

Endeca® Latitude
Glossary
Version 2.2.2 • December 2011





Copyright and disclaimer

Product specifications are subject to change without notice and do not represent a commitment on the part of Endeca Technologies, Inc. The software described in this document is furnished under a license agreement. The software may not be reverse engineered, decompiled, or otherwise manipulated for purposes of obtaining the source code. The software may be used or copied only in accordance with the terms of the license agreement. It is against the law to copy the software on any medium except as specifically allowed in the license agreement.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of Endeca Technologies, Inc.

Copyright © 2003-2011 Endeca Technologies, Inc. All rights reserved. Printed in USA.

Portions of this document and the software are subject to third-party rights, including:

Corda PopChart® and Corda Builder™ Copyright © 1996-2005 Corda Technologies, Inc.

Outside In® Search Export Copyright © 2011 Oracle. All rights reserved.

Rosette® Linguistics Platform Copyright © 2000-2011 Basis Technology Corp. All rights reserved.

Teragram Language Identification Software Copyright © 1997-2005 Teragram Corporation. All rights reserved.

Trademarks

Endeca, the Endeca logo, Guided Navigation, MDEX Engine, Find/Analyze/Understand, Guided Summarization, Every Day Discovery, Find Analyze and Understand Information in Ways Never Before Possible, Endeca Latitude, Endeca InFront, Endeca Profind, Endeca Navigation Engine, Don't Stop at Search, and other Endeca product names referenced herein are registered trademarks or trademarks of Endeca Technologies, Inc. in the United States and other jurisdictions. All other product names, company names, marks, logos, and symbols are trademarks of their respective owners.

The software may be covered by one or more of the following patents: US Patent 7035864, US Patent 7062483, US Patent 7325201, US Patent 7428528, US Patent 7567957, US Patent 7617184, US Patent 7856454, US Patent 7912823, US Patent 8005643, US Patent 8019752, US Patent 8024327, US Patent 8051073, US Patent 8051084, Australian Standard Patent 2001268095, Republic of Korea Patent 0797232, Chinese Patent for Invention CN10461159C, Hong Kong Patent HK1072114, European Patent EP1459206, European Patent EP1502205B1, and other patents pending.

A

Related Links

- [Administration Web Service](#) on page 5
- [assignment](#) on page 5
- [attribute](#) on page 5
- [attribute group](#) on page 6
- [attribute type](#) on page 6
- [attribute value](#) on page 6

Administration Web Service

Used by IT engineers and administrators to integrate the MDEX Engine server and its reporting with third-party IT tools.

See also Endeca Web services.

Related Links

- [Endeca Web services](#) on page 14

assignment

When a record has a value for an attribute, it is referred to as an assignment.

attribute

Basic unit of a record schema. Attributes describe records in the MDEX Engine.

For data records, attributes provide information about a record. For system records, an attribute is a configuration setting.

The term *attribute* collectively refers to both standard attributes and managed attributes.

- Standard attributes are described by attribute schema records. The attribute schema records that describe standard attributes are known as Property Description Records (PDRs).
- Managed attributes are also described by attribute schema records. The attribute schema records that describe managed attributes are known as Property Description Records (PDRs) and Dimension Description Records (DDR).

See also attribute schema record, Property Description Record, Dimension Description Record, standard attribute, and managed attribute.

Related Links

- [primary key attribute](#) on page 29
- [single-assign attribute](#) on page 33

multi-assign attribute on page 25

unique attribute on page 39

standard attribute on page 34

managed attribute on page 25

attribute group

An group of attributes defined by the administrator or power user. Attributes are displayed in the context of their groups in Latitude Studio. All attributes that are not members of user-defined groups automatically belong to a group named `other`. Additional groups can be created using Latitude Studio.

attribute type

The required format for an attribute value (such as string, integer, boolean, date/time).

attribute value

An assignment from an attribute on a record, used as a tag, or label, to classify a record in your data set. Tagging a record with a value identifies that record as a valid result when a user queries for the value. A record can have more than one assignment from a specific attribute—such record is known as multi-assign.

The term *attribute value* applies to values on both standard and managed attributes.

