SEMI E105-0701
PROVISIONAL SPECIFICATION FOR CIM FRAMEWORK SCHEDULING COMPONENT

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 Japanese Regional Standards Committee on February 1, 2001. Initially available at www.semi.org April 2001; to be published July 2001. Originally published October 2000.

INACTIVE

1 Purpose

1.1 The Scheduling Component supports Factory Operations, Material Transport and Storage, Production Machine, and Equipment Tracking and Maintenance components by ordering, in time, jobs that process material on equipment, move material, and maintain equipment. The scheduler uses knowledge of product demand, equipment and material state, process flows, throughput bottlenecks, operational policy and constraints, and other information to recommend jobs that maximize effective utilization of factory resources to satisfy product demand and planned objectives.

1.2 Increased control over operations requires an active Scheduling Component that can respond to factory events and changes in state and dynamically adjust the schedules for material processing, material transport and equipment maintenance. The Scheduling Component can react to inventory levels of material in the factory to adjust priorities to minimize queue sizes and ensure that use of bottleneck equipment is optimized to keep WIP inventory levels at desired levels.

1.3 The Scheduling Component can minimize turn around time (TAT) by coordinating material transport (for substrates and durables) with processing to reduce equipment idle time. The Scheduling Component can sequence activities to minimize setup time. It can also respond to scheduled and unscheduled equipment down-time to minimize impact on turn around time. In addition to minimizing overall TAT, it can react to the priorities for urgent lots to move them through the process flow in the minimum possible time while adjusting the schedules of lower-priority lots that are impacted.

2 Scope

2.1 The primary run-time responsibilities of the Scheduling Component are to monitor resource and material state and apply scheduling and dispatching decision mechanisms to identify the next activity (dispatching) or sequence of activities (scheduling) for factory resources. The Scheduling Component includes an interface that supports both scheduling and dispatching. Figure 1 illustrates the interactions between the Scheduling Component and other components of the CIM Framework. This illustration does not reflect all of the many inputs to the Scheduling Component that are required to provide it with the current status of the factory resources.

2.2 As described in this standard, the Scheduling Component produces activity option and activity forecast lists. These lists are produced by combining the factory model and status information from other components with scheduling policies in the Scheduling Component. For example, the Scheduling Component combines data from the Specification Component on how products are made with status data from the Equipment and Product Management Components to give activity options for a machine.

2.3 The Factory Operations Component uses the Scheduling Component services to orchestrate the management of machines and production of lots. For example, when a tool becomes available, Factory Operations uses the Activity Options list to select the next lot to process on the tool. Factory Operations then works with other components to execute the production job for the lot on the machine. Executing the production job changes the state of lot and tool in other components. These status changes are then used by the Scheduling Component when it produces new activity lists.

2.4 The Scheduling Manager supports Factory Operations by providing an answer to questions like, “What is next for this material or resource?” The answer may be based on evaluation of current or future constraints and objectives. Although the dispatcher's output takes the form of a decision for the next activity for the target resource, the interface may also support manual scenarios by providing a list of prioritized activities from which the decision is selected.

2.5 The Scheduling Manager interface also provides forecasts of future activities projected to occur after the next activity. By simulating anticipated future activity sequencing and execution timing the scheduler can generate forecasts that predict future responses from the dispatcher for subsequent requests. These forecasts are subject to change as factory conditions change, but they offer the best current projection of future activity decisions.

