \dots, a_p(t) \rangle = \langle a_1(t'), \dots, a_p(t') \rangle]$$

The base situation $S_e(d)$ is a collection of entity e states (the time-stamped attribute vectors) that have the same value during a time interval d .

Consequently, a situation has a duration, i.e. a start-time and an end-time.

2. Relational entity-based situation. Let $\{b_1, \dots, b_q\}$ be set of situational attributes of relation R . Situation $S_R(d)$ on relation R during a time interval d , $d \subseteq \delta$, where δ is the lifespan of relation R is defined as

$$S_R(d) = \langle b_1(t), \dots, b_q(t) \rangle \in v_1 \times \dots \times v_q / \forall (t, t') \in d [\langle b_1(t), \dots, b_q(t) \rangle = \langle b_1(t'), \dots, b_q(t') \rangle]$$

3. Relational situation. Let $R \subseteq E_i \times E_j$, where $E_i, E_j \subseteq U$, $(e_i, e_j) \in R$, and δ_i, δ_j are lifespans of e_i, e_j , accordingly, then

$$S_{(e_i, e_j)}(d) = e_i R e_j$$

is a situation, where $d \subseteq \delta$, $\delta = \delta_i \cap \delta_j$, where δ is the lifespan of the relation R . R is defined as a structural, spatial or domain specific relation.

Dynamic Domain Modeling - Complex Situations

Complex situations could be constructed from other situations using set-theoretical union and inter-section operations.

1. If $S_{B_1}(d_1)$ and $S_{B_2}(d_2)$ are two situations, where $B_1, B_2 \subseteq U$ and d_1, d_2 are subsets of common lifespans of all entities in B_1, B_2 , correspondingly, then,

$$S_B(d) = S_{B_1}(d_1) \cup S_{B_2}(d_2) \text{ and } S'_{B'}(d') = S_{B_1}(d_1) \cap S_{B_2}(d_2)$$

are situations, where, correspondingly

$$d = d_1 \cap d_2 \text{ and } B = B_1 \cup B_2 \text{ and } d' = d_1 \cap d_2 \text{ and } B' = B_1 \cap B_2$$

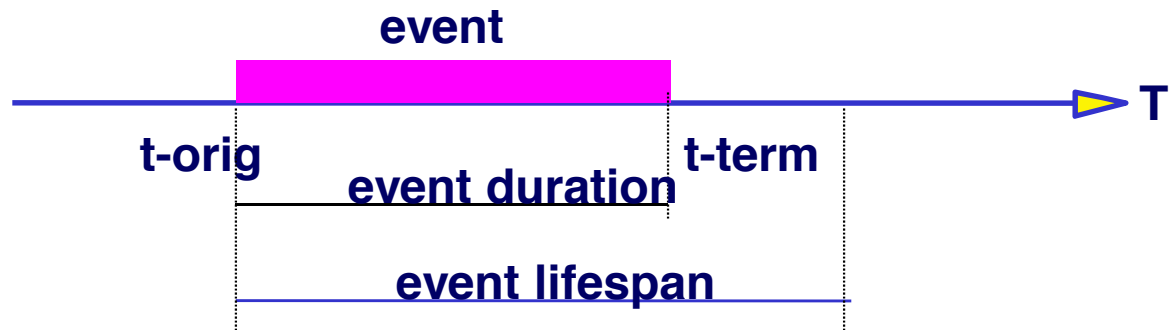
2. Due to the use of active entities and situational attributes, multiple different situations can be defined on the same set of entities and relations
3. Logical and temporal relations could be defined between situations, however only as between predicates, e.g. S' & S'' and S' AFTER S'' are predicates not situations

Events

- **Event is a time-dependent internal change of the system situation (state)**
- **Some events are manifested by the built-in external event messages**
- **Happenings of other events are captured by external sensing and surveillance equipment or human site observers**
- **Missions and business processes are events**

Event Time Dependency

a) Interval Time



b) Point Time



Ontology for Knowledge Representation

- **Situation Management is a knowledge-intensive process and requires a large body of well-represented and organized knowledge.**
- **One of the most effective organizational principle and tool for handling knowledge is Ontology.**
- **The term ontology is borrowed from philosophy, where it refers to a systematic account of existing reality, i.e. all notions, objects, relations, properties, etc. that exist in reality. Note: the key word here is exist.**
- **Contrary to that, Computer Science (and specifically, AI) looks on ontology as a classification of things, not so much that might exist in reality, but rather those ones, which can be represented.**
- **References**
<http://www.aaai.org/AITopics/html/ontol.html>
Blackwell Guide to the Philosophy of Computing and Information, Oxford: Blackwell, 2003, 155–166.

Ontology Languages and Tools

- **Ontology Programming Languages and Tools**
 - Majority of the current ontology representation languages are based on XML extensions
 - OWL – Web Ontology Languages <http://www.w3.org/TR/owl-features/>
 - SWRL – Semantic Web Rule Language <http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/>
 - RDF – Resource Description Framework <http://www.w3.org/RDF/RDF>
 - DAML – DARPA Agent Markup Language <http://www.daml.org/>
DAML+OIL, OIL Ontology Inference Language (precursor to OWL)
- **Ontology Visualization Languages**
 - UML is used to visualize ontology;
 - programs exist to generate RDF, DAML, OWL code from UML specifications

K. Baclawski, M. M. Kokar, P. A. Kogut, L. Hart, J. Smith, W. S. Holmes III, J. Letkowski and M. L. Aronson. *Extending UML to Support Ontology Engineering for the Semantic Web*. Proceedings of the Fourth International Conference on the Unified Modeling Language, Toronto, Canada, 2001.

Situation Awareness

- In early nineties the term “situation awareness” was almost synonym to industrial ergonomics and human factors studies of human operator safety and effectiveness, e.g. pilot in a cockpit
- Several situation awareness models were proposed, most notably the models developed by Endsley and Garland
- Nowadays situation awareness has found important place in information fusion research and engineering, initially related to military applications of signal fusion for target identification and tracking.
- The abstraction to more general model of fusion prompted the development of JDL fusion model, where the level 2+ was directly associated with operational situation awareness and threat prediction.

M. R. Endsley, “Towards a theory of situation awareness in dynamic systems,” *Human Factors*, 37(1), 1995, pp. 32-64.

A. N. Steinberg, C. L. Bowman, and F. E. White, “Revisions to the JDL data fusion model,” in *Proceedings of the NATO IRIS Conference*, Quebec, Canada, October 1998.

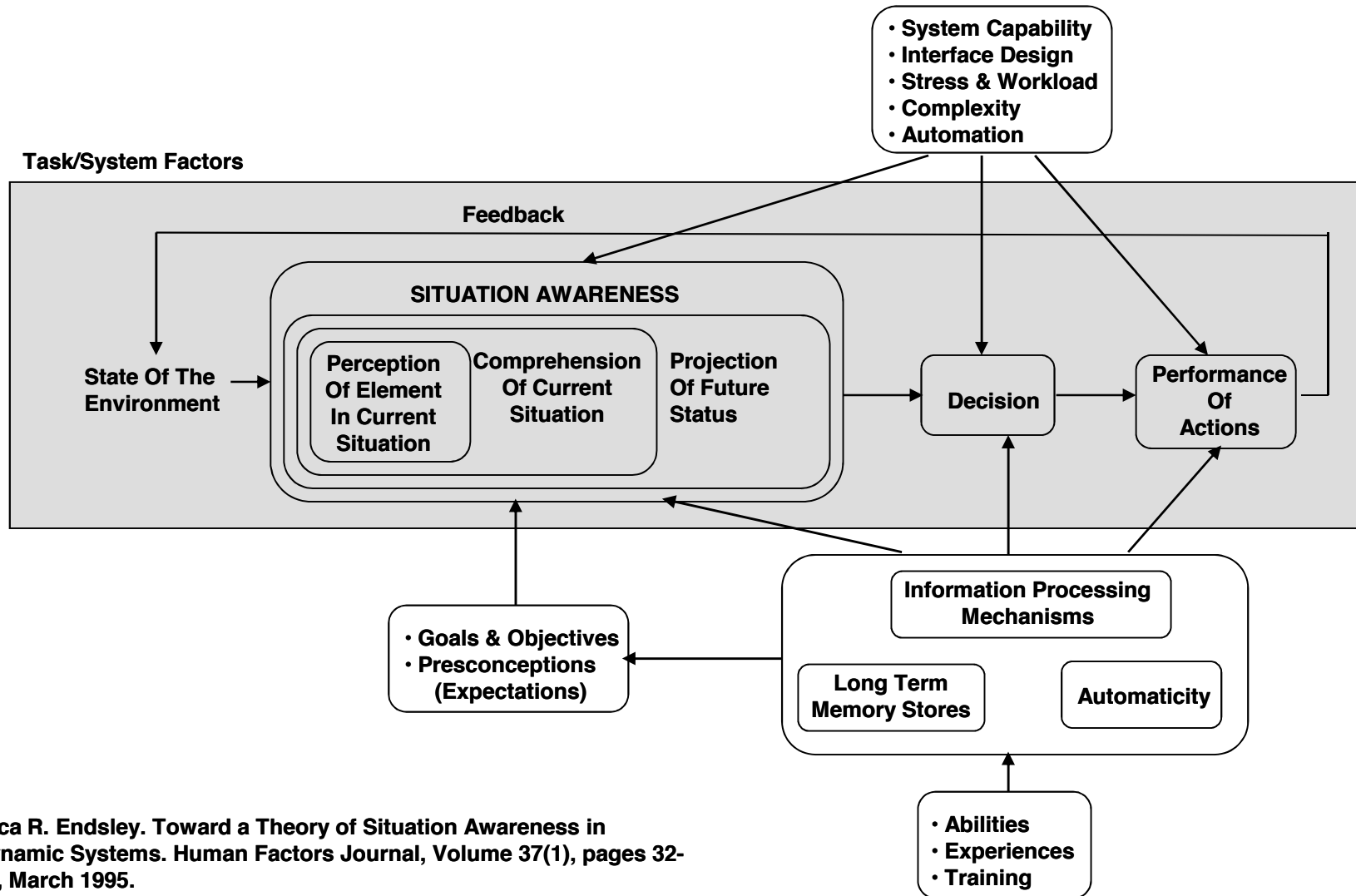
Multi-Agent Systems

- **The paradigm of multi-agent systems (MAS) has its roots in distributed artificial intelligence, object oriented systems and human team cognition.**
- **MAS is currently one of the most powerful approaches used in building distributed computing systems.**
- **MAS has several important features which correspond to our specific interests, particularly:**
 - **Adaptivity: the ability to reorganize and improve behavior with experience**
 - **Autonomy: goal-directedness, proactive and self-starting behavior**
 - **Collaboration: the ability to work with other agents to achieve a common goal**
 - **Inference: the ability to act on abstract task specifications**
 - **Mobility: migration in physical or cyber space**

Current MAS Approach to Situation Awareness

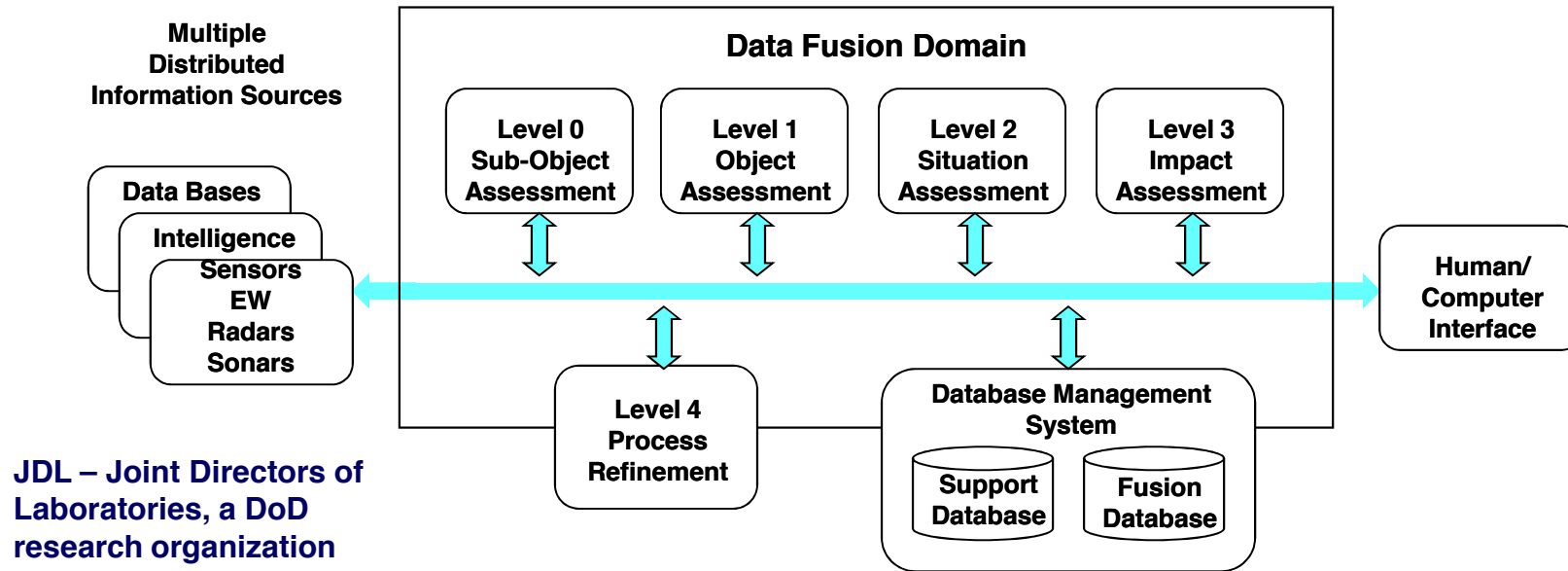
- A typical MAS solution to situation awareness, and consequently to the whole process of command and control, is based on dividing situation awareness (command and control) into several dedicated agents either across functional tasks, e.g. data detection, classification, visualization, etc., or across levels of abstraction of information, e.g. signal, data and semantic information levels.
- Most of the MAS complexity is in the internal agent architecture, the data/knowledge representation and the inference procedures, while inter-agent communication is simplified.
- More sophisticated MAS architectures establish inter-agent communication rules guiding the flow of data and control.
- A few MAS have introduced an ontology-based architecture which allows a semantically deeper data structure, and most importantly, the unifying of conceptually different data representations from different agents.

Endsley's Situation Awareness Model



Mica R. Endsley. Toward a Theory of Situation Awareness in Dynamic Systems. Human Factors Journal, Volume 37(1), pages 32-64, March 1995.

JDL Data Fusion Model



Data Fusion is a process dealing with the association, correlation, and combination of data and information from single and multiple sources to achieve refined position and identity estimates, and complete timely assessment of situations and threats, and their significance. The process is characterized by continuous refinements of its estimates and assessments, and the evaluation of the need for additional sources, or the modifications of the process itself, to achieve improved results.

A. N. Steinberg, C.L. Bowman, and E.F. White, "Revisions to the JDL Model", Joint NATO/IRIS Conference Proceedings, Quebec, October, 1998.