B

Related Links

[Bulk Load interface](#) on page 7

Bulk Load interface

An MDEX Engine interface that is intended to achieve high performance for strictly additive ingests of varying amounts of data. An Integrator connector is available for this interface.

Related Links

[Data Ingest Web Service \(DIWS\)](#) on page 11

C

Related Links

- [component](#) on page 9
- [cluster](#) on page 9
- [configuration update](#) on page 9
- [Configuration Web Service](#) on page 9
- [Conversation Web Service](#) on page 10

component

See Latitude Studio standard component or Integrator component.

Related Links

- [Latitude Studio standard component](#) on page 24
- [Integrator component](#) on page 20

cluster

A cluster is composed of a set of MDEX Engine nodes each of which can serve query requests. Only one node is identified as the leader node; All other nodes are follower nodes. There is one copy of the MDEX Engine index that is shared and used by all MDEX Engine nodes. Nodes can be added or removed dynamically, without having to stop the cluster.

The Cluster Coordinator provides communication between the nodes in the cluster and notifies the nodes about index updates and updates to the configuration.

See also leader node and follower node.

Related Links

- [leader node](#) on page 24
- [follower node](#) on page 15

configuration update

The process of loading changes made to the configuration of the MDEX Engine index.

Configuration Web Service

Web service that allows you to update the schema and configuration.

See also Endeca Web services.

Related Links

[Endeca Web services](#) on page 14

Conversation Web Service

Provides the primary means of querying data in the MDEX Engine. Used by Latitude Studio to query the MDEX Engine.

See also Endeca Web services.

Related Links

[Endeca Web services](#) on page 14

[Latitude Studio](#) on page 24

D

Related Links

- [Data Ingest Web Service \(DIWS\)](#) on page 11
- [data record](#) on page 11
- [Dimension Description Record \(DDR\)](#) on page 11
- [Discovery Framework](#) on page 12

Data Ingest Web Service (DIWS)

A Web service that provides an interface to ETL tools to load data into the MDEX Engine. Besides adding new records to the MDEX Engine, the data ingest operations include modifying (updating) and deleting existing records in the MDEX Engine data set.

See also Endeca Web services.

Related Links

- [Bulk Load interface](#) on page 7
- [Endeca Web services](#) on page 14

data record

Data records, also known as Endeca records, represent the actual data that is being analyzed and manipulated using Latitude Studio. These records are the individual items that the user navigates to in an Endeca Latitude application.

Data records generally correspond to traditional records in a source database. Unlike source records, however, data records have been standardized for consistency, and then classified with attribute values.

See also record, primordial record, system record, and source record.

Related Links

- [record](#) on page 31
- [primordial record](#) on page 30
- [source record](#) on page 34

Dimension Description Record (DDR)

A system record used to define the behavior of a managed attribute. Each managed attribute has a DDR. The DDR configuration includes rules for displaying the managed attribute and using it in searches.

See also system record, Property Description Record (PDR), and Global Configuration Record (GCR).

Discovery Framework

Name used for Latitude Studio in versions of Latitude prior to 2.1.

See Latitude Studio.

Related Links

[Latitude Studio](#) on page 24

E

Related Links

- [edge](#) on page 13
- [end user \(Latitude Studio\)](#) on page 13
- [Endeca Latitude](#) on page 13
- [Endeca MDEX Engine](#) on page 14
- [Endeca Web services](#) on page 14

edge

In a Latitude Data Integrator graph, edges represent data flowing from one component to another.

Each edge is characterized by metadata that you can assign to it. Metadata specifies information about which fields of data are being provided from one component to the next.

Since data produced by one component needs to be received by another component through the edge, the metadata on the edge describe the fields of data that are involved in this operation.

Related Links

- [graph](#) on page 17
- [project](#) on page 30
- [Latitude Data Integrator](#) on page 23
- [Integrator component](#) on page 20

end user (Latitude Studio)

End users are Latitude Studio content consumers. This includes executives seeking a dashboard view as well as others who need to drill through interactive visualizations and reports. Typically, Latitude Studio is configured so that end users cannot access the edit controls found on the edit view of each component.

See also Latitude Studio and power user.

