PRELIMINARY DISCUSSION DRAFT

Framework for Cyber-Physical Systems

Release 0.7

3/3/2015 5:27 PM

Cyber Physical Systems Public Working Group

Table of Contents

Table of Figures

Table of Tables

Disclaimer

This document (including any preliminary discussion drafts) has been prepared by the Cyber-Physical Systems Public Working Group (CPS PWG), an open public forum established by the National Institute of Standards and Technology (NIST) to support stakeholder discussions and development of a framework for cyber-physical systems. This document is a freely available contribution of the CPS PWG and is published in the public domain.

Certain commercial entities, equipment, or materials may be identified in this document in order to describe a concept adequately. Such identification is not intended to imply recommendation or endorsement by CPS PWG (or the National Institute of Standards and Technology), nor is it intended to imply that these entities, materials, or equipment are necessarily the best available for the purpose.

Executive Summary

Cyber-Physical Systems (CPS) are smart systems that include co-engineered interacting networks of physical and computational components. CPS and related systems (including the Internet of Things, Industrial Internet, and more) are widely recognized as having great potential to enable innovative applications and impact multiple economic sectors in the world-wide economy.

In 2014, NIST established the CPS Public Working Group (CPS PWG) to bring together a broad range of CPS experts in an open public forum to help define and shape key aspects of CPS to accelerate development and implementation within and across multiple "smart" application domains, including smart manufacturing, transportation, energy and healthcare.

The objective of the CPS PWG is to develop a shared understanding of CPS and its foundational concepts and unique dimensions (as described in this "CPS Framework") to promote progress through exchanging ideas and integrating research across sectors and among disciplines and to support development of CPS with new functionalities. While in principle there are multiple audiences for this work, a key audience is the group of CPS experts, architects and practitioners who will benefit from an organized presentation of CPS conceptual reference architecture facets and aspects, which identifies key concepts and issues informed by the perspective of the five expert subgroups in the CPS PWG: reference architecture, security, timing, data interoperability, and use cases. This foundation then enables further use of these principles to develop a comprehensive standards and metrics base for CPS to support commerce and innovation. As an example, the CPS Framework could support identification of the commonalities and opportunities for interoperability in complex CPS, at scale. A broader audience for this work includes all CPS stakeholders, who may be interested in broadening individual domain perspectives to consider CPS in a holistic, multi-domain context.

The three-stage work plan of the CPS PWG has been to first develop initial "Framework Element" documents in each of the five subgroups: reference architecture, cybersecurity and privacy, timing and synchronization, data interoperability, and use cases. Then in the second phase, these documents have been combined into an initial draft CPS Framework and revised and improved to create this draft document. The (future) third phase is a roadmapping activity to both improve the CPS Framework and develop understanding and action plans to support its use in multiple CPS domains.

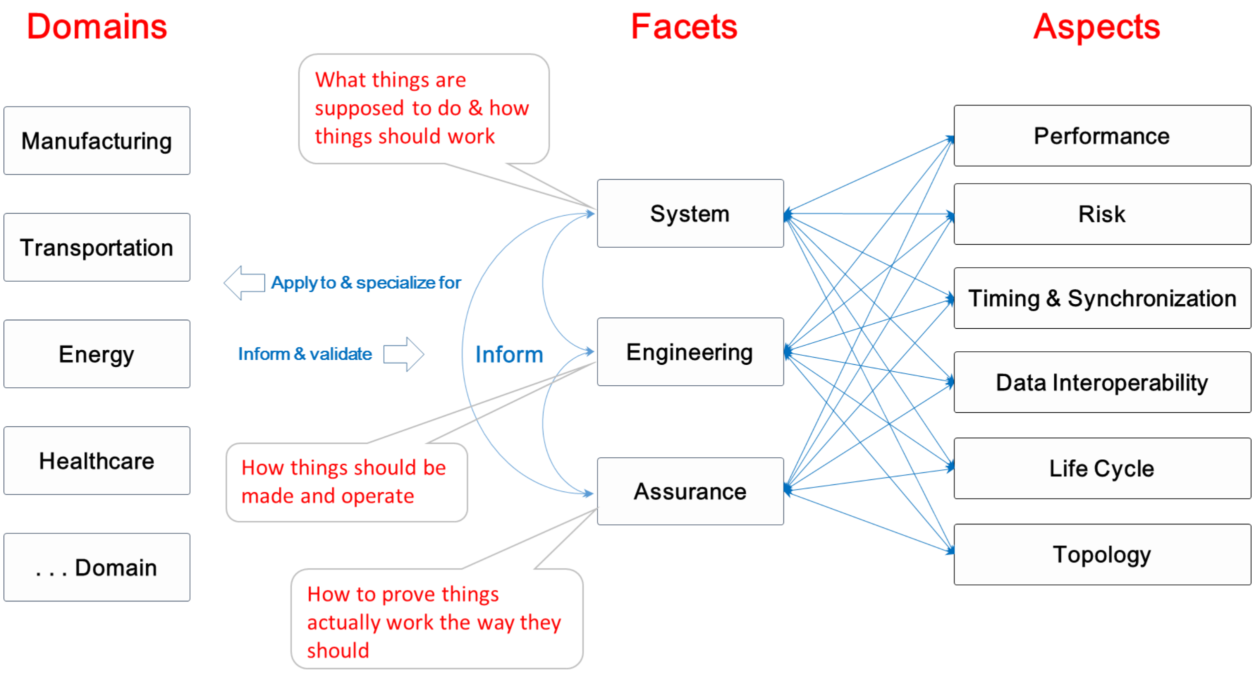
With respect to this draft CPS Framework, the goal has been to first derive a unifying framework that covers, to the extent understood by the CPS PWG participants, the range of unique dimensions of CPS. The second goal is then to populate a significant although not yet complete portion of the framework with detail, drawing upon content produced by the CPS PWG subgroups and CPS PWG leadership team.

The diagram below shows this analysis proceeding in a series of steps as undertaken within the reference architecture activity:

- Start with the enumeration of application domains of CPS

- Identify concerns, e.g., societal, business and technical, and others; stakeholders can have concerns that overlap or are instances of broader conceptual concerns
- Derive from these generic concerns, the fundamental “Facets” of “System,” “Engineering,” and “Assurance”
- Analyze cross-cutting concerns to produce “Aspects”

Through two iterations of integration and analysis, the following view was distilled from the work:



Domains:

It is intended that the identification and description of the activities, methods and outcomes in each of these Facets can be applied to concrete CPS application domains, e.g., manufacturing, transportation, energy, etc. as a specialization of these common conceptions and descriptions and as a means for integrating domains for coordinated functions. Conversely, these specializations may validate and help to enhance the overarching CPS conceptions and descriptions.

Facets:

The System Facet of the CPS architecture captures the functional requirements and organization of CPS as it pertains to what a CPS or its components are supposed to do and how things should work. If we consider the design of a building as a metaphor, this represents the view of the building as a whole – what the customer wants; how many floors; windows, etc. Domain experts – those knowledgeable about the nature and operation of a CPS domain, typically assemble the System Aspect in Use Cases.

The Engineering Facet addresses the concerns of the stakeholders that design, maintain and operate the system. Using a layered approach, it captures the different activities surrounding the processes and activities surrounding the design and implementation of such systems. Topics such as system engineering processes and tools are pertinent here. Also, modeling and simulation and other activities that help inform and actuate the design process.

The Assurance Facet deals with the verification of the design. It addresses the processes, tools, and activities that deal with testing and certification of implementations of CPS. Additionally the verification of the requirements as met by designs is a topic of the assurance facet.

Aspects:

Sets of cross-cutting concerns, identified as “Aspects,” are listed below, and in this draft CPS Framework:

- Performance
- Risk (which includes Cybersecurity & Privacy, Safety, Reliability, and Resiliency)
- Timing and Synchronization
- Data Interoperability
- Life Cycle
- Topology

During the second phase of the CPS PWG, based on the reference architecture subgroup’s work described above to identify the concepts of facets and aspects to organize its work on reference architecture, an ambitious restructuring of this document along these organizing principles has been undertaken. As such, it is an intentional feature of this work that some newly discovered attributes and concepts (in particular, the “Assurance Facet” and several cross-cutting Aspects, including the “Lifecycle Aspect” and “Topology Aspect”) are not significantly developed at this time, and will be further addressed during the upcoming third road mapping phase of the CPS PWG.

In summary, this draft CPS Framework draws from content developed within the CPS PWG subgroups, which has been integrated and reorganized to follow an overarching document structure based on the identified reference architectural concepts of facets and aspects.

Further input and comments from a broad audience will be useful to inform CPS PWG efforts to build out and improve the CPS Framework.

1 Purpose and Scope

1.1 Overview and Background

Cyber-physical systems (CPS) are smart systems that include co-engineered interacting networks of physical and computational components.[1] These highly interconnected and integrated systems provide new functionalities to improve quality of life and enable technological advances in critical areas, such as personalized health care, emergency response, traffic flow management, smart manufacturing, defense and homeland security, and energy supply and use. In addition to CPS, there are many words and phrases (Industrial Internet, Internet of Things, smart cities, and others)[2] that describe similar or related systems and concepts.[3]

The impacts of CPS will be revolutionary and pervasive – this is evident today in emerging autonomous vehicles, intelligent buildings, robots, and smart medical devices. Realizing the full promise of CPS will require interoperability among heterogeneous components and systems, supported by new reference architectures using shared vocabularies and definitions. Addressing the challenges and opportunities of CPS requires broad consensus in foundational concepts, and a shared understanding of the essential new capabilities and technologies unique to CPS. To this end, NIST has established the CPS Public Working Group (CPS PWG), which is open to all, to foster and capture inputs from those involved in CPS, both nationally and globally.

The CPS PWG was launched in mid-2014 with the establishment of five subgroups (reference architecture, use cases, security, timing, and data interoperability).[4] Initial “Framework Element” documents were produced by each of the subgroups in December 2014, then integrated, reorganized and refined to create this draft CPS Framework Release 0.7. The CPS Framework is intended to be a living document and will be revised over time to address stakeholder community input and public comments; some sections of the document are incomplete and will be developed and extended over time.

The core element of the CPS Framework is a common vocabulary and reference architecture. The reference architecture should capture the generic functionalities that CPS provide, and the

[1] A technical definition of CPS is provided in Section 2.1, and for convenience, CPS may be considered to be either singular or plural.

[2] Some of these terms are defined in Section 1.4; also note some are used for marketing purposes.

[3] CPS will be the focus of this document; however, terminology distinctions may be introduced to aid the reader where beneficial or informative. As an example, CPS may sound like the notion of ‘mechatronics’, however mechatronics designers and manufacturers have more ‘product control’ of the entire process. In the case of CPS they are by intent cross product in their conception, design, and execution.

[4] Additional information on the NIST CPS PWG is available at www.cpspwg.org and http://www.nist.gov/cps/

activities and outputs needed to support engineering of CPS. Together these will be called the CPS Reference Architecture (CPS RA). Accordingly, the design of a CPS instance is one of the engineering activities and among its output is a domain specific CPS architecture, a schematic or complete set of requirements for the desired CPS instance. Thus the CPS RA will consist of a set of activities and their outputs. The approach of this document is to gather the key elements of current thinking and practice relating to CPS in order to:

- Assemble high level concepts that capture the key elements, present in current applications of CPS, needed in a reference architecture
- Identify the relationships between these elements and categorize them relative to their role in the system
- Determine how these categories of elements of the reference architecture interact generally
- Present the set of these elements as a many-sorted structure, including elements and relations and functions. Provide a common language and common constructs that will facilitate future development and maintenance of the reference architecture. Together this structure and language provide a framework for a CPS Reference Architecture
- Use this common framework to structure efforts to address key examples of critical concerns to advance CPS.

As an example of the desired comprehensiveness of this framework, the language and constructs of the CPS Reference Architecture should also address organizational needs and should assist in addressing such issues as:

  - Life Cycle Process
  - Design, Verification and Validation
  - Manufacturing
  - Service and Retirement

1.2 Purpose

The success of this CPS Framework can be assessed by its usefulness as guidance in designing CPS and as a tool for describing and demonstrating properties of CPS. It should aid users in determining whether a system is an instance of the CPS RA, and provide guidance such that two CPS instance architectures, independently derived or tailored from the CPS RA, are in substantial alignment.

It should also serve as a design template or methodology by providing a general decomposition of CPS and the CPS design activity, including the activities and tasks associated with developing the elements of a cyber-physical system. An example, the framework should facilitate the decomposition of a CPS instance into layers corresponding with the categories of elements noted above. The successful delivery of the components and communication network of a CPS instance requires systematic coordination between the groups that design and deliver those components and the group that is responsible for developing the communications for the CPS.

By providing a framework for discussion, design of, and reasoning about CPS, a common foundation or 'starting point' will be established, from which a myriad of interoperable CPS can be developed, safely and securely combined and delivered to the public, government and researchers. If broadly adopted, this framework will serve to stimulate activity in research and provide the 'glue' that will support development of CPS-based products and economy.

A simple CPS conceptual domain model is shown in Figure 1. This figure is a simplification of a CPS Functional Domain diagram (Figure 6) presented later in the Framework, and is presented here to highlight the potential interactions of a CPS (e.g., a device) and a system of systems (e.g., a CPS infrastructure). The CPS has multiple flows, including information, decision, action, energy/material, and management flows, occurring within and between the domains.

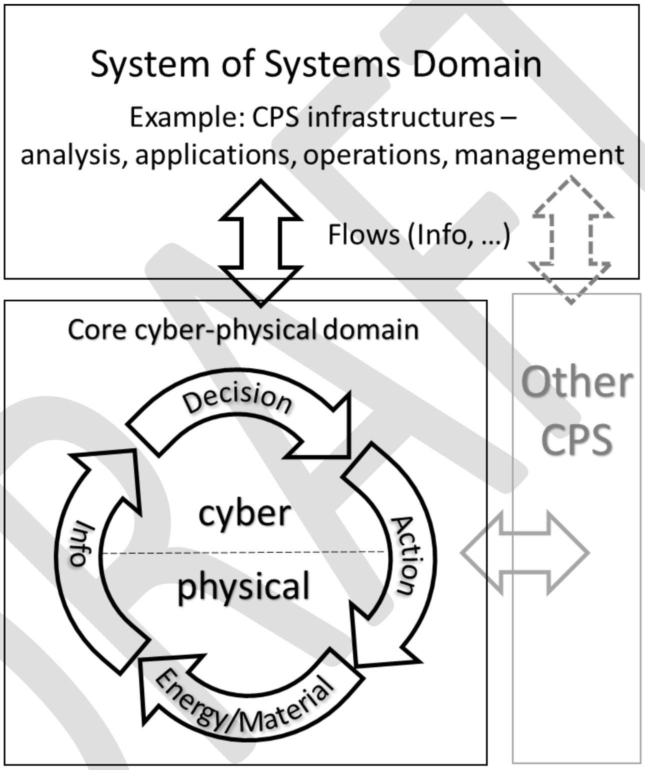

System of Systems Domain
Example: CPS infrastructures –
analysis, applications, operations, management
Flows (Info, …)
Core cyber-physical domain
Decision
Info
cyber
physical
Action
Energy/Material
Other
CPS

Figure 1: Simplified CPS conceptual domain model

1.3 Scope

The scope of CPS is very broad by nature, as demonstrated in Figure 1 by the large number and variety of domains, services, applications and devices in a visual representation of CPS focused on the Internet of Things. This broad CPS scope includes cross-cutting functions that are likely to impact multiple interacting CPS domains. The CPS Framework will facilitate users' understanding of cross-cutting functions, i.e. functions that are derived from critical and overriding CPS concerns. Addressing such concerns in CPS may impact multiple 'layers' in a CPS instance architecture. Examples include safety, security, interoperability and others.

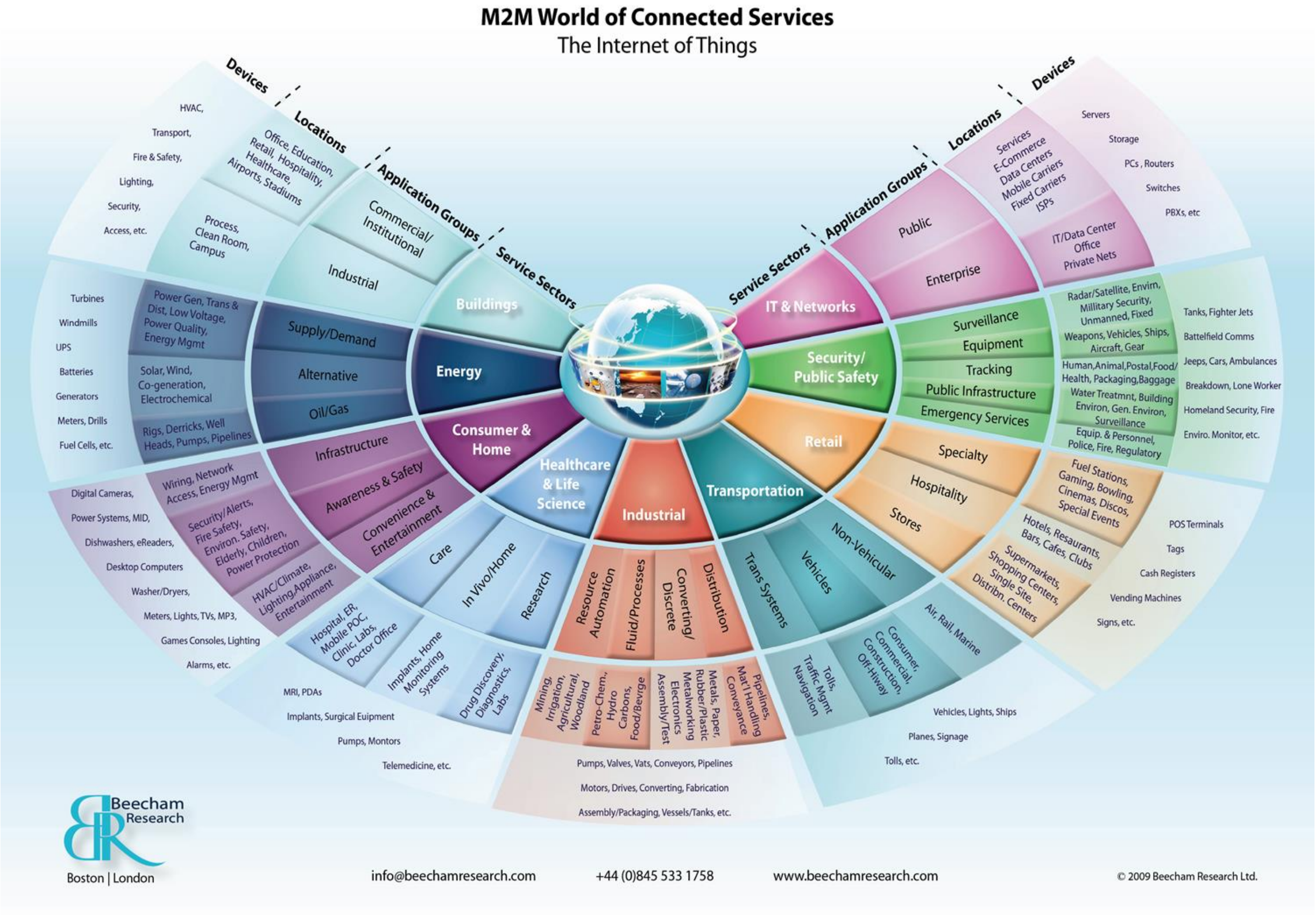

M2M World of Connected Services
The Internet of Things
Devices
Locations
Application Groups
Service Sectors
HVAC,
Transport,
Fire & Safety,
Lighting,
Security,
Access, etc.
Office, Education, Retail, Hospitality, Healthcare, Airports, Stadiums
Process, Clean Room, Campus
Commercial/ Institutional
Industrial
Buildings
Turbines
Windmills
UPS
Batteries
Generators
Meters, Drills
Fuel Cells, etc.
Power Gen, Trans & Dist, Low Voltage, Power Quality, Energy Mgmt
Solar, Wind, Co-generation, Electrochemical
Rigs, Derricks, Well Heads, Pumps, Pipelines
Supply/Demand
Alternative
Oil/Gas
Energy
Digital Cameras,
Power Systems, MID,
Dishwashers, eReaders,
Desktop Computers
Washer/Dryers,
Meters, Lights, TVs, MP3,
Games Consoles, Lighting
Alarms, etc.
Wiring, Network Access, Energy Mgmt
Security/Alerts, Fire Safety, Environ. Safety, Elderly, Children, Power Protection
HVAC/Climate, Lighting, Appliance, Entertainment
Infrastructure
Awareness & Safety
Convenience & Entertainment
Consumer & Home
MRI, PDAs
Implants, Surgical Euipment
Pumps, Montors
Telemedicine, etc.
Hospital, ER, Mobile POC, Clinic, Labs, Doctor Office
Implants, Home Monitoring Systems
Drug Discovery, Diagnostics, Labs
Care
In Vivo/Home
Research
Healthcare & Life Science
Pumps, Valves, Vats, Conveyors, Pipelines
Motors, Drives, Converting, Fabrication
Assembly/Packaging, Vessels/Tanks, etc.
Mining, Irrigation, Agricultural, Woodland
Petro-Chem, Hydro Carbons, Food/Bevrge
Assembly/Test Electronics Metalworking
Metals, Paper, Rubber/Plastic
Pipelines, Mat'l Handling Conveyance
Resource Automation
Fluid/Processes
Converting/ Discrete
Distribution
Industrial
Vehicles, Lights, Ships
Planes, Signage
Tolls, etc.
Tolls, Traffic Mgmt Navigation
Consumer, Commercial, Construction, Off-Hiway
Air, Rail, Marine
Trans Systems
Vehicles
Non-Vehicular
Transportation
POS Terminals
Tags
Cash Registers
Vending Machines
Signs, etc.
Fuel Stations, Gaming, Bowling, Cinemas, Discos, Special Events
Hotels, Resaurants, Bars, Cafes, Clubs
Supermarkets, Shopping Centers, Single Site, Distribn. Centers
Specialty
Hospitality
Stores
Retail
Tanks, Fighter Jets
Battlefield Comms
Jeeps, Cars, Ambulances
Breakdown, Lone Worker
Homeland Security, Fire
Enviro. Monitor, etc.
Radar/Satellite, Envir., Military Security, Unmanned, Fixed
Weapons, Vehicles, Ships, Aircraft, Gear
Human, Animal, Postal, Food/ Health, Packaging, Baggage
Water Treatmnt, Building Environ, Gen. Environ, Surveillance
Equip. & Personnel, Police, Fire, Regulatory
Surveillance
Equipment
Tracking
Public Infrastructure
Emergency Services
Security/ Public Safety
Servers
Storage
PCs, Routers
Switches
PBXs, etc
Services E-Commerce Data Centers Mobile Carriers Fixed Carriers ISPs
IT/Data Center Office Private Nets
Public
Enterprise
IT & Networks
Beecham Research
Boston | London
info@beechamresearch.com
+44 (0)845 533 1758
www.beechamresearch.com
© 2009 Beecham Research Ltd.

Figure 2: Sector map showing segmentation of M2M (machine-to-machine) market and identifying 9 key service sectors, key applications groups, and examples of connected devices[5]

The figure shows the dimensionality of the CPS Domain space – that, the application areas where CPS devices exist.

1.4 Definitions

These referenced definitions are presented as a ready reference to the intended meaning of their use in the text of this document. It is recognized that within various technical domains, many of these terms have multiple meanings. The intent here is to provide clarity for the

[5] Courtesy of Beecham Research, used by permission, http://www.beechamresearch.com/article.aspx?id=4, 2009

interpretation of this framework and not to make a definitive statement about the “universal” definition of the terms.

Term	Definition	Source
access control	A means to ensure that access to assets is authorized and restricted based on business and security requirements Note: Access control requires both authentication and authorization	[12]
accuracy	Closeness of the agreement between the result of a measurement and the true value of the measurand.	ITU-R Rec. TF.686
actors	A person or system component who interacts with the system as a whole and who provides stimulus which invoke actions.	[116]
actuator	A device which conveys digital information to effect a change of some property of a physical entity.	[5]++
ageing	The systematic change in frequency with time due to internal changes in the oscillator. NOTE 1 – It is the frequency change with time when factors external to the oscillator (environment, power supply, etc.) are kept constant.	ITU-R Rec. TF.686
architecture layer		
architecture view	An ‘architecture view’ consists of ‘work product expressing the architecture of a system from the perspective of specific system concerns’.	
architecture viewpoint	An ‘architecture viewpoint’ consists of work product establishing the conventions for the construction, interpretation and use of architecture views to frame specific system concerns’.	
aspect	Conceptually equivalent concerns, or major categories of	

Term	Definition	Source
	concerns. Sometimes called “cross-cutting” concerns.	
assurance	The level of confidence that software is free from vulnerabilities, either intentionally designed into the software or accidentally inserted during its life cycle, and that the software functions in the intended manner.	
assurance level	The Evaluation Assurance Level (EAL1 through EAL7) of an IT product or system is a numerical grade assigned following the completion of a Common Criteria security evaluation, an international standard in effect since 1999. The increasing assurance levels reflect added assurance requirements that must be met to achieve Common Criteria certification. The intent of the higher levels is to provide higher confidence that the system's principal security features are reliably implemented. The EAL level does not measure the security of the system itself, it simply states at what level the system was tested.	
assured time	Time derived from a known good time reference in a secure manner.	
attribute	A characteristic or property of an entity that can be used to describe its state, appearance, or other aspects.	[10]
authenticated identity	Identity information for an entity created to record the result of identity authentication.	[10]
authentication	Provision of assurance that a claimed characteristic of an entity is correct.	[12]
authorization	Granting of rights, which includes the granting of access based on access rights.	[7]
automatic	Working by itself with little or no direct human control.	[16]
automation	The use or introduction of automatic equipment in a manufacturing or other process or facility.	[16]

Term	Definition	Source
	Note: Automation emphasizes efficiency, productivity, quality, and reliability, focusing on systems that operate without direct control, often in structured environments over extended periods, and on the explicit structuring of such environments.	
calibration	The process of identifying and measuring offsets between the indicated value and the value of a reference standard used as the test object to some determined level of uncertainty. NOTE 1 – In many cases, e.g. in a frequency generator, the calibration is related to the stability of the device and therefore its result is a function of time and of the measurement averaging time.	ITU-R Rec. TF.686
choreography	Type of composition whose elements interact in a non-directed fashion with each autonomous part knowing and following an observable predefined pattern of behavior for the entire (global) composition	[13]
clock	A device that generates periodic signals for synchronization. Note: Other definitions are provided in different references that are tailored to particular applications. Suitable references include ITU-T Rec. G.810, ITU-R Rec. TF.686 and IEEE Std. 1377-1997.	
co-design		
collaboration	Type of composition whose elements interact in a non-directed fashion, each according to their own plans and purposes without a predefined pattern of behavior.	[13]
component	Modular, deployable, and replaceable part of a system that encapsulates implementation and exposes a set of interfaces.	[8]
composition	Result of assembling a collection of elements for a	[13]

Term	Definition	Source
	particular purpose.	
controller	A User that interacts across a network to affect a physical entity.	[5] ++
CPS architecture	A concrete realization of a reference CPS architecture designed to satisfy use-case-specific constraints.	
CPS domain	A CPS domain is a logical group of CPS nodes and bridges which form a network with their own timing master.	
CPS network manager	A work-station or CPS node connected to a CPS domain that manages and monitors the state and configuration of all CPS nodes in one or more CPS domains.	
CPS reference architecture (CPS RA)	Abstract framework for understanding and deriving application-domain-specific CPS architectures. Activities and outputs to support engineering of CPS.	
cross-cutting concern		
cross-cutting function		
data	Re-interpretable representation of information in a formalized manner suitable for communication, interpretation, or processing. NOTE Data can be processed by humans or by automatic means.	[104]
data accuracy	Closeness of agreement between a property value and the true value. NOTE 1: In practice, the accepted reference value is substituted for the true value.	[104]
device	A physical entity embedded inside, or attached to, another physical entity in its vicinity, with capabilities to	[17]

Term	Definition	Source
	convey digital information from or to that physical entity.	
device	A device, such as a laptop, sensor, smartphone, MEMS or nanotechnology chip, may be viewed as a physical component of a cyber-physical system, but its utility derives from the digital entities that may be accessed from, associated with or embedded in the device. A simple device typically has a make/model/serial number associated with it that assists in identifying and locating it, while more complex devices may be capable of performing or executing more complex operations. A device, as well as the digital entities embedded therein, may be composite in form, i.e., made up of other devices or digital entities.	
device endpoint	An endpoint that enables access to a device and thus to the related physical entity.	[17]
digital entity	An entity represented as, or converted to, a machine-independent data structure consisting of one or more elements in digital form that can be parsed by different information systems; and the essential fixed attribute of a digital entity is its associated unique persistent identifier, which can be resolved to current state information about the digital entity, including its location(s), access controls, and validation, by submitting a resolution request to the resolution system.	
element	Unit that is indivisible at a given level of abstraction and has a clearly defined boundary. Note: An element can be any type of entity	[13]
emergent behavior		
endpoint	One of two components that either implements and exposes an interface to other components or uses the interface of another component.	[11]

Term	Definition	Source
endpoint address	Data element designating the originating source or destination of data being transmitted.	[9]
entity	Item inside or outside an information and communication technology system, such as a person, an organization, a device, a subsystem, or a group of such items that has recognizably distinct existence	[10]
entity	Anything that has a separate and distinct existence that can be uniquely identified. Examples of entities include subscribers, users, network elements, networks, software applications, services and devices. An entity may have multiple identifiers	
epoch	Epoch signifies the beginning of an era (or event) or the reference date of a system of measurements.	ITU-R Rec. TF.686
facet	Facets are perspectives on CPS that each express a distinct set of well-defined processes, methods and tools for expressing the architecture of a system.	
formal syntax	Specification of the valid sentences of a formal language using a formal grammar NOTE 1 A formal language is computer-interpretable. NOTE 2 Formal grammars are usually Chomsky context-free grammars. NOTE 3 Variants of Backus-Naur Form (BNF) such as Augmented Backus-Naur Form (ABNF) and Wirth Syntax Notation (WSN) are often used to specify the syntax of computer programming languages and data languages. EXAMPLE 1 An XML document type definition (DTD) is a formal syntax. EXAMPLE 2 ISO 10303-21, contains a formal syntax in WSN for ISO 10303 physical files.	

Term	Definition	Source
fractional frequency deviation	The difference between the actual frequency of a signal and a specified nominal frequency, divided by the nominal frequency.	ITU-T Rec. G.810
frequency	If *T* is the period of a repetitive phenomenon, then the frequency *f = 1/T*. In SI units the period is expressed in seconds, and the frequency is expressed in hertz (Hz).	ITU-R Rec. TF.686
frequency drift	A systematic undesired change in frequency of an oscillator over time. Drift is due to ageing plus changes in the environment and other factors external to the oscillator. See “ageing”.	ITU-R Rec. TF.686
frequency instability	The spontaneous and/or environmentally caused frequency change of a signal within a given time interval. NOTE 1 – Generally, there is a distinction between systematic effects such as frequency drift and stochastic frequency fluctuations. Special variances have been developed for the characterization of these fluctuations. Systematic instabilities may be caused by radiation, pressure, temperature, and humidity. Random or stochastic instabilities are typically characterized in the time domain or frequency domain. They are typically dependent on the measurement system bandwidth or on the sample time or integration time. See Recommendation ITU-R TF.538.	ITU-R Rec. TF.686
frequency offset **(see also fractional frequency deviation)**	The frequency difference between the realized value and the reference frequency value. NOTE 1 – The reference frequency may or may not be the nominal frequency value.	ITU-R Rec. TF.686
frequency standard	An accurate stable oscillator generating a fundamental frequency used in calibration and/or reference applications. See Recommendation ITU-T G.810.	ITU-R Rec. TF.686
functional		

Term	Definition	Source
component		
functional framework		
functional requirement	Functional requirements define specific behavior (functions) or particular results of a system and its components, what the system is supposed to accomplish.	
gateway	A forwarding component, enabling various networks to be connected.	[5] ++
identification	A process of recognizing an entity in a particular identity domain as distinct from other entities.	[10]
identifier	Identity information that unambiguously distinguishes one entity from another one in a given identity domain.	[10]
identity	The characteristics determining who or what a person or thing is.	[16]
identity authentication	Formalized process of identity verification that, if successful, results in an authenticated identity for an entity.	[10]
identity domain	An environment where an entity can use a set of attributes for identification and other purposes.	[10]
identity information	A set of values of attributes optionally with any associated metadata in an identity. Note: In an information and communication technology system an identity is present as identity information.	[10]
identity management	Processes and policies involved in managing the lifecycle and value, type and optional metadata of attributes in identities known in a particular identity domain.	[10]

Term	Definition	Source
identity verification	A process to determine that presented identity information associated with a particular entity is applicable for the entity to be recognized in a particular identity domain at some point in time.	[10]
industrial internet	An internet of things, machines, computers and people, enabling intelligent industrial operations using advanced data analytics for transformational business outcomes.	[17]
infrastructure services	Specific services that are essential for a CPS/Internet of Things (IoT) implementation to work properly. Such services provide support for essential features of the IoT.	[5]
interface	Named set of operations that characterize the behavior of an entity.	[5]
internet	A global computer network providing a variety of information and communication facilities, consisting of interconnected networks using standardized communication protocols.	[16]
ip endpoint	An endpoint which has an IP address.	[17]
jitter	The short-term phase variations of the significant instants of a timing signal from their ideal position in time (where short-term implies here that these variations are of frequency greater than or equal to 10 Hz). See also “wander”.	ITU-R Rec. TF.686
latency	The latency of a device or process is the time delay introduced by the device or process.	
master data	Data held by an organization that describes the entities that are both independent and fundamental for that organization, and that it needs to reference in order to perform its transactions.	[104]
network	A generic concept that depicts the way of distributing a	ITU-T

Term	Definition	Source
synchronization	common time and/or frequency to all elements in a network.	Rec. G.810
network time protocol (ntp)	The network time protocol (NTP) is used to synchronize the time of a computer client or server to another server or reference time source, such as a terrestrial or satellite broadcast service or modem. NTP provides distributed time accuracies on the order of one millisecond on local area networks (LANs) and tens of milliseconds on wide area networks (WANs). NTP is widely used over the Internet to synchronize network devices to national time references. See www.ntp.org. See also IETF documents (e.g. RFC 5905).	ITU-R Rec. TF.686
non-functional requirement	Non-functional requirements specify criteria useful to evaluate the qualities, goals or operations of a system, rather than specific behaviors or functions of a system.	
observer	A user that interacts across a network to monitor a physical entity.	[5] ++
orchestration	The type of composition where one particular element is used by the composition to oversee and direct the other elements. Note: the element that directs an orchestration is not part of the orchestration.	[13]
oscillator	An electronic device producing a repetitive electronic signal, usually a sine wave or a square wave.	ITU-R Rec. TF.686
phase coherence	Phase coherence exists if two periodic signals of frequency M and N resume the same phase difference after M cycles of the first and N cycles of the second, where M/N is a rational number, obtained through multiplication and/or division from the same fundamental frequency.	ITU-R Rec. TF.686
phase	The term phase synchronization implies that all	ITU-T

Term	Definition	Source
synchronization:	associated nodes have access to reference timing signals whose significant events occur at the same instant (within the relevant phase accuracy requirement). In other words, the term phase synchronization refers to the process of aligning clocks with respect to phase (phase alignment). NOTE 1 – Phase synchronization includes compensation for delay between the (common) source and the associated nodes. NOTE 2 – This term might also include the notion of frame timing (that is, the point in time when the timeslot of an outgoing frame is to be generated). NOTE 3 – The concept of phase synchronization (phase alignment) should not be confused with the concept of phase-locking where a fixed phase offset is allowed to be arbitrary and unknown. Phase alignment implies that this phase offset is nominally zero. Two signals which are phase-locked are implicitly frequency synchronized. Phase-alignment and phase-lock both imply that the time error between any pair of associated nodes is bounded	Rec. G.8260
physical entity	An entity that is the subject of monitoring and control actions.	[5] ++
policy	A course or principle of action adopted or proposed by an organization or individual.	[16]
precision time protocol (ptp)	A time protocol originally designed for use in instrument LANs now finding its way into WAN and packet based Ethernet network applications. PTP performance can exceed NTP by several orders of magnitude depending on the network environment. See IEEE 1588.	ITU-R Rec. TF.686
reference timing signal	A timing signal of specified performance that can be used as a timing source for a slave clock.	ITU-T Rec. G.810
repeatability	Closeness of agreement between the results of successive	ITU-R

Term	Definition	Source
	measurements of the same measurand carried out under the same conditions as follows: • with respect to a single device when specified parameters are independently adjusted to a stated set of conditions of use, it is the standard deviation of the values produced by this device. It could also be termed “resettability”; • with respect to a single device put into operation repeatedly without readjustment, it is the standard deviation of the values produced by this device; • with respect to a set of independent devices of the same design, it is the standard deviation of the values produced by these devices used under the same conditions.	Rec. TF.686
reproducibility	With respect to a set of independent devices of the same design, it is the ability of these devices to produce the same value. With respect to a single device, put into operation repeatedly, it is the ability to produce the same value without adjustments. NOTE 1 – The standard deviation of the values produced by the device(s) under test is the usual measure of reproducibility.	ITU-R Rec. TF.686
satisfiability	In mathematical logic, a formula is satisfiable if it is possible to find an interpretation that makes the formula true.	[125]
second	The SI unit of time, one of the seven SI base units. The second is equal to the duration of 9 192 631 770 periods of the radiation corresponding to the transition between the two hyperfine levels of the ground state of the cesium-133 atom. Note: The symbol for second, the SI unit of time, is *s*.	found in IEEE Std 270-2006 (Revision of IEEE Std 270-1966); IEEE

Term	Definition	Source
		Standard Definitions for Selected Quantities, Units, and Related ... |
sensor	A sensor is a special Device that perceives certain characteristics of the real world and transfers them into a digital representation.	[5]
service	A distinct part of the functionality that is provided by an entity through interfaces.	[15]
stability	Property of a measuring instrument or standard, whereby its metrological properties remain constant in time.	ITU-R Rec. TF.686
subsystem	A discrete part of a system that groups some functionality that is part of the whole.	
syntonization	The relative adjustment of two or more frequency sources with the purpose of cancelling their frequency differences but not necessarily their phase difference.	ITU-R Rec. TF.686
system	A system is a composite set of logical components that together satisfy a concrete set of Use Cases.	
system function	What the system does. Formalized requirements.	
system of systems	Systems of systems exist when there is a presence of a majority of the following five characteristics: operational and managerial independence, geographic distribution, emergent behavior, and evolutionary development.	[103]
TAI : international	The time-scale established and maintained by the BIPM	ITU-R

Term	Definition	Source
atomic time	on the basis of data from atomic clocks operating in a number of establishments around the world. Its epoch was set so that TAI was in approximate agreement with UT1 on 1 January 1958. The rate of TAI is explicitly related to the definition of the SI second as measured on the geoid. See "second", "universal time", "UT1" and SI Brochure.	Rec. TF.686
temporal determinism	Property of a device or process whereby the latency introduced is known a priori.	
thing	Generally speaking, any physical object. In the term 'Internet of Things' however, it denotes the same concept as a physical entity.	[5]
time interval	The duration between two instants read on the same time-scale.	ITU-R Rec. TF.686
time scale (timescale; time-scale)	A system of unambiguous ordering of events. NOTE – This could be a succession of equal time intervals, with accurate references of the limits of these time intervals, which follow each other without any interruption since a well-defined origin. A time scale allows to date any event. For example, calendars are time scales. A frequency signal is not a time scale (every period is not marked and dated). For this reason "UTC frequency" must be used instead of "UTC".	ITU-T Rec. G.810
time stamp (timestamp; time-stamp)	An unambiguous time code value registered to a particular event using a specified clock.	ITU-R Rec. TF.686
time standard	A device used for the realization of the time unit. A continuously operating device used for the realization of a time-scale in accordance with the definition of the second and with an appropriately chosen origin.	ITU-R Rec. TF.686

Term	Definition	Source
time synchronization:	Time synchronization is the distribution of a time reference to the real-time clocks of a telecommunication network. All the associated nodes have access to information about time (in other words, each period of the reference timing signal is marked and dated) and share a common time-scale and related epoch (within the relevant time accuracy requirement. Examples of time-scales are: • UTC • TAI • UTC + offset (e.g. local time) • GPS • PTP • local arbitrary time Note that distributing time synchronization is one way of achieving phase synchronization	ITU-T Rec. G.8260
time-scales in synchronization	Two time-scales are in synchronization when they, within the uncertainties inherent in each, assign the same date to an event and have the same time-scale unit. NOTE 1 – If the time-scales are produced in spatially separated locations, the propagation time of transmitted time signals and relativistic effects are to be taken into account.	ITU-R Rec. TF.686
timing signal	A nominally periodic signal, generated by a clock, used to control the timing of operations in digital equipment and networks. Due to unavoidable disturbances, such as oscillator phase fluctuations, actual timing signals are pseudo-periodic ones, i.e. time intervals between successive equal phase instants show slight variations.	ITU-T Rec. G.810
traceability	The property of a result of a measurement whereby it can be related to appropriate standards, generally international or national standards, through an unbroken chain of comparisons. (ISO/IEC 17025:2005).	found in IEEE Std 1159-1995;

Term	Definition	Source
	Ability to compare a calibration device to a standard of even higher accuracy. That standard is compared to another, until eventually a comparison is made to a national standards laboratory. This process is referred to as a chain of traceability.	IEEE Recommended Practice for Monitoring Electric Power Quality; also ITU-R Rec. TF.686
universal time (ut)	Universal time is a measure of time that conforms, within a close approximation, to the mean diurnal motion of the sun as observed on the prime meridian. UT is formally defined by a mathematical formula as a function of Greenwich mean sidereal time. Thus UT is determined from observations of the diurnal motions of the stars. The time-scale determined directly from such observations is designated UT0; it is slightly dependent on the place of observation See Recommendation ITU-R TF.460. UT0: UT0 is a direct measure of universal time as observed at a given point on the Earth's surface. In practice, the observer's meridian (position on Earth) varies slightly because of polar motion, and so observers at different locations will measure different values of UT0. Other forms of universal time, UT1 and UT2, apply corrections to UT0 in order to establish more uniform time-scales. See "universal time", "UT1" and "UT2" and Recommendation ITU-R TF.460. UT1: UT1 is a form of universal time that accounts for polar motion and is proportional to the rotation of the Earth in space. See "universal time" and Recommendation ITU-R TF.460.	ITU-R Rec. TF.686

Term	Definition	Source
	UT2: UT2 is a form of universal time that accounts both for polar motion and is further corrected empirically for annual and semi-annual variations in the rotation rate of the Earth to provide a more uniform time-scale. The seasonal variations are primarily caused by meteorological effects. See “universal time” and Recommendation ITU-R TF.460. NOTE 1 – The UT2 time-scale is no longer determined in practice.	
user	An entity that is interested in interacting with a particular physical entity.	[5] ++
user endpoint	An endpoint used by a user to interact.	[17] proposed
utc : coordinated universal time	The time scale, maintained by the *Bureau International des Poids et Mesures* (BIPM) and the International Earth Rotation Service (IERS), which forms the basis of a coordinated dissemination of standard frequencies and time signals. See Recommendation ITU R TF.460. It corresponds exactly in rate with TAI, but differs from it by an integer number of seconds. The UTC scale is adjusted by the insertion or deletion of seconds (positive or negative leap seconds) to ensure approximate agreement with UT1. See “universal time” and Recommendation ITU R TF.460.	ITU-T Rec. G.810 and ITU-R Rec. TF.686
virtual entity	Computational or data element representing a physical entity.	[5]
wander	The long-term phase variations of the significant instants of a timing signal from their ideal position in time (where long-term implies here that these variations are of frequency less than 10 Hz). See “jitter”.	ITU-R Rec. TF.686

Term	Definition	Source
	Note: there is work in ITU-T SG15/Q13 to address wander/jitter associated with time signals such as 1PPS where the 10Hz breakpoint is not meaningful.	

2 Reference Architecture [RA Subgroup]

2.1 Overview

The focus of this framework is on developing a reference architecture (RA) and a vocabulary that describes it. The reference architecture will include the identification of foundational goals, characteristics, common roles and features, actors, and interfaces, across CPS domains, while considering cybersecurity and privacy and other cross-cutting concerns.

There are several key attributes of CPS that must be captured by a reference architecture for cyber physical systems (CPS). We define a CPS as follows:

> Cyber Physical Systems (CPS) integrate computation, communication, sensing and actuation with physical systems to fulfill time-sensitive functions with varying degrees of interaction with the environment, including human interaction.

We can 'unpack' this definition to identify some of these key attributes. Others are not definitional, but stem from our understanding of the context of CPS, their typical use, and their impact on the environment.

A CPS reference architecture must reflect the fact that CPS has a computational component. The range of platform and algorithm complexity is broad, so the architecture must be able to accommodate a variety of computational models.

A CPS reference architecture must also support the variety of modes of communication within and among CPS (to include no inter-device communication). The architecture must address systems that range from standalone to highly networked; limited protocols to more expressive protocols; power constrained to resource rich.

The notion of a sensing and control loop with feedback is central to CPS and must be well addressed in any reference architecture. Here again there is a wide range of complexity that should all be accommodated by the architecture including sensors that range from dumb to smart, static and adaptive sensors and control, single mode and multi-faceted sensors, control schemes that can be local, distributed, federated, or centralized, control loops that rely on a single data source and those that fuse inputs, and so on.

Equally central to the notion of a CPS is the fact that they are a product of co-design. The design of the hardware and the software are considered jointly, and tradeoffs can be made between the cyber and physical components of the system.

There is typically a time-sensitive component to CPS, and timing is a central architectural concern.

Timing requirements are generally expressed as constraints on the time intervals (TI) between pairs of system significant events. For example, the TI between the acquisition of a sensor reading and the time at which an actuator is set as a result of that reading may be *specified* to be 100µs±1µs. Similarly a bound may be required on the TI, i.e. the *latency*, between when a sensor measurement event actually occurred and the time at which the data was made available to the CPS. Likewise the accuracy of event timestamps is a constraint on a TI, in this case between the actual time of the event and the value of the timestamp. Constraints on TIs can be categorized based on their degree of time-awareness in terms of *bounded TIs*, *deterministic TIs*, and *accurate TIs.* Bounded TIs are required for CPS whose timing behavior is based on deadlines. Deterministic TIs (meaning temporal determinism as discussed in 4.3.1.3) are necessary for CPS where repeatable and precise timing relative to the system timescale is required. Accurate TIs are useful for coordinating actions in CPS of large spatial extent. Accurate TIs are sometimes required due to legal or regulatory requirements. Details on these constraints are further addressed in section 4.3.2.

Timing in general can take many forms with diverse requirements. A more extensive discussion of these can be found in section 4.3, the Timing Viewpoint, and in the Annex on Timing [144].