Documents Document Sharing Hub
Full document details: SEMI E105-0701

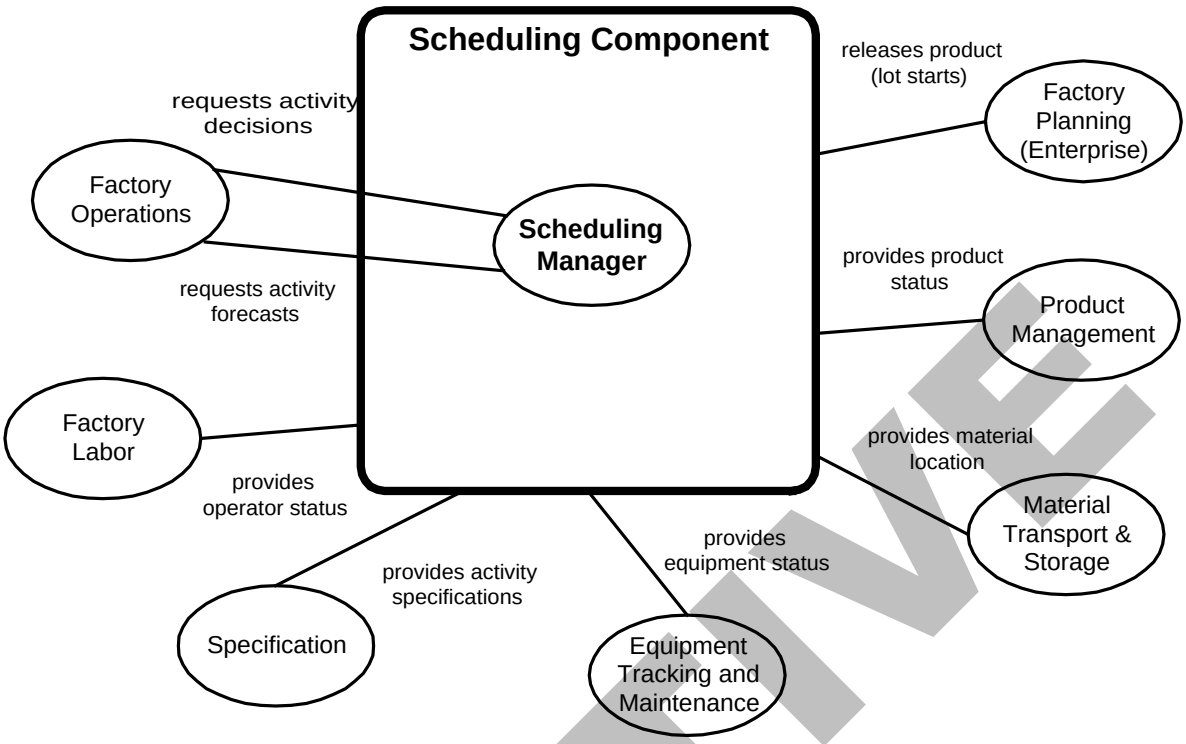

Figure 1
Scheduling Component Interactions

2.6 This specification does not describe the definition of the policies used by the Scheduling Component to produce activity lists. This is left to component implementors.

2.7 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 *Provisional Status* — This specification is designated as provisional due to known areas that need to be completed. The following items summarize the deficiencies of the provisional specification to be addressed before a subsequent ballot to upgrade it to full standard status.

3.1.1 *Dependence on Factory Operations Component* — The Scheduling Component provides specifications of activities that are executed by the Factory Operations Component in the form of production jobs, transport jobs and maintenance jobs. The specification of Factory Operations may require changes to the activity specification properties of the Scheduling Component. When the Specification for CIM Framework Factory Operations Component is complete, this dependency can be resolved.

3.1.2 *Specification of Event Subscriptions* — The Scheduling Component maintains a current status of the factory resources and material by receiving event notifications of occurrences that indicate changes in other components. Events are specified with the component responsible for publishing the event notifications. Until all of the required event formats are defined as part of the publishing component specification, the inputs to the Scheduling Component will be defined only in general terms. When event specifications are complete, these event subscriptions can be specified.

3.1.3 *Initialization* — The Scheduling Component does not currently provide interfaces that allow the representation of factory conditions to be initialized for the start-up of dispatching and scheduling. The addition of initialization interfaces will be added or deemed out of scope prior to upgrade to full standard status.

4 Referenced Standards

4.1 *SEMI Standards*

SEMI E81 — Provisional Specification for CIM Framework Domain Architecture

INACTIVE

Documents
Document Sharing Hub
Full document details: SEMI E105-0701

SEMI E97 — Provisional Specification for CIM Framework Global Declarations and Abstract Interfaces

SEMI E102 — Provisional Specification for CIM Framework Material Transport and Storage Component

4.2 *Other Documents*

UML Notation Guide, Version 1.1, document number ad/97-08-05, Object Management Group.[1]

ISO/IEC International Standard 14750 (also ITU-T Recommendation X.920): Information Technology – Open Distributed Processing – Interface Definition Language [2]

5 Terminology

5.1 *Definitions*

5.1.1 *activity* — work performed as part of the manufacturing operations of a factory. Activities may be specified formally by a predefined type of job specification (for example, Production Job, Transport Job or PM Job), or they may be represented by identifying the minimal set of resources and material needed to allow subsequent completion of the job specification. An activity is the result of dispatching or scheduling.

5.1.2 *dispatching* — generation of a decision or option for the next activity involving a particular factory resource or material. The dispatch result is determined by evaluating the current state of the factory, the priorities and requirements for the activities, and the relationship of the activities to one another. Dispatching returns only the immediately applicable part of a schedule.

