

## ISO/WD 18629-1

### Process specification language: Overview and basic principles

#### COPYRIGHT NOTICE:

This ISO document is a committee draft and is copyright protected by ISO. While the reproduction of committee drafts in any form for the use by Participants in the ISO standards development process is permitted without prior permission from ISO, neither this document nor any extract from it may be reproduced, stored or transmitted in any form for any other purpose without prior written permission from ISO.

Request for permission to reproduce this document for the purposes of selling it should be addressed as shown below (via the ISO TC 184/SC4 Secretariat's member body) or to ISO's member body in the country of requester.

Copyright Manager  
ANSI  
11 West 42<sup>nd</sup> Street  
New York, New York 10036  
USA  
phone: +1-212-642-4900  
fax: +1-212-398-0023

Reproduction for sales purposes may be subject to royalty payments or a licensing agreement  
Violators may be prosecuted.

#### ABSTRACT:

This document provides a general overview of the different series of parts of the ISO 18629 standard, which defines a process specification language aimed at identifying, formally defining and structuring the semantic concepts intrinsic to the capture and exchange of process information related to discrete manufacturing.

#### KEYWORDS:

**Manufacturing, process information, specification language, ISO 15531, manufacturing process, process language**

#### COMMENTS TO READER:

**Project Leader: M Gruninger**

**Address:**

**National Institute of Science and Technology  
Gaithersburg MD 20879  
Telephone: +1 301-975-6536  
Telefacsimile: +1 301-258-9749  
Electronic mail: gruning@cme.nist.gov**

**Project Editor: AF Cutting-Decelle**

**Address :**

**Université de Savoie / ESIGEC  
Campus scientifique Savoie Technolac  
73376 Le Bourget du Lac Cedex, France  
Telephone: + 33 4 79 75 88 16  
Telefacsimile: +33 4 79 75 81 12  
Electronic mail: afcd@univ-savoie.fr**

# Contents

page

## Foreword

- 1 Scope .....
  - 1.1 Scope of ISO 18629 .....
  - 1.2 Scope of ISO 18629-1 .....
- 2 Normative references .....
- 3 Terms, definitions and abbreviations .....
  - 3.1 Terms defined in ISO 10303-1.....
  - 3.2 Terms defined in ISO 15531-1 .....
  - 3.3 Other terms and definitions .....
  - 3.4 Abbreviations .....
- 4 Overview of ISO 18629 .....
  - 4.1 Concepts provided .....
  - 4.2 The process specification language
  - 4.3 Organisation of the ISO 18629 standard
  - 4.4 Parts 18629-1x series (Core theories) .....
  - 4.5 Parts 18629-2x series (External mappings).....
  - 4.6 Parts 18629-4x series (Extensions)
  - 4.7 Parts 18629-2xx series (Implementation guidelines).....
- 5 Conformance testing methodology and framework
  - 5.1 Overview
  - 5.2 Conformance of ontologies with ISO 18629
  - 5.3 Conformance of applications with ISO 18629

Annex A (Background to the Development of PSL-Core)  
Annex B (The Need for Semantics)  
Annex C (Interoperability)  
Annex D (Logical Foundations)  
Annex E (Semantic Architecture)

## Bibliography

## Index

## Foreword

ISO (the International Organisation for Standardisation) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organisations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardisation.

Draft International Standards adopted by technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75% of the member bodies casting a vote.

ISO 18629 consists of the following parts under the general title *Industrial automation systems and integration – Process specification language*:

- Part 1, Process Specification Language : Overview and basic principles(this document);
- Part 1x series addresses the foundational theories;
- Part 2x series addresses the external mappings onto representation methods or languages;
- Part 4x series addresses the representation of the extensions of the language ;
- Part 2xx series specifies implementation protocols.

This document is part 1 of ISO 18629. Part 1 was jointly prepared by Sub-Committee 4 *Industrial data*, and Sub-Committee 5 *Architecture, Communications, and Integration Frameworks* of Technical committee ISO/TC184 *Industrial automation systems and integration*.

Annexes A to E are informative.

## Introduction

As the use of information technology in manufacturing has matured, the necessity for software applications to inter-operate has become crucial to the conduct of business and operations in organizations. To be competitive and maintain good economic performance, manufacturing organizations need to employ increasingly effective and efficient systems. Such systems should result in the seamless integration of manufacturing applications and exchange of manufacturing processes between applications. Organizations should also be able to conserve and retrieve on demand the knowledge contained in their business and operational processes, regardless of the applications used to produce and handle these processes.

Many manufacturing engineering and business software applications use process information, including manufacturing simulation, production scheduling, manufacturing process planning, workflow, business process reengineering, product realization process modeling, and project management. However, each of these applications utilizes process information in a different way, and each representation of process information inherent to these applications is also different. Thus interoperability is difficult to achieve. Consequently, these concerns have led to the development of a process specification language that complements the process representations utilized in manufacturing engineering and business software applications. The International Standard ISO 18629 provides a generic language for process specifications applicable to a broad range of specific process representations in manufacturing applications.

ISO 18629 is an International Standard for the computer-interpretable exchange of information related to manufacturing processes. Taken together, all the parts contained in the ISO 18629 Standard provide a generic language for describing a manufacturing process throughout the entire production process within the same industrial company or across several industrial sectors or companies, independently from any particular representation model. The nature of this language makes it suitable for sharing process information related to manufacturing during all the stages of a production process.

The process representations used by engineering and business software applications are influenced by the specific needs and objectives of the applications. Therefore, the use of the process specification language also varies from one application to another. A major purpose of the Process Specification Language is to enable the interoperability of manufacturing processes between software applications that utilize different process models and process representations. As a result of implementing process interoperability, economies of scale are made in the integration of manufacturing applications.

This part of ISO 18629 and all other parts in ISO 18629 are independent of any specific process representation or model used in a given application. Collectively, they provide a structural framework for interoperability.

The International Standard ISO 18629 describes what elements inter-operable systems should encompass, but not how a specific application implements these elements. It is not the purpose of the International Standard ISO 18629 to enforce uniformity in manufacturing process representations. Objectives and design of software applications vary. Therefore the implementation of an inter-operable application must necessarily be influenced by the particular objectives and processes of each specific application. This part of ISO 18629 provides an overview and the basic principles for the ISO 18629 International Standard. This part provides an overview of the principal concepts contained in the International Standard, and guidance on selection and use of the parts contained in the ISO 18629 International Standard.





# Industrial automation systems and integration — Process specification language — Part 1: Overview and basic principles

## 1 Scope 1.1 Scope of ISO 18629

ISO 18629 specifies a language for the representation of process information, limited to the realm of discrete processes related to manufacturing, including all processes in the design and manufacturing life cycle. Business processes and manufacturing engineering processes are included in this work both to ascertain common aspects for process specification and to acknowledge the current and future integration of business and engineering functions.

The goal of this standard is to create a process specification language, not a process characterisation language. The process specification language is composed of a lexicon, an ontology, and a grammar for process descriptions, as further described in 4.2.

