April 2002

OMG News

THE ARCHITECTURE FOR A CONNECTED WORLD

IN THIS ISSUE:

SUCCESS STORIES

FEATURES & COLUMNS:

SUCCESS STORY: UML

Smiths Aerospace

Tools Used:

ARTiSAN Real-time Studio™, Unified Modeling Language™ (UML™)

Description:

Smiths Aerospace was formed from the merger between Smiths Industries Aerospace and TI Group's Dowty. The new company with combined sales in excess of £1.1 billion is a first-tier supplier for Boeing and Airbus and is a major supplier for retrofit and upgrade of commercial and defense aircraft. Replacement of outdated avionics can effectively prolong the life of an aircraft, potentially doubling its length of service. Through continuous research and development investment, the Smiths Aerospace manufacturing facility at Bishops Cleeve, near Cheltenham in the UK, incorporates the latest technologies into leading-edge avionic display products, mission management, and power-switching systems.

In the late 1990s, the Smiths software development teams at Cheltenham produced a standard approach to the analysis and design of software. Structured analysis techniques were used to provide a functional breakdown of the requirements placed on the software. However, in place of the structured design techniques traditionally used, the company had adopted an Object Oriented (OO) design approach. This technique was seen to offer significant advantages over structured, top down

continued on page 6

SUCCESS STORY: UML

THALES

Tools Used:

I-Logix' Rhapsody®, Unified Modeling Language™ (UML™)

Description:

I-Logix' Rhapsody UML-based application development platform combined with services and support, aid THALES in unifying software and systems development across all 12 of its business units.

Problem:

When companies become large, the lack of information sharing, data, technologies and best practices can cause severe problems. Even companies with only a few dozen employees will experience cases where engineers tackle common tasks in different ways, use different tools and who have their own spin on what it is they think they are making. In such cases, jobs are often done twice, sometimes not at all – and the lack of standardization, in terms of tools and methodologies, within the company leads to inefficiencies across the board.

Now, consider how it must be for large corporations – organizations employing thousands of people, many of whom are not even based in the company's country of origin. Most of these organizations have teams of engineers, multiple projects, and systems comprising electronic, software,

continued on page 4

Lockheed Martin

Tools Used:

Kennedy Carter's iCCG, UML™, iUML™, Model Driven Architecture™ (MDA™)

Description:

Lockheed Martin Aeronautics at Fort Worth, Texas has used the OMG's Model Driven Architecture to develop the F-16 Modular Mission Computer (MMC) Application Software. Their goal was to achieve cross-platform compatibility and increased productivity and quality, all in the context of the demanding environment of avionics software development.

Problem:

The F-16 MMC team originally used traditional CASE tools with an OO modeling notation to specify the software before manually coding in Ada. When they migrated their development to Kennedy Carter's iUML tool, they gained the ability to use a UML action language which made their UML models executable. They could then test their UML models to verify their intended behavior before hand-coding the implementation.

More recently they have used Kennedy Carter's iCCG product to specify, in eXecutable UML, an Ada code generator which can automatically generate 100% of the Ada implementation.

By this means they guarantee that their UML models are entirely platform independent and portable across any future platform.

In Lockheed Martin's case, what constitutes the platform?

The software execution platform effectively raises the abstraction level of the hardware to provide a platform on which code generated from eXecutable UML models can run directly. Lockheed Martin's goal of complete cross-platform compatibility implies a very strong form of platform independence whereby the UML models which specify the Application software behavior can be ported without change, even if the Application Software Interface changes.

Solution:

The use of MDA allows the mission software functionality to be formalized as eXecutable UML models (xUML); such models are Platform Independent Models (PIM) in MDA. Platform independence is essential if the goal of decoupling the models from any changes to the Software Execution Platform is to be achieved. We use the term xMDA to mean an MDA approach augmented by the use of xUML. xUML models are expressed using a UML action language based on the newly adopted precise action semantics for the UML (see http://www.omg.org/technology/documents/modeling_spec_catalog.htm#Action_Semantics).

