

[MS-OXGLOS]: Exchange Server Protocols Master Glossary

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [Open Specification Promise](#) or the [Community Promise](#). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1	Major	Initial Availability.
04/25/2008	0.2	Editorial	Revised and updated property names and other technical content.
06/27/2008	1.0	Major	Initial Release.
08/06/2008	1.01	Editorial	Updated references to reflect date of initial release.
09/03/2008	1.02	Editorial	Changed title of document.
12/03/2008	1.03	Editorial	Revised and edited technical content.
03/04/2009	1.04	Editorial	Revised and edited technical content.
04/10/2009	2.0	Major	Updated technical content for new product releases.
07/15/2009	3.0	Major	Revised and edited the technical content.
11/04/2009	4.0.0	Major	Updated and revised the technical content.
02/10/2010	4.1.0	Minor	Updated the technical content.
05/05/2010	5.0.0	Major	Updated and revised the technical content.
08/04/2010	6.0	Major	Significantly changed the technical content.
11/03/2010	6.1	Minor	Clarified the meaning of the technical content.
03/18/2011	6.2	Minor	Clarified the meaning of the technical content.
08/05/2011	6.3	Minor	Clarified the meaning of the technical content.
10/07/2011	6.4	Minor	Clarified the meaning of the technical content.
01/20/2012	6.5	Minor	Clarified the meaning of the technical content.
04/27/2012	6.6	Minor	Clarified the meaning of the technical content.
07/16/2012	6.7	Minor	Clarified the meaning of the technical content.
10/08/2012	6.8	Minor	Clarified the meaning of the technical content.
02/11/2013	6.9	Minor	Clarified the meaning of the technical content.
07/26/2013	6.10	Minor	Clarified the meaning of the technical content.
11/18/2013	6.11	Minor	Clarified the meaning of the technical content.
02/10/2014	6.12	Minor	Clarified the meaning of the technical content.

Table of Contents

1	Non-Alphanumeric	5
2	0-9	6
3	A	7
4	B	10
5	C.....	11
6	D	14
7	E.....	16
8	F.....	18
9	G	19
10	H	20
11	I	21
12	J	22
13	K	23
14	L.....	24
15	M.....	25
16	N	28
17	O	29
18	P.....	30
19	Q	32
20	R	33
21	S.....	36
22	T.....	40
23	U	42
24	V	43
25	W	44
26	X	45
27	Y.....	46
28	Z.....	47

29 Change Tracking.....48

1 Non-Alphanumeric

3 A

action: (1) The smallest unit of work in a workflow system. An action can contain one or more tasks that define work that **actors** need to do. Actions are deployed and registered in the workflow system to be activated by protocol client users.

(2) A unit of work that can be performed by a workflow and is typically defined in a workflow markup file.

(3) A discrete operation that is executed on an incoming **Message object** when all **conditions** in the same **rule (4)** are TRUE. A rule contains one or more actions.

Action: A type of **MetadataObject** that represents a URL that triggers the display or manipulation of data related to an Entity or EntityInstance. Actions are contained by an Entity. Actions contain **ActionParameters**.

action instance: The runtime instance of a specific **action (1)**. Action instances are building blocks for an **activity flow**. Several action instances can be chained together to form an activity flow, and multiple action instances of the same action (1) can exist in a single activity flow.

ActionParameter: A type of **MetadataObject** that defines how to parameterize the URL of an **Action** with specific data about an EntityInstance. ActionParameters are contained by Actions.

active reminder: A **reminder** that is enabled on an object and is either pending or overdue, depending on whether the **signal time** has passed.

active replica: A name given to a server that hosts content and is expected to serve that content to clients.

activity flow: A running instance of a workflow that consists of a sequence of **action instances** and **activity model** instances. Action instances and activity model instances can be sequenced in any order to create a single activity flow.

activity model: A predefined sequence of **actions (1)**.

actor: A person or process that starts or participates in an **activity flow**. An actor can be an initiator or a target.

address book: A collection of **Address Book objects**, each of which are contained in any number of **address lists**.

address book container: An **Address Book object** that describes an **address list**.

address book hierarchy table: A collection of **address book containers** arranged in a hierarchy.

Address Book object: An entity in an **address book** that contains a set of **attributes (1)**, each attribute with a set of associated values.

address creation table: A table containing information about the templates that an address book server supports for creating new email addresses.

address creation template: A template that describes how to present a dialog to a messaging user along with a script describing how to construct a new email address from the user's response.

address list: A collection of distinct **Address Book objects**.

address type: An identifier for the type of email address, such as **SMTP** and EX.

alias: An alternate name that can be used to reference an object or element.

animation: A record of synthetic, successive still images that produce an illusion of movement when played back.

app for Office: A cloud-enabled app that integrates rich, scenario-focused content and services into an Office application or equivalent protocol client.

application server: A computer that provides infrastructure and services for applications that are hosted on a server farm.

Appointment object: A **Calendar object** that has an organizer but no attendees.

archive policy: A feature that determines when items are moved into an alternate **mailbox** for archival purposes.

archive tag: An element that contains information about the **archive policy** of a **Message object** or folder.

Association: A **MethodInstance** that enables the traversal and manipulation of a data model relationship between a set of source **Entities** and a single destination Entity. An Association can retrieve, associate, and disassociate EntityInstances of a destination Entity if given EntityInstances of other source Entities.

asynchronous context handle: A **remote procedure call (RPC)** context handle that is used by a client when issuing RPCs against a server on AsyncEMSMDB interface methods. It represents a handle to a unique session context on the server.

Asynchronous JavaScript + XML (AJAX): A web programming model that incorporates a set of web technologies including Extensible HyperText Markup Language (XHTML), **cascading style sheets (CSS)**, Document Object Model (DOM), **XML**, Extensible Stylesheet Language Transformation (XSLT), **XMLHttpRequest (XHR)**, and JavaScript. AJAX is designed to make user interaction with the web more responsive.

atom: A unit of information that cannot be divided into smaller parts, and is accepted or rejected in its entirety. See also **Atom Publishing Protocol (AtomPub)**.

atom feed: An **XML** structure that contains metadata about content, such as the language version and the date when the content was last modified, and is sent to subscribers by using the Atom Publishing Protocol (AtomPub), as described in [\[RFC4287\]](#).

Atom Publishing Protocol (AtomPub): An application-level protocol for publishing and editing web resources, as described in [\[RFC5023\]](#).

Attachment object: A set of properties that represents a file, **Message object**, or structured storage that is attached to a Message object and is visible through the **attachments table** for a Message object.

attachments table: A **Table object** whose rows represent the **Attachment objects** that are attached to a **Message object**.

Autodiscover client: A client that queries for a set of server locations where setup and configuration information for an [\[RFC2821\]](#)-compliant email address is stored.

Autodiscover server: A server in a managed environment that makes setup and configuration information available to **Autodiscover clients**. The location of Autodiscover servers is made available via the Autodiscover HTTP Service Protocol, as described in [\[MS-OXDISCO\]](#).

4 B

base64 encoding: A binary-to-text encoding scheme whereby an arbitrary sequence of bytes is converted to a sequence of printable **ASCII** characters, as described in [\[RFC4648\]](#).

basic flag: A flag on a **Message object** that indicates that the object has an associated work item or shares a defining characteristic with other Message objects with such flags.

best body: The text format that provides the richest representation of a **message body (2)**. The algorithm for determining the best-body format is described in [\[MS-OXBBODY\]](#).

binary large object (BLOB): A discrete packet of data that is stored in a database and is treated as a sequence of uninterpreted bytes.

binding handle: A data structure that represents the logical connection between a client and a server.

bitmap (BMP): A representation of characters or graphics by individual pixels. The pixels can be arranged in rows (horizontal) and columns (vertical). Each pixel can be represented by one or more bits.

blind carbon copy (Bcc) recipient: An addressee on a **Message object** that is not visible to recipients of the Message object.

body part: A part of an Internet message, as described in [\[RFC2045\]](#).

bookmark: (1) An entity that is used in a document to denote the beginning and ending character positions of specific text in the document, and optionally, metadata about that text or its relationship to other referenced parts of the document.

