

- [Geoplace Home](#)
- [Ad Index](#)
- [Advertising](#)
- [Archives](#)
- [Bookstore](#)
- [Classifieds](#)
- [Contact Us](#)
- [Events](#)
- [Forum](#)
- [GeoDirectory](#)



- [GeoNews](#)
- [Help](#)
- [Links](#)
- [MarketPlace](#)
- [Publications](#)
- [Research](#)
- [Services](#)
- [Site Map](#)
- [Subscriptions](#)
- [Weather](#)

Copyright © 1999 Adams Business Media, Inc.
All Rights Reserved. Reproduction Prohibited.

[Would You Like A Reprint of this Article?](#)
[CLICK HERE!](#)

technology trends

The objects of our desire



GeoEurope investigates the benefits that objects bring to our mapping systems.

Object-oriented programming: everyone's talking about it, but how many people know what it is? With most computing fashions, there are normally obvious benefits for users, but what exactly do we stand to gain from this most esoteric of disciplines? The answer is far more than we could expect: interoperability between different pieces of software, the ability to customise and add to existing software with ease, and even the addition of "intelligence" to gis programs.

So what is in an object? Loosely, it is a portion of computer code combined with data. For example, a piece of code designed to draw a line-feature six centimetres long on a map could be an object. Indeed, the line itself could be an object. An object has "methods" associated with it. These are really just things it can do, such as draw a line. Objects can exchange messages asking each other to perform methods. An object for creating buffer zones could ask the line-drawing object to make a line as part of the buffer zone. Lastly, objects have "behaviours". Change the scale on your map to a tenth of its size and object-oriented features will redraw themselves as appropriate or even exhibit more

Columns

**DANGLING SEGMENT
SUSTAINING SPATIAL
MOMENTUM**

GEO JOBS
Manuel Pallage,
commercial director of
Star Informatic, tells all
to *GeoEurope*.

**TECHNOLOGY
TRENDS
THE OBJECTS OF OUR
DESIRE**

reviews
SOFTWARE REVIEW

News

[euro news](#)
[business news](#)
[people news](#)
[new business forum](#)
[product news](#)
[calendar](#)
[notice board](#)

Listings
[products & services](#)

complicated behaviour: imagine a name label that shrunk only slightly but moved so that it didn't cover any other features (Laser-Scan has a system that does something similar).

Big deal. On the face of it, none of this is much to get excited about. So what's the fuss?

Object appeal

The main appeal of objects is that you don't need to know how they work to use them. You don't need to know how the line-drawing object draws the line, only that it can. Provided you know what an object can do and how to send messages to it, you can use it. The result is that once you've created an object, it's very easy to combine it with other objects to create whole programs or to manipulate and store multi-dimensional data such as feature geometry. It's even possible to combine objects on different computers using a network. So, if your gis program were to contain an object for accessing features in certain file formats, another program on another computer networked to yours, perhaps via the Internet, could access and possibly change the features in these files, even if it didn't understand the same file formats. At the far end of the scale is the possibility of creating "intelligent agents" – applications that can try to understand your data or you! Tests and even some practical applications are in progress at various sites throughout Europe.

The problem is getting objects to talk to one another: they need to have a common language. This is where the gis heaven starts to fall apart. There are a variety of ways for objects to communicate with one another, all incompatible with each other. JavaBeans, dcom (aka ActiveX aka ole/com) and corba are the main object standards and each has virtues that appeal to developers.

Microsoft's com (Component Object Model) is built into Windows 98 and nt (now called Windows 2000) and is becoming the de facto standard for the desktop. Microsoft Office is a collection of com objects, as are Intergraph's GeoMedia, Alsoft's Geo Concept, and Innogistic's Cartology among others. A basic example of the advantages offered by object-oriented technology is the ability to embed a gis map in an Office document and still be able to edit it within the gis.

At the moment, com only exists under Windows, although Microsoft is developing a unix version of the technology. This means that if you want to develop for any other platforms, you'll be unable to take advantage of com. The advantages offered by object-oriented programming, together with Windows' dominance of the desktop market and the inroads Windows 2000 is making into the GIS market, are making this prospect attractive even to unix developers such as esri.

