softwaremag.com

Building

ADVANCED APPLICATIONS

UML Models E-Business

How to achieve faster time to market, higher quality, and greater return on investment by adhering to OMG's latest UML and XML software standards.

A SPONSORED SUPPLEMENT TO
SOFTWARE MAGAZINE®
Field Guide for IT Managers Tackling Business Priorities

nterprise integration | Ma
ethodology framework XM
Enterprise portals | Mainfra
nethodology | **XML** | iPortal Appl
Orbix | B2B integration | Enterprise portals |
Enterprise integration | **B2Bi** | Mainframe inte
Orbix | B2B integration | Mainframe integratic
ramework | B2Bi XML/Web services | iPortal
egration | **iPortal** | Mainframe integration | La
odology framework | iPortal
ration | Large architecture |
tion | **J2EE** | Time to marke
/Web services | Orbix | B2B
ion | **Time to market** | Large
odology framework | J2EE |
ix | **EJB** | XML/Web service
ration | Large architecture/m
es | **iPortal** | Orbix | **EAI** |
ion | Mainframe integration
odology framework | Time to
Application Server | Orbix |
ion | Mainframe integration
gration | **End-to-end** | Mainf
odology framework | XML/W
B integration | Enterprise por
frame | **CORBA** | Enterprise
| B2B integration | Orbix
ration | **XML** | iPortal Applic
Enterprise integration | Mair
www.iona.com/one

One vendor. One platform.

IONA®

Total Business Integration™

UML Models e-business

EVERY DAY, IN HUNDREDS OF THE LARGEST companies worldwide, software developers use tools like Rational Software's Rose, Computer Associates' Paradigm Plus, and Microsoft's Visual Modeler to plan and design software applications. In all these cases, the developers use the Unified Modeling Language (UML) to create virtual models for the software systems they plan to build.

Companies model complex software applications for the same reason they hire architects to design buildings before actually having them built. In the long run, a detailed blueprint saves time and money. It allows developers to consider alternatives, select the best option, work out details, and achieve agreement before anyone starts building the application. It's much less costly to use a model than it is to modify a building or an e-commerce application after it has been assembled. More importantly, a good model documents the application's structure and simplifies modifying it later. This is critical when you consider that 90% of the costs involved in large applications occur as they are changed, extended, and otherwise maintained.

UML is an open standards success story of the late '90s. In three years, a dozen object-oriented modeling methodologies were replaced by UML, which is currently supported by all major software modeling tools. The Object Management Group (OMG) drove the rapid transition to UML and continues to maintain it as the world's premier open, object-oriented analysis and design standard.

As companies work to develop Internet portals, e-commerce, and business-to-business applications, they use UML models to drive their designs and maintain consistency. UML turns out to be a key ingredient in the new model-driven development approach to software development proving so popular with e-business developers.

At the same time, UML has served as the basis for another software breakthrough—a universal approach to metamodel integration. Companies will soon use metamodels derived from OMG's work with UML to automatically integrate information about many types of applications that use XML.

Established as the premier software specification language, UML is now underpinning OMG's Model Driven Architecture.

Standardizing a Solution

In 1996 OMG announced it was interested in creating an open, standard object-oriented notation and called for proposals. Rational Software, Cupertino, Calif., submitted UML Version 1.0, which had been developed by Booch, Rumbaugh, and Jacobson. Ultimately, 21 other companies sent proposals to OMG. In 1997, the submitting companies blended their work into a seamless specification that covered most user and vendor needs. OMG's version of UML represented the best efforts of the teams from such companies as Hewlett Packard Co., Palo Alto, Calif.; IBM Corp., Armonk, N.Y.; Microsoft Corp., Redmond, Wash.; Oracle Corp., Redwood Shores, Calif.; Ptech Inc., Boston; Reich Technologies, Brest, France; Softeam, Baltimore; and Taskon, Trondheim, Norway. The OMG board approved the resulting UML Version 1.1 specification in November 1997. Since then, OMG has managed UML as an open standard. An OMG task force gath-

By Paul Harmon

IONA Targets Total Integration

IONA®, Waltham, Mass., is the leading e-business platform provider for Total Business Integration™ and markets IONA Suite, a set of integrated infrastructure products that supports the design, development, implementation, and management of e-business Internet applications and other large-scale, distributed computing projects. IONA Suite provides the core application deployment, integration, management, and Web service components needed by IT administrators and is based on IONA's proven distributed computing experience.

IONA Global Services, the company's consulting arm, provides IT solutions aimed at helping customers achieve total business integration. From its network of locations worldwide, IONA Global Services helps customers design, build, and integrate their internal applications and systems with supplier, partner, and customer systems.

IONA Global Services excel in virtually all leading standards-based software technologies such as CORBA, Java, EJB, COM+, and IONA's own suite of products, including iPortal Integrator, iPortal OS/390 Server, iPortal Server, iPortal SureTrack, Orbix, iPortal Application Server, and XML Bus.

IONA's Global Services are designed to provide business value by helping companies define overall IT strategies and select and deploy enterprise-scale software applications as well as platforms and toolsets. IONA's services also address knowledge transfer and education on tools and industry best practices. IONA's services are organized to aid strategic IT decision makers, project and business managers, and developers. These areas are staffed with highly experienced consultants who can deliver sophisticated, Total Business Integration solutions to any business.

For more information on IONA products or Global Services, visit **www.iona.com.**

By Supplement Staff

ers information about problems and improvements, and also schedules revisions.

OMG's specification defines UML as "a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system."

Because UML is an open standard, anyone can download the complete specification from OMG's Web site free of charge. Similarly, OMG controls the process of maintaining and extending UML, and any company can join the standards organization.

Unlike the earlier methods that combined a graphical notation system and a procedural methodology, UML doesn't include a software development methodology. When OMG issued its call for proposals, it indicated it thought it was too early to try to standardize an object-oriented methodology; the steps involved in developing object-oriented systems vary too greatly. Developers creating a transaction processing system follow very different steps from those developing a real-time embedded system, for example. OMG suggested it would standardize on a notation first and put off an effort to standardize a procedural methodology.

OMG's UML is based on a common UML metamodel. The UML metamodel is, in effect, a class diagram and a set of semantic and syntactic rules that defines the core elements and relationships used in UML. As such, the metamodel derives all other terms from that core set of elements and relationships. Varying application development efforts call for differing diagrams.

The UML metamodel ensures that UML is more systematic and internally consistent than any earlier software modeling technology. At the same time, it means that tool vendors know exactly how to transform an element in a class diagram into its equivalent element in a sequence diagram or a state diagram, and so forth.

