# **C****PSFramework**

NIST CPS Framework.

## **FunctionalDecomposition diagram**



FunctionalDecomposition

## **Maturity diagram**

The Maturity Model is intended to capture lifecycle in the context of the NIST CPS Framework. All of the instances of Framework objects are subject to revisions and approvals as they mature through maturity levels of approval, including submitted, verified, approved, delegated, rejected and reviewed. In addition there is a maturity status that can take the values of as developed, as designed, as planned, as built, etc.



Maturity

## **CPS diagram**

The CPS class diagram summarizes the nature and constituent parts of a CPS. Functions of the CPS can be designated as 'Cyber' or 'Physical that in turn consists of at least one device and of a system or a system of systems.



CPS

## **Facets diagram**

The CPS class diagram summarizes the facets of the CPS Framework, including Conceptualization, Realization and Assurance.



Facets

## **CPSFramework diagram**

The CPS class diagram that summarizes the types of functions, as well as those of type facet and of type aspect.



CPSFramework

## **CPS to Use Case Cross Associations diagram**

The CPS class diagram that summarizes the types and enumeration types associated with a use case (following the IEC UML Model of Use Case) and contains the association of the use case model with the model of the CPS Framework.



CPS to Use Case Cross Associations

## **A****ctivities**

This package contains canonical process activities performed during conceptualization, realization, and assurance. Each industry or system engineering practitioner has its own set of process activities virtually all of which are map-able to this canonical set.

### **Activities diagram**



Activities

### **AnalyzeEvidence**

### **AssuranceActivity**

### **AssuranceArtifacts**

The Assurance Artifacts consists of the outputs of the assurance facet, including evidence, argumentation and judgments that the evidence and argumentation is sufficient to conclude that the properties of the CPS Model have been achieved with a sufficient level of confidence.

### **BusinessCaseAnalysis**

### **ConceptualizationActivity**

### **ConceputalizationArtifacts**

The Conceptualization Artifacts consists of the outputs of the conceptualization facet, including the properties of the system as well as their analysis and, finally, the Model of the CPS.

### **ConfigureAudit**

### **ControlAssuranceEvidence**

### **CyberPhysicalAbstractionLayerFormation**

### **DefineAssuranceStrategy**

### **DesignCPS**

### **Disposal**

### **FunctionalDecomposition**

### **IdentifyAssuranceObjectives**

### **InterfaceRequirementsAnalysis**

### **LifecycleManagement**

### **ManufacturingImplementation**

### **MissionAndBusinessCaseDevelopment**

### **Operations**

### **PhysicalLayerRealization**

### **ProductCertificationAndRegulatoryComplianceTesting**

### **ProvideAssuranceArgument**

### **ProvideEstimateOfConfidence**

### **RealizationActivity**

### **RealizationArtifacts**

The Realization Artifacts consists of the outputs of the realization facet, including the design requirements, design artifacts and, finally, the system or CPS itself (all builds including the final release build of the system).

### **RequirementsAllocation**

### **RequirementsAnalysis**

### **RequirementsVerification**

## 

## **Assurance**

This package contains definitions for the assurance facet.

### **ArgumentationKind**

ArgumentationKind is an enumeration type that consists of the kinds of argumentation used in an assurance judgement, including standard, best practice, formal method, regulation and expertise.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| standard |  |  |

|  |  |  |
| --- | --- | --- |
| best practice |  |  |

|  |  |  |
| --- | --- | --- |
| formal method |  |  |

|  |  |  |
| --- | --- | --- |
| regulation |  |  |

|  |  |  |
| --- | --- | --- |
| expertise |  |  |

### **Assurance**

The assurance facet is the set of activities, including evidence management and definition of argumentation required to assert that the system produced during the realization facet satisfies the properties defined by the conceptualization facet with a sufficient level of confidence.

### **AssuranceCase**

The Assurance Case aggregates the judgements that the evidences and argumentation, are sufficient to establish that the properties hold of the system, with a sufficient level of certainty.

### **Claim**

The Claim class consists of one or more references to properties of a system. A claim asserts that a system has the properties referred to.

### **Evidence**

The Evidence in assurance judgements can consist of test, inspection, analysis or demonstration.

### **Judgement**

A Judgement is an assertion that the argumentation establishes that the evidence is sufficient to show that the claim/property holds of the system with a sufficient level of confidence. Argumentation can take different forms, including reference to a standard, best practice, formal method, regulation or expertise.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| claim | Claim |  |

|  |  |  |
| --- | --- | --- |
| evidence | Evidence |  |

|  |  |  |
| --- | --- | --- |
| argumenation | ArgumentationKind |  |

|  |  |  |
| --- | --- | --- |
| confidence | float |  |

### **VerificationKind**

VerificationKind is an enumeration type that consists of test, inspection, analysis and demonstration. It describes the desired form of test procedure and artifact produced.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| test |  |  |

|  |  |  |
| --- | --- | --- |
| inspection |  |  |

|  |  |  |
| --- | --- | --- |
| analysis |  |  |

|  |  |  |
| --- | --- | --- |
| demonstration |  |  |

## 

## **Conceptualization**

This package contains definitions for the conceptualization facet.

### **ActorDecomposition diagram**



ActorDecomposition

### **Concern diagram**



Concern

### **Aspects diagram**



Aspects

### **Actuation**

Concerns related to the ability of the CPS to effect change in the physical world.