According to David Maquire, formerly

According to David Maguire, formerly managing director of esri's UK office, "If the development environments, consistent interface, and compilers were available for Java, we'd develop using that as well. com is now a mature object-oriented technology, and Java might well be in three years' time. At the moment, it doesn't have the capabilities we need." That said, the company has been working on a Java version of its MapObjects collection of com objects.

This preference for com is echoed by David Swinburne, European product manager for MapInfo. "com on the desktop, Java on the server. We're only developing for Windows on the desktop so we don't need Java's cross-platform capabilities there."

Java's advantage

Java's main advantage over com is that it will run on any computer, provided that system has a "virtual machine" - a program that can run Java applications or "applets". This virtual machine can be part of the computer's operating system, included within a Web browser such as Netscape Communicator or Internet Explorer, or included within another program such as a gis or cad system. In this regard, within the gis industry, the main exponents of Java are the cad companies, Bentley Systems and Autodesk. Both bundle Java virtual machines with their own packages, MicroStation (now re-titled MicroStation/J in acknowledgement of the development) and Autocad respectively.

Autocad doesn't have quite the Java integration of MicroStation/J, the former simply being able to run Java programs from its AutoLisp prompt line. MicroStation/J has an entire set of Java objects that more or less duplicate MicroStation's capabilities; you can access these objects from your own Java programs. Since many other programs, including the enterprise resource planning system SAP, now offer similar capabilities, you can use Java applets to link MicroStation/J to these programs, without having to rewrite or recompile your applet to take account of different operating systems at either end. Yoav Etiel, senior vice president at Bentley, describes this as "enterprise engineering".

The more ubiquitous use of Java is on the Internet. While gis companies can gamble that their customers will be Windows users, the Web offers an equal playing field for unix and Mac OS users. Even if they could, ActiveX (or Distributed com), the networked version of com, has proven unpopular and a potential security risk to unsuspecting Web surfers, so Java has become the main standard for networked applications.

Autodesk's forthcoming MapGuide 4.0 shows how even those committed to a Windows desktop can change their policy for the more open Web. MapGuide 3.0 is available on the

Open Web. MapGuide 3.0 is available on the client-side only as a plug-in for browsers running on Windows 95, 98 and NT. MapGuide 4.0 will enable servers to send a Java applet to browsers so they can process the same data as browsers using the plug-in, no matter what platform they're running on. Joe Astroth, Autodesk's vice-president for gis, explains, "We chose to develop superior technology for pc users first and then broaden our audience reach with a Java viewer."

Two-pronged approach

Since no one format is dominating both Web and desktop, the Opengis Consortium is wisely choosing to set standards for both Java (via corba, another networked object system which has proved of limited popularity) and dcom. While gis programs based on Java objects (known as JavaBeans) will not be able to talk to those based on dcom objects or vice versa since their messaging systems are different, they will be able to talk to programs based on the same object formats and Opengis standards. At the moment, the Consortium is still working on these standards, although specifics have already been approved for letting objects tell each other how to manipulate simple features such as lines. The first programs to support these initial standards are almost ready to appear on the market.

As more companies move towards using object-oriented technology in their programs and further Opengis standards are approved, the life of the average gis user will be improved by the removal of unnecessary obstacles caused by the "monolithic" programming techniques used to create yesterday's and some of today's software. Objects won't be the total panacea some people believe they are, but they will at least smooth the way to creating an environment where you don't have to keep employing more technology to overcome the limitations of your existing technology.

Copyright © 1999 Adams Business Media, Inc.
All Rights Reserved. Reproduction Prohibited.

Would You Like A Reprint of this Article?
[CLICK HERE!](#)

	If you have questions or suggestions for the GeoPlace Web Site, please let us know, email: geomgr@geoplace.com	
--	--	--

© 1996-1999 GeoTec Media All rights reserved.
Any unauthorized reproduction of site contents is expressly prohibited.
Usage constitutes an agreement to abide by the [GeoPlace.com User Agreement](#).

To obtain reprints or subscriptions, click here

An **Adams Business Media Company**.
2101 South Arlington Heights Road
Arlington Heights, IL 60005-4142
phone - 847.427.9512

fax - 847.427.2006