In addition to core symbols, the metamodel contains supplementary symbols, called extensions. Extensions are usually represented by graphical adornments added to existing core symbols. For example, an arrowhead added to a line indicates the relationship's direction. Also featured are stereotypes, which are created by those interested in customizing UML. The name of a stereotyped symbol is always put between guillemets (e.g., <<stereotype>>). Using stereotypes, an industry group can extend UML to incorporate some special-purpose notational system. If the group follows the rules, and stereotypes are derived from core UML elements,

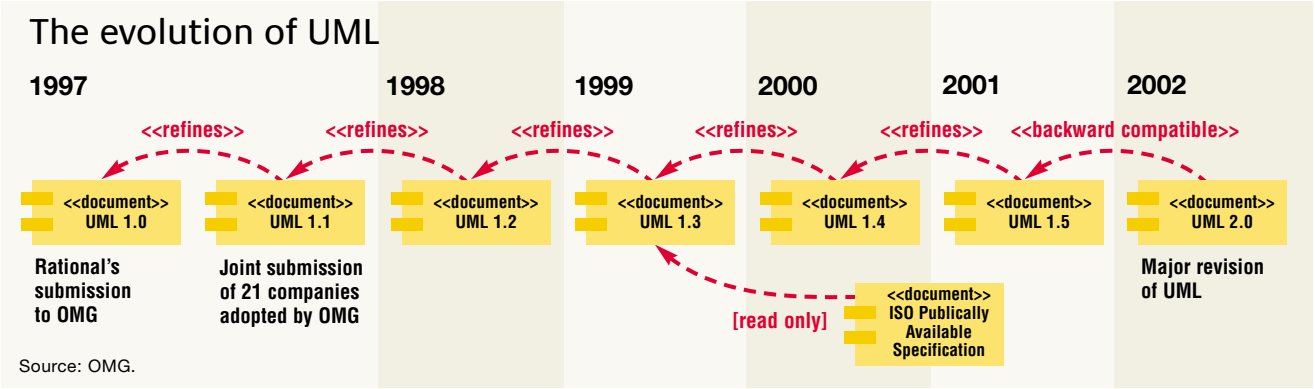


Fig. 1. The OMG revision task force process was designed to guarantee that the language evolved systematically and that changes were made at specified intervals.

they can be incorporated into UML diagrams and used by UML tools. An extensive package of stereotypes is referred to as a UML profile. Any introductory book on UML will introduce all the core symbols and most extensions.

UML allows developers to specify constraints on any diagram by describing the constraint in ordinary language and placing it between braces; for example, {constraint}. The OMG Analysis and Design task force selected the Object Constraint Language (OCL), because it gives developers a formal notation with which they can define complex sets of constraints. OCL is based on the "design by contract" approach to object-oriented design and calls for the specification of preconditions, postconditions and invariants an object must fulfill. The actual constraints are specified by rules.

OMG has provided for the automatic generation of Interface Definition Language (IDL) code from UML diagrams. IDL is OMG's language-neutral way of specifying object or component interfaces. Common languages such as Java, C++, and Visual Basic are mapped to IDL. Thus, with an appropriate tool, developers can begin with UML diagrams, generate IDL descriptions of the diagrams, and then use IDL to generate a description of the application in a regular computer language. Likewise, common data structure formats like XML can be automatically generated from UML diagrams.

Once the OMG revision task force process was in place, it decided on a schedule that would guarantee that the language evolved systematically, incorporating minor changes frequently and major changes only at intervals that would enable developers and tools vendors to keep up with the changes. (See Fig. 1.)

Beyond Software Architecture

E-commerce and supply chain integration systems are complex and are developed in object-oriented languages like Visual Basic 6.0 or Java. They rely on components, are distributed in multiple tiers, and often require application servers like WebLogic from BEA Systems Inc., San Jose, Calif.; WebSphere from IBM; iPortal from Iona Technologies, Waltham, Mass.; and iPlanet from Sun Microsystems, Inc., Palo Alto, Calif. They also must be integrated with existing applications. In many cases, the applications spread across several companies, run different platforms, and use legacy applications written in different languages.

To build these systems, developers must specify software architectures that provide an overview of the systems and interfaces the elements will use to facilitate communication. Companies also need to develop component middleware and infrastructure to support locating components and passing messages between diverse languages and database formats.

Increasingly, sophisticated companies are going beyond simply specifying a software architecture. They are creating business models that provide a framework for all corporate applications. The model describes the major components used in the business and how they relate to each other and to the company's computer infrastructure. If the model is specified in UML, as it is in most popular visual modeling tools, corporate developers can later use the UML diagrams and the visual modeling tool's

From its beginnings in the mid-'90s, UML

has grown into one of the central software technologies of this decade.

facilities to generate IDL interfaces for components. Similarly, the developers could automatically generate XML from the UML diagrams.

This approach, which uses high-level corporate models or frameworks to guide the actual development of specific applications, is termed Model or Achitecture Driven Development. It is an increasingly popular way of managing complex, distributed e-business systems, and government organizations have taken notice.

Last November, the U.S. Department of Defense Science Board released its "Report of the Defense Science Board Task Force on Defense Software."

This report described the department's current status of software development. It also made recommendations for reducing the cost and improving the quality of future efforts. One key recommendation was to adopt UML for describing application architectures. A second was to use Model Driven Development.

OMG's Architecture of Choice

To maintain the momentum behind UML and metamodeling interest, OMG recently voted to adopt and promote Model Driven Architecture. (See Fig. 2.) In effect, this architecture builds on OMG's decade of work on CORBA and middleware services and uses UML, MOF, and CWM to integrate CORBA with other middleware technologies and services. A company using Model Driven Architecture would begin by creating a description of an application using UML. Subsequently, using UML tools from leading vendors, the company could generate code for CORBA IDL, XMI for XML, or IDL links for Enterprise JavaBeans applications. Revisions would begin with UML models and then be automatically propagated to system runtime elements.

In a sense, Model Driven Architecture is a return to a more structured approach to large-scale application generation and integration. It will

Rösch Consulting: Tailor-Made Development

Simplicity, precision, component specification, and automation—these four tool-supported and integrated techniques from Rösch Consulting, Kaarst, Germany, bring the efficiency and precision of industrial production to the development of software systems and components, fitting any software development process.

Very Quick Modeling (VQM) is a tool-supported technique for structuring projects. VQM takes less than a day. It has been judged by customers as very helpful in structuring the knowledge needed for software systems and components. VQM gives project teams a head start on requirements analysis and ensures smooth development up to the verified UML model, where Software-Robots® take over to automatically create the rest of the software.

Objects 9000® is a tool-supported technique for specifying small software systems and software component interiors. Specifications can be so precise that software developers can verify their UML models. Verification is automated, takes a few seconds, and can be started at any time.

