

O

Object An entity within an OO program or system which has methods (or internal code) that determines its behaviour in response to stimuli such as events and messages. Objects are the currency of OO systems.

(See *C++*, *Java* and *OO*.)

Object-based (See *C++*, *Java** and *OO*.)

Object factory A term sometimes used to describe software publishers, who offer objects such as ActiveX controls, Java applets, JavaBeans, Plug-ins or OLE objects.

(See *ActiveX**, *Java**, *Plug-in* and *OLE*.)

Object implementation A coded solution which dictates an object's behaviour and response to events. The code represents the object's methods.

(See *OO*.)

Object interface An object's outer conceptual layer, which intercepts messages and directs them appropriately. The layer is sometimes referred to as a shell. It is the first entity which an inbound message meets. The message may then be processed internally by the object's methods.

(See *Object** and *OO**.)

Object-oriented (See *C++*, *Java** and *OO*.)

Object-Oriented Database Management System (See *OODBC*.)

Object-oriented language (See *C++*, *Java** and *OO*.)

Object-oriented programming (See *C++*, *Java** and *OO*.)

Object-oriented programming language (See *C++*, *Java** and *OO*.)

Object-oriented UI

Object-oriented UI (*See OO user interface.*)

Object schema A structure that defines the interactions and relationship of objects in an OO system.

(*See OO*.*)

Object scraping A method of mapping data from a server to objects. The objects are used to perform transactions or other types of processing. It may be applied in an application renovation solution.

(*See Application renovation and Screen scraping.*)

Occam 1. A programming language intended for developing software targeted at parallel systems such as supercomputers. It is closely aligned with the Transputer model for parallel processing. Named after William of Occam (circa 1300 to circa 1349), Occam is based on Co-operating Sequential Processes (CSP), which were originated by Tony Hoare. CSP puts forward the *monitor* concept, which coordinates tasks. Put simply, executing tasks are placed in monitors which are then locked, thus preventing other tasks from entering them. When the task is complete the monitor is unlocked, and may be entered by other tasks. Java is the first mainstream programming language to feature such concurrent programming features. Java objects have monitors, whose state is determined by the keyword *synchronised*. This is one of Java's key attributes, but is rarely publicised. When synchronised, other threads cannot execute Java object methods that have begun executing as the result of a thread. In order to run such methods, threads must be able to obtain the object's monitor lock. (*See Java and MPP.*) 2. An ancient philosophical principle that states we should not presume more to exist than is absolutely necessary. In the guise of Occam's razor it loosely translates to: if you have two or more possible paths, then select the easiest. However, the convenient route in the short term often proves the most difficult later on.

OCR (Optical Character Recognition) A process that converts scanned images of text into text files.

Octet A term used to describe eight consecutive bits.

OCX A control or object which was a forerunner to ActiveX. OCX controls can be integrated into compatible applications, yielding functionality gains that may take the form of complete applications such as grammar checkers. OCX controls can be written using Visual C++.

(*See ActiveX*, C++ and Visual Basic.*)

ODBC (Open Database Connectivity) An internationally agreed standard, covering database connectivity. Most modern databases offer ODBC compliance.

(See DBMS.)

ODBC 3.0 SDK A set of tools, libraries and headers which can be used to integrate ODBC 3.0 connectivity access in Web sites.

(See Data warehouse and ODBC.)

OLAP (On-Line Analytical Processing) A data analysis technique used predominantly in the client/server computing environment. It is a decision-making support technique, which may be applied to interrogate data from disparate sources. The resulting data may also be analysed. OLAP implementations can be assumed to embody multidimensional data analysis techniques, and to integrate:

- an OLAP GUI for user communication
- OLAP analytical processing logic
- OLAP data processing logic.

OLAP empowers users to generate query data in order to answer complex questions based on what-if scenarios or on current and historical data. It is an advancement of the primitive querying techniques harnessed in RDBMS designs. These include Borland (now Inprise) QBE and even query languages such as the industry standard SQL.

(See Data warehouse, MOLAP and ROLAP.)

