Sessions

1. Registration
2. Welcome (Greer)
3. Goals (Griffor)
4. Keynote (Ross)
5. CPS Framework Overview (Wollman)
6. CPS Framework Applications
   1. Math (Griffor)
   2. Transportation (McShane, Brandao)
   3. IES City (Burns)
   4. Security to Trustworthiness (Vishik)
   5. Ontology (Balduccini)
7. Panel Discussion (Greer)
8. Systems Engineering and CPS Framework (Roth)
9. Modeling (Burns/Song)
10. Community Building (Griffor)
2. Welcome - Greer

NIST and the Smart Grid and Cyber-Physical Systems Program Office – CPS Program

CPS Framework Open Source: Continuous Integration for CPS Development
3. Workshop Goals - Griffor

This workshop aims to address key CPS challenges: how we conceive, design, build, deliver and maintain them.

1. What is CPS?
2. How do we design, build and assure CPS throughout their lifecycle?
3. What discipline do we need to address the concerns that drive requirements and engineering?
4. What needs to be the common core tooling?
3.1 What is CPS?

Cyber-Physical Systems (CPS) comprise interacting digital, analog, physical, and human components engineered for function through integrated logic and physics.

Internet of Things (IoT) emphasizes digital infrastructure for widely connected, interacting, physical ‘things,’ forming systems that integrate logic and physics for function.

NIST Smart Grid and Cyber-Physical Systems Program Office
3.2 How do we design, build and test CPS?

- Develop requirements.
- Specify the system, sub-systems and components.
- Build components.
- Unit test components.
- Assemble and test sub-systems.
- Assemble and test/validate full system.
3.3 What discipline do we need to address the concerns?

Concern Structure:
- Develop a full set of concerns.
- Develop the relationships between the concerns.

Systems Engineering Activities:
- Determine requirements needed to address each concern.
- Design, build and test to each set. (composition of concerns).
- Build the Assurance Case
3.4 What needs to be the common core tooling?

CPS Framework Open Source provides:

1) ‘Type Structure’ for:
   • Aspects and concern; and
   • Facets, engineering activities and outcomes

2) That type and sort compositionally:
   • properties/requirements and
   • artifacts

3) **Encoded** in a portable, reusable XML format.
3.5 Expanded Concern Risk and Risk Mitigation Surface

<table>
<thead>
<tr>
<th>Primary Impact of Failure</th>
<th>Mitigation Mechanisms</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT System</td>
<td>CPS</td>
</tr>
<tr>
<td>Digital</td>
<td>Physical</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

“E.g. Better cybersecurity through physics!”
4. Achieving Trustworthy Systems - Ross
Rethinking Cybersecurity from the Inside Out

An Engineering and Life Cycle-Based Approach for Achieving Trustworthy Secure Systems

Dr. Ron Ross
Computer Security Division
Information Technology Laboratory
Our appetite for *advanced technology* is rapidly exceeding our ability to protect it.
Complexity.
The n+1 vulnerabilities problem.
Reducing susceptibility to *cyber threats* requires a multidimensional systems engineering approach.

- **Harden the target**
- **Limit damage to the target**
- **Make the target survivable**

Achieving Trustworthiness and Resiliency

Security Architecture and Design
Security.

An emergent property.
Risk assessment.
Assets and consequences.
Systems Security Engineering

Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems
Multidisciplinary integration of security best practices.
Systems and software engineering
— System life cycle processes

Technical Processes

- Business or mission analysis
  - Stakeholder needs and requirements definition
    - System requirements definition
    - Architecture definition
    - Design definition
    - System analysis
    - Implementation
    - Integration
    - Verification
  - Transition
  - Validation
- Operation
- Maintenance
- Disposal
Nontchnical Processes

- Project planning
- Project assessment and control
  - Decision management
  - Risk management
    - Configuration management
    - Information management
      - Measurement
      - Quality assurance
      - Acquisition and Supply
    - Life cycle model management
  - Infrastructure management
  - Portfolio management
  - Human resource management
  - Quality management
  - Knowledge management

Systems and software engineering
— System life cycle processes
Systems and software engineering
— System life cycle processes

Nontchnical Processes

- Project planning
- Project assessment and control
  - Decision management
  - Risk management
    - Configuration management
    - Information management
    - Measurement
    - Quality assurance
    - Acquisition and Supply
  - Life cycle model management
  - Infrastructure management
  - Portfolio management
  - Human resource management
  - Quality management
  - Knowledge management
Appendices

A Wealth of Trusted Systems Development
Principles, Concepts, and Best Practices

- References
- Glossary
- Acronyms
- Summary of Security Activities / Tasks
- Roles, Responsibilities, and Skills
- Design Principles for Security
- Engineering and Security Fundamentals
Security should be a by-product of good design and development practices—integrated throughout the system life cycle.
Institutionalize.

The ultimate objective for security.

Operationalize.
Security is a team sport.
NIST Systems Security Engineering Project

Race to the Top — Better Security Through Engineering
Ron Ross

100 Bureau Drive  Mailstop 7730
Gaithersburg, MD USA 20899-7730

Email
ron.ross@nist.gov

Mobile
301.651.5083

LinkedIn
www.linkedin.com/in/ronross-cybersecurity

Twitter
@ronrossecure

Web
csrc.nist.gov

Comments
sec-cert@nist.gov
5. CPS Framework Review-Wollman
5.1 Frameworks – NIST Convening of Stakeholders

- **Frameworks**: documented conceptual structures that organize and make clear collective wisdom (vision, principles, underlying structure, functions, requirements, …)
  - Frameworks are created with technical expertise and consensus-based process

- Perspectives, Viewpoints, Views, …
- Communities of practice, processes, …
5.2 Frameworks – NIST Convening of Stakeholders

NIST Special Publication 1108r3
NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 3.0

Smart Grid and Cyber-Physical Systems Program Office
and Energy and Environment Division, Engineering Laboratory

Priority Action Plans (PAPs)

Framework for Improving Critical Infrastructure Cybersecurity

Version 1.0
National Institute of Standards and Technology

Function | Category | ID | 2014
--- | --- | --- | ---
Identify | Asset Management | ID.AM | 
| Business Environment | ID.BE | 
| Governance | ID.GV | 
| Risk Assessment | ID.RA | 
| Risk Management Strategy | ID.RM | 
Protect | Access Control | PR.AC | 
| Awareness and Training | PR.AT | 
| Data Security | PR.DS | 
| Information Protection Processes & Procedures | PR.IP | 
| Maintenance | PR.MA | 
| Protective Technology | PR.PT | 

NIST Special Publication 1500-4
NIST Big Data Interoperability Framework: Volume 4, Security and Privacy

Smart Grid – EISA SG Legislation
Dec 2007

Cloud Computing
2010

Executive Order – Feb 2013

Cybersecurity

Community Disaster Resilience
June 2013

Climate Action Plan

Big Data
June 2013

Smart America/Global Cities

Cyber-Physical Systems
June 2014
5.3 NIST CPS Public Working Group

- **Goal:** create CPS Framework to support CPS research, development and deployment (applicable to CPS and Internet of Things IoT)
- **Need:** multi-domain perspective baked in
  - Applicable within all CPS domains, supports cross-CPS domain applications

**CPS Framework**

- Smart infrastructure (grid, water, gas, …)
- Smart buildings
- Smart emergency response
- Smart healthcare
- Many more!!
- Smart manufacturing
- Smart transportation
5.4 NIST CPS Public Working Group

