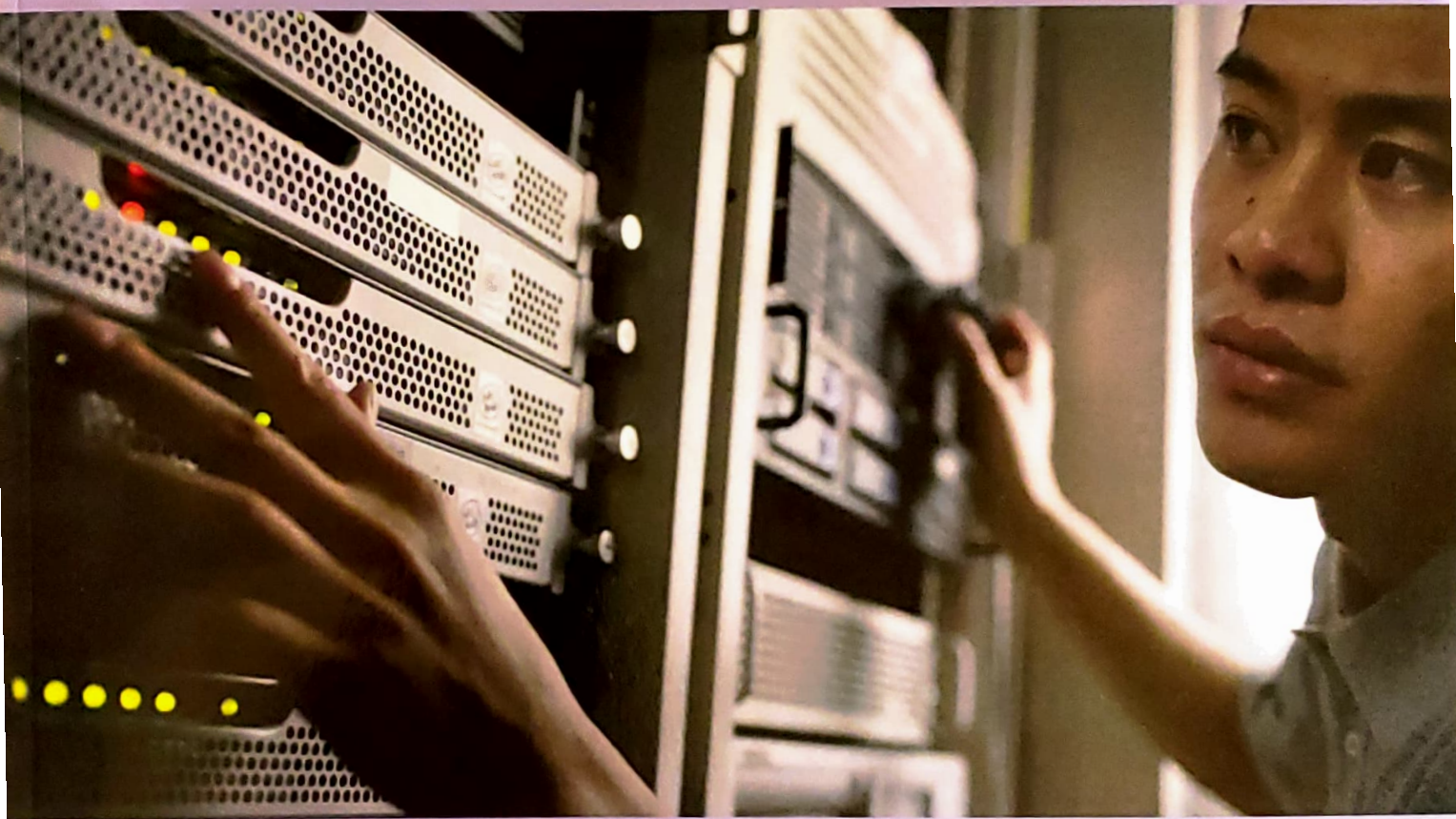


# DICTIONARY OF COMPUTER AND INTERNET TERMS



JOHN C. RIGDON

BlackBerry 1035  
U.S. Patent No. 8,359,016

# **Dictionary of Computer and Internet Terms (Vol. 1)**

**John C. Rigdon,  
Editor**

## Dictionary of Computer and Internet Terms

1<sup>st</sup> Edition – AUG 2016 0/0/0/CS

This dictionary is adapted from Microsoft Language Portal with additions and edits by John C. Rigdon.

© 2016 Eastern Digital Resources. All Rights Reserved. No part of this book may be reproduced by any means without the express written consent of the copyright holder. "Entries from the Microsoft Language Portal. © 2016 Microsoft Corporation. All rights reserved." Distribution release by Microsoft. See <https://www.microsoft.com/Language/en-US/LicenseAgreement.aspx>

Published by:  
Eastern Digital Resources  
31 Bramblewood Dr SW  
Cartersville, GA 30120  
<http://www.wordsrus.info>  
EMAIL: [Sales@wordsrus.info](mailto:Sales@wordsrus.info)  
Tel. (678) 739-9177



## Overview

This dictionary contains over 32,000 terms that are specific to Computers and the Internet. Each term includes a definition / description. With more than 750 pages, this dictionary is one of the most comprehensive resources available.