OLE1 and OLE2 (Object Linking and Embedding) An object architecture. It is a method by which one application may be linked with or embedded into another. An OLE server application is the underlying source of an OLE client application. Objects may be video, wave audio, speech synthesis, MIDI files, graphics or text. The objects may be shown in the client OLE document or application as an icon, and can be launched by double clicking that icon. OLE can be used to embed Windows Media Player into client applications to add voice or video annotations to documents. Using OLE1-compliant applications the process of embedding an object is more intensive than that associated with OLE2 applications. Object embedding is made easier using OLE2-compliant applications, because objects can simply be dragged from one application to another. An increasing number of Window applications are OLE2-compliant.

(See ActiveX.)*

OLE client

OLE client (Object Linking and Embedding client) An application which has an embedded object or application from an OLE server application.

(See Object and Visual Basic.)*

OLE DB SDK A Microsoft database access specification which bases itself on OLE and COM object architectures. It complies with SQL and non-SQL databases.

(See Database, ODBC and ODBC 3.0 SDK.)

OLE server (Object Linking and Embedding server) An application which provides an object for an OLE client application, providing a means of running that object from within the client application.

(See Object and Visual Basic.)*

Online state A state in which a computer or peripheral can perform its intended purpose. For example, a printer may print when it is online, and it is possible to communicate with a remote computer when it is online.

OO (Object-Orientated) A prefix used in object-oriented systems, software and development tools.

(See ActiveX, C++, Java, JavaBeans, Object and OOP.)*

OODBMS (Object-Oriented Database Management System.) A database used to store and to retrieve complete objects, including their code and their data. Stored objects may be categorised and stored in compound structures or objects. OODBMSes are characterised by their ability to:

- store complex objects
- be renovated or updated without radical renovation of data table structures associated with RDBMS implementations
- be extensible, providing a means of defining new data types
- support OO methodologies and concepts, including encapsulation, where objects' inner workings are hidden, and inheritance, where objects may be granted the methods and data of other objects. Multiple inheritance may also be supported where subclasses inherit methods and data from more than one superclass.

(See Data warehouse, DBMS, OO and OOP.)

OODL (Object-Orientated Dynamic Language) A programming language that is both object-oriented and dynamic, of which Dylan is a commercial example.

(See Dylan, Dynamic and OOP.)

OOL (Object-Oriented Language) A programming language which adheres to the object-oriented programming model.

(See C++, Java and OOP.)

OOP (object-oriented programming) A modular programming approach that depends upon reusable objects. OOP programming tools include Inprise Delphi, Optima++ and PowerSoft PowerBuilder. OOP languages include C++, Java and Visual Basic. In the real world we unconsciously place objects in classes. We know, for example, that cars, holiday chalets and computers are from different classes, but each time we see a car we don't ask ourselves: which class does a car belong to? Or why is it different from a holiday chalet? We know that it is a member of the class of vehicles because we have learned how it behaves, and that behaviour, with all its methods, is in our mind. We do not have to learn or consider an object's behaviour each time we come in contact with it. For example, you know that you cannot drive the holiday chalet because of its behaviour and the class to which it belongs. You know these things without having to repeatedly decide that a holiday chalet cannot be driven because it has no wheels, no axle, no engine and so on. Object-oriented programs are much the same. Classes of objects are carefully defined. If the class `HolidayChalet` were defined appropriately, we could not drive it by using, for example, the following command:

```
HolidayChalet.drive()
```

Only the correctly defined class, `Car`, could be made to drive using the command:

```
Car.drive()
```

Hierarchies form another important part of classes where, once again, as in the real world, classes are subdivided into further classes. This helps distinguish between, say, a sports car and a jeep. The jeep would be a member of the class `OffRoadVehicles`, which is a member of the class `Cars`, which in turn is a member of the class `Vehicles`. This additional information tells us that a sports car cannot be driven up a steep muddy slope etc. Everything in the real world is a member of a class, of which there are an infinite number. Some Windows databases come with a number of in-built methods to choose from. These cover standard activities such as opening tables and forms and even opening the Help window. This type of database building is achieved through a so-called 'pick and build' interface. The OOP model embraces:

- data hiding
- encapsulation
- reuse
- polymorphism.