NIST CPS PWG leadership: David Wollman and Chris Greer

Co-Chairs | Reference Arch | Use Cases | Security | Timing | Data Interop
---|---|---|---|---|---
NIST | Abdella Battou, Ed Griffor | Eric Simmon | Vicky Pillitteri, Steve Quinn | Marc Weiss | Marty Burns
Academia | Janos Sztipanovits | John Baras | Bill Sanders | Hugh Melvin | Larry Lannom
Industry | Stephen Mellor, Shi-Wan Lin | Stephen Mellor | Claire Vishik | Sundeep Chandhoke | Peggy Irelan, Eve Schooler

NIST SP 1500-201 and 1500-202

Framework for Cyber-Physical Systems

Release 1.0

May 2016

Cyber Physical Systems Public Working Group

pages.nist.gov/cpspwg
5.5 CPS Framework Development

CPS Stakeholders (Societal, Business & Technical) → Raw CPS concerns → Aspects and Concerns → CPS Framework

- Conceptualization Facet: Model of CPS
- Realization Facet: Instance of CPS
- Assurance Facet: CPS Assurance

CPS Framework → Assured CPS

What things should be and what things are supposed to do

How things should be made and operate

How to prove things actually work the way they should
5.6 NIST CPS PWG – CPS Framework

‘Concern-driven’: holistic, integrated approach to CPS/IoT concerns.

- CPS Framework Release 1.0 (May 2016) available at https://pages.nist.gov/cpspwg/
5.7 Purpose of the CPS Framework

• **Concern-driven structuring of development artifacts:** to facilitate assurance cases (by representing or analyzing a system along these dimensions, points of commonality or interoperability with other systems are revealed)

• **A normal-form for CPS/IoT system** (common way of presenting CPS/IoT that enables comparison of what is done, across the system, for the sake of any individual concern)

• Provides a **method for integrating CPS/IoT across domains** – the future of CPS/IoT is cross-domain integration. While some domains may have robust, integrated approaches to some concerns, there are typically radically different standards across domains.

CPS Framework is NOT A PROCESS!!
It is a method for integrating concerns into systems engineering processes!
5.8 CPS Framework Structure

Facets

<table>
<thead>
<tr>
<th>Conceptualization</th>
<th>Realization</th>
<th>Assurance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case, …</td>
<td>Design / Produce / Test / Operate</td>
<td>Argumentation, Claims, Evidence</td>
</tr>
</tbody>
</table>

Activities and their Artifacts

- Model of a CPS
- CPS
- CPS Assurance

Domains

- Manufacturing
- Transportation
- Energy
- Healthcare
- others …

Aspects

- Functional
- Business
- Human
- Trustworthiness
- Timing
- Data
- Boundaries
- Composition
- Lifecycle
5.9 CPS Framework Structure

Facets

<table>
<thead>
<tr>
<th>Conceptualization</th>
<th>Realization</th>
<th>Assurance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case, …</td>
<td>Design / Produce / Test / Operate</td>
<td>Argumentation, Claims, Evidence</td>
</tr>
</tbody>
</table>

Activities and their Artifacts

- Model of a CPS
- CPS
- CPS Assurance

Aspects

- Functional
- Business
- Human
- Trustworthiness
- Timing
- Data
- Boundaries
- Composition
- Lifecycle

Domains

- Manufacturing
- Transportation
- Energy
- Healthcare
- Others …
5.10 Aspects (groupings/categories of concerns)

<table>
<thead>
<tr>
<th>Aspects</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Functional</strong></td>
<td>Concerns about function including sensing, actuation, control, communications, physicality, etc.</td>
</tr>
<tr>
<td><strong>Business</strong></td>
<td>Concerns about enterprise, time to market, environment, regulation, cost, etc.</td>
</tr>
<tr>
<td><strong>Human</strong></td>
<td>Concerns about human interaction with and as part of a CPS.</td>
</tr>
<tr>
<td><strong>Trustworthiness</strong></td>
<td>Concerns about trustworthiness of CPS including security/cybersecurity, privacy, safety, reliability, and resilience.</td>
</tr>
<tr>
<td><strong>Timing</strong></td>
<td>Concerns about time and frequency in CPS, including the generation and transport of time and frequency signals, timestamping, managing latency, timing composability, etc.</td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>Concerns about data interoperability including fusion, metadata, type, identity, etc.</td>
</tr>
<tr>
<td><strong>Boundaries</strong></td>
<td>Concerns related to demarcations of topological, functional, organizational, or other forms of interactions.</td>
</tr>
<tr>
<td><strong>Composition</strong></td>
<td>Concerns related to the ability to compute selected properties of a component assembly from the properties of its components. Compositionality requires components that are composable: they do not change their properties in an assembly. Timing composability is particularly difficult.</td>
</tr>
<tr>
<td><strong>Lifecycle</strong></td>
<td>Concerns about the lifecycle of CPS including its components.</td>
</tr>
</tbody>
</table>
## 5.11 CPS Framework Structure

### Domains
- Manufacturing
- Transportation
- Energy
- Healthcare
- Others ...

### Aspects
- Functional
- Business
- Human
- Trustworthiness
- Timing
- Data
- Boundaries
- Composition
- Lifecycle

### Facets
<table>
<thead>
<tr>
<th>Conceptualization</th>
<th>Realization</th>
<th>Assurance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case, Requirements, …</td>
<td>Design / Produce / Test / Operate</td>
<td>Argumentation, Claims, Evidence</td>
</tr>
</tbody>
</table>

### Activities and their Artifacts
- Model of a CPS
- CPS
- CPS Assurance
5.12 Activities and Artifacts

In using the framework to analyze and document CPS, a series of activities is performed. For example, a typical waterfall-like process will include:

• use case development
• functional decomposition
• requirements analysis
• design
• etc.

An activity produces one or more artifacts.

For example, the activities and associated artifacts of the conceptualization facet commonly include:

Mission and Business Case Development
Artifact: Business use cases

Functional Decomposition
Artifact: Detailed use cases, actors, information exchanges

Requirements Analysis
Artifact: Functional and non-functional requirements

Requirements Allocation
Artifact: HW/SW configuration items

Interface Requirements Analysis
Artifact: Interface requirements
5.13 Analyzing and Developing CPS: Decomposition

Safety “Properties” of a Function: Automatic Emergency Braking (AEB)

- Vehicle provides automated collision safety function
- Vehicle provides/maintains safe stopping
- Braking function reacts as required
- Stopping algorithm provides safe stopping
- Messaging function receives distance to obstacles and speed from propulsion function
- Distance and speed info is understood by braking function
- Friction function provides appropriate friction, depending on the road, tire pressure, etc.

CPS/Function Types

- Business Case
- Use Case ‘feature’
- CPS
- Logical
  - Messages
  - Info
- Physical
  - Influences
  - Energy

*transductions

Functional Decomposition (Logical and Physical)

Generate System Properties

Apply Aspects/Concerns
6. Applications - Griffor

1. Mathematics of CPS and the CPS Framework (E. Griffor)

2. Applications to Transportation (D. McShane, F. Brandao/Ricardo LLC)

3. IES City Tables – CPS Framework as Benchmarking Tool (M. Burns)

4. From Security to Trustworthiness (C. Vishik/Intel)

5. Trustworthiness Ontology (M. Balduccini/St. Joseph’s University)
Cyber-Physical Systems (CPS) comprise interacting digital, analog, physical, and human components engineered for function through integrated logic and physics.

Internet of Things (IoT) emphasizes digital infrastructure for widely connected, interacting systems.

