# Oracle Endeca Commerce

## Glossary

## Version 6.3.0 • July 2012

# Copyright and disclaimer

Copyright © 2003, 2012, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

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

# A

**Related Links**

*Access Control List* on page 5
*Access Control System* on page 5
*Agidx* on page 5
*Analytics API* on page 5
*ancestors* on page 6
*aggregated Endeca record* on page 6
*aggregated record query* on page 6
*Agraph* on page 6
*Application Controller* on page 6

## Access Control List

See Endeca Access Control List.

**Related Links**

*Endeca Access Control List* on page 20

## Access Control System

See Endeca Access Control System.

**Related Links**

*Endeca Access Control System* on page 20

## Agidx

The name of the program that runs in a distributed environment and aggregates the Agraph's index with the current data subset.

## Analytics API

See Endeca Analytics API.

**Related Links**

*Endeca Analytics API* on page 20

# ancestors

The dimension values between the navigation descriptor (the current location in the dimension tree) and the dimension root (the beginning of the dimension tree). Although a parent is technically an ancestor, in general the term is used for those values above the parent in the dimension tree. See also child dimension value and parent dimension value.

**Related Links**

*parent dimension value* on page 47

# aggregated Endeca record

Multiple Endeca records grouped by a dimension or property and treated as a single record. Aggregated Endeca records are commonly used to minimize the effect of duplication in the data display. See Endeca record.

**Related Links**

*Endeca record* on page 24

# aggregated record query

An aggregated record query is a query used to obtain a single aggregated record from the MDEX Engine.

# Agraph

The name of the process used in distributed configurations to coordinate the sharing of data across machines. The Agraph resides on one of the distributed MDEX Engine processors and is responsible for receiving requests from clients, forwarding the requests to the distributed Dgraph processes, and aggregating the results.

# Application Controller

See Oracle Endeca Application Controller.

**Related Links**

*Oracle Endeca Application Controller* on page 45

# B

**Related Links**

*base SSL* on page 7
*baseline update* on page 7

## base SSL

See SSL.

**Related Links**

*SSL* on page 58

## baseline update

A baseline update (also called a full update) produces a complete re-indexing of the entire data set. It runs the update process for the whole data set.

In your baseline update pipeline, you can add, change or remove records, dimensions, dimension values and properties. In addition, configuration changes, such as dimension reordering or stop word changes require a baseline update.

You can run baseline updates nightly and use this method as your update strategy. For small to medium-sized data sets, baselines can be run frequently, as often as every few minutes. Alternatively, you can run as many partial updates as needed for those changes that can be done through partial updates, and periodically run baseline updates for those changes that require a baseline update.

**Related Links**

*delta update* on page 14
*full update* on page 28

# C

**Related Links**

*cartridge* on page 9
*certificate* on page 9
*certificate authority file* on page 10
*Charting API* on page 10
*child dimension value* on page 10
*Cluster Discovery* on page 10
*collapsible dimension hierarchy* on page 10
*Component Instance Manager* on page 11
*compound dimension query* on page 11

## cartridge

A cartridge is a functional component that a content administrator can place on a dynamic page using Experience Manager. Examples of cartridges may include a Guided Navigation cartridge, results list cartridge, or rotating banner cartridge. A cartridge is comprised of several parts:

- A configuration file (XML **template**) that defines the content structure and the editing interface in Experience Manager
- Zero or more Experience Manager **editors** that allow the content administrator to configure cartridge content
- One or more Content Assembler **tag handlers** to do additional query processing before returning results to the application.
- **Rendering code** to display the content in the Web application

**Related Links**

*Experience Manager* on page 26

## certificate

A security file that is used by all Endeca clients and servers to specify their identity when using SSL. The certificate, which is created by the Endeca enecerts utility, contains information about the certificate owner and a public key used to encrypt and decrypt data that is being exchanged between SSL-enabled components. The certificate file should be thought of as the identity of the Endeca system, or as the identify of all components of the Endeca system.

**Related Links**

*SSL* on page 58

# certificate authority file

A security file that is used in an Endeca system that is configured for mutually authenticated SSL. The CA file, which is created by the Endeca enecerts utility, is used by all Endeca clients and servers to authenticate the other endpoint of a communication channel.

**Related Links**

*SSL* on page 58

# Charting API

See Endeca Charting API.

**Related Links**

*Endeca Charting API* on page 22

# child dimension value

Sub-dimension values are known as child dimension values. Child dimension values are always more specialized than their parents and help the users to further refine their navigation query. A dimension value may have multiple child values. Because these values refine query results, child dimension values are often referred to as refinement values or refinements. See also parent dimension value.

**Related Links**

*parent dimension value* on page 47
*dimension hierarchy* on page 16
*leaf dimension value* on page 39
*parent dimension value* on page 47
*refinement dimension values* on page 53

# Cluster Discovery

See Endeca Cluster Discovery.

**Related Links**

*Endeca Cluster Discovery* on page 22

# collapsible dimension hierarchy

A dimension hierarchy in which some or all of the internal (non-root and non-leaf) dimension values are flagged as collapsible. The MDEX Engine automatically removes, or collapses, these dimension values when there are only a few leaves available for refinement, creating a more streamlined, user-friendly navigation experience for your users.

# Component Instance Manager

The Component Instance Manager manages Record Store instances. Management means that the Component Instance Manager is responsible for creating, deleting, and listing all Record Store instances.

