UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

RAFAEL HUMANN PETRY

Improving Interoperability on Industrial Standards through Ontologies

Monograph presented as a partial requirement for obtaining the degree of Bachelor of Science in Computer Science.

Supervisor: Prof. Dr. Mara Abel
Co-supervisor: M.Sc. Eng. Nicolau Oyhenard dos Santos

Porto Alegre
2023

ACKNOWLEDGMENTS

I want to express my heartfelt gratitude to my family for their unwavering guidance, love, respect, and care and for providing me with a solid foundation. Specifically, I want to extend special thanks to my father, who has been my role model and mentor; my mother, who has been my unwavering source of support; my sister, my lifelong companion; and my girlfriend, for her incredible partnership and morale.

My supervisors and colleagues have played a fundamental role in introducing me to research, offering invaluable insights, and serving as a supportive network that continuously fuelled my understanding of the world. I'm incredibly grateful to my supervisor for her valuable guidance and my co-supervisor for his patience and meticulous planning.

I sincerely thank FINEP and the LIBRA consortium for their generous support, which has enabled the realization of this research endeavor.

I am indebted to the dedicated professors at UFRGS's Informatics Institute for their tireless efforts in imparting knowledge and facilitating my learning. And I also express gratitude to the Brazilian people, my people, for allowing me to receive my education.

Lastly, I would like to acknowledge pnb plants & bytes for providing access to their pnb Toolbox Library, which was pivotal in developing this research.

ABSTRACT

Interoperability refers to the effective exchange of information and understanding to collectively pursue common objectives. System developers commonly use ontologies to enhance semantic and syntactic interoperability within this context. This work aims to evaluate the contribution of ontology in making explicit the meaning of the entities described in a Piping and Instrumentation Diagram (P&ID) model and to provide an architecture that allows the representation of a P&ID in ontological knowledge bases.

To understand the semantics of the P&ID entities and relations, we map each class of the P&ID to the corresponding entity of the Offshore Petroleum Production Plant Ontology (O3PO). The ontology describes the definition of each vocable associated with the axioms that clarify and regulate the meaning and utilization of this vocabulary. We intend to guarantee that the integration of P&ID with other models respects the original semantics and avoids unintended data exchanges. We follow this ontological analysis with a case study of a model that conforms to the Data Exchange in the Process Industry (DEXPI) specification, intended to provide homogeneous data interchange between CAD systems from diverse vendors. The ontological analysis of the DEXPI P&ID specification, to build a relation with a well-founded ontology, raises a set of desirable properties for a model intended for use in interoperability.

While achieving technical interoperability between DEXPI P&IDs and ontologies represented in OWL is evident, we identified several challenges within the realm of semantic interoperability, specifically concerning clarity/intelligibility, conciseness, extendibility, consistency, and essence. These issues present significant hurdles to achieving seamless systems integration. Moreover, if the DEXPI standard were to evolve into a de facto standard for representing P&IDs across a broader range of domains than initially intended, these highlighted issues could potentially bottleneck its adoption and hinder its integration into different systems.

Keywords: Ontology. Interoperability. DEXPI. Piping and Instrumentation Diagram (P&ID). Industrial standard. Oil & Gas.

RESUMO

Interoperabilidade se refere à troca efetiva de informação e entendimento na busca por objetivos comuns. Neste contexto, desenvolvedores de sistemas comumente utilizam ontologias para aprimorar a interoperabilidade semântica e sintática. O objetivo deste trabalho é avaliar a contribuição da ontologia para tornar explícito o significado das entidades descritas em um modelo de Diagrama de Tubulação e Instrumentação (DTI) e fornecer uma arquitetura que permita a representação de um DTI em bases de conhecimento ontológicas.

Para entender a semântica das entidades e relações do DTI, mapeamos cada classe do DTI para a entidade correspondente da Ontologia de Planta de Produção de Petróleo Offshore (O3PO). A ontologia descreve a definição de cada vocábulo associado com os axiomas que esclarecem e regulam o significado e a utilização desse vocabulário. Pretendemos garantir que a integração do DTI com outros modelos respeite a semântica original e, assim, evite trocas de dados não intencionais. Seguimos essa análise ontológica com um estudo de caso de um modelo que se conforma à especificação “Data Exchange in the Process Industry” (DEXPI), destinada a fornecer uma troca de dados homogênea entre sistemas CAD de diversos fabricantes. A análise ontológica da especificação DEXPI DTI, para construir uma relação com uma ontologia bem fundamentada, levanta um conjunto de propriedades desejáveis para um modelo destinado a ser usado na interoperabilidade.

Embora a conquista da interoperabilidade técnica entre DTIs DEXPI e ontologias representadas em OWL seja evidente, diversos desafios foram identificados no âmbito da interoperabilidade semântica, especificamente em relação à clareza/inteligibilidade, concisão, extensibilidade, consistência e essência. Essas questões representam obstáculos significativos para alcançar uma integração de sistemas perfeita. Além disso, se o padrão DEXPI evoluir para um padrão de facto para a representação de DTIs em um conjunto mais amplo de domínios do que inicialmente pretendido, essas questões destacadas poderiam potencialmente atrasar sua adoção e dificultar sua integração em sistemas diferentes.

Palavras-chave: Ontologia. Interoperabilidade. DEXPI. Diagrama de Tubulação e Instrumentação.

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

O3PO	Offshore Petroleum Production Plant Ontology
DEXPI	Data EXchange in the Process Industry
CAE	Computer-Aided Engineering
CAD	Computer-Aided Design
P&ID	Piping and Instrumentation Diagram
e-P&ID	Electronic Piping and Instrumentation Diagram
EIF	European Interoperability Framework
EP/EPC	Engineering-Procurement-Construction
OWL	Ontology Web Language
ISO	International Organization for Standardization
HAZOP	Hazard and Operability Study
CIB	Comprehensive Information Base

SUMMARY

1 INTRODUCTION

In ancient small nomadic societies, people had a holistic understanding of production and exchanged knowledge within their communities. The industrial revolution ushered in a change, bringing advanced tools and a distinct focus on specialization. Our tools have evolved into many intricate and specialized software applications in today's technology and digital communication era. To achieve efficient specialization, contributions toward a common goal must be possible with minimal friction (TURK, 2020). The proliferation of diverse software systems, which makes systems and decision-makers struggle to communicate and interact cohesively, results in isolated islands of automation (ENCARNAÇÃO; LOCKEMANN, 1990) and reduced project performance (HOWARD et al., 1989). We can solve the challenge of highly segregated architectures by meticulously connecting scattered pieces of information spread across these isolated silos (GUIZZARDI, 2020). This laborious task becomes impossible as soon as the number of applications and information grows.

One way to achieve consensus and enhance interoperability is through standardization, using formally agreed-upon documents referencing common and recurrent practices. When viewed through a technological lens, interoperability splits into two key facets: Semantic Interoperability and Technical Interoperability (TURK, 2020). Technical interoperability addresses data exchange, while Semantic interoperability is fixated on the meaning of the data, ensuring its usability across diverse systems. Enabling effective data exchange and utilization across various systems requires implementing standardized procedures. Standardization serves as the bridge for system integration, fostering a shared understanding among relevant stakeholders. The extent to which a standard enhances semantic interoperability between systems hinges on its proficiency in accurately representing the fundamental concepts within its particular domain of discourse.

Currently, one of the common approaches for enhancing syntactic and semantic interoperability is using ontologies, allowing data from different domains to be better used and understood across different systems. The effectiveness of ontologies grows when they find adoption within active communities, are applied as de facto standards within significant data repositories, and demonstrate seamless interoperability with other ontologies (BODENREIDER, 2008).

In oil and gas context, one particular use case of ontologies is establishing formal definitions and relations between domain-specific elements. The conceptual modeling oil and gas community proposes ontologies to describe many aspects of oil production, from geological characteristics to the equipment and connections of a petroleum production plant (DINIZ et al., 2012; GARCIA et al., 2020; SANTOS et al., 2022b). More recently, engineers and geologists have used it as a baseline to express the semantics of the RESQML (KING et al., 2012) and ISO 15926 (ISO/TS, 2019) standards.

Throughout the history of the process industry, Piping and Instrumentation Diagrams (P&IDs), which are documents employed to exchange information concerning installations, equipment, and related elements (TOGHRAEI, 2019), have traditionally served as representations of production plants, initially as physical drawings and more recently in digital formats. Within this context lies a significant opportunity to utilize the domain knowledge encapsulated within a production plant ontology to enhance the semantic richness of plant representations by integrating specialized domain vocabulary.

This work's objective is to evaluate the contribution of ontology to make explicit the meaning of the entities described in a P&ID model and provide an architecture to represent a P&ID in ontological knowledge bases. We achieve this clarification through a case study using the Offshore Petroleum Production Plant Ontology (O3PO), which is a well-founded domain ontology of production plant physical assets and associated properties and P&IDs conforming to the Data EXchange in the Process Industry (DEXPI) specification, which serves the purpose of proportioning homogeneous data interchange between 2D CAD from diverse vendors.

We organize the following of this work into distinct sections. The theoretical background delves into key concepts such as interoperability, standards, plant topology models, and ontologies, followed by the related work focuses on the specific plant topology model, the oil and gas domain ontology, ontology evaluation, and enhancements made to plant topology models. The section on semantic interoperability analysis outlines the approach employed to identify correspondences between the two models and presents the results obtained from this analysis. The technical interoperability analysis provides an architecture to represent plants described following plant topology models in ontological knowledge bases, followed by a brief discussion and a conclusion.

2 THEORETICAL BACKGROUND

We divide this section into three parts. The first delves into interoperability and standardization, portraying diverse perspectives and definitions in the literature. The second part centers on piping and instrumentation diagrams, elucidating their significance, applications, and representations. The final part addresses ontologies in computer science, spotlighting the role of domain ontologies in structuring data within specific domains for enhanced comprehension and organization.

2.1 Interoperability and standardization

In the realm of interoperability, a multitude of aspects warrant consideration. Some proponents advocate that they can establish a solid basis for effective interoperability and collaboration within intricate networks of entities when delving into the interactions between systems, individuals and systems, and individuals with each other (TURK, 2020).

As shown in Figure 1, this division neatly classifies the specific interoperability concerns for humans and machines. The central issues revolve around legal and organizational interoperability for humans, while machines focus on semantic and technical interoperability domains.

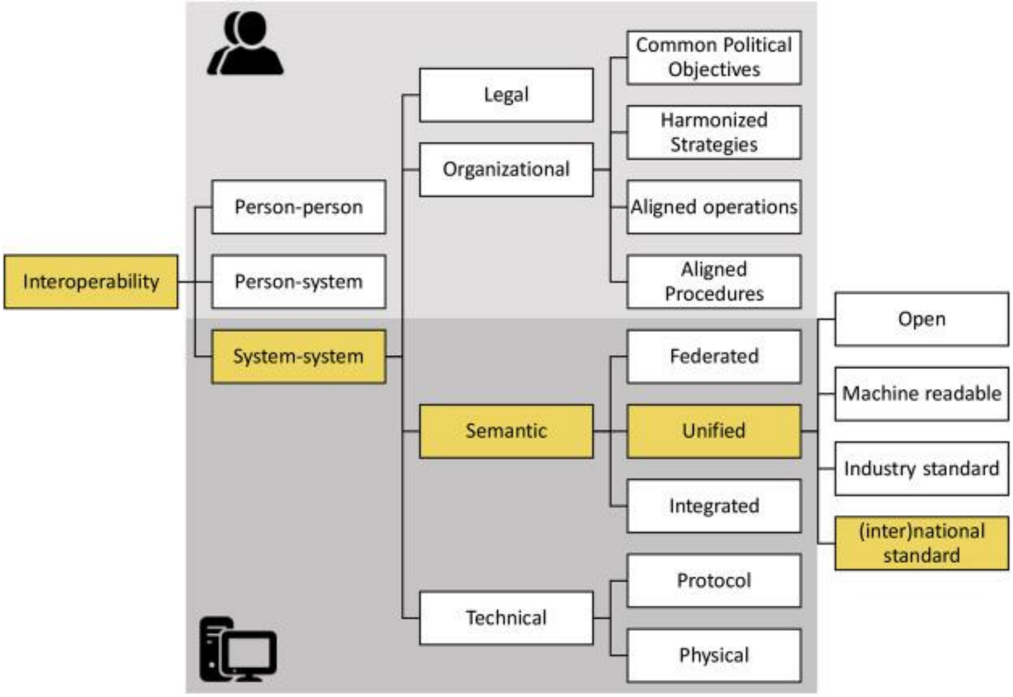


Figure 1 - Map of interoperability in construction (TURK, 2020)

We define interoperability between systems as the ability of two or more systems or components to exchange information and use the information that has been exchanged (IEEE, 1991). Also, in the context of system interactions, it can be understood as the capability of two or more systems or components to exchange information effectively and utilize the exchanged data (ISO, 2013).

From other sources, we further define interoperability as the ability of multiple systems, differing in hardware, software platforms, data structures, and interfaces, to exchange data with minimal loss of content and functionality (RILEY, 2017). It is even seen as the capability of distinct information systems to communicate with each other, sharing data, information, and knowledge efficiently and safely (GARCIA et al., 2017).

Interoperability is multidimensional, encompassing diverse perspectives and approaches across various application domains. This complexity is evident in the multitude of definitions provided by different communities, with some studies even identifying as many as 34 definitions of interoperability across the literature (FORD et al., 2007; GÜRDÜR; ASPLUND, 2018).

The separation of different levels of interoperability, such as the differentiation between technical interoperability as the capability of exchanging information and semantic interoperability as the recipient's capability of using the given information (BENSON TIMAND GRIEVE, 2021), is further outlined by the European Union's interoperability framework (COMMISSION; FOR INFORMATICS, 2017), and plays a crucial role in facilitating smooth and effective collaboration among various entities and systems, breaking interoperability in:

- Legal interoperability ensures that organizations operating under different legal systems, contexts, policies, and strategies can seamlessly work together.
- Organizational interoperability aligns business processes, responsibilities, and expectations toward shared objectives, fostering inter-organizational relationships between service providers and users.
- Semantic interoperability safeguards the meaningful exchange of data between parties, encompassing both semantic and syntactic aspects within the EIF framework.
- Technical interoperability involves the physical connection of systems, encompassing networks, cloud computing, interfaces, and security measures.

Another similar view is the separation of interoperability in the four layers of institutional, data, human, and technology (PALFREY; GASSER, 2012). This view served as a base for the medical separation of interoperability in layers (BENSON TIMAND GRIEVE, 2021), which puts regulations as defined by the European standard under the Institutional umbrella and understands that processes fall under the human layer, as shown in Figure 2.

The interoperability types represent only a portion of the different interoperability types in the literature, where (FORD et al., 2007; GÜRDÜR; ASPLUND, 2018) identified 64 types.

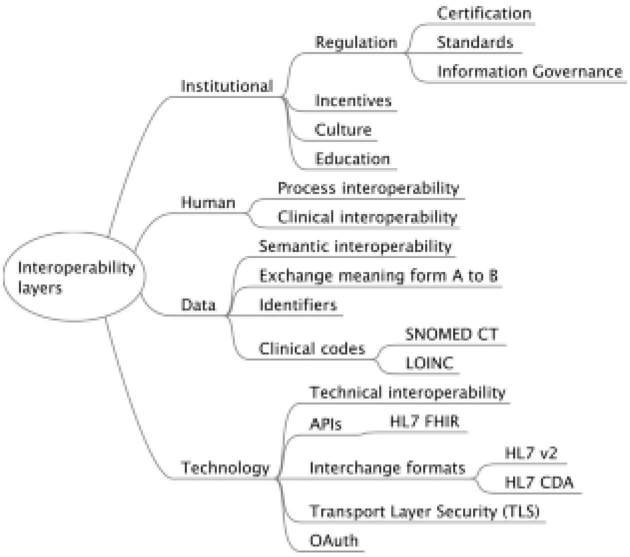


Figure 2 - Layers of interoperability in the medical domain (BENSON TIMAND GRIEVE, 2021)

One aspect said to impact semantic interoperability in information systems directly is their level of openness, with less open systems posing challenges in sharing and utilizing data outside their environment. Open systems facilitate interaction through well-documented methods, enabling seamless information sharing and fostering developer communities to enhance system capabilities with add-ons and extensions (TURK, 2020).

The oil and gas community sees standardization as a key factor in promoting interoperability. A standard is a formally agreed-upon document approved by an authorized body, serving as a reference for common and recurrent practices, outlining regulations, instructions, or attributes applicable to activities or their outcomes, all to promote an optimal level of structure within a specific environment (ISO/IEC, 2004).

2.2 The piping and instrumentation diagram (P&ID)

Industry has made significant efforts to standardize data exchange in process industries, particularly in domains pertinent to the planning phase (FILLINGER et al., 2017). Of notable importance, the P&ID assumes a central role as a critical document for plant design and operation. The simplified visual depiction of processes inherent in P&IDs facilitates the quick grasp of the functional context of a system.

Toghraei and colleagues (TOGHRAEI, 2019) consider the piping and instrumentation diagram (P&ID) as the bible of a petrochemical process plant, providing information essential for equipment manufacturing, installation, commissioning, start-up, and ongoing plant operation. This document maintains its significance throughout the project's life cycle, proving invaluable from the initial design stages through the operational phase, playing a role in generating supplementary documents, spanning piping isometric drawings, equipment and instrument lists, control philosophy, loop diagrams, material take-offs, and more. Figure 3 shows the different data structures used to represent the different aspects of an asset's life cycle.

