



---

## Glossary

# **Solstice Enterprise Manager™ 4.1**

---

Sun Microsystems, Inc.  
901 San Antonio Road  
Palo Alto, CA 94303  
U.S.A. 650-960-1300

Part No. 806-7974-10  
October 2001, Revision A

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Sun Enterprise Manager, SunOS, Java, Java Coffee Cup logo, JavaBeans, Java Dynamic Management, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

**RESTRICTED RIGHTS:** Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

---

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, Sun Enterprise Manager, SunOS, Java, Java Coffee Cup logo, JavaBeans, Java Dynamic Management, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPENDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



Please  
Recycle



Adobe PostScript

# Glossary

---

This glossary offers brief descriptions of terms that appear in the discussion of the Solstice Enterprise Manager (Solstice EM) system, either because they are used in the Telecommunications Management Network (TMN) or because they have specific meanings in the Solstice EM context.

This glossary contains the following types of terms:

- Those specific to OSI management and OMNIPoint 1
- Those specific to the SNMP environment
- Those specific to the CMIP environment
- Those with definitions common to SNMP and CMIP environments
- Those with definitions different for SNMP and CMIP environments
- Those specific to SEM-HA
- Those specific to SEM CORBA

Definitions specific to the SNMP environment and SNMP terms that have a different meaning in CMIP are indicated by [S]. Definitions specific to the CMIP environment and CMIP terms that have a different meaning in SNMP are indicated by [C].

Definitions common to both environments have no indicator.

Definitions specific to the SEM-HA are indicated by [HA].

Definitions specific to SEM CORBA are indicated by [CR].

Terms in *italics* are defined in this glossary.

<b>ACSE</b>	Association Control Service Element. ACSE is a generic grouping of service elements within the application layer that offers services related to the establishment and management of a cooperative relationship between peer application processes.
<b>action</b>	<p>In the Design Advanced Requests service, an action is the response that a request manager makes when a <i>transition</i> occurs. In a <i>request template</i>, the definition of a transition, lists the actions (if any) to take when a transition occurs. The actions are selected from a small set of supported actions. However, each action takes an argument that gives it great latitude in specifying details of the action, particularly when the action is a <i>condition</i> action.</p> <p>What can happen when a transition from one <i>state</i> to another occurs as the result of a condition returning a value of true. The <i>Nerve Center</i> supports three types of actions: execution of a condition, execution of a UNIX command, or sending a mail message. The default is that no action be taken upon a transition. You can have any number of actions in any combination of types.</p>
<b>AE-title</b>	A character string that identifies a domain, it could either be a <i>managed object</i> domain or a manager domain. In the context of <i>EGW</i> , AE-title and <i>EventPort</i> have a one-to-one relationship and hence the AE-title is used to identify the EventPort and hence the receiver of <i>events</i> .
<b>agent</b>	<p>A module residing in a <i>managed resource</i> on a network, capable of reporting the status of the resource and/or responding to inquiries about it. Described in standards documents X.701   ISO/IEC 10040.</p> <p>In a general sense, it is software running on a <i>managed object</i> that responds to and reports to the Solstice EM <i>MIS</i> with current information about the object. Also called Network Management Agent. See also <i>proxy agent</i>.</p>
<b>agentCME object</b>	See <i>system object</i> .
<b>alarm</b>	An <i>event</i> or <i>trap</i> that has been registered for (by a <i>request</i> ) in the <i>MIS</i> .
<b>API</b>	Application Programming Interface. A set of software routines that enable an applications developer to access and use the features of a product.
<b>ARP</b>	Address Resolution Protocol. A procedure for finding the network hardware address corresponding to an <i>Internet</i> address (RFC 826).
<b>ASN.1</b>	Abstract Syntax Notation One. A specification understood by network management protocols and used for encoding information between a manager and <i>agents</i> in a machine- and network-independent manner.
<b>ASN.1 compiler</b>	A Solstice EM program ( <i>em_asn1</i> ) that accepts <i>ASN.1</i> descriptions for ultimate inclusion in the <i>MDR</i> of a running Solstice EM <i>MIS</i> .

<b>attachment instance</b>	<p>An <i>object class</i> instance of a line attachment to a node. See also <i>class instance</i>.</p> <p>You can attach each end of a <i>Network Views</i> tool line icon to a node. The attachment has two aspects: it is a graphical representation of one portion of the physical network; it is an object class on the node (such as an <i>object</i> for port or connection information).</p>
<b>attribute</b>	<p>An item of management information that describes some property of a <i>managed object</i>—its operational characteristics, <i>conditions</i> of operation, or current <i>state</i> or status. In the <i>SNMP</i> world, it is also called a Non-Aggregate <i>object</i>. In the context of Solstice EM, it is an object type in an <i>MIT</i> or <i>MIB</i> module that is part of an object class (a child of the object class). An attribute has an identification and a value. Instances of managed objects belonging to the same class have the same set of attributes with the same set of identifications; only the values of the attributes can differ from one <i>instance</i> to another. See also <i>object class</i> and <i>class instance</i>.</p> <p>A characteristic of a managed object. The <i>Nerve Center</i> may <i>poll</i> for attributes as part of a <i>request</i>. In a request, you can poll only for those attributes that are defined in the <i>MIS</i>. Attributes are defined in <i>GDMO</i> documents, which are compiled into the <i>MIS</i>. Solstice EM is shipped with a set of <i>GDMO</i> documents in which managed objects, and their attributes, are defined.</p>
<b>authentication entity</b>	That portion of an <i>SNMP agent</i> responsible for verifying that an <i>SNMP</i> entity is a member of the community it claims to be in. The authentication entity is also responsible for encoding and decoding <i>SNMP</i> messages according to the authentication algorithm of a given community. This entity is used in security alarms.
<b>AVA</b>	Attribute Value Assertion. An assertion that a particular <i>attribute</i> has a particular value. See also <i>RDN</i> .
<b>AV pair</b>	Attribute-Value pair.
<b>base object</b>	See <i>object class</i> .
<b>base-object instance</b>	See <i>class instance</i> .
<b>BSS</b>	Backing Storage System. A set of methods included in an <i>API</i> to enable <i>managed objects</i> to be written to and read from a database.
<b>callback</b>	The <i>object</i> (or method) to be notified at completion of an asynchronous process. Typically, when a client process requests something from an <i>MIS</i> , the <i>request</i> includes a pointer to the function to be invoked when the <i>MIS</i> is ready.
<b>child</b>	A subordinate object contained in an <i>instance</i> of a <i>class</i> and directly below that <i>class instance</i> . [C]
<b>CI</b>	Component interface. The <i>DMI</i> layer used by <i>component instrumentation</i> .