### **Adaptability**

Concerns about the ability of the CPS to achieve an intended purpose in the face of changing external conditions such as the need to upgrade or otherwise reconfigure a CPS to meet new conditions, needs, or objectives.

### **Aggregation**

Concerns about the aggregation of data to form summaries or collections from multiple source

### **Analytics**

Concerns about analytics

### **Authentication**

Concerns about the authenticity of data

### **Authorization**

Concerns about the authorizations constraining access to data

### **Availability**

Concerns about ensuring timely and reliable access to and use of a system.

### **BOL**

### **BasicConnectivity**

Concerns about connectivity of CPSDevices

### **Behavioral**

Concerns about interdependence among behavioral domains. Concerns related to the ability to successfully operate a CPS in multiple application areas.

### **BoundariesAspect**

Concerns about the demarcations of topological, functional, organizational, or other forms of interactions.

### **BusinessAspect**

Concerns about enterprise, time to market, environment, regulation, cost, etc.

### **ClassIdentification**

Concerns about identification of data representing classes

### **Communication**

Concerns related to the exchange of information internal to the CPS and between the CPS and other entities.

### **Complexity**

Concerns about our understanding of the behavior of CPS due to the richness and heterogeneity of interactions among its components, such as existence of legacy and the variety of interfaces.

### **ComponentInventory**

Concerns about component inventory or 'bill of materials' (BoM) and versioning.

### **CompositionAspect**

Concerns about 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.

### **Conceptualization**

The conceptualization facet is the set of activities, including requirements development and requirements analysis and others, that produce as an output the set of properties of the system under consideration, or the Model of the CPS.

### **Concern**

Dimensions of conceiving, realizing and assuring a CPS through the lifecycle.

### **Confidentiality**

Concerns about preserving authorized restrictions on access and disclosure.

### **Configuration**

Concerns about the management of the configuration of data producing entities

### **ConfigurationForLeastFunctionality**

Concerns about minimizing the configuration needed.

### **ConfigurationManagement**

Concerns about managing of configuration, versions, etc., including identifying these and their relationships.

### **Constructivity**

Concerns about the ability to combine CPS modular components (hardware, software, and data) to satisfy user requirements.

### **Controllability**

Ability of a CPS to control a property of a physical thing. There are many challenges to implementing control systems with CPS including the non-determinism of cyber systems, the uncertainty of location, time and observations or actions, their reliability and security, and complexity. Concerns related to the ability to modify a CPS or its function, if necessary.

### **Cost**

Concerns related to the direct and indirect investment or monetary flow or other resources required by the CPS throughout its lifecycle.

### **Cybersecurity**

Concerns about cyber-security, which encompasses confidentiality, integrity and availability.

### **DataAspect**

Concerns about data interoperability including data semantics, identity, operations on data, relationships between data, and velocity of data.

### **DataIntegrity**

Concerns about the integrity of the data elements

### **DataModels**

Concerns about the structure of data/information

### **DataQuality**

Concerns about associated quality metrics of data

### **DataSemantics**

Concerns related to the agreed and shared meaning(s) of data held within, generated by, and transiting a system.

### **Deployability**

Concerns about the ease and reliability with which a CPS can be brought into productive use.

### **DeviceIdentification**

Concerns about identification of data representing devices

### **Directionality**

Concerns about directional/non-directional relationships between data

### **Discoverability**

Concerns about the ease and reliability with which a CPS component can be observed and understood (for purposes of leveraging the component’s functionality) by an entity (human, machines). Concerns related to the ease and reliability with which a CPS component’s functions can be ascertained (for purposes of leveraging that functionality) by an entity (human, machines).

### **Disposability**

Concerns about the impacts that may occur when the CPS is taken physically out of service.

### **EOL**

### **Engineerability**

Concerns about the ease and reliability with which a CPS design concept can successfully be realized via a structured engineering process.

### **Enterprise**

Concerns related to the economic aspects of CPS throughout their lifecycle.

### **Environment**

Concerns related to the impacts of the engineering and operation of a CPS on the physical world.

### **Functional**

### **FunctionalAspect**

The Functional Aspect is a set of concerns related to the sensing, computational and actuation functions of the CPS.

### **Functionality**

Concerns related to the function that a CPS provides.

### **Fusion**

Concerns about the fusion and aggregation of data.

### **Granularity**

Concerns about the granularity of data

### **HumanAspect**

Concerns about human interaction with and as part of a CPS.

### **HumanFactors**

Concern about the characteristics of CPS with respect to how they are used by humans.

### **Identity**

Concerns about the ability to accurately recognize entities (people, machines, and data) when interacting with or being leveraged by a CPS.

### **Inheritance**

Concerns about inheritance relationships of data

### **Integrity**

Concerns about guarding against improper modification or destruction of system, and includes ensuring non-repudiation and authenticity.

### **IntervalAndLatencyControl**

Concerns about time intervals. A time-interval is the duration between two instants read on the same timescale. CPS timing requirements are generally expressed as constraints on the time intervals (TI) between pairs of system significant events. There can be more general constraints on TIs that can be categorized based on their degree of time awareness in terms of bounded TIs or latency, deterministic TIs, and accurate TIs.

### **LifecyscleAspect**

Concerns about the lifecycle of CPS including its components.

### **LocationOfData**

Concerns about location of data

### **LogicalTime**

Concerns related to the order in which things happen (causal order relation).

### **MOL**

### **Maintainability**

Concerns about the ease and reliability with which the CPS can be kept in working order.