NIST Smart Grid and Cyber-Physical Systems Program Office
6.1.2 The Category CyPhy

• The cyber-physical category CyPhy has as objects:
  o Action/Actuation
  o Sense
  o Phys_State
  o Decision

• The morphisms of CyPhy are given by:
  o $\text{Mor}(\text{Act,Physical\_State}) = \{\text{phy\_act-phys}\}$
  o $\text{Mor}(\text{Decision,Act}) = \{\text{log\_dec-act}\}$
  o $\text{Mor}(\text{Sense,Decision}) = \{\text{log\_sen-dec}\}$
  o $\text{Mor}(\text{Sense,Act}) = \{\text{phys\_sen-act}\}$
  o $\text{Mor}(\text{Phys\_State,Sense}) = \{\text{phy\_Phys\_State-Sense}\}$. 
6.1.3 Symmetric Monoidal Categories

• For purposes here systems will be viewed as processes and interactions between them (process algebra in the sense of Milnor for example)

• We distinguish two sorts of interactions between processes:
  o Logical interactions (exchanges of information)
  o Physical interactions (exchanges of energy)

• Math model of physical interactions is algebraic systems of ODEs

• Math model of logical interactions are formalizations of agent-based models such as complex adaptive systems (J. Holland)

• We choose symmetric monoidal categories (SMC) as an example of a model of systems in category
A cyber-physical system, in the sense of process algebra, can be represented as a **functor from a symmetric monoidal category to the category CyPhy.**

Such a functor represents:

• Processes as instances of **Sensing, Decision, Action or Physical**

• Interactions as **exchanges of information or exchanges of energy**

Benefit of this representation can be derived from:

• Structural representation of one CPS ‘in another’ (isomorphic with a sub-CPS)
6.1.5 The category CPS

Given two representations of CPS as functors $F$ and $G$, let $\text{SM}(F)/\text{SM}(G)$ denote the symmetric monoidal categories that $F$ and $G$ map into CyPhy

$\text{Mor}(F,G)$ is the functors $T$ from $\text{SM}(F)$ to $\text{SM}(G)$ such that the following diagram commutes:

```
          T
         / \    \
SM(F)   --->  SM(G)
        /     \
   F       G
    \     /   \
       CyPhy
```
6.1.6 Mathematics of CPS Framework

**Property-Tree** of a CPS

<table>
<thead>
<tr>
<th>Legend</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_{MB/C} ) = Mission/Business Case</td>
</tr>
<tr>
<td>( P_{ARCH} ) = Integration Steps</td>
</tr>
<tr>
<td>( P_{ASS} ) = Assumptions</td>
</tr>
<tr>
<td>( P_{SUCC} ) = Success Criteria</td>
</tr>
<tr>
<td>( P_{Aspect/Concern} ) = Aspect/Concern</td>
</tr>
</tbody>
</table>

- Branches capture the ‘genealogy’ of a property
- Branching gives assurance conditions for the branching node property
- Concerns may give rise to multiple properties in the Functional Decomposition
- ‘Edges’ should be read ‘depends on’ (L2R) or ‘needed to satisfy’ (R2L)

<table>
<thead>
<tr>
<th>Mission/Business Case (CPS Service/Function)</th>
<th>Functional Decomposition (Subservices)</th>
<th>Requirements Analysis (Decomposition and Concern Application)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_{MB/C} )</td>
<td>( P_{ARCH} )</td>
<td>( P_{Aspect/Concern} )</td>
</tr>
<tr>
<td>( P_{ASS} )</td>
<td>( P_{SUCC} )</td>
<td>( P_{Aspect/Concern} )</td>
</tr>
<tr>
<td>( P_{Aspect/Concern} )</td>
<td>( P_{Aspect/Concern} )</td>
<td>( P_{Aspect/Concern} )</td>
</tr>
<tr>
<td>( P_{Aspect/Concern} )</td>
<td>( P_{Aspect/Concern} )</td>
<td>( P_{Aspect/Concern} )</td>
</tr>
</tbody>
</table>

**Semantics** of CPS Framework

\[ P \in \overline{Concern}^{CPS} \]

\[ \overline{p}^{CPS} = \{ \text{tests } T \text{ for } P \} \]

\[ \text{Supp}_M(T) = \{ \text{measurement support } \mu_1, \ldots, \mu_k \text{ of } T \} \]

\[ \text{Evidence}^{CPS}(P) = \sum_{T \in \overline{PCPS}} T^{CPS} \]

... defines composition of concerns

\[ \overline{C_1} * \overline{C_2}^{CPS} = \overline{C_1}^{CPS} \cup \overline{C_2}^{CPS} \]

**Formal Methods for Assurance** of a CPS

\( < d, e, a > \in P(CPS) \equiv_{Def} \text{ design element } d, \text{ test evidence } e \text{ are sufficient based on argument } a \text{ to conclude that the CPS satisfies } P \)

\[ \text{AssuranceCase}^{CPS} = \sum_{C \in \text{Aspect}^{CPS}} \sum_{P \in \text{CPS}} \sum_{d \in \text{Design}^{CPS}} \sum_{e \in \text{Evidence}(P)^{CPS}} \text{Argumentation}^{CPS}(P) \]
6.2 Applications to Transportation – McShane/Brandao
• A global, multi-industry, multi-discipline consultancy and niche manufacture of high-performance products

• The objective throughout our history has been to maximize efficiency and eliminate waste in everything we do
6.2.3 Strategy for growth: Global engineering, environmental consulting and niche product manufacture…

Megatrends driving focus for solutions:

- **Climate change**
  (Emissions and waste)
- **Resource scarcity**
  (Oil/water usage)
- **Urbanization**
  (Transport, energy, efficiency)
- **Energy security**
  (Renewables, bio-fuels)
6.2.4 Products & services that cover global engineering and test, consultancy, independent assurance & niche product manufacture

A broad range of capabilities and expertise
6.2.5 Automotive Use Case Setup

Typical Development Process

Requirements / Standards

System

Performance reports
Pass/Fail checklists

CPS
6.2.6 Use Case Setup - System

System can be:
- Automotive
- Water Process and Distribution System
- Electrical Grid
- Medical/Health System application
- …

Diverse testbed options/configurations:
- All virtual prototype using MiL and SiL
- HiL system(s) at different phases of the development process
  - with emulated HW
  - HW as they are made available
- Real-time
- Sub-systems and components may have diverse ownership / suppliers
- IP protections
- On-board and off-board interactions and attacks
6.2.7 Sample of multiple control units’ communication

ECU

<<Actor>>

Local CAN

TCU

<<Actor>>

CAN / FLEXRAY

ESP
6.2.8 Automatic Emergency Braking - Example

1. A small LKAS camera beside the rear-view mirror monitors the road markings on either side and feeds data to a computer.

2. LKAS applies correct steering torque to keep car in the centre of the lane.

3. ACC radar sensors behind the Honda badge monitor the distance from the car in front.

4. If the gap with the car in front decreases, the car automatically brakes, then accelerates again to maintain a safe distance.

