\ISO TC184/SC4/JWG8 N255

Date: 2001-09-10

Supersedes ISO TC184/SC4/JWG8 N232

ISO/CD 18629-1
Industrial automation system and integration -- Process specification language:
Part 1: 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 42nd 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 d'Evry / IUT
91025 Evry Cedex, France
Telephone: + 33 1 69 47 73 11
Telefacsimile: +33 1 69 47 73 06
Electronic mail: afcd@univ-savoie.fr

Contents

Page

Figures

Foreword

ISO (the International Organisation for Standardisation) is a world-wide 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.

Attention is drawn to the possibility that some of the element of this part of ISO 18629 may be the subject of patents rights. ISO shall not held the responsible for identifying any or all such patent rights. 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 organisations. To be competitive and maintain good economic performance, manufacturing organisations 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. Organisations 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 realisation process modelling, and project management. However, each of these applications utilises 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 (PSL) that complements the process representations utilised 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 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 utilise 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.

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 ISO 18629 to enforce uniformity in manufacturing process representations. Objectives and design of software applications vary. Therefore the implementation of an interoperable application must necessarily be influenced by the particular objectives and processes of each specific application. 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 ISO 18629.

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 Concepts within a process characterisation language include the dynamic or kinematics properties of a process, independently of specific processes, such as: tool chatter, a numerical model capturing the dynamic behaviour 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 the 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:[1], *Industrial automation systems and integration - Parts library - Part 1: Overview and fundamental principles*.

— ISO 15531-1:[2], *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)*.

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;

— discrete manufacturing;

— industrial process;

— manufacturing;

[1] To be published
[2] To be published

— manufacturing process;

— process;

— process planning;

— ; 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

3.3.3
bi-conditional sentence
KIF sentence of the form (□□□ P Q) for well-formed formulæ P and Q

NOTE KIF is currently on-going standardisation process in ISO/IEC JTC1

3.3.4
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

3.3.5
defined lexicon
set of symbols in the non-logical lexicon which denote defined concepts. It is divided into constant, function and relation symbols

EXAMPLE terms with conservative definitions.

3.3.6
definitional extension
extension of PSL-Core that introduces new linguistic items which can be completely defined in terms of the PSL-Core.

NOTE: Definitional extensions add no new expressive power to PSL-Core.

3.3.7
extension
augmentation of PSL-Core containing additional axioms

NOTE 1 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 PSL-Core. Rather than clutter the PSL-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 PSL-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 foundational theories, definitional extensions or non-definitional extensions.

3.3.8
foundational theory
theory that turns into axioms relations and function symbols that denote primitive concepts

3.3.9
grammar
specification of how logical symbols and lexical terms can be combined to make well-formed formulae

3.3.10
language
combination of a lexicon and a grammar.

3.3.11
lexicon
set of symbols and terms

NOTE The lexicon consists of logical symbols (such as Boolean connectives and quantifiers) and non-logical 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.

3.3.12
manufacturing entity
any concrete or abstract thing in the universe of discourse of a manufacturing enterprise.

3.3.13
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.

3.3.14
core theory
an extension that involves at least one concept that cannot be using PSL-Core

3.3.15
non-logical lexicon
constant, function and relation symbols that denote concepts in the application domain

3.3.16
ontology
set of terms together with a specification of the meaning of the terms in a formal language

NOTE 1 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 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.

3.3.17
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.

3.3.18
primitive concept
lexical term that has no conservative definition

3.3.19
primitive lexicon
set of symbols in the non-logical lexicon which denote primitive concepts. It is divided into constant, function and relation symbols

3.3.20
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 PSL-Core, the Outer Core and the extensions.

3.3.21
PSL-Core
theory that axiomatizes the concepts of activity, activity-occurrence, time-point, and object.

NOTE The motivation for PSL-Core is any two process-related applications must share these axioms in order to exchange process information, and hence is adequate for describing the fundamental concepts of manufacturing processes. Consequently, this characterisation of basic processes makes few assumptions about their nature beyond what is needed for describing those processes, and the PSL-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.

