Architecture Viewpoint Definition (AVPD) for a System-of-Interest situated in its environment

Abstract

This Architecture Viewpoint Definition (AVPD) describes the design of the System Description Architecture Description Framework for a System-of-Interest situated in its environment.. The Architecture Viewpoint Definition (AVPD) has been created using the COMPASS Architectural Framework Framework (CAFF) from the COMPASS Project.

The System Description Architecture Description Framework provides the viewpoints to create the views for a System Description. The System Description provides a complete and testable definition of a system-of-interest. The System Description template is patterned after the Root Definition used in the Soft System Methodology (SSM) to describe a Human Activity System or the list of 10 Things to understand about systems, from the book, Architecting Systems, by Hillary Sillitto.

In addition, the System Description that is produced from the System Description Architecture Description Framework can be used in the Ontology Definition View for creating Architecture Viewpoint Definition (AVPD) for the system-of-interest.

Author and Version

Bruce McNaughton, Prototype, Version 0.5, 11-January-2023

Contents

Revision History

V0.5 11-January-2023 Revise to align with ISO 42010:2022 and Change of AFD to AVPD

V0.4 21-May-2021 Revise with latest models and system definitions

V0.3 26-August-2019 Revised various diagrams as a result of revising the system conceptual model.

V0.2 11-August-2019 System Description for SOI, improved description of function / capability and behaviour, updated VRV.

V0.1 28-June-2019 Included abstract class diagram; revised template based upon usage; some text edits.

V0.0 15-June-2019 Initial Draft

Introduction: System (Abstract) Architecture Viewpoint Definition

This Architecture Viewpoint Definition (AVPD) describes the design of the System Description Architecture Description Framework for a System-of-Interest situated in its environment.. The Architecture Viewpoint Definition (AVPD) has been created using the COMPASS Architectural Framework Framework (CAFF) from the COMPASS Project.

The System Description Architecture Description Framework provides the viewpoints to create the views for a System Description. The System Description provides a complete and testable definition of a system-of-interest. The System Description template is patterned after the Root Definition used in the Soft System Methodology (SSM) to describe a Human Activity System or the list of 10 Things to understand about systems, from the book, Architecting Systems, by Hillary Sillitto.

In addition, the System Description that is produced from the System Description Architecture Description Framework can be used in the Ontology Definition View for creating Architecture Viewpoint Definition (AVPD) for the system-of-interest.
This System Description (SDSF) is the result of applying the Structuring Formalism to the abstract system as the system-of-interest.

Structuring Formalism: System Description (SDSF)

A Structuring Formalism provides a way to organize and gain insights from AD Elements for a system-of-interest Architecture Description Framework (System-of-Interest ADF). This Structuring Formalism uses a two step approach aligned to the early life cycle processes found in ISO 15288:2015. These steps are to create:

- A System Description that captures the basic structure, behaviour and properties of a system-of-interest to establish a common language and understanding within the team.
- A System-of-Interest ADF that provides stakeholders with specific viewpoints and views of the system-of-interest based upon a whole system understanding of the system-of-interest.

The System Description is created using a SysDesc ADF that allows the System Description to be considered as an AD Element in the scope of the SysDesc ADF. This allows correspondences to be identified that promote re-use of AD Elements across a wider set of system-of-interests.
The concept of a System Description is used as the basis for this Structuring Formalism: System Description (SDSF). This document contains a description of the System Description (SDSF) that is used to gain insights about the AD Elements (including other System Descriptions) to structure the System-of-Interest ADF. This Structuring Formalism contains the following:

- A conceptual model of the Structuring Formalism: System Description (SDSF) used to organize and gain insight from the AD Elements for the System-of-Interest ADF.
- The various steps for creating and using System Descriptions when gaining insight and enhancing reuse of the AD Elements .
- The Structural Categories that allow sharing and reuse of AD Elements using correspondences.
- Various links to examples and results from using the two step approach.
- The types of benefits that can be achieved using the System Description as an AD Element.

PDF: Structuring Formalism: System Description (SDSF), Version 0.4, 07-February-2023
used to create System Descriptions and related System-of-Interest ADF:

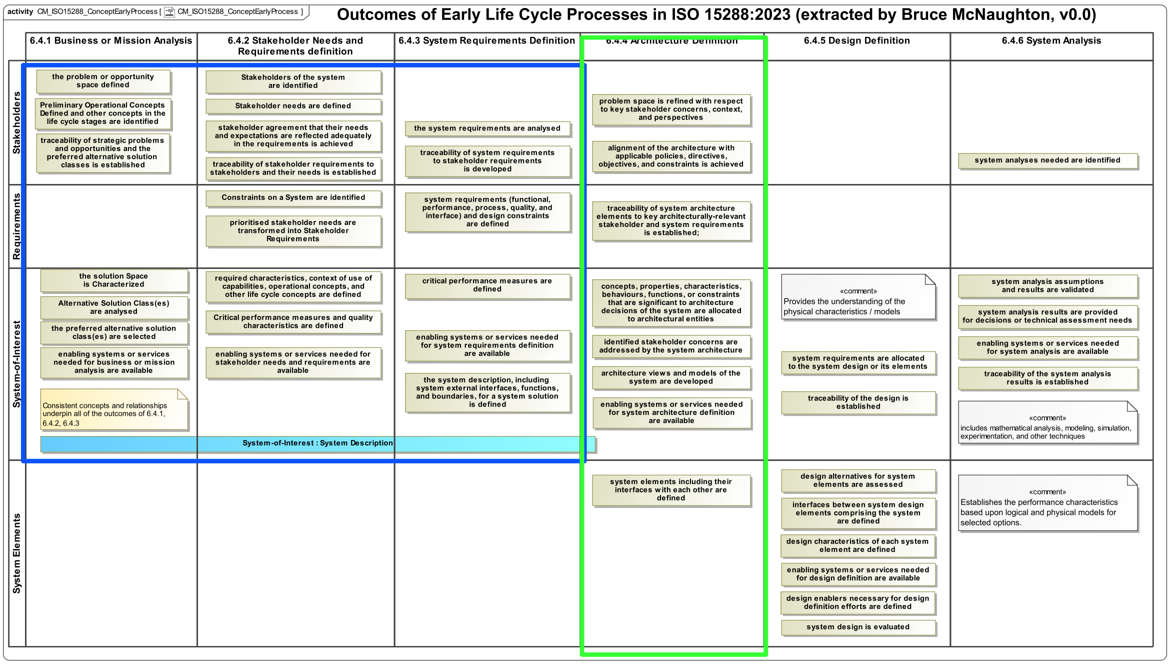
Link to the System Description Architecture Description Framework
and the example Enterprise (SoS) Architecture Description Framework
Link to the Enterprise (SoS) Architecture Description Framework

Context: Fit within the early ISO 15288:2015 life cycle processes

This System Description (SDSF) is used within the context of the life cycle of a system-of-interest identified in ISO 15288:2015. The early technical processes of the life cycle provide an opportunity to engage stakeholders about their concerns and needs for the system-of-interest. Establishing a common language and vision require conversations and co-creation of information. This can be seen as a two step process:

- Shaping the system-of-interest using a System Description (6.4.1, 6.4.2, 6.4.3) Blue Rectangle
- Creating the system-of-interest Architecture Description Framework based upon the system-of-interest System Description (6.4.4) Green Rectangle

The following picture shows the two areas where these conversations can occur:



Shaping the system-of-interest (Blue Rectangle)

The first three processes (6.4.1, 6.4.2, and 6.4.3) build an understanding of the system-of-interest through identifying stakeholders and their needs and concerns, identifying stakeholder and system requirements and understanding the language (Ontology) and structure, behavior and system properties (capabilities) of the system-of-interest. The system-of-interest row (third row from the top) co-creates an understanding of the whole system through the creation of a System Description. The system engineering role or the system architect role is generally working to build this understanding across the team. One of the outcomes from the process 6.4.3 System Requirements Definition is a System Description. This step is similar to the process "Architecture Conceptualization" in ISO 42020:2019. The System Description is created using the SysDesc ADF as an Architecture Description.

Also during this period, insights can be gained through the identification and reuse of other system descriptions. This reuse is captured through correspondences across the various AD Elements, including other System Descriptions.

Creating the system-of-interest Architecture Description Framework (Green Rectangle)

The System-of-Interest ADF provides viewpoints and views that address the full set of stakeholders and their concerns based upon the system description created in 6.4.3. The System Description provides a strong foundation for the creation of the additional AD elements needed for the System-of-Interest ADF. This step is very similar to the "Architecture Elaboration" process in ISO 42020:2019.

The insights gained from the System Description through identification of correspondences provides a way of reusing AD Elements across other Architecture Description Frameworks, such as viewpoints, model kinds, ADLs, etc.

This System Description (SDSF) Document

The Structuring Formalism consists of three key parts:

- The conceptual model of the System Description (SDSF)
- The approach to creating a SoI ADF based on the System Description (SDSF)
- The System Classification Framework for identification and use of existing systems and system descriptions.

Each of these parts are described in this System Description (SDSF).

System Classification Framework

System Classification Framework

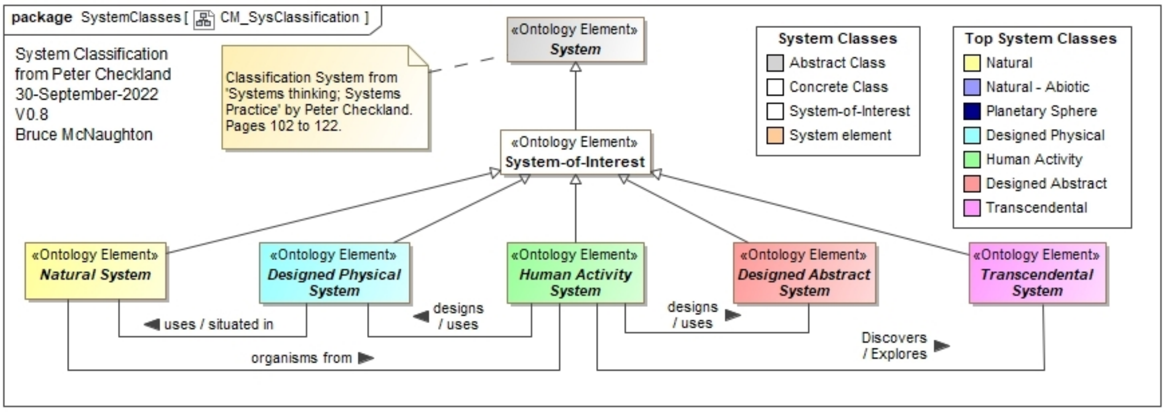
A System Classification Framework provides a way to position a system-of-interest in a wider context of systems. This System Classification Framework is used to:

- Identify types of systems.
- Promote reuse across a set of systems and system types
- Ensure alignment of similar types of systems and reduce duplicate definitions.

The System Classification Framework provides the following benefits:

- A top level set of system types that can be used for any system-of-interest.
- A way to reuse aspects of systems using generalizations that allow inheritance of the key elements of a system.
- A way to integrate across systems based upon consistent references to defined systems using a single abstract system class..
- A way to reuse AD Elements across the full set of defined systems (e.g. viewpoints, views, view components, other system descriptions, etc).