# compound dimension query

A combination of one or more dimension values (or, more specifically, dimension value IDs). A compound dimension query instructs the MDEX Engine to return the set of records that represents the intersection of all the dimension values that it contains.

# D

**Related Links**

*Data Foundry* on page 13
*data pipeline* on page 13
*data pipeline component* on page 14
*dead end query* on page 14
*delta update* on page 14
*Deployment Template* on page 14
*deprecated* on page 14
*descriptor* on page 15
*Dgidx* on page 15
*Dgraph* on page 15
*Dgraph Cluster* on page 15
*differential crawl* on page 15
*dimension* on page 15
*dimension adapter* on page 16
*dimension descriptor* on page 16
*dimension group* on page 16
*dimension hierarchy* on page 16
*dimension root* on page 16
*dimension search* on page 17
*dimension server* on page 17
*dimension value* on page 17
*dynamic ranking* on page 17

## Data Foundry

See Endeca Information Transformation Layer.

**Related Links**

*Endeca Information Transformation Layer* on page 23

## data pipeline

The data pipeline, as displayed in Developer Studio, is the graphical representation of how the Endeca Information Transformation Layer transforms source data into tagged Endeca records.

**Related Links**

*pipeline* on page 48

# data pipeline component

The pipeline is composed of a collection of components. Each component performs a specific function during the transformation of your source data into Endeca records, such as loading data, standardizing properties, and tagging records with dimension values. Components are linked together, by means of cross-references, providing a sequential flow and a pipeline feel.

# dead end query

When an intersection of common records doesn't exist between all of the dimension values in a navigation query, that query is considered a dead end. The MDEX Engine automatically eliminates such dead end queries in the way it structures the follow-on query information that it returns in its query results. See also Guided Navigation.

**Related Links**

*Guided Navigation* on page 29

# delta update

A full baseline index, similar to the baseline update, that is performed by joining a smaller extraction of source data (only the added, updated, or removed source records) with the output from the previous baseline update. A delta update is ideal when you need to reduce the time required for loading the source data repository and for extracting the data.

See also partial update and baseline update.

**Related Links**

*partial update* on page 47
*baseline update* on page 7

# Deployment Template

The Deployment Template is a utility that you run to create a new Endeca application with the complete directory structure required for deployment, including Endeca Application Controller (EAC) control scripts, configuration files, and batch files or shell scripts that wrap common script functionality.

# deprecated

Deprecated means that an item (platform, feature, flag, etc.) is fully supported, as in prior releases, but that Oracle intends to discontinue support in a subsequent release.

# descriptor

See dimension descriptor and navigation descriptor.

**Related Links**

*dimension descriptor* on page 16
*navigation descriptor* on page 43

# Dgidx

The name of the program that indexes tagged Endeca records and produces indices in the Endeca MDEX Engine format.

# Dgraph

The name of the program that launches the MDEX Engine. You can run more than one Dgraph process on a single computer.

# Dgraph Cluster

Dgraph clusters are used in the configuration of an application installed by the Deployment Template utility to apply actions to an entire cluster of Dgraphs, rather than manually iterating over a number of Dgraphs. They also contain logic associated with Dgraph restart strategies.

# differential crawl

See incremental crawl.

**Related Links**

*incremental crawl* on page 33

# dimension

A collection of related dimension values, organized into a navigable tree with a dimension root at the top. See also externally created dimension, and externally managed dimension.

**Related Links**

*externally created dimension* on page 26
*externally managed dimension* on page 26

# dimension adapter

A pipeline component that loads dimension information. Dimension adapters can load dimension information in either XML or delimited format.

# dimension descriptor

The dimension that has been used in a navigation query to generate a set of query results. Every descriptor dimension has one or more descriptors (dimension values) associated with it. A descriptor dimension provides a handle for accessing the dimension value(s) used to refine the index records to create the current query results. See also navigation descriptor.

**Related Links**

*descriptor* on page 15
*navigation descriptor* on page 43
*navigation descriptor* on page 43

# dimension group

An implicit or explicit grouping of dimensions used to organize dimensions. An implicit group consisting of a single dimension is automatically generated for each dimension that is not included in an explicit, user-defined grouping (which generally contains multiple dimensions).

# dimension hierarchy

A logical (though not physical) tree structure with a dimension root at the top and related dimension values below. Dimension hierarchies allow you to exercise a higher level of control over the number of follow-on choices that are presented to users as they navigate. See also parent dimension value and child dimension value.

**Related Links**

*parent dimension value* on page 47
*child dimension value* on page 10
*flat dimension* on page 27
*hierarchical dimension* on page 31

# dimension root

The first dimension value in a dimension tree. A dimension root generally has the same name and ID as its first child dimension.

# dimension search

A search that finds all of the dimension values that have names that contain terms the user provides. The result of a dimension search is a set of dimension values, organized by dimension.

**Related Links**

*filter* on page 27
*keyword search* on page 37
*record search* on page 53
*search query* on page 57

# dimension server

Working in conjunction with dimension adapters, dimension servers serve as a centralized source of dimension information for all other pipeline components.

# dimension value

Member of a dimension; used as a tag, or label, to classify a record in your data set. Tagging a record with a dimension value identifies that record as a valid result when a user queries for the dimension value. Dimension values can be auto-generated, defined explicitly, or imported from external taxonomy systems, for example Stratify.