NOTE 1- A process specification language is a language needed to specify a process or a flow of processes, including supporting parameters and settings. This may be done for prescriptive or descriptive purposes. This is different from a process characterisation language (or process modelling language), which can be defined as a language describing the behaviours and capabilities of a process.

NOTE 2 – Examples of concepts within a process specification language include activities, activity-occurrences, timepoints, and process-plans.

NOTE 3 – Examples of concepts within a process characterization language include the dynamic or kinematic properties of a process, independently of specific processes, such as: tool chatter, a numerical model capturing the dynamic behavior of a process, limits on the process's performance or applicability.

The following are within the scope of ISO 18629:

- the representation of process information related to discrete manufacturing processes;
- the exchange and sharing of process information within one industrial sector or through several industrial sectors.

The following are outside the scope of ISO 18629:

- language, architecture and methodologies for the representation of an enterprise as a whole;
- representation and exchange of product information ;
- representation and exchange of computer-interpretable parts library information;

— technical maintenance information.

EXAMPLE - technical information included in devices repair, operation and maintenance manuals.

## 1.2 Scope of ISO 18629-1

Part 1 of ISO 18629 gives an overview of ISO 18629 and of the main underlying principles of the Process Specification Language . It specifies the characteristics of the various series of parts in ISO 18629 and the relationships among them.

The following are within the scope of Part 1 of ISO 18629:

- general overview of the standard and of the main principles used;
- structure of the standard and relationships between the series of parts of which this standard is composed;
- definitions of terms used throughout this International Standard ISO18629
- 
- conformance criteria for process-related applications
- conformance criteria for other ontologies
- 
- conformance criteria for parts of the standard

The scope of this part includes providing explanations addressing the following items.

Annex A: Background to the development of the standard.

- Annex B: Need for semantics
- Annex C: Interoperability
- Annex D: Logical foundations
- Annex E: Semantic architecture

The scope of each of the other parts of ISO 18629 is defined within the relevant parts of each series.

## 2 Normative references

The following standards contain provisions that, through reference in this text, constitute provisions for this part of ISO 18629. At the time of publication the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO 18629 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

- ISO 10303-1: 1994, *Industrial automation systems and integration - Product data representation and exchange - Part 1: Overview and fundamental principles.*



— ISO 10303-11: 1994, *Industrial automation systems and integration - Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference manual.*

— ISO/IEC 8824-1:1995, *Information technology - Open systems interconnection - Abstract syntax notation one (ASN.1) - Part 1: Specification of basic notation.*

— ISO 13584-1: 1997, *Industrial automation systems and integration - Parts library - Part 1: Overview and fundamental principles.*

— ISO 15531-1: 2000, *Industrial automation systems and integration – Industrial manufacturing management data - Part 1: General overview.*

— ISO 14258: 1998, *Industrial automation systems and integration - Concepts and rules for enterprise models.*

— ISO 8879: 1986, *Information processing - Text and office systems - Standard Generalised Markup Language (SGML)*

**Placeholder: ISO/IEC JTC1 SC32 (KIF, the Knowledge Interchange Format) reference will move here when it is agreed upon as a standard, but not yet.**

### **3 Terms, definitions, and abbreviations**

#### **3.1 Terms defined in ISO 10303-1**

This part of ISO 18629 makes use of the following terms defined in ISO 10303-1:

- data;
- exchange structure;
- generic resource;
- information;
- product;
- product data;
- product information.

#### **3.2 Terms defined in ISO 15531-1**

This part of ISO 18629 makes use of the following terms defined in ISO 15531-1:

- continuous process;

## ISO/WD 18629-1:2000(E)

- discrete manufacturing;
- industrial process;
- manufacturing;
- manufacturing process;
- process;
- process planning;
- production facilities; manufacturing facilities;
- resource;
- scheduling;

### 3.3 Other terms and definitions

For the purpose of this part of ISO 18629, the following definitions apply:

#### 3.3.1 axiom

well-formed formula in a formal language that provides constraints on the interpretation of symbols in the lexicon of a language

#### 3.3.2 axiomatisation

set of axioms in a theory

#### biconditional sentence

KIF sentence of the form  $(\Leftrightarrow P Q)$  for well-formed formulae P and Q

#### 3.3.3 conservative definition

definition that specifies necessary and sufficient conditions that a term must satisfy and that does not allow new inferences to be drawn from the theory

#### 3.3.4 constant

symbol that denotes an element in the domain of a model

### 3.3.5

#### PSL Core

theory that axiomatizes the concepts of activity, activity-occurrence, timepoint, and object such that any two process-related applications must share these axioms in order to exchange process information

NOTE - The core axioms provide a syntactic representation of the process specification language model theory, in that they are sound and complete with regard to the model theory. That is to say, every axiom is true in every model of the language of the theory, and every sentence of the language that is true in every model of ISO 18629 can be derived from the axioms. Because of this tight connection between the core axioms and the model theory, the core itself can be said to provide a *semantics* for the terms in the ISO 18629 language. The purpose of the core is to axiomatise a set of intuitive semantic primitives that is adequate for describing basic processes. Any two process-related applications must share the Core in common in order to be PSL-compliant. The PSL core is sound and complete.

### 3.3.6 defined lexicon

set of symbols in the nonlogical lexicon which denote defined concepts (i.e. terms with conservative definitions). It is divided into constant, function, and relation symbols

### 3.3.7

#### foundational theory

theory that axiomatise relations and function symbols that denote primitive concepts

### 3.3.8

#### definitional extension

extension of PSL-Core that introduces new linguistic items which can be completely defined in terms of the PSL- core.

### 3.3.9

#### extension

augmentation of PSL-Core containing additional axioms

NOTE 1: The core is a relatively simple theory that is adequate for expressing a wide range of basic processes. However, more complex processes require expressive resources that exceed those of the core. Rather than clutter the core itself with every conceivable concept that might prove useful in describing one process or another, a variety of separate, modular extensions need to be developed and added to the core as necessary. In this way a user can tailor the language precisely to suit his or her expressive needs.

NOTE 2: All extensions are either foundational theories or definitional extensions

### 3.3.10

#### grammar

## **ISO/WD 18629-1:2000(E)**

specification of how logical symbols and lexical terms can be combined to make well-formed formulae

### **3.3.11 isomorphism**

mapping between structures that preserves constants, function values for all functions, and tuples in all relations

### **3.3.12 language**

combination of a lexicon and a grammar.

### **3.3.13 lexicon**

set of symbols and terms

NOTE - The lexicon consists of logical symbols (such as boolean connectives and quantifiers) and nonlogical symbols. For ISO 18629, the non logical part of the lexicon consists of expressions (constants, function symbols, and relation symbols) chosen to represent the basic concepts of the ontology.

### **3.3.14 manufacturing entity**

any concrete or abstract thing in the universe of discourse of a manufacturing enterprise.