5. The ACC radar is also used for CMBS and recognises if a collision is imminent. CMBS warns the driver to take action first by an audio and visual warning on the dash. If no action is taken seat belt pre-tensioners will lightly tug the driver to give a physical warning and if still no action is taken, the system will apply strong braking to reduce the impact of a collision.
6.2.9 Automotive System Functional Level: Brake System
6.2.10 Sub-System Behaviors: Brake System

- **Passive Braking** – Basic functionality
  - Brake pressure applied no feedback (Open loop)

- **ABS** – Avoid locking of wheels
  - Brake pressure applied, feedback based on wheel speed sensors (Closed loop)
  - Basic Stability control – not loosing control of vehicle due to braking, based on wheel speed and other sensors

- **Automated – Collision avoidance**
  - Proximity sensors trigger braking event due to;
    - Car brakes by itself (Distracted Driver / reaction time)
    - Driver not braking soon enough or hard enough
  - Keep in the direction of travel, Systems controls steering and brake pressure
    - Similar to LKA
6.2.11 Demo Plan

Concept:

- **Multiple control systems** developed by different suppliers
- **Communication via CAN Bus** using encrypted signals
- **Confidential/proprietary information** passing.
- **Potential to be Hil /Sil** or a combination of both
- **Federated experiments** could be;
  - **Cyber attack** through the infotainment system or on-board component
  - **Braking system Hardware malfunctions** and doesn’t send the correct signals

6.3 IES City Tables: CPS Framework as Benchmarking Tool - Burns
6.3.1 The Challenge - Divergent CPS/IoT Technology Landscape

Internet of Things Landscape 2016

Applications (Verticals)
- Personal
  - Wearables
  - Home Automation
  - Security
  - Entertainment
  - Family
  - Sports
  - Electronics

Vehicles
- Automobiles
  - Autonomous
- Vehicles
- Healthcare
- Enterprise
- Industrial Internet
- Machines
- Energy

Platforms & Enablement (Horizontal)
- Software
- Platforms
- Connectivity
- Interfaces
- 3D
- Augmented Reality
- Other

Building Blocks
- Hardware
- Software
- Connectivity
- Partners

© Matt Turck (@mattturck), David Roff (@davidjroff) & FirstMark Capital (@firstmarkcap)
6.3.2 Internet of Things-Enabled Smart (IES) City Framework

• IES-City (“Yes-City”) Int’l Working Group
  NIST and its partners have convened a public working group to distill a common set of smart city architectural features and to identify “Pivotal Points of Interoperability”
  
  o 3 working groups, collaboration site: https://pages.nist.gov/smartcitiesarchitecture/
  o Completion in fall 2017
6.3.3 NIST Public Working Groups

- Learning by Doing
- Workshops and Analysis
- Application Framework
  - Review exemplar smart-city applications
  - Summarize scope
  - Develop structure of sub-domains
  - Identify Metrics for evaluation

- Technical Architectures
- Workshops and Analysis
- Consensus PPIs
  - Device specifications
  - Document overlaps and gaps
  - Identify PPIs
  - Find consensus standardized interfaces

- Deployments
- Convergence Action Cluster
- Consensus Deployed PPI
  - Discover PPIs from existing Deployments
  - Super Action Clusters e.g. GCTC
  - Identify opportunities to develop more PPIs

Participants: City CTOs, Experts, Companies, Technical Stakeholders, …
6.3.4 Application Framework Model

- Apps/SubApps
- Requirements (CPS Framework/ITU)
- Readiness
- Benefits
- Case Studies

Working SubGroup 1
Working SubGroup 2
Working SubGroup 3
6.3.5 Application Framework Data Analysis

Spreadsheet Database Model of Application Framework

**Category: Water and Wastewater**
- to collect, manage, distribute, use, reuse and recycling water
- to reduce water consumption and contamination, enable the effective utilization of water resources
- to reduce costs and increase the reliability and transparency of water distribution

**Sub-Category: Water collection and management**
- to map and monitor the hydrology network
- to monitor groundwater level
- to predict and manage events (like storm) in time
- to monitor water quality and take corrective action in case of any degradation of water quality
- to analyze, predict and manage water consumption
6.3.6 Consensus PPI

Complex Specifications and Architectures

Simplified by Aspects and Concerns

Zone of Concerns = bundles of services that address a “vector” of concerns
6.4 From Security to Trustworthiness - Vishik
6.4.1 Definition of CPS

Cyber-Physical Systems (CPS) comprise interacting digital, analog, physical, and human components engineered for function through integrated logic and physics.

Internet of Things (IoT) emphasizes digital infrastructure for widely connected, interacting systems.
6.4.2 Trustworthiness in the CPS Framework

CPS Framework Structure

Facets

<table>
<thead>
<tr>
<th>Conceptualization</th>
<th>Realization</th>
<th>Assurance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case, Requirements, …</td>
<td>Design / Produce / Test / Operate</td>
<td>Argumentation, Claims, Evidence</td>
</tr>
</tbody>
</table>

Activities and their Artifacts

- Model of a CPS
- CPS
- CPS Assurance

- CPS Framework Release 1.0 (May2016) available at https://pages.nist.gov/cpswg/
6.4.3 Reach of cyberattacks is expanding

Adequate protection mechanisms have to include privacy, safety, security, and other areas (reliability, resilience) treated in an integrated fashion.

- **1970s**
  - Server attacks: Creeper
  - Server, PC: Brain
  - Server, PC, Mobile: Morris Worm
  - Server, PC, Mobile, ICS: Michelangelo
  - Server, PC, Mobile, ICS: Leandro
  - Server, PC, Mobile, ICS: Tribe Flood
  - Server, PC, Mobile, ICS: DDOS
  - Server, PC, Mobile, ICS: Melissa

- **1980-90s**
  - Server, PC, Mobile: Code Red
  - Server, PC, Mobile, ICS: Sasser
  - Server, PC, Mobile, ICS: SQL Slammer
  - Server, PC, Mobile, ICS: Cabir Premium

- **2000-2005**
  - Server, PC, Mobile, ICS: Stuxnet
  - Server, PC, Mobile, ICS: Duqu
  - Server, PC, Mobile, ICS: GSM interface attack
  - Server, PC, Mobile, ICS: Steel plant attack
  - Server, PC, Mobile, ICS: Ransomware
  - Server, PC, Mobile, ICS: Heart Bleed

- **2005-2010**
  - Server, PC, Mobile, ICS: Confliker
  - Server, PC, Mobile, ICS: ICS-CERT
  - Server, PC, Mobile, ICS: I Love You
  - Server, PC, Mobile, ICS: CAESS

- **2010-2017**
  - Future: Further expansion

- **2017+**
  - Future: Further expansion
6.4.4 Trustworthiness: integrated concept


- **Safety**: Ability to ensure the absence of catastrophic consequences on the life, health, property, or data of stakeholders.
- **Security**: Internal or external protection from unintended and unauthorized access, change, damage, destruction, or use of systems.
- **Privacy**: Risks to individuals arising from the processing of their personal information.
- **Reliability**: Ability to deliver stable and predictable performance in expected conditions.
- **Resilience**: Ability of withstand instability, unexpected conditions, and gracefully return to predictable, but possibly degraded, performance.

**Definition:** Demonstrable likelihood that the system performs according to designed behavior under a typical set of conditions as evidenced by its characteristics, such as safety, security, privacy, reliability and resilience.
6.4.5 Integrated trustworthiness: sample categories of use cases

- **Modeling**: Use the integrated approach to model current and future systems and complex environments.
- **Design and development**: Anticipate connections, features, and constraints in designing CPS.
- **Analysis and assessment**: Assess existing systems to understand their potential vulnerabilities and to optimize deployment.
6.4.5 Integrated trustworthiness: some challenges

- Misalignment of metrics: Even for probabilistic models, the values for failure rates for, e.g., security and safety are very different.
- Mutually exclusive requirements: Requirements for adjacent areas are misaligned. E.g., transparency requirements are fundamentally different for security and privacy.
- No mechanism for composition: No reliable tools to model a component in context, e.g., autonomous vehicle in a Smart City.
6.4.6 Next steps: from TW positioning to ontology & reasoning

Current state

CPS framework positions trustworthiness within CPS and allows the technologists to decompose this concept

Ontology and reasoning

Future work needs to create a trustworthiness ontology matching the current CPS framework and reasoning algorithms to analyze trustworthiness

Trust language

Trust language will need to be developed as a tool for trustworthiness calculus (building on existing ‘reasoning languages’ such as OCL, OWL, etc.)
6.4.7 Useful concept: trust evidence

Ability to rely on a broader list of characteristics (evidence) to assess trustworthiness. Some examples below.

