

SEMI E90-0301

SPECIFICATION FOR SUBSTRATE TRACKING

This specification was technically approved by the Global Information & Control Committee and is the direct responsibility of the North American Information & Control Committee. Current edition approved by the North American Regional Standards Committee on November 22, 2000. Initially available at www.semi.org January 2001; to be published March 2001. Originally published September 1999; previously published October 2000.

1 Purpose

1.1 The purpose of this standard is to provide the standard services of equipment to track substrates (manufactured product) in manufacturing equipment. This standard defines the concepts and behaviors for the information management of substrates, as well as the messages/services.

2 Scope

2.1 Essentially, information about substrates must be managed by the factory system, while the equipment is required to provide the services for the substrate information management. This standard addresses the requirement for the equipment services to manage information of substrates that reside in the equipment.

2.2 The scope of this standard is to define the information services of equipment that can be requested by the user. To clarify required services, the concepts and behaviors of the substrate and the substrate location are defined.

2.3 This standard is applicable to any manufacturing equipment that handles substrates. To implement these services, the equipment and factory system must be integrated by means of a communication link.

2.4 This standard does not purport to address safety issues, if any, associated with its use. It is the responsibility of the users of this standard to establish appropriate safety and health practices and determine the applicability of regulatory limitations prior to use.

3 Limitations

3.1 This standard assumes that the substrate(s) to be managed has been resident in the equipment and has been given a substrate identifier.

3.2 The service for a substrate is valid while the substrate is registered in the equipment. That is, the service is available from the moment that the substrate is registered into the equipment and, to the moment that the substrate is removed from the equipment.

3.3 This standard is only for information services. It does not address any mechanism to read or write information from/to the substrate.

4 Referenced Standards

4.1 SEMI Standards

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

SEMI E39 — Object Services Standard: Concepts, Behavior, and Services

SEMI E53 — Event Reporting

SEMI E87 — Provisional Specification for Carrier Management (CMS)

NOTE 1: As listed or revised, all documents cited shall be the latest publications of adopted standards.

5 Terminology

5.1 Definitions

5.1.1 *carrier ID* — the name to identify a specific substrate carrier.

5.1.2 *carrier slot* — physical location capable of holding a substrate within cassette type carrier.

5.1.3 *carrier slot map* — the registry of substrates to the substrate carrier slots.

5.1.4 *carrier substrate location* — a substrate location within a substrate carrier capable of holding a substrate.

5.1.5 *default substrate ID (default ID)* — the substrate ID assigned to the substrate when no substrate ID information is given by the user but the carrier ID for the source carrier is known. The default ID is the combined text of the source carrier ID and the slot number.

5.1.6 *equipment substrate location* — a substrate location on a equipment resource.

5.1.7 *location ID* — the name of a material location.

5.1.8 *lot* — a group of one or more substrates of the same type. A lot must be organized by the user. The group may be referred to for tracking of substrates in the factory.

5.1.9 *material location* — an identifiable place within the equipment or carrier where material can be held.

5.1.10 *register* — an operation that adds the substrate object to the equipment's database. This operation is

performed automatically when the equipment receives both a carrier and information from the host about the contents of the carrier. The operation is also performed automatically when the equipment detects the substrate ID.

5.1.11 *remove* — the operation that removes a substrate from the equipment.

5.1.12 *service* — represents a function offered to a user by a provider. A service consists of a sequence of service primitives, each described by a list of parameters.

5.1.13 *substrate* — the basic unit of material on which work is performed to create a product. Examples include wafers, die, plates used for masks, flat panels, circuit boards, and leadframes.

5.1.14 *substrate carrier* — a carrier to hold substrates to be transferred to/from the equipment. A substrate carrier has one or more position to hold substrates (carrier substrate location).

5.1.15 *substrate history* — ordered set of information about the locations visited by the substrate.

5.1.16 *substrate ID* — identifier of a substrate.

5.1.17 *substrate location* — a material location which is capable of holding a substrate. For example, but not limited to, process modules, transfer subsystems, wafer chucks, robot end effector, and carrier slots.

5.1.18 *substrate type* — represents the type of the substrate, such as wafers, CDs, flat panels, or masks.

5.2 Data Type

5.2.1 *Boolean* — may take on one of two possible values, equating to TRUE or FALSE.