### **3.3.15 model**

combination of a set of elements and a truth assignment that satisfies all well-formed formulae in a theory

NOTE – A model is typically represented as a set with some additional structure (partial ordering, lattice, or vector space). The model then defines meanings for the terminology and a notion of truth for sentences of the language in terms of this model. Given a model, the underlying theory of the mathematical structures used in the theory then becomes available as a basis for reasoning about the concepts intended by the terms of the language and their logical relationships, so that the set of models constitutes the formal semantics of the ontology.

### **3.3.16 non-definitional extension**

a non definitional extension is an extension that involves at least one notion that cannot be defined in terms of the PSL core

### 3.3.17 nonlogical lexicon

constant, function, and relation symbols that denote concepts in the application domain

### 3.3.18 ontology

set of terms together with a specification of the meaning of the terms in a formal language

Note1: the primary component of a process specification language such as ISO 18629 is an ontology , i.e. a lexicon of specialised terminology along with some specification of the meaning of terms in the lexicon. The ontology is designed to represent the primitive concepts that, according to ISO 18629, are adequate for describing basic manufacturing, engineering, and business processes.

Note 2: that the focus of an ontology is not only on terms, but also on their meaning. An arbitrary set of terms is included in the ontology, but these terms can only be shared if there is an agreement about their meaning. It is the intended *semantics* of the terms that is being shared, *not simply* the terms.

Note 3: Any term used without an explicit definition is a possible source of ambiguity and confusion. The challenge for an ontology is that a framework is needed for making explicit the meaning of the terms within it. For the ISO 18629 ontology, it is necessary to provide a rigorous mathematical characterisation of process information as well as a precise expression of the basic logical properties of that information in the ISO 18629 language.

### 3.3.19 outer core

set of foundational theories that are extensions of PSL-Core and that are so generic and pervasive in their applicability that they have been put apart.

NOTE – in practice, extensions incorporate the axioms of the outer core.

### 3.3.20 primitive concept

lexical term that has no conservative definition

### 3.3.21 primitive lexicon

set of symbols in the nonlogical lexicon which denote primitive concepts. It is divided into constant, function, and relation symbols

**3.3.22**  
**proof theory**

set of theories and lexical elements necessary for the interpretation of the semantics of the language.

NOTE : it consists of three components : the core, the outer-core and the extensions.

**3.3.23**  
**satisfiable**

a set of sentences is satisfiable if there exists a model for the sentences

**3.3.24**  
**structure**

combination of a set of elements, a set of functions, and sets of tuples for each relation

**3.3.25**  
**theory**

set of axioms

**3.3.26**  
**translation definition**

a biconditional KIF sentence in which the expression on the left-hand side of the biconditional is a term in the application's nonlogical lexicon and the expression on the right-hand side of the biconditional uses only terminology from extensions in the ISO 18629 standard.

## **3.4 Abbreviations**

For the purposes of this part of ISO 18629, the following abbreviations apply:

<b>BNF</b>	Backus-Naur Formalism
<b>CEN</b>	Comité Européen de Normalisation (European Committee for Standardisation)
<b>EDI</b>	Electronic Data Interchange
<b>ENV</b>	European Pre-standard
<b>IDEF3</b>	ICAM DEFinition language 3 Process description capture method
<b>KIF</b>	Knowledge Interchange Format
<b>MANDATE</b>	MANufacturing management DATa Exchange
<b>MMS</b>	Manufacturing Message Services

<b>MRP</b>	Material Requirement Planning
<b>MRP II</b>	Manufacturing Resources Planning
<b>P-LIB</b>	Parts Library
<b>PSL</b>	Process Specification Language
<b>STEP</b>	STandard for the Exchange of Product model data
<b>XML</b>	EXtensible Mark-up Language

## 4 Overview of ISO 18629

### 4.1 Concepts provided

ISO 18629 provides a neutral language for process specification to serve as an interchange language to integrate multiple process-related applications throughout the manufacturing process life cycle, from initial process conception all the way through to process retirement.

ISO 18629 is intended to mainly address industrial discrete manufacturing activities. However ISO 18629 is not limited to this type of process and can be applicable to many other industrial processes, in particular continuous processes, through the development of specific extensions.

ISO 18629 is divided into this overview and a series of parts. The parts are strongly related and developed in close co-operation in order to preserve the consistency of the whole standard.

ISO 18629 addresses information dealing with manufacturing processes, and makes use of product and component descriptions. Process information must also be consistent with process management information (ISO 15531). This requires consistency between ISO 18629, ISO 10303, ISO 13584 and ISO 15531. In particular ISO 18629 is consistent with the ISO 10303 architecture.

### 4.2 The Process Specification Language (PSL)

A goal of ISO 18629 is to facilitate application interoperability by means of the development of translators between the native formats of those applications and ISO 18629. To support this goal,

each Part of the standard will be composed of one or more extensions to PSL Core. For each extension, the standard will include the following:

- nonlogical lexicon
- specification of models
- axiomatization
- satisfiability and axiomatizability theorems
- grammar for process descriptions that use the terminology of the nonlogical lexicon.

### 4.3 Templates for PSL extensions

### 4.3.1 Nonlogical Lexicon

The nonlogical lexicon is the terminology of the standard that corresponds to the concepts and relationships related to manufacturing processes. All terms in the nonlogical lexicon of the standard are either constant, function, or relation symbols in KIF.

Each extension specifies a unique nonlogical lexicon, and any term in the standard must belong to the nonlogical lexicon of a unique extension.

#### 4.3.2. Specification of Models

The model theory of ISO 18629 provides a rigorous abstract mathematical characterization of the semantics of the terminology of ISO 18629. This characterisation defines the meanings of terms with respect to some mathematical structures together with a notion of truth with respect to those structures for sentences of the language.

PSL Core

### 4.3.3 Axiomatisation

The axiomatisation of ISO 18629 is the set of KIF sentences which constrain the interpretation of the terminology in the nonlogical lexicon of ISO 18629.

The axioms of ISO 18629 are organized into PSL-Core and a set of extensions. PSL Core is the set of axioms written in KIF and using only the nonlogical lexicon of PSL Core. The extensions form a lattice of extensions to PSL Core.

Note: The purpose of PSL Core is to axiomatise a set of intuitive semantic primitives that is adequate for describing the fundamental concepts of manufacturing processes. Consequently, this characterization of basic processes makes few assumptions about their nature beyond what is needed for describing those processes, and the Core is therefore rather weak in terms of logical expressiveness. In particular, PSL Core is not strong enough to provide definitions of the many auxiliary notions that become necessary to describe all intuitions about manufacturing processes.

To supplement the concepts of PSL Core, the standard includes a set of extensions that introduce new terminology. An ISO 18629 extension provides the logical expressiveness to express information involving concepts that are not explicitly specified in PSL Core.

For each symbol in the nonlogical lexicon of an extension of ISO 18629, there exist one or more axioms, written in KIF, that constrain its interpretation. A distinction is drawn between definitional and nondefinitional extensions of PSL Core. A definitional extension is an extension whose nonlogical lexicon can be completely defined in terms of PSL Core. Definitional extensions add no new expressive power to PSL Core, Nondefinitional extensions (also called core theories) are extensions of PSL Core that involve at least one new primitive not contained in PSL Core.