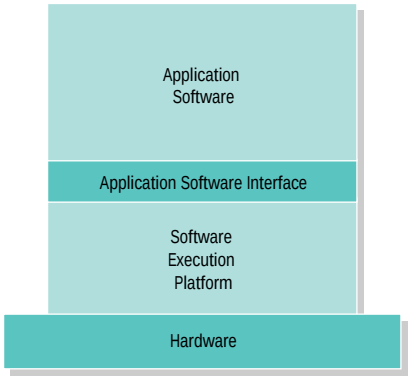

FIGURE 1: Mission Software Main Architectural Elements

Figure 1 shows the basis for the F-16 Mission Software architecture; it depicts the main architectural elements:

- Software that is unique to the application(s) for which the embedded computer exists, it represents some 80-90% of the total software (in terms of long-term development cost);
- Application Software Interface, the boundary between the Application Software and the Software Execution Platform. This provides the methods by which the Application Software can make requests and use the services of the Software Execution Platform and the Software Execution Platform can provide its services to the Application Software;
- Software Execution Platform, low-level software, the purpose of which is to allow the Application Software to run on the hardware; The software execution platform incorporates device drivers, the built-in test and the RTOS;
- Hardware, the embedded system hardware for the F-16 Mission Management system.

Executable models support the MDA approach in two main ways. First, they allow early testing using simulation and debugging tools. Secondly, since they are a full and formal specification of the system behavior, they allow generation of the target code. Defining a mapping from the rigorously defined PIM (expressed in xUML) to the implementation, is what is at the heart of xMDA. Since xUML models are executable and rigorous they act as much more than a simple visual agenda for the software developers, they actually embody all the business logic required to execute and verify the system.

Definition of the Platform Independent Model (PIM) to Platform Specific Model (PSM) Mapping:

The basis of defining a mapping that allows full and automatic translation of the PIM models, expressed in xUML into the implementation is to build an xUML model of the translation system itself and execute it! The elegant conceptual coherence of employing xUML both in the business-modeling realm and in building the translator means that a minimum set of new skills has to be learned.

Figure 3 shows the basis for defining the mapping from PIMs (expressed in xUML) to their platform specific implementation. "Ordinary" analyst models (level 1) are used to populate the metamodel of xUML (level 2), which has all the necessary processing defined (remember it is an executable model in its own right) to populate a model of the implementation (level 3). This final model, which again is expressed in xUML, is executed to produce the implementation. The analyst models (level 1) are augmented with tags that act like "compiler directives" to the translation system and allow such facts as limited instance populations to be exploited in order to produce efficient target code.

Tool Support:

There are two main areas where tool support is vital in order to reap the full benefits of the MDA process. First, iUML is used to provide the modeling and

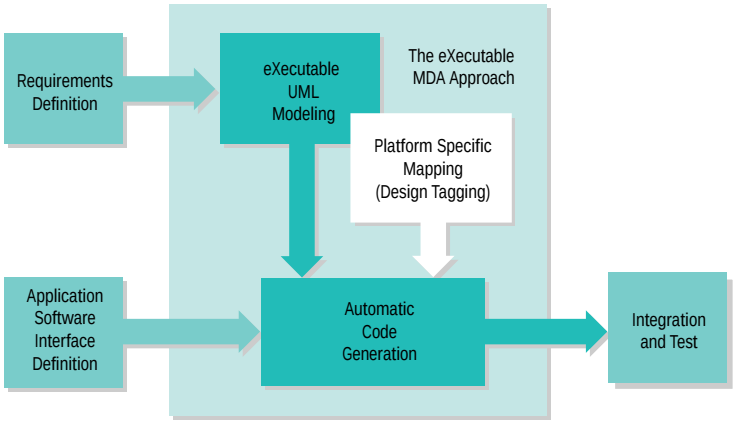

FIGURE 2:
Lockheed Martin's Executable MDA (xMDA) Process

simulation environment for the xUML models. This tool provides dedicated intelligent support for xUML and allows models to be simulated on host and "debugged" visually at the UML level of abstraction. The second part of the toolchain is the translation environment (level 2 in Figure 3 above) where application models are extracted from the iUML database and used to populate the translation engine. The translation engine is a specialization of the intelligent Configurable Code Generator (iCCG). iCCG allows developers to capture their mapping rules as xUML models and so produce any target implementation of which they can conceive. The specification of a mapping from PIM to PSM in eXecutable UML is itself highly reusable, allowing any set of application models to be generated onto the target.

Further details of these and other products that support xMDA may be obtained at www.kc.com.