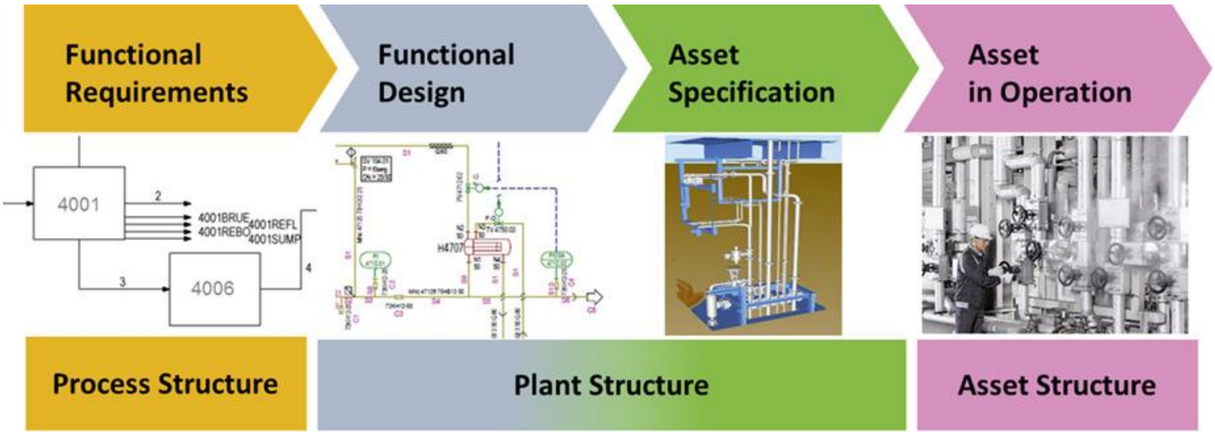


Figure 3 - Four asset life cycle aspects and their data structures (WIEDAU et al., 2019)

P&IDs are used in many different process plants, ranging from oil refineries and gas processing facilities to food processing and pharmaceutical complexes, often employed by plants producing non-discrete products (TOGHRAEI, 2019). In contrast, industries like automotive manufacturing, producing discrete items like cars, veer from utilizing P&IDs. They comprehensively depict the process steps within a process plant, essentially translating the plant into a visual representation. Presented in schematic format, P&IDs, as shown in Figure 4, utilize predefined symbols to illustrate pipes, process equipment, and control systems without adhering to scale or geographical orientation (TOGHRAEI, 2019). These equipment symbols often reflect a lateral view of the equipment's shape, striving to maintain proportional accuracy where feasible.

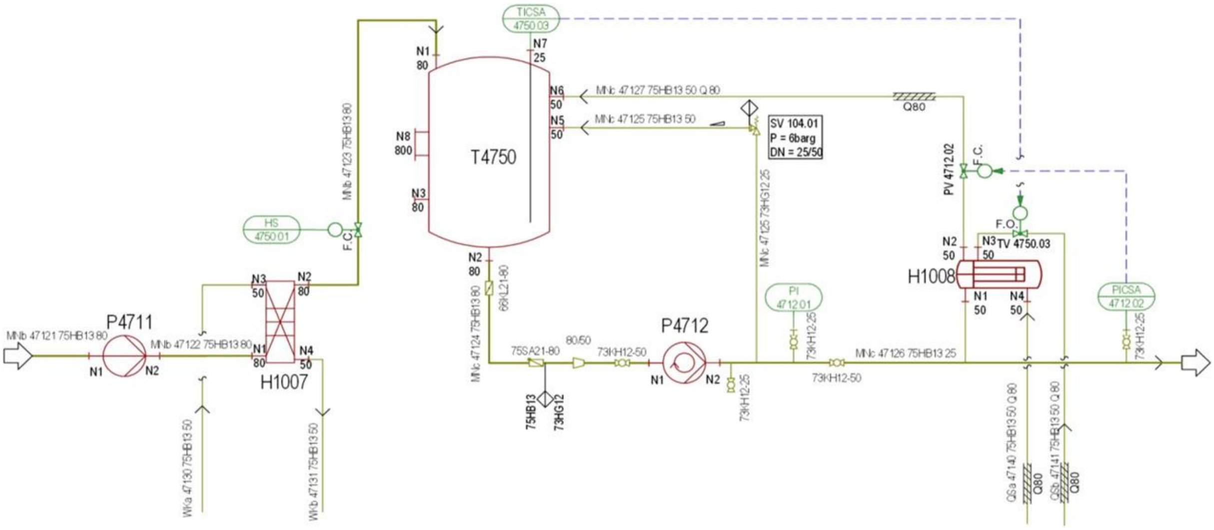


Figure 4 - Example of a P&ID (WIEDAU et al., 2019)

Distinct lines featured on P&IDs denote pipes and signals, although the line lengths do not mirror the dimensions of pipes or signal carriers, such as wires. Notably, P&IDs serve a range of stakeholders, including operations personnel, control technicians, engineers, and maintenance staff, acting as crucial resources for various functions (TOGHRAEI, 2019), as depicted by Figure 5. The increasing adoption of electronic P&IDs (e-P&IDs) is driven by their ease of transfer and incorporation of “smart” capabilities, further contributing to their growing popularity.

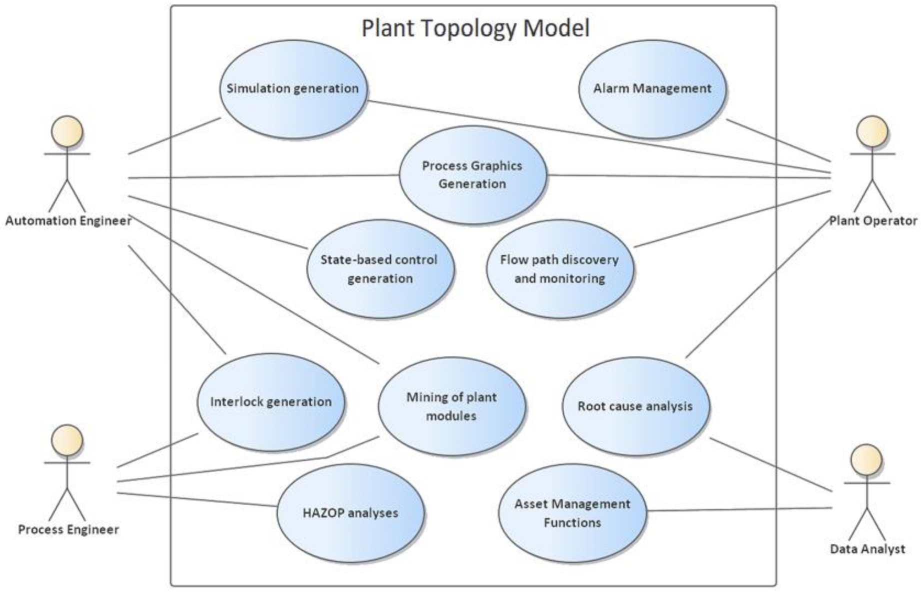


Figure 5 - Use cases of plant topology models (KOZIOLEK; RÜCKERT; BERLET, 2020)

By adopting a topology model founded on object-oriented principles and imbued with semantic information, the potential for algorithms to process P&IDs becomes evident, paving the way for the automation of numerous plant engineering tasks that currently demand manual execution (KOZIOLEK; RÜCKERT; BERLET, 2020). This collaboration between process engineers and software developers underscores the ongoing efforts towards establishing standardized topology models, fostering the possibility of facilitating model exchange across diverse CAD tools from various vendors.

When discussing the oil and gas sector, these diagrams can encompass more than just chemical process plants. They are also utilized to depict elements related to the entire process of extracting oil from the well to the platform. This context involves highly specific industry terminology and considerations.

2.3 Ontology

Experts often exchange domain knowledge informally, with their concepts well comprehended but not necessarily interpretable by machines. To transform this expertise into machine-readable forms, it becomes essential to capture and formalize the collective conceptualization of these experts in a language that computers can process effectively. Figure 6 depicts the main steps of knowledge capture into conceptual models.

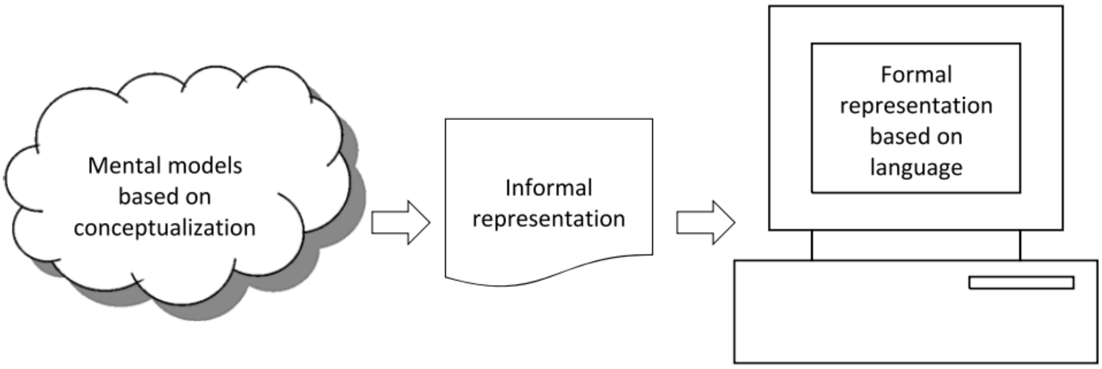


Figure 6 - Steps of conceptual model creation (ABEL; PERRIN; CARBONERA, 2015)

In the oil and gas industry context, inadequate modeling practices are the main drivers of challenges related to data integration (ABEL; PERRIN; CARBONERA, 2015). This underscores the significant role of inadequate modeling in addressing information heterogeneity and improving interoperability.

Semantic interoperability relies on the combined support of two critical components: ontologies and Ontology (GUIZZARDI, 2020). On the one hand, ontologies act as meaning contracts that capture the conceptualizations represented in information artifacts. On the other hand, Ontology is a philosophical discipline, namely the branch of philosophy that deals with the nature and structure of reality (GUARINO NICOLA AND OBERLE, 2009).

An ontology is a logical theory that accounts for the intended meaning of a formal vocabulary and the commitment of this vocabulary to an individual's worldview (GUARINO, 1998). It encompasses the study of existence and the entities that truly inhabit reality, serving as an inventory of the constituents of reality (BAKER, 2007). Logical theories, among other semantic models, and their structure complexity and expressivity levels are portrayed on a spectrum in Figure 7.

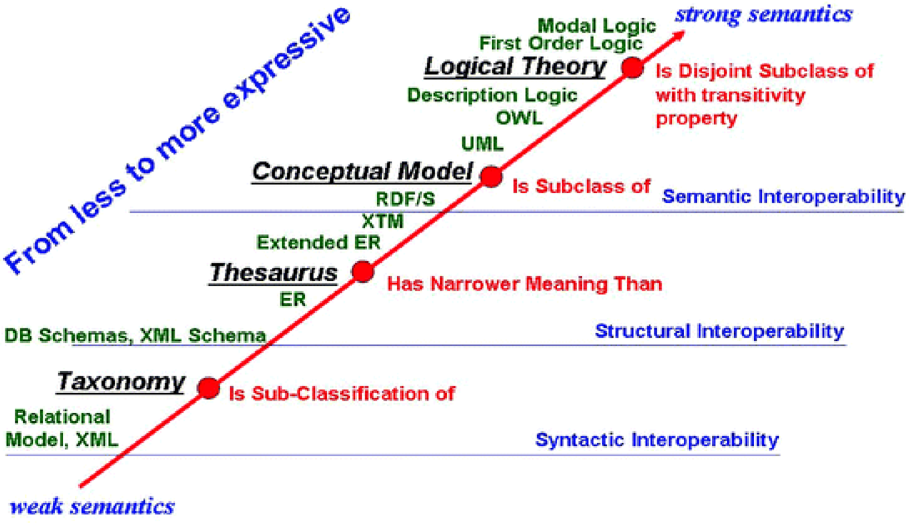


Figure 7 - Spectrum of semantic models (OBRST, 2010)

In Computer Science, ontological definitions diverge based on whether they lean towards representing domain conceptualizations as a reflection of someone's perspective and a mental abstraction of the domain or direct representations of real-world domain entities (SMITH, 2004). Applying these theories in constructing conceptual models leads to the development of an ontology, which functions as a computational artifact encapsulating the description of a domain's concepts and their interconnected relationships (GUARINO NICOLAAND OBERLE, 2009).

One definition characterizes ontology as a formal, explicit specification of a shared conceptualization (STUDER; BENJAMINS; FENSEL, 1998), where 'conceptualization' entails constructing an abstract model of a phenomenon in the world by identifying its relevant concepts. The term 'explicit' signifies that the types of concepts used and the limitations placed on their application are meticulously defined. 'Formal' characterizes the ontology as machine-readable, excluding the use of natural language. Moreover, 'shared' underscores the ontology's capacity to capture consensual knowledge, emphasizing its acceptance by a group rather than being confined to an individual perspective.

Conversely, an alternate viewpoint posits that the purpose of an ontology is to portray reality, distinct from individuals' subjective interpretations (SMITH, 2004). This stance molds the concept of ontology into that of a representational artifact, comprising a taxonomy as the proper part, whose representations are intended to designate some combination of universals, defined classes, and certain clear relations between them (ARP; SMITH; SPEAR, 2015).

A domain represents a distinct segment of reality aligned with a scientific discipline or specific area of knowledge (ARP; SMITH; SPEAR, 2015). In the context of our study, this refers to Oil and Gas production plants and equipment. What we call domain ontologies are ontologies encompassing a main taxonomy of entities of a given domain, organized by the subsumption relationship, and allowing the inclusion of restrictions, axioms, and relations that characterize the behavior of this particular domain. Figure 8 depicts how elements from the Oil and Gas production environment can be described using a domain ontology.

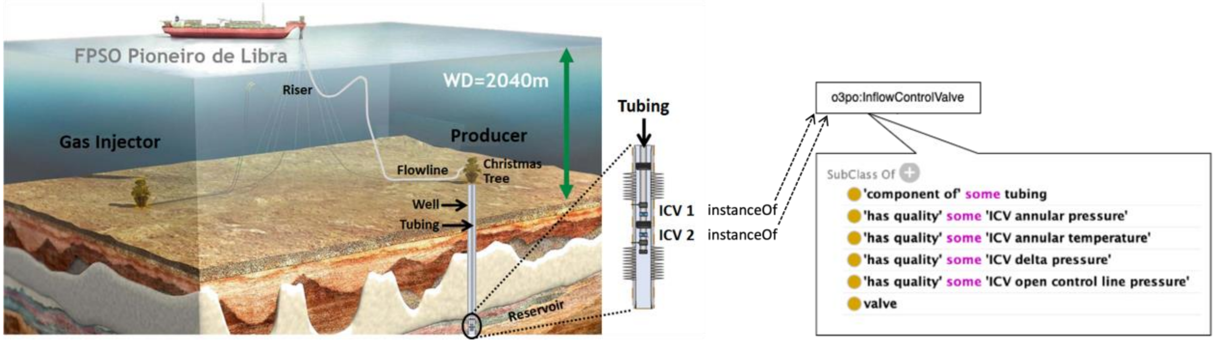


Figure 8 - Scheme displaying entities in a petroleum production system (SANTOS et al., 2022a)

Although domain ontology is defined as the “consensual” specification of the conceptualization of a domain, numerous research teams are currently developing distinct domain ontologies to cater to their localized requirements, inadvertently leading to the emergence of isolated information silos. Addressing this challenge involves adopting a broad representation of categories and relationships common across various domains, serving as a guiding framework for ontology development. This is referred to as a *top-level ontology* aimed at fostering consistent ontology development practices across diverse domains, thereby mitigating the issue of incompatible ontologies (ARP; SMITH; SPEAR, 2015).

A domain ontology structures data within a particular domain, augmenting its comprehensibility, accessibility, and computability. In contrast, a top-level ontology assumes a unique role in organizing data derived from numerous domain ontologies. This coordination fosters heightened levels of semantic interoperability across information systems that incorporate these ontologies (ARP; SMITH; SPEAR, 2015).

3 RELATED WORK

This chapter describes the technology adopted in the industry to face the problem of data interoperability in the petroleum plant design and operation domain. We describe the standards we have studied and the previous works that applied ontologies as a technology to enhance the semantic content of industrial standards.

3.1 Data EXchange in the Process Industry

The Data EXchange in the Process Industry (DEXPI) initiative addresses the interoperability challenges across the process industry's life cycle, spanning process development, engineering, operation, and maintenance (WIEDAU et al., 2019). By involving diverse organizational units and disciplines within and beyond process industry companies, the initiative tackles the imperative need for standardized digital communication norms in an era characterized by digitalization and globalization.

Data interoperability and exchange challenges pervade the process industry, particularly within process plants' planning, construction, and operation, necessitating streamlined practices. This stems from the lack of consensus and coherence among various systems, resulting in significant efforts for data interoperability. Stakeholders involved in these projects, encompassing EP/EPCs, owner-operators, vendors, site services, and authorities, encounter obstacles while striving for seamless data exchange. The absence of an industry-wide standardized exchange format and limitations posed by conventional manual formats exacerbate these predicaments (DEXPI INITIATIVE, 2021).

In response, a comprehensive data exchange model rooted in the ISO 15926 standard is being pursued, aiming to enhance efficiency throughout the life cycle of (petro-)chemical plants. Core objectives drive this initiative to facilitate smooth data exchange, ensure interoperability, and promote seamless data integration. The envisioned approach seeks to mitigate diverse industry challenges, such as circumventing data loss during format conversions, simplifying engineering data handover during and after projects, dismantling data exchange barriers across varying CAE systems, enabling CAE system-independent plant data storage, and facilitating harmonious coexistence of disparate CAE systems within single organizations (DEXPI INITIATIVE, 2021).

In summation, the absence of standardized data exchange standards has posed significant hurdles during project execution and operational phases within the process industry. The pursuit of a data exchange model grounded in the ISO 15926 standard underpins efforts to conquer these challenges, offering a comprehensive solution that amplifies efficiency, fosters interoperability, and seamlessly integrates data across the entire life cycle of process plants (DEXPI INITIATIVE, 2021).

The DEXPI initiative facilitates the exchange of P&IDs, serving as a prime illustration of homogeneous data interchange between 2D CAD tools from diverse vendors. This undertaking is advanced by the DEXPI consortium, which is dedicated to formulating a standard primarily concentrated on seamless data exchange among intelligent P&ID systems. Comprising prominent owners, operators, EPCs (engineering, procurement, and construction), research institutions, universities, and software developers, the DEXPI consortium collectively strives towards this shared objective (FILLINGER et al., 2017).

The DEXPI initiative's initial milestone centered on achieving standardized data exchange, particularly on Piping and Instrumentation Diagrams (P&IDs). To achieve this, several design principles were established, including the separation of specification (DEXPI) and implementation (ProteusXML), the distinction between engineering content and graphical representation, and the utilization of internationally recognized specifications in its taxonomy definition, which is exemplified by Figure 9 (WIEDAU et al., 2019).