The top level System Classification Framework is based upon Peter Checkland's system classification model. Peter Checkland includes a system classification approach in his book Systems Thinking, System Practice. The following form the top level set of systems in this classification scheme:



The top level System Classification Framework is described in the book from page 102 to page 122. Figure 4, page 112 highlights the 5 system classes. These classes are used as a top level classification for system types. Link to the Top System Classifications PDF

Russell Ackoff's System Classification

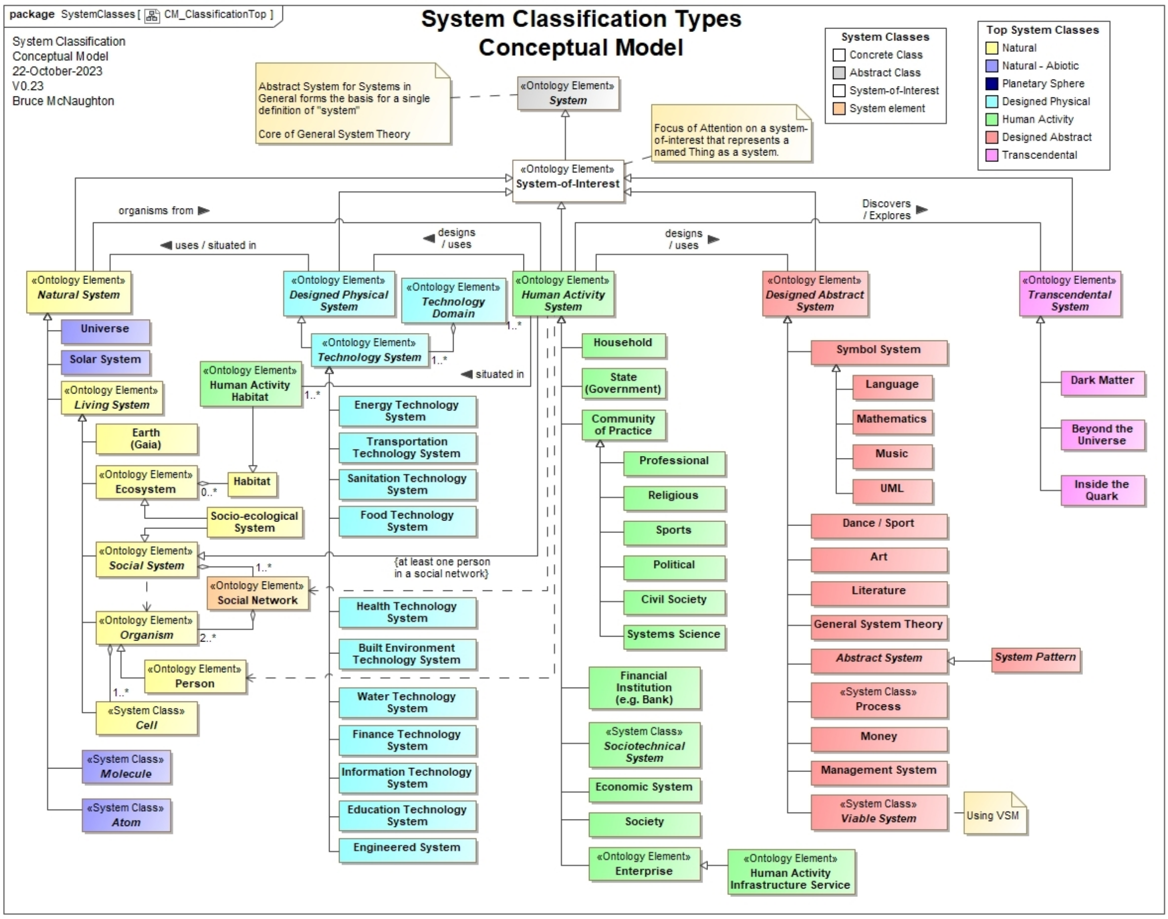
Russell Ackoff's System Classifications were also considered. The following types of systems comes from Re-Creating the Corporation

- Deterministic System
- Animated System
- Social System
- Ecological System.

These classifications were considered; however, they use are use "Purposeful System" as a differentiator between system types and was considered too narrow for this System Classification Framework.

Current Systems in the System Classification Framework

. The current systems that have been identified using the top level classification types are shown in the diagram below:

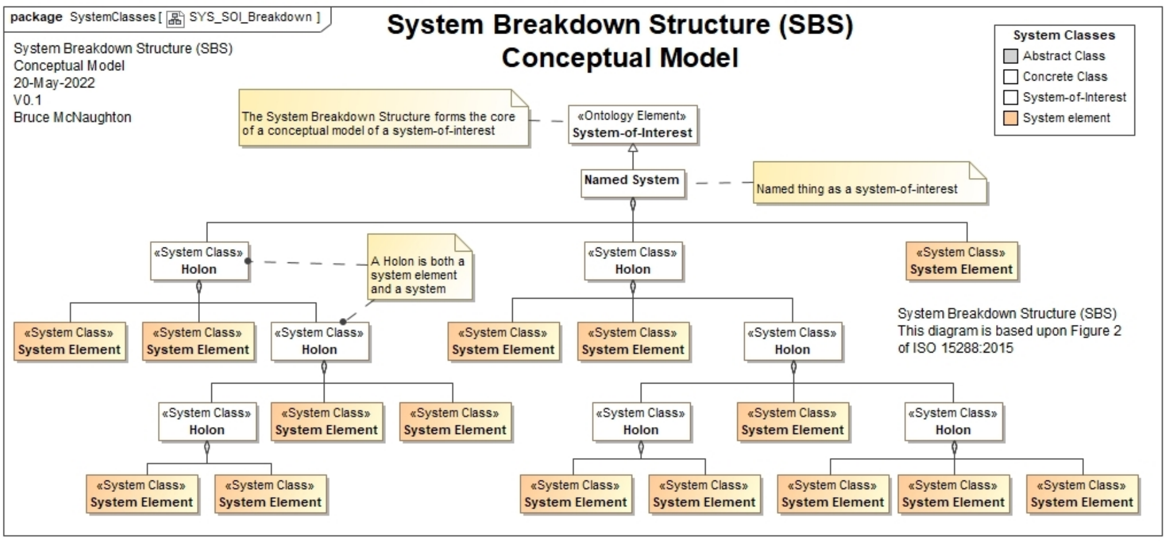


Note: that all of the types of systems are based upon a single definition and model of an abstract system. Each system inherits the single definition of system. This provides a consistent way to describe each type of system using a System Description based upon the SysDesc ADF.

System Classes

Given the consistent inheritance of a single definition of System (Abstract), any of the systems in the System Classification Framework can provide a generalization / inheritance path to retain the essence of the System Description (AD Element).

The identification of the above types of systems allows a consistent breakdown of systems. Here are some examples of further exploration of these systems: The system Breakdown Structure (equivalent to Figure 2 in ISO 15288:2015) is shown below:



Each holon can be considered a system-of-interest and may have an associated System Description and / or System-of-Interest ADF. The top named system should be the primary candidate for the Architecture Description Framework. All holons can have a System Description. These holons can also be a mix of top level system types.

The Enterprise (SoS) System Description is a good example of multiple systems in a system description that mirrors the SBS.

PDF: System Description: Enterprise as a System of Systems (SoS), Version 0.17, 24-June-2023

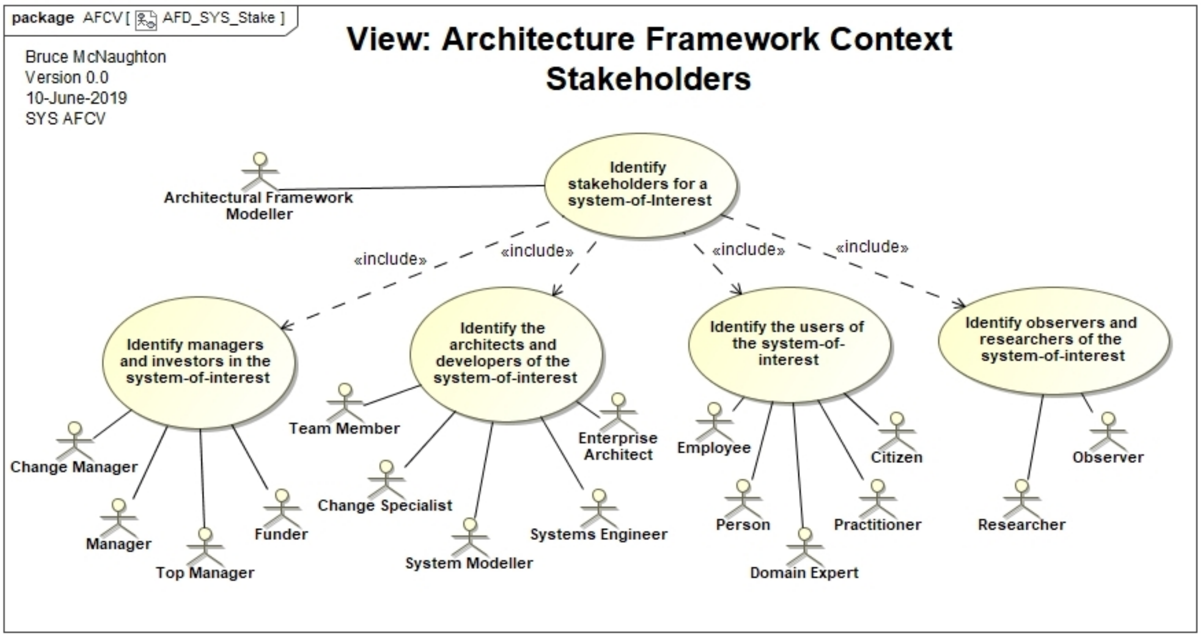
Correspondences

In addition, the correspondences section of any System Description can also provide relationships between AD Elements where an AD element can also be a system Description. This allows correspondences across systems to be documented. This also allows for the sharing of system description AD Elements such as viewpoints, model kinds, correspondences or other AD Elements.

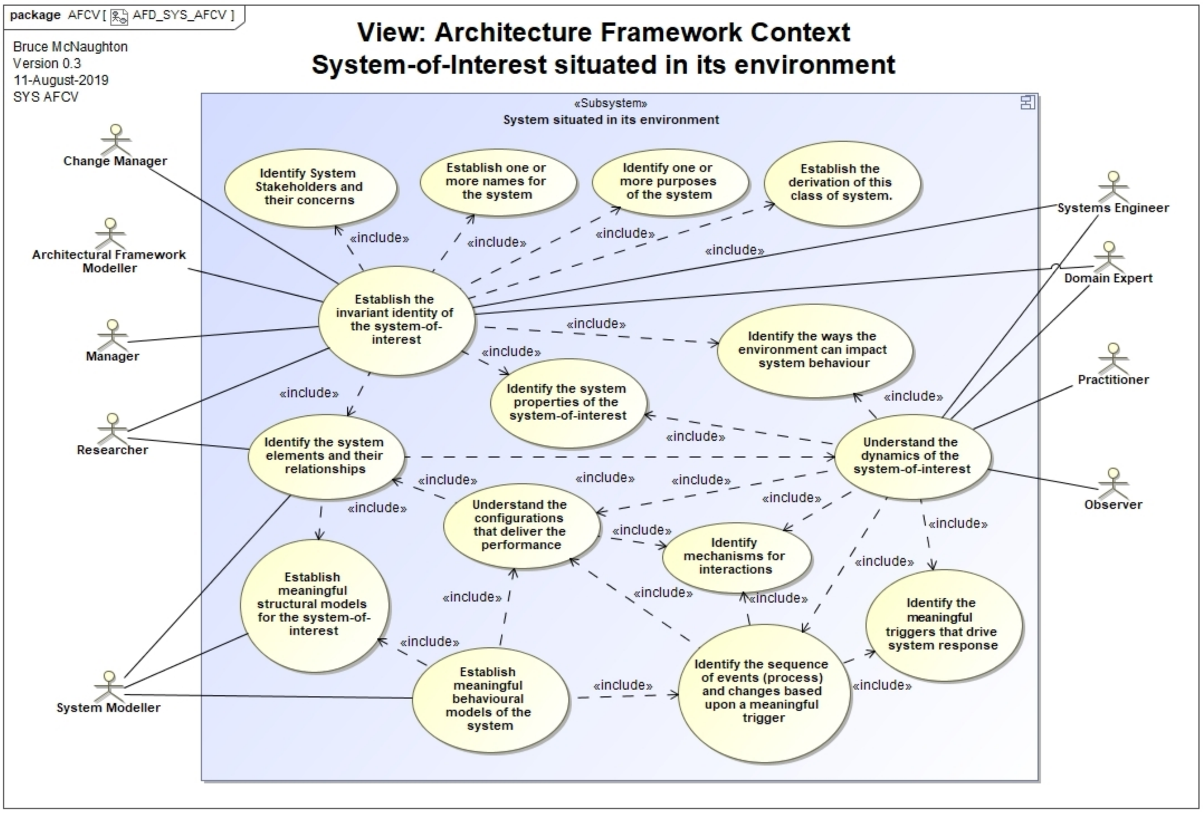
View: Architecture Description Framework Context

Answers the question: What is the purpose of the System Description Architecture Description Framework?