Related Links

- [Latitude Studio](#) on page 24
- [power user \(Latitude Studio\)](#) on page 29

Endeca Latitude

Endeca Latitude applications guide people to better decisions by combining the ease of search with the analytic power of business intelligence. Users get self-service access to the data they need without having to specify in advance the queries or views they want. At the same time, the user experience is data driven, continuously revealing the salient relationships in the underlying data for them to explore.

Endeca MDEX Engine

The query engine that is the core of the Endeca Latitude. The MDEX Engine maintains the indices of your records in memory, receives queries, executes them against the stored indices, and returns the results. The MDEX Engine is an online process that must remain running as long as you want clients to be able to access data.

See also Endeca Web services.

Related Links

[Endeca Web services](#) on page 14

[index](#) on page 19

Endeca Web services

Include the Data Ingest Web Service, the Conversation Web Service, the Configuration Web Service, and the Administration Web Service.

Together, they provide an API interface to an Endeca implementation.

Related Links

[Data Ingest Web Service \(DIWS\)](#) on page 11

[Conversation Web Service](#) on page 10

[Configuration Web Service](#) on page 9

[Administration Web Service](#) on page 5

F

Related Links

- [follower node](#) on page 15
- [full index load](#) on page 15
- [full index refresh](#) on page 15

follower node

A node in a cluster of MDEX Engine nodes responsible for processing queries. The follower node does not update the index although it has read-only access to its latest copy. Each cluster can have more than one follower nodes. In a single-node cluster, a leader node is also a follower node. Each follower node must have a unique name across the cluster. All nodes (including follower nodes) must have write access to a shared file system.

See also leader node and cluster.

Related Links

- [cluster](#) on page 9
- [leader node](#) on page 24

full index load

A load of the full index, either as an initial load or a complete reload. This includes all user-defined data.

Related Links

- [incremental index update](#) on page 19
- [full index refresh](#) on page 15
- [index](#) on page 19

full index refresh

The process of reloading the full data set into the MDEX Engine. This entails cleaning out the old data and replacing it with the new data. In SQL terminology, full index refresh is equivalent to the "Truncate and load" operation on a table in the database.

Unlike in a full index load, in a full index refresh the MDEX Engine retains some user-defined data.

Related Links

- [incremental index update](#) on page 19
- [full index load](#) on page 15
- [index](#) on page 19

G

Related Links

- [Global Configuration Record \(GCR\)](#) on page 17
- [graph](#) on page 17
- [Guided Navigation](#) on page 17

Global Configuration Record (GCR)

A single system record used to define global configuration information. The GCR includes rules for searches and spelling correction.

See also system record, Property Description Record (PDR), and Dimension Description Record (DDR).

graph

In the Latitude Data Integrator, a graphical layout that contains a set of integrator components.

Related Links

- [edge](#) on page 13
- [project](#) on page 30
- [Latitude Data Integrator](#) on page 23
- [Integrator component](#) on page 20

Guided Navigation

The presentation of valid follow-on refinement choices to the user, implemented using the Guided Navigation and Breadcrumbs components in Latitude Studio.

See also Latitude Studio and component.

Related Links

- [incremental index update](#) on page 19
- [index](#) on page 19
- [index configuration documents](#) on page 19
- [Integrator component](#) on page 20

incremental index update

The process of loading changes made to the data records, which updates the index in the MDEX Engine.

Related Links

- [full index load](#) on page 15
- [full index refresh](#) on page 15
- [index](#) on page 19

index

A collective term that refers to the many types of indices in the MDEX Engine. For example, in the standard search index, each entry corresponds to a searchable document containing the correct term; in the wildcard search index, each entry corresponds to a document enabled for wildcard search that contains the correct term. Other indices are also used.

An empty MDEX Engine index is created when you install the MDEX Engine, run the `mkmdex` command, and start the MDEX Engine. When you make a change to your data records or configuration, depending on the change, this causes the index to be updated, either through an incremental index update, a configuration update, or a full index refresh.

Related Links

- [incremental index update](#) on page 19
- [full index load](#) on page 15
- [full index refresh](#) on page 15

index configuration documents

A set of XML-based configuration files that define how your Endeca records, Endeca standard attributes, and Endeca managed attributes are indexed by the MDEX Engine. The index configuration is the mechanism for implementing a number of Endeca features such as record search, value search, snippeting, relevance ranking, precedence rules, and thesaurus entries.