Components 9000® is a tool-supported technique for specifying the mutual dependencies of software components. The specifications can be so precise that software developers can automatically verify a set of components (including static interfaces and dynamic behavior). This allows teams to easily and safely integrate their work, resulting in a software system whose correctness can be measured objectively.

Software-Robots® automate software. Rösch Consulting builds Software-Robots to a company's specifications. The company also delivers software architectures for Internet applications. A Software-Robot automates all programming tasks needed to create ready-to-run software from Objects 9000 and Components 9000 results. Software-Robots generate code and also perform necessary design tasks before coding begins. They have been used for Java, C++, and COBOL, and can be used for all other languages. Developers need no longer worry about coding errors in the software development process, so they can guarantee that the final software is correct. And, using Software-Robots, developers can build and change software up to 10 times faster than before. Among the companies having achieved good results with the software are DePfa Group and Signal Iduna, Germany's third-largest insurance company.

For more information, visit **www.roesch.com.**

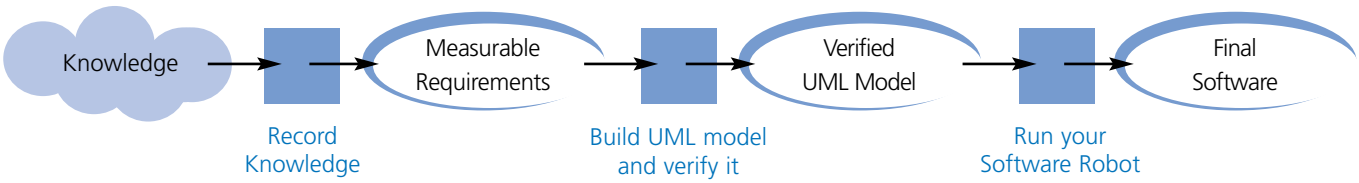
By Supplement Staff

The Future of Software

Error-free Software – Guaranteed

Must Software Really Have Errors?

Software used to be a nightmare. But now it's up to you to change the situation. Rösch Consulting supplies the tools and the knowledge. Ask us: **info@roesch.com**. We deliver.



Process Elements

The following process elements can be integrated into any software process: ISO 9000, CMM, Bootstrap and V-Model.

All software development begins with the knowledge the software should automate

❶ The first process element converts the knowledge into documented requirements. This step is manual and cannot be verified according to ISO 9000:2000, but it can be validated.

❷ The second process element creates UML models from the requirements and verifies these models. UML modeling is manual, but ISO 9000 verification is automatic.

❸ The third process element is 100% automated by Software Robots®. It usually creates more than 95% of the final software. The remaining (less than) 5% of the code have already been verified in step 2.

The final software is as precise as you wish. If your requirements are complete, the final software can even be guaranteed to be free of errors.

Advantages

- **The final software does exactly what it is supposed to do – and nothing else.**
- **The final software is as precise as you decide. You can even guarantee its correctness.**
- **The times of bug-ridden software are over. Demand the same 100% quality from your software suppliers.**
- **Requirements are traced automatically, down to the last element of the implementation – and vice versa.**
- **Object Traces and Interaction Diagrams are created automatically.**
- **No more maintenance problems – software can be re-verified at any time and within seconds.**
- **Proven technology: The Rösch Method® has been used in many projects, with up to 140 developers.**

How To Get There

Ask Rösch Consulting for

- **Tools (e.g. Software Robots®)**
- **Trainings (Zero-Defect UML, Components 9000®, Software Robots®, ISO 9000 for software)**
- **Coaching**
- **Turnkey pilot projects**

Rösch Consulting
Corporation for Innovative Software Development
P.O. Box 101440 • D-41546 Kaarst/Germany
info@roesch.com • www.roesch.com
Phone: +49 (2131) 986-300 • Fax: +49 (2131) 986-320

RÖSCH Consulting

remind some of Computer-Assisted Software Engineering (CASE). Unlike CASE, however, this current effort sponsored by OMG is much more flexible. In most cases, companies will not seek to generate complete applications from diagrams, but will use preexisting components and middleware to link them. The Model Driven Architecture approach is more focused on enterprise application integration than on code generation.

In the next year or two, as more companies rely on Model Driven Architecture to integrate their software applications and data resources, the rule of UML, MOF, and CWM will continue to move toward the center of enterprise application development efforts.

Wells Fargo's Business Object Services

The Business Object Services organization at San Francisco-based Wells Fargo Bank exemplifies UML's sophisticated approach. That approach was described at a recent UML conference by Tony Mallia, a senior architect with CIBER Inc., the Englewood, Colo.-based consultancy that helped Wells Fargo develop its Model Driven Development methodology.

Essentially, Wells Fargo's approach is to create a single business model that, in turn, drives the development of specific applications. The business model developed by the Business Object Services group describes key elements required by these applications. It includes business objects like "customer" and "account" and interfaces for every component that might need to be accessed across the network.

The bank's business processing environment is complex and multiplatform. (See Fig. 3.) BEA's ObjectBroker (CORBA) manages interactions between components and applications running on different platforms.

Wells Fargo's model is specified in UML and implemented in Rational Software's Rational Rose modeling tool. The model provides a detailed, logical specification of the major software components and how they interact, including messages passed between components. Wells Fargo can generate OMG IDL structure code for any message in the model.

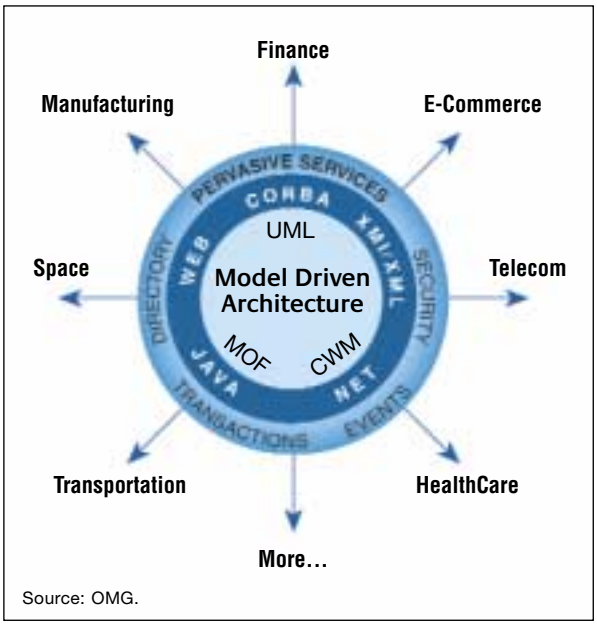


Fig. 2. OMG recently adopted Model Driven Architecture. A company using this could generate code for CORBA IDL, XMI for XML, or IDL links for EJB applications. Revisions would be automatically propagated to the system.