**Related Links**

*synonym* on page 59

# dynamic ranking

The ordering of dimension values according to how frequently they appear within a data set, rather than in the MDEX Engine's default alphabetical order. Dynamic ranking is particularly useful in large data sets.

**Related Links**

*static ranking* on page 58

# E

**Related Links**


*Endeca Access Control List* on page 20
*Endeca Access Control System* on page 20
*Endeca Analytics API* on page 20
*Endeca Analytics module* on page 20
*Endeca APIs* on page 20
*Endeca application* on page 21
*Endeca Assembler* on page 21
*Endeca CAS Console for Oracle Endeca Workbench* on page 21
*Endeca CAS Extension API* on page 21
*Endeca CAS Server* on page 21
*Endeca CAS Service* on page 22
*Endeca Charting API* on page 22
*Endeca Cluster Discovery* on page 22
*Endeca Content Acquisition System* on page 22
*Endeca Data Foundry* on page 22
*Endeca Developer Studio* on page 22
*Endeca HTTP Service* on page 23
*Endeca implementation* on page 23
*Endeca Information Transformation Layer* on page 23
*Endeca instance* on page 23
*Endeca Logging and Reporting System* on page 23
*Endeca MDEX Engine* on page 24
*Endeca Presentation API* on page 24
*Endeca record* on page 24
*Endeca Record Store* on page 25
*Endeca Relationship Discovery* on page 25
*Endeca system* on page 25
*Endeca Term Discovery* on page 25
*Endeca Tools Service* on page 25
*Endeca Web Crawler* on page 25
*Experience Manager Editor SDK* on page 26
*Experience Manager* on page 26
*expression* on page 26
*externally created dimension* on page 26
*externally managed dimension* on page 26

# Endeca Access Control List

A property on an Endeca record that lists user and group permissions, such as the right to view the record. In an implementation using the Endeca Access Control System, the MDEX Engine reads the Endeca ACL property and uses the user entitlement filter to determine if the user that made the query is authorized to view the record.

**Related Links**

*Access Control List* on page 5

# Endeca Access Control System

A security infrastructure that authenticates a user's identity against an external directory (such as an LDAP directory) and creates a user entitlement filter that limits access to only those records that the user is authorized to see. One or more login modules are configured to perform the actual authentication. See also file-based login module, LDAP login module, and user entitlement filter.

**Related Links**

*file-based login module* on page 27
*LDAP login module* on page 39
*user entitlement filter* on page 63

# Endeca Analytics API

Available with Endeca Analytics, the Analytics API extends the Endeca Presentation API to enable interactive applications that allow users to explore aggregate and statistical views of information using a Guided Navigation interface. See also Endeca Presentation API.

**Related Links**

*Analytics API* on page 5
*Endeca Presentation API* on page 24

# Endeca Analytics module

Endeca Analytics builds on the core capabilities of the Endeca MDEX Engine to enable applications that examine aggregate information such as trends, statistics, analytical visualizations, comparisons, and so on, all within the Guided Navigation interface.

# Endeca APIs

The collection of APIs, such as the Presentation API, Logging API, and Forge API, that provide an interface to an Endeca implementation.

**Related Links**

*Endeca MDEX Engine* on page 24
*Endeca Presentation API* on page 24

# Endeca application

The end-user-facing portion of an Endeca implementation.

**Related Links**

*Endeca implementation* on page 23

# Endeca Assembler

The Endeca Assembler API enables a Web application to query the MDEX Engine and retrieve the appropriate dynamic content based on a user's navigation state or other triggers. The Endeca Assembler returns both Endeca query results familiar from the Presentation API as well as a content item object that encapsulates the page configuration specified by the content administrator. All the content for a page, including the results of any additional queries needed for spotlighting or merchandising, are wrapped in the content item object, simplifying the logic in the front-end application by reducing the need to manage sub-queries in the application layer.

# Endeca CAS Console for Oracle Endeca Workbench

The CAS Console for Oracle Endeca Workbench is a Web-based application used to crawl various data sources including file systems, content management systems, and custom data source extensions. During the Content Acquisition System installation, the CAS Console is installed as an extension to Oracle Endeca Workbench.

**Related Links**

*Oracle Endeca Workbench* on page 45

# Endeca CAS Extension API

The CAS Extension API provides interfaces and classes to build extensions to the Content Acquisition System such as data sources and manipulators. An extension developer packages extensions into a JAR, and a CAS application developer installs the JAR and any additional JARs (for third-party dependencies) into the Content Acquisition System. After installation, the extensions are available and configurable using the CAS Console, the CAS Server API, and the CAS Server Command-line Utility.

# Endeca CAS Server

A component of the Endeca Content Acquisition System that gathers source data by managing file system and CMS crawls. The output of a crawl is a set of Endeca records that are used in a Developer Studio project.

# Endeca CAS Service

The Endeca CAS Service is a servlet container that runs the CAS Server, the Component Instance Manager, and any number of Record Store instances (one per data source).

# Endeca Charting API

Available with Endeca Analytics, the Charting API extends the Endeca Presentation API to support graphical visualization of Endeca analytics results. See also Endeca Presentation API.

**Related Links**

*Charting API* on page 10
*Endeca Presentation API* on page 24

# Endeca Cluster Discovery

A feature of the Endeca MDEX Engine that uses the salient terms created by Term Discovery and intelligently clusters them into groupings that represent distinct and coherent sets of records. These clusters are similarly leveraged as intuitive ways for end users to refine or broaden their Endeca queries.