<table>
<thead>
<tr>
<th>Trust posture</th>
<th>Measurement</th>
<th>Evaluation</th>
<th>Identification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nuanced rather than black &amp; white approach to trust</td>
<td>Verified proof of domain integrity and software integrity, integrity measurement, time stamp</td>
<td>Evidence of self certification &amp; evaluation, conformance to standards, data quality</td>
<td>Provable device identification, proof of ownership, provable transaction identification</td>
</tr>
</tbody>
</table>

**Definition:** trust evidence is an agreed upon system of parameters that could help define trustworthiness in a complex environment
6.4.8 Relevant area: human/technology connection

- **Intent semantics**: The intent of the user or developer while designing and using a system.
- **Usability**: Ease, with which a system can be used, learned, and integrated into an environment or process.
- **Transparency**: Ease, with which a system can be understood for functions that a user or other systems need to know.

Although the majority of interactions are machine to machine, the human aspect is very important!
6.5 Trustworthiness Modeling - M. Balduccini
6.5.1 Modeling Methodology: Conceptual Ontology

**Use-case(s)**

**Domain Knowledge**

**Commonsense/World Knowledge**

---

**Ontological Analysis**

**Formalization**

**Evaluation**

**Questions**

**Queries**

*Types of queries*
1. Retrieval queries:
   - DB-like queries, purely propositional
2. Competency/adequacy queries
3. Reasoning queries
   - Require guess-and-check:
     - planning, diagnostics from first principles

**Objects, Classes, Functions, Relations**

**Ontology Specification**
6.5.2 Modeling Methodology: Parametrization, Ontology Calculus

Ontological Analysis
- Numerical Variables + Constraints:
  - use case parameterization
  - equalities and inequalities over numerical variables and numerical constants/functions

Objects, Classes, Functions, Relations
- Numerical Variables + Constraints

Formalization

Use-case(s)  Domain Knowledge  Commonsense/World Knowledge

Evaluation

Ontology Specification

Questions

Queries

Types of queries
1. Retrieval queries
   - DB-like queries, purely deductive
2. Competency/Capability Queries
3. Reasoning queries
   - Require guess-and-check: planning, diagnostics from first principles
4. Numerical queries
   - Find values of numerical parameters that satisfy all given constraints
5. Ontology Computations: combine 1+3, 2+3, 1+2+3
6.5.3 Modeling Tools

UML

Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering, that is intended to provide a standard way to visualize the design of a system. (Wikipedia)

Focus: systems, system design, code generation

Knowledge Representation (KR) Languages, OWL

The W3C Web Ontology Language (OWL) is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things. OWL is a computational logic-based language such that knowledge expressed in OWL can be exploited by computer programs, e.g., to verify the consistency of that knowledge or to make implicit knowledge explicit (W3C)

Focus: “world” knowledge, commonsense, automated reasoning

Which Sensor can detect Stimulus x?
6.5.4 Modeling Use-Case

Device model

- Police body cameras
  - Body cameras, location sensors, alarm sirens

Trustworthiness elements

- Security
  - Physical Security: broken, stolen
  - Cyber Security: CIA of data streams
    - Important if used in court
    - Potential approach: timestamp

- Privacy
  - Is face recognition in use?
  - Who has access to the information?

- Reliability: will the camera work 24/7?
  - Data reliability:
    - Multiple cameras, multiple streams
    - Stored on a server
    - Who has access? Who can access all streams?

- Resilience
  - What if a camera does not work?
  - Can the stream from a nearby camera be used as a substitute?
    - “Stitch camera feeds from home security systems”
6.5.5 Earlier Work: Cone-of-Impact Vulnerability Assessment

MCS Architecture Model
- Hierarchy of systems
- Physical/network links

Vulnerability Model
- Vulnerabilities
- Affected systems

Threat Model
- Target
- Consequences

Mitigation/Control Model
- Mitigation type
- Effect
7. Panel Discussion - Greer

• **Discussion Topic:** “Why do we need holistic concern-driven engineering? “

• **Moderator:** Dr. Chris Greer

• What kinds of questions keep CPS leaders “up at night”?

• How should a CPS engineering process address questions like: Where are we in the process, how do we stand? What’s the degree of completion? What’s the test coverage?

• How do current practices reveal and resolve competing/interacting concerns in complex CPS?

• What has to change in education and training to succeed in engineering CPS? To drive a holistic concern-driven culture into the skills-based engineering curriculum of today?
8. Systems Engineering and the CPS Framework - Roth
8.1 Our goals for the CPS Framework

• Our goal is **not to replace** systems engineering processes!

• We believe that **existing approaches do not explicitly consider the breadth of concerns** required for CPS

• Our goal is to **enhance** existing systems engineering processes with a methodology **to apply a rich set of concerns** that are traceable throughout the CPS life cycle
8.2 What is 15288?

• An international standard that describes “a common framework of process descriptions for describing the life cycle of systems created by humans”

• Defines a set of processes that span the system life cycle separated into 4 categories (see right)

• Each process description has:
  1. a statement of purpose
  2. a set of outcomes
  3. a list of activities and their tasks

Source: ISO/IEC/IEEE 15288
8.3 15288 is designed to be adaptive

- It **does not prescribe a development methodology** for the implementation of process descriptions in a project.

- It recommends to **use only the sub-set of relevant processes** for a given system of interest.

- It **defines a tailoring method** to modify existing life cycle processes or create new processes.
NIST SP 800-160

Special Publication on Systems Security Engineering
8.4 How can we build a secure system?

- **15288 provides no guidance** on what must be considered at each stage of the system life cycle to build a secure system.

- **Modern systems are too complex** for concerns such as security to be separated from the system life cycle processes.

- Trustworthiness is achieved by **holistic consideration of security concerns during system engineering processes**.
8.5 What is 800-160?

- Tailors 15288 process descriptions (purpose, outcomes, and activities) to incorporate trustworthiness concerns

<table>
<thead>
<tr>
<th>ID</th>
<th>PROCESS</th>
<th>ID</th>
<th>PROCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>AQ</td>
<td>Acquisition</td>
<td>MS</td>
<td>Measurement</td>
</tr>
<tr>
<td>AR</td>
<td>Architecture Definition</td>
<td>OP</td>
<td>Operation</td>
</tr>
<tr>
<td>BA</td>
<td>Business or Mission Analysis</td>
<td>PA</td>
<td>Project Assessment and Control</td>
</tr>
<tr>
<td>CM</td>
<td>Configuration Management</td>
<td>PL</td>
<td>Project Planning</td>
</tr>
<tr>
<td>DE</td>
<td>Design Definition</td>
<td>PM</td>
<td>Portfolio Management</td>
</tr>
<tr>
<td>DM</td>
<td>Decision Management</td>
<td>QA</td>
<td>Quality Assurance</td>
</tr>
<tr>
<td>DS</td>
<td>Disposal</td>
<td>QM</td>
<td>Quality Management</td>
</tr>
<tr>
<td>HR</td>
<td>Human Resource Management</td>
<td>RM</td>
<td>Risk Management</td>
</tr>
<tr>
<td>IF</td>
<td>Infrastructure Management</td>
<td>SA</td>
<td>System Analysis</td>
</tr>
<tr>
<td>IM</td>
<td>Information Management</td>
<td>SN</td>
<td>Stakeholder Needs and Requirements Definition</td>
</tr>
<tr>
<td>IN</td>
<td>Integration</td>
<td>SP</td>
<td>Supply</td>
</tr>
<tr>
<td>IP</td>
<td>Implementation</td>
<td>SR</td>
<td>System Requirements Definition</td>
</tr>
<tr>
<td>KM</td>
<td>Knowledge Management</td>
<td>TR</td>
<td>Transition</td>
</tr>
<tr>
<td>LM</td>
<td>Life Cycle Model Management</td>
<td>VA</td>
<td>Validation</td>
</tr>
<tr>
<td>MA</td>
<td>Maintenance</td>
<td>VE</td>
<td>Verification</td>
</tr>
</tbody>
</table>

Source: NIST SP 800-160
CPS Framework
A Holistic Concern-Driven Approach
8.6 How does the CPS Framework fit?