### **Manageability**

Concerns about managing a CPS through its life cycle.

### **Meaning**

Concerns about the semantic meaning of data elements

### **Measurability**

Concerns related to the ability to measure the characteristics of the CPS.

### **Monitorability**

Concerns related to the ease and reliability with which authorized entities can gain and maintain awareness of the state of a CPS and its operations. Includes logging and audit functionality.

### **NetworkInteroperability**

Concerns about networked message exchange

### **Networkability**

Concerns about the ease and reliability with which a CPS can be incorporated within a (new or existing) network of other systems.

### **OSIApplication**

Concerns about application layer exchange patterns

### **OSIDataLink**

Concerns about data links

### **OSINetwork**

Concerns about networked exchange

### **OSIPhysical**

Concerns about physical communication links

### **OSIPresentation**

Concerns about the syntax of message encoding

### **OSISession**

Concerns about session creation, maintenance, and tear-down

### **OSITransport**

Concerns about information transport

### **ObjectIdentification**

Concerns about identification of data representing objects

### **Operatability**

Concerns about the operation of the CPS when deployed.

### **OperationView**

### **OperationsOnData**

Concerns about the ability to create/read/update/delete system data and how the integrity of CPS data and behaviors may be affected.

### **Performance**

Concerns related to the ability of a CPS to meet required operational targets.

### **PhysicalContext**

Concerns relating to the need to understand a specific observation or a desired action relative to its physical position (and uncertainty.) While this information is often implied and not explicit in traditional physical systems, the distributed, mobile nature of CPS makes this a critical concern.

### **PhysicalProperties**

Concerns about purely physical properties of CPS including seals, locks, safety, and EMI.

### **PhysicalSecurity**

Concerns about physical security.

### **Planning**

Concerns about planning.

### **Policy**

Concerns related to the impacts of treaties, statutes, and doctrines on a CPS throughout its lifecycle.

### **Polymorphism**

Concerns about polymorphism of data

### **Preservability**

### **Privacy**

Concerns related to the ability of the CPS to prevent entities (people, machines) from gaining access to data stored in, created by, or transiting a CPS or its components such that individuals or groups cannot seclude themselves or information about themselves from others. Privacy is a condition that results from the establishment and maintenance of a collection of methods to support the mitigation of risks to individuals arising from the processing of their personal information within or among systems or through the manipulation of physical environments.

### **Procurability**

Concerns about the ease and reliability with which a CPS can be obtained.

### **Producibility**

Concerns about the ease and reliability with which a CPS design can be successfully manufactured.

### **Property**

Property of a CPS, included in the Model of a CPS, that is the result of applying one or more of the concerns of the CPS Framework.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| statement | char | Property value statement. |

|  |  |  |
| --- | --- | --- |
| trace | char | Trace is here to enable tracing of one property or requirement to another (typically capturing the relationships 'satisfies', 'tests', 'complies with' |

|  |  |  |
| --- | --- | --- |
| priority | float | Priority is included to enable a case where properties can be weighted or prioritized. |

|  |  |  |
| --- | --- | --- |
| reference | char | Reference such as standard, regulatory best practice, .... See ArgumentationKind. |

|  |  |  |
| --- | --- | --- |
| description | char | Description of the property |

### **Provenance**

Concerns about the provenance of the source of data

### **Quality**

Concerns related to the ease and reliability of assessing whether a CPS meets stakeholder (especially customer) expectations.

### **Regulatory**

Concerns related to regulatory requirements and certifications.

### **RelatedData**

Concerns about metadata and other related data

### **RelationshipsBetweenData**

Concerns about how and why sets of data must, may, or may not be associated with each other and the value or harm that can be derived from those associations.

### **Reliability**

Concerns related to the ability of the CPS to deliver stable and predictable performance in expected conditions.

### **Resiliance**

Concerns related to the ability of the CPS to withstand instability, unexpected conditions, and gracefully return to predictable, but possibly degraded, performance.

### **ResourceIdentification**

Concerns about identification of data.

### **Responsibility**

Concerns about the ability to identify the entity or entities authorized to control the operation of a CPS.

### **Safety**

Concerns related to the ability of the CPS to ensure the absence of catastrophic consequences on the life, health, property, or data of CPS stakeholders and the physical environment.

### **Security**

Concerns related to the ability of the CPS to ensure that all of its processes, mechanisms, physical and data, and services are afforded internal or external protection from unintended and unauthorized access, change, damage, destruction, or use. Security encompasses both cyber-security and physical security.

### **Sensing**

Concerns related to the ability of a CPS to develop the situational awareness required to perform to its function.

### **StatesAndModes**

Concerns about states and modes of a CPS.

### **Storage**

Concerns about the volume and storage of data.

### **Synchronization**

Concerns about whether all associated nodes have access to the same reference timing signals. There are three kinds of synchronization: time, phase, and frequency synchronization, although frequency synchronization is also called syntonization.

### **SyntacticInteroperability**

Concerns about the presentation and exchange patterns of data

### **TimeAwareness**

Concerns about enabling time correctness by design. 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.

### **TimeOfData**

Concerns about time stamping of data through its life cycle

### **TimeToMarket**

Concerns related to the time period required to bring a CPS from need realization through deployment.

### **TimingAndLatency**

Control and measurement applications in CPS generally require specific management of timing and latency. General important concerns for managing timing and latency include the coordination of timing domains and multiple timing references; latency management and control, including scheduling of network timing guarantees; and the mixing of time-sensitive and best-effort data in both networks and CPS nodes.