**Related Links**

*Cluster Discovery* on page 10

# Endeca Content Acquisition System

The Endeca Content Acquisition System is a set of components that add, configure, and crawl data sources for use in an Endeca application. Data sources include file systems, Content Management System crawls, Web servers, and custom data sources. The Endeca Content Acquisition System crawls data sources, converts documents and files to Endeca records and stores them for use in an Endeca pipeline.

# Endeca Data Foundry

The component of the Endeca Information Transformation Layer that aggregates information from disparate sources, transforms it into Endeca records, and produces indices for use by the Endeca MDEX Engine.

# Endeca Developer Studio

A Windows application used to define all aspects of the instance configuration including pipeline components, Endeca properties and dimensions, precedence rules, dynamic business rules, and user profiles. Developer Studio uses a project file, with an .esp extension, that contains pointers to the XML files that support an instance configuration. See also Oracle Endeca Workbench.

**Related Links**

*Oracle Endeca Workbench* on page 45

# Endeca HTTP Service

The Endeca HTTP Service is an application server that runs the Oracle Endeca Application Controller.

# Endeca implementation

An entire Endeca system, comprising the Endeca instance (or back end), the Endeca application (or front end), and any machines included in the resource collection. See also Endeca instance, Endeca application, and resource collection.

**Related Links**

*Endeca instance* on page 23
*Endeca application* on page 21
*resource collection* on page 54
*Endeca system* on page 25
*reference implementation* on page 53

# Endeca Information Transformation Layer

The Endeca Information Transformation Layer (ITL) is the component of Oracle Endeca Commerce that reads in your source data and manipulates it into an index for the MDEX Engine. It consists of the Content Acquisition System, Forge, and the indexing processes. The ITL components are offline processes that you run at intervals that are appropriate for your business needs.

**Related Links**

*Data Foundry* on page 13
*Oracle Endeca Guided Search*

# Endeca instance

The back end of an Endeca implementation, which is not visible to the end user. This typically includes Developer Studio project files and system configuration files.

**Related Links**

*Endeca implementation* on page 23

# Endeca Logging and Reporting System

The Endeca Logging and Reporting System provides an application-level logging solution, using a stand-alone logging server. See also Logging API, Log Server and Report Generator.

**Related Links**

*Logging API* on page 39
*Log Server* on page 39
*Report Generator* on page 54

# Endeca MDEX Engine

The query engine that is the core of Oracle Endeca Commerce. The MDEX Engine consists of the indexer (Dgidx), the Dgraph, and the Agraph. The MDEX Engine stores the indices generated by the Data Foundry. After the indices are stored, the MDEX Engine 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 APIs.

**Related Links**

*Endeca APIs* on page 20
*MDEX Engine* on page 41

# Endeca Presentation API

The interface between client browsers and the MDEX Engine. The Endeca API must be available online, on a Web or application server, as long as you want clients to be able to access data. See also Endeca APIs.

**Related Links**

*Endeca Analytics API* on page 20
*Endeca Charting API* on page 22
*Endeca APIs* on page 20
*Presentation API* on page 48

# Endeca record

The individual items that the user is trying to navigate to in an Endeca application. Endeca records generally correspond to traditional records in a source database. Unlike source records, however, Endeca records have been standardized for consistency, and then classified with dimension values. The production of Endeca records is a primary result of the pipeline development process.

**Related Links**

*aggregated Endeca record* on page 6
*record page* on page 52

# Endeca Record Store

The Endeca Record Store provides persistent storage for generations of records that can later be accessed by Forge for baseline and incremental updates. The CAS Server writes crawl output from each data source to a unique Record Store instance.

# Endeca Relationship Discovery

An optional feature of Oracle Endeca Commerce that includes Endeca Term Discovery and Endeca Cluster Discovery. It enables users to discover relationships in their source data records by extracting salient terms from records and using them to build sets (clusters) of related records.

**Related Links**

*Relationship Discovery* on page 54

# Endeca system

See Endeca implementation.

**Related Links**

*Endeca implementation* on page 23

# Endeca Term Discovery

A feature of the Endeca Information Transformation Layer that identifies and extracts salient noun phrases (terms) from documents, and subsequently tags them to the associated records. The terms, which are identified through natural language processing and statistical inference techniques, constitute words or groups of words that are significant in both the corpus and in individual records. These terms, when presented in the application's front end, allow users to refine and broaden their queries.

**Related Links**

*Term Discovery* on page 61

# Endeca Tools Service

The Endeca Tools Service is an application server that runs Oracle Endeca Workbench.

# Endeca Web Crawler

A component of the Endeca Content Acquisition System that gathers source data by crawling HTTP and HTTPS Web sites and outputs Endeca records for use in a Developer Studio project.

# Experience Manager Editor SDK

The Experience Manager Editor SDK enables application developers to introduce new functionality into Experience Manager via custom editors. The SDK consists of the Experience Manager Editor API, a sample editor project, and associated documentation.

# Experience Manager

The Experience Manager is a powerful template-based tool that enables rapid creation of rich, dynamic landing pages. Experience Manager gives merchandisers and content administrators unprecedented control over site content without the need for IT intervention.

# expression

Analogous to functions, expressions tell the Forge program which records, properties, or dimensions to affect, and how to affect them. See also Forge.

**Related Links**

*Forge* on page 28

# externally created dimension

A logical hierarchy of a data set that is transformed from its source format to Endeca compatible XML outside of Developer Studio. The logical hierarchy of the dimension conforms to Endeca's external interface for describing a data hierarchy before you import the dimension into your project.