<b>class</b>	In general, the formal description of a set of <i>objects</i> . In the <i>OSI</i> world, objects with similar <i>attributes</i> and behavior are grouped into classes. In C++, the rules governing a set of data structures (which are said to be an <i>instance</i> of the class) and the methods (also called member functions) which give access to an instance's data.
<b>class instance</b>	A collection of <i>attribute instance</i> values that specify one example of a <i>class</i> . For example, if the class comprised port information for a router port, you could specify an instance of the class by providing a router board and port number for a particular port. The information you provide to specify a class instance is called the <i>instance identifier</i> . Other related terms are instance string, <i>RDN</i> , Index, and Named Object.
<b>class string</b>	A text string that identifies a <i>group</i> outside the context of a particular <i>component</i> declaration. Identical group definitions will have identical class strings.
<b>CLNS</b>	Connection-Less Network Service. A network service provided at Layer 3 of the <i>OSI</i> protocol stack.
<b>CM-API</b>	See <i>X resource</i> .
<b>CME</b>	Conformant Management Entity. Any management system that supports the interoperable interface defined by the <i>NMF</i> . (Solstice EM <i>MIS</i> is one.)
<b>cmeID</b>	The object identifier of a <i>CME</i> . A unique identifier assigned to each <i>MIS</i> running Solstice EM.
<b>CMIP</b>	Common Management Information Protocol. The protocol specified as part of <i>CMIS</i> ; part of the <i>OSI</i> protocol stack. CMIP is a connection-oriented protocol, with reliable delivery. It is the same as Recommendation X.711, as defined by the International Telecommunications Union (ITU). CMIP was developed to support distributed management appropriate to complex wide-area networks as well as local-area networks. It is object-oriented, and often used with a programming <i>agent</i> between a <i>device</i> and a management system.
<b>CMIP MPA</b>	CMIP Management Protocol Adapter. A Solstice EM <i>component</i> that allows access to <i>OSI managed resources</i> via <i>CMIP</i> . The CMIP MPA receives management directives from the <i>MIS</i> and translates the directives into CMIP messages. The CMIP MPA also receives <i>notifications</i> from <i>OSI managed objects</i> and sends them to the <i>MIS</i> .
<b>CMIS</b>	Common Management Information Services. An <i>ISO</i> standard that defines the services required of systems that exchange network management information, such as <i>requests</i> to supply the values of <i>attributes</i> or to send reports of <i>events</i> . Management services as defined by the ITU Recommendation X.710.
<b>CMOL</b>	Common Management Information Protocol Over LLC (Logical Link Control). A lightweight version of <i>CMIP</i> , standardized by the IEEE.
<b>CMOT</b>	Common Management Information Protocol Over TCP/IP. <i>CMIP</i> for the <i>TCP/IP</i> environment.

<b>command block</b>	The concatenation of data blocks (data structures) that constitute a command to be sent between a <i>management application</i> and the service provider and between the service provider and the <i>component instrumentation</i> .
<b>community string</b>	An <i>SNMP</i> concept. A simple password that you provide when adding a node icon to the network view. The <i>agent</i> running on the node requires this password from Solstice EM before providing information about the node. [S]
<b>compiler</b>	Generally, a process that analyzes a statement into its syntactic components. Specifically, Solstice EM provides compilers that process the descriptions of <i>managed object classes</i> written according to the <i>GDMO</i> specifications ( <i>em_gdmo</i> ), according to <i>ASN.1</i> specifications ( <i>em_asn1</i> ), and descriptions of managed object classes written in <i>MIB</i> format ( <i>em_cmib2gdmo</i> ). Output of these compilers supply the internal formats needed for the <i>MDR</i> and the <i>ASN.1</i> repository.
<b>component</b>	Any hardware, software, or firmware element contained in (or primarily attached to) a computer system.
<b>component instrumentation</b>	The executable code that provides <i>DMI</i> management functionality for a particular <i>component</i> .
<b>Concise MIB</b>	In the <i>SNMP</i> world, an extensible system for classifying information about <i>managed objects</i> in a network. Specifically, a particular extension offered by the manufacturer of a <i>device</i> in order to accommodate information about that device within the general <i>MIB</i> structure. Refer RFCs 1156 and 1213, which define, respectively, <i>MIB I</i> and <i>MIB II</i> , which are the <i>MIBs</i> for <i>TCP/IP</i> -based internetworks. See also <i>MIB</i> .
<b>Concise MIB compiler</b>	A Solstice EM program ( <i>em_cmib2gdmo</i> ) that accepts the description of an <i>Internet MIB</i> and generates parallel descriptions in <i>GDMO</i> (which may then be submitted to the <i>GDMO compiler</i> ( <i>em_gdmo</i> )) and <i>ASN.1</i> (which may then be submitted to the <i>ASN.1 compiler</i> ( <i>em_asn1</i> )) for ultimate inclusion in the <i>MIT</i> and <i>MDR</i> of a running Solstice EM <i>MIS</i> .
<b>condition</b>	<p>A term used in Solstice EM requests. A set of instructions written in the <i>RCL</i>.</p> <p>A condition serves two functions in a <i>request</i>:</p> <ul style="list-style-type: none"> <li>- Its primary function tests whether a request for a specific set (one or more) of <i>managed objects</i> will cause a <i>transition</i> from one <i>state</i> to another (or loop back to the same state). You must have at least one <i>condition</i> associated with each transition. Where there are multiple conditions, the <i>request facilities</i> evaluate conditions in the order that they are entered in the <i>request template</i>.</li> <li>- It is also used as an action taken in response to a transition. A condition is one of the three alternatives you have for <i>actions</i>.</li> </ul>

When used to test whether to make a transition, the value of the condition (true is nonzero, false is zero) is the value returned by the last statement in the condition. When used as an action, the value returned by the condition is not used by the request facilities.

<b>confirm</b>	The final response from a <i>request</i> .
<b>confirm buffer</b>	The area of memory where a <i>component instrumentation</i> or service provider puts response data.
<b>container icon</b>	In <i>Network Views</i> , an icon that can contain other icons, or more precisely, an icon representing a <i>managed object</i> that can contain other managed objects. For example icons such as networks and subnetworks. A container icon is equivalent to a <i>view</i> and is distinguished from an <i>element</i> or a link icon.
<b>CORBA</b>	Common Object Request Broker Architecture. Structure for interchange of <i>objects</i> in a network management environment. Developed by <i>OMG</i> .
<b>COSE</b>	Common Open Software Environment. Common network system management environment proposed by IBM, HP, Sun, and UNIX Systems Labs. Rival to <i>DME</i> from <i>OSF</i> .
<b>CosEventChannel</b>	
<b>Admin</b>	Administrative interface that allows consumers and suppliers to establish logical connections with it. It handles <i>EventChannel</i> administration. [CR]
<b>Criteria and Key</b>	The combination of Criteria and Key uniquely identify the software running on a node that responds to and reports current information about another node to the Solstice EM <i>MIS</i> , which has no agent running on it. It is often used to shield network devices from <i>CMIP</i> overhead. Both are character strings. See also <i>agent</i> . [CR]
<b>data service</b>	Highly available software which is able to run within a Sun Cluster. [HA]
<b>DBM</b>	Database Manager. A simple database manager that is useful where a more powerful relational database package is not needed.
<b>device</b>	A <i>network element</i> . In the Solstice EM documentation, it refers to bridges, routers, and other network nodes that are not <i>hosts</i> (cannot accept logins).
<b>Direct interface</b>	Method by which a <i>component instrumentation</i> informs the service provider that it is already running. Rather than starting the code to service incoming <i>requests</i> , the service provider will use the code already running.
<b>discriminator construct</b>	The proposition that a log object tests in order to decide whether to keep a record of a <i>notification</i> . Each log object has exactly one discriminator construct. The discriminator construct is an expression that can be evaluated for the <i>attributes</i> contained in the notification message. A discriminator construct is written in the language prescribed for a <i>CMIS</i> filter, as specified in ISO/IEC 9595. A discriminator construct is also used by the <i>EFD</i> to determine whether a received notification needs to be forwarded to an application.