5.1.3 *factory planning* — recommendation of lot starts for a particular production facility over an extended period of time. The factory plan is determined by predicting future changes in factory state and available capacity as lots progress through production. This prediction is used to determine the optimum sequence of lot starts to best achieve the production goals of the facility. Factory planning is typically the responsibility of enterprise systems.

5.1.4 *scheduling* — generation of a forecast of future time sequenced activities involving factory resources or material. The schedule is based on the current state of the factory, the priorities and requirements for the activities, the relationship of the activities to one another and knowledge of factory level goals and capacity. Scheduling covers activities projected to occur over a longer future time interval than dispatching.

6 Requirements

6.1 *Scheduling Component* — The Scheduling Component provides a single interface called the SchedulingManager with operations for both dispatching and scheduling.

6.1.1 Any resource or material within the factory may have a next activity defined for it. The SchedulingManager interface is responsible for identifying activities that answer questions such as:

- What's next for this Machine?
- Where next for this Lot?
- What's next for this Operator?

6.1.2 The SchedulingManager interface is also responsible for generating a time-sequenced forecast of future activities for a resource or for material within the factory. This forecast is based on projections from the current factory state and may be updated as actual activities are dispatched and executed.

6.1.3 The Scheduling Component Information Model is shown in Figure 2.

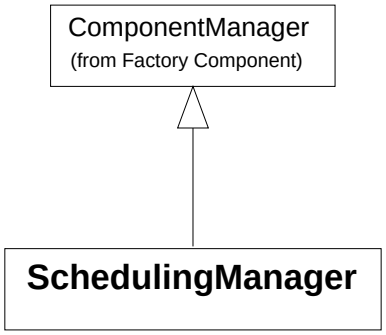

Figure 2
Dispatching Component Information Model

1 UML Notation Guide v1.1 is available to the general public at http://www.omg.org/cgi-bin/doclist.pl, +1-508-820 4300, Object Management Group, Inc. , Framingham Corporate Center, 492 Old Connecticut Path, Framingham, MA 01701.

2 ISO Central Secretariat, 1, rue de Varembé, Case postale 56, CH-1211 Genève 20, Switzerland

INACTIVE

Documents Document Sharing Hub
Full document details: SEMI E105-0701

6.1.4 *Scheduling Component Declarations*

/* The ActivityType enumerated value will determine which kind of activity is represented and the appropriate properties present in the activitySpec. */

```
enum ActivityType {
      ProductionActivity,
      TransportActivity,
      MaintenanceActivity,
      DoNothing };
```

/* ActivityOption represents a single activity that is eligible to be dispatched for execution. The prioritySequenceNumber represents the priority ordering of a list of ActivityOptions in the case where a manual process includes selection of one ActivityOption from the list. The deadline is a timestamp that is used to convey start time constraints for activities. It may also be used in a DoNothing activity to specify the timeout for a retry of the dispatch request. */

```
struct ActivityOption {
      ActivityType activityType;
      short prioritySequenceNumber;
      Global::TimeStamp deadline;
      AbstractIF::ResourceSequence resources;
      AbstractIF::MaterialGroupSequence materialGroups;
      DurablesManagement::DurableSequence durables;
      Global::Properties activitySpec;
}; // ActivityOption
```

/* ActivityForecast represents a single Activity that may be dispatched and executed in the future. An ActivityForecast includes information that is equivalent to an ActivityOption, but with added start and finish time projections. The timeSequenceNumber represents the time ordering of activities consistent with the projectedStart and projectedFinish. */

```
struct ActivityForecast {
      ActivityType activityType;
      short timeSequenceNumber;
      AbstractIF::ResourceSequence resources;
      AbstractIF::MaterialGroupSequence materialGroups;
      DurablesManagement::DurableSequence durables;
      Global::TimeStamp projectedStart;
      Global::TimeStamp projectedFinish;
      Global::Properties activitySpec;
}; // ActivityForecast
```

/* An ActivityList contains one or more ActivityOptions for a Resource, MaterialGroup or Durable which are all eligible for immediate execution. A list of one ActivityOption represents a decision from the SchedulingComponent. A list of more than one ActivityOption represents recommendations from the SchedulingComponent which require some external interaction for a decision to be reached. */

```
typedef sequence <ActivityOption> ActivityList;
```

/* A ForecastList contains one or more ActivityForecasts for a Resource or MaterialGroup which are not yet dispatched for execution. */

```
typedef sequence <ActivityForecast> ForecastList;
```

6.1.5 *Use of ActivitySpec Properties*— The ActivitySpec part of the ActivityOption and ActivityForecast are provided to allow implementations to extend the specifications of activities for specific implementations. The names and value types of properties included in these ActivitySpec structure are not part of the standard.