CPS are also characterized by interaction with their environment (as indicated by the sensing and control loop discussed above) and that environment typically includes humans. The architecture must support a variety of modes of human interaction with CPS to include: human as host of CPS; human as controller, or partner in control, of CPS; human as user of CPS; human as consumer of output of CPS.

There are other key CPS concepts which do not flow directly from the CPS definition but which need to be reflected in a CPS architecture: CPS are frequently systems of systems and the architectural constructs should be able to be applied recursively to support this nested nature of CPS. The sensing/control and computational nature of CPS generally leads to emergent higher levels of behavior and to a level of system intelligence.

To support these key concepts, the architecture itself must be constructed with several principles in mind.

The architecture must provide well-defined components. It should provide components whose characteristics are well known and described using standardized semantics and syntax. Components should use standardized component/service definitions, descriptions, and component catalogs.

The architecture must support application and domain flexibility. To do this, the definition of the components should be flexible and open ended. The architecture should support the provision of accurate descriptions of things to allow for flexibility in virtual system creation and adaption and to promote innovation. It should also support a large range of application size, complexity, and workload. The same components that are used in a very simple application should also be useable in a very large complex distributed system. Ideally the components can be adjusted and scaled quickly (even during runtime). CPS architecture should allow composition from independent, decoupled components for flexibility, robustness, and resilience to changing situations. Decoupling should also exist between vertical architectural layers allow each layer to be modified and replaced without affecting the other layers. In order for the system to integrate different components, the interfaces to these components should be based on well defined, interpretable, and unambiguous standards. Further, standardization of interfaces will allow for easy provisioning of various components by any systems envisioned today and into the future. By allowing internal component flexibility while providing external interoperability through standardized interfaces, customization can be achieved. This supports diversity of application and scalability.

CPS frequently performs critical applications, so the CPS architecture must support the level of reliability to meet requirements.   It should provide the ability of an application to resist change due to external perturbations or to respond to those changes in a way which preserves the correct operation of the critical application.

Security is a necessary feature of the CPS architecture to ensure that actions taken by CPS are not compromised by malicious agents and the information processed and transferred preserves its integrity and is kept confidential where needed. The nature of CPS not only increases the consequences of a breach but adds additional types of vulnerabilities. For example, timing in a CPS has unique vulnerabilities different from traditional data vulnerabilities considered in cybersecurity. Security needs to be built into CPS by design and to be flexible to support a diverse set of applications. This security should include component security, access control, and communications security.

Components that contain sensors or actuators (or a combination of sensors and actuators) should have an awareness of physical location. The accuracy requirement for location will change based upon the application.   It is therefore important that components can describe not only their location, but associated uncertainty of the location.

Finally, CPS architecture should support legacy component integration and migration. Legacy devices have aspects (including devices, systems, protocols, syntax, and semantics) that exist due to past design decisions, and these aspects may be inconsistent with the current architectural requirements. New components and systems should be designed so that present or legacy aspects do not unnecessarily limit future system evolution. A plan for adaptation and migration of legacy systems must be planned to ensure legacy investments are not prematurely stranded. Legacy components should be integrated in a way that ensures that security and other essential performance and functional requirements are met.

2.2 Derivation of the Framework

A useful reference for the terminological and definitional conventions relating to systems architecture and systems architecture frameworks is ISO/IEC/IEEE 42010 [2]. Let's review a couple of these for purposes of this section. An 'architecture framework' consists of the 'conventions, principles and practices for the description of architectures established within a specific domain of application and/or community of stakeholders'. An 'architecture view' consists of 'work product expressing the architecture of a system from the perspective of specific system concerns'. And an 'architecture viewpoint' consists of work product establishing the conventions for the construction, interpretation and use of architecture views to frame specific system concerns'.

We propose an extension of the ISO/IEC/IEEE 42010 terminology that will be useful in understanding our methodology. It recognizes two distinct groupings of concerns from 42010.

The first is that of a ‘facet’. Facets are perspectives on CPS that each express a distinct set of well-defined processes, methods and tools for expressing the architecture of a system. The second is the notion of an ‘aspect’, consisting of conceptually equivalent concerns[6]. Finally, we reserve the much-used term ‘domain’ to represent the different application areas of CPS as shown in Figure 4.

Two simple diagrams will help us understand how to analyze CPS using these concepts.

The first diagram shows this analysis proceeding in a series of steps:

- Start with the enumeration of domains of CPS
- **Identify** concerns; like societal, business and technical, etc.; stakeholders can have concerns that overlap or are instances of broader conceptual concerns
- **Derive** from these generic concerns, the fundamental facets of “system”, “engineering”, and “assurance”
- **Analyze** cross-cutting concerns to produce “aspects”

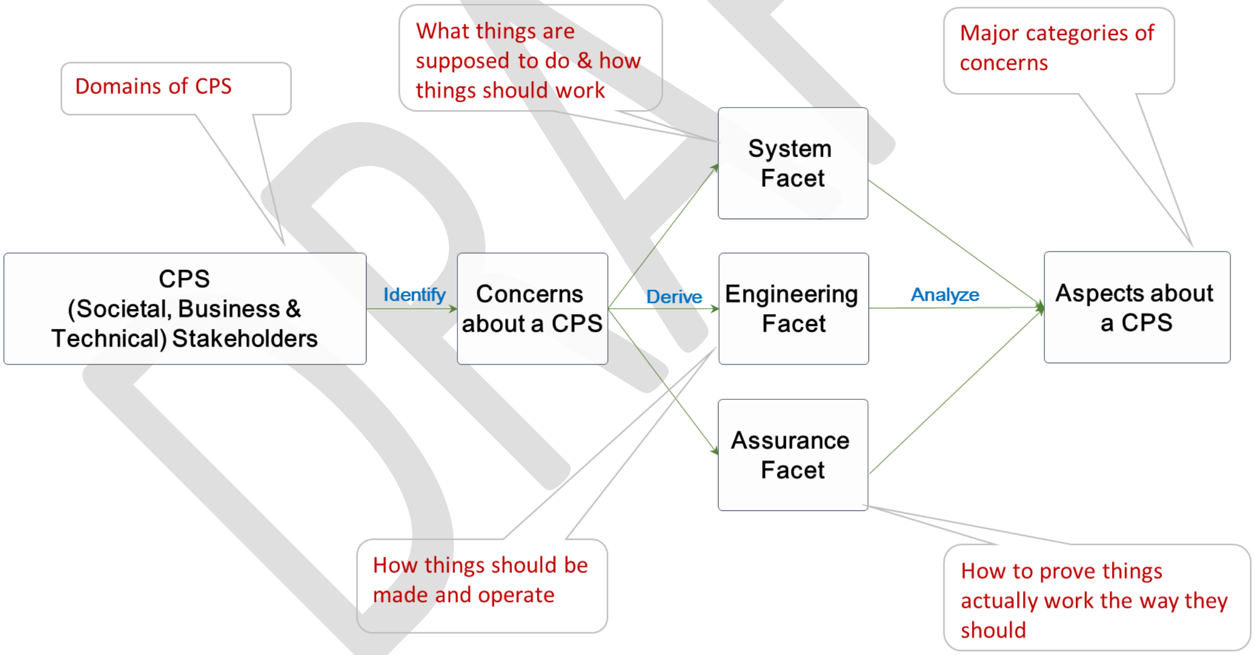

Domains of CPS
What things are supposed to do & how things should work
Major categories of concerns
CPS (Societal, Business & Technical) Stakeholders
Identify
Concerns about a CPS
Derive
System Facet
Engineering Facet
Assurance Facet
Analyze
Aspects about a CPS
How things should be made and operate
How to prove things actually work the way they should

Figure 3: Analysis of CPS and derivation of Framework

The system facet of the CPS architecture captures the functional requirements and organization of CPS as it pertains to what a CPS or component of a CPS are supposed to do and how things

[6] Aspects are sometimes called cross-cutting concerns.

should work. If we consider the design of building as a metaphor, this represents the view of the building as a whole – what the customer wants; how many floors; windows, etc… The system aspect is typically assembled in Use Cases by domain experts – those knowledgeable about the nature and operation of a CPS domain.

The Engineering facet addresses the concerns of the stakeholders that design, maintain and operate the system. Using a layered approach, it captures the different activities surrounding the processes and activities surrounding the design and implementation of such systems. Topics such as system engineering processes and tools are pertinent here. Also, modeling and simulation and other activities that help inform and actuate the design process.

The Assurance facet deals with the verification of the design. It addresses the processes, tools, and activities that deal with testing and certification of implementations of CPS. Additionally the verification of the requirements as met by designs is a topic of the assurance facet.

It is intended that the identification and description of the activities, methods and outcomes in each of these Facets can be applied to concrete CPS application domains, e.g., manufacturing, transportation, energy, etc. as a specialization of these common conceptions and descriptions. Conversely, these specializations may validate and help to enhance these conceptions and descriptions.

The domains, concerns, and facets were further analyzed producing a set of cross-cutting concerns called facets. These facets were “factored” out of the work of the various working groups that produced this framework – namely, the reference architecture, cybersecurity, timing, and data interoperability.

The result is Figure 4 which follows:

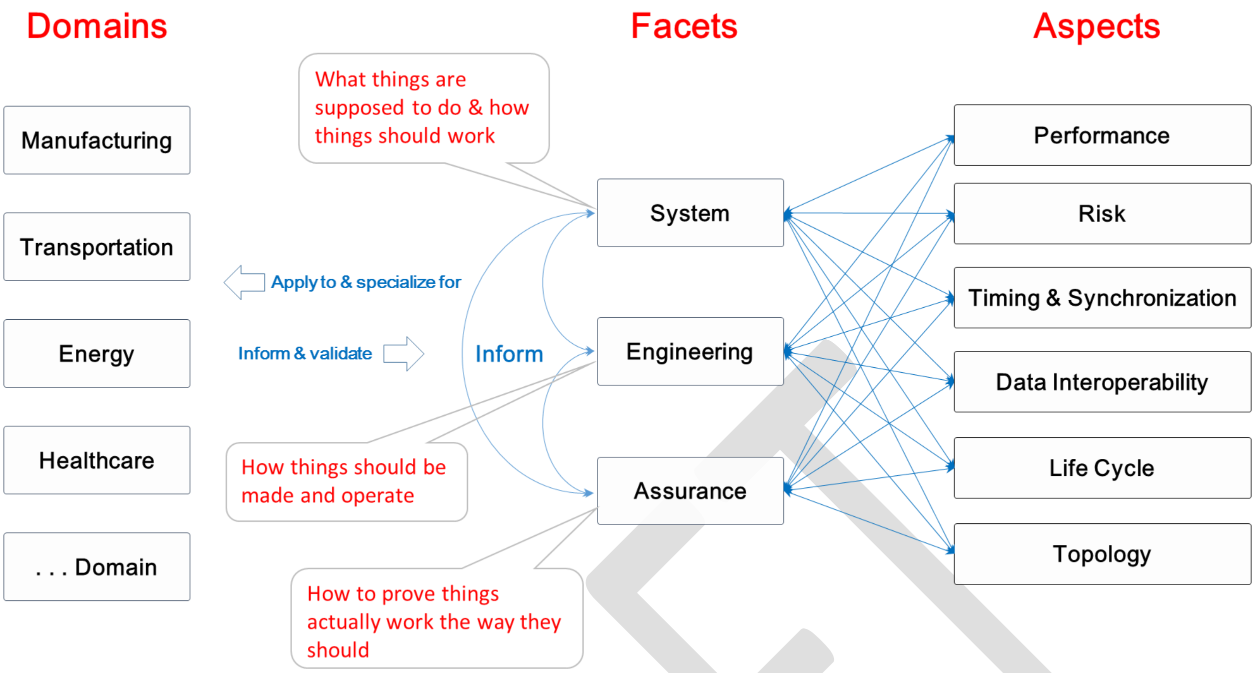


Figure 4: CPS Framework Reference Architecture – Domains, Facets, Aspects

The present section will describe the nature and content of the facets and how the reference architecture can provide for the systematic analysis, design, and verification of CPS over their life cycle.

The balance of this document will detail the aspects of the framework.

The aspects identified are:

- Performance
- Risk (which includes Security & Privacy, Safety, Reliability, and Resiliency)
- Timing and Synchronization
- Data Interoperability
- Life Cycle
- Topology

2.3 The Role of Use Cases in the Framework Development

The CPS RA should be created to serve all, or most, of the CPS requirements identified by the Use-Cases sub-group. Thus, the Vocabulary and Reference Architecture sub-group shall take into account the abstractions created by the Use-Cases sub-group when defining the CPS RA meta-model. These abstractions include business goals, domain-specific functional and non-functional requirements, and use-case constraints. The CPS RA meta-model will be constructed with these objectives in mind:

- CPS RA shall be a cognitive aid; therefore, it shall define a vocabulary and concepts unambiguously
- CPS RA shall be a common model for various CPS domains
- CPS RA shall be independent from domain-specific and application-specific standards; however, it shall be compatible with these standards
- CPS RA shall address the unique and particular challenges of CPS
- CPS RA shall enable cooperation with different standardization organizations and their activities (e.g., IEEE P2314)
- CPS RA shall adhere to the ISO/IEC/IEEE 42010 standard to be accessible to a broad audience and facilitate its adoption
- The CPS RA shall be organized in architectural views and architectural viewpoints
- CPS RA shall facilitate the analysis of CPS along their life cycle
- CPS RA shall embrace architecture divergence and provide the means for measuring qualitative and quantitative similarities and differences between the derived CPS architectures.

2.4 Related Standards and Activities

The purpose of this Section is to identify the relationships between the NIST CPS PWG activities and other related standards and working groups.

From 2010 to 2013, the European Lighthouse Integrated Project “Internet of Things – Architecture” (IoT-A) developed and proposed an architectural reference model for the IoT, referred to as the IoT Architectural Reference Model (IoT ARM) [1]. The goal of the project was to introduce a common language for fostering the inter-operability between vertical “silos” (domains) in emerging IoT applications. The IoT ARM introduces top-down architectural principles and design guidelines.

IoT-A explicitly separates itself in scope from CPS. The IoT-ARM’s functional view is organized in service layers (including communication, services, management, and security) on top of CPS. CPS, in IoT-A’s terminology, are IoT Devices (devices) and IoT Resources (software) and their architecting guidelines are not covered by the IoT ARM. It is important for the NIST CPS PWG Vocabulary and Reference Architecture sub-group to determine possible interactions with the IoT ARM.

The IEEE P2413 working group [4] was formed in 2014 to promote cross-domain interaction, aid system interoperability and functional compatibility in the IoT. The IEEE P2313 also defines an architectural framework for the IoT, including abstractions and a common vocabulary. It emphasizes a “blueprint for data abstraction and the quality quadruple (protection, security, privacy, and safety)”.

The IoT ARM and IEEE P2413 share a few important characteristics that are worth noting. Both initiatives adhere to the ISO/IEC/IEEE 42010 standard, their functional model is inspired by the OSI reference model, and they explicitly take into consideration architecture divergence. Also, both identify architecture divergence as a major topic. It is important for the NIST CPS PWG to find similarities and key differences between the scopes of IoT-related activities and CPS. This will help the reader of this document to distinguish between CPS and IoT and use the NIST CPS Vocabulary and Reference Architecture to define CPS-specific architectures that may be compatible with IoT services and standards.

2.5 Example -- Smart Traffic: an example to illustrate key architectural notions

Smart Traffic systems consisting of smart traffic monitoring and control infrastructure, advanced traffic control centers powered by predictive analytic on real-time traffic data, autonomous vehicles interacting with peer vehicles in proximity, and traffic control systems. This example will be used through this functional reference architecture to elaborate or explain the main features of the functional architectures.

CPS controls have a variety of levels of complexity ranging from automatic to autonomic

- A prominent example of cyber-physical systems in Smart Traffic, as outlined in _____ are the autonomous vehicles which are themselves system of cyber-physical systems. The functions of the cyber-physical systems within an autonomous vehicle are orchestrated, collaborated, coordinated to achieve the overall autonomous functions. (The exact technical meaning of orchestration, collaboration, coordination and autonomy will be illustrated later.)
- Another example of cyber-physical systems are the on-location smart traffic control systems installed in street intersections to sense and measure local traffic patterns and conditions, to apply commands to the traffic signals to orchestrate the movement of vehicles passing the intersections based on prescribed objectives. On the other hand, these on-location smart traffic control systems may be orchestrated by regional traffic control centers to optimize overall traffic flows.

CPS often collaborate with each other to produce larger effects.

- An example of collaboration of the cyber-physical systems is the collaboration of vehicles in proximity to avoid collisions. These vehicles communicate with each other in the cyber space dynamically forming ad hoc communities to inform others the actions each of them is taking that may affect the communities of vehicles. Examples of such actions include applying a brake or changing lanes. They also interact, albeit indirectly, in the physical space by continuously sensing and measuring the movement and trajectory neighboring vehicles. The information gathered from both the cyber and the physical spaces is then synthesized to gain an understanding of the state and intent of the vehicles in proximity. From this understanding and based on prescribed objectives (e.g. to avoid collision, a physical effect), control decisions are continuously made to produce the desired physical effects in the vehicle in question, e.g. to slow down, stop, accelerate or change course, in order to avoid the undesired ones, such as collision between vehicles or between vehicles and other objects.

CPS can be orchestrated by a cyber system that communicates logically with them

- An example of this is the computational unit in an autonomous vehicle strongly orchestrating the activities between the steering, braking and power chain cyber-physical systems. Another example of this is a traffic control unit uses wireless signaling to orchestrate autonomous vehicles passing through a street intersection.

System of Systems domain enables the complex management of CPS and supports emerging behavior.

- In Smart Traffic, traffic monitoring systems send data to the on-location traffic control units and to their respective regional traffic control centers. Vehicles also report driving data to the traffic internet, which can in turn be routed to the relevant traffic control centers. The Information component for the regional traffic control centers analyzes these data to understand the traffic conditions and patterns. The Application component synthesizes these information with other information such as traffic patterns in the neighboring regions, current and forecast weather conditions, current and pending large public events, and road accident reports. It takes into account in its model of the constraints imposed by the objectives such as minimizing traffic delay, minimizing air and noise pollution, increasing safety and enhancing security, and reducing energy consumption. It optimizes the traffic routing patterns and sends high level instructions to on-location traffic control units to orchestrate regional traffic patterns. It coordinates traffic flows of vehicles by broadcasting advices to vehicle to suggest alternative routes. The Application component may assist emergency response to accident sites for rescue and recovery. It may interact with the Business component to plan road or facility repairs on the account of both material or work crews. It may interact with the Business component to schedule predictive maintenance or repairs on the traffic control infrastructure based on information provided by the Information and Entity Management component that managing the cyber-physical systems in the traffic control infrastructure.

Furthermore, sensory data gathered from the vehicles collaborated with Geolocation, climate, and season data as well as road construction and maintenance records, can be analyzed to derive information on road and bridge conditions on precise locations, and their relations to the interworking of climate, season, pattern of usages, construction materials and procedures, maintenance frequency. Optimal preventive maintenance can be planned in relation to usage pattern, season and cost. New material and procedure can be developed that are optimal on specific usage patterns and climate.

2.6 Summary

We have presented the NIST CPS PWG Cross-Sector Reference CPS Architecture Model (CPS RA) which includes the identification of foundational goals, characteristics, common roles and features across CPS domains, while considering cybersecurity and privacy and other cross-cutting concerns. Work remains to be done to further specify this high level architecture and to identify actors and interfaces to facilitate cross-sector CPS interoperability. The CPS RA is an abstract framework, or meta-model, for understanding and deriving application-domain-specific CPS architectures. Work remains to be done to further specify this high level architecture independent from specific application domains, problems, standards, technologies, protocols, and implementations, and to identify interfaces to facilitate cross-sector CPS interoperability.

The CPS RA consists of multiple viewpoints, two of which, the Engineering Viewpoint and the Functional Viewpoint, are discussed in this section.

In the Engineering Viewpoint, CPS are described using layers typical for engineered systems: business, life-cycle, operation, CPS abstraction, and physical. However, CPS have unique characteristics, specific combinations of cross-cutting design concerns, and domain-specific architectures that span a wide range from Industrial Internet systems to different sector-specific product categories, all of which must be addressed by this Engineering Viewpoint.

The CPS Functional Viewpoint provides the building blocks to functionally derive domain-specific CPS architectures from the CPS RA and it aims at being adaptable to many industry sectors. This viewpoint is divided into two major domains, the core cyber-physical domain and the system of systems domain. The core cyber-physical domain consists of functional components that contribute to or involve in the designed functions of the cyber-physical systems. These functions at a very high level include the sensing of the physical condition and state of physical entities, executing control logic and exercising actuation to produce the desired physical effects. The system of systems domain is responsible for connecting to the cyber-physical systems, gathering data from these systems, transforming the data into information, performing analytic on the information to gain insights on a global scale about the operational states of the cyber-physical systems or the environments that the cyber-physical systems are monitoring or with which they are interacting. The system of systems domain consists of four major functional components of Information, Application, Business and Entity Management.

The Functional Viewpoint also identifies several cross-cutting functional components which require concerted behaviors among the functional components to be realized. These are connectivity, timing and synchronization, security, trust, and privacy, data analytics and interoperability, intelligent and resilient control, operational support, system integration, interoperability and composability.

Future work will address the development of additional viewpoints including Security Viewpoint, Data Integration Viewpoint, Timing Viewpoint, Usage Viewpoint, and Viewpoints to address other cross cutting concerns such as reliability, resiliency, dependability, safety, integration and composition.

The CPS RA presented here provides a set of high-level concepts, their relationships, and a vocabulary for clear communication among stakeholders (e.g., architects, engineers, users). The ultimate goal of the CPS RA is to provide a common language for describing inter-operable CPS architectures in various domains so that these CPS can inter-operate within and across domains and form systems of systems.

3 Facets of the CPS Framework

3.1 System Facet [RA Subgroup]

The CPS Functional viewpoint provides the building blocks to functionally derive domain-specific CPS architectures from the CPS RA and it aims at being adaptable to many industry sectors. For this objective, we emphasize the generality of the CPS RA and are keen not to impose unnecessary constraints to its wide applicability. At the same time, we are mindful to strike a balance between the usefulness of the CPS RA and its general applicability.

There are many ways to functionally decompose a system. Given the vast diversity in cyber-physical systems in different consumer and industrial sectors, some decomposition or abstraction approaches are more suitable to specific systems than others.

The CPS Functional viewpoint divides the overall system functions into key constituent building blocks (or functional components) and describes the structures in which these building blocks are put together to form the whole system. It describes the relationships and interactions between the building blocks to provide system-wide functions.

The functional components are recursively decomposable, some of which are done within this CPS Functional viewpoint. As the decomposition progresses, it is expected that the resulting functional decomposition will be specific and consequently less adaptable. It is foreseeable that domain-specific CPS architectures developed with this framework will contain functional structures that meet their specific use-case requirements.

This CPS Functional viewpoint describes the functional components at an abstract level and does not constrain them to any specific technologies or implementations. Furthermore, it does not make a distinction between whether a cyber-function is implemented in hardware or software. This is left to the implementation to make the best choice based on the functional requirements described in this general framework and those drawn from the specific use cases. It does, however, make a distinction between the cyber and physical functions where it is appropriate and to highlight the cyber-physical co-design requirements where it is important from the functional point of view.

There are technical requirements that can be met entirely within the functional space while other that cannot. For example, security requires functional components such as those that implement cryptography. It also requires best practice process, governance and even regulations in design, development, testing and certification across the cyber-physical boundary of a system.

There are certain capabilities that are commonly required in many functional components. To realize these capabilities, it often requires different functional components to act consistently and cohesively as a whole. For example, system security cannot be achieved by functional components in isolation and any weak link in the system would render whole system vulnerable. Consequently, these capabilities must be considered across functional components. In this framework, these functional capabilities are categorized and described as cross-cutting functions.

With this CPS functional viewpoint, we hope to provide a common and accessible framework to deal with complex cyber-physical systems. We hope that most of the functional components identified in this open and horizontal architecture can be implemented as interoperable, better yet, composable and interchangeable building-blocks regardless if they are implemented as products, hardware, software or services. Leveraging the advantage of the efficiency from specialization and the economy of scale, this would make it possible to build large and complex cyber-physical systems at lower cost by employing proven off-the-shelf system building blocks.

3.1.1 Conceptual Functional View: Systems of Systems

In this section we explore a broad concept that CPS are systems of systems which are engineered products with integrated computational and physical capabilities for automatic and, increasingly, autonomous operations, in interaction with physical entities/environment and human, to produce the desired physical outcomes. At a simpler level, a cyber-physical system may be deployed to sense and measure of the states and conditions of the physical world for a better understanding of the world we live in and the impacts that we bring about to it. This better understanding would enable better decision-making in the human interest. More often, on the other hand, a cyber-physical system may be deployed for the purpose of changing of the states of the physical entities or environment to bring about physical effects desirable by human.

At an abstract level, cyber-physical systems may be deployed

To control the flow of energy (e.g. electric grid);

To control the flow of material (e.g. oil pipeline and freight transportation);

To control the transformation from material to objects to goods (e.g. mining, fabrication, chemical refinery and production, manufactory, farming, generic engineering, etc.);

To control the movement of objects (e.g. autonomous vehicle, robots, traffic control);

To control the conversion of energy (e.g. power generation).

To control the flow of signals (e.g., air traffic control).

To control the conversion of energy, material, and signals

While some CPS may operate in isolation, many others may be required to operate in concert in order to produce these desired physical effects at large scale. To the concerted action, the cyber-physical systems are connected into clusters of systems. The cyber-physical systems in such clusters communicate with each other in the cyber space. They may also interact in the physical space. Some of the connectivity may be statically configured while some others may be dynamically established.

To orchestrate the operations of the cyber-physical systems at a global level for a given use case, the clusters of cyber-physical systems are increasingly brought online with broader systems, predominately the vast computation and communication infrastructure and business processes that have been established in the past decades, forming systems of cyber-physical systems. This is a defining concept that directly influences the consideration of the scope and structure of this functional framework.

With the global technology trends in advanced computing and manufacturing, pervasive sensing and ubiquitous network connectivity, cyber-physical systems will likely advance in two major directions:

Cyber-physical systems are rapidly shifting from the programmed to autonomous mode of operations, in other words, becoming more intelligent.

Cyber-physical systems are increasingly connected horizontally with each other and vertically with the broader systems. The horizontal connectivity paves the way for cyber-physical systems to collaborate directly. The vertical connectivity brings about the possibility of realizing a global view of the states of the vast network of the cyber-physical systems.

These new capabilities in the cyber-physical systems, fusing with the other important evolution of technologies such as social media, mobile computing, cloud computing and big data analytic is expected to bring transformational changes to the economy, the society, our knowledge of the world, and ultimately the way we live. It is important that the reference architecture should foresee and accommodate the engagements and interactions between the cyber-physical systems and these important technological developments.

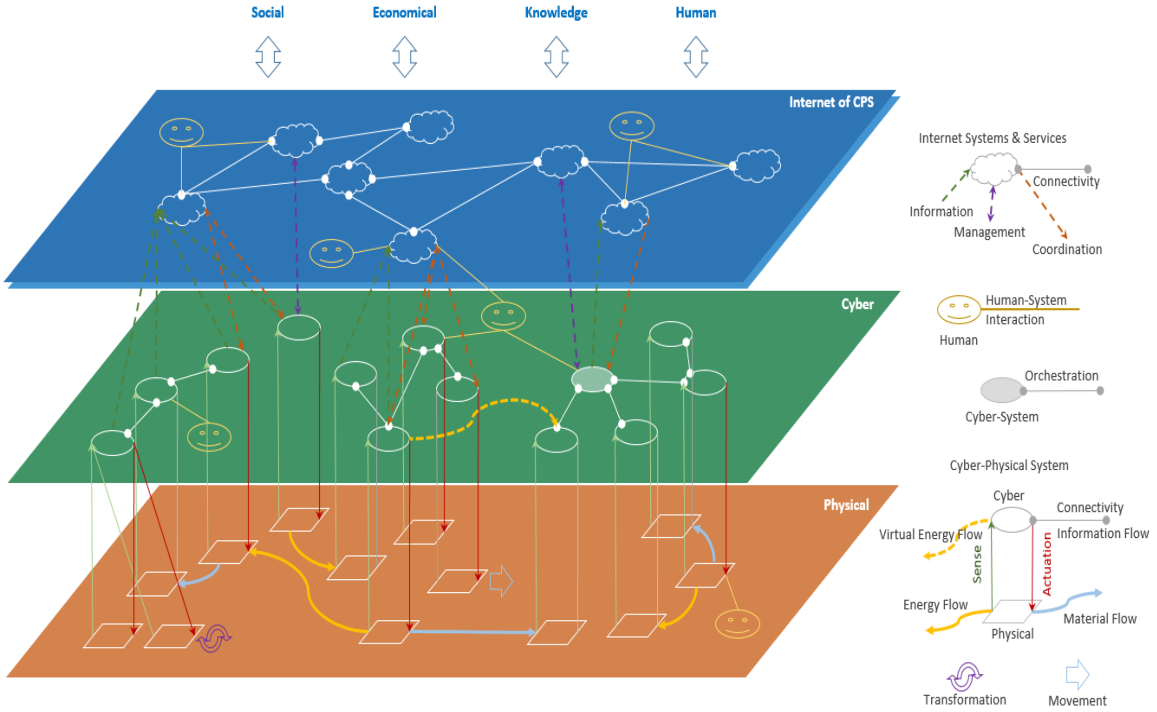


Figure 5: A CPS View: Systems of Systems

3.1.2 A Logical Functional Decomposition of the cyber-physical systems

With the general systems of systems view of the cyber-physical systems and their basic characteristics, as outlined in the previous section, a cyber-physical system functional architecture can be naturally divided into two major domains, the core cyber-physical domain and the system of systems domain, as shown in figure 3 below:

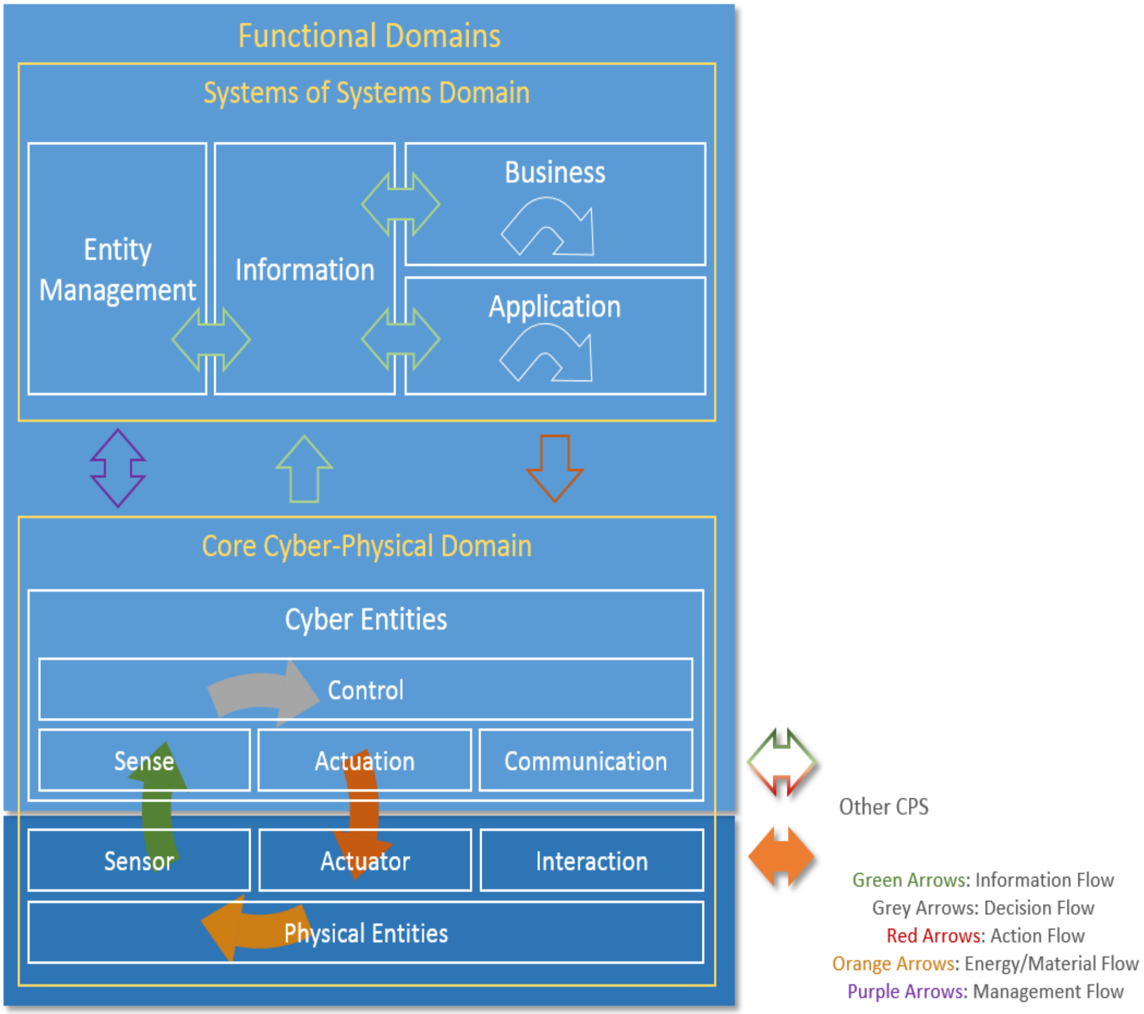


Figure 6: CPS Functional Domains

3.1.2.1.1 The core cyber-physical Domain

The core cyber-physical domain consists of functional components that contribute to or involve the designed functions of the cyber-physical systems. These functions include the sensing of the physical condition and state of physical entities, executing control logic and exercising actuation to produce the desired physical effects. Some cyber-physical systems may perform only parts of these high-level functions, such as sensing and reporting of the observed physical properties. A complete cyber-physical system typically includes all four high-level functions with the full cycle of sensing, control, actuation and the physical process forming closed-loop control to produce the desired physical effects.

This domain includes physical entities which carry out functions in the physical world; sensors, actuators and interactions which mediate between the cyber and physical entities; and cyber entities which exert control on physical entities through sense, actuation and communication. The sense/actuate control loop is a key feature of CPS.

The cyber-physical systems may have different levels of sophistication in performing the closed-loop control functions. The control logic may be fully programmed in some systems. In others it may be more flexible and open-ended allowing intelligent response based on prescribed objectives and situation-awareness. Some systems are merely automatic and others are autonomous. Some systems may only handle single input-output stream and others may be able to synthesize inputs from multiple sources and respond with multiple concerted outputs.

To complete complex tasks, many cyber-physical systems may connect to and interact with each other, forming a community or a system of systems either by configuration or dynamically. The interactions between the cyber-physical systems can be realized through either logical communication between their respective cyber components or through the physical interaction between their physical counterparts, or both. They can even be relayed across the cyber-physical boundary. Which path of communication or interaction to take is specific to the systems in question and the context in which they are operating, and it is in the domain of cyber-physical co-design. The result of co-design should be a coherent model of concerted cyber-communications and physical interactions among the cyber-physical systems to produce the desired physical effects.

In some scenarios, the activities of cyber-physical systems may be orchestrated by a cyber-system that communicate logically with the cyber-physical systems. These orchestrating cyber-systems produce no direct physical effort themselves but are required to maintain the operations of a system of cyber-physical systems. These orchestration functions depended on by the on-going operations of the cyber-physical systems are considered within the core cyber-physical functional domain.

While connectivity is important for many systems of cyber-physical systems to operate, it is important to note that connectivity should be by-design a non-deterministic factor in maintaining the operations of cyber-physical systems, at least for most of the cases. In the event that the connectivity becomes unavailable, the cyber-physical systems should be able to continue to operate locally based programmed logic or autonomous smart control, albeit in a non-optimal or even degraded mode of operations.

3.1.2.1.2 The System of Systems Domain

The system of systems domain is responsible for connecting to the cyber-physical systems, gathering data from these systems, transforming the data into information, performing analytic on the information to gain insights on a global scale about the operational states of the cyber-physical systems or the environments that the cyber-physical systems are monitoring or interacting with. The information can be synthesized with the information from other cyber-physical systems as well as the information about the environment, business, economy, social and government for better decision-making. They can also be used to achieve better effectiveness and efficiency in operations by automatically or autonomously orchestrating or coordinating the activities of the cyber-physical systems at a global scale.

The system of systems domain consists of four major functional components of Information, Application, Business and Entity Management.

The Information component provides functions for gathering data from the cyber-physical systems, transforming and persisting them where it is required, and analyzing them to provide information on the operational states of the cyber-physical systems, synthesizing information from other sources to inform the Business components and to aid the Application component in its orchestration or coordination of activities of the cyber-physical systems.

The Application component provides functions that take in information from the Information component and process these information based on prescribed objectives, rules, models to orchestration or coordinate the activities of the cyber-physical systems to achieve better effectiveness and efficiency in operations. It also interacts with the Business component to complete the activities that are required to maintain the operation of the cyber-physical systems.

The Business component provides functions that enable the end-to-end operations of the cyber-physical systems including business processes and procedural activities. These include the enterprise resource management (ERM), customer relationship management (CRM), payment systems, order systems, work planning and scheduling systems etc.

The Entity Management component provides manageability functions to the cyber-physical systems including provisioning, configuration, monitoring, update, de-commissioning, etc.

3.1.3 Crosscutting Functions

In any architecture of a complex system, there are common system capabilities and requirements that must be considered across many functional components. These capabilities, required in various functional components, may share a set of common characteristics. Furthermore, these capabilities often require concerted behaviors among the functional components to be realized. These capabilities are called crosscutting functional components. Within this functional architecture, the following crosscutting functional components are highlighted:

Connectivity

- The Connectivity deals with the functional aspects of connecting various cyber-physical entities within the cyber-physical domain and to systems in the internet domain. It covers communications, transport protocols, network structures by which the connecting entities are organized. It is within the cyber-space confine. (To be developed – need volunteers to collaborate and contribute.)

Timing and Synchronization

Timing and synchronization are essential to many CPS. Fundamentally, timing involves a physical signal, whose transfer delays must be accounted for at the required level of accuracy for the system. The physical signal may be accompanied by data, which describe it or is meant to be used with the signal. The physical nature of timing is at odds with the way data systems work, leading to core difficulties in CPS. Data systems, computer hardware, software, and networking, all isolate timing processes, allowing the data to be processed with maximum efficiency due in part to asynchrony. However, coordination of processes, time-stamping of events, latency measurement and real-time control are enabled and enhanced by a strong sense of timing.

CPS involve a marriage of the cyber and the physical: a marriage of data networking and processing systems with systems that live within the laws of physics. Generally speaking, CPS currently overcome this fundamental conflict of modern system design by using dedicated hardware and customized software for timing-critical systems. Things that require strong temporal determinism are processed as much as possible with systems that do little or no data processing. However, in many cases CPS must include significant data processing, in which case worst-case execution times are determined statistically. Computation within sensitive timing control is accomplished with statistical measures of software execution times. Development is underway to allow mixing specialized hardware for time-sensitive operations with traditional cyber techniques for best-effort systems. This is leading to converged networks safely mixing both time-sensitive and best-effort traffic.

Networks also require specialized structures to support time-sensitive operations. These issues are discussed in section 4.3, the Timing Viewpoint. We discuss the current status of such systems and point out problems and new directions that are currently in development. A later document will more fully show a roadmap for future timing systems.

Cybersecurity and Privacy

To support the definition of key aspects of CPS and accelerate their development and implementation, the CPS Public Working Group (PWG) Cybersecurity and Privacy Subgroup will identify and address the cybersecurity and privacy elements unique to CPS application domains and contexts, cumulating in the development of a set of tailored cybersecurity requirements for CPS. The work of this subgroup leverages existing approaches in traditional IT/enterprise cybersecurity and physical security. Practitioners in those disciplines have developed extensive bodies of work which were important to, scope of, this effort. The primary goal of the subgroup is to develop a cybersecurity and privacy strategy for CPS with a focus on the identification, implementation, and monitoring of specific cybersecurity activities (including the identity, security protection, detection, response and recovery of CPS elements) and outcomes for CPS in the context of the risk management process.

The following objectives address the Cybersecurity and Privacy Subgroup’s main goal, and may evolve as work progresses:

- Develop a set of qualities that can be used to describe appropriate cybersecurity objectives (e.g., confidentiality, integrity, and availability) for CPS,

- Ensure that cybersecurity is included in the overall reference architecture for CPS, and
- Identify cybersecurity and privacy requirements for the reference architecture.

As the work of the Cybersecurity and Privacy Subgroup is completed, it can be leveraged as a resource for CPS stakeholders to review and consider in the design, implementation, and maintenance of their CPS systems.

Data Analytics and Interoperability

The Data Interoperability subgroup will address the simplification and streamlining of cross-domain data interactions by developing a sound underlying framework and standards base for CPS data interoperability, in part by developing an inventory of relevant existing practices and standards. There are many CPS domains in which data is created, maintained, exchanged, and stored. Each datum has a data flow and a life cycle. Each domain naturally defines its own data semantics and exchange protocols, but those data can be difficult to understand and process when moved across domains and ownership boundaries, an increasing requirement of an increasingly connected world. This is as much, if not more so, the case in cyber physical systems as it is in other data management domains. We will address these cross-cutting data interoperability issues and point the way to the development of new efficient and scalable approaches to managing CPS data.

Intelligent and Resilient Control

- The Intelligent and Resilient Control deals with the functional aspects on how to achieve intelligent and resilient control within a cyber-physical system, among a cluster of cyber-physical systems and globally in an internet of cyber-physical systems.

Operational Support

- The Operational Support deals with the functional aspects of managing the cyber-physical systems and other functional entities in both functional domains to ensure normal operations. It covers a wide-ranging of functions entity registration, configuration, operation state monitoring, system update, decommissioning, etc.

System Integration, Interoperability and Composability

- The System Integration, Interoperability and Composability deals with how the functional building blocks are assembled together to form a complete system, how the functional building blocks interface with each other with what binding mechanisms (e.g. dynamic or static, agent-based or peer-to-peer). Interoperability and composability are both important topics in both the cyber and physical spaces. Composability imposes a stronger requirement than interoperability in that it requires building blocks not only compatible in their interfaces but exchangeable by other building blocks of the same kind that share the same set characteristics and properties such as in timing behaviors, performance, scalability and security. When a building block is replaced by another of the same kind that is composable, the overall system functions and characteristics is unchanged.

3.2 Engineering Facet [RA Subgroup]

CPS are engineered systems with a definition referring to its construction. , CPS are differentiated from other types of engineered systems in that they are constructed via the integration of cyber and physical component types and not by the specific functionalities they jointly deliver, the services they provide, or the application domain where they are used. While various definitions create stronger or weaker expectations regarding the characteristics of interactions among cyber and physical components, they all agree that CPS functionalities are the result of the tight integration of the cyber and physical sides.

The Engineering Facet of the CPS RA focuses on how CPS are made (see Figure x). Similarly to other engineered systems, the make process can be described using layers typical for engineered systems, such as business, life-cycle, operation and physical. However, CPS have unique characteristics, specific combinations of concerns, and domain-specific architectures that span a wide range of technology and application domains from Industrial Internet systems to sector-specific product categories and to societal scale infrastructures. These areas need to be understood, and then developed and supported by new foundations, methods, technologies and standards.

Figure 7 intends to capture key conceptual layers of the Engineering Facet. Each layer is associated with concepts, components and notional architectures that can be instantiated into layer and domain specific CPS architectures. Below is a short summary of the individual layers.

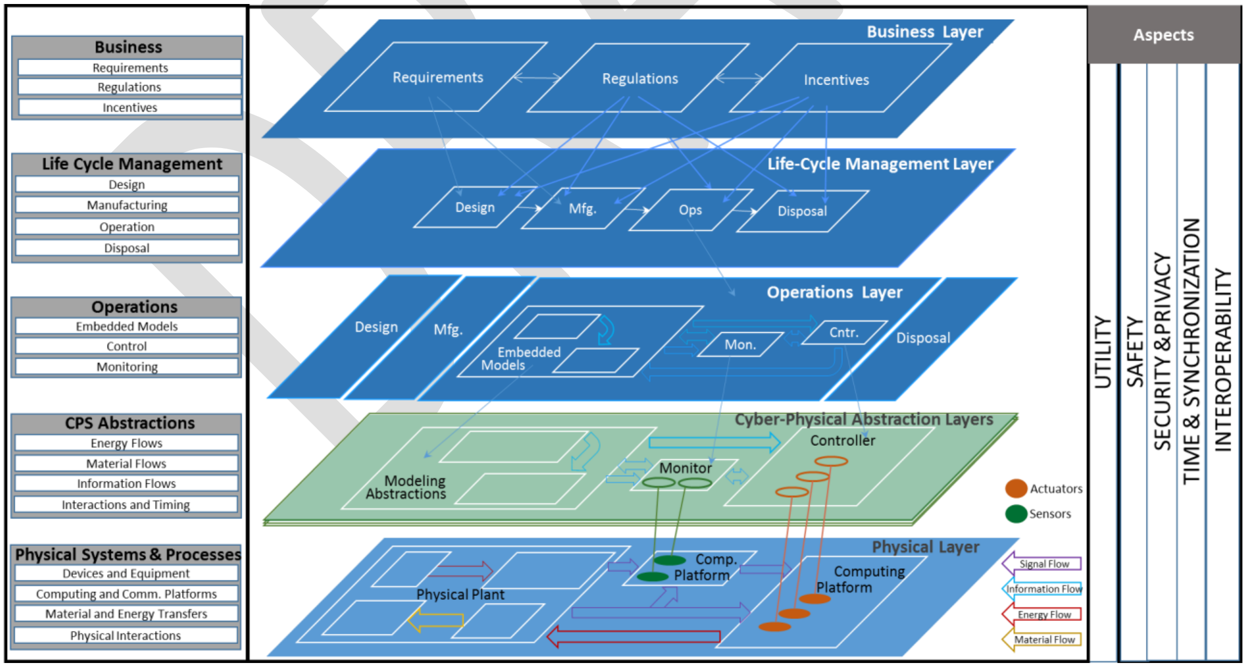


Figure 7: Engineering Facet

3.2.1 Business Layer

Evolution of CPS is driven by societal, business and individual needs, such as making transportation systems safer and more energy efficient, medical devices more interoperable, safe and secure, or the national power grid more resilient against cyber-attacks. These needs are the source of ‘Requirements’ that business enterprises respond to. A unique aspect of CPS is that in many industrial sectors CPS products are safety critical. In these areas existing and emerging government ‘Regulations’ establish constraints in addition to the requirements. In industrial sectors such as medical devices, aerospace and defense, regulations require certification processes. Frequently, existing certification methods designed for previous generation systems are in conflict with CPS technologies and create technical challenges that are not yet answered. The third essential element of the business layer is ‘Incentives’. Incentives are important tools for coupling the business layer to all phases of CPS life cycle. The emerging field of incentives engineering views the design of incentives and market mechanisms as a tool for optimizing the operation of large, distributed CPS with many conflicting operational objectives.