All extensions within ISO 18629 must be consistent extensions of PSL Core, and may be consistent extensions of other ISO 18629 extensions. However, not all extensions within ISO 18629 need be mutually consistent.

Example: There may be an extension of that axiomatizes a discrete timeline and another extension that axiomatizes a dense timeline; although these are mutually inconsistent, they are both individually consistent with PSL Core.

### 4.3.4 Grammar for process descriptions

The underlying grammar used for ISO 18629 is that of KIF (Knowledge Interchange Format).



Note: KIF is a formal language based on first-order logic developed for the exchange of knowledge among different computer programs with disparate representations. KIF provides the level of rigour necessary to unambiguously define concepts in the ontology.

Process descriptions shall be sentences in KIF that use the nonlogical lexicon of ISO 18629. In particular, process descriptions shall be restricted to sentences that are satisfied by elements in a model of the axioms of ISO 18629. Process descriptions are not arbitrary sentences.

Example: the process descriptions for deterministic activities will not contain disjunctive sentences over subactivity occurrences.

Each extension shall have an associated BNF grammar for process descriptions, which extends the BNF grammar associated with PSL Core.

### 4.3.6 Format for extensions

All extensions within ISO 18629 will have the following header information:

- Extension Name: Foundational theory names must have a .th suffix, and definitional extension names must have a .def suffix.
- Primitive Lexicon
- Defined Lexicon
- Foundational theories Required by the Extension : This is a list of core theories, each of which is an extension of PSL Core. The given extension is an extension of the set of axioms which is the union of all theories in the list, together with the axioms in the extension.
- Definitional Extensions Required by the Extension This is a list of definitional extensions, each of which is an extension of PSL Core.

The content of any ISO 18629 extension will be a set of KIF sentences. For foundational theories, there will be a set of arbitrary KIF sentences for the primitive lexicon of the theory. For definitional extensions and terms in the defined lexicon of a foundational theory, each term shall be defined as a biconditional sentence in KIF

In addition, each KIF sentence must have corresponding text in English which summarizes the key intuitions captured by the sentence.

## 4.4 Organisation of the ISO 18629 Standard

The components of ISO 18629 are grouped into the following parts:

### Part 1: Overview and basic principles

There shall be two types of extensions within ISO 18629 – foundational theories and definitional extensions. Foundational theories introduce and axiomatise new relations and functions that are primitive. All terminology introduced in a definitional extension shall have conservative definitions using the terminology of the foundational theories.

### Part 1x series: Foundational theories

The current contents of these series of parts addresses:

- Part 110:** PSL Core
- Part 121:** Outer core
- Part 132:** Time and ordering theories
- Part 14:** Resource theories, and
- Part 15:** Activity Performance theories agents

Any new foundational theory shall be included in the Part 1x series.

#### **Part 2x series: External mappings**

The current expected content of this series of part includes:

**Part 21:** EXPRESS

**Part 22:** XML

**Part 23:** UML

This set of mappings may evolve according to industry needs and technology changes.

#### **Part 4x series: Definitional extensions**

In addition to the foundational theories, ISO 18629 provides a series of definitional extensions that shall be used to capture the semantics of process terminology in different applications. All definitions in these extensions use the terminology of the foundational theories in Parts 10, 11, and 12. The series currently includes:

**Part 41:** Activitiesy  
ordering

**Part 42:** Time and state

**Part 43:** Ordering

**Part 44:** Resource roles

**Part 45:** Kinds of resource sets

**Part 46:** Processor activity

**Part 47:** Process intent

Additional extensions are to be developed later according to industry needs by any standardisation committee and others. Any new extension that is a definitional extension of PSL Core shall be included in the Part 4x series.

#### **Part 2xx series: Implementation Protocols**

Parts of this series will be developed later according to industry needs and technology changes.

NOTE – The numbering system for the parts of this standard has been made consistent with the system adopted for the other standards developed within ISO TC184/SC4.

(Placeholder: The roadmap of who uses what part of PSL for which purposes goes here).

## **4.5 ISO 18629-1x series Foundational theories**

### **4.5.1 ISO 18629-110 PSL Core**

The PSL Core is based upon a precise, mathematical, first-order theory, that is a formal language, a precise mathematical semantics for the language, and a set of axioms that express the semantics in the language.

The basic elements of the language for PSL Core are four primitive ~~classes~~one-place predicates, two primitive functions, two primitive binary predicates, and ~~seven~~five primitive-defined predicates ~~relations of the ontology of the PSL Core~~.

The primitive ~~classes~~one-place predicates are activity, activity occurrence, timepoint, and object. ~~OBJECT, ACTIVITY, ACTIVITY\_OCCURRENCE and TIMEPOINT~~. The two functions are beginof and endof. ~~BEGINOF, and ENDOF~~. The primitive binary predicates are before and participates-in; the five defined predicates are between, between-eq, is-occurring-at, and exists-at. ~~The seven relations are BEFORE, BETWEEN, BEFORE EQ, BETWEEN EQ, IS OCCURRING AT, PARTICIPATES IN, EXISTS AT~~.

#### 4.5.2 ISO 18629-1~~21~~ Outer core

There is a set of extensions of PSL Core that are so generic and pervasive in their applicability that they have been set them apart by calling them the ISO 18629 Outer Core.

These extensions are:

- Occurrence Tree
- Subactivity/Activity Occurrence;
- Atomic activities;
- Complex Activities;
- ~~Occurrence Tree~~;
- State;
- Activity-Occurrence ~~Subactivity~~.

~~(Placeholder: we need to re-order the list to correspond to the text ordering; atomic activities)~~

~~The Subactivity Extension describes how activities can be aggregated and decomposed. It also defines the concept of primitive activity, that is, one which can not be decomposed into any further activities.~~

The Occurrence Tree Extension introduces a tree structure over the set of possible activity occurrences; branches in the tree correspond to different sequences of primitive activity occurrences. This extension also introduces the notion of state, in particular, preconditions and effects for activities.

The Subactivity Extension describes how activities can be aggregated and decomposed. It also defines the concept of primitive activity, that is, one which can not be decomposed into any further activities.

The Atomic Activity Extension introduces the occurrence of concurrent activities.

The Complex Activity Extension specifies the relationship between occurrences of the subactivities of an activity and occurrences of the activity itself.

~~The Activity Occurrence Extension defines relations that allow the description of how activity occurrences relate to one another with respect to the time at which they start and end.~~

The State Extension specifies the basic concepts for states and their relationships to activity occurrences. In particular, all states are changed by activity occurrences.

~~The Activity Occurrence Extension defines relations that allow the description of how activity occurrences relate to one another with respect to the time at which they start and end.~~

These extensions, together with ISO 869-Core, provide much of the infrastructure for specifying the definitions of terminology within ISO 18629. Every extension within the current version of ISO 18629 is an extension of one or more of these theories.