**Related Links**

*dimension* on page 15

# externally managed dimension

A logical hierarchy for a data set that is built and managed using a third-party tool. Once you include an externally managed taxonomy in your Developer Studio project, it becomes a dimension whose hierarchy is managed by the third-party tool that created it. In Developer Studio, you cannot add or remove dimension values from it. If you want to modify a dimension or its dimension values, you have to edit the taxonomy using the third-party tool and then update the taxonomy in your project.

**Related Links**

*dimension* on page 15

# F

**Related Links**

*file-based login module* on page 27
*filter* on page 27
*flat dimension* on page 27
*Forge* on page 28
*full crawl* on page 28
*full update* on page 28

## file-based login module

A component of the Endeca Access Control System that authenticates a user's identity and group membership against information in a local directory file.

**Related Links**

*Endeca Access Control System* on page 20
*stacked authentication* on page 58

## filter

Query settings that limit what the query can return. See also dimension search, record search, and user entitlement filter.

**Related Links**

*dimension search* on page 17
*record search* on page 53
*user entitlement filter* on page 63

## flat dimension

A dimension that has only one level of hierarchy underneath the dimension root. See also dimension hierarchy.

**Related Links**

*dimension hierarchy* on page 16

# Forge

A component of the Data Foundry that processes your source data records or documents into tagged Endeca records.

**Related Links**

*expression* on page 26

# full crawl

A crawl in which a crawler retrieves all the source documents that it is configured to access. See also incremental crawl.

**Related Links**

*incremental crawl* on page 33
*incremental crawl* on page 33

# full update

See baseline update.

**Related Links**

*baseline update* on page 7

# G

**Related Links**

*Guided Navigation* on page 29

## Guided Navigation

Guided Navigation is the presentation of valid follow-on refinement choices to the user. You can also think of Guided Navigation as the elimination of invalid refinement queries, or dead ends.

**Related Links**

*dead end query* on page 14

# H

**Related Links**

*hidden dimension* on page 31
*hierarchical dimension* on page 31

## hidden dimension

A dimension that is not accessible from an Endeca application. If a dimension is marked as hidden on the Dimension editor of Developer Studio, the Endeca MDEX Engine will not return the dimension or any of its values as refinement options. Although hidden dimensions are not rendered in the user interface, records are still tagged with relevant values from these dimensions. Therefore, a user is able to search for records based on dimension values within hidden dimensions.

**Related Links**

*inert dimension value* on page 33

## hierarchical dimension

See dimension hierarchy.

**Related Links**

*dimension hierarchy* on page 16

# I

**Related Links**

*incremental crawl* on page 33
*indexer adapter* on page 33
*inert dimension value* on page 33
*Information Transformation Layer* on page 33
*instance configuration* on page 34
*intermediate dimension value* on page 34

## incremental crawl

A crawl in which the crawler retrieves only the source documents that have changed since the last crawl. Incremental crawls are also known as differential crawls. See also full crawl.

**Related Links**

*differential crawl* on page 15
*full crawl* on page 28
*full crawl* on page 28

## indexer adapter

A pipeline component that saves data that is ready to be indexed.

## inert dimension value

A dimension value that is not included in the navigation state. When a user selects an inert dimension value, the navigation state is not changed, but the children of the dimension value are displayed for selection. See also hidden dimension.

**Related Links**

*hidden dimension* on page 31

## Information Transformation Layer

See Endeca Information Transformation Layer.

**Related Links**

*Endeca Information Transformation Layer* on page 23

# instance configuration

A set of project files that configure the back-end processes (Forge, Dgidx, Dgraph) of an Endeca implementation.

# intermediate dimension value

A non-navigable dimension value that changes the query parameters but does not affect the current Endeca record set. For example, A-D, E-H, I-L, and so forth.

# J

**Related Links**

*Java manipulator* on page 35
*Join* on page 35

## Java manipulator

A pipeline component that you can write in Java and add to your pipeline, to manipulate records. A Java manipulator that you create must adhere to the Content Adapter Development Kit (CADK) Java Adapter interface classes. A Java manipulator transforms source records and Endeca records as part of data processing in the Endeca Information Transformation Layer. For example, Java manipulators are used by the Endeca Term Discovery feature to extract terms from source records, provide filtering and scoring mechanisms, and tag the terms to the associated Endeca records.

## Join

A join combines records from two or more tables in a relational database. In the Structured Query Language (SQL), there are three types of joins: inner, outer, and cross. Outer joins are subdivided further into left outer joins, right outer joins, and full outer joins.

# K

**Related Links**

*key properties* on page 37
*keyword search* on page 37

## key properties

Available with Endeca Analytics, key properties are property- and dimension-level metadata that allow customized application behavior.

## keyword search

A query that returns results based on a user-specified text string (keyword). See also record search and dimension search.

**Related Links**

*record search* on page 53
*dimension search* on page 17

# L

**Related Links**

*LDAP login module* on page 39
*leaf dimension value* on page 39
*Log Server* on page 39
*Logging API* on page 39

## LDAP login module

A component of the Endeca Access Control System that authenticates a user's identity and group membership against a lightweight directory access protocol (LDAP) directory.

**Related Links**

*Endeca Access Control System* on page 20
*stacked authentication* on page 58

## leaf dimension value

The bottom-most dimension value in a dimension tree. A leaf dimension value has no children. See also child dimension value.