(2) A data structure that the server uses to point to a position in the **Table object**. There are three pre-defined bookmarks (beginning, end, and current). A custom bookmark is a server-specific data structure that can be stored by the client for easily navigating a **Table object**.

business logic: (1) A set of rules, formulas, validation, and code that define the limits and methods for processing data that is entered into an InfoPath form.

(2) The part of an application that processes data according to the requirements defined in a **line-of-business (LOB) system**. It refers to the routines that perform the data entry, update, query, and report processing, and more specifically to the processing that takes place behind the scenes rather than the presentation logic that is required to display the data.

5 C

cabinet (.cab) file: A single file that stores multiple compressed files to facilitate storage or transmission.

calendar: A date range that shows availability, **meetings**, and appointments for one or more users or **resources**. See also **Calendar object**.

Calendar folder: A **Folder object** that contains **Calendar objects**.

Calendar object: A **Message object** that represents an event, which can be a one-time event or a recurring event. The Calendar object includes properties that specify event details such as description, organizer, date and time, and status.

calendar options dictionary: A dictionary that contains calendar configuration data. It is stored in a **folder associated information (FAI)** message that is in a **Calendar special folder**.

Calendar special folder: A **Calendar folder** that is in a user's **mailbox** and in which meetings are created by default.

carbon copy (Cc) recipient: An address on a **Message object** that is visible to recipients of the Message object but is not necessarily expected to take any action.

cascading style sheet (CSS): An extension to **HTML** that enables authors and users of HTML documents to attach style sheets to those documents, as described in [\[CSS-LEVEL1\]](#) and [\[CSS-LEVEL2\]](#). A style sheet includes typographical information about the appearance of a page, including the font for text on the page.

category: (1) A custom string that is used to group one or more documents.

(2) A string that is used as a suggestion for a document category on a site.

(3) A subdivision of items into useful groups such as geographical regions. For example, categories that represent geographical regions could be North, South, East, and West.

(4) An enhanced presence concept that is used by a **Session Initiation Protocol (SIP)** client to publish or subscribe to **presence (2)** information. A category enables basic identification of the data that is being published; it implies an agreed-upon schema for interpreting the data. A category name identifies a contract between a publisher and a subscriber.

(5) A grouping of rows in a **Table object** that all have the same value for a specified property.

change number: A number that identifies a version of a messaging object. A change number is identical in format to a message ID (MID) or folder ID (FID).

change number set (CNSET): A data structure that is similar to an IDSET, in which the global counters (GLOBCNTs) represent changes instead of messaging objects.

character set: (1) A mapping between the characters of a written language and the values that are used to represent those characters to a computer.

(2) The range of characters used to represent textual data within a **MIME body part**, as described in [\[RFC2046\]](#).

checkpoint ICS state: An **Incremental Change Synchronization (ICS)** state that is provided by a server in the middle of an ICS operation, which reflects the state of the **local replica**, indicated by the **initial ICS state**, after applying all differences transmitted in the ICS operation.

claim: (1) A set of operations that are performed on a workflow task to specify the user who owns it.

(2) A statement that one subject makes about itself or another subject. For example, the statement can be about a name, identity, key, group, privilege, or capability. Claims have a provider that issues them, and they are given one or more values. They are also defined by a claim value type and, possibly, associated metadata.

clear-signed message: An Internet email message that is in the format described by [\[RFC1847\]](#) and is identified with the media type "multipart/signed", or the **Message object** representing such a message. An important class of clear-signed message, based on a "multipart/signed" format, is the S/MIME clear-signed message, as described in [\[RFC5751\]](#) and [\[RFC3852\]](#).

client-side rule: A **rule** that has at least one **action** that is executed by a client because it cannot be executed by a server.

codec: An algorithm that is used to convert media between digital formats, especially between raw media data and a format that is more suitable for a specific purpose. Encoding converts the raw data to a digital format. Decoding reverses the process.

color flag: A flag that extends the concept of a **basic flag** by associating one of a chosen set of color values with a flagged **Message object**.

column: (1) See **field (3)**.

(2) A single set of data that is displayed vertically in a worksheet or a table.

(3) See **column chart**.

column chart: A chart that displays data in vertical bars to facilitate data comparison.

Common Views folder: A **special folder** that contains the data for default views that are standard for a message store and can be used by any user of a client that accesses the message store.

condition: A logical expression comparing one or more properties in all incoming **Message objects** against a set of clauses. This logical expression can evaluate to TRUE or FALSE.

conflict detection: A process that is used to determine whether two versions of the same object conflict with each other, that is, one is not a direct or indirect predecessor of another.

conflict reporting: An automated process that notifies a system actor of a previously detected conflict.

conflict resolution: An automated or semi-automated process that is used to resolve a previously detected conflict between versions of an object. The process replaces conflicting versions with a successor version. How a successor version relates to a conflicting version depends on the algorithm that is used.

contact: (1) A presence entity (**presentity**) whose presence information can be tracked.

(2) An object of the contact class that represents a company or person whom a user can contact.

(3) A person, company, or other entity that is stored in a directory and is associated with one or more unique identifiers and **attributes (2)**, such as an Internet message address or login name.

Contact object: A **Message object** that contains properties pertaining to a **contact (3)**.

Contacts folder: A **Folder object** that contains **Contact objects**.

Container class: The value of the PidTagContainerClass property on a folder, which indicates the default **Message object** type for the folder.

content synchronization: The process of keeping synchronized versions of **Message objects** and their properties on a client and server.

contents table: A **Table object** whose rows represent the **Message objects** that are contained in a **Folder object**.

conversation: A single representation of a send/response series of email messages. A conversation appears in the Inbox as one unit and allows the user to view and read the series of related email messages in a single effort.

conversation action: A limited set of actions that a user applies to all **Message objects** that have the same PidTagConversationId value. The action is applied to all Message objects that are currently in the store or are delivered in the future.

conversation ID: A unique value that is associated with a conversation. It is assigned to each **Message object** that is part of a conversation and it is used to identify the conversation to which the message belongs.

conversation index: A value that specifies the location of a message within a conversation. A client can use this value to identify the parent and child messages of a message, and then generate a tree view of the conversation that contains those messages.

cookie: A small data file that is stored on a user's computer and carries state information between participating protocol servers and protocol clients.

counter proposal: A request that an attendee sends to an **organizer** when requesting a change to the date or time of a meeting.

6 D

database object: An object such as a table, query, form, report, macro, or module that can be referenced by name in a database, database application, or database project.

DataClass: A type of **MetadataObject** that represents a type of a business data object obtained from a **line-of-business (LOB) system**. Instances of a DataClass have transient identity. DataClasses are contained by **LobSystems** and Methods.

Deferred Action Folder (DAF): A **special folder** where a server places all Deferred Action Messages and Deferred Error Messages to be acted on by a client. The Deferred Action Folder is not visible to a user.

Deferred Action Message (DAM): A hidden message indicating to a client that it needs to execute one or more **rules** on another user-visible message in the store.

Deferred Error Message (DEM): A hidden message indicating to a client that it needs to present the user with an error indicating that a **server-side rule** failed to execute.

delegate: A user or resource that has permissions to act on behalf of another user or resource.

delegate access: The access that is granted by a delegator to a delegate and is used by the delegate to access the delegator's account.

Delegate Information object: A **Message object** that contains properties specifying delegate access settings for resources in a delegator's mailbox.

delegate rule: A **server-side rule** that is used to send mail to delegates on behalf of a delegator.

delegator: A user or resource for which another user or resource has permission to act on its behalf.

Deleted Items folder: A **special folder** that is the default location for objects that have been deleted.

delivery receipt: A report message that is generated and sent by a client or server to the sender of a message or another designated recipient when an email message is received by an intended recipient.

Department object: An **Address Book object** that describes a department within an organization.

departmental group: A **distribution list** that describes a department within an organization.

dictionary: A collection of key/value pairs. Each pair consists of a unique key and an associated value. Values in the dictionary are retrieved by providing a key for which the dictionary returns the associated value.

display name: A text string that is used to identify a principal or other object in the user interface. Also referred to as title.

display template: A template that describes how to display or allow a user to modify information about an **Address Book object**.

distribution list: A collection of users, computers, contacts, or other groups that is used only for email distribution, and addressed as a single recipient.