Most Wells Fargo applications are created by development teams that require the cooperative work of a client group (e.g., retail banking), Business Object Services, and the systems organization that maintains back-end systems and records. In effect, the Business Object Services group contributes its model, which defines the core business objects and the interfaces, and the middleware infrastructure t. The application is created by reusing objects defined in the Business Object Services infrastructure, and in some cases, extending those objects. Application developers create UML diagrams that describe the application. They then generate IDL message code for the CORBA stubs and repository entries that will be used in passing messages between application components.

Metamodels and Enterprise Integration

At the same time that OMG has systematically evolved UML, it has created complementary technologies that extend the power of UML beyond its developers' original goals. Metamodeling has made this possible.

Metadata refers to any data that describes other data. Thus, database record definitions are a kind of metadata. Similarly data-type definitions in conventional languages and class definitions are metadata in object-oriented systems. Data warehouse and enterprise information portals also depend on metadata in the form of tables, schemas, and dimensions. Middleware formats are also a kind of metadata. Increasingly, as companies try to integrate applications to build e-business systems or warehouse data, architects focus on metadata and the models that define its various permutations . Ultimately, system integration is the ability to translate from one metadata format to another. Compatible metamodels allow tool vendors to automate transformations between islands of data. At the same time, a metamodel allows companies to catalog, index, and search enterprise data and metadata assets in heterogeneous systems in ways never possible before now.

A metamodel is a model of a set of

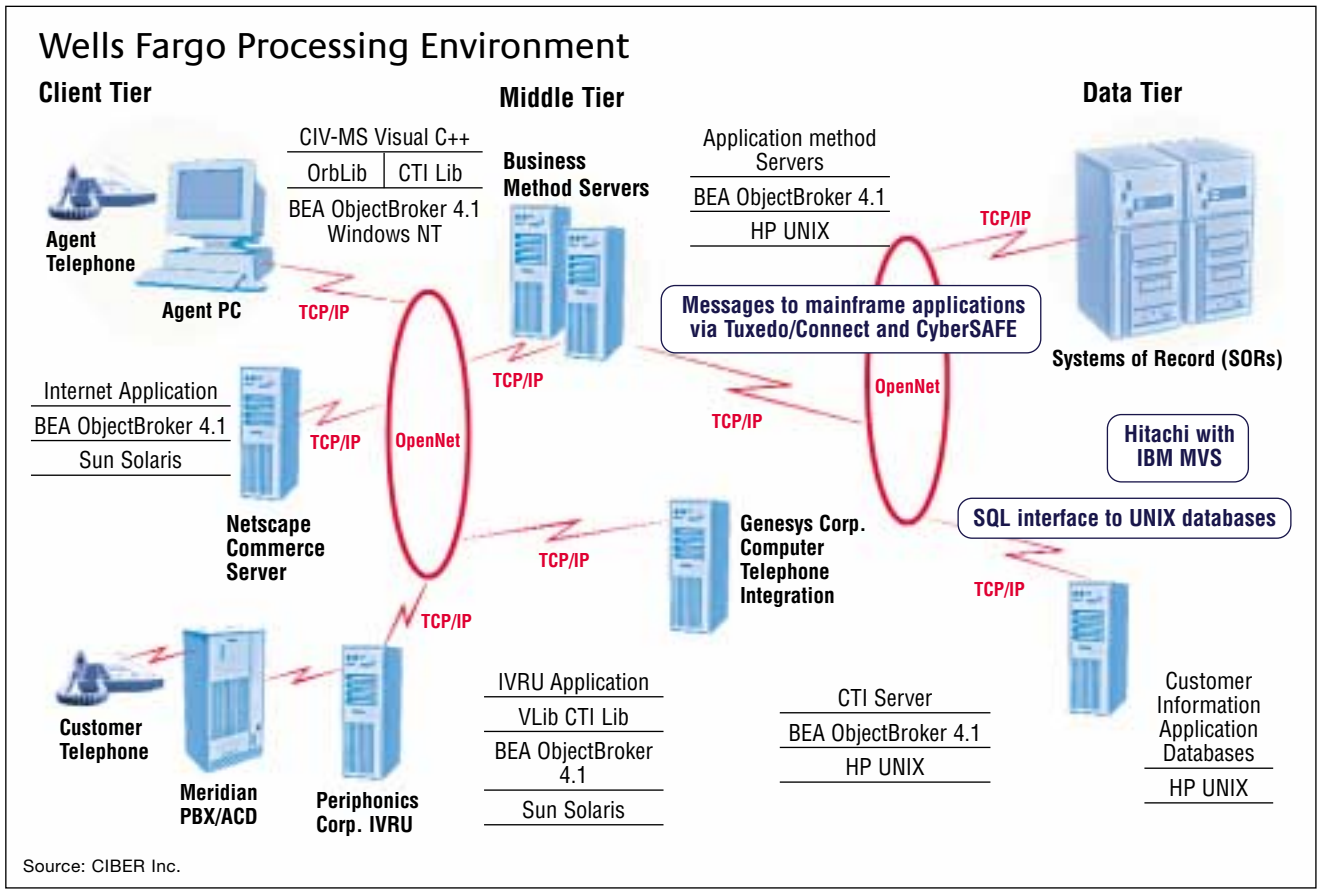


Fig. 3. Shown here is Wells Fargo's complex, multiplatform business processing environment. BEA's ObjectBroker (CORBA) manages interactions between components and applications.

models. UML is itself a metamodel that describes all specific UML models or diagrams. Developers use UML to create models of software applications. Then, based on those application models, developers generate software to implement the application.

Similarly, using the relational database model, developers generate models of specific database applications and then populate those models with actual data.

The OMG has formalized a vocabulary for metamodeling. It was found that four hierarchical layers were sufficient to represent models. (See Fig. 3.)

The OMG's meta-metamodel is called the Metamodel Object Facility (MOF). MOF is a subset of UML that can provide formal descriptions of metamodels, including both UML and MOF itself.

Once the OMG task force realized the power of using MOF to standardize metamodel descriptions,, it decided it needed a standard way to pass information between metamodels and, by extension, between all models.

The OMG Task Force selected XML as a way of packaging MOF and UML information. OMG created an XML DTD language to pass metamodel information. The DTD language is called XMI, the Metadata Interface (XMI).

Using XMI, developers can specify Document Type Definition (DTD) tags that become, in effect, words in a specialized language. XMI can be transported over the Internet on HTTP, IIOP, or other wire protocols. UML visual modeling tools are currently adding XMI capabilities so that they can pass UML models from one tool to another. In a similar way, developers can move information about UML models from tools to databases that support MOF.

