

SEMI E125-0923

SPECIFICATION FOR EQUIPMENT SELF DESCRIPTION (EqSD)

This Standard was technically approved by the Information & Control Global Technical Committee. This edition was approved for publication by the global Audits and Reviews Subcommittee on May 10, 2023. Available at www.semiviews.org and www.semi.org in September 2023; originally published July 2003; previously published October 2022.

NOTICE: The section symbol (§) is used to reference a numbered section, paragraph, or header and all subordinate paragraphs, headers, and embedded material (i.e., EXCEPTIONs, lists) therein. The paragraph symbol (§) is used to reference a particular numbered paragraph and embedded material therein. When duplicated (i.e., §§ and §§) the symbols are used to reference more than one section or paragraph, respectively.

NOTICE: Paragraphs entitled ‘NOTE:’ are not an official part of this Standard or Safety Guideline and are not intended to modify or supersede the official Standard or Safety Guideline. These have been supplied by the global technical committee to enhance the usage of the Standard or Safety Guideline.

Table of Contents

1 Purpose	3
2 Scope	3
3 Limitations	3
4 Referenced Standards and Documents	3
5 Terminology	4
6 Conventions	5
6.1 Document and Notation Conventions	5
6.2 Terms	7
6.3 Generic Class Definitions	7
7 Clients and Consumers	7
7.3 Client and Consumer Conventions	7
7.4 Common Errors	8
8 Background	9
9 Overview	9
9.1 Equipment Metadata Organization	9
9.2 Accessing Equipment Metadata	11
9.3 Mapping SECS Variables, Events, and Alarms	11
10 Interfaces for Managing Equipment Metadata	11
10.1 Equipment Metadata Manager	11
10.2 Metadata Client	16
11 Equipment Metadata	17
11.1 Management of Equipment Metadata	17
11.2 Equipment Structure	17
11.3 Equipment Nodes	17
11.4 Parameters	19
11.5 Exceptions	26
11.6 StateMachines	27
11.7 ObjTypes	37
11.8 SimpleEvent	39
11.9 Metadata Fingerprint	40
12 Privileges	41
13 Requirements for Compliance	41
14 Related Documents	42
Related Information 1	43
Related Information 2	48
Related Information 3	50

Related Information 4.....	51
Related Information 5.....	52

1 Purpose

1.1 This Specification describes a method for allowing equipment suppliers to provide a description of the variables, events, exceptions, and physical equipment configuration available from their equipment. With this information available for consumption by software systems, it can be used as a tool to aid the process of integrating equipment into a factory's automation system.

2 Scope

2.1 *In-Scope*

2.1.1 This Standard specifies the classes that suppliers are to use to describe essential data, events, and exceptions provided by their equipment. This Specification only describes information that is static in nature (i.e., information that does not change dynamically while the equipment is running). This Standard also specifies an interface that Clients can use to access this information.

2.1.2 This Specification applies to all semiconductor manufacturing equipment that supports the data acquisition interface defined in the SEMI Specification for Data Collection Management.

2.2 *Out of Scope*

2.2.1 This Specification does not define any new behavior required of the equipment other than that necessary for retrieving information describing equipment configuration, interfaces, and available data, and keeping this information current.

2.2.2 The details of any underlying concepts and behavioral models (e.g., carrier management, process/control job, etc.) that can be described by metadata are to be separately specified in a document dedicated to those concepts. Only the ability to describe the fact that a supplier has implemented such a concept and that a Client can discover this implementation and any data it can produce is in scope for this Specification.

2.2.3 This Specification does not require that the metadata provided by the equipment be directly human readable. It is expected that applications will be written to organize and present this information to human users in a form that is easier for end users to digest.

NOTICE: SEMI Standards and Safety Guidelines do not purport to address all safety issues associated with their use. It is the responsibility of the users of the Documents to establish appropriate safety and health practices, and determine the applicability of regulatory or other limitations prior to use.

3 Limitations

3.1 *Abstract Model*

3.1.1 This Specification is an abstract model only. Adjunct specifications shall be developed to bind this Specification to an implementation technology.

3.2 *Other Limitations*

3.2.1 This Specification does not define the mechanism used for determining how to locate EDA Applications that are to receive notifications defined by this Specification. It assumes that the equipment has established a communications context with any EDA Applications that can submit requests, and has provided any context necessary to send notifications to EDA Applications.

3.2.2 This Specification does not define the mechanism used for identifying or authenticating Clients. It assumes that Clients can be uniquely identified, and that the equipment is capable of distinguishing requests originating from different Clients.

4 Referenced Standards and Documents

4.1 *SEMI Standards and Safety Guidelines*

SEMI E5 — Specification for SEMI Equipment Communications Standard 2 Message Content (SECS-II)

SEMI E30 — Specification for the Generic Model for Communications and Control of Manufacturing Equipment (GEM)

SEMI E39 — Specification for Object Services: Concept, Behavior, and Services

SEMI E120 — Specification for the Common Equipment Model (CEM)

SEMI E132 — Specification for Equipment Client Authentication and Authorization

NOTE 1: SEMI E132 is being updated by the Information & Control Global Technical Committee.

SEMI E134 — Specification for Data Collection Management

NOTE 2: SEMI E134 is being updated by the Information & Control Global Technical Committee.