3.2.2 Life Cycle Management Layer

CPS lifecycle, similarly to other engineered products, covers phases from engineering design through manufacture, to operation and to disposal of products. Cyber-physical system construction has strong impact on all phases of the life-cycle. While each life-cycle phase could be further elaborated to show CPS impact, Figure xxx elaborates only of the Operations phase - to restrict the scope of the discussion.

Design: Current engineering design flows are clustered into isolated, discipline-specific verticals, such as CAD, thermal, fluid, electrical, electronic control and others. Heterogeneity and cross-cutting design concerns motivate the need for establishing horizontal integration layers in CPS design flows. This need can be answered only with the development of new standards enabling model and tool integration across traditionally isolated design disciplines.

Manufacturing: CPS manufacturing incorporates both physical and cyber components as well as their integration. As product complexity is increasingly migrating toward software components, industries with dominantly physical product lines need to change. This transformation is frequently disruptive, requires the adoption of new manufacturing platforms, design methods, tools and tighter integration of product and manufacturing process design.

Operations: CPS operations cover the phase of the life cycle where benefits of new technologies are manifested in terms of better performance, increased autonomy, new services, dependability, evolvability and other characteristics.

Disposal: Cost of disposing physical components is integral part of the overall life-cycle management process.

3.2.3 Operations Layer

CPS operations deliver the utility for users. Accordingly, the operations layer extends to functionalities and services implemented by the networked interaction of cyber and physical components. While the functional architecture of CPS is domain-specific, there are common functionalities that most systems incorporate. These common functionalities can be captured in the CPS RA. The common elements include physical and cyber entities, information flows among them; functionalities such as hierarchical control layers, monitoring, anomaly detection, self-diagnostics and contingency management systems, models that support operation, and human operators.

3.2.4 Cyber-Physical Abstraction Layers

The CPS abstraction layer(s) form a suite of structural and behavior models of systems that span both cyber and physical aspects. The abstraction layers and related modeling languages are selected according to the essential properties that need to be verified and tested during design and monitored during operation. Some of these models (for example, lumped-parameter physical dynamics of controllers of physical processes) represent behaviors that are refined during implementation to software and to physical computation platforms. Similarly, physical interactions may also be virtualized by mapping them to information flows connected to the physical world through sensors and actuators. Timing is an essential component in many CPS that relies on precisely coordinated interactions between physical and computational processes. In these systems, challenges go well beyond the introduction of physical time abstractions in computing that has a rich history in real-time computing. New challenges and opportunities emerge from integrating the rich concurrency models in computing with time abstractions in physical systems and finding solutions for managing timing uncertainties.

Abstraction layers are usually defined by modeling languages that capture the concepts, relations and well-formedness rules that each model must satisfy. In another word, modeling languages introduce invariants that all design (captured in the modeling language) satisfies. An important role of selecting modeling languages (i.e. abstraction layers) is to ensure that essential properties (such as stability or timing) are guaranteed by the introduced invariants.

Among the many abstractions that are applied to CPS, functional abstractions are of special interest. A functional abstraction involves a decomposition of a complex CPS into its logical and abstract constituent functional components, which can be integrated and composed to form the overall functions. Because of reduced complexity, these functional components are easier to understand, design and implement. The logical decomposition also allows the grouping of similar functions into their respective components. This in turn offers the opportunity for specialization of functional components. All these make it easier to understand and design the overall functions of a CPS.

The functional abstraction describes how a CPS is logically decomposed into components and a structure in which these components relate to and interact with each other to form the full system functions. It is abstract in nature and does not constrain the technology and implementation choices by which functional components are realized. Specifically, it does not necessarily prescribe if the functions are implemented solely in the cyber domain, the physical ones or both.

In this document, we refer this functional abstraction as the CPS Functional Facet and discuss it in substantial details in Section 2.4.

3.2.5 Physical Layer

All CPS incorporate physical systems and interactions implementing some forms of energy and material transfer processes. Physical systems include plants, computation and communication platforms, devices and equipment. CPS abstraction layers explicitly model the structure and behavior of these physical processes and express their relations to cyber models by linking information flows to physical variables via sensors and actuators and modeling the deployment of computations and information flows to platforms. Consequently, CPS design flows do not abstract out physicality in computations but consider the implementation side effects of computations and networking on abstracted behaviors.

3.2.6 Crosscutting Aspects <to be revised>

While system complexity largely depends on the extent and richness of interactions across components, design complexity is strongly influenced by the number of and interdependence among design concerns. Just like restricting and controlling interactions in systems is a key to decrease behavioral complexity, “separation of concerns” is the most frequently applied engineering principle to mitigate design complexity. Crosscutting concerns are essential in the engineering process of making, operating and retiring CPS because they have influence in all (or most) layers thereby limiting the applicability or effectiveness of the separation of concerns principle.

Figure xxx captures five major categories concerns: utility, safety, security and privacy, time and synchronization and interoperability.

Utility is the primary driver of creating a CPS. It captures concerns that carry values for users, and usually expressed as delivered functionalities, related capabilities and performance metrics. Many design tradeoffs are expressed in terms of compromise between utility and some other category of concerns.

Safety properties of CPS express their capabilities for mitigating and avoiding hazards. In many CPS domains safety considerations are one of the key factors that influence decisions in all system layers. For example, in safety critical CPS, regulations may require certification of safety properties that in turn motivate the selection of architectures and design methods for verifiability, exert influence on manufacturing, testing and system operation, and determine the level of abstractions used for modeling physical components and processes and impose restrictions on acceptable physical architectures.

Cybersecurity and privacy have emerged as a major concern in CPS. As opposed to information technology (IT) cybersecurity that focuses only on mitigating the impact of cyber-attacks, CPS cybersecurity and privacy is extended to the coordinated exploitation of both physical and cyber vulnerabilities. Impacts of cybersecurity and privacy considerations are pervasive on multiple layers of a CPS instance.

Time and synchronization are fundamental concerns due to the inherent role of time in the physical side of CPS. This category of concerns lead to services and protocols necessary to:

- Ensure that the temporal aspects of data that are common to more system components are based on a common understanding of time so that logical operations and computations on these data are meaningful.
- Ensure that ordering of system-wide operations based on some defined temporal relationships are correct.
- Enable the explicit use of timing and synchronization abstractions in complex, distributed CPS.

These services and protocols may include one or more of the following (and perhaps others):

- Implementation of and interfaces to a system-wide common timescale.
    - Any service providing time synchronized to a timescale or a translation between timescales, includes a method for removing the delay from the source of the timescale and is a real-time service. "Real-time" here means that the time accuracy relative to the source timescale is sufficient for the needs of the application when it arrives at the client.
    - When required or preferred the timescale is traceable to TAI or UTC
    - Provide translation functions to relate timescales internal to a layer or an entity within a layer to the system-wide common timescale. For example, at the physical layer often only self-consistent time is required. If data from such a timescale must be used elsewhere in the system then the translation service is invoked.
- Using synchronized time in a network to achieve determinism (bounded latency and guaranteed bandwidth) over the network which is key for distributed control loop operation.
- Global timeout and/or global event notification services.
- Scheduling at all layers, varying from precision scheduling in control loops to scheduling for billing and shipping.
- Logical ordering protocols and services, e.g. a system-wide mutex, system-wide event queue.

Interoperability and Compositionality are key concepts in engineering systems from components or developing system of systems. Interoperability means that system components are able to exchange data based on a shared interpretation and able to interact to coordinate operations. Compositionality means that properties of composed systems can be computed from properties of its components. Compositionality is crucial in integrating large systems. If conditions for compositionality are satisfied, it ensures "correct by construction" i.e. the elimination of design-manufacture- build- test –re-design iterations. Achieving interoperability and compositionality in CPS have many open challenges due to the impacts of heterogeneity.

3.3 Assurance Facet [TBD Subgroup]

TBD

4 Aspects of the CPS Framework

4.1 Risk Aspect

4.1.1 Overview

Complex systems-of-systems integrating the cyber and physical worlds, often referred to as cyber-physical systems (CPS)[7], will extend the functionality and capabilities of existing information technology (IT), operational technology (OT)/industrial control systems (ICS), and embedded systems. CPS provide an opportunity to leverage multi-disciplinary approaches as technologies converge to shape continued and future innovation across countless sectors of national and international economies. Influenced by common technical and business drivers such as interoperability and standards-based platforms, a need for common reference architectures, and growing consumer/user needs, CPS will require international, cross-sector collaboration to realize anticipated benefits.

CPS will provide the next generation of “smart,” co-engineered interacting components connected over diverse networks. Composed of heterogeneous, potentially distributed, components and systems, CPS bridge the digital and physical worlds. Assuring that these systems are trustworthy (e.g., reliable, resilient, secure, available, and safe) and protect the privacy of information and users poses unique cybersecurity challenges. Traditional approaches to cybersecurity and privacy, reliability, resiliency, and safety may not be sufficient to address the risks to CPS. This results in a need for a cross-property risk management [18] approach for CPS that understands the risk management approaches from historically disparate areas of expertise. To support the co-design aspect of CPS, a deeper understanding of the relative significance and interaction between each of these properties is necessary to ensure the functionality of the CPS is not compromised or results in unintended outcomes. Through this cross-property understanding, appropriate CPS design trade-offs and complementary cross-property design decisions can be made.

The following sections will highlight the unique elements for the risk properties of CPS and how they relate to and impact the other properties in the context of CPS: i) Cybersecurity and Privacy, ii) Safety, iii) Reliability, and iv) Resiliency.

4.1.2 CPS Cybersecurity and Privacy Risk

In its broadest sense, cybersecurity for CPS will require significant operational and use changes that will impact how systems and applications are deployed across legacy and new systems. New standards, affecting design, engineering configuration, automation, and communication

[7] The draft consensus definition (October 17, 2014) by the Cyber Physical Systems (CPS) Public Working Group (PWG) Reference Architecture Subgroup follows: *CPS integrate computation, communication, sensing and control with physical systems to fulfill time-sensitive functions with varying degrees of collaboration and interaction with the environment, including human interaction.*

need to be instituted to ensure a favorable outcome. When considering cybersecurity for CPS, it is important to focus on the physicality of these systems, and the operational constraints that are attendant upon that physicality, makes to our CPS cybersecurity strategy. Certainly many of the cybersecurity challenges that apply to IT systems apply to CPS as well. However, some challenges may not have the same criticality in the CPS space as they do in IT systems, and CPS may pose additional challenges that are not present in the IT space. Further, the mechanisms used to address IT challenges may not be viable in the world of CPS. The physicality of CPS also presents some opportunities for cybersecurity solutions that are not available to IT solution providers.

4.1.2.1 Cybersecurity Challenges

4.1.2.1.1 Overarching Issues

Perhaps the most significant challenge in providing cybersecurity for CPS is addressing the requirement for resilience. CPS cybersecurity must protect operational goals from the impacts of malicious cyber-attack, so cybersecurity mechanisms must enable safe and live operations even in compromised conditions. Cybersecurity for CPS must address how a system can continue to function correctly when under attack and provide mechanisms that support graceful degradation in accordance with mission- or business-driven priorities, and enable the system to fail-safe or be fault-tolerant in those circumstances in which resilience cannot be provided in the face of threat.

Providing cybersecurity for CPS is further complicated by the fact that CPS operate under a wide range of operational conditions. Security solutions must encompass that breadth. On one extreme are the safety-critical systems. These systems are often highly regulated, generally physically protected, and almost always the product of careful design and significant capital investment. On the other end of the spectrum are consumer convenience or entertainment devices. These systems assume no limits on access, and are produced in a variety of development environments (some of which are relatively unstructured). Cybersecurity and Privacy professionals cannot afford to focus more on one end of the spectrum than the other, because these operating conditions are converging. Consider wearable or implantable medical device: they are safety critical, somewhat regulated, but exhibit limited physical protection, are almost always accessible, and produced and used in environments similar to the consumer goods environment. Yet security and privacy considerations are as critical to this system's safety and integrity as they are for an industrial controls, or critical element of the power grid.

The system-of-systems nature of many CPS introduces another challenge to the cybersecurity of CPS. A system-of-systems emerges, and is not necessarily designed as a coherent system. Understanding and addressing upstream and downstream dependencies of the component system, boundaries of the "system" are often unclear and ever changing, making cybersecurity analysis and the design of cybersecurity mechanisms more complicated. Where the composite system consists of components owned by multiple entities, there is also the issue of determining responsibility for the security of the whole CPS or how responsibility is shared or trust relationships are established among responsible entities to assure global protection.

The extreme scalability of CPS also presents challenges. The emergence of the Internet of Things increases the number of connected entities on a scale that dwarves current IT networks. Huge networks of small sensors are becoming more commonplace. Security mechanisms, and the infrastructure to manage them, must be able to scale up to accommodate these structures.

4.1.2.1.2 Challenges due to interaction with physical world

Another set of challenges for CPS cybersecurity stems from interaction with the physical world. Perhaps the most obvious of these it that the impact of attacks on a CPS can be physical catastrophic – attack a CPS, and things can result in impacts on quality and safety, damage, and in some cases, lead to catastrophic effects. This means there is a different level of tolerance for threats against CPS, and a different level of urgency in addressing attacks. A denial of service attack against a website means loss of access, perhaps loss of revenue or even damage to a server, but if the attack is addressed in minutes, it is generally not difficult to recover. By contrast a denial of service attack against the system that regulates the safe operation of an industrial plant can lead to irreparable damage to capital equipment that could take months to replace. In this case, the time scale to address the attack cannot be minutes. In addition, CPS are deployed in ways that preclude physically securing all the components. This increases the likelihood that cybersecurity processes will be operating in a compromised environment.

Because CPS interact with the physical world, they are subject to the time constraints of the physical process they are executing. These processes are generally time-aware and deadline-sensitive, so security processes must fit within the time constraints of the application. Current IT cybersecurity controls may need to be modified significantly, or be completely replaced, because those solutions cannot be applied to CPS. Further, the real-time constraints on addressing attacks rule out human-in-the-loop solutions. This drives requirements for continuous, autonomous, real-time detection and response.

4.1.2.1.3 Challenges due to operational constraints

The fact that the operational settings of CPS are often very different from those of IT systems, particularly enterprise systems, challenges application of existing cybersecurity paradigms for CPS. Moreover, the operational settings and requirements vary greatly across the range of CPS, so the challenges are not uniform for all CPS, thus, it is useful to consider a variety of operational implications for CPS cybersecurity.

CPS often exist on resource-constrained platforms. As a result, security mechanisms must be lightweight in terms of storage space, memory use, processor use, network connectivity, and electrical power consumption. Furthermore, constrained platforms are often distributed; the individual components must perform global tasks using local information exchange and limited computation at the nodes.

Cybersecurity for CPS generally must accommodate the in-place business processes. access controls, authentication, and authorization mechanisms must accommodate the fact that CPS are often deployed in operational situations which require immediate access to control systems or access by any member of a group. “Strong” passwords, passwords that are lengthy or complicated to enter, or require frequent updates are often inappropriate for such environments. These passwords are often shared among all the individuals holding a particular role to eliminate potential discontinuity between shifts and provide rapid emergency access to the system. New mechanisms to establish trust between machines and people are needed for these conditions.

CPS often have "always on" requirements. This makes rebooting and patching non-viable strategies for many systems. Furthermore, the software that executes processes in many of these systems has often undergone extensive analysis and testing to meet safety requirements, so cannot be easily changed since the cost of implementing changes is prohibitive.

In several CPS sectors (e.g. transportation, emergency response), the domain of use is dynamic. Actors, be they people or machines, come and go. The set of valid users is constantly changing, and at an ever quickening pace. Therefore traditional key management is ineffective over large “accidental” populations of this type. For example, the impact of providing keys all the driver-assisted or autonomous vehicles on any major road during peak traffic. Without new keying mechanisms and protocols under such dynamic conditions, encryption mechanisms are not likely to work. The dynamism of system configuration is increased by two other facts: in many use cases nodes are intermittently unavailable; and some nodes change context (and the attendant security requirements) depending on the task at hand. The variable reliability of human participants also adds to system dynamism.

4.1.2.1.4 Lifecycle Issues

A number of lifecycle issues also complicate the cybersecurity of CPS. Some operational technology and infrastructure CPS have very long lifetimes (30 years or more). These systems are difficult to change; industry needs strategies that both “future-proof” designs and allow for integration with legacy systems. In some cases, the verification cost of these systems locks owners into old technology; they need methods that enable rapid reassessment and conjoined maintenance of new and legacy systems. This raises challenges associated with composability; therefore, the new system designs should include consideration of accommodating existing devices.

The more agile consumer and sensor CPS also highlight the problem of orphaned equipment or stranded assets that remain in use long after support has been discontinued. This equipment cannot be made resistant to emerging threats; rather, it poses a risk to any network to which it is connected. Additional challenges can be introduced by inappropriate use of throwaway systems, which have a limited lifespan by design, but which are never removed from the environment and can be co-opted in an attack. In both the static and the agile environments, there is a need to understand lifecycle threats and take a systems engineering approach to address the security of the manufacturing process, supply chain, commissioning, operation, and decommissioning of devices.

4.1.2.2 Privacy Challenges

When considering privacy protection in CPS, it is critical to keep in mind how CPS interact with the physical world. In short, the impact of a CPS privacy violation can be quite different from that of an information privacy violation. If an individual's privacy is broached in a CPS context, attackers do not merely gain access to the individual's information, they can impact physical systems without permission, manipulate or modify individuals' behavior by constraining choices and opportunities in the physical world.

There are cases in which the actual data in a CPS has no privacy implications in isolation but can be used in aggregate that would be privacy intrusive. The by-product of CPS data collection without regard to privacy is an issue. The privacy analysis of a system should consider not just what data is created by that system, but what set of data could reasonably be created (or aggregated) without users' knowledge on basis of these presumably "innocuous" observations. The privacy risk assessment must also consider other systems that are receiving the data. Complete consideration should also address the "data exhaust" problem and provide a strategy for the deletion or protection of data such as long tail measurements produced over the life of a system.

Some privacy concerns that plague information systems can be exacerbated in CPS. The system-of-systems nature of CPS produces highly complex interrelationships among systems. How can the threshold for aggregation of data points across these many interconnected elements be determined? CPS data is often collected for the sake of the management of the system, not for any user-driven purpose. Ownership of data is unclear; For instance, does a Utility or its customers own meter data? The value of the data is in many instances divorced from its owner/target. In these cases, designers are responsible for characterizing the tradeoffs between the gains made by the collection of such data (forecasting, non-technical losses/revenue protection, etc.) vs. the privacy costs/losses experienced by consumers.

In addition to data leakage, users also leak information through simple "data exhaust". Non-Intrusive Load Monitoring (NILM) leaks device usage information through the power line. The simple act of turning an automobile leaks information on route. Water and gas flow changes leak information about control structures. CPS in general leak information that no amount of encryption can protect. From the privacy viewpoint, this question must be considered expansively.

4.1.2.3 Opportunities

Though the nature of CPS introduces many challenges to cybersecurity, it also present some opportunities that may enable novel approaches to securing these systems or make viable some approaches that are difficult to implement in the more open world of IT. The laws of physics often constrain operations of CPS and the normal behavior range of CPS is often well understood. These features may make anomaly detection and control easier. CPS have comparatively simple network dynamics: servers change rarely, the topology is often fixed, the user population is relatively stable, communication patterns are regular, the number of protocols of protocols is limited. These parameters can be modeled, and the model of the dynamics of the system can be used to detect a compromised node or identify out-of-norm behavior. Because of these more limited dynamics, it is possible to consider use models which can adjust connectivity of a system based on criticality and business needs and limit connectivity that does not address some mission or business need. However, the drive to smart systems is fueled by increased connectivity and fusion of information; thus, security professionals' desire to limit connectivity will constantly intrude upon the potential for improved functionality that additional connectivity will enable.

The deployment strategies used for CPS present several possibilities for novel protection strategies. CPS are often highly distributed and provide multiple observations of the same, or highly related, phenomena. This multiplicity could be used to devise new means of providing data integrity by leveraging the multiple viewpoints. Although the challenges associated with upgrading legacy CPS are discussed above, the addition of new systems into the legacy environment also provides opportunities. The new components can monitor or protect their older comrades, or serve as wrappers that enable the old technology to participate in new protection strategies. As more "smarts," processing power, capability, and control move to the system edges, additional protection nodes that are robust enough to protect themselves and the system of which they are a part can be added.

The fact that many CPS are safety-critical systems also provides some opportunity for improved cybersecurity. Systems that often undergo rigorous analysis for safety and cybersecurity may be able to leverage this analysis in the context of a threat model to devise protections. Some of the safety controls already in place in CPS can mitigate the effects of some types of cyber-attack, thus providing mechanical and non-cyber solutions to cybersecurity problems. Safety-critical systems are also often designed with redundancy, which cybersecurity engineers can leverage to provide resilience. In contrast, low power systems are minimally designed. This opens the door for resilience strategies that rely on redundancy in infrastructure rather than at the endpoints.

The CPS PWG Cybersecurity and Privacy Subgroup concludes that identifying the specific properties of CPS that are unique from IT can help system designers and cybersecurity professionals to tailor existing cybersecurity solutions or identify new ones that are well suited to this domain.

4.1.2.4 The Design Response

The unique characteristics of CPS must be considered when designing and developing secure CPS. Trust analysis of CPS architecture must understand the physical properties and constraints of the system; such analysis must include design, analysis and up-to-date adversary models. There is also a need to design proactive, real-time, autonomic algorithms and architectures that can defend dynamically against given changing adversary models. Incorporating dynamic models of systems being controlled can help increase understanding of impacts of attacks and leverage this understanding of the consequences of attacks to reason about what the attacker might do should he/she gain access. To address privacy protection, CPS owners and operators need purpose-aware collection of data, which enables system owners to collect only what is needed, at intervals that are tuned to the needs of the application. System designers should consider privacy risk and trade operational gain versus privacy loss.

4.1.3 CPS Cybersecurity: Moving from Classic Cybersecurity Properties to Cross-Property Risk Management

Note to Reviewers: *The ideas presented in this section, CPS Cybersecurity: Moving from Classic Cybersecurity Properties to Cross Property Risk Management, do not yet reflect consensus of the Cybersecurity and Privacy working group, rather intends to capture many of the concepts initially discussed by a smaller subteam.*

4.1.3.1 Properties Defined

This section defines the properties of CPS risk management and explains the relevance of these properties to CPS.

- **Security (or cybersecurity):** A condition that results from the establishment and maintenance of protective measures that enable a system to perform its mission or critical functions despite risks posed by threats to its use. Protection measures may involve a combination of deterrence, avoidance, prevention, detection, recovery, and correction that should form part of the enterprise's risk management approach [CNSSI 4009].
- **Privacy:** [In development]
- **Safety:** Absence of catastrophic consequences on the user(s) and the environment (IEEE Transactions on Dependable and Secure Computing) or freedom from unacceptable risk of physical injury or of damage to the health of people, either directly, or indirectly as a result of damage to property or to the environment (IEC).
- **Reliability**: The ability to provide a consistent level of service to end users (Disaster Resilience Framework, 50% draft) or continuity of correct service (IEEE Transactions on Dependable and Secure Computing).
- **Resilience:** The ability to prepare for and adapt to changing conditions and withstand and recover rapidly from disruptions. Resilience includes the ability to withstand and recover from deliberate attacks, accidents, or naturally occurring threats or incidents. (Presidential Policy Directive (PPD)-21: Critical Infrastructure Security and Resilience).
- **Timing**: is a fundamental dimension of the design and operation of CPS. The use of time in a CPS node is typically complemented by the node's positional coordinates, as the space-time continuum of physics is now an engineering reality. Since timing is fundamental to the operations of CPS, any disruption or corruption of the timing will affect CPS operations, and under some circumstances could cause CPS operations to cease altogether. Hence, the disruption or corruption of timing poses another one of the many risks to CPS in general.

Together in the context of CPS, the risk management properties defined above support the trustworthiness of the system – "the system does what is required despite environmental disruption, human user and operator error, and attacks by hostile parties and not other things" (Fred B. Schneider, Trust in cyberspace). To achieve trustworthiness of a system is greater than the sum of trustworthy parts.

As defined by the CPS PWG Reference Architecture Subgroup, *"CPS integrate computation, communication, sensing and control with physical systems to fulfill time-sensitive functions with* *varying degrees of collaboration and interaction with the environment, including human interaction."* Given the scope of CPS, traditional enterprise IT approaches and solutions cannot exclusively address the relevant cybersecurity and privacy needs. CPS owners and operators may need additional risk management properties. These will vary based on system functionality and operational needs. The analysis of illustrative examples by this subgroup led its members to concludes that the above five properties of risk management applied most broadly across the diverse breadth of CPS.

4.1.3.2 Cross Property Nature of the Threat

CPS owners and operators, who have traditionally been concerned with system risk in terms of safety, reliability, resilience, physical security and privacy, have good reason to also be concerned about cybersecurity. Users need systems that will behave as expected, even under stress due to attacks [52]. Confidence that the system will perform as expected is especially critical to CPS because they have potential to cause harmful effects in the physical world. To gain that confidence, we need a risk management approach that considers cybersecurity in the same analysis as safety, reliability, resilience, physical security, and privacy. The case of the Stuxnet worm [53] illustrates the importance of cross-property risk analysis for CPS.

> [Stuxnet] was a 500-kilobyte computer worm that infected the software of at least 14 industrial sites in Iran, including a uranium-enrichment plant. It targeted Microsoft Windows machines and networks, repeatedly replicating itself. Then, it sought out Siemens Step7 software, which is also Windows-based and used to program industrial control systems that operate equipment, such as centrifuges. Finally, it compromised the programmable logic controllers.
>
> The key compromise was that Stuxnet placed itself in a critical path where it could not only disrupt the plant process, but also disrupt/manipulate the information flow to the system operator. In this particular instance of Stuxnet, it caused the fast-spinning centrifuges to tear themselves apart, while fabricating monitoring signals to the human operators at the plant to indicate processes were functioning normally.
>
> Stuxnet could spread stealthily between computers running Windows—even those not connected to the Internet [via infected USB drives]. It exploits vulnerabilities associated with privilege escalation, designed to gain system-level privileges even when computers have been thoroughly locked down. That malware is now out in the public spaces and can be reverse engineered and used again against CPS.

Stuxnet used the cyber interface to the target system to impact its physical operation and cause safety and reliability concerns. In concept, malware with capabilities similar to those displayed by Stuxnet could maliciously alter the operational state of any CPS by compromising cyber subsystems (e.g. digital data feeds from sensors, digital files used by cybernetic control systems to control machine operation, and digital data storage used to record system state information) in ways that adversely affect safety, reliability, resilience, privacy and financial bottom lines. Such malware could also collect and exfiltrate intellectual capital that could inform attackers' future attempts to threaten system performance. Managing risk associated with CPS cybersecurity, therefore, requires consideration of these properties along with classic IT security concerns.

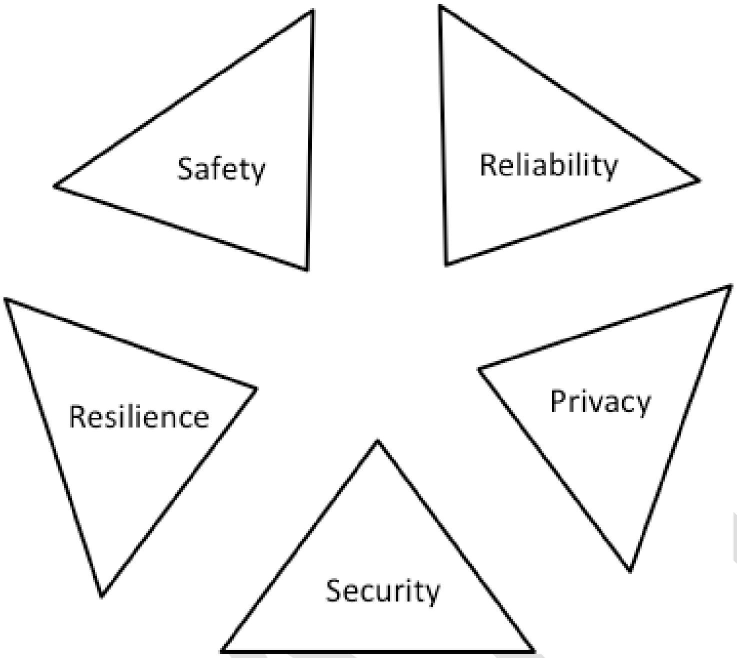


Figure 8 - Historically systems design occurred within disparate disciplines. The disciplines were prioritized based on domain-specific (energy, manufacturing, transportation) requirements and perspectives.

The properties of safety, reliability, privacy, security and resilience have, for the most part, evolved within distinct silos. Large systems engineering and integration projects often have property-specific leads, who represent discrete viewpoints within the trade-off process overseen by the chief systems engineer/integrator. Functional requirements often have lead engineers and designers to prioritize each property differently, but achieving a level of success in each property typically is vital to the overall success of the system. Likewise, risk management activities have often been conducted within each silo, rather than across them. The future of CPS design, integration and risk management, however, appears to be evolving toward a multi-disciplinary approach where systems designers and integrators are increasingly required to work across properties, with the increasing imperative to provide cybersecurity becoming a common requirement for all. Ideally, personnel responsible for each property will consider the interdependencies among all five properties throughout the system lifecycle.

Stuxnet illustrates how the continuing integration of cyber technology into traditional systems is breaking down silo walls. “Cyber technology” exploited by Stuxnet included the data interfaces, digital data pathways and digital sensors used to compromise the PLCs associated with centrifuge control. Machines built with locally isolated controls were “connected” by a USB interface designed to offer greater convenience to workers. The interface unwittingly permitted transfer of cyber-attack payloads across an air gap. The operational systems used to deliver services in many critical infrastructure sectors and in plants that manufacture goods, including national security systems, use similar configurations.

Stuxnet's principal objective appears to have been to cause physical damage to centrifuges. Its developers determined that a cyber-payload could use digital data to manipulate the mechanical and digital components of the centrifuge system such that the centrifuges would damage or destroy themselves. Having designed the payload, the individuals behind Stuxnet only needed a way around the cyber protections to achieve harmful effects that were typically the concern of other risk management properties. Stuxnet used the cyber interface to effectively overcome the safety, reliability, privacy, security and resilience provisions of the target systems.

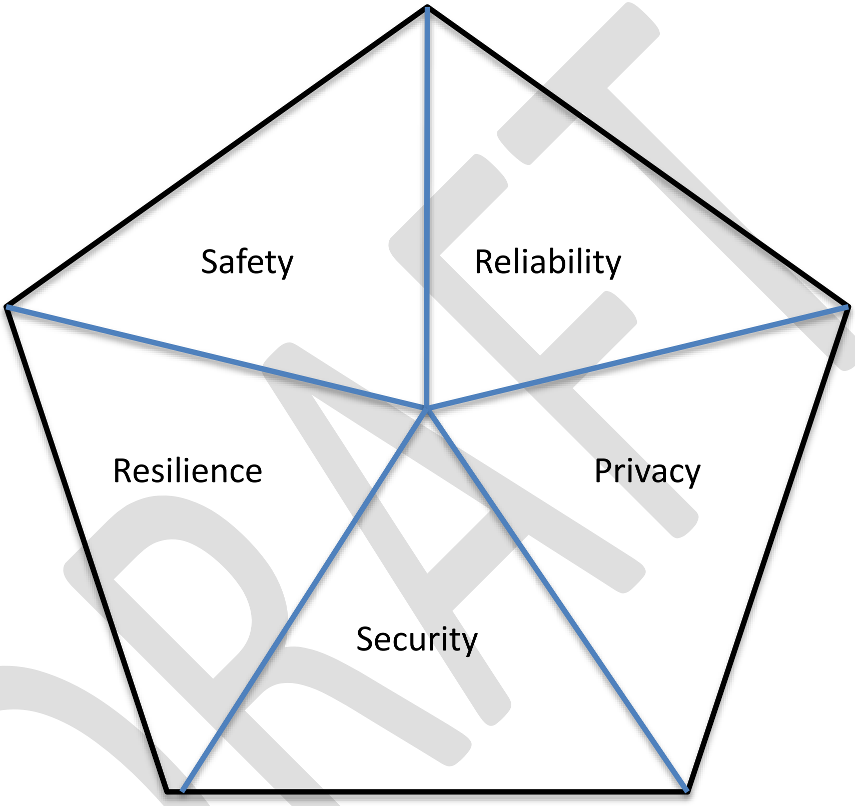


Figure 12 Recommended interdisciplinary design approach to CPS Engineering

Industry trends suggest that discrete systems engineering disciplines are converging toward increased interdependency [55] as illustrated in Figure 12. This is particularly important for CPS in which co-design to support objectives such as safety, reliability, resilience, privacy and security must be considered. The relative importance and interaction of the various risk-related properties must be considered so that problems arising with respect to one property, or protections inserted to address one dimension of concern, does not compromise other primary system objectives or cause deleterious unintended effects. An interdisciplinary approach to systems design and integration is therefore required to establish an overall system-of-systems design objective and contemplate how to make appropriate trade-offs in the service of that objective, if possible.

Because earlier CPS were custom designed over time and mostly isolated, it was believed there were few common processes or software systems by which a cybersecurity incident could affect CPS, let alone spread to multiple systems. Due to the implementation of commonly used software and communication protocols, increasing interconnections between different systems, and connection to the Internet, CPS cybersecurity is becoming increasingly important to CPS owners and operators. Cyber-attacks can now affect CPS operations in a variety of ways, some with potentially significant adverse effects.

The development of trustworthy [56], networked CPS requires a deep understanding of potential impacts resulting from intentional and unintentional cyber-attacks or incidents, on both the cyber and the physical aspects of the system. Such an effort must address cybersecurity jointly with safety, reliability, resilience and privacy.

4.1.3.3 The Need for Cross-Property Risk Analysis for CPS

Cyber-physical systems are composed of physical and cyber components with an abstraction layer that mediates between them.   An objective of CPS systems is to achieve optimum behavior through the correct allocation of requirements to each of the three elements through a process of co-design. “Optimum” in this context involves determination of the desired balance point for cost, benefit and risk.

Systems designers and integrators often assign a ‘risk budget’ to manage the degree of allowable impact security, safety, reliability, privacy and resilience may have on system performance. With the co-design of risk-relevant properties, this budget should not be meted out with a separate share to each concern, but rather viewed as a common resource that each property can draw on. System designers must develop a risk model which indicates the level of protection required for each of the properties and the level of the system in which these protections are best addressed. Since this budget is fixed, designers need to determine the allocation that best achieves the overall objective. Tradeoffs will be required if the budget is not adequate to address all concerns. Obviously, determination of specific priorities will be situation-dependent and the risk budget need not be apportioned equally.

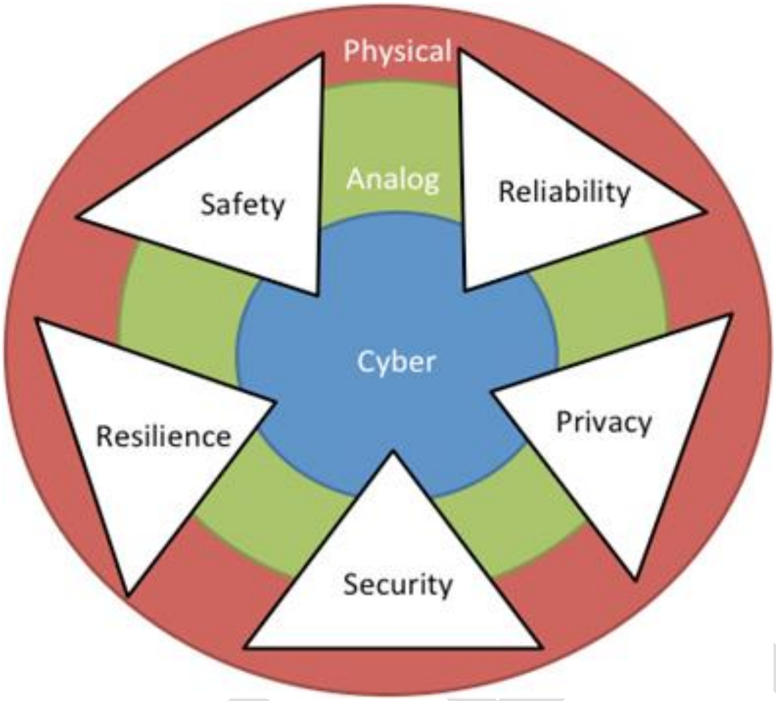


Figure 13 - Cyber-physical systems are composed of physical, analog and cyber components - – notion of co-design in ref. architecture (have to think of functional requirements over the whole, can argue for co-design in security domain as well]

When considering solutions involving cyber, physical or abstraction layer components, engineers must determine how to evaluate the effect of their choices on the system in terms of relevant trade-off metrics. In simplistic terms, security now considers operational and reputational risk, safety considers, error rates, reliability considers failure rates, privacy considers unwanted disclosure rates and resilience considers recovery rates. The complexity, interconnectivity and dynamism typical of cyber solutions may argue for a greater consideration of protections at that level.

4.1.3.4 Cybersecurity as a CPS Risk Management Property

It is interesting to consider how cybersecurity interacts with the other risk-relevant properties to provide trust that the system will work as expected in the face of changing conditions, faults and threats. By adding cyber components to systems, we are introducing new loci of faults and new vectors of threat, as well as a more complex environment. This provides new challenges in providing safety, resilience, reliability and privacy for the system. However, by adding a cyber-component to the system and considering cybersecurity as an integral part of that component, we are also adding a new locus of protections and protection mechanisms ("smarts") that cannot be instantiated in the physical domain alone.

Safety and resilience requirements are perhaps the most challenged by the addition of a cyber-component to the system. Safety is the absence of catastrophic consequences on the user(s) and the environment [57]. The primary focus of any system safety program is to implement a comprehensive process to systematically predict or identify the operational behavior of each safety-critical failure condition, fault condition or human error that could lead to a hazard and potential mishap. This process is used to influence requirements to drive control strategies and safety attributes in the form of safety design features or safety devices to prevent, eliminate and mitigate safety risk. The cyber component greatly increases the complexity of the set of possible behaviors and so greatly complicates this analysis. Modern system safety is comprehensive. It is risk-based, requirements-based, function-based and criteria-based. It includes specific objectives aimed at producing engineering evidence to verify whether safety functionality is deterministic and provides acceptable risk in the actual operating environment.

Cyber components that command, control and monitor the safety-critical functions of physical systems require extensive system/software safety analyses to influence detail design requirements, especially in relatively autonomous or robotic systems that require little or no operator intervention. Cybersecurity must be able to deal with system complexity, and system designers and engineers must consider cybersecurity principles that support separation of functions and assured composition.

The safety of a CPS depends on its resilience, which includes fault-tolerance, ability to degrade gracefully and pre-defined fail-safe states (and triggers). Resilience gives a system “tolerance to degraded and failed conditions that permits continued performance of all or at least critical functions [59].” In the event of significant system failure that could compromise safety, a resilient system must provide a highly reliable way to achieve pre-defined fail-safe status. Alternatively, the system may reconfigure process streams and control parameters to meet new functional objectives, including establishing new operational priorities such as shutting down low-priority processes in order to direct remaining resources to higher-priority ones (graceful degradation). Cybersecurity protections can also support the identification of the more critical aspects of the system and provide additional protections to those system components.

System reliability is a critical requirement of CPS. An unreliable CPS can produce system malfunctions, service disruptions, poor-quality products, financial losses, and even endanger human life and the environment. Each component (and component system) of the CPS must provide a sufficiently low failure rate to enable the CPS to achieve sufficient aggregate system-level reliability. Resilience gained through redundancy and synchronization (fault-tolerant approach) among different CPS components, in combination with high-confidence detection of failures, are the major means used to provide required level of reliability and availability of a system [60]. Cybersecurity practices and mechanisms can be used to provide software assurance and to improve failure detection.

Reliability has some commonalities with cybersecurity (e.g. providing the required level of availability). The major difference is that reliability has traditionally primarily addressed physical/environmental defects/problems or unintentional human (operational) errors. Cybersecurity, on the other hand, aims first to protect against and mitigate the effects of intentional disruptions caused by human-related attacks that may target:

- **System/data availability**—the ability to provide required functions/data (including control functions, specifications and state indicators);

- **System integrity**—the ability to execute the correct instructions using the correct data. It is important to recognize that attacking the cyber subsystem can disrupt proper functioning of the physical subsystem(s) of the CPS or cause the system to function in accordance with an improper set of instructions;
- **Data confidentiality**—the ability to protect system data (including internal programs) from disclosure to unauthorized individuals or use of data for unauthorized purposes.

Traditionally, reliability mechanisms concentrated on detection, protection and mitigation of CPS component failures (fault-tolerance) while cybersecurity concentrated on detection, prevention and mitigation of attacks and compromises (threat-tolerance). Enabling the seamless convergence of reliability and cybersecurity will help provide CPS resilience and the required level of safety.

4.1.3.5 Cyber-Physical Systems Trends and Risk Analysis

Traditional information technology (IT) cybersecurity provides information protection (integrity, confidentiality, privacy) and readiness for correct services (availability). CPS Cybersecurity has the same goals as traditional IT cybersecurity, though perhaps with different priorities, but in addition to that it should be focused on how to protect physical components from the results of cyber-attacks. Two challenges are typical for CPS cybersecurity:

- Detection and prevention of deception attacks (e.g. attacks on sensors that can lead them to input malicious data to the cyber component and, as a result, to provide wrong, or even dangerous, output from the cyber component)
- Detection of compromised cyber component and prevention of incorrect cyber functioning (or stop functioning).

These challenges are not unique to CPS; rather their consequences are potentially more severe because they impact the physical world. More importantly, the means to prevent these problems include not only cybersecurity controls but also safety and reliability controls that are not available to IT systems.

Thus CPS cybersecurity requirements should be determined *in conjunction with* safety, reliability, and privacy requirements. In its turn, CPS resilience should provide ways and means to continue not just IT services, but also critical CPS operations in case of a failure or a cyber-attack, with full CPS recovery. This can be done only through co-design of CPS cybersecurity, including privacy, with safety, reliability, and resilience. As a result, consideration of the traditional tenets of Confidentiality, Integrity and Availability is no longer the sole focus of cybersecurity for CPS. Nor is providing CPS cybersecurity simply a matter of prioritization and application of existing controls. Rather, it involves the tradeoff of risks. This process of risk management becomes even more critical when one considers the potential impact of cybersecurity failures on the ability to deliver capability across the disciplines.

In addition to this, to develop effective CPS cyber protection and mitigation actions, one must understand the nature, functions, and interactions of all three layers of CPS: cyber, abstraction, and physical.

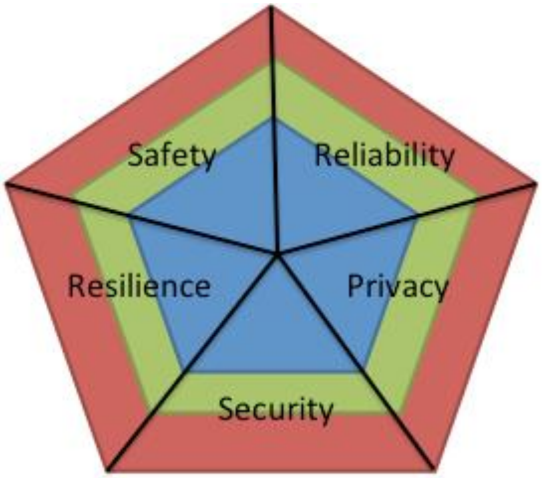


Figure 9. Cyber-Physical Systems Risk Disciplines

CPS designers and integrators should consider both the intended and unintended effects resulting from the combination of properties where the goals of each may contradict or be complimentary to their counterparts. Trade-off decisions should be considered in light of the system-of-systems objective, if known. This is much more challenging than it sounds.

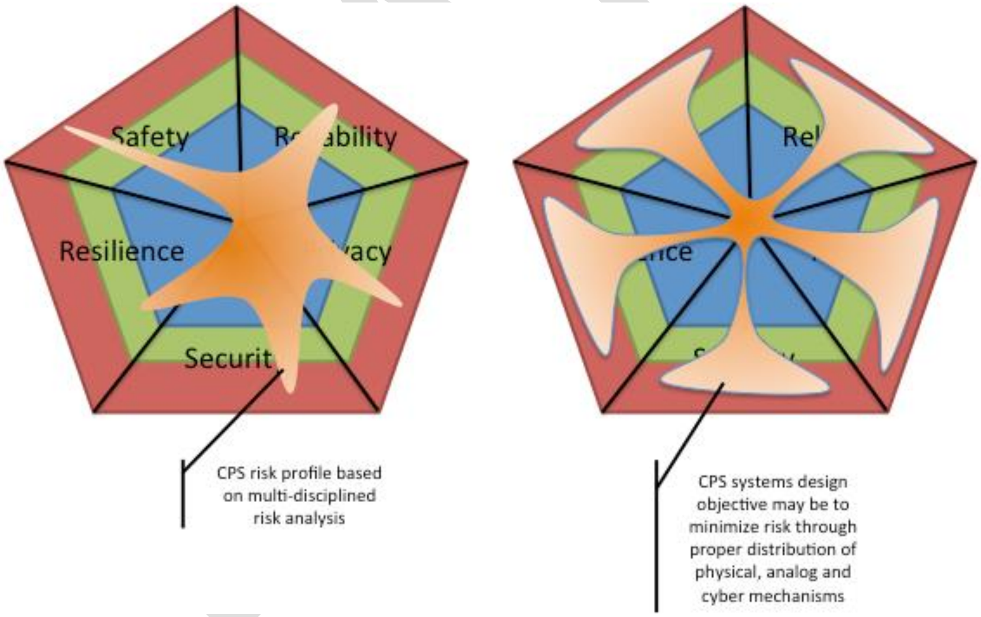

Safety
Re ability
Resilience
acy
Securit
Rel
nce
CPS risk profile based on multi-disciplined risk analysis
CPS systems design objective may be to minimize risk through proper distribution of physical, analog and cyber mechanisms

Figure 10 - CPS Risk Analysis

A system-of-systems design or integration approach for CPS may benefit from ‘risk model’ analysis that considers the impact to each system objective individually and the system of systems objective as a whole. For example, a system of systems whose highest priority goal is to deliver safety should have a risk model that favors safety. Risk models may also aid in placing emphasis on the most appropriate layer – physical, abstraction or cyber. System risk analysis may provide helpful context when considering how best to apply desired CPS risk-related properties. While their specific equities and priorities may be different, CPS owners and operators should use a similar process when evaluating risk in operational situations. This requires a detailed understating of the strengths and weaknesses of the system in place, the role of each layer, and the interactions among the layers.

It is useful to look at a few illustrative examples of risk models to get a clearer understanding of the kind of analysis and tradeoffs that take place in the design of a CPS. Because these are high level examples, this discussion does not address the allocation of concerns across the cyber, physical, and abstraction levels of the system, which varies based on implementation. We can however, describe the relationship among the risk-related properties in a number of example systems.