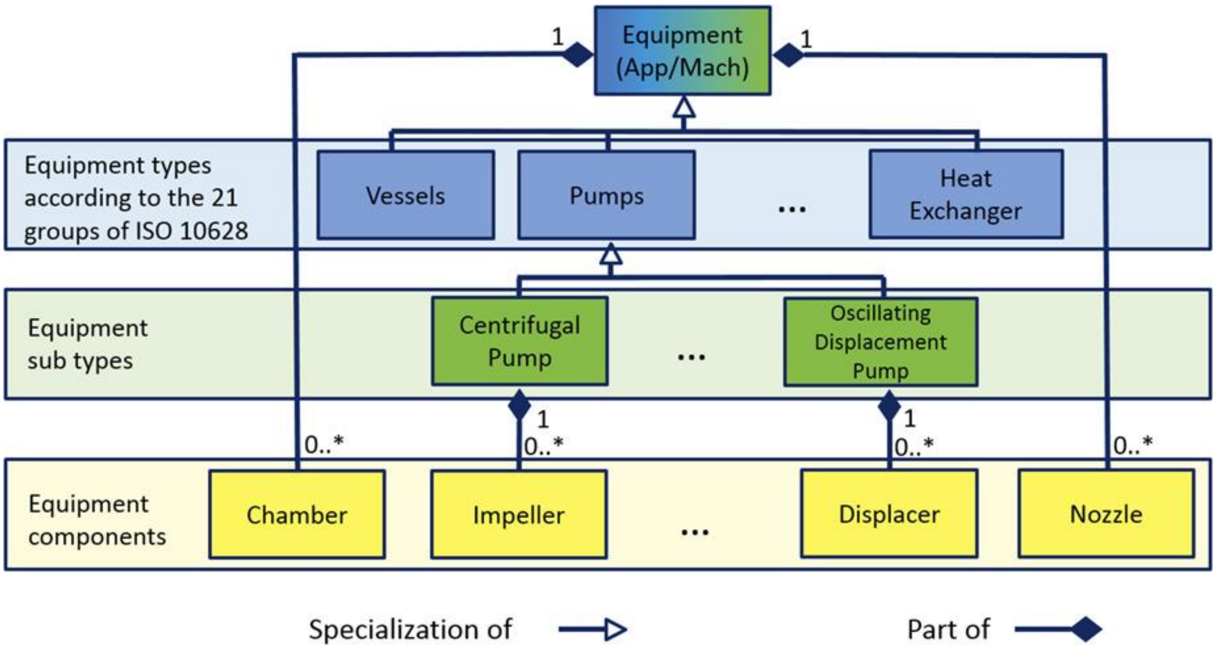


Figure 9 - DEXPI equipment taxonomy (WIEDAU et al., 2019)

Separating specification (DEXPI) from implementation (ProteusXML) design necessitates aligning P&ID files, initially in the ProteusXML format, with the DEXPI conceptual model. To proficiently manipulate Proteus files within the framework of the DEXPI conceptualization, the P&ID Toolbox offers a comprehensive implementation of the DEXPI information model. Furthermore, this toolbox streamlines the import and export of DEXPI P&IDs and their visualization as images (PNB PLANTS & BYTES GMBH, [s.d.]).

3.2 Smart P&IDs

In terms of facilitating P&ID functions, another work introduces an integration framework aimed at incorporating the HAZOP expert system “LDGHAZOP” with the commercial process design package “Smart Plant P&ID (SPPID, Intergraph)” to establish HAZOP as an integral element of process design (CUI; ZHAO; ZHANG, 2010). Also, from using the DEXPI standard, another work proposes using AI techniques to accelerate the drawing process of making a P&ID (OEING et al., 2022).

Upon reviewing the preceding chapters, it becomes evident that Piping and Instrumentation Diagrams hold significant utility across various industries, exchanging information across diverse groups to achieve various purposes. Furthermore, numerous approaches exist for extending their functionality to cater to specific tasks. These applications encompass both ad-hoc models and ontological approaches, with the latter harnessing the enriched semantic capacities of ontologies and the available tools for reasoning and fact-checking.

3.3 Ontologies and P&IDs

The concept of the Comprehensive Information Base, shown in Figure 10, refers to an approach for tackling the hurdles of information integration, collaborative engineering, and concurrent engineering using ontologies (WIESNER; SAXENA; MARQUARDT, 2010). This concept encompasses consistency management, covering the execution of consistency rules to detect inconsistencies and activities involving detecting, diagnosing, and resolving inconsistencies between systems.

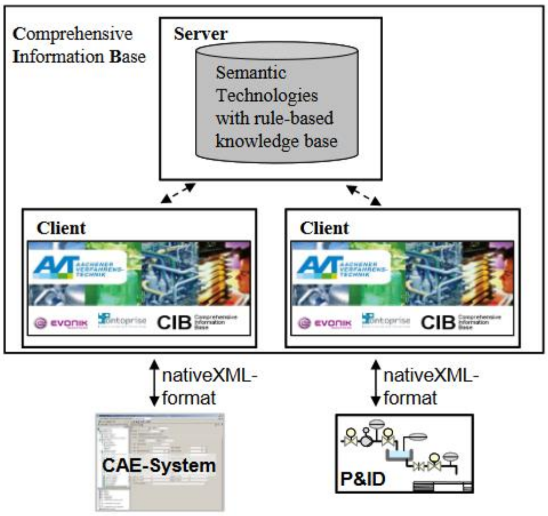


Figure 10 - Comprehensive Information Base (WIESNER; SAXENA; MARQUARDT, 2010)

Regarding finding correspondences between ontologies and chemical processes, some works within the process industry context use ontology mapping to synchronize simulation and diagram models, effectively creating process information models referred to as intelligent diagrams (LEHTONEN; OTHERS, 2007). Another proposes an ontology-based framework for automatically generating redundancy-free interlock code from piping and instrumentation diagrams (STEINEGGER; MELIK-MERKUMIANS; SCHITTER, 2017).

A similar work presents an application for importing DEXPI plant topology models in the CAYENNE topology model (KOZIOLEK; RÜCKERT; BERLET, 2020). This approach automatically maps classes from one model to the other, relying on manual intervention for cases where there is no way to do it directly.

Another work introduces a novel rule ontology designed to facilitate the representation of control engineering rules in a digital format (SCHMIDBERGER; DRATH; FAY, 2006). The application of these rules to electronically accessible plant information could unlock their potential value through the use of neutral and tool-agnostic object-oriented file formats, offering the capability to store industrial plant information using hierarchical object models.

3.4 Offshore Petroleum Production Plant Ontology

The Offshore Petroleum Production Plant Ontology (O3PO) is a well-founded domain ontology of production plant physical assets and associated properties, with the primary objective of establishing a unified, formal vocabulary encompassing entities related to offshore petroleum production plants, responding to the voluminous data in contemporary offshore operations and the extensive collaboration among numerous companies during field production, including assets integral to the oil path between reservoir and platform, like wells and subsea equipment (SANTOS et al., 2022a, 2022b).

In addition to O3PO, alternative ontologies are available for representing the production environment. Some of these ontologies were specifically developed for the petrochemical process domain (DINIZ et al., 2012), while others have broader applications. An example of an alternative ontology is the "Plant Equipment, Topology, and Instrumentation Ontology" (STEINDL; KASTNER, 2020), which the authors have designed for Industrial Energy Systems. For the oil and gas domain, the work of (ABEL, PERRIN, CARBONERA, 2015)proposes ontologies focusing on geological characteristics.

O3PO's development is rooted in the NeOn methodology (SUÁREZ-FIGUEROA; GÓMEZ-PÉREZ; FERNÁNDEZ-LÓPEZ, 2015), drawing from data sourced across diverse industry forums, encompassing industry-wide glossaries and oil and gas-specific standards and references (SANTOS et al., 2022b).

Framed upon the Basic Formal Ontology (BFO) (OTTE; BEVERLEY; RUTTENBERG, 2022) and incorporating components from the GeoCore Ontology (GARCIA et al., 2020) and IOF-Core Ontology (SMITH et al., 2019), O3PO takes shape as an oil and gas domain ontology. O3PO incorporates diverse elements such as classes, individuals, properties, and property restrictions. Since we have implemented O3PO in OWL 2 language, a computer-processable format, it can use tools for logical consistency checks and inferential reasoning based on defined data model relations (SANTOS et al., 2022a).

3.5 Ontology evaluation

Within the realm of ontology development and utilization, the significance of high-quality ontologies becomes evident. This importance naturally leads us to the fundamental matter of assessing ontology quality. Literature has proposed various approaches to address this issue. The quality assessment of ontologies can be approached from various dimensions, such as their coverage of a specific domain, their compatibility for mapping to an upper ontology, and the types of reasoning they support, alongside many others (OBRST LEOAND CEUSTERS, 2007). Alternatively, we can evaluate ontologies based on intrinsic characteristics like consistency, completeness, conciseness, expandability, and sensitiveness (GÓMEZ-PÉREZ, 2001).

We summarize the various metrics for ontology evaluation into the following quality categories: structural, functional, analytical, pragmatic, syntactic, cognitive, semantic, social, and practical (LOURDUSAMY; JOHN, 2018). At the same time, different ontologies may vary in their design in degrees of freedom along the aspects of Vocabulary, Syntax, Structure, Semantics, Representation, and Context (VRANDEČIĆ, 2009).

In recent decades, there has been a notable surge in research papers dedicated to assessing ontology quality (GÓMEZ-PÉREZ, 2001; GRUBER, 1995; GUARINO; WELTY, 2002; NEUHAUS et al., 2013; OBRST LEOAND CEUSTERS, 2007; VRANDEČIĆ, 2009), with publications exploring an extensive range of quality metrics and ontology design methodologies, primarily focusing on evaluating ontology quality across diverse contexts (MCDANIEL; STOREY, 2019). Figure 11 provides a timeline showcasing some of these evaluation initiatives. Additionally, researchers have proposed methodologies for ontology evaluation, considering a wide spectrum of quality metrics derived from the existing literature (BANDEIRA et al., 2016).

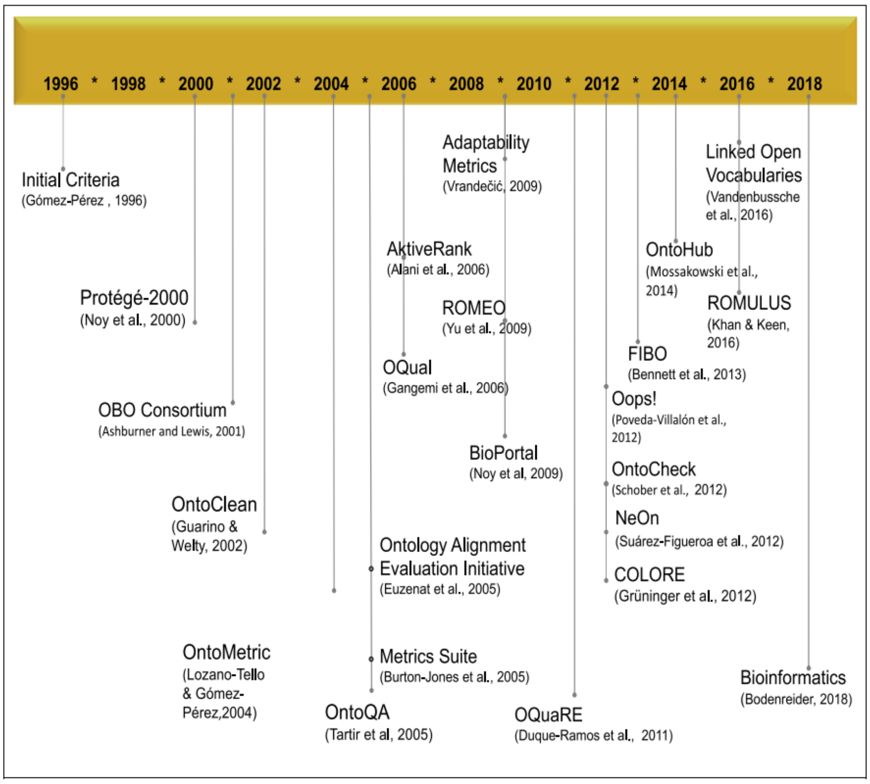


Figure 11 – Timeline of ontology evaluation initiatives (MCDANIEL; STOREY, 2019)

4 SEMANTIC INTEROPERABILITY ANALYSIS

One of our goals is to display the potential for semantic interoperability of process plants adhering to the DEXPI standard, streamlining their integration with the Offshore Petroleum Production Plant Ontology. In the Oil and Gas industry context, we have selected the O3PO ontology due to the intersection of the modeled domain entities with the entities referred to by DEXPI. Also, the O3PO capacity to integrate data from diverse sources about an oil plant enriches the model of reality and facilitates more useful queries.

This work is primarily motivated by the significant overlap observed between certain elements present in the DEXPI standard and O3PO, portrayed by Figure 12, facilitating a pathway for integration. Both models refer to *Physical quantities*, *Component types*, and *connection topology*, presenting an opportunity to bridge the gap between the broader DEXPI model and the domain-specific O3PO. This integration approach entails setting aside components unique to DEXPI, such as graphical representations and elements exclusive to O3PO, like Oil and Gas-specific vocabulary.

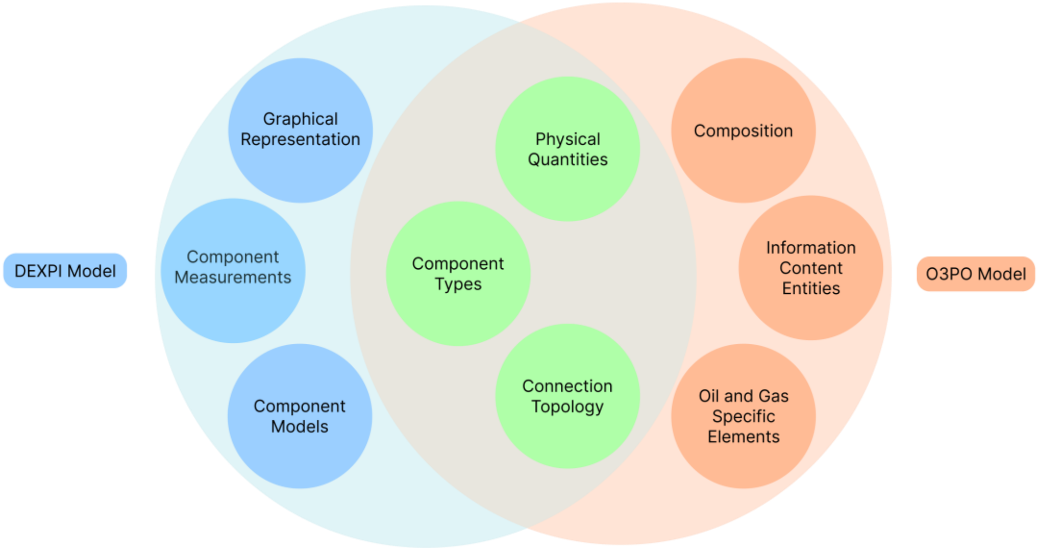


Figure 12 - Overlap of representation between DEXPI and O3PO

Within the framework of the DEXPI standard, distinct packages focus on various facets of a production plant integration possibilities within specific packages, as shown in Figure 13. Package Piping stands out for its depiction of connection topology, while both Package Equipment and Package Piping contribute by outlining diverse equipment types within the plant. Additionally, Package PhysicalQuantities proves valuable in elucidating the array of physical quantities associated with production elements, such as surface areas of frequencies.

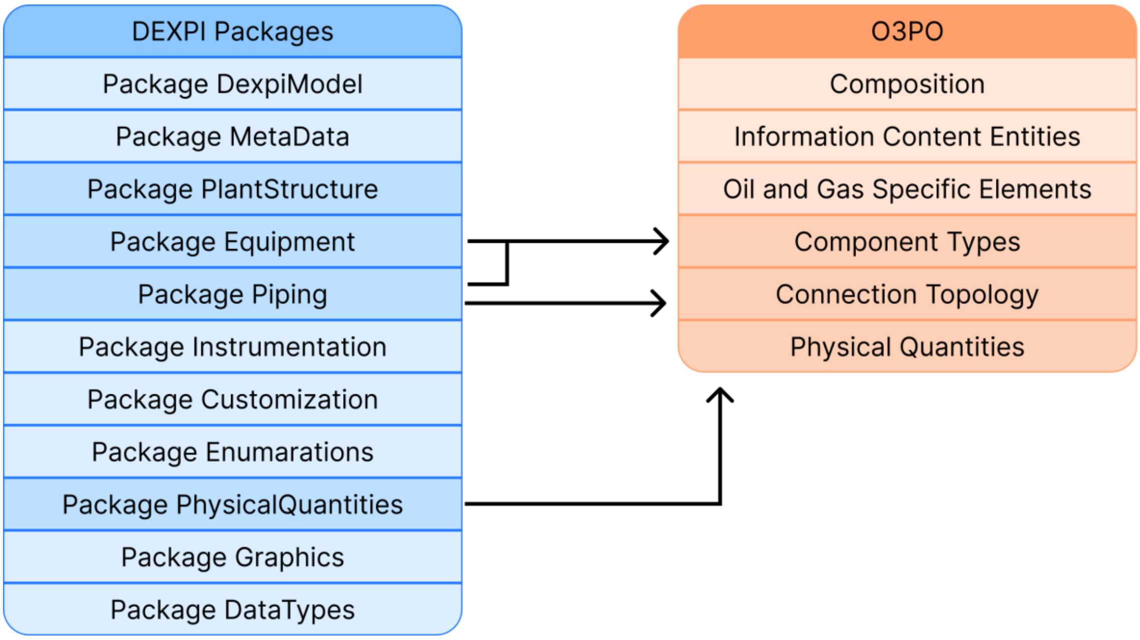


Figure 13 - DEXPI packages and their representation overlap with O3PO

4.1 Methodology

To assess the overlap of intended models between the conceptualizations of DEXPI and O3PO, a classification process is proposed to establish a depiction of DEXPI classes with O3PO classes and relationships, using elements from its imported ontologies when necessary.

We selected the packages from the DEXPI P&ID specification based on their apparent overlap with represented elements in O3PO. Package piping and equipment were chosen for this study, aligning with the component types and relations in O3PO.

From the selected DEXPI packages, after a more in-depth analysis, the most representative classes identified were *PipingComponent* and *Equipment*. From there, their subclasses were analyzed down the taxonomy, stopping whenever new terms brought out the same meaning as their parent classes. In addition, other classes help to explain dependent concepts.

We accomplish the ontological analysis to represent DEXPI classes in terms of O3PO classes and relations. In the cases where we could not represent the classes adequately, we explain the reasons for this limitation.

While some of the established correspondences rely on concepts directly defined in O3PO, many require definitions from O3PO's imported ontologies, such as IOF-Core for more general terms like assemblies and components or BFO for capabilities and functions. It is essential to note that we considered only classes and relations defined in these important ontologies and imported by O3PO for the analysis.

4.2 Results

By applying this methodology, we have arrived at the corresponding O3PO classifications for some DEXPI piping and equipment classes. While simultaneously pinpointing the specific challenges for the cases that render the classification unfeasible. For reference, the definitions from the DEXPI P&ID specification used across this volume are shown in <ATTACHMENT A>, in the same order they appear in the text.

Consider the case of the DEXPI:PipingComponent, as outlined in the DEXPI specification as “A piping component”. Given the vague nature of its definition, by taking into consideration its subtypes and the understanding that Piping and Instrumentation Diagrams (P&IDs) primarily illustrate assemblies rather than individual components, one can classify DEXPI:PipingComponent as a BFO:MaterialEntity, and BFO:bearer_of some BFO:Function, and Core:prescribedBy some Core:DesignSpecification and O3PO:Component_of some O3PO:Pipeline.