Integrator component

In the Latitude Data Integrator, a graphical object that performs some kind of data manipulation and that you add to your graph.

Components are characterized by several types, with various components belonging to each type. Some of the component types are readers, writers, transformers, and joiners.

Each component has an input and output port. Even though components are joined sequentially in a graph, all components that are in the same phase of the graph run in parallel.

Related Links

[edge](#) on page 13

[graph](#) on page 17

[project](#) on page 30

[Latitude Data Integrator](#) on page 23

K

Related Links

[key-value pair](#) on page 21

key-value pair

Assignments on standard attributes use key-value pairs, or KVPs, where *key* is the name of an attribute, and *value* is an assigned value for this attribute.

See also standard attribute and standard attribute value.

L

Related Links

- [Latitude Data Integrator](#) on page 23
- [Latitude Data Integrator component](#) on page 23
- [Latitude Information Integration Suite](#) on page 23
- [Latitude Query Language](#) on page 24
- [Latitude Studio](#) on page 24
- [Latitude Studio standard component](#) on page 24
- [leader node](#) on page 24

Latitude Data Integrator

Latitude Data Integrator (or LDI) is a high-performance data integration platform that lets you extract source records from a variety of source types (from flat files to databases) and send that data to the Data Ingest Web Service, which in turn loads the records into the MDEX Engine.

Related Links

- [edge](#) on page 13
- [graph](#) on page 17
- [project](#) on page 30
- [Integrator component](#) on page 20
- [Latitude Information Integration Suite](#) on page 23

Latitude Data Integrator component

See Integrator component.

Related Links

- [Integrator component](#) on page 20

Latitude Information Integration Suite

The suite of products that enable the process of loading various types of source data into the Endeca Latitude. After the data is consumed, it is indexed by the MDEX Engine and made available in the Latitude Studio for exploration, navigation, and further configuration to improve the ability to find results.

See also Latitude Data Integrator.

Related Links

- [Latitude Data Integrator](#) on page 23

Latitude Query Language

The Latitude Query Language, or LQL, enables interactive applications that allow users to explore aggregate and statistical views of information using LQL-enabled Latitude Studio components.

Latitude Studio

A customizable, component-based portal application, built on the Liferay Portal, that offers an interactive Guided Navigation user experience across a range of structured and unstructured enterprise data. Granular layout and configuration control enable users to manage and personalize their own experiences within the portal.

Related Links

[end user \(Latitude Studio\)](#) on page 13

[power user \(Latitude Studio\)](#) on page 29

Latitude Studio standard component

A Latitude Studio portlet created and provided by Endeca, containing logic needed to retrieve and manipulate data from the MDEX Engine and render results. Each Latitude Studio standard component provides specific MDEX Engine features or other application support.

leader node

A single node in a cluster of MDEX Engine nodes responsible for processing queries and for receiving updates to the index and to the configuration. This node is responsible for obtaining information about the latest index and propagating this information to the follower nodes through the Cluster Coordinator.

Each cluster must have one and only one leader node. All nodes must have write access to a shared file system on which the MDEX Engine index is stored. The modules outside the cluster of MDEX Engine nodes, (such as connectors in the Latitude Data Integrator and components in Latitude Studio), must have access to the leader node.

See also follower node and cluster.

Related Links

[cluster](#) on page 9

[follower node](#) on page 15

M

Related Links

- [managed attribute](#) on page 25
- [managed attribute value](#) on page 25
- [multi-assign attribute](#) on page 25

managed attribute

An attribute for which a hierarchy of attribute values is attached. Managed attributes are used to support hierarchical navigation. For example, when using a Location attribute to filter records, users may navigate by North America > United States > California.

Managed attributes are described by schema records — Property Dimension Records and Dimension Description Records. An assignment from managed attribute is known as a managed attribute value (or mval).

See also attribute and standard attribute.