Document object: A **Message object** that represents a single file, such as a document generated by a word-processing application. The Message object contains the file as an **Attachment object** and includes additional properties to describe the file.

double-byte character set (DBCS): A **character set (1)** that can use more than one byte to represent a single character. A DBCS includes some characters that consist of 1 byte and some characters that consist of 2 bytes. Languages such as Chinese, Japanese, and Korean use DBCS.

Draft Message object: A **Message object** that has not been sent.

Drafts folder: A **special folder** that is the default location for **Message objects** that have been saved but not sent.

7 E

electronic discovery (eDiscovery): A type of legal discovery that focuses on electronic (computer) communications and files.

email address: A string that identifies a user and enables the user to receive Internet messages.

Email object: A **Message object** that represents an email message in a message store and adheres to the property descriptions that are described in in [\[MS-OXOMSG\]](#).

Embedded Message object: A **Message object** that is stored as an **Attachment object** within another Message object.

embedded object: An object that is created by using one application and is hosted in a document that was created by using another application. Embedding an object, rather than inserting or pasting it, ensures that the object retains its original format. Users can double-click an embedded object and edit it with the toolbars and menus from the application that was used to create it. See also **Object Linking and Embedding (OLE)**.

encapsulation: A process of encoding one document in another document in a way that allows the first document to be re-created in a form that is nearly identical to its original form.

encrypted message: An Internet email message that is in the format described by [\[RFC5751\]](#) and uses the EnvelopedData CMS content type described in [\[RFC3852\]](#), or the **Message object** that represents such a message.

endpoint: (1) A client that is on a network and is requesting access to a network access server (NAS).

(2) A network-specific address of a remote procedure call (RPC) server process for remote procedure calls. The actual name and type of the endpoint depends on the RPC protocol sequence that is being used. For example, for RPC over TCP (RPC Protocol Sequence ncacn_ip_tcp), an endpoint might be TCP port 1025. For RPC over Server Message Block (RPC Protocol Sequence ncacn_np), an endpoint might be the name of a named pipe. For more information, see [\[C706\]](#).

(3) A participant that uses the Microsoft Groove Dynamics Protocol, as described in [\[MS-GRVDYNM\]](#), to synchronize with a shared space. An endpoint is identified by the combination of an identity URL and a client device URL. Each endpoint maintains a copy of the data in a shared space.

(4) A communication port that is exposed by an **application server** for a specific shared service and to which messages can be addressed.

(5) A device that is connected to a computer network.

enterprise/site/server distinguished name (ESSDN): An **X500 DN** that identifies an entry in an abstract naming scheme that is separate from an **address book**. The naming scheme defines enterprises, which contain sites, and sites contain servers and users. There is no concrete data structure that embodies an ESSDN. Instead, an address book entry can contain an ESSDN as a property of the entry.

Entity: A type of **DataClass** that represents a type of business data object that is stored in a line-of-business (LOB) system and whose instances have a persistent EntityInstanceId.

entry ID: See **EntryID**.

EntryID: A sequence of bytes that is used to identify and access an object.

event: (1) Any significant occurrence in a system or an application that requires users to be notified or an entry to be added to a log.

(2) An action or occurrence to which an application might respond. Examples include state changes, data transfers, key presses, and mouse movements.

event notification: A message that is sent by a subscribed resource to its subscribers or implied subscribers, notifying them of an event on that resource.

Exception Attachment object: An **Attachment object** on a **Recurring Calendar object** that contains the data for an exception, including an Exception Embedded Message object.

Exception Embedded Message object: An **Embedded Message object** that contains the changes for an Exception object.

Exception object: An instance of a **recurring series** that differs from the rest of the recurring series, for example by start time.

extended rule: A **rule** that is added to, modified, and deleted from a server by using a mechanism other than standard rules, but is otherwise functionally identical to a standard rule.

external identifier: A globally unique identifier for an entity that represents either a **foreign identifier** or an **internal identifier (2)**. It consists of a GUID that represents a namespace followed by one or more bytes that contain an identifier for an entity within that namespace. If an external identifier represents an internal identifier (2), it can be also called a **global identifier**.

external OOF message: An **OOF message** that is sent to external users.

8 F

FAI contents table: A table of **folder associated information (FAI)** Message objects that are stored in a Folder object.

FastTransfer download context: A **Server object** that represents a context for a FastTransfer download.

FastTransfer stream: A binary format for encoding full or partial folder and message data. It can also encode information about differences between mailbox replicas.

FastTransfer upload context: A **Server object** that represents a context for a FastTransfer upload.

field: (1) An element or **attribute (1)** in a data source that can contain data.

(2) A container for metadata within a SharePoint list and associated list items.

(3) A discrete unit of a record that has a name, a data type, and a value.

(4) The data elements that constitute an **Entity** in a line-of-business (LOB) system.

File Transfer Protocol (FTP): A member of the TCP/IP suite of protocols that is used to copy files between two computers on the Internet if both computers support their respective FTP roles. One computer is an FTP client and the other is an FTP server.

final ICS state: An **Incremental Change Synchronization (ICS)** state that is provided by a server upon completion of an ICS operation. A final ICS state is a **checkpoint ICS state** that is provided at the end of the ICS operation.

floating: A time that is interpreted in an observer's location and does not necessarily translate into the same **Coordinated Universal Time (UTC)** time in different locations, as described in [\[RFC2445\]](#). For example, a reminder for an appointment that starts at 1/1/2008 at 2:00 P.M. floating time would signal two hours earlier in Athens than in London.

folder associated information (FAI): A collection of **Message objects** that are stored in a Folder object and are typically hidden from view by email applications. An FAI Message object is used to store a variety of settings and auxiliary data, including forms, views, calendar options, favorites, and category lists.

Folder object: A messaging construct that is typically used to organize data into a hierarchy of objects containing Message objects and **folder associated information (FAI)** Message objects.

foreign identifier: An identifier that is assigned to an entity by a foreign system, typically a client. It always has a form of an **external identifier**, but not all external identifiers are foreign identifiers.

free/busy message: A message that is stored in a public folder and contains free/busy data.

free/busy status: A property of an appointment that indicates how an appointment on the **calendar** of an attendee or resource affects their availability.

9 G

Gateway Address Routing Table (GWART): A list of values that specifies the **address types** that are supported by transport gateways.

ghosted: (1) A property that is not deleted by the server if the element is not included in a Sync <Change> request message. By default, elements that are not included in a Sync <Change> request are deleted from the store.

(2) See uncustomized.

ghosted folder: A folder whose contents are located on another server.

Global Address List (GAL): An **address list** that conceptually represents the default address list for an **address book**.

global counter: A 6-byte value that is incremented automatically. If a global counter is paired with a **REPLID** it forms a message ID (MID), folder ID (FID), or change number. If a global counter is paired with a **REPLGUID** it forms a **global identifier**.

global identifier: A form of encoding for an internal identifier that makes it unique across all stores. Global identifiers are a subset of **external identifiers**, and they consist of a **REPLGUID** followed by a 6-byte **global counter**.

Globally Routable User Agent URI (GRUU): A **URI** that identifies a **user agent** and is globally routable. A URI possesses a GRUU property if it is useable by any **user agent client (UAC)** that is connected to the Internet, routable to a specific user agent instance, and long-lived.

Graphics Interchange Format (GIF): A compression format that supports device-independent transmission and interchange of bitmapped image data. The format uses a palette of up to 256 distinct colors from the 24-bit **RGB** color space. It also supports **animation** and a separate palette of 256 colors for each frame. The color limitation makes the GIF format unsuitable for reproducing color photographs and other images with gradients of color, but it is well-suited for simpler images such as graphics with solid areas of color.

group header: A **navigation shortcut** that groups other navigation shortcuts.