**Aspects**
- Functional
- Business
- Human
- Trustworthiness
  - Timing
  - Data
  - Boundaries
  - Composition
  - Lifecycle

**Facets**
- Conceptualization
  - Use Case, Requirements, ...
- Realization
  - Design / Produce
  - Test / Operate
- Assurance
  - Argumentation, Claims, Evidence

**Activities**
- Model of a CPS
- CPS
- CPS Assurance

**Artifacts**

NIST CPS PWG Framework Release 1.0
8.7 The need for a holistic approach

• All CPS aspects/concerns intrinsically depend on each other and we need a holistic approach for such cross-cutting concerns

• Examples:
  o Cyber security mechanisms can be defeated by physical attacks
  o The most secure system is one that does nothing
  o The dichotomy between being fast and secure
CPS Framework Open Source

The Road to a Development Process Tool
8.8 Our current state

- We **need** a holistic, concern-driven methodology for the development of CPS

- We **have** multiple system engineering processes that provide an outline for how to develop a CPS

- We **have** an ontology of cross-cutting concerns extracted from domain experts in CPS
8.9 Our plan moving forward

• CPS Framework satisfies our need at a conceptual level

• There is still a gap on **how to implement** the framework:
  o How to annotate the artifacts (process outcomes) with concerns?
  o How to manage and exchange the artifacts that are produced?
  o How to trace concerns across artifacts throughout the life cycle?

• We are working towards a **tool and data exchange format** to manage artifacts produced by 15288 / 800-160 / …
9. Modeling for a ‘CPS Framework Tool’ (Burns/Song)

1. Moving the CPS Framework Forward
2. Use Case Methodology
3. Model Realization
4. Modeling the CPS Framework and Use Case
5. Tools Demonstration
9.1 Moving the CPS Framework Forward

• We wanted to move adoption of the CPS Framework concepts into common practice in engineering CPS

• We believe that the CPS Framework enhances and extends existing system engineering processes and does not alter or replace them

• We hoped to quantify the discussion of CPS so that it can be studied from multiple disciplines

• So we developed a useable model of the CPS Framework
9.1.1 CPS Framework Model Requirements

• Capture Concern-Driven Analysis
• Supports traceability of requirements, designs that realize them, tests that verify them, and argumentation that validates them
• Allows for maturity and versioning of parts
• Allows for reuse
  o composition of existing and new parts
• Supports referencing external artifacts and specifications
  o External documents and specifications
  o Standards and certification test references
  o Development process tool artifacts
• Supports reasoning over data set in single XML Document
9.1.2 Tech Transfer Concept

• We built this model so it can accessorize existing tool suites for system engineering process execution

• The result is a simple XML data file that can be an import or export to any tool

• The data structure of the XML document object is composable so that various tools can add/edit detail at any time

• The UML model and XMLSchema is provided as an open source tool set (more on this later) at:
  - https://github.com/usnistgov/cpsframework

• We encourage interested parties to evolve this with us to suit your collective needs
9.1.3 Evolution of CPS Design Instance

Tools for conceptualization, realization, and assurance of CPS

Requirements Capture ⇒ Design ⇒ Testing ⇒ Assurance

Continuous integration

Configuration management

CPS Framework Document Instance
And now the details ....
9.1.4 A Union of Technologies

IEC 62559 Use Case Methodology

NIST CPS Framework Methodology

Standardized XML Schema

Conceptualization
- Business Case
- Use Case
- Requirements

Realization
- Design
- Traceability to Requirements

Assurance
- Algorithmically Prove Design Meets Requirements
9.1.5 Framework Open Source Project

Continuous Integration for CPS Development

Common XML format – Model of CPS
CPS and Assurance of CPS

Requirements modeling tool

Design Exploration / Model Driven Development / Continuous Integration Tools

Design Verification and Validation and Assurance Tools

CPS Framework Use
Case/Aspects/Concerns Analysis

https://github.com/usnistgov/cpsframework
9.1.6 Methodology

1. Capture the CPS Framework in UML
   - Class hierarchy of facets, aspects, concerns, ...
   - Functional decomposition based on an IEC Use Case standard

2. Generate an XMLSchema of the model
   - Which governs an XML instance document of a CPS Framework

3. Produce a test example CPS
   - A smart communicating thermostat
9.2 Use Case Methodology - Song
9.2 IEC 62599 Standard-based Smart Thermostat Use Case

1) What Is A Use Case?
2) IEC 62599 Standard - Use Case Methodology
3) IEC 62599-2 Standard - Template Format for Use Case
4) IEC 62599-2 Standard-based Smart Thermostat Use Case
5) Benefits of Standard-based Use Case
9.2.1 What Is A Use Case?

• A use case is an abstraction of a function of a system.
• A use case is a specification of a set of actions performed by a system. (ISO/IEC 19505-2:2012)
• Use cases are used to capture functional requirements of a system.

For example: Smart thermostat has heating, cooling and automatic control modes (three actions) to control HVAC system to maintain room temperature near a user’s set point.
9.2.2 IEC 62559 - Use Case Methodology

IEC 62559 - Use Case Methodology:

- **Part 1** – **Concept and processes in standardization**
- **Part 2** – **Definition of a template format** for use cases, actor list and requirements list
- **Part 3** – **UML Model and XML Serialization** of use case template artifacts
9.2.3 IEC 62559-2 - Template Format for Use Case

1. Description of the use case
2. Diagrams of use case
3. Technical details
4. Step by step analysis of use case
5. Information exchanged
6. Requirements (optional)
7. Common terms and definitions
8. Custom information (optional)
9.2.4 IEC 62559-2 Standard-based Smart Thermostat Use Case

A Business Case for Smart Thermostat

• Build a **smart thermostat (ST)** that has **heating**, **cooling** and **automatic control modes** to **maintain room temperature near a user’s set point** and uses a WiFi local area network (LAN) to **interact with a temperature sensor** and an **HVAC system** in a home.

• The thermostat should be able to retail for less or equal to **$79**. It must be **intrinsically safe**, **reliable**, **secure**, **protect privacy**, **easy to use** and **upgradable**.
## 1. Description of use case

### 1.1 Name of Use Case

<table>
<thead>
<tr>
<th>Use case identification</th>
<th>Name of Use Case</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ID</strong> Domain(s)</td>
<td></td>
</tr>
<tr>
<td>1.1.1 User</td>
<td>Maintain room temperature near a user’s set point</td>
</tr>
</tbody>
</table>

### 1.2 Version Management

#### Version management

<table>
<thead>
<tr>
<th>Version No.</th>
<th>Date</th>
<th>Name Author(s)</th>
<th>Changes</th>
<th>Approval Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>2016-04-06</td>
<td>Eugene/Cuong</td>
<td>Initial</td>
<td>Initial</td>
</tr>
<tr>
<td>0.8</td>
<td>2017-08-31</td>
<td>Eugene/Ed</td>
<td>Remove sensor gateway, change control message</td>
<td></td>
</tr>
</tbody>
</table>

### 1.3 Scope and Objectives of Use Case

#### Scope and objectives of use case

<table>
<thead>
<tr>
<th>Scope</th>
<th>A smart thermostat to remotely control the HVAC system through a local area network (LAN) in the home</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective(s)</td>
<td>Provide the functional requirements for a smart thermostat to control the HVAC system based on user inputs or set points.</td>
</tr>
<tr>
<td>Related business case(s)</td>
<td>See it before</td>
</tr>
</tbody>
</table>
1.4 Narrative of Use Case

**Narrative of use case**

**Short description**

This use case describes the operations of a smart thermostat (ST) to control an HVAC system. It has three operational modes – heating, cooling, and automatic control.