Related Links

- [attribute](#) on page 5
- [primary key attribute](#) on page 29
- [single-assign attribute](#) on page 33
- [multi-assign attribute](#) on page 25
- [unique attribute](#) on page 39
- [standard attribute](#) on page 34

managed attribute value

Values of a specific attribute organized in an enumerated list or a hierarchy of values. Often, managed attribute values are sources from an external system. Managed attribute values can be managed in Latitude Studio.

multi-assign attribute

An attribute for which a record may have more than one value. For example, because a book may have more than one author, the Author attribute would be multi-assign.

Related Links

- [attribute](#) on page 5
- [primary key attribute](#) on page 29
- [single-assign attribute](#) on page 33
- [unique attribute](#) on page 39

[standard attribute](#) on page 34
[managed attribute](#) on page 25

N

Related Links

[navigation query](#) on page 27

navigation query

Used to return a list of matching records based on the current navigation or search.

P

Related Links

- [power user \(Latitude Studio\)](#) on page 29
- [precedence rule](#) on page 29
- [primary key attribute](#) on page 29
- [primordial record](#) on page 30
- [project](#) on page 30
- [Property Description Record \(PDR\)](#) on page 30

power user (Latitude Studio)

Power users configure Latitude Studio content. One example would be a business analyst who configures Latitude Studio for end users and determines what components and data they can access.

See also Latitude Studio and end users (Latitude Studio).

Related Links

- [Latitude Studio](#) on page 24
- [end user \(Latitude Studio\)](#) on page 13

precedence rule

A relationship between two attributes that establishes a navigation or display preference based on a set of predefined criteria, known as triggers for the rule.

primary key attribute

An attribute used to uniquely identify a record. A primary key attribute must be both unique and single-assign. For example, for a book, the ISBN number could be a primary key attribute, because each book has only one unique value.

See also unique attribute and single-assign attribute.

Related Links

- [attribute](#) on page 5
- [single-assign attribute](#) on page 33
- [multi-assign attribute](#) on page 25
- [unique attribute](#) on page 39
- [standard attribute](#) on page 34
- [managed attribute](#) on page 25

primordial record

The most basic infrastructure of an MDEX Engine. Primordial records are created automatically and used as the basis for the system records.

See also record, system record, and data record.

Related Links

[data record](#) on page 11

[record](#) on page 31

[source record](#) on page 34

project

In Latitude Data Integrator, the location where you create operations to manipulate your data. You can put one or more graphs into a single project.

Related Links

[edge](#) on page 13

[graph](#) on page 17

[Latitude Data Integrator](#) on page 23

[Integrator component](#) on page 20

Property Description Record (PDR)

A system record used to define the format and behavior of a single attribute. Each attribute has a PDR. The PDR configuration includes rules for uniqueness, searches, and navigation.

See also system record, Dimension Description Record (DDR), and Global Configuration Record (GCR).

R

Related Links

- [record](#) on page 31
- [record search](#) on page 31
- [record query](#) on page 31
- [refinement](#) on page 32
- [relevance ranking](#) on page 32

record

The fundamental unit of data in an MDEX Engine. Records are assigned attribute values. An assignment indicates that a record has a value for an attribute. A record typically has assignments for multiple attributes. For each assigned attribute, the record may have one or more values.

A record may be a primordial record, a data record, or a system record.

See also assignment, attribute, primordial record, data record, and system record.

Related Links

- [data record](#) on page 11
- [primordial record](#) on page 30
- [source record](#) on page 34

record search

A query that returns results based on a user-specified text string by filtering the record set to include only those Endeca records that have at least one attribute whose value matches a specified search term (keyword). The result of a record search is a set of records based on the user-defined keyword(s), plus any follow-on query information.

See also value search.

Related Links

- [value search](#) on page 41

record query

Used to return the details for a single record.

refinement

An attribute whose values may be used to reduce or refine the current query's record set.

relevance ranking

A search interface feature that lets the developer control the order in which record or value search results are displayed to the end user. A relevance ranking module assigns ranking scores to results based on its predetermined criteria, such as the frequency of a user's query terms in the result text. Modules can be combined to produce a complex ranking strategy for a search interface.