Situation Calculus

- The first formal specification of a situation was given by McCarthy and Hayes in their Situation Calculus, where they used first order logic (FOL) expressions to define a situation as a snapshot of a complete world state at a particular time.
- Since it was computationally inefficient to consider a situation as a complete state of the world, Reiter and Pirri in their approach to situation calculus defined a situation as a sequence of actions enabling calculation of the current state knowing the initial state and the sequence of actions transforming the initial state.
- For example, if s_0 is an initial state and $s' = \text{do}(a, s)$ is an situation resulting from applying action a in situation s , then $\text{do}(\text{put}(A, B), \text{do}(\text{put}(B, C), s_0))$ is a situation resulting in putting block B on block C, and then putting block A on block B.
- Along with “fixed” actions Situation Calculus defines fluents, i.e. such functions and predicates whose value depends on situations, where they are applied.
- Situation Calculus has several well-known associated problems, namely, the Frame Problem (how to describe those aspects of a state, which are not changed by an action), the Ramification Problem (what are the ramifications and side-effects of performing of an action), and the Qualification Problem (what preconditions are required for performing and action).
- As the basis of the situation calculus a programming language GOLOG (aIGOL LOGic) was developed and applied for several planning tasks, e.g. robot planning

Situation Semantics

- **A deviation from a complete world state specification was also argued by Barwise, who looked on situations from the viewpoint of understanding speech acts by “intelligent situated agents”.**
- **Barwise and his colleagues developed situation semantics theory based on FOL. The emphasis of the Barwise theory was not so much in exploring in under what circumstances an utterance is true, but rather what is the semantics (meaning) of speech acts.**
- **In his later work Barwise made an important comment stating that in understanding language, thought and inference it is crucial to handle situations as first class objects that can have properties and stand in relations.**

Situation Control

- Another school of understandings of situations and the use of them in control of large engineering, systems was developed by Pospelov, Klykov and others in Russia in late 70-ies
- Known as situational control theory, it was based on semiotic models of the domain developed in linguistics and language psychology. Semiotics as a science of signs, explores the syntactic, semantic and pragmatic aspects of signs.
- Pospelov considered situations as states of the relations between objects referred to some point in time. The formalism was based initially on a graph theory and finite state machines, and later on formal relational expressions close to FOL.

D. Pospelov, *Situational Control: Theory and Practice*, Nauka, Moscow, 1986

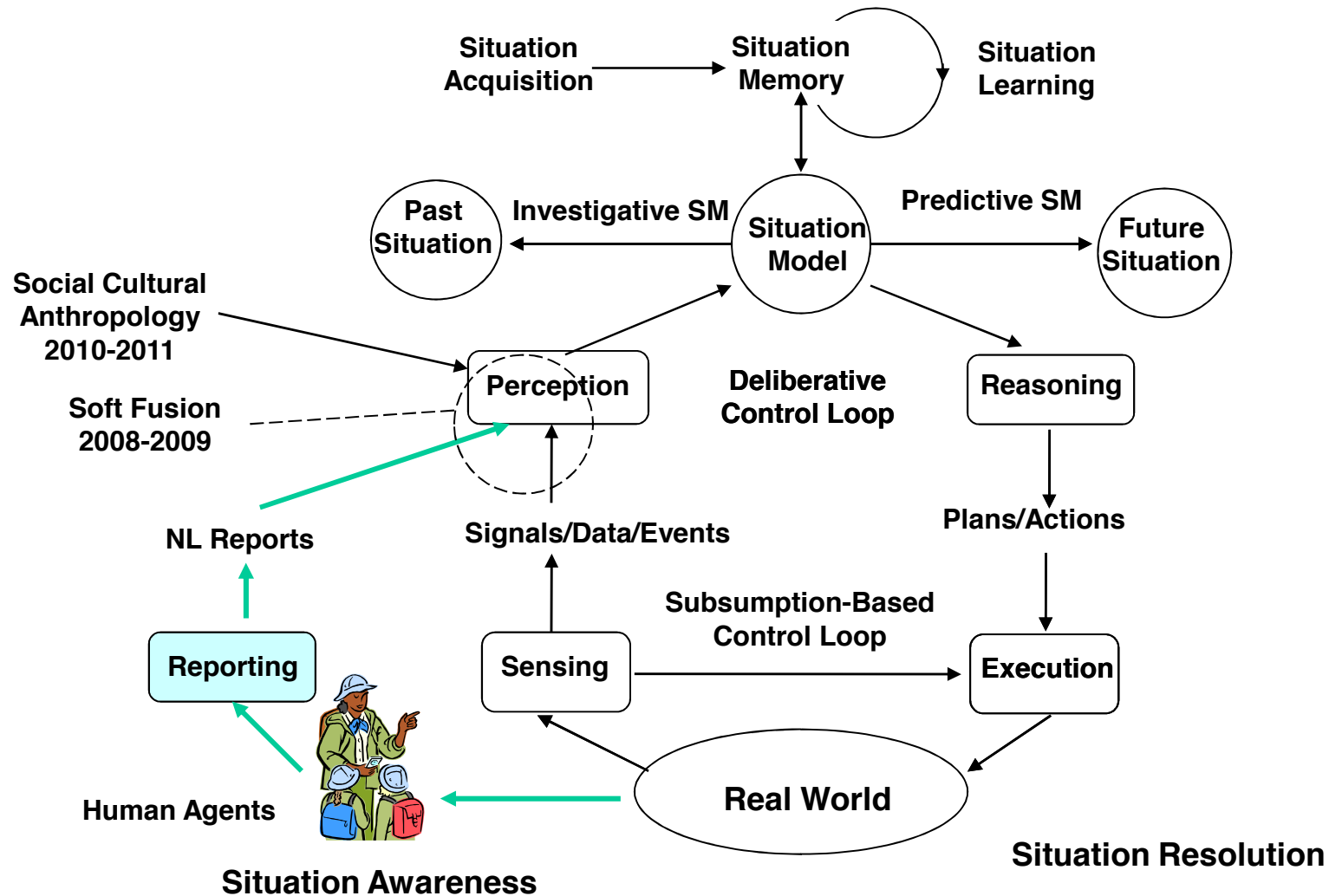
G. S. Osipov, D. A. Pospelov, V.F. Khoroshevsky, A.I. Ehrlich. *Semiotic Modelling in Control Systems*. Proc. 10-th IEEE International Symposium on Intelligent Control, Monterey, California, 1995.

An Ontology-Based Approach

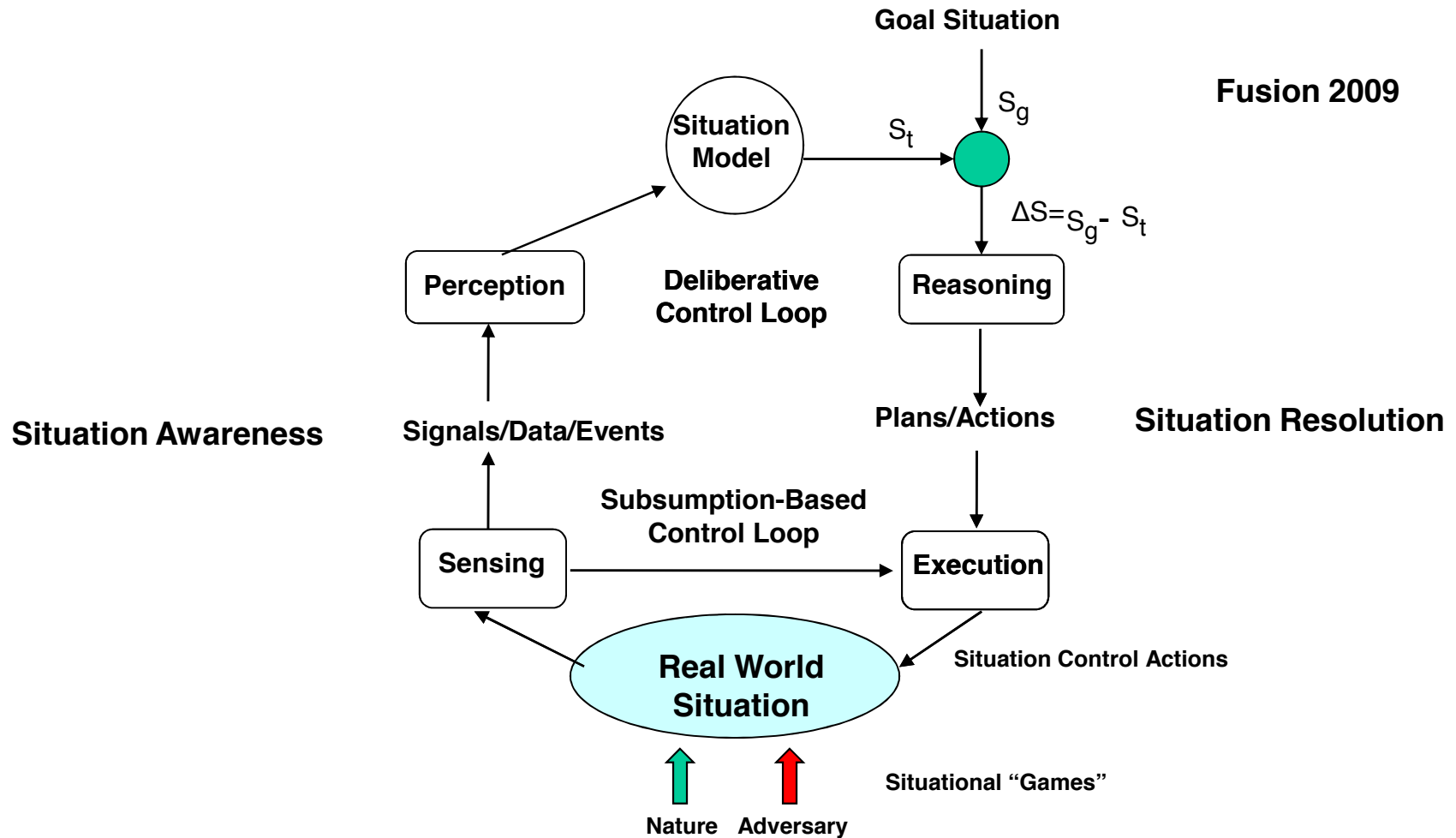
- An ontology-based approach to situation awareness was developed by M. Kokar and his colleagues
- The approach uses formal ontologies to describe events, domain objects and the relations between them, and logical rules to define the process of recognition of situations and situation transitions.
- A set of typical situations, e.g. “under-the-fire situation” were examined to define a core library of situations.

C. J. Matheus, M. M. Kokar, and K. Baclawski, “A Core Ontology for Situation Awareness,” in proceedings of the 6th International Conference on Information Fusion, 2003, PP. 5454-552.

High-Level View on Situation Management



High-Level View on Situation Management: More on Situation Control Aspects



3. Mission Cyber Security Modeling Framework

- **Cyber Attack**
- **Attack Impact**
- **Cyber Terrain**
- **Assets**
- **Services**
- **Missions**
- **Cyber Situations**
- **Impact Dependency Graphs**
- **Principle of Plausible Situations**
- **Asset Similarity**

Cyber Attacks

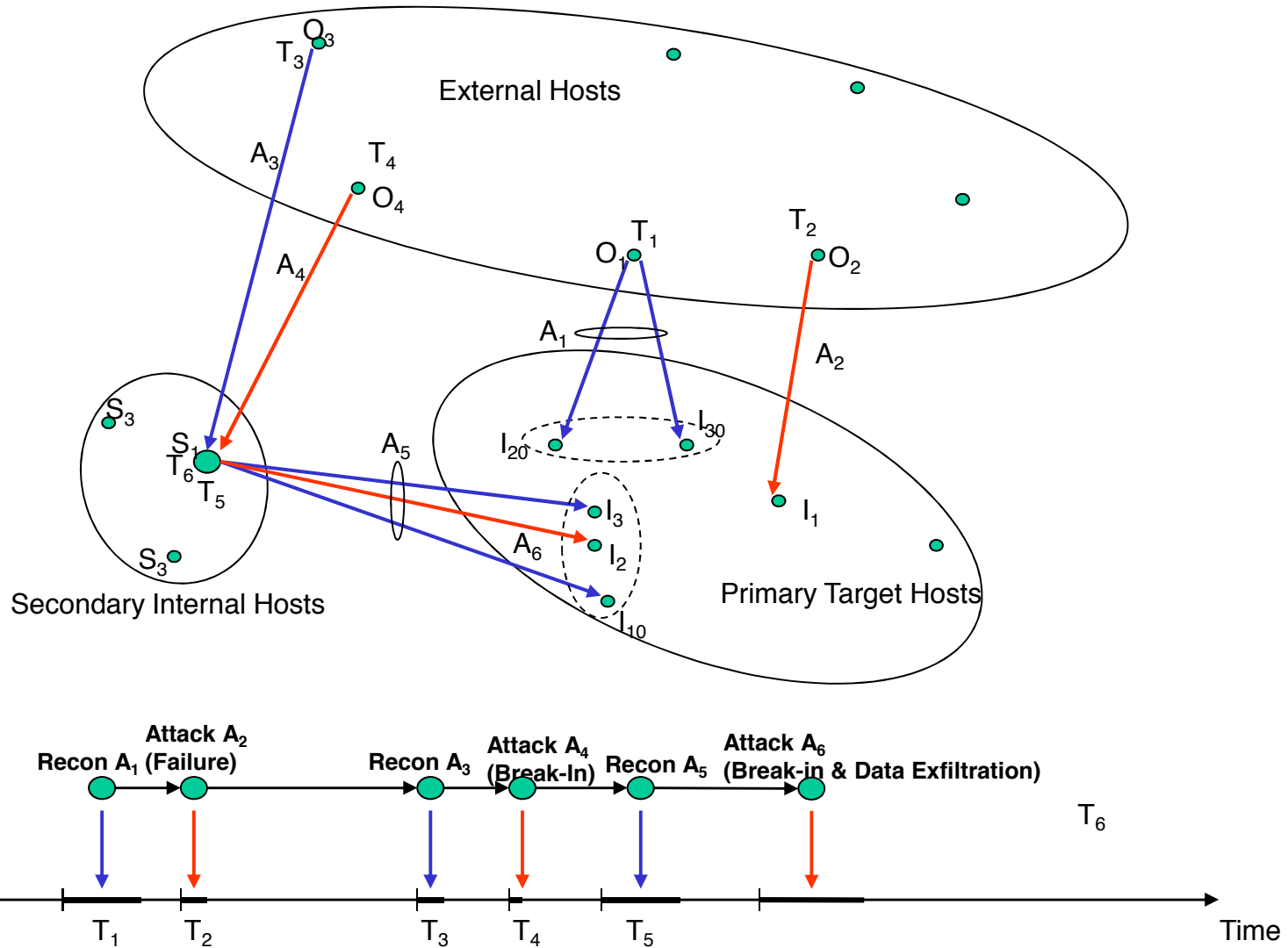
- **Cyber attacks are sequences of actions performed by malicious attackers to gain access to protected information, incapacitate network assets, and ultimately interrupt missions that are running on those networks and assets.**
- **it is important to describe not only the specifics of the actions undertaken by the attacker, but also investigate such factors of the attacks as motivation of the attacker, the time and location of the attack, the different types of actions performed by the attacker during the attack, the tools that the attacker is using, the specific knowledge that the attacker might have about the attacked assets.**
- **We will present these aspects in a model that will be called Cyber Attack Impact Assessment Model (CAIAM)**
- **CAIAM will be described using Conceptual Graph of Sowa***

*J. F. Sowa. Knowledge representation: Logical, Philosophical, and Computational Foundation. Brooks Cole Publishing Co., Pacific Grove, CA, 2000.