Last year, OMG completed the Common Warehouse Metamodel (CWM) specification. CWM defines how specific data warehouse models relate to each other. CWM models are described via UML, and the CWM metamodel is, in turn, defined in terms of MOF. Oracle; IBM; Unisys Corp., Blue Bell, Pa.; Hyperion Solutions, Sunnyvale, Calif.; and Dimension EDI, Surrey, U.K., have already released CWM-compliant versions of their

warehouses. Because CWM is MOF compliant, CWM database users can pass information about their models to MOF-compliant databases, via XML. In effect, any CWM database can automatically reformat information, via XMI, into an XML document that can then be sent over the Internet to other CWM databases or data warehouses.

The Government Computer-Based Patient Record Project

Another example of UML use is the Government Computer-based Patient Record project (GCPR), developed with CIBER's assistance. This system was designed to enable a variety of medical systems, from Veterans Affairs and Defense Department treatment facilities to civilian health services here and abroad, to exchange information.

The GCPR project was a joint effort that included the Department of Defense, Veterans Affairs, the U.S. Indian Health Service and the Louisiana State University Medical Center. These organizations, along with the others that will eventually use the system, maintain medical records on patient care, including illnesses, injuries, and inoculations. Unfortunately, they maintain these records on different computer platforms and have diverse ways of organizing, naming, and coding the information.

The key to gathering all the information on a single patient is to have a standard format into which information can be organized. Doing that involves transforming data from various formats into a common one. The infrastructure that does this is called the GCPR Framework.

The prime contractor for the GCPR Framework effort was Litton/PRC, McLean, Va. Litton/PRC subcontracted CIBER to develop the model. Two models were created. The government created the Reference Information Model to describe information in patient healthcare records. Litton/PRC developed the Framework Computation Model with CIBER's help. In effect, Litton/PRC and CIBER collaborated with the government to ensure that the UML reference model that described patient information could be incorporated and processed by the framework.

Litton/PRC and CIBER also created

OMG Establishes Model Driven Architecture

Founded in 1989 by 11 companies, the Object Management Group™ (OMG™) began independent operations as a not-for-profit corporation. Through OMG's commitment to developing technically excellent, commercially viable, and vendor-independent specifications for the software industry, the consortium now includes approximately 800 members. OMG is moving forward in establishing the Model Driven Architecture™ (MDA™) through its worldwide standard specifications including CORBA, CORBA/IIOP, UML, XMI, MOF, Object Services, Internet Facilities, and Domain Interface specifications.

OMG MDA™ strives to ensure that mission-critical information systems are rooted in standards that will adapt to new hardware capabilities and software platforms. This initiative addresses the complete life cycle of designing, deploying, integrating, and managing applications as well as data using open standards. MDA-based standards enable organizations to integrate whatever they have in place with whatever they build.

OMG is headquartered in Needham, Mass., with a subsidiary in Japan. The organization has marketing offices in Bahrain, Brazil, Germany, India, and the U.K., along with a government representative in Washington, D.C.

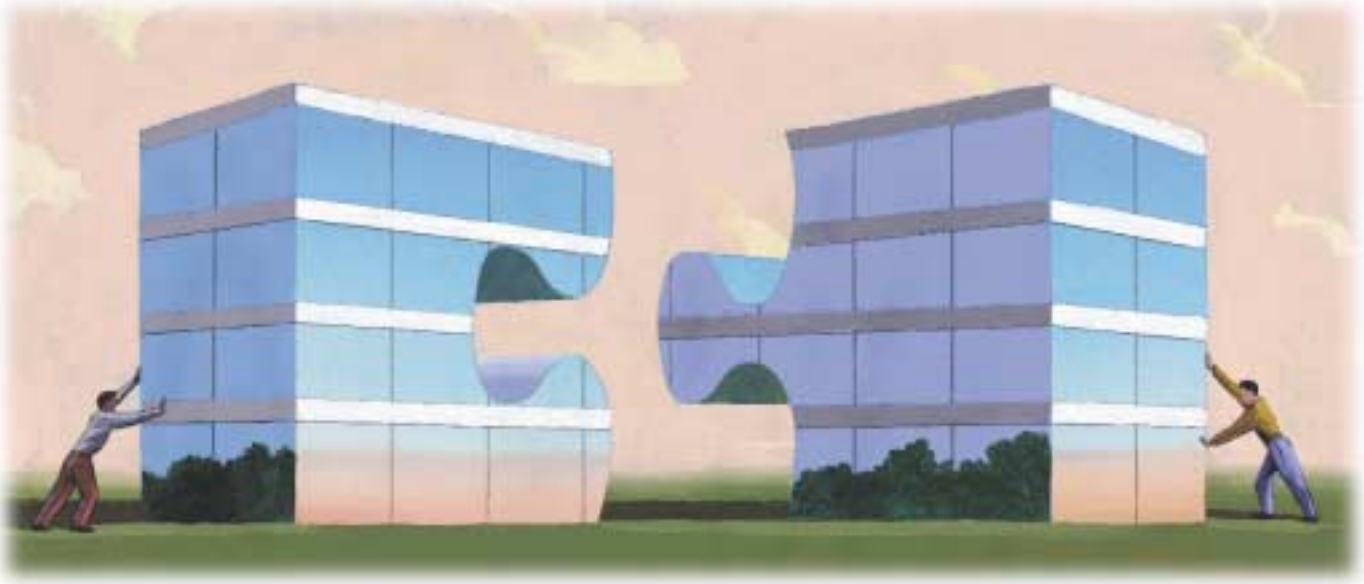
OMG was formed to create a component-based software marketplace by accelerating the introduction of standardized object software. The organization's charter includes the establishment of industry guidelines and detailed object management specifications to provide a common framework for application development.

Hundreds of member companies have chosen OMG's Internet protocol IIOP (Internet Inter-ORB Protocol). Vertical markets worldwide use OMG specifications to develop and deploy distributed applications. These markets include manufacturing, finance, telecom, electronic commerce, real-time systems, and healthcare.

OMG is structured into three major bodies: the Platform Technology Committee (PTC), the Domain Technology Committee (DTC), and the Architectural Board. Within the technology committees and Architectural Board rest all the task forces, special-interest groups, and working groups that drive the OMG's technology adoption process.

For more information, visit **www.omg.org.**

By Supplement Staff



Keeping up with the constantly changing landscape of high technology is often like fitting together pieces of a puzzle over and over again. How do you manage to fit it all together to make sure your strategic business objectives are being met?