S

Related Links

- [schema for records](#) on page 33
- [search interface](#) on page 33
- [single-assign attribute](#) on page 33
- [snippeting](#) on page 34
- [source record](#) on page 34
- [standard attribute](#) on page 34
- [standard attribute value](#) on page 35
- [system record](#) on page 35

schema for records

A set of metadata that describes the data model for your records. During the data modeling process, a data architect for the application powered by Endeca Latitude defines the schema for records. For example, the schema defines which of your attributes are searchable. It also defines display names for the attributes on your records, and other characteristics.

A schema for your data records is itself represented by records. However, unlike data records which have attributes describing your data, schema records have attributes that describe the schema.

See also Property Description Record (PDR), and Dimension Description Record (DDR).

Related Links

- [Property Description Record \(PDR\)](#) on page 30
- [Dimension Description Record \(DDR\)](#) on page 11

search interface

A named collection of attributes, each of which is enabled for record search. The search interface may include features that control the search behavior, such as relevance ranking modules and partial match. Search interfaces thus allow end users to search multiple attributes simultaneously.

single-assign attribute

An attribute for which each record can only have one value. For example, a book may have only one ISBN number, so the ISBN attribute would be single-assign.

Related Links

- [attribute](#) on page 5
- [primary key attribute](#) on page 29
- [multi-assign attribute](#) on page 25

[unique attribute](#) on page 39

[standard attribute](#) on page 34

[managed attribute](#) on page 25

snippeting

The snippeting feature provides the ability to return an excerpt from a record—called a snippet—to an application user who performs a record search query. A snippet contains the search terms that the user provided along with a portion of the term's surrounding content to provide context. A Web application displays these snippets on the record list page of a query's results. With the added context, users can more quickly choose the individual records they are interested in.

source record

The data that is input into Endeca Latitude. Latitude supports source records in a variety of formats.

Related Links

[data record](#) on page 11

[record](#) on page 31

[primordial record](#) on page 30

standard attribute

Attributes whose values are not organized in an enumerated list or hierarchy. A standard attribute contains information about a record. An assignment from a standard attribute is known as a standard attribute value (or KVP, key-value pair). Each Endeca record is described by a set of attribute values.

A standard attribute differs from a managed attribute in that it does not have an hierarchy or an enumeration of attribute values attached to it. Standard attributes (along with managed attributes) are intended for display once the end user has searched or navigated to a record set or an individual record. Standard attributes are described by a type of schema record — Property Description Records (PDRs).

Related Links

[attribute](#) on page 5

[primary key attribute](#) on page 29

[single-assign attribute](#) on page 33

[multi-assign attribute](#) on page 25

[unique attribute](#) on page 39

[managed attribute](#) on page 25

standard attribute value

An assignment from a non-hierarchical attribute on a record, used as a tag, or label, to classify a record in your data set. For values on standard attributes, use key-value pairs, or KVPs, where *key* is the name of an attribute, and *value* is an assigned value for this attribute.

system record

Used to control the behavior of the schema. Each attribute in a system record represents a configuration setting.

See also Property Description Record (PDR), Dimension Description Record (DDR), and Global Configuration Record (GCR).

Related Links

[Property Description Record \(PDR\)](#) on page 30

[Dimension Description Record \(DDR\)](#) on page 11

[Global Configuration Record \(GCR\)](#) on page 17

T

Related Links

[thesaurus](#) on page 37

thesaurus

The thesaurus feature allows the system to return matches for related concepts to words or phrases contained in user queries. For example, a thesaurus entry may allow searches for Mark Twain to match text containing the phrase Samuel Clemens.

U

Related Links

[unique attribute](#) on page 39

unique attribute

An attribute for which the value must be unique for each record across the data set. For example, for a book, the ISBN number would be a unique attribute.

Related Links

[attribute](#) on page 5

[primary key attribute](#) on page 29

[single-assign attribute](#) on page 33

[multi-assign attribute](#) on page 25

[standard attribute](#) on page 34

[managed attribute](#) on page 25

V

Related Links

[value search](#) on page 41

[value query](#) on page 41

value search

A search that finds all of the attribute values that have names containing terms the user provides. The result of a value search is a set of attribute values, organized by attribute. The "type ahead" feature in a search box in Latitude Studio returns value search results.

Related Links

[record search](#) on page 31

value query

Used during navigation to return the next level of values in an attribute hierarchy.