Based on that, we can also specify DEXPI:InlinePrimaryElement, defined in the DEXPI specification as “An inline primary element”. Since the definition is also lacking specificity, the fact that it is a subclass of DEXPI:PipingComponent and all its subclasses are used to perform some measurement, we can say that a DEXPI:InlinePrimaryElement is a BFO:MaterialEntity, and a BFO:bearer_of some BFO:Function, and is Core:PrescribedBy some Core:DesignSpecification, and a O3PO:Component_of some O3PO:Pipeline, and Core:hasCapability some Core:MeasurementCapability.

Even though all the elements represented in a P&ID diagram are part of piping, that seems to be the only difference between DEXPI:Equipment, defined as “An apparatus or machine”, and DEXPI:PipingComponent.

For the full results table, please refer to <APPENDIX A>.

4.3 Discussion

Through this classification, we clarified the semantics of the entities represented in a P&ID and provided support for the operator to understand the meaning and intrinsic restriction of the modeled entities. Also, O3PO supports a description of the DEXPI entities while utilizing domain-specific vocabulary tailored to the Oil and Gas industry. Moreover, this approach equips us with a foundational framework built upon ontological principles, fostering open accessibility, collaborative development, and enhanced interoperability.

Further possibilities remain open for mapping the two models, such as establishing a correspondence between DEXPI and O3PO topology models. DEXPI's topology is structured around nodes associated with elements and their interconnections, whereas O3PO presents its topology through connections and fluid supply properties.

Another possibility is the use of O3PO for the creation of a CIB, integrating data from various sources, including DEXPI P&IDs. By allowing the complementing of the data in the P&ID with other information sources, we could bridge the gap between classifications. This enhancement could help accurately categorize valve types such as GasLiftValve, ChokeValve, or InflowControlValve, as depicted in Figure 14.

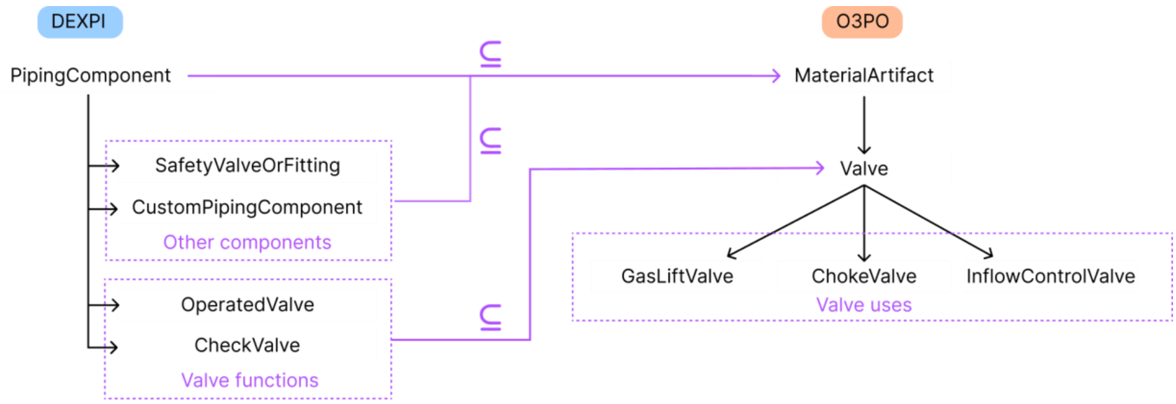


Figure 14 - Gaps in direct mapping between DEXPI and O3PO valves

Integrating the DEXPI standard with the O3PO presented several challenges, including imprecise term definitions, potentially redundant or overly vague definitions that rely on previous understanding of those familiar with the subject matter, the absence of a superclass in definitions, the complexities of multiple inheritances and collapsed entities or relations. These limitations lead us to search the correspondence between the entities mostly based on the entity name, a common practice for mapping models that results in false agreements. This section will elaborate on these challenges to offer a more comprehensive understanding.

4.3.1 Vague definitions

Certain elements within the DEXPI standard exhibit vague definitions and require a comprehensive grasp of the standard's structure and access to industry-specific glossaries for proper comprehension. A few cases of this include DEXPI:PipingComponent, defined as “A piping component”, and DEXPI:ColumnSection, defined as “A column section”, lacking contextualization or specific and explicit identity criteria for the identification of its instances.

4.3.2 Missing definitions

Despite numerous classes in the DEXPI standard, such as DEXPI:PipingComponent, DEXPI:PipingNetworkSegment, and DEXPI:PipingConnection, which define elements in terms of being part of a piping, the standard lacks a clear definition of what constitutes “piping”. This absence of definition poses challenges when attempting to state, for instance, that DEXPI:PipingComponent is a component of piping, as the concept of a piping system is not explicitly defined. Consequently, one must resort to terms like O3PO:Pipeline as an approximation to the implicit DEXPI notion of what a piping system entails.

Another case of missing definitions seems to be the DEXPI:PipeFitting, defined as “A pipe fitting”, without resorting to any external definition. By analyzing its subclasses, the class seems to refer to passive elements assembled between two pipes, like strainers, fittings, and line blinds, that we could categorize as valves.

4.3.3 Collapsed definitions

An illustrative example of a lack of definition and collapsed terms is DEXPI:SafetyValveOrFitting, denoted in the specification as “A safety valve or fitting”. From an ontological viewpoint, this definition muddies the distinction between different types of entities: pipe fittings and valves, both serving the role of ensuring safety. Within the O3PO model, the sole common class that encompasses both pipe fittings and valves is Core:MaterialArtifact. Had the standard implemented separate classes for each, it would have enabled the differentiation between the elements that define their identity and those that specify their utility.

The absence of a valve class raises further questions, particularly regarding how one might quantify the number of valves within a plant. Despite the seemingly straightforward nature of this query, it necessitates a manual count of instances, encompassing DEXPI:OperatedValves, DEXPI:CheckValve, and non-safety fitting instances of DEXPI:SafetyValveOrFitting. Given that these instances are distributed among various subclasses of the DEXPI:PipingComponent and no clear criteria exist within the DEXPI:CustomSafetyValveOrFitting class to differentiate between valves and fittings, this tallying process necessitates human intervention.

4.3.4 Custom Object Role

One parent class present across the DEXPI specification is the DEXPI:CustomObject, differentiating classes originally defined in the standard from user-defined ones. The standard currently counts 57 custom classes, all sharing the same problem.

The issue with custom classes is that instead of relying on the entity responsible for defining a class like “this class is defined in DEXPI 1.3” or “this class was defined by organization X”, its members are defined in a case-by-case manner, using the same pattern of “a DEXPI:CustomY is a Y and is not covered by any of the other subclasses of Y”. Which means that to support the representation of any of the custom classes in DEXPI, one needs to be able to represent and differentiate all other subclasses of the parent class.

Had the DEXPI standard adopted a definition for custom classes, such as “a class not present in the standard definition” instead of a definition reliant on not being other classes, the standard would have failed to clarify the definition. That way, it allows for differentiating standard classes from custom classes in systems that cannot represent all non-classes.

4.3.5 Differentiability of equipment

While the DEXPI specification excels in distinguishing various types of equipment, a similar level of differentiation is not inherently present in O3PO. O3PO cannot differentiate several types of equipment on its own. For example, DEXPI effectively differentiates between Pumps and Compressors, with compressors designated for pressurizing gases exclusively. However, O3PO does not inherently possess this specific capability. The standard could address this limitation by introducing capabilities within O3PO to represent the act of pressuring various fluids.

As far as redundancy goes, there are also classes like DEXPI:TaggedColumnSection that exist to convey that an item is a DEXPI:ColumnSection and is tagged. In the same way, as in previous examples, it is quite interesting to note that a column section is still the same whether it is tagged or not, and the only difference is this relation with a tag that does not affect the identity of the column. The precision and clarity of a model presuppose that the entity has a single representation and that non-specialization properties, like “having a tag”, should not define new entities but be inserted as dependent properties of the entity.

4.3.6 The same definitions in different places

Within the DEXPI specification, the class DEXPI:InlinePrimaryElement encompasses ten distinct subclasses. Two of these subclasses stand out: the DEXPI:FlowMeasuringElement and the DEXPI:ElectromagneticFlowMeter. We have defined the former as “A FLOW MEASURING ELEMENT is a MEASURING ELEMENT that measures a FLOW RATE”. In contrast, the latter is “A velocity flow meter that measures the flow rate of a conductive fluid running through a magnetic field by measuring the charge created when fluid interacts with the field”.

This scenario perfectly illustrates a case where an electromagnetic flow meter inherently functions as a flow-measuring element due to its ability to measure flow. However, in the DEXPI specification, both classes are categorized as direct subclasses of DEXPI:InlinePrimaryElement, thus implying that we can not classify one entity under both classes.

A similar scenario arises when considering DEXPI:HeatExchanger and DEXPI:CoolingTower. The former is “An apparatus or machine that has the capability of heat exchanging”. In contrast, we defined the latter as “A cooler and an air-cooled heat exchanger that is a tall structure through which air circulates by convection”. Notably, the definition of DEXPI:CoolingTower explicitly states that it is an “air-cooled heat exchanger”. Despite this clear relationship, the DEXPI standard does not classify DEXPI:CoolingTower as a subclass of DEXPI:HeatExchanger, showcasing a discrepancy between the definitions and the classification within the standard.

5 TECHNICAL INTEROPERABILITY ANALYSIS

The inherent capacity of representing elements from the DEXPI standard in OWL using the O3PO domain ontology affords a dual advantage. From the technical interoperability perspective, it enables the execution of queries leveraging languages such as SPARQL, facilitating information retrieval from process plants structured in alignment with the DEXPI standard. Simultaneously, from a semantic standpoint, it empowers the utilization of a domain-specific vocabulary and conceptual framework to encapsulate plant-related data. Beyond these advantages, it facilitates seamless integration and aggregation of information from diverse sources and supports fact-checking and reasoning for instances within the O3PO context.

The subsequent illustration (Figure 15) portrays the diverse information transitioned from the initial P&ID to the final O3PO instances, showcasing the potential outcomes achievable by mapping topological correlations and equipment between O3PO and DEXPI.

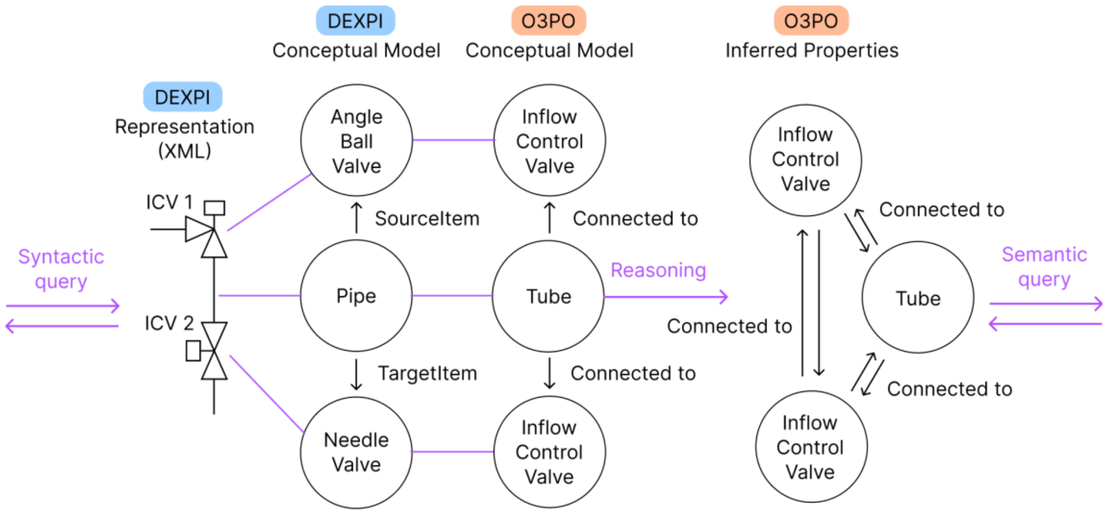


Figure 15 - Enabling domain-specific semantic queries in P&IDs

5.1 Proof of concept application

We this section, we aim to demonstrate the technical viability of this approach. Then we devise a concise proof of concept utilizing a compact plant example depicted within a ProteusXML file. This process involved the conversion of the Proteus XML representation into O3PO instances, subjecting the resulting triples to reasoning procedures.

5.1.1 Application architecture

To address the initial challenge of mapping from the DEXPI file representation (current ProteusXML) to the DEXPI conceptual model, we employed a dedicated parser developed by pnb plants & bytes. This parser abstracts the intricacies of the representation format, mitigating the complexities and potential future alterations.

Following this, we have carried out the conversion from the DEXPI conceptual model contained within the parser application, creating an N-Triples file using the previously outlined mapping rules. This process involved a limited selection of elements derived from the DEXPI equipment and piping packages. Subsequently, the Hermit reasoner was employed to conduct reasoning over the O3PO instances detailed within the N-Triples file. This use of the reasoner provided the potential to generate information and detect inconsistencies due to the reasoning process.

For the concept's viability validation, we develop an application that takes a ProteusXML file as input and produces well-founded instances as output. We applied the parsing of the ProteusXML file utilizing the pnb Toolbox Library offered by pnb plants & bytes. Following this, the file transforms into sets of triples, which we have further translated from the DEXPI model to the O3PO model.

We then created a customized module using the Owlready2 Python library to facilitate reasoning and information querying. This Python library provides many useful tools for building applications with ontologies. It allows for software development in an “ontology-oriented” way, object-oriented programming in which objects and classes are the entities of an ontology, also providing support for reasoning (JEAN-BAPTISTE, 2021). Below, in Figure 16, is an overview of this architecture description.

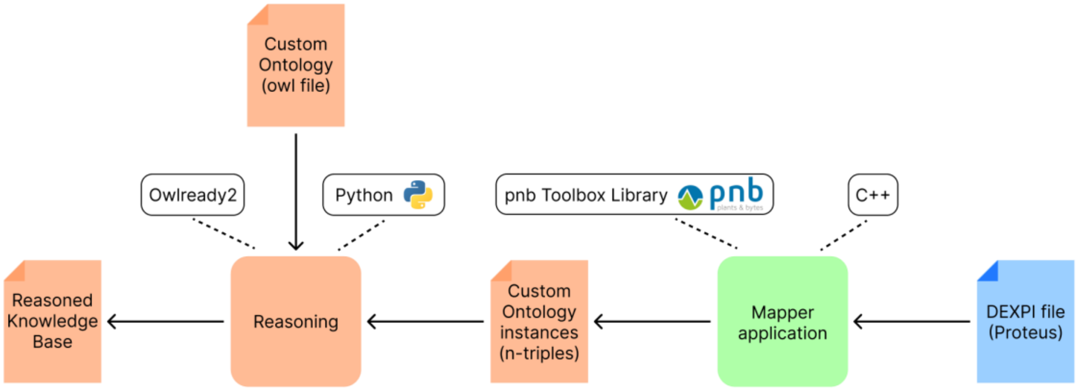
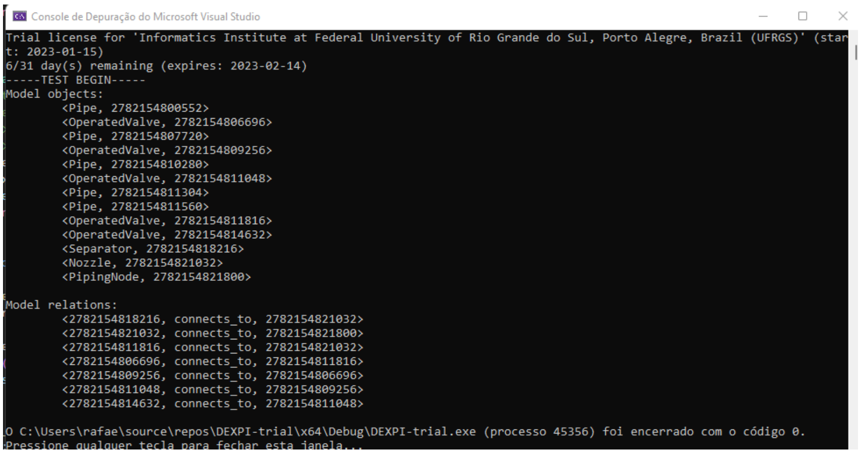


Figure 16 - Architecture for technical interoperability

5.1.2 Application demonstration

Upon receiving a ProteusXML file adhering to the DEXPI specification that delineates a plant featuring components like pipes, valves, separators, and nozzles, the application undertakes the task of mapping these elements to the O3PO model and subsequently infers their inherent attributes. Below, Figure 17 shows a screenshot showcasing that identifyes the elements present in a P&ID.