5.2.2 *enumerated* — may take on one of a limited set of possible values. These values may be given logical names, but they may be represented by any single-item data type except floating point.

5.2.3 *floating point* — may take on any single numeric value, positive or negative. Messaging protocol may impose a limit on the range of possible values.

5.2.4 *form* — type of data: positive integer, unsigned integer, integer, floating point, enumerative, Boolean, text, formatted text, structure, list, and ordered list.

5.2.5 *formatted text* — text with an imposed format. This could be by position, by use of special characters, or both.

5.2.6 *integer* — may take on the value of any negative or unsigned integer. Messaging protocol may impose a limit on the range of possible values.

5.2.7 *list* — a set of one or more items that are all of the same form (one of the above forms).

5.2.8 *ordered list* — a list for which the order in which items appear is significant.

5.2.9 *positive integer* — may take the value of any positive whole number. Messaging protocol may impose a limit on the range of possible values.

5.2.10 *structure* — a complex set of information consisting of specific sets of items of possibly mixed data types, in a specified arrangement.

5.2.11 *text* — a text string. The message protocol restricts its length or ASCII representation. Messaging protocol may impose restrictions, such as length or ASCII representation.

5.2.12 *unsigned integer* — may take the value of any positive integer or zero. Messaging protocol may impose a limit on the range of possible values.

6 Convention

6.1 Harel State Model

6.1.1 This document uses the Harel state chart convention for describing dynamic operation of defined objects. The outline of this convention is described in an attachment of SEMI E30. The official definition of this convention is described in “Statecharts: A Visual Formalism for Complex Systems” written by D. Harel in Science of Computer Programming 8, 1987.¹

6.1.2 A transition table is used with the state chart for clearly describing the character of each state transition. The table contains the following: a transition number, current state, trigger, new state, and operation in transition.

6.2 OMT Object Information Model

6.2.1 The object models are represented using the Object Modeling Technique (OMT) developed by Rumbaugh, James, et al, in Object Oriented Modeling Design, Prentice Hall, Englewood Cliffs, NJ, 1991.²

6.2.2 An overview of this notation is provided in the Appendix 1 of SEMI E39.

6.3 Object Attributes Representation

6.3.1 The object information model for standardized objects will be supported by an attribute definition table with the following column headings:

¹ Elsevier Science, P.O. Box 945, New York, NY 10159-0945, <http://www.elsevier.nl/homepage/browse.htm>

² Prentice Hall, Inc., Upper Saddle River, NJ 07458, <http://www.prenhall.com/divisions/ecs/cscat.html>

<i>Attribute Name</i>	<i>Definition</i>	<i>Access</i>	<i>Reqmt</i>	<i>Format</i>
The formal text name of the attributes.	Description of information contained.	RO or RW	Y or N	Refer to the description below.

6.3.1.1 The Access column uses RO (read only) or RW (read/write) to indicate the access that a service user has to the attribute.

6.3.1.2 A “Y” or “N” in the Requirement (Reqmt) column indicates if this attribute must be supported in order to meet the fundamental requirement for the service.

6.3.1.3 The Format column is used for showing the data type of the attribute. (See Section 5.2.)

6.4 Service Message Representation

6.4.1 *Service Resource Definition* — A service resource definition table defines the specific sets of messages for a service group, as shown in the following table:

<i>Message Service Name</i>	<i>Type</i>	<i>Description</i>
Message name	N or R	Purpose of the service

6.4.1.1 Type can be either N = Notification or R = Request. Notification type messages are initiated by a service provider and the provider does not expect to get the response from the service user or requester.

6.4.1.2 Request messages are initiated by a service user or requester. The Request message asks for data or an activity from the provider. Request messages expect a specific response message (no presumption on the message context).

6.4.2 *Service Parameter Dictionary* — A service parameter dictionary table defines the parameters for one or more services, as shown in the following table:

<i>Parameter</i>	<i>Form</i>	<i>Description</i>
Parameter X	Data type	The parameter called X is B in A.

6.4.2.1 A row is provided in the table for each parameter of the service. The first column contains the name of the parameter. This is followed by columns describing form and contents of the corresponding primitive.

6.4.2.2 The Form column is used to indicate the type of data contained in a parameter. (See Section 5.2 for definitions.)

6.4.2.3 The Description column in the Service Parameter Dictionary table describes the meaning of the parameter, the values it can assume, and any interrelationships with other parameters.

