TU Wien

On the Cognitive Complexity of Cyber-Physical System of Systems (CPSoS)

H.Kopetz
September 2015

Outline

- Introduction
- Cyber-Physical System of Systems (CPSoS)
- Communication in a CPSoS
- Cognitive Complexity
- Simplification Strategies
  - Abstraction
  - Emergence
  - Partitioning
  - Segmentation
- Conclusion

Motivation

- The major cost elements during the specification, design, operation, evolution and maintenance of a large CPSoS are accrued in the *non-physical domain*.
- Compared to the cost of the *engineering effort*, the hardware costs of a CPSoS are modest—and *getting even smaller* as the hardware technology moves forward.
- The *engineering effort* depends to a considerable degree on the *cognitive complexity*, i.e., the time needed to *understand the behavior of a system*.
- Any reduction of the cognitive complexity of a large system is thus of utmost economic significance and reduces the probability of the occurrence of design errors.

System of Systems (SoS)

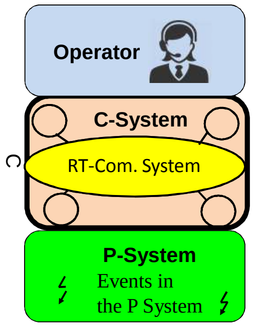
An SoS is an integration of a finite number *of autonomous constituent systems (CS) e.g., embedded systems,* which are independent and operable, and which are networked together for a period of time to achieve a certain higher goal (refer to Jamshidi, 2009, T-Area SoS)**.**

SoSs are qualitatively different from Embedded Systems

Embedded System versus a *CPSoS*

Characteristic	**Embedded System**	**SoS**
Scope of System	**Fixed (known)**	**Not known**
Requirements and Spec.	Fixed	Changing
Context	Single	Multiple
Evolution	Version control	Uncoordinated
Testing	Test phases	Continuous
Implementation Technology	Given and fixed	Unknown
Faults (Physical, Design)	Exceptional	Normal
Control	Central	Autonomous
Emergence	Insignificant	Important

A CPS consists of three Subsystems



Cyber Space

meets

Physical Space

Physical Space versus *Cyber Space*

Physical Space	**Cyber Space**
Laws of physics	Program execution
Physical time	**Execution time**
Time base dense	Time-base sparse

We need a computational model, where physical time and execution time are properly integrated.

The widely used *time-triggered frame-based data flow model* meets this requirement.

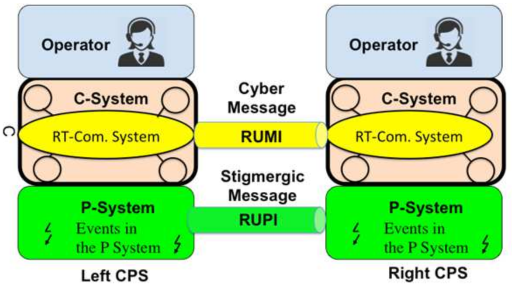
Time-Triggered Frame-Based Data Flow Model

- Physical Time is partitioned into a periodic sequence of constant durations, called *Frames*, where each frame consists of two phases, a processing phase and a communication phase.
- At the beginning of the processing phase of each frame, the input data from the environment and the current state are provided to the application.
- At the end of the processing phase, the output data are produced for the environment or the (phase-aligned time-triggered) communication system.
- During the communication phase the output data is transported to the successor by the communication system and a new version of the internal state is produced for the following frame.

There should be no interaction with the environment during a frame.

November 27, 2015 Page 8 AMADEOS Consortium

Communication Channels in a CPSoS



The Relied-Upon Interfaces (RUI). i.e the RUMI (*Relied Upon Message Interface*) and the RUPI (*Relied Upon Physical Interface*), form the backbone of an SoS Architecture.

Relied Upon Message Interfaces (RUMI)

- A RUMI is *an SoS Internal message interface* between constituent systems (CS).
- *RUMIs should be standardized* by an accepted industry consortium or standard setting organization.
  - Standardization of the Data Formats
  - Standardization of the Explanation
- *Controlled Evolution* of the RUMIs—often backward compatibility required.
- *Gateway component* between a RUMI and the external world to *buffer* changes.