### **TimingAspect**

Concerns about time and frequency in CPS, including the generation and transport of time and frequency signals, timestamping, managing latency, timing composability, etc.

### **Transformation**

Concerns about the transformation of data from one form/semantic to another

### **TrustworthinessAspect**

Concerns about trustworthiness of CPS including cybersecurity, privacy, safety, reliability, and resilience.

### **Uncertainty**

Managing the effects of uncertainties is a fundamental challenge in CPS. Sources of uncertainty in CPS can be grouped into statistical (aleatoric), lack of knowledge (epistemic) uncertainty, or systematic uncertainty. In CPS statistical uncertainty is caused by randomness of accuracy of sensing and actuation, often caused by uncertainty of manufacturing processes. Systematic uncertainty is caused by incomplete knowledge either due to limits of acquired knowledge or due to simplification in modeling. Typical manifestations of epistemic uncertainty are limited validity of models of physical processes or limits of computability of properties of mathematical models.

### **Usability**

Concerns related to the ability of CPS to be used to achieve its functional objectives effectively, efficiently, and to the satisfaction of users (adapted from ISO 9241-210.) The combination of physical and cyber into complex systems creates challenges in meeting usability goals. Complexity is a major issue. The diversity of interfaces creates a significant learning curve for human interaction.

### **UserIdentification**

Concerns about identification of data representing users.

### **Utility**

Concerns related to the ability of a CPS to provide benefit or satisfaction through its operation. Utility reflects a business concern, especially when considered as the numerator when computing value, which equals utility divided by costs.

### **Velociy**

Concerns about the velocity of data.

### **Verification**

Concerns about the verification of the identity of data

## 

## **Realization**

This package contains definitions for the realization facet.

### **Design**

This class contains the design of the system, including the design requirements and design elements such as drawings of the physical elements of the system and source and implementation code of the logical elements of the system.

### **Realization**

The realization facet is the set of activities, including design requirements generation, prototyping, verification and validation, that produce as an output the system itself.

### **TestCase**

The Test Case class consists of the test cases, including test, inspection, analysis or demonstration, that relate to one or more of design requirement.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| kind | VerificationKind |  |

## 

## **UseCaseModelDerived**

This package contains definitions derived from IEC standard 62559-2.

### **ContextualModel\_Dependency diagram**



ContextualModel\_Dependency

### **Assumption**

May be used to define further, general assumption for a use case, such as: which systems already exist, which contractual relations exist, and which configurations of systems are probably in place. Initial states of information exchanged shall also be identified.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | Name of the assumption |

|  |  |  |
| --- | --- | --- |
| content | string | Description of the assumption |

### **Author**

This field is used to document who has provided the current version. It can be a person, organisation or e.g. standardization committee like TC or WG.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

### **BusinessCase**

Provides a description or reference with some rationale for the suggested use case. Usually the business case is related to several use cases.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| Domain | Ref\_Domain | Use cases can be used in various areas (e.g. energy system). Within these areas different domains are used to define / determine a more specific subgrouping. One or more domains and zones, comma separated, can be specified in the template. These are the other ones. |

### **BusinessCaseLibrary**

The Business Case Library contains the business cases of the repository. These are high-level descriptions of the value proposition of the CPS, i.e., a high-level statement of what the CPS 'does' as well as a description of the desired level of effort to deliver that function, i.e., direct and indirect costs, etc.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| BusinessCase | BusinessCase | the list of business cases |

### **CPS**

Entity that communicates and interacts.

These actors can include people, software applications, systems, databases, and even the power system itself.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| type | string | Type of the actor (business, system, ....) |

### **CPSGrouping**

Group of actors used to organize an actor list.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| CPS | UseCase\_Ref\_CPS | list of grouped Actors |

### **CPSLibrary**

This repository contains all of these actors or 'subsystems' of the CPS, that may interact logically, physically or cyber-physically.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| cps | IrreducibleCPS | An irreducible CPS. |

|  |  |  |
| --- | --- | --- |
| system | System | A system or subsystem of irreducible CPS. |

|  |  |  |
| --- | --- | --- |
| systemofsystems | SystemOfSystems | A system of systems CPS. |

### **CommonTerm**

Represents a glossary term and its definition.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

### **CommonTermLibrary**

Common glossary defining terms for all use cases.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| CommonTerm | CommonTerm | list of common terms |

### **Condition**

Describes either a precondition or a postcondition.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | Name of the condition. |

|  |  |  |
| --- | --- | --- |
| content | string | Description of the condition. |

### **CustomInformation**

Provides a flexible option to include miscellaneous custom, semi-structured information which does not fit into other parts of the template.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| key | string | Use case wide unique key for identification purposes. |

|  |  |  |
| --- | --- | --- |
| value | string | Provides the corresponding information for the provided key. |

|  |  |  |
| --- | --- | --- |
| reference | string | This field refers to relevant template section. |

### **Domain**

A domain is an area of knowledge or activity characterized by a set of concepts and terminology understood by the practitioners in that area. Examples include transportation, medical, energy, etc.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

### **DomainLibrary**

Domains of the repository.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| Domain | Domain | the list of domains within the library |

### **Drawing**

For clarification, in general it is recommended to provide drawing(s) by hand, by a graphic or as UML graphics (preferred in this clause). The drawing should show interactions which identify the steps where possible.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | Name of the drawing. |

|  |  |  |
| --- | --- | --- |
| drawingType | DrawingClassification | Type of drawing. |

|  |  |  |
| --- | --- | --- |
| URI | Resource\_String | resource Path to an image file  ex: http://www..../image.jpg  or  path to an XMI file with a relative diagram path from the root of the UML model. ex: http://www.../Model.xmi#Package1/Package2/.../DiagName |

### **Function**

Entity that communicates and interacts.