INACTIVE

Documents
Document Sharing Hub
Full document details: SEMI E105-0701

6.1.6 *SchedulingManager Interface*

Module: Scheduling

Interface: SchedulingManager

Inherited Interface: ComponentManager

```
interface SchedulingManager : FactoryOperations::ComponentManager {
```

Description: The SchedulingManager provides the concrete interfaces that support the registration and control of the Scheduling Component and enable access to the Dispatcher and Scheduler interfaces.

Exceptions:

/* An attempt was made to request an activity for a resource that is not found within the factory data available to this SchedulingManager. */

```
exception ResourceNotFound {
      AbstractIF::Resource requestedObject;};
```

/* An attempt was made to request an activity for a material group that is not found within the factory data available to this SchedulingManager. */

```
exception MaterialGroupNotFound {
      AbstractIF::MaterialGroup requestedObject;};
```

/* An attempt was made to request an activity for a durable that is not found within the factory data available to this SchedulingManager. */

```
exception DurableNotFound {
      DurablesManagement::Durable requestedObject;};
```

Published Events:.

/* The ActivityListAvailable event is published by the Scheduling Component to notify event subscribers that a new ActivityList containing changes is available. The generation of ActivityListAvailable events is controlled as an implementation specific configuration choice. */

`ActivityListAvailable`

/* The ForecastListAvailable event is published by the Scheduling Component to notify event subscribers that a new Forecast is available. ForecastList generation is potentially a compute-intensive operation which may benefit from event notification when an updated forecast is available. The generation of ForecastListAvailable events is controlled as an implementation specific configuration choice. */

`ForecastListAvailable`

Provided Services:

/* Return an activity list for this resource. The activity list will contain the number of existing activities requested in listCount in maximum. If “listCount” is blank, the activity list shall contain all existing activities. A decision will be represented by an activity list with only one activity option. The ActivityOptions returned by this operation may be ProductionActivity, MaintenanceActivity or DoNothing ActivityTypes. Although Resources are typically immobile (such a Equipment), some Resources may also be subject to TransportActivities. */

```
ActivityList whatNextForResourceByCount (
       in AbstractIF::Resource aResource)
       in short listCount)
       raises (Global::FrameworkErrorSignal,
                    ResourceNotFound);
```

INACTIVE

Documents Document Sharing Hub
Full document details: SEMI E105-0701

/* Return an activity list for this material group. The activity list will contain the number of existing activities requested in listCount in maximum. If "listCount" is blank, the activity list shall contain all existing activities. A decision will be represented by an activity list with only one activity option. The ActivityOptions returned by this operation may be ProductionActivity, TransportActivity or DoNothing ActivityTypes. */

```
ActivityList whatNextForMaterialGroupByCount (
      in AbstractIF::MaterialGroup aMaterialGroup,
      in short listCount)
      raises (Global::FrameworkErrorSignal,
            MaterialGroupNotFound);
```

/* Return an activity list for this durable. The activity list will contain the number of existing activities requested in listCount in maximum. If "listCount" is blank, the activity list shall contain all existing activities. A decision will be represented by an activity list with only one activity option. The ActivityOptions returned by this operation may be ProductionActivity, MaintenanceActivity, TransportActivity or DoNothing ActivityTypes. */

```
ActivityList whatNextForDurableByCount (
       in DurablesManagement::Durable aDurable)
       in short listCount)
       raises (Global::FrameworkErrorSignal,
              DurableNotFound);
```

/* Return an activity list containing the next TransportActivity for this material group. A decision will be represented by an activity list with only one activity option. The ActivityOptions returned by this operation may be TransportActivity or DoNothing ActivityTypes. This operation is defined to support a factory operating with a "push" model that pushes material to the next storage or production machine. */

```
ActivityList whereNextForMaterialGroup (in AbstractIF::MaterialGroup aMaterialGroup)
      raises (Global::FrameworkErrorSignal,
            MaterialGroupNotFound);
```

/* Return an activity list containing the next TransportActivity for this durable. A decision will be represented by an activity list with only one activity option. The ActivityOptions returned by this operation may be TransportActivity or DoNothing ActivityTypes. This operation is defined to support a factory operating with a "push" model that pushes material to the next storage or production machine. */

```
ActivityList whereNextForDurable (in DurablesManagement::Durable aDurable)
      raises (Global::FrameworkErrorSignal,
            DurableNotFound);
```