Benefits:

The use of MDA with executable UML (xMDA) has provided many benefits to the F-16 project:

- The application models are expressed in a completely platform independent way and so can be reused across multiple hardware and software platforms;
- UML modelers are isolated from the software and hardware details and so can concentrate on a thorough exploration of the problem space;
- The hardware and software platforms can be upgraded without impacting the application models;
- Models can be tested at the earliest opportunity by executing them in the iUML Simulation environment;
- Rework is reduced with validated models;
- The mapping from PIM to PSM is specified in xUML with iCCG and is highly reusable;
- Code generation eliminates manual coding and eliminates the defects traditionally introduced in the coding phase;
- The xUML models are the primary source. Code is not maintained.

Taken altogether these MDA benefits have reduced application development time by 20% on the F-16 MMC program in addition to helping them achieve complete cross-platform compatibility. ■

Acknowledgments

This article was based upon a presentation prepared by:

Lauren E. Clark
Chief Engineer
F-16 Modular Mission Computer Software

Terry Ruthruff
Staff Specialist
Software Engineering Core

Bary D. Hogan
Methodology Lead
F-16 Modular Mission Computer Software

All of the Lockheed Martin Aeronautics Company

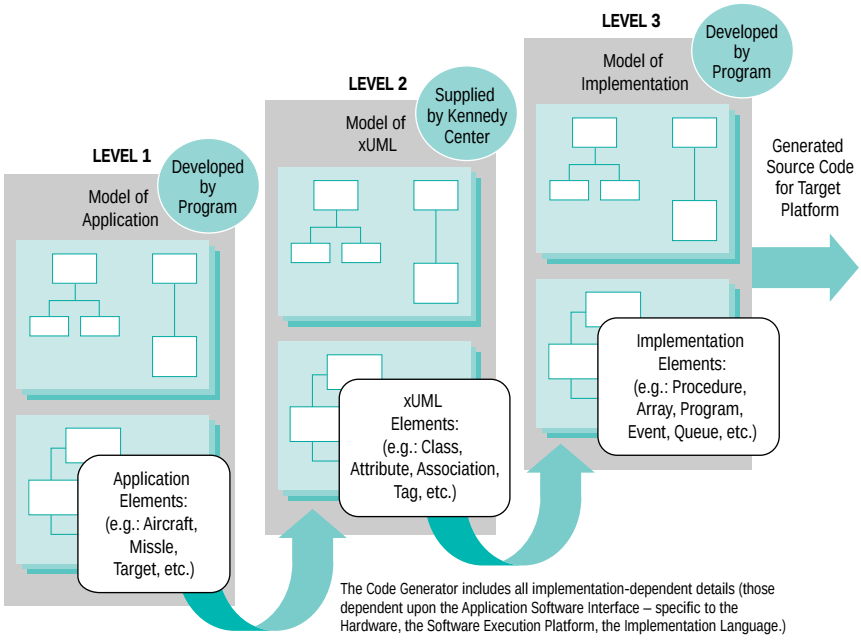

FIGURE 3:
Mapping PIM to PSM

THALES

continued from page 1

mechanical elements – coming together from different corners of the world. It is extremely difficult to ensure that the systems that are developed actually meet the customers' requirements.

It is all too easy to see how you could soon have a situation where whole teams are working in different ways – unless you standardize on tools and methodologies across the entire development organization: which is exactly what defense company THALES (formerly Thompson-CSF) is in the process of doing under the guidance of its THALES/TTM (THALES Tools and Methodologies) business unit. THALES is at present, pooling its technical and industrial disciplines, procedures and practices.

Solution:

THALES is divided into 12 business groups, each devoted to specific areas of business such as: Avionics, Simulation, Air Traffic Management, and Airborne Systems. THALES/TTM, operating as a corporate technical business unit over all 12 business groups, is charged with identifying and selecting "technological building blocks" in order to shorten project lead times,

It is all too easy to see how you could soon have a situation where whole teams are working in different ways – unless you standardize on tools and methodologies across the entire development organization.

reduce risks, and guarantee technology adaptation fit for specific and demanding roles (for example, defense and aerospace).

Michel Galinier, Software and Systems Manager at THALES/TTM in Orsay, France, comments: "TTM focuses on three main activities. One, adopting methods, processes, and development tools for our technical and industrial disciplines. Two, the selection of technologies, for example the assessment of COTS and architecture "building blocks." And three, information sharing – for example knowledge management."

Galinier goes on to say within these distinct activities, common disciplines are addressed. "For example, all three encompass program management, systems engineering, software, hardware, mechanical, logistics, purchasing, technical documentation, and web technologies."

Key to TTM's strategy is THALES standardizing on CAD tools for electronics, software, mechanical and systems development – and integrating all within a project management strategy that will unify the whole of THALES.

