Guide to Use of Industry Foundation Classes in Exchange of Reinforcement Models

Reported by ACI Committee 131

ACI 131.2R-17

American Concrete Institute
Always advancing

First Printing
October 2017
ISBN: 978-1-945487-81-1

Guide to Use of Industry Foundation Classes in Exchange of Reinforcement Models

Order information: ACI documents are available in print, by download, on CD-ROM, through electronic subscription, or reprint and may be obtained by contacting ACI.

Most ACI standards and committee reports are gathered together in the annually revised ACI Manual of Concrete Practice (MCP).

American Concrete Institute
38800 Country Club Drive
Farmington Hills, MI 48331
Phone: +1.248.848.3700
Fax: +1.248.848.3701

www.concrete.org

ACI 131.2R-17

Guide to Use of Industry Foundation Classes in Exchange of Reinforcement Models

Reported by ACI Committee 131

Christopher D. Brown, Chair

Allan P. Bommer, Secretary

Mark Douglas Agee
Daniel D. Berend
Richard H. Birley
Daniel Bittrich
Brady G. Buckley
Barry B. Butler
Peter J. Carrato
Jeffrey N. Cochrane
James T. Davy
Charles M. Eastman
Dennis J. Fontenot
Sidney Freedman
David A. Grundler Jr.
Michael Gustafson
Michael G. Hernandez
Harrison Rolfe Jennings
Julian Kang
William M. Klorman
Andrew R. Lloyd
Ronald L. O'Kane
Andrew Pinneke
Joseph C. Sanders
William J. Shebetka
Carl Taylor
John B. Turner
Peter Zdgiebloski

Special acknowledgement and thanks to D. Yang, Research Scientist at the Digital Building Laboratory in the Georgia Institute of Technology, for his contributions to this guide.

This guide provides a protocol for the exchange of data related to reinforcing steel between software applications. This guide presents a human-readable list of reinforcing steel entities, attributes, property sets, and relationships, with sufficient specificity so that the format and syntax for machine-readable exchanges based on Industry Foundation Classes (IFC) can be employed, enhanced, or developed. This specific set of exchange requirements is referred to as a model view definition (MVD). Material and geometric attributes, property sets, and relationships, both required and optional, that address most reinforced concrete applications for buildings and nonbuilding structures are presented. This guide is intended to be used by building information modeling (BIM) software developers to assist in the development of consistent and accurate exchanges of reinforcing steel information between applications.

Keywords: attribute; building information modeling; model view definition; Industry Foundation Classes; reinforcing steel.

CONTENTS

ACI 131.2R-17 was adopted and published October 2017.

CHAPTER 1—INTRODUCTION AND SCOPE

1.1—Introduction

This guide provides a model view definition (MVD) that describes both minimum and optional exchange requirements for concrete reinforcement models through the use of Industry Foundation Classes (IFC). The MVD is intended to be used by software developers to create interoperable applications that will allow reinforcement detailers, reinforcement fabricators, and others to exchange detailed reinforcement models between all participants in the concrete construction supply chain. The terminology and concepts used in this guide are those of the intended audience: software developers creating and modifying software applications that produce or consume reinforcement models.

The National BIM Standard-United States (NBIMS-US) (NIBS 2007) defines standard and efficient terminology and semantics to be exchanged in building information models to support various business-use cases throughout architecture, engineering, construction, and operations projects. The project committee responsible for developing the NBIMS-US is a committee of the buildingSMART alliance, a council of the National Institute of Building Sciences (NIBS). The NBIMS-US establishes the standard process to develop the NIBS standard. The development process includes four phases:

1. *Program*—Defines information exchange requirements that may be standardized by developing process models and defining specifications and business rules for each exchange. An example of an information exchange is the transfer of data in context between various entities along the concrete supply chain—for example, from the architect to the structural engineer. In this phase, a process model that identifies the required tasks, the information exchanges that take place in the project lifecycle, the actors (those entities such as engineers and reinforcing bar detailers who develop or use information), and the software applications that are the senders and recipients of these exchanges, is developed. The information exchanges are defined by exchange models, which specify the functional requirements (content) of data exchanges to be implemented. When the process models and exchange models are combined, they form an information delivery manual (IDM). This IDM serves as the overall functional requirements specification for one or more exchanges.
2. *Design*—Develops exchange requirement models and qualitative MVD.
3. *Construct*—Develops software implementation specifications for MVD and facilitates product testing and certification of information exchanges.
4. *Deploy*—Provides generic and product-specific building information modeling (BIM) guides, validates data exchange, and extends the complexity of information that can be included in the BIM data.

ACI 131.1R addressed portions of the first and second of these four phases, Program and Design, by establishing a flowchart (an IDM) for data exchange across the concrete design and supply chain. This document addresses portions of the second and third of these phases, Design and Construct, for the reinforcing steel portion of concrete data exchanges. Entities, attributes, property sets, and relationships, both required and optional, are presented that can be used to employ, enhance, and develop data exchange standards. These exchange requirements make up the MVD for reinforcing steel.

This guide is organized as follows. Chapter 3 reviews the IFC concepts upon which the MVD has been developed. Chapter 4 describes modeling concepts that are used for all information exchanged. Chapter 5 describes the modeling of projects, sites and buildings. Chapters 6, 7, and 8 describe modeling of reinforcement materials, reinforcing bars, and welded-wire reinforcement, respectively. Chapters 9, 10, and 11 describe modeling of bar couplers, bar terminators, and other reinforcement accessories, respectively. Chapters 12 and 13 describe grouping of bars into callouts, cages, bundles, and releases. Appendix A shows standard IFC instance diagrams for the exchange, and Appendix B provides examples of the MVD in use.

1.2—Scope

This guide is intended to be used in conjunction with IFC 2x4 (IFC4), Addendum 1 (buildingSMART International 2015). This guide provides human-readable description of entities, attributes, property sets, and relationships for reinforcing steel used in buildings and nonbuilding structures, that can be used by software developers to prepare machine-readable data exchange protocols. Geometry, position, attributes, and other information are provided for reinforcing bars, welded-wire reinforcement, bar couplers, bar terminators, and other items that are commonly included in reinforcement models. In addition, information related to the bundling of bars for shipment and the tracking of releases is included.

This guide describes the MVD that is used in the exchange of reinforcement information, but does not provide detailed descriptions of IFC syntax and modeling requirements. buildingSMART International (2013) provides that information.

A level of development (LOD) specification (BIMForum 2015; American Institute of Architects 2013) is a reference that enables practitioners in the architecture, engineering, and construction industry to specify and articulate, with a high level of clarity, the content and reliability of building information models at various stages in the design and construction process. Data exchanges described in this guide vary from preliminary modeling of reinforcing steel to fabrication-ready information to drive computer-controlled shop equipment. This range of LODs supported by this exchange is from 300 to 400, described as follows.

a) LOD 300: The model element is graphically represented within the model as a specific system, object, or assembly in terms of quantity, size, shape, location, and orientation. Nongraphic information may also be attached to the model element.

b) LOD 350: In addition to the information provided by LOD 300, interfaces with other building systems are provided for the modeled element.

c) LOD 400: In addition to the information provided by LOD 350, detailing, fabrication, assembly, and installation information are provided for the modeled element.

CHAPTER 2—DEFINITIONS

Many IFC terms, such as IfcRoot and IfcPositiveLengthMeasure, are used in this guide. These terms all have detailed definitions that can be found in the official IFC4 documentation (buildingSMART International 2013).

ACI provides a comprehensive list of definitions through an online resource, “ACI Concrete Terminology”, https://www.concrete.org/store/productdetail.aspx?ItemID=CT13. Definitions provided herein complement that source.

bundle—a set of reinforcing bars tied or otherwise packaged together to facilitate shipping and related logistics.

cage—a rigid assembly of reinforcement ready for placing in position.

callout—placing drawing label describing the requirements for a bar or group of bars at an individual location.

Industry Foundation Classes—platform-neutral open data model for construction and facilities management developed by buildingSMART; published as ISO 16739.

information delivery manual—documentation that captures the business process and includes detailed specifications of the information that a user fulfilling a particular role would need to provide at a particular point within a project.

model view definition—formal subset of Industry Foundation Classes designed to satisfy particular data exchange requirements, typically as defined in an information delivery manual.

release—set of reinforcement and accessories, typically of a specific area of a project, that is approved for fabrication and delivery.

template—definition that is intended to be applied to multiple instances of an item.

CHAPTER 3—REVIEW OF INDUSTRY FOUNDATION CLASSES CONCEPTS

3.1—Industry Foundation Classes

Industry Foundation Classes (IFC) is a vendor-neutral data model that is used for exchanging and sharing information among various participants in a building construction or facility management project. This chapter briefly describes the core concepts of the IFC data model. Full descriptions of IFC are available through buildingSMART International (2015).

A model view definition (MVD) restricts the usage of IFC to a predictable subset to facilitate dependable and efficient exchanges of IFC data in a specified use case. Without the constraints applied by MVDs, it is difficult to write fully compatible software applications.

IFC data can be conceptually considered as a hierarchy, as shown in Fig 3.1.

3.2—Entities and occurrences

Entity data types are the primary data structures of Industry Foundation Classes (IFC). An entity defines a type of physical or conceptual item that may or may not exist in the real world. It is a named data structure that corresponds closely to a “class” in most object-oriented programming languages (with the exception of the lack of “methods” in an entity). In object-

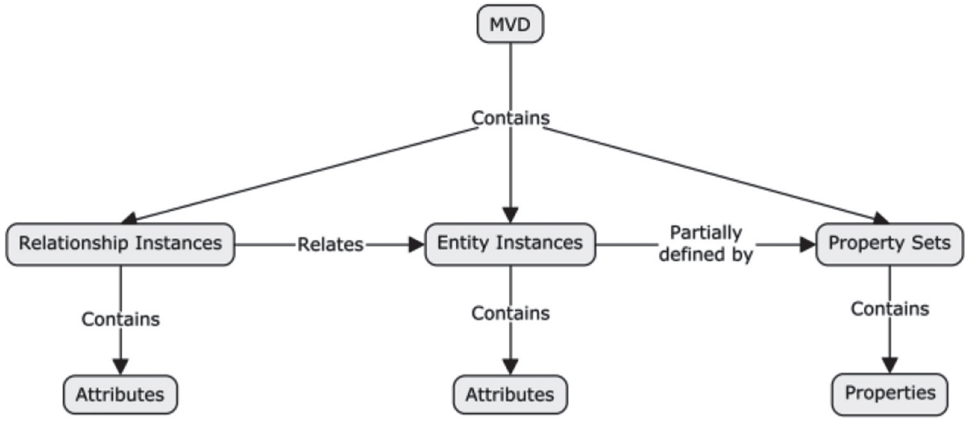


Fig. 3.1—Simplified view of IFC data concepts.

oriented programming, methods are procedures or functions that are defined as part of a class. Examples of entities are IfcReinforcingBar, IfcRelAggregates, IfcMaterial, IfcCurve, and IfcMechanicalFastener. Over 700 entities are defined in IFC. All IFC entities' names start with the "Ifc" prefix.

Entities can (and almost always do) inherit characteristics from other entities; for example, IfcReinforcingBar inherits from ("is a subtype of" in IFC terminology) IfcReinforcingElement and, hence, has all the characteristics of IfcReinforcingElement in addition to the characteristics it explicitly defines. Inheritance is widely used in IFC to identify commonalities in entities and reduce duplication of definitions. The use of inheritance in IFC creates a rich hierarchy of entities.

An entity data structure is defined once in the IFC model, but can be used an unlimited number of times. For example, there is one definition of IfcReinforcingBar, but that definition can be used to represent any and all actual reinforcing bars. Each usage (representing a single item) is referred to as an "instance." Commonly, the entity name is used to refer to the instance; for example, "each IfcReinforcingBar may be related to an IfcGroup."

Model view definitions (MVDs) such as the one described in this guide cannot define new IFC entities, but can choose which entities are valid in an exchange and can constrain some aspects of their usage.

3.3—Attributes

Some characteristics of entities are defined through attributes. Attributes are defined directly in the entity definitions. For example, the IfcRoot entity has the following attributes:

a) GlobalId—a globally unique identifier

b) OwnerHistory—information relating to the current owner and last modifier of the object

c) Name—human-readable label

d) Description—human-readable description

Attributes may be thought of as properties, but the term "properties" is not used to avoid confusion with property sets, which are described in the following section. Attributes have clearly specified data types that are sometimes more restrictive than the data types available in programming languages; for example, IfcPositiveLengthMeasure is an attribute type that limits its values to lengths greater than zero. Each instance has its own set of values for the attributes. Attributes are one of the types of characteristics that are inherited in the IFC entity hierarchy.

Model view definitions (MVDs) such as the one described in this guide cannot define new attributes, but can choose which attributes are valid in an exchange and can constrain an attribute's usage.

3.4—Property sets

Property sets are the extensibility mechanism provided by IFC for dynamically defining properties and their values. Each property set defines one or more properties with a name and a value type. Property sets are implemented through the IfcPropertySet entity. IfcPropertySet is related to entities by the IfcRelDefinesByProperties relationship. By convention, property sets have a name beginning with "Pset_."

Model view definitions (MVDs) such as the one described in this guide can define new property sets and specify how they are used. Many property sets are defined in this exchange.

3.5—Relationships

Relationships are IFC entities that have the purpose of relating entities to each other and defining the characteristics of their relationship. Relationships can either be 1-to-1 (1 entity related to 1 entity) or 1-to-many (1 entity related to multiple entities). Relationships entity names begin with "IfcRel". There are also ternary relationships such as ifcRelConnectsWithRealizingElements.

As is the case with other entities, model view definitions (MVDs) such as the one described in this guide cannot define new relationships, but can choose which relationships are valid in an exchange and can constrain some aspects of their usage.

3.6—Entity diagrams

In this document, EXPRESS-G (ISO 10303-11) diagrams are used to illustrate the relationships between IFC entities. Figure 3.6 shows an example of these diagrams; to navigate from From_Entity_Name to To_Entity_Name, Attribute_Name is used.

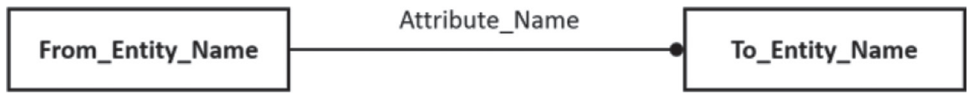


Fig. 3.6—EXPRESS-G diagram.

Table 4.2—IfcRoot attribute usage

Attribute name	Type	Usage	Required or optional
GlobalId	IfcGloballyUniqueId	Unique identifier	Required
OwnerHistory	IfcOwnerHistory	Ownership of object	Optional
Name	IfcLabel*	Human-readable label	Optional
Description	IfcText*	Human-readable description	Optional

*The IfcLabel and IfcText types are referenced frequently in this guide. The difference between them is primarily usage. IfcLabel is intended to be used as a human-readable identifier whereas IfcText is intended for other human-readable information.

Table 4.3—Pset_ACI_ItemStatus Usage

Property name	Type	Usage	Required or optional
LOD	IfcInteger	The level of development of this item, according to BIMForum (2015)	Optional
Status	IfcLabel	Workflow status of the modeled item, such as "For Fabrication"	Optional
StatusDate	IfcDate	Date the status was determined	Optional

CHAPTER 4—COMMON MODELING REQUIREMENTS

4.1—General

Modeling approaches that are common to most entities in this exchange are described in this chapter. Section 4.2 describes the usage of attributes common to all items, 4.3 describes the usage of property sets common to all elements, 4.4 describes the geometric placement of physical elements, and 4.5 describes the use of products and types.

4.2—IfcRoot attributes

All items modeled in this exchange are represented using entities that are subtypes of IfcRoot. Table 4.2 defines the usage of attributes defined in IfcRoot. IfcRoot attributes are part of every item in the exchange and are therefore not repeated or discussed further in the item-specific chapters that follow.

4.3—Status and level of development

All IfcElements, such as IfcReinforcingBar, IfcReinforcingMesh, IfcMechanicalFastener, and IfcDiscreteAccessory, modeled in this exchange may use the property set Pset_ACI_ItemStatus to provide information related to the element's level of development (LOD) and status. Table 4.3 defines the usage of properties in Pset_ACI_ItemStatus.

For model-size efficiency, it is assumed that this property set will be widely shared by elements in the model with identical status information, but sharing is not required.

4.4—IfcProduct, ObjectPlacement, and IfcRelContainedInSpatialStructure

All physical components modeled in this exchange are represented using entities that are subtypes of IfcProduct. These components are positioned relative to either the IfcSite ("world coordinates") or an IfcBuilding, if any. All items are positioned relative to the IfcSite with one limited exception; all items contained in an IfcBuilding are positioned relative to the same reference and may be positioned relative to the IfcBuilding that contains them. The positioning is defined using the ObjectPlacement attribute of IfcProduct. Placement relative to the IfcSite is shown in Fig. 4.4a; placement relative to an IfcBuilding is shown in Fig. 4.4b.

In addition to relative positioning, IFC supports the concept of containment to allow a hierarchy of spatial regions that contain items. This exchange uses the IfcRelContainedInSpatialStructure relationship to define the containment and the IfcSite and IfcBuilding entities to define the spatial zones. An IfcProduct can either be contained in the IfcSite (as shown in Fig. 4.4a) or contained in an IfcBuilding (as shown in Fig. 4.4b).

4.5—IfcTypeProduct and IfcProduct

All physical items included in this exchange are represented with IfcProduct subtypes (such as IfcReinforcingBar) that reference IfcTypeProduct subtypes (such as IfcReinforcingBarType) to define their primary attributes and geometry. IfcTypeProduct subtypes act as templates. This modeling strategy clearly identifies commonalities, reduces duplicate information, and leads to smaller IFC files. IfcRelDefinesByType defines the primary relationship between the product and the type, as shown in Fig. 4.5.

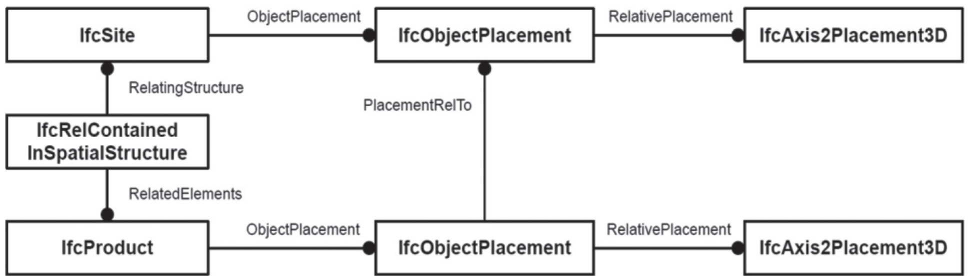


Fig. 4.4a—Placement and containment relative to IfcSite.

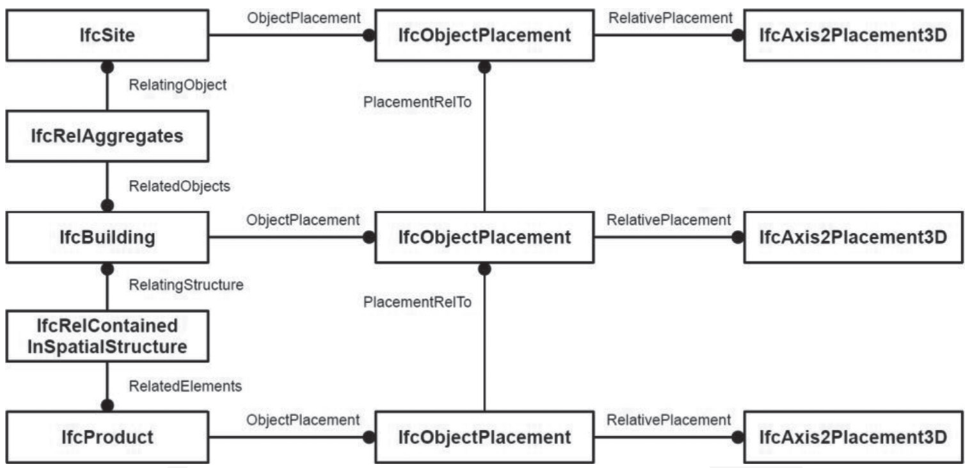


Fig. 4.4b—Placement and containment relative to IfcBuilding.