Stigmergic Channels

- The biologist *Grasse* introduced the term *stigmergy* to describe the indirect information flow among the members of a termite colony when they coordinate their nest building activities.
- According to the present understanding, the nearly blind ants orient themselves on the information captured by the *olfactory sense* following the intensity of the smell of the chemical substance *pheromone*.
- **A *stigmergic information channel* is present if one CPS acts on the environment common to many CPSs, changes the state of this physical environment and another CPS observes the changed state at some later point in time.**
- Since stigmergic information channels operate in the *physical environment* (not in cyber space) they exposed to the full spectrum of *environment dynamics*.

Traffic Flow

The information flow among drivers on a busy road is mainly of the *stigmergic* type.

Relied Upon Physical Interface (RUPI)

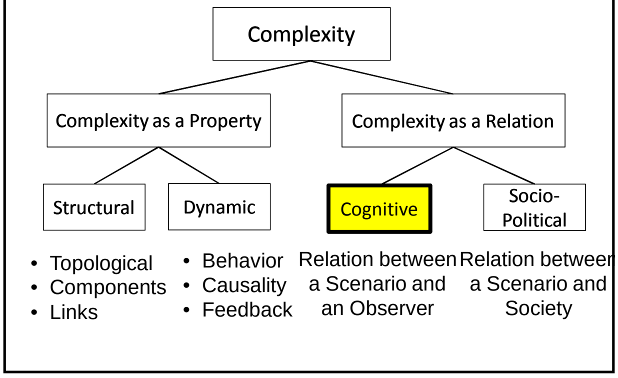
The state of the physical environment is *dynamic*—it changes as time progresses:

- A RUPI is *an SoS interface* between constituent systems (CS) that is based on *physical entities* and *stigmergic channels*.
- The cyber-model of the physical environment may be incomplete (a car that does not support C2C communication—***environmental dynamics*** are not fully captured).
- Up-to-date state of the physical environment (full environmental dynamics) is captured by the stigmergic channels.
- The sensor fusion of the information in cyber-space and of the information in physical space is ***time-sensitive.***

Summary: Stigmergic versus Message Based Itoms

Characteristic	**RUPI (Stigmergic)**	**RUMI (Cyber Message)**
Information Type	Properties of Things	No Restriction
Inform. Transfer	Pull	Push
Tense	Present	Past, Present, Future
Observation Mode	Direct	Indirect
Observation Delay	None	Existent
Comm. Delay	Unbounded	Bounded
Source	Unknown	Known
E-Dynamics	**Considered**	**Not Considered**
Representation	Single Context	Multiple Contexts

Complexity



Cognitive Complexity

Cognitive complexity is concerned with the questions:
*How much mental effort is required in order to **understand** a **given scenario** for the given **purpose** by an **identified user?***

The *time* it takes for an *average representative* from the *intended user group* to *understand* the scenario is linked to the *cognitive complexity* of a scenario.

The time required for understanding will depend upon

- the purpose of understanding
- the *conceptual basis* of the intended user group
- the inherent characteristics of the scenario
- the representation of the scenario.

Understanding and Certification

- Many CPSoS are safety critical, e.g.
  - Flight control systems
  - Nuclear reactor control
  - Automotive control systems
  - Medical systems
- Many safety-critical systems require certification by an independent certification authority!
- How can we certify a system, if we don't understand its behavior in every detail?

Mental Model

According to Craig,
understanding the behavior
means that a *mental model* that establishes *causal links* between the *inputs, the state* and the *outputs* of the system has been formed.

The basis for our reasoning is information, not data.

Information versus *Data*

An ***Information Item*** *(Itom)* is *a proposition about some state or behavior of the world*.

An *Itom* consists of ***data*** and an ***explanation of the data***.

- In cyber-space, *data* is represented by a *bit-pattern*.
- While the *data* is carried explicitly in a message, the ***explanation* is often implied by context**.
- In a large *SoS the* ***context of the sender can be different from the context of the receiver.*** If this is the case, then a message that carries *data* without an explanation can be interpreted differently by the sender and the receiver.

Example: **86° F** means the same as **30° C**

Some Consequences

- The input and output Itoms (information Items) of the constituent systems must be *observable at the relied upon interfaces*.
- In case the behavior depends on the internal state of a system, this internal state *must also be observable*.
- ***Causal Order*** **presupposes *Temporal Order.***
- The *Temporal Order* of all input and output messages can be established if the messages are time-stamped with a *global time*.