4.2 ISO® Standards¹

ISO 8601 — Data Elements and Interchange Formats – Information Interchange – Representations of Dates and Times

4.3 OMG® Standards²

OMG 01-09-67 — Unified Modeling Language®³

NOTICE: Unless otherwise indicated, all documents cited shall be the latest published versions.

5 Terminology

5.1 Abbreviations and Acronyms

5.1.1 Descriptions of many of the abbreviations and acronyms used in this Specification may be found in the SEMI International Standards Compilation of Terms, available on the SEMI Web site, <http://www.semi.org/>. In most cases, these terms are not included in this section.

5.1.2 *SQL* — Standard Query Language

5.1.3 *UML*®³ — *Unified Modeling Language*

5.2 Definitions

5.2.1 Definitions or descriptions of many of the terms used in this Specification may be found in the SEMI International Standards Compilation of Terms, available on the SEMI Web site, <http://www.semi.org/>. In most cases, these terms are not included in this section.

5.2.2 Related Information 5, § R5-1 contains useful definitions of Unified Modeling Language (UML) terms taken directly from the UML standard. UML terms used in this Specification conform to these definitions. Please refer to this section as needed.

5.2.3 *Client* — an application process that communicates with the equipment to access equipment resources and initiates the connection with the Equipment Server. The Client performs management functionality, and can receive notification and operational messages from the Equipment Server. [SEMI E132]

5.2.4 *Consumer* — an application process that communicates with the equipment to get data. The Equipment Server initiates the connection to a specific endpoint to communicate with the Consumer. The Consumer does not perform any management functionality, can receive notification messages from the Equipment Server, and always receives operational messages from the Equipment Server when it exists. [SEMI E132]

5.2.5 *EDA Application* — software application that implements at least one Client or Consumer. [SEMI E132]

5.2.6 *metadata* — data used to describe data. For example, if a tool can report an event with several associated variables under certain conditions, the metadata for that event would provide a description of what condition will produce the event, what the type and units are of each variable, and the id of the event itself.

¹ International Organization for Standardization, ISO Central Secretariat, 1, ch. de la Voie-Creuse, CP 56, CH-1211 Geneva 20, Switzerland; Telephone: +41.22.749.01.11, Fax: +41.22.733.34.30, <http://www.iso.org>. ISO trademark is owned by the International Organization for Standardization.

² Object Management Group, Inc., 250 First Ave. Suite 100, Needham, MA 02494, USA; Telephone: +1.781.444.0404; Fax: +1.781.444.0320; <http://www.omg.org>. OMG trademark is owned by Object Management Group.

³ Trademark is owned by Object Management Group.

5.2.7 *notification message* — type of message sent by the Equipment Server to notify the recipient something has changed on the equipment (typically related to status, management, or metadata). [SEMI E132]

5.2.8 *operational message* — type of message sent by the Equipment Server to either a Client or Consumer with collected or analysis data from the equipment. [SEMI E132]

6 Conventions

6.1 Document and Notation Conventions

6.1.1 *Unified Modeling Language (UML)* — This Specification uses the UML conventions for representing all class, state, object, and sequence diagrams. Refer to the UML specification for a detailed description of these conventions.

6.1.2 *Attribute Tables* — The table below provides an example of the tables used to list and describe attributes of classes defined in this Specification.

Table 1 Attribute Table Format

Attribute Name	Definition	Required	Form
		Y, N, or C	Refer to list below

6.1.2.1 *Required* — Specifies whether the attribute is required. The possible values are yes (Y), no (N), and conditional (C). Where the attribute is conditionally required, the condition is specified with the attribute definition.

6.1.2.2 *Form* — Defines the data type of the attribute. The terms used to describe data types in this column are defined in the SEMI International Standards Compilation of Terms, or are included as part of the Specification. Refer to the compilation of terms for the definition of SEMI type name meanings.

6.1.3 *UML Associations* — The mechanism used for representing UML associations between classes is implementation dependent. This Standard is abstract in nature, and does not specify or imply any such mechanism. Any adjunct standard that provides an implementation of this Specification shall include a description of the mechanism used for representing the UML associations shown in this Specification.

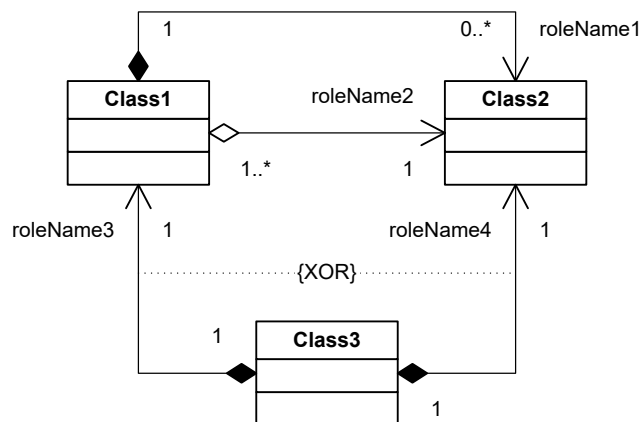


Figure 1
Association Notation

6.1.3.1 *Association Notation* — This Specification uses the UML aggregation diamond adornment, role names, end multiplicities, and navigability in all class diagrams specifying associations. Unadorned associations are not used, for economy of notation. Refer to OMG 01-09-67 for details. All role names are public by default; therefore the visibility symbols are not used.