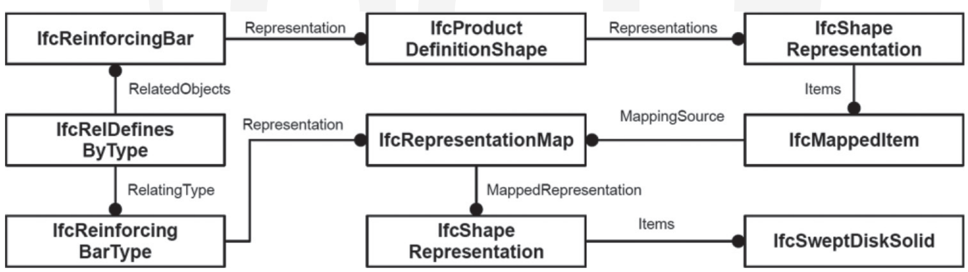


Fig. 4.5—IfcProduct-IfcTypeProduct relationships (using IfcReinforcingBar as example).

The geometry of the IfcTypeProduct is defined through an IfcRepresentationMap and an IfcRepresentation with the RepresentationType of "body". The geometry of the IfcProduct is defined through an IfcProductDefinitionShape, an IfcShapeRepresentation, and an IfcMappedItem. The relationship of these entities is shown in Fig. 4.5.

CHAPTER 5—PROJECT, SITE, AND BUILDING

5.1—Conceptual modeling

In the model exchange described in this guide, IfcProject, IfcSite, and IfcBuilding provide the hierarchical and spatial organization of the model. There is exactly one IfcProject and exactly one IfcSite. There may be zero or more IfcBuildings. The relationships of IfcProject and IfcSite are shown in Fig. 5.1a and 5.1b and described in Table 5.1a.

If one or more instances of IfcBuilding exist, their relationship to IfcSite is as shown in Fig. 5.1b and described in Table 5.1b. The relationship between IfcSite and IfcBuilding has also been discussed in 4.3.

5.2—IfcProject

IfcProject is used to represent the context in which the information is being exchanged, typically a construction project. There is always exactly one IfcProject instance in any IFC model. In addition to the attributes described in 4.2, the attributes defined in Table 5.2 are used for IfcProject.

IfcUnitAssignment of UnitsInContext is a collection of IfcUnit. Within that collection, IfcUnits defining the following IfcUnitEnum values are included:

a) LENGTHUNIT
b) MASSUNIT
c) PLANEANGLEUNIT
d) AREAUNIT
e) FORCEUNIT
f) PRESSUREUNIT

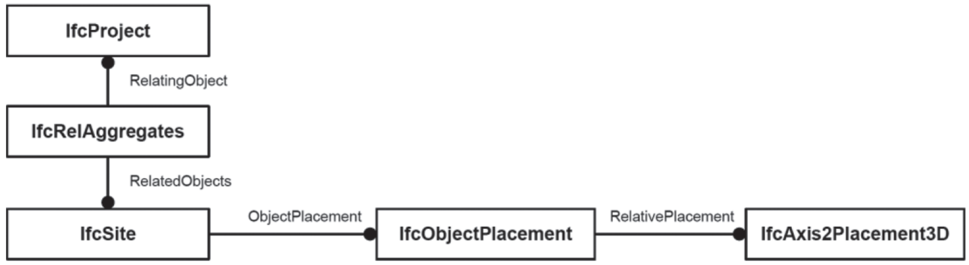


Fig. 5.1a—Spatial composition of IfcProject and IfcSite.

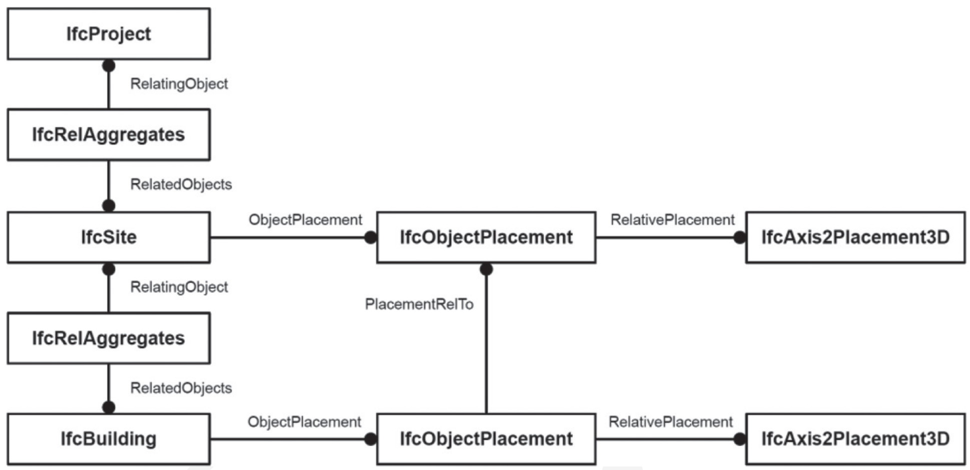


Fig. 5.1b—Spatial composition of IfcProject, IfcSite and IfcBuilding.

Table 5.1a—IfcProject-IfcSite relationship (IfcRelAggregates)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcSite	[1:?]	Required
RelatingObject	IfcProject	[1:1]	Required

Table 5.1b—IfcSite-IfcBuilding relationship (IfcRelAggregates)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcBuilding	[1:?]	Required
RelatingObject	IfcSite	[1:1]	Required

Table 5.2—IfcProject attribute usage

Attribute name	Type	Usage	Required or optional
UnitsInContext	IfcUnitAssignment	Specifies the units used in the model	Required

Table 5.3—IfcSite attribute usage

Attribute name	Type	Usage	Required or optional
CompositionType	IfcElementCompositionEnum	Always ELEMENT	Required

There are no property sets used with IfcProject.

5.3—IfcSite

IfcSite is used to represent an area of land where a project is located. There is always exactly one IfcSite instance in any IFC model for this exchange. In addition to the attributes described in 4.2, the attributes defined in Table 5.3 are used for IfcSite. There are no property sets used with IfcSite.

5.4—IfcBuilding

IfcBuilding is used to represent a structure. There may be zero or more instances of IfcBuilding in any IFC model for this exchange. In addition to the attributes described in 4.2, the attributes defined in Table 5.4 are used for IfcBuilding. There are no property sets used with IfcBuilding. There is no geometric representation of the building.

Table 5.4—IfcBuilding attribute usage

Attribute name	Type	Usage	Required or optional
CompositionType	IfcElementCompositionEnum	Always ELEMENT	Required

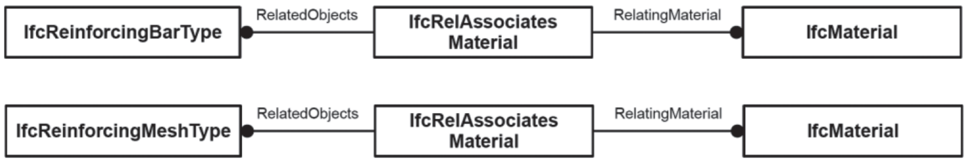


Fig. 6.1—Material relationships.

Table 6.1a—Reinforcing bar-material relationship (IfcRelAssociatesMaterial)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingBarType	[1:?]	Required
RelatingMaterial	IfcMaterial	[1:1]	Required

Table 6.1b—Reinforcing mesh-material relationship (IfcRelAssociatesMaterial)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingMeshType	[1:?]	Required
RelatingMaterial	IfcMaterial	[1:1]	Required

CHAPTER 6—REINFORCEMENT MATERIAL MODELING

6.1—Conceptual modeling

In the model exchange described in this guide, materials are defined only for reinforcing bars and for welded wire reinforcement, as shown in Fig. 6.1 and described in Tables 6.1a and 6.1b.

Typically, each IfcMaterial will define the material and coating for many IfcReinforcingBarTypes or IfcReinforcingMeshTypes through the IfcRelAssociatesMaterial relationship. IfcReinforcingBarType is described in Chapter 7. IfcReinforcingMeshType is described in Chapter 8.

6.2—IfcMaterial

IfcMaterial is used to represent a steel material and optional coating used for reinforcing bars and reinforcing mesh. IfcMaterial has no attributes other than those defined in 4.2. The property set Pset_ACI_ReinforcingMaterial is needed to provide additional information regarding the IfcMaterial. Table 6.2 defines the properties available in Pset_ACI_ReinforcingMaterial.

Table 6.2—Pset_ACI_ReinforcingMaterial property set usage

Property name	Type	Usage	Required or optional
Specification	IfcLabel	The specification under which the material is produced. Standard names for ASTM materials use this pattern: ASTM_A615, ASTM_A706.	Required
SpecificationVersion	IfcLabel	The year or other version number for the specification. For ASTM documents, values such as "2016" and "2016A" are used.	Required
Grade	IfcPressureMeasure	Yield strength as defined in the specification	Required
Subtype	IfcLabel	Material subtype (if any) in the specification. For ASTM specs, this is currently only used for stainless steel and low-carbon chromium steel. For ASTM_A955, use alloy names such as 316.	Optional
CoatingSpecification	IfcLabel	The specification under which the coating material is produced. Standard names for ASTM materials use this pattern: ASTM_A775. Use UNCOATED when no coating is used. In rare case of proprietary coating, this field may be used to specify the coating.	Required
CoatingSpecificatioVersion	IfcLabel	The year or other version number for the coating specification. For ASTM documents, values such as "2016" and "2016A" are used.	Required if CoatingSpecification is not UNCOATED
CoatingSubtype	IfcLabel	Coating subtype (if any) in the coating specification. For ASTM_A767 mass of zinc coating use: CLASS_I or CLASS_II	Optional
CoatedBeforeFabrication	IfcBoolean	True or false	Required if CoatingSpecification is not UNCOATED
RequiredOrigin	IfcLabel	Location of material origin. For countries, use the two-letter ISO 3166-1 alpha-2 names (example, US for United States)	Required (if project requires)

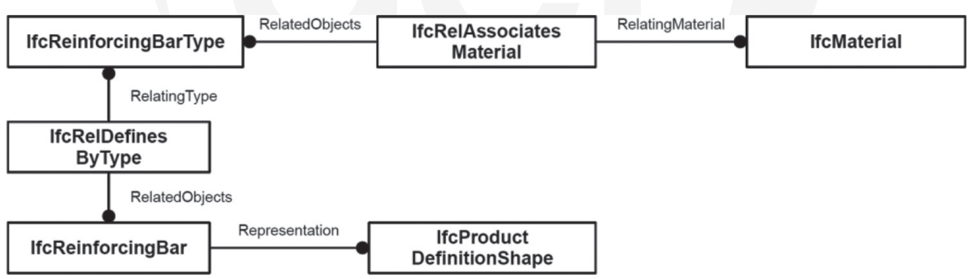


Fig. 7.1—Reinforcing bar relationships.

Table 7.1a—Bar type-material relationship (IfcRelAssociatesMaterial)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingBarType	[1:?]	Required
RelatingMaterial	IfcMaterial	[1:1]	Required

Table 7.1b—Reinforcing bar-bar type relationship (IfcRelDefinesByType)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingBar	[1:?]	Required
RelatingType	IfcReinforcingBarType	[1:1]	Required

CHAPTER 7—REINFORCING BAR MODELING

7.1—Conceptual modeling

In the model exchange described in this guide, reinforcing bars are modeled using a combination of IfcMaterial, IfcReinforcingBarType, and IfcReinforcingBar, as shown in Fig. 7.1 and described in Tables 7.1a and 7.1b.

IfcMaterial is used to define a reinforcing bar coating and steel material, as described in Chapter 6. Typically, each IfcMaterial will define the material and coating for many

Table 7.2a—IfcReinforcingBarType attribute usage

Attribute name	Type	Usage	Required or optional
PredefinedType	IfcReinforcingElementTypeEnum	Always NOTDEFINED	Required
NominalDiameter	IfcPositiveLengthMeasure	Nominal bar diameter	Required
CrossSectionArea	IfcAreaMeasure	Nominal bar area	Required
BarLength	IfcPositiveLengthMeasure	"Theoretical length" calculated per CRSI (2009) standards*	Required
BarSurface	IfcReinforcingBarSurfaceEnum	Always PLAIN or TEXTURED†	Required
BendingShapeCode‡	IfcLabel	Shape code name per project or regional standard	§
BendingParameters‡	[1:?] IfcBendingParameterSelect	Array of lengths, angles, or both	Optional

*Refer to Fig. 7.2a.

†While the term deformed is commonly used in North America, TEXTURED is the proper value for data of the type IfcReinforcingBarSurfaceEnum.

‡The BendingShapeCode and BendingParameters attributes are not the primary mechanism for transferring bending information, as they lack the ability to name the parameters and, hence, optional parameters cannot be skipped. ACI 315 bending shapes typically have many optional parameters.

§If the BendingParameters is provided, then the BendingShapeCode is also provided.

Table 7.2b—Pset_ACI_ReinforcingBarType property set usage

Property name	Type	Usage	Required or optional
Size	IfcLabel	Bar name per local standards Bar numbers (#4, #5, etc.) are used where appropriate for the location.	Required
BarMark	IfcLabel	Unique human-readable name as used to identify the bar type.	Optional
BarMass	IfcMassMeasure	Total mass of bar, consistent with BarLength attribute.	Optional
StartEndPrep*†	IfcLabel	Preparation required at beginning of bar to accommodate couplers, terminators, or other accessories. Also used for bar end treatments such as forged heads.	Required if special prep is needed
EndEndPrep*†	IfcLabel	Preparation required at end of bar to accommodate couplers, terminators, or other accessories. Also used for bar end treatments such as forged heads.	Required if special prep is needed

*Standard end prep values are listed in Table 7.2c. Proprietary names may be used for proprietary processes.

†"Start" and "End" are defined by the Directrix in the IfcSweptDiskSolid defining the bar shape.

IfcReinforcingBarTypes through the IfcRelAssociatesMaterial relationship.

IfcReinforcingBarType is used to define template bars with a fixed size and shape. Through its IfcRelAssociatesMaterial relationship with IfcMaterial, the material of the IfcReinforcingBarType is defined. Typically, each IfcReinforcingBarType will define the shape, size, material, and coating for many IfcReinforcingBars through the IfcRelDefinesByType relationship. IfcReinforcingBar is used to define individual bars in specific positions. It is mostly defined through its IfcRelDefinesByType relationship to IfcReinforcingBarType.

7.2—IfcReinforcingBarType

IfcReinforcingBarType is used to define a template for bars with the same size, shape, and material. In addition to the attributes described in 4.2, the attributes defined in Table 7.2a are used for IfcReinforcingBarType.

The property set Pset_ACI_ReinforcingBarType is used to provide additional information regarding the IfcReinforcingBarType. Table 7.2b defines the properties available in Pset_ACI_ReinforcingBarType.

The property set Pset_ACI_BarShape may be used to provide additional information regarding the shape of the IfcReinforcingBarType. Table 7.2d defines the properties available in Pset_ACI_BarShape.

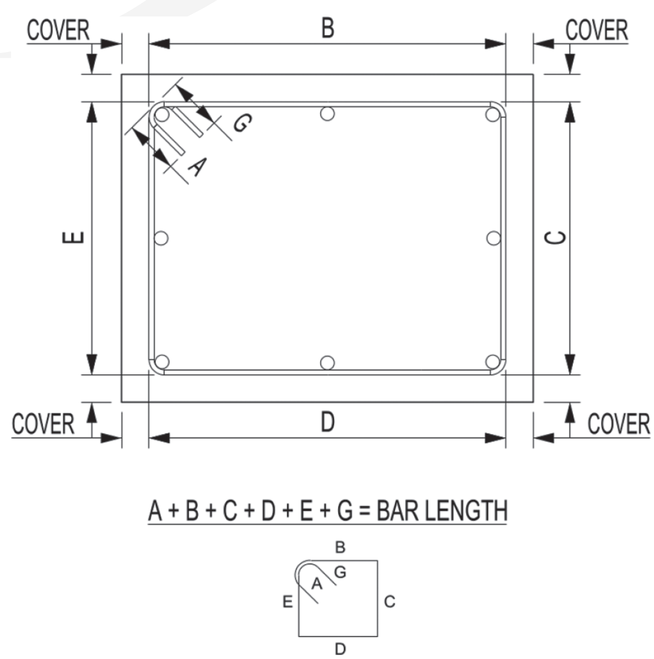


Fig. 7.2a—BarLength calculation.

Table 7.2c—Standard values for StartEndPrep and EndEndPrep

Name	Meaning
SHEAR_CUT	The most common method for cutting reinforcing bar. Uses a technique that shears the ends of the bars and does not produce clean cut end.
SAW_CUT	In this method, a saw is used to cut the bar instead of a shear. It produces a smooth, flat end on the bar. It is sometimes required for threading operations and for smooth dowels that are to be used in expansion joints.
STRAIGHT_THREAD	A standard thread used for attaching mechanical couplers and terminators is applied to the end of the bar
TAPERED_THREAD	A special, usually proprietary, tapered thread used for attaching mechanical couplers and terminators is applied to the end of the bar
SAW_CUT_STRAIGHT_THREAD	SAW_CUT combined with STRAIGHT_THREAD Some threading operations require a smooth end.
SAW_CUT_TAPERED_THREAD	SAW_CUT combined with TAPERED_THREAD Some threading operations require a smooth end.
FORGED_HEAD	End of bar is forged to provide terminator at end of bar.
FORGED_COUPLER_HEAD	End of bar is forged to provide head for coupler attachment.

Table 7.2d—Pset_ACI_BarShape property set usage

Property name	Type	Usage	Required or optional
StandardName	IfcLabel	Detailing standard identifier, such as ACI_315 BS_4466 BS_8666 RSIC	Required
StandardVersion	IfcLabel	Year of standard or other version identifier. Valid values for ACI_315: 1999 Valid values for BS_4466: 1989 Valid values for BS_8666: 2000, 2005	Required
ShapeName	IfcLabel	Name of shape per the standard	Required
DefaultInsideBendRadius	IfcPositiveLengthMeasure	Inside radius of bar bends unless otherwise specified by shape	Required
Parameters*	IfcPropertyTableValue	The parameter names are specified in the DefiningValues field. The values are specified in the DefinedValues field, using: a) IfcLengthMeasure for distances b) IfcPlaneAngleMeasure for angles c) IfcInteger for integer values d) IfcReal for nondimensional real values	Required

*Typical ACI 315 bar parameters are illustrated in Fig. 7.2b.

Table 7.2e—MappedRepresentation for IfcReinforcingBarType

Geometry type	Usage	Required or optional
IfcSweptDiskSolid	For bars	Required
IfcBooleanResult*†	For forged heads	Required
IfcBooleanResult†‡	For thread tapers	Required

*IfcBooleanResult operation is a union.

‡IfcBooleanResult operation is a difference.

†Refer to Table 7.2f for geometry modeling of SecondOperand.

The geometry of the IfcReinforcingBarType is provided by an IfcRepresentationMap set in the RepresentationMaps attribute. The options for the MappedRepresentation attribute of the IfcRepresentationMap are listed in Table 7.2e.

When a single forged head or thread taper is present, a single Boolean operation will be required. The IfcBooleanResult should have the IfcSweptDiskSolid (representing the bar) as the FirstOperand. IfcBooleanResult is the geometric result of one three-dimensional (3D) volume being intersected with, added to, or subtracted from another 3D volume.

When multiple Boolean operations are needed (such as for thread tapers on both ends), the IfcBooleanResults will be nested in a manner that results in the bar's original IfcSweptDiskSolid representation being accessible through the recurring FirstOperands of the nested IfcBooleanResults, as shown in Fig. 7.2c through 7.2f.