Work on establishing common engineering and management technologies in THALES began in the 1990s – tightening (standardizing) practices for hardware and software development. Galinier recalls: "Three years ago, to manage the object technology and risk/opportunity in software we decided to set up a common approach with the selection of Jacobson for Object Oriented Design and the Unified Modeling Language (UML) as a notation."

OMG's UML, the most common modeling language in use for software development, is frequently used in firmware and real-time embedded systems development. In many organizations, object-oriented techniques and the UML are being used by systems engineers to specify the architectural structure of the overall system.

TTM reviewed a number of UML supportive products including: I-Logix' Rhapsody®, Rational's ROSE, Object-Team's Objecteering and Artisan's Real Time Studio. Galinier comments: "We set up a benchmark for tools supporting UML and this benchmark focused on two sets of users; software engineers and systems engineers. For software engineering, the requirements included Jacobson methodology support and automatic code generation. For systems engineering, the requirements included being able to describe system architecture with UML, the need for a good user interface, documentation generation and simulation/animation."

Benefits:

The benefits of using UML for system specification are two-fold. First, for systems that are software centric, the use of UML allows easy transition from the systems engineers to the software developers. And secondly, the behavioral model of UML is based on Harel Statecharts – a well defined executable language, which means that a specification of a system in UML can be executed and evaluated for consistency, accuracy and fidelity before it is partitioned into (amongst others) mechanical, electronic and software "blocks".

Galinier resumes: "Systems engineering is much more than just partitioning hardware and software – systems also include

In many organizations, object-oriented techniques and the UML are being used by systems engineers to specify the architectural structure of the overall system.

mechanical and optical components, and it is crucial to be describing all of your system if you are to see how it interacts. For example, an infra-red camera is in fact mechanics, optics, software and electronics – and to describe the architecture of the system, we use UML."

On TTM's recommendations, the THALES' shareholders have chosen to adopt I-Logix' Rhapsody for both software and systems development. Rhapsody automatically generates production-ready code at any point in development. It does this by employing a real-time execution framework "beneath" the GUI, which means Rhapsody can automatically generate not only the source code, but also make files and even invoke the compiler and linker. As for Rhapsody's use for systems development, Galinier praises the tool's GUI and simulation capabilities.

But Rhapsody's aptness for software and systems generation was, according to Galinier, only part of the reason I-Logix was selected: "The tool's selection is not only based on technical satisfaction but also on the tool provider – their world-wide subsidiaries, and the level of technical co-operation they can provide. We often need to customize tools and for that we need the help of I-Logix – for example to set up interfaces with other tools such as Telelogic's requirements management tool DOORS. In addition I-Logix is in the process of introducing bridges based on XMI™, the XML-based Model Interchange

standard that allows models to be imported and exported into both its Statemate MAGNUM™ and Rhapsody tools. Statemate MAGNUM is a widely used tool for systems engineering following a functional decomposition approach.

The fact that I-Logix is focusing on standards in terms of languages (like C, C++ and Java) and in terms of notations/interfaces (UML and XMI) is perceived, by THALES, as a major plus because it then becomes possible to share designs with other tools utilizing the same standards. Also, by adopting and building on standards, I-Logix' longevity is perceived as more guaranteed than that of vendors developing tools that use proprietary languages, notations and interfaces: and supplier longevity is very important to THALES as projects can be as long 25 years.

THALES uses many programming languages and needs to maintain code throughout its numerous projects and UML is certainly helping define, generate and maintain such code. In addition though, UML is also proving a worthwhile notation and methodology in defining system architectures – and here its uses extend outside of THALES.

With the executability of the Rhapsody UML models, these designs can be visualized, easily understood, and validated by virtually all of the team members involved.

"By defining and describing system architectures in UML it is possible to communicate with the customer," concludes Galinier. "It's possible to have reviews with the customer and to share the same vision." In addition, with the executability of the Rhapsody UML models, these designs can be visualized, easily understood, and validated by virtually all of the team members involved.

THALES, having standardized on Rhapsody, plans to integrate with other CAD/CAE tools to create a powerful systems engineering environment – one that is both model and simulation based. ■

OMG MDA Seminars Travel to 7 Major Cities Across the US

In accompaniment with OMG's Fall 2001 Model Driven Architecture Seminar Series, OMG launched a Spring Series during the first week of April 2002. Traveling to Seattle, San Diego, Dallas, Chicago, Atlanta, Philadelphia and Washington, DC, the spring Seminar Series reached new audiences across the US. The goal was to inform and educate the IT community about Model Driven Architecture, OMG's standard that promises to extend software lifecycle to twenty years and beyond.