Terms relate to applications, commands, functions, operating systems, image processing and networking. No other dictionary of computing terms even comes close to the breadth of this one. It is designed to be used by everyone from the novice seeking the most basic information to the mainframe systems programmer and MIS professional looking for sophisticated and hard-to-find information that's not available in most reference books. It's all here in one indispensable reference source.

- artificial intelligence
- computer-integrated manufacturing
- data communication
- databases
- distributed data processing
- fiber optics
- fundamental terms
- local area networks
- multimedia
- office automation
- open systems interconnection
- peripheral equipment
- personal computing
- processing units
- programming
- system development
- text processing

This dictionary is ideal not only for students of computing but for those studying the related fields of Information Technology, mathematics, physics, media communications, electronic engineering, and natural sciences.

We also publish a companion volume (Vol.2) of Computer Acronyms and Abbreviations with an additional 4,500 terms. Volume 2 also includes a section on file name extensions showing the most commonly used extensions and their association with various software systems.

This dictionary is available in more than 100 languages. See our website for pricing and availability.

[http://www.wordsrus.info/catalog/computer\\_dictionary.html](http://www.wordsrus.info/catalog/computer_dictionary.html)

Abbreviations used:

- (n) – noun
- (pn) – Proper Name
- (v) – verb
- (adj) – adjective
- (oth) – other

## Contents

Overview .....	3
Miscellaneous Terms .....	5
A ~ .....	6
B ~ .....	51
C ~ .....	79
D ~ .....	137
E ~ .....	175
F ~ .....	197
G ~ .....	218
H ~ .....	229
I ~ .....	243
J ~ .....	266
K ~ .....	270
L ~ .....	274
M ~ .....	292
N ~ .....	334
O ~ .....	353
P ~ .....	399
Q ~ .....	404
R ~ .....	439
S ~ .....	505
T ~ .....	531
U ~ .....	542
V ~ .....	558
W ~ .....	584
X ~ .....	587
Y ~ .....	587
Z ~ .....	587



performance of an algorithm. An algorithm or the method that implements it is said to be  $O(1)$  if its execution is a constant,  $O(\log n)$  if its execution is proportional to the natural log of  $n$ , or  $O(n)$  if its execution is proportional to  $n$ .  $O$  notation is frequently used as a way to compare the relative performance of operations on collections, in which case  $n$  is usually the number of elements in the collection.

**O/R Designer (PN)**~ A tool that provides a visual design surface for creating LINQ to SQL entity classes and associations (relationships) based on objects in a database. The Object Relational Designer also provides functionality to map stored procedures and functions to System.Data.Linq.DataContext methods for returning data and manipulating entity classes.

**OAB (n)**~ A snapshot of information from an address book that has been downloaded so that an Outlook user can access it while disconnected from the server. It contains the properties of a user, such as e-mail addresses, that Outlook requires to send an e-mail message and display information about the sender.

**OAB distribution (n)**~ The method by which the offline address book (OAB) can be accessed by users when they are working remotely or over a dial-up connection. To distribute the OAB, administrators can use Web-based distribution, public folder distribution, or both. Public folder distribution is required for Outlook 2003 or earlier clients.

**OAB distribution point (n)**~ The HTTP Web address or public folder where client computers can download an offline address book. Public folder distribution is required for Outlook 2003 or earlier clients.

**OAB generation (n)**~ The process by which Exchange Server creates and updates the OAB. Administrators can define when the OAB generation process will run.

**OAuth (PN)**~ An open standard for authorization which allows users to share resources stored on one site with another site without having to share credentials.

**OAuth Client ID (PN)**~ A type of client ID that allows Open Authorization (OAuth).

**OAuth token (n)**~ An entity encapsulating the security credentials for an OAuth login session.

**Obfuscation (n)**~ In software development, the deliberate act of creating obfuscated code to make it difficult for humans to understand and conceal its purpose or its logic, to prevent tampering, deter reverse engineering, or as a puzzle or recreational challenge for someone reading the source code.

**object (n)**~ An entity, such as a file, folder, shared folder, printer, or Active Directory object, described by a distinct, named set of attributes. For example, the attributes of a File object include its name, location, and size; the attributes of an Active Directory

User object might include the user's first name, last name, and e-mail address.

**object (n)**~ In object-oriented programming, an instance of a class. An object comprises data and methods that act on the data, and is treated as a discrete entity.