**Complete description**

This use case describes the operations of a ST to control an HVAC system. It has three operational modes – heating, cooling, and automatic control.

*User* can set the temperature set point for ST locally. *Thermostat controller* in ST can pull the room temperature from the *temperature sensor*, compare it to the set point and then remotely control heating and cooling systems of an HVAC system via an *HVAC controller* through a WLAN to maintain room temperature near the desired set point.

The ST communicates over the network and is globally reachable from the WLAN. This allows remote client applications to read the status of the ST and manipulate its set points.
9.2.4.3 IEC 62559-2 Standard-based Smart Thermostat Use Case (Cont’d)

2. Diagram of Smart Thermostat Use Case

Information exchanged

User

Smart Thermostat

Thermostat Controller

HVAC System

Temperature Sensor

Compressor

Fan

Furnace

Temperature

Set point

Status

Control

Actors
### 3.1 Actors: People, Systems, Applications, Databases, the Power System, and Other Stakeholders

<table>
<thead>
<tr>
<th>Grouping (Community)</th>
<th>Group description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home Energy System</td>
<td>The components of a home energy management system</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actor name</th>
<th>Actor type</th>
<th>Actor description</th>
<th>Further information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thermostat Controller</td>
<td>Controller</td>
<td>A controller in smart thermostat can send and receive messages, as well as control the HVAC system</td>
<td></td>
</tr>
<tr>
<td>HVAC Controller</td>
<td>Controller</td>
<td>A controller in the HVAC can send and receive messages from the thermostat, as well as trigger the HVAC operations</td>
<td></td>
</tr>
<tr>
<td>Temperature Sensor</td>
<td>Sensor</td>
<td>Thermostat reads data from temperature sensor</td>
<td></td>
</tr>
<tr>
<td>User</td>
<td>Person</td>
<td>The owner of the thermostat. A User would provide the inputs or set points for the operation of the thermostat</td>
<td></td>
</tr>
</tbody>
</table>
4 Step by Step Analysis of Use Case

4.1 Overview of scenarios

<table>
<thead>
<tr>
<th>No.</th>
<th>Scenario name</th>
<th>Scenario description</th>
<th>Primary actor</th>
<th>Triggering event</th>
<th>Pre-condition</th>
<th>Post-condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Heating Mode</td>
<td>This is the heat mode setting on the ST</td>
<td>Thermostat Controller</td>
<td>Temperature difference</td>
<td>Temperature is <strong>lower than</strong> or <strong>equal</strong> to temperature set point</td>
<td>The HVAC is running until the temperature is <strong>higher than</strong> the set point</td>
</tr>
<tr>
<td>4.2</td>
<td>Cooling Mode</td>
<td>This is the cool setting on the ST</td>
<td>Thermostat Controller</td>
<td>Temperature difference</td>
<td>Temperature is <strong>higher than</strong> or <strong>equal</strong> to temperature set point</td>
<td>The HVAC is running until the temperature is <strong>lower than</strong> the set point</td>
</tr>
<tr>
<td>4.3</td>
<td>Automatic Control Mode</td>
<td>This is the automatic mode setting on the ST</td>
<td>Thermostat Controller</td>
<td>Temperature difference</td>
<td>Temperature is <strong>lower than</strong> or <strong>equal</strong> to temperature set point</td>
<td>The HVAC is running until the temperature is <strong>higher than or equal</strong> to the set point</td>
</tr>
</tbody>
</table>

Temperature is **lower than** or **equal** to temperature set point

The HVAC is running until the temperature is **higher than** the set point
### 4.2.1 Steps – Scenarios

<table>
<thead>
<tr>
<th>Scenario Name</th>
<th>Heating Mode</th>
<th>(The commands and statuses reference to both furnace and fan)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step No.</strong></td>
<td><strong>Event</strong></td>
<td><strong>Name of process/activity</strong></td>
</tr>
<tr>
<td>1</td>
<td>Set Temperature Set Point</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Temperature Change</td>
<td>Temperature update</td>
</tr>
<tr>
<td>3</td>
<td>HVAC Operation</td>
<td>HVAC switch on</td>
</tr>
<tr>
<td>4</td>
<td>Status Update</td>
<td>HVAC On</td>
</tr>
<tr>
<td>5</td>
<td>Temperature Change</td>
<td>Temperature update</td>
</tr>
<tr>
<td>6</td>
<td>HVAC Operation</td>
<td>HVAC Off</td>
</tr>
<tr>
<td>7</td>
<td>Status Update</td>
<td>HVAC Off</td>
</tr>
</tbody>
</table>
## 5 Information Exchanged

<table>
<thead>
<tr>
<th>Information exchanged ID</th>
<th>Name of information</th>
<th>Description of information exchanged</th>
<th>Requirements IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature</td>
<td>Temperature</td>
<td>Float temperature value</td>
<td></td>
</tr>
<tr>
<td>TemperatureSetPoint</td>
<td>Temperature</td>
<td>Float temperature value</td>
<td></td>
</tr>
<tr>
<td>Control</td>
<td>Control</td>
<td>Control: 1 = On, 0 = Off Example:</td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td>Status of HVAC</td>
<td>Contains the current status of the HVAC: 0 = Off, 1 = On Example:</td>
<td></td>
</tr>
</tbody>
</table>

### Example:
- **Control**: 
  - Cool: 0
  - Fun: 0
  - Heat: 0

- **Status**: 
  - Cool: 1
  - Fun: 1
  - Heat: 0
9.2.4.8 IEC 62559-2 Standard-based Smart Thermostat Use Case (Cont’d)

Smart Thermostat Use Case (IEC 62599-2) (Microsoft Word file)

Software Tool (Conversion)

Smart Thermostat Use Case (IEC 62599-3) (XML Data file)
9.2.5 Benefits of a Standard-based Use Case Methodology

• provide a standardized format and common understanding of use cases (including functionalities, actors and interactions) of CPS systems.

• help to easily understand functions and requirements of CPS systems.

• help to easily exchange or share of use cases among CPS system development processes.
9.3 Model Realization - Burns
9.3.1 CPS Framework Modeling Tools

**Queries and Assessments**
- owl:sparql

**CPS Mathematics**
- Property
  - statement: char
  - trace: char [0..1]
  - priority: float [0..1]
  - reference: char [0..*]
  - description: char [0..1]

**CPS Facets**
- xml:xsd
9.3.2 NIST CPS Framework

‘Concern-driven’: holistic, integrated approach to CPS/IoT concerns.