OOUI

OO languages include:

- Java
- C++
- Smalltalk
- Visual Basic.

(See C++, Data hiding, Dylan, Encapsulation, Java, Parallel programming and Polymorphism and Smalltalk.)

OOUI (Object-oriented user interface) *(See OO user interface.)*

OO user interface A user interface which uses the object model as its underlying interface components. They are typically graphical user interfaces (GUIs or 'gooeys'). The Apple Macintosh is remembered as one of the first systems to feature a commercially successful OO UI, followed by NeXTStep, which was founded by Apple Computer's co-founder Steve Jobs. This was followed by the Microsoft Windows and IBM OS/2 OSes, which featured OO user interfaces.

(See ActiveX, C++, Java*, Object*, OO*, UI builder and Windows.)*

OpenDoc A standard object or component architecture initially aimed at the creation of compound documents. It is not as widespread as OLE or Microsoft ActiveX component technology.

(See ActiveX, Compound document, JavaBeans and OLE.)*

OpenGL A 3-D graphics language. OpenGL is remembered as having started life in the 1980s as IRIS GL, a graphics programming interface for Silicon Graphics workstations. It provides foundation components for the development of applications that base themselves largely upon 3-D graphics. It contains 120 functions to draw 2-D and 3-D images. Among other things, OpenGL provides developers with the necessary high-level programming components to shade polygons, texture-map or colour polygons with bit-maps, light scenes, create animations, perform atmospheric effects, and to depth cue or dim distant objects.

(See VRML.)

Open Inventor A 3-D developer's toolkit based upon OpenGL. It provides a library of C++ objects and methods, including geometric primitives, routines that interface objects with data entry devices, and tangible elements for lighting. Open Inventor is licensed to several companies including Template Graphics Software and Portable Graphics. Open Inventor has an ASCII file format.

(See VRML.)

OpenScript A programming language included in many versions of Asymetrix ToolBook. It fits into the same programming language category as Lingo, but is dedicated to the ToolBook environment and is not used as widely.

Coding a ToolBook presentation using OpenScript ToolBook applications are based on the metaphor of a book. Books are applications, and are divided into pages which are displayed in windows called viewers. Pages can have objects such as fields, buttons and graphics. Such objects can be shared among pages by placing them in a background that may be common to two or more pages. Usually each page is used for each screen in your application. For example, each definition in an encyclopaedia might be shown on an individual page. Equally, each record in a form-based database application may be confined to a single page. An authentic object-oriented environment, the buttons, fields, graphics, pages and backgrounds that make up a book are all objects. The properties of an object determine both its appearance and its response to events such as a mouse click. There is an array of ready-made objects in the tool palette. The tool palette lets you implant a variety of different objects including buttons, fields, lines, rectangles, polygons and ellipses. ToolBook operates in two basic modes: Author level and Reader level. The Author level mode provides access to ToolBook's drawing and programming tools. It lets you create books, create and modify page objects, and program in OpenScript. The Reader level mode provides all necessary features to run ToolBook applications, but does not provide access to the development tools. It lets users:

- navigate through pages and add pages
- type, edit and format text in fields
- print
- run OpenScript programs.

To prevent users changing ToolBook applications you can use the runtime version of ToolBook. While authoring an application you can switch between Author level and Reader level instantly in order to test applications under development. You toggle between Reader and Author levels either by selecting Reader or Author from the Edit menu or by pressing F3.