This context view uses a Use Case Diagram to highlight the concerns and needs of the stakeholders of the System-of-Interest situated in its environment The key stakeholders of the System-of-Interest situated in its environment represent the following domains:



The purpose of the System Description Architecture Description Framework is to provide instructions to create a System Description for a System-of-Interest situated in its environment.

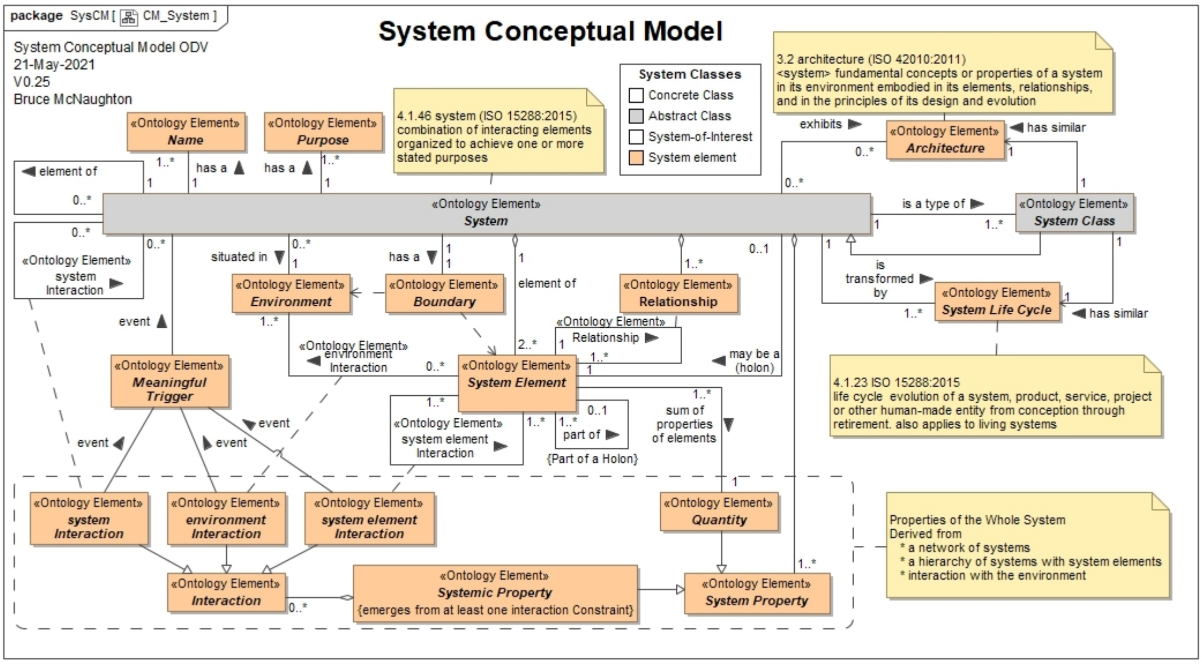


View: Ontology Definition

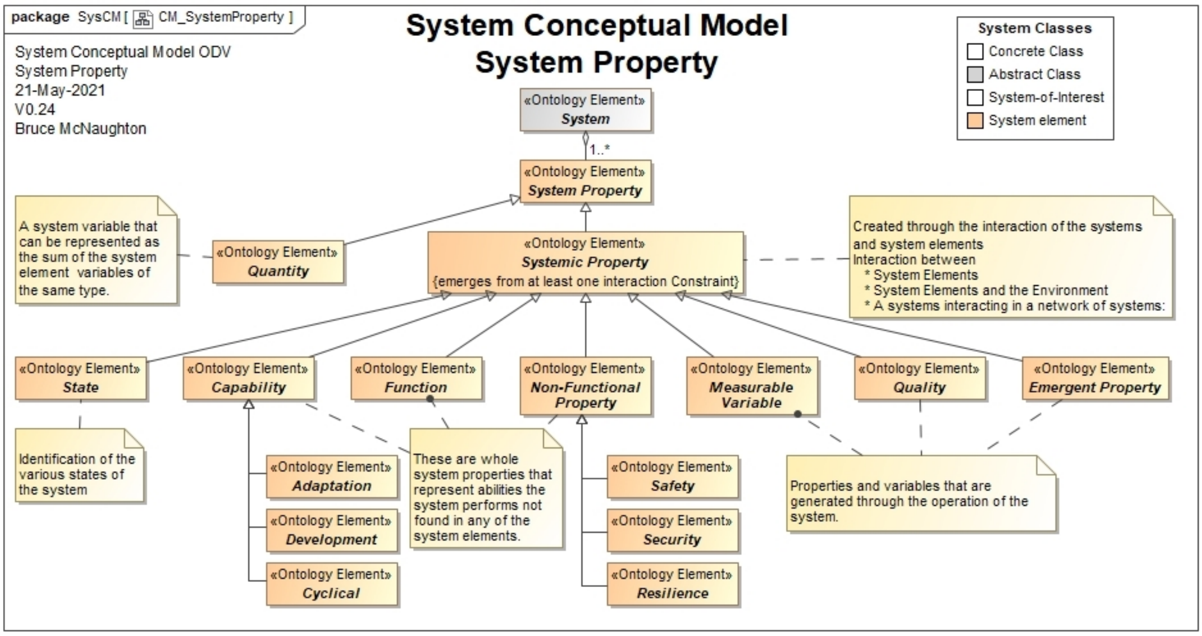
Answers the question: What domain concepts must the AF support?

This view identifies the concepts that are to be used within the System Description Architecture Description Frame-work These concepts are the underlying conceptual models for the System-of-Interest situated in its environment. This set of concepts is also called a 'concept model' or 'meta model'. This approach to the ontology focuses on whole systems. The following types of classes and models are defined in this section:

- Abstract (or Conceptual) Class
  - System (as an abstract class)
- Concrete (or Physical) Class
  - **System-of-Interest** as a Concrete Class

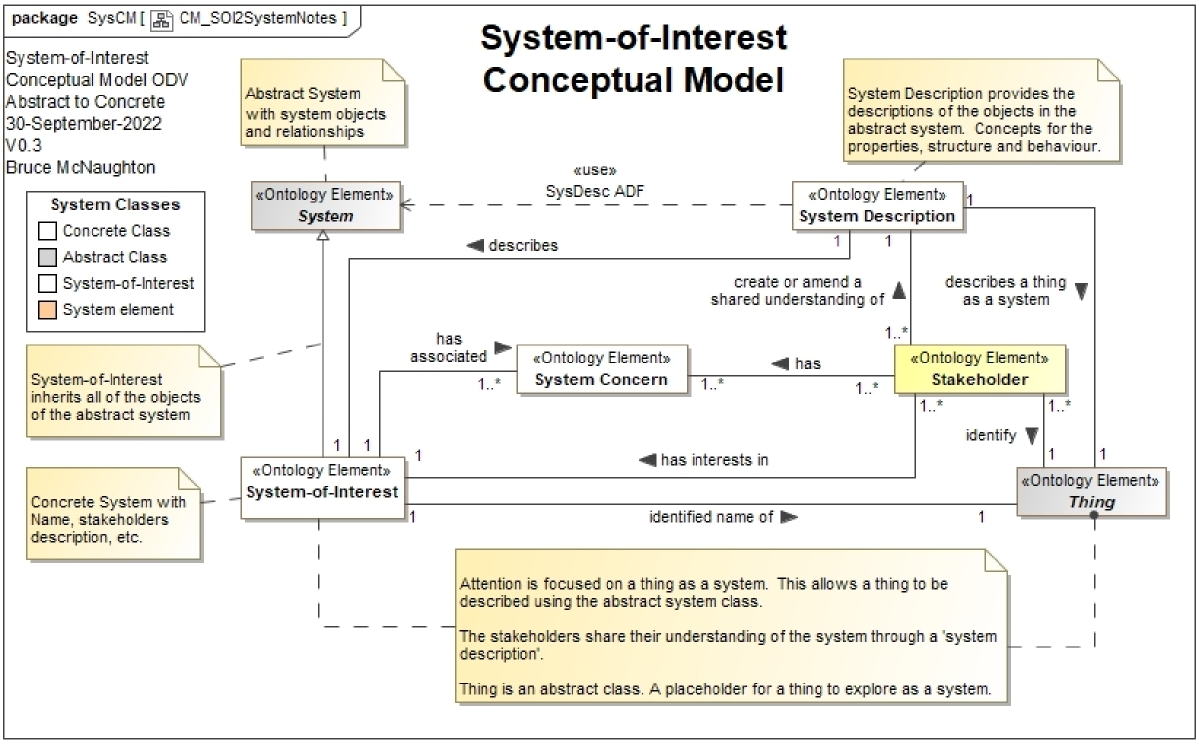


This diagram provides more details on the system property section of the system conceptual model.



System-of-Interest Conceptual Model

The following highlights the relationship of the abstract system and the system of interest.



The System Conceptual Model includes the **'System'** element as a UML Abstract Class where the class does not have any implementation or instances (See UML Definitions). The **'System-of-Interest'** element, on the other hand, is a UML Concrete Class that can have implementation and instances and can be seen through the interests and concerns of the stakeholders of the system. The stakeholders capture their shared understanding of the system-of-interest using a System Description.

The **System** as an abstract class provides the root definitions for all subsequent **System-of-Interest**. The terminology and elements of a **system** are inherited by any **system-of-interest.**

The UML Generalization association is used to show the inheritance of these definitions in the system-of-interest or any further derived system definitions.

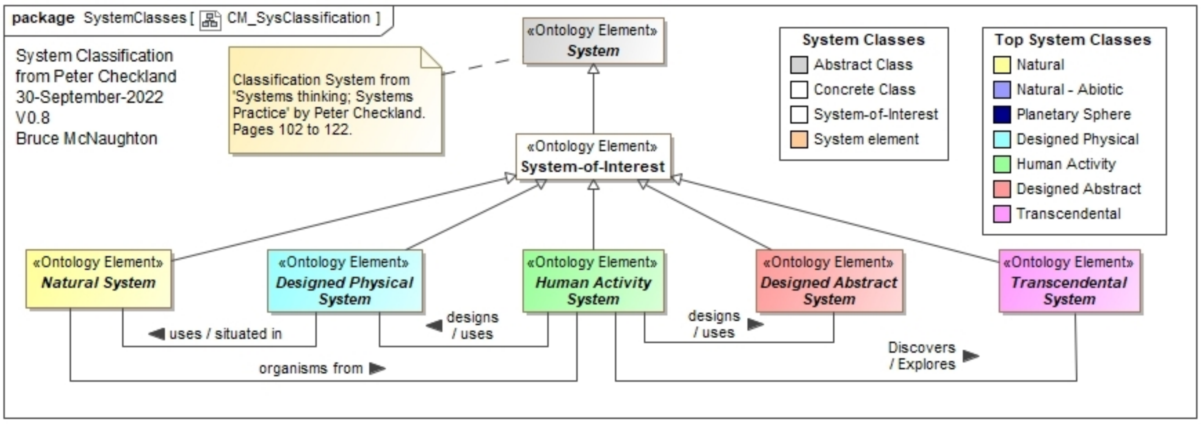
The system-of-Interest

"Thing" represents an "abstract class" representing a place holder for the specific thing that the stakeholders focus their attention and intention through the lens of a system. In this case, the system description provides a description of the "Thing as a System".

The **'System Class'** element provides a way to associate a **system** to a system classification scheme. As an example, Peter Checkland has identified a system classification system that uses the following system classes:

- Natural Systems
- Designed Physical Systems
- Designed Abstract Systems
- Human Activity Systems
- Transcendental Systems

See System Classes covered later in this document. The system classes provide a way to classify types of systems that can become a system-of-interest.



System-of-Interest situated in its environment

Concepts and Terminology

The concepts and terminology for the System situated in its environment describe an abstract system that can be used to describe any system-of-interest. These concepts and terminology are used in the design of the System Description Architecture Description Framework that can be used to create a System Description of a system-of-interest.

