

Visual Modelling: past, present and future

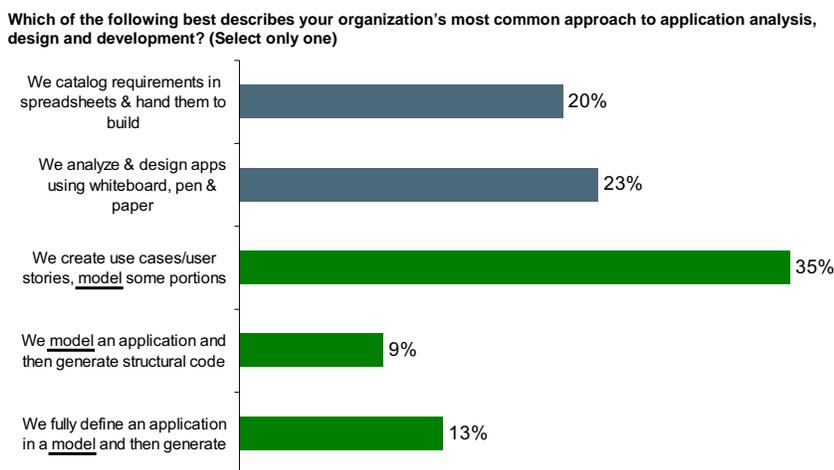
Andrew Watson
Vice-President and Technical Director
Object Management Group™

The history of visual modelling in the software industry divides cleanly into two eras - "Before UML®" and "After UML".

Before the first Unified Modelling Language™ (UML) standards were published in the mid-1990s, visual software modelling was plagued by the incompatibility of different notations created by different modelling gurus. The absence of a standardised notation deterred potential users, and as an inevitable result the modelling tools market was tiny and fragmented. The few tools that were available suffered from a lack of investment; many only allowed sketching of software designs, lacking facilities for checking the diagrams' internal consistency or automatically processing the information they held. These early visual diagrams were useful as design aids or documentation, but were rarely integrated into the software development lifecycle.

The UML standard changed all that, and triggered the dramatic growth in visual modelling that has led to its widespread use not only in software design, but also in non-software disciplines such as systems engineering, and in the business domain.

In the mid-1990s the Object Management Group (OMG™) acted as forum for agreement between the thought-leaders in the nascent software modelling field. The time was exactly



From a Commissioned study conducted by Forrester Consulting on behalf of Unisys
Base: 132 respondents

Figure 1: According to Forrester, most software developers use modelling

right for the emergence of a standard. Researchers and early-adopters had accumulated a great deal of modelling experience, but were being held back by the lack of a widely-used notation. Once UML eliminated this major obstacle to widespread use of visual modelling, its use grew spectacularly. In 1995 modelling tools were used in a tiny fraction of software projects; by 2006 Gartner estimated that more than 10 million IT professionals used UML, and by 2008 over 70% of software development organisations worldwide were using it.

UML has become the lingua franca of software development, allowing engineers to exchange their designs freely. Nowhere is this better illustrated than in the software for the new James Webb space telescope, scheduled for launch in 2013. To aid communication and help meet stringent reliability and performance goals, all the software being built for the telescope by NASA, the Canadian and European space agencies and all their subcontractors is being designed using UML. Organisations across the world are cooperating on guidance software, a command data handling system and software for the science module housing four different light-receiving instruments. All will be integrated in the telescope itself, destined for earth orbit at an altitude of 940,000 miles (four times the distance to the moon).

UML is supported by every major commercial IT vendor, as well as a flourishing selection of Open Source tools. UML books & training are widely available, and the OMG Certified UML Professional (OCUP™) and OMG-Certified Real-time and Embedded Specialist (OCRES™) certification programmes have allowed ten of thousands of engineers and architects to establish their UML credentials. UML has changed the software world.