4.1.3.5.1 Implanted Medical Device

An implanted medical device has high requirements for safety because incorrect operation could cause direct harm to patients and threaten life itself. It also has high reliability requirements because the patient's welfare depends on the continued operation of the device. Privacy requirements are medium, patients have legitimate concern that their health metrics remain private, but for this example we assume there is personally identifying information associated with the device. There is a 3rd piece of information required for this to become personal and that is the unit number as it is related not to the name but to the Medical Record Number. This becomes a risk only if a direct falsification of values is to be implanted. Otherwise, any wireless, implanted device could be compromised. This brings to light that there are high requirements for cybersecurity protections on the command and control paths of implanted devices, and lesser requirements on their reporting paths. In fact, the privacy requirements might more than cover the cybersecurity requirements on the data reporting paths. Given the high reliability requirement, one might think resilience is critical, but the small size and low power typical of implanted devices make the usual methods for providing resilience (e.g. redundancy, fail over) impractical and lead us to think about alternative strategies such as frequent monitoring, scheduled replacement or early detection of degradation.

4.1.3.5.2 Chemical Manufacturing Plant

A chemical manufacturing plant has high requirements for safety that refer to two aspects. One is process safety itself, to prevent unwanted or uncontrolled chemical reactions. The other is equipment safety, which seeks to prevent equipment failure or breakage. An example would be preventing pressure in the reactor exceeding safety limits to stave off reactor burst [64]. Today, more than 100 million Americans live close enough to one of the more than 470 chemical facilities across the country that could put 100,000 people at risk if there were a deliberate or accidental release of chemicals at those sites [65]. Safety of Smart chemical plants relies on reliability and security. High reliability, by minimizing defects and implementing one or more alternative control structures in parallel, can compensate for possible failures. But in case of cyber-attacks, such as integrity attacks (sensor manipulation attacks), denial of service (DoS) attacks, and attacks on situational awareness (attack on a Human Machine Interface console), only cybersecurity provides the necessary detection and protection. Given the high reliability and cybersecurity requirements, the resilience of the control process to failures and intentional attacks is critical. Resilience provided by improving the tolerance period during an attack, can give operators more time to intervene. Privacy requirements are low, since there is no personally identifying information associated with the chemical process or plant's equipment.

4.1.3.5.3 Wearable computing and Internet of Things (IoT)

Wearable computing is the use of a miniature, body-borne computer or sensory device worn on, over, under or integrated within, clothing. Constant interaction between the user and the computer, where the computer "learns" what the user is experiencing at the time he or she is experiencing it and super-imposes on that experience additional information, is an objective of current wearable computing design [66]. According to a 2013 market research report [67], there are currently four main segments in the wearable technology marketplace:

- Fitness, wellness and life tracking applications (e.g. smart clothing and smart sports glasses, activity monitors, sleep sensors) which are gaining popular appeal for those inclined to track many aspects of their lives;
- Infotainment (smart watches, augmented reality headsets, smart glasses);
- Healthcare and medical (e.g. continuous glucose monitors, wearable biosensor patches) and
- Industrial, police and military (e.g. hand worn terminals, body-mounted cameras, augmented reality headsets).

Security and privacy issues should be considered very seriously as the wearable devices work through IoT that deals not only with huge amount of sensitive data (personal data, business data, etc.) but also has the power of influencing the physical environment with its control abilities. Cyber-physical environments must, therefore, be protected from different kind of malicious attacks. Security, privacy, resilience and safety requirements depend on the particular application. For example, fitness tracking applications have low requirements for risk-related CPS properties. Police or military applications should have high safety, security, and resilience requirements based on their mission.

4.1.4 Applying Cybersecurity Controls to CPS

In development; it is likely that this content will be published as a separate document.

4.1.5 Parking Lot: CPS Cybersecurity and Privacy

4.1.5.1 Uncategorized: Difference between ICS and CPS

4.1.5.2 Related Efforts

[Note: This section was previously part of the introduction to the Cybersecurity and Privacy Subgroup Framework Element. However, in the context of the full CPS Framework, this section should be moved to another location since it applies to all aspects of the CPS PWG work and provides context into other ongoing CPS efforts.]

Other ongoing efforts support the enhancement of CPS, Industrial Internet, and "Internet of Things." These efforts support, impact, and influence the efforts of the CPS PWG. Examples include:

- **Cybersecurity Research Alliance (CSRA) [46]** is an industry-led, non-profit consortium focused on research and development strategy to address evolving cyber security environment through partnerships between government, industry, and academia. This effort was established in response to the growing need for increased public-private collaboration to address R&D issues in cyber security. The founding members of the CSRA are Advanced Micro Devices, Inc. (AMD), Honeywell International, Inc., Intel Corporation, Lockheed Martin Corporation, and RSA (Security Division of EMC).
- **CPS Voluntary Organization (National Science Foundation) [47]** is an online site to foster collaboration among CPS professionals in academia, government and industry.
- **The Networking and Information Technology Research and Development (NITRD) CPS Senior Steering Committee [48]** coordinates programs, budgets, and policy recommendations for CPS research and development (R&D). This includes identifying and integrating requirements, conducting joint program planning, and developing joint strategies for the CPS R&D programs conducted by agency members of the NITRD Subcommittee. CPS includes fundamental research, applied R&D, technology development and engineering, demonstrations, testing and evaluation, technology transfer, and education and training; and "agencies" refers to Federal departments, agencies, directorates, foundations, institutes, and other organizational entities.
- **NIST Privacy Engineering [49]** focuses on providing guidance that can be used to decrease privacy risks, and enable organizations to make purposeful decisions about resource allocation and effective implementation of controls in information systems.
- **Industrial Internet Consortium [50]** brings together the organizations and technologies necessary to accelerate growth of the Industrial Internet by identifying, assembling and promoting best practices. This goal of the IIC is to drive innovation through the creation of new industry use cases and testbeds for real-world applications; define and develop the reference architecture and frameworks necessary for interoperability; influence the global development standards process for internet and industrial systems; facilitate open forums to share and exchange real-world ideas, practices, lessons, and insights; and build confidence around new and innovative approaches to security.
- **National Security Telecommunications Advisory Committee (NSTAC) [51]** brings together up to 30 industry chief executives from major telecommunications companies, network service providers, information technology, finance, and aerospace companies. These industry leaders provide the President with collaborative advice and expertise, as well as robust reviews and recommendations. The NSTAC's goal is to develop recommendations to the President to assure vital telecommunications links through any event or crisis, and to help the U.S. Government maintain a reliable, secure, and resilient national communications posture.

4.1.5.3 Development Methodology

Note to Reviewers: *This section focuses on the Cybersecurity and Privacy methodology. Based on interest in reorganizing this focus from a "Cybersecurity and Privacy Aspect/Viewpoint" to address other Risk Management areas (e.g., resilience, reliability, safety, etc.), this methodology will need to be updated. This section may also be more appropriate as background for a cybersecurity-specific CPS document.*

Task 1. Identification and Analysis of Illustrative Examples: In order to better understand the characteristics of CPS across the breadth of domains, the first task is to identify illustrative examples that highlight the unique cybersecurity needs and challenges associated with existing CPS. The goal is to identify common and "architecturally significant" cybersecurity requirements with emphasis placed on the unique operational frameworks within which CPS operate. Through this analysis, the subgroup will identify unique CPS cybersecurity and privacy challenges and opportunities. The illustrative examples, supplemented by the repository of use cases developed by the CPS PWG Use Case Subgroup, will provide a general framework for performing risk assessments across different CPS.

Task 2. Identification and Analysis of Unique Challenges and Properties of CPS Cybersecurity and Privacy: While many of the cybersecurity and privacy challenges that apply to IT systems also apply to CPS, it is important to identify and understand what is unique to CPS. These unique challenges and opportunities stem from operational needs, the potential impacts of interconnections, physicality of the systems, and the potential impact on the environment. Identifying the attributes of CPS that are unique from IT enables tailoring of existing cybersecurity approaches and solutions or identifying new ones that are well suited to this domain.

Task 3. Cross Property Approach to Risk Management: The findings from Task 2 highlighted a unique opportunity for cross-property risk management in CPS. This approach builds upon and leverages the relationship of cybersecurity and privacy with the fields of safety, reliability, and resilience.

Future Task. Risk Assessment: The risk assessment of CPS will be based on analysis of a set of illustrative examples and use cases. Key risk assessment tasks include identifying the assets, vulnerabilities, threats, and potential impacts. The risk assessment process includes identification of vulnerabilities, consideration of well-understood problems that need to be addressed (such as user and device authorization and authentication), a top-down analysis of priority areas (to be identified based on the Reference Architecture subgroup output and through dialogue and analysis of unique CPS cybersecurity properties and challenges) to determine groupings of CPS with similar cybersecurity characteristics and constraints. The risk assessment process leads to identification of the cybersecurity objectives (confidentiality, integrity, and availability) for CPS (or subgroupings of CPS). Feedback on the cybersecurity objectives, characteristics, and constraints will be provided to the Reference Architecture subgroup with recommendations for how to include cybersecurity into the overall CPS Reference Architecture(s). Additionally, as the CPS PWG developed a vocabulary, the Cybersecurity and Privacy Subgroup supplemented it with relevant terminology from this work.

Future Task. Specification of Cybersecurity Requirements: A compendium of relevant, existing cybersecurity requirement/control source documents from different domains of CPS has been collected [See Appendix A]. Leveraging the output of Tasks 1 and 2, a subset of from the compendium documents will be selected to serve as input/source documents for the specification of specific cybersecurity requirements or a methodology to tailor existing security requirements for CPS. The output of this task will be determined as Tasks 1 and 2 continue to progress. This task will be an ongoing effort that may result in a separate document.

4.1.5.4 Recommendations

CPS that address a more complete set of tenets will be more complete and hence will present less risk to the greater system-of-systems envisioned by IoT. Safe, reliable or resilient systems that lack attention to security or privacy may increase these risks when connected to other systems whose primary objective is security or privacy. CPS cybersecurity is concerned with managing risk for the entire system-of-systems as well as for sub-systems. Development of a common approach to cyber security design, integration and operation is an important next step. In particular, CPS designers need to consider the following when addressing cybersecurity controls.

1. Proactive mechanisms in sensor network security have focused on integrity and availability from a communication network point of view. They have not considered how deception and DoS attacks affect the application layer service, i.e. how successful attacks affect estimation and control algorithms –and ultimately, how they affect the physical world. Novel robust control and estimation algorithms should be designed that consider realistic attack models from a security point-of-view. These attack models should simulate deception and DoS attacks.
2. Cybersecurity controls have not considered algorithms for detecting deception attacks against estimation and control algorithms. In particular, previous detection of deception attacks launched by compromised sensor nodes assumes a large number of redundant sensors: they have not considered the dynamics of the physical system and how this model can be used to detect a compromised node. Furthermore, there has not been any detection algorithm to identify deception attacks launched by compromised controllers.
3. Many cybersecurity controls involve a human in the loop. Because CPS use autonomous, real-time decision making algorithms for controlling the physical world, they introduce new challenges for the design and analysis of secure systems: a response by a human may impose time delays that may compromise the safety of the system. Therefore, autonomous and real-time detection and response algorithms should be designed for safety-critical applications.
4. CPS security should be defined with respect to an adversary model. Previous research has not studied rational adversary models against CPS. The field of automatic control is more mature in comparison to information security; however, despite great achievements in the field of nonlinear and hybrid systems theory, robust, adaptive, game-theoretic and fault-tolerant control, much more needs to be done for design of secure control algorithms to ensure survivability of CPS.

5. In addition to the state of the system to be controlled, the state of communication network should be jointly estimated. Approaches to estimate the indicators of performance and integrity of the communication network based on available network data should be developed. The estimated state of the network should be used to design transmission policies for sensors and actuators as well as scheduling policies for controllers to optimize performance.
6. Physical and analytical redundancies should be combined with security principles (e.g., diversity and separation of duty) to adapts or reschedules its operation during attacks. For example, under sensor faults or when only intermittent sensory information is available, the system should be able to operate using open-loop control for a sufficient amount of time.

A notion of trustworthiness should be associated with different components of CPS and trust management schemes should be designed when the above redundancies are in place.

4.1.6 Safety Risk [TBD Subgroup]

4.1.7 Reliability Risk [TBD Subgroup]

4.1.8 Resiliency Risk [TBD Subgroup]

4.1.9 Timing Risk [Timing Subgroup]

With the fundamental need for accurate timing in ensuring coordination, synchronization, operational accuracy and integrity of CPS nodes, it is necessary for the designers of a CPS to understand the risks associated with acquiring and distributing accurate time. Timing is subject to both physical and cybersecurity risks, both accidental and deliberate. Acquiring a reference time traceable to a national standard for global synchronization of centralized or decentralized CPS involves a physical signal, whether it's a GNSS signal, other RF signal, or if it’s transmitted through a network. The RF physical signals are subject to interference from space or earth weather effects, or from jamming and/or spoofing. Distribution of timing signals through networks is similarly subject to cybersecurity and physical risks. These risks as well as the elements of securing time in CPS are discussed in detail in Section 4.3.4 and in the Annex on Timing, Section 1.3.2 [144].

4.2 Data Aspect [DI Subgroup]

4.2.1 Overview

Data may be created, maintained, exchanged and stored in many domains. Each datum has a lifecycle and can be moved among any number of systems and components. Each domain naturally defines its own data semantics and exchange protocols. But both humans and systems can find it difficult to process, understand and manage data that has been moved across domains and ownership boundaries. In an ever more connected world, processing and understanding data is a growing necessity. A cyber-physical system (CPS) is a system of collaborating computing elements that monitor and control physical entities. Understanding data exchanged among independent computing elements is as much, if not more important than it is in other data management domains.

CPS components collect, process, share and examine data to provide actionable inputs to other CPS components. Data are acquired, shared and examined at multiple “levels” within “scales.” A “scale” is a spatial, temporal, quantitative, or analytical dimension used to measure and examine the data. A “level” is a unit of analysis on a scale. For example temporal scale can be thought of as divided into different “levels” (time frames) related to rates, durations, or frequencies.

The dynamics of cross-scale and cross-level interactions are affected by the interactions among collaborating computing elements and entities at multiple levels and scales. Addressing these complexity issues in an efficient and effective manner will require new approaches to managing data integration and all boundaries (ownership, scales, and levels) need to be more widely understood and used.

The challenges of data integration complexity and CPS boundaries include:

- Data fusion that is done at any time from multiple sensor or source types or use of a single data stream for diverse purposes
- Data fusion of streaming data and predictive analytics capabilities
- Complex data paths that cross-scale and cross-level connecting architectural layers, dedicated systems, connected infrastructure, systems of systems, and networks
- Data-driven interactions between dependent and independent cyber physical systems
- Privacy-protecting data policies and procedures in light of the ubiquitous nature of IoT
- Data interoperability issues including metadata, identification of type and instance, data quality and provenance, timing, governance, and privacy and cybersecurity

The goal of this data interoperability aspect is to provide a sound underlying description and standards base that simplifies and streamlines the task of understanding of cross-domain data interactions.

4.2.1.1 Organization of the Data Interoperability Section

The Data Interoperability section begins with the overview above. It then follows with a presentation of key topics about data interoperability from the CPS viewpoint. Each of these sections in turn has an overview to discuss the topic and an example of what the topic is about to give it some context.

Then, a section summarizing the critical dimensions of Data Interoperability provides for detailed discussion of data and metadata, identification, data quality and provenance, governance, privacy and cybersecurity, and verifiability and assurance.

Then, since this is being developed in a consensus process, a “parking lot” captures issues that have not yet been resolved.

The CPS-PWG bibliography has a section for data interoperability references presented.

In this framework, we cite a significant number of references. However, the scope of data interoperability is broad and a more exhaustive study could include many more substantive references. Further, there are mentions of specific references that are helpful in illustrating the concepts presented. However, these descriptions are intended to be exemplary rather than prescriptive.

4.2.1.2 Data Interoperability Discussion

The concept "data interoperability" involves how and to what the extent systems and devices can exchange and interpret data. It assumes a requirement to understand the exchanged data to realize the intended benefits of the exchange. We define the "dimensions of interoperability" as the extent to which exchanged data can be understood. Note that data interoperability is but a subset of all dimensions of interoperability necessary to establish an interoperable architecture of exchange. However, this section focuses only on the data dimensions – syntactical, semantic, and contextual.

- Syntactical interoperability defines the structure or format of data exchange, where there is uniform movement of data from one system to another such that the purpose and meaning of the data is preserved and unaltered. Syntactical interoperability defines the syntax of the data – organization of the bits and bytes – and certain structural descriptions of intermediate processing such as processing for storage, manifesting descriptions, and pipelining. It ensures that data exchanges between systems can be interpreted at the individual data field level.
- Semantic interoperability provides for the ability of two or more information systems or elements to exchange information and to enable the use of the information that has been exchanged, processed, interpreted or otherwise used, independent of the syntax by which it was exchanged. Semantic interoperability is about a shared, common interpretation of data. This degree of interoperability supports the exchange and other operations on data among authorized parties via potentially dependent and independent systems, if required.
- Contextual interoperability includes Business Rules about the validation and authorization of data.

As with any interaction between systems, the data exchanged will be driven by how the data is used. The content, format and frequency of systems-to-system data exchanges is driven by the intended purpose of the exchange—specifically, where, when, how and why the receiving system will use the exchanged data. In addition to physical connectivity that permits data movement, use of data across disparate systems often requires translation of data objects from the syntax of the sender's data into a form that is compatible with the receiver's syntax. For systems that require integration, the exchange of data between systems is done through data models and data objects that describe the data semantics. The receiving system must understand the context, for example metadata that describe the nature and constraints on the data, in which the data was created to properly apply the semantics to its purpose.

In practice, data exchange requires the interoperability framework to encompass the physical connection of sensors and system components accounting for transmission of data through various protocol standards. These data are then processed through system software data ingest functions according to specified rules and procedures.

4.2.1.3 Canonical Models and Adaptors

Many cyber physical systems are composed, at least in part, of legacy components and data implementations. These legacy components may not implement current best practices and protocols. A descriptive semantic model relies on the data types and the relationships between the data types within a given data model. Redesigning applications to use a given semantic model may not be straightforward or even feasible. This means that the source system's data model must be transformed into each destination system's data model for integration.

A set of common canonical data models that can be mapped to a set of disparate semantic models can reduce complexity in these cases. The models can be maintained for critical systems within each infrastructure and, at the highest level, between infrastructures. The use of common canonical models reduces the number of transformations between systems required from "n(n-1)" to "n" (where n is the number of disparate system that must ultimately exchange data), because in the more complex case, each pairwise exchange domain must have its own bilateral transformation.

Data related to time, privacy and security are also important within the context of data exchanges between applications. The integration of time-series data should express time information in a manner that can be aligned to a global time, including drift. This is similar to how GPS can be used for geo-level data integration to enable consistent understanding across system boundaries. Privacy, security, and authentication data are also essential to the contextual understanding of information because they embody essential trustworthiness requirements.

Adaptors can minimize the impact on cost and complexity of interoperability achieved. In traversing many network segments and protocols, a standard interface can be inserted at any point rendering everything upstream from it interoperable in view.

Higher degrees of interoperability achieved has implications for reducing the complexity of the data exchange and use. Data exchange adapters between systems should be strategically located for maximum effect and minimum cost. This will reduce the risk to these systems as they evolve and expand.

4.2.1.4 CPS Data Interoperability and the Term "System of Systems"

A CPS is a cyber-physical *system*, and every *system* must have clearly identified boundaries. When data crosses a system's boundary, it may flow to another system. The movement of data may be to an "actor" (e.g., person, component, device or system) which (by definition) is closely involved with the operation of the CPS, or it may be to an actor having no direct connection to the original one. From the perspective of the first CPS, some systems may appear to passively consume data. When other systems exist outside the CPS boundary, it is possible that a collection of such systems could interact, with new behaviors emerging from this interaction. In this way, the original CPS may become part of a "system-of-systems." Whether or not the CPS interacts at this scale may be of little or no import to the individual CPS. Ideally, well-crafted interfaces from the CPS to other systems will permit the circulation of data among systems, while limiting data use to authorized users and purposes. From the data interoperability perspective, the challenge lies in the design of the CPS' data interface. The focus of this document is to raise data interoperability issues and recommend how they may be addressed in practice. These issues include:

- The identity of the sender
- The identity of the data
- The integrity of the data
- The semantic meaning (including context) of the data
- The authorization to acquire and use the data (for specified purposes)

Whether a particular Cyber Physical System is able to interact with other systems to become part of a System-of-Systems is perhaps a test of the quality of the handling of these issues. When a CPS is designed, it may be expected to occupy a particular position in a large and well-defined ecosystem. Or, it may part of a small collection of systems, or even standing alone. Ideally, such matters would be immaterial to the interface. However, interfaces that support exchanges to among multiple stakeholder systems are difficult to realize. In information systems, the very nature of "identity" and "meaning" are usually arrived at by mutual agreement. There is no global authority to certify all identities and all semantic meaning for all applications. It is thus left to the technical community to arrive at useful solutions to some of these issues. These arrangements must be balanced by other practical concerns such as:

- The costs associated with communication (and thus the degree of implicit versus explicit semantic content)
- Safety concerns, and the risks associated with data errors to the application or other actors
- The extent and reliability of security required by the application
- The provision of version control and the support of newer/older versions of an interface

4.2.2 Data Interoperability Topics from the CPS Viewpoint

4.2.2.1 Data Fusion

4.2.2.1.1 Data fusion from multiple sensor or source types or use of such data for diverse purposes

4.2.2.1.1.1 Overview

Researchers and practitioners have offered several strong definitions of the term data fusion. The US Department of Defense's Joint Director of Laboratories Workshop (JDL Workshop 1991 [85]) defined it as a "… multi-level process dealing with the association, correlation, combination of data and information from single and multiple sources to achieve refined position, identify estimates and complete and timely assessments of situations, threats and their significance." Hall and Llinas [83] synthesized prior research to define data fusion as "… techniques [that] combine data from multiple [sources], and related information from associated databases, to achieve improved accuracies and more specific inferences than could be achieved by the use of a single [source] alone." Taking a narrower view for their Linked Data effort, Bizer, Heath and Berners-Lee [84] define data fusion as "… the process of integrating multiple data items representing the same real-world object into a single, consistent, and clean representation." Castanedo [86] groups data fusion techniques into "three nonexclusive categories: (i) data association, (ii) state estimation, and (iii) decision fusion."

Error! Reference source not found. illustrates the JDL fusion framework, which comprises " … four] processing levels, an associated database, and an information bus …." [86]. Elaboration of the details of this design-oriented framework is beyond the scope of this document.

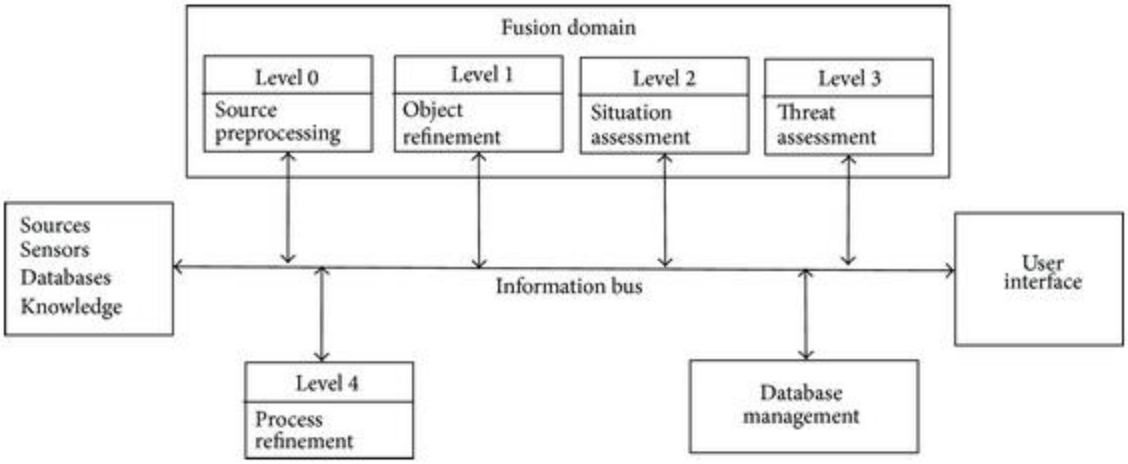


Figure 11: JDL Fusion Framework

Cyber physical systems (CPS) are increasingly leveraging capabilities provided by improved sensors, processing techniques and computing power to monitor, analyze (sometimes in near-real time) and control increasingly sophisticated systems and processes in domains as diverse as manufacturing, robotics, the operation of medical devices (both free-standing and implanted), environmental control, energy generation and distribution, and transportation. As the desire for additional data fusion grows, CPS users are likely to rely on data fusion in the sense of all of the definitions provided above.

Efforts to fuse data from multiple sources face significant data interoperability challenges. These challenges include, but are not limited to: identifying and resolving differences in vocabulary, context and semantic meaning; structure (schema); attributing data to their source and maintaining an accurate "trail of provenance" (with attendant issues in identity management); resolving differences among different data formats; and detecting and resolving issues of accuracy vs. timeliness.

An international standard, Recommendation ITU-T X.1255 [20], was approved in September 2013. The recommendation adopts a fundamental approach toward defining core concepts for purposes of interoperability across heterogeneous information systems. It describes a digital entity data model that provides a uniform means to represent metadata records as digital entities, and can also be used to represent other types of information as digital entities (whether also referred to as data, data item, data fusion, or other terminology). It is a logical model that allows for multiple forms of encoding and storage, and enables a single point of reference (i.e., the identifier) for many types of information that may be available in the Internet.

4.2.2.1.1.2 Example

A typical Air Traffic Control System is a cyber-physical system that leverages data fusion. Each air traffic controller is the man-in-the-loop in a control system that directs aircraft to certain flight paths and altitudes at specific speeds. Controllers also advise pilots of potentially hazardous traffic and weather. The air traffic control system combines data from two types of sensors to provide an annotated image used by air traffic controllers to monitor and control the flight of thousands of aircraft a day. The first type of sensor is fixed-site surveillance radar. The surveillance radar provides bearing and slant range from a known point (the radar antenna's location) and can detect some forms of hazardous weather. The displayed aircraft geographic position (the "blip" or "primary return" on a radar screen) is a function of slant range and the known geographic location of the radar antenna. The second sensor is one of a pair of redundant "Identification Friend or Foe" (IFF) transponders on each aircraft. The transponder collects altitude data from the aircraft's flight instruments and combines this data with the aircraft's identification code, then transmits this data to a receiver mounted on top of the surveillance radar. The system that displays the images on the controller's radar screen must merge and continuously update the primary and secondary data to present an accurate and integrated picture over time to enable controllers to help ensure proper routing and safe separation of aircraft from each other and possible hazards.

4.2.2.1.2 Data fusion of streaming data and predictive analytics capabilities

4.2.2.1.2.1 Overview

There is a need for a common interpretation of data to support the exchange of information. Data from today's CPS in various domains are collected separately; each domain exhibits its own data structure and may use different protocols. Data fusion techniques are needed if a user wishes to combine data from various systems.

Among the protocols that seek to help federate data, so that data from multiple sources can be acquired and fused, is OPC UA [128]. Supervisory Control and Data Acquisition (SCADA) systems are examples that, when using OPC UA, combine the data into a common structured dataset accessible via web services. Software like Hadoop [129] enables distributed processing of large data sets across clusters of computers. However, obtaining and harmonizing the data can be a challenge due to the differences in format and variance in protocols. Identification is also an issue since often in today's systems, as many systems may offer no identity other than a tag name, which may not provide the required level of assurance. While some modern systems tag data with IP or MAC addresses, these are insufficient for a positive determination of device type, device owner, device operator and device trustworthiness. Realistic projections indicate solutions to similar requirements must scale to trillions of devices.

CPS today are beginning to transition to a “semantic” form whereby metadata information can be used to describe the device and related information. This metadata can include guidance on how to handle the information. Also gaining popularity is use of identifiers that can be captured in the form of a Quick Response (QR) Code [130].

CPS have begun to use IPv6 and 6LOPAN [131] to be able to capture sensor data and represent unique identifiers for the source of data. Widespread use of this identifier within CPS is a few years out, and faces considerable challenges using the IPv6 address as the primary identification. A client of the data must be configured to use the sensor device address to represent its identity. This has proven useful on a small scale (e.g., in smart phones and some sensor systems deployed in homes and buildings). However, obtaining the information across a backhaul where there have been many local network segments using different protocols from Wi-Fi to Broadband over Power lines (BPL) remains an outstanding challenge.

Additionally, varied approaches to information exchange protocols exist (e. g., the Simple Object Access Protocol (SOAP) [132] and Representational State Transfer (REST) [133]). One is service oriented – SOAP, and, the other data oriented – REST. Thus, a challenge still exists to move the information in a common format that would facilitate data fusion easily.

For the immediate future, data collection and fusion for data analytics are also complicated by security concerns – particularly the confidentiality of the information. Today, data mining is often achieved through access to databases and/or data sets that have been exposed to the public via web pages. Cyber Physical Systems used in healthcare, by utilities, and other critical systems are often maintained on closed networks with understandable reluctance to share the information with third parties.

Presently, a migration to WEB Application Programming Interfaces (APIs) based on SOAP and REST provide a flexible means of serving up data in loosely coupled systems allowing “mashups” of data from multiple sources into analytic services which fuse the data for predictive and other purposes.

4.2.2.1.2.2 Example

The diagram below, which shows the merger of different data sources (often from distinct databases), is the model that is generally used today for obtaining information from data and integrating them into a common data source. This approach, though commonly used, may be inadequate to handle the scale of the Internet of Things.

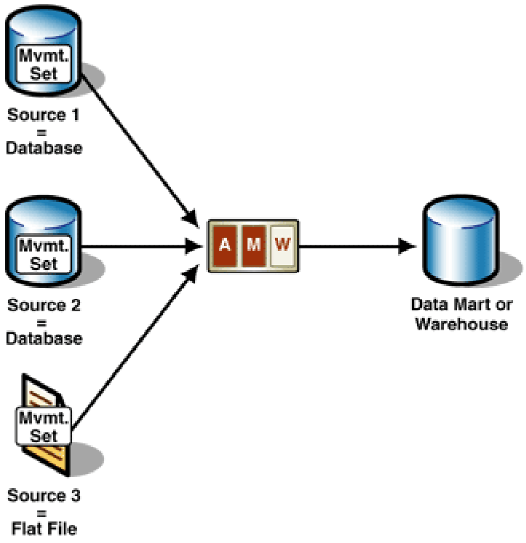


Figure 12: Merger of Different Sources of Data

The diagram below is an example of data fusion today.

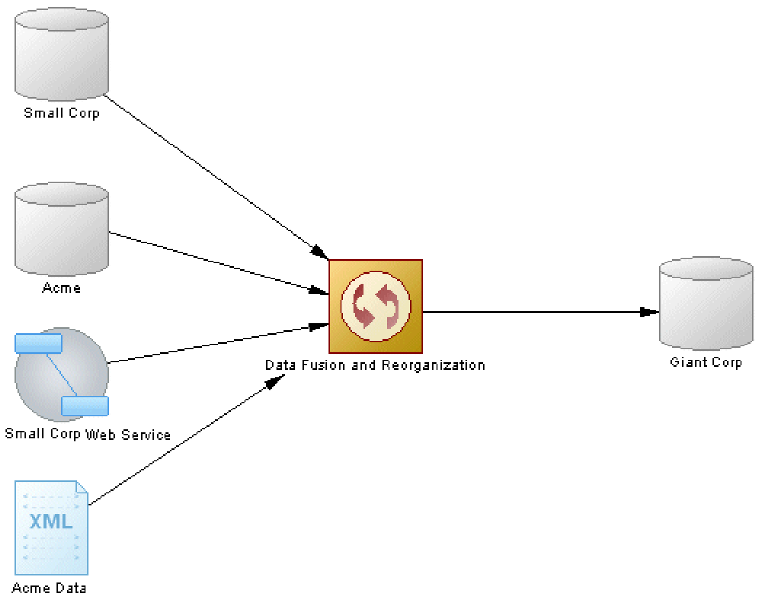


Figure 13: Data Fusion Today

Today's profusion of data sources and uses imposes an additional requirement in that the data flows may need to be shared with multiple locations simultaneously. This drives a requirement for multicast capabilities with extended trust that preserves the data's integrity and rights.

In the case of a sensor device, the end point in the second diagram could be a sensor or group of sensors collecting information, but there would still be a need for data concentration and forwarding to an end point collection system. System owners must decide whether to disseminate the information directly from the end point via a local or regional server/concentrator or use a federated cloud repository that contains the information. Distributing the information is more practical as long a trust engagement is used to assure integrity of the devices with a data sharing capability.

4.2.2.1.2.3 Discussion of relevant standards

There are systems such as the Interface for Metadata Access Points (IF-Map) that have a data binding using SOAP [133][134]. Another standards set that secures the use of SOAP was developed by the OASIS Foundation [136]. Though widely used, it suffers from cyber vulnerabilities stemming from lack of security within the core protocol as well as reliance on web servers to provide the information. The use of OPC UA with Microsoft SQL Server has vulnerabilities to cyber-attack. All of these systems make use of layered on security model that has proven to be highly vulnerable to cyber-attack.

There is a joint effort known as ISO/IEC/IEEE P21451-1-4 (also known as Sensei-IoT*) [97] which has defined a common transport language with built-in security. It offers the data in a common form utilizing XML constructs known as IoT XEPs (Extensions) to the eXtensible Messaging and Presence Protocol (XMPP). This approach has security built into the protocol using TLS (Transport Layer Security) and makes use of trust engagement whereby all devices must be registered to participate in a network. Assuming the root of trust is reliable, this trust relationship allows the data to be trusted and shared with other domains under the control of the owner of a participating device. The resulting structure can be converted to any format since data are held in a common format of XML. The common XML form makes merging information with systems that use XML semantics easier. Moreover, an additional benefit is that during the transition of the original protocol it provides metadata isolation and the ability to apply policy to the data preventing access to certain information to be controlled on a more granular basis. It is among the first Semantic Web 3.0 standard to address the complexities of the Internet of Things (IoT) [109].

The XMPP protocol is used extensively in social networks such as Skype™, Yahoo™, MSN™ and data sharing systems such as GotoMeeting™ and WebEx™. However, while they use XMPP to set up the security session, they often use other protocols to secure the exchange of proprietary information.

4.2.2.1.2.4 Summary analysis

Trustworthy data fusion will continue to be a challenge until systems can assure the integrity and confidentiality of the data, non-repudiable identification of relevant actors and devices, and creation of justified trust among users, devices and applications. CPS present a challenge if the Internet is to be used as a vehicle to transport the information. Each of the technologies presented in this section have deficiencies noted in one aspect or another. New approaches are needed to provide the assurance that data fusion results in integrity and that the information from those systems is interoperable across different domains of use.

4.2.2.2 Complex data exchange and other management issues for interoperability across heterogeneous systems

4.2.2.2.1 Overview

When the Internet protocol (IP) was being developed in the mid-70’s and early 80’s, most computers were large, stationary, expensive to own, and generally had limited interaction with other computing environments. The foundations of both computing and internetworking, including use of the domain name system to facilitate recalling IP addresses, have therefore been rooted in a location-centric mindset; data and other information in digital form is counted on to be accessible at a location and, for the most part, is immobile. Thus, the broadly accepted view that such information cannot be addressed directly through a persistent and unique address but must instead be referenced via a computer address followed by a data pathway within that computational environment.

This method of naming, storing, and moving digital information has become increasingly problematic in the face of trends such as mobile computing, data-producing smart ‘things’, increasing size and volume of data files, and decreasing costs for both bandwidth and storage. More data is being stored, in more formats, for more widely varied uses than ever before. Information and analytics have become commonly traded commodities and are often moved across trust and privacy boundaries, touching multiple administrative domains. Data pathways are becoming increasingly complex and increasingly vulnerable to loss of availability, integrity, or confidentiality.

Additionally, the role of a client of data may determine the nature of access. For example, in manufacturing precision and control are critical, access to read and write data are highly constrained. The relationship between controller and actuator nodes is often termed “tightly-coupled”. On the other hand, such a control system may have access to measurement data that might be of value to other CPS clients outside the control system or even outside the CPS domain. It may be of benefit to provide access to at least read such data. This client relationship can be termed “loosely-coupled”. From this example, the tight or looseness of the coupling between communicating parties may be based on their respective roles in CPS.

An example of this complexity occurs in the manufacturing domain. Many of today’s medium-to-large manufacturing enterprises have multiple lines of business, each with multiple plants each of which contains multiple communication networks that are logically layered. Giving decision makers access to information produced by these plants in a timely manner and in a form normalized for useful understanding is quite a challenge.

The communications networks within a plant often have a hierarchical topology where lower layers become increasingly specialized to meet requirements of the manufacturing functions and systems they support and the conditions in which they operate. The communications equipment in these lower layers is considered manufacturing equipment which has a long lifecycle and is expensive to take offline; thus it is rarely replaced or upgraded. The figure below (from ChemicalProcessing.com) shows a simplified view of a topology of networks for a process plant in the continuous process industry. Much of the production equipment and sensors that produce manufacturing data reside at the bottom of this hierarchy. This equipment is infrequently replaced, leading to a set of equipment that is diverse in type, era, and technology.

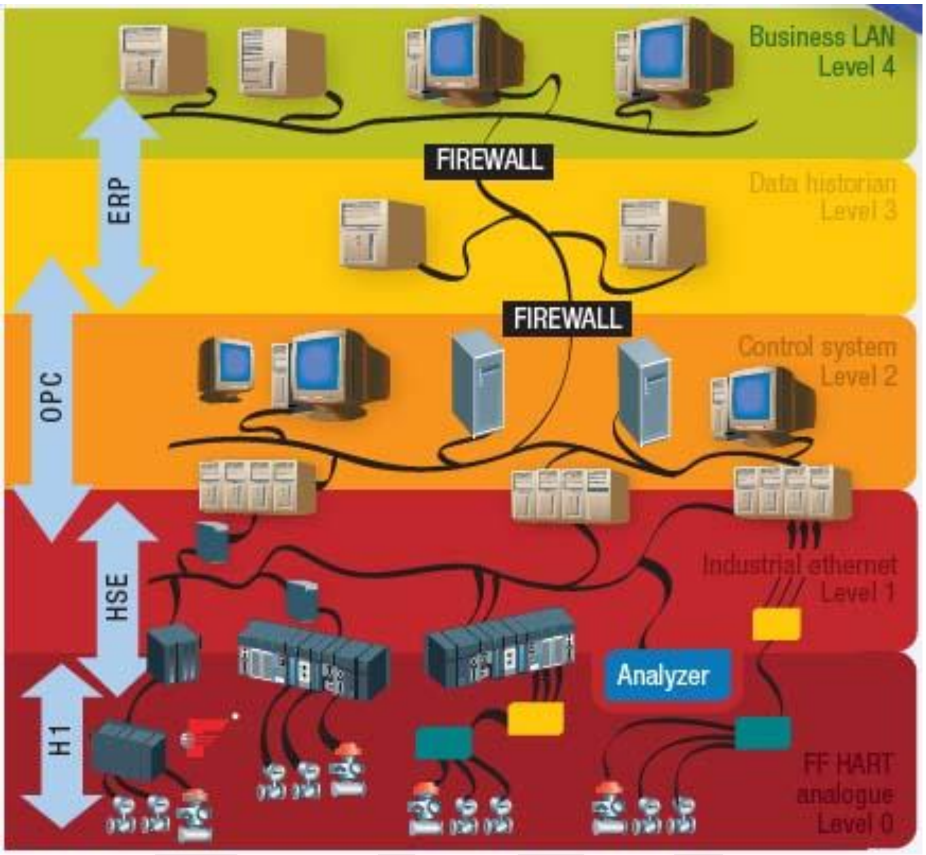


Figure 14: Simplified Topology of Networks for a Chemical Plant

Typically, data from production equipment must flow through its supporting specialized networks upward to reach the enterprise network where business applications support corporate decision making. Such data is typically refined and digested to produce a smaller aggregate result. The raw data itself, however, is being increasingly found to be important for manufacturing and business intelligence, once characterized and transferred. Various approaches are being investigated to achieve more timely and easy access to this data. These approaches include: (1) using machine-to-machine technologies and standards to connect equipment or specialized equipment networks directly to corporate clouds and (2) adapting elements of ubiquitous network technologies to factory networks while maintaining performance characteristics such as determinism, availability, security, and robustness that are needed to insure safe and proper operations. While hierarchies won’t disappear, plant architectures will slowly become more homogeneous and provide a common means for collection of data from lower layers. Challenges lie in avoiding adverse impacts on the performance of production systems and networks, in providing confidentiality of data (at and after collection), and in providing means to normalize and merge diverse data such that it provides correct views of an entire portion of an enterprise. The expected lifespan of capital assets, issues of safety and availability, and many characteristics required for manufacturing control networks also apply in other domains.

Approaches, technologies, or architectural elements that address data integration problems in many or all domains of cyber-physical systems will have a broader and longer impact than those that apply narrowly. Two standards that seek to provide a comprehensive solution to data integration are: the Digital Object (DO) Architecture [88] and Recommendation ITU-T X.1255 [87]. They represent a basic architectural foundation whereby mobile programs, smart applications and services, and devices of various kinds involved in managing information in digital form can exchange information on the location and provenance of data. Also of note is the recent establishment of an infrastructure to manage the evolution and deployment of this DO Architecture globally [124].

4.2.2.2.2 Example

An embedded-control boiler system that has been in service for decades is being migrated into an IT infrastructure through a new capability. Previously, the data generated from this system was generated, stored, and could be accessed by known parties using locally known infrastructure through set data paths. Now this data must be made accessible globally, for use in unknown and potentially complex systems, through unknown infrastructure. A tool is required that would enable such transactions or operations.

The simplest method of storing and locating data in this scenario employs a repository that is part of a secure cloud computing service that can be uniformly accessed by any number of authorized third parties. This may present challenges to data privacy and ownership as once the data moves outside of the originating entity's infrastructure, it becomes subject to the cloud computing service provider's trust framework. In addition, if the originator wants to move the data from one service to another, the data pathway changes and must be changed with all accessing parties as well. Credentials may also have to change.

The originating entity might instead choose to host the data in their own infrastructure for better privacy; however, this introduces the same kind of complexities as described above, and may increase security and privacy concerns. As the data ages, originators might need to move old data into storage or destroy it altogether. Network locations and naming conventions may change over time as the originator's system evolves. Abstraction can be used to limit the amount of manual work required to maintain data in such a scenario; however, this increases the complexity of initial setup. All these factors increase the complexity of maintaining persistent data pathways for accessing parties and present major challenges to efficiently realizing value from the data. The resulting inability of users to store and manage their own data is a challenge for maintaining an open, competitive, secure, and privacy-enhancing data marketplace.

4.2.2.2.3 Discussion of relevant standards

Modern web standards and practices provide many tools for describing, fusing, sharing and accessing distributed heterogeneous data (see Christian Bizer, Tom Heath, and Tim Berners-Lee, "Linked Data – The Story So Far" [91]. The standard web infrastructure and protocols [92] provide a means for accessing and sharing distributed data. Any kind of element can be considered a resource and named using an internationalized resource identifier (IRI) following guidelines in the standards and practices associated with linked data. These standards include the Resource Description Framework (RDF) [93]and the Web Ontology Language [94] for describing the data (i.e. these are languages for metadata), formats for encoding the data and related metadata for sharing and fusing [95], and a protocol and language called the SPARQL Protocol and RDF Query Language [96] for merging (via SPARQL endpoints) and querying the data. These standards and approaches have been used to integrate industrial data in the electric power industry, oil drilling industry, and the manufacturing shop floor among others.

Digital entity data model: This is a standardized approach that makes use of components of an infrastructure that are distributed and interoperable with each other in practice [87] [88]. It is compatible with existing Internet standards.

4.2.2.3 Data-driven interactions between dependent and independent cyber physical systems

4.2.2.3.1 Overview

For effective and controlled data interaction to occur between the various elements of a particular CPS system, roles, procedures, rights, and permissions of the humans who create and manage each system must be defined. These humans will ultimately be responsible to setup, manage and maintain both the cyber and physical components of such systems.

The primary challenge then, is to develop a set of definitions that is comprehensive and unambiguous, so that interactions between systems can be appropriately described and standardized. The multiple dimensions involved are discussed herein.

As it regards data-driven interactions between dependent and independent cyber physical systems, for clarity the author identifies three sub-sections (actors, roles and permissions).

ACTORS: While any particular CPS instance may involve different *actors*, they will generally fall into three categories:

1. Those who manage the data elements; (*Data Mangers)*
2. Operations/Production personnel who interact at some level with a CPS element; (*Operations Staff)*
3. Governance, Risk and Compliance (GRC) personnel who manage the various security and governance elements that may be required; (*GRC Staff)*

These *Actors* will interact with each other based upon their defined *roles (see below)*, with each role consisting of a series of *permissions* that will govern such interaction.

ROLES of actors:

Data Managers will be responsible to create the processes that will manage all data elements that initiate an action to, or are the result of an action from, a CPS device. Data Managers' roles will include program development, testing and deployment, database management and data analysis management.

Operations Staff will be responsible for the physical devices that are employed as well as those that perform the actual human tasks that may be inclusive in any set of managed processes.

GRC Staff will be responsible to define and manage all processes and rules that may be required to meet governance and oversight standards that apply to certain processes.

PERMISSIONS

Permissions will be established for each role of each actor and will govern the actions that each actor will be responsible for.

The following permissions and their associated definitions will be present in most CPS systems:

- Define interaction points between devices
- Initiate specific interaction points between devices
- Monitor interaction points between devices
- View Data
- Modify Data
- Create new workflows
- Import Data from other sources
- Export Data to other sources

Control Processes and Procedures - The actual control processes and procedures must be clearly defined. Some examples of these processes and procedures include:

- *Interaction points between devices* - CPS devices, whether dependent or independent, will need precise parameters by which they can interconnect. This may vary even with the same device, based on what other input/output is being employed for any particular instance.
- *Initiate specific interaction points between devices* - Once the interaction points have been established there must be a trigger, or event, which initiates the ensuing process(s). In a dependent CPS device, the triggered data event will most likely begin once the output of its dependent source begins transmission. In an independent CPS device that initiation will range from simple 'human' kick-off to timing devices that auto-start the independent device.[8]
- *Monitor interaction points between devices* - It might be argued that this is a combination of 'Presence' and other factors defined below, but nonetheless procedures

[8] From a pure definition standpoint it must be determined if such an action makes the 'independent' device a 'dependent' device, as its function is 'dependent' on the stated action.

must be clearly defined that continuously monitor these interaction points to ensure that they are reporting and functioning as required for each specific interaction.

