Making Reuse a Reality — Introducing the SOMobjects Developer Toolkit

Object oriented (OO) technology represents a fundamental change in the concept of software development. The promise of OO technology is for software objects — segments of code combining data and procedures — to become standard, off–the–shelf parts. A programmer can then incorporate these objects in a software system as easily as an engineer incorporates an off–the–shelf semiconductor chip in a circuit board design.

The Future is Now

Creating interchangeable, reusable software components is very much a reality today: you can choose from a variety of object oriented programming languages, design and analysis aids, and visual programming tools. You can also find OO enablers and frameworks built into the latest releases of some operating systems, including IBM Operating System/2® (OS/2®) Versions 2.0 and 2.1. What you haven't been able to find — until now — are some key pieces needed to complete a system that works as smoothly and openly as the hardware development process.

Now, SOMobjects™ tools fill in some of these missing pieces by breaking objects free from ties to a specific programming language. On a single machine, the System Object Model (SOM) provides an object–structured protocol that allows applications to access and use objects and object definitions, regardless of what programming language created them, with no need to recompile the application. SOM's language–neutral character not only allows robust software objects to be easily used and reused wherever they're needed, it also enables a greater degree of openness than ever before in the development and use of object oriented programming (OOP) facilities across multiple operating platforms.

What's more, SOMobjects incorporates Distributed SOM (DSOM) technology that provides a base for OOP development and use over entire networks. With the IBM SOMobjects Developer Toolkit Version 2.0, programmers can start taking advantage of SOM and DSOM immediately. SOMobjects is available for both the OS/2 2.0 (or higher) and the IBM AIX/6000™ 3.2 (or higher) operating systems and is planned for the Windows™ environment. Complying with industrywide standards of the Object Management Group's (OMG®) Common Object Request Broker Architecture (CORBA), SOMobjects provides an extensive set of facilities for putting the power and speed of OO technology to work for you.

Highlights of SOMobjects

IBM's SOMobjects gives professional programmers a comprehensive toolkit for creating OO class libraries and instances of those classes — that is, objects that belong to particular classes — using SOM and DSOM technology. Among the product's powerful functions and benefits are the following:

- **Language–neutral support** for the definition, construction and use of OO systems — by defining an application programming interface (API) to SOM objects based on simple procedure calls — allows applications written in different programming languages to use a common class library.
- **Upward binary compatibility** of SOM class libraries, as dynamic link libraries (DLLs) that can be replaced without requiring recompilation of application source code, makes it easy for end users to install new releases of system libraries.
- **SOM/IDL Compiler** is key to language neutrality and supports the OMG standard Interface Definition Language (IDL). IDL gives individual object implementations the means to tell potential clients what operations are available and how they should be invoked. By mapping the public and private types for a given SOM class to language–specific bindings, the compiler allows SOM objects to be used by different programming languages.

- **Distributed SOM** supports OMG's CORBA standards: • Workstation DSOM provides support for cross–process and cross–address transparent access to objects in a single–system environment. • Workgroup DSOM enables you to define or access objects remotely in a distributed workgroup local area network (LAN) configuration across supported platforms (e.g., OS/2 to AIX/6000).
- **Language bindings for C and C++** allow programmers in those languages to use SOM objects and create new SOM object classes with ease. Other SOM bindings may be available from language vendors.
- **Replication Framework** makes copies of a single object available concurrently to multiple clients, and maintains consistency among the copies, with updates to any one copy automatically reflected in all other copies. It's especially useful for collaborative applications when you need updated information instantly. Fault tolerance is provided to ensure object integrity in the event of a node failure.
- **Persistence Framework** allows you to save and restore SOM objects to and from a repository that can be a file system, database, or object database. You can also change to another format (e.g., Lotus® to Excel®).
- **Emitter Framework** that produces an output file representing part or all of an object interface definition — structured to minimize effort, improve quality, and simplify maintenance — makes it easy to develop additional language bindings for SOM.
- **Collection Classes Framework** gives programmers such frequently needed data structures as lists, sets, queues, and dictionaries, and lets them inherit from and use these SOM classes in applications, with no need to recode or retest the functions.
- **Workstation Enabler**, for OS/2 or AIX/6000, enables the execution of SOM–based applications in a single–machine environment.
- **Workgroup Enabler**, for OS/2 or AIX/6000, enables the execution of SOM–based applications across a multiple–node workgroup LAN distributed environment.

What You Get

IBM's SOMobjects offers you several packages for the development and execution of SOM–based applications in OS/2 and AIX/6000 environments. Each package is available separately.

- **SOMobjects Developer Toolkit** Version 2.0 includes 3.5–inch diskettes containing the basic SOM kernel (SOM/IDL Compiler and language bindings for C and C++) and all of the frameworks, including Distributed SOM (DSOM). SOMobjects documentation is provided in hardcopy with the SOMobjects Developer Toolkit, along with the product license agreement. [Part number 96F8647 for OS/2; part number 96F8648 for AIX]
- **SOMobjects Workstation Enabler** Version 2.0, for SOMobjects applications operating on a single machine, includes the following: • SOM kernel runtime • DSOM runtime • Persistence runtime • Replication runtime • Collection Classes runtime • 3.5–inch diskettes containing the runtimes, as well as a hardcopy installation/configuration guide and the product license agreement. [96F8673 for OS/2; 96F8674 for AIX]
- **SOMobjects Workgroup Enabler** Version 2.0 includes 3.5–inch diskettes containing all of the runtimes listed above for the Workstation Enabler, but geared for SOMobjects applications operating across a multi–node LAN. A hardcopy installation/configuration guide and the product license agreement are also included. [96F8675 for OS/2; 96F8676 for AIX]
- **SOMobjects Documentation** is included with the SOMobjects Developer Toolkit. This set of hardcopy publications is also available separately, as the SOMobjects Developer Toolkit Publications Version 2.0 package. [Part number 96F8649]

Ordering and Training Information

- U.S. 1–800–342–6672
  Training 1–800–426–8322
- Canada 1–800–465–7999
  Training 1–800–661–2131