These actors can include people, software applications, systems, databases, and even the power system itself.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| type | string | Type of the actor (business, system, ....) |

### **InformationModel**

Information model exchanged in detailed activity step.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| Requirement | Ref\_Requirement | can be used to define requirements referring to the information and not to the step as in the step by step analysis |

### **InformationModelLibrary**

Recall that a 'use case' of a CPS contains one or more actions, 'scenarios' or features, each of which consists of a finite number of steps. These steps consist of interactions between the actors of the scenario.

The elements of the Interaction Library can be either logical or physical or cyber-physical. A logical interaction is an exchange of information between the actors of the CPS. The 'business objects' of the repository are the data that make up the information in such an exchange.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| InformationModel | InformationModel | list of business objects within the library |

### **KeyPerformanceIndicator**

Key Performance Indicator (KPI) related to the objectives of the Use Case.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| Objective | Ref\_Objective | the related objective |

### **MacroActivity**

Macro-activity.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| PrimaryCPS | Ref\_CPS | reference to the PrimaryActor |

|  |  |  |
| --- | --- | --- |
| Requirement | Ref\_Requirement | the associated requirements |

|  |  |  |
| --- | --- | --- |
| ChildMacroActivity | MacroActivity | The list of sub macro activities |

|  |  |  |
| --- | --- | --- |
| Drawing | Drawing | the list of associated drawing. |

|  |  |  |
| --- | --- | --- |
| Step | Step | list of steps |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Narrative**

Describes the narrative of the use case.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| shortDescription | string | Short text intended to summarize the main idea as service for the reader who is searching for a use case or looking for an overview.  Recommendation: This short description should have not more than 150 words. |

|  |  |  |
| --- | --- | --- |
| completeDescription | string | Provides a complete narrative of the use case from a user’s point of view, describing what occurs when, why, with what expectation, and under what conditions. This narrative should be written in plain text so that non-domain experts can understand it.  The length of the complete description can range from a few sentences to a few pages, depending on the complexity and / or newness of the use case. This description often helps the domain expert to reflect about the requirements for the use case before getting into the details in the next sections of the use case template.  The description may include drawings for explanation. |

### **Objective**

Objective of the use case.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

### **Ref\_BusinessCase**

Provides a description or reference with some rationale for the suggested use case. Usually the business case is related to several use cases.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Ref\_CPS**

Entity that communicates and interacts.

These actors can include people, software applications, systems, databases, and even the power system itself.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Ref\_CommonTerm**

Represents a glossary term and its definition.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Ref\_Domain**

Area of knowledge or activity characterized by a set of concepts and terminology understood by the practitioners in that area .

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Ref\_InformationModel**

Reference to an information model exchanged in detailed activity step.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Ref\_Objective**

Objective of the use case.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Ref\_Requirement**

Requirement of a part of the use case.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Ref\_UseCase**

Specification of a set of actions performed by a system, which yields an observable result that is, typically, of value for one or more actors or other stakeholders of the system

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

### **Reference**

Any references shall be identified that might restrict or affect the design, understanding, and requirements of the use case, including contracts, regulations, policies, financial considerations, engineering constraints, pollution constraints, and other environmental quality issues.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| type | string | There are different reference types (e.g. standards, regulation, contract, others like publications). |

|  |  |  |
| --- | --- | --- |
| impact | string | This information describe the nature of the influence of the document on the use case. |

|  |  |  |
| --- | --- | --- |
| status | string | Status of the referenced document. |

|  |  |  |
| --- | --- | --- |
| link | string | If available, a public link to the document can be provided. |

|  |  |  |
| --- | --- | --- |
| originatorOrganisation | string | This describes the name of the organisation publishing the document. |

### **Remark**

Used for further comments which are not considered elsewhere.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| content | string | Content of the remark. |

### **Requirement**

Requirement of a part of the use case.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

### **RequirementCategory**

Requirements can be sorted in categories. For complex categories the category name can be "dot-delimited".

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| Requirement | Requirement | information about the requirement |

### **RequirementLibrary**

Requirements of the repository.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| RequirementCategory | RequirementCategory | The category for the requirement |

### **Resource\_String**

String Primitive type.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| type | ResourceType |  |

### **Resource\_String\_value\_ValueType**

### **Scenario**

A possible sequence of interactions.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| PrimaryCPS | Ref\_CPS | Describes which actor triggers this scenario. |

|  |  |  |
| --- | --- | --- |
| Drawing | Drawing | the list of associated drawing. |

|  |  |  |
| --- | --- | --- |
| TriggeringEvent | TriggeringEvent | list of triggering events for the scenario |

|  |  |  |
| --- | --- | --- |
| Precondition | Condition | Describes which condition(s) should prevail after this scenario happens The post conditions may also define "success" or "failure" conditions for each use case. |

|  |  |  |
| --- | --- | --- |
| Postcondition | Condition | Describes which condition(s) should have been met before this scenario happens. |

|  |  |  |
| --- | --- | --- |
| Requirement | Ref\_Requirement | information about the requirement |

|  |  |  |
| --- | --- | --- |
| MacroActivity | MacroActivity | list of macro activities for the scenario |