The following presentations were made at each free seminar;

- **Model Driven Architecture: An Introduction** by Richard Soley, Chairman and CEO of the OMG
- **OMG MDA: An Idea Whose Time Has Come** by Paul Harmon, Senior Consultant Distributed Architecture, and eBusiness Service for Cutter Consortium
- **MDA Reality and Implementation** by David Frankel, Chief Consulting Architect at IONA Technologies.

If you missed this most recent Seminar Series but would like more information about MDA, please visit our website for an extensive overview and explanation of the standard, as well as for case studies and cutting-edge products/services that currently implement Model Driven Architecture. The URL is www.omg.org/mda. ■

CORBA®, The Information Brokerage®, CORBA Academy®, IIOP®, XMI®,and the Object Management Group logo® are registered trademarks of the Object Management Group. OMG™, Object Management Group™, the CORBA Logo™, "The Middleware That's Everywhere™", the CORBA Academy logo™, MOF™, CWM™, OMG Interface Definition Language™, IDL™, CORBAservices™, CORBAfacilities™, CORBAmed™, CORBAnet™, UML™, the UML Cube Logo, "We're Known By The Companies We Connect™", Model Driven Architecture™, MDA™, OMG Model Driven Architecture™, OMG MDA™, "The Architecture of Choice for a Changing World™", and Unified Modeling Language™ are trademarks of the Object Management Group. All other products or company names mentioned are used for identification purposes only, and may be trademarks of their respective owners.

What's Happening

Upcoming OMG Technical Meetings, Workshops and Conferences / Exhibits

April 2-9, 2002
OMG Model Driven Architecture Seminars
Seattle, WA, San Diego, CA, Dallas, TX, Chicago, IL, Washington, DC, Atlanta, GA and Philadelphia, PA

April 22-26, 2002
OMG Technical Meeting
Yokohama, JAPAN
Sponsored by Distributed Objects Promotions Group (DOPG)

April 28-May 2, 2002
DAMA INTERNATIONAL SYMPOSIUM AND WILSHIRE META-DATA CONFERENCE
San Antonio, TX USA
OMG Members presenting on the Common Warehouse Meta-Model (CWM) specification

May 15 & 16, 2002
Enterprise UML
Le Meridien Palace, Manchester UK
20% discount for OMG members

June 17-21, 2002
7th International Conference on Reliable Software Technologies, Ada-Europe 2002

June 24-28, 2002
OMG Technical Meeting
Orlando, FL USA

July 15-18, 2002
Realtime & Embedded Workshop
Arlington, VA USA

September 2002
OMG Technical Meeting
Helsinki, Finland
Co-sponsored by HM&V Research, BEA Systems Oy (Additional sponsor TBA)

September 17-20, 2002
EDOC 2002 Conference
Lausanne, Switzerland

October 2002
UML for Enterprise Applications Workshop
San Francisco, CA USA

November 18-22, 2002
OMG Technical Meeting
Washington, DC USA

Smiths Aerospace

continued from page 1

approaches to design and implementation. The OO approach was successfully used even though the languages used for implementation (mainly Ada 83 and C) are not OO languages.

Problem:

While the analysis and OO design and implementation phases were well specified and successful in isolation, the transition from structured analysis into an OO design and implementation was difficult. Also, the software tools used to support the software development process were becoming dated, requiring the engineers to use dedicated workstations when the rest of their work could be done on their desktop PCs. A better development process was required – one that would allow a smooth transition from requirements analysis into design and that could be deployed on the desktop PCs.

Solution:

According to Shaun Cullimore, Smiths Aerospace Software Technology Manager, the process chosen needed to be capable of supporting a wide range of software developments for products in both the defense and civil marketplaces. The software in these products would also vary in size, implementation language and customer. To minimize risk selecting a PC-based tool, the selection criteria required that the vendor be able to provide training and consultancy support.

Smiths selected Real-time Studio from ARTiSAN, says Cullimore, because of the company's strong focus on real-time embedded applications, their technical expertise in the UK, and the price of the product. The relative maturity of their tool in a volatile market, together with the strength of their consulting capability (engineers with experience of real time embedded system development), and emphasis on process, were also key factors in making ARTiSAN the preferred partner for Smiths.