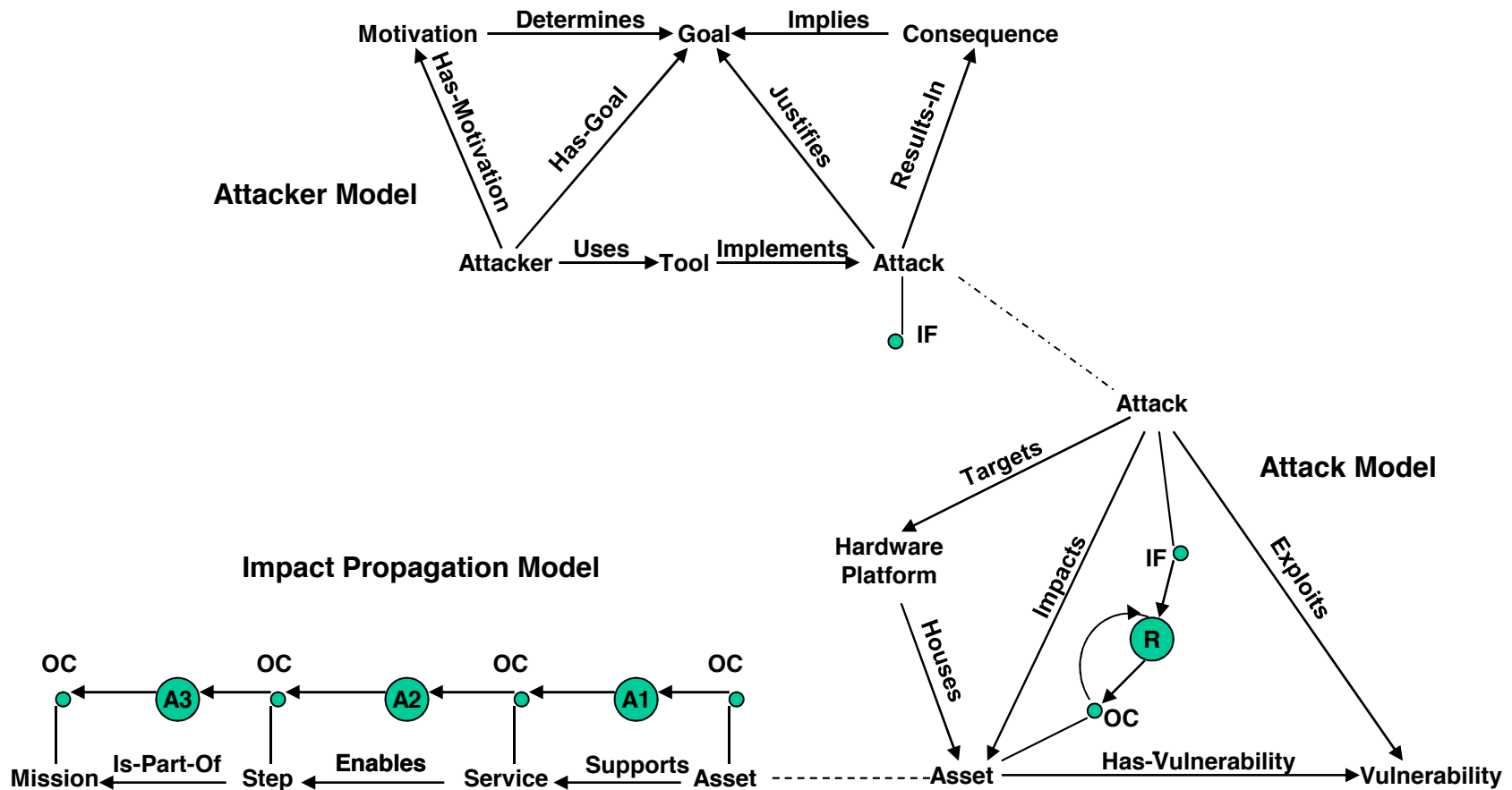
Cyber Attack Step Types

- **Reconnaissance**
 - **Observation** –sniffing, traffic data collection, flow analysis in order to construct the behavioral models of users, the use of assets, services, missions
 - **Targeting** - scanning to determine the IP addresses in the target network that are associated with live hosts; reverse DNS lookups
 - **Vulnerability Identification** - determine certain properties of the host, such as which OS or applications are installed, what versions, patches, etc.
- **Break-in**
 - **Penetration** – exploiting a target by obtaining unauthorized access
 - **Control** – achieving target privileges, usually system administrator privileges
 - **Camouflage** – presenting compromised assets as uncompromised ones, e.g. not launching future attacks from that asset during certain time period
 - **Deception** - cover-up (“noisy”) of attacks to disorient or mislead the security personnel
- **Exploitation**
 - **Embedding** – downloading software that allows re-entering the target, even if the original exploitation has been detected and the target has been re-configured
 - **Data Extraction and Modification** – re-entering the target, collecting data, moving data to another system.
 - **Target Destruction** - incapacitating the target’s operational capabilities, functions and performance

Sample Multi-Step Attack



Cyber Attack Impact Assessment Model



Cyber Attack Impact Assessment Model

- **Attack Model and Impact Propagation Model**
- **Concepts (nodes)**
 - Attack
 - Hardware Platform
 - Asset
 - (Asset) Vulnerability
- **Conceptual Relations**
 - Targets (Attack, Hardware-Platform)
 - Exploits (Attack, Vulnerability)
 - Houses (Hardware-Platform, Asset)
 - Has-Vulnerability (Asset, Vulnerability)
- **Computational Relations**
- **Formally, we will represent cyber attacks as extended conceptual graphs of Sowa however with two important extensions:**
 - first, we will parameterize concepts, and
 - second, we will use computational relations between the parameters of the concepts.

Attack Impact Factor

- **Attack Impact Factor (IF)** of takes its value from an interval [0, 1] and indicates to what degree the attack is capable to compromise the attacked asset.

IF = 0 - attack has no impact on the asset

IF = 1 - attack is capable to destroy the asset by bringing its operational capacity to 0.

- **Computational relation R** between the IF and asset operational capacity (OC) calculates new OC of the attacked asset depending on the existing OC and the IF.
- **Assigning values for IF** is an important knowledge acquisition task, which requires analysis of historic attack data as well consultation with cyber security experts.
- **Common Vulnerability Scoring System (CVSS)** data is used to calculate IF. CVSS has a range of asset vulnerability scores (VS) from 0 to 10, where VS = 0 means that asset is not vulnerable to the cyber attack and VS = 10 means that the asset is most vulnerable to the attack; $IF = VS / 10$.
Alternatively, IF can be computed from IDS alert severity (priority) data

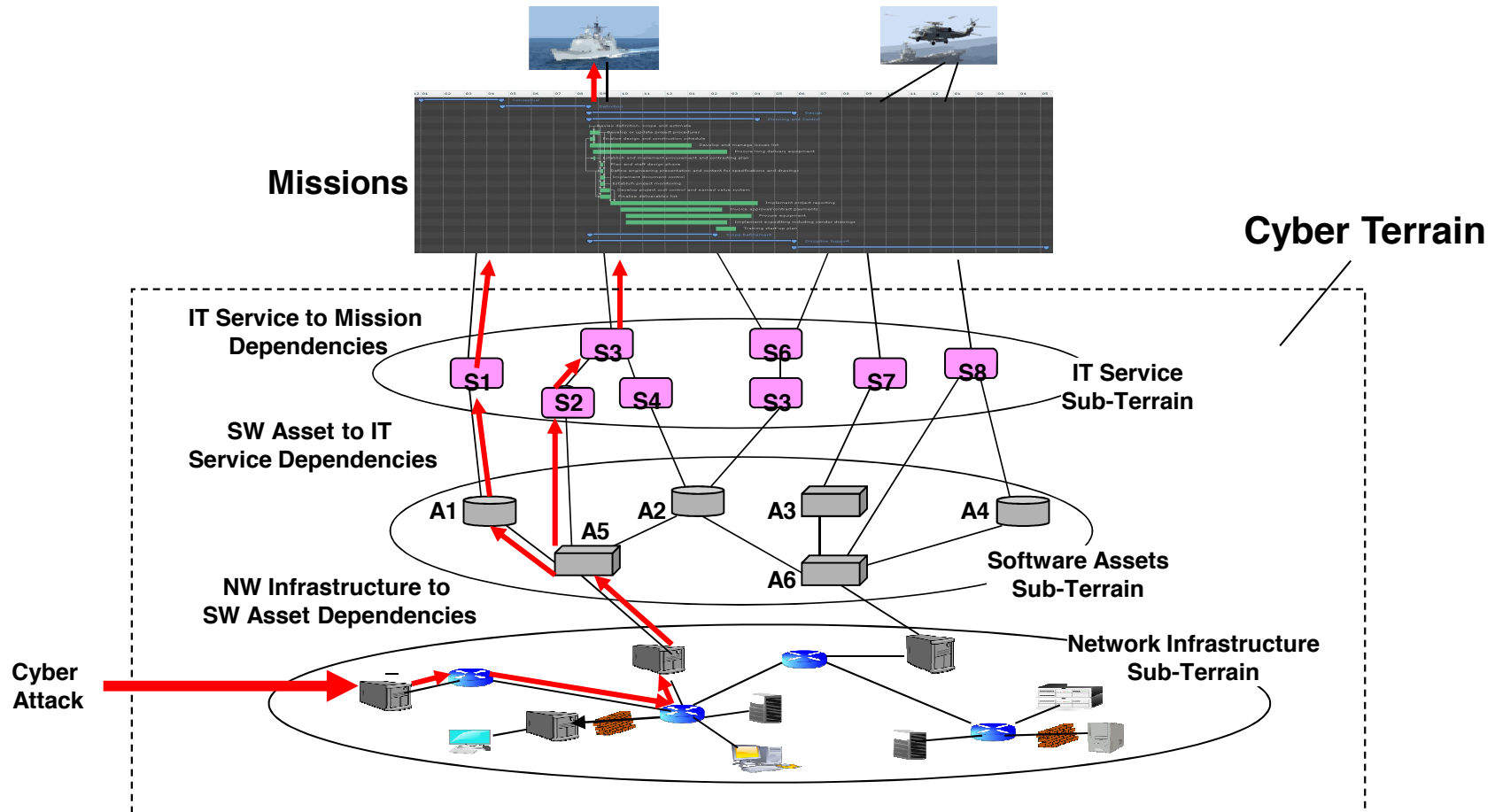
Cyber Terrain

- **We introduce Cyber Terrain (CT) as a complex multi-level information space that describes cyber assets and their inter-dependencies.**
- **It contains three sub-terrains: network infrastructure, software assets, and IT services sub-terrains.**
- **CT is a dynamic information structure: the content of its each element as well the dependencies among its elements might change over the time.**
- **Missions are relying on resources and services provided by the CT. Such dependency of missions on the CT is also a function of time.**
- **Conceptually, it is possible to interpret missions as agents that “act” on CT. While supporting the missions, the CT possesses certain “operational capacity”, i.e. the ability to provide resources and services to the missions with a certain level of quantity, quality, effectiveness, and cost to the missions.**

Operational Capacity

- **Operational Capacity (OC) is ability of a system or its component to produce things, provide services, or accomplish tasks at a given unit of time; OC can be for example, available disk space, number of database transactions, or completed service orders.**
- **Operational Capacity will be measured at an interval $[0, 1]$, where value measures a relative OC to its maximum value. $OC = 0$ denotes that system has lost its total operational capacity, i.e. the system is not operating at all, and $OC = 1$ means that the system operates at its maximum capacity**
- **We will characterize network infrastructure components, software assets, services, and missions with their corresponding operational capacity**

Illustration of a Cyber Terrain



Sub-Terrains

- **Network Infrastructure Sub-Terrain**
 - Collection of connected network infrastructure components like routers, servers, switches, firewalls, communication lines, terminal devices, sensors
 - Dependencies: connectivity, containment, location, and other relations represent the logical topology of the Network Infrastructure sub-terrain.
- **Software Asset Sub-Terrain**
 - Operating systems, middleware, applications
 - SW might be characterized by attributes like functionality, vendor, release number, vulnerability, etc.
 - SW sub-terrain defines different types of dependencies among the components, e.g. containment, data dependency, control dependency, support
- **Service Sub-Terrain**
 - On-line data service, file transfer, video service, e-mail, GIS, GPS, security
 - Dependencies between two services include: service enabling and containment
 - Services are dependent on assets that are delivering them
 - Service operational capacity is determined by the assets that they are dependent upon.
 - Often service quality is defined in so-called service-level agreements (SLAs)

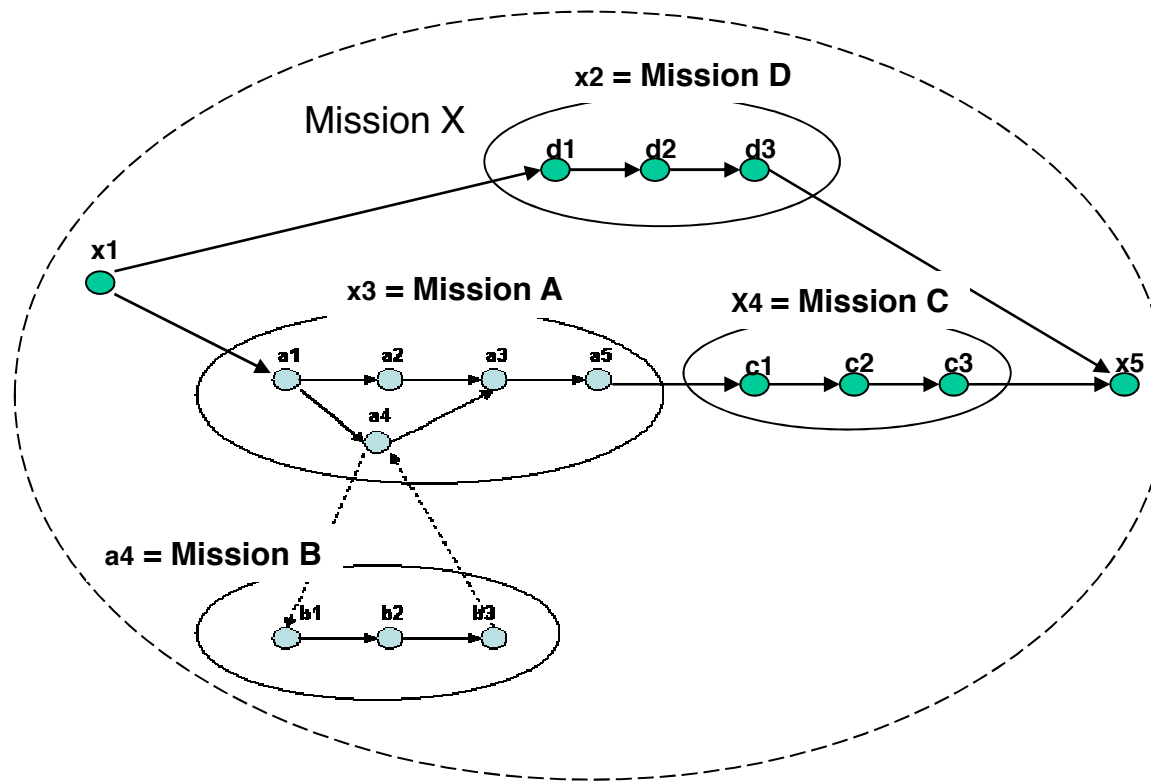
Dependencies Between Sub-Terrains

- **As among the components of a sub-terrain the dependencies exist between the sub-terrains : a Network Infrastructure sub-terrain component may “house” zero or more SW sub-terrain components and a SW sub-terrain component may enable services in the service sub-terrain.**
- **The provisioning of services follows service level agreements that identify what services, under what constraints and during what time are providing support to the steps of a mission.**
- **CT is a dynamic information structure: its components and their inter-dependencies are a function of time.**

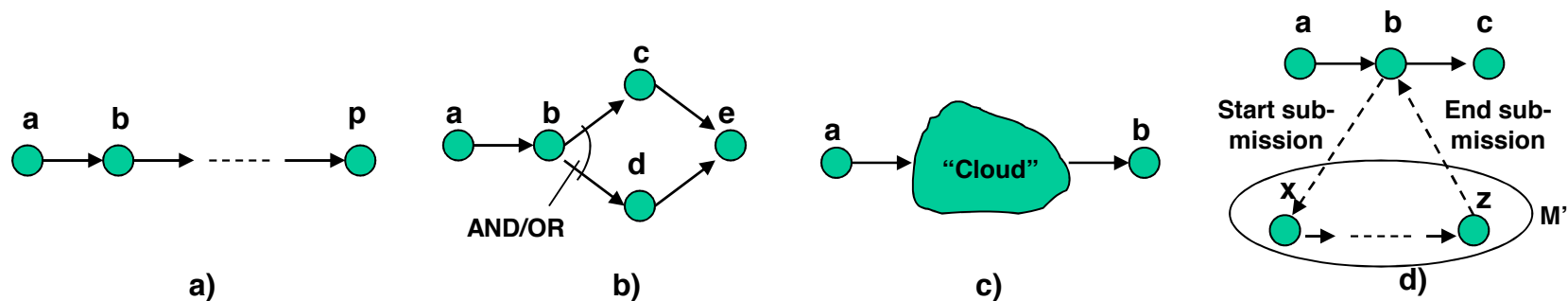
Missions

- **Missions are goal-directed time-phased flows of mission steps executed by human or machine agents. Each mission step can be another step, a mission, or a task, i.e.**
mission := (step) ...
step := step | mission | task
- **Mission task is the terminal element of a mission: it is the concrete procedure (algorithm) that has to be implemented during the execution of a mission step. It is possible that one and the same task will be implemented twice during two different mission steps.**
- **Mission steps and missions are time-dependent entities; they have their start time, duration, and end-time. The start time of a mission is the start time of the first steps belonging to the mission, and the end of the mission is the end time of the last step of the mission.**