10 H

handle array: An array of object handles that are sent to and received from a server as part of a **remote procedure call (RPC)** accompanying **ROP request buffers** and **ROP response buffers**, respectively. Also referred to as a **Server object handle table** or an HSOT table.

hard delete: A process that removes an item permanently from the system. If an item is hard deleted, a server does not retain a back-up copy of the item and a client cannot access or restore the item. See also **soft delete**.

header: (1) A line, or lines, of content in the top margin area of a page in a document or a slide in a presentation. A header typically contains elements such as the title of the chapter, the title of the document, a page number, or the name of the author.

(2) A name-value pair that supplies structured data in an Internet email message or **MIME entity**.

header field: A component of a Session Initiation Protocol (SIP) message header, as described in [\[RFC3261\]](#).

header row: (1) A row in a table, typically the first row, that contains labels for **columns (2)** in the table.

(2) A row at the beginning of a **category (5)** that does not represent data in the **Table object**, but provides information about a grouping.

hierarchy synchronization: The process of keeping synchronized versions of folder hierarchies and their properties on a client and server.

hierarchy table: A **Table object** whose rows represent the **Folder objects** that are contained in another Folder object.

Hypertext Markup Language (HTML): An application of the Standard Generalized Markup Language (SGML) that uses tags to mark elements in a document, as described in [\[HTML\]](#).

11 I

icon file format (ICO): A file format used for icons in the operating system.

ICS state: A set of properties that determine the state of a **local replica** narrowed down to a specific **synchronization scope**.

inactive search folder: A **search folder (2)** that does not have a **search folder container**.

Inbox folder: A **special folder** that is the default location for **Message objects** received by a user or resource.

Incremental Change Synchronization (ICS): A data format and algorithm that is used to synchronize folders and messages between two sources.

Information Rights Management (IRM): A technology that provides persistent protection to digital data by using encryption, **certificates (1)**, and **authentication (2)**. Authorized recipients or users acquire a license to gain access to the protected files according to the rights or business rules that are set by the content owner.

informational update: A **Meeting Update object** that includes a change that does not require attendees to respond again, such as additional agenda details.

initial ICS state: An **Incremental Change Synchronization (ICS)** state that is provided by a client when it configures an ICS operation.

instance: A unique publication of data for a **category (4)**. It enables a **publisher** to publish data for the same category (4) multiple times. An example is a publisher who uses two different **endpoints (5)** to publish data. These endpoints (5) can publish the same category (4). However, each endpoint (5) requires a different instance number to be considered a distinct publication by the **server (2)**. An instance number is provided by the publishing client.

instant messaging: A method of real-time communication over the Internet in which a sender types a message to one or more recipients and the recipient immediately receives the message in a pop-up window.

Integrated Services Digital Network (ISDN): A high-speed digital technology that uses existing telephone lines to provide Internet access and digital network services.

internal identifier: (1) An integer that uniquely identifies any item in a **term store**.

(2) A Folder ID or Message ID, as described in [\[MS-OXCDATA\]](#).

Internet Message Access Protocol - Version 4 (IMAP4): A protocol that is used for accessing email and news items from mail servers, as described in [\[RFC3501\]](#).

Inter-Personal Mail (IPM): Typical user messaging items, such as email and **calendar** items.

interpersonal messaging subtree: The root of the hierarchy of folders commonly visible in a messaging client. This includes mailbox folders (such as the Inbox folder and Outbox folder) and user-created folders, including user-created public folders.

item: A unit of content that can be indexed and searched by a search application.

12 J

JavaScript Object Notation (JSON): A text-based, data interchange format that is used to transmit structured data, typically in **Asynchronous JavaScript + XML (AJAX)** web applications, as described in [\[RFC4627\]](#). The JSON format is based on the structure of ECMAScript (Jscript, JavaScript) objects.

Joint Photographic Experts Group (JPEG): A raster graphics file format for displaying high-resolution color graphics. JPEG graphics apply a user-specified compression scheme that can significantly reduce the file sizes of photo-realistic color graphics. A higher level of compression results in lower quality, whereas a lower level of compression results in higher quality. JPEG-format files have a .jpg or .jpeg file name extension.

journal: A process that generates a Journal-Report for an original-message.

Journal folder: A **Folder object** that contains **Journal objects**.

Journal object: A **Message object** that represents an entry in a journal or log and adheres to the property descriptions that are described in in [\[MS-OXOJRN\]](#).

Junk Email folder: A **special folder** that is the default location for **Message objects** that are determined to be junk email by a Junk Email rule.

Junk Email rule: An **extended rule** that describes a **spam filter**.

14 L

LDAP Data Interchange Format (LDIF): A standard that defines how to import and export directory data between directory servers that use the **Lightweight Directory Access Protocol (LDAP)**, as described in [\[RFC2849\]](#).

Lempel-Ziv Extended (LZX): An LZ77-based compression engine, as described in [\[UASDC\]](#), that is a universal lossless data compression algorithm. It performs no analysis on the data.

Lempel-Ziv Extended Delta (LZXD): A derivative of the Lempel-Ziv Extended (LZX) format with some modifications to facilitate efficient delta compression. Delta compression is a technique in which one set of data can be compressed within the context of a reference set of data that is supplied both to the compressor and decompressor. Delta compression is commonly used to encode updates to similar existing data sets so that the size of compressed data can be significantly reduced relative to ordinary non-delta compression techniques. Expanding a delta-compressed set of data requires that the exact same reference data be provided during decompression.

line-of-business (LOB) system: A software system that is used to store business data and can also contain business rules and **business logic (2)** that support business processes.

linked object: An object that is inserted into a document and continues to exist in a separate source file. If the object in the source file changes, the object in the document is updated automatically to reflect those changes.

LobSystem: A type of **MetadataObject** that represents a specific version of a line-of business (LOB) system. An LOB system can be a database or a web service.

local replica: A copy of the data in a **mailbox** that exists on the client.

local site: A directory services unit that defines the physical structure or topology of a local network.

locale: A collection of rules and data that are specific to a language and a geographical area. A locale can include information about sorting rules, date and time formatting, numeric and monetary conventions, and character classification.

Logon object: A **Server object** that provides access to a private **mailbox** or a **public folder**. A client obtains a Logon object by issuing a RopLogon **remote operation (ROP)** to a server.

long ID (LID): A 32-bit quantity that, in combination with a GUID, defines a **named property**.

15 M

mail app: An **app for Office** that enhances an email or appointment item.

mail tip: A note that is presented to the author of a message when the author is composing the message. A mail tip provides information about the recipients of a message and issues that might impact delivery of the message, such as moderation or delivery restrictions.

mail user: An **Address Book object** that represents a person or entity that can receive deliverable messages.

Mail User Agent (MUA): A client application that is used to compose and read email messages.

mailbox: A **message store** that contains email, calendar items, and other **Message objects** for a single recipient.

marker: An unsigned 32-bit integer value that adheres to **property tag** syntax and is used to denote the start and end of related data in a **FastTransfer stream**. The property tags that are used by markers do not represent valid properties.

meeting: An event with attendees.

Meeting Cancellation object: A **Message object** that represents a meeting organizer's cancellation of a previously scheduled meeting.

Meeting Forward Notification object: A **Message object** that represents a notification that is sent to the meeting organizer when an attendee forwards a meeting request.

Meeting object: A **Calendar object** that has both an organizer and attendees.

meeting request: An instance of a **Meeting Request object**.

Meeting Request object: A **Message object** that represents an invitation from the meeting organizer to an attendee.

Meeting Response object: A **Message object** that represents an attendee's response to a meeting organizer's invitation. The response indicates whether the attendee accepted, tentatively accepted, or declined the meeting request. The response can include a proposed new date or time for the meeting.

meeting update: An instance of a **Meeting Update object**.

Meeting Update object: A **Message object** that represents a meeting organizer's changes to a previously scheduled meeting. The update is categorized as either a full update or an informational update.

Meeting Workspace: A website that is created by using the Meetings Web Services protocol, as described in [\[MS-MEETS\]](#). It can host documents, discussions, and other information about a meeting.

meeting-related object: A **Message object** that represents a relay of information between a meeting organizer and an attendee. It can be any of the following: **Meeting Request object**, **Meeting Update object**, **Meeting Cancellation object**, or **Meeting Response object**.

message body: (1) The content within an HTTP message, as described in [\[RFC2616\]](#).