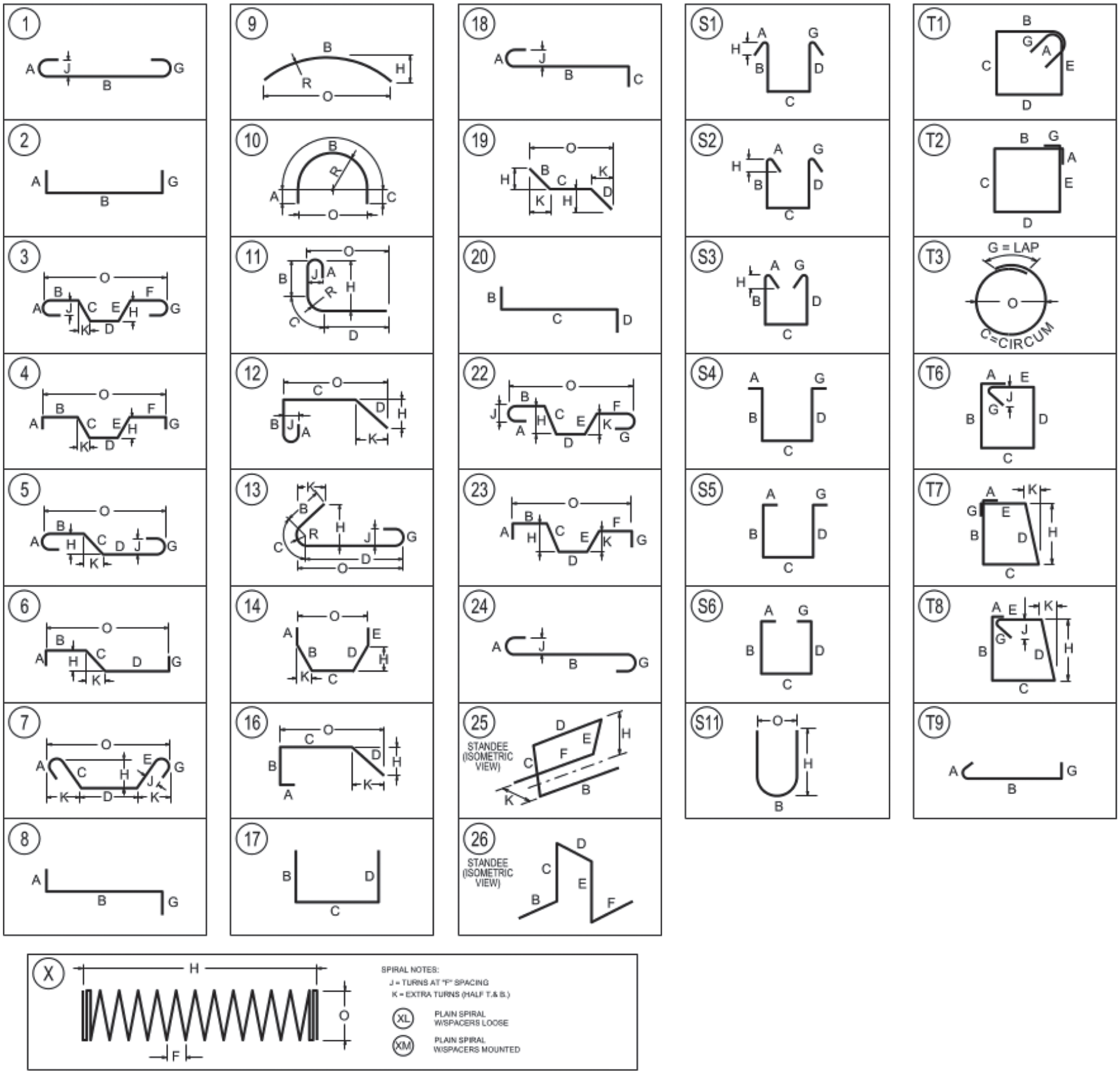


Fig. 7.2b—Pset_ACI_BarShape parameters.

Table 7.2f—Generic geometry for forged heads and tapers

Feature	Geometry type	Dimensions
Forged head	IfcRightCircularCylinder	Height = 0.75 bar diameter Radius = 1.15 bar diameter
Thread taper	IfcRevolvedAreaSolid of IfcArbitraryClosedProfileDef	Angle = 2π Profile is triangle with 6-degree slope Height = 1.5 × bar diameter

7.3—IfcReinforcingBar

IfcReinforcingBar is used to represent individual bars. There is a 1:1 correspondence between a real bar and an IfcReinforcingBar. The geometry of the bar and many of its characteristics are defined via the bar's relationship to IfcReinforcingBarType, as shown in 4.4 and 6.1. IfcReinforcingBars are positioned in space using the ObjectPlacement described in 4.3. In addition to the attributes described in 4.2, the attributes defined in Table 7.3a are used for IfcReinforcingBar.

It is important to note that the attributes NominalDiameter, CrossSectionArea, BarLength, and BarSurface are

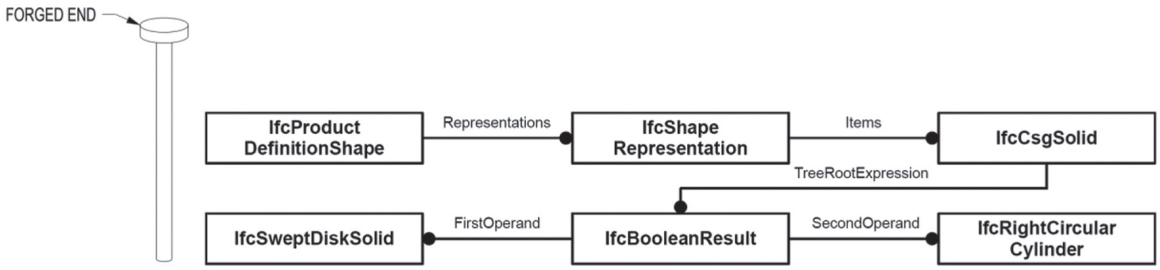


Fig. 7.2c—Representing bar with forged head with IfcBooleanResult.

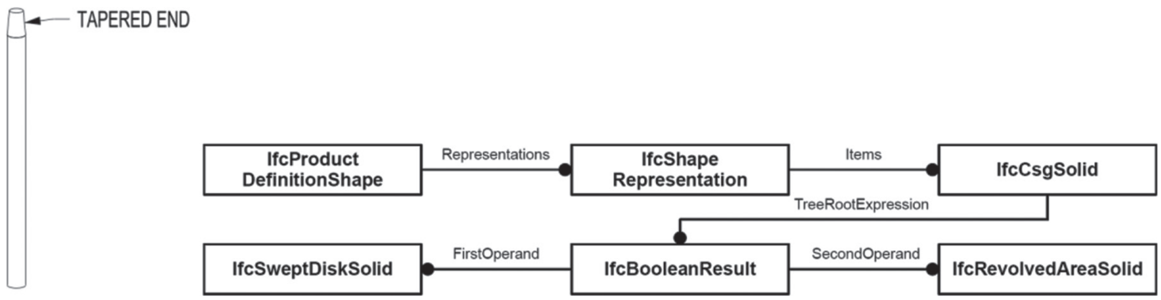


Fig. 7.2d—Representing bar with tapered end with IfcBooleanResult.

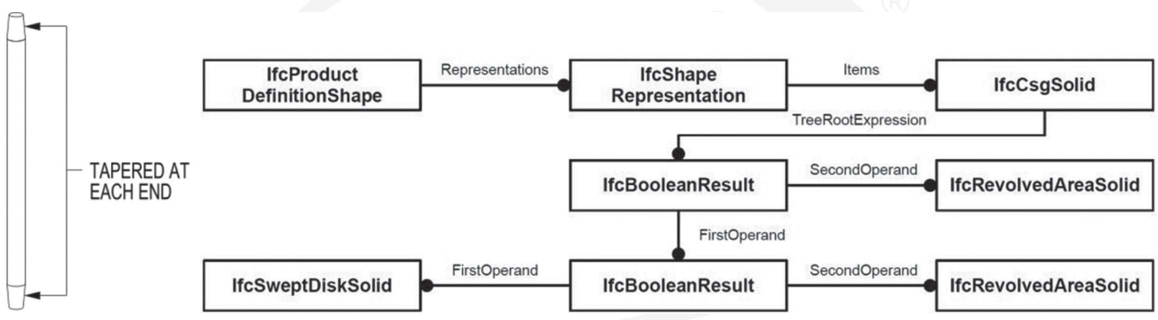


Fig. 7.2e—Representing bar with two tapered ends with IfcBooleanResult.

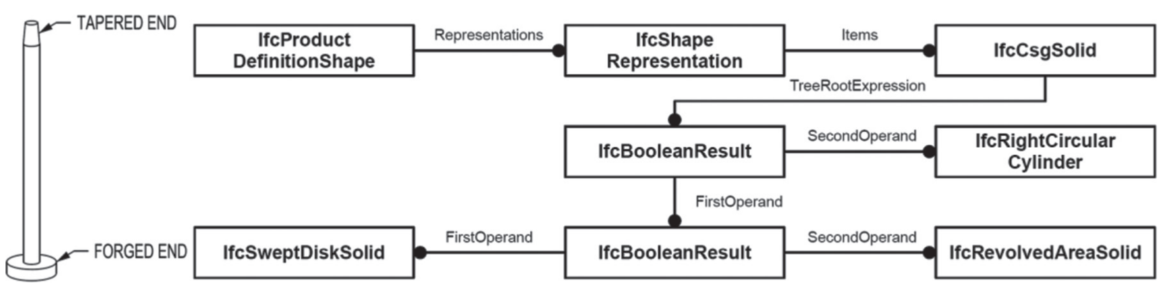


Fig. 7.2f—Representing bar with forged head and tapered end with IfcBooleanResult.

Table 7.3a—IfcReinforcingBar attribute usage

Attribute name	Type	Usage	Required or optional
ObjectType	IfcLabel	Always STRUCTURAL or NONSTRUCTURAL	Required
PredefinedType	IfcReinforcingElementTypeEnum	Always USERDEFINED	Required

Table 7.3b—Pset_ACI_ReinforcingBar property set usage

Property name	Type	Usage	Required or optional
BarElement	IfcLabel	Always one of the following: BEAM COLUMN CORBEL FOOTING PIER PILE_CAP (applies to pier caps also) SLAB WALL Refer to Table 7.3c for usage details.	Optional
BarUse	IfcLabel	Always one of the following: CORNER DOWEL HORIZONTAL LONGITUDINAL STIRRUP SUPPORT TIE TRANSVERSE TRIM VERTICAL Refer to Table 7.3d for usage details.	Optional
BarPosition	IfcLabel	May be user-defined or one of the following standard values: BOTTOM INSIDE CENTER OUTSIDE SIDE TOP Refer to Table 7.3e for usage details.	Optional

Table 7.3c—BarElement property usage

Value	Meaning
BEAM	The bar is in a beam, girder, or similar element
COLUMN	The bar is in a column or similar element
CORBEL	The bar is in a corbel or similar element
FOOTING	The bar is in a footing or similar element
PIER	The bar is in a pier, caisson, or similar element
PILE_CAP	The bar is in pile cap, pier cap, or similar element
SLAB	The bar is in a slab, ramp, or similar element
WALL	The bar is in a wall or similar element

never used, as those attributes are instead defined via the relationship to an IfcReinforcingBarType.

In addition to the property set Pset_ACI_Status described in 4.3, the property set Pset_ACI_ReinforcingBar is available to provide additional information regarding the IfcReinforcingBar. Table 7.3b defines the properties available in Pset_ACI_ReinforcingBar. Tables 7.3c to 7.3e describes the usage of the BarElement, BarUse, and BarPosition properties, respectively; Fig. 7.3a to 7.3f assist in determining the appropriate usage of these properties.

Table 7.3d—BarUse property usage

Value	Meaning
CORNER	The bar is used to continue reinforcement around a corner.
DOWEL	The bar is used to connect elements placed at different times.
HORIZONTAL	The bar is horizontal reinforcement. This value should only be used for bars in a wall-like member.
LONGITUDINAL	In linear elements, such as columns, beams, and strip footings, the bar is reinforcement along the element longitudinal axis. In surface elements such as slabs and mat foundations, the bar is approximately parallel to the surface; typically, bars in the direction of the longer spans or longer element length are chosen as LONGITUDINAL (see also TRANSVERSE)
STIRRUP	The bar is normal to (or at an acute angle to) longitudinal reinforcement in a beam or beam-like element. Stirrups may be open or closed.
SUPPORT	The bar is primarily used to support other reinforcement.
TIE	The bar is encircling the longitudinal reinforcement in a column or column-like element, or the bar is linking two faces of a slab-like or wall-like element.
TRANSVERSE	In surface elements, such as slabs and mat foundations, the bar is approximately parallel to the surface and is perpendicular to (or at an acute angle to) the LONGITUDINAL bars. Although engineers may refer to stirrups and ties as "transverse reinforcement," this value is only intended for reinforcement parallel to the surfaces of slab-like members.
TRIM	The bar is additional reinforcement at an opening or edge.
VERTICAL	The bar is vertical reinforcement. This value should only be used for bars in a wall-like member.

Table 7.3e—BarPosition property usage

Value	Meaning
BOTTOM	The bar is near the bottom face of the element. This should only be used for elements that are primarily horizontal.
INSIDE	The bar is near the inside face of the element. This value should only be used in elements with a clear inside and outside face.
CENTER	The bar is in the center of the element.
OUTSIDE	The bar is near the outside face of the element. This value should only be used in elements with a clear inside and outside face.
SIDE	The bar is on the side face of the element.
TOP	The bar is on the top face of the element. This should only be used for elements that are primarily horizontal.
(user-defined)	User defined values should only be used when none of the above values is appropriate.

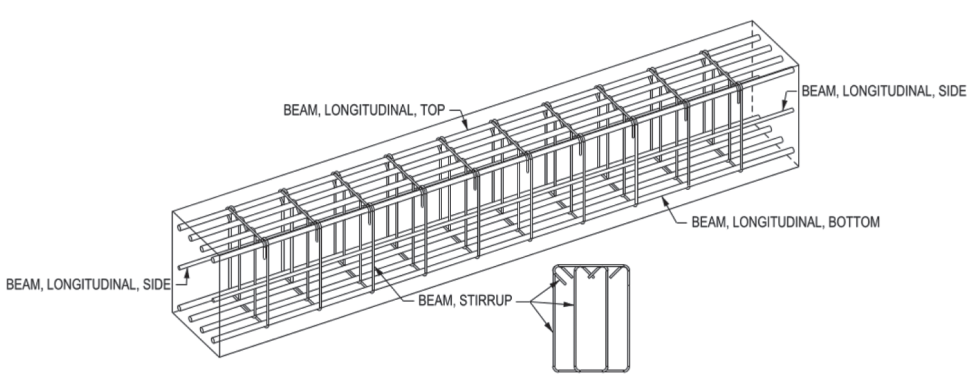


Fig. 7.3a—BarElement, BarUse, and BarPosition in beams.

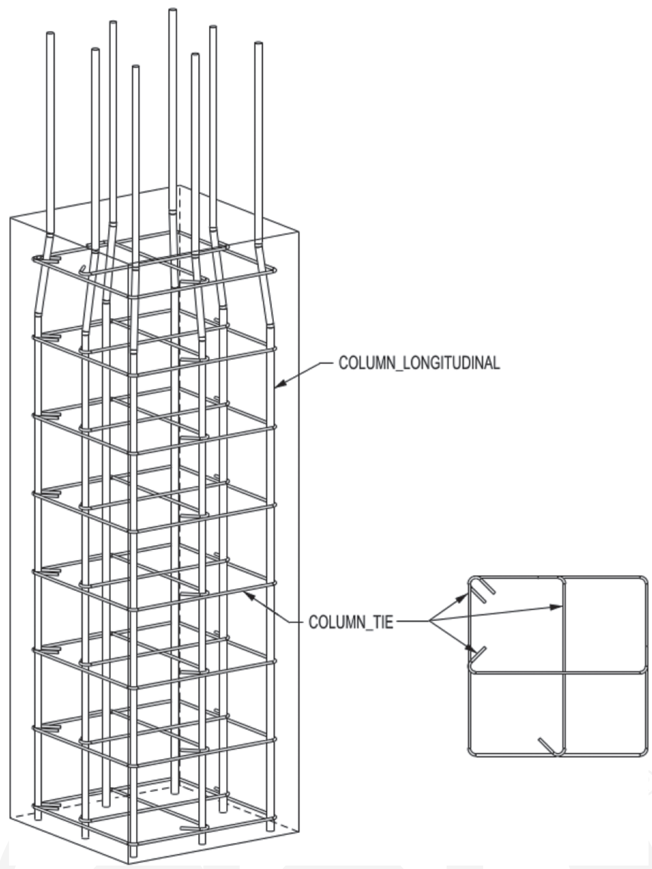


Fig. 7.3b—BarElement, BarUse, and BarPosition in columns.

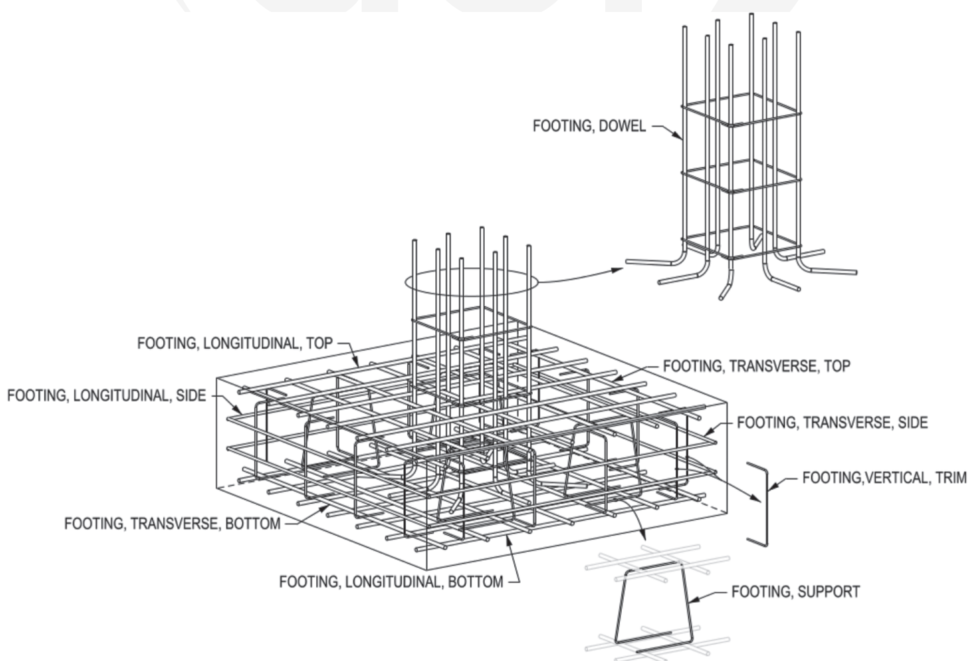


Fig. 7.3c—BarElement, BarUse, and BarPosition in spread footings.

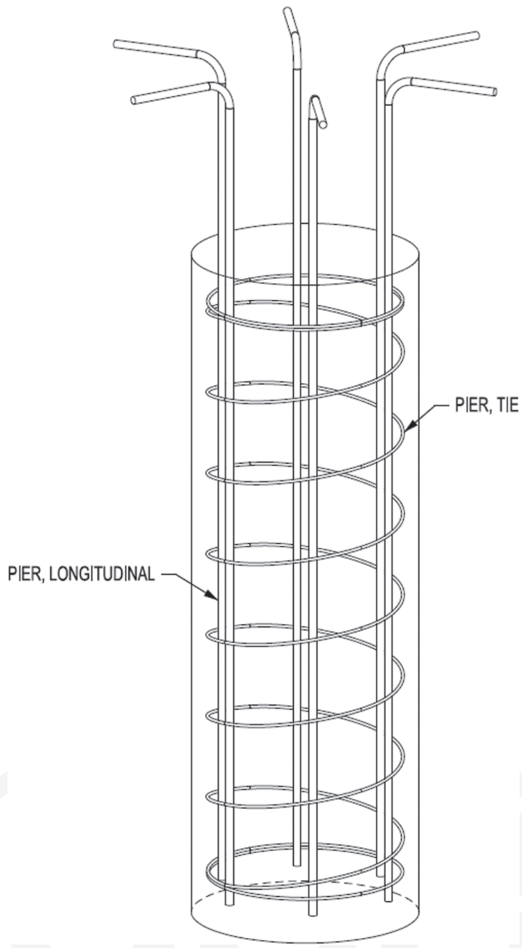


Fig. 7.3d—BarElement, BarUse, and BarPosition in piers and caissons.