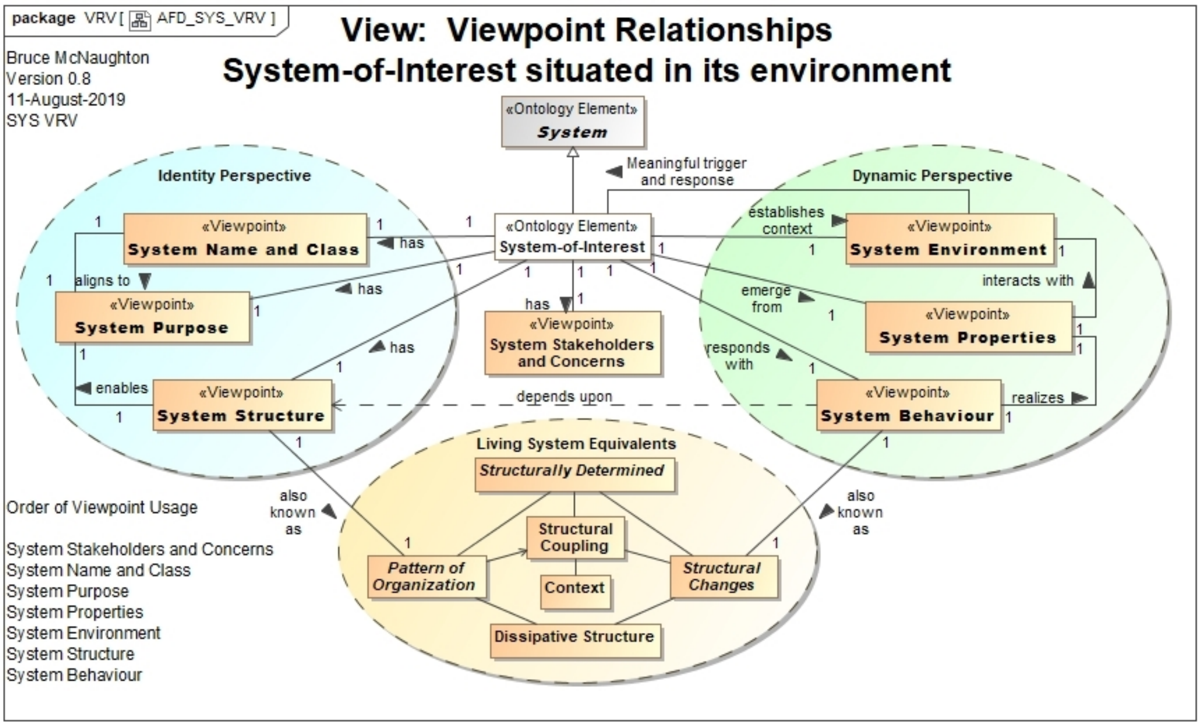
View: Viewpoint Relationships

Answers the question: What viewpoints are required?

The viewpoints have been grouped into two perspectives:

- Identity Perspective
- Dynamic Perspective

These two perspectives provide the set of viewpoints for the System Description Architecture Description Framework



The viewpoints that have been identified for the System Description Architecture Description Framework are:

- **Identity Perspective.**
  - **System Name and Class** Establishes the name of the system-of-interest and the base class.
  - **System Purpose**: Establishes the purpose of the system-of-interest. There may be many reasons for being.
  - **System Structure (Pattern of Organization)**: Identifies the system elements and their relationships
- **Dynamic Perspective**
  - **System Environment** Identifies the elements of the system environment that have an impact on the system-of-interest.
  - **System Properties** Identifies the properties of the whole that are not found in any of the system elements. These properties are only seen when the system-of-interest exists and is operating or alive.
  - **System Behaviour (Structural Changes)** Identifies the way the system-of-interest responds (process) to meaningful triggers or disturbances.

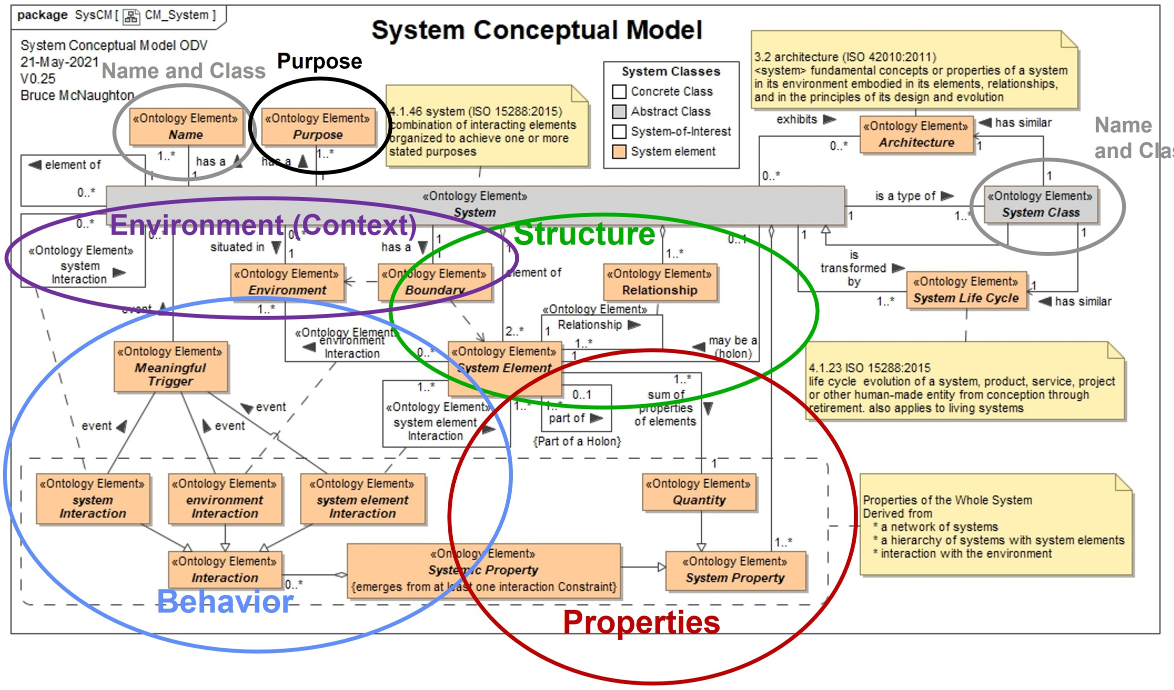
The living system equivalents section highlights terminology differences across disciplines. The following terms have equivalences in this model:

- **Living System**: Generally characterized as open to the flow of energy, matter and information and closed operationally.
- **Pattern of Organization**: The pattern of organization is a description of the components, or parts and their relationships. This is usually referred to the structure of a system. The typical pattern is a network pattern.
- **Structurally Determined**: As a living system is closed operationally, a living system responds to a meaningful trigger or disturbance (event) through a sequence or process of changes that are structurally determined. This response is constrained by the structure or pattern of organization.
- **Structural Coupling**: Autonomous living systems are structurally coupled. This means that they can raise a meaningful trigger or disturbance to another living system but cannot define the change that will occur. The living system will respond to the disturbance with appropriate structural changes.
- **Structural Changes**: In response to a meaningful trigger or disturbance, the system carries out a sequence or process which may alter the structure based upon the type of change. This may result in a response that

may change the state of a system, create a condition where the same occurrence may cause a different response (learning effect) or may create novelty or creativity and improve or alter the identity of the system.

- **Context**: Meaningful disturbances that arise from structural coupling come from the environment.
- **Dissipative Structure**: A dissipative structure can form in living and non living systems to form a repeatable process / structure that transforms material, energy and information. This structure forms dynamically in the living system in response to a meaningful trigger or disturbance. This dissipative structure creates order and moves the system to a place far from equilibrium. The dissipative structure forms a type of process through the various system elements.

The following System Conceptual Model has the mapping of the Ontology Elements to viewpoints. This is a recommended way to identify viewpoints from the Ontology Elements.

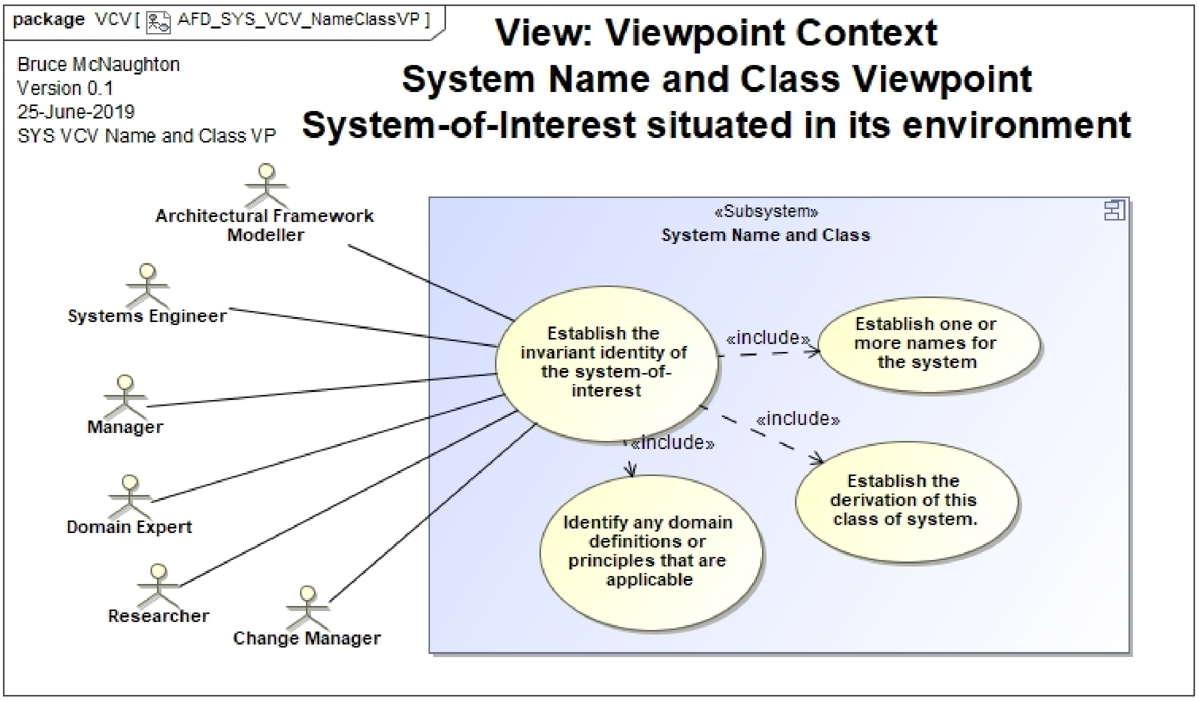


Views: System Name and Class Viewpoint

View: Viewpoint Context

Answers the question: What is the purpose of this Viewpoint?

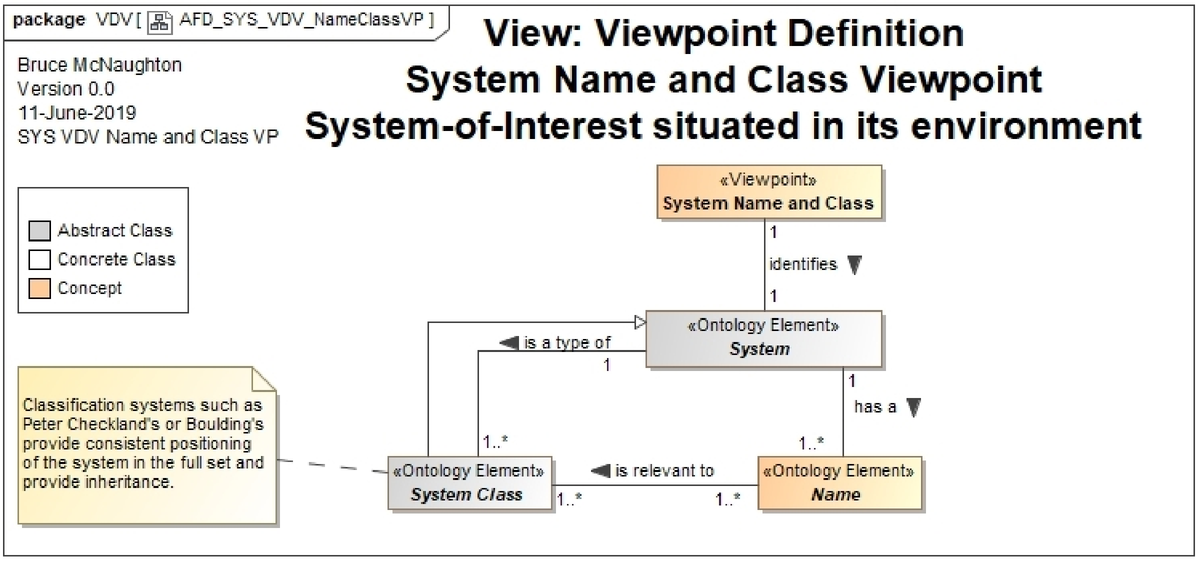
This viewpoint identifies the Name and Class of the system. These attributes provide a way to inherit definitions from other system classes and be initiate the formation of the identity of the system. In addition, other domain definitions and principles can be included in this section.