(2) The main message text of an email message. A few properties of a **Message object** represent its message body, with one property containing the text itself and others defining its **code page** and its relationship to alternative body formats.

message class: A property that loosely defines the type of a message, contact, or other Personal Information Manager (PIM) object in a mailbox.

Message object: A set of properties that represents an email message, appointment, contact, or other type of personal-information-management object. In addition to its own properties, a Message object contains recipient properties that represent the addressees to which it is addressed, and an **attachments table** that represents any files and other Message objects that are attached to it.

message part: A **message body (2)** with a string property that contains only the portion of an email message that is original to the message. It does not include any previous, quoted messages. If a message does not quote a previous message, the message part is identical to the message body.

message store: A unit of containment for a single hierarchy of Folder objects, such as a mailbox or public folders.

message transfer agent (MTA): An **SMTP** server that accepts mail from a client or another MTA and delivers the mail or relays it to another MTA.

Messaging Application Programming Interface (MAPI): A messaging architecture that enables multiple applications to interact with multiple messaging systems across a variety of hardware platforms.

messaging object: An object that exists in a **mailbox**. It can be only a **Folder object** or a **Message object**.

MetadataObject: An abstract data structure that consists of a set of **attributes (1)** that represent a LobSystem, LobSystemInstance, DataClass, Entity, Method, MethodInstance, Parameter, TypeDescriptor, Identifier, FilterDescriptor, **Action**, **ActionParameter**, or **Association**.

metafile: A file that stores an image as graphical objects, such as lines, circles, and polygons, instead of pixels. A metafile preserves an image more accurately than pixels when an image is resized.

meta-property: An entity that is identified with a **property tag** containing information (a value) that describes how to process other data in a **FastTransfer stream**.

MethodInstance: A type of **MetadataObject** that associates a normalized or stereotypical semantic with a Method that represents a native API in a line-of-business (LOB) system. MethodInstances identify which part of the data that is returned by a Method is relevant for the semantic by defining a **ReturnTypeDescriptor**. MethodInstances are contained by Methods.

MIME body: The content of a **MIME** entity, which follows the header of the MIME entity to which they both belong.

MIME content-type: A content type that is as described in [\[RFC2045\]](#), [\[RFC2046\]](#), and [\[RFC2047\]](#).

MIME Encapsulation of Aggregate HTML Documents (MHTML): A MIME-encapsulated HTML document, as described in [\[RFC2557\]](#).

MIME entity: An entity that is as described in [\[RFC2045\]](#), [\[RFC2046\]](#), and [\[RFC2047\]](#).

MIME message: A message that is as described in [\[RFC2045\]](#), [\[RFC2046\]](#), and [\[RFC2047\]](#).

MIME part: A message part that is as described in [\[RFC2045\]](#), [\[RFC2046\]](#), and [\[RFC2047\]](#).

Minimal Entry ID: A property of an **Address Book object** that can be used to uniquely identify the object.

multibyte character set (MBCS): A mixed-width **character set (1)** in which some characters are encoded into more than 1 byte. A multibyte character set (MBCS) is used in languages such as Japanese, Chinese, and Korean, where the 256 possible values of a single-byte character set are not sufficient to represent all possible characters.

Multimedia Messaging Service (MMS): A communications protocol that is designed for messages containing text, images, and other multimedia content that is sent between mobile phones.

Multipurpose Internet Mail Extensions (MIME): A set of extensions that redefines and expands support for various types of content in email messages, as described in [\[RFC2045\]](#), [\[RFC2046\]](#), and [\[RFC2047\]](#).

multivalue property: A property that can contain multiple values of the same type.

16 N

named property: A property that is identified by both a GUID and either a string name or a 32-bit identifier.

named property set: A GUID that groups related named properties into a set.

navigation shortcut: An object that contains identifying information to locate a folder in a message database or an object that groups other navigation shortcuts.

Network News Transfer Protocol (NNTP): A protocol for the distribution, inquiry, retrieval, and posting of news articles, as described in [\[RFC3977\]](#).

non-delivery report: A report message that is generated and sent by a server to the sender of a message if an email message could not be received by an intended recipient.

non-interpersonal messaging subtree: A hierarchy of folders at the root level not commonly visible in a client. This includes server and client-created folders used principally for containing operational metadata.

non-read receipt: A message that is generated when an email message is deleted at the expiration of a time limit or due to other client-specific criteria.

non-Unicode: A **character set (1)** that has a restricted set of glyphs, such as Shift_JIS or ISO-2022-JP.

normal message: A message that is not a **folder associated information (FAI)** message.

Note object: A **Message object** that represents a simple text note in a messaging store and that adheres to the property descriptions that are described in [\[MS-OXONOTE\]](#). A Note object functions as an electronic equivalent of a paper sticky note.

Notes folder: A **Folder object** that contains **Note objects**.

NTLM message: A message that carries **authentication (2)** information. Its payload data is passed to the application that supports embedded NTLM authentication by the NTLM software installed on the local computer. NTLM messages are transmitted between the client and server embedded within the application protocol that is using NTLM authentication. There are three types of NTLM messages: NTLM NEGOTIATE_MESSAGE, NTLM CHALLENGE_MESSAGE, and NTLM AUTHENTICATE_MESSAGE.

NTLM software: Software that implements the **NT LAN Manager (NTLM) Authentication Protocol**.

17 O

OAL data sequence number: An integer that is associated with **offline address list (OAL)** data that represents the generation number of this data. The value of the initial sequence number is "1". Each subsequent data generation process that produces a data set that is not identical to the previous data set is incremented by one.

Object Linking and Embedding (OLE): A technology for transferring and sharing information between applications by inserting a file or part of a file into a compound document. The inserted file can be either embedded or linked. See also **embedded object** and **linked object**.

offline: The condition of not being connected to or not being on a network or the Internet. Offline can also refer to a device, such as a printer, that is not connected to a computer, and files that are stored on a computer that is not connected to or not on a network or the Internet.

offline address book (OAB): A collection of **address lists** that are stored in a format that a client can save and use locally.

offline address book (OAB) data file: A file that contains **offline address book (OAB)** version 4-specific data, as described in [\[MS-OXOAB\]](#).

offline address list (OAL): A portion of data that is in an **offline address book (OAB)** and is related to a single **address list**.

OLE DB: A set of interfaces that are based on the Component Object Model (COM) programming model and expose data from a variety of sources. These interfaces support the amount of Database Management System (DBMS) functionality that is appropriate for a data store and they enable a data store to share data.

one-off EntryID: A special address object **EntryID** that encapsulates electronic address information, as described in [\[MS-OXCDATA\]](#).

OOF message: A message that is sent in response to incoming messages and indicates that the user is currently **Out of Office (OOF)**.

opaque-signed message: An Internet email message that is in the format described by [\[RFC5751\]](#) and uses the SignedData CMS content type described in [\[RFC3852\]](#), or the **Message object** that represents such a message.

optional attendee: An attendee of an event whom the organizer lists as an optional participant.

Organization object: An **Address Book object** that describes an entire organization.

organizer: The owner or creator of a conference or event.

orphan instance: An instance of an event that is in a **recurring series** and is in a Calendar folder without the recurring series. For all practical purposes, this is a single instance.

Out of Office (OOF): One of the possible values for the **free/busy status** on an appointment. It indicates that the user will not be in the office during the appointment.

Out of Office rule: A **rule** that is only evaluated when the mailbox is in an Out of Office state.

Outbox folder: A **special folder** that contains **Message objects** that are submitted to be sent.

18 P

Permanent Entry ID: A property of an **Address Book object** that can be used to uniquely identify the object.

permission: A rule that is associated with an object to regulate which users can gain access to the object and in what manner. See also **rights**.

Personal Distribution List object: A **Message object** that contains properties pertaining specifically to user-created **distribution lists**.