3.3.22
satisfiable
a set of sentences is satisfiable if there exists a model for the sentences

3.3.23
structure
combination of a set of elements, a set of functions, and sets of tuples for each relation

3.3.24
theory
set of axioms

3.3.25
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 non-logical 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:

BNF Backus-Naur Formalism

CEN Comité Européen de Normalisation (European Committee for Standardisation)

EDI Electronic Data Interchange

ENV European Pre-standard

IDEF3 ICAM DEFinition language 3 Process description capture method

KIF Knowledge Interchange Format

JTC 1 Joint Technical Committee between ISO and IEC

MANDATE MANufacturing management DATa Exchange

MMS Manufacturing Message Services

MRP Material Requirement Planning

MRP II Manufacturing Resources Planning

P-LIB Parts Library

PSL Process Specification Language

STEP STandard for the Exchange of Product model data

UML Unified Modelling Language

NOTE For further information, see [1]

XML 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 a series of parts, the first of which is this overview. 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:

— non-logical lexicon

— specification of models

— axiomatisation

— satisfiability and axiomatisability theorems

— grammar for process descriptions that use the terminology of the non-logical lexicon.

4.3. Templates for PSL extensions

The templates for the development of extensions are defined in terms of non-logical lexicon, specification of models, axiomatization, grammar for process description and format of the extensions. Those templates are described in the following clauses:

4.3.1. Non-logical Lexicon

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

Each extension specifies a unique non-logical lexicon. Any term in the standard shall belong to the non-logical lexicon of a unique extension.

4.3.2. Specification of Models

The model theory of ISO 18629 provides a rigorous abstract mathematical characterisation 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.

4.3.3. Axiomatisation

The axiomatisation of ISO 18629 is the set of KIF sentences that constrain the interpretation of the terminology in the non-logical lexicon of ISO 18629.

The axioms of ISO 18629 are organised into PSL-Core and a partially ordered set of extensions to PSLCore. An ISO 18629 extension provides the logical expressiveness to express information involving concepts that are not explicitly specified 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.

There may be an extension of that turn into axioms a discrete timeline and another extension that turn into axioms a dense timeline; although these are mutual EXAMPLE ly 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 non-logical 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 sub-activity occurrences.

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

4.3.5. Format for extensions

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

— Extension name: Foundational theory names must have a suffix of the form .th, and definitional extension names must have a suffix of the form .def;

— Primitive lexicon;

— Defined lexicon;

— Foundational theories required by the extension: This is a list of PSL-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 by a biconditional sentence in KIF. In addition, each KIF sentence must have corresponding text in English that summarises 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:

a) Part 1: Overview and basic principles

There shall be two types of extensions within ISO 18629– foundational theories and definitional extensions.

b) Part 1x series: Foundational theories

The current contents of these series of parts addresses:

- Part 11: PSL-Core;

- Part 12: Outer Core;
- Part 13: Time and ordering theories;
- Part 14: Resource theories;
- Part 15: Activity performance theories.

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

c) 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.

d) 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. The series currently includes:

- Part 41: Activities;
- Part 42: Time and state;
- Part 43: Ordering;
- Part 44: Resource roles;
- Part 45: Kinds of resource sets;
- Part 46: Processor activities;
- Part 47: Process intent.

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

e)Part 2xx series: Translator Implementation Guidelines

Parts of this series will be developed 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.

4.5. ISO 18629-1x series Foundational theories

4.5.1. ISO 18629-11 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 are four primitive classes, two primitive functions, and seven primitive relations of the ontology of the PSL-Core.

The primitive classes are activity, activity_occurrence , timepoint, and object.

The two functions are beginof and endof.

The seven relations are before, occurrence_of, between, before-eq, between-eq, is-occurrin-at, participates-in, exist-at.

4.5.2. ISO 18629-12 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 Outer Core.

These extensions are:

— Occurrence Trees

— Discrete States

— Subactivity

— Atomic Activity

— Complex Activity

— Activity Occurrences

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.

The Discrete State extension specifies the basic concepts for states and their relationships to activity occurrences. In particular, all discrete states are changed by activity occurrences, but they do not change during an activity occurrence.

The Subactivity extension describes how activities can be aggregated and decomposed.

The Atomic Activity extension introduces the class 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.

These extensions, together with PSL-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.