### 4.5.3 ISO 18629-13~~2~~ Time and Ordering Theories, Resources and Agents

There is also a set of theories that are nondefinitional extensions of the Outer Core, and which are required by some, though not all, definitional extensions within ISO 18629. These extensions are:

- Subactivity Occurrence Ordering;
- Duration;
- Resource Requirements;
- Resource Sets;
- Activity Performance.

The Subactivity Occurrence Ordering extension specifies the relations required to represent various kinds of partially ordered sets of activities. This includes sequences, parallelism, AND splits/junctions, and OR splits/junctions.

The Duration extension introduces the concept of duration as a relationship among timepoints (intuitively, durations are the “difference” between two timepoints in the timeline). This allows the introduction of quantitative concepts related to time, as well as providing the basis for defining the duration of activities, activity occurrences, and objects.

### 4.5.4 ISO 18629-14 Resource Theories

The Resource Requirements extension axiomatises the concept of resource as any object which is required by an activity. In particular, resources are defined with respect to the possible interactions among activities.

The Resource Set extension axiomatises the concept of a set of resources which as a whole also satisfy the axioms for resources. Different kinds of resource sets are defined in Part 44, and they include such concepts as resource pools (sets of machines) and buffers (sets of inventory resources).

### **4.5.5 ISO 18629-15 Activity Performance**

The Activity Performance ~~extension~~-theory axiomatises the relationship that holds between an activity and the actor (such as a human or machine) who performs the activity.

(Placeholder for agents)

## **4.6 ISO 18629-2x series External mappings**

In addition to specifying the grammar for process descriptions, ISO 18629 also specifies mappings between this grammar and languages used by other manufacturing standards. In particular, there will be mappings between the grammar of ISO 18629 and EXPRESS (to facilitate interoperability with applications using ISO 10303), as well as mappings to XML and UML.

These other languages are used as alternative ways of representing particular process descriptions, rather than in the specification of the semantics of the terminology of ISO 18629.

## **4.7 ISO 18629-4x series Definitional Extensions**

### **4.7.1 ISO 18629-41: Activity Ordering**

The extensions in this part are defined with respect to the Outer Core and Subactivity Occurrence Ordering foundational theory. They include:

- Nondeterministic Activities;
- Ordering Relations over Activities;
- Complex Sequence Ordering Relations;
- Junctions.

The Nondeterministic Activities extension defines different kinds of activities with respect to constraints on the occurrence of subactivities. An activity is deterministic if all subactivities occur, although they may occur in different orderings, whereas an activity is nondeterministic if not every subactivity occurs when the activity itself occurs.

The Ordering Relations over Activities extension is restricted to deterministic activities. It specifies different kinds of activities in which there is a partial ordering over subactivity occurrences.

The Complex Sequence Ordering Relations extension is restricted to nondeterministic activities. It supports the specification of activities in which there is branching and conditional occurrence of subactivities.

The Junctions extension defines kinds of activities which contain splits and joins. It encompasses both deterministic and nondeterministic activities.

### **4.7.2 ISO 18629-42: Time and State**

The extensions in this part are defined with respect to the Outer Core and the Duration core theory; they include:

## ISO/WD 18629-1:2000(E)

- Activity and Occurrence Duration;
- Temporal Ordering Constraints;
- State Constraints;
- Interval Activities.

The Activity and Occurrence Duration extension defines the concept of duration for activities and activity occurrences.

The Temporal Ordering Constraints extension specifies relations among the occurrences of activities with respect to the times at which they occur. These relations are particularly used within scheduling applications.

The State Constraints extension defines relations between states and activity occurrences. These include concepts such as preconditions and effects, as well as the notions of an activity achieving or falsifying various states.

The Interval Activities extension considers kinds of activities that can be defined using the relations in the State Constraints extension. In particular, it considers activities that have the property of being interruptable or non-interruptable.

### 4.7.3 ISO 18629-43 Resource Roles

The extensions in this part are defined with respect to the Outer Core and the Resource Requirements foundational theory; they include:

- Resource Role definitions;
- Capacity-based Concurrency;
- Resource Divisibility;
- Resource Usage.

The Resource Roles extension defines various roles that resources play with respect to activities; these include reusable, consumable, and renewable.

The Capacity-based Concurrency extension defines different kinds of activities and resources using concurrency constraints.

The Resource Divisibility extension considers the different ways in which resources can be shared by multiple activities.

The Resource Usage extension characterizes constraints on resources over the intervals in which activities occur.

### 4.7.4 ISO 18629-44 Kinds of resource sets

The extensions in this part are defined with respect to the Outer Core, the Resource Requirements foundational theory, and the Resource Set foundational theory; they include:

- Resource Set-based Activities;
- Substitutable Resources;
- Homogeneous Resource Sets;

- Inventory Resource Sets;
- Resource Pools.

The primary purpose of these extensions is the axiomatisation of discrete capacity resources. The major problem in this case is that the discreteness of the resource arises from the fact that it is actually composed of a set of resources, and any activity requires or provides some subset of resources in this set.

- Resource Set-based Activities defines kinds of activities which use resource sets;
- Substitutable Resources, which makes the distinction between sets of arbitrary resources and sets of resources that can be substituted for others in an activity.
- Homogeneous Sets, which defines different kinds of substitutable resources
  - Resource Pools, which are equivalent to discrete capacity resources;
  - Inventory Resource Sets, which are related to buffers.

#### 4.7.5 ISO 18629-45 Processor Activities

~~(Place holder for theories required)~~ The extensions in this part consider a particular kind of activities that are defined with respect to the roles of the resources required by the activities; this includes :

- Processor Actions;
- Resource Paths.

Processor actions are actions that use some set of resources, consume some set of material resources, and produce or modify some other set of material resources.

Resource paths are partially ordered sets of processor actions in which the output material resource of one processor action is the input material of the next processor action.

#### 4.7.6 ISO 18629-46 Process intent

~~(Placeholder)~~ The extensions in this part define constraints on agent behaviour, such as goals, policies, commitments, and responsibilities.

### 4.8 ISO 18629-2xx series Implementation protocols

As discussed earlier, ISO 18629 is intended to support interoperability among manufacturing process applications by providing a neutral representation for the exchange of process specifications.

The Implementation protocols in the 200 series of ISO 18629 identify the different extensions within the ontology that are necessary for developing translators among applications in different manufacturing domains. These currently include:

- Process modelling;
- Process planning;
- Scheduling;
- Simulation;

- Project management;
- Production planning.

## **5 Conformance testing methodology and framework**

### **5.1 Conformance of Applications with ISO 18629**

A manufacturing process application will be conformant to ISO 18629 if and only if the following conditions hold:

- The nonlogical lexicon of the manufacturing process application must be specified. This is the set of terms used by the application that refer either to elements in the domain or relations among these elements.
- There exist translation definitions for each term in the application's nonlogical lexicon, and these definitions are consistent with PSL Core and a subset of the set of extensions in the standard.
- There exists an implemented bi-directional translator between ISO 18629 and the application
- There exists a BNF grammar for the process descriptions of the application.