Personal Information Manager (PIM): A category of software packages for managing commonly used types of personal information, including contacts, email messages, calendar appointments, and meetings.

phishing: The luring of sensitive information, such as passwords or other personal information, from a recipient by masquerading as someone who is trustworthy and has a real need for such information.

phishing message: An email message that is designed to trick a recipient into divulging sensitive information, such as passwords or other personal information, to a non-trustworthy source.

plain text: Text that does not have markup. See also **plain text message body**.

plain text message body: A **message body (2)** for which the Content-Type value of the Email Text Body header field is "text/plain". A plain text message body can be identified explicitly in the content, or implicitly if it is in a message that is as described in [\[RFC822\]](#) or a message that does not contain a Content-Type header field.

point: A unit of measurement for fonts and spacing. A point is equal to 1/72 of an inch.

policy tip: A message that is displayed in the client user interface to inform the user that an email does not comply with an email policy configured on the server.

port: A TCP/IP numbered connection point that is used to transfer data.

Portable Network Graphics (PNG): A bitmap graphics file format that uses lossless data compression and supports variable transparency of images (alpha channels) and control of image brightness on different computers (gamma correction). PNG-format files have a .png file name extension.

Post object: A **Message object** that represents an entry in a discussion thread stored in a messaging store.

Post Office Protocol - Version 3 (POP3): A protocol that is used for accessing email from mail servers, as described in [\[RFC1939\]](#).

Predecessor Change List (PCL): A set of **change numbers** that specify the latest versions of a messaging object in all replicas that were integrated into the current version. It is used for conflict detection.

presence: (1) A status indicator on a client device that is transmitted by using the Wide Area Network Device Presence Protocol (WAN DPP).

(2) A setting for the User field that determines whether instant-messaging status information appears with user names in that field.

presentity: An entity that provides presence information to a presence service.

primary recipient: A person for whom a message is directly intended.

property ID: A 16-bit numeric identifier of a specific **attribute (1)**. A property ID does not include any **property type** information.

property list restriction table: A set of restrictions, expressed in tabular form, that is imposed on an array of properties and the values of those properties.

property name: A string that, in combination with a **property set**, identifies a **named property**.

property tag: A 32-bit value that contains a property type and a property ID. The low-order 16 bits represent the property type. The high-order 16 bits represent the property ID.

property type: A 16-bit quantity that specifies the data type of a property value.

public folder: A **Folder object** that is stored in a location that is publicly available.

publisher: A **SIP protocol client** that is making a publish request.

publishing license: An XrML 1.2 license that defines the usage policy for protected content and contains the content key with which that content is encrypted. The usage policy identifies all authorized users and the actions that they are authorized to take with the content, in addition to any usage conditions. The publishing license tells a server which usage policies apply to a specific piece of content and grants a server the right to issue Use Licenses (ULs) based on that policy. The publishing license is created when content is protected. Also referred to as "Issuance License (IL)."

pure MIME message: A **MIME** representation of an email message that does not contain a **Transport Neutral Encapsulation Format (TNEF)** body part.

20 R

read receipt: An email message that is sent to the sender of a message to indicate that a message recipient received the message.

Really Simple Syndication (RSS): An XML-based syndication format for content, as described in [\[RSS2.0\]](#).

Receive folder: A **Folder object** that is configured to be the destination for email messages that are delivered.

recipient: (1) An entity that can receive email messages.

(2) An entity that is in an **address list**, can receive email messages, and contains a set of **attributes (1)**. Each attribute has a set of associated values.

recipient information cache: An information store that contains a list of the **contacts (3)** with whom a user has interacted most often and most recently, and with whom the user is likely to interact again.

Recipient object: A set of properties that represent the recipient of a **Message object**.

recipient table: The part of a **Message object** that represents users to whom a message is addressed. Each row of the table is a set of properties that represents one **recipient (2)**.

recurrence BLOB: The **binary large object (BLOB)** encoding of a recurrence pattern, a recurrence range, and recurrence exceptions.

recurrence part: A name/value pair in a property of type Recurrence Rule, as described in [\[RFC2445\]](#).

recurrence pattern: Information for a repeating event, such as the start and end time, the number of occurrences, and how occurrences are spaced, such as daily, weekly, or monthly.

recurrence range: The range of time for which a recurrence pattern continues.

Recurring Calendar object: A **Calendar object** that describes an event that repeats according to a recurrence pattern.

recurring series: An event that repeats at specific intervals of time according to a recurrence pattern.

recurring task: A series of **Task objects** that are described by a recurrence pattern.

red-green-blue (RGB): A color model that describes color information in terms of the red (R), green (G), and blue (B) intensities in a color.

reminder: A generally user-visible notification that a specified time has been reached. A reminder is most commonly related to the beginning of a meeting or the due time of a task but it can be applied to any object type.

reminder properties: A set of properties that specify the attributes of a reminder. These attributes include the time at which and the method by which a reminder is signaled or displayed.

remote operation (ROP): An operation that is invoked against a server. Each ROP represents an action, such as delete, send, or query. A ROP is contained in a **ROP buffer** for transmission over the wire.

remote user: (1) A user who has a persistent identity within an enterprise and is connected from outside the enterprise network boundary.

(2) An **Address Book object** known to be from a foreign or remote messaging system.

replica: (1) A server that hosts an instance of a message item in a folder.

(2) A copy of the data that is in a user's **mailbox** at a specific point in time.

replica GUID (REPLGUID): A value that represents a namespace for identifiers. If a REPLGUID is combined with a GLOBSET, the result is a set of **global identifiers**. A REPLGUID value has an associated **replica ID (REPLID)** that is used in its place on disk and on the wire.

replica ID (REPLID): A value that is mapped to a **replica GUID (REPLGUID)** that identifies a namespace for IDs within a given logon. REPLIDs are used on disk and on the wire for compactness, and are replaced with the corresponding REPLGUID for external consumption.

required attendee: An attendee of an event whom the organizer lists as a mandatory participant.

Resource object: An **Address Book object** that represents an asset that can be reserved, such as a room or equipment.

response message: A Traversal Using Relay NAT (TURN) message that is sent from a protocol server to a protocol client in response to a request message. It is sent when the request message is handled successfully by the protocol server.

restriction: (1) A set of conditions that an item meets to be included in the search results that are returned by a query server in response to a search query.

(2) A filter used to map some domain into a subset of itself, by passing only those items from the domain that match the filter. Restrictions can be used to filter existing **Table objects** or to define new ones, such as **search folder (2)** or rule criteria.

retention policy: A policy that specifies the length of time during which data, documents, and other records must be available for recovery.

retention tag: An element that contains information about the **retention policy** of a **Message object** or folder.

ReturnTypeDescriptor: An **attribute (1)** of a **MethodInstance**. It is the TypeDescriptor that identifies the portion of a Method's return or output Parameters to extract and return as the result of executing the MethodInstance. It defines the View of the EntityInstances returned, with its child TypeDescriptors denoting the Fields of the View.

Rich Text Format (RTF): Text with formatting as described in [\[MSFT-RTF\]](#).

rights: Tasks that a user is permitted to perform on a computer, site, domain, or other system resource. See also **permission**.

rights policy template: An XrML 1.2 document that contains a predefined usage policy that is used to create a **publishing license** when content is protected.

rights-managed email message: An email message that specifies permissions that are designed to protect its content from inappropriate access, use, and distribution.

Root folder: The **special folder** that is the top-level folder in a message store hierarchy. It contains all other **Folder objects** in that message store.

ROP buffer: A structure containing an array of bytes that encode a **remote operation (ROP)**. The first byte in the buffer identifies the ROP. This byte is followed by ROP-specific fields. Multiple ROP buffers can be packed into a single **remote procedure call (RPC)** request or response.

ROP request: See **ROP request buffer**.

ROP request buffer: A **ROP buffer** that a client sends to a server to be processed.

ROP response: See **ROP response buffer**.

ROP response buffer: A **ROP buffer** that a server sends to a client to be processed.

RSS item: An item element in an RSS feed, as described in [\[RSS2.0\]](#).

RSS object: A Message object that represents an entry from an **RSS item** or **atom feed**.

rule: (1) A condition or action, or a set of conditions or actions, that performs tasks automatically based on events and values.