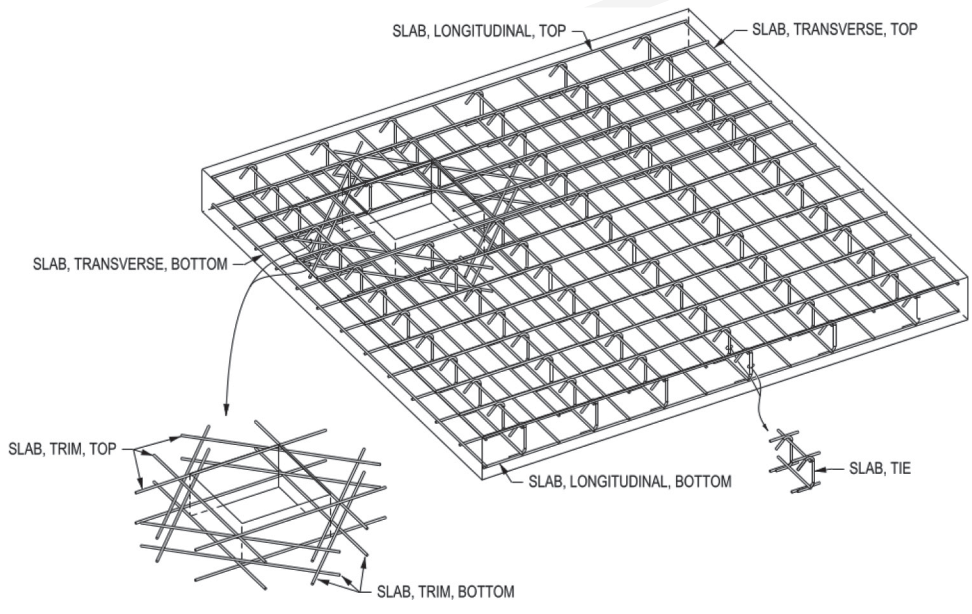


Fig. 7.3e—BarElement, BarUse, and BarPosition in slabs.

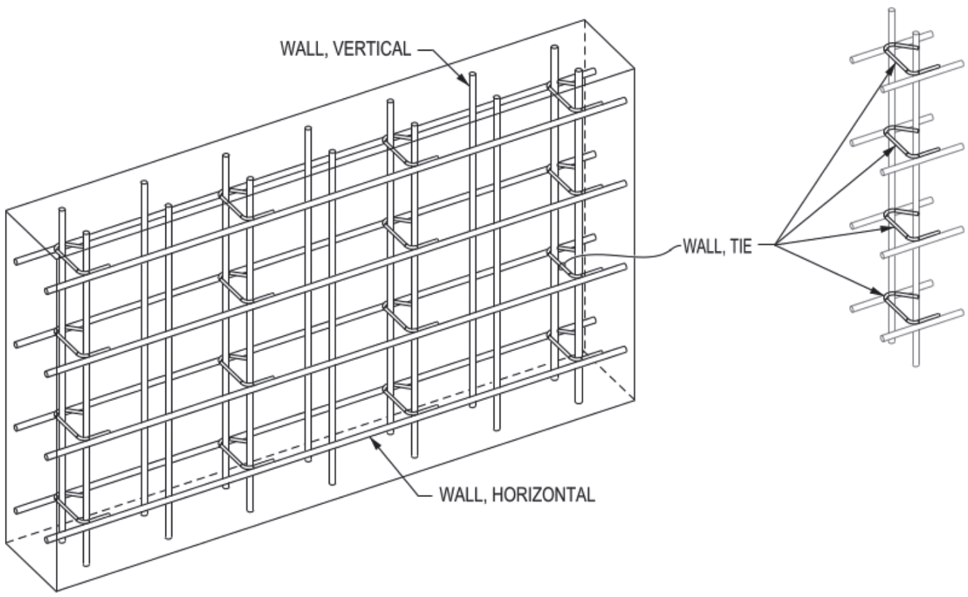


Fig. 7.3f—BarElement, BarUse, and BarPosition in walls.

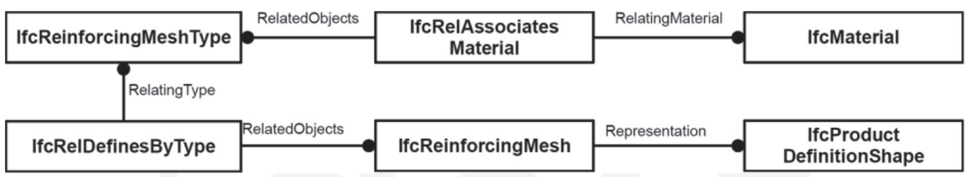


Fig. 8.1—Welded-wire reinforcement relationships.

Table 8.1a—Mesh type-material relationship (IfcRelAssociatesMaterial)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingMeshType	[1:?]	Required
RelatingMaterial	IfcMaterial	[1:1]	Required

Table 8.1b—Reinforcing mesh-mesh type relationship (IfcRelDefinesByType)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingMesh	[1:?]	Required
RelatingType	IfcReinforcingMeshType	[1:1]	Required

CHAPTER 8—WELDED-WIRE REINFORCEMENT MODELING

8.1—Conceptual modeling

This chapter refers to welded-wire reinforcement as reinforcing mesh to be consistent with IFC terminology. In the model exchange described in this guide, reinforcing mesh sheets are modeled using a combination of IfcMaterial, IfcReinforcingMeshType, and IfcReinforcingMesh, as shown in Fig. 8.1 and described in Tables 8.1a and 8.1b. When rolls of reinforcing mesh are used, the individual pieces cut from the rolls are modeled as sheets.

IfcMaterial is used to define a reinforcing mesh coating and steel wire material as described in Chapter 6. Typically, each IfcMaterial will define the material and coating for many IfcReinforcingMeshTypes through the IfcRelAssociatesMaterial relationship.

IfcReinforcingMeshType is used to define template reinforcing mesh sheets with a fixed size and arrangement. Through its IfcRelAssociatesMaterial relationship with IfcMaterial, the material of the IfcReinforcingMeshType is defined. Typically, each IfcReinforcingMeshType will define the shape, size, material, and coating for many IfcReinforcingMeshes through the IfcRelDefinesByType relationship. IfcReinforcingMesh is used to define individual sheets in specific positions. It is mostly defined through its IfcRelDefinesByType relationship to IfcReinforcingMeshType.

8.2—IfcReinforcingMeshType

IfcReinforcingMeshType is used to define a template for sheets of reinforcing mesh with the uniform material and wires of fixed size and spacing; the wire sizes and spacings maybe different in the two directions. In addition to the attri-

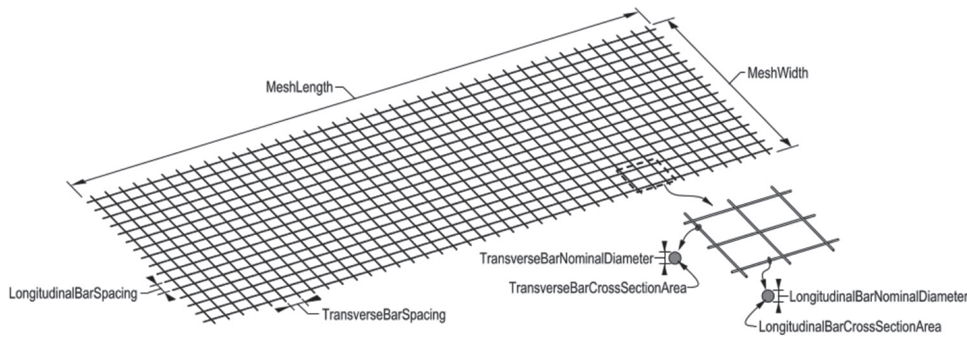


Fig. 8.2—IfcReinforcingMeshType attributes.

Table 8.2a—IfcReinforcingMeshType attribute usage

Attribute name	Type	Usage	Required or optional
PredefinedType	IfcReinforcingElementTypeEnum	Always NOTDEFINED	Required
MeshLength	IfcPositiveLengthMeasure	Overall length of mesh sheet in longitudinal direction	Required
MeshWidth	IfcPositiveLengthMeasure	Overall width of mesh sheet in transverse direction	Required
LongitudinalBarNominalDiameter*	IfcPositiveLengthMeasure	Nominal diameter of bars or wires in longitudinal direction	Required
TransverseBarNominalDiameter	IfcPositiveLengthMeasure	Nominal diameter of bars or wires in transverse direction	Required
LongitudinalBarCrossSectionArea*	IfcAreaMeasure	Effective cross section area of each longitudinal bar or wire	Required
TransverseBarCrossSectionArea	IfcAreaMeasure	Effective cross section area of each transverse bar or wire	Required
LongitudinalBarSpacing*	IfcPositiveLengthMeasure	Center-to-center spacing of longitudinal bars or wires	Required
TransverseBarSpacing	IfcPositiveLengthMeasure	Center-to-center spacing of transverse bars or wires	Required

*The "longitudinal" bars are sometimes referred to as the line wires.

Table 8.2b—Pset_ACI_ReinforcingMeshType property set usage

Property name	Type	Usage	Required or optional
LongitudinalStartOverhang*†	IfcNonNegativeLengthMeasure	Extension of longitudinal wires beyond first transverse wire.	Optional
LongitudinalEndOverhang*†	IfcNonNegativeLengthMeasure	Extension of longitudinal wires beyond last transverse wire.	Optional
TransverseLeftOverhang†	IfcNonNegativeLengthMeasure	Extension of transverse wires beyond leftmost longitudinal wire.	Optional
TransverseRightOverhang†	IfcNonNegativeLengthMeasure	Extension of transverse wires beyond rightmost longitudinal wire.	Optional
LongitudinalBarSurface*	IfcReinforcingBarSurfaceEnum	PLAIN or TEXTURED	Required
TransverseBarSurface	IfcReinforcingBarSurfaceEnum	PLAIN or TEXTURED	Required

*The "longitudinal" bars are sometimes referred to as the line wires.

†Overhang dimensions, if provided, are consistent with the LongitudinalSpacing, TransverseSpacing, MeshLength, and MeshWidth attributes of the IfcReinforcingMeshType.

butes described in 4.2, the attributes illustrated in Fig. 8.2 and defined in Table 8.2a are used for IfcReinforcingMeshType.

The BendingShapeCode and BendingParameters attributes available in IfcReinforcingMeshType are not used in this exchange, as only unbent sheets of welded wire reinforcement are considered.

The property set Pset_ACI_ReinforcingMesh is used to provide additional information regarding the shape of the IfcReinforcingMeshType. Table 8.2b defines the properties available in Pset_ACI_ReinforcingMesh.

The geometry of the IfcReinforcingMeshType is provided by an IfcRepresentationMap set in the RepresentationMaps attribute. The MappedRepresentation attribute of the IfcRepresentationMap should include a "body" representation of type "AdvancedSweptSolid," which holds multiple IfcSweptDiskSolids, each swept along a line segment.

8.3—IfcReinforcingMesh

IfcReinforcingMesh is used to represent individual mesh sheets. There is a 1:1 correspondence between a real mesh sheet and an IfcReinforcingMesh. Rolls of mesh are considered as sheets in their unrolled and cut positions. The geometry of the mesh and many of its characteristics are defined via the mesh's relationship to IfcReinforcingMeshType, as shown in 4.4 and 8.1. IfcReinforcingMeshes are positioned in space using the ObjectPlacement described in 4.3.

The attributes MeshLength, MeshWidth, LongitudinalBarNominalDiameter, TransverseBarNominalDiameter, LongitudinalBarCrossSectionArea, TransverseBarCrossSection-

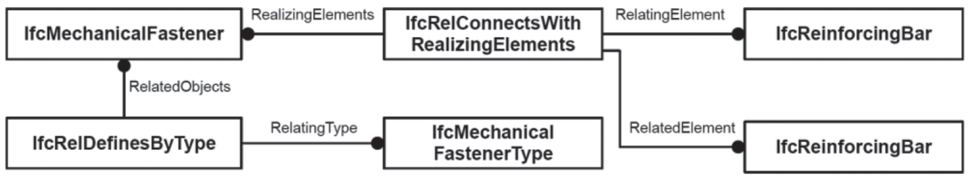


Fig. 9.1—Bar coupler relationships.

Table 9.1a—Bar-fastener relationship (IfcRelConnectsWithRealizingElements)

Attribute	Type	Cardinality	Required or optional
RelatedElement	IfcReinforcingBar	[1:1]	Required
RelatingElement	IfcReinforcingBar	[1:1]	Required
RealizingElements	IfcMechanicalFastener	[1:?]	Required

Table 9.1b—Fastener-fastener type relationship (IfcRelDefinesByType)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcMechanicalFastener	[1:?]	Required
RelatingType	IfcMechanicalFastenerType	[1:1]	Required

Table 9.2a—IfcMechanicalFastenerType attribute usage

Attribute name	Type	Usage	Required or optional
PredefinedType	IfcMechanicalFastenerTypeEnum	USERDEFINED	Required
ElementType	IfcLabel	BAR_COUPLER or BAR_COUPLER_PART	Required

Area, LongitudinalBarSpacing, TransverseBarSpacing, and PredefinedType are never used, as those attributes are instead defined via the relationship to an IfcReinforcingMeshType. There are no property sets available for IfcReinforcingMesh other than Pset_ACI_Status described in 4.3.

CHAPTER 9—BAR COUPLER MODELING

9.1—Conceptual modeling

In the model exchange described in this guide, bar couplers are modeled using a combination of IfcMechanicalFastenerType and IfcMechanicalFastener, as shown in Fig. 9.1 and described in Tables 9.1a and 9.1b.

IfcMechanicalFastenerType is used to define a template for couplers of a particular size and type (or manufacturer's model). Typically, each IfcMechanicalFastenerType will define the shape and size for many IfcMechanicalFasteners through the IfcRelDefinesByType relationship.

IfcMechanicalFastener is used to define individual couplers in specific positions. It is mostly defined through its IfcRelDefinesByType relationship to IfcMechanicalFastenerType. The IfcRelConnectsWithRealizingElements relationship defines the connection of the coupler to the bars that it connects.

When two-piece couplers are used, the pieces may be considered as a single IfcMechanicalFastener or as two IfcMechanicalFasteners. When two IfcMechanicalFasteners are used, both the pieces are referred to (the "RealizingElements") by the single IfcRelConnectsWithRealizingElements relationship. Treating each individual piece as an individual IfcMechanicalFastener enables tracking of the individual components in situations where they are not delivered together.

9.2—IfcMechanicalFastenerType

IfcMechanicalFastenerType is used to define templates for bar couplers and components of bar couplers. In addition to the attributes described in 4.2, the attributes defined in Table 9.2a are used for IfcMechanicalFastenerType.

The property set Pset_ACI_BarCouplerType is used to further define the coupler type. Table 9.2b defines the properties available in Pset_ACI_BarCouplerType.

"Relating" and "Related" in Pset_ACI_BarCouplerType property names refer to the "RelatingElement" and "RelatedElement" of the IfcRelConnectsWithRealizingElements relationship that has the IfcMechanicalFastener as its RealizingElements. The properties are used to indicate where the connection between the coupler and the bar is made. For two-piece couplers, the coupler-to-coupler connection is assumed to always be made in the field and hence does not need to be specified.

The geometry of the coupler (or component) is provided by an IfcRepresentationMap set in the RepresentationMaps attribute. The MappedRepresentation attribute of the IfcRepresentationMap should include a "body" representation using either a BRep of the actual coupler (or component) required or may be a generic IfcSweptDiskSolid as described in Table 9.2c and illustrated in Fig. 9.2. A boundary representation (BRep) defines a solid shape by defining the shape's surface.

Table 9.2b—Pset_ACI_BarCouplerType property set usage

Property name	Type	Usage	Required or optional
Manufacturer	IfcLabel	Manufacturer name or identifier	Optional
Model	IfcLabel	Coupler type name or identifier	Optional
Function	IfcLabel	PLAIN, POSITION, TRANSITION, or POSITION-TRANSITION	Required
RelatingApplication*	IfcLabel	SHOP, FIELD or MANUFACTURER	Required
RelatedApplication*	IfcLabel	SHOP, FIELD or MANUFACTURER	Required

*The "Relating" and "Related" bars are defined by the IfcRelConnectsWithRealizingElements relationship described in 9.1.

Table 9.2c—Generic coupler representations

Condition	Geometric type	Recommended diameter*	Recommended height*
Full coupler	IfcRightCircularCylinder	2 x bar diameter	6 x bar diameter
Half of two-piece coupler	IfcRightCircularCylinder	2 x bar diameter	3 x bar diameter

*More accurate dimensions may be used if available.

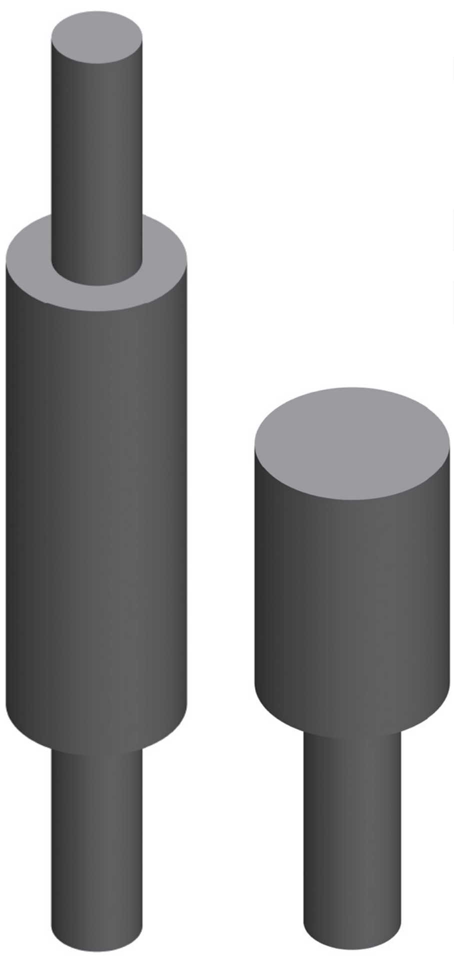


Fig. 9.2—Generic geometric modeling of one-piece and two-piece couplers.

9.3—IfcMechanicalFastener

IfcMechanicalFastener is used to represent individual bar couplers or components of bar couplers. There is a 1:1 correspondence between a real coupler or coupler component and an IfcMechanicalFastener. The geometry of the coupler or coupler component and some of its characteristics are defined via the relationship to IfcMechicalFastenerType, as described in 4.4 and 9.1. IfcMechanicalFasteners are positioned in space using the ObjectPlacement described in 4.3. In addition to the attributes described in 4.2, the attributes defined in Table 9.3a are used for IfcMechanicalFastener.

In addition to Pset_ACI_Status described in 4.3, the property set Pset_ACI_BarCoupler is used to further define the coupler. Table 9.3b defines the properties available in Pset_ACI_BarCoupler.

9.4—Coupler modeling when one bar is out of the model

In the circumstance where a coupler is used to connect a bar that is in the model with a bar that is out of the model, an IfcBuildingElementProxy is used to represent the bar that is out of the model. The use of the proxy (instead of IfcReinforcingBar) reduces the potential for misunderstandings regarding which bars are in the model while still allowing couplers to be modeled. In addition to the attributes described in 4.2, the attributes defined in Table 9.4 are used for IfcBuildingElementProxy.

IfcBuildingElementProxies representing bars out of the model are positioned in space using the ObjectPlacement described in 4.3. Their geometry is defined using a representation of IfcSweptDiskSolid; the representation may be nominal (a short segment representing the end of the bar) or may provide the complete geometry of the out-of-model bar that connects to the coupler.