### **Step**

Elementary step within a scenario representing the finest-grained description level of interactions in the use case.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| identifier | string | Unique global identifier for the object.  This identifier is unique among several exchange of information. A global repository has to maintain the uniqueness. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| event | string | The event that triggers the step. This might be completion of previous steps (in that case the event is the comma-separated list of their numbers) or the condition to exit a loop. |

|  |  |  |
| --- | --- | --- |
| service | string | This column identifies the nature of the information flow and the originator of the information. Available options are CREATE, GET, CHANGE, DELETE, CANCEL, EXECUTE derived from IEC 61968-100:2013.  Additionally, REPORT, TIMER and REPEAT are suggested.   * CREATE means that an information object is to be created at the Producer. * GET (this is the default value if none is populated) means that the Receiver requests information from the Producer (default). * CHANGE means that information is to be updated. Producer updates the Receiver’s information. * DELETE means that information is to be deleted. Producer deletes information from the Receiver. * CANCEL, CLOSE imply actions related to processes, such as the closure of a work order or the cancellation of a control request. * EXECUTE is used when a complex transaction is being conveyed using a service, which potentially contains more than one verb. * REPORT is used to represent transferral of unsolicited information or asynchronous information flows. Producer provides information to the Receiver. * TIMER is used to represent a waiting period. When using the TIMER service, the Information Producer and Information Receiver fields shall refer to the same actor. * REPEAT is used to indicate that a series of steps is repeated until a condition or trigger event. The condition is specified as the text in the "Event" column for this row or step. Following the word REPEAT, shall appear, in parenthesis, the first and last step numbers of the series to be repeated in the following form REPEAT(X-Y) where X is the first step and Y is the last step.   These common service definitions are related to automation / information or communication systems. In case the use case template is applied in other domains, further services might be used and described. |

|  |  |  |
| --- | --- | --- |
| Drawing | Drawing | the list of associated drawing. |

|  |  |  |
| --- | --- | --- |
| Requirement | Ref\_Requirement | Requirement(s) for the step. It is allowed to list several requirement in one step, comma separated. |

|  |  |  |
| --- | --- | --- |
| InformationReceiver | Ref\_CPS | This identifies the receiver of the information. This should be one of the actors defined in the repository. |

|  |  |  |
| --- | --- | --- |
| InformationProducer | Ref\_CPS | This identifies the producer or source of the information. This should be one of the actors defined in the repository. |

|  |  |  |
| --- | --- | --- |
| BusinessObject | Ref\_InformationModel | This describes briefly the information to be exchanged between the two actors – information producer and information receiver. It is possible to several information in one step, comma separated. |

|  |  |  |
| --- | --- | --- |
| interaction | Ref\_Message |  |

### **TriggeringEvent**

Describes which event triggers a scenario.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | Name of the triggering event. |

|  |  |  |
| --- | --- | --- |
| content | string | Description of the triggering event. |

### **UseCase**

Specification of a set of actions performed by a system, which yields an observable result that is, typically, of value for one or more actors or other stakeholders of the system

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| nature | string | Classifies the main focus of the use case. |

|  |  |  |
| --- | --- | --- |
| classification | string | On international level the use case description might be generic enough to describe a use case in a more general way independently from the national or regional market design. But use cases might be used to describe regional or national specific circumstances like laws or even project specific details. If the use case reflects to those circumstances, it should be characterized accordingly. |

|  |  |  |
| --- | --- | --- |
| keywords | string | Keywords can be defined in order to support extended search functionalities within a use case repository. Multiple keywords should be provided as a comma-separated list. |

|  |  |  |
| --- | --- | --- |
| levelOfDepth | string | Use cases can be described on different levels, for example high-level use case, detailed use case, specialised use case, ... |

|  |  |  |
| --- | --- | --- |
| prioritisation | string | Considering a larger number of use cases it might be interesting to cluster them according to priority.  This prioritisation might be different from country to country. |

|  |  |  |
| --- | --- | --- |
| scope | string | The scope defines the limits of the use case. |

|  |  |  |
| --- | --- | --- |
| RelatedUseCase | Ref\_UseCase | Known relations to other use cases can be provided here if e.g. the use case is a more detailed one related to a high level use case, or it is an alternative to an existing use case. |

|  |  |  |
| --- | --- | --- |
| CPSGrouping | CPSGrouping | a Use Case references several set of Actors |

|  |  |  |
| --- | --- | --- |
| PrimaryCPS | Ref\_CPS | Identification of the PrimaryActor linked by the use case. |

|  |  |  |
| --- | --- | --- |
| RelatedObjective | Objective | List of objectives of the use case. |

|  |  |  |
| --- | --- | --- |
| Assumption | Assumption | set of assumptions related to the use case |

|  |  |  |
| --- | --- | --- |
| Prerequisite | Condition | Describes what condition(s) should have been met before initiation of the use case, such as prior state of the actors and activities. |

|  |  |  |
| --- | --- | --- |
| KeyPerformanceIndicator | KeyPerformanceIndicator | information about the KPI related to the use case |

|  |  |  |
| --- | --- | --- |
| BusinessCase | Ref\_BusinessCase | list of business cases associated to the use case |