**Related Links**

*child dimension value* on page 10

## Log Server

The Log Server translates log requests into log files. These log files can be used by the Report Generator to create HTML-based, human-readable log reports. Log requests are passed from the application modules to the Log Server via the Logging API. See also Logging API and Report Generator.

**Related Links**

*Endeca Logging and Reporting System* on page 23
*Logging API* on page 39
*Report Generator* on page 54

## Logging API

The Logging API receives log requests from an Endeca application's modules, and passes them to the Log Server via HTTP.

**Related Links**

*Endeca Logging and Reporting System* on page 23
*Log Server* on page 39

# M

**Related Links**

*MDEX Engine* on page 41
*MDEX Engine query request* on page 41
*mutually authenticated SSL* on page 41

## MDEX Engine

See Endeca MDEX Engine.

**Related Links**

*Endeca MDEX Engine* on page 24

## MDEX Engine query request

A client browser request that has been altered to use MDEX Engine-specific parameters. Not to be confused with navigation query, which is a specific type of query. See also navigation query.

**Related Links**

*navigation query* on page 43

## mutually authenticated SSL

See SSL.

**Related Links**

*SSL* on page 58

# N

**Related Links**

*navigation descriptor* on page 43
*navigation query* on page 43
*navigation reference* on page 43
*navigation state search query* on page 43

## navigation descriptor

The specific dimension value for a dimension that is specified in an MDEX Engine query. The navigation descriptor describes the user's current dimension value choice within a dimension; this is the user's current location within that dimension's tree. See also dimension descriptor.

**Related Links**

*descriptor* on page 15
*dimension descriptor* on page 16
*dimension descriptor* on page 16

## navigation query

A query that returns a set of records based on user-selected characteristics along with any follow-on information. See also record query.

**Related Links**

*MDEX Engine query request* on page 41
*record query* on page 52

## navigation reference

A collection of dimension values that can be used to create a navigation query. Navigation references, in essence, are navigation queries waiting to happen.

## navigation state search query

A query to obtain the set of valid navigation references whose constituent dimension values have names that match all of the search terms.

**Related Links**

*search query* on page 57

# O

**Related Links**

*Oracle Endeca Application Controller* on page 45
*Oracle Endeca Commerce* on page 45
*Oracle Endeca Tools and Frameworks* on page 45
*Oracle Endeca Workbench* on page 45

## Oracle Endeca Application Controller

The Oracle Endeca Application Controller is the interface you use to control, manage, and monitor your Endeca implementations. It provides the infrastructure to support Endeca projects from design through deployment and runtime. It replaces the Control Interpreter, while leaving the Endeca tools (Developer Studio, Oracle Endeca Workbench) largely intact.

## Oracle Endeca Commerce

The platform upon which Guided Navigation solutions are built. Oracle Endeca Commerce is made up of the Endeca Information Transformation Layer, the MDEX Engine, and the Endeca Assembler.

## Oracle Endeca Tools and Frameworks

The Oracle Endeca Tools and Frameworks enable the development of Web applications that present dynamic and contextual content to end customers across multiple channels. The main components of Tools and Frameworks are:

- Oracle Endeca Workbench with Experience Manager, a Web-based tool that enables business users to create and manage dynamic content configurations across multiple channels or applications.
- Endeca Assembler, a system for collecting and sending queries to the Endeca MDEX Engine and returning the resulting data to an application as ready-to-render content items.
- The Discover Electronics reference application, an application that demonstrates best practices for Oracle Endeca Commerce applications.

## Oracle Endeca Workbench

Oracle Endeca Workbench is a Web-based tool that provides a way for business users and merchandisers to configure portions of their Endeca application and provides system administrators with a means to configure and administer an Endeca implementation.

Unlike Developer Studio, which provides a rich development environment for configuring all aspects of an Endeca implementation, Oracle Endeca Workbench focuses on a smaller set of common, every day

configuration and maintenance tasks. This reduced focus gives Workbench a smaller footprint that can exist within the bounds of a Web-based application.

# P

**Related Links**

*parent dimension value* on page 47
*partial update* on page 47
*Perl manipulator* on page 48
*pipeline* on page 48
*precedence rule* on page 48
*Presentation API* on page 48
*property* on page 48
*property mapper* on page 48
*property name* on page 49
*property value* on page 49

## parent dimension value

A dimension value that has sub-dimension values. Each dimension value can have only one parent value. See also child dimension value and ancestors.

**Related Links**

*child dimension value* on page 10
*dimension hierarchy* on page 16
*ancestors* on page 6
*child dimension value* on page 10

## partial update

A partial update is a change in the overall data set that does not require restarting the MDEX Engine. Partial updates allow you to update only those portions of the MDEX Engine index that have changed since the last baseline update.

A partial update lets you implement a number of the source data changes. For project configuration changes, run a baseline update.

Even if you are only making source data changes, keep in mind that some configuration information that is derived from the data, such as dictionary or wildcarding information, can become outdated. Therefore, to keep dictionaries up-to-date, periodically run baseline updates.

**Related Links**

*delta update* on page 14

# Perl manipulator

A pipeline component that uses Perl to efficiently manipulate source records and Endeca records as part of data processing performed in the Endeca Information Transformation Layer. The Forge API for Perl provides the means to plug a perl manipulator into Forge with four plug-in methods that you write. From the methods that you write, you can also call methods in the EDF name space that Endeca provides to perform data manipulation.