Because there was no opportunity to free staff for extended tool and process evaluation exercises, Smiths chose to use Real-time Studio on a "live" project. And, to minimize risk, Smiths incorporated a training and consultancy package from ARTiSAN.

"We needed to adapt ARTiSAN's 'out-of-the-box' lifecycle so we could maintain a separate 'requirements view' and 'design/implementation view' without using two models for every project," comments Cullimore. "Agreement with the customer of exactly what constitutes the software requirements baseline is a key milestone in our programs. A process which distorts that view through elaboration or pollutes it with implementation detail, would not be acceptable to us. We also wanted a process that was independent of any tool, though we would obviously be seeking the best tool to support it."

A single ARTiSAN model is now used to accommodate both the requirements view and the design and implementation view. "This suits us better than maintaining two separate models as it is so difficult to maintain consistency," says Cullimore. From these discussions, Smiths has produced a Code of Practice for OO Analysis and Design that is used across all new projects. Smiths uses industry standards whenever they can, and benefits from ARTiSAN's active participation in the Object Management Group (OMG) who manages UML. ARTiSAN's Real-time Studio supports Unified Modeling Language (UML) but also provides unique real-time extensions that go beyond UML.

Smiths has produced a Code of Practice for OO Analysis and Design that is used across all new projects. Smiths uses industry standards whenever they can, and benefits from ARTiSAN's active participation in the Object Management Group™ (OMG™), who manages UML.

"Since the beginning, we have actually implemented the new methodology and tools on four major real-time projects and many projects developing supporting test equipment," says Cullimore. Smiths continues to utilize ARTiSAN training and consultancy as engineers begin to use the process on new projects. The tool and process have been employed on projects ranging from equipment to switch electrical loads on military aircraft, to helicopter display systems.

Benefits:

"As a result of adopting our new process and the ARTiSAN tool, we now find it easier to move software engineers between projects. This is a major benefit given the diversity of our projects," Cullimore claims.

ARTiSAN continues to develop Real-time Studio, and following their initial experience with the tool, Smiths has indicated they are keen to use some of the newer facilities that it provides. For example, the "Package-based Object Repository" of Real-time Studio, enables a peerless combination of upstream team collaboration and downstream development stability. Teams now have the flexibility to simultaneously share the centralized project repository as well as partition it for separate controlled development. A new feature of Real-time Studio that appeals to Smiths is Configuration Management of the model via package controls that reduces or eliminates problems traditionally encountered with control of a model by numerous engineers.

ARTiSAN supports open systems and industry standards such as UML, which enables prime contractors, sub-contractors and their customers to communicate effectively. Support for automatic documentation generation from templates means that generated documents can be adapted to conform to standards such as: MIL-STD-498, J-STD-016, IEEE/EIA 12207 and ISO 12207. This enables developers to focus on engineering activities while highly evolved tools automate the documentation tasks.

Because ARTiSAN's Real-time Studio makes it easy to maintain code and design consistency, and as the development teams become more familiar with the new process and technology, Cullimore anticipates that Smiths will achieve even greater time-to-market acceleration. Additionally, he believes by maintaining greater code and design consistency Smiths will find it easier to leverage the software for future projects because most components developed are in a form that can be readily and easily reused.

Ultimately, this should result in significant development cost reductions. ■

The Siegel Guide

Jon Siegel , Ph.D.

Vice President, Technology Transfer

Object Management Group

OMG's First MDA-Based Specification: Gene Expression, from the Life Science Research DTF

In September 2001, OMG members voted to make the Model Driven Architecture (MDA) the base for future specifications. Under the MDA, the organization can now adopt specifications defined in UML and expressed on virtually any middleware platform: as Web Services, in XML/SOAP, or Enterprise JavaBeans, or many others.

Less than five months later, OMG members adopted the first MDA-based, non-CORBA OMG specification – Gene Expression, from the Life Science Research (LSR) Domain Task Force (DTF). This specification takes good advantage of the MDA: It's a data model, defining no operations, and even though the only platform-specific realization in the standard is on XML, its standard data structures have been programmed in Java, Perl, and C++. Based on a single detailed UML model, the XML DTD and declarations in Java, Perl, and C++ are all compatible, allowing data to be passed from a program in any of these languages, via XML generated from the DTD, to any other language, without loss of information.