The Need for *Global Time*

A dependable *global (physical) time* is needed to understand the temporal behavior of an SOS.

A global time is needed to

- Interpret the time-stamps across the diverse embedded systems
- Control and synchronize the *durations of the physical time frames*
- Specify the *temporal properties* of interfaces
- Synchronize inputs and output actions
- Synchronize *stigmergic* and *message-based* information
- Allocate resources *conflict-free* (e.g, in time-triggered communication, scheduling)
- *Detect errors* promptly
- Strengthen *security protocols*

A Counter-Example

On August 14, 2003 a major power blackout occurred in parts of the US and Canada. In the final report about this blackout it is stated on p. 162:

A valuable lesson from the August 14 blackout is the importance of having time-synchronized system data recorders. The Task Force's investigators labored over thousand of data items to determine the sequence of events, much like putting together small pieces of a very large puzzle. That process would have been significantly faster and easier if there had been wider use of synchronized data recording devices.

How much valuable engineering time has been wasted by coping with this unnecessary complexity?

Simplification Strategies

- Abstraction
- Emergence
- Partitioning
- Segmentation

Abstraction

- Abstraction (from the Latin *abs*, meaning *away from* and *trahere*, meaning *to draw*) is the process of taking away or removing properties from a *detailed model* of an entity in order to arrive at a *simpler model* that is still sufficient for the stated purpose.
- Abstraction forms a category of all entities that share the properties of the *simpler model*.
- The notion of category is *recursive*: the *elements of a category* can themselves be *categories,* thus forming a hierarchy of categories..
- A category that is augmented by a *set of beliefs* about its relations to other entities is called a *concept.*

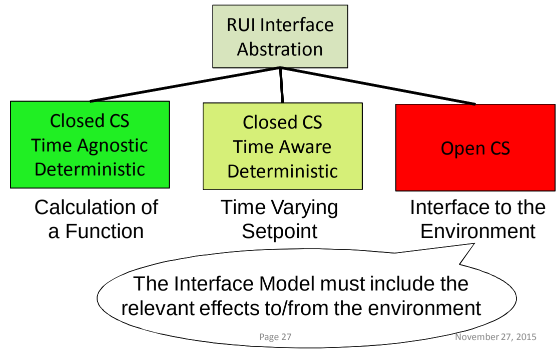
Key to Success: Finding Proper Abstractions

In celestial mechanics, when we are interested in the interactions between heavenly bodies, we build an *abstraction* where we put aside the diversity of our world and consider it to be a ***single mass point***--the ultimate simplicity.

Relied Upon Interface Abstraction

- In a CPSoS the services of a constituent system (CS), i.e the service provided at the *Relied Upon Interfaces (RUI)*, are specified in the *Service Level Agreement (SLA)*.
    - Service: *Intended Behavior*
    - Behavior: *Functions plus Time*
- **The SLA *abstracts* from the mechanism internal to a CS and its environment.**
- The SLA must be *complete*, it must specify the behavior of *the CS and its environment* in the functional, temporal and non-functional domains.

RUI Interface Abstraction—Case Distinction


RUI Interface Abstraction
Closed CS Time Agnostic Deterministic
Closed CS Time Aware Deterministic
Open CS
Calculation of a Function
Time Varying Setpoint
Interface to the Environment
The Interface Model must include the relevant effects to/from the environment
Page 27
November 27, 2015

Emergence

Definition of Emergence: *A phenomenon of a whole at the macro-level (the SoS level) is emergent if and only if it is new with respect to the non-relational phenomena of any of its proper parts at the micro level.*

- Emergent Phenomena at the CPSoS level are caused by the *interactions (e.g. information flow* and *material flow)* among the Constituent Systems (CS) of an SoS across message-based and stigmergic channels.
- Parts are integrated into a ***New Whole*** that is characterized by a ***new conceptualization*** (e.g, the concept of *transistor*).
- Sometimes emergent phenomena appear unexpectedly with negative side effects (e.g., deadlock).

Classification of Emergent Behavior

Contribution of emergent behavior to the overall goal of a system **Emergent behavior**	**Beneficial**	**Detrimental**
Expected	Normal case	Avoided by appropriate rules
Unexpected	Positive surprise	Problematic case

Emergence, Supervenience and Simplification

Supervenience is a relation between the emergent phenomena of adjacent levels in a hierarchy:

- **Sup_1:** A given emerging phenomenon at the macro level can emerge out of many different arrangements or interactions of the parts at the micro-level
- **Sup_2:** A difference in the emerging phenomena at the macro level requires a difference in the arrangements or the interactions of the parts at the micro level.