We're Known By The Companies We Connect.™

The Object Management Group™ (OMG™), founded in 1989, connects hundreds of talented, driven professionals whose IT experience ranges from systems infrastructure and security to e-commerce, finance and healthcare. The best and the brightest work through OMG's proven, open consensus-making process to create software integration standards that enhance our members' competitive positions in the marketplace. OMG's early success with CORBA® expanded in the late '90s to include the UML™ "software blueprint" language, and now includes solutions for data warehousing and repository integration. Over the last four years, OMG members have defined, adopted and published standard solutions for air traffic control, manufacturing plant integration, financial ledger systems, clinical decision-making and many other areas of information technology.

OMG membership allows you to help shape the direction of the software industry, to ensure that you will be able to integrate the technology you already have with the "next big thing" that comes along in the future. We invite you to learn more about the Object Management Group and see for yourself how connected we are.

For more information, contact the Object Management Group at +1-781-444 0404 or visit our Web site, www.omg.org/connect.

OMG®

OBJECT MANAGEMENT GROUP

CORBA®, and the Object Management Group logo® are registered trademarks of the Object Management Group. OMG™, Object Management Group™, UML™, and "We're Known By The Companies We Connect". are trademarks of the Object Management Group.

the UML-based computation model that described how any record could be reformatted to be compatible with the GCPR reference model and released based on security and privacy policies. The records involved can vary greatly. Some, for example, might describe patient laboratory results, while others might describe prescriptions. The computation model was created using a metamodel that describes, in a generic way, how any system might transfer medical records. At runtime, the metamodel is "downshifted" to a model and drives Java code development. Without this approach, the framework would have to be regenerated each time a record's format was extended or changed. By using metamodeling techniques, CIBER was able to help Litton/PRC develop a powerful, generic solution to the transformation problem. This is important because GCPR is expected to grow to incorporate more medical information.

The UML models were developed with Rational Software's Rose tool. Unlike the Wells Fargo UML models, which generate OMG IDL code for use with that company's CORBA middleware, the GCPR Framework Computational Model is used for code design. The Reference Information Model is currently transferred to the Litton/PRC Framework development team using Rational Rose's proprietary format, which will eventually be replaced with XMI.

Experiments are under way to generate XML DTDs directly from the model to define XML exchange, but this will likely be overtaken by Healthcare HL7, Version 3 standard XML messages, which are developed from a Healthcare model developed in UML. Investigation is also under way to merge government and HL7 reference models.

The GCPR system is in the pilot-testing phase. It will be ready for general use after security and privacy designs are implemented.

What's Next

In the early '90s, OMG set the pace in advanced software development by pioneering technologies that facilitated

MSTI's Message-Passing Middleware Complies with Standards

MPI Software Technology Inc. (MSTI) designs middleware for companies that want to control their cost of ownership and maximize the return on their investment by solving problems important to their mission, rather than tangling with free but costly-to-own software that has no guaranteed support path, performance, or compliance.

MSTI's Message Passing Interface (MPI) middleware defines how data is transferred among multiple processors working together on related programs. The company's message-passing products are based on de facto standards developed and accepted by the organizations they serve. MSTI, Starkville, Miss., recently released Version 1.6.3 of MPI/Pro for NT/2000 and Linux, which is fully compliant with the MPI Forum's 1.2 standard.

The company also produces Cluster Controller, a job scheduler and resource management tool for NT/2000. MSTI'S most recent addition to its product line, VSI/Pro, is a math library compliant with the emerging VSIPL (vector, signal, and image processing library) standard. VSI/Pro is available for LynxOS for G3 and Altivec/G4 and VxWorks for G3. Altivec/G4 support will soon follow for the VxWorks operating system.

MSTI sells its products worldwide. Users include the U.S. Department of Energy labs (Sandia, Argone), Compaq Computer, Dell Computer, General Electric, Schlumberger, Northrup-Grumman, Cornell University, and Mississippi State University. MSTI's OEM customers include Paralogic Software, JPA Electronic Supply, Ansys, Radstone, and Thales (formerly CETIA, a subsidiary of Thomson Marconi). The company's networking partners are Giganet (recently acquired by Emulex), Compaq, and Myricom. MSTI began in August 1996 as a spin-off of research done at Mississippi State by Dr. Anthony Skjellum.

MSTI has 21 full-time employees. Most of them work on high-performance implementations of MPI middleware, as well as on the new MPI/RT.

For more information, visit **www.mpi-softtech.com.**

By Supplement Staff

The MPI Software Technology, Inc. Family of Products

Teragrade™ middleware for real-world parallel computing.

CORBA™ is called "the middleware that's everywhere™." MPI Software Technology, Inc. provides MPI and MPI/RT middleware for scalable, parallel computing with scheduling support that's everywhere you want scale.

MPI Software Technology, Inc has undertaken strategic efforts to exploit CORBA for scalable, cluster scheduling in ClusterController®, and to support emerging performance, scaling extensions of CORBA including Data Parallel CORBA. Also, MPI Software Technology will be supporting parallel processing with VIPL and Infiniband™ compatible ORBs as soon as they become available.

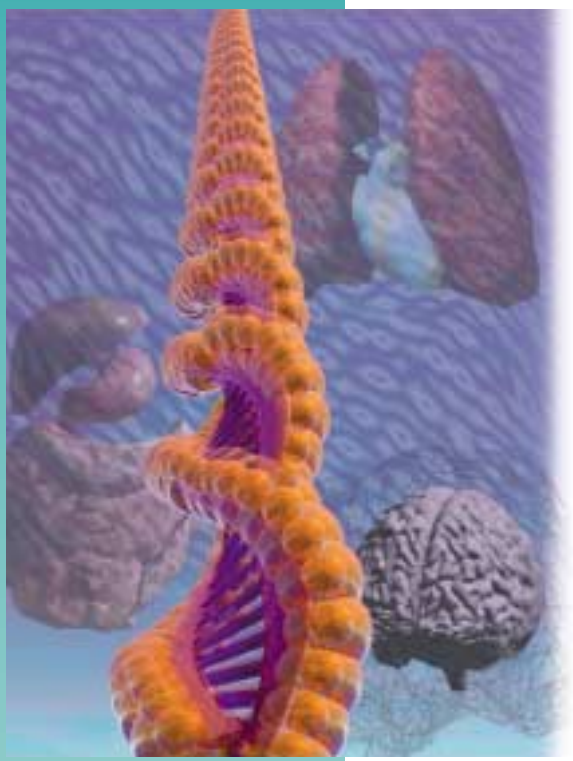
Of course, MPI Software Technology, Inc. - like CORBA itself - works across platforms and operating systems, from embedded systems with VxWorks to Scalable Clusters of Linux, Windows, and MacOS.