6.1.3.1.1 Open diamond adornments indicate that instances of the target class may be shared among aggregate classes. Closed diamond adornments indicate that target instances belong to at most one composite class. The part-whole semantics of the aggregation/composition symbol is not significant in this Specification. Refer to OMG 01-09-67 for further information.

6.1.3.1.2 Some diagrams use the UML ‘XOR’ constraint where applicable. An ‘XOR’ constraint indicates that only one of many possible associations can be instantiated at any one time for a given instance of the association class. Refer to OMG 01-09-67 for further information.

6.1.4 *Association Tables* — The table below provides an example of the tables used to list and describe associations between classes defined in this Specification.

Table 2 Association Table Format

<i>Association Role Name</i>	<i>Definition</i>	<i>Comments</i>

6.1.4.1 *Association Role Name* — The name of the association role being specified.

6.1.4.2 *Definition* — Describes the function or purpose of the association.

6.1.4.3 *Comments* — Any additional comments or notes regarding the association.

6.1.5 *Operation Definition Tables* — The table below provides an example of the tables used to list and describe the interface operations defined in this Specification.

Table 3 Operation Definition

<i>Operation</i>	<i>Description</i>	<i>Type</i>
		Refer to list below

6.1.5.1 *Operation* — Specifies the name of the operation.

6.1.5.2 *Type* — Specifies the messaging semantics of the operation. Only Request-Reply (RR) semantics are used in this Specification. Request-Reply messages are messages that involve an initiator and a receiver. In a Request-Reply exchange, the initiator sends a single request message to the receiver, and the receiver sends a single reply message to that request back to the initiator. Fire-and-Forget messages are messages that involve a sender and a receiver. In a Fire-and-Forget exchange, the sender sends a single message to the receiver, with no associated response.

6.1.6 *Operation Argument Definition Table* — The table below provides an example of the tables used to list and describe arguments for interface operations defined in this Specification.

Table 4 Operation Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
		Refer to list below	Refer to list below

6.1.6.1 *Argument* — Specifies the name of the argument.

6.1.6.2 *Kind* — Specifies whether the argument is an ‘in’, ‘out’, or ‘error’ argument for the operation. ‘Error’ arguments always function as ‘out’ arguments, but indicate that the operation did not complete successfully.

6.1.6.3 *Form* — Defines the data type of the argument. The terms used to describe data types in this column are defined in the SEMI International Standards Compilation of Terms, or are included as part of the Specification. Refer to the compilation of terms for the definition of SEMI type name meanings.

6.2 Terms

6.2.1 *Class, Type* — This Specification uses the terms ‘class’ and ‘type’, interchangeably to refer to a UML class.

6.3 Generic Class Definitions

6.3.1 This Specification does not define a parameter type or unit class required by classes Parameter and Attribute. A Subordinate Standard to this Specification is expected to identify the parameter type and unit classes that are used with the specific implementation technology. The parameter type class shall include a text name attribute for unique identification. The unit class shall include a text unitId attribute that identifies each unit uniquely and a text unitSymbol attribute.

7 Clients and Consumers

7.1 With a networked environment, it is possible for software applications to be on different machines and fulfill different functions for the equipment users.

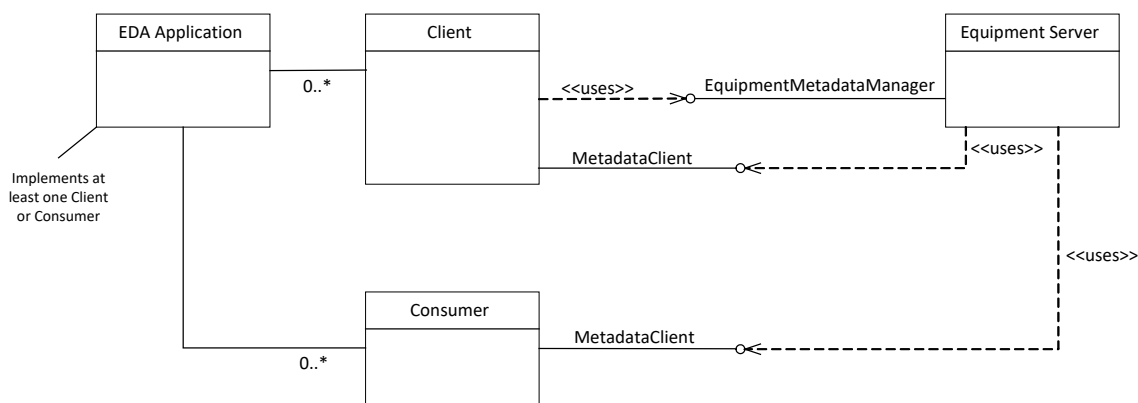


Figure 2
Session Interfaces

7.2 This Specification uses SEMI E132 for authentication and authorization.

7.2.1 Refer to SEMI E132 for details on notification messages, operational messages and how to specify who receives them when establishing a session.

7.3 *Client and Consumer Conventions* — For this Specification, the following conventions are used:

7.3.1 *Client* — This application process initiates the connection to the Equipment Server.