- *View Data* - Define which actors are given access to specific data sets.
- *Modify Data* - Define which actors have the authority to modify data once transmitted. [9]
- *Import Data from other sources* - Define which actors have the authority to import data to other systems or databases.
- *Export Data to other sources* - Define which actors have the authority to export data to other systems or databases.
- *Create* workflow for each component process - The fact that there will be differences in precisely how each component process occurs or interacts necessitates clearly defined workflows so that consistency is maintained regardless the origin of any particular process. A critical companion of each ‘flow’ must be an audit trail that is never allowed to be modified or deleted at any point. Unless such an absolute audit trail is in place, it will be impossible to determine with certainty what may have occurred should any such component procedure fail or be compromised in any way.
- *Sanitize data to conform with regulatory and privacy requirements* - Owing to the magnitude of ‘big data’ that may be produced by CPS devices, there may be a need to sanitize data to remove extraneous elements that result during specific operations. Any such ‘sanitizing’ must be strictly controlled, including which actors/devices may perform such action and a requirement that all actions must be instantly and permanently archived in a way that prevents tampering after the fact.
- *Interact with other data sources* - CPS devices may need to interact with other non-CPS sources of data, such as an on-line security check of personnel. Such interaction may be automated, or conducted by humans. The procedures and methodologies must be clearly defined and data-maps must pre-established for consistency between such sources.
- *Report outcomes to stakeholders/actors* - As cited in the example below, there must be defined procedures and process by which each actor/stakeholder interacts with the output data. In certain instances it may be simple reporting for archiving purposes, while in other instances notification may need to be immediate and redundant if mission-critical actions are required.
- *Request Permission to Modify/Delete Data* - Define the process by which individual users may initiate a request for their ability to modify/delete data, including which specific sets of data to which the permission applies. Any such action must be strictly controlled and instantly and permanently archived for future auditability.
- *Define Rights and Permissions* –Strict controls must be emplaced that determine the rights and permissions that each actor may be granted or restricted to. These determinations must include not only audit trails, but multi-level redundancy in

[9] Precise audit trails must be in place for compliance and regulatory oversight permission be granted to modify viewed or transmitted data in any way.

managing these processes and procedures to ensure compliance and enable regulatory oversight. Rights will include, but not be limited to: View Data Only; View and Suggest Data Modification; and, View and Modify Data

Finally, various mechanisms must be developed and specified to ensure that these processes are implemented correctly and reliably. Examples of these mechanisms include:

- Assure that all required CPS devices are functional
- Assure that all required CPS devices are in place to monitor their intended functions
- Assure that data is bring transmitted in the form and format needed
- Establish Trust Factors

4.2.2.3.2 Example

The following illustrative example discusses how various CPS devices will interact to improve security to help manage the procedures and processes to control the safety of the more than 6,000,000 shipping containers that enter US ports annually [110].

While this example represents a potential comprehensive end-to-end solution, the author recognizes that it will be impossible to 'boil the ocean' in an attempt to reach all of the stated goals as a single project.

Therefore, this project must be divided into smaller subsets that will be effective on their own, while leading to a full implementation over some undefined period of time. Given the importance of securing our ports and the dramatic increase of the rhetoric of terrorist efforts to create a major adverse event in the US, it cannot be understated that substantive changes to the system must begin or such an event will become likely, rather than theoretical.

A suggested roadmap of some of these iterative steps will follow the example detail that follows below:

- There exist multiple vulnerability points from the time that a shipment originates in a foreign country until that shipment arrives at its final US destination.
- In this case, the manufacturing point of origin is the primary point of vulnerability where goods can be tampered with, or hazardous materials can be packaged and concealed as the product is being prepared for shipment.
- RFID tags could be placed on each item and locator tags then placed on each pallet used to load a container to track the movement of all items. An assigned freight supervisor will monitor the loading of the shipping container, assuring that each item has a RFID tag. As each RFID tag is attached a resultant scan will transmit the data to a secure storage system.
- Finally, the supervisor will place a Digital GPS Tracking device within the container and secure that container with a digital seal that will instantly report any tampering to the secure storage system cited above. All associated and/or resultant actions will result in that data being transmitted to a Cloud database or other monitoring system.
- Standard screening inspections of the shipping container at port of exit are then

performed using devices such as a gas chromatograph or mass spectrometer working through container air vents to ensure that no explosives or harmful chemicals are present.

- These CPS devices will instantly upload data to a central data repository that will alarm should any negative feedback result using the Digital GPS Tracking device mounted inside or attached to the container.
- If final packaging and consolidation was not performed in the factory as described above, it will usually take place at a warehouse or staging area which prepares the product shipment for truck or rail transport to the port. At this stage, illicit activity can occur while products are being consolidated into larger shipping loads, and while being trucked or railed to their maritime port of debarkation. Constant surveillance within the warehouse facility, final load inspection, and employee background checks for both warehouse and transport personnel are effective to improve security. As a prepared load is being transported, a truck can easily be diverted from its given route, providing the opportunity to tamper with the shipment. The use of Global Positioning System (GPS) technology gives transportation management the ability to better track adherence to routes. Truck drivers often have broad discretion over their routes, and are subject to last-minute changes.
- Freight dock supervisors will constantly monitor the RFID tags of each piece of freight or pallet to ensure that it remains on its proper path.
- These RFID devices will instantly upload data to a central data repository that will alarm should deviation from established routes occur for any tagged piece of freight.
- Once the container is at sea, procedures must be in place to prevent tampering. Containers typically do not have a uniform seal or any way to exhibit obvious signs of tampering. Ocean carrier personnel may not routinely check containers for seals or signs of container tampering while onboard. Container ships often stop at various seaports to unload and load containers. The container ship's transits through various routes and ports pose different levels of security risks
- A digital tampering device combines a covert Assisted GPS tracking & sensing device with a re- usable electronic seal that can be affixed to a conveyance door. The GPS tracker can be hidden within a pallet. This system is web based, so when a seal is compromised, the GPS device sends the event and location information to the stakeholder for immediate action. This system can be used for cross border or domestic trailer tracking using cellular and web based technology.
- As above, this CPS device will instantly upload data to a central data repository that will alarm should any tampering occur.

Once the container arrives at the Port of Entry the shipping containers may be at risk of tampering, especially if they must sit for extended periods of time before being staged and loaded onto a cargo ship. Terminal operators may not routinely check containers for seals or signs of container tampering so a device such as is described above will help to further ensure the integrity of each container.

Re-conceptualizing basic legal documentation in the maritime industry, in particular *bills of lading,* may also serve to enhance security and reliability across various related industries such as shipping, banking, and insurance. Where unique persistent identifiers are associated with information structured as digital entities (aka digital objects), it is possible to move beyond static information and create more dynamic data structures. As an example, if a storm occurs at sea, and a container is swept into the ocean, video information captured at the time of its lading when compared to conditions at the time the container broke loose may be used to identify possible negligence in strapping down the cargo; and the relevant insurance companies may be notified as appropriate [89][117].

4.2.2.3.2.1 Suggested order of the first two iterative projects

PHASE 1

A. Standard screening inspections of the shipping container at the port of exit are performed using devices such as a gas chromatograph or mass spectrometer working through container air vents to ensure that no explosives or harmful chemicals are present.

B: Place a Digital GPS Tracking device within the container that will track that container's movements so that any diversion of previously specified routes will cause an instant notification to appropriate authorities.

C: Once the inspection is completed, secure that container with a digital seal that will instantly report any tampering to the secure storage system. All associated and/or resultant actions will result in that data being transmitted to a Cloud database or other monitoring system.

PHASE 2

A. RFID tags could be placed on each item and locator tags then placed on each pallet used to load a container to track the movement of all items. An assigned freight supervisor will monitor the loading of the shipping container, assuring that each item has a RFID tag. As each RFID tag is attached, a resultant scan will transmit the data to secure storage system.

B: As in Phase 1.C above: Once the inspection is completed, secure that container with a digital seal that will instantly report any tampering to the secure storage system. All associated and/or resultant actions will result in that data being transmitted to a Cloud database or other monitoring system.

Instituting these two phases will dramatically improve the possibility of spotting or preventing a major adverse event berfore it becomes a disaster.

4.2.2.3.3 Discussion of relevant standards

There are few relevant standards that apply to this aspect of CPS Data Interoperability. However, the above-referenced example [110] was required to comply with a variety of other applicable standards. These standards include:

- Department of Homeland Security, US Customs and Border Protection, Container

Security Initiative (CSI) program [110]
- Department of Homeland Security, US Customs and Border Protection, C-TPAT (Customs Trade Partnership Against Terrorism) program [111]
- Department of Homeland Security, US Customs and Border Protection, Bonded Warehouse Manual for CBP Officers and Bonded Warehouse Proprietors [112]
- Department of Homeland Security, US Customs and Border Protection, *Amendment to the Current Reporting Requirements for the Ultimate Consignee at the Time of Entry or Release*, [113]
- Department of Homeland Security, US Customs and Border Protection , *International Carrier Bonds for Non-Vessel Operating Common Carriers (NVOCCs)* [118]

4.2.2.3.4 Summary Analysis

Data-driven interactions between dependent and interdependent CPS require a precise unambiguous set of definitions to describe and regulate these interactions. Some of the needed definitions include roles of actors, control processes and procedures, and monitoring mechanisms. There may be an opportunity to describe and standardize these definitions to enable robust interactions between dependent and interdependent CPS.

4.2.2.4 Privacy-protecting data infrastructures (the ubiquitous nature of IoT/CPS creates the potential for data in these environments to be intrusive)

4.2.2.4.1 Overview

Protecting the privacy of the humans, businesses, nation states, non-profit institutions, and other entities involved in a complex CPS is an increasingly difficult proposition as data is being produced in greater volumes, from a greater variety of sources. Complex proprietary data infrastructures have combined to make the overall data infrastructure more opaque, and data access controls vary dramatically as the number of vendors and products that produce data in a CPS grow into the thousands. Data is often ‘mined’ in ways that don’t currently require a user’s explicit permission. Data storage is increasingly moving away from the users that own the data and is being centralized in third-party ‘cloud’ servers. Movement of data often includes multiple third party brokers or aggregators. Data ‘leakage’ is often a side effect of data collection (e.g., an observer can use appliance data to determine if a user is at home). Ironically, attempting to impose access control and integrity protections can actually serve to decrease user privacy as the authenticating information, which is stockpiled with increasing numbers of security administrators, grows.

Lack of a uniform way to identify, secure, store, and access data across proprietary system boundaries has made it difficult for users and institutions to effectively manage their privacy. Indeed, companies such as Google have recently made it clear to regulators in places such as the EU that, given today’s infrastructure, it is virtually impossible to give a user the ability to be ‘forgotten’ in the internet.

The release of personal information, even to support the normal functioning of a system (e.g., the provision of services individuals request) can still raise privacy risks. These risks could include stigmatization of the individual or loss of trust from the unanticipated revelation of personal information or from the release of inaccurate information. Thus, any standard or implementation needs to incorporate design requirements and privacy-enhancing controls to support the protection of privacy and civil liberties in the developing CPS ecosystem. User management of the release of attributes is one such control.

Although user control is important, individuals are not always in the best position to mitigate all privacy risks.

Therefore any potential approach should include design requirements and controls that do not rely solely on user management. Requirements that provide the capability for claims to be derived instead of releasing actual values can limit the unnecessary disclosure of personal information. For example, if an online credential can get a teenager into a movie theater, the only exposure necessary is that the teenage is older than seventeen. Full birthdate, even birth year, is not needed. Metadata should also have privacy enhancing controls. For example, if 'over 17' is asserted, the implementation should consider that a 'valid DMV' asserted that fact, not that 'the Virginia DMV' asserted it, causing unnecessary data leakage. The objective is to consider the full range of privacy risks and appropriate mitigation strategies that can be incorporated into executable, implemented systems, and not just rely on manual, management policies.

4.2.2.4.2 Example

An advanced utility grid is using data from millions of syncrophasers, heat sensors, vibration sensors, and other data production points to balance power generation against system load through "sense, actuate, and control" CPS systems. Sources of data generation in this environment include power generation assets owned by a variety of vendors, Independent Service Operators (ISOs), public distribution infrastructures, to local municipal infrastructures, to right inside the consumer's home.

The information collected comes from a variety of different sources, through a variety of infrastructures, and via a variety of different market pathways. Consumer data such as power consumption information from appliance vendors may be used to estimate potential load on the grid but can also "leak" information such as when a person is at home, what specific electricity-consuming activities the person is engaging in, and even what media a person is consuming on their devices. Asset operators may expose proprietary operational information, such as which assets are utilized in certain scenarios and how assets are being utilized and managed, just by providing data to central aggregation/analytics points. Even public information may be collected and analyzed. For example social media surrounding popular sporting events that may give a hint of load spikes to the grid, but may also reveal information about individual participants in the aggregated data.

A user - whether institutional or individual - who wishes to protect their privacy in such a system of systems may have a very difficult time simply locating all the different collection points and data stores that track their usage patterns, and may not even be aware of the individual data collection practices of the vendors involved. A user in such a scenario has very little expectation of privacy and very little capability to control what information of theirs is being shared with whom, and for what purpose.

4.2.2.4.3 Discussion of relevant standards

To truly enhance privacy in the conduct of online transactions, the Fair Information Practice Principles [139], must be universally and consistently adopted and applied in the CPS Ecosystem. The FIPPs are the widely accepted framework of defining principles to be used in the evaluation and consideration of systems, processes, or programs that affect individual privacy.

However, the FIPPS may not be enough when engineering automated systems. As such, NIST, in a public and private partnership, is exploring “Privacy Engineering” methodologies to integrate privacy-preserving controls directly into systems as opposed to depending solely on documented paper policy. As illustrated in the following graphic, the FIPPS provides the baseline input to an overall privacy engineering methodology, but is not the sole tool used to impact effective privacy management.

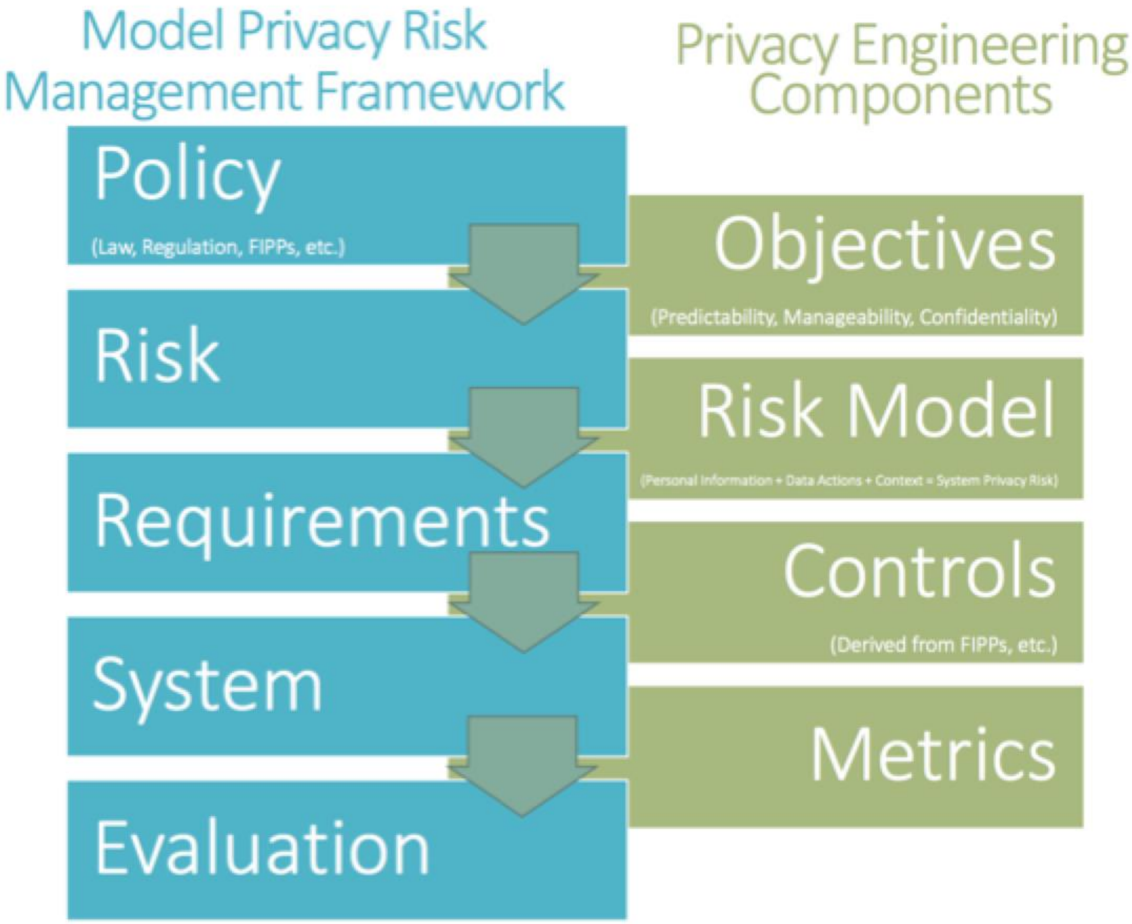

Model Privacy Risk Management Framework
Privacy Engineering Components
Policy
(Law, Regulation, FIPPs, etc.)
Risk
Requirements
System
Evaluation
Objectives
(Predictability, Manageability, Confidentiality)
Risk Model
(Personal Information + Data Actions + Context = System Privacy Risk)
Controls
(Derived from FIPPs, etc.)
Metrics

Figure 15: Continuous Refinement of Privacy Risk Management

These concepts are under continuous refinement, but could serve as another data point in CPS efforts to engineer privacy directly into systems that handle potential personal information.

Specifications like OAuth, OpenID Connect, and User Managed Access (UMA) allow explicit user control over information release. During transactions governed by these specifications, where a third party is requesting information, the user is required to consent prior to disclosure. Fine-grained user controls are possible that allow individuals to manage consent in a myriad of ways. For example, a user can allow one-time release, white list entities where release doesn't require consent, turn consent on/off for an individual datum, or revoke consent for any or all previously authorized entities. Emerging concepts such as Personal Data Stores (PDS) can and should influence attribute standards and should be built upon existing standards that give users explicit control and choice over the information they share.

Other approaches include, but are not limited to, cryptographic profiles that include zero-knowledge assertions such that intermediaries or brokers cannot see attribute values, and design requirements that limit the building of user profiles by preventing identity providers from knowing the consuming relying parties. Commonly known as double or triple blind, this latter approach is not codified in any singular standard, but is becoming a de-facto implementation technique to limit traceability of users online. Figure 16 is a data instance diagram of a possible double-blind scheme.

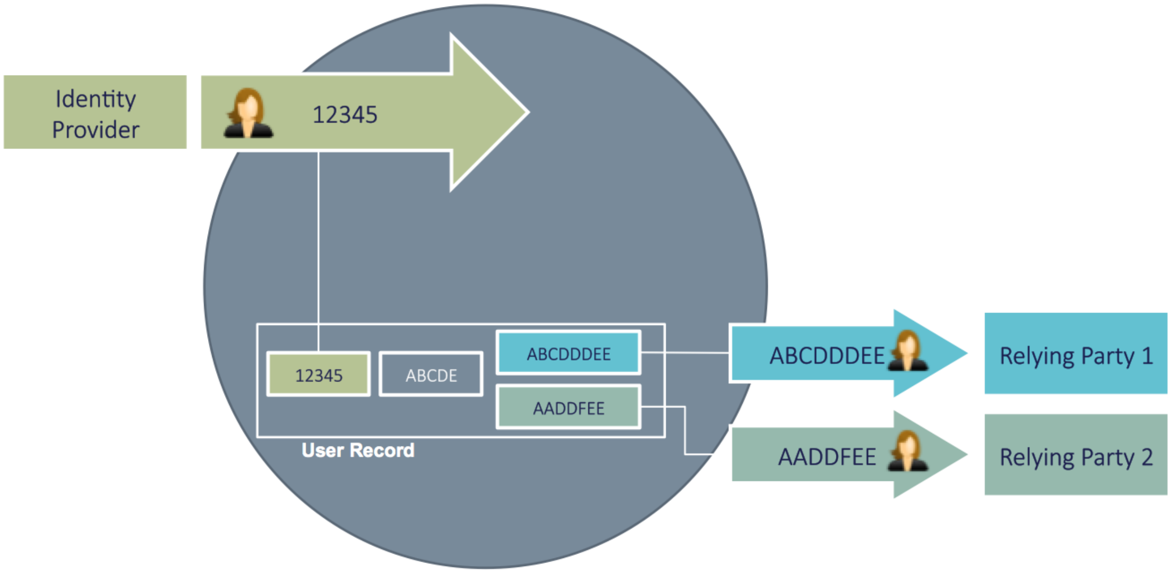


Figure 16: Double-blind Authentication Scheme

This model is designed specifically to ensure that privacy requirements of anonymity, unlinkability and unobservability are built in from the start. However, without the appropriate cryptography, this model allows user information to flow freely through the broker depicted by the gray circle. Although great care is taken to generate pseudonymous identifiers throughout the system, any personal information provided by the identity provider needs to be encrypted in a manner that keeps the broker from viewing information. This is simple in traditional PKI systems where the source system (the IDP) encrypts the data for the destination system (the RPs) using the RP public key. Yet, traditional PKI breaks the design requirements of anonymity, unlinkability and unobservability because knowing which public key to use means the IDP knows where the user is going. Open, tested, and approved cryptography algorithms must be used to keep attributes encrypted without exposing the user destination to the IDP. Such cryptographic techniques are not yet available in common use. Finally, the broker is in an extreme position of power, as well as being a prime attack vector for those who wish to do harm. Automated compensating controls, in addition to paper policy (contracts, laws, regulations, etc.), are still under development to reduce or eliminate the vulnerabilities of the double-blind, broker-centric architecture.

4.2.3 Traditional data interoperability issues

4.2.3.1 Data Models, Relationships between Data and Data Type

4.2.3.1.1 Overview

Terminology has evolved from the ANSI notion of data modeling that described three types of data schema (or model): a conceptual schema, a logical schema, and a physical schema. Often, the key distinction now is between data models and information models. The discussion below is largely derived from a presentation by Ed Barkmeyer [122] to the Ontolog Forum in 2007, though there are other sources that similarly distinguish data models from information models such as RFC3444 [123] entitled "On the Difference between Information Models and Data Models"

Data models and information models differ both in nature and purpose.

Data models relate data to data. They support software implementations and organize data for access, encoding, or processing. Their classifiers (i.e. primary language constructs) describe the structure and type of the data.

Information models relate things to other things, as well as things to information about those things. These models are used to support a set of business processes or describe a domain and organize information for human comprehension. They use classifiers to collect properties. Transformation rules often exist for information modeling formalisms to data modeling formalisms to enable generation of data models from information models.

Semantic models (many of which are called "ontologies") are information models that are meant for machine "comprehension". These models use information to classify things. Semantic models are often constructed using knowledge representation methods, languages, and technologies. Such languages are sufficiently formal to support machine reasoning that provides this comprehension. Examples of inferences this can support include: revealing relationships between elements of independently authored ontologies or data sets (classifying both types and things), determining the logical consistency of a model, and determining the satisfiability of particular elements of a model (i.e. whether or not it is possible for any instance to exist that satisfies all the constraints of its type).

A way to distinguish these different kinds of models is by what their classifiers classify and how they do it. If the main classifier in a modeling language describes a data structure (such as an Element in XML Schema) then it is a data modeling language; if it describes properties associated with an entity (such as attributes and associations for a class in UML or relations to an entity in ER diagrams) then it is an information modeling language.

As one moves up this spectrum, the models become less prescriptive and more descriptive. Semantic models have flexibility that is quite useful for integrating information, but data models have the specificity needed for insuring its integrity for use in implementations of critical systems, thus both are useful for data integration in CPSs.

An obvious goal of data exchange is conveyance of understanding from the data source to a destination user of the data. There has been much work on defining interoperability and understanding; it has been developed from very theoretical first principles to quite practical terms. Some examples can be found in the Web Ontology standards from the WC3 [119][120].

This section describes the three key dimensions that allow conveyance of understanding. Note that other aspects of data interoperability are covered in other parts of section 4.2.3, but this one deals with the data itself.

The first subsection describes the concept of data models (sometimes called semantic models or information models) and how they typically scoped and described.

The second section describes metadata as data related to other data, outlines the major kinds of metadata used in the library community and how these kinds relate to our concerns, describes the importance of metadata to data interoperability for CPS, and enumerates some things that may need to be done with respect to metadata standards to enable data interoperability across CPS.

The third section describes data type and structure.

4.2.3.1.2 Data Models

"A message to mapmakers: Highways are not painted red, rivers don't have county lines running down the middle, and you can't see the contour lines on a mountain." [Kent, William, updated by Steve Hoberman. "Data & Reality: A Timeless Perspective on Perceiving and Managing Information in Our Imprecise World." Westfield, NJ: Technics Publications, 2012. Print]

The above tongue-in-cheek quote begins the 1978 preface to William Kent's classic book on data modeling, *Data and Reality*, and shows that everyone understands data modeling to a certain degree. Reducing, for the moment, the nice distinctions made above among data, information, and semantic modeling to a single concept, we can address the general challenge with modeling, which is the difficulty of mapping some subset of the real world, including cyber-physical systems, onto a conceptual structure that allows us to more easily understand and/or manipulate that real world subset, within certain constraints. Those constraints include the limits of the modeling language used, i.e., what can and cannot be expressed using the language, and the difficulty of capturing all of the relevant information. Furthermore, even using the same modeling language, multiple individuals can easily create variant conceptual structures describing the same real world subset. With this in mind, the relevance of data modeling to data interoperability is quite clear. Data captured from a given cyber-physical system will be structured according to a certain model and that model will be constrained by the modeling language used, by the level of granularity of the data collected, and, now going back to the distinctions among data, information, and semantic models described above, the basic type of modeling being done. Combining data streams from multiple cyber-physical systems at multiple times structured according to multiple data models using multiple approaches to structuring the data is a specific and challenging subset of the general and well-known problem of making sense of heterogeneous data sets.

Approaching specific data interoperability problems in CPS will require understanding the data modeling, or even lack of modeling, that has resulted in the available data structured or presented as it is. As noted elsewhere in this document, a clear requirement for data interoperability among cyber-physical systems is that many cyber-physical systems are legacy systems that must be accommodated in any data interoperability scenario and that clean slate solutions ignoring that legacy are unacceptable.

It is tempting to compare modeling approaches to each other and to favor one over another, but that ignores both the issue of legacy systems and the even more basic fact that different situations and different points of view require different approaches to modeling and no single solution fits all cases. Contrast, for example, OMG's UML (Unified Modeling Language) and W3C's OWL (Web Ontology Language). Both are widely used, historically by separate communities for different purposes, both are appropriate to those purposes, and both can be used synergistically within the same domain. UML comes out of the software engineering and more traditional data modeling community while OWL comes more out of the artificial intelligence community and looks at knowledge representation. One cannot be favored over the other in general, but each is appropriate to and solidly in place in its own community. It is beyond the scope of this document to compare modeling approaches but furthering the work of data interoperability in CPS will require understanding those approaches and the tools that can help in mapping from one to another.

One issue that will come up over and over in data modeling is the issue of metadata, which is further discussed below. Data, including data relevant to CPS, goes through a life cycle. At each stage the difference between data and metadata is not in kind of data but in the relationship of that data to other data. Thus, what is considered primary data and what is considered metadata can vary through the life cycle?

Here are some examples of typical names of data sets where this consideration could apply. These may not be orthogonal depending on the detailed definitions:

- Status – often derived states from other data categories
- Control – actuators and supervisory control points
- Measurements – sensor data

- Settings – set points for algorithms and alarms
- Documentation – manufacturer information, schema references
- Configuration – parameters that bind the device to its system
- Capability – possible degrees of freedom for settings and configuration
- Faults – logs of significant events and problems and their management
- Access management – authorization and authentication information (see privacy and cybersecurity section below)
- Identification – identifiers both traceable and opaque (people, processes, devices and systems); as well as identifiers associated with the digital entities in which such pre-existing identifiers are incorporated for operational purposes.

Note that typically, the ability to communicate these values is often regulated by access rights which include authentication as well as authorizations. This is a type of metadata.

Going back to Bill Kent, in his introduction to Entities:

"As a schoolteacher might say, before we start writing data descriptions let's pause a minute and get our thoughts in order. Before we go charging off to design or use a data structure, let's think about the information we want to represent. Do we have a very clear idea of what that information is like? Do we have a good grasp of the semantic problems involved?"

Paraphrasing that for purposes of thinking about the interoperability of data coming out of different cyber-physical systems, we might ask if we have a very clear idea of the data we are trying to integrate and a good grasp of the semantic problems involved.

4.2.3.1.3 Relationships between Data

In his 1968 dissertation, Philip Bagley may have coined the term “metadata” as data about data. In his Extension of Programming Language Concepts [98], Bagley says: "As important as being able to combine data elements to make composite data elements is the ability to associate explicitly with a data element a second data element which represents data 'about' the first data element. This second data element we might term a 'metadata element'."

The way that a "metadata element" in Bagley's definition relates to the data element it describes can be thought of as a role of the metadata element with respect to the described data element. All it means, then, to say that something is metadata is it that it relates to other data in a particular way. However, communities differ on which relationships constitute a metadata role. In some communities, everything but raw measurements are considered metadata, while in others complex data structures may capture many of the important relationships among data with metadata only providing data about the entire collection.

Types of metadata correspond to different ways that data can relate to other data. The library community makes heavy use of metadata to describe information resources. NISO, the National Information Standards Organization, describes three main types of metadata [121] used in this community that are also important in the information technology realm. These three types are structural, descriptive, and administrative.

According to NISO, "*Structural metadata* indicates how compound objects are put together, for example, how pages are ordered to form chapters." In the IT realm this type of metadata can include data models, data type identifiers and descriptions, and models used to describe structural metadata (aka metamodels). In other words, structural metadata is data about the containers of data.

NISO asserts that "*Descriptive metadata* describes a resource for purposes such as discovery and identification. It can include elements such as title, abstract, author, and keywords." This kind of metadata relates to the nature and identity of the data or the thing the data is describing.

Finally, NISO asserts that *Administrative metadata* provides information to help manage a resource, such as when and how it was created, file type and other technical information, and who can access it. There are several subsets of administrative data; two that are sometimes listed as separate metadata types are:

- Rights management metadata, which deals with intellectual property rights, and
- Preservation metadata, which contains information needed to archive and preserve a resource.

In the IT realm administrative metadata will include provenance data as well data on who may access which information and how.

Metadata may be structured or freeform (e.g. freeform text tags assigned by users to web links, files or services). Metadata describing metadata is also important to evaluating its use.

Metadata is critical to integrating data across diverse systems and having confidence in the implications of the results. Structural metadata provides a means to agree on common forms for exchange or determine commons forms for aggregation. It also provides information on how to parse the data and assess its integrity (e.g. by its conformity to the structure and rules specified in its data model). Descriptive metadata supports finding data relevant to a particular purpose, assessing its veracity, and assessing its compatibility with other data. Administrative metadata supports assessing freshness, trust, and availability of data, as well as the means of access and use.

There are many standards for these different kinds of metadata. For data interoperability to work quickly and safely in cyber-physical systems, one must assess what is needed from each type of metadata, which metadata standards are in use in different CPS domains, how they relate, and how they should be extended or narrowed to meet time, availability, and safety requirements for data interoperability for cooperating cyber-physical systems.

On the other hand, Bagley recognizes that metadata represents the need to be able to associate explicitly one data set with another. For example, for a control application, the data might be temperature or energy or relay state. The metadata might be units of measure, scaling, uncertainty, precision, etc. Additional metadata might include make/manufacturer/model/serial-number for the sensor monitoring temperature or energy or for the device having the state or attribute being monitored such as the relay. Yet to an asset management application the make/manufacturer/model/serial-number is the data.

The use of the term *metadata* may have evolved beyond Bagley's original usage to include analogous types of data about things such as devices and processes. A device data sheet typically describes characteristics of a class of device or machine and may be referred to as device metadata. This is analogous to the role of data type and data models with respect to the data it describes. Additionally, there may be calibration data associated with a particular device that is analogous to provenance information on the source and history of data instances. Since it may be useful to apply the same mechanisms used for managing data about data to these analogous kinds of data about other types of things, it may be wise to broaden the CPS interpretation of metadata to include these other uses of the term.

4.2.3.1.4 Data Type

Automated processing of large amounts of data, especially across domains, requires that the data can be parsed without human intervention. Within a given domain that functionality can simply be built into the software, e.g., the piece of information that appears in this location is always a temperature reading in centigrade or, at a different level of granularity, this data set is structured according to Domain Standard A including base types X, Y, and Z where the base types are things like temperature readings in centigrade. This knowledge, easily available within a given domain or a set of closely related organizational groups, can be built into processing workflows. But outside of that domain or environment the 'local knowledge' approach can begin to fail and more precision in associating data with the information needed to process it is required. This also applies across time as well as domains. What is well known today may be less well-known twenty years hence but age will not necessarily reduce the value of a data set and indeed may increase it.

We are using the term 'type' here as the characterization of data structure at multiple levels of granularity, from individual observations up to and including large data sets. Optimizing the interactions among all of the producers and consumers of digital data requires that those types be defined and permanently associated with the data they describe. Further, the utility of those types requires that they be standardized, unique, and discoverable.

Simply listing and describing types in human readable form, say in one or more open access wikis, is certainly better than nothing, but full realization of the potential of types in automated data processing requires a common form of machine readable description of types, i.e., a data model and common expression of that data model. This would not only aid in discoverability but also in the analysis of relations among types and evaluation of overlap and duplication as well as possible bootstrapping of data processing in some cases.

Types will be at different levels of granularity, e.g., individual observation, a set of observations composed into a time series, a set of time series describing a complex phenomenon, and so forth. The ease of composing lower level, or base, types into more complex composite types would be an advantage of a well-managed type system.

An immediate and compelling use case for a managed system of types comes directly out of persistent identifiers (PIDs) for data sets. Accessing a piece of data via a PID, either as a direct reference or as the result of a search, requires resolving the identifier to get the information needed to access the data. This information must be understandable by the client, whether that client is a human or a machine, in order for the client to act on it. For a machine, it must be explicitly typed. A type registry for PID information types would appear to be an early requirement for coherent management of scientific data.

Finally, assigning PIDs to types would aid in their management and use. All of the arguments for using persistent identifiers for important digital information that must remain accessible over long periods of time will apply equally well to whatever form of records are kept for data types.

A recent effort to codify types, still very much in development, is a Research Data Alliance (RDA) Working Group on Data Type Registries [137].

4.2.3.2 Identification of type and instance

How do I know what a piece of metadata is referencing? How do I find the metadata for a given digital entity? What ties all of these things together? And, finally, because we want people and processes that did not create the data to understand and reuse it, how do I understand them, and which are key to data interoperability?

Unique, persistent, and resolvable identifiers are essential to managing distributed data in the Internet and other computational environments. A digital entity that is referenced from outside its local domain must be uniquely identified and that identifier must be resolvable to allow for access to relevant and timely state information about the entity, e.g., current location or access conditions. This allows the identifier for a digital entity to persist over changes in the state of the entity, i.e., the identifier itself remains constant while the returned state data from a resolution request can change as needed.

Allotting a persistent identifier for a digital entity and maintaining that identifier for at least as long as the identified entity exists is a commitment, the success of which depends in the end on the organization or process that mints and maintains the identifier. Not all entities require this level of identification. However, an entity which is never referenced from outside of its local context would still require an identifier for local management purposes, subject only to local policies and procedures.

The conditions, under which the changes to an existing digital entity are judged to be sufficient to declare it to be a new entity, and thus requiring a new identifier, are application and domain-dependent. Moving a data set from one location to another, for example, clearly seems not to be essential to its identity, as it is still the same data set. Moving a sensor, however, from one location to another, might be seen as sufficient, as the core identity of a sensor might be seen as sensor type plus location. An assertion that two things are or are not the same must be made in the context of 'same for what purpose'.

An identifier may serve as a single point of reference to access a service that provides the required current state information as part of its service, including perhaps the digital entity itself. An identifier resolution system can be used as a late binding mechanism to connect current attributes to entities, e.g., current public key for a person or process.

Such an identifier system needs a method for dealing with fragments or subsets of identified entities, e.g., seconds N through M of a given video in digital form, where it would be impractical or impossible to assign unique identifiers for each potential fragment or subset.

Trust is a key issue in identifier resolution and takes multiple forms. On what basis do I trust that the resolution response received is indeed the response that was sent? On what basis do I trust that the resolution response reflects the data that was entered in the system by the party responsible for the identifier? And do I trust the information itself, i.e., on what basis do I trust the party that stands behind it?

The structure of the identifier string itself is of some importance. Experience has shown that building semantics into the string, while perhaps useful for minting and administering identifiers, can be dangerous in that it can tempt people and processes to make assumptions about the identified entity which are not justified. Any changeable attribute baked into the identifier itself, as opposed to the changeable record to which it resolves, results in a brittle identifier, e.g., identifying an entity by its location or ownership when either may change.

Although the TCP protocol was implemented to provide a virtual circuit mechanism, the notion of end-to-end in the Internet was never a requirement of the early protocol design work undertaken by Robert Kahn and Vint Cerf. As the Internet moves forward to embrace the so-called Internet of Things (IoT) [22], however, substantiation of a data “endpoint” is still of some interest in a scalable, unified data identification system. Also problematic is a location-centric or owning-entity-centric structure. The core of many challenges in sharing and managing data lies in our treatment of data entities as second-class entities, existing without continuous and credentialed identification. This means that we have a paradigm of securing servers, and then managing access to those servers. A key weakness in today's technological landscape is PKI-based credentialing systems that don't allow for interoperability across trust domains. The method of credentialing is therefore an important issue in data interoperability.

There are two distinct classifications of identifiers – traceable and untraceable. The discussion above provides clear rationales for where traceable and navigable identification schemes are valuable. The Universally Unique Identifier (UUID) typifies a second class of identifier [127]. A UUID may be necessary when anonymity is required, often for privacy purposes. Application requirements must dictate which and when identifiers of each kind, or both, are required.

4.2.3.3 Data quality and provenance

ISO/IEC 2382-1 differentiates information from data through the following definitions:

- Information
  knowledge concerning objects, such as facts, events, things, processes or ideas, including concepts, that within a certain context has a particular meaning

- Data
  Re-interpretable representation of information in a formalized manner suitable for communication, interpretation, or processing

ISO 9000 defines:

- quality
  degree to which a set of inherent characteristics fulfils requirements

ISO 8000, the international standard for data quality, defines data quality as data that: (1) references a syntax, and (2) is semantically explicit, and (3) meets stated requirements. By its very definition quality data is portable data (explicit syntax and explicit semantic encoding).

ISO 22745-30 is the international standard for stating requirements for data in a computer processable form using an open technical dictionary.

ISO 22745-40 is the international standard for the exchange of characteristic data in a computer processable form using an open technical dictionary.

ISO 8000 data quality can automatically be assessed by comparing ISO 22745-40 data to an ISO 22745-30 data requirement.

ISO 8000-120, the international standards for quality data with provenance, requires that provenance be provided for all characteristic values. Provenance is the identifier of the organization that provided the data, and the date and time the data was extracted. Provenance must be provided at the data element level, and not at the record or exchange level.

Quality data relies on a concept dictionary for semantics. A concept dictionary will contain the explicit definition of all encoded concepts to include metadata and code lists (reference data). A metadata registry typically only includes attributes (name of the characteristic) and their definitions, but a concept dictionary also includes code lists.

Note: examples of a code list: state code is the characteristic, CA would be a possible value, however it needs to be defined in a dictionary as CA=California, for example, other examples include material codes (SS=stainless steel), etc.

4.2.3.4 Governance

Data governance[10] is the collection of stated rules, regulations, and policies that govern data. Data governance is associated with a system of decision rights and accountabilities for information-related processes, executed according to agreed-upon models which describe who

[10] Note that the term Data Governance has little to do with legal and regulatory issues and is mainly concerned with enterprise-level policies and procedures.

can take what actions with what information, and when, under what circumstances, using what methods.

Data governance covers all data as shown Figure 17: Taxonomy of data below:

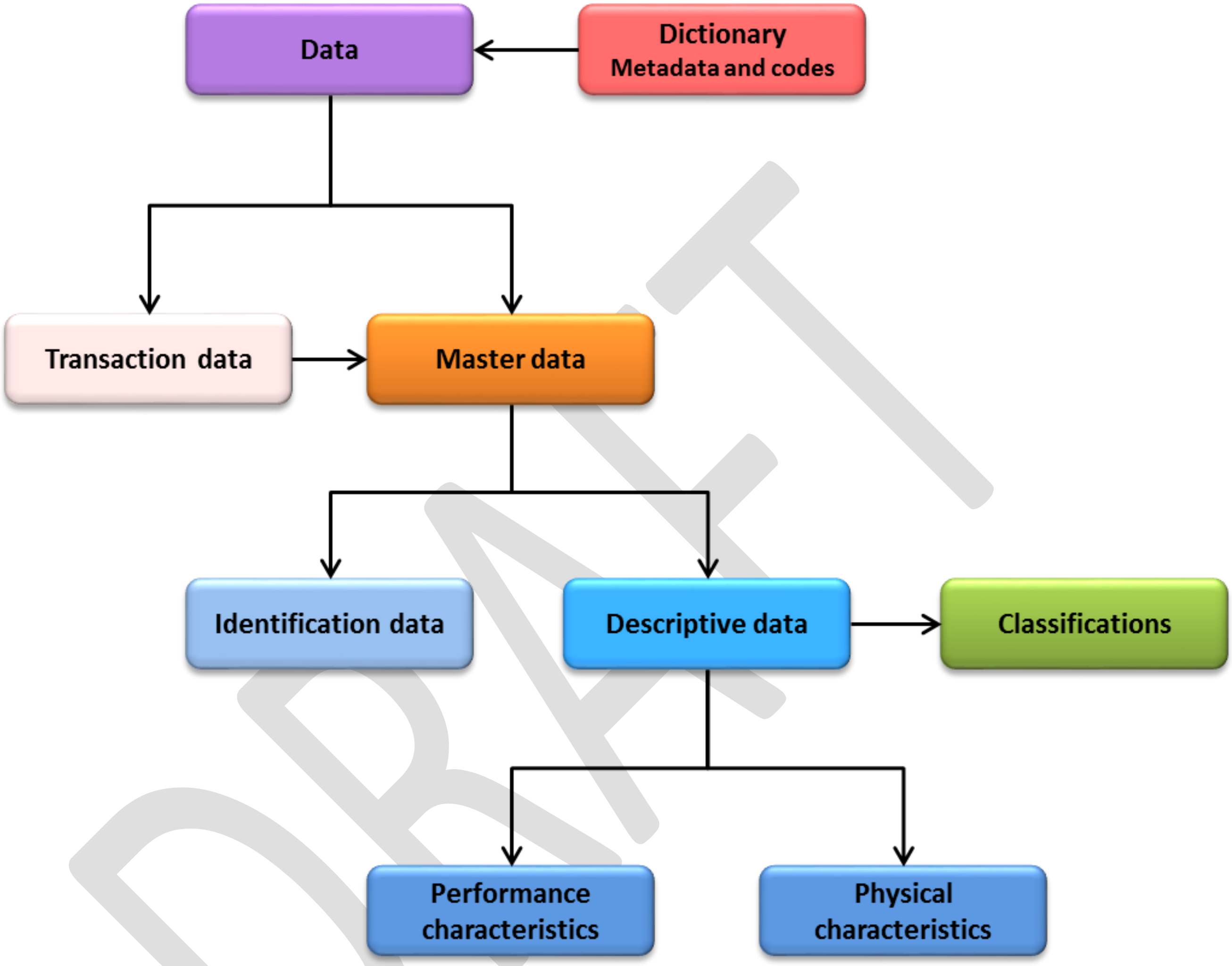


Figure 17: Taxonomy of data

Master data is defined as "data held by an organization that describes the entities that are both independent and fundamental for that organization, and that it needs to reference in order to perform its transactions. [104]"

Examples of master data include records that describe customers, products, employees, materials, suppliers, services, shareholders, facilities, equipment, and rules and regulations.

For CPS and data interoperability, the information exchange by the CPS is described as transaction data that is dependent upon the quality of the master data. A key requirement of data quality for CPS, in addition to syntactic and semantic definitions, is the notion that the data is portable; the data is application independent.

4.2.3.5 Privacy and cybersecurity

This section discusses the relationship between Cybersecurity and Data Interoperability.

Cybersecurity and privacy are often discussed using measurements of "Confidentiality, Integrity, and Availability," each holding more or less importance depending on the environment. Without comparing value, we'll use these anchor points to address traditional data interoperability issues with Cyber Security.

Confidentiality is obviously vital for privacy, as well as for control of information and the system itself. Control of information can make certain attacks (physical and cyber) on an entity more difficult to plan and execute successfully. Control of the system itself is vital for data integrity, which we'll talk about next. Standard solutions to confidentiality involve encryption. Encryption is only as good as the implementation of its algorithm, key exchange between parties, and key data storage. If any of these is poorly implemented, an attacker may be able to compromise the encryption, potentially leading to breach of privacy and/or control of the system.

Integrity of a given system is vital for trusting any of the data or behaviors the system provides. Attacks (e.g. credential compromise, memory corruption exploit, or man-in-the-middle attack) that allow for unauthorized modification of the information maintained by the system, or control of the system jeopardize the value and trustworthiness of the system. For instance, if a system generates, transports, or interprets sensor data from power equipment in the field to a control center, modifying that information along the path could lead to disastrous decisions by the people consuming the information. Likewise, if information about a crop report is intercepted and modified before being delivered to the agricultural market, decisions would be made which could destroy an entire portion of our society's food chain.

Typically authentication and authorization are used to ensure correct controls over a system, and cryptographic integrity checks (aka "digital signatures") ensure data has not been altered since creation. In addition, most networking layers provide integrity checks, but these are intended to identify accidental bit-errors, not to keep an attacker from modifying the data. Authentication is the art of ensuring the identity of an actor on a system. Several common methods are used to verify the identity of an actor, including shared keys/passwords, and multi-factor authentication which attempts to make impersonation more difficult. Passwords/shared keys mean that both sides have some type of pre-shared data. These passwords can be stolen if stored on a compromised device, and in many cases, they can be guessed and/or cracked offline. Multi-factor authentication attempts to ensure that the entity has at least two of the following: knowledge of some pre-shared key, some offline device, or some biometric evaluation. Multi-factor is currently only good at identifying human entities since it relies on the interpretation of something that is not network-attached (thus more difficult to compromise), but the key value of multi-factor is that an attacker must overcome multiple hurdles to impersonate an entity on the network. Best practices for each of these involve cryptographic means to verify the identity of a given entity, such that information is not immediately compromised over a network by an attacker who may be capturing and analyzing the data.

Authorization is ensuring that a particular entity is supposed to be performing an activity. This verification allows a system to have many verifiable entities, each only allowed to perform certain tasks. This concept of constraining information on a “need to know” is also known as the principle of least privileges.

There are numerous methods of verifying that data has not been modified in transit, including CRC, Checksums, and any given hash (MD5/SHA256/etc.) of the data. However, these methods only provide protection from accidental modification. An attacker need simply re-<method> their modified data and pass all checks. For this reason, cryptographic integrity checks (aka digital signatures) were created to ensure that the calculation of any integrity check was based on information only maintained by the original sender. This type of check has been integrated into most common encryption schemes, to ensure both confidentiality and integrity of the data... assuming no compromise of the information used to sign/encrypt the data.