Mission Flow Graph

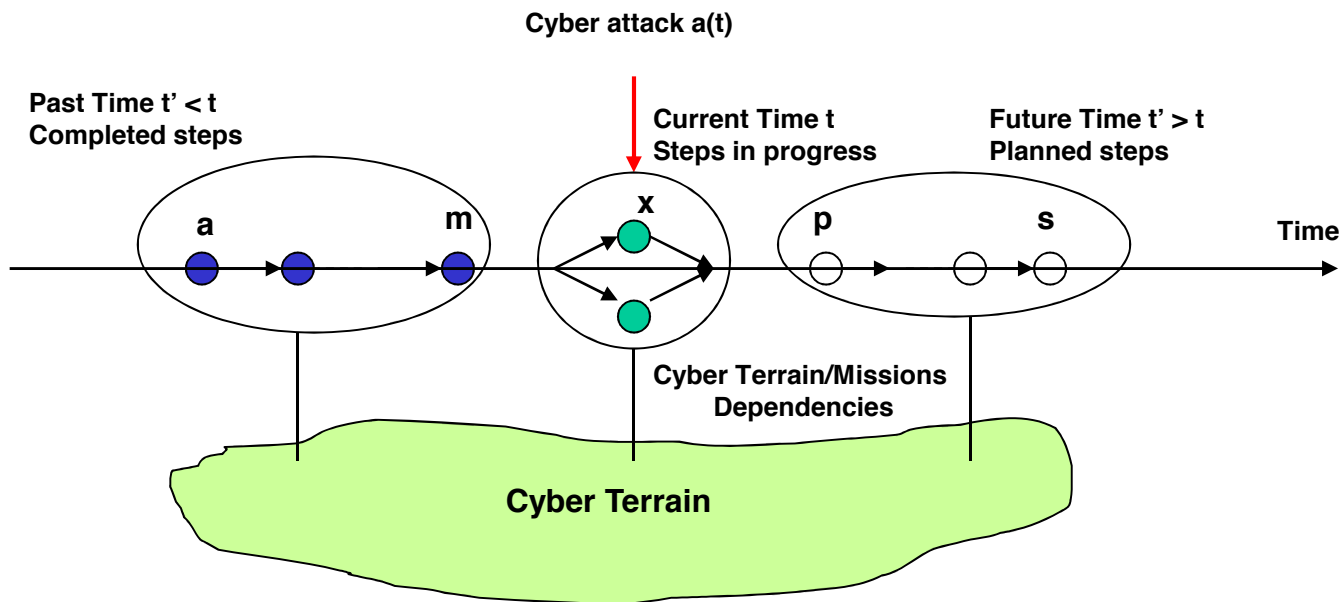


Time Dependency in Mission Modeling



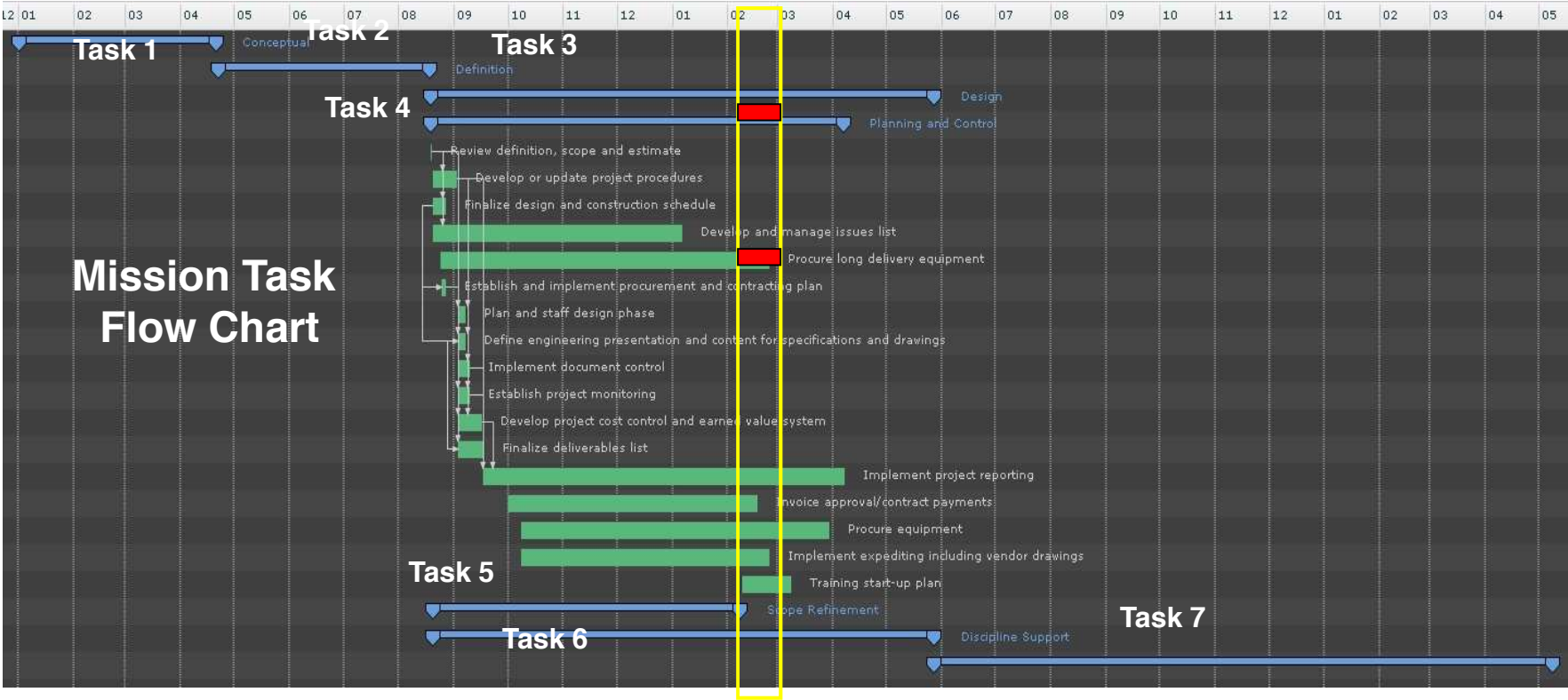
- a) **Sequential flow of mission steps: mission depends on all steps that are executed in a linear order**
- b) **Flow diagram contains parallel branches that that can be forked either by AND or OR-nodes. The AND-node requires that both branches should be executed, while the OR-node prescribes that at least one branch should be taken.**
- c) **Flow diagram, where all steps from a “cloud”, a set of tasks should be taken without any particular order.**
- d) **A sub-mission is defined by a step in a higher-level mission.**

Time-Dependent Mission States



- From mission monitoring viewpoint at each particular time a mission step could be in one of the three different states
 - Completed
 - In progress
 - Planned for execution
- State of the mission depends on the state of the mission tasks

Mission Task Flow Chart

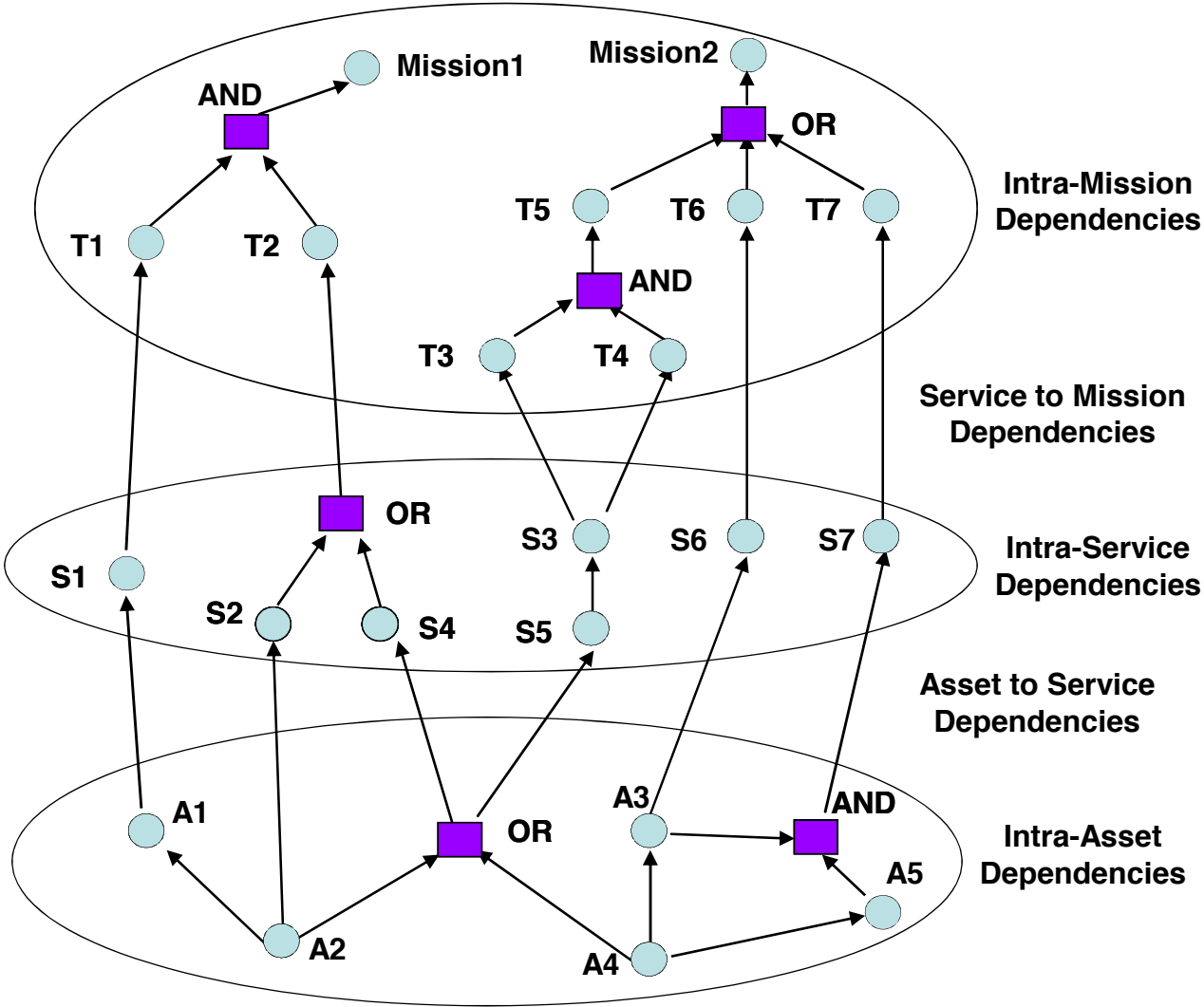


Mission Task Flow Chart

Impact Dependency Graph

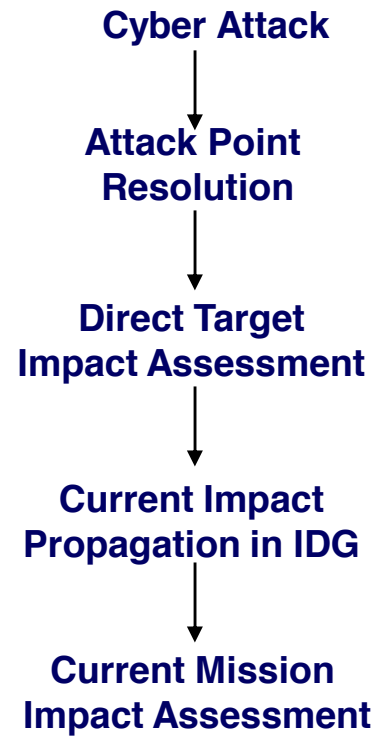
- **Impact dependency graph (IDG) is a mathematical abstraction of the dependencies that exist among cyber assets, services, mission steps and missions.**
- **IDG nodes include cyber assets, services, missions steps and missions, as well two types of special nodes: AND-nodes and OR-nodes that represent logical dependencies among nodes in IDG.**
- **AND-node defines that the parent node depends on all of its children nodes, while the OR dependency defines the required presence of at least one child node. The OR dependency is introduced to capture system redundancy or for alternative functionality, performance, cost, reliability or for some other reason..**
- **The structure of IDG is derived from the Cyber Terrain and from the structural composition of missions. The arcs in IDG represent relations “Node X Depends-On node Y”.**
- **For the purpose of calculating the cyber attack impact propagation through IDG we can abstract from the specific semantics of assets, services and missions, and consider them as generic nodes in IDG.**

Impact Dependency Graph



4. Real-Time Cyber Attack Impact Assessment

Real-Time Mission Impact Assessment Process



Attack Point Resolution

- **Not every cyber attack causes damage. The attack might succeed if the following logical condition holds:**

IF (Attack C targets hardware platform H)

AND (Hardware platform H houses software asset A)

AND (Asset A has vulnerability V)

AND (Attack C exploits vulnerability V)

THEN (Attack C succeeds in impacting the asset A)

- **The above-given logical condition is defined over binary relations that are established between hosts, assets, attacks and vulnerabilities per the cyber attack model. These relations can be formulated as a system of logical constraints in terms of logic constraint programming:**

C1: Alert_Constraint (SID, IP)

C2: Network_Constraint (IP, Asset_ID)

C3: Asset_Constraint (Asset_ID, Vulnerability_ID)

C4: Vulnerability_Constraint (SID, Vulnerability_ID)

Cyber Attack Constraint Resolution

C1: Alert_Constraint (SID, IP)

C2: Network_Constraint (IP, Asset_ID)

C3: Asset_Constraint (Asset_ID, Vulnerability_ID)