When multi-part couplers are used to connect bars in the model with bars out of the model, only the coupler components to be furnished with the bars in the model are included in the model.

Table 9.3a—IfcMechanicalFastener attribute usage

Attribute name	Type	Usage	Required or optional
ObjectType	IfcLabel	BAR_COUPLER or BAR_COUPLER_PART	Required
PredefinedType	IfcMechanicalFastenerTypeEnum	USERDEFINED	Required

Table 9.3b—Pset_ACI_BarCoupler property set usage

Property name	Type	Usage	Required or optional
RelatingBarEnd*	IfcLabel	Location of coupler on the relating bar Always START or END†	Required
RelatedBarEnd*	IfcLabel	Location of coupler on the related bar Always START or END†	Required

*The “Relating” and “Related” bars are defined by the IfcRelConnectsWithRealizingElements relationship described in 9.1.

†Start and End are defined by the Directrix in the IfcSweptDiskSolid defining the bar shape.

Table 9.4—IfcBuildingElementProxy attribute usage

Attribute name	Type	Usage	Required or optional
ObjectType	IfcLabel	Always BAR_OUT_OF_MODEL	Required

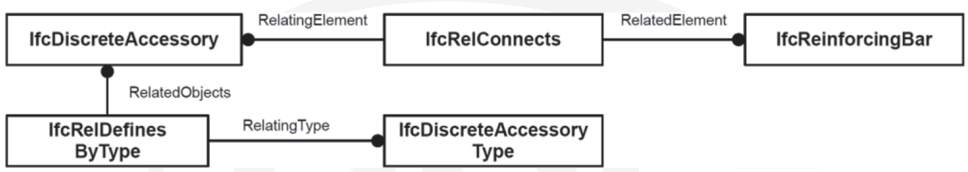


Fig. 10.1—Bar terminator relationships.

Table 10.1a—Bar-terminator relationship (IfcRelConnects)

Attribute	Type	Cardinality	Required or optional
RelatedElement	IfcReinforcingBar	[1:1]	Required
RelatingElement	IfcDiscreteAccessory	[1:1]	Required

Table 10.1b—Terminator-terminator type relationship (IfcRelDefinesByType)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcDiscreteAccessory	[1:?]	Required
RelatingType	IfcDiscreteAccessoryType	[1:1]	Required

CHAPTER 10—BAR TERMINATOR MODELING

10.1—Conceptual modeling

In the model exchange described in this guide, bar terminators are modeled using a combination of IfcDiscreteAccessoryType and IfcDiscreteAccessory, as shown in Fig. 10.1 and described in Tables 10.1a and 10.1b.

IfcDiscreteAccessoryType is used to define a template for terminators of a particular size and type (or manufacturer’s model). Typically, each IfcDiscreteAccessoryType will define the shape and size for many IfcDiscreteAccessory through the IfcRelDefinesByType relationship.

IfcDiscreteAccessory is used to define individual terminators in specific positions. It is mostly defined through its IfcRelDefinesByType relationship to IfcDiscreteAccessoryType. The IfcRelConnects relationship defines the bar that it terminates.

10.2—IfcDiscreteAccessoryType

IfcDiscreteAccessoryType is used to define templates for bar terminators. In addition to the attributes described in 4.2, the attributes defined in Table 10.2a are used for IfcDiscreteAccessoryType.

The property set Pset_ACI_BarTerminatorType is used to further define the terminator type. Table 10.2b defines the properties available in Pset_ACI_BarTerminatorType.

The geometry of the terminator is provided by an IfcRepresentationMap set in the RepresentationMaps attribute. The MappedRepresentation attribute of the IfcRepresentationMap should include a body representation using either a

Table 10.2a—IfcDiscreteAccessoryType attribute usage

Attribute name	Type	Usage	Required or optional
PredefinedType	IfcDiscreteAccessoryTypeEnum	USERDEFINED	Required
ElementType	IfcLabel	BAR_TERMINATOR	Required

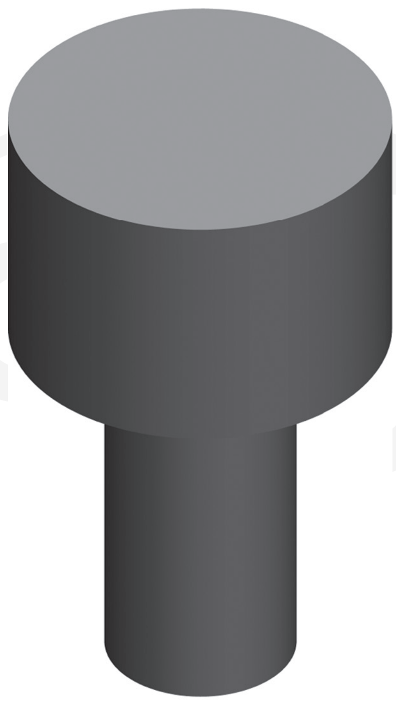
Table 10.2b—Pset_ACI_BarTerminatorType property set usage

Property name	Type	Usage	Required or optional
Manufacturer	IfcLabel	Manufacturer name or identifier	Optional
Model	IfcLabel	Terminator type name or identifier	Optional
Application	IfcLabel	SHOP, FIELD, or MANUFACTURER	Required

Table 10.2c—Generic terminator representations

Geometric type	Recommended diameter*	Recommended height*
IfcRightCircularCylinder	3 × bar diameter	2 × bar diameter

*More accurate dimensions may be used if available.



Terminator (with bar)

Fig. 10.2—Generic geometric modeling of terminator.

BRep of the actual terminator required or may be a generic IfcSweptDiskSolid as described in Table 10.2c and illustrated in Fig. 10.2.

10.3—IfcDiscreteAccessory

IfcDiscreteAccessory is used to represent individual bar terminators. There is a 1:1 correspondence between a real bar terminator and an IfcDiscreteAccessory. The geometry of the bar terminator and some of its characteristics are defined via the relationship to IfcDiscreteAccessoryType, as described in 4.4 and 10.1. IfcDiscreteAccessories are positioned in space using the ObjectPlacement described in 4.3. In addition to the attributes described in 4.2, the attributes defined in Table 10.3a are used for IfcDiscreteAccessory.

In addition to Pset_ACI_Status described in 4.3, the property set Pset_ACI_BarTerminator is used to further define the terminator. Table 10.3b defines the properties available in Pset_ACI_BarTerminator.

Table 10.3a—IfcDiscreteAccessory attribute usage

Attribute name	Type	Usage	Required or optional
ObjectType	IfcLabel	BAR_TERMINATOR	Required
PredefinedType	IfcDiscreteAccessoryTypeEnum	USERDEFINED	Required

Table 10.3b—Pset_ACI_BarTerminator property set usage

Property name	Type	Usage	Required or optional
BarEnd	IfcLabel	Location of terminator on the bar Always START or END*	Required

*START and END are defined by the Directrix in the IfcSweptDiskSolid defining the bar shape.

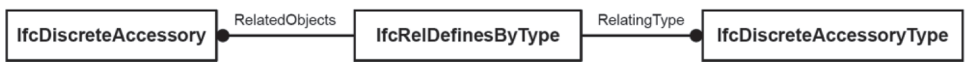


Fig. 11.1—Bar accessory relationships.

Table 11.1—Accessory-accessory type relationship (IfcRelDefinesByType)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcDiscreteAccessory	[1:?]	Required
RelatingType	IfcDiscreteAccessoryType	[1:1]	Required

Table 11.2a—IfcDiscreteAccessoryType attribute usage

Attribute name	Type	Usage	Required or optional
PredefinedType	IfcDiscreteAccessoryTypeEnum	USERDEFINED	Required
ElementType	IfcLabel	One of the following: CHAIR BOLSTER DOBIE SIDE_FORM_SPACER SPIRAL_PITCH_SPACER BOOT	Required

CHAPTER 11—ACCESSORY MODELING

11.1—Conceptual modeling

In the model exchange described in this guide, bar accessories are modeled using a combination of IfcDiscreteAccessoryType and IfcDiscreteAccessory, as shown in Fig. 11.1 and described in Table 11.1.

IfcDiscreteAccessoryType is used to define a template for accessories of a particular size and type (or manufacturer's model). Typically, each IfcDiscreteAccessoryType will define the shape and size for many IfcDiscreteAccessory through the IfcRelDefinesByType relationship.

IfcDiscreteAccessory is used to define individual accessories in specific positions. It is mostly defined through its IfcRelDefinesByType relationship to IfcDiscreteAccessoryType.

11.2—IfcDiscreteAccessoryType

IfcDiscreteAccessoryType is used to define templates for bar accessories. In addition to the attributes described in 4.2, the attributes defined in Table 11.2a are used for IfcDiscreteAccessoryType.

The property set Pset_ACI_BarAccessoryType may be used to further define the accessory type. Table 11.2b defines the properties available in Pset_ACI_BarAccessoryType.

The geometry of the accessory is provided by an IfcRepresentationMap set in the RepresentationMaps attribute. The MappedRepresentation attribute of the IfcRepresentationMap should include a "body" representation using either a BRep of the actual accessory required or may be a generic representation as described in Table 11.2f and illustrated in Fig. 11.2.

Table 11.2b—Pset_ACI_BarAccessoryType property set usage

Property name	Type	Usage	Required or optional
Manufacturer	IfcLabel	Manufacturer name or identifier	Optional
Model	IfcLabel	Accessory type name or identifier	Optional
Standard	IfcLabel	The standard that defines values for material and type. CRSI is only predefined value.	Required if material or type is defined
Material	IfcLabel	Material type. Standard values for CRSI are: WIRE, WIRE_WITH_PLASTIC_TIPS, PLASTIC, CONCRETE	Optional
Type	IfcLabel	The bar accessory type. For CRSI, refer to Tables 11.2c through 11.2e	Optional
Size	IfcPositiveLengthMeasure	Height, pitch, or, cover distance	Required for CHAIR, BOLSTER, DOBIE, SIDE_FORM_SPACER, SPIRAL_PITCH_SPACER
LegSpacing	IfcPositiveLengthMeasure	Spacing of legs. This is typically used for continuous high chairs for metal deck (CHCM)	Optional
BarSize	IfcPositiveLengthMeasure	Nominal bar diameter	Optional (use where applicable)

Table 11.2c—Standard CRSI wire bar support names (symbols)

Symbol	Meaning
SB	Slab bolster
SBU	Slab bolster upper
BB	Beam bolster
BBU	Beam bolster upper
BC	Individual bar chair
JC	Joist chair
HC	Individual high chair
HCM	High chair for metal deck
CHC	Continuous high chair
CHCU	Continuous high chair upper
CHCM	Continuous high chair for metal deck
JCU	Joist chair upper
CS	Continuous support (aka zig-zag)
SBC	Single bar centralizer

Note: Table derived from CRSI (2009), Table 3-1.

Table 11.2d—Standard CRSI precast concrete bar support names (symbols)

Symbol	Meaning
PB	Plain block
WB	Wired block
CB	Combination block
DB	Dowel block
DSSS	Side-form spacer: wired
DSBB	Bottom bolster: wired
DSWS	Side-form spacer for drilled shaft applications

Note: Table derived from CRSI (2009), Table 3-3.

Table 11.2e—Standard CRSI plastic bar support names (symbols)

Symbol	Meaning
BS	Bottom support
BS-CL	Bottom support with clamping action
HC	High chair
HC-V	High chair: variable
WS	Wheel side-form spacer
DSWS	Side-form spacer for drilled shaft applications
VLWS	Locking wheel side-form spacer for all vertical applications
DSBB	Bottom bolster (gripping)
SB	Slab bolster
HDHC	Heavy-duty high chair
OGC	On-grade chair
SBU	Slab bolster upper

Note: Table derived from CRSI (2009), Table 3-4.

Table 11.2f—Generic representations

Accessory	Geometric type	Recommended dimensions*
CHAIR	IfcRightCircularCone	Height = chair height BottomRadius = height/4
BOLSTER	IfcExtrudedAreaSolid with triangle IfcArbitraryClosedProfileDef	Equilateral triangle with Height = bar cover and Extrusion length = bolster length
DOBIE	IfcBlock	XLength = bar cover YLength = bar cover ZLength = bar cover
SIDE_FORM_SPACER	IfcRightCircularCylinder	Height = radius/4 Radius = bar cover + bar radius
SPIRAL_PITCH_SPACER	IfcExtrudedAreaSolid with L-shaped IfcArbitraryClosedProfileDef	L-shaped extrusion with Leg = 2 × spiral bar diameter Thickness = spiral bar diameter/4 Extrusion length = extent of spacer
BOOT	IfcRightCircularCylinder	Height = 2 × bar cover Radius = bar cover/2

*More accurate dimensions may be used if available.

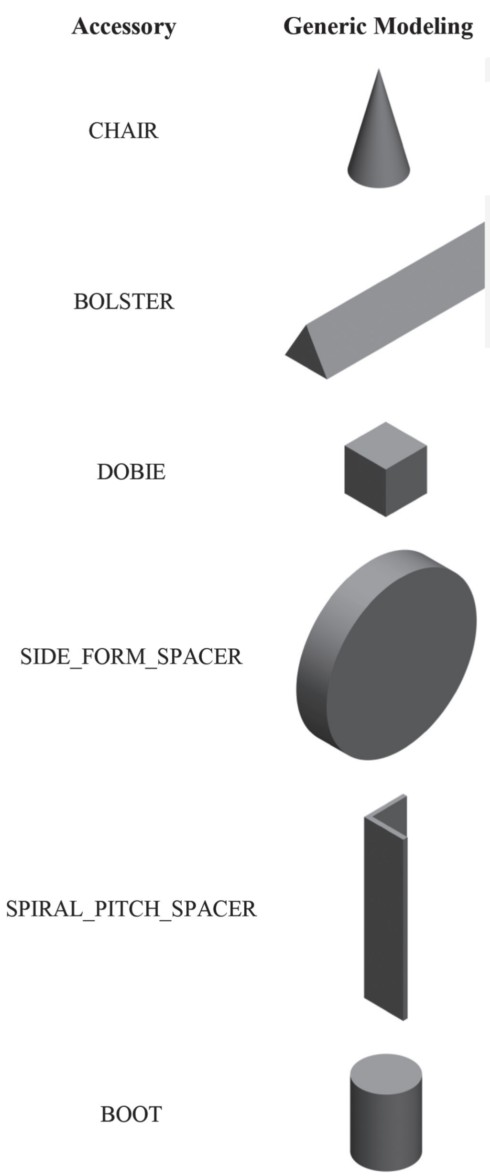


Fig. 11.2—Generic geometric modeling of accessories.

11.3—IfcDiscreteAccessory

IfcDiscreteAccessory is used to represent either an individual bar accessory in a specific location or a set of identical bar accessories that may not be explicitly positioned. The geometry of the accessory and some of its characteristics are defined via the relationship to IfcDiscreteAccessoryType, as described in 4.4 and 11.1. IfcDiscreteAccessories are positioned in space using the ObjectPlacement described in 4.3. In addition to the attributes described in 4.2, the attributes defined in Table 11.3a are used for IfcDiscreteAccessory.

When IfcDiscreteAccessory represents an individual bar accessory, there is a 1:1 correspondence between a real accessory and an IfcDiscreteAccessory. The individual IfcDiscreteAccessory is positioned in its true location.

When IfcDiscreteAccessory represents a set of identical bar accessories, the positioning of the IfcDiscreteAccessory may be arbitrary (outside of the building, for example) or may be in the actual location of one of the identical accessories.

In addition to Pset_ACI_Status defined in 4.3, the property set Pset_ACI_BarAccessory is used to provide additional information regarding the IfcDiscreteAccessory. Table 11.3b defines the properties available in Pset_ACI_BarAccessory.

Table 11.3a—IfcDiscreteAccessory attribute usage

Attribute name	Type	Usage	Required or optional
ObjectType	IfcLabel	One of the following: CHAIR BOLSTER DOBIE SIDE_FORM_SPACER SPIRAL_PITCH_SPACER BOOT	Required
PredefinedType	IfcDiscreteAccessoryTypeEnum	USERDEFINED	Required

Table 11.3b—Pset_ACI_BarAccessory property set usage

Property name	Type	Usage	Required or optional
Count	IfcInteger	The number of accessories at this location (always positive)	Required
AccessoryElement*	IfcLabel	Always one of the following: BEAM COLUMN CORBEL FOOTING PIER PILE_CAP (applies to pier caps also) SLAB WALL	Optional

*AccessoryElement usage is identical to that of the BarElement property described in Table 7.3c.

CHAPTER 12—CALLOUT AND CAGE MODELING

12.1—Conceptual modeling

A callout in this exchange is analogous to callout on a reinforcement drawing. At minimum, a callout contains the quantity, size, and length or mark; it may also contain additional properties such as grade, coating, and placing instructions as necessary.

A cage in this exchange contains not only the reinforcing bars but also accessories that are considered as components of the cage. For example, a pile cage might contain side form spacers and bar boots as well as reinforcing bars.

In the model exchange described in this guide, bar callouts and cages are modeled using an IfcElementAssembly, as shown in Fig. 12.1a through 12.1c, and described in Tables 12.1a and 12.1b. Unlike the modeling strategy for bars, mesh, couplers, terminators, and accessories, IfcElementAssemblyType is not used.

Due to the hierarchical decomposition nature of IfcElementAssembly, an IfcReinforcingBar may be directly related to a maximum of one cage or callout. A reinforcing bar may be indirectly defined as part of a cage by its callout being part of the cage. Cages may be nested.

12.2—IfcElementAssembly for callouts

IfcElementAssembly is used to define callouts for groups of bars (often identical bars). In addition to the attributes described in 4.2, the attributes defined in Table 12.2a are used for IfcElementAssembly when used as a callout.

The property set Pset_ACI_BarCallout is used to further define the callout. Table 12.2b defines the properties available in Pset_ACI_BarCallout.

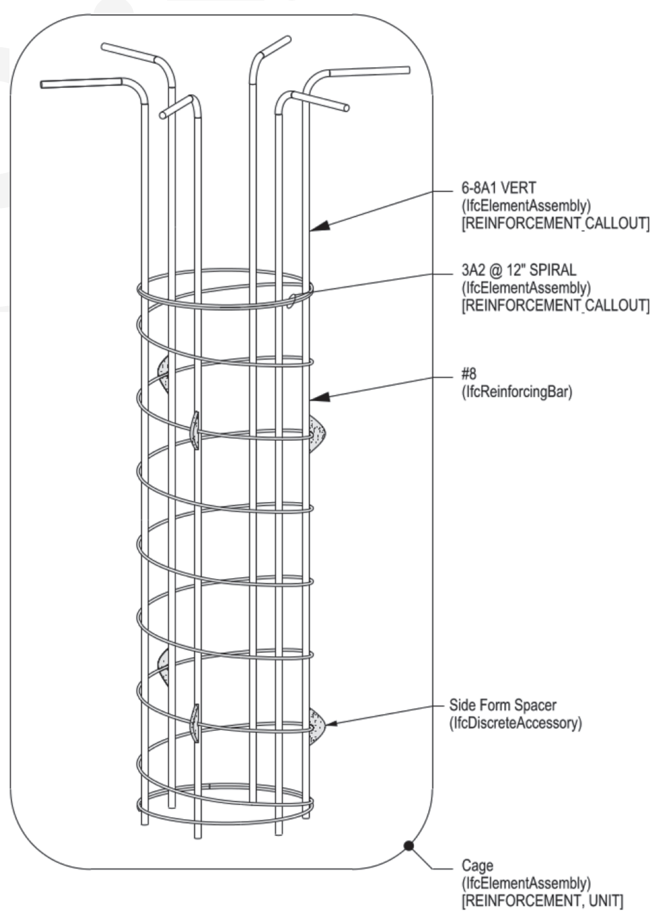


Fig. 12.1a—Callouts and cages.

Fig. 12.1b—Relationships for callouts.