<b>distinguished name</b>	A name that identifies a position within the <i>MIT</i> . A name may be an <i>FDN</i> and include the complete path from the root of the tree, an <i>RDN</i> with respect to a current position in the tree, or an <i>LDN</i> .
<b>DME</b>	Distributed Management Environment. Network management environment of the <i>OSF</i> .
<b>DMI</b>	Definition of Managed Information.
<b>EDF</b>	Event Discriminating Filter. A <i>CMIS filter</i> which selects <i>events</i> to be forwarded based on a certain/specified criteria. The events selected are forwarded to all the listeners who meet the criteria. [CR]
<b>EDS</b>	Event Distribution Server. A UNIX daemon process. There could be more than 2 EDS running (default being 2), responsible for reception of <i>events</i> from the <i>MIS</i> and distributing them amongst the registered clients. [CR]
<b>EFD</b>	Event Forwarding Discriminator. In <i>OMNIPoint 1</i> terminology (and in Solstice EM), an <i>object</i> that decides whether and where to forward a <i>notification</i> received from a <i>managed object</i> . Conceptually equivalent to a <i>sieve</i> .
<b>EGW</b>	Event Gateway. The name given to a collection of <i>EPR</i> and <i>EDS</i> UNIX processes, handles delivery of <i>CMIS events/notifications</i> . [CR]
<b>element</b>	Sometimes referred to as <i>network element</i> . In <i>OMNIPoint 1</i> terminology, a <i>component</i> or <i>device</i> to be managed (that is, part of a <i>managed resource</i> ). In <i>GUI</i> terminology, a graphical component of the user interface (for example, a line, rectangle, ellipse, etc.).
<b>Element library</b>	Collection of C++ classes that extend the Solstice EM <i>GUI API</i> to permit manipulation of graphic <i>elements</i> within the display.
<b>environment variable</b>	A <i>variable</i> set in the UNIX shell environment to which an application may make reference. For example, used to specify the Solstice EM <i>MIS</i> to which a Solstice EM service connects.
<b>EPR</b>	Event Port Registry. A UNIX daemon process. There is one EPR for every <i>SEM CORBA Gateway</i> installation, it handles registration of clients for reception of <i>events</i> . [CR]
<b>EPS</b>	Events Per Second. The number representing the rate at which an <i>event</i> is distributed from the event source to the event listener.
<b>event</b>	A <i>notification</i> to which a software system must respond when it occurs, but which it did not solicit or control. In network management, a notification of a status change that arises externally, rather than being solicited by the manager. In <i>CMIP</i> usage, an event is a report that is automatically sent from a node agent. In <i>SNMP</i> , this is referred to as a <i>trap</i> .
<b>EventChannel</b>	Allows multiple suppliers and consumers to be connected to them. EventChannel presents itself as a consumer to suppliers and as a supplier to consumers. [CR]

<b>event consumer</b>	A software entity that has registered with the service provider through the <i>MI</i> with a non-null indication <i>callback</i> procedure address.
<b>event generator</b>	A hardware or software device that has undergone a change in <i>state</i> , or in which a certain <i>condition</i> of interest has occurred. This change of state or condition will directly or indirectly cause a new <i>event</i> to be processed by the service provider which then produces and delivers an <i>indication</i> data structure to <i>event consumers</i> that have registered their interest in receiving indications.
<b>EventPort</b>	Is created according to certain constraints and contains the <i>AE-title</i> that identifies the manager domain. [CR]
<b>EventPortFactory</b>	An interface that facilitates dynamic creation of <code>JIDM::EventPort</code> objects by providing the <code>create_event_port()</code> method. [CR]
<b>event reporter</b>	The software entity that causes a new <i>DMI event</i> to be processed by the service provider. Events are 'reported' by calling the service provider entry point <code>DmiIndicate()</code> .
<b>Event Service</b>	Acts as a mediator that decouples a supplier from a consumer. It allows suppliers to send messages to one or more consumers with a single call. [CR]
<b>failover</b>	When service is switched from the master or primary node to one of the backup nodes. [HA]
<b>FDN</b>	Fully Distinguished Name. In the <i>MIT</i> it is the complete path to a <i>managed object instance</i> . See also <i>RDN</i> and <i>name</i> .
<b>filter</b>	<p>The use of a Boolean expression to test a set of <i>attributes</i> in order to select the <i>object</i> to which a network management command is addressed. Object <i>instances</i> that successfully pass the filtering tests become those on which a management operation is performed. Defined by the <i>CMIS</i> specification (ISO/IEC 9595), <i>CMIP</i> filtering capabilities help reduce the network traffic overhead of a management protocol. See also <i>scope</i>.</p> <p>This usage of filter is distinct from the usage in UNIX systems, where a filter is a program that accepts input from one stream and supplies output in another, so that it can be piped to other functions as needed.</p>
<b>FIPS</b>	Federal Information Processing Standards. Summary of standards for computer and communication systems developed by US National Institute of Standards and Technology. Includes <i>GNMP</i> .
<b>Forum</b>	See <i>NMF</i> .
<b>gateway</b>	A computer that interconnects two networks and then routes packets from one to the other. A gateway has more than one network interface.
<b>GDMO</b>	Guidelines for the Definition of Managed Objects. An <i>ISO</i> document (ISO/IEC 10165-1 and ITU Recommendation X.722) that describes a formal specification language used to define the <i>MIT object class</i> syntax, and consequently, the form in which <i>managed objects</i> are defined for client services.