Note that an application need not be compliant with all extensions of ISO 18629, but it must be compliant with PSL Core. Two applications will be interoperable if they are compliant with the same set of ISO 18629 extensions.

### **5.2 Conformance of Ontologies with ISO 18629**

#### **5.2.1 Options for conformance of ontologies**

There are two ways in which other ontologies interact with ISO 18629 -- they may be new extensions to ISO 18629, or they may be external to the scope of PSL e.g. ontologies for products or geometry.

#### **5.2.2 Adding extensions**

An extension consisting of a user-specific set of concepts that are used in conjunction with the ISO 18629 standard shall be conformant with ISO 18629 if it uses only the terminology in the nonlogical lexicon of the ISO 18629 standard. Any extension that introduces new primitives shall be consistent with PSL Core.

#### **5.2.3 Conformance of external ontologies**

An external ontology that includes its own axiomatization of process-related terminology is conformant if it is consistent with the axioms of the ISO 18629 standard.

Example: The axioms in any external ontology of time must be consistent with the axioms of PSL Core.

### **5.3 Conformance of new parts**

#### **5.3.1 Specification of Models**

(place holder for changing this text below on new extensions that will become part of the standard)

(The model theory of ISO 18629 provides a rigorous abstract mathematical characterization of the semantics, or meaning, of the terminology of ISO 18629. This characterization defines the meanings of terms with respect to



some set of mathematical structures together with a notion of truth with respect to these structures for sentences of the language.

For each extension, there must be a specification of the models of the axioms within the extension. This specification must contain the following:

- definition of a set of mathematical structures, including the underlying set of elements, the functions and relations on this set of elements, and any distinguished elements of the set;
- for each constant, function, and relation symbol in the lexicon, a specification of which substructures are isomorphic to the extension of the constant, function, or relation that is denoted by the symbol;
- a classification of the structures up to isomorphism, and a proof that shows this classification is correct with respect to the definition of the class of structures.

In addition, for every extension in ISO 18629, the associated class of structures must contain the models of PSL Core as substructures. )

### 5.3.2 Satisfiability and Axiomatizability Theorems

(place holder for new extensions that will become part of the standard)

(For each extension of ISO 18629, the following two theorems must be proven:

- **Satisfiability:** every structure in the set of structures associated with the extension is a model of the axioms in the extension.
- **Axiomatizability:** every model of the axioms in the extension is isomorphic to some structure in the set of structures associated with the extension. )

**Annex A**

(Informative)

**Background to the development of the standard**

This standard is one outcome of the PSL project at the National Institute of Standards and Technology. The approach in developing the language involved five phases: requirements gathering, existing process representation analysis, language creation, pilot implementation and validation, and submission as a candidate standard. The completion of the first phase resulted in a comprehensive set of requirements for specifying manufacturing processes. In the second phase, twenty-six process representations were identified as candidates for analysis by the development team and analyzed with respect to the phase one requirements. Nearly all of the representations studied focused on the syntax of process specification rather than the meaning of terms, the semantics. While this is sufficient for exchanging information between applications of the same type, such as process planning, different types of applications associate different meanings with similar or identical terms. As a result of this, a large focus of the third phase involved the development of a formal semantic layer (an ontology) for ISO 18629 based on the Knowledge Interchange Format (KIF) specification. By using this ontology to define explicitly and clearly the concepts intrinsic to manufacturing process information, the language was used to integrate multiple existing manufacturing process applications in the fourth phase of the project.

Four families of manufacturing process applications have been used in the pilot implementations: process modelling, process planning, scheduling, and manufacturing simulation. Representative applications from each family were used in the demonstration of interoperability. In each case, the terminology of the application was identified and the implicit semantics of this terminology was specified. Based on this semantics, the terminology of the application was mapped to the semantically equivalent terminology within the developing standard. If there were no terminology within the existing standard that was equivalent to the semantics of the application, an extension was made to the terminology and semantics of the developing standard. Completion of the pilot implementations will validate the breadth of the set of concepts within ISO 18629 with respect to the applications.

In addition to the design and validation of ISO 18629 using pilot implementations for interoperability, the project also provides formal validation of the semantics of the terminology of the standard.

## Annex B (Informative)

### The Need for Semantics

Existing approaches to process modeling lack an adequate specification of the semantics of the process terminology, which leads to inconsistent interpretations and uses of information. Analysis is hindered because models tend to be unique to their applications and are rarely reused. Obstacles to interoperability arise from the fact that the systems that support the functions in many enterprises were created independently, and do not share the same semantics for the terminology of their process models.

EXAMPLE : consider Figure 1 in which two existing process planning applications are attempting to exchange data. Intuitively the applications can share concepts; for example, both *material* in Application A and *workpiece* in Application B correspond to a common concept of *work-in-progress*. However, without explicit definitions for the terms, it is difficult to see how concepts in each application correspond to each other. Both Application A and B have the term *resource*, but in each application this term has a different meaning. Simply sharing terminology is insufficient to support interoperability -- the applications must share their semantics, i.e., the *meanings* of their respective terminologies.

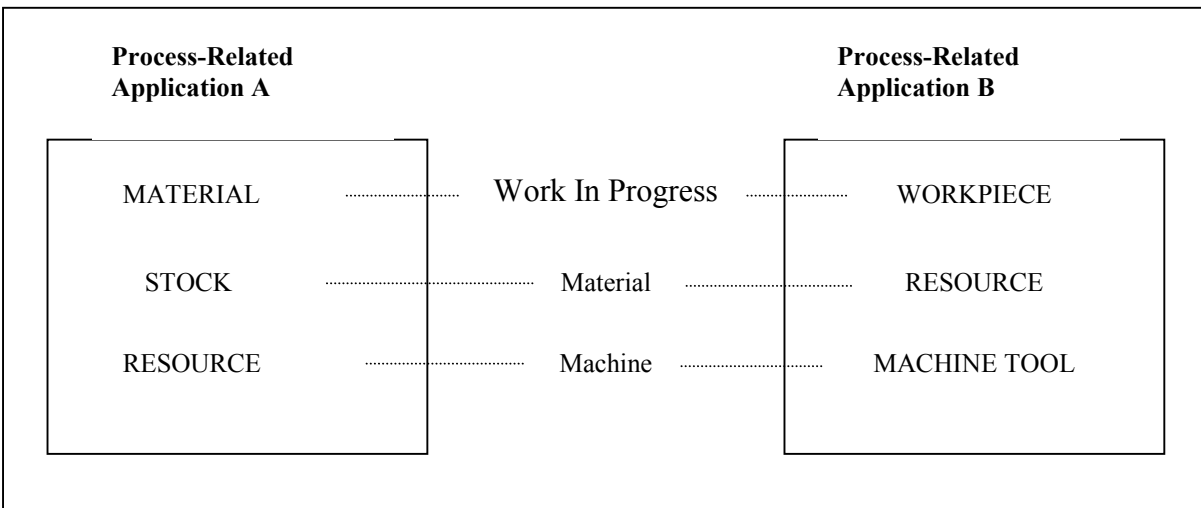


Figure 1: Why Semantics?