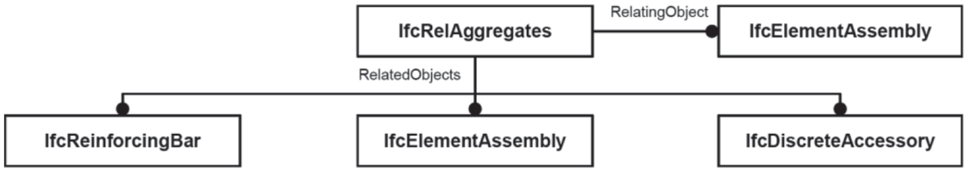


Fig. 12.1c—Relationships for cages.

Table 12.1a—Element-element assembly relationship for callouts (IfcRelAggregates)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingBar	[1:?]	Required
RelatingObject	IfcElementAssembly	[1:1]	Required

Table 12.1b—Element-element assembly relationship for cages (IfcRelAggregates)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingBar* IfcElementAssembly representing a callout IfcElementAssembly representing a cage IfcDiscreteAccessory representing bar accessories	[1:?]	Required
RelatingObject	IfcElementAssembly	[1:1]	Required

*At least one reinforcing bar is always included in the cage, directly or indirectly through a cage or callout.

Table 12.2a—IfcElementAssembly attribute usage for callout modeling

Attribute Name	Type	Usage	Required or optional
ObjectType	IfcLabel	REINFORCEMENT_CALLOUT	Required
PredefinedType	IfcElementAssemblyTypeEnum	USERDEFINED	Required

Table 12.2b—Pset_ACI_BarCallout property set usage

Property name	Type	Usage	Required or optional
BarCount	IfcInteger	The number of bars	Required
DesignSpacing	IfcPositiveLengthMeasure	The spacing of the bars specified in the design	Optional
DesignLapSpliceLength	IfcPositiveLengthMeasure	The length of the required lap splice specified in the design	Optional
PlacementStartDate*	IfcDate	Date placement will start	Optional
PlacementEndDate*	IfcDate	Date placement will be completed	Optional
PlacementTaskId	IfcLabel	Identifier of the placement task in scheduling software. Same TaskId may appear on multiple callouts.	Optional

*If PlacementStartDate or PlacementEndDate are specified, the values represent the projected dates at the time of the creation of the exchange data.

"Design" in the names DesignSpacing and DesignLapSpliceLength is intended to clarify that these values are what has been specified in the design and may not exactly match what has been modeled. DesignLapSliceLength applies at any end of the bar where a lap splice occurs; this property set does not allow for specifying different lap splices at each end of the bar. No explicit geometry or placement is provided for the callout; the geometry and location are defined indirectly by the elements in the assembly.

Table 12.3a—IfcElementAssembly attribute usage for cage modeling

Attribute name	Type	Usage	Required or optional
AssemblyPlace	IfcAssemblyPlaceEnum	Always SITE or FACTORY	Required
PredefinedType	IfcElementAssemblyTypeEnum	REINFORCEMENT_UNIT	Required

Table 12.3b—Pset_ACI_BarCage property set usage

Property name	Type	Usage	Required or optional
BarCount	IfcInteger	Total number of bars in the cage	Required
Mass	IfcMassMeasure	Total mass of the cage including any nonbar items (not always a sum of the individual bar masses)	Optional

Fig. 13.1a—Bundle relationships.

Fig. 13.1b—Release relationships.

Table 13.1a—Bar-group relationship for bundles (IfcRelAssignsToGroup)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingBar	[1:?]	Required
RelatingGroup	IfcGroup	[1:1]	Required

12.3—IfcElementAssembly for cages

IfcElementAssembly is used to define cages of reinforcement. In addition to the attributes described in 4.2, the attributes defined in Table 12.3a are used for IfcElementAssembly when used as a cage.

The property set Pset_ACI_BarCage is used to further define the cage. Table 12.3b defines the properties available in Pset_ACI_BarCage.

No explicit geometry or placement is provided for the cage; the geometry and location are defined indirectly by the elements in the assembly.

CHAPTER 13—BUNDLE AND RELEASE MODELING

13.1—Conceptual modeling

Bar bundles are sets of reinforcement bars tied or packaged together to facilitate shipping and logistics. Bundles typically consist of one size, length, or mark of reinforcing bars tied together, with the following exceptions: 1) very small quantities of nonsimilar bars may be bundled together for convenience; and 2) groups of varying bar lengths or marks that will be placed adjacent may be bundled together. Maximum weight of bundles is dependent on regional practices and site conditions. A bundle in this document does not refer to bars bundled together to act as a single reinforcement unit.

A release is a set of reinforcement and accessories that is approved for fabrication and delivery. A release typically includes material intended for a specific area of a project.

In the model exchange described in this guide, bar bundles and releases are modeled using IfcGroup, as shown in Fig. 13.1a and 13.1b and described in Tables 13.1a and 13.1b. If releases are modeled, each reinforcing bar can only be related to a single release. If bundles are modeled and a reinforcing bar may be contained in one of a set of bundles, the reinforcing bar is related to each of the bundles in which it may be contained.

13.2—IfcGroup for bundles

IfcGroup is used to define fabrication and shipping bundles of reinforcing bars. This bundle information is only included in the model when the detailer needs to communicate bundle information prior to fabrication of the reinforcement. In addition to the attributes described in 4.2, the attributes defined in Table 13.2a are used for IfcGroup when used as a bundle.

The property set Pset_ACI_BarBundle is used to further define the bundle. Table 13.2b defines the properties available in Pset_ACI_BarBundle.

Table 13.1b—Bar-group relationship for releases (IfcRelAssignsToGroup)

Attribute	Type	Cardinality	Required or optional
RelatedObjects	IfcReinforcingBar, IfcElementAssembly representing a callout, or both, optionally together with IfcDiscreteAccessory representing bar accessories or terminators, optionally together with IfcMechanicalFastener representing bar couplers	[1:?]	Required
RelatingGroup	IfcGroup	[1:1]	Required

Table 13.2a—IfcGroup attribute usage for bundle modeling

Attribute name	Type	Usage	Required or optional
ObjectType	IfcLabel	REINFORCEMENT_BUNDLE	Required

Table 13.2b—Pset_ACI_BarBundle property set usage

Property name	Type	Usage	Required or optional
BarCount	IfcInteger	Total number of bars in the bundle	Required
MillCertifications	IfcLabel	The mill certificate numbers. If the bars in the bundle may be fabricated from one of multiple heats, multiple mill certification numbers may be provided, separated by commas.	Optional

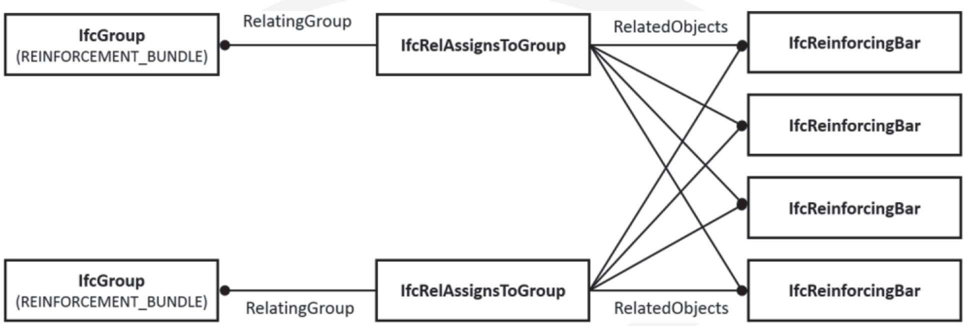


Fig. 13.2—Bars assigned to multiple bundles.

In most cases, the BarCount property in Pset_ACI_BarBundle will not be equal to the count of the bars related to the bundle, as an individual bar may be related to multiple bundles if it is not known precisely which bundle the bar will be contained in. Figure 13.2 shows two bundles that each contain two bars (BarCount = 2); as it is not known which bundle will contain which bar, each bundle is related to four bars.

13.3—IfcGroup for releases

IfcGroup is used to define releases of reinforcing bars, couplers, terminators, and other accessories. The inclusion of couplers and other accessories in the IfcGroup representing the release is optional. This release information is only included in the model when the detailer needs to communicate release information prior to fabrication of the reinforcement. In addition to the attributes described in 4.2, the attributes defined in Table 13.3a are used for IfcGroup when used as a release.

The property set Pset_ACI_BarRelease may be used to further define the release. Table 13.3b defines the properties available in Pset_ACI_BarRelease.

Table 13.3b—Pset_ACI_BarRelease property set usage

Property name	Type	Usage	Required or optional
Status	IfcLabel	Human-readable status of the release. Recommended values are: Draft Pending award Pending RFI Pending approval Approved Ready for fabrication Shipped Installed Invoiced Paid	Optional
StatusDate	IfcDate	Date status property was effective. Never set unless status is set.	Optional
ControlCode	IfcLabel	An alpha-numeric code that uniquely identifies a material release throughout the fabrication process	Optional
FabricationFacility	IfcLabel	Text that specifies the facility where the material is to be fabricated	Optional
ReleaseNumber	IfcLabel	Number or text that identifies the current material release as a portion of the entire job	Optional
Description	IfcText	Text that identifies the purpose for which the material is being provided	Optional
TruckIdentifiers	IfcLabel	Text that identifies the truck/trailer that will be used to deliver the material to the jobsite	Optional
ReferenceDrawings	IfcLabel	Text that refers to the drawing(s) or document(s) on which the material is referenced	Optional
OtherReferences	IfcText	Text that can be used to provide an additional description or message related to the material on the release	Optional
SpecialTransportationInstructions	IfcText	Text that can be used to provide any special transportation requirements that apply	Optional
MaxStraightBundleMass	IfcMassMeasure	Maximum mass (weight) allowed per bundle of straight reinforcement	Optional
MaxBentBundleMass	IfcMassMeasure	Maximum mass (weight) allowed per bundle of bent reinforcement	Optional
TransportationMassLimit	IfcMassMeasure	Maximum mass (weight) permitted on a delivery, including the weight of the truck/trailer	Optional
ReleaseCreationDate	IfcDate	Date that the material release was created	Optional
RequestedDeliveryDate	IfcDate	Date that the material should be delivered to the jobsite	Optional
CreatedBy	IfcLabel	Text that identifies the person who created the material release	Optional

CHAPTER 14—REFERENCES

Committee documents are listed first by document number and year of publication followed by authored documents listed alphabetically.

American Concrete Institute

ACI 131.1R-14—Information Delivery Manual for Cast-in-Place Concrete

ACI 315-99—Details and Detailing of Concrete Reinforcement (available in ACI SP-66)

International Standards Organization

ISO 10303-11:2004—Industrial Automation Systems and Integration—Product Data Representation and Exchange—Part 11: Description and Methods: The EXPRESS Language Reference Manual

ISO 10303-21:2016—Industrial Automation Systems and Integration—Product Data Representation and Exchange—Part 21: Implementation Methods: Clear Text Encoding of the Exchange Structure

ISO 16739:2013—Industry Foundation Classes (IFC) for Data Sharing in the Construction and Facility Management Industries

ISO 3166-1:2013—Codes for the Representation of Names of Countries and their Subdivisions—Part 1: Country Codes

Authored documents

American Institute of Architects, 2013, *G202-2013 Building Information Modeling Protocol Form*, Washington, DC.

BIMForum, 2015, “Level of Development Specification,” https://bimforum.org/lod/ (accessed November 14, 2016)

buildingSMART International, 2013, “Industry Foundation Classes Release 4 (IFC4),” http://www.buildingsmart-tech.org/ifc/IFC4/final/html/ (accessed November 14, 2016)

buildingSMART International, 2015, “Industry Foundation Classes Release 4 Addendum 1 (IFC4 Add 1),” http://www.buildingsmart-tech.org/ifc/IFC4/Add1/html/ (accessed November 14, 2016)

Concrete Reinforcing Steel Institute (CRSI), 2009, *Manual of Standard Practice*, 28th edition, CRSI, Schaumburg, IL.

National Institute of Building Sciences, 2007, “United States National Building Information Modeling Standard,” Version 1—Part 1: Overview, Principles, and Methodologies, NIBS, Washington, DC, 183 pp.

APPENDIX A—INDUSTRY FOUNDATION CLASSES INSTANCE DIAGRAMS

A.1—Standard Industry Foundation Classes instance diagrams

Industry Foundation Classes (IFC) documentation has traditionally included instance diagrams that illustrate object instances, object attributes, and relationships simultaneously. This appendix includes a selection of instance diagrams for the instances described in the previous chapters.

A.2—Selected diagrams

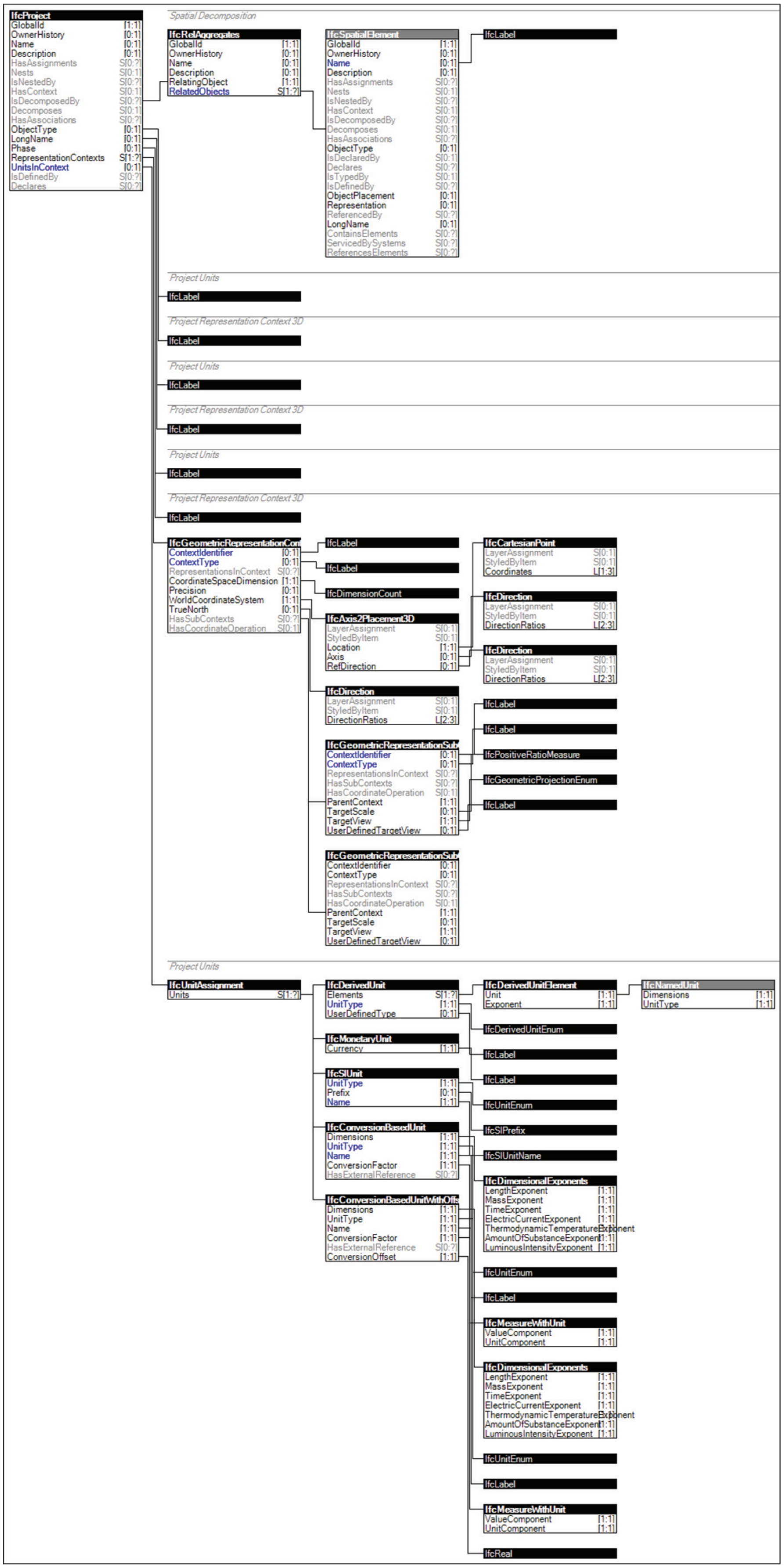


Fig. A.2a—IfcProject.

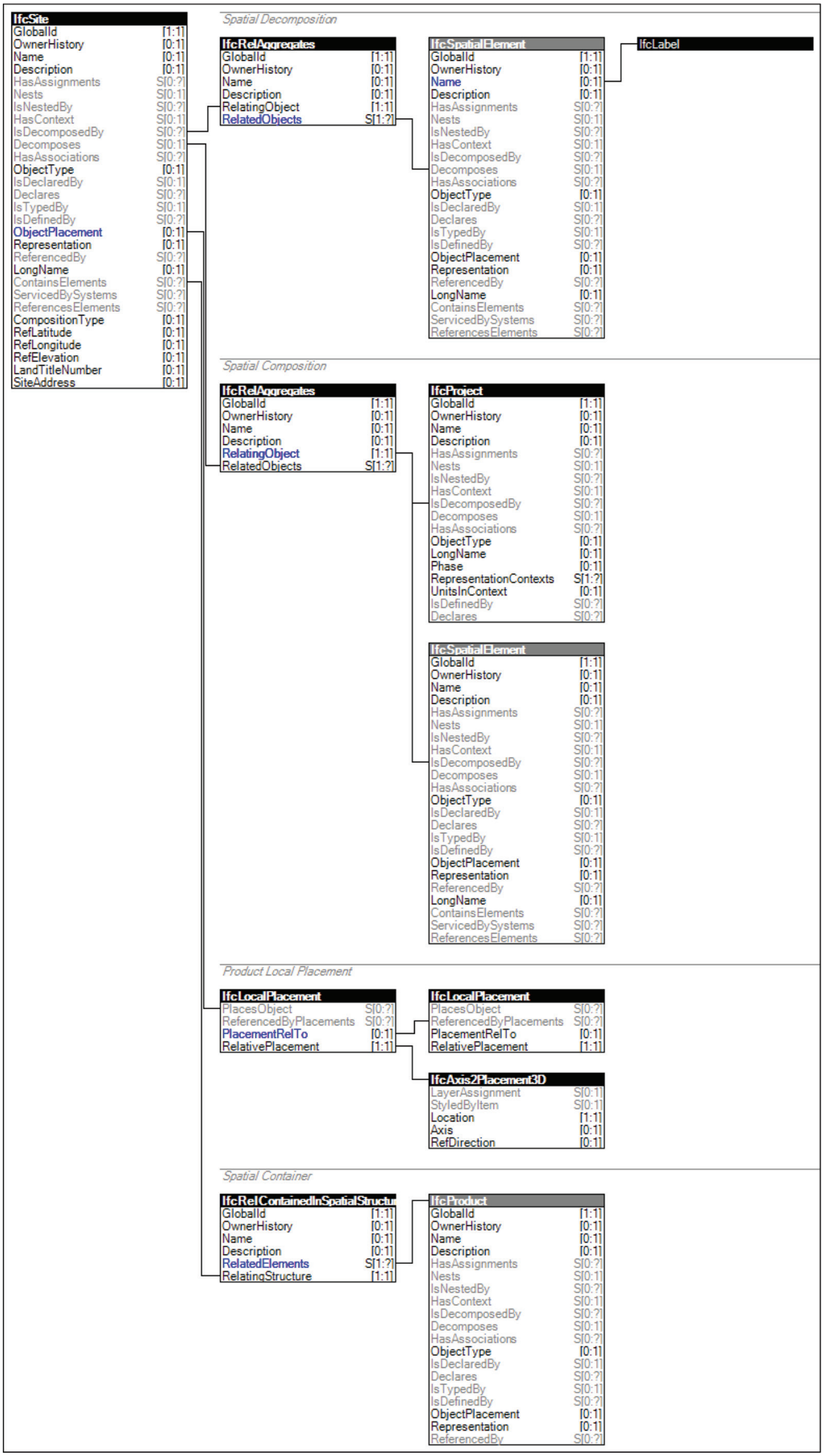


Fig. A.2b—IfcSite.