- It is associated with one SessionId.
- It uses the EquipmentMetadataManager interface to perform management functionality (e.g., querying the equipment model structure to determine what data is available for collection).
- The Client can receive notification messages from Equipment Server by implementing the MetadataClient interface.
- The Client can receive operational messages sent from the Equipment Server. (When a Consumer is specified for the session, the Client does not receive operational messages.) Currently, SEMI E125 does not define operational messages. (Refer to ¶ 10.2.2.)

7.3.2 *Consumer* — The Equipment Server initiates the connection to this application process.

- It is associated with one SessionId.
- It does not do any management functionality.
- The Consumer always implements the MetadataClient interface.

- Can receive notification messages.
- When present, the Consumer always receives operational messages sent from the Equipment Server. (When the Consumer is specified for the session, the Client cannot receive operational messages.) Currently, SEMI E125 does not define operational messages. (Refer to ¶ 10.2.2.)

7.3.3 *Equipment Server* — This software application implements the EquipmentMetadataManager interface. It uses the MetadataClient interface to send notification messages to the appropriate recipients (i.e., Clients and Consumers).

7.3.4 *EDA Application* — Software application that implements one of the following scenarios:

- one or more Clients
- one or more Consumers
- one or more Clients and one or more Consumers

7.4 *Common Errors* — Common errors independent of this Specification can be reported when they are applicable and not explicitly defined as SEMI E125-specific errors.

7.4.1 A Subordinate Standard shall identify how common errors are used with the implementation technology described in that Subordinate Standard.

7.4.2 Common errors defined in Table 5 shall be supported by all SEMI E125 operations. (For example, the GetUnits() operation can return the common error ‘System shutting down’, even though it is not explicitly listed as an error parameter in Table 8, GetUnits Argument Definitions.)

Table 5 Common Errors Parameter Definitions

<i>Parameter</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
error	System not ready.	error	Text, shall include the text ‘System not ready’.
error	System shutting down.	error	Text, shall include the text ‘System shutting down’.
error	Specified configuration is invalid.	error	Text, shall include the text ‘Specified configuration is invalid’, and a description of the invalid configuration.

7.4.3 Common SEMI E132 errors defined in Table 6 shall be supported by all SEMI E125 operations. (For example, the GetUnits() operation can return the SEMI E132 error ‘Unrecognized session’, even though it is not explicitly listed as an error parameter in Table 8, GetUnits Argument Definitions.)

Table 6 Common SEMI E132 Errors Parameter Definitions

<i>Parameter</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
error	Unauthorized operation.	error	Structured data, of type SEMI E132 UnauthorizedOperationError
error	Unrecognized session.	error	Text, shall include the text ‘Unrecognized session’ and the bad session id.
error	Unrecognized equipment identifier	error	Text, shall include the text ‘Unrecognized equipment identifier’ and the bad equipment identifier.

8 Background

8.1 Motivation

8.1.1 This Specification is designed to help address the problem of integrating semiconductor equipment from a variety of suppliers into an automated factory environment, and to provide a standard mechanism for describing information and data that the equipment can provide for a wide variety of uses. Refer to Related Information 1, ¶ R1-3.1 for an overview of some example use cases.

8.1.2 This Specification defines the information that is needed to communicate essential features of the equipment to software applications, and provides extensibility features to allow suppliers to communicate information unique to their equipment or that may not be specified in a SEMI Standard.

8.1.3 Equipment metadata is designed to be useful to applications that provide an interface to human end users for the purposes of understanding and browsing equipment configuration, available data items, events, exceptions, etc. In many cases, these users will be referring to equipment metadata so they can perform data collection for a variety of objectives (diagnostics, troubleshooting, process control, etc.).

8.1.4 Equipment metadata is also designed to be useful to applications written to take advantage of standardized state models and other data so that discovery and usage of this information can be automated, without requiring human intervention. Diagnostics, health monitoring, or other supplier-provided applications that understand the details of specific equipment types, for example, can also discover and use common equipment-specific configuration and data without requiring manual assistance.

9 Overview

9.1 Equipment Metadata Organization

9.1.1 Figure 3 shows the concepts that are included in this Specification and their relationship to the SEMI E120 Common Equipment Model. A brief description of the purpose of each metadata concept is provided in this section only as a conceptual overview. For the detailed specification of the corresponding classes, refer to § 11.