ClusterController® — Cluster Scheduling

MPI/Pro® For Linux — Message Passing

VSI/Pro® — Vector Signal Image Processing

Champion/RT™ — Real-time Message Passing



MPI Software Technology™

http://www.mpi-softtech.com

MPI Software Technology, Inc., Technology Square Bldg., Suite 33, 101 South Lafayette St., Starkville, MS 39759
The Source for MPI support, MPI-2, Embedded & High Performance Middleware™
For more information: sales@mpi-softtech.com

VSI/Pro®, MPI/Pro® and ClusterController® are registered trademarks, and Champion/RT™ is a trademark of MPI Software Technology, Inc. Other names are trademarks or registered trademarks of their respective owners. Artwork copyright © 2001 by MPI Software Technology, Inc. This information is subject to change without notice or obligation.

Secant: From Vision Model to Reality

Secant advances the functionality of model-driven development to enable organizations to rapidly build, power, and evolve high-performance applications.

Cleveland-based Secant Technologies provides model-driven application generation and integration software. Secant's products also ensure scalability to exponential levels of complexity and demand.

According to one Secant customer, Mark McCulloch, Dallas-based XTRA On-Line vice president, "[XTRA On-Line] saved six months of development time and $400,000 using Secant's products. We expect this ROI could be as high as 20 person-months during the maintenance and evolution of the PowerTrip system."

Secant provides a model-driven application server platform (Secant Extreme Enterprise Server™) and a model-driven object-to-RDBMS integration technology (Secant Extreme Persistent Object Service™). These products integrate with development tools to support model-driven development, which allows developers to specify enterprise features using industry-standard UML models. Secant's code generator takes these models and automatically creates the glue-code needed to connect the application to the fully integrated, standards-based server.

Secant's time-saving code generation technology combined with support for distributed transactions, high-availability clustering, multi-level in-memory caching, and object-to-relational integration technology provide a unique platform solution. As a result, Secant customers frequently report as much as 80% reduction in development time and an increase of 10 times or more in runtime performance. In addition, because developers don't have to write middleware code, companies can maximize developer involvement in demanding enterprise applications.

Secant provides technical support and a full line of professional services that include a comprehensive product-based educational curriculum, consulting, and mentoring in enterprise system architecture, application design, and systems implementation.

Secant and its products have received prestigious awards, including "Model-Driven Infrastructure for Custom E-Business Applications," Crossroads A-List Award, January 2001; "Best Java Application Server," *Network Computing* Real-World Labs, June 2000; "Best Database Product," *Java Developer's Journal,* Dec. 1999.

Large organizations such as Detroit Edison Co., i2 Technologies Inc., Incyte Genomics Inc., Open Telecommunications, and Susquehanna Partners, GP have benefited from Secant's open, standards-based solutions.

For more information, visit **www.secant.com.**

By Supplement Staff

distributed computing, leveraging object-oriented techniques. Specifically, OMG created the CORBA middleware architecture and its associated facilities to allow companies to link object-oriented languages, components, and databases to build large enterprise systems.

Today, developers have accepted objects and component languages but are now focused on how best to architect Internet-based applications, and integrate all their shiny, new legacy systems. And they now face complex distribution and integration issues. Once again, OMG offers an open, comprehensive solution. OMG's approach is based on the UML, MOF, XML, and associated Internet standards. Using UML and MOF, companies can describe any application they want to build and any tools or facilities they want to use with that application, including databases, warehouses, or middleware systems. By specifying applications in UML, developers gain access, via XMI, to models or languages also specified in MOF or UML.

Put another way, UML and its associated technologies can help change the way developers think about modeling and integration. UML, coupled with MOF and XML, is about to become the heart of a metamodeling architecture that will make it possible for developers to transparently link almost any software application with any other.

Metamodeling with UML is set to serve the next decade of developers in the way that Object Request Brokers served developers in the '90s.

As OMG begins its planned upgrade of UML to Revision 2.0, 37 companies have signed on to define the new vision, the most ever to submit to any OMG technology adoption process. It's going to be interesting to watch OMG technology continue to evolve.

UML really has no competitors

today. There are some real-time methodologies like Shlaer-Mellor and component methodologies like D'Souza and Wills Catalysis, but these have rapidly redescribed their notations in terms of UML. It is the great strength of UML's stereotype system that it can be extended to incorporate more specific notational systems. Thus, although some developers may use earlier methodologies, when they update to the latest version, they will probably find that the UML stereotypes describe the notation they were using before.

Some developers and organizations have complained that UML is too large and complex. In fact, when you consider all the diagrams and the stereotypes UML includes, it can be overwhelming. In reality, however, developers use only the notational elements and diagrams they need for a specific project. Thus, most organizations use UML without being aware of most of its more esoteric notations.

Microsoft developers provide a good example of this. Microsoft includes Visual Modeler, the company's graphical tool, with its Visual Studio package. Originally, Visual Modeler was developed in conjunction with Rational and in fact supports only a few UML diagrams. Most Visual Modeler users neither know nor care about complex UML diagrams. At the same time, however, because they use UML, if they run into an application that gets more complex than Visual Modeler is designed to handle, they can readily switch their work to a more comprehensive UML tool like Rational's Rose or Islandia, N.Y.-based Computer Associates' Paradigm Plus. UML has certainly become the structured methodology of this era. However, today's visual modeling tools aren't designed to generate complete applications from diagrams the way the best of the CASE tools were in days past. The problems involved in creating such comprehensive tools has been avoided in favor of tools that help developers conceptualize their applications but require that developers generate significant portions of the application by other means. In part vendors supported this approach to avoid complications encountered when trying to generate a complete application. In part this approach recognizes the fact that modern developers are much more likely to reuse components, and that today's modeling tools are used more for design than for code generation.