|  |  |  |
| --- | --- | --- |
| Narrative | Narrative | the information about the narrative part of the use case. |

|  |  |  |
| --- | --- | --- |
| Reference | Reference | information about referenced documents. |

|  |  |  |
| --- | --- | --- |
| Remark | Remark | remark associated to the use case. |

|  |  |  |
| --- | --- | --- |
| Drawing | Drawing | the list of associated drawing. |

|  |  |  |
| --- | --- | --- |
| Version | Version | list of versions describing changes about the use case |

|  |  |  |
| --- | --- | --- |
| CustomInformation | CustomInformation | Related additional information |

|  |  |  |
| --- | --- | --- |
| CommonTerm | Ref\_CommonTerm | the associated common terms. |

|  |  |  |
| --- | --- | --- |
| Scenario | Scenario | reference to the scenarios belonging to the use case. |

### **UseCaseLibrary**

Use cases of the repository.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| UseCase | UseCase | list of use cases within the library |

### **UseCaseRepository**

### **UseCaseRepository\_Type**

Database, based on a given use cases template, for editing, maintenance and administration of use cases, actors and requirements including their interrelations.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | string | Description of the object. |

|  |  |  |
| --- | --- | --- |
| DomainLibrary | DomainLibrary | the associated library of domains |

|  |  |  |
| --- | --- | --- |
| BusinessObjectLibrary | InformationModelLibrary | the library of business objects |

|  |  |  |
| --- | --- | --- |
| RequirementLibrary | RequirementLibrary | The associated library |

|  |  |  |
| --- | --- | --- |
| ActorLibrary | CPSLibrary | Library of actors |

|  |  |  |
| --- | --- | --- |
| CommonTermLibrary | CommonTermLibrary | the associated library of common terms |

|  |  |  |
| --- | --- | --- |
| BusinessCaseLibrary | BusinessCaseLibrary | the library associated to the repository |

|  |  |  |
| --- | --- | --- |
| UseCaseLibrary | UseCaseLibrary | the associated library of use cases |

### **UseCase\_Ref\_CPS**

Entity that communicates and interacts.

These actors can include people, software applications, systems, databases, and even the power system itself.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | string | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | string | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| furtherInformation | string | Individual or additional information that relates to the use case can be provided. |

### **Version**

Version of the use case(s) description.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| versionNumber | string | Sequential number to identify the version of the document. |

|  |  |  |
| --- | --- | --- |
| date | dateTime | Date of creation of the version, in format YYYY-MM-DD. |

|  |  |  |
| --- | --- | --- |
| changes | string | Differences with the previous version.  Multiple changes are separated with paragraphs. |

|  |  |  |
| --- | --- | --- |
| approvalStatus | string | Information used within the standardization process. |

|  |  |  |
| --- | --- | --- |
| Author | Author | list of authors participating in the changes of this version |

### **DrawingClassification**

Possible types of drawing.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| Domain Overview | string |  |

|  |  |  |
| --- | --- | --- |
| Use Case Overview | string |  |

|  |  |  |
| --- | --- | --- |
| Documentation | string |  |

|  |  |  |
| --- | --- | --- |
| Scenarios Flowchart | string |  |

|  |  |  |
| --- | --- | --- |
| Scenario Overview | string |  |

|  |  |  |
| --- | --- | --- |
| MacroActivities Flowchart | string |  |

|  |  |  |
| --- | --- | --- |
| MacroActivity Overview | string |  |

|  |  |  |
| --- | --- | --- |
| Actions Flowchart | string |  |

|  |  |  |
| --- | --- | --- |
| Action Overview | string |  |

|  |  |  |
| --- | --- | --- |
| Business Objects Overview | string |  |

|  |  |  |
| --- | --- | --- |
| other | string |  |

|  |  |  |
| --- | --- | --- |
| Role Model | string |  |

### **ResourceType**

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| Image | string |  |

|  |  |  |
| --- | --- | --- |
| UMLDiagram | string |  |

## **Aspect**

Aspects are high level concerns or collections of concerns whose relationships and tradeoffs are more fully understood.

## **Aspects**

This is the set of aspects or high-level concerns of the CPS Framework. They may be applicable at the levels of the decomposition of the CPS that takes place in the conceptualization facet of the CPS Framework.

The decomposition of a CPS in the CPS Framework proceeds from the Business Case, Use Case, CPS's (in the case of a system of systems) and Allocation to the Logical and the Physical.

These aspects, nine of them in the CPS Framework, are the high-level concerns or 'categories of concerns' of the CPS Framework. Applying a concern to a level of the decomposition of a CPS results in a set of 'properties' (or requirements), designed to assure that the concern is addressed by the CPS.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| functional | FunctionalAspect |  |

|  |  |  |
| --- | --- | --- |
| business | BusinessAspect |  |

|  |  |  |
| --- | --- | --- |
| human | HumanAspect |  |

|  |  |  |
| --- | --- | --- |
| trustworthiness | TrustworthinessAspect |  |

|  |  |  |
| --- | --- | --- |
| timing | TimingAspect |  |

|  |  |  |
| --- | --- | --- |
| data | DataAspect |  |

|  |  |  |
| --- | --- | --- |
| boundaries | BoundariesAspect |  |

|  |  |  |
| --- | --- | --- |
| composition | CompositionAspect |  |

|  |  |  |
| --- | --- | --- |
| lifecylce | LifecyscleAspect |  |

## **CPSFramework**

The CPS Framework class denotes the CPS Framework UML Model. All the class objects in this model inherit Maturity attributes such as effective date, maturity level, responsible party, revision level and maturity status.