View: Viewpoint Definition

Answers the question: What is the definition of this Viewpoint in terms of the identified domain concepts?

This viewpoint provides concise information about the identity of the system.



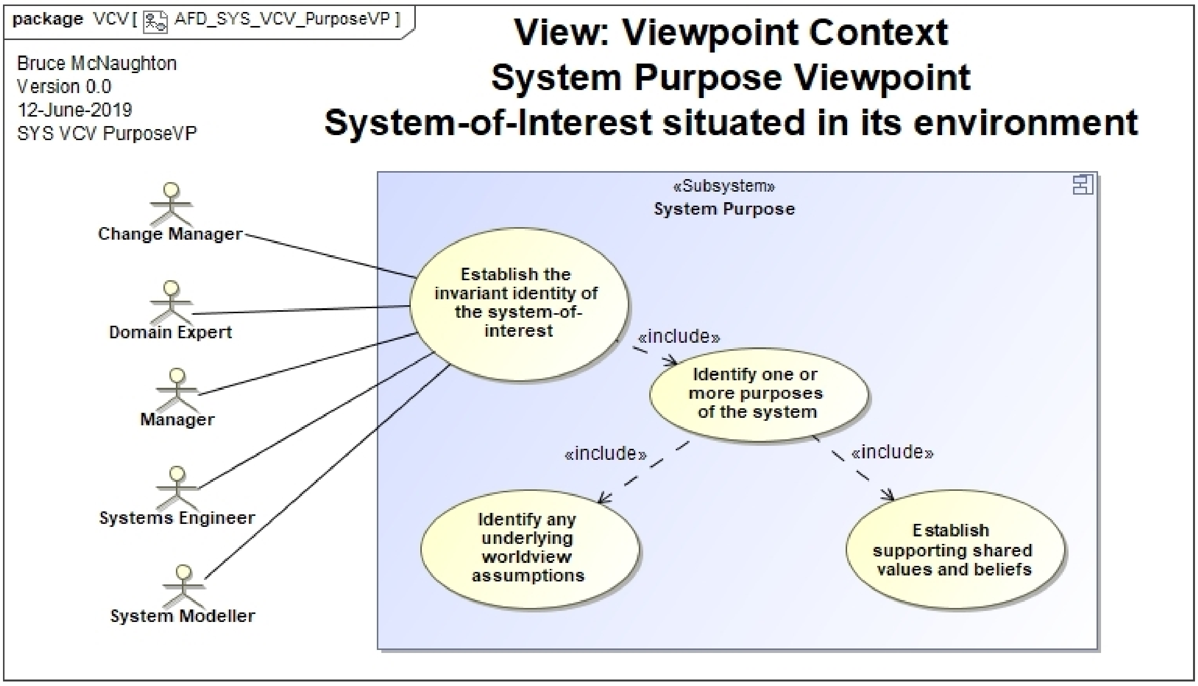
See System Classes for further information

Views: System Purpose Viewpoint

View: Viewpoint Context

Answers the question: What is the purpose of this Viewpoint?

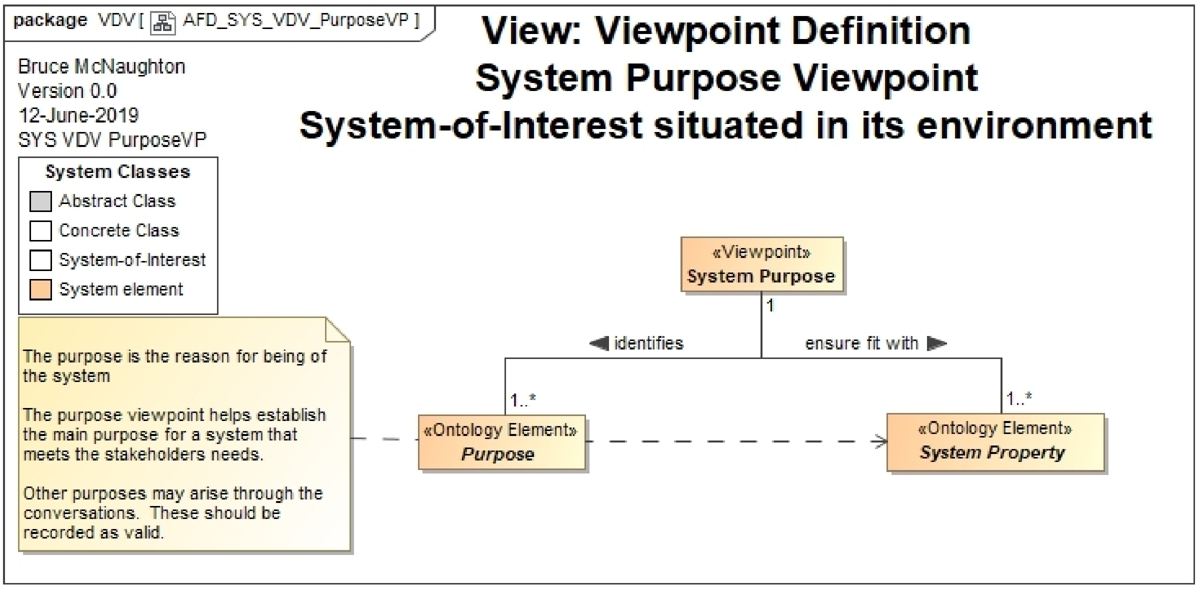
This viewpoint provides the 'reason for being' for this system. There may be many purposes associated with the system. These are discovered using this viewpoint.



View: Viewpoint Definition

Answers the question: What is the definition of this Viewpoint in terms of the identified domain concepts?

This viewpoint provides a clear set of statements about the 'reason for being' of this system. This may include other attributes that support the definition of this system.

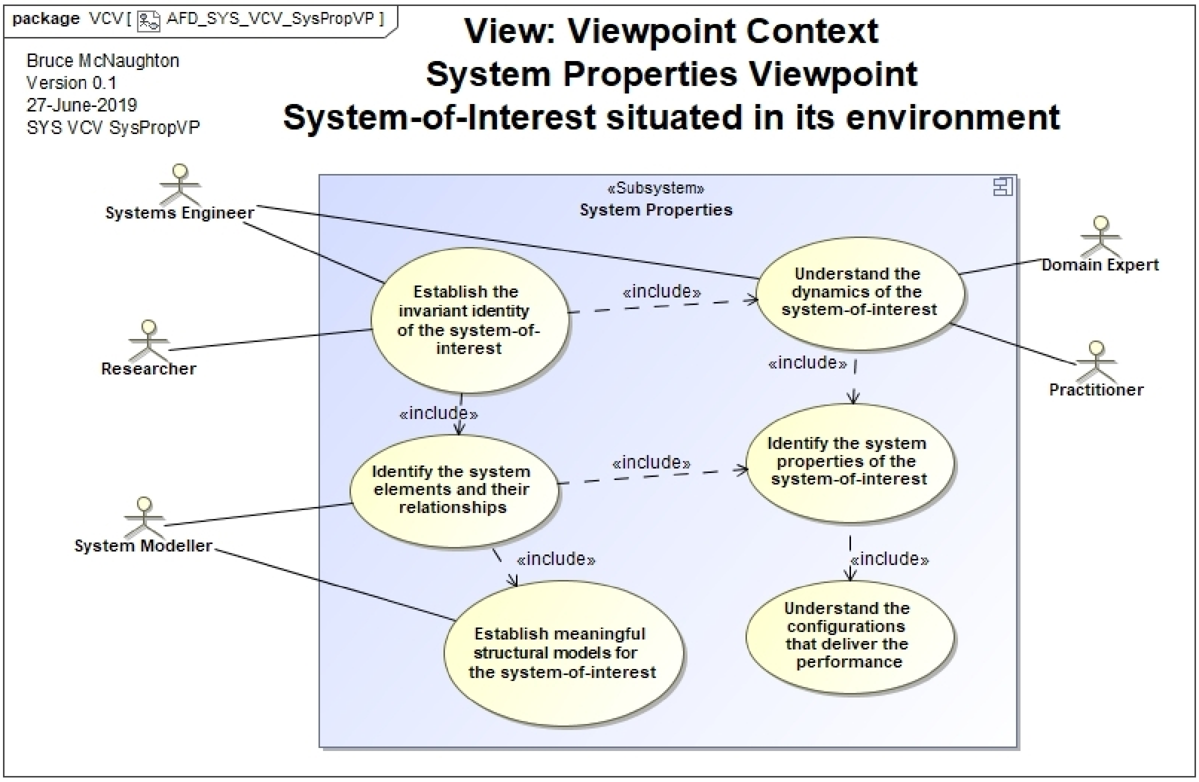


Views: System Properties Viewpoint

View: Viewpoint Context

Answers the question: What is the purpose of this Viewpoint?

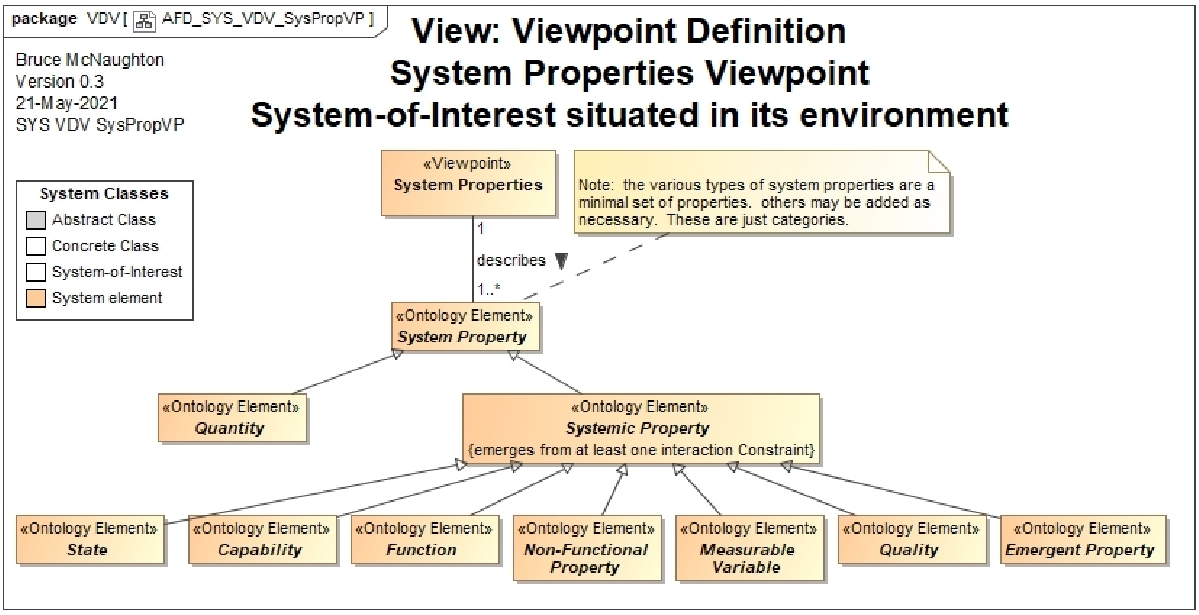
This viewpoint Identifies the types of system properties that relate to the system as a whole. These system properties are properties of the whole and provide functions or abilities to the environment. or context



View: Viewpoint Definition

Answers the question: What is the definition of this Viewpoint in terms of the identified domain concepts?

This viewpoint identifies and defines the various system properties of the system. This definition establishes the key names and types of system properties.