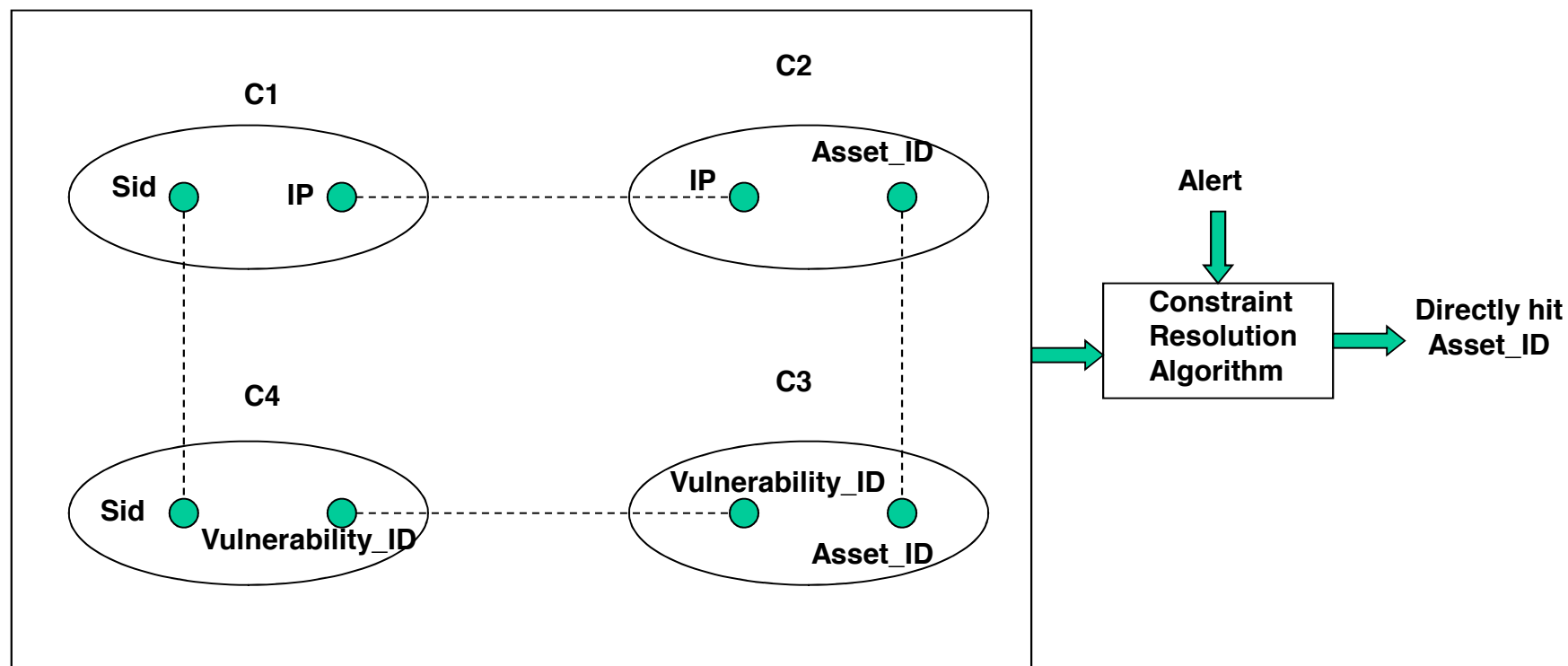
C4: Vulnerability_Constraint (SID, Vulnerability_ID)

IDS alert file

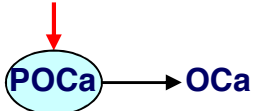
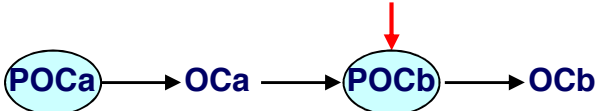
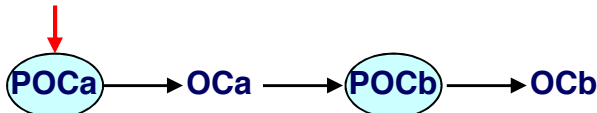
Network configuration database

Asset configuration database

Asset vulnerability database



Calculation of Asset's Operational Capacities

	Terminal SW asset	Non-terminal SW asset
Direct hit	$POC_a(t') := \text{Max} [POC_a(t) - IF_x(t'), 0]$ $OC_a(t') := POC_a(t')$ 	$POC_b(t') := \text{Max} [POC_b(t) - IF_x(t'), 0]$ $OC_b(t') := \text{Min} [OC_a(t'), POC_b(t')]$ 
Indirect hit	Not possible situation	$POC_b(t') := POC_b(t)$ $OC_b(t') := \text{Min} [OC_a(t'), POC_b(t')]$ 

$OC_a(t)$ - operational capacity of asset a at time t

$POC_a(t)$ – permanent operational capacity of asset a at time t

$IF_x(t')$ - impact factor of cyber attack x at time t' , $t' > t$

POC describes a permanent damage caused to an asset by an attack. POC is an internal feature of a software asset only. POC stays unchanged until either its value is reduced by the next direct cyber attack, or it can be changed by a human (usually to reset $POC = 1$).

Attack Impact Propagation

- While calculating attack impact propagation through IDG we can abstract from the specific semantics of assets, services and missions, and consider them as generic nodes in IDG along with AND and OR-nodes.
- During the attack propagation from the terminal nodes (the asset nodes) the operational capacities of all dependent nodes will be calculated. The node that is in a linear path in the IDG gets the operational capacity from its child node
- The operational capacities of the AND and OR nodes are calculated as follows:

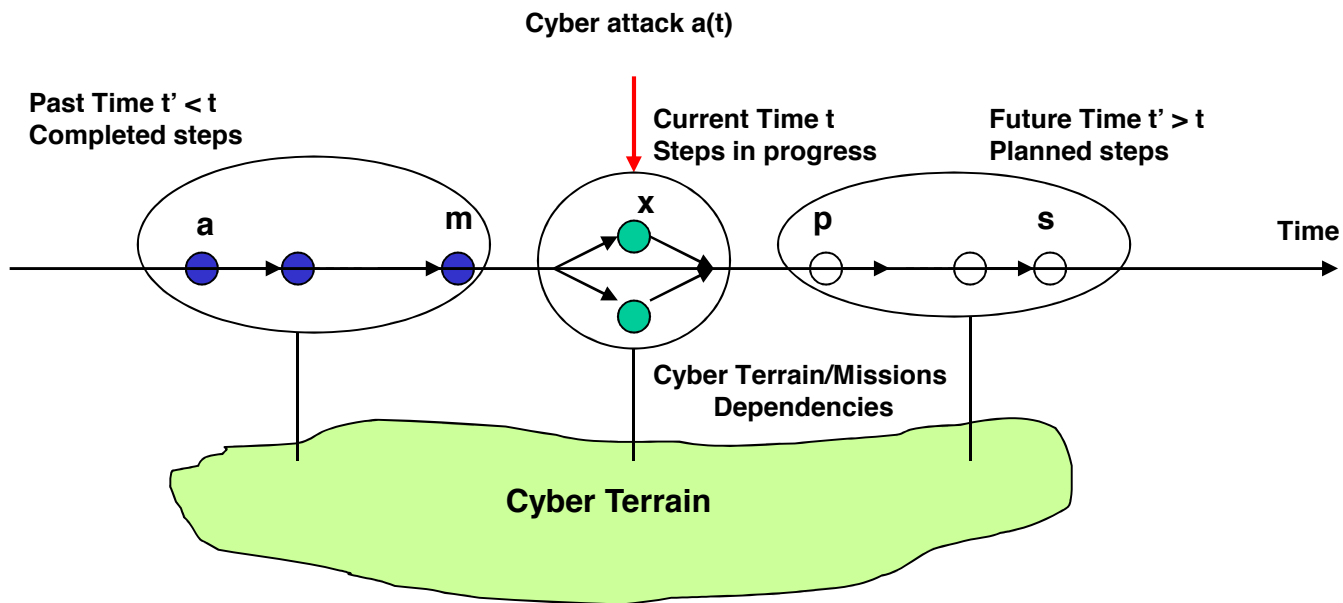
$$\begin{aligned} OC_{OR(t)} &= AVE(OC_{1(t)}, OC_{2(t)}, \dots, OC_{n(t)}) \\ OC_{AND(t)} &= MIN(OC_{1(t)}, OC_{2(t)}, \dots, OC_{n(t)}), \text{ where} \end{aligned}$$

$OC_{OR(t)}$ is the operational capacity for an OR-node

$OC_{AND(t)}$ is the operational capacity for an AND-node

$OC_{1(t)}, OC_{2(t)}, \dots, OC_{n(t)}$ are the operational capacities of the child nodes for the OR and AND nodes.

Time-Dependent Mission States



- From mission monitoring viewpoint at each particular time a mission step could be in one of the three different states
 - Completed
 - In progress
 - Planned for execution
- State of the mission depends on the state of the mission tasks

Mission Impact Assessment

- **Cyber attack impact on a mission depends on what state the mission steps are at the moment when the attack occurred:**
 - If a cyber attack impacts assets and services that support steps that have been already completed then the impact of the attack should be irrelevant as far as these steps are concerned
 - The ongoing steps at the cyber attack will be affected by the attack
 - The mission steps that are waiting their execution at the moment of the attack will not be accounted in the calculation of the operational capacity of the overall mission
- **Strategies to handle to be affected mission steps:**
 - Calculation of potential impacts to be affected steps
 - Reconfigure cyber terrain
 - Reconfigure mission

Mission Impact Calculation

- Since mission is a process that unfolds step-by-step as time progresses, its operational capacity is getting its starting value $OC=1$, and then it is steadily decreasing depending on the operational capacities of its executed steps.
- The operational capacity of a mission for all types of the mission step, except the OR-ed parallel flow is calculated as follows:

$$OCA(t) = OC1(t) \times OC2(t) \times \dots \times OCn(t), \text{ where,}$$

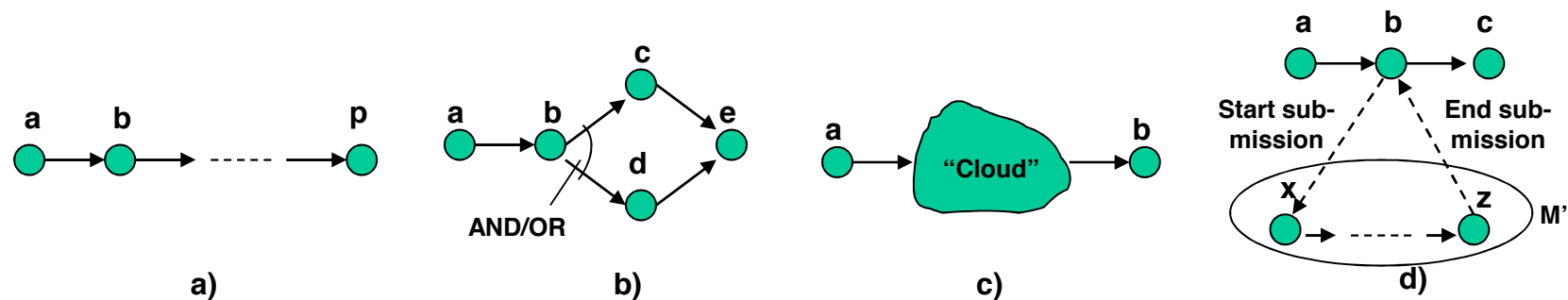
t is the time of a cyber attack

$OCA(t)$ is the operational capacity of mission A

$OC1(t), OC2(t), \dots, OCn(t)$ are the operational capacities of all tasks of mission A that where under execution at a time t of a cyber attack.

- For the OR-ed flow of mission steps the calculation of the mission impact depends on what parallel branch of the mission step flow is taken by the mission monitoring system. .

Time Dependency in Mission Modeling



- a) **Sequential flow of mission steps: mission depends on all steps that are executed in a linear order**
- b) **Flow diagram contains parallel branches that that can be forked either by AND or OR-nodes. The AND-node requires that both branches should be executed, while the OR-node prescribes that at least one branch should be taken.**
- c) **Flow diagram, where all steps from a “cloud”, a set of tasks should be taken without any particular order.**
- d) **A sub-mission is defined by a step in a higher-level mission.**

5. Assessment of Plausible Future Cyber Situations

Introduction

- **In recent years several approaches are emerging to detect cyber attack plans and predict future attacks, including**
 - **probabilistic reasoning (Valdes & Skinner 2001, Goldman 2001)**
 - **statistical alert analysis (Qin & Lee 2004)**
 - **clustering algorithms (Debar & Wespi 2001)**
 - **methods based on causal network analysis (Qin & Lee 2004)**
 - **cyber attack condition matching (Cheung, Lindqvist & Fong 2003)**
- **All these approaches are based on the analysis of cyber attack patterns, either focusing on internal features of the attack patterns, or on causality between the attack patterns.**
- **We will propose an alternative approach on analyzing future attacks and their impacts on cyber assets, services and missions based on the notion of plausible situations**

Related Work on Plausible Future Analysis

- The area of research of plausible cyber security situations is in its infancy and only few research results can be cited. For example, a plausible futures analysis has been conducted, where plausible future situations are projected using information from unfolding multi-step cyber attacks (Li & Lei 2007). In (Holsopple & Yang 2008) a plausible future is defined as “an event that extends from a current progression of events”.
- The plausibility of events is measured by a plausibility score, where this score is interpreted as how strongly the current evidence suggests that the given object could be acted upon.
- The plausible futures model has been used in other domains like judging on economic developments, potential future political situations, and spreading of diseases (Centers for Disease Control and Prevention, 2008).

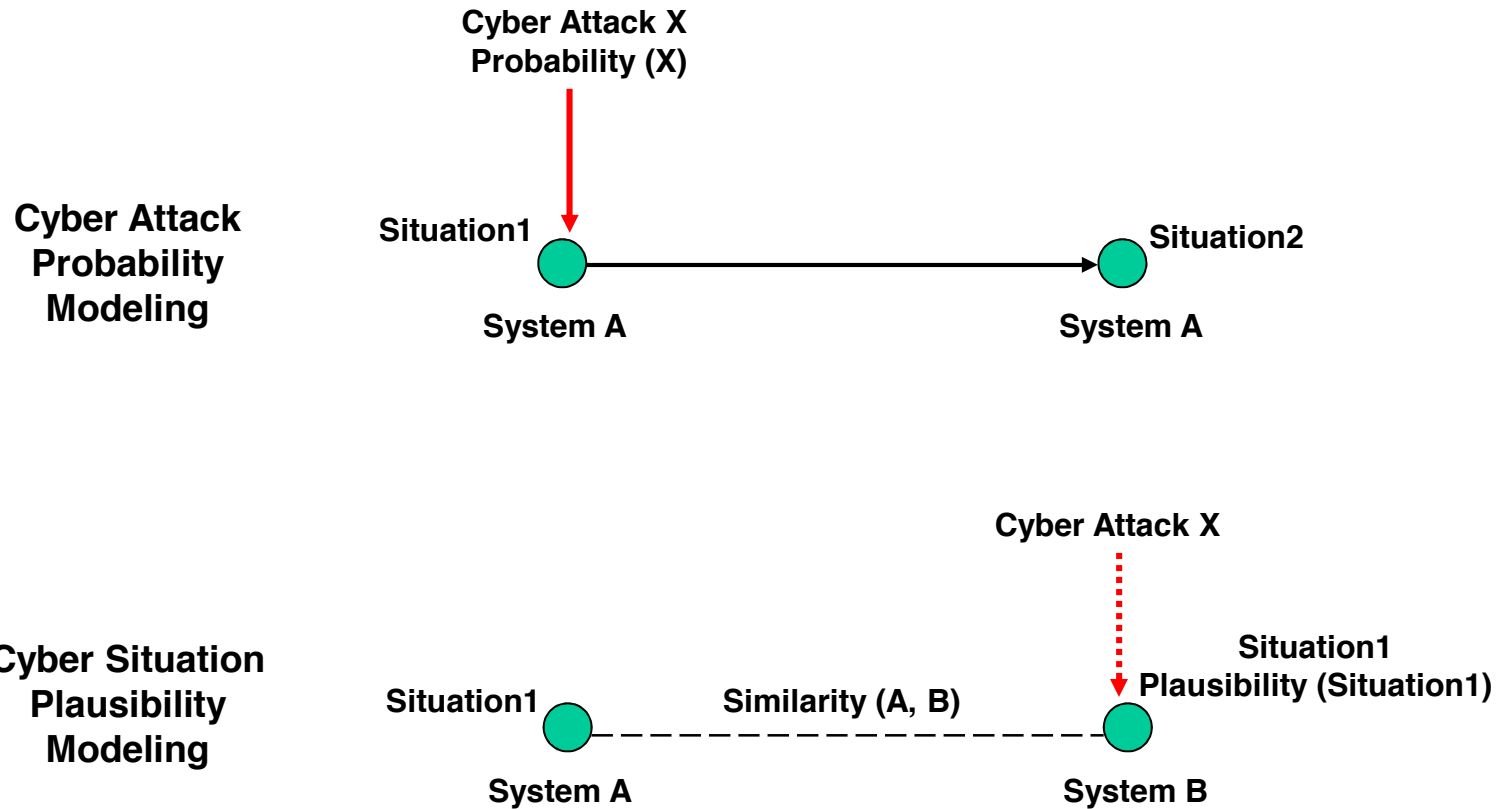
Plausible Situations

- **Notion of Plausible Future Situation.**

Situation in a dynamic system is called plausible if with some degree of likelihood it could happen in future

- **Our premise is that if some cyber security situation happened, e.g. certain asset was compromised due to a cyber attack then the detected cyber security situations could happen in future with other asset that to some degree of likelihood is similar to the already attacked cyber asset.**
- **The semantics of the relation “similar” can be fairly wide, like functional, configuration, location and usage similarity. Although in this paper we will focus on plausible cyber security situations the presented approach can be also applied to physical attacks, as well to impacts caused by not-attack type events, e.g. impacts caused by system internal faults, operator errors, or natural events.**

Cyber Attack Probability Modeling Vs. Cyber Situation Plausibility Modeling



Temporal Fuzzy Sets and Situations

- Temporal Fuzzy Set

Let X be a universe of elements x , T be a discrete time of points t , and $[0, 1]$ be an interval of real numbers, then a temporal fuzzy set is defined as (X, f, T) , where $f: X \times T \rightarrow [0, 1]$. We will interpret the value of $f(x, t)$ as “likelihood that at an observed time moment t the element x belongs to X ”.