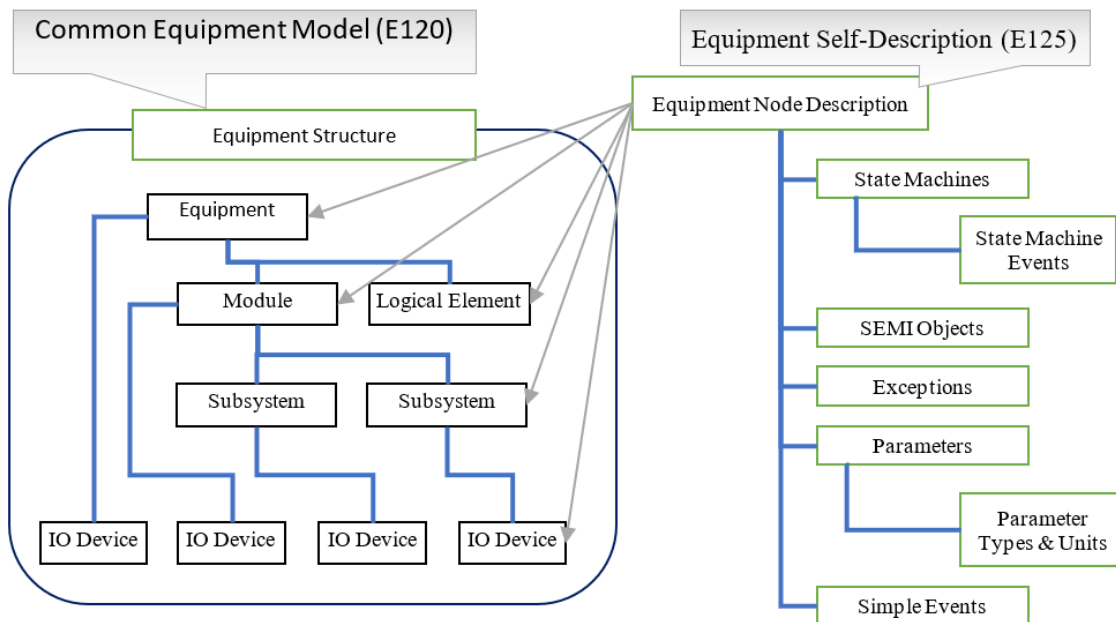


Figure 3
Organization of Metadata Concept

9.1.2 Equipment Node Description

9.1.2.1 This Specification defines a class for the Equipment Node Description that associates events, exceptions, parameters, state machines and SEMI objects with any node in the equipment hierarchy. SEMI E120 defines the Nameable class that is used to describe the physical equipment structure and related elements; this Specification defines the ‘node’ representing the equipment component as a derivative of the Nameable class. Equipment node description classes are defined in § 11.3. Figure 4 shows the relationship between the Equipment Node Description and other classes. Descriptions of events (StateMachine Event and Simple Event), exceptions, parameters, and SEMI objects are defined outside of the Equipment Node Description and are linked to the Equipment Node Description via Reference or Instance.

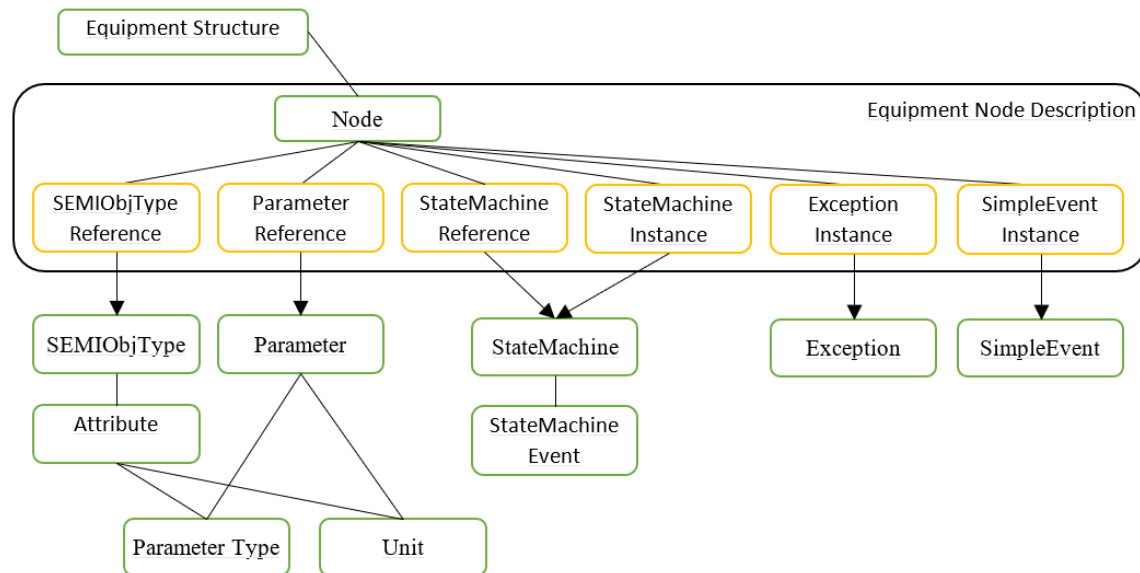


Figure 4
Relationship of Metadata Components

9.1.3 Parameters

9.1.3.1 This Specification defines classes that can be used to describe any variables that are provided by the equipment, and refers to all such variables as ‘Parameters’. Parameters can be used to model any concept that can be represented as a data type. For example, the SEMI E30 concepts of equipment constants, data variables, and status variables can all be represented by a parameter. Any type of data that the equipment can produce (e.g., measurement results on metrology equipment) can be represented by a Parameter. Parameter description is defined in § 11.4.

9.1.4 State Machine and StateMachine Events

9.1.4.1 This Specification defines classes that can be used to describe the state machines (and the corresponding transitions and state machine events) that are implemented by the equipment. These classes can be used to describe any SEMI-defined state machine or any supplier-defined state machine. This Specification also defines classes that can be used to describe the Parameters that are available for reporting when a specific state machine event occurs. State machine description is defined in § 11.6.

