Model Driven Development Maturity

In a recent study conducted by Forrester Research for Unisys, the usage of model-driven development (MDD) was noted to be widespread amongst the IT community in general.

Clear benefits can be derived from this type of approach – but the two most important benefits defined in the report were the ability to enhance productivity and the ability to rapidly respond to business changes. Both of these key enhancements should be crucial defining factors in our current economic climate (and indeed in business development in general), where the smallest of differences in these two areas can deliver dramatic business results.

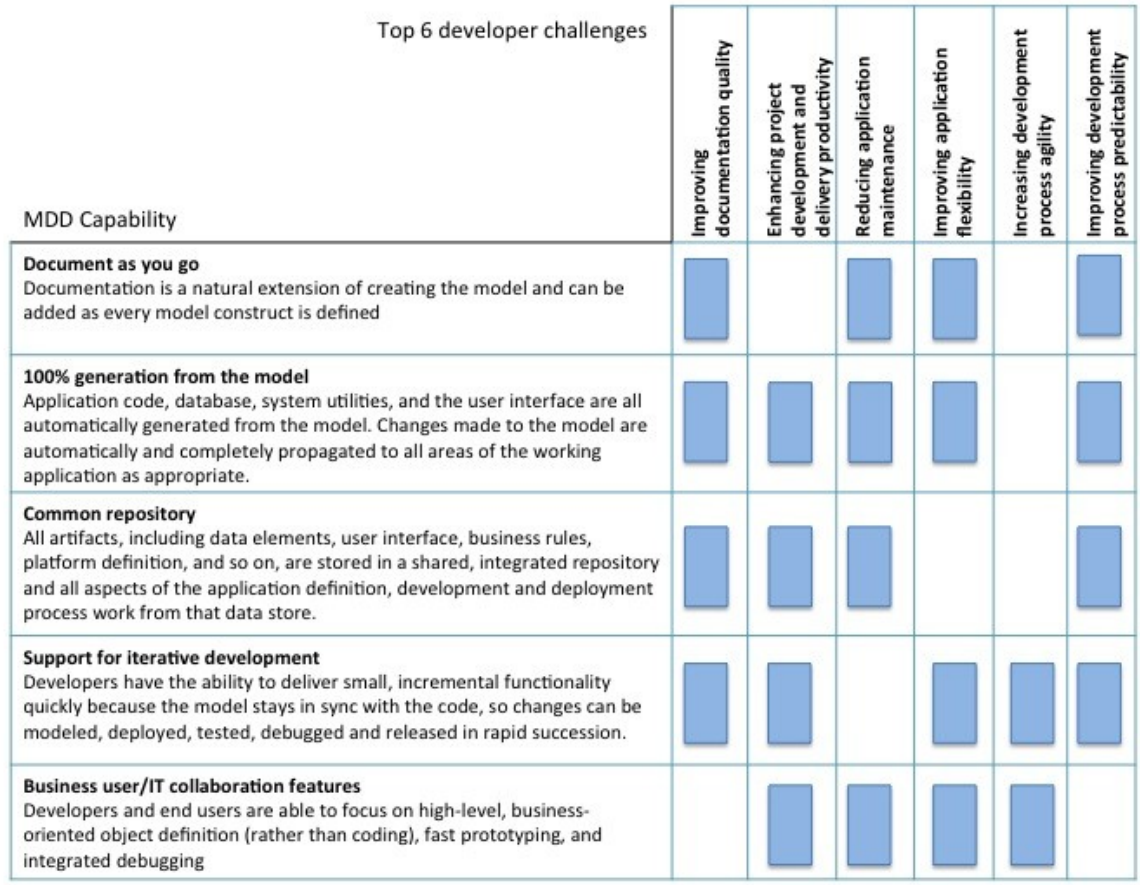
The report itself defines 6 common problems that companies are trying to resolve:

- Improving the productivity of project development and delivery
- Reducing the amount of time spent maintaining existing applications
- Increasing the flexibility of applications
- Improving the agility of development processes
- Improving the predictability of development processes
- Improving the quality of documentation

With a constant barrage of new technologies being launched onto the market, it is important to find the right approach that works for your organisation, but which can also provide the key tangible benefits you need to remain at the cutting edge of technology, operating in a competitive and challenging market.

MDD provides such an opportunity – but how do you know what to look for? The report from Forrester has a summary of the key features you should be looking for:

- "Document as you go" features that make the task of describing the model and application integral to the process – and relatively painless for developers
- 100% generation of the application – including the code, database, user interfaces and system utilities – directly from the model
- A common repository to enable effective sharing and versioning of artifacts, which improves development process predictability
- Support for iterative development, which helps developers deliver better quality software faster
- Business user/IT collaboration features that help developers and user experts communicate more effectively.