Availability means that a system or data is accessible as needed or desired. This data or system may provide important information for a given process or may be part of a designed system of trust. For example, TLS, as used in HTTPS and other encrypted services, uses cryptographic certificates and a Public Key Infrastructure (PKI). This PKI uses something called a Certificate Revocation List (CRL), which is often just a web page with a list of certificates that are no longer trusted. If that CRL is not available when a TLS-enabled service is accessed, known compromised keys will still be considered valid, because the mechanism required to verify that a certificate has not been compromised is unavailable. From a process control standpoint, if a system is unavailable during manufacturing, chemical mixing, power drains, and a myriad of other physical events, products can be destroyed (or simply not produced), chemicals may explode, electrical components can be damaged, and otherwise "bad things" can happen. For this reason, control systems engineers tend to favor Availability over anything else, whereas common Information Technology (IT) engineers tend to favor Confidentiality and Integrity primarily and consider Availability more valuable when money and reputation are involved.

Availability is ensured through careful design and use of redundancy. Poor design can leave many single-points of failure that lead to services and data being unavailable when needed. Proper design of a system includes sufficiently redundant network connectivity, identifier name resolution (if necessary), and in many cases, redundant services and data. Services themselves may be provided behind a load-balancer or use some other fail-over method (which itself then has redundancy). Data may be served by one of these redundant services, and be mirrored between different storage media, providing further redundancy and availability. These are potentially complex solutions which require deep knowledge and understanding of their technology, which also has to be considered in proper design. Many operational technology (OT) devices do not have the luxury of redundancy. Many OT devices were designed before redundancy technology was cost-effective. Redundancy in these legacy systems tends to be nonstandard and difficult to work with.

Data Interoperability and cybersecurity are significantly intertwined. Cybersecurity requires that both sides of communication understand the security protocols in use for communications to take place. This communication is a key part of Availability. Data Interoperability is nothing if the data is not transmitted and stored securely, which indicates Integrity. What good is data if you cannot trust it? Data Interoperability does not necessarily require Confidentiality, although most data valuable enough to require data interoperability is not for prying eyes.

Data terms related to cybersecurity discussed include:

- Certificate Revocation List (CRL)
- Certificates
- Checksum and CRC
- Credentials
- cryptographic certificates
- Cryptographic Hash
- Cryptographic Keys
- Digital Signatures
- Hash
- Key Data Storage
- Passwords
- Preshared Key
- Signatures

4.2.3.6 Data about Timing and Timestamps

Many data require time stamps for when the data were created. For example, a sensor of a moving part in a motor might need to take data at a regular rate, and each data point would need a time stamp with enough accuracy with respect to the appropriate reference time scale to make the data useful. There are several issues here:

1. The local oscillator determines the time-stamping rate, in short term. In longer term, this oscillator may be locked to an external reference. It may be, however, that the time transfer through the network is insufficiently accurate to meet the needs of the local CPS node. A better oscillator can be both more accurate and stable. With an external reference, requisite stability up to the loop time constant is the requirement. Without a sufficiently accurate external reference, the local oscillator needs both accuracy and stability; note that these two are rather independent requirements. A significant trade-off here is that the better the oscillator generally the more size, weight, power, and cost it might demand.
2. The time-stamping rate largely determines the quantization error of the time-stamp. This, along with clock stability, is the source of jitter on the measurement times and other stochastic noise.
3. A stable but inaccurate time-stamping oscillator produces a deterministic offset in the data collection rate. If this can be measured it can be removed.
4. Traceability of the oscillator is a function of the time-transfer accuracy from the reference timescale. If data need to be correlated between nodes, a common reference

timescale is required. Often this is best done using an international timescale such as UTC or TAI.

5. Missing data need to be accounted for. If the user of the data is expecting data at a certain rate, there needs to be a method of acknowledging missing data, for the user to maintain the correct data rate.
6. Formats used to write or create timestamps can be serious issues. Consider in a networked system of possibly dissimilar nodes, the potential for different time stamp formats (e.g. 48 bits vs. 64 bits or the order of significance reversed, high to low versus low to high) as well as varying granularity of time stamp clocks. One system might generate timestamps at 40 MHz and another at 250 MHz. The period of the slower clock allows for greater local oscillator error influence.
7. Uniformity in any networked system of shared time stamps is mandatory.

These issues suggest the need for the following parameters for data timestamps:

1. The nominal data rate
2. An indication if data are missing at a regular measurement time
3. Enough significant digits in the data and timestamp to meet requirements
4. The stochastic uncertainty of the timestamps
5. The deterministic uncertainty of the timestamps
6. The traceability accuracy and reference timescale
7. A common timestamp format, such as ISO 8601.
8. Perhaps a period of validity and/or expiration date of the data

Timing data can contribute to security and monitoring issues, for example, knowing that a user cannot be in two places at the same time. Accurate time-stamping can contribute to root-cause-analysis of when a failure or incursion happened where in a network.

4.2.3.7 Safety and Configuration Assurance

Design and implementation assurance is an important part of CPS with regard to safety, reliability and resilience. It is essential that any given CPS component can be verified, to some level certainty, that it conforms to required levels of safety assurance.

There are two key dimensions to this that pertain to data interoperability:

1. Determining that the software running on the CPS device is indeed that which is believed to be running, and,

2. Determining that the running configuration is as established by authorized configuration management software, policies, and procedures

Software images are typically verified through secure hash checksums that ensure that the code in firmware is as expected by design.

Changes to CPS device configuration can be managed through event recording of changes and the maintenance of a change history.

ANSI C12.19 [126] is a standard used throughout North America for automated meter reading. This standard tackled these issues from the perspective of Data Interoperability with a function they called “Event Logger”. The principle used is that configuration changes that can be made to what is essentially the cash register of the utility must be tracked and auditable. Further, since communications can be intermittent, and changes can be imparted locally or remotely to such devices, a persistent record of some depth must reside within the CPS device itself, with a larger less limited record “spooled up” to the owner – typically the utility. A series of secure hashes and time stamped event records are performed that guarantee that any current state of the CPS device can be recreated by executing the logged sequence of changes and only in the order that they were recorded.

Many CPS devices provide for configuration management through communications interfaces. Inadvertent, incorrect, or malicious changes can cause havoc in a CPS, depending on the role of such a device in the system. Therefore, best practices on the version and state control need to be specified for many components of CPS.

4.3 Timing Aspect [Timing Subgroup]

4.3.1 Introduction

4.3.1.1 Overview

In this timing section, we have the following chapters:

- This introduction discusses fundamental concepts needed for understanding the chapters that follow.
- Chapter 4.3.2 presents the current status of and needs for time-awareness in system elements of a CPS.
- Chapter 4.3.3 discusses timing and latency in CPS. Latency is a core concept for timing in CPS. Latency is a critical issue in all CPS but is especially critical where control systems span several nodes with significant spatial separation, and especially in Systems of Systems or any systems that include cloud computing or virtualization technologies in the control system. Also, the temporal relationships between acquired data (e.g. simultaneity) are of paramount importance. The challenges of predictability in software are increased by the non-determinism of the layers of software managing data-transfer and non-determinism of the network connecting these nodes.
- Chapter 4.3.4 discusses special security issues that arise with timing. General trust disciplines relating to CPS include security, resilience, safety, reliability, and privacy. Timing plays a key role in many of these and thus the provision of secure timing raises specific challenges relating to security, and resilience. Security of a timing signal requires security of both the physical signal and of the data associated with the signal. Security of the data in a timing signal is similar to other cybersecurity problems. Security of the physical signal brings in a number of aspects unique to timing. The user

is typically remote from the source of the timing signal representing the particular system time-scale. For security, the user needs to know both that the physical signal came from the correct source, and that the transmission delay has not been tampered with. In addition to these two aspects, denial-of-service can be created for timing signals in a number of ways.

4.3.1.2 Core Concepts

There are many aspects to timing, but fundamentally all timing includes a physical signal. The physical signal may be accompanied by data, which describes it or is meant to be used with the physical signal. The physical nature of timing is at odds with the way data systems work, leading to core difficulties in Cyber-Physical Systems (CPS). Data systems, computer hardware, software, and networking have been optimized by abstracting away the timing properties of the physical layer. These systems all isolate timing processes, allowing the data to be processed with maximum efficiency due in part to asynchrony. However, coordination of processes, time-stamping of events, latency measurement and real-time control are enabled and enhanced by a strong sense of timing. CPS involve a marriage of the cyber and the physical: a marriage of data networking and processing systems with systems that live within the laws of physics. Generally speaking, CPS currently overcome this fundamental conflict of modern system design by using dedicated hardware and customized software for timing-critical systems. Things that require strong temporal determinism are processed as much as possible with systems that do little or no data processing. However, in many cases CPS must include significant data processing. Here, both software and hardware must be calibrated to ensure agreement with timing specifications, and this calibration is done for the specific chosen hardware and software. Any changes or upgrades to hardware or software can trigger a re-calibration of the entire system.

This document, the timing framework elements of cyber-physical systems, discusses the current status of such systems and points out problems and new directions that are currently in development. A later document will more fully show a roadmap for future timing systems.

4.3.1.3 Types of Timing and Timing Requirements

There are three different types of timing signals for synchronization: frequency, phase, and time. Accurate frequency can be supplied by an individual clock, a cesium standard, though practicality drives the use of oscillators that require calibration and active reference signals. By contrast, phase and time synchronization *always* require transport of signals and perhaps data. Unlike the transfer of data, the transfer of time and phase requires compensation for the transmission delay of these timing signals to the required synchronization accuracy. For example, GPS provides positioning by sending synchronized time signals from known locations in space. The transmission delay is of order 70 ms. To provide ranging accurate to 1 m, the true delay must be removed to better than 3 ns, a factor of about 1 part in 20 million.

Data often accompany physical timing signals, though phase synchronization may not need it. The simplest timing data are for time, sometimes called “time-of-day,” where the signal indicates when the time information is correct, but the actual date and time-of-day of that time signal must be transferred as data. In this case, the time signal is sometimes called the "on-time marker." The time *data* can be transferred with significant noise and latency, as long as when it arrives it is clear which on-time marker the data refer to. Depending on the applications, many other data may be associated with timing signals. For example, a quality level of the source clock is often required with timing data.

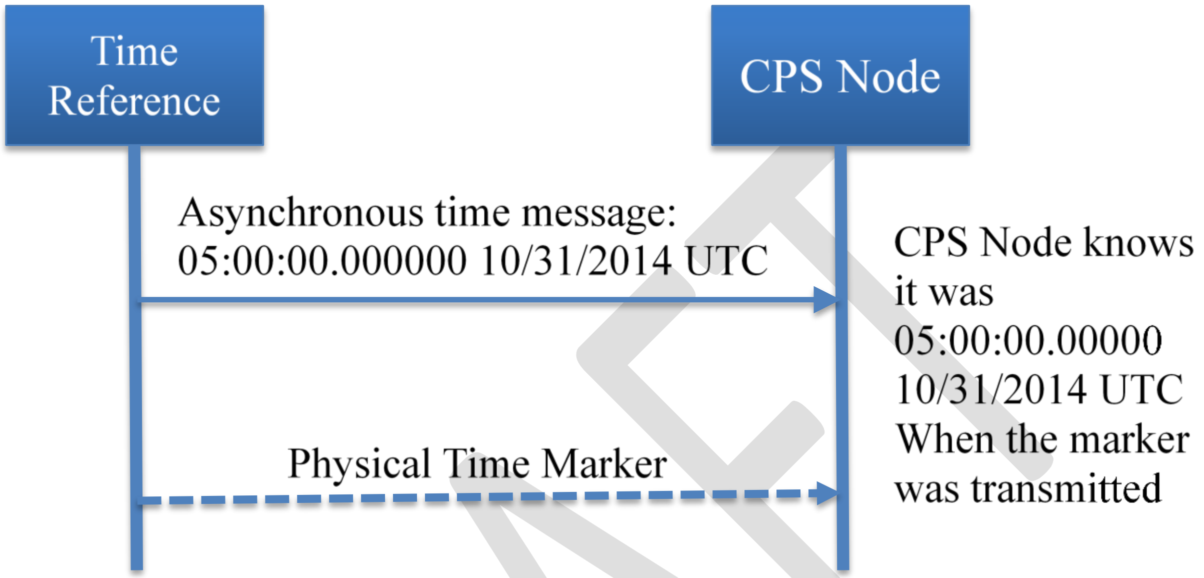

Time Reference
CPS Node
Asynchronous time message:
05:00:00.000000 10/31/2014 UTC
Physical Time Marker
CPS Node knows it was 05:00:00.00000 10/31/2014 UTC When the marker was transmitted

Figure 18: On Time Marker

Figure 18 is an illustration of the relationship between the physical time signal and associated data, an asynchronous time message, in this case. Note that the time of arrival of the marker is the transmission time plus the delay. The CPS node will need to either know or cancel the transmission delay commensurate with its time accuracy requirements.

Synchronization through networks will generally involve the transmission of such time markers and data using a two-way time protocol to cancel the delay through the network. Two-way time transfer is discussed in the Timing Framework Annex Section 1.1 [144]. Common protocols for this are the Network Time Protocol (NTP) [172] and the Precise Time Protocol (PTP) [149] [150] [151] [152]. Other protocols are discussed later, in section 4.3.2. Systems whose timing requirements are coarse enough that the time-transfer delay is not important will not need to cancel or remove the transmission delay.

A specific set of CPS nodes will be synchronized against a single reference timescale forming a CPS synchronization domain, the CPS domains as described in section 4.3.3. Section 4.3.3 also discusses how timescales will need to be synchronized across domains if they need to coordinate functions such as timestamps of data or control. This will apply to all forms of synchronization depending on what is needed for the specific CPS function: time, phase, or frequency synchronization. Synchronization across domains can require more care if they are connected through a Cloud or across a network with virtualization. The impact of new networking paradigms such as Software Defined Networking (SDN) on timing performance needs to be carefully considered as does the role of Network Function Virtualization (NFV), as discussed in section 4.3.2.4.

CPS timing requirements can be specified in terms of the time interval between significant events. The concept of a time interval specification implies that the system supports a time-scale against which intervals can be measured (time-scale is defined in [141]). A time-scale is characterized by two features: the epoch which marks the origin, i.e. time zero, and the rate at which time advances, typically the definition of the second.

The concept of a "second" is defined in the International System of Units (Système International d'unités, SI) developed and maintained by the International Bureau of Weights and Measures (Bureau International des Poids et Mesures, BIPM), in terms of energy levels of Cesium atoms. Thus, a clock is accurate (in frequency) to the extent its rate agrees with the definition of the second. The clock is accurate as a wall-clock if it is traceable to UTC or TAI. TAI is the time-scale called International Atomic Time (Temps Atomique International), which is generated by the BIPM with the rate that best realizes the SI second, and the time origin determined by the transition to atomic time from astronomical time in 1958. UTC is considered "discontinuous" due to leap second adjustments. These are inserted into UTC to keep it within 0.9 seconds of UT1, the time scale linked with the Earth time. Note that any real-time UTC or TAI signal is only a prediction of the exact value, since UTC and TAI are post-processed time scales [142]. The following table identifies some of the time-scales in use and the choice of time origin (epoch).

In many CPS systems the time-scale need only be self-consistent, with no requirement to agree with time-scales external to the system. However, due to the inherent connectedness of the Internet of Things (IoT), some level of accuracy of time that is traceable (traceable is defined in [141]) to an international scale such as Universal Time Coordinated (UTC) [142] will often be available, though perhaps not at the accuracy the system requires. Thus, in many systems, the precision timing of the epoch is an application specific event, e.g. when the power was turned on, and the rate is typically a count of the oscillations of a local oscillator in one of the nodes. In other systems the time-scale is required to agree with an internationally defined time-scale, e.g. UTC or TAI [142]. In this case the rate must be the SI second. The Timing Framework Annex Section 1.1 [144] contains a detailed discussion of time-scale issues and metrics.

Equally important aspects of CPS timing are predictability and determinism. There are two aspects to determinism. The first, and the typical computer science meaning, is that a system is deterministic if for the same set of input values and system state (ignoring timing) the resulting output values and system state is always the same. Thus for example 2+2 is *always* 4 and the command "initialize" *always* puts the system into a defined initial state. This is clearly a requisite property for CPS systems. However, CPS systems often require *temporal determinism*, i.e. identical or at least very similar timing behavior. Due to inherent variability of execution time on modern high-performance architectures, system significant time intervals can only be identical (deterministic) if identical input, identical initial architectural state, and the absence of external interference can be guaranteed. Issues of temporal determinism are discussed in

Chapter 4.3.2. Throughout this document the term determinism refers to temporal determinism.

Timing predictability means that the timing behavior can be predicted within appropriate parameters that a specific system requires. This is discussed in more detail in the Timing Framework Annex Section 1.1 [144]. To the extent the timing is predictable, it can be predicted at any future time, given the initial values of input and state. The BIPM has developed a standard method for determining uncertainty, by breaking it into type A, typically the statistical uncertainty, and type B, typically a deterministic uncertainty, or an uncertainty of how large a bias there may be in the data [142]. Thus, uncertainty is in a sense the opposite of accuracy, i.e. uncertainty is the amount of inaccuracy. An example of this is in the IEEE 1588 protocol, or PTP. Short term noise is caused by packet delay variation (PDV) also called jitter. This would be a type A uncertainty, i.e. it is a statistical uncertainty. Asymmetry in the delay between the two directions of timing packet transfer causes a constant time error in the resultant time transfer. This would be a type B error; it cannot be seen in the measurements, even with a very small standard deviation in the stochastic effects. Thus, an estimate of the magnitude of the asymmetry would be part of the type B uncertainty. Timing uncertainty is discussed in detail in the Timing Framework Annex Section 1.1 [144].

4.3.1.4 Benefits Introduced from Precise Timing

Timing is inherent in CPS. Precise timing capability in a CPS can enable better control, more robust correlation of acquired data, and permit CPS that have large spatial extent and/or higher degrees of complexity.

Perhaps more significantly, the increasing use of explicit time in networks, and the nodes themselves, holds the possibility of designing CPS that are correct by construction. In the future the presence of appropriate support for explicit time will lead to new and more robust designs for the applications themselves. Both these points are discussed in section 2.

Precision timing may mean very many different things. Besides the different types of timing, frequency, phase, and time, there are many orders of magnitude of variation in timing requirements. These are illustrated in the Timing Framework Annex Section 1.4 [144].

In the absence of a CPS time-aware architecture that infuses appropriate timing into the components on which applications are built, today's CPS are increasingly being rolled out, complete with many limitations due to the lack of availability of precise time. Emerging CPS application domains include Smart Systems (Grid, Cities, Buildings, Transportation Systems), Location-based systems, Medical Devices, Environmental Monitoring, and Entertainment.

The need urgently exists to revisit conventional Information and Communications Technology paradigms so they maintain appropriate time-awareness, such that next generation CPS will not be held back by design and engineering constraints. This will then signal an era whereby CPS will have the potential to transform our lives by facilitating huge performance leaps in existing application domains and setting a foundation block for as-of-yet unheard of domains.

4.3.2 Time-Awareness in CPS

This section examines the components of a Cyber Physical System (CPS) from the perspective of the presence or absence of explicit time in the models used to describe, analyze, and design CPS and in the actual operation of the components.

Such systems take many forms and have diverse timing requirements as indicated in the Timing Framework Annex Section 1.4 [144]. Timing requirements are generally expressed as constraints on the time intervals (TI) between pairs of system significant events. For example the TI between the acquisition of a sensor reading and the time at which an actuator is set as a result of that reading may be *specified* to be 100 µs±1µs. Similarly a bound may be required on the TI, i.e. the *latency*, between when a sensor measurement event actually occurred and the time at which the data was made available to the CPS. Likewise the accuracy of event timestamps is a constraint on a TI, in this case between the actual time of the event and the value of the timestamp.

Constraints on TIs can be categorized based on their degree of time-awareness in terms of *bounded TIs*, *deterministic TIs*, and *accurate TIs.* Bounded TIs are required for CPS whose timing behavior is based on deadlines. Deterministic TIs (meaning temporal determinism as discussed in section 4.3.1) specify the interval between two significant events, but allow for a specified deviation. Deterministic TIs are necessary for CPS where repeatable and precise timing relative to the system time-scale is required. Accurate TIs are deterministic TIs where the system time-scale is TAI or UTC. Accurate TIs are useful for coordinating actions in CPS of large spatial extent, where accessing a traceable timescale is often more convenient than propagating a self-consistent and system-specific one. Accurate TIs are sometimes required due to legal or regulatory requirements. Details on these constraints are further addressed in The Timing Framework Annex Section 1.1 [144].

4.3.2.1 Bounded TI

A bounded TI is always less than some stated value Δ_{MAX} (and sometimes always greater than some stated value Δ_{MIN}), i.e. $\Delta_{MIN} < TI < \Delta_{MAX}$. To be useful $\Delta_{MAX} < \Delta_{REQ}$, where Δ_{REQ} is an application specific requirement on the bound.

Bounded TIs are the basis for operation in deadline oriented CPS. For example in an airplane the TI between the pilot's signal that the landing gear should be lowered and the gear being in place and locked must have a predictable bound but need not be deterministic. Failure occurs if the bound is exceeded but there are no issues if the operation completes earlier.

Similarly in a power plant the TI between a loss of load and shutting off the energy input to the generator turbine must have a predictable bound to prevent damage to turbines or other equipment that must dissipate the energy. In all such cases Δ_{MAX} must be small enough to meet the application requirements. The verification of such bounds is a major task in designing and certifying CPS in many industrial and safety-critical applications.

4.3.2.2 Deterministic TI

In contrast to a bounded TI, a deterministic TI is always within some stated error ε of the application specification Δ_{REQ} on the TI, i.e. $|TI- \Delta_{REQ}| \leq \varepsilon$. In most CPS the attributes Δ_{REQ} and ε are specified in terms of a system-defined time-scale rather than on international standards.

For example smart highway designs require that cars be able to determine the distance to the car in front. Acoustic or electromagnetic ranging can be used to determine the TI between the transmitted signal and the signal returned from the other car. For acoustic-based ranging and assuming the allowed error is one foot a reasonable value for ε is one millisecond. That is the difference between the actual and the measured time interval is the error of 1 foot divided by the speed of sound. If electromagnetic ranging is used a reasonable value for ε is one nanosecond. Here ε is the required precision of the measurement, i.e. the CPS must be able to measure the ranging time with a resolution of ε. However the accuracy requirement is much less severe, i.e. the second defined by the system time-scale can differ from the SI second. In this case 0.1%, e.g. allowing an error of 1 foot in 1000 feet, is probably more than adequate and would easily be met by a time-scale governed by a quartz crystal oscillator with no need for calibration against international standards.

Engine control units are another example where the TIs must be deterministic rather than simply bounded. The intervals between fuel injections must have a precise timing relationship to the sensed position of the shaft. Again the time-scale is local, as consistency within the engine is required, but it is not required for function that timing be based on the SI second.

4.3.2.3 Accurate TI

An accurate TI is a deterministic TI but with the added requirement that the time-scale be traceable to international standards. These are discussed in section 4.3.1.3. Accurate TIs based on a time-scale traceable to international standards are often needed to meet regulatory or legal requirements. For example it is quite common in the medical industry for CPS specifications, including time, be certified based on metrics defined by international standards.

However the use of accurate as opposed to just deterministic TIs often provides a simpler and more robust solution for a CPS. This is particularly true where the CPS is sufficiently large spatially that it is difficult to establish a deterministic time-scale. For example in North America, power systems often need to be coordinated over distances of thousands of miles. Synchrophasor technology is likely to be a critical part of the smart grid and will need to function over these distances. Synchrophasor technology requires the determination of the phase angles between the voltage waveforms at various parts of the grid.

The only realistic way this can be done on a continental scale is to make local measurement of phase with respect to a 60 (or 50) Hz cosine waveform synchronous with TAI. In principle one could establish a consistent continental time-scale by distributing time, frequency and phase from a central location but the effort would far exceed that of simply using GPS. Power systems and telecommunication systems are similar in that both are continental scale and both are implemented by independent companies rather than by a monolithic organization. So for example in North America prior to the breakup of the Bell System a continental frequency standard was established by Bell based on distribution from a central location. Consistent frequency not necessarily based on the SI second was all that was required. Since the breakup the only practical way to achieve the continental agreement on frequency is for each of the operating companies to implement their frequency distribution based on the SI second, again typically relying on GPS. More recent protocols require time as well as frequency agreement which has led the ITU-T to publish standards on the use of protocols such as IEEE 1588 in combination with GPS for this purpose.

4.3.2.4 CPS Nodes

A CPS node typically samples the physical world via one or more sensors, performs some computation based on the sensed values, often along with data obtained from other CPS nodes, possibly including the time of sensing, and delivers the computed results either to another CPS node or as an instruction to an actuator. In the case of a bounded TI there need not be any explicit reference to the time of a time-scale, while in the case of an accurate TI, the time is not only explicit but traceable to international standards.

To dispel any doubt about the central role that time awareness plays in CPS one need only look at the measures currently used in industry to achieve such awareness: time triggered architectures [145], TDMA network protocols, and architectures such as PROFINET [146], IRIG-B [147], GNSS [148], IEEE 1588 [149] [150] [151] [152], FPGAs for critical local timing control, and finally analysis and reasoning techniques to determine code execution bounds, i.e. worst case execution time (WCET) [153] [154] [155] [156], and the correctness of programs in meeting timing requirements [157] [158]. Conspicuously absent is timing-correctness by design, a term we discuss later in this section.

Next consider how the architecture of typical CPS devices supports, or fails to support, timing. Figure 19 is a block diagram of a typical networked node of a CPS. Note that a CPS need not be networked, but may consist of one or more autonomous nodes. At the other end of the spectrum, very large scale CPS may form Systems of Systems which introduce further challenges. Furthermore, many CPS nodes have multiple network interfaces to permit daisy-chained or more complex topologies.

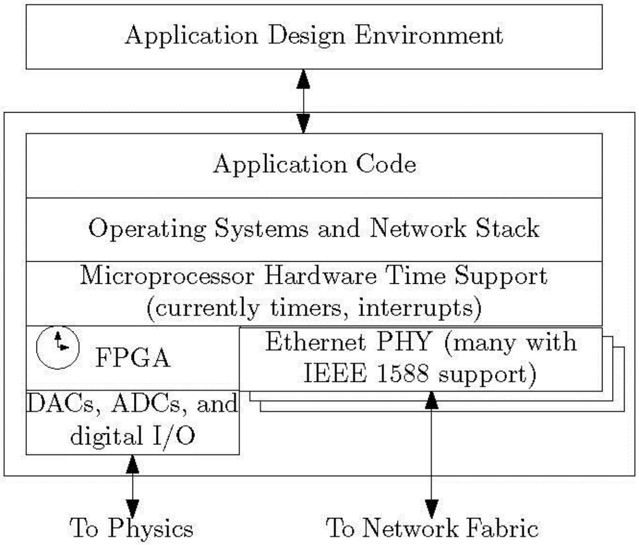

Application Design Environment
Application Code
Operating Systems and Network Stack
Microprocessor Hardware Time Support (currently timers, interrupts)
FPGA
Ethernet PHY (many with IEEE 1588 support)
DACs, ADCs, and digital I/O
To Physics
To Network Fabric

Figure 19: Architecture of a CPS Node and Environment

Consider the "P" or "physics" part of a CPS node and here we include physical things such as biological, electrical, thermodynamic and chemical processes. For the most part CPS physics models for natural and many man-made target devices include time explicitly, e.g. Maxwell's and Newton's equations, the diffusions equation, etc. However there are definitely targets of interest where time is not explicit in our physics models, e.g. radioactivity, Ethernet network traffic, etc. Here our models are more likely to be state or statistical models.

Considering the CPS microprocessor of Figure 19, timers and interrupts are the principal explicit means for supporting time constraints in modern microprocessors. With very few exceptions, it is not possible to specify or control the actual execution time of a code segment or the time to react to an interrupt. Furthermore, these times are often not even repeatable given the same inputs and code due to process scheduling, memory caches, pipelining, speculative execution and similar features that have been introduced to increase the performance of modern microprocessors. In effect, modern general purpose microprocessor operation is no longer time-aware; execution time is at best construed as a performance metric rather than as a correctness criterion. The result is that operating systems and commonly used programming languages also lack time-awareness. It is clear that modern microprocessors cannot by themselves support deterministic or accurate TI requirements [159].

Under some restrictions, particularly on processors with no operating system or operating systems with non-preemptive scheduling, it is possible, albeit difficult, to analyze code execution timing and predict safe upper bounds [155]. Many safety-critical systems are based on these timing analysis techniques. For example the aviation/aerospace industry uses these techniques, but only uses qualified and certified processors and in applications that are deadline based, or use timing support hardware that can add determinism.

Time-triggered architectures illustrate how the separation of timing at the boundary between the cyber and physical parts of a CPS allow deterministic, or if needed, accurate timing at this interface, while requiring only bounded TIs on the computation phase [145]. This is a general principle not fully explored in today's design practices, CPS architectures, and applications.

Next consider the network interface. With the exception of TDMA protocols, network latency between two microprocessors is as unpredictable as code execution within the microprocessors. A lower bound can be set on latency but that is the extent of network time-awareness.

Where explicit and accurate time constraints are required within a CPS node, timing constraints are typically implemented in FPGAs, ASICs, or custom hardware logic where time is explicit, as opposed to depending on microprocessor code execution timing. If the CPS is distributed it is possible to order events by means of messages passing over the network, but the enforcement of accurate timing requirements requires system-wide explicit time, i.e. a clock synchronized to its peers. In some cases frequency and (relative) phase will suffice, e.g. ensuring that all converters between analog and digital (and vice versa) in a system use a common sampling rate, and/or a common sampling phase/time. In safety-critical systems, system-wide time is used to establish time-triggered architectures where applicable sampling, code execution, actuation and network traffic are all based on schedules, generally periodic, enforced by special hardware, ASIC or FPGA logic based on the node's synchronized clock.

Synchronized clocks are readily, but not universally, implemented in a CPS node. The network time protocol, NTP, can be made available at the application level but this is of little help for accurate timing at the interface to "physics". As shown in Figure 19, newer physical layer network interface chips, e.g. Ethernet PHYs, typically contain hardware support for implementing synchronized clocks using protocols such as IEEE 1588, enabling the establishment of system-wide time to levels of accuracy and stability appropriate to the majority of CPS applications [160]. GNSS, e.g. GPS, technology is often used to provide a source of time for synchronizing clocks in a distributed CPS. However to be truly useful, the time from the clock needs to be a key and explicit feature of timing support in microprocessors. This is not the case at present. At a minimum, standardized interfaces for time-sensitive operations should be inherent in the microprocessor architecture itself.

If explicit time from synchronized clocks was inherent in microprocessor timing support, it would be possible to conceive of operating systems and languages that could enforce designers' timing requirements to a high degree of accuracy and determinism. It should be noted that if time were made explicit throughout the CPS along the lines outlined, the way designers conceive applications would change. The best example is the Google Spanner project [161], a world-wide database where they replace the usual message passing logic for commits with logic based on reasoning about time stamps associated with transactions. The time stamps are generated by a world-wide explicit time base implemented by synchronized clocks. While not a CPS, Spanner does illustrate the change in design philosophy possible given the presence of system-wide explicit time.

"Time correctness by design" includes this concept of: designers including accurate timing in designs, independent of hardware [162] [163]. Designers need to be able to specify timing in a CPS as an abstraction, much as most modern systems are designed as abstractions, without reference to specific hardware. This is necessary to allow a design to persist through upgrades in the hardware and software. There is a lot of work to be done to realize time correctness by design in full. In its ideal realization, a designer could include timing as an abstraction in a GUI design system. Upon choosing the target hardware, the system determines if that hardware can support the timing, and if so, generates the code and implementations to support the design.

Finally some recommendations for the design of future CPS systems:

- Incorporate explicit time at the lower levels, e.g. network and hardware, of the systems.
- As they become available, use microprocessors and other COTS hardware that provide explicit support for time.
- Use networks with on-path support for clock synchronization. There are numerous examples of bridges and routers for Ethernet that incorporate such support.
- Explore ways in which the use of explicit time, particularly in distributed systems, can be used to improve application designs.

From an architectural viewpoint, CPS nodes rarely exist in isolation and will typically form part of large scale, geographically distributed systems. The concept of Systems of Systems introduced in the Reference Architecture section illustrates the potential scalability of CPS. In such cases Cloud Computing will play increasingly important roles in CPS. The networks that support such systems will also see adoption of Software Defined Networking (SDN) and Network Function Virtualization (NFV) technologies. This raises a range of timing-related challenges:

- Cloud – The role of the Cloud in CPS will dictate the degree of time-awareness that is necessary. At a minimum, data analytics will require simultaneity as described earlier, and a mapping from local to global timescales. If the Cloud plays a more time-sensitive role, then requirements similar to those discussed above re execution time need to be met. Such challenges are made more difficult by virtualization which is a foundation block of Cloud Computing.
- Network- The impact of Software Defined Networking (SDN) on timing performance needs to be carefully considered as does the role of Network Function Virtualization (NFV). While both technologies may reduce complexity and cost, and increase flexibility, their abstracted architectures may degrade timing performance.

Finally, CPS exist to fulfill Business needs, and as shown in the Reference Architecture, the timing requirements at this ‘layer’ need to be met.

4.3.3 Time and Latency in CPS

This section addresses the use of time to provide bounded latency in a Cyber-Physical System. The aim is to provide reference architectures/frameworks that enable building time-aware Cyber-Physical Systems to solve control and measurement applications.

Given the diversity in CPS applications and scale it is not surprising that temporal considerations vary considerably over the range. For example, in small closed systems such as a packaging machine, the primary temporal concern is that all components respect a self-consistent timing design. In such systems, networking temporal considerations, e.g. design of a TDMA scheme, are part of the design itself. However in large scale, and more critically in environments characterized as "System of Systems", timing issues are more difficult, as outlined above. For example "smart highways" will involve many different systems, some in the vehicle, some in the infrastructure, some in a traffic management center, etc. Each will have its own temporal requirements which must be met while sharing network bandwidth and in some cases computation bandwidth on servers. Today the technology for managing the timing in such systems is still a work in progress. The remainder of this section discusses both the general issues as well as some of the current thinking on these issues. Some of these can be applied to smaller systems. There is no doubt that the work on larger systems will result in improvements, e.g. in time-sensitive network technology, that will make small system temporal design much easier and more robust.

CPSs are used in both control and measurement applications. The requirement of bounded latency is obvious in control systems where latency from when a physical input is read to when a physical output is written has to be proven by timing and schedulability analysis. In large-scale control systems this requirement becomes even more challenging since the input, computation and output may be occurring on different nodes that are spatially distributed. The challenges of predictability in software are added to by the non-determinism provided by layers of software managing data-transfer on the network connecting these nodes. As described above, the impact on timing of Cloud Computing and Networking concepts such as SDN and NFV need to be carefully considered.

In CPS-based measurement systems, the deterministic relationship between acquired data (e.g. simultaneity) is of paramount importance. However, what is typically overlooked is the efficiency and complexity of transferring the acquired data from thousands of nodes to one or more aggregating units, where analytics or logging is being performed. Misaligned data can result is faulty conclusions. In many CPS-based applications, the data measurements are used for asset or structural-health monitoring and in many cases a timely response based on real-time analytics is required. Time, when applied to data-transfer can enable bandwidth reservation in networks used in these measurement applications, thereby enabling faster analytics, a smaller memory footprint, and increased efficiency in data-reduction techniques (for logging). Moreover, bounded latency is extremely useful in distributing triggers to multiple nodes inside a CPS.

Similar to CPUs, computer networking has traditionally been optimized for "best effort delivery", and that has worked extremely well in the past and will continue to do so in the future for many uses. However, it is not good enough when the same networking technology is used for time-sensitive applications that are served by CPSs. Time-based CPSs can be built using standard Ethernet technologies to enable seamless integration with the Internet. "Time-Awareness" in standard Ethernet is paving the way to enable time-sensitive (bounded latency) traffic to coexist on the same network as traditional best-effort (no latency guarantees) traffic. There are several standards being developed in the IEEE and other SDOs for this purpose.

A time-aware CPS should guarantee bounds on latency of data delivery and guarantees on synchronization accuracy as it applies to timing correlation of physical I/O. To build such large-scale systems with these guarantees the following two concepts of *CPS Domain* and *CPS Network Manager* are defined.

CPS Domain: A CPS domain is a logical group of CPS nodes and bridges which form a network with their own timing master. The master may synchronize to a globally traceable time source (e.g. GPS). Each CPS domain has its own primary (or self-consistent as described earlier) time-scale. This time-scale provides a strong monotonically increasing clock to applications for performing input/output functions and time-based scheduling. The timing master of a CPS domain should not produce a discontinuity of time once time-sensitive data transfer within the domain has commenced, even if the master loses connectivity to its global source (e.g. GPS) sporadically.

If a global traceable time is required inside a CPS node, then the node can implement a second time-scale called the *Global Traceable Time-Scale*. This time-scale can be managed independent of the CPSs primary Time-scale. To correlate the CPS's primary time-scale to the Global Traceable Time-Scale, the offset of the primary time-scale from the Global Traceable Time-Scale can be maintained at all times by the CPS node. The Global Traceable Time-Scale can be used to correlate CPS Time-Scales from multiple CPS domains.

Many CPS will be small enough that they don't need an external time-scale and the primary time-scale will suffice. However with many things becoming networked, some level of traceable timing may be available, though perhaps not at the needed precision.

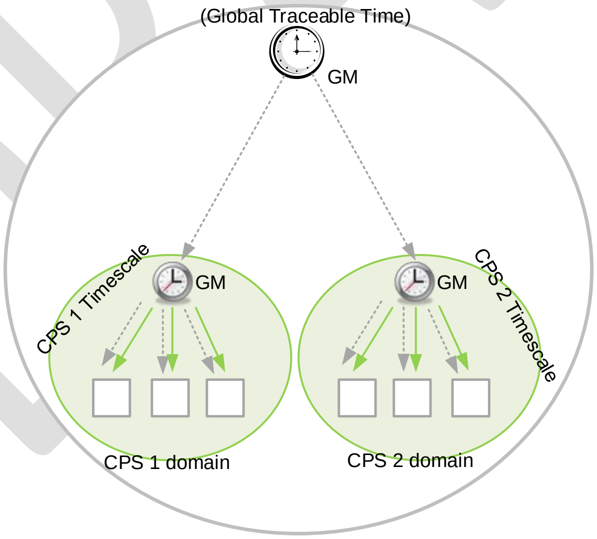


Figure 20: Domains and Multiple Time-scales in Time-aware CPSs[11]

[11] **Source: Sundeep Chandhoke, National Instruments**

CPS Network Manager (CNM): A work-station or CPS node connected to a CPS domain that manages and monitors the state and configuration of all CPS nodes in one or more CPS domains, or in a more scalable System of Systems. The CPS Network Manager (CNM) interfaces with a schedule generator and path computation engine to generate the schedule for the CPS. This may be done by interfacing with a centralized network controller. For performance, reliability and/or scalability reasons, functions of a CPS Network Manger may be distributed among multiple devices.

The functions of a CNM vary depending on the size of the system. These functions include:

- Control and manage the state of all CPS nodes in a CPS domain.
- Coordinate with a centralized network controller to configure bridges in a CPS domain.
- Configure transmission schedules on CPS nodes
- Monitor the health of the CPS domain (for handling errors, changing schedules and bringing new CPS nodes online, etc.).
- Configure application and I/O timing on each CPS node
- Configure any static timing requirements for time-based synchronization

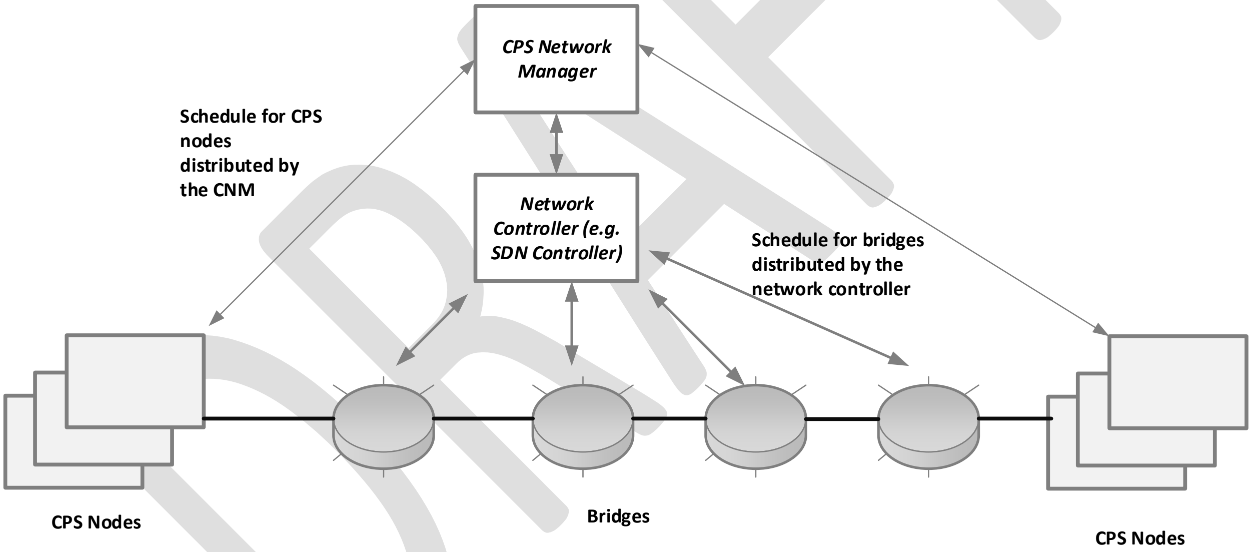

CPS Network Manager
Schedule for CPS nodes distributed by the CNM
Network Controller (e.g. SDN Controller)
Schedule for bridges distributed by the network controller
CPS Nodes
Bridges
CPS Nodes

Figure 21: CPS Network Manager configuring a CPS [12]

Either the CNM or the centralized network controller has to gather performance metrics and determine the topology of CPS nodes in a CPS domain in order to create a schedule. The relevant performance metrics include Bridge Delays, Propagation Delays, and Forwarding/Transmission delays. There are multiple ways to detect topology. For example, one

[12] Source: Sundeep Chandhoke, National Instruments

approach to Software Defined Networking (SDN) defines a “Packet-In” “Packet-Out” protocol which uses Openflow [164] with Link Layer Discovery Protocol (LLDP) [165]. Some other protocols like PROFINET [166] use Simple Network Management Protocol (SNMP) [167] along with LLDP. The Centralized Network Manager computes the topology for the CPS domain using these mechanisms, and determines the bandwidth requirements for each time-sensitive stream based on application requirements. The bandwidth can be specified by the period and the size of the frame. Optionally the application can also specify a range <min, max> for the offset from start of a period. This information is provided to the Centralized Network Controller. The Centralized Network Controller computes the path for the streams and gathers performance metrics for the stream (latency through the path and through the bridges). This information is then used to compute the schedule for the transmission time of each time-sensitive stream and the bridge shaper/gate events to ensure that each time-sensitive stream has guaranteed latency through each bridge. Additionally, queues in bridges are reserved for each stream to guarantee bandwidth for zero congestion loss. It should be noted that schedule computation is the subject of continuing research as the problem becomes intractable for large systems.

It should also be noted that there is considerable activity in the IEEE 802.1 and other standards communities in providing additional tools for controlling network temporal properties, see the Timing Framework Annex Section 1.2 [144] for additional details.

An illustration of a possible device model for a time-aware CPS node is shown in Figure 22, below.

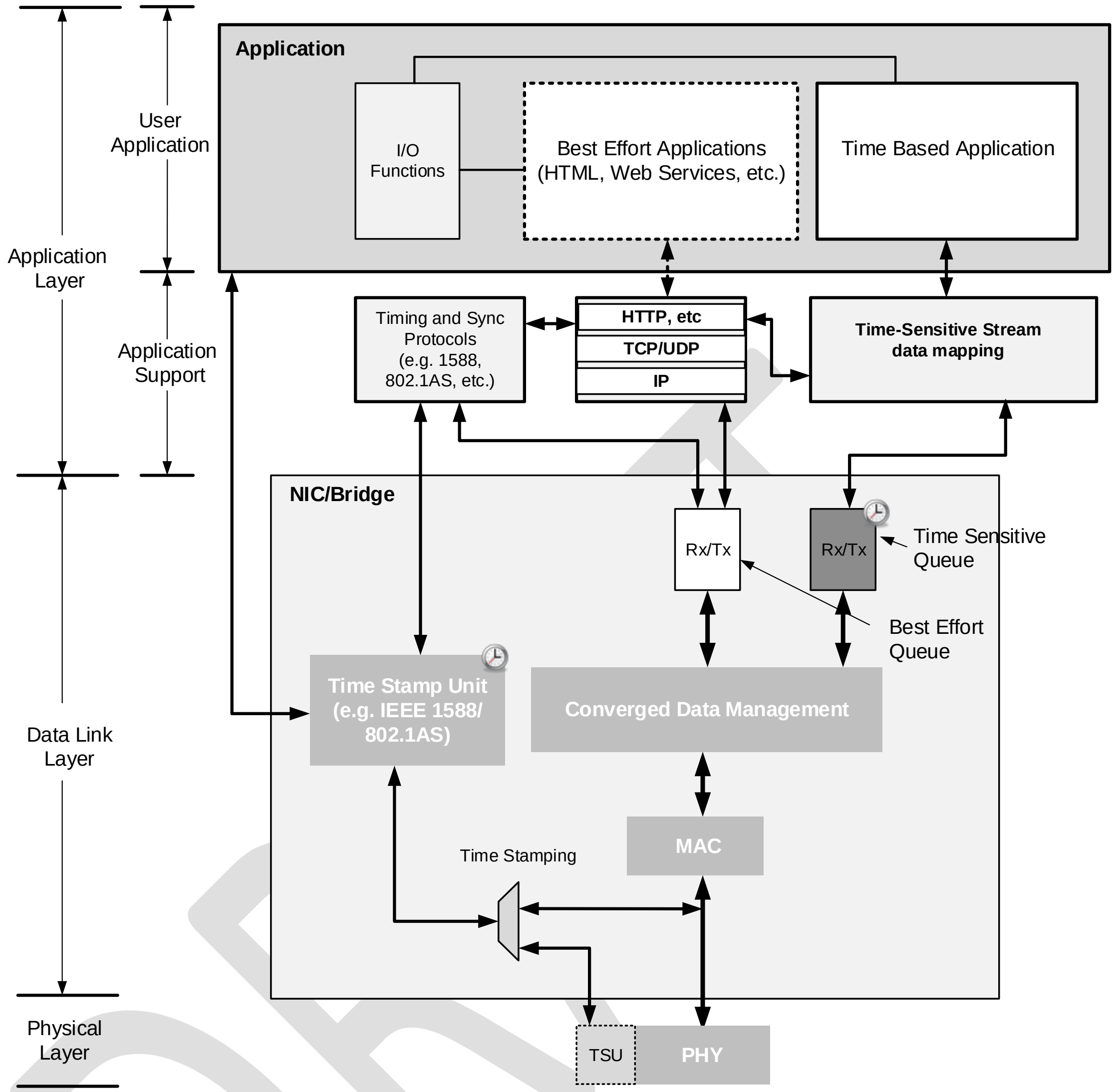


Figure 22: Time-Aware CPS Device Model[13]

The physical layer receives data units from the data link layer and encodes the bits into signals and transmits the resulting physical signals to the transmission medium connected to the CPS node. If the physical layer supports a time stamp unit (TSU) then its management interface should be connected to the data link layer so that a time stamp can be retrieved as and when required by the timing and synchronization protocol (e.g. IEEE Std. 1588TM).