To see how this specification benefits from MDA, we'll take a closer look starting with a brief biology lesson:

A gene is a section of DNA that defines a protein. Cells control what they do by controlling the types and quantities of proteins they produce. The process of transcribing a gene inside a cell and producing the protein that it defines is called gene expression. When something goes wrong with this system, a cell can do unexpected things – it could become cancerous, for example, or create the symptoms of a genetic disease.

By taking cells from a culture or other source (a tumor, perhaps) and breaking down the cell walls to release the contents, a scientist can prepare a solution rich in the genes that were being expressed when the cells were, well, interrupted. How does the scientist find out exactly which genes these are? Companies and laboratories have samples of thousands of DNA strings and fragments, each labeled with the gene that it represents. By comparing DNA from an experimental cell culture with known DNA, a scientist could tell which genes were being expressed, but it's impractical to do thousands of comparison experiments one-at-a-time. How can this process be automated?

Enter the Microarray, an array of many hundreds of different known DNA samples bonded to a microscope slide or other suitable carrier. Because of the way DNA strands match up to form the double helix that you remember from biology class, DNA from an experimental preparation will bond to the spots on a Microarray that contain DNA from its gene, and not bond to the other spots. A scan then produces a tiff file of the Microarray. Software converts data for each spot into the level of expression of the gene that the spot represents.

These experiments produce unbelievable amounts of data. Each microarray contains thousands of spots (as many as four hundred thousand), each of which must be described completely (species, cell type, pathology, and so on), and an overall investigation may include many microarrays, each used multiple times. And, each experiment produces a profile with a multiply-dimensioned data value for each spot on an array.

This is too much data to deal with by hand. Data collection, storage, indexing, and analysis must be done by computer, and the widespread use of this technology means that data, analyses, and even software must be shared. The Gene Expression Specification is OMG's answer to this requirement. Defined in UML, the specification is a data model for gene expression. Michael Miller of Rosetta Biosoftware and editor of the specification tells how it was produced:

"Last March, after some initial teleconferences, the submitters decided on the initial set of packages and classes for the model. That model, which became MAGE-OM (Microarray Gene Expression Object Model), allowed all of us to focus on the domain of Gene Expression Data without worrying about any particular implementation. When we finally thought we were ready, we used the model, as exported to its XMI version, to custom generate the MAGE-ML.dtd. These two documents (lifesci/2001-01-02 and 03) along with the revised Gene Expression submission (lifesci/2001-01-01) make up the normative Adopted Specification, dtc/02-02-04.

"Subsequent efforts have produced open source implementations of the model in Java, Perl and C++ along with parsing code for the Java and Perl to and from MAGE-ML (http://sourceforge.net/projects/ mged/). There are plans for a reference implementation to a Relational Database as well. Three organizations are well on their way to using MAGE as the basis of creating their repositories: EBI is using MAGE as the basis of their ArrayExpress; NCBI for GEO; and NCGR with GeneX. A host of other private and public organizations that deal with Gene Expression data, including Rosetta Biosoftware, Agilent, Affymetrix, Iobion, and several others have announced their support for MAGE."

Thanks to the many advantages of the MDA, companies are taking a fresh look at OMG DTFs as the best place to establish industry standards. Gene Expression may be the first MDA-based specification, but the next few are already in the pipeline and many more are scheduled to start up in the near future. ■

CORVAL2 Delivers CORBA 2.3 Test Suites for Conformance Testing and Branding Program

In February 2002, the European Commission funded EUR 1.7 million CORVAL2 project was successfully concluded (www.opengroup.org/corval2). The project goals were to develop new techniques and test tools, and a Branding Program, for CORBA 2.3. The IST Enhanced Techniques for CORBA Validation Project (to give it its complete title) was a 26 month project involving OMG and five OMG member partners.

The key project deliverables are the test suites, VSOrb 2.3 and VSJOrb 2.3. The VSOrb 2.3 test suite, which has been available since November 2001, ensures correct and consistent interfaces and functionality for CORBA 2.3 conformant ORBs written in C++. A Beta version of VSOrb 2.3 for C++ is available for free download from the CORVAL2 Web site (www.opengroup.org/corval2/download.html). The VSJOrb 2.3 test suite is identical in purpose to VSOrb 2.3, but is designed for validating CORBA 2.3 conformant ORBs written in Java. This test suite was released in March 2002.