UML turns out to be a key ingredient

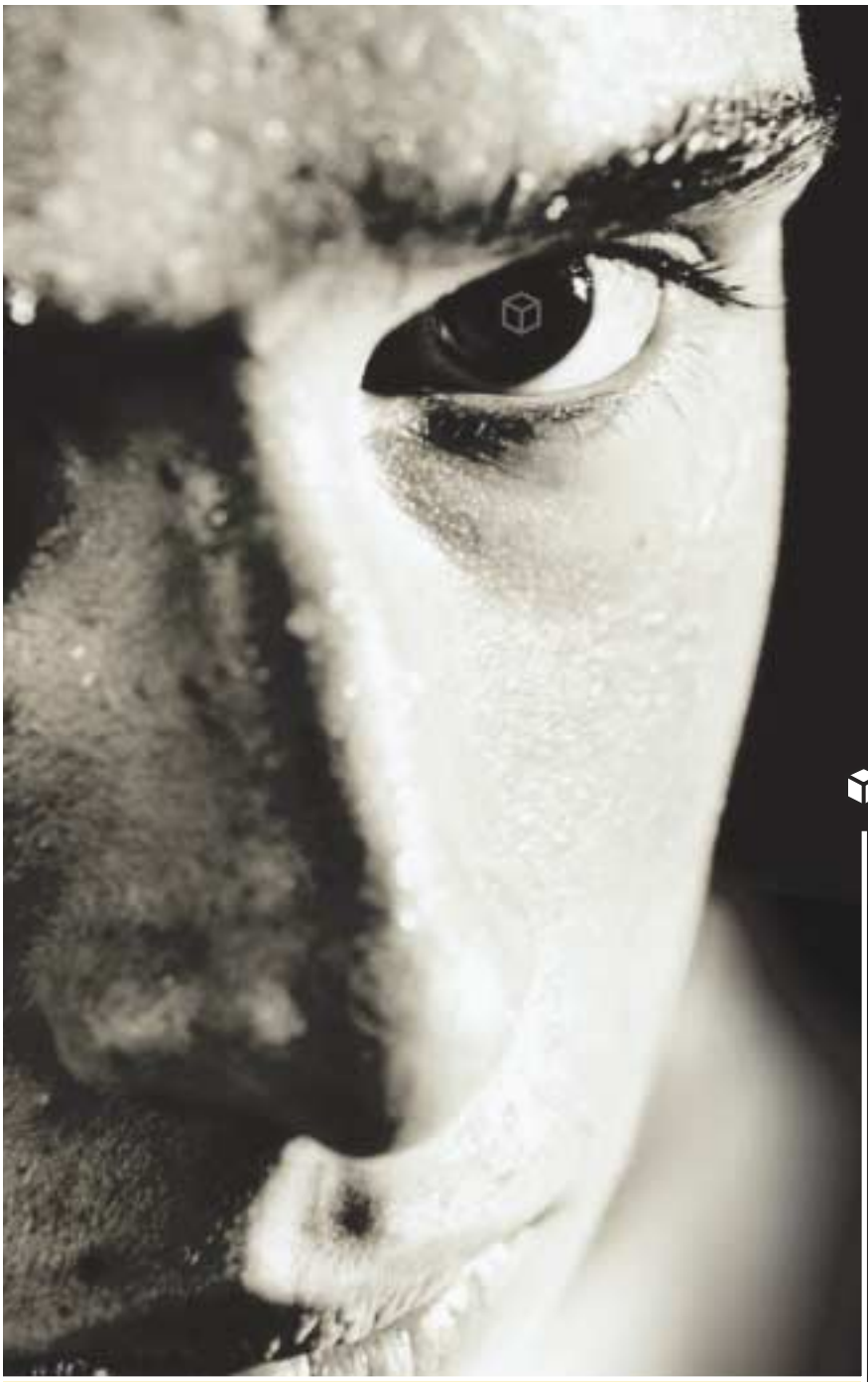
in the new Model Driven Development approach to software development proving so popular with e-business developers.

Marketing Challenges

Probably the biggest marketing challenges facing analysts and vendors who promote UML is the desire of programmers to work informally and companies to develop applications quickly and cheaply. This problem will continue to haunt the software development industry as it struggles to shift from a craft to an engineering discipline that can reliably produce effective products. It's no secret that even today, few large software projects are delivered on time or within budget. Indeed, many large software projects are never delivered or fail when they are. Most studies of this problem return to the same solution that has been suggested for the last two decades: better analysis of the initial problem and a more systematic development methodology. Both of these depend on companies adopting and enforcing methods like UML. Organizations of any size can take advantage of UML. Small organizations can use tools like Microsoft's Visual Modeler simply to examine the components they wire together to create interfaces. More sophisticated organizations can use a wider variety of diagrams and techniques to help design and deliver more complex applications. The use of metamodeling and XMI to transmit UML between tools and databases and to integrate diverse applications, however, is just being explored and developed. Today, these techniques are mostly confined to large companies with sophisticated groups of systems developers. It will be several years before these newer techniques become embedded into tools that smaller companies can easily use.

Today's UML users range from Microsoft developers, who use Visual Modeler to diagram departmental COM-based applications, to the Department of Defense, which uses more sophisticated tools to design huge applications it takes years and hundred of developers to create. By being more flexible and more modest, UML has appealed to a much wider range of developers than CASE ever did. •

Paul Harmon is a senior consultant with Cutter Consortium's Distributed Architecture and E-Business service in San Francisco. He recently co-authored Developing E-Business Systems *and* Architectures: A Manager's Guide *with Michael Rosen and Michael Guttman (both of IONA).*



When winning is your *only* option.

When only one in four development projects are successful, how can you beat the heat and make sure your new application is on time, on budget, and on target?

The answer is **ModelMethods**™ software and services from Secant.

ModelMethods bridges the chasm between your ideal business model and the software applications and infrastructure that run your business.

From a model of your business, **ModelMethods** automatically generates most of your application and provides the award-winning **Model-Driven Infrastructure**™ to ensure the successful performance and scalability of even the most complex and demanding applications.

When you have to win, **ModelMethods** is your only option.

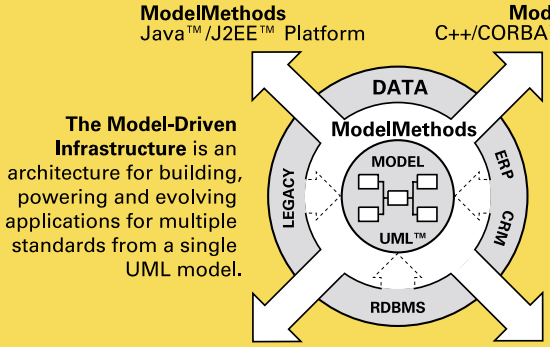
Put yourself in the winner's circle now and visit **www.modelmethods.com** or call 1-888-473-2268.

ModelMethods™
by Secant

From Model to Reality™

ModelMethods™
by Secant

The Award-Winning
Model-Driven Infrastructure™
that's here today.



The Model-Driven Infrastructure is an architecture for building, powering and evolving applications for multiple standards from a single UML model.

CROSS ROADS

A-LIST AWARDS 2001

Copyright ©2001, Secant Technologies, Inc. All rights reserved. Secant Technologies, Inc., ModelMethods, the ModelMethods logo, Model-Driven Infrastructure and From Model to Reality are trademarks of Secant Technologies, Inc. All other trademarks are the trademarks or registered trademarks of their respective rights holders.