As UML use has grown, continuous feedback from the user community and investment by tool vendors has helped the standard evolve and mature. The original UML 1 standard of 1997 was backed by 21 OMG member companies; feedback from dozens more submitted via OMG's issue-reporting system helped refine it, flushing out remaining inconsistencies. In 2005 OMG published UML 2, a major revision largely based on the same familiar diagram notations, but using a more rigorous underlying modelling infrastructure specified using OMG's Meta-Object Framework (MOF™). While some designers still use UML merely for sketching designs to share with colleagues, UML 2's MOF foundation means that today's UML diagram is more than just a pretty picture. A MOF-aware modelling tool can capture the meaning of diagram elements and their relationships in machine-readable form, and use this to reason about the design, perform consistency checks, and even automatically generate parts of the application code. Creating, storing and transforming machine-readable models in this way puts modelling at the heart of the software production process, and forms the basis of OMG's Model Driven Architecture® (MDA®).

Putting modelling at the heart of software development

MDA helps software users cope with two key realities of today's software environment; multiple implementation technologies, and the need for maintenance over extended software lifetimes. MDA uses OMG modelling standards to create and manipulate precise, detailed, machine-readable models that represent application structure and behaviour independently

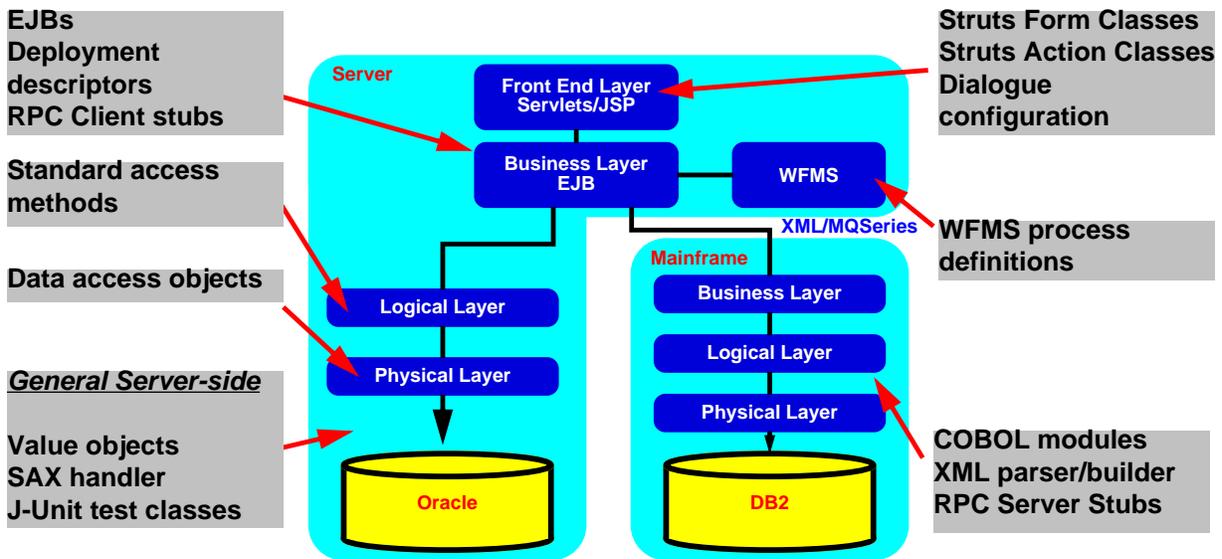


Figure 2: Code artefacts generated in Deutsche Bank's MDA application

of what programming languages, operating systems, databases or other technological platform being used to implement them. MDA's basis in freely-available standards has resulted in a thriving community of tool vendors and open-source tool offerings. Use of standards allows tools from multiple vendors to be used together on a single project - a vital feature, since even the largest tool vendor may not provide support for all necessary software platforms

A case study on the use of UML and MDA at Deutsche Bank Bauspar shows how model-centred approaches are reducing software ownership costs at major international corporations. This project involved a maintenance upgrade to an eighties-vintage COBOL back-office mainframe running CICS & DB2, marrying it to a Web-based front-end to give users sitting at 30,000 client machines in 1250 Deutsche Bank offices across Germany access to details of their savings-and-loan accounts (which had previously only been available by submitting paperwork to Bauspar's back office). A new three-tier system based on the original mainframe was designed using UML and coded largely through transformations directly from the machine-readable UML models using Interactive Objects' ArcStyler tool. Not only was Java code for the Oracle- and Unix-based middle-tier created in this way, so was the all-important COBOL interface code to run on the existing mainframe. A dozen or so different kinds of software artefacts, including COBOL modules, Oracle database schemas, DB2 database schemas, EJBs, XML-RPC interface definitions and JUnit test classes were all created from UML design models using MDA techniques, with around 60% of the new business logic and 90% of the database-related code being created this way. The customer estimated the cost savings at around 40% compared to hand-writing code from sketched design diagrams, comparable to the savings found in several other studies.