(2) A set of qualifiers, such as enumeration values, and quantifiers, such as numeric arguments, that are specified as usage guidelines for a set of objects or data.

(3) A mapping of a file type to a location in a document repository.

(4) An item that defines a **condition** and an action. The condition is evaluated for each **Message object** as it is delivered, and the action is executed if the new Message object matches the condition.

Rule FAI message: A **folder associated information (FAI)** message stored in the Inbox special folder where the client can store extra rule-related information that is opaque to the server.

rules table: A **Table object** whose rows represent the rules that are contained in a **Folder object**.

21 S

S/MIME (Secure/Multipurpose Internet Mail Extensions): A set of cryptographic security services, as described in [\[RFC5751\]](#).

search criteria: A criteria used to determine which messages are included in a folder with specific characteristics. It is composed of a restriction, which is the filter to be applied, and a search scope, which are the folders that contain the content to search.

search folder: (1) A collection of related items to be crawled by a search service.

(2) A **Folder object** that provides a means of querying for items that match certain criteria. The search folder includes the **search folder definition message** and the **search folder container**.

search folder container: A **Folder object** that is created according to the specifications in the definition message. It is in the Finder folder of the message database.

search folder definition message: A **folder associated information (FAI)** message that persists all the information that defines a search folder. It is in the associated contents table of the **Common Views folder** in the message database.

search key: A binary-comparable key that identifies related objects for a search.

security descriptor: A data structure that contains the security information associated with a securable object. A security descriptor uses a security identifier (SID) to identify the owner of an object.

security principal: (1) A unique entity that is identifiable through cryptographic means by at least one key. It frequently corresponds to a human user, but also can be a service that offers a resource to other security principals. Also referred to as principal.

(2) An identity that can be used to regulate access to resources. A security principal can be a user, a computer, or a group that represents a set of users.

security principal identifier: A value that is used to uniquely identify a **security principal (2)**. In Windows-based systems, it is a **security identifier (SID)**. In other types of systems, it can be a user identifier or other type of information that is associated with a **security principal (2)**.

security provider: (1) A Component Object Model (COM) object that provides methods that return custom information about the security of a site.

(2) A pluggable security module that is specified by the protocol layer above the **remote procedure call (RPC)** layer, and will cause the RPC layer to use this module to secure messages in a communication session with the server. The security provider is sometimes referred to as an authentication service.

security token service (STS): A web service that issues **claims (2)** and packages them in encrypted security tokens.

Sent Items folder: A **special folder** that is the default location for storing copies of **Message objects** after they are submitted or sent.

sequence number: (1) A numeric value that is used to define the order in which a series of events occurs in an execution sequence or transaction.

(2) The revision number of a **Meeting object**. The sequence number is used to determine the most recent **meeting update** that was sent by the organizer.

Server object: An object on a server that is used as input or created as output for **remote operations (ROPs)**.

Server object handle: A 32-bit value that identifies a **Server object**.

Server object handle table: An array of 32-bit handles that are used to identify input and output **Server objects** for **ROP requests** and **ROP responses**.

server replica: A copy of a user's **mailbox** that exists on a server.

server-side rule: A **rule** for which all actions are executed by a server.

service connection point: An object that is made available by a directory service and that clients can use to discover **Autodiscover servers**.

Session Context: A server-side partitioning for client isolation. All client actions against a server are scoped to a specific Session Context. All **messaging objects** and data that is opened by a client are isolated to a Session Context.

session context handle: A **remote procedure call (RPC)** context handle that is used by a client when issuing RPCs against a server on EMSMDB interface methods. It represents a handle to a unique session context on the server.

Session Initiation Protocol (SIP): A protocol that is used to establish multimedia communication sessions between two or more users over an IP network, as described in [\[RFC3261\]](#).

shared folder: A folder for which a sharing relationship has been created to share items in the folder between two servers.

Short Message Service (SMS): A communications protocol that is designed for sending text messages between mobile phones.

signal time: The time at which a **reminder** has been specified to notify the user or an agent acting on behalf of the user. For example, the signal time for a meeting that starts at 11:00 A.M. can be 10:45 A.M., thus allowing the user 15 minutes to prepare for or travel to the meeting upon receiving the notification.

Simple Mail Transfer Protocol (SMTP): A member of the TCP/IP suite of protocols that is used to transport Internet messages, as described in [\[RFC5321\]](#).

single-instance object: An **Appointment object**, **Meeting object**, or **Task object** that occurs only once.

SIP protocol client: A network client that sends **Session Initiation Protocol (SIP)** requests and receives SIP responses. An SIP client does not necessarily interact directly with a human user. **User agent clients (UACs)** and proxies are SIP clients.

site: (1) A group of related webpages that is hosted by a server on the World Wide Web or an intranet. Each website has its own entry points, metadata, administration settings, and workflows. Also referred to as web site.

(2) A group of related pages and data within a SharePoint site collection. The structure and content of a site is based on a site definition. Also referred to as SharePoint site and web site.

site mailbox: A repository comprised of a mailbox and a web-based collaboration environment that is presented to users as a mailbox in an email client. A site mailbox uses team membership to determine which users have access to the repository.

skip block: The block in a **binary large object (BLOB)** that acts as padding, reserving space that can be used by future versions to insert data. The block consists of a ULONG that describes how many additional ULONGs to skip ahead.

SOAP envelope: The outermost element of a **SOAP message**, as described in [\[SOAP1.1\]](#).

soft delete: A process that removes an item from the system, but not permanently. If an item is soft deleted, a server retains a back-up copy of the item and a client can access, restore, or permanently delete the item. See also **hard delete**.

sort order: (1) A set of rules in a search query that defines the order of relevant results. Each rule consists of a managed property, such as modified date or size, and a direction for order, such as ascending or descending. Multiple rules are applied sequentially.

(2) A specific arrangement of cells that is based on cell content. The order can be ascending or descending.

(3) The order in which the rows in a **Table object** are requested to appear. This can involve sorting on multiple properties and sorting of **categories (5)**.

spam: An unsolicited email message.

spam filter: A filter that checks certain conditions in a message to determine a spam confidence level.

special folder: One of a default set of **Folder objects** that can be used by an implementation to store and retrieve user data objects.

storage: An element of a compound file that is a unit of containment for one or more storages and streams, analogous to directories in a file system, as described in [\[MS-CFB\]](#).

Store object: An object that is used to store **mailboxes** and **public folder** content.

stream: (1) An element of a compound file, as described in [\[MS-CFB\]](#). A stream contains a sequence of bytes that can be read from or written to by an application, and they can exist only in storages.

(2) A flow of data from one host to another host, or the data that flows between two hosts.

Stream object: A **Server object** that is used to read and write large string and binary properties.

string property: A property whose property type is PtypString8 or PtypString.

structured document: A document that is internally composed of multiple **streams (1)** that specify data for individual pieces of the document, such as style information, images, or embedded objects. The streams (1) allow pieces of the document to be addressed and manipulated individually.

subject: For a folder, the messages and subfolders that are contained in that folder. For a message, the **recipients (2)** and attachments to that message. For an attachment, the **Embedded Message object** for that attachment.

synchronization context: See **synchronization download context** or **synchronization upload context**.

synchronization download context: A **Server object** that represents a context for an **ICS** download.

synchronization scope: A set of complex criterion that defines a superset of all the **messaging objects** that are within a specific mailbox and are considered for a single synchronization operation.

synchronization type: The type of synchronization that is occurring, either a **hierarchy synchronization** or a **content synchronization**.

synchronization upload context: A **Server object** that represents a context for an **ICS** upload.

22 T

Table object: An object that is used to view properties for a collection of objects of a specific type, such as a **Message object** or a **Folder object**. A Table object is structured in a row and column format with each row representing an object and each column representing a property of the object.

tagged property: A property that is defined by a 16-bit property ID and a 16-bit property type. The property ID for a tagged property is in the range 0x001 – 0x7FFF. Property IDs in the range 0x8000 – 0x8FFF are reserved for assignment to **named properties**.