A rigorous foundation for process design, analysis, and execution therefore requires a formal specification of the semantics of process models. One approach to generating this specification is through the use of *ontologies*. An ontology is a formal description of the entities within a given domain: the properties they possess, the relationships they stand in, the constraints they are subject to, and the patterns of behaviour they exhibit. It provides a common terminology that helps to capture key distinctions among concepts in different domains, which aids in the translation process.

**Annex C**  
(Informative)

**Interoperability**

A goal of ISO 18629 is to facilitate application interoperability by means of the development of translators between the native formats of those applications and ISO 18629. Without an overarching language like ISO 18629 to serve as a medium of information interchange between applications, a unique translator must be written for every two-party exchange. However, this approach requires  $n(n-1)$  translators for  $n$  different ontologies. With ISO 18629 serving as a standardised medium of information interchange, the number of translators for  $n$  different ontologies is reduced to  $n$ , since it only requires translators between native ontologies and the interchange ontology. The other feature of this approach is that the applications interact primarily through the exchange of files that contain process information.

## **Annex D** (Informative) **Logical Foundations**

### **Specification of Models**

The first step in the methodology is the specification of some set of structures. These structures provide a rigorous mathematical characterisation of the semantics of concepts in the domain. The objective is to identify each concept with an element of some mathematical structure (graphs, linear orderings, partial orderings, groups, fields, and vector spaces); the underlying theory of the mathematical structure then becomes available as a basis for reasoning about the concepts and their relationships. In particular, given the nonlogical lexicon in some language, structures are isomorphic to the extensions of the relations, functions, and constants denoted by the relation symbols, function symbols, and constant symbols of the lexicon.

To model some domain, the necessary properties need to be captured by the structures. Ideally, properties of the structures need to be reflected by the properties of the corresponding concepts in the domain. These characteristics can be used to evaluate the adequacy of the intended structures. If some property is not captured, then a decision must be taken about this property. If it is not necessary, then it can be ignored. If it is deemed necessary, then the characterisation of the intended structures must be extended so that it includes some formalisation of this property. For example, there are domains where it is necessary to make distinctions between discrete and continuous processes. Since these properties are not considered within the scope of the work within this standard, they are not formalised within any structures. Of course, to represent these properties, it is then necessary to extend the set of structures appropriately.

This relationship between the empirical interpretations and the structures is, of course, informal, but these domain interpretations can be considered as providing a physical interpretation of the structures. In this sense, it is possible to adopt an experimental or empirical approach to the evaluation of the set of intended structures to attempt to falsify these structures. If some objects or behaviour are found within the domain without any correspondence to an intended structure, there is a counter-example to the set of structures. In response, it is possible to either redefine the scope of the set of structures (without including the behaviour within the characterisation of the structures) or modify the definition of the set of structures so that they capture the new behaviour.

**EXAMPLE :** For example, physicists use various kinds of differential equations to model different phenomena. However, they do not use linear differential equations to model heat diffusion, and they do not use second-order partial differential equations to model the kinematics of springs. To model some phenomena using a class of differential equations, it is possible to use the equations to predict behaviour of the physical system; if the predictions are falsified by observations, then the set of equations is incorrect. Similarly, in this case, it is possible to use some set of structures to predict behaviour or characterise states of affairs; if there is no physical scenario in the domain which corresponds to these behaviours or states of affairs, then intuitively the set of structures is incorrect.

### **Axiomatisation and Extensions**

The axiomatisation is the set of KIF sentences which constrain the interpretation of the terminology in the nonlogical lexicon. Within the standard these sentences are aggregated into the PSL Core and a set of extensions.

The PSL Core is a set of axioms written in the basic language of ISO 18629. Informally, an ISO 18629 extension gives one the theoretical machinery to express information involving concepts that

are not part of the Core. Extensions give the ISO 18629 a clean, modular character. The PSL Core is a relatively simple theory that is adequate for expressing a wide range of basic processes. However, more complex processes require expressive resources that exceed those of the core. Rather than clutter the core itself with every conceivable concept that might prove useful in describing one process or another, a variety of separate, modular extensions have been (and continue to be) developed that can be added to the core as needed. In this way a user can tailor ISO 18629 precisely to suit his or her expressive needs

To define an extension, new constants and/or predicates are added to the basic ISO 18629 language, and, for each new linguistic item, one or more axioms are given that constrain its interpretation. In this way one provides a “semantics” for the new linguistic items.

EXAMPLE : A good example of such an extension is the theory of time durations. The PSL Core itself does not provide the resources to express information about time durations. However, in many contexts, such a notion might be useful or even essential. Consequently, a theory of time durations has been developed which can be added as to the core, thus providing the user with the desired expressive power.

A distinction can be drawn between definitional and non-definitional extensions. As the name suggests, a definitional extension is an extension whose new linguistic items can be completely defined in terms of the PSL Core. Theoretically, then, definitional extensions add no new expressive power to the core, and hence involve no new theoretical overhead. However, because definitions of many subtle notions can be quite involved, definitional extensions can prove extremely useful for describing complex processes in as succinct a manner as possible. Non-definitional extensions, of course, are extensions that involve at least one notion that cannot be defined in terms of the PSL Core.

### **Satisfiability and Axiomatisability Theorems**

One of the advantages of taking the formal approach to ontology design is that it is possible to prove that the axioms and definitions of the ontology are consistent, and that they are complete with respect to the intended models which are specified in the model theory of the ontology. The axioms in an ISO 18629 extension are proven to be consistent by constructing a model that satisfies all of the axioms. A complete characterisation of these models shows that any structure that satisfies the axioms in the extension is one of the intended models.

This approach to the semantics of the standard has several benefits. First, it is possible to demonstrate that the standard itself is sound, as well as provide a characterisation of its completeness. Second, because the terminology of the standard is represented in KIF, which is machine-readable, process descriptions that are specified using the standard can be checked for consistency using automated deduction software. This will avoid problems that may arise when attempting to translate an incorrect process description from one application to another.

Finally, the axiomatisation of the terminology within the standard enables the use of translation definitions as a basis for process description exchange among applications. Recall that the primary role that ISO 18629 will play is that of a neutral representation of process information to support interoperability. This will primarily be done by mapping the terminology from one process application into ISO 18629, and then mapping the terminology of the standard back into another application’s terminology. By using a logical language such as KIF, these mappings can be represented as sentences within the language, which can then be manipulated by automated deduction software.

Strictly speaking, it is only necessary to show that a model exists in order to demonstrate that a theory is satisfiable. However, in the axiomatisation of domain theories, a complete characterisation of the possible models is needed.

EXAMPLE : For example, in the domain of processes, to show that a theory is satisfiable, it is only necessary to specify a process description which together with the axioms is satisfied by some structure.

The risk with this approach is the possibility of having demonstrated satisfiability only for some restricted kind of processes. For example, a theory of processes may be shown to be consistent by constructing a satisfying interpretation, but the interpretation may require that there is no concurrency, or that only sequences of activities are allowed; although such a model may be adequate for such processes, it would in no way be general enough for our purposes.