- Temporal Fuzzy Situation

Let S be a set of situations, then fuzzy temporal situation (S, f, T) will be defined as a class of temporal fuzzy set over S .

- One can define many specific classes of temporal fuzzy situations depending of domain semantics. For the cyber security domain we will introduce a class of situations based on the notion of predicate “compromised”.

Fuzzy Predicate “Compromised”

- We will introduce a fuzzy predicate *Compromised* ($a, h(a, t)$) that defines a temporal fuzzy set (A, h, T) of compromised entities A , where $a \in A$ and $h: A \times T \rightarrow [0, 1]$. The value of $h(a, t)$ represents a degree of likelihood that at time t entity a is compromised, particularly

$h(a, t) = 0$ - entity a is completely compromised

$h(a, t) = 1$ - entity a is not compromised at all

- In case of assets, the compromise level could be expressed by different characteristics of assets, e.g. level of internal resources, asset availability, performance, etc. In this work we use the asset’s operational capacity (OC) as an aggregated measure of its compromise level $h(a, t)$.
- We can use the above-given definition for defining compromised services and missions. The difference is in the semantics: an asset is compromised because it got hit by a cyber attack; while the missions and services become compromised only because the services depend on compromised assets or other services, and missions depend on services.
- Predicate *Compromised* ($a, h(a,t)$) represents a fuzzy situation

Asset Similarity

- Asset Similarity

Let A be a set of assets, and $a, b \in A$. Let's introduce fuzzy predicate *Similar* $((a, b), q(a, b))$, which defines a fuzzy relation (R, q) between similar cyber assets,

$$R \subseteq A^2, q: R \rightarrow [0, 1]$$

$q(a, b)$ - degree (strength) of similarity between assets a and b .

$q(a, b) = 1$ - assets are identical

$q(a, b) = 0$ - assets are totally dissimilar.

- We will describe several different interpretations of the fuzzy relation *Similar*, including structural, functional, and location similarity.

Plausible Situations

- Plausible Situations.

Let S be a set of situations in some domain, and $s \in S$. We will introduce a predicate *Plausible* ($s, p(s)$), which defines a fuzzy set (S', p) of plausible situations, where $S' \subseteq S$, $p: S' \rightarrow [0, 1]$, and $p(s)$ is a degree of our confidence that situation s belongs to the set of plausible situations S' . In other words, $p(s)$ is a confidence factor (CF) that the situation s is plausible.

The Principle of Plausible Future Situations (PFS)

For any assets $a, b \in A$

Compromised ($a, h(a, t)$) & Similar ($(a, b), q(a, b)$)

Plausible (Compromised ($b, h(b, t') = h(a, t)$), $p(b, t') = q(a, b)$), $t' > t$)

The PFS principle states that if at some time moment t an asset a is compromised to the level of $h(a, t)$ and the strength of similarity between the assets a and b is equal to $q(a, b)$, then the plausibility that the asset b could be compromised at a future time $t' > t$ to the same level that the asset a was compromised is equal to the level (to the strength) of similarity between the assets a and b .

Example: Using PFS

- Let's assume that at some time t a database was hit by a cyber attack with an impact factor 0.3, which, by exploiting a vulnerability of the database reduced its operational capacity from the original value 1.0 to $1.0 - 0.3 = 0.7$.
- It is known that in the targeted network in some other host resides about the same database, however a different release. Let's declare that due to the difference in releases the similarity level of the databases is 0.85.
- Application of the PFS allows us to come up with a conclusion that some time in future (not exactly when) it is plausible with certainty 0.85 the other database could loose its operational capacity to the level of 0.7.

Adjusted Operational Capacity

- The confidence factor CF defines the level how strongly we believe that some situation is plausible. In cases when the situation itself is a fuzzy entity and is measured by some level of truthfulness, e.g. by the operational capacity (OC) level of an asset, it is reasonable to combine the confidence factor CF and the operational capacity OC into one adjusted operational capacity (AOC), e.g. using the following simple formula:

$$\text{AOC} = \text{CF} \times \text{OC}$$

- For the example that we gave above, the AOC for the other database will be $0.88 \times 0.7 = 0.595$. Throughout of this work we will operate with the adjusted operational capacity while calculating the plausible impacts on missions.

Asset Similarity Classes

- (Vulnerability-Similarity (x, b) , q_{cs})
 - Are A and B the same products?
 - Do A and B have the same configuration?
 - Do A and B have the same version (release)?
 - Do A and B have the same vulnerabilities?
- (Location-Similarity (x, b) , q_{ls})
 - Do A and B belong to the same sub-net?
 - What is the geographic distance between A and B?
- (Functional-Similarity (x, b) , q_{fs})
 - Do assets A and B perform the same function, e.g. both are DBMS
- (Temporal-Similarity (x, b) , q_{ts})
 - Are assets operational during coinciding time intervals?
- (Mission-Similarity (x, b) , q_{ms})
 - Do assets support the same mission?
- (Usage-Similarity (x, b) , q_{us})
 - Do A and B have the same usage pattern (traffic)?
 - Do A and B are characterized by anomalous traffic between corresponding nodes?

Vulnerability-Based Asset Similarity Calculation

Vulnerabilities from Open Source Vulnerability Database (OSVDB).

OSVDB records identify the versions of a vendor/product that have the same vulnerability.

For example, vulnerability # 22919 "Oracle Database XML Database DBMS_XMLSCHEMA_INT Multiple Procedure Remote Overflow" affects the product/versions from the vendor "Oracle Corporation" (Table 1).

Analysis of software configurations among different product #, releases and versions and introduction of a metrics to estimate closeness of products allows construction of vulnerability-based similarity function q_{vs} (Table 2).

Each product has a unique vulnerability identifier, even if some other product has the same vulnerability.

A products might have multiple vulnerabilities.

Table 1

Product	Product #	Release	Version
Database	10g	2	10.2.01.1
Database	10g	1	10.1.0.3 10.1.0.4 10.1.0.5 10.1.0.4.2
Database	9i	2	9.2.0.6 9.2.0.7
Database	8i	3	8.1.7.4
Database	9i	1	9.0.1.4 9.0.1.5 9.0.1.5
Database	8	8.0.6	8.0.6.3

Table 2

Similarity Class	Product	Product #	Release	Version	q_{vs}
1	1	1	1	1	1.0
2	1	1	1	0	0.9
3	1	1	0	0	0.75
4	1	0	0	0	0.5
5	0	0	0	0	0.0

Asset Similarity Index

Two assets might be related with multiple similarity relations, and in order to calculate a combined affect of them, we will introduce the notion of asset similarity index (ASI). ASI index is calculated using a formula of combining two evidence factors*

$$E(x, y) = E(x) + E(y) \times (1 - E(x))$$

Let's assets x and y have the following similarity scores:

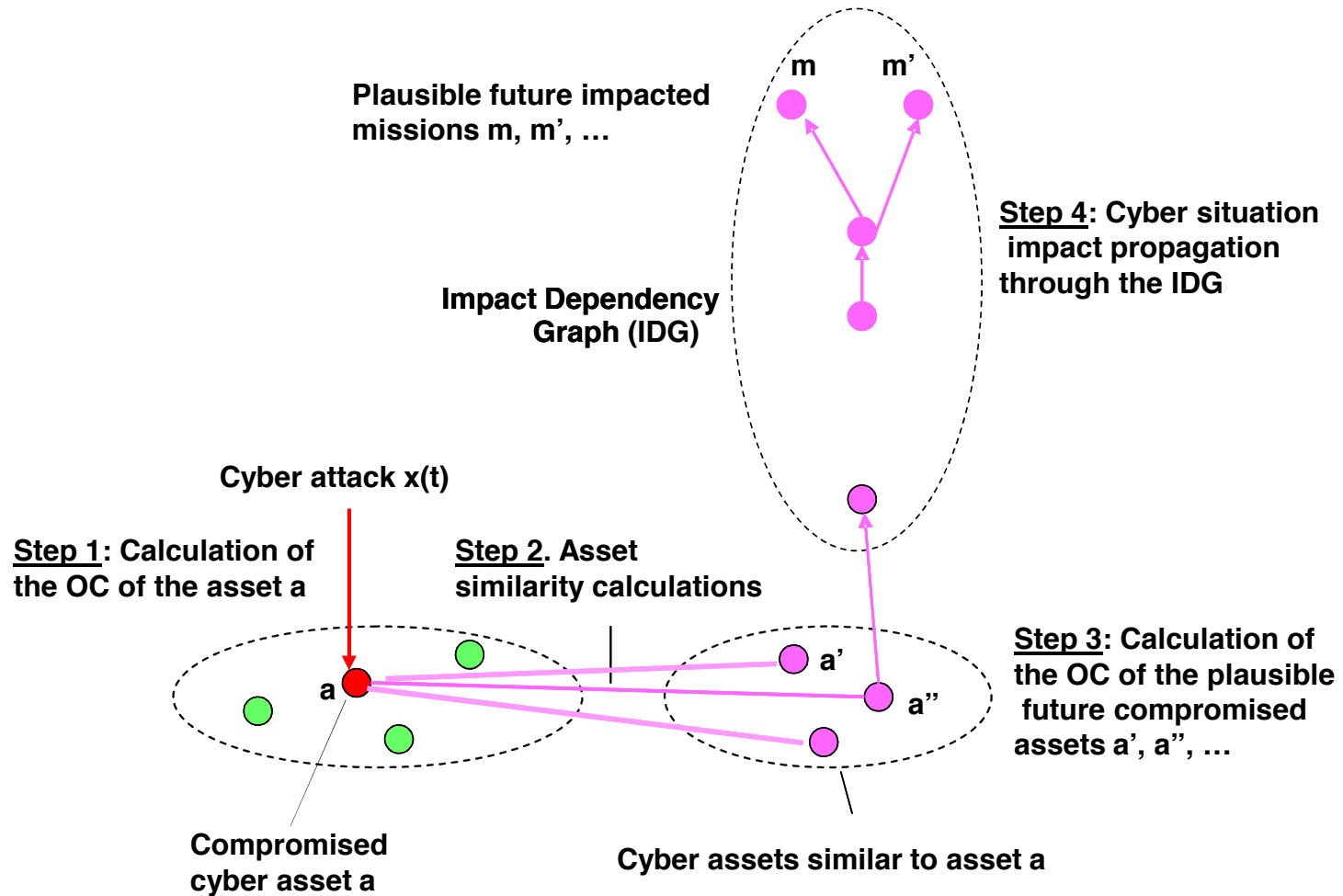
Vulnerability-Similarity qvs = 0.8
Configuration-Similarity qcs = 0.8
Location-Similarity qls = 0.5
Functional-Similarity qfs = 0.5

Successive calculation of the scores leads to the following results:

$$\begin{aligned}0.8 + (0.8 \times (1 - 0.8)) &= 0.96 \\0.96 + (0.5 \times (1 - 0.96)) &= .98 \\0.98 + (0.5 \times (1 - 0.98)) &= 0.99\end{aligned}$$

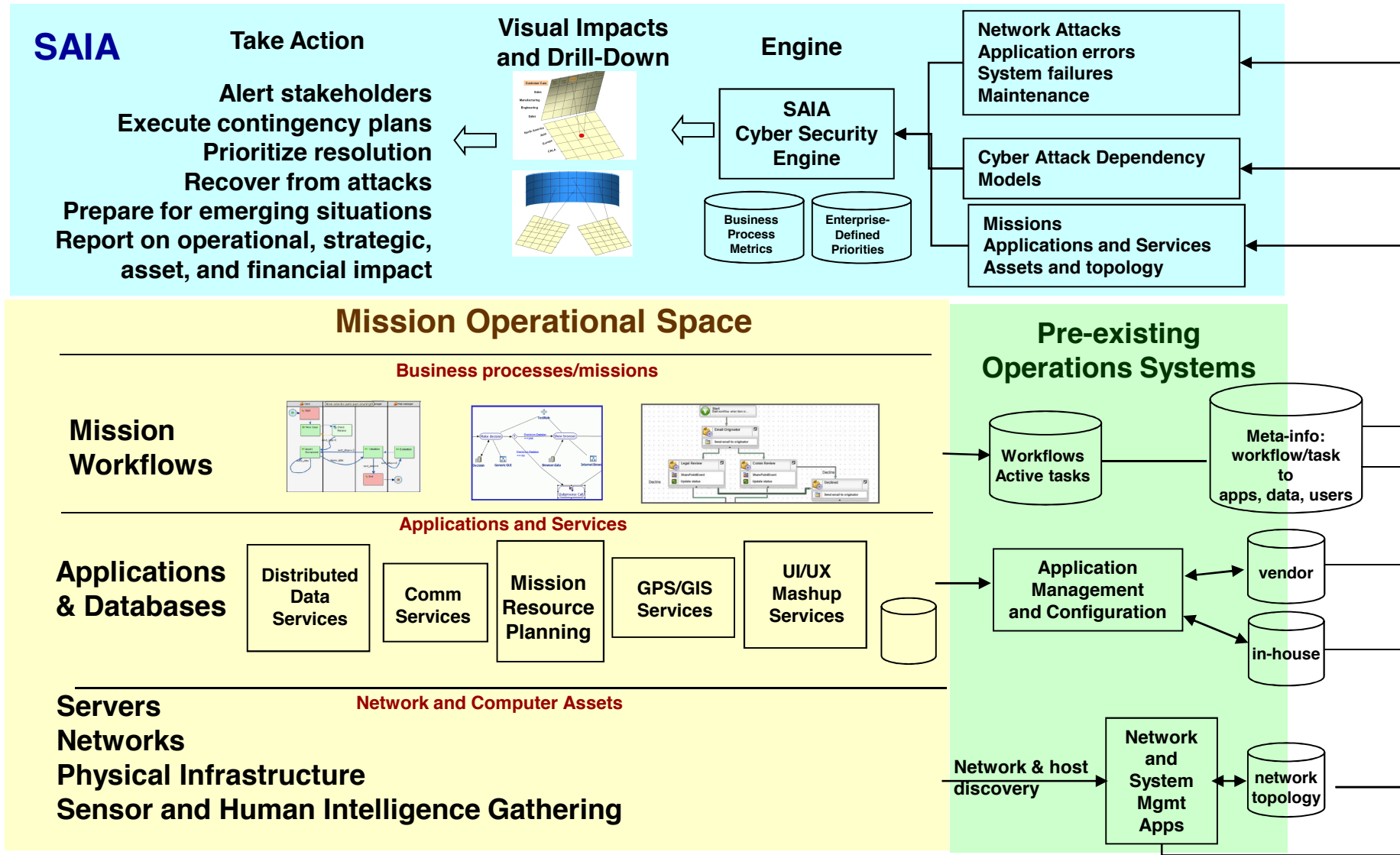
*Jackson, Peter (1998). Introduction to Expert Systems, ISBN 0-201-87686-8.