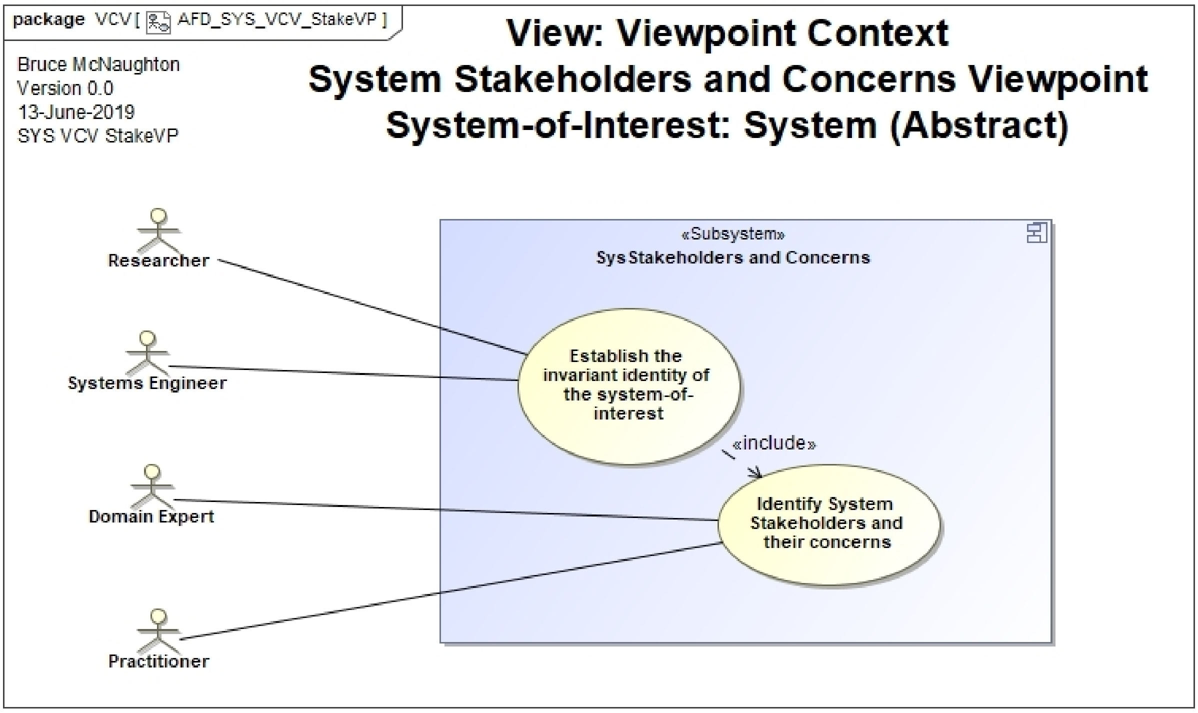


Views: System Stakeholders and Concerns Viewpoint

View: Viewpoint Context

Answers the question: What is the purpose of this Viewpoint?

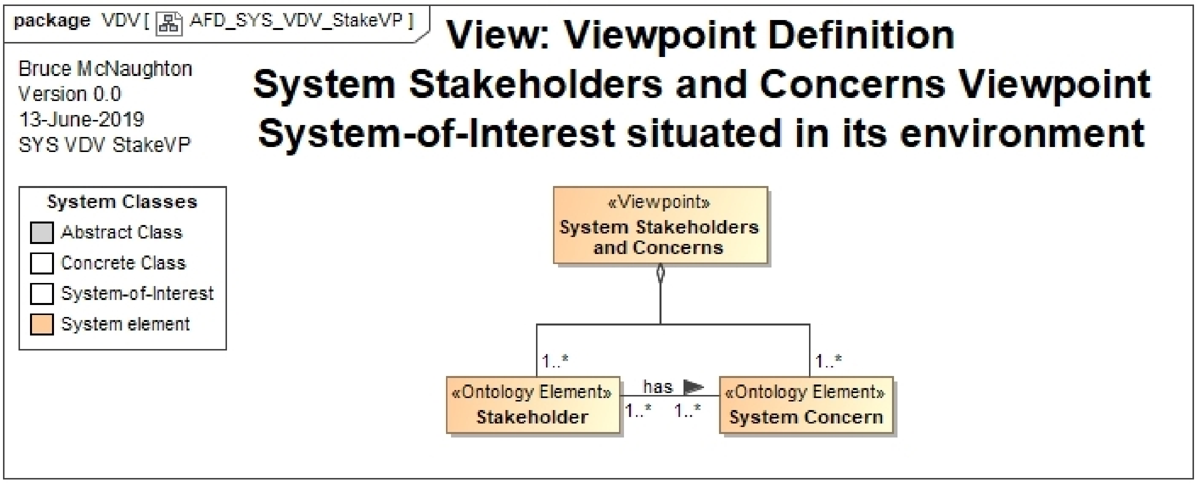
This viewpoint Identifies the types of stakeholders that have an interest in the system-of-interest. These Stakeholders have relevant roles related to the system-of-interest.



View: Viewpoint Definition

Answers the question: What is the definition of this Viewpoint in terms of the identified domain concepts?

This viewpoint identifies and defines the various concerns or interests raised by the stakeholders. This view allows the stakeholders and their concerns to be seen as part of the description of a system-of-interest. These stakeholders and concerns will be used in any Architecture Description Framework that is being used.

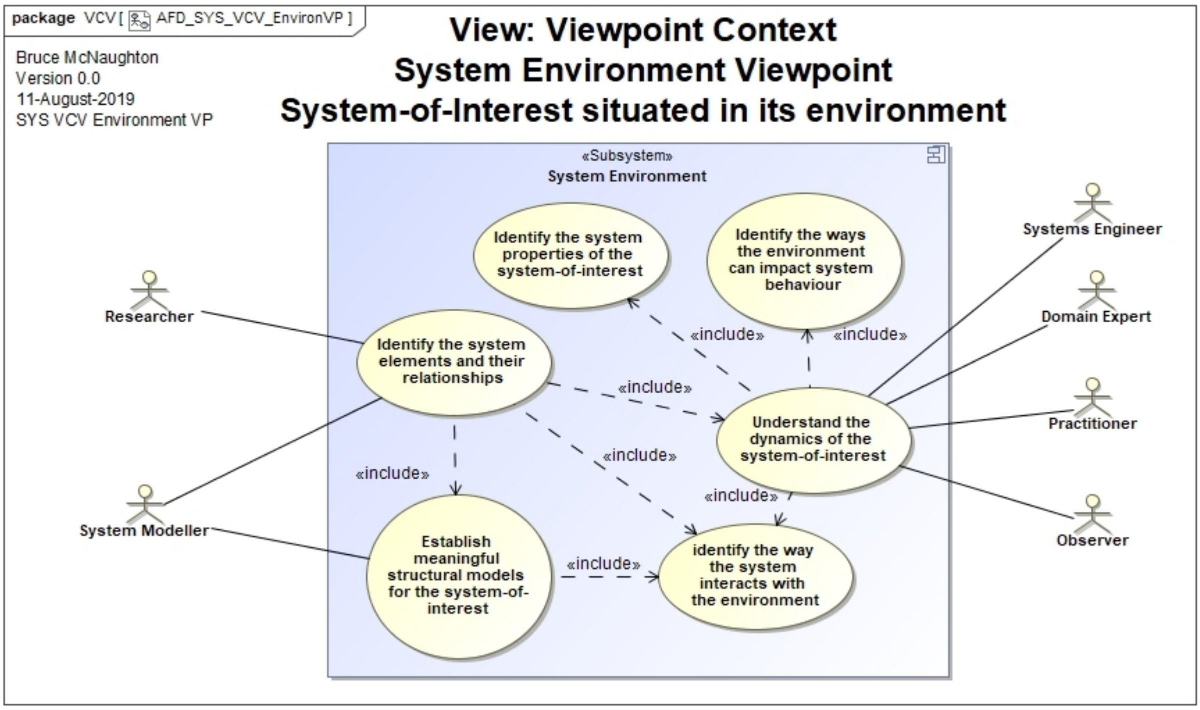


Views: System Environment Viewpoint

View: Viewpoint Context

Answers the question: What is the purpose of this Viewpoint?

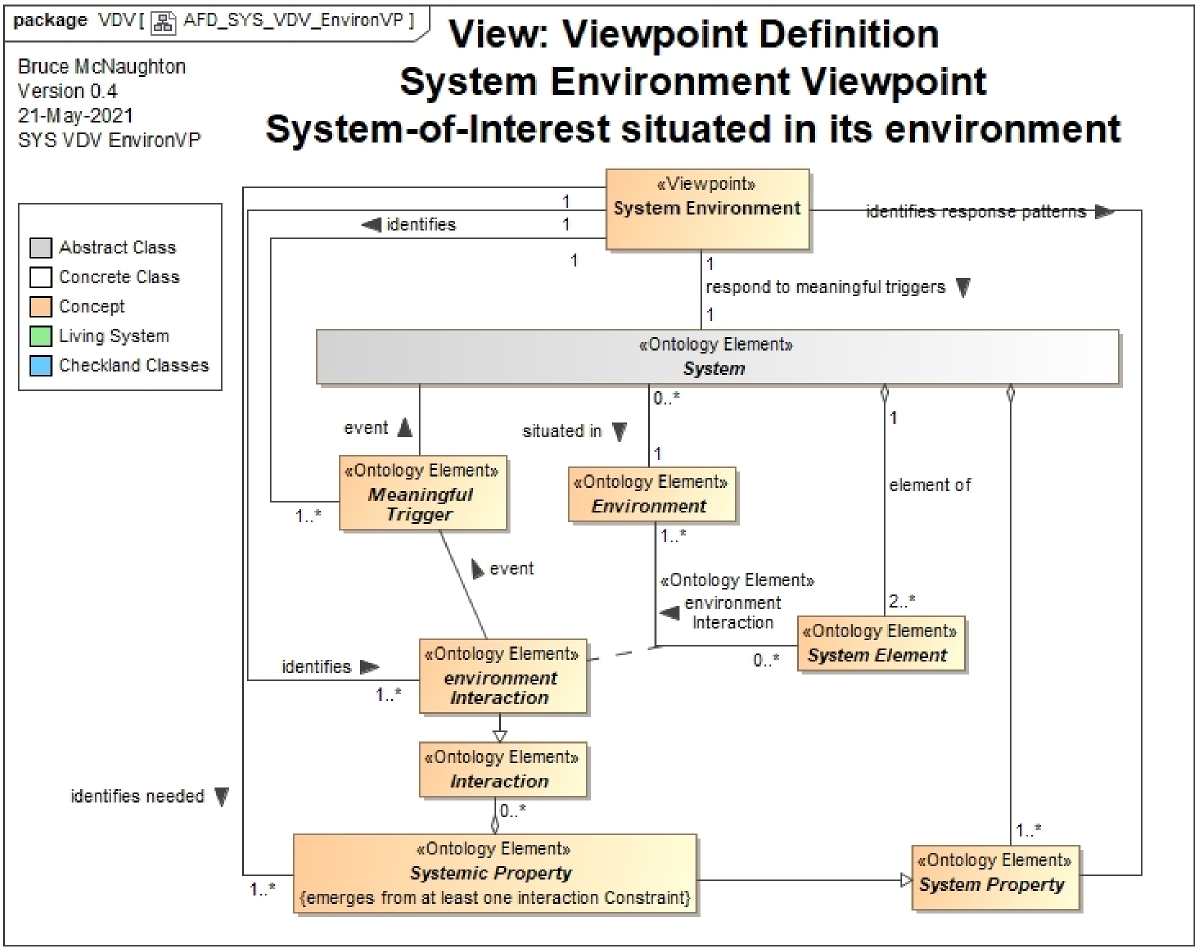
This viewpoint identifies the aspects of the environment that shape the responses and provide the context of the system.



View: Viewpoint Definition

Answers the question: What is the definition of this Viewpoint in terms of the identified domain concepts?

This viewpoint defines the context of the system and the key interactions with the environment.