The following sets of theories define other non-definitional extensions of the Outer Core, required by some, though not all, definitional extensions within ISO 18629.

4.5.3. ISO 18628-13 Time and ordering theories

The extensions related to time and ordering theories are:

— Duration;

— Subactivity occurrence ordering.

The duration extension introduces the concept of duration as a relationship among timepoint. 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.

NOTE intuitively, duration are the "difference" between two timepoint in the timeline

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.

4.5.4. ISO 18629-14 Resource theories

The extensions related to resource theories are:

— Resource requirements;

— Resource sets.

The resource requirements extension turn into axioms 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 turn into axioms 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 theories

The extension related to activity performance theories is:

— Activity performance.

The Activity performance extension turn into axioms the relationship that holds between an activity and the actor (such as a human or machine) who performs the activity.

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: Activities

The extensions in this part are defined with respect to the Outer Core theory (Part 12). They include:

— Non-deterministic activities.

The non-deterministic 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 non-deterministic if not every subactivity occurs when the activity itself occurs.

4.7.2. ISO 18629-42: Time and State

The extensions in this part are defined with respect to the Part 12 Outer Core and the Part 13 Time and ordering theories. They include:

— Activity and occurrence duration;

— Interval activities;

— State constraints.

The activity and occurrence duration extension defines the concept of duration for activities and activity occurrences.

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 interruptible or non-interruptible.

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.

4.7.3. ISO 18629-43: Ordering

The extensions in this part are defined with respect to the Part 12 Outer Core and the Part 13 Time and ordering theories. They include:

— Complex sequence ordering relations;

— Ordering relations over activities;

— Temporal ordering constraints.

The complex sequence ordering relations extension is restricted to non-deterministic activities. It supports the specification of activities in which there is branching and conditional occurrence of sub-activities.

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 sub-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.

4.7.4. ISO 18629-44 Resource roles

The extensions in this part are defined with respect to the Part 12 Outer Core and the Part 14 Resource theories. They include:

— Capacity-based concurrency;

— Resource divisibility;

— Resource role;

— Resource usage.

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

The resource usage extension characterises constraints on resources over the intervals in which activities occur.

4.7.5. ISO 18629-45 Kinds of resource sets

The extensions in this part are defined with respect to the Part 12 Outer Core and the Part 14 Resource theories. They include:

— Homogeneous resource sets;

— Inventory resource sets;

— Resource pools;

— Resource set-based activities;

— Substitutable resources.

The primary purpose of these extensions is the axiomatisation of discrete capacity resourcesfor which 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.

Homogeneous sets define different kinds of substitutable resources.

Inventory resource sets are related to buffers.

Resource pools are equivalent to discrete capacity resources.

Resource set-based activities define kinds of activities which use resource sets.

Substitutable resources make the distinction between sets of arbitrary resources and sets of resources that can be substituted for others in an activity.

4.7.6. ISO 18629-46 Processor Activities

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.

4.7.7. ISO 18629-47 Process intent

This number has been reserved for the development of appropriate extensions.

4.8. ISO 18629-2xx series Translator implementation guidelines

ISO 18629 supports interoperability among manufacturing process applications through the implementation of translators between an application's process descriptions and ISO 18629 process descriptions that use the ISO 18629 ontology.

The guidelines 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;

— Production planning;

— Project management;

— Scheduling;

— Simulation;

— Process execution.

5. Conformance testing methodology and framework

5.1. Conformance of Applications with ISO 18629

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

— The non-logical lexicon of the manufacturing process application is specified;

NOTE: This is the set of terms used by the application that refer either to elements in the domain or relations among these elements.

— There exists a BNF grammar for the process descriptions of the application;

— There exists an implemented bi-directional syntactic translator between ISO 18629 process descriptions and the application process descriptions;

— There exist translation definitions for each term in the application's non-logical lexicon, and these definitions are consistent with the axioms of PSL-Core and a subset of the set of extensions in ISO 18629;

NOTE 1: An application need not be compliant with all extensions of ISO 18629, but it shall be compliant with PSL-Core.

NOTE 2: Two applications are interoperable if they are conformant with the same set of ISO 18629 extensions.