Calculating Plausible Future Cyber Security Situations

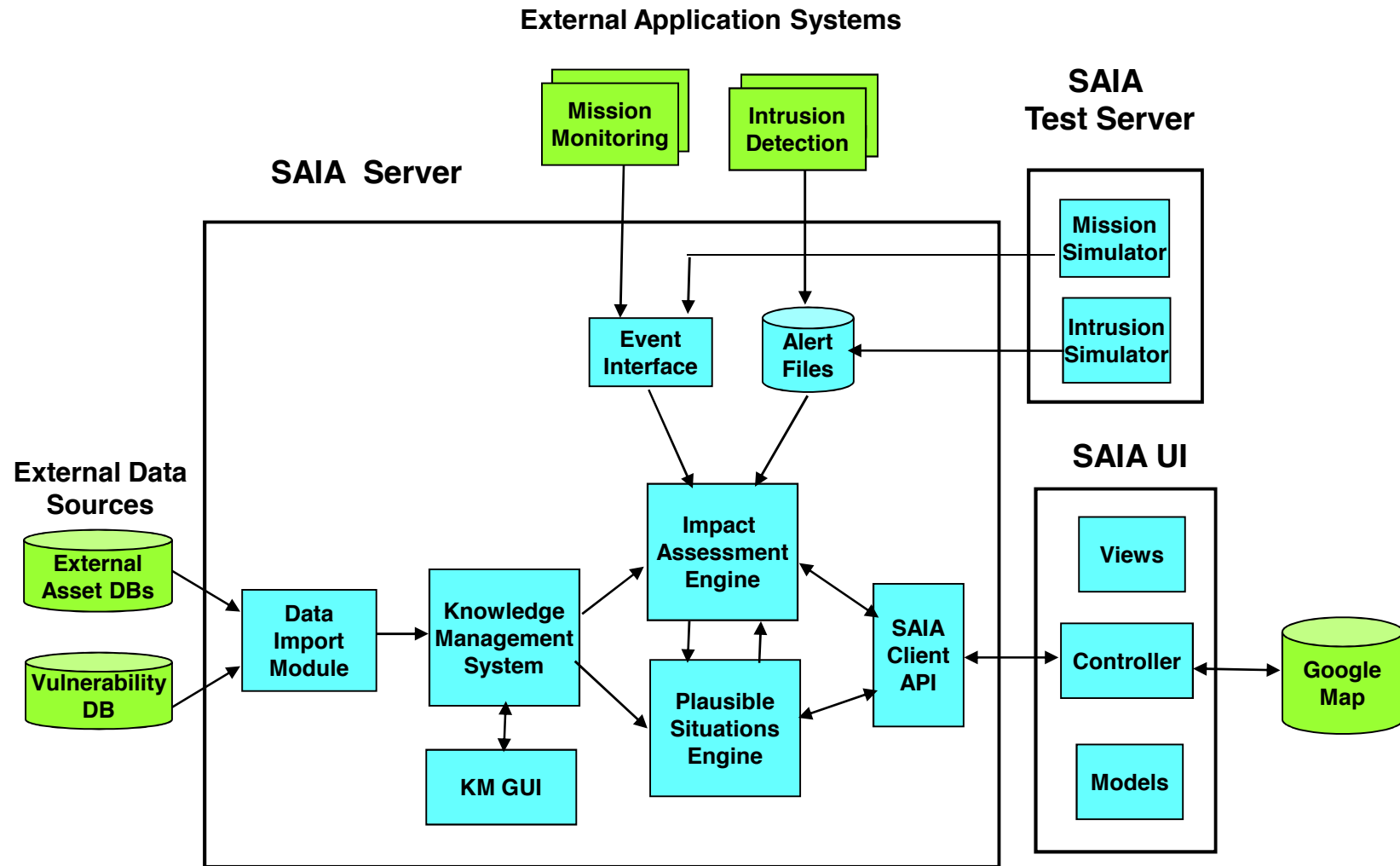


6. SAIA – Situation Awareness and Impact Assessment System

SAIA: What and How?



SAIA Internals



SAIA Software Architecture

- **Web-based architecture leveraging Ruby on Rails**
 - Ruby programming language
 - Rail web infrastructure and object/relational mapping (ORM)
 - RSS and email support
- **MySQL, Oracle, SQL Server and other DBMSs directly supported for deployment**
- **Mongrel application server, Apache web server**
 - Mongrel cluster provides scalability and fault tolerance
- **Web services supported for integration with external services such as IDS alert import**
- **Knowledge Management UI uses standard HTML and Javascript**
- **SAIA UI/UX – Mashup JackBe's Presto, Adobe's Flex**

SAIA User Interface - Features

- **Easy to learn, responsive, web-based UI to providing an integrated visual presentation of:**
 - **Active missions under management**
 - **Mission impacts**
 - **Network attacks**
- **Drill down to mission operations and detailed views**
 - **Services and assets on which missions depend**
 - **Dependencies between missions, services and assets**
- **Predicted impacts on other missions based on current attack**

7. Sample Application

Domain Expertise & Sierra Leone Mission

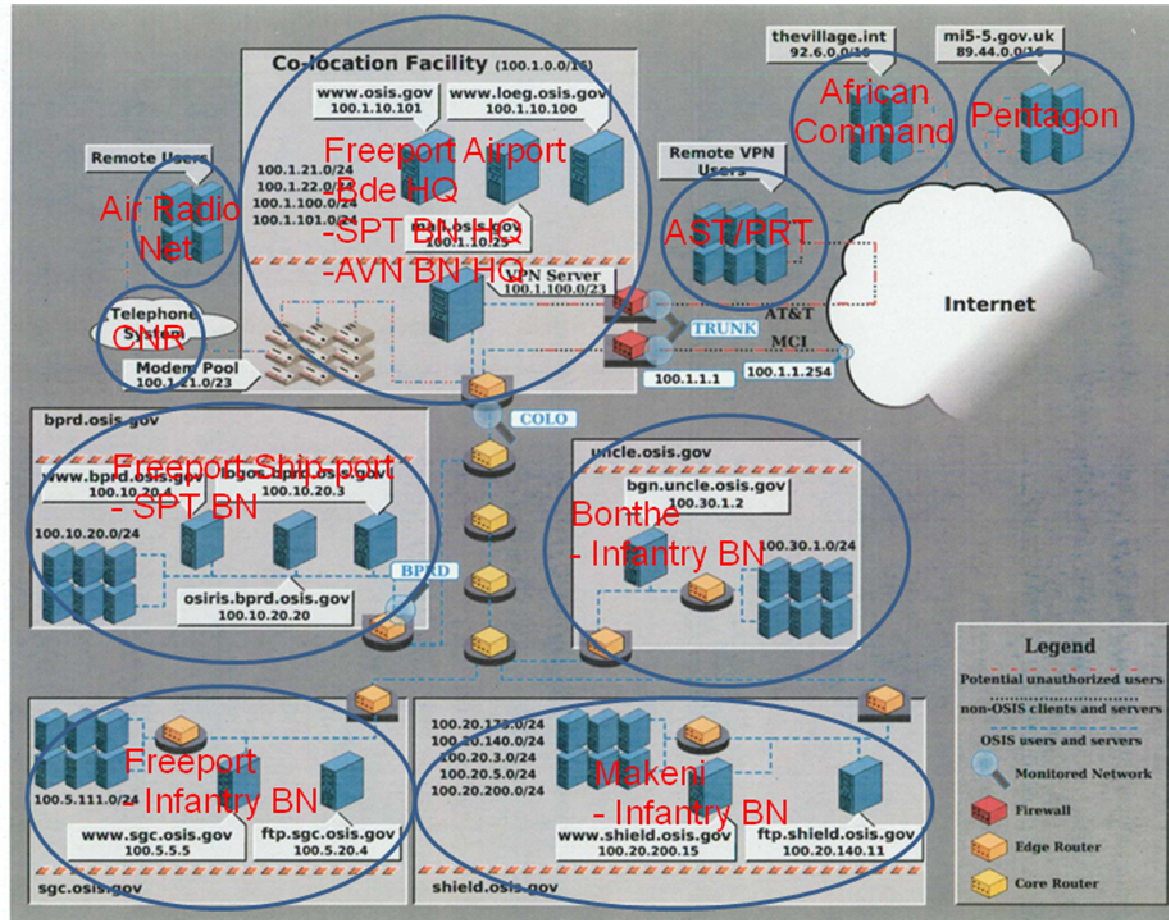
- Mission expected for a US Army Brigade Combat Team (BCT - Light) to support a Humanitarian Mission in the African country of Sierra Leone.



Background Information

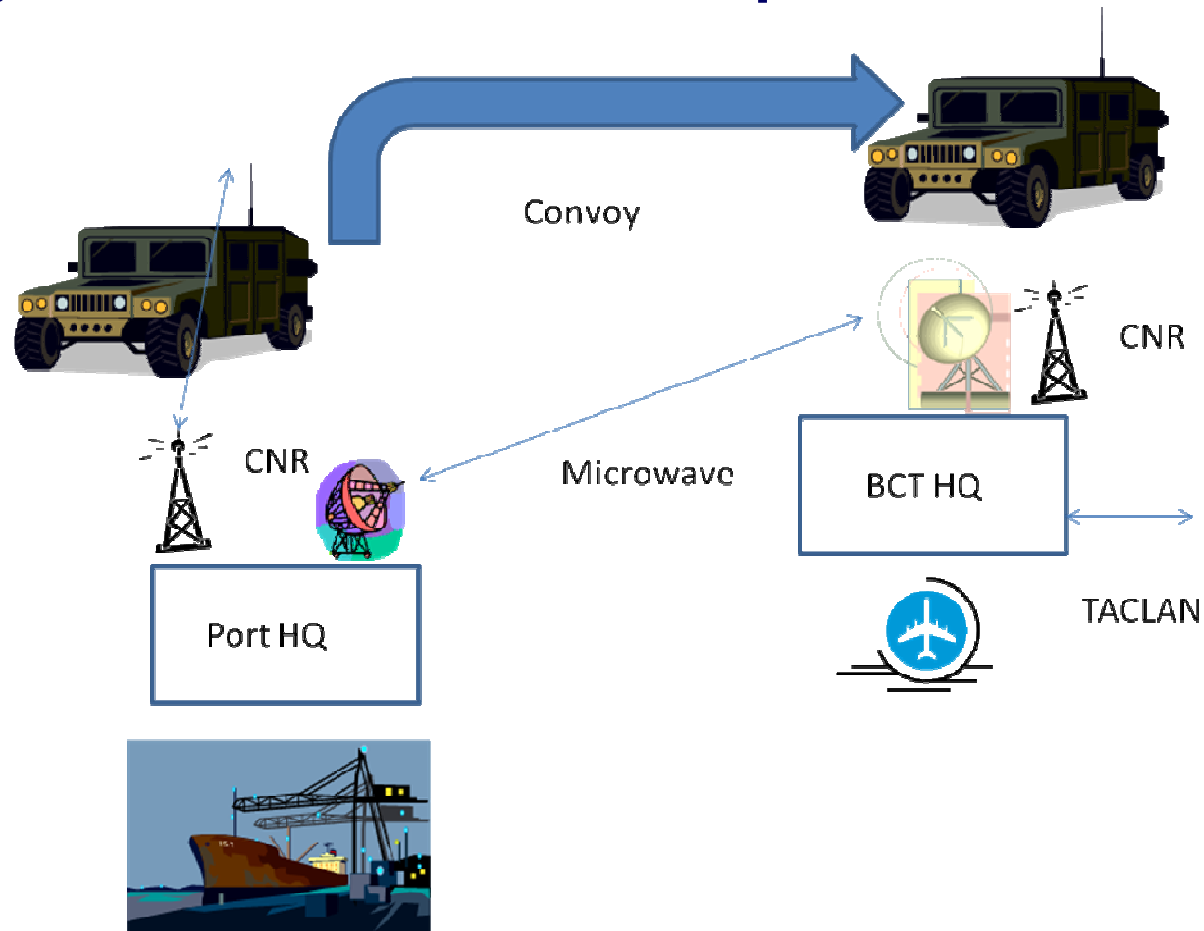
- **After long-lasting civil war in Sierra Leone the US and the Sierra Leone governments reached an agreement on deploying a Brigade Combat Team (BCT) of the 25th Infantry Division (Light) in Sierra Leone, Africa.**
- **The deployed BCT will consist of approximately 4,000 soldiers and is broken down into the following units:**
 - **Brigade Headquarters Company (BHQ) – Located at the airport in Freetown.**
 - **Aviation Battalion (AVB) – Located at the airport in Freetown.**
 - **Support Battalion (SUB) – Located at both the airport and main shipping port in Freetown.**
 - **Infantry Battalion (INB1) – HQ at Freetown airport, but has small units deployed in the Freetown metropolitan area.**
 - **Infantry Battalion (INB2) – HQ at the airport in the port city of Bonthe, but has small units deployed in the coastal region.**
 - **Infantry Battalion (INB3) – HQ located at the main sports complex in the city of Makeni, but has small units deployed throughout the central region.**

Sierra Leone Mission Network Configuration

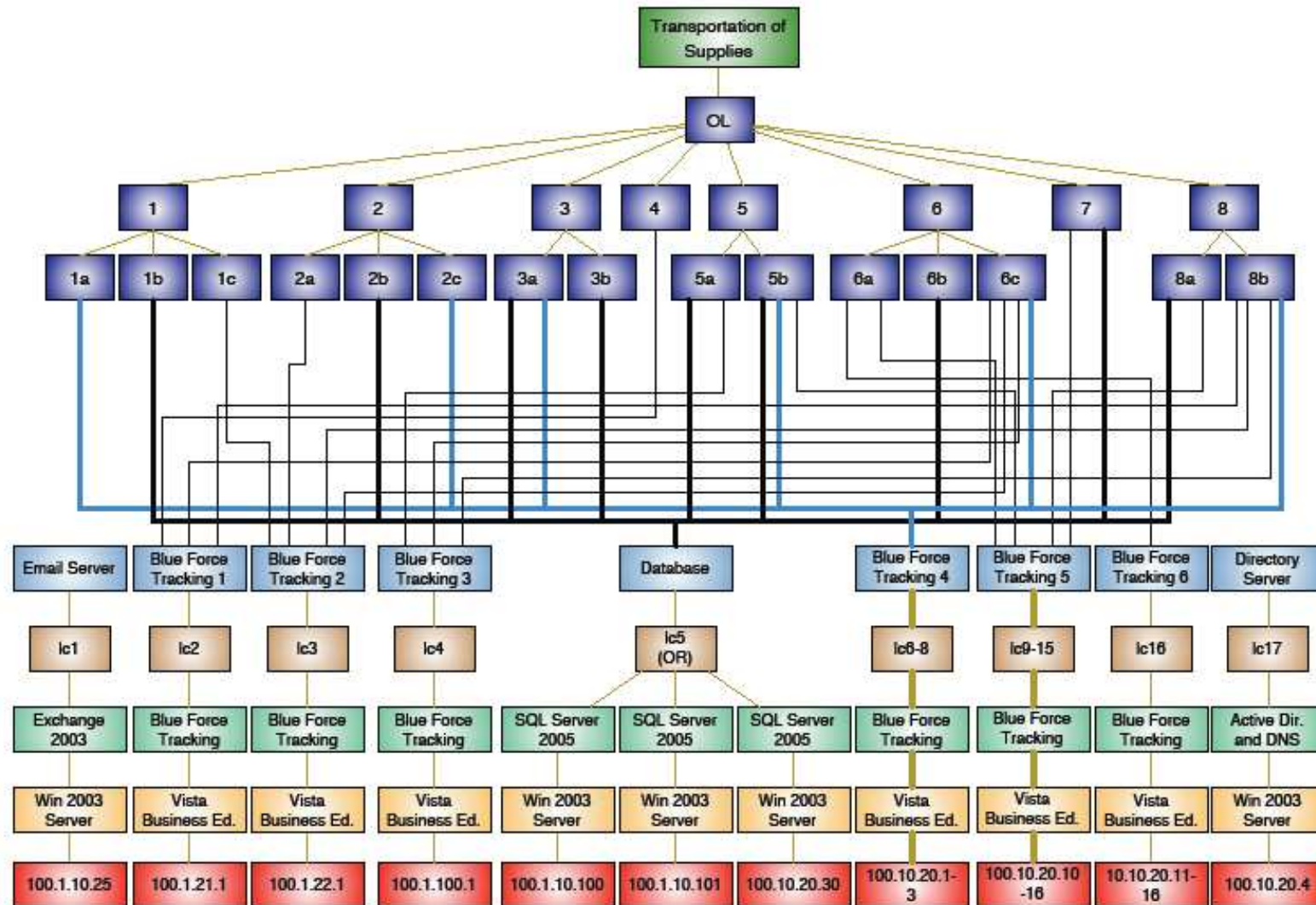


(Based on SKAION's OSIS Network)