Given the large, measurable return on investment from using model-driven development, it's

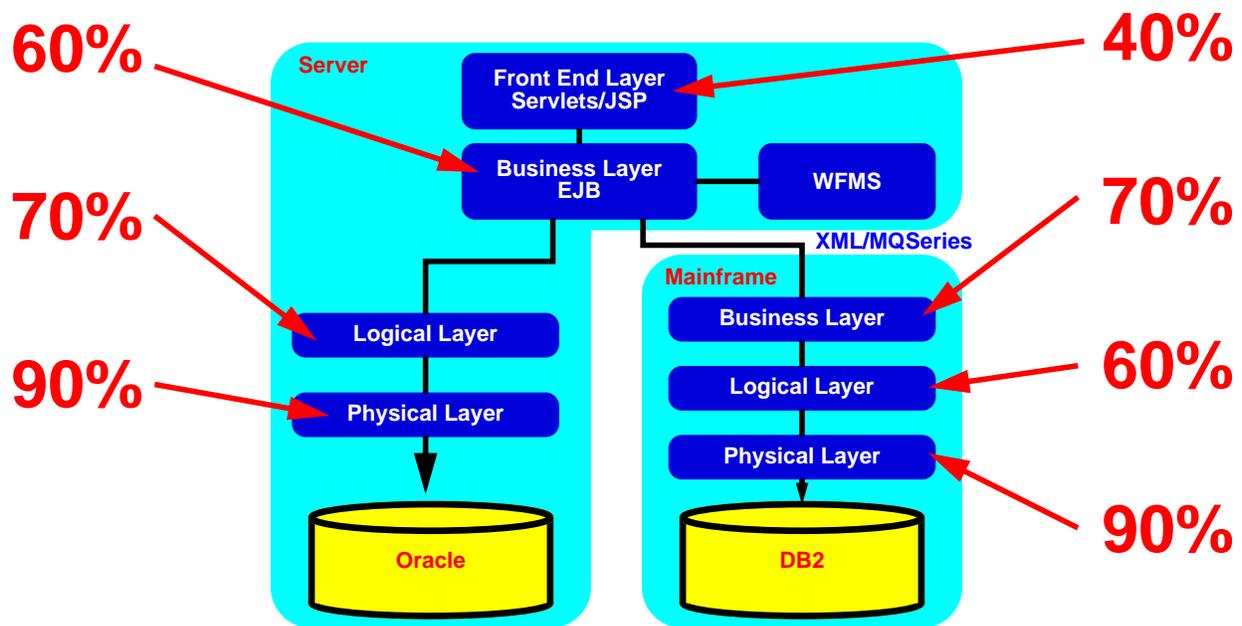


Figure 3: Code generation percentages in Deutsche Bank's MDA application

hardly surprising that its use is rapidly taking off. As recently as 2004 BZ Research found only 4% of developers were actually using such techniques, with a further 12% planning to at some later time. By 2008, a Forrester study for Unisys showed that 22% of software development projects were deriving at least some of their application code directly from models (see figure 1).

OMG's visual modelling family

Although UML is OMG's best-known modelling standard, it is in fact just one of a diverse family of OMG modelling standards, all based on a MOF foundation. UML itself is deliberately domain-independent and platform-agnostic, but includes a lightweight customisation mechanism for creating UML "dialects" (called "profiles"). Loading a profile definition into a standard UML tool allows it to support (and enforce) the characteristics of a particular target platform or application domain. OMG has defined standard profiles for (amongst others) real-time systems, fault-tolerant systems and CORBA-based distributed computing platforms, and work is underway on a standard profile for Service Oriented Architecture (SOA) platforms. Standard UML profiles are also available to customise UML for use with specific languages such as Java and XML. UML's profile mechanism allows UML tools to support UML-based languages customised for specific platforms and domains; in other words, to support Domain-Specific Languages (DSLs).