<b>GDMO compiler</b>	A Solstice EM utility ( <i>em_gdmo</i> ) that parses a description written in <i>GDMO</i> format and compiles it for inclusion in the <i>MDR</i> .
<b>GIOP</b>	General Inter-ORB Protocol. This is an abstract that defines transfer syntax and a standard set of message formats to allow different <i>ORBs</i> to communicate between them. [CR]
<b>GNMP</b>	Government Network Management Profile. A US government checklist of network management system features. See also <i>FIPS</i> .
<b>GOSIP</b>	Government OSI Profile. Standards and recommendations for government use of computers and communications. Separate US and UK GOSIP standards exist, but cover the same general topics.
<b>ground state</b>	In the Design Advanced Requests tool, the <i>state</i> that a <i>request</i> is in when it is first created; the first state in a <i>request template</i> . The sole requirement for this state is that it have a <i>severity</i> of normal.
<b>group</b>	A collection of <i>attributes</i> . A group with multiple <i>instances</i> is called a <i>table</i> .
<b>GUI</b>	Graphical User Interface. Solstice EM uses the Motif Window Manager to generate application interfaces using the Motif library of interface routines.
<b>hop</b>	The traversing of a <i>router</i> . The hop count is a metric for distance used in IP routing. Used in the <i>Network Discovery</i> application to delimit the extent of Network Discovery's search.
<b>host</b>	Any network node that accepts logins. In <i>Internet</i> terminology, a host is an end-system. However, in the Solstice EM screen interface, a host is any node in the network that is running an <i>SNMP</i> or <i>CMOT agent</i> (such as a workstation, bridge, <i>router</i> , terminal concentrator, or switch).
<b>ICMP</b>	Internet Control Message Protocol. Protocol that specifies error and control messages used with <i>Internet protocols</i> .
<b>IDL</b>	Interface Definition Language. A high-level, declarative language for defining the interface of a distributed <i>object</i> . [CR]
<b>IETF</b>	Internet Engineering Task Force. Source of <i>MIT</i> , <i>SNMP</i> .
<b>IIOP</b>	Internet Inter ORB protocol. This is a <i>TCP/IP</i> implementation of <i>GIOP</i> , which is mandated by <i>OMG</i> to be implemented by all <i>ORB</i> vendors who conform to CORBA 2.0. [CR]
<b>index</b>	See <i>class instance</i> .
<b>indication</b>	An unsolicited report, either from a <i>component instrumentation</i> to the service provider, or from the service provider to a <i>management application</i> .

<b>instance</b>	In C++, a piece of data whose structure is described by its membership in a <i>class</i> . Access to the data is provided only by the member functions defined by the class. For <i>managed objects</i> , a specific case of a managed object. For example, <i>router</i> might be taken as an <i>object class</i> ; one particular router would be an <i>instance</i> of that class.
<b>instance identifier</b>	<p>In the <i>OSI</i> world, the <i>FDN</i> of an <i>instance</i> in an <i>MIT</i>. The instance identifier is unique within a given <i>system</i>.</p> <p>In <i>SNMP</i>, the values for the <i>attributes</i> of a <i>base object</i> take the form of a <i>table</i>. Attributes define the columns, and <i>class instances</i> define the rows. The instance identifier comprises the elements in a row that uniquely identify that row. Some base objects in vendor <i>MIB</i> are extensions of base objects in standard <i>MIB</i>. For such a base object, the instance identifier might be found in the original base object, not the extension.</p>
<b>instance specifier</b>	See <i>instance identifier</i> .
<b>Internet</b>	A large collection of connected networks, primarily in the US, running the <i>Internet</i> suite of <i>protocols</i> . The term Internet refers to a collection of <i>TCP/IP</i> internetworks.
<b>interoperability</b>	The capability of two or more systems to meet user requirements by communicating through specific mechanisms in a known environment.
<b>IP address</b>	A 32-bit quantity used to represent a point of attachment in a <i>TCP/IP</i> -based <i>Internet</i> .
<b>ISO</b>	International Standards Organization. Develops standards, by international agreement, over a wide range of technical areas.
<b>JIDM</b>	Joint Inter-Domain Management. A set of specifications/standards for internetworking of <i>CORBA</i> , <i>OSI</i> and <i>SNMP</i> systems management reference models. [CR]
<b>JIDMExt::ProxyAgent</b>	This <i>object</i> is derived from <i>JIDMExt::ProxyAgent</i> , it implements text equivalent of <i>CMIS</i> commands. [CR]
<b>JIDM::ProxyAgent</b>	A <i>CORBA object</i> that implements the <i>JIDM</i> specified <i>OSI</i> Network Management functionality ( <i>cmis_get</i> , <i>cmis_set</i> , <i>cmis_create</i> , <i>cmis_create_sync</i> , <i>cmis_delete</i> and <i>cmis_action</i> ). This <i>CORBA object</i> is created dynamically by the SEM <i>CORBA RGW</i> . Also called <i>ProxyAgent</i> . [CR]
<b>JIDM::ProxyAgentController</b>	A <i>singleton object</i> which is created and exposed by the SEM <i>CORBA RGW</i> . The Controller object is used by the <i>RGW</i> in the process of destroying the <i>ProxyAgent</i> on request from the client applications. Also called <i>ProxyAgentController</i> . [CR]

<b>JIDM::ProxyAgentFinder</b>	A <i>singleton object</i> which is created and exposed by the SEM CORBA <i>RGW</i> . It is used by the client applications to find the <i>JIDM::ProxyAgent</i> required. If the required <i>ProxyAgent</i> does not exist, the finder <i>object</i> will create one, and return the reference of the created object. Also called <i>ProxyAgentFinder</i> . [CR]
<b>key</b>	An identifier of a particular <i>instance</i> (row) of a <i>table</i> .
<b>LDN</b>	In the <i>MIT</i> it is the path to a local <i>managed object instance</i> .
<b>localized string</b>	A version of a display string that is a translation of the original string into an equivalent string in the appropriate local language.
<b>logical host</b>	Has one of its <i>physical hosts</i> impersonate the logical <i>host's</i> host name and <i>IP address</i> . When a <i>failover</i> occurs, one of the other surviving physical hosts will impersonate the logical host's host name and IP address. [HA]
<b>log object</b>	An <i>object</i> that contains several <i>attributes</i> used to test incoming <i>events</i> . If an event occurs, and passes all the log object tests, a <i>log record</i> of the event is stored in the <i>system</i> . The tests consist of a discriminator (test against the attributes of the event), and a test against a specified size (if the size of the log records, in octets, is greater than the allowable size, the new record is not created).
<b>log record</b>	An <i>object</i> that contains the <i>attributes</i> for an <i>event</i> which passed the <i>log object</i> tests.
<b>major trap number</b>	The <i>SNMP</i> standard RFC1157 defines seven categories of <i>traps</i> , numbered 0 through 6.
<b>managed line</b>	Any line, created as a Solstice EM line icon, with manageable line attachments.
<b>managed node</b>	A network computer, <i>router</i> , hub, or other piece of equipment on the network that has <i>object classes</i> entered in the Solstice EM <i>MIT</i> and a network <i>agent</i> running on it.