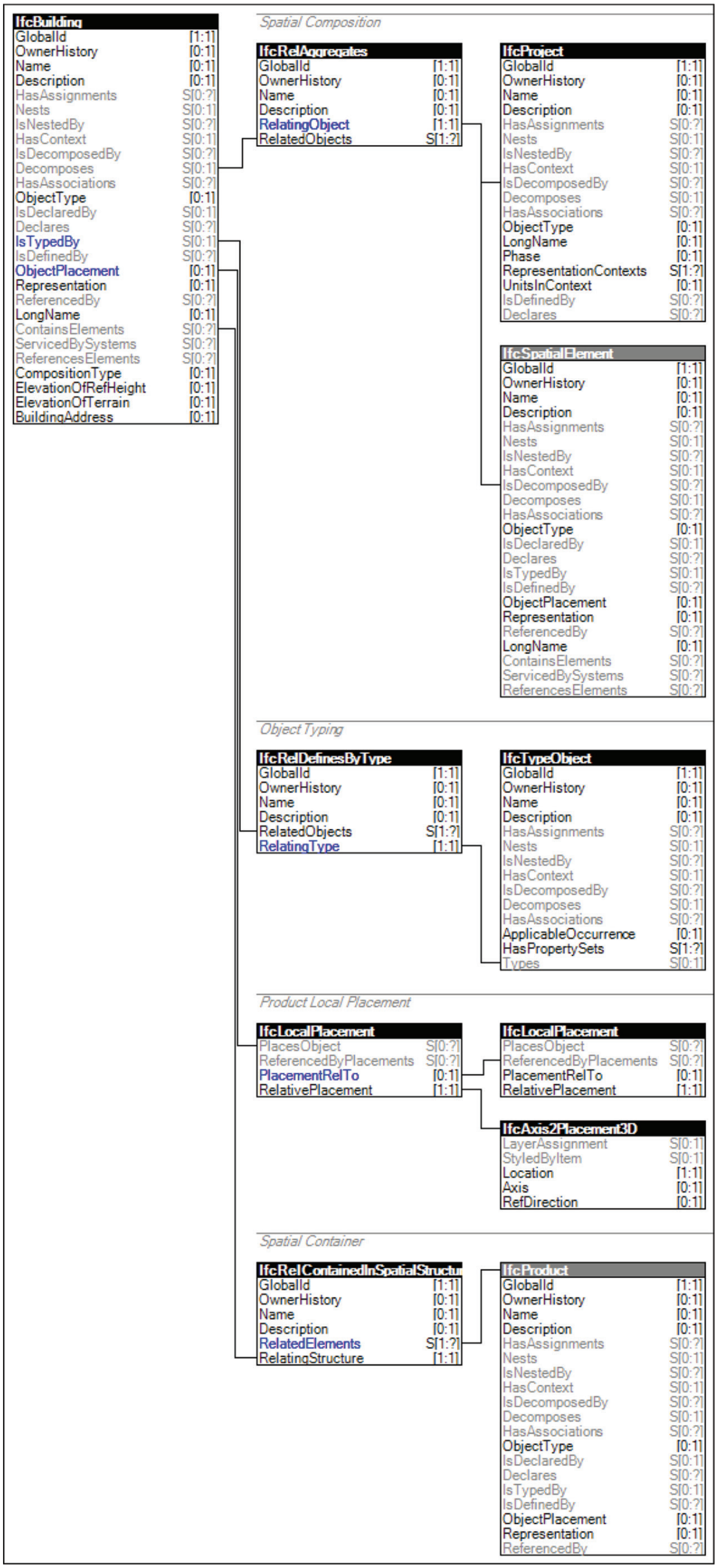


Fig. A.2c—IfcBuilding.

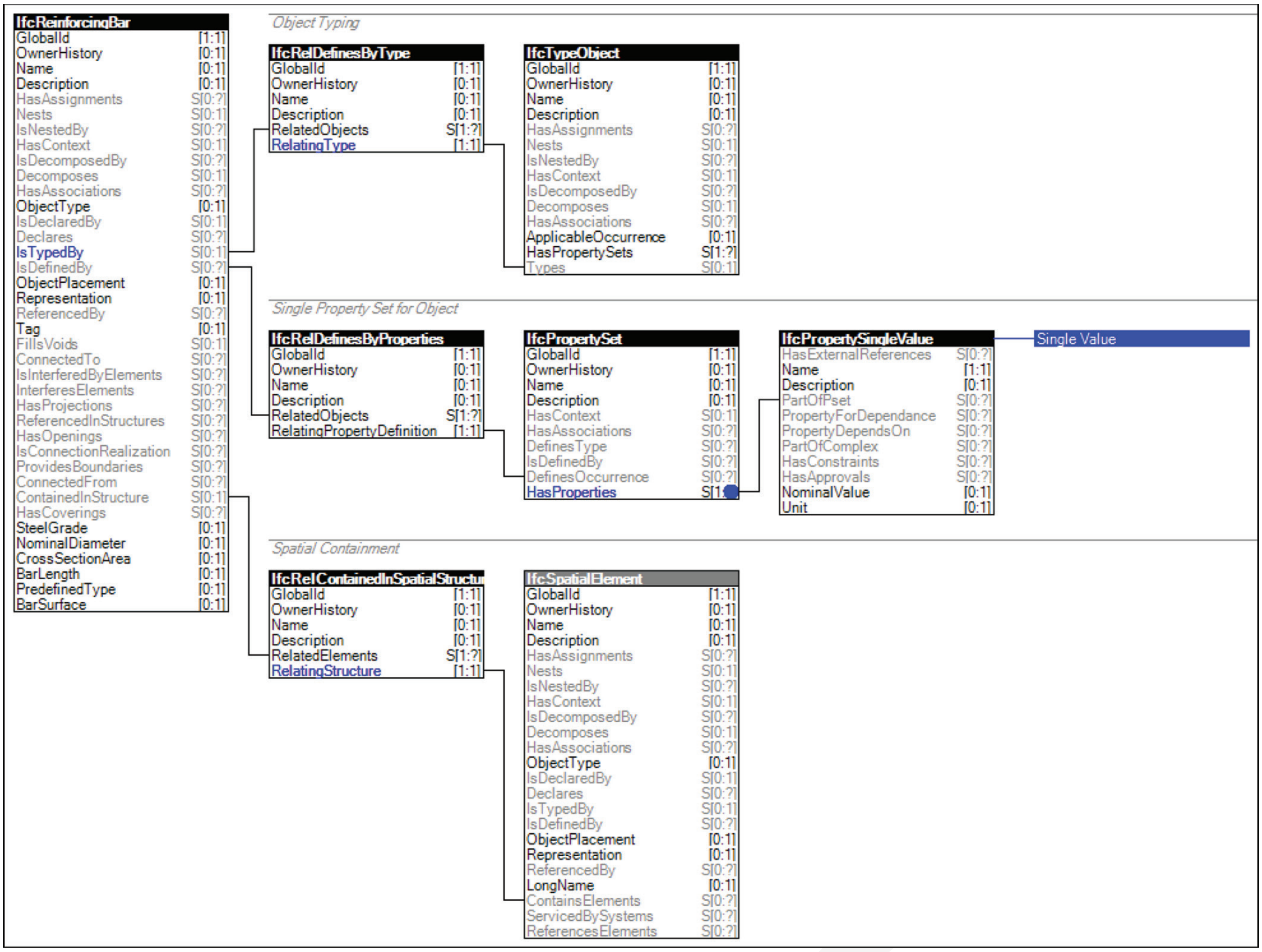


Fig. A.2d—IfcReinforcingBar.

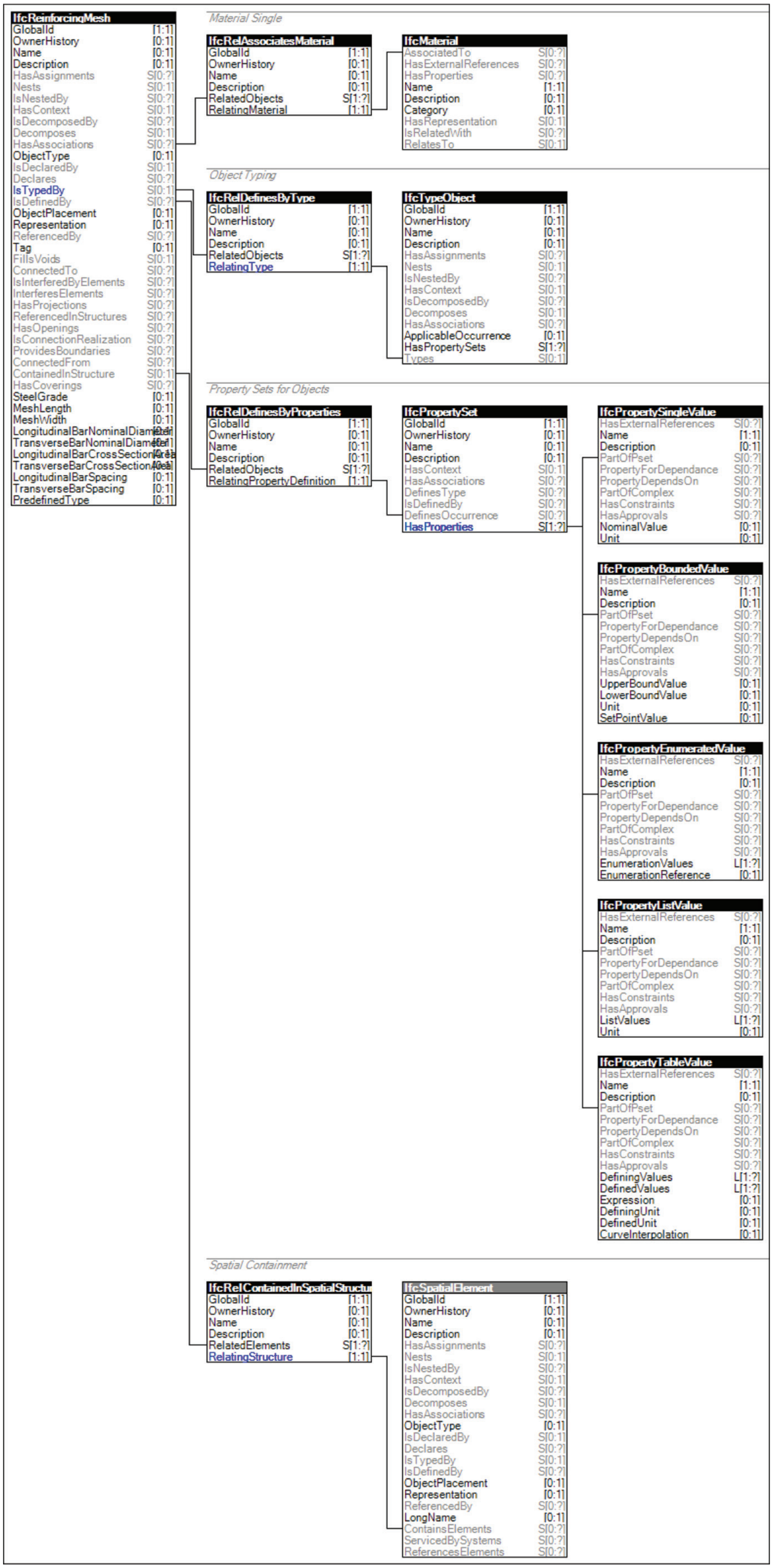


Fig. A.2e—IfcReinforcingMesh.

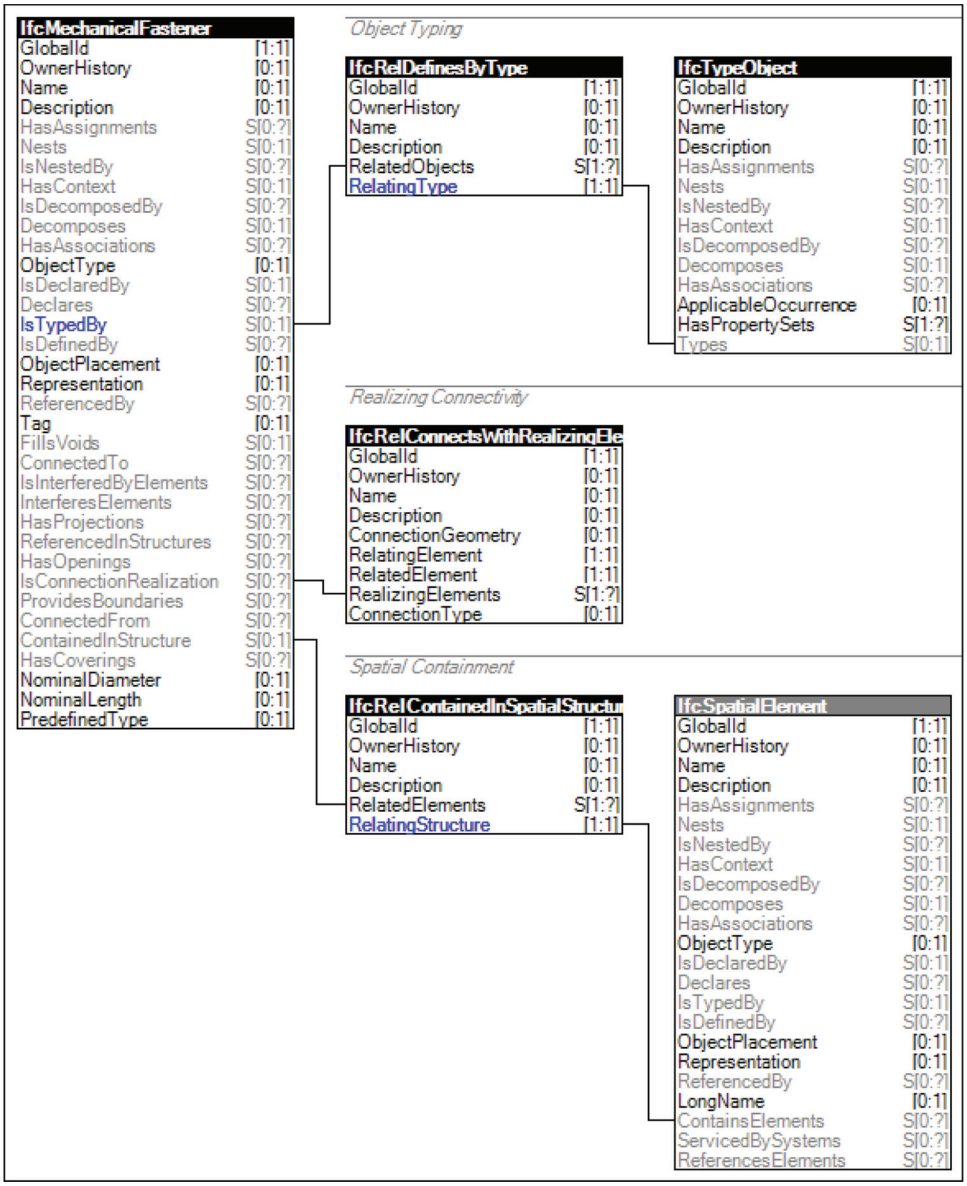


Fig. A.2f—IfcMechanicalFastener.

APPENDIX B—EXAMPLES

B.1—Selected example files

This appendix contains two examples of IFC files. The first example is a single straight bar; the second example is a single straight bar with a forged head. The examples give a glimpse of what an actual exchange file will look like.

These files are ISO 10303-21:16, otherwise known as a STEP part 21 files. They are written with the IFC4 schema. The information in the HEADER typically describes the application and authoring data while the DATA section describes the information of the model and visual representation.