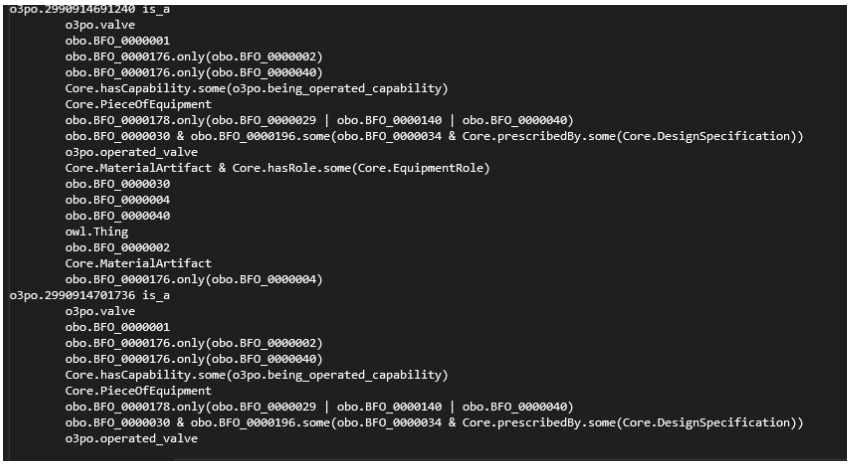
```
Trial license for 'Informatics Institute at Federal University of Rio Grande do Sul, Porto Alegre, Brazil (UFRGS)' (star
t: 2023-01-15)
6/31 day(s) remaining (expires: 2023-02-14)
-----TEST BEGIN-----
Model objects:
        <Pipe, 2782154800552>
        <OperatedValve, 2782154806696>
        <Pipe, 2782154807720>
        <OperatedValve, 2782154809256>
        <Pipe, 2782154810280>
        <OperatedValve, 2782154811048>
        <Pipe, 2782154811304>
        <Pipe, 2782154811560>
        <OperatedValve, 2782154811816>
        <OperatedValve, 2782154814632>
        <Separator, 2782154818216>
        <Nozzle, 2782154821032>
        <PipingNode, 2782154821800>

Model relations:
        <2782154818216, connects_to, 2782154821032>
        <2782154821032, connects_to, 2782154821800>
        <2782154811816, connects_to, 2782154821032>
        <2782154806696, connects_to, 2782154811816>
        <2782154809256, connects_to, 2782154806696>
        <2782154811048, connects_to, 2782154809256>
        <2782154814632, connects_to, 2782154811048>

O C:\Users\rafae\source\repos\DEXPI-trial\x64\Debug\DEXPI-trial.exe (processo 45356) foi encerrado com o código 0.
Pressione qualquer tecla para fechar esta janela...
```

Figure 17 - Identification of various elements from a DEXPI P&ID

Subsequently, following the described procedures, the process of categorizing DEXPI classes into corresponding O3PO classes is executed, followed by reasoning to extract comprehensive information associated with the identified elements. An illustration depicting the classification of two operated valves is provided below in Figure 18.



```
o3po.2990914691240 is_a
        o3po.valve
        obo.BFO_0000001
        obo.BFO_0000176.only(obo.BFO_0000002)
        obo.BFO_0000176.only(obo.BFO_0000040)
        Core.hasCapability.some(o3po.being_operated_capability)
        Core.PieceOfEquipment
        obo.BFO_0000178.only(obo.BFO_0000029 | obo.BFO_0000140 | obo.BFO_0000040)
        obo.BFO_0000030 & obo.BFO_0000196.some(obo.BFO_0000034 & Core.prescribedBy.some(Core.DesignSpecification))
        o3po.operated_valve
        Core.MaterialArtifact & Core.hasRole.some(Core.EquipmentRole)
        obo.BFO_0000030
        obo.BFO_0000004
        obo.BFO_0000040
        owl.Thing
        obo.BFO_0000002
        Core.MaterialArtifact
        obo.BFO_0000176.only(obo.BFO_0000004)
o3po.2990914701736 is_a
        o3po.valve
        obo.BFO_0000001
        obo.BFO_0000176.only(obo.BFO_0000002)
        obo.BFO_0000176.only(obo.BFO_0000040)
        Core.hasCapability.some(o3po.being_operated_capability)
        Core.PieceOfEquipment
        obo.BFO_0000178.only(obo.BFO_0000029 | obo.BFO_0000140 | obo.BFO_0000040)
        obo.BFO_0000030 & obo.BFO_0000196.some(obo.BFO_0000034 & Core.prescribedBy.some(Core.DesignSpecification))
        o3po.operated_valve
```

Figure 18 - O3PO instances generated from the DEXPI P&ID

6 DISCUSSION

Our ontological analysis of the DEXPI P&ID specification to build a relation with a well-founded ontology, such as O3PO, raises a set of desirable properties of a model intended to be used to enhance interoperability. Here, we list and explain the identified properties.

1- Clarity/Intelligibility: Elements in the model should possess clear and transparent definitions, ensuring that they communicate the intended meanings effectively. This property, inspired by definitions from the literature (GRUBER, 1995; NEUHAUS et al., 2013; VRANDEČIĆ, 2009), hinders effects akin to the vague and missing definitions described in the analysis.

2- Conciseness: The model should not have useless definitions, nor should it have explicit redundancies between definitions. Inspired by definitions from various authors (GÓMEZ-PÉREZ, 2001; VRANDEČIĆ, 2009), concise models prevent effects like the redundancy described in the differentiability of equipment part of the analysis.

3- Extendibility: The model should be designed to support anticipation of its uses in different tasks and should also allow for its monotonical specialization. This property, inspired by another definition from the literature (GRUBER, 1995), accounts for similar cases to the collapsed definitions situation presented in the analysis.

4- Consistency: The model should not have contradictory definitions. Based on other authors' definitions (GÓMEZ-PÉREZ, 2001; OBRST LEOAND CEUSTERS, 2007; VRANDEČIĆ, 2009), this property encourages the creation of models that do not present the characteristics outlined when discussing the same definitions in different places in the analysis.

5- Essence: The modeler should define the elements in the model in terms of the essential properties that guarantee their identity. This property, inspired by another definition from the literature (GUARINO; WELTY, 2002), accounts for similar cases to the custom object role situation presented in the analysis.

These outlined properties are among the many others in the literature for assessing and producing higher-quality models. They serve as guidelines to address the challenges in making explicit the meaning of the entities in the DEXPI P&ID model. To address the presented issues, we provide recommendations to enhance the semantic interoperability capabilities of the DEXPI P&ID specification.

1. Avoid vague definitions: definitions need further specification to tackle the problem of vague definitions. It is crucial to make explicit the intended relations between definitions. For instance, the definition of DEXPI:PipingComponent could be refined to read as “Something that is part of a DEXPI:Piping” or “Something that has the capability of being part of a DEXPI:Piping”. This approach offers a clear and comprehensible way to determine whether a given element qualifies as a piping component.
2. Assure completeness in the model scope: We recommend establishing a clear definition for the terms used to define classes. Piping, that is used across many definitions, could be defined as “A composition of one or more valves, sensors, ..., and pipes that are connected to one another maximally”.
3. Make each definition explicit: To confront the challenge of collapsed definitions as observed in DEXPI:SafetyValveOrFitting, we suggest either splitting DEXPI:SafetyValveOrFitting into two distinct types (one for safety valves and another for safety fittings) or creating separate classes for Valve and Fitting. This would help differentiate these elements consistently across all classes while still ensuring compatibility with ISO 10628.
4. Define in terms of essential properties: We cannot create a class of “all things that are not something else” without ignoring the elements essential properties. To resolve the issues related to custom objects, we propose replacing custom classes with a role denoted as “elements whose class is not explicitly defined in the DEXPI P&ID specification”. This modification would facilitate interoperability with systems that lack equivalent classes at the same hierarchical level.
5. Unify common notions: The definition of elements with and without tags, exemplified by DEXPI:ColumnSection, DEXPI:TaggedColumnSection and DEXPI:SubTaggedColumnSection, alongside other elements with tags such as the DEXPI:TaggedPlantItems, are a clear case where the notion of having a tag could be unified. We recommend eliminating such specialization classes. Instead, a tag property could be introduced, allowing physical elements to have a tag.
6. Avoid Overlapping definitions: The issue of overlapping definitions observed in DEXPI:ElectromagneticFlowMeter vs. DEXPI:FlowMeasuringElement, or DEXPI:CoolingTower vs. DEXPI:HeatExchanger, necessitates changes to these

definitions to more explicitly represent their essential properties. Alternatively, some of these classes could be established as parent classes of others, for instance, making DEXPI:FlowMeasuringElement a parent class of DEXPI:ElectromagneticFlowMeter, to better align with intended distinctions.

This practice would lead to better model in terms of semantics, while providing a better foundation for automatic interoperability.

7 CONCLUSION

This study approaches interoperability from multiple perspectives by facilitating technical interoperability between the P&ID representation standard (DEXPI) utilized by diverse software vendors and domain ontologies like O3PO tailored to represent the oil and gas production plant environment. Additionally, the study highlights instances where semantic interoperability between the standard and ontology conceptualizations is feasible and instances where it is not attainable.

While various tools are available to handle standards and technologies, the practical feasibility of achieving technical interoperability between DEXPI P&IDs and ontologies represented in OWL becomes evident. However, the predominant challenges remain within the realm of semantic interoperability. The presence of ambiguous and often vague definitions poses a significant hurdle to seamless systems integration.

As a potential avenue for further contribution, future work could extend this approach of categorizing instances into well-founded ontologies to encompass the entire DEXPI standard by taking advantage of different ontologies to represent different aspects of the standard, thus serving as a valuable guide for future standards development and patching. Additionally, an opportunity exists to employ ontology alignment techniques with the DEXPI P&ID information model and other relevant ontologies, enabling a more extensive coverage of classes.

Although the primary focus of the DEXPI revolves around enhancing interoperability between CAE tools, it is increasingly observed in the literature that various other applications adopt the description as a standard. These applications leverage the available tooling and the standard's publicly available documentation. However, if the DEXPI standard were to evolve into a de facto standard for representing P&IDs across a broader range of domains than initially intended, the highlighted issues could bottleneck its adoption and hinder the ease of integration in different systems.

REFERENCES

ABEL, M.; PERRIN, M.; CARBONERA, J. L. Ontological analysis for information integration in geomodeling. **Earth Science Informatics**, v. 8, n. 1, p. 21–36, 2015.

ARP, R.; SMITH, B.; SPEAR, A. D. **Building Ontologies with Basic Formal Ontology**. [s.l.] The MIT Press, 2015.

BAKER, L. R. **The Metaphysics of Everyday Life: An Essay in Practical Realism**. [s.l.] New York: Cambridge University Press, 2007.

BANDEIRA, J. et al. FOCA: A methodology for ontology evaluation. **arXiv preprint arXiv:1612.03353**, 2016.

BENSON TIM AND GRIEVE, G. Why Interoperability Is Hard. Em: **Principles of Health Interoperability: FHIR, HL7 and SNOMED CT**. Cham: Springer International Publishing, 2021. p. 21–40.

BODENREIDER, O. **Ontologies and Data Integration in Biomedicine: Success Stories and Challenging Issues**. (S. and F. C. Bairoch Amos and Cohen-Boulakia, Ed.)Data Integration in the Life Sciences. **Anais**...Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.

COMMISSION, E.; FOR INFORMATICS, D.-G. **New European interoperability framework : promoting seamless services and data flows for European public administrations**. [s.l.] Publications Office, 2017.

CUI, L.; ZHAO, J.; ZHANG, R. The integration of HAZOP expert system and piping and instrumentation diagrams. **Process Safety and Environmental Protection**, v. 88, n. 5, p. 327–334, 2010.

DEXPI INITIATIVE. DEXPI P&ID Specification. **Version 1.3**, 2021.

DINIZ, A. A. R. et al. **Ontology for industrial petrochemical processes: Case study of a DEA process**. Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation (ETFA 2012). **Anais**...2012.

ENCARNAÇÃO, J. L.; LOCKEMANN, P. C. **Engineering databases: connecting islands of automation through databases**. [s.l.] Springer, 1990.

FILLINGER, S. et al. Improving interoperability of engineering tools–Data exchange in plant design. **Chemie ingenieur technik**, v. 89, n. 11, p. 1454–1463, 2017.

ISO. **ISO 25964-2:2013: Information and Documentation : Thesauri and Interoperability with Other Vocabularies. Interoperability with other vocabularies**. [s.l.] ISO, 2013.

FORD, T. C. et al. A survey on interoperability measurement. **Gateways**, v. 2, p. 3, 2007.

GARCIA, L. F. et al. A Conceptual Framework for Rock Data Integration in Reservoir Models Based on Ontologies. **Int. J. Monit. Surveill. Technol. Res.**, v. 5, n. 1, p. 71–82, jan. 2017.

GARCIA, L. F. et al. The GeoCore ontology: A core ontology for general use in Geology. **Computers & Geosciences**, v. 135, p. 104387, 2020.

GÓMEZ-PÉREZ, A. Evaluation of ontologies. **International Journal of intelligent systems**, v. 16, n. 3, p. 391–409, 2001.

GRUBER, T. R. Toward principles for the design of ontologies used for knowledge sharing? **International Journal of Human-Computer Studies**, v. 43, n. 5, p. 907–928, 1995.

GUARINO, N. **Formal Ontology in Information Systems: Proceedings of the 1st International Conference June 6-8, 1998, Trento, Italy**. 1st. ed. NLD: IOS Press, 1998.

GUARINO, N.; WELTY, C. Evaluating Ontological Decisions with OntoClean. **Commun. ACM**, v. 45, n. 2, p. 61–65, fev. 2002.

GUARINO NICOLA AND OBERLE, D. AND S. S. What Is an Ontology? Em: STAAB STEFFEN AND STUDER, R. (Ed.). **Handbook on Ontologies**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 1–17.

GUIZZARDI, G. Ontology, ontologies and the "I" of FAIR. **Data Intelligence**, v. 2, n. 1–2, p. 181–191, 2020.

GÜRDÜR, D.; ASPLUND, F. A systematic review to merge discourses: Interoperability, integration and cyber-physical systems. **Journal of Industrial Information Integration**, v. 9, p. 14–23, 2018.

HOWARD, H. C. et al. Computer Integration: Reducing Fragmentation in AEC Industry. **Journal of Computing in Civil Engineering**, v. 3, n. 1, p. 18–32, 1989.

IEEE. IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. **IEEE Std 610**, p. 1–217, 1991.

ISO/IEC. **ISO/IEC Guide 2:2004: Standardization and related activities — General vocabulary**. , 2004. Disponível em: <https://www.iso.org/standard/39976.html>. Acesso em: 13 ago. 2023

ISO/TS. ISO/TS 15926-4:2019: Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 4: Initial reference data. 2019.

JEAN-BAPTISTE, L. **Ontologies with Python: Programming OWL 2.0 Ontologies with Python and Owlready2**. [s.l.] Springer, 2021.

KING, M. J. et al. Reservoir Modeling: From RESCUE To RESQML. **SPE Reservoir Evaluation & Engineering**, v. 15, n. 02, p. 127–138, ago. 2012.

KOZIOLEK, H.; RÜCKERT, J.; BERLET, A. **Industrial plant topology models to facilitate automation engineering**. Systems Modelling and Management: First International Conference, ICSMM 2020, Bergen, Norway, June 25–26, 2020, Proceedings 1. **Anais**...2020.

LEHTONEN, T.; OTHERS. Ontology-based diagram methods in process modelling and simulation. **Master's thesis, Helsinki University of Technology**, 2007.

LOURDUSAMY, R.; JOHN, A. **A review on metrics for ontology evaluation**. 2018 2nd International Conference on Inventive Systems and Control (ICISC). **Anais**...2018.

MCDANIEL, M.; STOREY, V. C. Evaluating Domain Ontologies: Clarification, Classification, and Challenges. **ACM Comput. Surv.**, v. 52, n. 4, set. 2019.

NEUHAUS, F. et al. Towards Ontology Evaluation across the Life Cycle: The Ontology Summit 2013. **Appl. Ontol.**, v. 8, n. 3, p. 179–194, jul. 2013.

OBRST, L. Ontological Architectures. Em: POLI ROBERTO AND HEALY, M. AND K. A. (Ed.). **Theory and Applications of Ontology: Computer Applications**. Dordrecht: Springer Netherlands, 2010. p. 27–66.

OBRST LEO AND CEUSTERS, W. AND M. I. AND R. S. AND S. B. The Evaluation of Ontologies. Em: BAKER CHRISTOPHER J. O. AND CHEUNG, K.-H. (Ed.). **Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences**. Boston, MA: Springer US, 2007. p. 139–158.

OEING, J. et al. Using artificial intelligence to support the drawing of piping and instrumentation diagrams using DEXPI standard. **Digital Chemical Engineering**, v. 4, p. 100038, 2022.

OTTE, J. N.; BEVERLEY, J.; RUTTENBERG, A. BFO: Basic Formal Ontology. **Applied ontology**, v. 17, n. 1, 2022.

PALFREY, J.; GASSER, U. **Interop: The Promise and Perils of Highly Interconnected Systems**. 2012. Disponível em: <https://api.semanticscholar.org/CorpusID:56526>

PNB PLANTS & BYTES GMBH. **pnb Toolbox Library**. Disponível em: <https://www.plants-and-bytes.de/en/p-id-and-dexpi>. Acesso em: 10 ago. 2023.

RILEY, J. Understanding metadata. **Washington DC, United States: National Information Standards Organization (http://www. niso. org/publications/press/UnderstandingMetadata. pdf)**, v. 23, p. 7–10, 2017.

SANTOS, N. et al. O3po: A Domain Ontology for Offshore Petroleum Production Plants. **Available at SSRN 4280151**, 2022a.

SANTOS, N. O. et al. Towards an Ontology of Offshore Petroleum Production Equipment. **FOMI 2022: 12th International Workshop on Formal Ontologies meet Industry**, 2022b.

SCHMIDBERGER, T.; DRATH, R.; FAY, A. **A rule ontology as a basis for automatic reasoning on industrial plant information**. 5th Mathmod Vienna. Vienna University of Technology. **Anais**...2006.

SMITH, B. Beyond Concepts: Ontology as Reality Representation. Em: VARZI, A. C.; VIEU, L. (Eds.). **Formal Ontology in Information Systems (FOIS)**. [s.l: s.n.]. p. 1–12.

SMITH, B. et al. A First-Order Logic Formalization of the Industrial Ontology Foundry Signature Using Basic Formal Ontology. Em: **Proceedings of the Joint Ontology Workshops (JOWO), Graz**. [s.l: s.n.].

STEINDL, G.; KASTNER, W. **Ontology-Based Model Identification of Industrial Energy Systems**. 2020 IEEE 29th International Symposium on Industrial Electronics (ISIE). **Anais**...2020.

STEINEGGER, M.; MELIK-MERKUMIANS, M.; SCHITTER, G. **Ontology-based framework for the generation of interlock code with redundancy elimination**. 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). **Anais**...2017.

STUDER, R.; BENJAMINS, V. R.; FENSEL, D. Knowledge engineering: Principles and methods. **Data & Knowledge Engineering**, v. 25, n. 1, p. 161–197, 1998.

SUÁREZ-FIGUEROA, M. C.; GÓMEZ-PÉREZ, A.; FERNÁNDEZ-LÓPEZ, M. The NeOn Methodology framework: A scenario-based methodology for ontology development. **Applied Ontology**, v. 10, p. 107–145, 2015.

TOGHRAEI, M. **Piping and instrumentation diagram development**. [s.l.] John Wiley & Sons, 2019.

TURK, Ž. Interoperability in construction – Mission impossible? **Developments in the Built Environment**, v. 4, p. 100018, 2020.

VRANDEČIĆ, D. Ontology Evaluation. Em: STAAB STEFFEN AND STUDER, R. (Ed.). **Handbook on Ontologies**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 293–313.

WIEDAU, M. et al. ENPRO Data Integration: Extending DEXPI Towards the Asset Life cycle. **Chemie Ingenieur Technik**, v. 91, n. 3, p. 240–255, 2019.

WIESNER, A.; SAXENA, A.; MARQUARDT, W. **An ontology-based environment for effective collaborative and concurrent process engineering**. 2010 IEEE International Conference on Industrial Engineering and Engineering Management. **Anais**...2010.