/* Return a forecast list containing future activities for this resource. The forecast list will contain all activities projected to start before the lookAheadTime, but which have not yet been started. The forecast list may include a mix of different activity types. */

```
ForecastList forecastForResourceByTime (
      in AbstractIF::Resource aResource,
      in Global::TimeStamp lookAheadTime)
      raises (Global::FrameworkErrorSignal,
            ResourceNotFound);
```

/* Return a forecast list containing future activities for this material group. The forecast list will contain all activities projected to start before the lookAheadTime, but which have not yet been started. The forecast list may include a mix of different activity types. */

```
ForecastList forecastForMaterialGroupByTime (
      in AbstractIF::MaterialGroup aMaterialGroup,
      in Global::TimeStamp lookAheadTime)
      raises (Global::FrameworkErrorSignal,
            MaterialGroupNotFound);
```

INACTIVE

Documents
Document Sharing Hub
Full document details: SEMI E105-0701

/* Return a forecast list containing future activities for this durable. The forecast list will contain all activities projected to start before the lookAheadTime, but which have not yet been started. The forecast list may include a mix of different activity types. */

```
ForecastList forecastForDurableByTime (
      in DurablesManagement::Durable aDurable,
      in Global::TimeStamp lookAheadTime)
      raises (Global::FrameworkErrorSignal,
            DurableNotFound);
```

/* Return a forecast list containing future activities for this resource. The forecast list will contain the number of upcoming activities requested in lookAheadCount. The forecast list may include a mix of different activity types. */

```
ForecastList forecastForResourceByCount (
      in AbstractIF::Resource aResource,
      in short lookAheadCount)
      raises (Global::FrameworkErrorSignal,
            ResourceNotFound);
```

/* Return a forecast list containing future activities for this material group. The forecast list will contain the number of upcoming activities requested in lookAheadCount. The forecast list may include a mix of different activity types. */

```
ForecastList forecastForMaterialGroupByCount (
      in AbstractIF::MaterialGroup aMaterialGroup,
      in short lookAheadCount)
      raises (Global::FrameworkErrorSignal,
            MaterialGroupNotFound);
```

/* Return a forecast list containing future activities for this durable. The forecast list will contain the number of upcoming activities requested in lookAheadCount. The forecast list may include a mix of different activity types. */

```
ForecastList forecastForDurableByCount (
      in DurablesManagement::Durable aDurable,
      in short lookAheadCount)
      raises (Global::FrameworkErrorSignal,
            DurableNotFound);
```

Contracted Services: None.

Dynamic Model: Inherited from ComponentManager.

```
}; // SchedulingManager
```

INACTIVE

Documents Document Sharing Hub
Full document details: SEMI E105-0701

APPENDIX 1
COMPLETE LISTING OF SCHEDULING COMPONENT IDL

NOTE: The material in this appendix is an official part of SEMI E105 and was approved by full letter ballot procedures on August 28, 2000 by North American Regional Standards Committee.

```
module CIMFW {

#include <Global.idl>
#include <AbstractIF.idl>
#include <FactoryOperations.idl>
#include <DurablesManagement.idl>
#ifndef _CIMFW_SCHEDULING_
#define _CIMFW_SCHEDULING_

module Scheduling {

      enum ActivityType {
            ProductionActivity,
            TransportActivity,
            MaintenanceActivity,
            DoNothing };

      struct ActivityOption {
            ActivityType activityType;
            short prioritySequenceNumber;
            Global::TimeStamp deadline;
            AbstractIF::ResourceSequence resources;
            AbstractIF::MaterialGroupSequence materialGroups;
            DurablesManagement::DurableSequence durables;
            Global::Properties activitySpec;
      }; // ActivityOption

      struct ActivityForecast {
            ActivityType activityType;
            short timeSequenceNumber;
            AbstractIF::ResourceSequence resources;
            AbstractIF::MaterialGroupSequence materialGroups;
            DurablesManagement::DurableSequence durables;
            Global::TimeStamp projectedStart;
            Global::TimeStamp projectedFinish;
            Global::Properties activitySpec;
      }; // ActivityForecast

      typedef sequence <ActivityOption> ActivityList;

      typedef sequence <ActivityForecast> ForecastList;

interface SchedulingManager : FactoryOperations::ComponentManager {

      exception ResourceNotFound {
            AbstractIF::Resource requestedObject;};

      exception MaterialGroupNotFound {
            AbstractIF::MaterialGroup requestedObject;};

      exception DurableNotFound {
            DurablesManagement::Durable requestedObject;};
```

INACTIVE

Documents
Document Sharing Hub
Full document details: SEMI E105-0701