**managed object** The representation of a network resource (or a set of resources). Note that in general a managed object is an abstraction that represents selected *attributes* of the resource it represents. The managed object resides within the *MIS*, where it represents a resource that is elsewhere. A managed object is characterized by:

- attribute visible at its boundary
- management operations that may be applied to it
- behaviors it exhibits in response to management operations
- *notification* that it emits

A *MIB* or *MIT* entry that represents some aspect of a network node or line that can be monitored and, in some cases, set, using Solstice EM services. The *MIS* manages the *object* by *polling* it, displaying the attribute values for current object *instances* of it, and in some cases changing the attribute values for instances of it.

**managed object class** The formal description of a set of *managed objects*.

**managed resource** A *device* or *component* in a network whose status or behavior is represented by a *managed object*.

**managed system** A network node running a *network management agent* and providing network data to a *managing system* on request.

**management agent** A *network management protocol agent* (such as *SNMP* or *CMOL*) that can communicate to the *DMI* through the *MI*.

**management application** Code that uses the *MI* to request management activity from *components*.

**managing system** The *system* requesting information from and setting information in a network node running a network-management system.

**MDR** Metadata Repository. Within the Solstice EM *MIT*, the storage devoted to the descriptions of data formats.

**metadata** The set of descriptions for forms of data used to describe *managed objects* in a network (as distinct from the data itself).

**MGW** Metadata Gateway. A UNIX daemon process, which provides the CORBA clients with access to *ASN.1 metadata* in Solstice EM. [CR]

**MI** Management Interface. The *DMI* layer between *management applications* and the service provider.

**MIB** Management Information Base. A hierarchical system for classifying information about resources in a network. By industry agreement, individual developers are assigned portions of the tree structure to which they may attach descriptions specific to their own *devices*.

<b>MIB II</b>	An extension to the <i>SNMP MIB</i> .
<b>MIB module</b>	A collection of <i>managed objects</i> .
<b>MIF</b>	Management Information Format. The format used by the <i>DMI</i> for describing <i>components</i> .
<b>MIF database</b>	The collection of known <i>MIF files</i> stored by the service provider (in an implementation-specific format) for fast access.
<b>MIF file</b>	A file that uses the <i>MIF</i> to describe a <i>component</i> .
<b>MIS</b>	Management Information Server. The Solstice EM software process that serves network management clients. The process running on a network workstation or server that maintains network management information in a database according to the definitions in its <i>MIT</i> , provides <i>polling</i> , filtering, and other services to Solstice EM services and various other applications.
<b>MIS Objects</b>	The Solstice EM tool used to inspect, and sometimes change, <i>objects</i> in the <i>MIT</i> .
<b>MIT</b>	Management Information Tree. A naming tree for an <i>MIS</i> or a set of <i>MIS</i> 's. The structure that organizes access to all information stored in the Solstice EM <i>MIS</i> . Each <i>object</i> within the tree is identified by its <i>FDN</i> , corresponding to the path through the tree to reach it.
<b>MOC</b>	Managed Object Class. In Solstice EM, a <i>class</i> for the description of classes of <i>managed objects</i> . An <i>instance</i> of MOC is a specific class.
<b>MOI</b>	Managed Object Instance. In Solstice EM, each set of facts about specific <i>managed resources</i> is stored in an <i>object</i> that is an <i>instance</i> of a <i>managed object</i> .
<b>MPA</b>	Management Protocol Adapter. In Solstice EM, the <i>CMIP MPA</i> provides communication services between <i>OSI managed resource</i> and the Solstice EM <i>MIS</i> via the <i>CMIP protocol</i> .
<b>MRI</b>	Metadata Repository Interface. The SEM CORBA <i>MGW</i> implements the <i>MRI</i> , which provides methods to access the <i>ASN.1 metadata</i> in Solstice EM. [CR]
<b>MRM</b>	Message Routing Module. In Solstice EM, the module that decodes the destination of a received message and transmits it to the appropriate part of the <i>MIS</i> . The MRM serves both messages transmitted through the network from network <i>agents</i> and messages sent internally from one portion of the <i>MIS</i> to another.
<b>name</b>	An <i>FDN</i> in the <i>MIT</i> specifies the complete path to a <i>MOI</i> . An <i>RDN</i> specifies the path to a <i>managed object instance</i> beginning from its parent. The <i>FDN</i> consists of concatenated <i>OID</i> -value pairs, one pair for each junction down the tree.
<b>Naming context</b>	An <i>object</i> that stores <i>name</i> bindings. [CR]
<b>Naming services</b>	The simplest and most basic of standardized CORBA services. It provides a mapping from <i>names</i> to <i>object</i> references. It is a service provided by the <i>ORB</i> vendor. [CR]

<b>Nerve Center</b>	A portion of the <i>MIS</i> that <i>polls</i> and receives <i>notifications</i> from the <i>agents</i> of <i>managed objects</i> . The main job of the Nerve Center is to start and maintain <i>requests</i> .
<b>Network Discovery</b>	A Solstice EM tool that locates manageable objects and adds them to the <i>MIS</i> .
<b>network element</b>	See <i>element</i> .
<b>network management agent</b>	The implementation of a <i>network management protocol</i> (a program) that exchanges network management information with a <i>network management station</i> .
<b>network management protocol</b>	The <i>protocol</i> used to convey management information. For example, <i>SNMP</i> and <i>CMIP</i> .
<b>network management station</b>	A computer on the network running Solstice EM.
<b>Network Views</b>	The Solstice EM tool that manages and displays a graphic representation of a set of <i>managed objects</i> in a network.
<b>NMF</b>	Network Management Forum. An association of vendors and developers of network hardware and software dedicated to the promotion of interoperable network management, based on the use of <i>OSI</i> techniques.
<b>non-container icon</b>	In <i>Network Views</i> , an icon for an <i>object</i> that cannot contain any objects. Examples of non-container icons are page -11s, <i>routers</i> , links, and hubs. A non-container icon is distinguished from a <i>container icon</i> .
<b>notification</b>	<p>A message initiated by an <i>agent</i> reporting a change in the <i>state</i> of a <i>managed resource</i>. The set of notifications that the <i>Nerve Center</i> knows about — and therefore the notifications that you can receive through a <i>request</i> — are defined in <i>GDMO</i> documents that are compiled and loaded into the <i>MIS</i>.</p> <p>Alarms, which are extraordinary events such as an equipment failure, are a proper subset of notifications. Solstice EM has a default set of <i>alarms</i>. Through the <i>request facilities</i>, <i>requests</i> can subscribe to receive incoming notifications and take appropriate <i>action</i>. See also <i>event</i>.</p>
<b>NSAP</b>	Network Service Access Point. Used to identify the remote system and subnetwork. This can be an X.25 address for CONS, a LAN address for CLNP, or an <i>IP address</i> for <i>TCP/IP</i> .
<b>OAM</b>	Object Access Module. In Solstice EM, translates requests for information that refer to objects in network terminology (for example, by network addresses or by the distinguished names of <i>objects</i> in the <i>MIT</i> ) to the methods required to access the C++ objects in which data is stored within the <i>MIS</i> .