Task object: A **Message object** that represents an assignment to be completed.

task request: A **Message object** that is used to issue a task assignment.

task response: A response to a task from a target. A task can have zero or more responses.

Tasks folder: A **Folder object** that contains **Task objects**.

telecommunications device for the deaf (TDD): A device that enables the transmission of typed messages over phone lines. These devices typically include keyboards for typing messages to send and printers to receive messages from one device to another.

tentative: One of the possible values for the **free/busy status** on an appointment. A tentative status indicates that the user is tentatively booked during the appointment.

term: A concept or an idea that is stored and can be used as metadata.

term set: A collection of terms that are arranged into and stored as a hierarchy or a flat list.

term store: A database in which managed metadata is stored in the form of **term sets** and **terms**.

TIFF: A high-resolution, tag-based graphics format. TIFF is used for the universal interchange of digital graphics.

time flag: A flag that extends the concept of a **basic flag** by associating time-related properties, such as start and due dates, with the flag information on a **Message object**. A time flagged Message object is also marked with a red **color flag**, but it is not considered to be color flagged by definition.

To recipient: See **primary recipient**.

token: A word in an item or a search query that translates into a meaningful word or number in written text. A token is the smallest textual unit that can be matched in a search query. Examples include "cat", "AB14", or "42".

top-level message: A message that is not included in another message as an **Embedded Message object**. Top-level messages are **messaging objects**.

transaction: (1) An object that stores the state and metadata for an item during a crawl.

(2) A single unit of work. If a transaction is successful, all data modifications that were made during the transaction are committed and become a permanent part of the database. If a transaction encounters an error and is canceled or rolled back, all data modifications are erased.

(3) The process of opening or creating an object on a server, and the subsequent committing of changes to the object by calling the required save function, at which time all changes to that instance of the object are either saved to the server, or discarded if a failure occurs before saving is finished successfully. Until successfully saved, changes are invisible to any other instances of the object.

Transport Layer Security (TLS): A protocol that provides a method to help client and server applications communicate more securely, as described in [\[RFC4346\]](#).

Transport Neutral Encapsulation Format (TNEF): A binary type-length-value encoding that is used to encode properties for transport, as described in [\[MS-OXTNEF\]](#).

Transport Neutral Encapsulation Format (TNEF) message: A **MIME** representation of an email message in which attachments and some message properties are carried in a **Transport Neutral Encapsulation Format (TNEF)** body part.

Transport Neutral Encapsulation Format (TNEF) Reader: An entity that decodes a **Transport Neutral Encapsulation Format (TNEF)** structure after receiving a message, for the purpose of reconstructing the rich properties that are contained in the stream.

Transport Neutral Encapsulation Format (TNEF) Writer: An entity that encodes or builds a **Transport Neutral Encapsulation Format (TNEF)** structure for the purpose of transporting rich properties.

23 U

Unified Messaging: A set of components and services that enable voice, fax, and email messages to be stored in a user's **mailbox** and accessed from a variety of devices.

Uniform Resource Identifier (URI): A sequence of characters that identifies an abstract or physical resource, as described in [\[RFC3986\]](#).

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the Internet.

unsendable attendee: An attendee to whom a **meeting request** or **meeting update** is not sent.

Use License: An XrML 1.2 license that authorizes a user to gain access to a protected content file and describes the applicable usage policies. Also referred to as "End-User License (EUL)."

user agent client (UAC): A logical entity that creates a new request, and then uses the client transaction state machinery to send it. The role of UAC lasts only for the duration of that transaction. If a process initiates a request, it acts as a UAC for that transaction. If a process receives a request later, it assumes the role of a **user agent server (UAS)** for that transaction.

user agent server (UAS): A logical entity that generates a response to a **Session Initiation Protocol (SIP)** request. The response either accepts, rejects, or redirects the request. The role of the UAS lasts only for the duration of that transaction. If a process responds to a request, it acts as a UAS for that transaction. If it initiates a request later, it assumes the role of a **user agent client (UAC)** for that transaction.

UTF-16LE: The Unicode Transformation Format - 16-bit, Little Endian encoding scheme. It is used to encode **Unicode** characters as a sequence of 16-bit codes, each encoded as two 8-bit bytes with the least-significant byte first.

UUEncoded attachment: A file that is attached to an email message that was encoded by using the uuencode utility, as described in [\[IEEE1003.1\]](#).

24 V

vCard: A format for storing and exchanging electronic business cards, as described in [\[RFC2426\]](#).

voice message: A **Message object** that contains audio content recorded by a calling party.

25 W

WBXML code page: A map used to convert XML to Wireless Application Protocol (WAP) Binary XML (WBXML) that specifies the one-to-one relationships between the XML tags of a namespace and the tags' numeric representations called WBXML tokens. Each WBXML code page corresponds to an XML namespace.

Web Distributed Authoring and Versioning Protocol (WebDAV): The Web Distributed Authoring and Versioning Protocol, as described in [\[RFC2518\]](#) or [\[RFC4918\]](#).

Web Distribution Point (WDP): A location on a server where **offline address book (OAB)** files are published for web distribution. A client can discover the URI of a WDP by using the Autodiscover Publishing and Lookup Protocol, as described in [\[MS-OXDSCLI\]](#).

web server: A server computer that hosts websites and responds to requests from applications.

web service: A unit of application logic that provides data and services to other applications and can be called by using standard Internet transport protocols such as **HTTP**, **Simple Mail Transfer Protocol (SMTP)**, or **File Transfer Protocol (FTP)**. Web services can perform functions that range from simple requests to complicated business processes.

Web Services Description Language (WSDL): An extensible **XML** format that describes network services as collections of communication **endpoints (4)**, as described in [\[WSDL\]](#).

WebDAV client: A computer that uses **WebDAV**, as described in [\[RFC2518\]](#) or [\[RFC4918\]](#), to retrieve data from a **WebDAV server**.

WebDAV server: A computer that supports **WebDAV**, as described in [\[RFC2518\]](#) or [\[RFC4918\]](#), and responds to requests from **WebDAV clients**.

Windows Metafile Format (WMF): A vector graphics format for Windows®-compatible computers. Windows Metafile Format is used primarily as a clip-art format in word-processing documents.

Wireless Application Protocol (WAP) Binary XML (WBXML): A compact binary representation of **XML** that is designed to reduce the transmission size of XML documents over narrowband communication channels.

working hours: Times of the day that are valid for meetings to be considered for an attendee.

WSDL message: An abstract, typed definition of the data that is communicated during a **WSDL operation**, as described in [\[WSDL\]](#).

WSDL operation: An abstract description of an action that is supported by a WSDL service, as described in [\[WSDL\]](#).

WSDL port type: A named set of abstract operations that is supported by one or more **endpoints (4)**, as described in [\[WSDL\]](#).

26 X

X500 DN: A distinguished name (DN), in Teletex form, of an object that is in an **address book**. An X500 DN can be more limited in the size and number of relative distinguished names (RDNs) than a full DN.

XML document: A document object that is well formed, as described in [\[XML\]](#), and might be valid. An XML document has a logical structure that is composed of declarations, elements, comments, character references, and processing instructions. It also has a physical structure that is composed of entities, starting with the root, or document, entity.

XML element: An **XML** structure that typically consists of a start tag, an end tag, and the information between those tags. Elements can have **attributes (1)** and can contain other elements.

XML namespace prefix: An abbreviated form of an **XML namespace**, as described in [\[XML\]](#).

XML schema: A description of a type of **XML document** that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by **XML** itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

XML schema definition (XSD): The World Wide Web Consortium (W3C) standard language that is used in defining XML schemas. Schemas are useful for enforcing structure and constraining the types of data that can be used validly within other XML documents. XML schema definition refers to the fully specified and currently recommended standard for use in authoring **XML schemas**.

XMLHttpRequest (XHR): A software component that is used by browser-based scripts to transfer data between a web browser and a web server.

29 Change Tracking

This section identifies changes that were made to the [MS-OXGLOS] protocol document between the November 2013 and February 2014 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- The removal of a document from the documentation set.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.
- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.

- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated**.

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact dochelp@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
G	Added a second definition to the term "ghosted".	N	Content updated.