5.2. Conformance of Ontologies with ISO 18629

5.2.1. Conformance of user-defined 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 a subset of extensions in ISO 18629 if it uses only the terminology in the non-logical lexicon of those extensions in ISO 18629.

The axioms in any extension that introduces new primitives shall be consistent with the axioms of PSL-Core.

5.2.2. Conformance of external ontologies

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

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

5.3. Conformance of future extensions

This clause specifies the conditions that shall be satisfied by any future extensions to the ontology of ISO 18629.

5.3.1. Specification of Models

For each extension, there shall be a technical document that includes a specification of the models of the axioms within the extension. This specification shall 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.

5.3.2. Satisfiability and Axiomatisability Theorems

For each extension of ISO 18629, the accompanying technical document shall contain proofs of the following two theorems:

— Satisfiability: every structure in the set of structures associated with the extension is a model of the axioms in the extension;

— Axiomatisability: every model of the axioms in the extension is isomorphic to some structure in the set of structures associated with the extension.

Annex Error! Unknown switch argument. (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 [7]. In the second phase, twenty-six process representations were identified as candidates for analysis by the development team and analysed with respect to the phase one requirements [5]. 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 [3], [8], [9]) for ISO 18629 based on the Knowledge Interchange Format (KIF) specification [4].

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 were 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 concepts within the existing ontology that preserved the semantics of the application, an extension was made to the terminology and semantics of the developing standard. For further information about the background of the development of the standard, see [6].

Annex Error! Unknown switch argument. (informative)

The Need for Semantics

Existing approaches to process modelling 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 B.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 : the meanings of their respective terminologies.

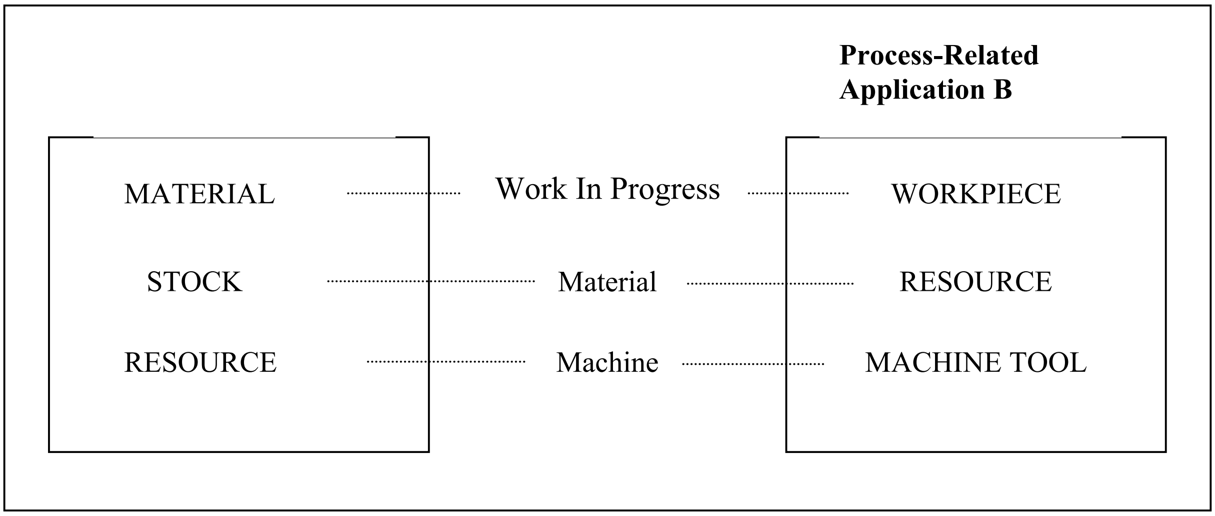

Figure B.1 — Why semantics?

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 Error! Unknown switch argument.
(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 Error! Unknown switch argument. (informative)

Logical foundations

Error! Unknown switch argument..1 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 non-logical 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.

D.2 Axiomatisation and Extensions

The axiomatisation is the set of KIF sentences which constrain the interpretation of the terminology in the non-logical 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 PSL-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 PSL-Core. Rather than clutter the PSL-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 PSL-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 timedurations. 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 PSL-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 PSL-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.