It's presence here guarantees that all of the elements of the CPS Framework inherit values of the attributes associated with the Maturity Class Object.

## **Energy**

Energy exchanged by an influence.

## **EnergyDomain**

The energy domain of CPS.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | char |  |

|  |  |  |
| --- | --- | --- |
| description | char |  |

## **Facet**

The facets of the CPS Framework are sets of activities performed during the development of a system. It includes the conceptualization facet, the realization facet and the assurance facet.

## **Human**

The human representation as a participant or part of a CPS.

## **Influence**

An interaction in the physical sense involving the exchange of energy.

## **Influences**

Physical parts of CPS interact through influences which involve the exchange of energy.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| influence | Influence | An influence. |

## **Interaction**

An interaction is re-usable description of a message or an influence. An interaction is governed by a set of properties.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | char | A short name. |

|  |  |  |
| --- | --- | --- |
| technicalId | char | Technical unique identifier for the object for the scope of a specific exchange of information (i.e This is unique only for the exchanged file). This identifier is used to create technical references of information within the exchange of information. Even if the global identifier has a global scope, the technicalId is always used instead when the purpose is to create reference information within the current exchange. |

|  |  |  |
| --- | --- | --- |
| description | char |  |

## **InteractionLibrary**

A library of interactions that define the messaging patterns between the logical elements of the CPS and the influences between physical elements of the CPS as well as the cyber-physical interactions between the logical and physical elements of the CPS.

## **IrreducibleCPS**

The minimal assembly of hardware and software that can be considered a CPS.

## **Logical**

The logical part of a CPS representing its state and algorithms.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| state | char |  |

|  |  |  |
| --- | --- | --- |
| Ref\_msgs | Ref\_Message | reference to messages (peer to influences) |

## **Maturity**

Describes the maturity level of a component of the CPSFramework model.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| maturity | maturityLevels |  |

|  |  |  |
| --- | --- | --- |
| effective | char | Effective date. ISO 8601 string. |

|  |  |  |
| --- | --- | --- |
| responsibleParty | char | responsible party |

|  |  |  |
| --- | --- | --- |
| revision | char | Version string |

|  |  |  |
| --- | --- | --- |
| status | MaturityStatus |  |

## **Message**

An interaction between logical parts of CPS involving the exchange of information.

## **Messages**

Logical parts of CPS interact via messages which involve the exchange of information.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| message | Message |  |

## **Physical**

The physical part of CPS that operate according to the laws of physics.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| state |  |  |

|  |  |  |
| --- | --- | --- |
| Ref\_Influences | Ref\_Influence | Reference to other Physical objects that this one influences. E.g. multiphysics influences between electrical and thermal properties. |

## **Ref\_Influence**

A reference to an influence.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| refInfluence | String | XPath reference to an influence. |

## **Ref\_IrreducibleCPS**

Reference to an irreducibleCPS.

## **Ref\_Message**

A reference to a message.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| refMessage | String | Contains a reference to a message. |

## **Ref\_Property**

Reference to a property.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| refProperty | char | URI reference to a property |

## **System**

A composition of 2 or more IrreducibleCPS.

## **SystemOfSystems**

A composition of 2 or more Systems of CPS.

## **TransportationDomain**

The transportation domain of CPS.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| name | char |  |

|  |  |  |
| --- | --- | --- |
| description | char |  |

## **reference**

An external reference to a schematic, document, model, plan, etc....

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| reference | char | external reference to document, etc... |

|  |  |  |
| --- | --- | --- |
| description | char | Description of what is being referred to. |

## **MaturityStatus**

This enumeration provides the values of the maturity status of the elements of the CPS decomposition, from as developed to as designed, as planned, as built, as maintained, as disposed to as presented.

The status of maturity of an instance of the CPS reflects where the CPS elements are in its lifecycle.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| asDeveloped |  |  |

|  |  |  |
| --- | --- | --- |
| asDesigned |  |  |

|  |  |  |
| --- | --- | --- |
| asPlanned |  |  |

|  |  |  |
| --- | --- | --- |
| asBuilt |  |  |

|  |  |  |
| --- | --- | --- |
| asMaintained |  |  |

|  |  |  |
| --- | --- | --- |
| asDisposed |  |  |

|  |  |  |
| --- | --- | --- |
| asPreserved |  |  |

## **maturityLevels**

This is an enumeration object in the maturity class diagram. Enumerations are used in a class diagram to define the value sets or possible values that the attributes of a class object can assume.

This enumeration is an enumeration of 'maturity levels', the values that the maturity attribute of the Maturity class can take. The Maturity Levels enumeration can be extended to include additional custom values. The maturity levels themselves correspond to the 'steps' of the effort to develop a CPS definition. They measure whether that effort is complete, from submitted to verified, approved, delegated, rejected and released.

| **Name** | **Type** | **Notes** |
| --- | --- | --- |
| submitted |  |  |

|  |  |  |
| --- | --- | --- |
| verified |  |  |

|  |  |  |
| --- | --- | --- |
| approved |  |  |

|  |  |  |
| --- | --- | --- |
| delegated |  |  |

|  |  |  |
| --- | --- | --- |
| rejected |  |  |

|  |  |  |
| --- | --- | --- |
| reviewed |  |  |