Because of *Sup_1* one can abstract from many different arrangements or interactions of the parts at the micro level that lead to the same emerging phenomena at the macro level.

Sup_1 entails a significant simplification of the higher-level models.

Emergence is our *Friend*, not our Enemy

The proper conceptualization of emergent phenomena can lead to an abrupt simplification at the next higher Level.

Examples:

- Fault-Tolerant Distributed Clock Synchronization → leads to the new concept of a *Dependable Global Time*
- The interactions among set of properly connected transistors → A *new whole* the behavior of which can be described by the concepts of *Boolean Logic.*
- A multitude of gas atoms leads to a *new whole* that can be characterized by the new concept *pressure*.

Partitioning

Also called: *divide and conquer*, *separation of concerns*, *functional independence*.

Independent functional requirements
should be implemented by
independent mechanisms.

The *separation of concerns* leads to mechanisms that can be analyzed in isolation without any complications caused by the entanglement of mechanisms.

Example: Communication in a CPSoS

Requirement 1—*data domain*: A CS should perform the data transformations specified in the SLA.

Requirement 2—*temporal domain*: Messages should be transmitted within the specified time constraints.

In an ***event-triggered communication system***, the triggers for the message transmission ***depend*** *on* the proper operation in the data domain.

In a **time-time triggered communication system**, the triggers for the message transmission ***are independent*** *of* the proper operation in the data domain.

Example: Interface Abstraction

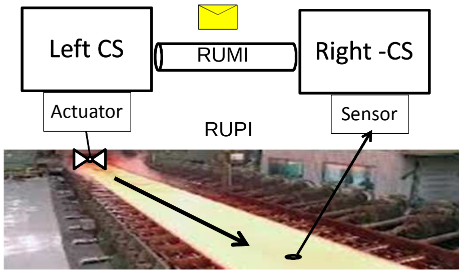
- Many sensors provide *raw data* that need a *sensor specific explanation* to transform the data to a standardized form.
- This sensor specific transformation of the data should be *hidden* from the user of the data and implemented by a separate mechanism.
- Only *standardized representations of data* should be provided at a RUI in order to take advantage of already *existing concepts in the mind of a user.*

Standardized representation of information **that is familiar to the intended user groups** simplifies the understanding.

Segmentation

- Segmentation refers to the *sequential split-up of action sequences in the temporal domain* such that each action can be analyzed in isolation within its limited temporal context.
- The widely used *frame-based data flow model* supports segmentation by implementing *a periodic finite state machine.*
- **The well-defined temporal order of events in a segmented system supports the causal analysis of action sequences.**
- The behavior of *concurrent action sequences* that interact among each other is *difficult to understand*.

Example Segmentation: Loop Analysis



A control loop in a CPS, consisting of message channels and stigmergic channels can be *segmented* into *discrete phases* in order to support the analysis.

Summary: Some Simplification Principles

Global Time: An established temporal order of events, based on global time stamps, supports *segmentation* and the construction of a ***causal mental model*** of behavior.

Divide and Conquer: *Independent* requirements should be implemented by *independent* mechanisms.

Emergence: The proper conceptualization of emergent phenomena can lead to an abrupt simplification.

Time-Triggered Communication eliminates the dependence of the communication system from the behavior in the data domain.

Relied-Upon Interfaces: The full specification of all RUIs (message based and stigmergic), is needed for the *interface abstraction*.

Identification of Interface State: The *interface state is required* in case the interface services are not state-agnostic.

Conclusions

- Any reduction of the cognitive complexity of a large system is of utmost economic significance and reduces the probability of the occurrence of design errors.
- Understanding the behavior means that a mental model that establishes *causal links* between the *input itoms*, the *state itoms* and the *output itoms* of the system is available.
- Since temporal order is a precondition of causal order, the availability of CPSoS wide global timestamps supports the causal analysis of behavior.
- The established simplification strategies of *abstraction*, *partitioning* and *segmentation* should be applied to reduce the cognitive effort that is needed to understand the behavior of a CPSoS.

An Example for *Environmental Dynamics*