9.1.5 SEMI Objects

9.1.5.1 This Specification defines classes that can be used to describe any SEMI E39 ObjTypes that are implemented by the equipment. ObjType attributes are described using similar classes that are used to describe Parameters and types. SEMI ObjType description is defined in § 11.6.

9.1.6 Exceptions

9.1.6.1 This Specification defines classes that can be used to describe the existence of error conditions that the equipment can detect and report as exceptions. An exception can represent any SEMI E30 alarm or equipment-supplier-defined exception that can be reported to EDA Applications. Exception description is defined in § 11.5.

9.1.7 Simple Events

9.1.7.1 This Specification defines classes that can be used to describe events implemented by the equipment that are not associated with state machines. This Specification also allows an association with any available parameter. SimpleEvents are defined in § 11.8.

9.2 Accessing Equipment Metadata

9.2.1 Equipment metadata is directly associated with each unique equipment installation within a factory. Access to metadata is provided by a collection of operations available from a specific equipment installation. This interface is described in § 10.1.

9.3 Mapping SECS Variables, Events, and Alarms

9.3.1 In many cases, data collection based on the SECS-II protocol (SEMI E5) and the Generic Equipment Model (SEMI E30/GEM) exists in parallel to data collection as defined in the combination of this Specification and SEMI E134 Data Collection Management. Much of the data specified in the equipment metadata is also expected to be available through SECS communication. As an aid to the user, the equipment metadata includes mappings from SECS/GEM variables, events, and alarms to the corresponding metadata concepts.

10 Interfaces for Managing Equipment Metadata

10.1 Equipment Metadata Manager

10.1.1 Figure 5 shows the interface used to access equipment metadata descriptions. The Equipment Server shall implement the EquipmentMetadataManager interface. SEMI E125 does not define any error codes specific to this Specification.

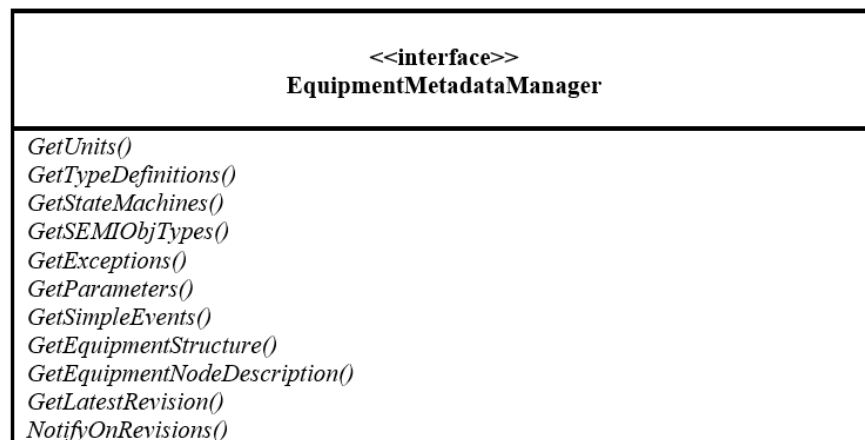


Figure 5
Interface for Accessing Equipment Metadata

10.1.2 EquipmentMetadataManager Operations

Table 7 EquipmentMetadataManager Operation Definition

<i>Operation</i>	<i>Description</i>	<i>Type</i>
GetUnits	Retrieves all unit metadata provided by the equipment.	RR
GetTypeDefinitions	Retrieves all type definitions provided by the equipment.	RR
GetStateMachines	Retrieves all state machine metadata provided by the equipment.	RR
GetSEMIObjTypes	Retrieves all SEMI ObjTypes metadata provided by the equipment.	RR
GetExceptions	Retrieves all exception metadata provided by the equipment.	RR
GetParameters	Retrieves all parameter metadata provided by the equipment.	RR
GetSimpleEvents	Retrieves all simple event metadata provided by the equipment.	RR
GetEquipmentStructure	Retrieves all equipment structural metadata provided by the equipment.	RR
GetEquipmentNodeDescriptions	Retrieves all requested equipment node description metadata.	RR
GetLatestRevision	Retrieve the last date and time at which the equipment metadata was revised.	RR
NotifyOnRevisions	Request that the equipment notify the requestor when changes to the metadata are made.	RR

10.1.2.1 *GetUnits* — Upon receiving this request, the equipment shall return all units metadata provided by the equipment.

10.1.2.1.1 *GetUnits Operation Arguments*

Table 8 GetUnits Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
units	All units defined for the equipment.	out	Unordered list of elements of a unit class identified by a Subordinate Standard to this Specification.

10.1.2.2 *GetTypeDefinitions* — Upon receiving this request, the equipment shall return all type definition metadata provided by the equipment.

10.1.2.2.1 *GetTypeDefinitions Operation Arguments*

Table 9 GetTypeDefinitions Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
typeDefinitions	All type definitions provided by the equipment.	out	Unordered list of elements of a parameter type class identified by a Subordinate Standard to this Specification.

10.1.2.3 *GetStateMachines* — Upon receiving this request, the equipment shall return all state machine metadata provided by the equipment.

10.1.2.3.1 *GetStateMachines Operation Arguments*

Table 10 GetStateMachines Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
stateMachines	All state machine metadata provided by the equipment.	out	Unordered list of elements of type StateMachine, described in § 11.6.

10.1.2.4 *GetSEMIObjTypes* — Upon receiving this request, the equipment shall return all OSS ObjType metadata provided by the equipment.