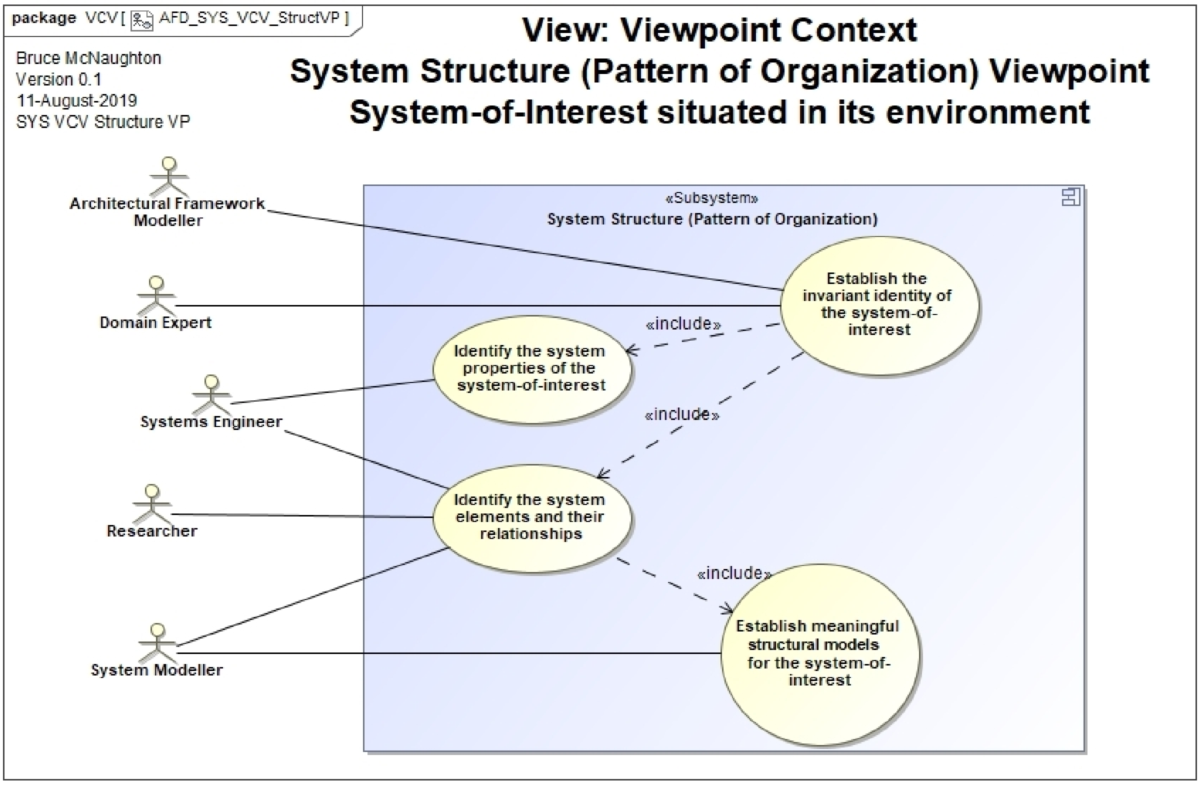
package VDV [AFD_SYS_VDV_EnvironVP]
View: Viewpoint Definition
System Environment Viewpoint
System-of-Interest situated in its environment
Bruce McNaughton
Version 0.4
21-May-2021
SYS VDV EnvironVP
Abstract Class
Concrete Class
Concept
Living System
Checkland Classes
«Viewpoint»
System Environment
identifies
identifies response patterns
respond to meaningful triggers
«Ontology Element»
System
event
situated in
element of
«Ontology Element»
Meaningful Trigger
«Ontology Element»
Environment
event
«Ontology Element»
environment Interaction
«Ontology Element»
System Element
identifies
«Ontology Element»
environment Interaction
«Ontology Element»
Interaction
identifies needed
«Ontology Element»
Systemic Property
{emerges from at least one interaction Constraint}
«Ontology Element»
System Property

Views: System Structure (Pattern of Organization) Viewpoint

View: Viewpoint Context

Answers the question: What is the purpose of this Viewpoint?

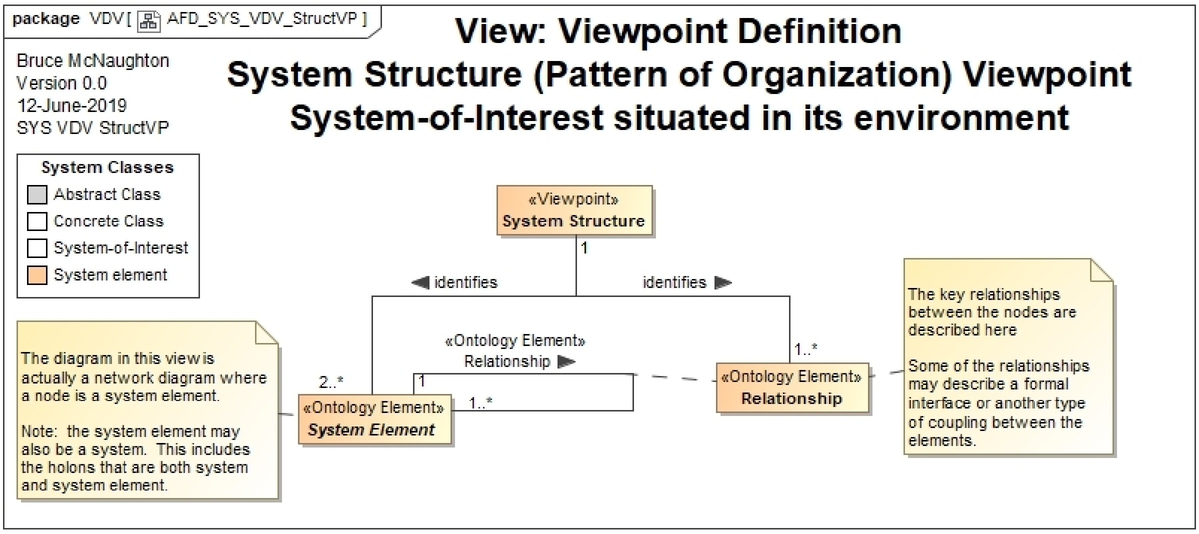
This viewpoint identifies the system elements and their relationships. The network is a key representation used in this viewpoint.



View: Viewpoint Definition

Answers the question: What is the definition of this Viewpoint in terms of the identified domain concepts?

This viewpoint provides the instructions for the creation of models that highlight the identified system elements and their relationships.

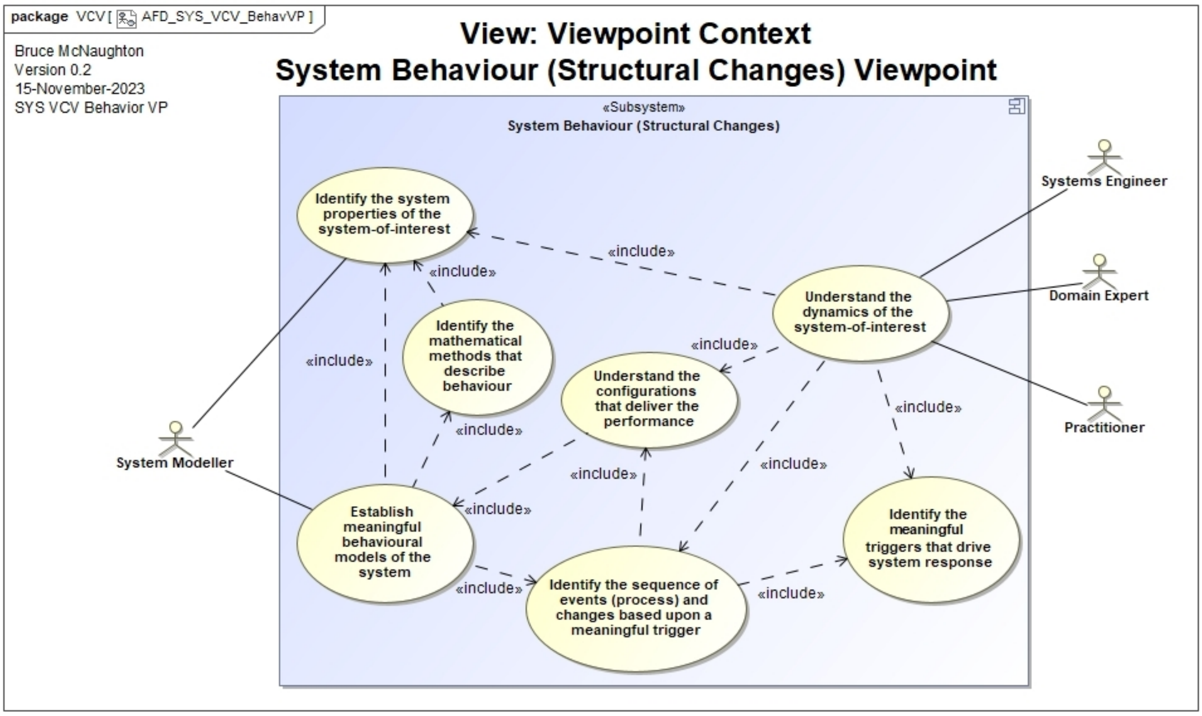


Views: System Behaviour (Structural Changes) Viewpoint

View: Viewpoint Context

Answers the question: What is the purpose of this Viewpoint?

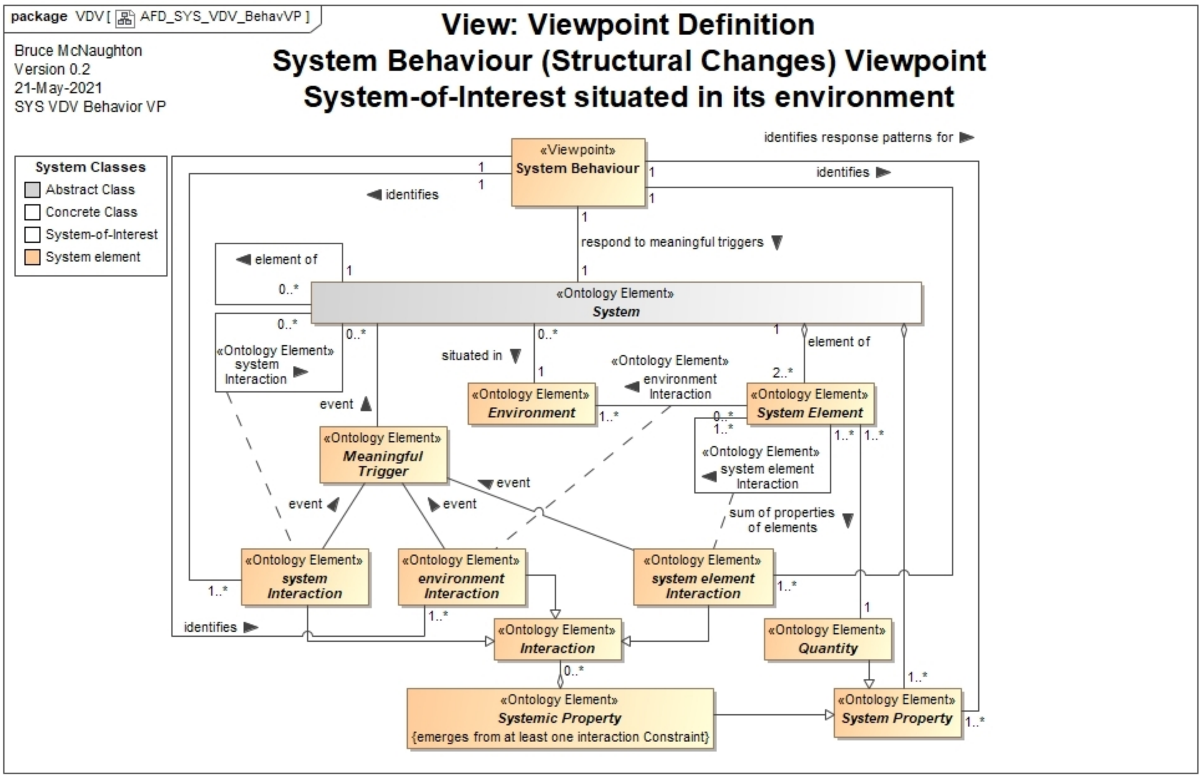
This viewpoint provides the way to view the operational dynamics of the system. The resulting view identifies the way specific triggers or disturbances are handled by the system over time. This assumes an understanding of the specific system elements and their characteristics for a speicific configuration of the system. This view also provides a way to view the behaviour of the systemic properties of the system.



View: Viewpoint Definition

Answers the question: What is the definition of this Viewpoint in terms of the identified domain concepts?