# pipeline

See data pipeline.

**Related Links**

*data pipeline* on page 13

# precedence rule

A trigger that causes dimension values that were not previously displayed to become available. Precedence rules reduce information overload, because the user is not presented with too many, or inappropriate, navigation choices.

# Presentation API

See Endeca Presentation API.

**Related Links**

*Endeca Presentation API* on page 24

# property

A name-value pair (a combination of property name and property value) containing information about a record. Generally, a source record is nothing more than a set of properties. Properties are intended for display once the end user has searched or navigated to a record set or an individual record. Properties can also be used to automatically generate dimensions and dimension values.

# property mapper

A pipeline component used to specify what your Endeca application should do with individual properties in your source data: explicitly map them to existing Endeca dimensions or properties, create new Endeca dimensions or properties to map them to, or ignore them.

# property name

Property names are mapped to dimensions. See property.

**Related Links**

*property* on page 48

# property value

Property values are mapped to dimension values. See property.

**Related Links**

*property* on page 48

# R

**Related Links**


*range dimension* on page 51
*rapid update* on page 51
*raw data* on page 52
*record adapter* on page 52
*record assembler* on page 52
*record cache* on page 52
*record manipulator* on page 52
*record page* on page 52
*record query* on page 52
*Record Relationship Navigation* on page 53
*record search* on page 53
*reference implementation* on page 53
*refinement dimension* on page 53
*refinement dimension values* on page 53
*Relationship Discovery* on page 54
*relevance ranking module* on page 54
*Report Generator* on page 54
*resident set size (RSS)* on page 54
*resource collection* on page 54
*resource configuration* on page 55
*results page* on page 55
*rollup key* on page 55


## range dimension

Dimension that can be organized into discrete ranges, for example, $10-$20, $21-$50, $51-$100, and so forth. Range dimensions are useful for data, such as price, that should be navigated as discrete values.

## rapid update

See partial update.

**Related Links**


*partial update* on page 47

# raw data

See source record.

**Related Links**

*source record* on page 57

# record adapter

A pipeline component used to read source data into the Endeca Data Foundry.

# record assembler

A pipeline component used to join source records originating from different files.

# record cache

A pipeline component that stores a temporary copy of record data that has been read in by a record adapter. Record caches are generally used in conjunction with record assemblers and are set up to contain data from secondary data sources.

# record manipulator

A pipeline component that changes the data that is associated with a record. This can include changing properties, property values, and dimension value tagging.

# record page

The displayed representation of a single Endeca record. Most Endeca-enabled applications are designed so that when an end-user selects an item (that is, an Endeca record) in their navigation or search results page, a record page appears displaying all or most of the record's properties. See also Endeca record.

**Related Links**

*Endeca record* on page 24

# record query

A query to obtain a single specific Endeca record from the MDEX Engine.

**Related Links**

*navigation query* on page 43

# Record Relationship Navigation

Record Relationship Navigation is an extension to Endeca Query Language for relational (normalized) data that enables more complex Guided Navigation at larger scale. Record Relationship Navigation is an optional module that is intended for use with complex relational data. It allows users to logically connect records at query time and navigate based on the connected relationships. It is based on EQL. Interrelationships between logical entities are not only preserved by the indexing process, but also allow for simultaneously navigating different types of entities based on attributes of any subsets of the data.

# record search

A keyword search that filters the record set to include only those Endeca records that have at least one property or dimension 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 dimension search.

**Related Links**

*filter* on page 27
*keyword search* on page 37
*dimension search* on page 17
*search query* on page 57

# reference implementation

A sample Endeca web application that provides skeleton examples of typical pages (navigation, record, and aggregated record pages) and the components that make up these pages (for example, navigation controls, navigation descriptors, and a record set). It is intended only as a guide for creating MDEX Engine queries and building pages from the query results. See also Endeca application.

**Related Links**

*Endeca implementation* on page 23

# refinement dimension

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

# refinement dimension values

The next set of dimension value choices the user can make to refine his or her navigation query. You refine a navigation query by navigating from a dimension value to one of its children. See also child dimension value.

**Related Links**

*child dimension value* on page 10

# Relationship Discovery

See Endeca Relationship Discovery.

**Related Links**

*Endeca Relationship Discovery* on page 25

# relevance ranking module

A search interface feature that lets the developer control the order in which record or dimension 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.

# Report Generator

The Report Generator uses the log files created by the Log Server to generate HTML-based reports.

**Related Links**

*Endeca Logging and Reporting System* on page 23

*Log Server* on page 39

# resident set size (RSS)

*Resident set size (RSS)* is the amount of physical memory currently allocated and used by the MDEX Engine process. As the MDEX Engine process runs, the active executable code and data are brought into RAM, becoming part of the RSS for the MDEX Engine.

For more information, see the *Oracle Endeca MDEX Engine Performance Tuning Guide*.

# resource collection

Represents the set of resources used to host an Endeca application, which generally consists of a set of computers with various application-specific roles (such as data processing, MDEX Engine servers, log collection and analysis, and so forth).

**Related Links**

*Endeca implementation* on page 23

# resource configuration

Specifies the configuration of the resources on which the application will run (such as machines in the resource pool, number of segments, number of replicas, and so on).

# results page

A page summarizing multiple records returned as a result of a navigation or search query.