APPENDIX A

Table 1 – DEXPI to O3PO Classification

Concept name	**DEXPI Definitions**	**Description**	**Observations**
CustomPipingComponent	A custom PipingComponent, i.e., a PipingComponent that is not covered by any of the other subclasses of PipingComponent (CheckValve, InlinePrimaryElement, OperatedValve, PipeFitting, or SafetyValveOrFitting).	N/A	Cannot define the role of a custom piping component. Could be: DEXPI:CustomPipingComponent is_a (DEXPI:CustomObject) and (Core:MaterialComponentRole) and (Core:roleOf some DEXPI:PipingComponent)
InlinePrimaryElement	An inline primary element.	DEXPI:InlinePrimaryElement is_a (Core:MaterialArtifact) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	All subclasses of InlinePrimaryElement are used to take or to facilitate some sort of measurement, but some of them are not directly sensors.
PipingComponent	A piping component	DEXPI:PipingComponent is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	DEXPI:PipingComponent's definition is vague, so we assume, based on its subtypes and the fact that P&IDs represent intended assemblies instead of components regardless of use, that a DEXPI:PipingComponent is Core:MaterialArtifact that constitutes part of a piping
PipingNetworkSegmentItem	An item that can be part of a PipingNetworkSegment.	DEXPI:PipingNetworkSegmentItem is_a (IAO:Identifier) or (IAO:Symbol) or ((BFO:MaterialEntity) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)))	DEXPI:PipingNetworkSegmentItem can be: * DEXPI:PipeOffPageConnector, that is a IAO:Identifier * DEXPI:PropertyBreak, that is a IAO:Symbol * DEXPI:PipingComponent, previously defined
Equipment	An apparatus or machine.	DEXPI:Equipment is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	No reason was found as to why PipingComponents are also not Equipments
Agglomerator	A machine that is capable of agglomerating. It is usually vertically aligned.	DEXPI:Agglomerator is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add agglomerating function to O3PO
Agitator	An Agitator is a dynamic mixer that stirs or shakes fluids by reaction force from moving vanes.	DEXPI:Agitator is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add mixing function to O3PO

Blower	A machine that is capable of blowing a medium volume flow.	DEXPI:Blower is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add blowing function to O3PO
Centrifuge	A 'separator' and 'machine' that uses centrifugal force to separate phases of different densities (from http://data.posccaesar.org/rdl/RDS420974).	DEXPI:Centrifuge is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add fluid separation function and centrifuge principle of operation to O3PO
Compressor	A machine that has the capability of compressing a gas.	DEXPI:Compressor is_a (O3PO:Pump) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add gas compressing function to O3PO
CoolingTower	A cooler and an air cooled heat exchanger that is a tall structure through which air circulates by convection (from http://data.posccaesar.org/rdl/RDS14072341).	DEXPI:CoolingTower is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add thermal exchange function and tower class to O3PO
HeatExchanger	An apparatus or machine that has the capability of heat exchanging (from http://data.15926.org/rdl/RDS304199).	DEXPI:Equipment is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add thermal exchange function to O3PO
Mixer	An apparatus or machine that has the capability of mixing (from http://data.15926.org/rdl/RDS222370).	DEXPI:Mixer is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add mixing function to O3PO
PackagingSystem	A system that is intended for the preparation of goods for transport, warehousing, logistics, sale, and end use (from http://data.15926.org/rdl/RDS2228725).	DEXPI:PackagingSystem is_a (Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add packing function to O3PO
PipeFitting	A pipe fitting.	DEXPI:PipeFitting is_a (Core:MaterialArtifact) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Could add fitting class as subclass of tube in O3PO
SafetyValveOrFitting	A safety valve or fitting.	DEXPI:SafetyValveOrFitting is_a (Core:MaterialArtifact or O3PO:Valve) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Could add pipe fitting class as subclass of tube, could add a pressure ensuring function in O3PO. The DEXPI class collapses both valves and fittings, impossibilitating automatic classification from the data model alone

ElectromagneticFlowMeter	A velocity flow meter that is measuring flow rate of a conductive fluid running through a magnetic field by measuring the charge created when fluid interacting with the field (from http://data.posccaesar.org/rdl/RDS1009664).	DEXPI:ElectromagneticFlowMeter is_a (O3PO:Sensor) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline) and (O3PO:FlowMeter) and (O3PO:measures some O3PO:FlowRate)	When classifying, it loses the information about the principle of operation of the sensor
Burner	A physical object that is intended to release thermal energy by burning a combustible mixture (from http://data.posccaesar.org/rdl/RDS284399)	DEXPI:Burner is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add thermal energy release function to O3PO
Dryer	An object that has the capability of drying (from http://data.15926.org/rdl/RDS1066939451).	DEXPI:Dryer is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add drying function to O3PO
ElectricGenerator	An electric rotating machine that transforms non-electric energy into electric energy (from http://data.posccaesar.org/rdl/RDS415709).	DEXPI:ElectricGenerator is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add energy conversion function and electric energy generation function to O3PO
Extruder	A machine that has the capability of extruding (from http://data.15926.org/rdl/RDS394044551).	DEXPI:Extruder is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add extruding function to O3PO
Fan	An object that is capable of delivering or exhausting volumes of vapour or gas at low differential pressure (from http://data.15926.org/rdl/RDS415169).	DEXPI:Fan is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add fanning function that displaces gas to O3PO
Feeder	A closed fluid transporter that is a gathering line tied into a trunk line (from http://data.15926.org/rdl/RDS300644).	DEXPI:Feeder is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:suppliesFluidTo some O3PO:Pipeline)	Could add feeding function to O3PO
Heater	An apparatus or machine that has the capability of heating	DEXPI:Equipment is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add heating function to O3PO

Mill	A physical object for grinding or pulverizing materials. Also a machine for shaping metal. In general a machine that manufactures by the continuous repetition of some simple action (from http://data.posccaesar.org/rdl/RDS11589220).	DEXPI:Mill is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add milling function to O3PO
MobileTransportSystem	A mobile system that is intended to transport, store or load/unload material.	DEXPI:MobileTransportSystem is_a (Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add a discrete element transportatation function to O3PO
Motor	A driver that is powered by electricity or internal combustion (from http://data.15926.org/rdl/RDS7191198).	DEXPI:Motor is_a (Core:MaterialArtifact) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add kinetic energy generation function to O3PO
ProcessColumn	A vertical vessel intended to enable chemical reactions or physical processes utilising differences in density of fluids and/or forced flow of fluid (from http://data.posccaesar.org/rdl/RDS4316825224).	DEXPI:ProcessColumn is_a (Core:MaterialArtifact) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add vessel class to O3PO
Sieve	A device that removes particles from a fluid when the fluid passes through or separates particles or molecules according to their size.	DEXPI:Sieve is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Coud add particle from fluid separation function to O3PO
StationaryTransportSystem	A transport system that is intended to transport, store or load/unload material and that, as a whole, remains in one place.	DEXPI:StationaryTransportSystem is_a (Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add a discrete element transportatation function to O3PO
Turbine	An object that is a rotary mechanical device that extracts energy from a fluid flow and converts it into useful work (from http://data.15926.org/rdl/RDS313289).	DEXPI:Turbine is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add the function of energy generation from fluid flow to O3PO to better specify the element
Vessel	A container intended for storage and/or processing of fluids or solids.	DEXPI:Vessel is_a (Core:MaterialArtifact) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add a container of vessel subclass of Core:MaterialArtifact to O3PO to better specify the element
WasteGasEmitter	A physical object that is intended to release/emit waste gas from the process.	DEXPI:WasteGasEmitter is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add a gas emitting function to O3PO to better specify the element

Weigher	A functional object that is capable of weighing.	DEXPI:Weighter is_a (O3PO:Sensor) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (Core:hasCapability some Core:MeasurementCapability)	Could add a weighter subclass of O3PO:Sensor to O3PO to better specify the element
CheckValve	A valve that permits fluid to flow in one direction only (from http://data.posccaesar.org/rdl/RDS292229).	DEXPI:CheckValve is_a (O3PO:Valve) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Could add function restricting flow from one direction
OperatedValve	A valve that includes an external means of operation. (E.g. handwheel / lever / actuator.) (from http://data. posccaesar.org/rdl/RDS11141590).	DEXPI:OperatedValve is_a (O3PO:Valve) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline) and (Core:hasCapability some O3PO:BeingOperatedCapability)	Very good match
Separator	A 'device' intended to separate different types of substances (from http://data.posccaesar.org/rdl/RDS2194378711)	DEXPI:Separator is_a (Core:MaterialArtifact or Core:EngineeredSystem) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Could add separation function to O3PO
Pump	A machine that is capable of pumping but may require parts and subsystems for that capability.	DEXPI:Pump is_a (O3PO:Pump) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification)	Not perfect match because DEXPI diferentiates pumps for liquid and compressors for gases
FlowNozzle	A nozzle with a smooth entry and a sharp exit (from http://data.posccaesar.org/rdl/RDS821024).	DEXPI:InlinePrimaryElement is_a (O3PO:Nozzle) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Nozzle construction method cannot be specified
FlowMeasuringElement	A FLOW MEASURING ELEMENT is a MEASURING ELEMENT that is used to measure FLOW RATE.	DEXPI:FlowMeasuringElement is_a (O3PO:Sensor) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline) and (O3PO:FlowMeter) and (O3PO:measures some O3PO:FlowRate)	Very good match
MassFlowMeasuringElement	A MASS FLOW MEASURING ELEMENT is a FLOW MEASURING ELEMENT that is used to measure MASS FLOW RATE.	DEXPI:MassFlowMeasuringElement is_a (O3PO:FlowMeter and O3PO:Sensor) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	It is acctualy both sensors, the Mass sensor and Flow sensor at the same time. What we can say in O3PO is that it is both a sensor (since it does not have mass sensors specialization) and a flow meter

PositiveDisplacementFlowMeter	A flow meter that measures the volumetric flow rate of a liquid or gas by separating the flow stream into known volumes and counting them over time (from http://data.posccaesar.org/rdl/RDS418094).	DEXPI:PositiveDisplacementFlowMeter is_a (O3PO:FlowMeter) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Measuring method cannot be specified
TurbineFlowMeter	A velocity flow meter that uses a multi bladed rotor to measure fluid flow rate in units of volumetric flow through a closed conduit (from http://data.posccaesar.org/rdl/RDS417914).	DEXPI:TurbineFlowMeter is_a (O3PO:FlowMeter) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Measuring method cannot be specified
VariableAreaFlowMeter	A flow meter consisting of a vertical tube with a conically shaped bore which widens to the top in which a solid body (float) is supported by the force exerted by the fluid stream (from http://data.posccaesar.org/rdl/RDS418229).	DEXPI:VariableAreaFlowMeter is_a (O3PO:FlowMeter) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Measuring method cannot be specified
VenturiTube	A 'measuring device' that has a constriction with a relative long passage with a smooth coned entry and exit (from http://data.posccaesar.org/rdl/RDS648044).	DEXPI:VenturiTube is_a (O3PO:FlowMeter and O3PO:Nozzle) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Measuring method cannot be specified
VolumeFlowMeasuringElement	A VOLUME FLOW MEASURING ELEMENT is a FLOW MEASURING ELEMENT that is used to measure VOLUME FLOW RATE.	DEXPI:VariableFlowMeasuringElement is_a (O3PO:FlowMeter) and (BFO:bearer_of some BFO:function) and (Core:prescribedBy some Core:DesignSpecification) and (O3PO:component_of some O3PO:Pipeline)	Measuring method cannot be specified

ATTACHMENT A

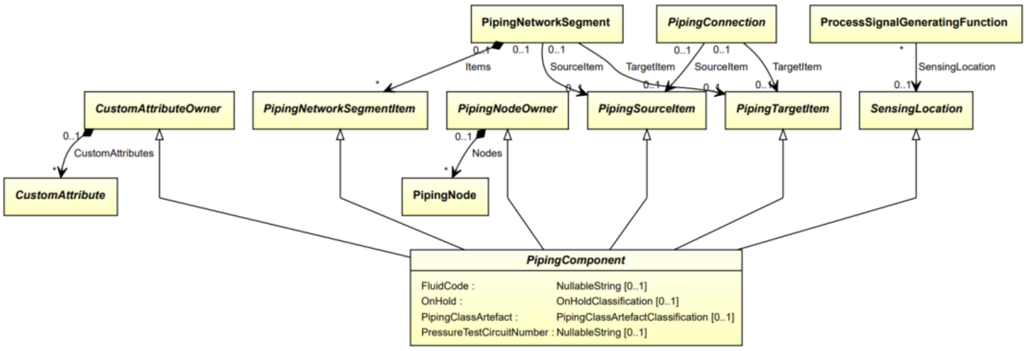
Following content originated from (DEXPI INITIATIVE, 2021)

8.52. PipingComponent

8.52.1 Overview

Abstract class

A piping component



Supertypes

- *CustomAttributeOwner*
- *PipingNetworkSegmentItem*
- *PipingNodeOwner*
- *PipingSourceItem*
- *PipingTargetItem*
- *SensingLocation*

Subtypes

- *CheckValve*
- *CustomPipingComponent*
- *InlinePrimaryElement*
- *OperatedValve*
- *PipeFitting*
- *SafetyValveOrFitting*

Attributes (data)

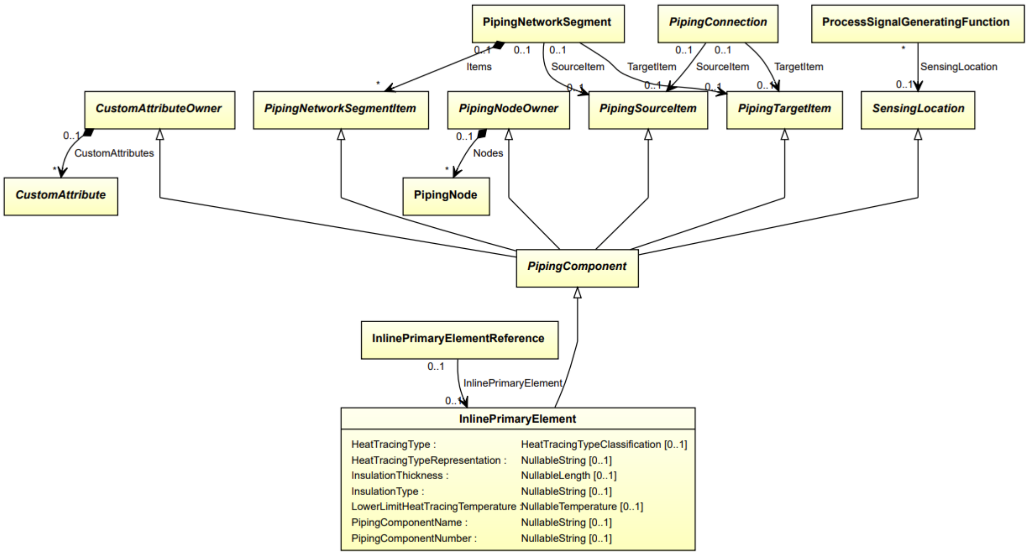
Name	Multiplicity	Type
FluidCode	0..1	*NullableString*
OnHold	0..1	*OnHoldClassification*
PipingClassArtefact	0..1	*PipingClassArtefactClassification*
PressureTestCircuitNumber	0..1	*NullableString*

8.35. InlinePrimaryElement

8.35.1 Overview

Class

An inline primary element.



Supertypes

- *PipingComponent*

Subtypes

- *CustomInlinePrimaryElement*
- *ElectromagneticFlowMeter*
- *FlowMeasuringElement*
- *FlowNozzle*
- *MassFlowMeasuringElement*
- *PositiveDisplacementFlowMeter*
- *TurbineFlowMeter*
- *VariableAreaFlowMeter*
- *VenturiTube*
- *VolumeFlowMeasuringElement*

Attributes (data)

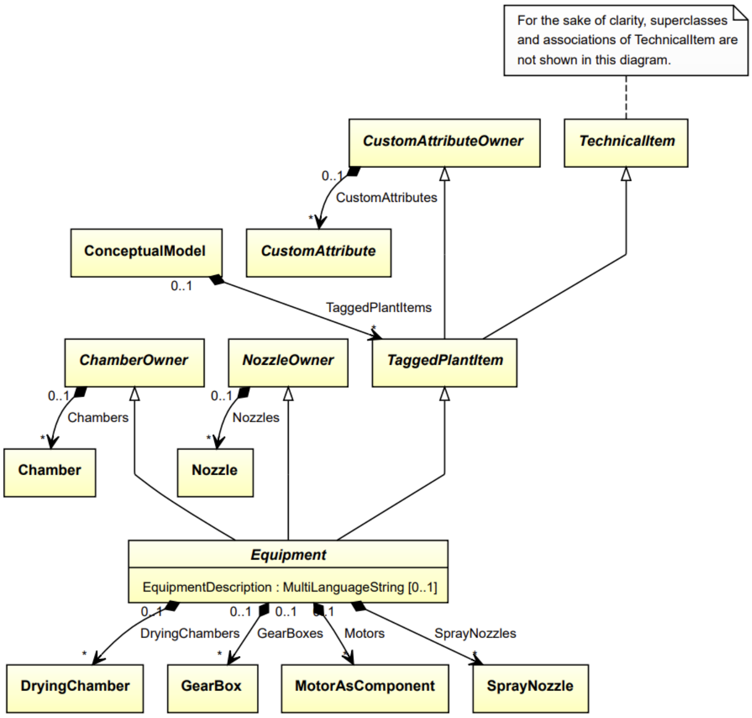
Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

7.74. Equipment

7.74.1 Overview

Abstract class

An apparatus or machine.



Supertypes

- *ChamberOwner*
- *NozzleOwner*
- *TaggedPlantItem*

Subtypes

- *Agglomerator*
- *Agitator*
- *Blower*
- *Burner*
- *Centrifuge*
- *Compressor*
- *CoolingTower*
- *CustomEquipment*
- *Dryer*
- *ElectricGenerator*
- *Extruder*
- *Fan*
- *Feeder*

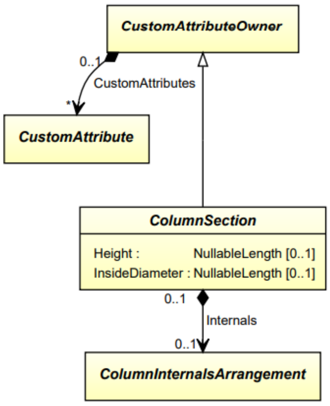
- *Filter*
- *HeatExchanger*
- *Heater*
- *Mill*
- *Mixer*
- *MobileTransportSystem*
- *Motor*
- *PackagingSystem*
- *ProcessColumn*
- *Pump*
- *Separator*
- *Sieve*
- *StationaryTransportSystem*
- *Turbine*
- *Vessel*
- *WasteGasEmitter*
- *Weigher*

7.26. ColumnSection

7.26.1 Overview

Abstract class

A column section.