The data link layer provides time-sensitive data communication among devices in a CPS domain. The data link layer implements a set of dedicated buffer pairs (Tx and Rx queues) for time-sensitive data. At a minimum two pairs of buffers are required so that time sensitive data can be managed independently from best effort data. The time-sensitive transmit buffer is

[13] Source: Sundeep Chandhoke, National Instruments

connected to a scheduled (time-triggered) transmit unit. This unit uses a schedule provided by the CPS Network Manager and reads data from the application and copies it into the time sensitive transmit frame and transmits the frame on to the CPS domain.

- The application layer consists of two parts:
- Application-support protocols: These are the protocols that support the conveyance of time sensitive data at the user's application level.
- Time-Sensitive Data Mapping: Protocol to manage the mapping of application data to time sensitive data exchange frames between devices. An example can be CANopen [168] which is used as a data-mapping protocol by multiple industrial protocols.
- Best-Effort protocols: Used for standard internet access, non- time-sensitive streams.
- Timing and Sync Protocols: These include protocols which propagate synchronized time from the network to the application (including I/O functions). Some examples of such protocols are IEEE 1588, IEEE 802.1AS etc.
- User application: User defined applications accessing time sensitive and best effort data, and time-sensitive I/O interfaces to allow decoupling of logical and physical time with enforcement only at the boundary to physics. An example of a realization of this capability is inherent in the design of the Texas Instruments DP8360 Ethernet PHY[14].

Currently time in CPUs is implemented via time-stamp counters (TSC) that increment time using the local clock driving the CPU. This clock does not maintain network time. The TSC can be disciplined via software to slave it to network time. However this leads to significant loss of precision and accuracy. For CPS nodes that synchronize to a single external clock source, it may be desirable to have the TSC driven directly by the network time. This may be implemented by linking the registers of the TSC with the timekeeper in the network interface or by providing a common time-base which can be atomically captured by the network interface before propagating the network time to the CPU or any peripheral device. More generally, CPS applications may choose to maintain offset/PPM state for each derived clock and translate on-the-fly as needed without physically disciplining the TSC. This is especially useful in cases where the applications care about multiple time sources.

Languages used for modeling and programming of time-aware CPS need time as a fundamental programming semantic. Time in the language is required when interfacing to physical I/O and the network. Functions that take future time events to read physical inputs and write physical outputs can enable coordination of physical I/O with scheduled data on the network. Additionally, time-triggered loops can enable coordination of logic execution with schedule of transmission of data. PTIDES [169] and LabVIEW[15] [170] are two examples of system design tools which implement these time-based programming semantics.

[14] Product names and models are included only for reference, with no endorsement implied.

[15] Product names and models are included only for reference, with no endorsement implied.

CPS can employ operating systems with a wide range of complexities, from a simple application-level infinite loop (e.g. the Arduino platform) to a virtual machine hypervisor running several instances of virtualized systems on a multi-blade, multi-core hardware platform. The issues that arise throughout these systems with respect to time-awareness are how to get time to the application with a bounded latency and with accuracy, and how to schedule tasks with time accuracy and bounded latency. Greater detail on this important facet of CPS can be found in the Timing Framework Annex Section 1.2 [144].

At the application layer, the introduction of explicit time will have a profound impact on the conception, design, execution, and robustness of CPS applications. This is a very active area of research, but there are hints of things to come. For example the concept of decoupling of logical and physical time with enforcement only at the boundary to physics mentioned above has yet to be fully exploited. In some cases, tradeoffs can be made between message passing, which consumes network bandwidth, and reasoning about timestamps can be exploited by applications.

Building CPSs using the above mentioned techniques will make it easier to analyze systems, which is a key requirement of safety-critical systems. CPSs with scheduled converged networks built with FPGAs and time-aware CPUs will provide static guarantees and always satisfy timing requirements. Architecture-specific analysis tools can derive these guarantees in the form of upper and sometimes also lower bounds on all execution times, since time is foundational in all elements of the CPS.

4.3.4 Secure and Resilient Time

Requirements for secure and resilient time exist at all layers of the network from the physical to the application layer. While time is physical, its abstraction into networks and complex information systems transform its security into cyber and physical concerns. Therefore, time affects both cyber and physical security architectures. As described in the Timing Framework Annex Section 1.3.2 [144], timing may be vulnerable to unintentional (interference, space weather impacts, network anomalies, etc.) or intentional threats such as jamming and spoofing (counterfeiting via RF signal injection or cyber-attack). The ability to meet timing performance requirements in CPS is also susceptible to vulnerabilities either related to time protocols in use or introduced by cybersecurity measures. For example, firewalls may isolate time between a network in protection and the external network at large. With time isolated, clock drift may occur between both the internal and external networks resulting in performance degradation and in some cases failure at one or more levels. More importantly, networks attempting to normalize or restore services from intentional or unintentional compromised time synchronization run a high risk of timing alignment issues.

Due to the increasingly wide range of timing dependent applications in critical infrastructure domains, secure time must be designed into the system in order to *detect* timing anomalies before performance degradation of the system and to seamlessly ensure sufficient time accuracy and precision can be maintained in the overall system during a compromise. This section describes the elements that constitute secure and resilient time, how time can be compromised, and methods for ensuring access to secure and resilient time.

4.3.4.1 Elements of Secure Timing

There are several widely application-dependent ways to distribute time. For example, a CPS in a closed system might need self-consistent time that can be achieved via a local implementation of PTP. Other CPS might need to be synchronized globally to UTC and depend on GPS, or a GPS-derived network timing source. Each of these timing sources enters the CPS from a different network layer and hardware chain.

Wherever possible and viable, timing distribution systems should provide some level of data and channel assurance. This source-provided timing assurance provides a baseline of security that individual CPS may or may not choose to enhance on an application specific basis. A timing source may securely distribute time (i.e. assured time) to CPS, or if source assurance is unavailable, as part of its timing module a CPS may be able to verify the authenticity of non-assured time. Additionally, regardless if it is using assured time, a CPS should fail predictably in the event its time is denied, disrupted, or detectably manipulated. A CPS with fully secured time must possess the necessary assurance and resilience attributes described in Table 1.

Table 1: Elements of Secure Timing

Source channel assurance	Opportunities to verify that timing information is delivered via an undistorted channel whose expected behavior is well characterized to ensure any deviations can be quickly detected. Distortion of the time-transfer channel may be driven by natural events (e.g. solar weather), unintentional actions (e.g. physically bumping an antenna), or intentional manipulation (e.g. introducing a time delay via spoofing). The data carried by a time-transfer channel may assist in verifying the channel itself. Enablers of channel verification may include unpredictable bits of a digital signature, or a symmetrically encrypted channel.
Source data assurance	Verification mechanisms to prove timing data are not forged. These may include digital signatures or symmetrically encrypted packets.
User provided assurance	User implemented security to verify unassured timing information. This may include anti-spoof GNSS receiver techniques or additional layers of network security.
Predictable failure	Known CPS failure modes that account for timing denial and other detected timing anomalies.
Diversity & Redundancy	Multiple sources and paths of secure time are available to a CPS. Where possible, sources are verified against each other, and in the event of a denial or spoofing attack on one source or other timing anomaly, a mechanism to switch to a redundant source is available.

When a timing source does not make assured time available, the CPS should implement timing assurance methods appropriate for the level of protection they need. Table 2 provides a survey of timing distribution methods and whether or not they provide any level of source channel or data assurance. Different levels of timing assurance are appropriate for different applications. For example, a car's timing network may require more security than a networked household appliance Table 2 indicates whether *any* elements of assured time are present in these distribution methods or whether they remain open to a trivial attack. Current timing distribution systems are generally lacking in source provided assurance and rely on users to implement their own security measures; however opportunities may exist to enhance their security.

Table 2: Survey of Time Distribution Methods

	Order of Timing	Source Channel Assurance Provided Today	Source Data Assurance Provided Today	Source Channel Assurance Possible via Enhancement	Source Data Assurance Possible via Enhancement
GPS L1 C/A	nanoseconds	No	No	No	No
GPS L2C/L5	nanoseconds	No	No	Yes	Yes
Galileo	nanoseconds	No	No	Yes*	Yes*
PTP [171]	nanoseconds	No	No	Yes	Yes
NTP [172]	milliseconds	No	No	Yes	Yes
eLoran [173]	nanoseconds	No	No	Yes	Yes
WWVB [174]	microseconds	No	No	Yes	Yes

*Galileo is not yet a fully operational GNSS constellation, but has indicated strong support for source channel and data assurance via navigation message authentication.

To safely and reliably operate in today's threat environment, a CPS should implement as many elements of secure timing as possible. Ideally, every CPS in a safety-critical application should have multiple, independent, assured, and traceable sources of time with safe and predictable failure modes should time be denied or perceptibly manipulated. Where a mix of secure and unsecure timing sources are available, and traceability to a common time standard exists between them, the unsecure timing sources may be validated against the secure timing sources.

Secured time signals and measurements should be assured for a CPS that uses well-defined performance metrics including: phase accuracy, frequency stability, holdover capability, mean detection time, traceability, and switchover time. Addressing the research needs for a fully secured time in safety critical CPS remains a high priority.

The Timing Framework Annex Section 1.3.8 [144] describes two possible use cases in the power system domain where secure time is necessary. The first use case describes how GNSS vulnerabilities can lead to synchrophasor measurement errors. To enable PMUs for real-time control, the power industry must ascertain the measurements are accurate and reliable. Erroneous measurements could appear as instabilities in the grid. Automatic protection schemes relying on the compromised measurements could trip generators. Tripping generators unnecessarily can cause blackouts and/or significant damage to power systems equipment. The use case illustrates how elements of secure time implemented on top of GNSS timing led to a hypothetical detection of the GNSS compromise. Subsequently, a predictable failover to an equally precise redundant timing distribution system would ensure access to trusted time.

Similarly, the second use case describes how digital substation automation can be compromised by network timing protocol attacks such as spoofing and Denial of Service (DoS). Again, both attacks can lead to erroneous measurements of synchrophasors, leading to inability to accurately monitor the state of the grid, and potentially impacting control decisions. Network time distribution that implements the secure time elements including: source channel and data assurance, user provided assurance, predictable failure, and diversity and redundancy would minimize any compromise’s impact on system timing performance.

Without assured time, critical infrastructure systems that people depend upon daily (power distribution, telecom, transportation, the Internet, etc.) are vulnerable to disruption. As Table 2 illustrates, time distribution methods available today require user or system enhancements to meet source channel and data assurance requirements. If there are existing security measures built into the time distribution method, the measures have vulnerabilities that are trivially compromised by a naïve attacker. Additionally, most end-use timing equipment is vulnerable to the disruption caused by source channel and source data disruption.

4.3.4.2 Current Security in Distributed Timing Systems

Timing is generally distributed to CPS via GNSS constellations or a network timing protocol. This section surveys the security mechanisms and vulnerabilities inherent in these two distribution methods.

4.3.4.2.1 GNSS Timing Directly to Devices/Equipment

Civil GNSS signals are the primary worldwide timing distribution mechanism, and are inherently vulnerable to jamming and spoofing.

Jamming refers to the denial of the signals-in-space by illegally broadcasting energy in the radio navigation spectrum. Low power (<1W) jammers are widely available to consumers and are marketed and used as "personal privacy devices." High power jammers are generally used to intentionally deny GNSS receivers over a wide-area. Though the effects of denial can be damaging, robust timing receivers should enter into pre-defined holdover, mitigation, or failure modes when it is detected.

GNSS spoofing is the RF injection of counterfeit or recorded GNSS signals into a receiver. Spoofing attacks may be data (e.g. replace the navigation data on the GPS signal) or timing oriented (e.g. induce a delay). Jamming may be intentional or incidental. Generally spoofing is intentional, though it may be possible for incidental spoofing to occur (e.g. through legal GPS repeaters). Unlike incidental jamming, many straightforward mitigations exist to incidental spoofing. Though spoofing is not yet as commoditized as jamming, publicly available research into spoofing techniques has been significantly increasing and software defined spoofers have been appearing in multiple and independent research universities.

As the majority of critical infrastructures rely on GNSS as a reference source, GNSS jamming and spoofing are known critical infrastructure vulnerabilities (due to its reliance on GNSS provided timing), and awareness of their consequences has been increasing significantly. Current areas of research include source channel and data assurance, anomaly detection before clocks are significantly impacted, and redundant distribution sources.

There has been significant work done on receiver side techniques to mitigate spoofing and jamming. Some GNSS providers (Galileo) have advanced toward securing the signal-in-space via navigation message authentication (NMA – that is, digitally signing the data transmitted by the satellites). An NMA implementation scheme that could be implemented on the modernized civil GPS signals is being considered [175]. A signal-side security scheme such as NMA provides an affordable and backwards compatible baseline of protection for civil GNSS receivers against spoofing, and would provide globally available time that is "source assured". Receivers could choose to ignore NMA, adopt it, or adopt it and implement additional measures of assurance. Asymmetric cryptography schemes can also be added to other timing signals and protocols (e.g. possibly WVVB or PTP) for source channel and data assurance.

The development of other methods for national-level reference time distribution to backup and augment GNSS in the event of a failure has become another active area of research. The Timing Framework Annex Section 1.3.3 [144] describes some currently available or researched alternatives to distribution of time traceable to a national reference. WWVB and eLORAN[173][176] are two alternatives that have been able to achieve wide area synchronization. Research efforts in alternative methods include achieving a timing accuracy comparable to GNSS as well as ensuring secure time in the alternative methods. Communication sector timing distribution methods such as network time distribution protocols over dedicated optical networks or a combination of SyncE and PTP can serve as an alternative source of national reference time. Another area of research is in Assisted Partial Timing Support (APTS) [177], which provides active monitoring and detection of synchronization deviations as well as automatic switchover to an alternative network time distribution source in the event the GNSS is deemed unreliable.

4.3.4.2.2 Network Timing

Network timing distribution leverages a packet-based protocol (e.g., PTP or NTP) to distribute timing information via a hierarchy of receivers. At the top of the hierarchy is a timing source that often derives a traceable national reference time from a satellite constellation (e.g. GPS) or another time transfer source (e.g. eLORAN[173][176], WWVB[174], etc.). Network timing distribution has a different set of security considerations than GNSS based timing. Network-based distribution methods are prone to common network vulnerabilities. The threats can compromise the integrity and availability of time in a CPS network. Securing network time distribution methods include assurance for authenticity to a traceable time reference, integrity of the time stamps and other metadata exchanged in the synchronization packets, and availability through redundant and diverse paths. Another key requirement to secure time in networks is the ability to detect the intrusion or other forms of anomaly in the network before the threat has impact on the network time. When anomalies in the timing distribution network are detected, the CPS would have the means to fail predictably with minimal impact on the function of the system. Ideally, the system would have diverse and redundant paths for timing distribution where the system can switchover readily once an anomaly is detected while maintaining the necessary timing accuracy and precision in the CPS.

4.3.4.2.2.1 Attack Vectors in Time Networks

Network timing distribution methods are susceptible to attacks characterized by an unauthorized third party, known as Man in the Middle (MitM) or interceptor, which can be manifested as different threat types. Table 3 outlines different principal threat vectors [178][179] and their impact on time networks. The impacts of the threats include limiting the availability of time distribution in the network, distributing completely erroneous time or distributing time with reduced accuracy. The threats can be passive (message interception) or active (message interruption, insertion or modification). Passive attacks tend to be the prerequisite to other attacks. Therefore, detecting passive attacks is one method to preventing an attack from having impact on the timing accuracy of the CPS.

Both external and internal perpetrators must be considered in a network security threat analysis. While external attackers do not have access to the network's security credentials, internal attackers do. The Timing Framework Annex Section 1.3.4 [144] provides more in-depth definition of terms for describing network time compromises, and The Timing Framework Annex Section 1.3.5 [144] provides a detailed external and internal threat analyses for network time distribution protocols.

Table 3: Principal threat vectors in an unsecured time network

Threat Type	Threat Characteristic	Impact	Example

Packet Manipulation	Modification (Man in the Middle (MitM))	False time	In-flight manipulation of time protocol packets
Replay Attack	Insertion / Modification (MitM or injector)	False time	Insertion of previously recorded time protocol packets
Spoofing	Insertion (MitM or injector)	False time	Impersonation of legitimate master or clock
Rogue Master (or Byzantine Master) Attack	Insertion (MitM or injector)	False time	Rogue master manipulates the master clock election process using malicious control packets, i.e. manipulates the best master clock algorithm
Interception and Removal	Interruption (MitM)	Reduced accuracy, depending on precision of local clock	Time control packets are selectively filtered by attacker
Packet Delay Manipulation	Modification (MitM)	Reduced accuracy, depending on precision of local clock	Intermediate / transparent clock relays packets with non-deterministic delay
Flooding-based general Denial of Service (DoS) or Time Protocol DoS	Insertion (MitM or injector)	• Impairment of entire (low-bandwidth) network • Limited or no availability of target (service)	• Rogue node floods 802.15.4 network with packets • Rogue node overwhelms single victim with time protocol packets
Interruption-based general	Interruption (MitM or possibly	• Impairment of entire network communicatio	• Rogue node jams network • Rogue node jams

DoS or Time Protocol DoS[16]	injector)	n • Limited or no availability of target	selectively certain time protocol packets
Master Time Source Attack	• Interruption (MitM or injector) • Insertion (MitM or injector)	• Reduced accuracy • False time	• GPS jamming • GPS spoofing
Cryptographic. Performance Attack	Insertion (MitM or injector)	Limited or no availability of target	Rogue node submits packets to master that trigger execution of computational expensive cryptographic algorithm (e.g. digital certificate validation)[17]

Current mitigation strategies for addressing network time distribution vulnerabilities include authentication of the synchronization source and integrity verification. NTP uses the AutoKey protocol to achieve end-to-end authentication, message integrity and replay protection. NTP is an end-to-end synchronization protocol whereas PTP is a hop-by-hop synchronization protocol using transparent/boundary clocks to achieve higher synchronization precision. The ability to secure a hop-by-hop protocol presents a unique security challenge. PTP has an experimental Annex K, which provides group source authentication, message integrity, and replay protection. The Timing Framework Annex Section 1.3.6 [144] describes some of the network timing distribution protocols' security extensions. With the increasing demand for security, existing security protocols such as MACsec and IPsec can be used to complement PTP. MACsec provides hop-by-hop integrity protection, whereas IPsec provides end-to-end integrity protection. The Timing Framework Annex Section 1.3.7 [144] details current countermeasures for achieving authentication and integrity.

Similar to GNSS, research continues with respect to detection of anomalies and the ability to maintain resilience of the clock synchronization network while maintaining the increasingly stringent precision and accuracy requirements. In large scale and dynamic networks, key

[16] This attack is blunter than the "Interception and Removal" attack above, as here all time-protocol related packets are omitted.

[17] The exchange and validation of a certificate as part of the authentication and authorization of a node can be the building block of such an attack.

management is a challenge in ensuring hop-by-hop timing protocol (e.g. PTP) security. Furthermore, there is a continuous need to improve countermeasures as new vulnerabilities arise. There is currently a demand on the network time distribution protocol standards efforts for guidance in achieving secure timing, while minimizing impact on time distribution performance. Current security extensions are susceptible to certain threats such as cryptographic spoofing and a variety of internal attacks. Standards efforts are currently underway to define optional security specifications for meeting source channel and source data assurance in NTP[180][181] and PTP[182].

4.3.4.3 Achieving secure time

Timing security in critical systems requires more than the availability of secured timing sources. Secure time requires including timing security in the CPS system architecture from its design in such a way that when the system detects potential timing compromises, it can failover to a redundant timing source (either internal or external to the system). Existing technologies utilize redundancy and diversity of routes to time and frequency sources as well as holdover capabilities of high stability oscillators. There continues to be research needs in the areas of timing compromise detection, alternative sources to traceable national standard reference time, timing network topologies to support diverse and redundant paths, and cybersecurity measures that minimize impact of timing performance. In addition, practical testing and validation of experimental results would ensure safety and predictability in failure modes.

Due to the lack of secured timing sources globally available today, a reasonable approach to securing time is to ensure systems can maintain timing within the tolerance of their application for the duration of a timing compromise. The future vision of secure time is to ensure timing compromises can be detected sufficiently early such that systems dependent on accurate and precise timing can seamlessly function under compromised conditions without any performance impact to the CPS.

4.4 Performance Aspect [tbd]

4.5 Life Cycle Aspect [tbd]

Specify /Engineer / Procure / Operate / Maintain / Dispose

4.6 Topology Aspect [tbd]

CPS consist of systems that include devices. There are many kinds of devices. Each device falls under the responsibility of one or more organizations that has responsibility its configuration, life cycle maintenance, and access rules to interact with it. Additionally, there is a network topology overlaying the organizational topology. Finally, topology includes the notion of location.

Due primarily to the existence of these location, network and organizational boundaries, and the intersecting interests in access-rights allocation, topology is a critical aspect of CPS.

In some cross-cps-domain use cases, access to data by client applications far removed from the actual administration of devices may be desirable.

The following are general classes of physical devices that can potentially interact:

Sensors & actuators: The simplest functionality that allows the interaction between cyber and physical.

Controller: Controllers combine data from sensors and produce control actions via actuators.

Gateway: Gateways provide the ability to forward information exchange between local devices within a proprietary network and a remote network (often the Internet). Gateways are often, but not always, the boundary between private and public networks.

Aggregator-concentrator: Aggregators and concentrators provide for data fusion and allow for managing the forwarding of information obtained from resource-constrained networks to more capable ones.

Broker: Message brokers supporting publish and subscribe message routing and certificate assurance services are examples of infrastructure components that enhance the function and security of information exchange.

Cloud-based analytics: “Big-Data” and other cloud-based services provide for the exploitation of large collections of data from many sources.

These classifications of devices are an initial starting point for the exploration of the impact of topology on CPS.

5 Use Case Analysis – Use Case Subgroup

5.1 Background

This section provides an overview of use cases as they are used in the NIST Cyber-Physical System Public Working Group. It serves to orient the reader and guide them through the remainder of the Use Cases section. It is not intended to serve as a treatise on use cases (there are plenty of references on that), nor as a (necessarily incomplete) list of use cases for CPS systems. This section does, however, describe how we can better understand the functional requirements for these systems, by examining functional examples and use cases describing CPS systems. This will help to validate the reference architecture being developed by the CPS PWG, guide standards development organizations in the development of supporting standards, and assist software and hardware developers in the creation of supporting products.

5.1.1 Requirements

To understand how to design a system it is important to understand what the goals of the system are, and what the requirements are that must be satisfied to achieve those goals. Developing use cases is one method of gathering functional requirements for a system based on the known ways the system will be used. Non-functional requirements are not captured in the use cases (but sometimes may be inferred from them). In the specific case of CPS, the CPS environment should support not just the known functions, but also promote innovation and provide the flexibility to develop the new functionality that will accompany this innovation. The use cases find only those requirements driven directly by known uses of the systems so the output of the Use Case subgroup must be used with other methods of gathering requirements.

CPS use cases exhibit certain system properties. The collection of these properties distinguish system that express them as "CPS." These properties include, but may not be limited to, timing, security, and data interoperability requirements and the like. We recognize other types of systems can have properties in these areas, but have determined that these system properties must be fulfilled by any realized CPS architecture, and so become requirements placed upon the reference architecture.

5.1.2 Relationship with Other CPSPWG Sub Groups

Because use cases provide a link between each user's goals and the system requirements, as described above, there is a tight coupling between the use cases and the system or infrastructure architecture. This implies the need for tight coupling between the CPSPWG use case subgroup and the CPSPWG architecture subgroup.

The use cases are used both to check the scope of the CPS definition created by the CPSPWG Architecture subgroup and to derive a set of requirements that the CPS reference architecture must support. In this way the output of the use case subgroup functions as input to the development of the CPS definition and the development of the CPS reference architecture. Once the CPS definition and architecture are complete, the use cases and requirements will be used to validate the definition and architecture.

The other three CPSPWG subgroups are also linked together with the use case subgroup. Each use case may have specific timing, security, and data interoperability requirements. Once these requirements are identified by the use case subgroup they will be fed to the appropriate subgroup for investigation. Additionally, any specific timing, security, or data interoperability use cases that are generated within the three subgroups will be fed into the use case subgroup and included in the CPS PWG use case repository.

The interactions are bidirectional and started at the beginning of the PWG process to ensure that there will not be any major gaps at the end of the process.

5.1.3 Overview of CPS Use Cases

Use cases are a common technique for gathering requirements in systems of many sorts, including cyber-physical systems. Each use case describes how an actor (the user) interacts with a system to achieve a goal. Use cases are used to elucidate functional behavior, with an emphasis on the value delivered to the users of the system. Each use case captures a function, or range of functions, required by the user, and acts as a guide to engineers responsible for developing the hardware and software that will make up the system.

A “user” refers to the actor that interacts with a system. A user can be a human or a constructed system. More generally, and especially in CPS, a “user” may be a person, machine, another system, or even the system itself, which may respond to an internally generated trigger. The actor concept represents a role that interacts with the system to cause it to carry out some function. To capture the “real” requirements, however, we must step back from the actors and also consider the constellation of entities affected by the system, such as regulators, corporate strategies, society or the environment at large, collectively known as “stakeholders.”

In the case of a single system, a complete collection of identified use cases should comprise a complete set of functional requirements for that system. Experienced engineers then scan the collection of use cases for common aspects that can be implemented once, and used in multiple places. For example, a control system for a chemical plant might need to control both temperature and pressure with a deadband. We might invent, or take off our mental shelf, an implementation of a PID loop, or, more broadly, a control loop. The same implementation can be used in multiple contexts. From the other direction, the concept of an acceleration profile can be applied for an elevator, a robot arm or a tape drive. Even though the specific application domains are different, the same pattern can be applied.

Because this process abstracts away from the specifics of a particular application, we may go one step further and observe *collections* of interlocking patterns that often appear in similar *types* of systems, such as batch, event-driven, service-oriented or cyber-physical systems. Such collections of interlocking patterns of the elements of a (type of) system, what they are and how they connect, is part of what is called the system’s “architecture.”

Colloquially, however, “architecture” does not require a careful definition. For our purposes, it is a convenient term to refer to the abstract organization of the elements of a system and how they connect one to another. This is why we are gathering use cases: We wish to identify the kinds of elements that comprise a cyber-physical system and how they are related, and from that we hope to identify requirements and gaps in the architectures of cyber-physical systems.

Broadly speaking, then, the process is to:

- Identify stakeholders
- Identify application categories
- Identify and elaborate CPS examples and use cases
- Identify architectural dimensions (high level view)
- Identify primitive requirements for CPS architecture

However, the number of potential CPS use cases is practically infinite, and will continue to expand as CPS systems are applied in new ways and unimagined markets. For this reason, we cannot hope to find all use cases. Instead we have developed a method (described in 5.2) to allow us to analyze sets of use cases with some repeatability at a high level and use the analysis to decide whether they need further elaboration. The method is based on clustering use cases based on a set of characteristics particular to the architectures of CPS. These characteristics can be broadly grouped together (shown in Table 5) and include groups such as functional concerns (device control or analytics) and cross-cutting concerns (security or timing). Each use case can then be categorized as imposing requirements on say, timing, or not. Additional categorization can be done based on actors, application types, and systems, each aspect providing a different view into the system.

This structure is reflected in the structure of the report, which begins with the stakeholders we have identified, and then the application types, and finally the requirement categories, showing relationship of the example/use-case to all the relevant requirements.

In the following subsection, we describe the method we use to evaluate and classify use cases, and how we then identify the requirements. The subsection after that describes just a few supporting use case examples (for this preliminary report).

Finally, we list the requirements we have identified on the architecture. We divide these into requirements into those placed on the functional architecture, and then the cross-cutting concerns of cybersecurity, timing and data management.

5.1.4 Stakeholders

The stakeholders of a system are by definition “a person or group that has an investment, share, or interest in something, as a business or industry [216].” The users are usually perceived as the key stakeholders, but often the primary focus is on the usability of the system and the system performance in meeting the user goals. The secondary stakeholders are also important, and understanding them and their needs will provide better understanding of the system requirements. Table 4 lists the stakeholder groups identified by the use case subgroup as important to the success of a system, are documented in Table 1 below.

Table 4: List of Stakeholders

Classes of Stakeholders	Who Are They?
Creators	The builder, system integrator, project manager, etc. of the CPS.
Owners	Those who own the CPS.
Operators	Those who operate the CPS.
Customers/users	These are those who benefit from the function performed by the system.
Supply chain providers	Third-party suppliers of components anywhere in the supply chain that end up in the CPS product.
Service providers	Consultants, contractors, lawyers, bankers, …
Insurers	Insurance companies.
Regulators	Mostly state and federal agencies responsible for developing and monitoring regulations.
Competitors	Companies in same market as the entity that experienced failure.
Government	Representatives of the three branches of government. Includes local, state and federal.

5.1.5 Application Categories

The application categories or types describe the different business areas in which CPS are predicted to be used. Some of the core application areas include: emergency response, where a CPS needs to be quickly assembled from an assorted set of (possibly not fully functional) components; manufacturing, where systems integration and maintainability can lead to cost savings and improved safety; defense systems with important reliability and security requirements; and even advertising that is linked into events in the physical world. These are only a few of the exciting possibilities; our entire list of application categories is shown in Table 5. This list will be updated as new categories are uncovered. The most up-to-date list will be found on the www.cpspwg.org website.

Table 5: Application Categories

Application Categories	
Transportation	Buildings
Manufacturing	Education
Cities	Social networks

Emergency response	Financial services
Healthcare	Environmental monitoring (e.g., greenhouse gas emission tracking)
Consumer	Science
Infrastructure (power, water)	Aerospace
Entertainment/sports	Disaster resilience (includes preparedness and crisis management activities)
Defense	Energy (included in “infrastructure,” but this is a very broad category)
Advertising	Communities
Agriculture	Leisure
Supply chain/retail	

5.2 Analysis Method

The pool of potential use cases is infinite. This makes filtering the examples and use cases to a set that effectively covers the requirements a daunting task. Additionally, the degree of similarly between use cases can vary greatly, making it even more difficult to process examples and use cases. To overcome this problem, there must be a thorough evaluation of each use case to identify common properties. This process will allow us to cluster the use cases based on architectural characteristics so as to get coverage where there are gaps in requirements for the reference architecture. For example, if our collection of uses exhibited only loose timing requirements, we might solicit another use case with stringent timing requirements.   (The specific properties are examined in more detail in Table 7 Requirement Categories.) For the evaluation process to be effective, it is imperative that each example and use case is evaluated in a consistent manner. To this end, the Subgroup developed a standard approach to use case evaluation.

This method provides an approach to identify patterns of use of CPS-based solutions from a set of use cases corresponding to different types of applications. These patterns of use will determine the specific architectural requirements that can be organized and described in a CPS reference architecture framework. The patterns also illustrate the capabilities needed to run the processes of the applications of interest. In general, the methodology is intended to help a CPS-based solution stakeholder to describe the requirements of an application, i.e. the problem description. These requirements are inputs to the CPS-based solution providers both directly – as a set of requirements needed for a specific system or type of system — and indirectly through the CPS PWG reference architecture.

For this effort, the CPS PWG Use Case subgroup will use a two-stage process designed to support differing uses for this information. The first step is to collect and analyze high-level CPS scenarios (which we will refer to as “CPS Examples” to prevent confusion with how scenarios are used in use case terminology). These examples can describe complex interactions between several systems and may cross one or more application category boundary. The examples will help us understand what requirements areas are important for that example and what the different actors and systems are (actors are a type of system, but in this case they are specific types of systems acting on another system). The CPS Example analysis phase will help us gain valuable knowledge about the types of actors, systems and their interactions along with a general understanding of the types of requirements need for each example. This first stage will not provide the specific simple requirements that will be needed to thoroughly validate the architecture (and can also be used to validate any systems designed to meet the full set or a subset of the requirements). Phase two will fill that need.

To gather the more detailed, specific requirements necessary to validate the CPS architecture, we will deconstruct CPS examples into a set of black box use cases (black box use cases describe the specific interaction between an actor and a system with no knowledge of what goes on within the system). We will then analyze these black box use cases can then be analyzed using a set of primitive requirements (which may be associated either with a use case or with a specific step within the use case). These primitive requirements will provide specific singular requirements that are mapped to specific steps within a use case (and therefore are associated with a specific actor and system). By looking at a set of these functional requirements for a specific instance of a system on can then a) build a system based on these requirements or b) test a system based on these requirements.

To generate the primitive requirements for CPS we are using a set of smart grid primitive requirements as the starting point. The thousand-plus requirements developed as part of the EPRI IntelliGrid project [217] are being modified and expanded to fit the more general needs of the CPS environment. We will map the primitive requirements will be mapped to high-level requirements categories described in part one.

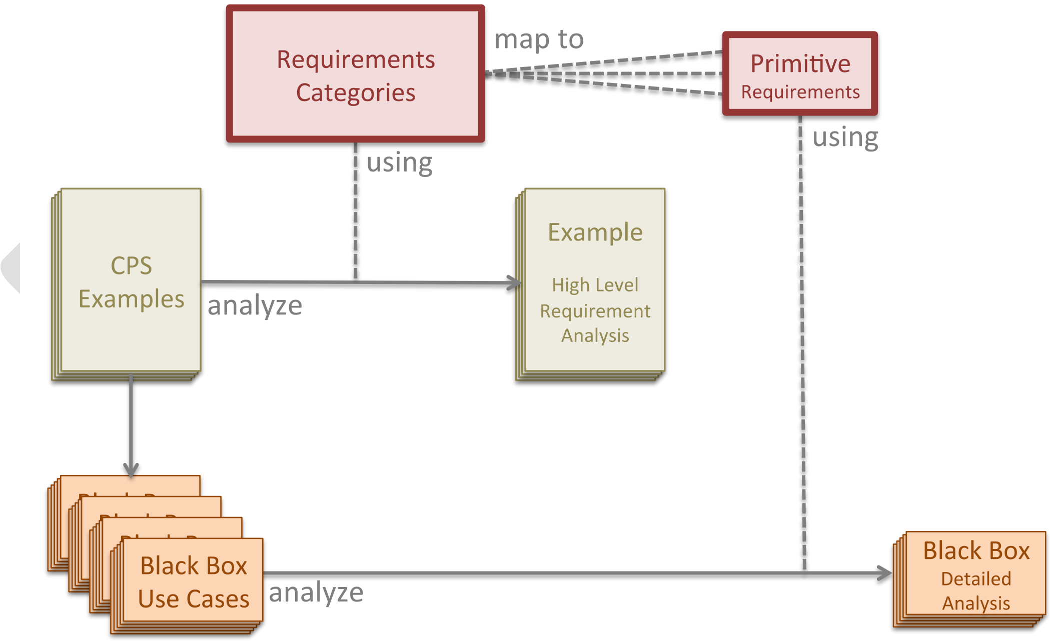


Figure 23: Requirements Decomposition into Primitives

The output of the CPS Use Case subgroup will be the requirement analyses of the set of CPS Examples and a set of primitive requirements for the set of black box use cases. While at first the output will only cover a selected set of important examples and use cases, over time it is desirable to cover all the requirements categories (5.2.2).

5.2.1 CPS Examples – method

The CPS Example is a use case summary describing a set of actors and systems that interact to achieve a variety of goals (not always the same goals). It contains information on the actors and systems (and systems can be actors as well – in this case a system is something that is acted upon and an actor is the entity doing the acting on the system). The CPS example differs in one major way from the black box use cases used in the second phase of this project – the example have multiple systems, actors, and interactions, where the black box use cases have only one.

Table 6: CPS Example Template

CPS Example TEMPLATE
CPS Example Name - phrase describes interaction btwn actor and system
Description - Brief description
Notes – any relevant notes that help in understanding the use case
Goals – what goals do the stakeholder want to see achieved?
Use Case Source Organization - Who developed the use case
Actors - The actor that intearact with the systems described in the example
Systems - The systems being acted on by the actors described in the example

5.2.2 Requirement Categories

Once the CPS examples have been collected, the next step is to evaluate them in terms of their architectural characteristics. These characteristics cover question like the volume and velocity of data, variability in data sizes, confidentiality, timing constraints, and computational effort. Since these characteristics are quite heterogeneous, they are grouped into two levels of categories, as shown in the first two columns in Table 1-4.

The architectural characteristics are directly related to the system properties described above. If a use case is part of a system that exhibits a need to collect data continuously (avionics that determine aircraft position, for example), then this implies styles of implementation that can realize continuous behavior (an analog subsystem that must be integrated with the rest of the system), or a digital system that operates periodically. The reference architecture should be able to cater to both architectural characteristics.

As each use case is evaluated, after it is compared against the known characteristics, we must also look for any unique characteristics that are not covered by the standard form. If there are such characteristics, we will modify the form will be modified to address the additional needs of the use case. We will also retroactively apply the modified form to previously processed and future use cases. This iterative approach will ensure that the methodology for evaluating use cases is comprehensive and adaptable to changing needs.

The table below, then, is a starting point, rather than comprehensive. Architectural characteristics may be added based on known properties of CPS systems that are not reflected in our current set of use cases.

Table 7: Requirements Categories

Cyber Physical System Characteristics	**Application Areas**	Does the use case require a system that crosses multiple application areas? If so, which application areas are included?
	Composition **Intersystem Interaction**	Does the use case require the interaction of heterogeneous subsystems?
		Are there specific requirements caused by the use case interacting with legacy systems
	Human Interaction	Are humans an important part of the system?
Cyber Physical System Data Characteristics	**Physical Properties**	What physical properties are being monitored?
		What physical properties are being acted upon?
	Volume and Velocity	Describe the size of the datasets being processed and the speed at which it comes into/out of the system.
	Computation	Describe the computation effort and processing required to achieve the use case goals
	Aggregation	Describe the requirements to aggregate different data types
	Variability	Is the size of data being generated/used consistent or is there a growth/shrinkage trend?
Accuracy	**Error Sensitivity**	Describe the sensitivity of the system to errors in the data
	Certainty	What is the level of uncertainty in the data being generated/processed and the assurance of the resulting actions taken by the system?
Physical Metadata	**Timeliness**	What are the use case timing constraints?
	Time synchronization	What are the use case time synchronization

		requirements?
	Physical Location	What are the location requirements of the use case?
Reliability	**Robustness**	What are the robustness requirements? (preventing a fault)
	Resilience	What are the resiliency requirements? (recovering from a fault or sub-fault)
Security	**Confidentiality**	What happens if information within the system leaks (or is pulled) out?
	Integrity	What happens if the system acts on incorrect data (including software)?
	Availability	What happens if the system or data it generates is not accessible and prepared to function properly when and where needed?

5.2.3 CPS Use Cases - Method

Once a CPS Example has been identified, along with the associated systems/actors, it will be broken down into a set of black-box use cases describing specific interactions between an actor and a system. The resulting use cases will be described using a template based on traditional use case design, focusing on the actor, the system, pre and post conditions, and the steps between the two. The full use case template is shown in Table 8.

Table 8: Black Box Use Case Template

BLACK BOX USE CASE TEMPLATE
Use Case Name - phrase describes interaction between actor and system
Use Case Description - Brief description **Notes** – any relevant notes that help in understanding the use case
Goal – what goal does performing the use case achieve?

Use Case Source Organization - Who developed the use case
Actor - The actor that performs the steps in the use case
System - The system being acted on in the use case
Pre-Conditions - A list of true conditions before the Use Case starts
Steps - A list of steps to perform the use case
Post-Conditions - A list of true conditions when the Use Case ends

Since our effort focuses on deriving CPS requirements from the use cases, we will use a list of simple (primitive) requirements will be used to associate each step of a black-box use case with a set of requirements. We will develop the primitive requirements using a set of simple requirement statements numbering in the thousands. These simple requirements will be generalized (as is appropriate for CPS covering a wide range of application types) and mapped to the requirement categories used in the high-level requirements analysis.

As we identify new simple requirements (during the use case analysis) we will add them to the set of requirements. We will use the set of primitive requirements to validate the CPS against a set of known CPS functions as the analysis effort approaches completion. The effort can never be finished, as more examples and use cases will be added as they are discovered (in fact this trend might increase as the new capabilities will drive our imagination).

Not only can we use these simple requirements be used to test the CPS reference architecture, we can also use them to describe and test any specific instance of a CPS. If these requirements are used in the development of CPS components, it will become easier to assemble systems and efficiently make use of available resources. We can use the primitive requirements in different ways:

- By grouping the set of requirements together for a use case - the specific use case can be tested
- By grouping the set of requirements for a specific system - the system can be designed and tested
- By grouping all the requirements together - the architecture can be validated (this is described in the next section).

5.2.4 Procedure for Identifying Reference Architecture Requirements

Once all the identified use cases have been processed using this method, the outcome will be a set of characteristics for the use case that the supporting system must be able to meet. While the specifics of these characteristics will be specific to each individual use case, the collection will represent a comprehensive set of use case needs. The next step is to translate the needs into requirement statements that will be levied against the reference architecture, timing and security working groups. We will analyze and abstract each characteristics and abstracting it away from its corresponding use case, grouping them based upon similarity and removing any duplicates.   The result of this process will be a generalized set of needs that will serve as requirements for the other working groups.

5.3 Supporting CPS Use Case Examples with Evaluation

Following are two CPS Examples that have been submitted and then analyzed by the use case group for an initial high level analysis based on the requirement categories.

5.3.1 CPS Example – Monitoring Energy Efficiency of Manufacturing System

In this example, the energy efficiency index of a manufacturing system is needed for reconfiguration and rescheduling, in a run-to-run basis.

Example Description – Level 3 manufacturing operations management obtains a set of production KPIs based on Level 2 and Level 1 operational data about the process, equipment and product. The energy efficiency indices are derived from the production KPIs and used to generate the new manufacturing system parameters for reconfiguration and adjustments to scheduling before the next set of production orders are done.

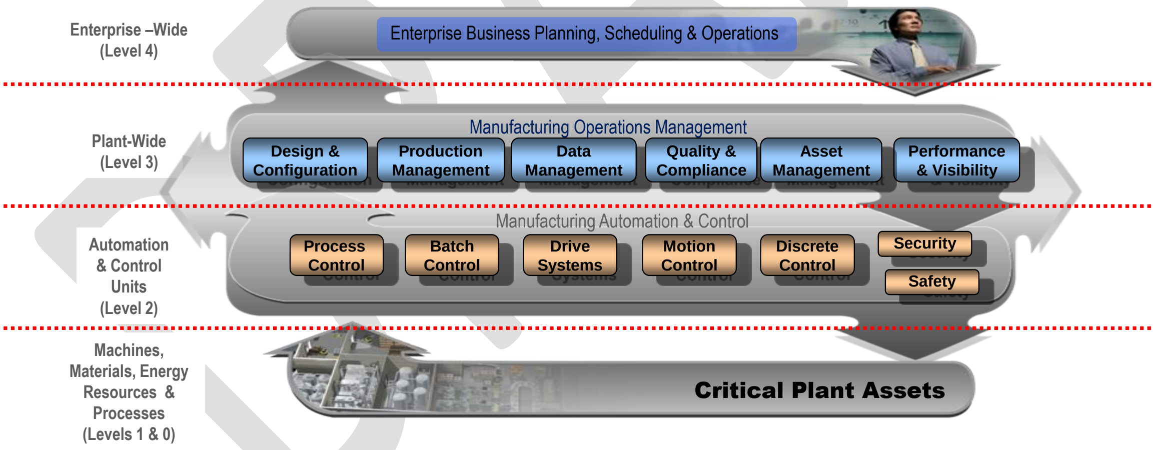


Figure 24: Example of Reference Architecture Model of "Manufacturing" System-of-Interest

Details - A production order prepared at Level 4 of the enterprise has been scheduled for execution at Level 3 with a set of manufacturing resources allocated, configured, validated and dispatched to process the provisioned materials and energy flows and output the required finished goods, at the lower levels (2,1,0), in a work request with detailed workflows;

A work request is sent by a Level 3 MOM application to Level 2 manufacturing control and automation application. Sequence of procedural automation steps are performed by Level 2 automation units to direct the Level 1 sensing, control and actuation units that conduct the production processes and machines (at Level 0) required to produce the desired outputs of the manufacturing system. A combination of data acquisition units collect real time data about the process, materials, energy flows, equipment and personnel that provide the basis for generating the relevant KPIs for evaluating the energy efficiency index of the manufacturing system. A Level 4 production performance tracking application evaluates the energy efficiency index of the current production run and estimates the needed changes to the configuration and scheduling parameters for the next production run to achieve the production objectives in quality, cost, timeliness and safety.

The architectural characteristics of this example use case are shown below in Table 9, without the first column used to group them, so as to save space.

Table 9: Analysis of Use Case

Application Areas	Does the use case require a system that crosses multiple application areas? If so, how many application areas are included?	Across several domains of an enterprise; among functional and resource levels;
Composition Intersystem Interaction	Does the use case require the interaction of heterogeneous subsystems?	Systems of processes, resources, and organizational units
	Are there specific requirements caused by the CPS-based solution interacting with legacy systems	Many of the identified heterogeneous subsystems can be considered as “legacy” types
Human Interaction	Are humans an important part of the system?	Critical to the objectives of an enterprise, e.g., in task prioritization, fault recognition & recovery
Physical Properties	What physical properties are being monitored?	Wide range of physical variables involved in the material and energy conversions plus equipment and personnel coordination to make a product
	What physical properties are being acted upon?	Process, product, equipment personnel properties to be set at target values needed to complete

		production
Volume and Velocity	Describe the size of the datasets being processed and the speed at which it comes into/out of the system.	PLC “I/O data tables” for control loops closed in millisecond cycles up to Manufacturing Operations Management (MOM) KPI targets and results composed and conveyed in seconds
Computation	Describe the computation effort and processing required to achieve the use case goals	Processing efforts scales according to size of enterprise and required throughput of products
Aggregation	Describe the requirements to aggregate different data types	Both composition & decomposition tasks performed on signals, data and information that are at and cross multiple levels and domains [100’s MBs per run or job]
Variability	Is the size of data being generated/used consistent or is there a growth/shrinkage trend?	“Data” associated with various forms, e.g., text, graphics, audio, video, or encoded/compressed bit streams typically span tens of bytes up to tens of MBs per transaction (more in future);
Error Sensitivity	Describe the sensitivity of the system to errors in the data	Critical product tolerances have to be maintained at parts per billion; [with or without fault tolerance mechanisms][exception reporting capabilities to mitigate]
Certainty	What is the level of uncertainty in the data being generated/processed and the assurance of the resulting actions taken by the system?	Very wide range; floating point and 64-bit integer computation mostly a starting point;
Timeliness	What are the use case timing	See above (Volume and Velocity)

	constraints?	
Time synchronization	What are the use case time synchronization requirements?	Tens of processing lines with 10K I/O points per line and job cycles up to 1800 items/hr per line
Physical Location	What are the location requirements of the use case?	Manufacturing and production sites occupy 1-2M/square ft per site, with multiple sites in different regional locations;
Robustness	What are the robustness requirements? (preventing a fault)	MTBF is 5K hours.
Resilience	What are the resiliency requirements? (recovering from a fault or sub-fault)	Fault recovery is ok if it doesn't affect production
Confidentiality	What happens if information within the system leaks (or is pulled) out?	intellectual property losses. Recommended encryption: 128-bit and higher (AES)
Integrity	What happens if the system acts on incorrect data (including software)?	Loss in productivity and work safety on the order of >$1M/month
Availability	What happens if the system or data it generates is not accessible and prepared to function properly when and where needed?	Fault causes loss in productivity.