Diagram for Mission 1: Transportation of Supplies



Sierra Leone Mission 1 Dependency Graph



Attack on SQL Database Server 100.1.10.100

SQL Injection (Attack from African Command unknown host to FBCB2-BFT3 Servers at: Freeport Airport (2), Freeport Ship-port, and Makeni Infantry BN)

- 09/29-11:11:48.039614 **[**] [1:456:4] SQL Inject Attempt [**] [Classification: Attempted Information Leak] [Priority: 1] {ICMP} 96.6.0.113 -> 100.1.10.100**
- 09/29-11:12:48.041165 **[**] [1:456:4] SQL Inject Attempt [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 96.6.0.113 -> 100.1.10.101**
- 09/29-11:13:48.041162 **[**] [1:456:4] SQL Inject Attempt [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 96.6.0.113 -> 100.10.20.30**
- 09/29-11:14:48.041172 **[**] [1:456:4] SQL Inject Attempt [**] [Classification: Attempted Information Leak] [Priority: 2] {ICMP} 96.6.0.113 -> 100.10.200.15**

SAIA Mission Monitoring Window

The screenshot displays the SAIA Mission Monitoring Window, which is divided into several functional areas:

- Top Left:** A hierarchical tree structure showing mission components and their relationships.
- Top Center:** A "Geospatial View" featuring a map of a coastal region with various locations labeled (e.g., Freeport, Bane, Makanshan). A tooltip for a mission is visible: "Mission: Transportation of Supplies by Ground Convoy", "Plausible Future Discovery Time: 2010-07-02 21:18:29 UTC", and "Operational Capacity: 0.25".
- Top Right:** A "SAIA Tasks" timeline for "June 18, 2010" from 2 pm to 6 pm. The timeline shows a sequence of tasks represented by colored bars (green and red) indicating task duration and status.
- Bottom:** A "Mission Monitor" panel with "GLANCE" and "DETAIL" tabs. The "GLANCE" tab is active, showing a "High" priority mission: "Transportation of Supplies by Ground Convoy". The status is "COMPLETED" with an "ALERT" icon. The completion time is "06/17/2010 17:09:32" and the "Impact Factor" is "0.3". The "Location" is also displayed with latitude and longitude fields.

Mission Cyber Situation Monitoring

Navigation for mission drill down

Click to select Mission from list

View:

Missions
Services
Assets
Alerts

Jan 27, 2010 **18:31:04**

New York - Eastern Time ⌵

User Profile Logout

>> Navigation bar

Priority	Mission
High	<p>Transportation of Supplies from Freeport Ship Terminal to Freeport Airport by Ground Convoy</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 15%;"> <p>Alert: Step 2.1 STATUS: Active. Today, 18:31:04 EST Impact Factor: 1.0 Oracle DBMS non responsive, target IP 100.10.20.3</p> </div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 25%;"> <p>Dependencies: Services and Assets Data services for this mission offline. target IP 100.10.20.3</p> </div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 20%; text-align: right;"> <p>Location: Sierra Leone Latitude: 80 N Longitude: 11 30 W</p> </div> </div>
High	<p>Transportation of Rare Blood type from Freeport airport to the forward deployed infantry at Makeni</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 15%;"> <p>Alert: Step 2.1 STATUS: Active. Today, 18:31:04 EST Impact Factor: 1.0 Pilot to fly helicopter delayed in convoy en route to Freeport airport</p> </div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 25%;"> <p>Dependencies: Services and Assets Data services for this mission offline. target IP 100.10.20.3</p> </div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 20%; text-align: right;"> <p>Location: Makeni Latitude: 85 N Longitude: 11 30 W</p> </div> </div>
Medium	<p>Intelligence gathering on person of interest X in location Y in Sierra Leone region</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 15%;"></div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 25%;"> <p>Dependencies: Services and Assets Data services for this mission offline. target IP 100.10.20.3</p> </div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 20%; text-align: right;"> <p>Location: Sierra Leone Latitude: 80 N Longitude: 11 30 W</p> </div> </div>
Medium	<p>Aircraft maintenance and repair mission at Freeport airport, Sierra Leone</p> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 15%;"></div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 25%;"> <p>Dependencies: Services and Assets Data services for this mission offline. target IP 100.10.20.3</p> </div> <div style="width: 15%; text-align: center;"> </div> <div style="width: 20%; text-align: right;"> <p>Location: Makeni Latitude: 85 N Longitude: 11 30 W</p> </div> </div>

Mission Cyber Situation Monitoring More Elaborate View

altusys		View:		Dec 12, 2009		New York - Eastern Time		User Profile		Logout			
		Missions		Services		Assets				18:31:04			
Stat	Priority	Mission	Steps for this Mission										
■	High	Transportation of Supplies from Freeport Ship Terminal to Freeport Airport by Ground Convoy <div style="border: 2px solid red; padding: 5px;"> Alert: Step 2.1 STATUS: Active Today, 18:31:04 EST Acknowledge Impact Factor: 1.0 Clear Oracle DBMS non responsive, target IP 100.10.20.3 </div>	<ol style="list-style-type: none"> ▼ 1. Transportation Unit schedules convoy mission to deliver ammunition to BCT Headquarters at Freeport Airport ▼ 2. Confirm receipt of go ahead to coordinate equipment, resources and security <ul style="list-style-type: none"> ■ 2.1 Confirm convoy is ready to depart the ship port 2.2 Create and send report that verifies that all vehicles, supplies, equipment, personnel are on Blue Force Tracking ▶ 3. GPS locations of all vehicles in convoy are updated throughout mission and sent via Blue Force Tracking to all key staff members ▶ 4. Convoy arrives at BCT HQ at Freeport Airport at expected time with no major issues to report. ▶ 5. Create and send Mission Complete report via Blue Force Tracking to all key staff members 									Alerts Today, 18:31:04 EST Impact Factor: 1.0 Oracle DBMS non responsive, target IP 100.10.20.3 Monday, 06:28 EST Impact Factor: 0.1 Mission 4 has had a minimal compromise to services	
■	High	Transportation of rare blood type to the forward deployed Infantry BN at Makeni	<ol style="list-style-type: none"> ▼ 1. Transportation Unit to schedule flight to deliver materials via air lift ▶ 2. Confirm that there is room on the aircraft for transporting the materials using medical transport equipment ▶ Confirm aviation unit, operations officer, and both sending and receiving medical officers are advised of the mission for tracking purposes ▶ 3. Confirm detailed plan of equipment and transport directions ▶ 4. Blood is carried in medical transport equipment onto the helicopter and loaded for shipment. ▶ 5. Report of this information is sent to key staff members through Blue Force Tracking ▶ 6. GPS locations of helicopter is updated throughout mission and sent via Blue Force Tracking to all key staff members ▶ 7. Helicopter arrives at destination location with no known issues ▶ 8. Create and send Mission Complete report via Blue Force Tracking to all key staff members 										
■	High	Information gathering on person X in location	▶ 1. GPS locations of all persons of interest are being updated throughout mission and sent via Blue Force Tracking to all key staff members										
■	Med	Aircraft maintenance and repair mission	▶ 1. GPS locations of all aircraft are being sent via Blue Force Tracking to all key staff members										
■	Low	Pilot training program in location X	▶ 1. GPS locations of all aircraft and pilot personnel are being updated throughout mission and sent via Blue Force Tracking to all key staff members										



View:

Missions Services Assets

Dec 12, 2009
18:31:04

New York - Eastern Time

User Profile

Logout

Stat	Priority	Mission	Steps for this Mission	Alerts
Red square	High	Transportation of Supplies from Freeport Ship Terminal to Freeport Airport by Ground Convoy Alert: Step 2.1 STATUS: Active Today, 18:31:04 EST Impact Factor: 1.0 Oracle DBMS non responsive, target IP 100.10.20.3 Acknowledge Clear	<ol style="list-style-type: none"> 1. Transportation Unit schedules convoy mission to deliver ammunition to BCT Headquarters at Freeport Airport 2. Confirm receipt of go ahead to coordinate equipment, resources and security <ol style="list-style-type: none"> 2.1 Confirm convoy is ready to depart ship port 2.2 Create and send report that verifies that supplies, equipment, personnel are on Blue Force Tracking 3. GPS locations of all vehicles in convoy are updated throughout mission and sent via Blue Force Tracking to all key staff members 4. Convoy arrives at BCT HQ at Freeport Airport at expected time with no issues 5. Create and send Mission Complete report via Blue Force Tracking to all key staff members 	Today, 18:31:04 EST Impact Factor: 1.0 Oracle DBMS non responsive, target IP 100.10.20.3
Green square	High	Transportation of rare blood type to the forward deployed Infantry BN at Makeni	<ol style="list-style-type: none"> 1. Transportation Unit to schedule flight to deliver materials via air lift 2. Confirm that there is room on the aircraft for transporting the materials using medical equipment 3. Coordinate with operations officer, and both sending and receiving units for purposes of transport directions 4. Load equipment onto the helicopter and loaded for shipment 5. Report status to key staff members through Blue Force Tracking 6. Update location throughout mission and sent via Blue Force Tracking 7. Report location with no known issues 8. Create report via Blue Force Tracking to all key staff members 	
Yellow square	High	Information gathering on person X in location	<ol style="list-style-type: none"> 1. GPS locations of all aircraft are being updated throughout mission and sent via Blue Force Tracking to all key staff members 	
Green square	Med	Aircraft maintenance and repair mission	<ol style="list-style-type: none"> 1. GPS locations of all aircraft are being sent via Blue Force Tracking to all key staff members 	
Green square	Low	Pilot training program in location X	<ol style="list-style-type: none"> 1. GPS locations of all aircraft and pilot personnel are being updated throughout mission and sent via Blue Force Tracking to all key staff members 	

List of active missions, with detailed tasks that define mission

Alert showing attack on this mission and cause of impact

Sample drill down on mission

Selected "lenses" displayed on available screen(s)

altusys **View:** Missions Services Assets Jan 27, 2010 18:31:04 New York - Eastern Time User Profile Logout

>> Home > Mission: Transportation of Supplies from Freeport Ship

Transportation of Supplies from Freeport Ship Terminal to Freeport Airport by Ground Convoy

ALERT **Alert: Step 2.1**
STATUS: Active.
Today, 18:31:04 EST
Impact Factor: 1.0
Oracle DBMS offline
target IP 100.10.20.3

Dependencies: Services and Assets
Data services for this mission offline.
target IP 100.10.20.3

Dependencies: Services and Assets
Data services for this mission offline.
target IP 100.10.20.3

Location: Sierra Leone
Latitude: 80 N
Longitude: 11 30 W

Dependencies: Services and Assets

```
graph TD; M1[Mission 1: Transportation of Supplies from Freeport Ship Terminal to Freeport Airport by Convoy] --> S21[Step 2.1: Confirm convoy is ready to depart ship port]; S21 --> IE[Intel Entry]; S21 --> DA1[Data Analysis]; IE --> AND1[AND]; AND1 --> OR1[OR]; OR1 --> ODBMS1[Oracle DBMS]; OR1 --> ODBMS2[Oracle DBMS]; OR1 --> ODBMS3[Oracle DBMS]; DA1 --> AND2[AND]; AND2 --> OR2[OR]; OR2 --> DA2_1[Data Analysis]; OR2 --> DA2_2[Data Analysis]; subgraph DBMS_Cluster [DBMS Cluster]; ODBMS1; ODBMS2; ODBMS3; end; subgraph App_Cluster [App Cluster]; DA2_1; DA2_2; end;
```

Location: Latitude 80 N, Longitude: 11 30 W

Technology

- **Ruby on Rails framework**
- **Prolog constraint processing engine**
 - Modified to use SQL input
- **MySQL RDBMS (initially)**
- **Apache web server**
- **Mongrel application server**
- **Linux OS**
- **(GUI technology TBD)**

References

- A. Valdes and K. Skinner, “Probabilistic alert correlation”. Proceedings of the Fourth International Symposium on Recent Advances in Intrusion Detection (RAID 2001), 54–68
- R. P. Goldman, W. Heimerdinger and S. A. Harp, “Information Modeling for Intrusion Report Aggregation”, In DARPA Information Survivability Conference and Exhibition, 2001.
- X. Qin and W. Lee, “Discovering Novel Attack Strategies from INFOSEC Alerts”, In Proceedings of the 9th European Symposium on Research in Computer Security, Sophia Antipolis, France 2004.
- H. Debar and A. Wespi, “The Intrusion Detection Console Correlation Mechanism”, In 4th International Symposium on Recent Advances in Intrusion Detection (RAID), 2001.
- X. Qin and W. Lee. “Attack Plan Recognition and prediction Using Causal Networks”, In Proceedings of the 20th Annual Computer Security Applications Conference, pp. 370-379, 2004.
- S. Cheung, U. Lindqvist, and M. W. Fong, “Modeling Multi-Step Cyber Attacks for Scenario Recognition”, In Proceedings of the 3rd DARPA Information Survivability Conference and Exhibition< Washington, D. C., 2003.
- G. Jakobson, J. Buford, L. Lewis. Situation Management: Basic Concepts and Approaches, Proceedings of the 3rd International Workshop on Information Fusion and Geographic Information Systems, ST. Petersburg, Lecture Notes in Geoinformation and Cartography, Springer-Verlag Berlin Heidelberg, 2007.
- Z. Li and J. Lei, “Assessing Attack Threat by Probability of Following Attacks,” In Proceedings of the International Conference on Networking Architecture & Storage, pp. 91-100, 2007.
- J. Holsopple and S. J. Yang, “FuSIA: Future Situation and Impact Awareness”, In Proceedings of the 11th International Conference on Information Fusion, Cologne, Germany, 2008.
- Simulation Modeling: Finding Plausible Futures for Diabetes Prevalence, Centers for Disease Control and Prevention, 2008.
- J. F. Sowa. Knowledge representation: Logical, Philosophical, and Computational Foundation. Brooks Cole Publishing Co., Pacific Grove, CA, 2000.
- B. Argauer, and S. Young, “VTAC: Virtual Terrain Assisted Impact Assessment for Cyber Attacks,” IN Proceedings of SPIE Security and Defense Symposium, Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security Conference, 2008.
- L. Lewis, Service Level Agreements, Artech House, 1999.
- B. Kosanovic, Physical System Modeling Using Temporal Fuzzy Sets", *Proc. of the International Joint Conference of NAFIPS/FIS/NASA'94*, pp. 429-433, San Antonio, TX, December 18-21, 1994.
- L. A. Zadeh (1965) "Fuzzy sets". *Information and Control* 8 (3) 338–353.
- OSVDB, The Open Source Vulnerability Database, 2010.
- Jackson, Peter (1998). Introduction to Expert Systems, ISBN 0-201-87686-8.

Thank You for Your Interest!