Supertypes

- *CustomAttributeOwner*

Subtypes

- *SubTaggedColumnSection*
- *TaggedColumnSection*

Attributes (data)

Name	Multiplicity	Type
Height	0..1	*NullableLength*
InsideDiameter	0..1	*NullableLength*

Attributes (composition)

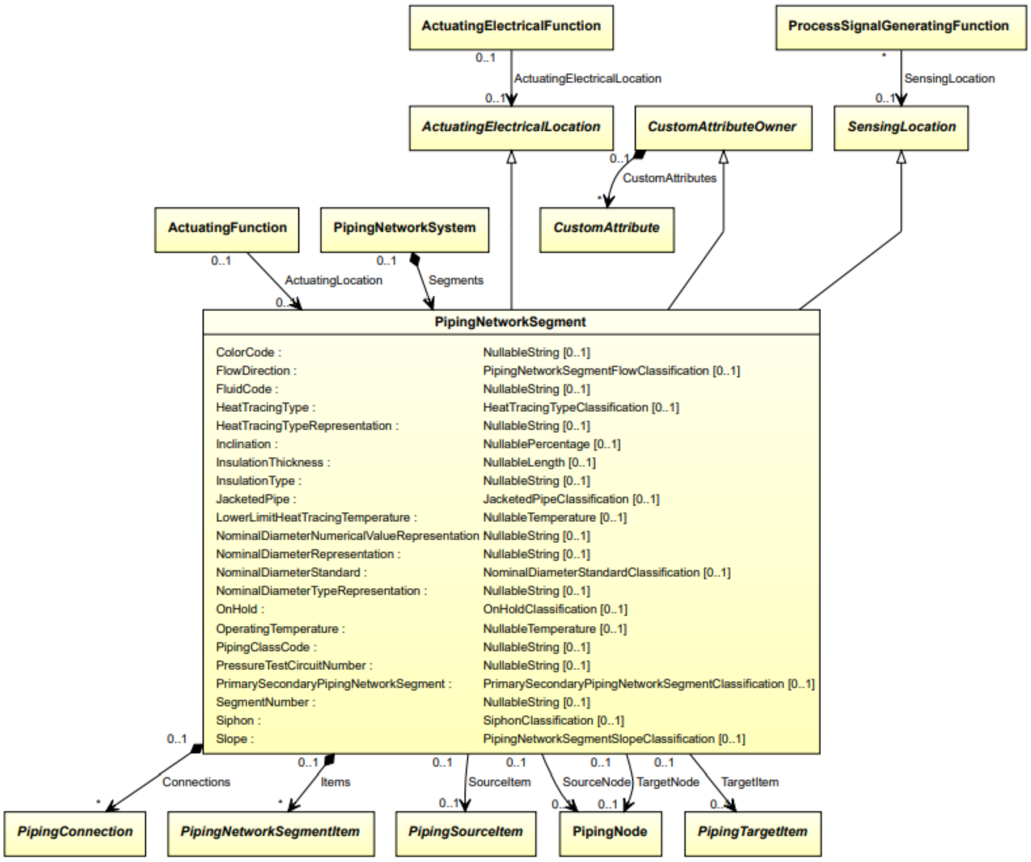
Name	Multiplicity	Type
Internals	0..1	*ColumnInternalsArrangement*

8.54. PipingNetworkSegment

8.54.1 Overview

Class

The piping limited by a Node and a Break, Node and Connector, two Nodes, two Breaks, two Connectors or a Break and a Connector. The last five providing there are no Breaks or Connectors in between. In the last three cases the Segment will coincide with a Piping Branch (from http://data.posccaesar.org/rdl/RDS267704).



Supertypes

- *ActuatingElectricalLocation*
- *CustomAttributeOwner*
- *SensingLocation*

Attributes (data)

Name	Multiplicity	Type
ColorCode	0..1	*NullableString*
FlowDirection	0..1	*PipingNetworkSegmentFlowClassification*
FluidCode	0..1	*NullableString*
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
Inclination	0..1	*NullablePercentage*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*

(continued on next page)

Name	Multiplicity	Type
JacketedPipe	0..1	*JacketedPipeClassification*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
NominalDiameterNumericalValueRepresentation	0..1	*NullableString*
NominalDiameterRepresentation	0..1	*NullableString*
NominalDiameterStandard	0..1	*NominalDiameterStandardClassification*
NominalDiameterTypeRepresentation	0..1	*NullableString*
OnHold	0..1	*OnHoldClassification*
OperatingTemperature	0..1	*NullableTemperature*
PipingClassCode	0..1	*NullableString*
PressureTestCircuitNumber	0..1	*NullableString*
PrimarySecondaryPipingNetworkSegment	0..1	*PrimarySecondaryPipingNetworkSegmentClassification*
SegmentNumber	0..1	*NullableString*
Siphon	0..1	*SiphonClassification*
Slope	0..1	*PipingNetworkSegmentSlopeClassification*

Attributes (composition)

Name	Multiplicity	Type
Connections	*	*PipingConnection*
Items	*	*PipingNetworkSegmentItem*

Attributes (reference)

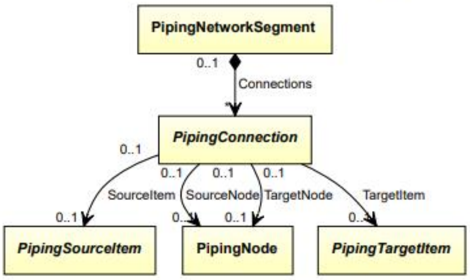
Name	Multiplicity	Type
SourceItem	0..1	*PipingSourceItem*
SourceNode	0..1	*PipingNode*
TargetItem	0..1	*PipingTargetItem*
TargetNode	0..1	*PipingNode*

8.53. PipingConnection

8.53.1 Overview

Abstract class

An elementary connection between two piping items.



Subtypes

- *DirectPipingConnection*
- *Pipe*

Attributes (reference)

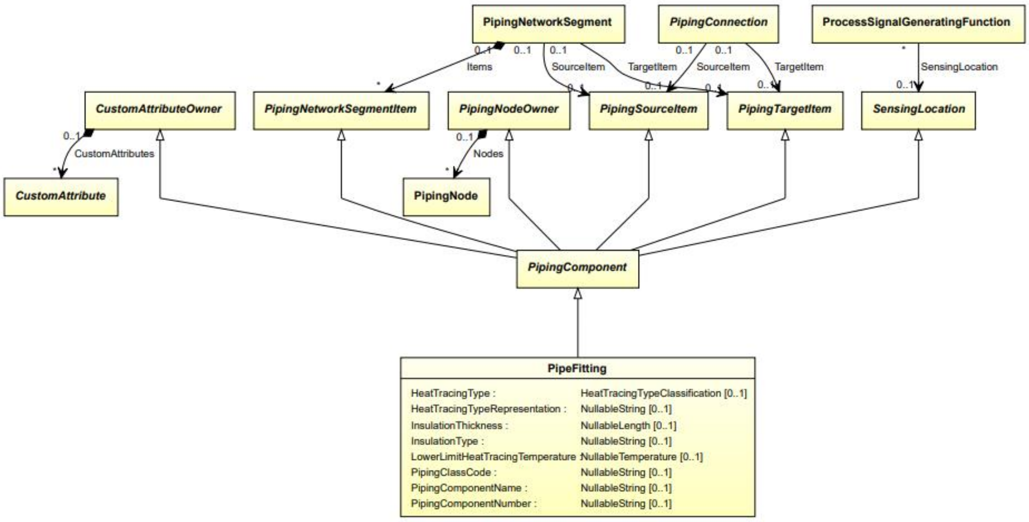
Name	Multiplicity	Type
SourceItem	0..1	*PipingSourceItem*
SourceNode	0..1	*PipingNode*
TargetItem	0..1	*PipingTargetItem*
TargetNode	0..1	*PipingNode*

8.43. PipeFitting

8.43.1 Overview

Class

A pipe fitting.



Supertypes

- *PipingComponent*

Subtypes

- *BlindFlange*
- *ClampedFlangeCoupling*
- *Compensator*
- *ConicalStrainer*
- *CustomPipeFitting*
- *Flange*
- *FlangedConnection*
- *Funnel*
- *Hose*
- *IlluminatedSightGlass*
- *InLineMixer*
- *LineBlind*
- *Penetration*
- *PipeCoupling*
- *PipeFlangeSpacer*
- *PipeFlangeSpade*
- *PipeReducer*
- *PipeTee*
- *RestrictionOrifice*
- *SightGlass*
- *Silencer*
- *SteamTrap*
- *Strainer*
- *VentilationDevice*

Attributes (data)

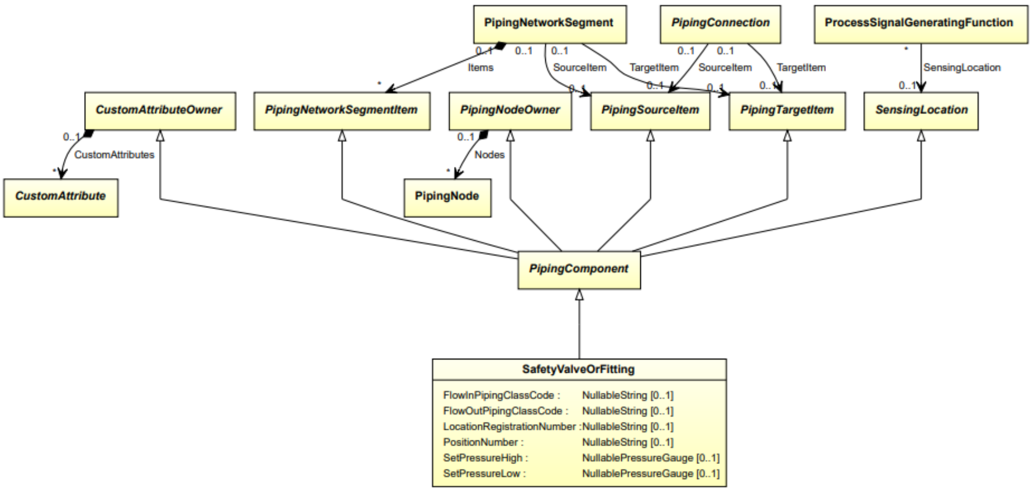
Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
PipingClassCode	0..1	*NullableString*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

8.66. SafetyValveOrFitting

8.66.1 Overview

Class

A safety valve or fitting.



Supertypes

- *PipingComponent*

Subtypes

- *BreatherValve*
- *CustomSafetyValveOrFitting*
- *FlameArrestor*
- *RuptureDisc*
- *SpringLoadedAngleGlobeSafetyValve*
- *SpringLoadedGlobeSafetyValve*

Attributes (data)

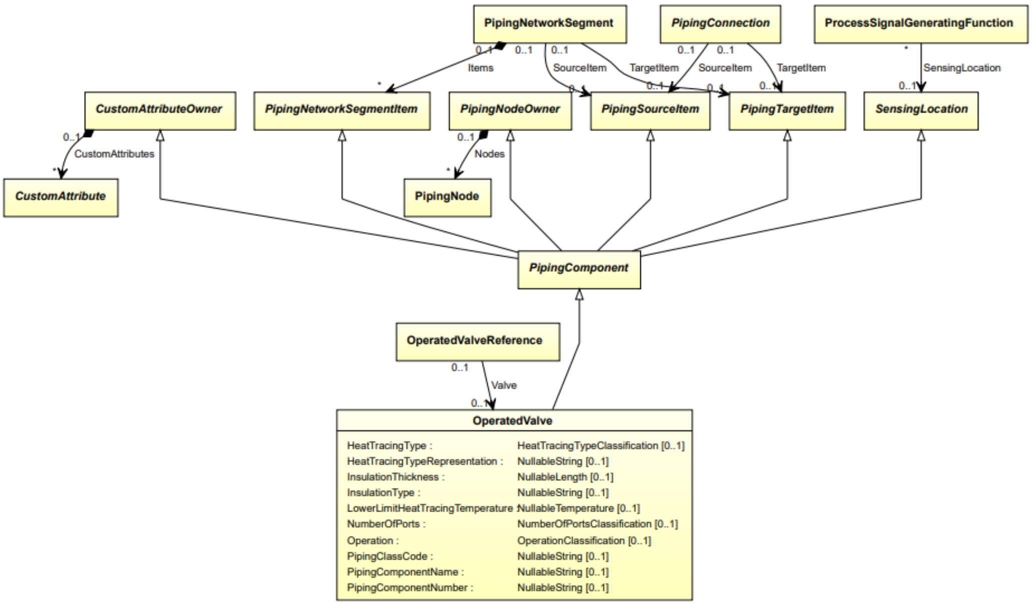
Name	Multiplicity	Type
FlowInPipingClassCode	0..1	*NullableString*
FlowOutPipingClassCode	0..1	*NullableString*
LocationRegistrationNumber	0..1	*NullableString*
PositionNumber	0..1	*NullableString*
SetPressureHigh	0..1	*NullablePressureGauge*
SetPressureLow	0..1	*NullablePressureGauge*

8.39. OperatedValve

8.39.1 Overview

Class

A valve that includes an external means of operation. (E.g. handwheel / lever / actuator.) (from http://data.posccaesar.org/rdl/RDS11141590).



Supertypes

- *PipingComponent*

Subtypes

- *AngleBallValve*
- *AngleGlobeValve*
- *AnglePlugValve*
- *AngleValve*
- *BallValve*
- *ButterflyValve*
- *CustomOperatedValve*
- *GateValve*
- *GlobeValve*
- *NeedleValve*
- *PlugValve*
- *StraightwayValve*

Attributes (data)

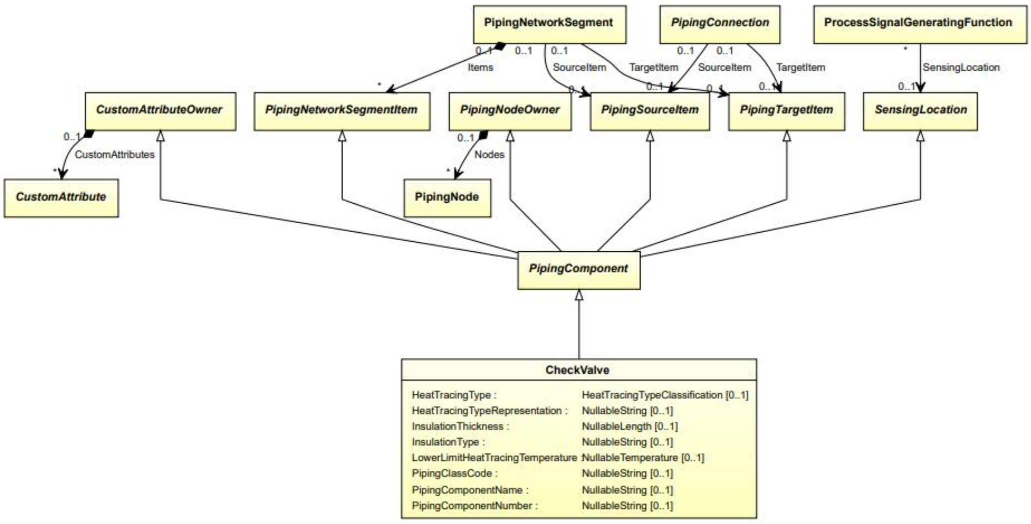
Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
NumberOfPorts	0..1	*NumberOfPortsClassification*
Operation	0..1	*OperationClassification*
PipingClassCode	0..1	*NullableString*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

8.9. CheckValve

8.9.1 Overview

Class

A valve that permits fluid to flow in one direction only (from http://data.posccaesar.org/rdl/RDS292229).



Supertypes

- *PipingComponent*

Subtypes

- *CustomCheckValve*
- *GlobeCheckValve*
- *SwingCheckValve*

Attributes (data)

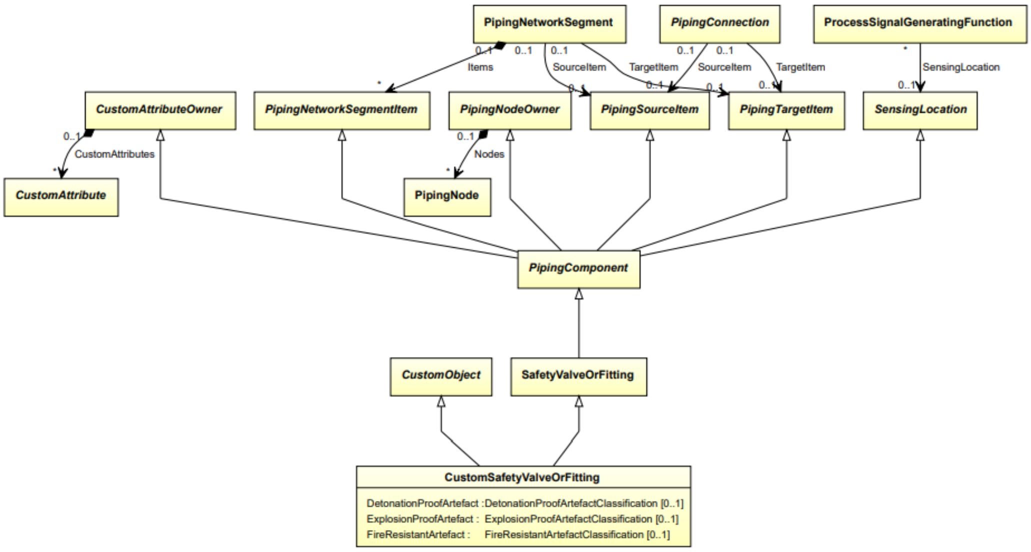
Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
PipingClassCode	0..1	*NullableString*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

8.18. CustomSafetyValveOrFitting

8.18.1 Overview

Class

A custom *SafetyValveOrFitting*, i.e., a *SafetyValveOrFitting* that is not covered by any of the other subclasses of *SafetyValveOrFitting* (*BreatherValve*, *FlameArrestor*, *RuptureDisc*, *SpringLoadedAngleGlobeSafetyValve*, or *SpringLoadedGlobeSafetyValve*).



Supertypes

- *CustomObject*
- *SafetyValveOrFitting*

Attributes (data)

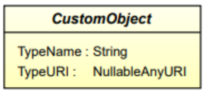
Name	Multiplicity	Type
DetonationProofArtefact	0..1	*DetonationProofArtefactClassification*
ExplosionProofArtefact	0..1	*ExplosionProofArtefactClassification*
FireResistantArtefact	0..1	*FireResistantArtefactClassification*

10.13. CustomObject

10.13.1 Overview

Abstract class

The abstract base class of all custom classes.