# rollup key

The property or dimension name by which an aggregated Endeca record is consolidated. Records that have the same value for the property or dimension are rolled up into an aggregated record.

# S

**Related Links**

*Search Engine Optimization Module* on page 57
*search interface* on page 57
*search query* on page 57
*source record* on page 57
*SSL* on page 58
*stacked authentication* on page 58
*static ranking* on page 58
*style* on page 58
*supplemental object* on page 58
*synonym* on page 59

## Search Engine Optimization Module

See URL Optimization API.

**Related Links**

*URL Optimization API* on page 63

## search interface

A named collection of properties and/or dimensions, 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 properties and/or dimensions simultaneously.

## search query

See dimension search, record search, and navigation state search query.

**Related Links**

*dimension search* on page 17
*record search* on page 53
*navigation state search query* on page 43

## source record

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

# SSL

The Secure Sockets Layer protocol that protects the privacy and integrity of data being transferred across a network. Base SSL provides security by encrypting communications among Endeca components. In addition to using encryption, mutually authenticated SSL provides a higher level of security by forcing each component to use a certificate to confirm its identity to other components. See also certificate and certificate authority file.

**Related Links**

*base SSL* on page 7
*mutually authenticated SSL* on page 41
*certificate* on page 9
*certificate authority file* on page 10

# stacked authentication

A feature of the Endeca Access Control System that allows multiple login modules to be used to authenticate a user's identity. See also file-based login module and LDAP login module.

**Related Links**

*file-based login module* on page 27
*LDAP login module* on page 39

# static ranking

The order in which dimensions and dimension values appear in the user interface, overriding MDEX Engine defaults. See also dynamic ranking.

**Related Links**

*dynamic ranking* on page 17

# style

See dynamic business rule.

**Related Links**

*dynamic business rule*

# supplemental object

Optional object that accompanies a basic navigation query. Supplemental objects are the general data structure used to create merchandising or analytics displays. Each supplemental object may contain one or more navigation references, one or more records, and one or more object attributes.

# synonym

Alternative terms used for the display and/or classification of dimension values. All dimension value definitions contain one or more synonyms. See also dimension value.

**Related Links**

*dimension value* on page 17

# T

**Related Links**

*tag handler* on page 61
*Term Discovery* on page 61

## tag handler

See Content Assembler community tag handler.

**Related Links**

*Content Assembler community tag handler*

## Term Discovery

See Endeca Term Discovery.

**Related Links**

*Endeca Term Discovery* on page 25

# U

**Related Links**

*user entitlement filter* on page 63
*URL Optimization API* on page 63

## user entitlement filter

A record filter that is created by the Endeca Access Control System for an authenticated user. The filter defines the user's access rights to the data in the Endeca implementation and is used by the MDEX Engine during queries to return only those records that the user is authorized to see.

**Related Links**

*Endeca Access Control System* on page 20
*filter* on page 27

## URL Optimization API

The URL Optimization API enables you to create application URLs that are optimized for internet search engines. In particular, the API provides the capability to shorten and canonicalize URLs and add search and navigation keywords to URLs. The resulting URLs are more optimized for internet search engines and more understandable to front-end application users.

**Note:** This API is sometimes referred to as the Search Engine Optimization (SEO) Module.

**Related Links**

*Search Engine Optimization Module* on page 57

# V

**Related Links**

*virtual process size* on page 65

## virtual process size

The *virtual process size (or address space)* for the Dgraph is the total amount of virtual memory allocated by the operating system to the MDEX Engine process at any point in time. This includes the Dgraph code, the MDEX Engine data as represented on disk, the Dgraph cache and any temporary work space.

For more information, see the *Oracle Endeca MDEX Engine Performance Tuning Guide*.

See also resident set size (RSS) and working set size (WSS) of the Dgraph.

**Related Links**

*working set size (WSS) of the Dgraph process* on page 67
*resident set size (RSS)* on page 54

# W

**Related Links**

*Web services and XQuery for Endeca* on page 67
*Workbench* on page 67
*working set size (WSS) of the Dgraph process* on page 67

## Web services and XQuery for Endeca

Web services and XQuery for Endeca provides Endeca application developers with a flexible, extensible, and standards-compliant query processing solution.You can use Web services and XQuery for Endeca alongside the Endeca Presentation API to extend the functionality of your Endeca application.

## Workbench

See Oracle Endeca Workbench.

**Related Links**

*Oracle Endeca Workbench* on page 45

## working set size (WSS) of the Dgraph process

The working set of the Dgraph process is a collection of pages in the virtual address space of the process that is resident in physical memory. The pages in the working set have been most recently and frequently referenced. In other words, the Dgraph working set is the amount of memory a Dgraph process is consuming now. This is the amount of memory that is needed to avoid paging.

The following statements describe the working set size (WSS):

- The WSS cannot be computed, although it is always less than or equal to the amount of virtual process size for the MDEX Engine.
- Determine the WSS experimentally: if you notice that increasing RSS (by adding RAM or subtracting competing processes) improves performance of the MDEX Engine, this means that the WSS was previously larger than the RSS. This was likely the cause of the performance degradation.
- If the size of the WSS grows too close to the amount of RAM, or starts to exceed it, paging to disk begins and you will notice rapid decreases in performance.

For more information, see the *Oracle Endeca MDEX Engine Performance Tuning Guide*.

# Z

**Related Links**

*zone* on page 69

## zone

See dynamic business rule.

**Related Links**

*dynamic business rule*