6.4.2.4 To prevent the definition of numerous parameters named “XxxList”, this document adopts the convention of referring to the list as “(List of) Xxxx”. In this case, the definition of the variable Xxx will be given, not of the list. The term “list” indicates a collection (or set) of zero or more items of the same data type. Where a list is used in both the request and the response, the list order in the request is retained in the response. A list must contain at least one element unless zero elements are specifically allowed.

6.4.3 *Service Message Definition* — A service message definition table defines the parameter used in a service, as shown in the following table:

<i>Parameter</i>	<i>Req/Ind</i>	<i>Rsp/Conf</i>	<i>Description</i>
Parameter X	see below	see below	A description of the service.

6.4.3.1 The columns labeled Req/Ind and Rsp/Conf link the parameters to the direction of the message. The message sent by the initiator is called the “Request”. The receiver terms this message the “Indication”. The receiver may then send a “Response”, which the original sender terms the “Confirmation”.

6.4.3.2 The following codes appear in the Req/Ind and Rsp/Conf columns, and are used in the definition of the parameters (e.g., how each parameter is used in each direction):

“M” — Mandatory Parameter – must be given a valid value.

“C” — Conditional Parameter – may be defined in some circumstances and undefined in others. Whether a value is given may be completely optional or may depend on the values of other parameters.

“U” — User-Defined Parameter

“-” — The parameter is not used.

“=” — (for response only) Indicates that the value of this parameter in the response shall match the value in the primary (if defined).

7 Overview

7.1 Purpose of Substrate Tracking

7.1.1 Substrate tracking consists of two kinds of capabilities, as described below:

7.1.2 *Substrate Location Tracking* — Substrate Location Tracking provides the capability to determine the current location of substrates in the equipment. The indication of the substrate location will help the user to understand the environment to which the substrate is exposed in the equipment.

7.1.3 *Substrate History Record* — Substrate History Record provides the capability to read the history of substrates in the equipment. The user can inquire about the history of a particular substrate or substrate

location. A history includes the series of locations that the substrate has visited in the equipment.

7.1.4 *Substrate Process Tracking* — Substrate Process Tracking provides the capability of tracking current substrate processing state. For example, the user may determine whether a substrate is processed or not by requesting its state when the process has been completed abnormally. The user may use the state transitions to trigger wafer level data collection.

7.2 View of Substrate Tracking

7.2.1 Figure 1 shows the view of Substrate Tracking using the OMT representation. It is assumed that substrates are available before usage of the services. Substrate carriers may or may not be used to load the substrates to the equipment. Substrates are loaded into the equipment and travel through substrate locations. The equipment must maintain all information necessary to track substrates in the equipment.

7.3 Substrate Location

7.3.1 A substrate location is a discrete position that can hold a substrate in the equipment. This document will not define where the substrate location exists. Substrate locations shall be determined by the design of equipment and the applications. For example, to track the order of process chambers that is applied to a substrate process in a multi-chamber equipment, it is appropriate to designate process chambers for substrate locations. All substrate locations shall be documented by the equipment supplier.

7.3.2 Figure 2 shows the concept of the substrate location using OMT representation. This figure is provided to clarify the definition of locations.