Scripts A typical OpenScript is shown below. It plays a MIDI file and shows a page of a video file in a stage. Take for example the script for page 1, which is shown below:

```
to handle enterPage
  mmOpen clip 'intro'
  mmShow clip 'intro' in stage 'intro'
  mmPlay clip 'music'
end
```

Like all the scripts associated with page objects this event handler is activated by simply opening, or moving to, the page number. The second line simply opens the video clip stored as `intro` in the Clip Manager. With the `intro` clip open the third line shows the first page, or a current page, of the `intro` video file. Finally, the fourth line plays a MIDI file clip stored as `music` in the Clip Manager. All the

Operating system

page scripts operate in the same way, except they have extra statements to close MIDI file clips. The button script below simply closes a MIDI file clip that is played by the page script and plays the video file in the stage. This requires just two lines of code using the `mmClose` and `mmPlay` commands:

```
to handle buttonClick
  mmClose clip 'music'
  mmPlay clip 'intro' in stage 'intro'
end
```

The script below illustrates a Pause button. It requires an additional `if...then` structure to determine whether or not the video clip is playing. This has to be included because an attempt to pause a clip that is not playing causes an application to crash.

```
to handle buttonClick
  if mmStatus of clip 'intro' = 'playing' then
    mmPause clip 'intro'
  end if
end
```

The status of any clip can be obtained by using the `mmStatus` command.

(See *Authorware Professional, IconAuthor, Lingo and ToolBook.*)

Operating system (OS) A generic term used to describe the software elements that manage system resources and so provide an interface between the user and the system, as well as between software and the system. The shell, user interface or front-end is sensitive to a number of user commands. Popular operating systems include Windows 95/98/NT, OS/2 Warp, MS-DOS, DR-DOS, OS/2 and Unix.

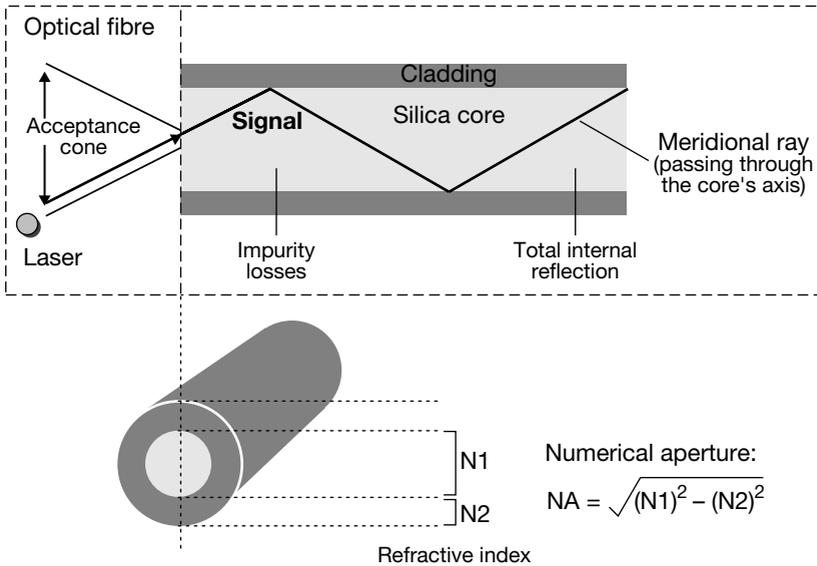
(See *Windows.*)

OPIE (One Time Passwords in Everything) (See *OTP.*)

Optical fibre A silica-based fibre that can propagate a light signal while inducing minimal losses. Light propagation is driven by total internal reflection. This is made possible using a core fibre and surrounding cladding of different refractive indexes. The light source must emit light into the cable at the critical angle in order to achieve total internal reflection. Applications include:

- lightwave communications
- flex sensing in gloves and bodysuits, which are used in VR.

Numerous different types of optical fibre exist, including graded and step-index. A step index is one in which the core silica is of one refractive index and its coating silica is of another index. A graded fibre consists of a core fibre that is coated with a number of grades of silica of differing refractive index. The advantages of optical fibre include:



Optical fibre – *monomode/single mode*

- low weight and ease of installation.
- immunity to electrical and reasonable levels of electromagnetic interference
- exceptionally wide bandwidth when compared to electrical conductors
- cost-effectiveness

The operation relies upon total internal reflection, given by reflecting injected rays in the cladding. The core and the cladding, therefore, have a different refractive index. The angle at which rays are injected into a fibre is critical in order to achieve total internal reflection and to propagate the ray appropriately. The numerical aperture (NA) of a fibre is a measure of the size of its acceptance cone or the range of angles at which rays must be injected. Propagated rays may be:

- meridional, which repeatedly intersect the core's axis
- skew, which spiral through the core without ever intersecting the axis.