B.2—Example 1—Single straight bar

```
ISO-10303-21;
HEADER;

FILE_DESCRIPTION(
/* description */ ('ViewDefinition [ACI131-EM15-v1.0]'),
/* implementation_level */ '2;1');

FILE_NAME(
/* name */ 'Rebar Example 1',
/* time_stamp */ '2016-07-26T12:14:51-05:00',
/* author */ ('barry'),
/* organization */ (''),
/* preprocessor_version */ 'ST-DEVELOPER v16 then by hand',
/* originating_system */ 'SDS/2 v2017.00',
/* authorisation */ '');

FILE_SCHEMA (('IFC4'));
ENDSEC;

DATA;
/*
* Project Data
*/

#10=IFCPROJECT('3tjlRgAab6OuwSW0_WYBCl',#3070,'ConcreteSetup',
'Top Level Container',$,'ConcreteSetup',$,(#1520),#2250);
#20=IFCRELAGGREGATES('0ZJtmKdOfDcOw3PHl9Rae2',#3070,'Project Contents',$,
#10,(#30));

/*
* Site in Project
*/

#30=IFCSITE('3ZUUUMeKz3ign0DBeoTqjo',#3070,'Job Site',$,$,#1400,$,$,
.ELEMENT.,$,$,0.,$,$);
#40=IFCRELAGGREGATES('3h5iOnJVnAhvF3ixCuEBhE',#3070,'Site Contents',$,#30,
(#50));

/*
* Building in Site
*/

#50=IFCBUILDING('2fC$TaudzF0vKgYVOHEmph',#3070,'Only Building',$,$,#1410,$,
$,.ELEMENT.,0.,0.,$);
#60=IFCRELCONTAINEDINSPATIALSTRUCTURE('1c06dmtpL1FB5dHGwPivft',#3070,
'Structure Contents',$,(#70),#50);

/*
```

```
* Bars in Building
*/

#70=IFCREINFORCINGBAR('11MsIzzDP0hv801vCAF$60',#3070,'rfb1','RB3/8','STRUCT
URAL',#1420,#1250,$,$,$,$,$,.USERDEFINED.,$);
#80=IFCRELDEFINESBYTYPE('2_0wrnSYPDtu2KGGdzeHZM',#3070,$,$,(#70),#110);

/*
* Material found on Bars
*/

#90=IFCMATERIAL('RB','#3','A615');
#100=IFCRELASSOCIATESMATERIAL('31wiplkcbDRQ2_AzvHYwKF',#3070,$,$,(#110),#90);

/*
* Definition of bar types will be associated with IfcReinforcingBar
*/
/*
dy: modified select data type
*/
#110=IFCREINFORCINGBARTYPE('3adL$u$JPCKeaIpBzWPd6m',#3070,'rfb1',$,$,(#120,#170),(#
1330),'rfb1',$,.NOTDEFINED.,0.03125,0.000763889,12.0,.PLAIN.,'NONE','NONE');

/*
* Property Sets
*/

#120=IFCPROPERTYSET('0WcMWR5OP9qQ25cHgaZ1gO',#3070,'
Pset_ACI_ReinforcingMaterial',$,(#130,#140,#150,#160));
#130=IFCPROPERTYSINGLEVALUE('Specification',$,IFCLABEL('ASTM_A615'),$);
#132=IFCPROPERTYSINGLEVALUE('SpecificationVersion',$,IFCLABEL('2016'),$);
#140=IFCPROPERTYSINGLEVALUE('Grade',$,IFCPRESSUREMEASURE(60.0),$);
#142=IFCPROPERTYSINGLEVALUE('CoatingSpecification',$,'UNCOATED',$);

#160=IFCPROPERTYSINGLEVALUE('RequiredOrigin',$,IFCLABEL('US'),$);

#170=IFCPROPERTYSET('2YxIKoP5L8ov3rDBSqCa4V',#3070,'
Pset_ACI_ReinforcingBarType',$,(#180));
#180=IFCPROPERTYSINGLEVALUE('Size',$,IFCLABEL('#3'),$);

#190=IFCPROPERTYSET('3$7jCkm6P5rA1h8UvRA1wu',#3070,'Pset_ACI_BarS
hape',$,(#200,#210,#220,#230));
#200=IFCPROPERTYSINGLEVALUE('StandardName',$,IFCLABEL('xyz'),$);
#210=IFCPROPERTYSINGLEVALUE('StandardVersion',$,IFCLABEL('xyz'),$);
#220=IFCPROPERTYSINGLEVALUE('ShapeName',$,IFCLABEL('xyz'),$);
/*
dy: positive length measure should be > 0
*/
#230=IFCPROPERTYSINGLEVALUE('DefaultInsideBendRadius',$,IFCPOSITIVELENGTHMEA
SURE(0.1),$);
/*#240=IFCPROPERTYSINGLEVALUE('Parameters',$,IFCPROPERTYTABLEVALUE(),$);*/

/*
* Geometric Shape definitions
*/

#1250=IFCPRODUCTDEFINITIONSHAPE($,$,(#1360));
#1260=IFCSTYLEDITEM(#1320,(#1270),$);
```

```
#1270=IFCPRESENTATIONSTYLEASSIGNMENT((#1280));
#1280=IFCURFACESTYLE('Color',.BOTH.,(#1290));
#1290=IFCSURFACESTYLERENDERING(#1300,0.,$,$,$,$,$,$,.MATT.);
#1300=IFCCOLOURRGB($,0.717647058823529,0.254901960784314,0.0549019607843137);
#1310=IFCCARTESIANTRANSFORMATIONOPERATOR3D(#1440,#1450,#1480,1.,#1430);
#1320=IFCMAPPEDITEM(#1330,#1310);
#1330=IFCREPRESENTATIONMAP(#1500,#1350);
/*
dy: modified on redefined attribute
*/
#1340=IFCGEOMETRICREPRESENTATIONSUBCONTEXT('Body','Model',*,*,*,*,#1520,1.,.M
ODEL_VIEW.,$);
#1350=IFCSHAPEREPRESENTATION(#1340,'Body','SweptSolid',(#1370));
#1360=IFCSHAPEREPRESENTATION(#1340,'Body','MappedRepresentation',(#1320));
#1370=IFCEXTRUDEDAREASOLID(#1390,#1510,#1430,12.);
#1380=IFCAXIS2PLACEMENT2D(#1490,#1460);
#1390=IFCCIRCLEPROFILEDEF(.AREA.,'rfb1',#1380,0.015625);
#1400=IFCLOCALPLACEMENT($,#1500);
#1410=IFCLOCALPLACEMENT(#1400,#1500);
#1420=IFCLOCALPLACEMENT(#1410,#1500);
#1430=IFCDIRECTION((0.,0.,1.));
#1440=IFCDIRECTION((1.,0.,0.));
#1450=IFCDIRECTION((0.,1.,0.));
/*
dy: true north should point to 2D IFCDIRECTION if populated
*/
#1451=IFCDIRECTION((0.,1.));
#1460=IFCDIRECTION((1.,0.));
#1470=IFCDIRECTION((0.,0.,-1.));
#1480=IFCCARTESIANPOINT((0.,0.,0.));
#1490=IFCCARTESIANPOINT((0.,0.));
#1500=IFCAXIS2PLACEMENT3D(#1480,#1430,#1440);
#1510=IFCAXIS2PLACEMENT3D(#1480,#1440,#1470);
/*
dy: true north should point to 2D IFCDIRECTION if populated
*/
#1520=IFCGEOMETRICREPRESENTATIONCONTEXT($,'Model',3,0.0,#1500,#1451);

/*
* Units needed by IFC
*/

#2000=IFCMEASUREWITHUNIT(IFCREAL(0.3048),#2190);
#2010=IFCMEASUREWITHUNIT(IFCREAL(0.0929),#2200);
#2020=IFCMEASUREWITHUNIT(IFCREAL(0.028317),#2210);
#2030=IFCMEASUREWITHUNIT(IFCREAL(0.45359237),#2220);
#2040=IFCMEASUREWITHUNIT(IFCREAL(4448.2216153),#2230);
#2050=IFCMEASUREWITHUNIT(IFCREAL(6894757.2932),#2240);

#2060=IFCDIMENSIONALEXPONENTS(1,0,0,0,0,0,0);
#2070=IFCDIMENSIONALEXPONENTS(2,0,0,0,0,0,0);
#2080=IFCDIMENSIONALEXPONENTS(3,0,0,0,0,0,0);
#2090=IFCDIMENSIONALEXPONENTS(0,1,0,0,0,0,0);
#2100=IFCDIMENSIONALEXPONENTS(1,1,-2,0,0,0,0);
#2110=IFCDIMENSIONALEXPONENTS(-1,1,-2,0,0,0,0);

#2120=IFCCONVERSIONBASEDUNIT(#2060,.LENGTHUNIT.,'FOOT',#2000);
#2130=IFCCONVERSIONBASEDUNIT(#2070,.AREAUNIT.,'SQUARE FOOT',#2010);
```

```
#2140=IFCCONVERSIONBASEDUNIT(#2080,.VOLUMEUNIT.,'CUBIC FOOT',#2020);
#2150=IFCCONVERSIONBASEDUNIT(#2090,.MASSUNIT.,'POUND',#2030);
#2160=IFCCONVERSIONBASEDUNIT(#2100,.FORCEUNIT.,'KIP',#2040);
#2170=IFCCONVERSIONBASEDUNIT(#2110,.PRESSUREUNIT.,'KSI',#2050);

#2180=IFCSIUNIT(*,.PLANEANGLEUNIT.,$,.RADIAN.);
#2190=IFCSIUNIT(*,.LENGTHUNIT.,$,.METRE.);
#2200=IFCSIUNIT(*,.AREAUNIT.,$,.SQUARE_METRE.);
#2210=IFCSIUNIT(*,.VOLUMEUNIT.,$,.CUBIC_METRE.);
#2220=IFCSIUNIT(*,.MASSUNIT.,.KILO.,.GRAM.);
#2230=IFCSIUNIT(*,.FORCEUNIT.,$,.NEWTON.);
#2240=IFCSIUNIT(*,.PRESSUREUNIT.,$,.PASCAL.);
#2250=IFCUNITASSIGNMENT((#2180,#2120,#2130,#2140,#2150,#2160,#2170));

/*
* Info needed for Owner History
*/

#3000=IFCTELECOMADDRESS(.OFFICE.,'Main headquarters',$,('800-443-0782',
'888-883-2492'),('402-441-4045','402-441-4043'),$,('info@sds2.com',
'support@sds2.com'),'http://www.sds2.com',$);

#3010=IFCPOSTALADDRESS(.OFFICE.,'Main headquarters',$,$,(
'1501 Old Cheney Road'),$,'Lincoln','Nebraska','68512','USA');

#3020=IFCAPPLICATION(#3040,'2016.13','SDS/2 v2016.13','SDS/2');
#3030=IFCORGANIZATION($,'SDS/2 User','End user',$,$);
#3040=IFCORGANIZATION($,'Design Data Corporation','Software vendor',$,(#3010,
#3000));
#3050=IFCPERSON($,'unknown',$,$,$,$,$,$);
#3060=IFCPERSONANDORGANIZATION(#3050,#3030,$);
#3070=IFCOWNERHISTORY(#3060,#3020,$,.ADDED.,0,$,$,1469553291);

/*
* End of Data
*/

ENDSEC;
END-ISO-10303-21;
```

B.3—Example 2—Single straight bar with forged head

```
ISO-10303-21;
HEADER;

FILE_DESCRIPTION(
/* description */ ('ViewDefinition [ACI131-EM15-v1.0]'),
/* implementation_level */ '2;1');

FILE_NAME(
/* name */ 'Rebar Example 1',
/* time_stamp */ '2016-07-26T12:14:51-05:00',
/* author */ ('barry'),
/* organization */ ("),
/* preprocessor_version */ 'ST-DEVELOPER v16 then by hand',
/* originating_system */ 'SDS/2 v2017.00',
/* authorisation */ ");

FILE_SCHEMA (('IFC4'));
```

```
ENDSEC;

DATA;
/*
* Project Data
*/

#10=IFCPROJECT(‘3tjlRgAab6OuwSW0_WYBCl’,#3070,’ConcreteSetup’,
‘Top Level Container’,$,’ConcreteSetup’,$,(#1520),#2250);
#20=IFCRELAGGREGATES(‘0ZJtmKdOfDcOw3PHl9Rae2’,#3070,’Project Contents’,$,
#10,(#30));

/*
* Site in Project
*/

#30=IFCSITE(‘3ZUUUMeKz3ign0DBeoTqjo’,#3070,’Job Site’,$,$,#1400,$,$,
.ELEMENT.,$,$,0.,$,$);
#40=IFCRELAGGREGATES(‘3h5iOnJVnAhvF3ixCuEBhE’,#3070,’Site Contents’,$,#30,
(#50));

/*
* Building in Site
*/

#50=IFCBUILDING(‘2fC$TaudzF0vKgYVOHEmph’,#3070,’Only Building’,$,$,#1410,$,
$,.ELEMENT.,0.,0.,$);
#60=IFCRELCONTAINEDINSPATIALSTRUCTURE(‘1c06dmtpL1FB5dHGwPivft’,#3070,
‘Structure Contents’,$,(#70),#50);

/*
* Bars in Building
*/

#70=IFCREINFORCINGBAR(‘11MsIzzDP0hv801vCAF$60’,#3070,’rfb1’,’#3
Rebar’,’STRUCTURAL’,#1420,#1250,$,$,$,$,$,.USERDEFINED.,$);
#80=IFCRELDEFINESBYTYPE(‘2_0wrnSYPDtu2KGGdzeHZM’,#3070,$,$,(#70),#110);

/*
* Material found on Bars
*/

#90=IFCMATERIAL(‘RB’,’#3’,’A615’);
#100=IFCRELASSOCIATESMATERIAL(‘31wiplkcbDRQ2_AzvHYwKF’,#3070,$,$,(#110),#90);

/*
* Definition of bar types will be associated with IfcReinforcingBar
*/

#110=IFCREINFORCINGBARTYPE(‘3adL$u$JPCKeaIpBzWPd6m’,#3070,’rfb1’,$,$,(#120,#170),(#
1330),’rfb1’,$,.NOTDEFINED.,0.03125,0.000763889,12.0,.PLAIN.,$,$);

/*
* Property Sets
*/

#120=IFCPROPERTYSET(‘0WcMWR5OP9qQ25cHgaZ1gO’,#3070,’
Pset_ACI_ReinforcingMaterial’,$,(#130,#140,#150,#160));
```

```
#130=IFCPROPERTYSINGLEVALUE('Specification',$,IFCLABEL('ASTM_A615'),$);
#132=IFCPROPERTYSINGLEVALUE('SpecificationVersion",$,IFCLABEL('2016'),$);
#140=IFCPROPERTYSINGLEVALUE('Grade',$,IFCPRESSUREMEASURE(60.0),$);
#142=IFCPROPERTYSINGLEVALUE('CoatingSpecification',$,IFCLABEL('UNCOATED'),$);

#160=IFCPROPERTYSINGLEVALUE('RequiredOrigin',$,IFCLABEL('US'),$);

#170=IFCPROPERTYSET('2YxIKoP5L8ov3rDBSqCa4V',#3070,'
Pset_ACI_ReinforcingBarType',$,(#180));
#180=IFCPROPERTYSINGLEVALUE('Size',$,IFCLABEL('#3'),$);
#182=IFCPROPERTYSINGLEVALUE('StartEndPrep',$,IFCLABEL('FORGED_HEAD'),$);

#190=IFCPROPERTYSET('3$7jCkm6P5rA1h8UvRA1wu',#3070,'Pset_ACI_BarS
hape',$,(#200,#210,#220,#230));
#200=IFCPROPERTYSINGLEVALUE('StandardName',$,IFCLABEL('xyz'),$);
#210=IFCPROPERTYSINGLEVALUE('StandardVersion',$,IFCLABEL('xyz'),$);
#220=IFCPROPERTYSINGLEVALUE('ShapeName',$,IFCLABEL('xyz'),$);
#230=IFCPROPERTYSINGLEVALUE('DefaultInsideBendRadius',$,IFCPOSITIVELENGTHMEA
SURE(0.1),$);

/*
* Geometric Shape definitions
*/

#1250=IFCPRODUCTDEFINITIONSHAPE($,$,(#1360));
#1260=IFCSTYLEDITEM(#1320,(#1270),$);
#1270=IFCPRESENTATIONSTYLEASSIGNMENT((#1280));
#1280=IFCURFACESTYLE('Color',.BOTH.,(#1290));
#1290=IFCSURFACESTYLERENDERING(#1300,0.,$,$,$,$,$,$,.MATT.);
#1300=IFCCOLOURRGB($,0.717647058823529,0.254901960784314,0.0549019607843137);
#1310=IFCCARTESIANTRANSFORMATIONOPERATOR3D(#1440,#1450,#1480,1.,#1430);
#1320=IFCMAPPEDITEM(#1330,#1310);
#1330=IFCREPRESENTATIONMAP(#1500,#1350);
#1340=IFCGEOMETRICREPRESENTATIONSUBCONTEXT('Body','Model',*,*,*,*,#1520,1.,
.MODEL_VIEW.,$);
/*#1350=IFCSHAPEREPRESENTATION(#1340,'Bar Geometry','lid',(#1371,#1600));*/
#1350=IFCSHAPEREPRESENTATION(#1340,'Bar Geometry','SweptDiskSolid',(#1650));
#1360=IFCSHAPEREPRESENTATION(#1340,'Bar','MappedRepresentation',(#1320));

/*
* Describe Bar Geomery
*/

#1371=IFCSWEPTDISKSOLID(#1374,0.375,$,$,$);
#1374=IFCPOLYLINE((#1480,#1375));
#1375=IFCCARTESIANPOINT((120.,0.,0.));

/*
*/

#1400=IFCLOCALPLACEMENT($,#1500);
#1410=IFCLOCALPLACEMENT(#1400,#1500);
#1420=IFCLOCALPLACEMENT(#1410,#1500);
#1430=IFCDIRECTION((0.,0.,1.));
#1440=IFCDIRECTION((1.,0.,0.));
#1450=IFCDIRECTION((0.,1.,0.));
#1451=IFCDIRECTION((0.,1.));
#1480=IFCCARTESIANPOINT((0.,0.,0.));
```

```
#1500=IFCAXIS2PLACEMENT3D(#1480,#1430,#1440);
#1520=IFCGEOMETRICREPRESENTATIONCONTEXT($,'Model',3,0.0,#1500,#1451);

/*
*/

#1600=IFCSWEPTDISKSOLID(#1610,0.7,$,$,$);
#1610=IFCPOLYLINE((#1480,#1620));
#1620=IFCCARTESIANPOINT((0.5,0.,0.));

#1650=IFCBOOLEANRESULT(.UNION.,#1371,#1600);

/*
* Units needed by IFC
*/

#2000=IFCMEASUREWITHUNIT(IFCREAL(0.3048),#2190);
#2010=IFCMEASUREWITHUNIT(IFCREAL(0.0929),#2200);
#2020=IFCMEASUREWITHUNIT(IFCREAL(0.028317),#2210);
#2030=IFCMEASUREWITHUNIT(IFCREAL(0.45359237),#2220);
#2040=IFCMEASUREWITHUNIT(IFCREAL(4448.2216153),#2230);
#2050=IFCMEASUREWITHUNIT(IFCREAL(6894757.2932),#2240);

#2060=IFCDIMENSIONALEXPONENTS(1,0,0,0,0,0,0);
#2070=IFCDIMENSIONALEXPONENTS(2,0,0,0,0,0,0);
#2080=IFCDIMENSIONALEXPONENTS(3,0,0,0,0,0,0);
#2090=IFCDIMENSIONALEXPONENTS(0,1,0,0,0,0,0);
#2100=IFCDIMENSIONALEXPONENTS(1,1,-2,0,0,0,0);
#2110=IFCDIMENSIONALEXPONENTS(-1,1,-2,0,0,0,0);

#2120=IFCCONVERSIONBASEDUNIT(#2060,.LENGTHUNIT.,'FOOT',#2000);
#2130=IFCCONVERSIONBASEDUNIT(#2070,.AREAUNIT.,'SQUARE FOOT',#2010);
#2140=IFCCONVERSIONBASEDUNIT(#2080,.VOLUMEUNIT.,'CUBIC FOOT',#2020);
#2150=IFCCONVERSIONBASEDUNIT(#2090,.MASSUNIT.,'POUND',#2030);
#2160=IFCCONVERSIONBASEDUNIT(#2100,.FORCEUNIT.,'KIP',#2040);
#2170=IFCCONVERSIONBASEDUNIT(#2110,.PRESSUREUNIT.,'KSI',#2050);

#2180=IFCSIUNIT(*,.PLANEANGLEUNIT.,$,.RADIAN.);
#2190=IFCSIUNIT(*,.LENGTHUNIT.,$,.METRE.);
#2200=IFCSIUNIT(*,.AREAUNIT.,$,.SQUARE_METRE.);
#2210=IFCSIUNIT(*,.VOLUMEUNIT.,$,.CUBIC_METRE.);
#2220=IFCSIUNIT(*,.MASSUNIT.,.KILO.,.GRAM.);
#2230=IFCSIUNIT(*,.FORCEUNIT.,$,.NEWTON.);

#2240=IFCSIUNIT(*,.PRESSUREUNIT.,$,.PASCAL.);
#2250=IFCUNITASSIGNMENT((#2180,#2120,#2130,#2140,#2150,#2160,#2170));

/*
* Info needed for Owner History
*/

#3000=IFCTELECOMADDRESS(.OFFICE.,'Main headquarters',$,('800-443-0782',
'888-883-2492'),('402-441-4045','402-441-4043'),$,('info@sds2.com',
'support@sds2.com'),'http://www.sds2.com',$);

#3010=IFCPOSTALADDRESS(.OFFICE.,'Main headquarters',$,$,(
'1501 Old Cheney Road'),$,'Lincoln','Nebraska','68512','USA');
```

```
#3020=IFCAPPLICATION(#3040,'2016.13','SDS/2 v2016.13','SDS/2');
#3030=IFCORGANIZATION($,'SDS/2 User','End user',$,$);
#3040=IFCORGANIZATION($,'Design Data Corporation','Software vendor',$,(#3010,
#3000));
#3050=IFCPERSON($,'unknown',$,$,$,$,$,$);
#3060=IFCPERSONANDORGANIZATION(#3050,#3030,$);
#3070=IFCOWNERHISTORY(#3060,#3020,$,.ADDED.,0,$,$,1469553291);

/*
* End of Data
*/

ENDSEC;
END-ISO-10303-21;
```

American Concrete Institute
Always advancing

As ACI begins its second century of advancing concrete knowledge, its original chartered purpose remains "to provide a comradeship in finding the best ways to do concrete work of all kinds and in spreading knowledge." In keeping with this purpose, ACI supports the following activities:

· Technical committees that produce consensus reports, guides, specifications, and codes.

· Spring and fall conventions to facilitate the work of its committees.

· Educational seminars that disseminate reliable information on concrete.

· Certification programs for personnel employed within the concrete industry.

· Student programs such as scholarships, internships, and competitions.

· Sponsoring and co-sponsoring international conferences and symposia.

· Formal coordination with several international concrete related societies.

· Periodicals: the ACI Structural Journal, Materials Journal, and Concrete International.

Benefits of membership include a subscription to Concrete International and to an ACI Journal. ACI members receive discounts of up to 40% on all ACI products and services, including documents, seminars and convention registration fees.

As a member of ACI, you join thousands of practitioners and professionals worldwide who share a commitment to maintain the highest industry standards for concrete technology, construction, and practices. In addition, ACI chapters provide opportunities for interaction of professionals and practitioners at a local level to discuss and share concrete knowledge and fellowship.

American Concrete Institute
38800 Country Club Drive
Farmington Hills, MI 48331
Phone: +1.248.848.3700
Fax: +1.248.848.3701

www.concrete.org

American Concrete Institute
Always advancing

38800 Country Club Drive
Farmington Hills, MI 48331 USA
+1.248.848.3700
www.concrete.org

The American Concrete Institute (ACI) is a leading authority and resource worldwide for the development and distribution of consensus-based standards and technical resources, educational programs, and certifications for individuals and organizations involved in concrete design, construction, and materials, who share a commitment to pursuing the best use of concrete.

Individuals interested in the activities of ACI are encouraged to explore the ACI website for membership opportunities, committee activities, and a wide variety of concrete resources. As a volunteer member-driven organization, ACI invites partnerships and welcomes all concrete professionals who wish to be part of a respected, connected, social group that provides an opportunity for professional growth, networking and enjoyment.

9 781945 487811