**object** In an *MIB*, an *object class* or *attribute* [S]. For Solstice EM, a collection of data that describes a manageable *network element*. Objects can be physical or logical. Examples of physical objects are network nodes; examples of logical objects are views and their icons, *requests*, and *queues*. [C]

The term object is used in several senses. In the *OSI* and *Forum* terminology for network management, an object is any *device*, process, or *event* that can be managed. In object-oriented programming such as the C++ in which Solstice EM is written, an object is similarly any process or device represented by a data structure, with methods (functions) provided by the *class* to which the object belongs. Solstice EM benefits from treating objects in the *OSI/Forum* sense as objects in the C++ sense.

**Object Adapter** Provides an interface between the object and a client application. [CR]

**object class** A particular set of attributes that define an *object instance*. The object class can have children object classes in the form of a tree beneath it. In the *CMIP* environment, an *SNMP MIB* is an object class, and each *base object* in the MIB is an object class.

An object class can be managed; the values of its attributes for each class instance can be polled, which implies that a software *agent* running on a node can report the instance values of the attributes to Solstice EM. See also *class instance* and *attribute*.

In Solstice EM, you are sometimes required to specify a class instance. For example, if the object class comprises port information for a *router* port, you could specify an instance of the object class by providing a router board and port number for a particular port. The information you provide to specify a class instance is called the *instance identifier*; synonyms are *RDN*, *index*, Instance String, and Named Object. Solstice EM also uses the term Instance Syntax. In Solstice EM, an object class is a type of property. Property groups contain two types of properties: object classes, which can be managed, and user-defined strings, which are used to define and limit the *scope of polls* and *requests*.

**octet** An 8-bit quantity.

**OID** Object IDentifier. A number that identifies an *object's* position in a global object registration tree. An example is 1.3.6.1.4.1.45.1.3.2, which corresponds to ios.org.dod.internet.private.enterprise.synoptics.1.3.2, and identifies a Synoptics3000 concentrator. There can also be an *MIB* name for the object identifier (for example, *cisco* for a Cisco router). [S] In *CMIP*, one half of the *RDN* pair, which identifies an object's position in an *MIT*. See *name*. [C]

An OID uses a *system* for describing an object's *class* by reference to a standard tree structure of descriptions. Each node of the tree is assigned a number, so that an object's identifier is a sequence of numbers. In *Internet* usage, the identifiers are shown as a string of numbers delimited by dots (for example, 0.128.45.12); in the *OSI* context (and in Solstice EM) the numbers are delimited by blanks and the entire sequence is surrounded by braces (for example, {0 128 45 12}).

<b>OMG</b>	Object Management Group. Consortium of users of object-oriented techniques. Sponsors of <i>CORBA</i> .
<b>OMNIPoint 1</b>	Set of standards, implementation specifications, and tools developed by the <i>NMF</i> .
<b>OMNIPoint Partners</b>	The sixteen organizations that collaborated in the formulation of <i>OMNIPoint 1</i> .
<b>ORB</b>	Object Request Broker. [CR]
<b>OSF</b>	Open Systems Foundation. UNIX consortium including Hewlett-Packard™, IBM™, and DEC™, founded in 1988. Sponsors of <i>DME</i> .
<b>OSI</b>	Open Systems Interconnection. General name for the set of network management conventions adopted by the <i>ISO</i> . An international effort (via <i>ISO</i> ) to facilitate communication among computers of varying manufacturers and technology.
<b>OSI/NMF</b>	OSI Network Management Forum. An <i>OSI</i> group formed to develop and promulgate definitions and standards for the <i>SNMP</i> , <i>PING</i> , and <i>CMIP protocols</i> .
<b>parent</b>	An <i>instance</i> of the <i>class</i> containing a ( <i>child</i> ) <i>object</i> . [C]
<b>physical host</b>	One of the physical nodes of a cluster. [HA]
<b>PING</b>	Packet Internet Groper. A program that uses the <i>ICMP protocol</i> for requesting acknowledgment from an <i>IP address</i> as a way of testing its existence or accessibility. Informally, used as a verb meaning “to send a signal to test response.” The program <i>ping</i> tests IP-level connectivity from one IP address to another.
<b>PMI</b>	Portable Management Interface. Within the Solstice EM <i>MIS</i> , the set of widely used <i>classes</i> and functions that provide fundamental services.
<b>P+M interface</b>	The set of <i>protocols</i> and messages (in those protocols) necessary for communication between management entities.
<b>POA</b>	Portable Object Adapter. The POA is a more complete object adapter and is fast replacing the Basic Object Adapter (BOA). It also makes better inter-operability of the <i>ORBs</i> possible. [CR]
<b>poll</b>	A periodic <i>request</i> for <i>MIB</i> or <i>MIT object class</i> status information sent to a <i>managed object</i> . Configurable in some cases by the network administrator via the Solstice EM Design Advanced Requests tool. <i>SNMP</i> tends to be poll-oriented, while <i>CMIP</i> tends to be <i>event-oriented</i> .
<b>polling</b>	The process by which the <i>request facilities</i> in the <i>MIS</i> periodically obtain data from an <i>agent</i> according to the specifications of a <i>request template</i> . The goal of polling is to obtain the current values of <i>attributes</i> of a <i>managed object</i> .

- poll rate** Specifies the delay until the first *poll* and the interval between successive polls. The Design Advanced Requests tool associates a name and a number (in seconds) with each poll rate. This tool offers a ready-made list of poll rates and allows you to create your own poll rate.
- Upon the conclusion of a poll rate interval, the *request facilities* begin to test the *condition* for the *transition* that lead from a given *state*. Each state has a poll rate and a *severity* associated with it.
- process monitor** A software component that monitors processes and restarts them automatically upon failure. [HA]
- protocol** A set of rules used by computers to communicate with each other. A protocol is also the private language and procedure of an *OSI* layer.
- proxy** A virtual object representing another *object* that cannot be addressed directly (because it is outside the domain of interest or because it requires a different *protocol*). The Solstice EM *MIS* (which uses *CMIP* for its internal protocol) creates a proxy to represent each *SNMP agent* with which it deals. The proxy then speaks Solstice EM's *CMIS*-like protocol internally, but uses *SNMP* to converse with *SNMP agents*.
- proxy agent** Software running on a node that responds to and reports current information about another node to the Solstice EM *MIS*, which has no agent running on it. See also *agent*. It is often used to shield network *devices* from *CMIP* overhead.
- PSEL** Presentation Selector. Used to select the entity above the presentation layer, the *CMIP* application (manager or *agent*).
- Pull model** Consumers pull *events* from the *EventChannel*, and the *EventChannel* pulls events from suppliers. [CR]
- Push model** Suppliers push *events* to the *EventChannel*, and the *EventChannel* pushes events to consumers. [CR]
- RCL** Request Condition Language. A script language, similar to the C programming language, used to create *conditions* for use in a *request*.
- RDN** Relative Distinguished Name. An RDN in the *MIT* specifies the path to a *managed object instance* beginning from its parent.
- request** The series of activities through which the *request facilities* in the *MIS* poll for the *attribute* of managed objects and receive *notification* from the *agents* of *managed objects*. A request is based on a *request template* and is targeted on a specific managed object. (A request that subscribes to receive *event* notifications may be launched without being targeted on a specific *object*.) Each request is made up of multiple *states*, with potentially, multiple *transitions* between those states.
- You can launch requests in the Requests window, which is started from the *Network Views* tool. Once started, a request remains alive until you stop it, in the Request Monitor.