Their launch angle tends to be greater than that of meridional rays.

The light source must be an LED or laser device which lases at an appropriate wavelength. A multimode step-index fibre may have a core diameter of ≥ 50 micrometres and an NA of the order of 0.15 to 0.4. They are able to propagate a substantial amount of emitted light from an LED. Injected light is dispersed into many thousands of paths called modes. A monomode step-index fibre has a smaller core and an equally low NA (which is typically 0.1).

(See LED.)

Optical medium

Optical medium A generic term used to describe a medium which is read using an optical read head. CD variants are optical media, and include Audio CD, Philips CD-I discs, CD-ROM, CD-V, CD-ROM XA, Photo CD, Video CD and DVD. Other optical media include LaserVision and Sony Mini disc.

(See CD-ROM and DVD.)

Oracle A database development environment produced by the company of the same name.

Oracle Media Objects An object-based multimedia authoring tool.

Orange Book A set of security standards developed by the United States Department of Defense, and formally named the Trusted Computer Standards Evaluation Criteria.

(See Security.)

OS/2 (Operating System/2) A PC operating system. Launched in mid-1987 to coincide with the release of the IBM PS/2 family of computers, it is a multitasking operating system able to run applications simultaneously, and is downwardly compatible with DOS. OS/2 is used in conjunction with Presentation Manager, a GUI. Relaunched in 1992 by IBM under the name OS/2 2.0, it is now a relatively successful 32 bit operating system. It is also able to run Microsoft Windows applications in either real or standard modes. The release of OS/2 2.1 in mid-1993 saw the inclusion of multimedia extensions, which compare with those of Windows.

OS/9 (Operating System/9) An operating system originally developed for the Motorola 6809 processor in the late 1970s, and used widely on micro-computers based on the 6809 and subsequent processors of the Motorola family. The Tandy Colour Computer, Atari ST and Dragon 32/64 were all able to run OS/9. Because CD-I is based on a Motorola processor, OS/9 was adopted and modified for Philips' purposes, calling it CD-RTOS (Compact Disc-Real-Time Operating System).

OSI (Open System Interconnection) A seven-layer industry-standard reference model which is applied extensively to client/server architectures and was introduced in 1984 by the ISO (International Organization for Standardization). It provides a standard infrastructure for the applications, glues and communications required of modern client/server implementations. The seven layers include the following.

- *Application*, which encompasses client- and server-side programs, such as e-mail clients and browsers at the front end.

- *Presentation*, which is the formatting layer, delivering such operations as protocol conversion and compression. A typical application sees clients' SQL requests converted to a format that complies with the SQL server.
- *Session*, which permits a conversation between programs, objects or processes.
- *Transport*, which provides error detection and correction operations for communicated data, and adds a transport layer ID.
- *Network*, which operates to break down transmitted data into packets (with sequence numbers), and to reassemble them into a readable message on reception. It may be assumed to route packets to an appropriate node.
- *Data-link*, which receives packets from the network layer and adds control information to their headers and trailing regions. The resulting frames are passed to the physical layer when appropriate access is detected.
- *Physical*, which converts frames into binary data so that it may be transmitted, and returns this data to frames upon reception at its intended destination.

(See *Application, Client/server* and Glue.*)

OSPF (Open Shortest Path First) A protocol used in routers.

OTP (One-Time Password) A password protection security policy to prevent illegal access. In many instances it does not prevent hackers from monitoring the network and gaining access to information. OTP variants include:

- Wietse Venema's LogDaemon
- Bellcore's S/KEY Version 1.0
- BellCore's Commercial S/KEY Version 2.0
- United States Research Laboratory's (NRL) One Time Passwords in Everything (OPIE).

(See *Encryption.*)

Overlay A screen image laid on top of another. It often applies to mixing computer-generated images with video.

OzEmail An Australian ISP that also operates as an international telecommunications company using Internet Telephony, which it calls *OzEmail Phone*.

(See *Internet telephony.*)