For the next production run, a new work request and associated workflow have been prepared with a set of resource configurations and schedules. The variances in the previous production run denoted in the KPIs and the energy efficiency index have been converted into a set of target production drivers for the next production run.

Notes – Obtaining information about the real time manufacturing system's capabilities and controlling the behavior of the automation units throughout the multiple physical, cyber and cyber-physical domains involve the use of human interface units, advanced sensing units, actuation units and control and optimization units.

Example Goals – highly energy efficient manufacturing with high quality and timely delivered products, as well as

Systems/Actors

- Manufacturing operations management (MOM) application
- Control and automation system
- Production equipment
- Materials, personnel, and energy handling units

5.3.2 CPS Example – Grain/Produce Monitoring and Delivery

Ingredients with specific characteristics are required for the production of a food product. Food producers and ingredient vendors collaborate to get appropriate ingredients delivered for production. Before shipment, vendors send ingredient samples to a lab for analysis and have the results sent to the food producer. The food producer uses the analysis results to adjust manufacturing plans. The adjustments may include stopping shipments of unacceptable ingredients, determining which food product batch is best to use the ingredients in, and/or modifying the production process for the food production batch that is to use the ingredients.

Since the properties of ingredients can change during transit, they may be monitored via sensors during the shipment. Manufacturing planning may make use of the sensor information if it exists.

The systems that need to interact include Supply Chain and Production Systems. The interactions involve multiple layers of communication systems – sensor communication over mobile network, business-to-business communication, and application-to-application communication. The communication topology may be peer-to-peer or ahub/intermediary-based. Sensors may need to be able to regularly join and adjourn different food producers' networks because trucks used for transporting ingredients likely do not belong to the food producer (may belong to a 3rd party logistics service provider or the grain vendor or farmer).

5.3.2.1 Example Goals

What goals does performing the use case achieve?

Information about variations in the characteristics of input ingredients is available in time for the food producer to reject unacceptable ingredients before shipment and for production planning to modify the food production process to account for ingredient variations.

5.3.2.2 Systems/Actors

- Farmer
- Testing lab
- Trucker/truck
- Container
- Customer

5.3.2.3 High Level Review

Table 10: High Level Review - Grain/Produce Analysis and Monitoring

Application Areas	Does the use case require a system that crosses multiple application areas? If so, how many application areas are included?	YES - Supply chain, manufacturing, transportation, agriculture
Composition **Intersystem Interaction**	Does the use case require the interaction of heterogeneous subsystems?	YES.
	Are there specific requirements caused by the use case interacting with legacy systems	YES, but not explicit (example – existing lab often can only send hardcopy of the data)
Human Interaction	Are humans an important part of the system?	with the monitoring aspects, the lab may employ humans in critical roles. but decision making aspects on the customer (manufacturer side).
Physical Properties	What physical properties are being monitored?	Temperature, humidity/moisture, light levels, time, location, biological, , grain/produce properties.
	What physical properties are being acted upon?	The produce/grain (location, manufacturing process, shipment acceptance)
Volume and Velocity	Describe the size of the datasets being processed and the speed at which it comes into/out of the system.	Not much data. At present. (need more information). Data does need to go through multiple heterogeneous systems. Truck monitoring data could get large.
Computation	Describe the computation effort and processing required to achieve the use	Some on the laboratory (measurement/calculation) side. Maybe some on the process reformulation side.

	case goals	
Aggregation	Describe the requirements to aggregate different data types	Test data needs to be combined. ID and other metadata needs to be combined. Customer specification (ingredient spec) may be created from multiple data sources.
Variability	Is the size of data being generated/used consistent or is there a growth/shrinkage trend?	Consistent.
Error Sensitivity	Describe the sensitivity of the system to errors in the data	Depends on property being measured. Can be HIGH – error can cause large monetary cost. If contaminated could lead to sickness or loss of life.
Certainty	What is the level of uncertainty in the data being generated/processed and the assurance of the resulting actions taken by the system?	Unknown. See error sensitivity. Predictive modeling causes additional uncertainties.
Timeliness	What are the use case timing constraints?	Truck monitoring data – minutes (resolution and latency). Lab turn around – time to send grain/produce to the lab + time for analysis and data transmission. Analysis and data transmission time – minutes to hours
Time synchronization	What are the use case time synchronization requirements?	Truck lab and farm data needs to be synchronized, but requirements are not very hard to meet. Need timestamps.
Physical Location	What are the location requirements of the use case?	Multiple locations. Supplier and OEM customer are possibly separated by large distances. Suppliers might not have good communication access. Truck is mobile dynamic locations. Location data for

		specific produce is important.
Robustness	What are the robustness requirements? (preventing a fault)	Cost of lack of production. High liability cost if something goes wrong (and not monitored). Failure is better than error.
Resilience	What are the resiliency requirements? (recovering from a fault or sub-fault)	Resilience is possible if it meets the other requirements of the use case (especially timing requirements)
Confidentiality	What happens if information within the system leaks (or is pulled) out?	The confidentiality of data is important to protect the manufacturers secret recipe. Sensors as well as data streams need to be protected. Data about produce may be authorized for specific actors. Devices, Farmers, lab staff, truckers/trucking staff, manufacturers staff all have different access needs.
Integrity	What happens if the system acts on incorrect data (including software)?	Misinformation could cause the customer large amounts of harm if the recipe used is dependent on the data from produce/grain measurement results.
Availability	What happens if the system or data it generates is not accessible and prepared to function properly when and where needed?	The manufacturer might not receive critical information about the produce shipment being purchased resulting in additional costs and time delays.

5.4 Black box use cases

The black box use cases will be developed from the CPS examples as key examples are identified.

5.4.1 Detailed analysis

Detailed analysis will be done on carefully selected black box use cases.

5.5 Current CPS Examples and Black Box Use Case

The CPS Examples and Black Box Use Cases will be available from the CPS PWG website as they are developed: http://www.cpspwg.org

6 References

6.1 Reference Architecture

[1] Internet of Things – Architecture (IoT-A), "Final architectural reference model for the IoT v3.0", http://www.iot-a.eu/public/public-documents/d3.1, 2013.

[2] ISO/IEC/IEEE 42010, "Systems and software engineering – architecture description", 2011, http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6129467

[3] Svetlana Nikitina, "Translating qualitative requirements into design choices – evaluating the method proposed in the architectural reference model for the internet of things", ERCIS, 2014.

[4] IEEE P2314, "Standard for an Architectural Framework for the Internet of Things (IoT)", Webinar, June 13, 2014.

[5] [IOT-A], EU IOT-A Terminology http://www.iot-a.eu/public/terminology/copy_of_term

[6] IHMC, http://www.ihmc.us/groups/datkinson/wiki/fcb0e/intelligent_system_autonomy_automation_robots_and_agents.html

[7] ISO 7498-2:1989, Information processing systems — Open Systems Interconnection — Basic Reference Model — Part 2: Security Architecture

[8] ISO TS 19104:2008, Geographic information – Terminology

[9] ISO/IEC 14814:2006, Road transport and traffic telematics — Automatic vehicle and equipment identification — Reference architecture and terminology

[10] ISO/IEC 24760-1:2011 , Information technology — Security techniques — A framework for identity management — Part 1: Terminology and concepts

[11] ISO/IEC 24791-1:2010, Information technology — Radio frequency identification (RFID) for item management — Software system infrastructure — Part 1: Architecture

[12] ISO/IEC 27000:2014, Information technology — Security techniques — Information security management systems — Overview and vocabulary http://standards.iso.org/ittf/PubliclyAvailableStandards/c063411_ISO_IEC_27000_2014.zip .

[13] ISO/IEC DIS 18834-1, RA SOA – Terminology and Concepts

[14] ISO/TS 19129:2009, Geographic information — Imagery, gridded and coverage data framework

[15] ISO/TR 14252:1996, Information technology -- Guide to the POSIX Open System Environment (OSE)

[16] OED, Oxford Dictionary of English, 2nd Edition

[17] Industrial Internet Consortium, http://www.industrialinternetconsortium.org/

6.2 Security Viewpoint

[18] The Framework for Improving Critical Infrastructure Cybersecurity, http://nist.gov/cyberframework/upload/cybersecurity-framework-021214.pdf , Use to guide what components need to be included in a framework intended to address multiple domains, Multiple

[19] NIST Interagency Report 7628 Rev. 1, Guidelines for Smart Grid Cybersecurity, http://nvlpubs.nist.gov/nistpubs/ir/2014/NIST.IR.7628r1.pdf , Use to guide what considerations need to be included in a framework intended to address multiple stakeholders within a single domain, Energy

[20] NIST SP 800-53 Rev. 4, Security and Privacy Controls for Federal Information Systems and Organizations, http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf , Use security controls to guide what CPS framework elements need to be considered and factored in, Multiple

[21] NIST SP 800-82 Rev. 1, Guide to Industrial Control Systems (ICS) Security, http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r1.pdf , Use to further bolster the list of differences between IT and CPS systems, Multiple

[22] ISO/IEC 2700x- Information security management, http://www.iso.org/iso/home/standards/management-standards/iso27001.htm, Use similar to how NIST 800-53 is used, but includes a more international perspective (see NIST 800-53 for a mapping between 800-53 and ISO controls), Multiple

[23] ISA/IEC 62443 Series, Industrial Automation and Control Systems Security, http://isa99.isa.org/ISA99%20Wiki/Master-Glossary.aspx, NOTE: Only the IEC62443-3-3 System Security Requirements and Levels is published (final), but requires membership to access the document. Use Master Glossary to determine common terminology being used across international community regarding industrial automation and control systems, Multiple

[24] Electric Sector Failure Scenarios and Impact Analyses, http://smartgrid.epri.com/doc/NESCOR%20failure%20scenarios09-13%20finalc.pdf, Use to guide creation of additional CPS cybersecurity use cases, Energy

[25] National Infrastructure Protection Plan (NIPP) 2013 - Partnering for Critical Infrastructure Security and Resilience, http://www.dhs.gov/sites/default/files/publications/NIPP%202013_Partnering%20for%20Critical%20Infrastructure%20Security%20and%20Resilience_508_0.pdf, Use to understand the various "operating environments" and how they interconnect across domains - consider the system of systems discusssion; read the Core Tenets to determine what characteristics can be incorporated in the CPS framework to address these Tenets and the following objectives:

•Identify, deter, detect, disrupt, and prepare for threats and hazards to the Nation’s critical infrastructure;
•Reduce vulnerabilities of critical assets, systems, and networks; and
•Mitigate the potential consequences to critical infrastructure of incidents or adverse events that do occur.
Consider how CPS Framework we develop needs to incorporate components that facilitate information sharing., Multiple

[26] Health Insurance Portability and Accountability Act (HIPAA) Security Rule

[27] HIPAA Privacy Rule

[28] Health Information Technology for Economic and Clinical Health (HITECH) Act

[29] 45 CFR Parts 160 and 164 - Modifications to the HIPAA Privacy, Security, Enforcement, and Breach Notification Rules Under the Health Information Technology for Economic and Clinical Health Act and the Genetic Information Nondiscrimination Act; Other Modifications to the HIPAA Rules; Final Rule, http://www.hhs.gov/ocr/privacy/hipaa/understanding/srsummary.html

[30] http://www.hhs.gov/ocr/privacy/hipaa/understanding/summary/index.html

[31] http://www.healthit.gov/sites/default/files/hitech_act_excerpt_from_arra_with_index.pdf (see Title XIII)

[32] http://www.gpo.gov/fdsys/pkg/FR-2013-01-25/pdf/2013-01073.pdf, Use to inform health information and information exchange security and privacy considerations that need to be reflected in the CPS framework, Healthcare and Public Health

[33] Modeling and Simulation for Cyber-Physical System Security Research, Development and Applications, http://prod.sandia.gov/techlib/access-control.cgi/2010/100568.pdf, The Virtual Control System Environment (VCSE) Framework and Architecture (p. 11 of 27) diagrams and descriptions assist in understanding a basic framework that is intended to describe the portions of a CPS that are similar across domains. Can use this to help guide our framework that must cover multiple CPS domains., Multiple

[34] NFPA 3: Recommended Practice for Commissioning of Fire Protection and Life Safety Systems, 2015 Edition, 2015 Edition, http://www.nfpa.org/catalog/category.asp?category_name=Codes+and+Standards&Page=1&src=catalog, Note: Documents require payment - cannot determine relevance without reading, Multiple

[35] NFPA 4: Standard for Integrated Fire Protection and Life Safety System Testing, 2015 Edition, http://www.nfpa.org/catalog/category.asp?category_name=Codes+and+Standards&Page=1&src=catalog, Note: Documents require payment - cannot determine relevance without reading, Multiple

[36] NFPA 1600®: Standard on Disaster/Emergency Management and Business Continuity Programs, 2013 Edition, http://www.nfpa.org/catalog/category.asp?category_name=Codes+and+Standards&Page=1&src=catalog , Note: Documents require payment - cannot determine relevance without reading, Multiple

[37] Whole Building Design Guide - Cybersecurity, http://www.wbdg.org/resources/cybersecurity.php, Use to see an example of how cybersecurity is applied to operational technology and industrial control systems; leverage concepts to guide establishment of cybersecurity framework for CPS broadly, Multiple

[38] Securing government assets through combined traditional security and information technology, http://www.dhs.gov/interagency-security-committee-standards-and-best-practices, Use to see how operational technology (OT) and information technology (IT) security considerations overlap and diverge to help understand unique security considerations for CPS Note: The link is only to the Interagency website; the actual report that is listed in Column A "Title" is not publicly available., Multiple

[39] Basic Concepts and Taxonomy of Dependable and Secure Computing, http://www.landwehr.org/2004-aviz-laprie-randell.pdf, Use to determine common terminology that can be referenced in our Framework publication, Multiple

[40] Homeland Security President Directive - 12 (HSPD-12) Implementation Standards and Testing, http://www.gsa.gov/portal/content/105233

[41] http://www.idmanagement.gov/ficam-testing-program , Use to inform efforts to incorporate credentialing into the Internet of Things (IOT) concept within CPS, Multiple

[42] Cyber Security Research Alliance - Roots of Trust for Cyber Physical Systems, http://cybersecurityresearch.org/

[43] http://cybersecurityresearch.org/documents/Roots_of_Trust_for_Cyber_Physical_Systems_Abstract_-_November_2014.pdf, Use full report (must request such via website; Abstract only is directly available) to leverage CPS taxonomy for CPS PWG report, Multiple

[44] Object Management Group (OMG) Industrial Internet of Things (IIOT), http://www.omg.org/hot-topics/iot-standards.htm, Acknowledge OMG's work in CPS PWG conclusions to express awareness of relevant parallel activity to further the credibility and usefulness of CPS PWG report , Multiple

[45] "An Immunity Based Network Security Risk Estimation", Li Tao, Ser. F Information Sciences 2005 Vol.48 No.5 ff7-578.

[46] http://www.cybersecurityresearch.org/about_us.html

[47] http://cps-vo.org/

[48] https://www.nitrd.gov/nitrdgroups/index.php?title=Cyber_Physical_Systems_(CPS_SSG)#title

[49] http://csrc.nist.gov/projects/privacy_engineering/index.html

[50] http://www.industrialinternetconsortium.org/

[51] http://www.dhs.gov/nstac

[52] "Trusted Computing Group: Where Trust Begins," Trusted Computing Group briefing Sept. 24, 2014, slide 7, http://www.trustedcomputinggroup.org/files/resroucefiles/.

[53] Kushner, David (2013). "The Real Story of Stuxnet" Spectrum.IEEE.Org (North American), Mar 2013, pp 49-53. http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet#

[54] http://arstechnica.com/security/2014/11/stuxnet-worm-infected-high-profile-targets-before-hitting-iran-nukes/

[55] Amit, "The Convergence of Engineering Disciplines in Modern Product Development," accessed 1 December 2014 at https://www.ibm.com/developerworks/community/blogs/invisiblethread/entry/the_convergence_of_engineering_disciplines_in_modern_product_development?lang=en_us, Sept 18, 2013.

[56] Fred B. Schneider, Trust in cyberspace, http://www.nap.edu/catalog/6161/trust-in-cyberspace

[57] Avizienis A., Laprie J.-C., Randell B., Landwehr C., Basic concepts and taxonomy of dependable and secure computing, Dependable and Secure Computing, IEEE Transactions on (Volume:1 , Issue: 1), http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1335465&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D1335465

[58] http://www.inl.gov/technicalpublications/documents/4680346.pdf, page 16

[59] http://web.ornl.gov/sci/electricdelivery/pdfs/ORNL_Cybersecurity_Through_Real-Time_Distributed_Control_Systems.pdf

[60] S. M. Amin, "U.S. electrical grid gets less reliable," IEEE Spectrum, p. 80, January 2011, http://dl.acm.org/citation.cfm?id=2244627

[61] http://tools.ietf.org/pdf/rfc4949.pdf

[62] http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4577833&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D4577833

[63] http://www.deviceauthority.com/solutions/m2m-authentication-for-government-and-iot

[64] http://www.utdallas.edu/~alvaro.cardenas/papers/NordSec2013.pdf

[65] http://www.usatoday.com/story/opinion/2014/02/20/christine-todd-whitman-chemicals/5612695/

[66] https://www.priv.gc.ca/information/research-recherche/2014/wc_201401_e.pdf

[67] http://www.marketresearchreports.biz/analysis-details/wearable-technology-market-global-scenario-trends-industry-analysis-size-share-and-forecast-2012-2018

6.3 Data Integration Viewpoint

[68] ISO/IEC CD 11179-1 Information Technology -- Metadata Registries (MDR) - Part 1: Framework Ed 3

[69] Satisfiability. (n.d.). Retrieved October, 2014, from http://en.wikipedia.org/wiki/Satisfiability

[70] Hodges, Wilfrid, "Model Theory", The Stanford Encyclopedia of Philosophy (Fall 2013 Edition), Edward N. Zalta (ed.). Retrieved October, 2014, from http://plato.stanford.edu/archives/fall2013/entries/model-theory.

[71] ISO 42010:2011

[72] http://www.rickmurphy.org/gag-modest.zip

[73] Package Java Util. Function. (2014). Retrieved October, 2014, from http://docs.oracle.com/javase/8/docs/api/java/util/function/package-summary.html

[74] Class LambdaMetafactory. (2014). Retrieved October, 2014, from http://docs.oracle.com/javase/8/docs/api/java/lang/invoke/LambdaMetafactory.html

[75] http://www.rickmurphy.org/gag-modest.zip

[76] OCaml. (2014). Retrieved October, 2014, from http://ocaml.org/

[77] The Scala Programming Language. (2014, January 1). Retrieved October, 2014, from http://www.scala-lang.org/

[78] Duggal, D. (2014, August 14). Semantic SOA makes Sense! Retrieved October, 2014, from http://www.dataversity.net/semantic-soa-makes-sense/

[79] Kahn, R., & Wilensky, R. (2006). A Framework for Distributed Digital Object Services. International Journal on Digital Libraries, 6(2), 115-123. Retrieved October, 2014, from http://link.springer.com/article/10.1007/s00799-005-0128-x?no-access=true

[80] Institute of Electrical and Electronics Engineers, IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries, New York, NY: 1990.

[81] http://en.wikipedia.org/wiki/Software_assurance Software assurance. (n.d.). Retrieved October 7, 2014, from http://en.wikipedia.org/wiki/Software_assurance

[82] http://en.wikipedia.org/wiki/Evaluation_Assurance_Level Evaluation Assurance Level. (n.d.). Retrieved October 7, 2014, from http://en.wikipedia.org/wiki/Evaluation_Assurance_Level

[83] http://en.wikipedia.org/wiki/National_Information_Assurance_Glossary Hall, David L. and James Llinas (1997). “An introduction to multisensory data fusion,” Proceedings of the IEEE, vol. 8, no. 1, pp. 6-23.

[84] Bizer, Christian, Tom Heath and Tim Berners-Lee (2009). “Linked data – The story so far,” in Heath, T., Hepp, M., and Bizer, C. (eds.). Special Issue on Linked Data, International Journal on Semantic Web and Information Systems (IJSWIS), vol. 5, no. 2, pp. 1-22.

[85] JDL (1991). “Data Fusion Lexicon,” Technical Panel For C3, F.E. White, San Diego, Calif: Code 4.

[86] Castanedo, Federico (2013). “A Review of Data Fusion Techniques,” The Scientific World Journal, Volume 2013, Article ID 704504, accessed 3 October 2014 at http://www.hindawi.com/journals/tswj/2013/704504/#B39.

[87] "Framework for discovery of identity management information" [available free of charge at http://www.itu.int/rec/T-REC-X.1255-201309-I; ITU-T announcement: http://newslog.itu.int/archives/137] was approved at an International Telecommunication Union (ITU) meeting in Geneva (ITU-T Study Group 17 (Security)) on September 4, 2013.

[88] “Overview of the Digital Object Architecture,” http://www.cnri.reston.va.us/papers/OverviewDigitalObjectArchitecture.pdf.

[89] Lyons, Patrice A. and Kahn, Robert E., "The Handle System and its Application to RFID and the Internet of Things,” RFIDs, Near-Field Communications and Mobile Payments; A Guide for Lawyers, edited by Sarah Jane Hughes, ABA Cyberspace Law Committee, 2013, pp. 257-270 (http://hdl.handle.net/4263537/5046).

[90] Berners-Lee, T., James Hendler, and O. Lassila, 2001. “The Semantic Web.” Scientific American, May, 29-37

[91] Berners-Lee, Tim, Christian Bizer, and Tom Heath. "Linked data-the story so far." International Journal on Semantic Web and Information Systems 5.3 (2009): 1-22.

[92] Ian Jacobs; Norman Walsh. Architecture of the World Wide Web, Volume One. 15 December 2004. W3C Recommendation. URL: http://www.w3.org/TR/webarch/

[93] Richard Cyganiak, David Wood, Markus Lanthaler. RDF 1.1 Concepts and Abstract Syntax. W3C Recommendation, 25 February 2014. URL: http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/. The latest edition is available at http://www.w3.org/TR/rdf11-concepts/

[94] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition). 11 December 2012. W3C Recommendation. URL: http://www.w3.org/TR/owl2-overview/ www.w3.org/TR/2012/REC-owl2-overview-20121211/

[95] Fabien Gandon; Guus Schreiber. RDF 1.1 XML Syntax. 9 January 2014. W3C Proposed Edited Recommendation. URL: http://www.w3.org/TR/rdf-syntax-grammar/

[96] W3C SPARQL Working Group. SPARQL 1.1 Overview. 21 March 2013. W3C Recommendation. URL: http://www.w3.org/TR/sparql11-overview/

[97] ISO/IEC/IEEE P21451-1-4

[98] http://www.dtic.mil/dtic/tr/fulltext/u2/680815.pdf

[99] http://en.wikipedia.org/wiki/Evaluation_Assurance_Level

[100] https://www.niap-ccevs.org/

[101] http://en.wikipedia.org/wiki/Common_Criteria

[102] ISO/IEC 2382-1:1993, definition 01.01.02

[103] Jamshidi, M. 2005, Theme of the IEEE SMC 2005, Waikoloa, Hawaii, USA, http://ieeesmc2005.unm.edu, Oct 2005.

[104] ISO/TS 8000 Data Quality

- ISO/TS 8000-1:2011, Data quality — Part 1: Overview
- ISO 8000-2:2012, Data quality — Part 2: Vocabulary
- ISO/TS 8000-100:2009, Data quality — Part 100: Master data: Exchange of characteristic data: Overview
- ISO 8000-102:2009, Data quality — Part 102: Master data: Exchange of characteristic data: Vocabulary
- ISO 8000-110:2009, Data quality — Part 110: Master data: Exchange of characteristic data: Syntax, semantic encoding, and conformance to data specification
- ISO/TS 8000-120:2009, Data quality — Part 120: Master data: Exchange of characteristic data: Provenance
- ISO/TS 8000-130:2009, Data quality — Part 130: Master data: Exchange of characteristic data: Accuracy
- ISO/TS 8000-140:2009, Data quality — Part 140: Master data: Exchange of characteristic data: Completeness
- ISO/TS 8000-150:2011, Data quality — Part 150: Master data: Quality management framework

[105] ISO 22745, Open technical dictionaries and their application to master data

- Part 1: Overview and fundamental principles
- Part 2: Vocabulary
- Part 10: Dictionary representation
- Part 11: Guidelines for the formulation of terminology
- Part 13: Identification of concepts and terminology
- Part 14: Dictionary query interface
- Part 20: Procedures for the maintenance of an open technical dictionary

- Part 30: Identification guide representation (data specification)
- Part 35: Query for characteristic data
- Part 40: Master data representation

[106] ISO 29002, Exchange of characteristic data

[107] ISO 3534-2

[108] http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

[109] P21451-1-4, Standard for a Smart Transducer Interface for Sensors, Actuators, and Devices - eXtensible Messaging and Presence Protocol (XMPP) for Networked Device Communication, active project

[110] Department of Homeland Security, CUSTOMS AND BOARDER PROTECTION, Container Security Initiative (CSI), http://www.cbp.gov/border-security/ports-entry/cargo-security/csi/csi-brief

[111] C-TPAT (Customs Trade Partnership Against Terrorism), http://www.cbp.gov/border-security/ports-entry/cargo-security/c-tpat-customs-trade-partnership-against-terrorism

[112] Bonded Warehouse Manual for CBP Officers and Bonded Warehouse Proprietors, http://www.cbp.gov/document/guidance/bonded-warehouse-manual-cbp-officers-and-bonded-warehouse-proprietors

[113] Amendment to the Current Reporting Requirements for the Ultimate Consignee at the Time of Entry or Release, http://www.cbp.gov/border-security/ports-entry/cargo-security/cargo-control/ult-consignee

[114] EN ISO 19115-1:2014, Geographic information -- Metadata -- Part 1: Fundamentals

[115] Richard C. Murphy, “NIST Cyber Physical Systems Working Group Data Interop Contributions”,http://www.rickmurphy.org/data-interop.html

[116] ISO/IEC 16500-8:1999, Information technology -- Generic digital audio-visual systems -- Part 8: Management architecture and protocols

[117] Kahn, Robert E. and Lyons, Patrice A., "Representing Value as Digital Objects: A Discussion of Transferability and Anonymity". *Journal on Telecommunications & High Technology Law*, Vol. 5, Issue 1, 189 (2006).

[118] International Carrier Bonds for Non-Vessel Operating Common Carriers (NVOCCs), http://www.cbp.gov/border-security/ports-entry/cargo-security/cargo-control/carrier-bonds

[119] http://en.wikipedia.org/wiki/Semantic_Web_Stack

[120] Turnitsa, C.D. (2005). Extending the Levels of Conceptual Interoperability Model. Proceedings IEEE Summer Computer Simulation Conference, IEEE CS Press, see http://en.wikipedia.org/wiki/Conceptual_interoperability

[121] NISO, (2004)" Understanding metadata", Bethesda, MD: NISO Press, p1, http://www.niso.org/standards/resources/UnderstandingMetadata.pdf

[122] Models as a Basis for Ontologies, Ed Barkmeyer, NIST, Ontolog Forum, April, 2007. http://ontolog.cim3.net/cgi-bin/wiki.pl?ConferenceCall_2007_04_12

[123] RFC3444 On the Difference between Information Models and Data Models, http://www.rfc-editor.org/rfc/rfc3444.txt

[124] http://www.dona.net

[125] http://en.wikipedia.org/wiki/Satisfiability

[126] ANSI C12.19-2008 American National Standard For Utility Industry End Device Data Tables, American National Standards Institute, Inc., February 24, 2009

[127] A Universally Unique IDentifier (UUID) URN Namespace, Leach, Mealling & Salz, Julu 2005, http://www.ietf.org/rfc/rfc4122.txt

[128] IEC 62541 Series-OPC Unified Architecture (OPC-UA), Version 1.01, Released,2009-02-09, http://www.opcfoundation.org/ua

[129] Hadoop, http://hadoop.apache.org/

[130] ISO/IEC 18004:2006 Information technology – Automatic identification and data capture techniques – QR code 2005 bar code symbology specification, *http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=43655*

[131] RFC 6282, Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks, Hui & Thubert, September 2011, https://tools.ietf.org/html/rfc6282

[132] SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), WC3 Recommendation 27 April 2007, http://www.w3.org/TR/soap12-part1/

[133] Architectural Styles and the Design of Network-based Software Architectures, Fielding, 2000, http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

[134] Trusted Computing Group, TNC Architecture for Interoperability , Revision 1.5 Rev 3, May 2012, http://www.trustedcomputinggroup.org/files/resource_files/2884F884-1A4B-B294-D001FAE2E17EA3EB/TNC_Architecture_v1_5_r3-1.pdf

[135] TNC IF-MAP Binding for SOAP, Specification version 2.2 rev 9, 26 Mar 2014, http://www.trustedcomputinggroup.org/files/static_page_files/FF3CB868-1A4B-B294-D093D8383D733B8A/TNC_IFMAP_v2_2r9.pdf

[136] OASIS Web Services Security (WSS), https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss-m

[137] https://rd-alliance.org/groups/data-type-registries-wg.html

[138] PRIVACY POLICY GUIDANCE MEMORANDUM, Memorandum Number: 2008-01, December 29, 2008, Department of Homeland Security, http://www.dhs.gov/xlibrary/assets/privacy/privacy_policyguide_2008-01.pdf

[139] FIPPs - http://www.dhs.gov/xlibrary/assets/privacy/privacy_policyguide_2008-01.pdf.

[140] Kent, William, updated by Steve Hoberman. "Data & Reality: A Timeless Perspective on Perceiving and Managing Information in Our Imprecise World." Westfield, NJ: Technics Publications, 2012. Print

6.4 Timing Viewpoint

6.4.1 References from Introduction

[141] ITU-R Recommendation TF,686-3 (12/2013) Glossary and Definitions of Time and Frequency Terms available from http://www.itu.int/rec/R-REC-TF.686-3-201312-I/en Note: this document contains references to additional glossary and definition material published by NIST, BIPM, IEC and the ISO.

[142] The time scales UTC and TAI and the International System of Units, SI, are defined and maintained by the International Bureau of Weights and Measures (Bureau International des Poids et Mesures, BIPM),. See http://www.bipm.org

[143] D.B. Sullivan, D.W. Allan, D.A. Howe, and F.L. Walls, “Characterization of Clocks and Oscillators,” NIST Tech. Note 1337, June 1, 1999, available from: http://tf.boulder.nist.gov/general/pdf/868.pdf

[144] Timing Framework for Cyber Physical Systems Technical Annex, <<TBD>>

6.4.2 References from Time-Awareness in CPS

[145] H. Kopetz and G. Bauer. The time-triggered architecture. Proceedings of the IEEE, 91(1):112–126, 2003.

[146] Jasperneite, J.; Feld, J., "PROFINET: an integration platform for heterogeneous industrial communication systems," Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on , vol.1, no., pp.8 pp.,822, 19-22 Sept. 2005

[147] Timing Committee Telecommunications and Timing Group- Range Commanders Council, “IRIG Serial time code formats,” September, 2004. [Online]. Available: http://www.irigb.com/pdf/wp-irig-200-04.pdf

[148] Kaplan, Elliott D., and Christopher J. Hegarty, eds. Understanding GPS: principles and applications. Artech house, 2005.

[149] IEEE Instrumentation and Measurement Society, “1588: IEEE standard for a precision clock synchronization protocol for networked measurement and control sytems” IEEE, Standar Specification, July 24, 2008

[150] K. Harris, “An application of IEEE 1588 to industrial automation,” in Precision Clock Synchronization for Measurement, Control and Communication, 2008, ISPCS. IEEE International Symposium on. IEEE, 2008, pp 71-76

[151] M. Shepard, D. Fowley, R. Jackson, and D. King, “Implementation of IEEE Std-1588 on a Networked I/O Node, “ in Proceedigns of the 2003 Workshop on IEEE-1588, NIST publication NISTIR 7070, Gaithersburg, MD, 2003.

[152] F. Steinhauser, C. Riesch, and M. Ridigier, “IEEE 1588 for time synchronization of devices in the electric power industry,” in Precision Clock Synchronization for Measurement, Control and Communication, 2010, ISPCS. IEEE International Symposium on. IEEE, 2010, pp 1-6

[153] Giorgio C. Buttazzo: Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications, Third Edition. Real-Time Systems Series 24, Springer 2011, ISBN 978-1-4614-0675-4, pp. 1-521

[154] R. Wilhelm, D. Grund: Computation takes time, but how much? Commun. ACM 57(2): 94-103 (2014)

[155] P. Axer, R. Ernst, H. Falk, A. Girault, D. Grund, N. Guan, B. Jonsson, P. Marwedel, J. Reineke, C. Rochange, M. Sebastian, R. von Hanxleden, R. Wilhelm, W. Yi: Building timing predictable embedded systems. ACM Trans. Embedded Comput. Syst. 13(4): 82 (2014)

[156] R. Wilhelm, J. Engblom, A. Ermedahl, N. Holsti, S. Thesing, D.B. Whalley, G. Bernat, C. Ferdinand, R. Heckmann, T. Mitra, F. Mueller, I. Puaut, P. Puschner, J. Staschulat, P. Stenström: The worst-case execution-time problem - overview of methods and survey of tools. ACM Trans. Embedded Comput. Syst. 7(3) (2008)

[157] J. Rushby and W. Steiner, “TTA and PALS: Formally verified design patterns for distributed cyber-physical systems,” in 30th IEEE/AIAA Digital Avionics Systems Conference (DASC), Seattle, WA, 2011.

[158] ARINC, “ARINC 653 family of standards,” November, 2010. [Online]. Available: https://www.arinc.com/cf/store/

[159] Edward A. Lee: Computing needs time. Commun. ACM 52(5): 70-79 (2009)

[160] PHYTER, DP83640 Precision. "IEEE 1588 precision time protocol transceiver." (2008).

[161] Corbett, James C., et al. "Spanner: Google's globally distributed database."ACM Transactions on Computer Systems (TOCS) 31.3 (2013): 8.

[162] Y. Zhao, E.A. Lee, and J. Liu. A programming model for time-synchronized distributed real-time systems. In Real-Time and Embedded Technology and Applications Symposium (RTAS), Bellevue, WA, USA, April 3-6 2007. IEEE.

[163] Broman, David, Patricia Derler, and John Eidson. "Temporal issues in cyber-physical systems." Journal of the Indian Institute of Science 93.3 (2013): 389-402.

6.4.3 References from Time and Latency in CPS

[164] Open Networking Foundation, "OpenFlow Switch Specification," October 14, 2013. [Online]Available from: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf

[165] Paul Congdon, "Link Layer Discovery Protocol Overview (LLDP)," March 8, 2003. [Online] Available from: http://www.ieee802.org/1/files/public/docs2002/LLDP%20Overview.pdf

[166] PROFINET. [Online] presentation at: http://www.profibus.com/technology/profinet/

[167] IETF Network Working Group, "Simple Network Management Protocol (SNMP)". [Online] available at: https://www.ietf.org/rfc/rfc1157.txt

[168] CANopen. [Online] available from: http://www.can-cia.org/index.php?id=canopen

[169] Center for Hybrid and Embedded Software (CHESS), UC Berkeley, "PTIDES". [Online] see: http://chess.eecs.berkeley.edu/ptides/

[170] National Instruments, "LabVIEW System Design Software". [Online] see: http://www.ni.com/labview/

6.4.4 References from Secure and Resilient Time

[171] IEEE Instrumentation and Measurement Society, IEEE 1588-2008 IEEE Standard for Precision Clock Synchronization Protocol for Measurement and Control Systems, 24 July 2008.

[172] NTP: The Network Time Protocol. [Online] http://www.ntp.org/

[173] (1994). SPECIFICATION OF THE TRANSMITTED LORAN-C SIGNAL. U.S Department of Transportation.

[174] ATIS COAST Standards Body, document SYNC-2014-00052R000 from NIST, "CONTRIBUTION TO STANDARDS PROJECT — COAST-SYNC: WWVB for Assisted Timing." John Lowe; Marc Weiss. October 2013.

[175] A.J. Kerns, K.D. Wesson, and T.E. Humphreys, "A Blueprint for Civil GPS Navigation Message Authentication," IEEE/ION PLANS, Monterey, CA, May 2014. [Online] available from: http://radionavlab.ae.utexas.edu/images/stories/files/papers/nmaimpPLANS2014.pdf

[176] Johnson, G. S. (2007). An Evaluation of eLoran as a Backup to GPS. *Technologies for Homeland Security, 2007 IEEE Conference on* (pp. 95-100). Woburn, MA: IEEE.Satisfiability. (n.d.). Retrieved October, 2014, from http://en.wikipedia.org/wiki/Satisfiability

[177] A. Pearson and K. Shenoi. “A Case for Assisted Partial Timing Support Using Precision Timing Protocol Packet Synchronization for LTE-A,” *IEEE Communications Magazine,* August 2014, pp. 136-143.

[178] T. Mizrahi, RFC 7384: Security Requirements of Time Protocols in Packet-Switched Networks. https://www.rfc-editor.org/rfc/rfc7384.txt

[179] T. Mizrahi, Time synchronization security using IPsec and MACsec, International IEEE Symposium on Precision Clock Synchronization for Measurement Control and Communication (ISPCS), 2011

[180] *D. Sibold, S. Roettger, K. Teichel*, “Network Time Security”, October 2014. https://tools.ietf.org/html/draft-ietf-ntp-network-time-security-05

[181] *D. Sibold et al.*, “Protecting Network Time Security Messages with the Cryptographic Message Syntax (CMS)”, October 2014, https://tools.ietf.org/html/draft-ietf-ntp-cms-for-nts-message-00

[182] *IEEE 1588 Working Group Website.* https://ieee-sa.centraldesktop.com/1588public/ 20 Nov. 2013.

[183] Caverly, R.J. “GPS Critical Infrastructure: Usage/Loss Impacts/Backups/Mitigation”, April 27, 2011. http://www.swpc.noaa.gov/sww/sww11/SWW_2011_Presentations/Wed_830/GPS-PNTTimingStudy-SpaceWeather4-27.pptx

[184] Kappenman, J. “Geomagnetic Storms and Their Impacts on the U.S. Power Grid.” January 2010. http://web.ornl.gov/sci/ees/etsd/pes/pubs/ferc_Meta-R-319.pdf

[185] The MITRE Corporation. “Detection, Localization, and Mitigation Technologies for Global Positioning System (GPS) Jamming and Spoofing (Final)”. *Redacted for Public Release.* February 2014.

[186] Jaldehag, K., Ebenhag, S., Hedekvist, P., Rieck, C., and Lothberg, P. “Time and Frequency Transfer Using Asynchronous Fiber Optical Networks: Progress Report,” *Proceedings of 41st Annual Precise Time and Time Interval (PTTI) Meeting*, 2009.

[187] R. Cohen, PTP Security Tutorial, International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS), 2007

[188] A. Treytl, G. Gaderer, B. Hirschler, Traps and pitfalls in secure clock synchronization, International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS), 2007

[189] A. Treytl, B. Hirschler, Validation and Verification of IEEE 1588 Annex K, International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS), 2011

[190] C. Önal and H. Kirrmann, Security improvements for IEEE 1588 Annex K, International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS), 2012

[191] S. Röttger, Analysis of the NTP Autokey Extension (in German), University of Braunschweig and Physikalisch-Technische Bundesanstalt Braunschweig, 2011

[192] A. Treytl, B. Hirschler, Security Flaws and Workarounds for IEEE 1588 (Transparent) Clocks, International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS), 2009

[193] A. Treytl, B. Hirschler, Practical Application of 1588 Security, International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication (ISPCS), 2008

[194] RFC 6066. Lnternet Engineering Task Force (IETF) Transport Layer Security (TLS) Extensions: Extension Definitions. January 2011. https://tools.ietf.org/html/rfc6066

[195] J. Tournier, O. Goerlitz, Strategies to Secure the IEEE 1588 Protocol in Digital Substation Automation, Fourth International Conference on Critical Infrastructures (CRIS), 2009

[196] Daniel P. Shepard, D.P.; Humphreys, T.E., Fansler, A.A. "Going Up Against Time: The Power Grid's Vulnerability to GPS Spoofing Attacks." GPS World, August 2012.

[197] D. P. Shepard, J. A. Bhatti, T. E. Humphreys, Evaluation of Smart Grid and Civilian UAV Vulnerability to GPS Spoofing Attacks, slide no. 11, September 21, 2012.

[198] "Time Anomaly Detection Applique", 2013. http://www.mitre.org/research/technology-transfer/technology-licensing/time-anomaly-detection-appliqu%C3%A9-tada

[199] Langley, R.B. "Innovation: GNSS Spoofing Detection." GPS World. http://gpsworld.com/innovation-gnss-spoofing-detection-correlating-carrier-phase-with-rapid-antenna-motion/

[200] Pearson, T. and Shenoi, K. "A Case for Assisted Partial Timing Support Using Precision Timing Protocol Packet Synchronization for LTE-A." *IEEE Communications Magazine,* 52 (8), August 2014, pp. 135-143.

[201] Amelot, J., Li-Baboud, Y., Vasseur, C., Fletcher, J., Anand, D., and Moyne, J. "An IEEE 1588 Performance Testing Dashboard for Power Industry Requirements," *Proceedings of International IEEE Symposium on Precision Clock Synchronization for Measurement Control and Communication (ISPCS),* pp. 132-137, 12-16 Sept. 2011.

[202] Crain, A. and Sistrunk, C. Advisory (ICSA-13-210-01). https://ics-cert.us-cert.gov/advisories/ICSA-13-219-01

[203] E. O. Schweitzer, E.O; Guzmán, A. "Real-Time Synchrophasor Applications for Wide-Area Protection, Control, and Monitoring." © 2009 by Schweitzer Engineering Laboratories, Inc.

[204] http://www.bpa.gov/news/newsroom/Pages/Synchrophasor-success-lands-B...

[205] Goldstein, A. Email to the CPS_Sync list dated 16 SEPT 2014.

[206] http://www.microsemi.com/products/timing-synchronization-systems/time-frequency-distribution/gps-instruments/xli-saasm ; also Symmetricom's GPS Disciplined Master Timing Reference (ATS 6501B). Warriner, J., private communication on 21 February 2014.

[207] *I. Fernández Hernández, "*Design Drivers, Solutions and Robustness Assessment of Navigation Message Authentication for the Galileo Open Service," ION GNSS+ 2014, Tampa, FL, September 2014.

[208] *J.T. Curran, M. Paonni, J. Bishop, "*Securing GNSS: An End-to-end Feasibility Analysis for the Galileo Open-service," ION GNSS+ 2014, Tampa, FL, September 2014.

[209] *Factsheet: National Risk Estimate: Risks to U.S. Critical Infrastructure from Global Position System Disruptions*, June 2013. http://www.gps.gov/news/2013/06/2013-06-NRE-fact-sheet.pdf

[210]

6.4.5 General Timing Definitions and Related Standards

[211] ITU-R Recommendation TF,686-3 (12/2013) Glossary and Definitions of Time and Frequency Terms available from http://www.itu.int/rec/R-REC-TF.686-3-201312-I/en Note: this document contains references to additional glossary and definition material published by NIST, BIPM, IEC and the ISO.

[212] All ITU-T published recommendations can be downloaded from: http://www.itu.int/rec/T-REC-G/e

We list ITU-T Published Recommendations associated with timing in telecom networks.

[213] ITU-T Published Recommendations (PDH/SDH)

- ITU-T Recommendation G.803, Architecture of transport networks based on the synchronous digital hierarchy (SDH).

- ITU T Recommendation G.810, Definitions and terminology for synchronization networks.
- ITU T Recommendation G.811, Timing characteristics of primary reference clocks.
- ITU T Recommendation G.812, Timing requirements of slave clocks suitable for use as node clocks in synchronization networks.
- ITU T Recommendation G.813, Timing characteristics of SDH equipment slave clocks (SEC).
- ITU-T Recommendation G.823, The control of jitter and wander within digital networks which are based on the 2048 kbit/s hierarchy
- ITU-T Recommendation G.824, The control of jitter and wander within digital networks which are based on the 1544 kbit/s hierarchy
- Recommendation ITU-T G.825, The control of jitter and wander within digital networks which are based on the synchronous digital hierarchy (SDH)

[214] ITU-T Published Recommendations (Packet Sync - Frequency)

- ITU T Recommendation G.8261, Timing and synchronization aspects in packet networks.
- ITU T Recommendation G.8262, Timing characteristics of Synchronous Ethernet Equipment slave clock (EEC).
- ITU T Recommendation G.8264, Distribution of timing through packet networks
- Recommendation ITU-T G.8261.1, Packet Delay Variation Network Limits applicable to Packet Based Methods (Frequency Synchronization).
- Recommendation ITU-T G.8263, Timing Characteristics of Packet based Equipment Clocks (PEC) and Packet based Service Clocks (PSC)
- ITU-T Recommendation G.8265), Architecture and requirements for packet based frequency delivery
- ITU-T Recommendation G.8265.1, Precision time protocol telecom profile for frequency sync
- ITU-T Recommendation G.8260, Definitions and terminology for synchronization in packet networks

[215] ITU-T Consented Recommendations (Packet Sync – Phase/Time)

- ITU T Recommendation G.8271, Time and phase synchronization aspects of packet networks
- ITU T Recommendation G.8272, Timing characteristics of Primary reference time clock
- ITU T Recommendation G.8271.1 , Network limits
- ITU T Recommendation G.8272, Primary Reference Timing Clock (PRTC) specification

- ITU T Recommendation G.8273, Clock General Requirements
- ITU T Recommendation G.8273.2 , Telecom Boundary Clock specification
- ITU T Recommendation G.8275 , Architecture for time transport
- ITU T Recommendation G.8275.1 , IEEE-1588 profile for time with full support from the network

6.5 Usage Viewpoint

[216] Dictionary.com

[217] Customer Communications Architecture Development: Metrics for Standards and Product Assessment. EPRI, Palo Alto, CA 94303, 20-Dec-2011, Product 1021945