10.1.2.4.1 *GetSEMIObjTypes Operation Arguments*

Table 11 GetSEMIObjTypes Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
objTypes	All SEMI ObjType metadata provided by the equipment.	out	Unordered list of elements of type SEMIObjType, described in ¶ 11.7.1.1. If no SEMI ObjTypes are provided, this list shall be empty.

10.1.2.5 *GetExceptions* — Upon receiving this request, the equipment shall return all exception metadata provided by the equipment.

10.1.2.5.1 *GetExceptions Operation Arguments*

Table 12 GetExceptions Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
exceptions	All exception metadata provided by the equipment.	out	Unordered list of elements of type Exception, described in § 11.5.

10.1.2.6 *GetParameters* — Upon receiving this request, the equipment shall return all parameter metadata provided by the equipment.

10.1.2.6.1 *GetParameters Operation Arguments*

Table 13 GetParameters Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
parameters	All parameter metadata provided by the equipment.	out	Unordered list of elements of type Parameter, described in § 11.4.

10.1.2.7 *GetSimpleEvents* — Upon receiving this request, the equipment shall return all Simple Event metadata provided by the equipment.

10.1.2.7.1 *GetSimpleEvents Operation Arguments*

Table 14 GetSimpleEvents Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
simpleEvents	All simple event metadata provided by the equipment.	out	Unordered list of elements of type SimpleEvent, described in § 11.8.

10.1.2.8 *GetEquipmentStructure* — Upon receiving this request, the equipment shall return all equipment structural metadata provided by the equipment.

10.1.2.8.1 *GetEquipmentStructure Operation Arguments*

Table 15 GetEquipmentStructure Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
equipmentStructure	All equipment structural metadata provided by the equipment.	out	Structured data, of a type derived from the SEMI E120 EquipmentElement, that represents the equipment as a whole.

10.1.2.9 *GetEquipmentNodeDescriptions* — Upon receiving this request, the equipment shall return all requested equipment node description metadata defined for the equipment. If the request includes unrecognized equipment node identifiers, the equipment shall return only node descriptions for recognized identifiers. All unrecognized identifiers shall be returned with the response.

10.1.2.9.1 *GetEquipmentNodeDescriptions Operation Arguments*

Table 16 GetEquipmentNodeDescriptions Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
equipmentNodeIds	The identifiers of the requested equipment node descriptions.	in	List of text, each entry equal to a valid SEMI E120 Locator value (refer to SEMI E120, Related Information 2). If an empty list is provided, the equipment shall return node descriptions for all equipment nodes.
filter	Specify what items, if any, shall be omitted from the returned equipmentNodeDescriptions	in	Positive Integer If the filter value is 0, then no filter is applied and all items are returned in the equipmentNodeDescriptions. If bit 1 is 1, then exceptionInstances shall be omitted. If 0, then exceptionInstances shall be included. If bit 2 is 1, then parameterReferences shall be omitted. If 0, then parameterReferences shall be included. If bit 3 is 1, then semiObjTypeReferences shall be omitted. If 0, then semiObjTypeReferences shall be included. If bit 4 is 1, then simpleEventInstances shall be omitted. If 0, then simpleEventInstances shall be included. If bit 5 is 1, then stateMachineInstances shall be omitted. If 0, then stateMachineInstances shall be included. If bit 6 is 1, then stateMachineReferences shall be omitted. If 0, then stateMachineReferences shall be included. For example, if the filter value is 5, then both exceptionInstances and semiObjTypeReferences are omitted in the equipmentNodeDescriptions.
equipmentNodeDescriptions	All equipment node description metadata.	out	Structured data of type NodeDescriptionResult.

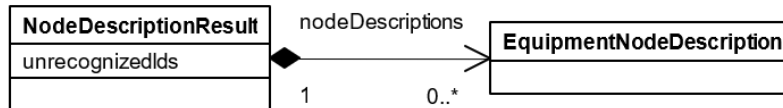


Figure 6
Interface for Accessing Equipment Metadata

10.1.2.10 *NodeDescriptionResult* — This class provides the list of requested EquipmentNodeDescription elements and any unrecognized equipment node id's that may have been provided with the request.

10.1.2.10.1 *NodeDescriptionResult* Attribute Definition Table

Table 17 NodeDescriptionResult Attribute Definition

Attribute Name	Definition	Form
unrecognizedIds	Any unrecognized equipment node identifiers provided in the original request.	List of text, each entry equal to an unrecognized equipment node identifier provided in the GetEquipmentNodeDescriptions request.

10.1.2.10.2 *NodeDescriptionResult* Association Definition Table

Table 18 NodeDescriptionResult Association Definition

Association Role Name	Definition	Comments
nodeDescriptions	The requested equipment node description metadata.	Zero or more elements of type EquipmentNodeDescription. The equipment shall return node description metadata for all recognized equipment node identifiers. If the same node identifier appears more than once in the request, the equipment shall return only one corresponding node description.

10.1.2.11 *GetLatestRevision* — Upon receiving this request, the equipment shall return the most recent date and time at which the metadata contents were changed. The equipment shall store the date and time of the most recent change in nonvolatile memory, along with the metadata fingerprint of the current equipment metadata. The equipment shall not generate false revision dates, in which there is no change in the available metadata, but the latest revision date/time is changed.