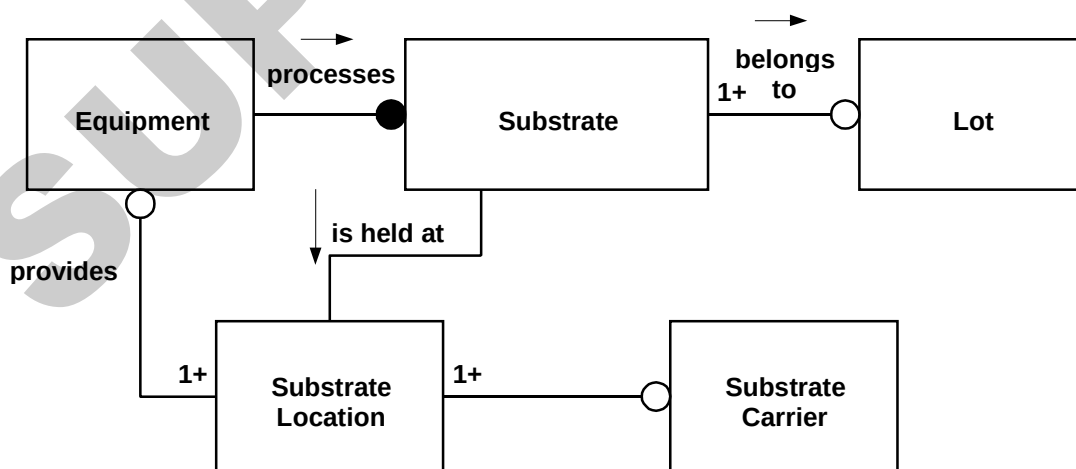


Figure 1
Overview of Substrate Tracking

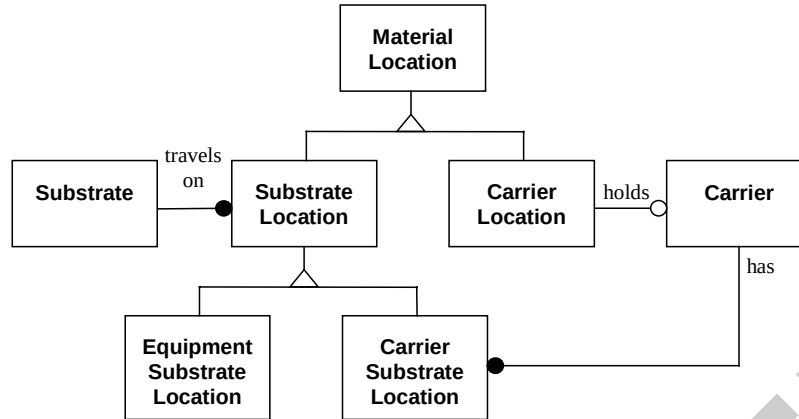


Figure 2
Concept of Substrate Location

7.3.3 Material Location is classified into Substrate Location and Carrier Location. A Carrier Location may hold a carrier and a Substrate Location may hold a substrate.

7.3.4 Substrate Location is classified into Equipment Substrate Location and Carrier Substrate Location. Carrier has Carrier Substrate Locations. A Carrier Substrate Location is the location which can hold a substrate in a carrier. An Equipment Substrate Location is the location which can hold a substrate on the equipment resource.

7.3.5 Substrate travels on substrate locations.

7.4 Identification of Substrate

7.4.1 Substrate shall be identified by substrate ID. There are options for assigning substrate ID to a substrate.

- When a carrier is used for loading substrates and the carrier is given its carrier ID, substrate IDs can be obtained by utilizing the carrier ID and carrier substrate location. This standard will define the requirement for the substrate ID assignment to the substrates in a carrier.
- Substrate ID may be read from the substrate directly by a substrate ID reader. The substrate ID read from the substrate can be used for the identification.
- Substrate ID may be assigned by the user (i.e., host or operator). The substrate ID can be used for the identification. This standard will define the service for registering substrates.

7.4.2 The recommendation of a particular option for assigning substrate IDs is not within the scope of this standard. It must be selected based on the equipment design and applications.

8 Substrate Tracking Requirement

8.1 Compliance to Substrate Tracking Service (STS) requires that an implementation also provide certain capabilities that are defined in other standards.

- STS requires compliance to SEMI E39 (OSS) for Substrate objects. The attributes of these objects may be accessed through use of the GetAttr and SetAttr services.
- STS requires an event reporting capability. This capability may be satisfied by the Event Reporting capability in SEMI E30 (GEM) or by SEMI E53 (ERS).

8.2 STS compliant equipment must implement capabilities defined in the following sections of this document:

- Section 9, Substrate Object Definition,
- Section 10, Substrate Location Object Definition,
- Section 11, Substrate Tracking Services, and
- Section 12, Variable Data, if SEMI E53 is not implemented.

8.3 Suppliers shall document the method by which the equipment identifies individual substrates. If the equipment has implemented a substrate ID reading capability, then the equipment shall provide the capability to assign the ID read from the substrate to the substrate object. Otherwise, the equipment shall provide the capability to assign substrate ID to the substrate object by the information provided by the host with the source carrier, either in ContentMap, where provided (see SEMI E87), or the default substrate ID. The default substrate ID shall be represented in text as the carrier ID, a period ".", and a two-digit slot number combined, as in "carrier ID" + "." + "slot number". For example,