- CPS Framework Release 1.0 (May 2016) available at [https://pages.nist.gov/cpswg/](https://pages.nist.gov/cpswg/)
9.3.3 Crash course in UML

### Inheritance (is-a) relationship

<table>
<thead>
<tr>
<th>Name</th>
<th>Base</th>
<th>Derived2 is-a Base</th>
</tr>
</thead>
<tbody>
<tr>
<td>Derived1</td>
<td>Derived2</td>
<td></td>
</tr>
</tbody>
</table>

#### Association (uses, interacts-with) relationship

- **A**
  - A’s role
  - B’s role

- **B**

#### Multiplicity in Aggregation, Composition, or Association

- * - any number 0..1 - zero or one
- 1 - exactly 1
- 1..* - 1 or more
- n - exactly n
- n..m - n through m

---

### Aggregation and Composition (has-a) relationship

#### Whole

- Whole
  - Whole has a Part

#### Part

- Part
  - Part is independent
  - Part is dependent

#### Navigability - can reach

- B starting from A

- B is associated with one A
- A is associated with 0 or more B

---

**Class**

<table>
<thead>
<tr>
<th>Attributes (member variables)</th>
<th>Methods (member functions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name</td>
</tr>
</tbody>
</table>

---

National Institute of Standards and Technology • U.S. Department of Commerce
9.4 Modeling the CPS Framework and Use Case
9.4.1 CPS Framework Object Model in UML

**See Aspects on subsequent slide**
9.4.2 CPS Framework Aspects

**Each Aspect has a hierarchically arranged set of concerns**
9.4.3 Functional Decomposition

CPS/Function Types

- Business Case
- Use Case
  - ‘feature’
- CPS
  - Logical
    - Message
    - Info
  - Physical
    - Influence
    - Energy

Functional Decomposition/Allocation

- Function
  - Logical
    - CDP
    - Info
  - Physical
    - Influence
    - Energy

Classes:
- UseCaseModelDerived::BusinessCase
- UseCaseModelDerived::UseCase
- UseCaseModelDerived::CPS
- UseCaseModelDerived::InformationModel
- UseCaseModelDerived::Scenario
- UseCaseModelDerived::MacroActivity

1..*
9.4.4 Allocation of Aspects to Model Elements

Note how each major component of functional decomposition can have apportionment of Aspects and Concerns.
### 9.4.6 Concerns and Properties

Properties, like requirements, are assertions intended to address a concern and evaluate to true or false to facilitate testing and verification.

**Concern**

- Properties define the requirements that address a concern
- Concerns are grouped hierarchically beneath Aspects

**Instance of Aspects/Concerns tree**

**Property**
- statement: char
- trace: char [0..1]
- priority: float [0..1]
- reference: char [0..*]
- description: char [0..1]

**Property is defined as containing:**
- statement: a requirements-like assertion that is either true or false
- trace: a reference to another Property elsewhere in the graph
- priority: a priority to be used to referee competing properties
- reference: a reference such as a standard, regulatory or best practice
- description: a more elaborate description of the statement
9.4.7 Model of a Facet: a collection of process activities

CPS are conceived and built in order to address certain needs while addressing any concerns the stakeholders may have. There will be activities, or sets of activities, with well-defined outcomes or deliverables that are designed to fulfill those needs and, at the same time, address stakeholder concerns.
9.4.8 Process Element Depiction of CPS Framework Facets

Conceptualization Facet

- Conceptualization Activities
- Model of CPS

Assurance Facet

- Assurance Activities
- CPS Assurance

Realization Facet

- Realization Activities
- CPS
9.4.9 Facets

CPS Framework Structure

Facets

Conceptualization:: Conceptualization
Activities:: ConceptualizationArtifacts

Realization:: Realization
Activities:: RealizationArtifacts

Assurance:: Assurance

Activities:: AssuranceArtifacts

Assurance:: AssuranceCase

Activities:: AssuranceArtifacts

Ref_Property
+ refProperty: char

«enumeration»
Assurance:: VerificationKind
+ test
+ inspection
+ analysis
+ demonstration

0..* 0..* 0..*

0..1

Assurance:: Evidence

«enumeration»
Assurance:: ArgumentationKind
+ standard
+ best practice
+ formal method
+ regulation
+ expertise

Assurance:: Claim

Conceptualization:: :Property

Realization:: Design

Realization:: TestCase
+ kind: VerificationKind

Claim
+ evidence: Evidence
+ argumenation: ArgumentationKind
+ confidence: float

Assurance:: Judgement

Activities and their Artifacts

Model of a CPS
CPS
CPS Assurance

Domains
Functional
Business
Human
Trustworthiness
Timing
Data
Boundaries
Composition
Lifecycle

Manufacturing
Transportation
Energy
Healthcare
others...

Activities:: AssistanceArtifacts

Activities:: AssistanceArtifacts

Activities:: AssistanceArtifacts
9.4.10 Activities: Conceptualization Facet
9.4.11 Activities: Realization Facet
9.4.12 Activities: Assurance Facet

- IdentifyAssuranceObjectives
- DefineAssuranceStrategy
- ControlAssuranceEvidence
- AnalyzeEvidence
- ProvideAssuranceArgument
- ProvideEstimateOfConfidence
- ConfigureAudit
- RequirementsVerification
- ProductCertificationAndRegulatoryComplianceTesting
9.4.13 Maturity and Versioning

Maturity

+ maturity: maturityLevels
+ effective: char
+ responsibleParty: char
+ revision: char
+ status: MaturityStatus

maturityLevels

- submitted
- verified
- approved
- delegated
- rejected
- reviewed

MaturityStatus

- asDeveloped
- asDesigned
- asPlanned
- asBuilt
- asMaintained
- asDisposed
- asPreserved

CPSFramework

InteractionLibrary

UseCaseModelDerived:: BusinessCaseLibrary

UseCaseModelDerived:: InformationModelLibrary

UseCaseModelDerived:: CPSLibrary
9.4.14 Turn the crank to generate the schema

Model in UML

Model in XMLSchema

Code Engineering / Generate XML Schema

Sparx Systems Enterprise Architect V13.5

Revise to optimize schema

Altova XMLSpy 2017
9.4.15 XML Editor of a Use Case
9.4.16 CPS Framework in XML Schema
9.4.17 CPS Framework Instance: Thermostat Design
9.5 Tools Demonstration

1) UML Model Review
2) XSD Export and Review
3) XML Model Browse
   1) XSLT (text view vs browser view)
   2) Xpath (//*/Aspects/trustworthiness/*/Property/..)
10. Building Community around CPS Framework
Open Source - Griffor

• What are the hoped-for outcomes
• Collaboration Tools – GitHub Environment
• Embedding this technology in your CPS Engineering Tool
• Open Discussion on Next Steps
10.1 Revisiting Workshop Goals

This workshop addresses key CPS challenges: what are the methods and tools needed to conceive, design, build, deliver and maintain Cyber-Physical Systems.

1. What is CPS?
2. How do we design, build and assure CPS throughout their lifecycle?
3. What discipline do we need to address the concerns that drive requirements and engineering?
4. What needs to be the common core tooling?
10.2 Concept common core tooling: CPS Framework Open Source

1) ‘Type Structure’ for:
   o Aspects and concern; and
   o Facets, engineering activities and outcomes

2) That type and sort compositionally:
   o Properties/requirements and
   o Artifacts

3) Encoded in a portable, reusable XML format.
Program Speakers and Panelists

NIST Participants
- E. Griffor (EL)
- Ron Ross (ITL)
- D. Wollman (EL)
- M. Burns (EL)
- C. Greer (EL)
- T. Roth (EL)
- E. Song (EL)

External Participants
- D. McShane (Ricardo LLC)
- F. Brandao (Ricardo LLC)
- C. Vishik (Intel)
- M. Balduccini (St. Joseph University)
- A. Rajhans (Mathworks)
- H. Neema (Vanderbilt University)
- S.-W. Lin (Thingswise)
- J. Weimer (UPenn-PRECISE)