Subtypes

- *CustomActuatingElectricalSystemComponent*
- *CustomActuatingSystemComponent*
- *CustomAgglomerator*
- *CustomBlower*
- *CustomCentrifuge*
- *CustomCheckValve*
- *CustomCompressor*
- *CustomCoolingTower*
- *CustomDryer*
- *CustomElectricGenerator*
- *CustomEquipment*
- *CustomExtruder*
- *CustomFan*
- *CustomFilter*
- *CustomHeatExchanger*
- *CustomHeater*
- *CustomInlinePrimaryElement*
- *CustomMill*
- *CustomMixer*
- *CustomMobileTransportSystem*
- *CustomMotor*
- *CustomOperatedValve*
- *CustomPipeFitting*
- *CustomPipingComponent*
- *CustomProcessSignalGeneratingSystemComponent*
- *CustomPump*
- *CustomSafetyValveOrFitting*
- *CustomSeparator*
- *CustomSieve*
- *CustomStationaryTransportSystem*
- *CustomTurbine*
- *CustomVessel*
- *CustomWasteGasEmitter*
- *CustomWeigher*

Attributes (data)

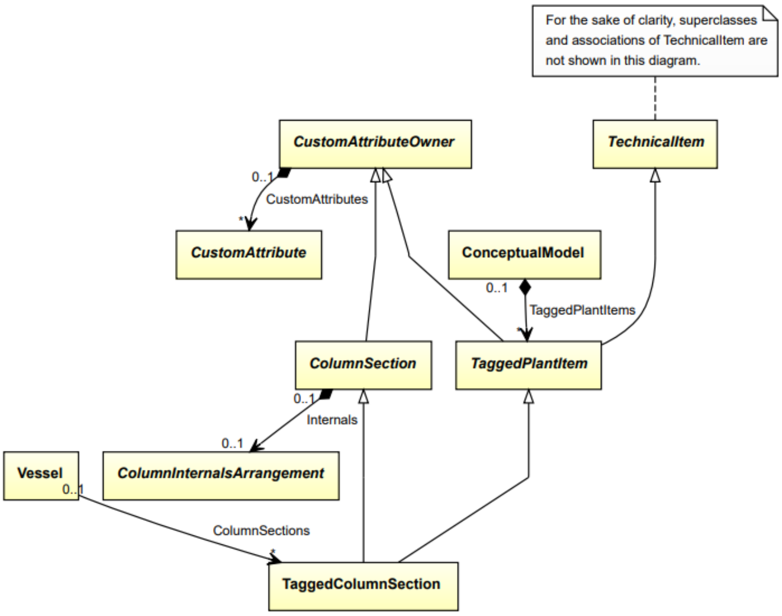
Name	Multiplicity	Type
TypeName	1	*String*
TypeURI	1	*NullableAnyURI*

7.146. TaggedColumnSection

7.146.1 Overview

Class

A fully tagged column section.



Supertypes

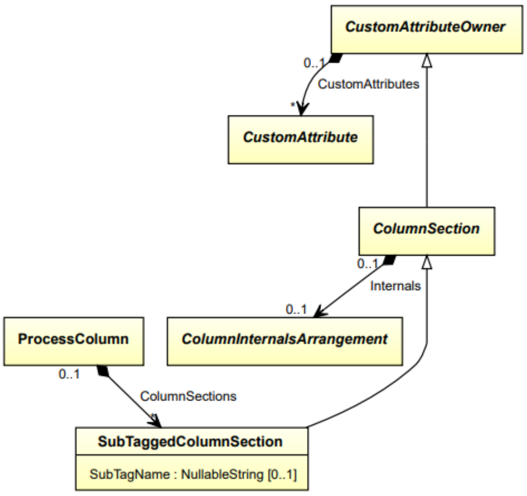
- *ColumnSection*
- *TaggedPlantItem*

7.145. SubTaggedColumnSection

7.145.1 Overview

Class

A sub tagged column section.



Supertypes

- *ColumnSection*

Attributes (data)

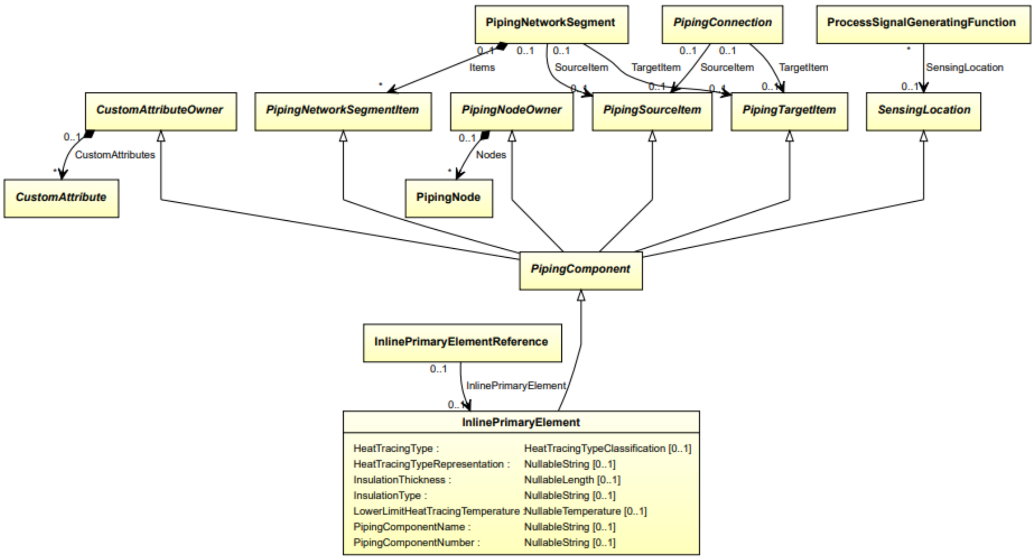
Name	Multiplicity	Type
SubTagName	0..1	*NullableString*

8.35. InlinePrimaryElement

8.35.1 Overview

Class

An inline primary element.



Supertypes

- *PipingComponent*

Subtypes

- *CustomInlinePrimaryElement*
- *ElectromagneticFlowMeter*
- *FlowMeasuringElement*
- *FlowNozzle*
- *MassFlowMeasuringElement*
- *PositiveDisplacementFlowMeter*
- *TurbineFlowMeter*
- *VariableAreaFlowMeter*
- *VenturiTube*
- *VolumeFlowMeasuringElement*

Attributes (data)

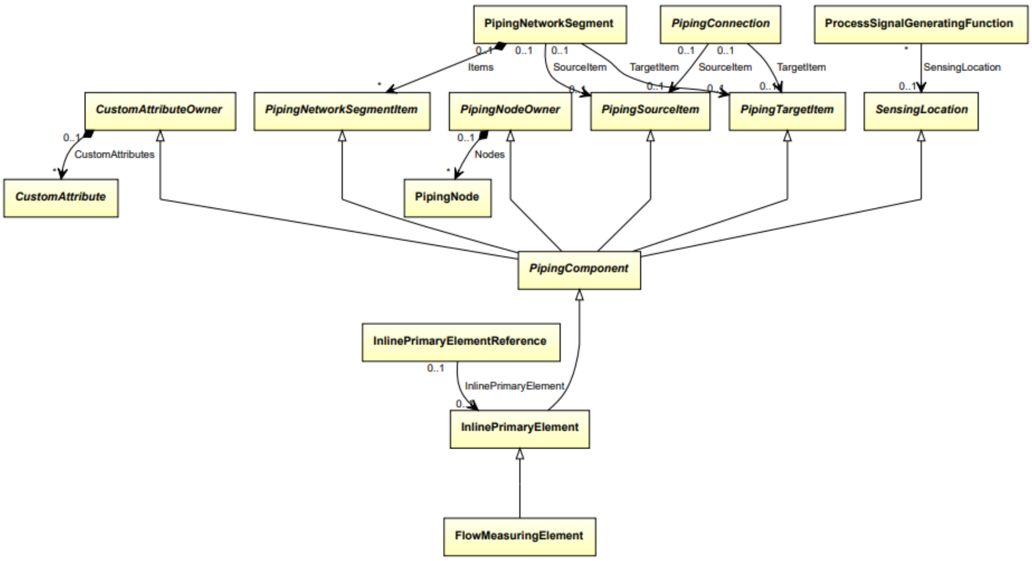
Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

8.25. FlowMeasuringElement

8.25.1 Overview

Class

A FLOW MEASURING ELEMENT is a MEASURING ELEMENT that is used to measure FLOW RATE.



Supertypes

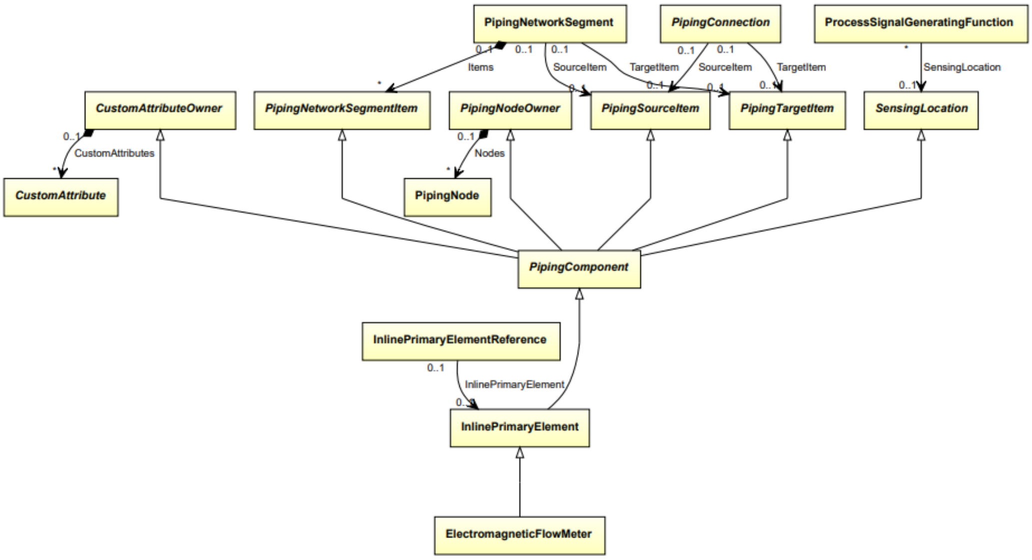
- *InlinePrimaryElement*

8.20. ElectromagneticFlowMeter

8.20.1 Overview

Class

A velocity flow meter that is measuring flow rate of a conductive fluid running through a magnetic field by measuring the charge created when fluid interacting with the field (from http://data.posccaesar.org/rdl/RDS1009664).



Supertypes

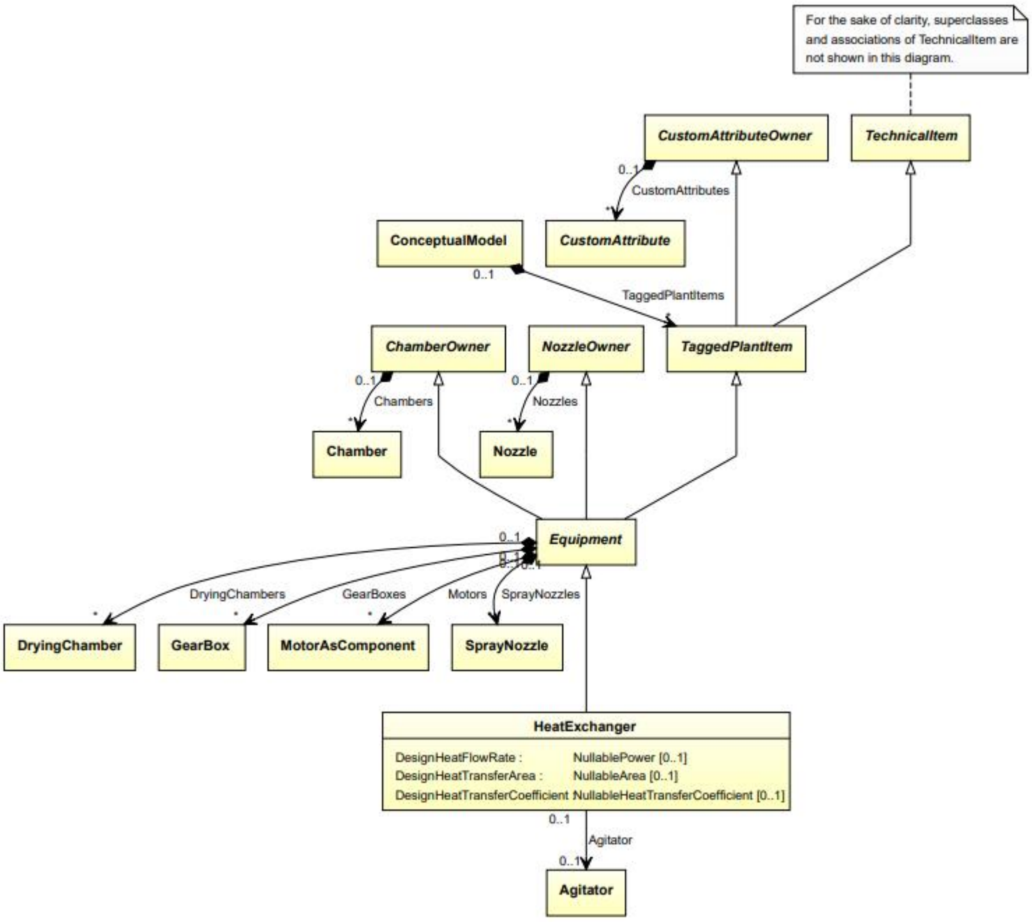
- *InlinePrimaryElement*

7.91. HeatExchanger

7.91.1 Overview

Class

An apparatus or machine that has the capability of heat exchanging (from http://data.15926.org/rdl/RDS304199).



Supertypes

- *Equipment*

Subtypes

- *AirCoolingSystem*
- *CustomHeatExchanger*
- *ElectricHeater*
- *PlateHeatExchanger*
- *SpiralHeatExchanger*
- *ThinFilmEvaporator*
- *TubularHeatExchanger*

Attributes (data)

Name	Multiplicity	Type
DesignHeatFlowRate	0..1	*NullablePower*
DesignHeatTransferArea	0..1	*NullableArea*
DesignHeatTransferCoefficient	0..1	*NullableHeatTransferCoefficient*

Attributes (reference)

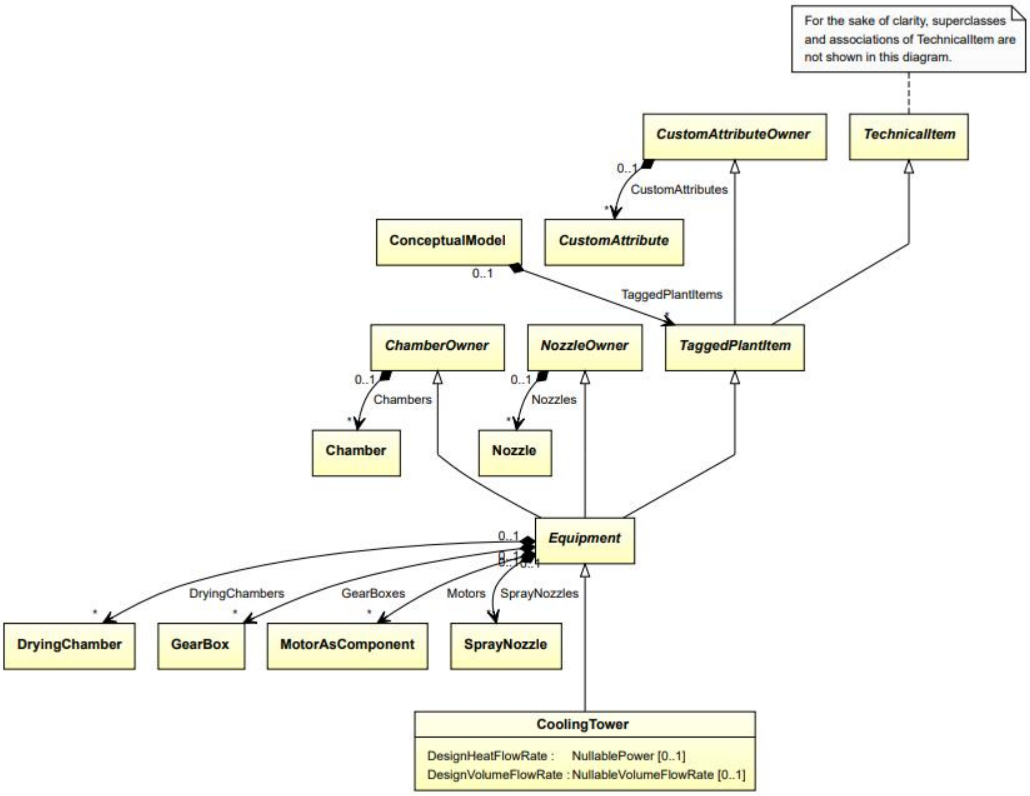
Name	Multiplicity	Type
Agitator	0..1	*Agitator*

7.34. CoolingTower

7.34.1 Overview

Class

A cooler and an air cooled heat exchanger that is a tall structure through which air circulates by convection (from http://data.posccaesar.org/rdl/RDS14072341).



Supertypes

- *Equipment*

Subtypes

- *CustomCoolingTower*
- *DryCoolingTower*
- *SprayCooler*
- *WetCoolingTower*

Attributes (data)

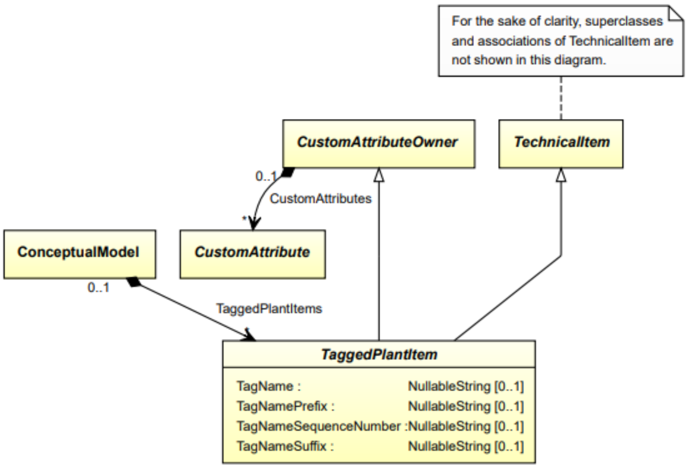
Name	Multiplicity	Type
DesignHeatFlowRate	0..1	*NullablePower*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

7.147. TaggedPlantItem

7.147.1 Overview

Abstract class

A fully tagged item in a plant.



Supertypes

- *CustomAttributeOwner*
- *TechnicalItem*

Subtypes

- *Equipment*
- *TaggedColumnSection*

Attributes (data)

Name	Multiplicity	Type
TagName	0..1	*NullableString*
TagNamePrefix	0..1	*NullableString*
TagNameSequenceNumber	0..1	*NullableString*
TagNameSuffix	0..1	*NullableString*