D.3 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 axiomatisability theorem states that every model of the theory is a structure in the set. The purpose of the axiomatisability 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 **Error! Unknown switch argument.**
(informative)

Semantic architecture

The general semantic architecture of ISO 18629 can be summarised as in Figure E.1. 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.

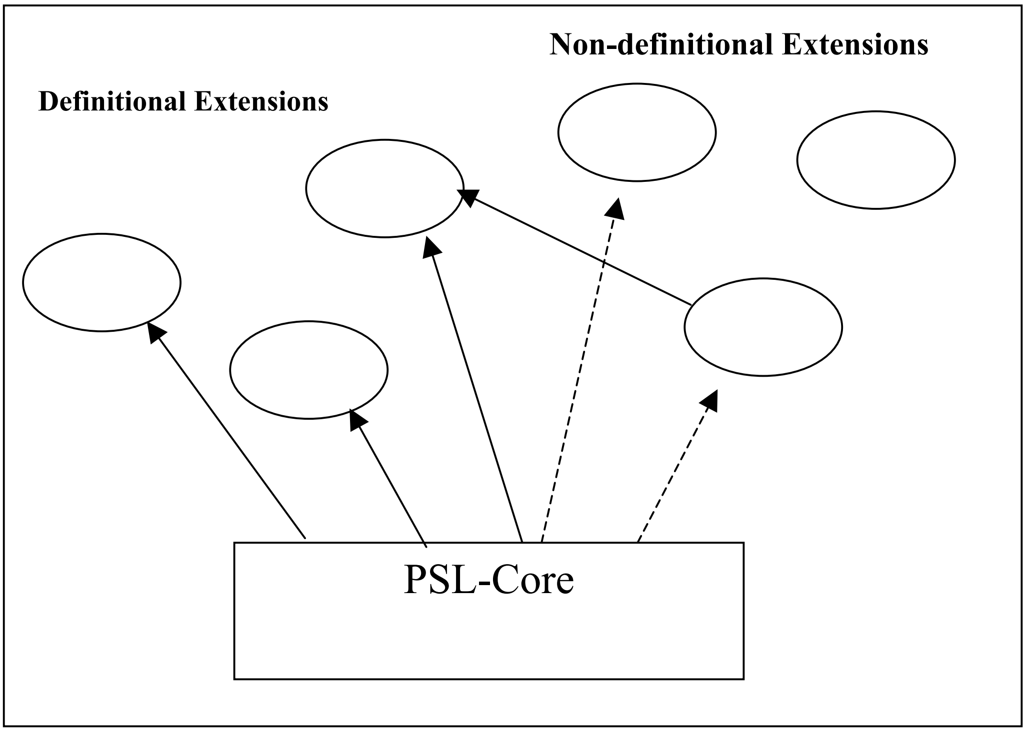

Figure E.1 — The ISO 18629 Semantic architecture

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 organising 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

[1] - Booch G., Rumbaugh J., Jacobson I., The Unified Mmodelling Language User Guide, Addison-Wesley, 1999
[2] – Extensible Markup Language (XML) 1.0, W3C Recommendation 10-February-1998, http://www.w3.org/TR/1998/REC-xml-19980210.
[3] – Fox, M., et al, An Organisation Ontology for Enterprise Modelling: Preliminary Concepts", Computers in Industry, 1996, Vol. 19, pp. 123-134.
[4] – Genesereth, M., Fikes, R.: Knowledge Interchange Format (Version 3.0) - Reference Manual, 1992, Computer Science Dept., Stanford University, Stanford, CA.
[5] – Knutilla, A., et al., Process Specification Language: An Analysis of Existing Representations, NISTIR 6133, 1998, National Institute of Standards and Technology, Gaithersburg, MD.
[6] – 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.
[7] – Schlenoff, C., Knutilla, A., Ray, S., Unified Process Specification Language: Requirements for Modelling Processes: NISTIR 5910, 1996, National Institute of Standards and Technology, Gaithersburg, MD.
[8] – 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.
[9] – Uschold, M. and Gruninger M., Ontologies: Principles, Methods, and Applications, Knowledge Engineering Review, 1996, Vol. 11, pp. 96-137.

Index