**object (n)**~ A table, chart, graphic, equation, or other form of information.

**object (n)**~ A drawing, image, shape, text, or other form of information that has been added to the artboard.

**object (n)**~ An entity within SQL Server.

**object (n)**~ An entity in a graph edge that the action is performed on.

**object anchor (n)**~ A format code in a desktop publishing or word processing document that keeps an element in the document, such as a figure or a caption or a label associated with the figure, in a certain position in the document. The anchored object is generally attached to another element in the document.

**object browser (PN)**~ A dialog box that displays information about objects, properties, methods, and constants in the current project and in referenced object libraries, and that you use to search for an element, get help on it, or paste it into a module.

**Object Browser (PN)**~ A dialog box that displays information about objects, properties, methods, and constants in the current project and in referenced object libraries, and that you use to search for an element, get help on it, or paste it into a module.

**object cache GC count (n)**~ The total number of objects from the object cache that have been removed from memory by the garbage collector.

**object cache live count (n)**~ The total number of objects that are currently active in the object cache. These objects can quickly be retrieved from the cache.

**object caching (n)**~ A type of caching that allows a reference to an object to remain accessible when the user navigates to a new domain.

**object class (n)**~ A category of objects that share a common set of characteristics.

**object construction (n)**~ A COM+ service available to .NET Framework classes that passes a persistent string value to a class instance on construction of the instance.

**object context (n)**~ The entity container defined in the conceptual schema. It contains a connection to the underlying data store and provides services such as change tracking.

**Object data type (n)**~ A fundamental data type representing any object that can be recognized by Visual Basic. Although you can declare any object variable as type Object, it is best to declare object variables according to their specific types.

**Object Dependencies pane (n)**~ A pane that shows objects that have a dependency on the selected object. Also shows objects on which

the selected object has dependencies.

**object dependency (n)**~ A dependency at the object level.

**Object Description Language (n)**~ A scripting language used to describe exposed libraries, objects, types, and interfaces. ODL scripts are compiled into type libraries by the MIDL compiler.

**Object Explorer (PN)**~ The section of the Visual Studio integrated development environment (IDE) that contains nodes that represent database objects.

**object file (n)**~ A file containing object code and/or data generated by a compiler or an assembler from the source code of a program.

**object flow (n)**~ In an activity diagram, a dashed arrow that indicates that an object is input or output by an action. An object flow arrow connects an object flow state either to a control flow arrow or to an action state.

**object flow state (n)**~ In an activity diagram, a state that defines an object flow between actions. An object flow state signifies the availability of an instance of a classifier in a given state, usually as the result of an operation.

**object frame (n)**~ A control you place on a form or report to contain an object.

**object handler (n)**~ A DLL loaded into an OLE container-ETM's process space that acts on behalf of a local server. The object handler responds to container requests that do not require the attention of the server, such as requests for drawing.

**object ID (n)**~ An attribute type in which the value of the attribute is a globally unique identifier (GUID) assigned by ILM2<sup>1</sup> to each resource when it is created.

**object identifier (n)**~ An attribute type in which the value of the attribute is a globally unique identifier (GUID) assigned by ILM2<sup>1</sup> to each resource when it is created.

**object in state (n)**~ In an activity diagram, an object that is manipulated by a number of successive activities. Each appearance of the object indicates a different phase in its life.

**object initializer (n)**~ A set of name/value pairs or unnamed variable initializers that are assigned to the public fields of an object in the same statement in which the object is declared.

**object layout (n)**~ The positioning of an object, such as a chart, shape, image or text box, on a page, along with related options such as text-wrapping styles.

**object library (n)**~ A file that contains definitions of objects and their methods and properties.

**object lifeline (n)**~ In a sequence diagram, a representation of the existence of an object at a particular time. If the object is created or destroyed during the time period the diagram represents, then the lifeline stops or starts at the appropriate point.

**object lifetime (n)**~ The span of time a cached object resides in cache and is available to be retrieved by cache clients. The



This dictionary contains over 32,000 terms that are specific to Computers and the Internet. Each term includes a definition / description. With more than 750 pages, this dictionary is one of the most comprehensive resources available.

Terms relate to applications, commands, functions, operating systems, image processing and networking.

No other dictionary of computing terms even comes close to the breadth of this one. It is designed to be used by everyone from the novice seeking the most basic information ... to the mainframe systems programmer and MIS professional looking for sophisticated and hard-to-find information that's not available in most reference books. It's all here in one indispensable reference source.

- artificial intelligence.
- computer-integrated manufacturing
- data communication
- databases
- fiber optics
- fundamental terms
- local area networks
- multimedia
- office automation
- open systems interconnection
- peripheral equipment
- personal computing
- processing units
- programming
- system development
- text processing

This dictionary is ideal not only for students of computing but for those studying the related fields of Information Technology, mathematics, physics, media communications, electronic engineering, and natural sciences.

ISBN 9781537264677



9 781537 264677

90000 >