Both test suites focus on defined APIs and software services, rather than product implementation. The test suites include tests for API Declaration, API Behavior, IDL Syntax, IDL Skeleton and GIOP/IIOP. Thus ORB developers retain the flexibility to differentiate their ORBs for operating environment, marketplace etc., while at the same time providing a standard interface for applications. A commercial test suite license for one ORB and one language binding costs $10,000. More information on the test suites, including more detailed pricing information, can be found at www.opengroup.org/testing/testsuites/vsorb2_3.htm. Test suites licenses can be purchased either from The Open Group or from Fraunhofer FOKUS' commercial arm, Testing Technologies IST GmbH (www.testingtech.de).

In October 2001, The Open Group launched The Open Brand for CORBA 2.3. The VSOrb 2.3 and VSJOrb 2.3 test suites are used to validate ORBs submitted for branding. For more information on branding, please see http://www.opengroup.org/registration/wotis.htm.

The bulk of the test specification and development was carried out by Fraunhofer FOKUS (formerly GMD FOKUS) of Berlin, Germany, with considerable assistance by CORBA vendors IONA/OOC of Karlsruhe, Germany and Fujitsu Services (formerly ICL) of Dublin, Ireland. Project management and technical support were supplied by The Open Group of Reading, UK, and marketing support and OMG liaison was supplied by Eric Leach Marketing of Ealing, UK. ■

Stay on Top of the OMG, MDA, CORBA and the UML!

Subscribe today to *OMG News* . Subscriptions are free of charge and new issues are published each quarter. Please specify whether you would prefer to receive an e-mail alert when the newest issues are available for download electronically or if you would prefer to receive a hardcopy via mail.

❑ YES, I want to receive OMG News.

❑ Please notify me quarterly, with an e-mail alert. ❑ I'd prefer to receive the quarterly hardcopies.

NAME _____ TITLE _____

COMPANY _____

STREET ADDRESS (required for hardcopy subscription) _____

CITY _____ STATE _____

POSTAL CODE _____ COUNTRY _____

EMAIL (required for email alert) _____

Please fax this form to OMG: ATTN: Nancy Lenehan, +1-781-444 0320, mail to Object Management Group, 250 First Avenue, Suite 201, Needham, MA 02494 or e-mail your information to subscriptions@omg.org.

EDITOR: Nancy E. Lenehan

OMG HEADQUARTERS
First Needham Place, 250 First Avenue
Suite 201, Needham, MA 02494 USA
Telephone: +1-781-444 0404
Fax: +1-781-444 0320
Email: info@omg.org
WWW: http://www.omg.org

OMG JAPAN
2-21-15 Akasaka OS Building
Akasaka Minato-ku,
107-0052
Tokyo, Japan
Telephone: +81-3-3560 6147
Fax: +81-3-3589 1903
Email: soken@omg.org

UNITED KINGDOM
Eric Leach
ELM
Sinclair House, The Avenue
West Ealing, London W13 8NT
United Kingdom
Telephone: +44-20-8758-7587
Fax: +44-20-8758-7517
Email: elm@omg.org

CONTINENTAL EUROPE
Roberto Zicari
LogOn Technology Transfer GmbH
Beleichstrasse 5, 61476 Kronberg
Germany
Telephone: +49-6173-9558-50
Fax: +49-6173-9558-68
Email: logon@omg.org

INDIA
D.K. Bose
ADA Software & Services
Vostok House, KB21 Sector III
Salt Lake City, Calcutta 700091, India
Telephone: +91-33-3377916
Fax: +91-33-9-10-00 12
Email: ada@omg.org

MIDDLE EAST
Mark Ragel
Al-Gosaibi Information Systems
Capital Centre, Ground Floor, Unit G6
PO Box 10528, Manama, Bahrain
Telephone: +973-274 090
Fax: +973-214 645
Email: markrage@batelco.com.bh

BRAZIL
Sergio Mainetti, Jr.
Director/Consultant
Visionnaire
R. Grã Nicco
113-Bl. B 5o. Andar
Curitiba, Parana 81200-200, Brazil
Tel: +55 (41) 373-7400
Fax: +55 (41) 373-7400
Email: mainetti@visionnaire.com.br

OMG GOVERNMENT REPRESENTATIVE
Hal Peterson
6701 Democracy Boulevard, Suite 300
Bethesda, MD 20817-1574
Tel: +1-410-987 3850
Fax: +1-301-564 9619
Email: jhp1@m-abc.com

OMG News is a quarterly publication from the Object Management Group, Inc. © 2002, All rights reserved. Reproduction or use of editorial or graphic content without the written permission of OMG is strictly prohibited.