More extensive use of profiles allows UML to be customised for entirely new domains outside the realm of software development, such as systems engineering. OMG Systems Modelling Language™ (OMG SysML™) is a UML-based language for graphical modelling of complex systems that include hardware, software, information, processes, personnel, and

facilities. It's a testament to UML's adaptability that a modelling language for a completely new domain can be created in this way. Because SysML is based on UML, it's easy for systems engineers familiar with UML to learn it, and for existing UML tool providers to support it. The standard MOF foundation means that MOF tools can manipulate SysML models as easily as UML models.

MOF also supports other modelling language standards unrelated to UML or software development, such as OMG's Semantics of Business Vocabulary and Rules (SBVR) standard. SBVR defines a MOF metamodel for storing definitions of business terms and the operational rules and policies by which all businesses operate, but represents them to the human user in an easily-comprehensible, text-based notation. Although SBVR uses no diagrams at all (let alone UML diagrams), because it's defined in terms of the same MOF foundation as UML, MOF model-manipulation tools can be used with SBVR-defined models to bring the benefits of precise, model-based reasoning to the task of representing how any businesses operates.

Another important business modelling specification in OMG's modelling stable is the Business Process Modelling Notation™ (BPMN™), for creating graphical representations of how a businesses functions. Like SBVR rules, BPMN diagrams are readily understandable by everyone involved in the business, including the business analysts that create the initial drafts of the processes, the technical developers responsible for implementing the technology that will perform those processes and the business people who will manage and monitor them. BPMN applies sound modelling principles to representing business processes, even those that are completely human-driven and involve no software at all. Today BPMN is working the same standards magic that UML did in the mid-90s; by removing uncertainties over notation, it has released a surge of pent-up demand for good business process modelling tools. More than 50 tools now support BPMN, and three books devoted exclusively to the language were published between April and August 2008 alone. Just as for UML, OMG has put in place a certification programme to allow practitioners to demonstrate their knowledge of the notation, and the existence of the standard is enabling rapid growth in business standards expressed in the new notation. A recent survey by Jan Recker of Queensland University of Technology found BPMN users spread across all continents and countries, with 40% working in government and 60% in the private sector. BPMN's growing use across industries from manufacturing to medicine is helping organisations become more agile and efficient, and moving visual modelling out of the software development world into widespread acceptance across business and government.

Conclusion: OMG and the future of visual modelling

Today OMG is the seminal organization in the modelling space, responsible not only for UML, the most widely-used visual modelling language for software, but also a growing family of successful Domain-Specific Languages for diverse application domains. SBVR and BPMN are repeating UML's standardisation success of the mid-1990s by bringing the benefits of standardised modelling syntax to business governance, allowing all aspects of all businesses to benefit from the precision and ease-of-understanding offered by good

modelling notations.

OMG's modelling expertise, based on the versatile, standardised MOF framework, allows the organisation to deliver modelling standards that answer the needs of all aspects of business and government. OMG has almost 15 years' experience creating modelling standards, and includes all the major modelling tool vendors and a host of end-user organisations with unparalleled modelling expertise in diverse domains from manufacturing to medicine. OMG's introduction of UML standards in the mid-1990s changed the software world by making visual modelling commonplace; its pioneering work on Model Driven Architecture uses precise software models throughout the development lifecycle to reduce dramatically the total cost of software ownership. Today, OMG's modelling standards for business and engineering are poised to change the world again by placing modelling at the heart of the way all businesses and governments operate.

Sources

"UML adoption making strong progress", Alan Zeichick, Software Development Times, 15 August 2004

"View DSLs and UML as 'Fraternal Twins', Not Competitors", David Norton, Gartner Research, 29 September 2006

"BPMN Modeling - Who, Where, How and Why", Jan Recker, BPTrends, May 2008, <http://tinyurl.com/5f2fh8>

"Modernising Software Development Through Model-Driven Development", A commissioned study conducted by Forrester Consulting on behalf of Unisys, 13 August 2008, <http://tinyurl.com/5nrfss>