<b>request</b>	A message the goes from the client applications to <i>SEM CORBA Gateway</i> . [CR]
<b>request facilities</b>	A portion of the <i>MIS</i> that <i>polls</i> and receives <i>notifications</i> from the <i>agents</i> of <i>managed objects</i> . The main job of the request facilities is to start and maintain <i>requests</i> .
<b>request template</b>	The specific form of a <i>request</i> , as created in the Design Advanced Requests tool. Any number of requests can be built on a single request template. A request template contains the <i>states</i> , <i>conditions</i> , and <i>transitions</i> that make up a request. Each request template has a name, which you can use in other tools to start, stop, or view a request. When you start a request, you are applying a specific request template to a specific <i>managed object</i> .
<b>resource group</b>	A collection of <i>X resources</i> that are managed by the Resource Group Manager (RGM) as a unit. Each resource that is to be managed by the RGM must be configured in a resource group. Typically, related and interdependent resources are grouped. [HA]
<b>response</b>	The final response from an <i>indication</i> .
<b>response</b>	A message that is sent back to the client in response to a <i>request</i> . [CR]
<b>RFC</b>	Request for Comment. The series of documents that formalize <i>protocols</i> within the <i>Internet</i> (TCP/IP-based) community. The last phase in the formal standardization process before the document is made official. RFCs are published by the <i>IETF</i> .
<b>RGW</b>	Request Gateway. A UNIX daemon process, started on startup of an <i>MIS</i> , handles initial client connections to the <i>SEM CORBA Gateway</i> , and <i>CMIS requests/responses</i> on any <i>managed object</i> . [CR]
<b>Root naming context</b>	Determines the points at which clients gain access to the naming graph. See also <i>Naming context</i> . [CR]
<b>router</b>	The term routing refers to the process of selecting a path over which to send packets, and router is any computer able to make such a selection. Although both <i>hosts</i> and gateways do routing, the term router is commonly used for a <i>device</i> that interconnects two networks. See also <i>gateway</i> .
<b>RPC MPA</b>	Remote Procedure Call Management Protocol Adapter. Provides the mechanism to get data and set <i>attribute</i> values for <i>devices</i> that are managed via RPC-based <i>agents</i> . The RPC MPA works as a <i>proxy agent</i> between the Solstice EM <i>MIS</i> and any device on the network having RPC agents installed on it.  The RPC MPA serves a major role in providing compatibility with SunNet™ Manager 2.2 or later products.
<b>SAP</b>	Service Access Point. The notional point at which a service user and a layer entity can meet so that services can be offered by the layer entity to the particular user.

<b>Schema compiler</b>	SNM 2.x schema files get compiled by the schema compiler (em_snm2gdmo) into <i>GDMO</i> and <i>ASN.1</i> descriptions which then get loaded into the Solstice EM <i>MIS</i> by other utilities. Users can then send <i>requests</i> to the <i>MIS</i> which get forwarded to the <i>RPC MPA</i> .
<b>scope</b>	<p>In general, a definition of the extent or boundaries of an <i>action</i> to be taken. For example, a <i>poll</i>'s scope is the set of nodes and/or lines to be polled. In <i>Network Discovery</i>, scope refers to the top level of a <i>subnet</i>, represented as a cloud icon. Scoping identifies the sub-tree of the <i>MIT</i> on which a <i>filter</i> is to be applied. <i>CMIP</i> scoping and filtering help reduce traffic overhead from management <i>protocols</i>.</p> <p>The sub-tree within an <i>MIT</i> to which a management command is applied. The scope is described by the node of the parent tree at which the sub-tree is rooted, and the depth (number of nodes) to which the scope extends.</p> <p>Typically, a network command is qualified both by its scope and by a filter that further selects <i>objects</i> with the scope.</p>
<b>SEM</b>	Solstice™ Enterprise Manager.
<b>SEM CORBA Gateway</b>	Name given to the conceptual entity that consists of several UNIX processes and libraries defining interfaces. Together these processes and libraries provide a means for the customer to develop and run client applications that interact with the <i>MIS</i> to achieve various management functions on <i>managed objects</i> . [CR]
<b>SEM CORBA ToolKit</b>	The <i>SEM CORBA Gateway</i> is delivered in the form of SEM CORBA ToolKit. It is more than a framework, but less than a complete application. You will need to use the ToolKit to build the SEM CORBA Gateway. It consists of <i>ORB</i> dependent source code (IDL and C++) and <i>ORB</i> independent shared libraries and Makefiles. [CR]
<b>severity</b>	<p>Describes the degree of importance you attach to each <i>state</i> in a <i>request</i>. A severity is made up of three items: a name, a number, and a color. The severity color is reflected in the color of the icon for the <i>managed object</i> as that <i>object</i> is displayed in the <i>Network Views</i> window. The Design Advanced Requests tool offers a ready-made list of severities and allows you to create your own severity. Each state has a severity and a <i>poll rate</i> associated with it.</p> <p>Severity has two meanings in the context of the Design Advanced Requests tool:</p> <p>(1) Each state has a severity and a poll rate associated with it. A severity is made up of three items: a name, a number, and a color. As a value attached to a state, a severity has a meaning that applies only within the tool itself — for</p>

example, by indicating the appropriate color to use in representing a state in the graphical State Machine display. The Design Advanced Requests tool offers a ready-made list of severities.

(2) A severity can also be attached as a value to a *Nerve Center alarm* posted to the alarm log, using one of the *request facilities* alarm log commands {`alarm()`, `alarmOi()`, `alarmStr()`}. This Nerve Center alarm severity is reflected in the color of the icon for the *managed object* as that *object* is displayed in *Network Views* window.