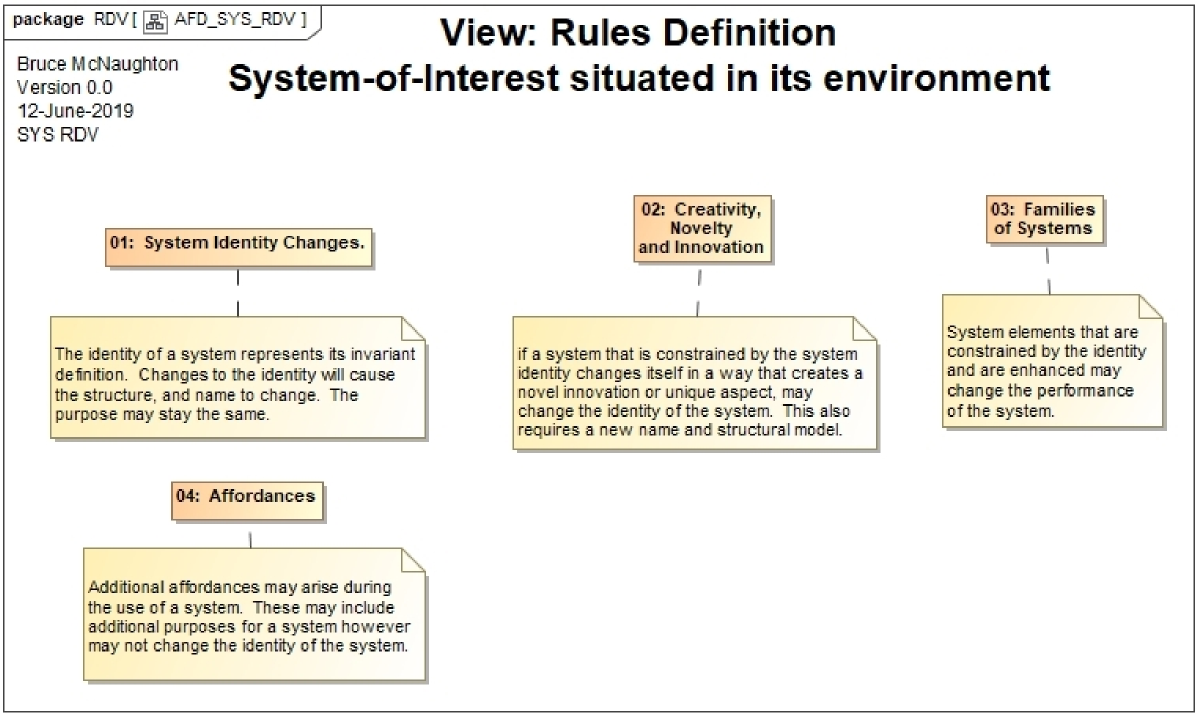
This viewpoint identifies the types of responses the system takes given meaningful triggers or disturbances. (In some cases, these are through well defined interfaces and others are a response to probabilistic (stochastic) events).


package VDV [AFD_SYS_VDV_BehavVP]
View: Viewpoint Definition
System Behaviour (Structural Changes) Viewpoint
System-of-Interest situated in its environment
Bruce McNaughton
Version 0.2
21-May-2021
SYS VDV Behavior VP
System Classes
Abstract Class
Concrete Class
System-of-Interest
System element
«Viewpoint» System Behaviour
identifies response patterns for
identifies
identifies
respond to meaningful triggers
element of
«Ontology Element» System
«Ontology Element» system Interaction
situated in
«Ontology Element» environment Interaction
element of
«Ontology Element» Environment
«Ontology Element» System Element
event
«Ontology Element» Meaningful Trigger
«Ontology Element» system element Interaction
event
event
event
sum of properties of elements
«Ontology Element» system Interaction
«Ontology Element» environment Interaction
«Ontology Element» system element Interaction
identifies
«Ontology Element» Interaction
«Ontology Element» Quantity
«Ontology Element» Systemic Property {emerges from at least one interaction Constraint}
«Ontology Element» System Property

View: Rules Definition

Answers the question: What rules constrain the use of the Architecture Description Framework?

This view provides the set of rules used to create the Architecture Description Framework.



Architecture Decisions and Rationale

The following decisions and rationale are recorded for this Architecture Viewpoint Definition (AVPD).

1) Why was the System (Abstract) Architecture Viewpoint Definition created?

The concept of a System Description has been used to create well formed descriptions of a system-of-interest that can be validated by various definitions of systems. The rationale for writing a system description is not visible in any method. The System Description is also an output from process 6.4.3 in ISO 15288:2015.

However, over time, there have been various investigations into system science publications and various authors who used the term view to describe an aspect of a system. This triggered the thinking that we write architecture descriptions as a normal way to describe the architecture of a system. In this way, a system description could be created from viewpoints to describe any system. The resulting document would be a system Architecture Description (AD).

In this final pass, the term architecture has been dropped leaving only System Description.

2) Revision to align with ISO 42010:2022

The following changes have been made to support the revision to ISO 42010:2022.

Output Name Change.

The name of the **Architecture Framework Definition (AFD)** has changed to **Architecture Viewpoint Definition (AVPD)**. The primary purpose of the document is to establish the terminology / ontology for a system-of-interest. The key output of this work is the definition of the viewpoints for the Architecture Description Framework.

Alignment with new terminology

The Architecture Framework name has changed to Architecture Description Framework.

Incorporate Structuring Formalism

Integrate the structuring formalism as part of the Architecture Viewpoint Definition (AVPD).

NOTE: CAFF terminology retained until the CAFF can be revised.

System: [System Name]

View: System Name and Class

Name:

Based on:

Also include any domain definitions or principles that relate to this type of system.

View: System Purpose

The stated or implied purposes of the system-of-interest

View: System Properties

This section identifies the system properties. Suggested headings have been included. These can be tailored..

Systemic Measurable Variables

Systemic Capabilities or Functions

System States

Systemic Quality Properties

System Quantity Properties

View: System Stakeholders and Concerns

The Stakeholders and their concerns or interests in this system-of-interest

View: System Environment (Context)

The environment and the potential impacts on the system-of-interest.

this section includes

- Transactional
- Contextual
- Regulatory

View: System Structure (Pattern of Organization)View: Pattern of Organization

Identification and Composition of System elements and their relationships. This section generally includes a system conceptual model.

System Element: Identification

Identification model separate

System Element: Relationships

Relationship model separate

System Element: Identification and Relationships

Single model combined

Concept: Identification and Relationships

Conceptual model

View: System Behavior (Structural Changes)View: Structural Changes

Describes a specific instance (configuration of components) of a system structure and this systems behaviour.

Describe the response of the system due to various triggers

Configuration / Scenario:

Describes any configuration / scenario attributes for a specific system-of-interest. This may not be appropriate for all system descriptions (e.g. patterns or abstract systems).

Cyclical (Repeating / Regular) Processes

Routine operational processes to maintain the system
Provides support to deliver the capabilities or functions of the system.

Development Life Cycle Processes

How this system is created, developed, used and released.
Describes the processes for major developmental changes.

Validate: System Name

Overview

This table validates the system definition for the system element name.

Validate using Russell Ackoff's Definition and 5 Conditions

Definition: System: A system is a whole consisting of two or more parts that satisfies the following five conditions:

1. The whole has one or more defining properties or functions

xxx

2. Each part of the set can affect the behavior or properties of the whole

xxx

3. There is a subset of parts that is sufficient in one or more environments for carrying out the defining function of the whole; each of these parts is necessary but insufficient for carrying out this defining function.

xxx

4. The way that each essential part of a system affects its behavior or properties depends on (the behavior or properties of)at least one other essential part of the system.

xxx

5. The effect of any subset of essential parts on the system as a whole depends on the behavior of at least one other such subset.

xxx

Validate using 10 things to understand about systems

The 10 things are used to validate the system-of-interest

1. A system exists within a wider 'context' or environment

xxx

2. A system is made up of parts that interact with each other and with the wider environment

xxx

3. A system has structure, function, performance, behaviour and a lifecycle.

xxx

4. A system has system-level properties ("Emergent properties") that are properties of the whole system not attributable to individual parts.

xxx

5. A system both changes it environment and adapts to its environment when it is deployed.

xxx

6. Systems contain multiple feedback loops with variable time constraints.

xxx

7. A system may be part of one or several wider 'containing systems'.

xxx

8. A system may have one of three basic types of relationship with its environment: distinct, close-coupled, fluid and dynamic.

xxx

9. A system may offer 'affordances' for interaction.

xxx

10. Types of system include technical, social, ecological, environmental and any combination of these.

xxx

References

Architecting Systems, Hillary Sillitto

Architecting Systems

The Compass Project, The Compass Club

The Comprehensive Modelling for Advanced Systems of Systems.or the COMPASS Project provides the terminology and concepts used in this document for a system of systems. This project is now closed and information about the COMPASS Project has been archived.

The COMPASS Architectural Framework Framework (CAFF) is closely aligned to the Framework for Architecture Frameworks (FAF) written by Simon Perry and Jon Holt. This CAFF is still used as the basis for the Architecture Viewpoint Definition (AVPD) work product.

Here is the final version of the COMPASS Architectural Framework Framework (CAFF) available for this work product:

Link to D21.5b Compass Architectural Framework Framework (Local): CAFF Viewpoint Definitions

Reference Materials (posted with Permission)

NOTE: The authors of the FAF below contributed to the Compass Guides

Some original information related to the earlier FAF from INCOSE UK

- Presentation on FAF from Simon Perry, INCOSEUK
- Paper on FAF from Simon Perry, INCOSEUK.

Fifth Discipline, Peter M. Senge

The Fifth Discipline: The art and practice of the learning organization: Second edition

The Five Disciplines described in the book are important to seeing systems and understanding the interaction of the parts.

The Five Disciplines are similar to the System of Profound Knowledge described by Deming.

Key elements of this book:

- An understanding of mental models and the impact they can have on decisions
- An understanding of the importance of personal visions both for individual motivation and later for building a shared vision.
- An understanding of the dynamics of systems thinking both in time and place.
- An understanding of the importance of practice in a safe environment.

Principles of Systems Science, George E. Mobus, Michael C. Kalton

Principles of Systems Science

Excellent visuals, principles and concepts about systems and system science.

System Engineering Handbook, INCOSE

Link to information about the: System Engineering Handbook

INCOSE

Systems Thinking, Systems Practice, Peter Checkland

Systems Thinking, Systems Practice: Includes a 30 Year Retrospective

This book contains a good description of Human Activity Systems (HAS) based on a root definition to describe a human activity system (CATWOE). These are both used in the Soft Systems Methodology (SSM).

The concept of the Root Definition has been extended to the System Description that is produced using the System Description Architecture Description Framework. The Human Activity System has also been extended from living social systems.

The book also contains a simple system classification scheme that is being used to define a Earth (Gaia) as a System of Systems model. The system classification system is described in the book from page 102 to page 122. Figure 4, page 112 highlights the 5 system classes. This book also has a good glossary of terms.

This system classification scheme is also being used as the System Classification Framework for the System Description Architecture Description Framework. This framework captures the identified systems and their type.

The Systems View of Life, Fritjof Capra and Pier Luigi Luisi

The Systems View of Life

This book is supported by the Capra Course which provides a 12 week course covering the four dimensions of life: Biological, Cognitive, Social, and Ecological.

A Capra Course Glossary is available in the Capra Course Alumni Network - A global Community of Practice related to the book.

See chapter 14 for information on social systems.

UML 2 and the Unified Process, Jim Arlow and Ila Neustadt

UML 2 and the Unified Process: Practical Object-Oriented Analysis and Design (2nd Edition)

This book provides a view of UML throughout the system life cycle.

ISO 15288:2023 Systems and software engineering — System Life Cycle Processes

ISO 15288 System Life Cycle Processes

ISO 15288:2023 Systems and software engineering — System Life Cycle Processes.

Integrated Management System

ISO 42010:2022 Software, Systems and Enterprise - Architecture Description

ISO 42010:2022 (Software, Systems and Enterprise - Architecture Description). ISO 42010:2022 replaces ISO 42010:2011 and IEEE 1472.

Main website for ISO 42010 is:

http://www.iso-architecture.org/42010/

Alternative website:

http://www.iso-architecture.org/

Conceptual Model contained in ISO 42010

http://www.iso-architecture.org/42010/cm/

Wikipedia: ISO 42010

The architectural concepts from the Compass Project and this international standard have been integrated into the change and transformation approach.

Rich Hilliard