10.1.2.11.1 *GetLatestRevision* Operation Arguments

Table 19 GetLatestRevision Argument Definitions

Argument	Description	Kind	Form
revisionDateTime	The most recent date and time at which the metadata was changed.	out	Text, formatted according to ISO 8601. Implementations may specify additional formatting restrictions within the ISO 8601 specification.
metadata Fingerprint	Fingerprint of the current equipment metadata, generated after the most recent metadata change.	out	Text, equal to the metadata fingerprint (refer to per § 11.9).

10.1.2.12 *NotifyOnRevisions* — Upon receiving this request, the equipment shall either enable or disable metadata revision notification messages for this session. The equipment shall store the session’s preference in nonvolatile memory.

10.1.2.12.1 *NotifyOnRevisions Operation Arguments*

Table 20 NotifyOnRevisions Argument Definitions

<i>Argument</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
enable	Specifies the notification preference.	in	Boolean. If true, the session’s Client and/or Consumer is to be notified of metadata revisions. If false, the session’s Client and/or Consumer is to not be notified of metadata revisions.

10.1.2.12.2 Clients shall invoke this operation with the ‘enable’ argument set to ‘true’ in order for this session’s Client and/or Consumer to receive notifications. The default behavior for revision notifications for a given session is disabled. The equipment shall automatically disable notification for the session if the equipment determines that the Client and/or Consumer is no longer accessible to receive such notifications. In this case, the Client shall invoke this operation in order to re-enable notifications.

10.1.2.12.3 If the ‘enable’ argument provided is ‘true’, the equipment shall begin notifying the session’s Client and/or Consumer (if it is not already doing so) whenever a change in the metadata is detected, starting with the first detected change after the Client invokes the operation. If the change is detected while the equipment is in a disabled communications state, the equipment shall notify the interested Client and/or Consumer for the session at the first opportunity when its communication status permits.

10.1.2.12.4 If the ‘enable’ argument provided is ‘false’, the equipment shall cease sending revision notifications to the session’s Client and/or Consumer, if it is currently doing so. The equipment shall ignore duplicate enable/disable requests (refer to § 3, Limitations).

10.2 *Metadata Client*

10.2.1 Figure 7 shows the interface and argument data types in the Metadata Client interface that the Equipment Server shall use to communicate metadata related events.

10.2.2 There are no SEMI E125 operational messages defined.

10.2.3 SEMI E125 notification messages shall be MetadataRevised.

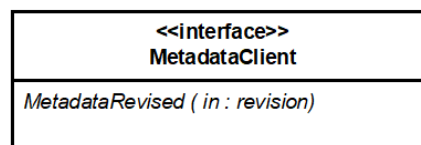


Figure 7
Metadata Client Interface

10.2.4 MetadataClient Operations

Table 21 MetadataClient Operation Definition

Operation	Description	Type
MetadataRevised	Notifies the Client and/or Consumer that the metadata has recently changed.	FF

10.2.4.1 *MetadataRevised* — The equipment shall send this notification whenever it detects a change in the metadata, if the Client has requested to be notified through a prior invocation of the ‘NotifyOnRevisions’ operation of the EquipmentMetadataManager interface.

10.2.4.1.1 MetadataRevised Operation Arguments

Table 22 MetadataRevised Argument Definitions

Argument	Description	Kind	Form
revisionDateTime	The date and time at which the metadata change was detected by the equipment.	in	Text, formatted according to ISO 8601. Implementations may specify additional formatting restrictions within the ISO 8601 specification.
metadata Fingerprint	Fingerprint of the current equipment metadata, generated after the most recent metadata change.	in	Text, equal to the metadata fingerprint (refer to § 11.9).

11 Equipment Metadata

11.1 Management of Equipment Metadata

11.1.1 When equipment configuration changes result in a change to any information described using the classes in § 11 of this Specification or by the SEMI Common Equipment Model, the equipment shall update the metadata description accordingly at the time of the change. If Clients have requested revision notification (refer to ¶¶ 10.1.2.11 and 10.1.2.12), the equipment shall send notification(s) at the time of the change, or at the first opportunity when its communication status permits. This includes reconfiguration of the equipment after its introduction into the factory. Ensuring that all aspects of equipment metadata are kept up-to-date and available to the factory is a critical aspect of maintaining equipment metadata.

11.2 Equipment Structure

11.2.1 The physical equipment structure shall be described according to the SEMI Common Equipment Model. Suppliers shall provide a model of those elements of their equipment for which it is necessary and practical to facilitate (at least) utilization tracking, process control, and technical support objectives.

11.3 Equipment Nodes

11.3.1 Equipment Node Description

11.3.1.1 Figure 8 shows how to describe the parameters, events, objTypes, state machines and exceptions available from an element (or node) of the equipment hierarchy. Providing these descriptions indicates that the specified node acts as the source of all Simple Events, Exceptions, ObjTypes, State Machines, StateMachine Events and Parameters that are included in the description. All Parameters, Exceptions, ObjTypes, Simple Events, State Machines and StateMachine Events associated with a given equipment node are independent of the Simple Events, Exceptions, ObjTypes, State Machine, StateMachine Events and Parameters associated with other equipment nodes.