**shared conceptual  
schema**

Basis for co-operation between managers:

- Management functions
- Managed object classes
- Available *instances* of the *managed object classes*
- Authorization

**sibling** An *object* that shares a common parent *class* with the object in question.

**sieve** In the *NMF* terminology, an *object* that decides whether and where to forward a *notification* received from a *managed object*. Superseded by the *OMNIPoint 1* term *EFD*.

**singleton object** An *object* whose reference count cannot be more than one (1), which means only one *instance* of the object is present at any given time. [CR]

**SMAE** System Management Application Entity. *OSI* terminology for a software *MIS* that manages a network. For example, Solstice EM.

**SMI** Structure of Managed Information. General term for various ways of specifying the information available about an *object* (including *GDMO* object descriptions, *MIBs*).

**SNMP** Simple Network Management Protocol. A *protocol* for exchanging information between network manager and *agent* processes within various *managed objects* that are able to report their status on *request*. The protocol was introduced as a simple interim solution, but is at present widely used in the *Internet* environment. It is a connection-less protocol, with the view of continuing to receive information from managed objects even when network performance is degraded and a connection-based reliable transport may fail.

**SNMP MPA** SNMP Management Protocol Adapter. Allows retrieving of data and setting of *attribute* values for *SNMP* managed devices. The *SNMP MPA* works as a *proxy agent* between the Solstice EM *MIS* and any *device* on the network which is *SNMP-manageable*.

**SSEL** Session Selector. Used to select the entity above the session layer, the presentation layer of Solstice *CMIP*.

<b>state</b>	<p>A description of a <i>managed object</i> in a point in time with respect to a <i>request</i>. At any given moment, a request, reflecting the target managed object, is in some state defined in that request or is undergoing a <i>transition</i> between states.</p> <p>You can think of a state as a receptacle that holds transitions to other states. While in a state, a request repeatedly, at intervals determined by the state's <i>poll rate</i>, tests the <i>condition</i> associated with each transition leading from that state.</p> <p>In addition to a poll rate, each state has a <i>severity</i> associated with it. It also has a name and a description. Between any two states, there is a single transition (one-way or two-way) with, potentially, multiple conditions associated with each transition.</p> <p>There is one required state, the ground (or init) state. The only requirement for this state is that it have a severity of "normal". Other states can be created by you. There is no limit to the number of states.</p>
<b>subnet</b>	In <i>Internet</i> parlance, a logical partition of a network. <i>OSI</i> attaches a more restricted meaning: the portion of a network attached to the same physical medium.
<b>subnet mask</b>	A 32-bit quantity indicating which bits in an <i>IP address</i> identify the physical network.
<b>switch latency</b>	The time taken to switch from a primary node to a backup node. [HA]
<b>system</b>	A computer.
<b>system object</b>	A Solstice EM <i>MIS</i> running on a network node. Also known as <code>agentCME</code> object.
<b>table</b>	An <i>SNMP</i> term that describes a set of <i>attribute</i> values for <i>object class instances</i> . The rows represent the attributes and the columns represent <i>class instances</i> .
<b>TCP/IP</b>	Transmission Control Protocol/Internet Protocol. The <i>Internet</i> suite of <i>protocols</i> is a group of protocols related to a common framework, or set of rules that defines how computers communicate with each other in an open (non-proprietary) system, typically a large communication infrastructure.
<b>TMN</b>	Telecommunications Management Network. A model proposed by the ITU.
<b>top class</b>	In the <i>NMF</i> system for the description of <i>managed objects</i> in a network, the root class from which all other <i>classes</i> are derived.
<b>traceroute</b>	A program that attempts to reconstruct the route from (and hence to) an <i>IP address</i> .
<b>transaction</b>	In database terminology, a sequence of operations that are to be regarded as indivisible. Either the entire set takes place, or none of it takes place. In practice, this means that the database must be able to stop a transaction that is aborted, so that the data is left as it was before the transaction was attempted.

**transition** In the context of a *request*, the change from one *state* to another, which occurs when a *condition* associated with a transition evaluates to true. From one state, a request can make transitions to multiple states, including the state from which the transition started. Within a pair of states (or from and to itself) there can be a single transition. Each transition is associated with one or more conditions. Where there are multiple conditions, the *request facilities* evaluate conditions in the order in which they are entered in the *request template*.

When a transition occurs, depending on the specifications in the request template, a set of *actions* might be performed to set variables or send *notifications*.

**trap** In *Internet* jargon, *notification* of a problem that an *agent* sends to a management *MIS* on its own initiative rather than in response to a *poll*. *SNMP* formally defines seven types of traps and permits subtypes to be defined. *OMNIPoint 1* uses the term *event* rather than trap.

**trap-directed polling** A hybrid form of trouble reporting in which a single *trap* initiated by an *agent* is followed up by *polls* in which the management *MIS* requests further information.

**trigger** The *condition* that causes a *state transition*.

**TSEL** Transport Selector. Used to select the entity above the transport layer, the session layer of Solstice *CMIP*.

**UDP** Universal Datagram Protocol. A connectionless *protocol* over which *SNMP* is usually implemented.

**unicode** A character encoding standard defined by the Unicode Consortium. Unicode characters are two octets each. When the first octet is zero, the second octet maps to the characters in ISO 8859-1.

**variable** A quantity associated with a *managed object* that, along with *attribute*, is the target of a *polling* operation. The *request facilities* allows you to specify variables as you need them when you create a *condition*. It types these variables at runtime. The request facilities also has a set of system variables, which you will find useful in a variety of different *request templates*.

**view** In Solstice EM, a graphical representation of a set of *managed objects* in a network, as presented by *Network Views*. A view can be part of a hierarchy, imitating the tree-form of a network topology, or non-hierarchical, wherein it is a logical grouping of managed objects in a flat space. In Solstice EM, a view is equivalent to a *container icon*.

**view file** A text file that holds your network's configuration information. The filenames of *view* and subview background pictures; the filenames and positions of icons representing computers, *routers*, hubs, and lines; *poll*, mask, and *request* configuration information; and property groups. Also called a View Database file or View\_db file.



- view icon** An icon in Solstice EM that represents one *view* in a set of nested views. The icon is displayed on the next higher level view. You can use the view icon provided with Solstice EM or define view icons of your own.
- view object** A collection of lines and icons that you situate in a set of *views* with optional, appropriate background pictures for the views to represent your network and its *subnets*.
- VM** Volume Manager. [HA]
- XMP** X/Open Management Protocol. An *API* that provides a management service interface to support the use of both *CMIP* and *SNMP protocols* by *management applications*. Also known as CM-API.
- XOM** X/Open OSI/ASN.1 Manipulation. Specification of a wrapper around *ASN.1* intended to offer the programmer a simplified *API* for ASN.1.
- X resource** In UNIX and X, a file containing set-up information that an application may consult during its operation (typically, when it is initialized) specifying values for fonts, colors, etc.