The Satisfiability Theorem states that every structure in the set are models of the theory, while the Axiomatizability Theorem states that every model of the theory is a structure in the set. The purpose of the Axiomatizability Theorem is to demonstrate that there are no unintended models of the theory, that is, no models which are not specified in the set of structures.

**Annex E**  
(Informative)

**Semantic Architecture**

The general semantic architecture of ISO 18629 can be summarized as in Figure2. The solid arrows indicate the definability relation. The dashed lines indicate partial definability. Partial definability occurs when some, but not all, additional linguistic items in the language of an extension are definable. Two or more solid arrows pointing to the same oval indicate the possibility that more than one given theory might be jointly used to define a new extension.

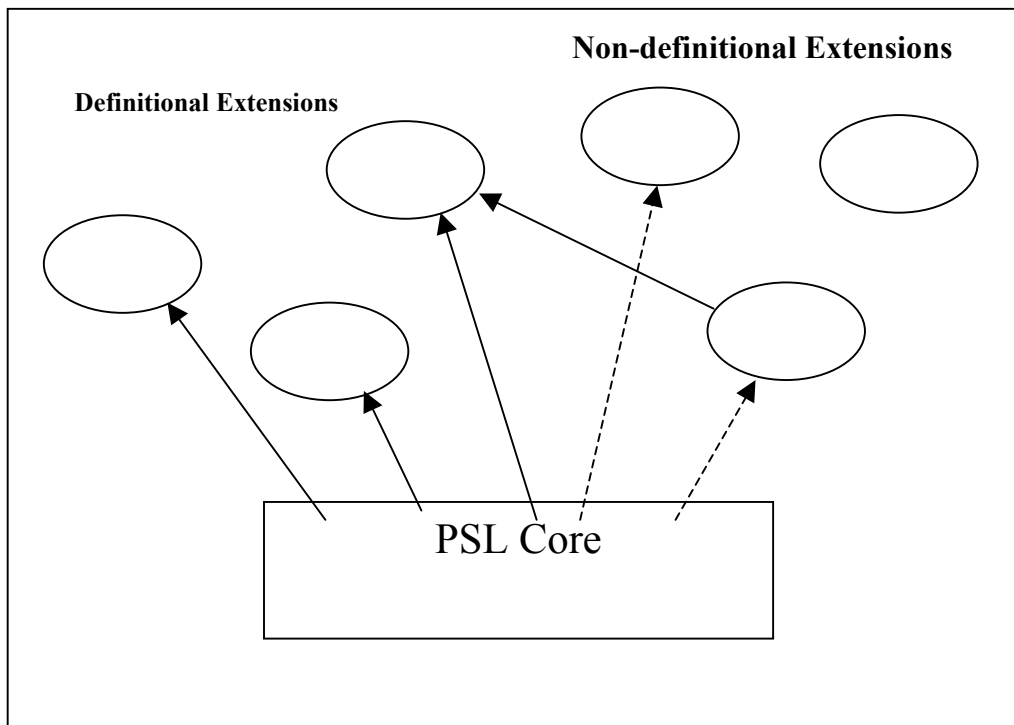


Figure 2: The ISO 18629 Semantic Architecture

PSL-Core is a minimal set of concepts and axioms that is shared by all process applications but is not sufficient by itself for representing all process-related concepts. The PSL core can be used to support interoperability by ensuring that concepts that are common across applications can be exchanged correctly, while also identifying those concepts on which the applications disagree and hence cannot share. By organizing all concepts within ISO 18629 as a partially ordered set of extensions, applications can explicitly declare which extensions provide the semantics for their application's terminology.



## Bibliography

- Schlenoff, C., Knutilla, A., Ray, S., Unified Process Specification Language: Requirements for Modeling Processes: NISTIR 5910, 1996, National Institute of Standards and Technology, Gaithersburg, MD.
- Knutilla, A., et al., Process Specification Language: An Analysis of Existing Representations, NISTIR 6133, 1998, National Institute of Standards and Technology, Gaithersburg, MD.
- Genesereth, M., Fikes, R.: Knowledge Interchange Format (Version 3.0) - Reference Manual, 1992, Computer Science Dept., Stanford University, Stanford, CA.
- Catron, B., Ray, S., ALPS: A Language for Process Specification, Int. J. Computer Integrated Manufacturing, 1991, Vol. 4, No. 2, 105-113.
- Fox, M., et al, An Organisation Ontology for Enterprise Modeling: Preliminary Concepts”, Computers in Industry, 1996, Vol. 19, pp. 123-134.
- Uschold, M., et. al., “The Enterprise Ontology”, The Knowledge Engineering Review, Vol. 13(1), pp. 31-89, Special Issue on “Putting Ontologies to Use” (eds. Uschold, M. and Tate, A.), Cambridge University Press.
- Pease, A., Core Plan Representation (CPR), <http://www.teknowledge.com/CPR2/>, November 13, 1998.
- Tate, A., “Roots of SPAR – Shared Planning and Activity Representation”, The Knowledge Engineering Review, Vol. 13(1), pp. 121-128, Special Issue on “Putting Ontologies to Use” (eds. Uschold, M. and Tate, A.), Cambridge University Press. <http://www.aii.ed.ac.uk/~arpi/spar/>
- Lee, J., et al, “The PIF Process Interchange Format and Framework”, The Knowledge Engineering Review, Vol. 13(1), pp. 91-120, Special Issue on “Putting Ontologies to Use” (eds. Uschold, M. and Tate, A.), Cambridge University Press.
- Workflow Management Coalition Members, Glossary: A Workflow Management Coalition Specification, Belgium, 1994.
- Uschold, M. and Gruninger M., Ontologies: Principles, Methods, and Applications, Knowledge Engineering Review, 1996, Vol. 11, pp. 96-137.
- Reiter, R., The frame problem in the situation calculus: a simple solution (sometimes) and a completeness result for goal regression, Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy, 1991, pages 418-440, Academic Press, San Diego.
- ISO 18629 Ontology – Current Theories and Extensions, June 28, 1999.
- Habrias, H., Dictionnaire encyclopédique du génie logiciel, MASSON, 1997

**ISO/WD 18629-1:2000(E)**

- Hayes, P., A Catalog of Temporal Theories, Tech Report UIUC-BI-AI-96-01, 1996, University of Illinois.
- Smith, S.J.J., et al., Integrating Electrical and Mechanical Design and Process Planning, Proceedings of the IFIP Knowledge Intensive CAD Workshop, 1996, Carnegie-Mellon University (CMU).
- Extensible Markup Language (XML) 1.0, W3C Recommendation 10-February-1998, <http://www.w3.org/TR/1998/REC-xml-19980210>.
- Resource Description Framework (RDF) Model and Syntax, W3C Recommendation 22 February 1999, <http://www.w3.org/TR/REC-rdf-syntax/>.
- Resource Description Framework (RDF) Schema Specification, W3C Proposed Recommendation 03 March 1999, <http://www.w3.org/TR/PR-rdf-schema/>.

## Index