MDD Capability \ Top 6 developer challenges	Improving documentation quality	Enhancing project development and delivery productivity	Reducing application maintenance	Improving application flexibility	Increasing development process agility	Improving development process predictability
Document as you go Documentation is a natural extension of creating the model and can be added as every model construct is defined	■		■	■		■
100% generation from the model Application code, database, system utilities, and the user interface are all automatically generated from the model. Changes made to the model are automatically and completely propagated to all areas of the working application as appropriate.	■	■	■	■		■
Common repository All artifacts, including data elements, user interface, business rules, platform definition, and so on, are stored in a shared, integrated repository and all aspects of the application definition, development and deployment process work from that data store.	■	■	■			■
Support for iterative development Developers have the ability to deliver small, incremental functionality quickly because the model stays in sync with the code, so changes can be modeled, deployed, tested, debugged and released in rapid succession.	■	■		■	■	■
Business user/IT collaboration features Developers and end users are able to focus on high-level, business-oriented object definition (rather than coding), fast prototyping, and integrated debugging		■	■	■	■	

Based on a commissioned study conducted by Forrester Consulting on behalf of Unisys "Modenizing Software Development Through Model-Driven-Development August 13 2008

Document as you go

Forrester's study indicated that this requirement was the biggest challenge most companies faced when looking at IT developments. A lack of documentation can hinder the ability for an IT department to provide enhancements to systems in a quick and efficient manner.

MDD itself should improve the ability to document as you go. At its very core, the model provides a description of the application that is required, but a multitude of outputs also exists to ensure that everything is tracked and documented.

A specific instance of an MDD tool is Mendix, which provides the use of business rules in a visual editor, allowing all sides of the business to understand the requirements in a standard format, which can then be linked in diagrams called 'microflows' to show the integration of business logic through the process. This ensures that the documentation is always up to date, and gives you the very latest view of the business process flows. All the rules are stored in a centralised repository, allowing for quick and easy access for any modifications that are required, as well as reuse across this or other applications.

Effectively then, any changes that are made become self-documenting, implemented throughout the solution by real-time changes to a model.

100% generation from the model

MDD by its very nature allows developers to move quickly from business requirements to working models, ensuring that the development time overall is reduced significantly compared to traditional development methods. Cloud deployment allows for the application model to be uploaded to the cloud in one single click.

The Mendix domain model is one of the most powerful concepts of the MDD framework. A Drag and drop format allows both the business and IT to manage complex relationships and high level entities, and ensures a central management of changes to real-life entities and relationships, rather than technical models, regardless of the underlying systems.

Mendix links design and run-time technologies, ensuring that the integrated solution is a true reflection of the business model. It also means that developers who are working with this form of solution are much closer to the business process, and much closer to the end users – highly beneficial whenever change is required. It also allows for a much lower possibility of interpretation by the programmers – giving a clear and transparent view of the business requirements.

Mendix manages to do this within an open environment, using Java and standard SQL databases such as Oracle or SQL Server.

Common repository

A common repository in itself provides complete transparency to all stakeholders involved in a development. It allows for all the documentation to be kept in sync, ensuring consistency (providing it is used by all functions). NO matter what the context of the required content, the repository can service all parts of a development function – debugging, modeling, deployment. Multiple access points ensure that the information is always available to whoever requires it when they require it.

Versioning also ensures that users know they are looking at the very latest documentation, and provides a method of managing multiple copies of documents.

Finally, a common repository can improve the development process predictability, as proven application objects can be reused. The repository itself linked to versioning can be used to track and manage what gets deployed and where.

Support for iterative development.

Iterative development allows for better integration between the model and the application through the delivery process. Ideally, there should be a debugging process that is integrated into this iterative process, improving the overall quality of delivery.

An iterative development approach that is integrated in this way will allow both the business and the developers to work on small increments of requirements or improvements, whilst supporting effective feedback from both sides.

Overall, this approach will improve project efficiency and delivery, as well as allowing for greater flexibility in the application itself. Mendix is now integrated with an agile tool called Sprintr to provide this requirement, along with a team server for effective communications. This enables iterative development, including direct feedback from the application to the developers.

Business user/IT collaboration

Responding to change remains a high priority for businesses, whether through economic uncertainty, increased/new competition or the speed of technology developments. The ability to respond can be the difference between success and failure, so the ability to improve the level of collaboration between the business and IT as part of application development and delivery is a fundamental requirement.

MDD tools allow for this collaboration, which also speeds up the ability of developers to address any change requirements. The improved communication between the business stakeholders and IT ensures a more efficient and accurate response in development activity. The ability of both sides to continue to see and improve the business rules and microflows within Mendix ensure that this collaboration is formed on solid ground, improving communication in general.

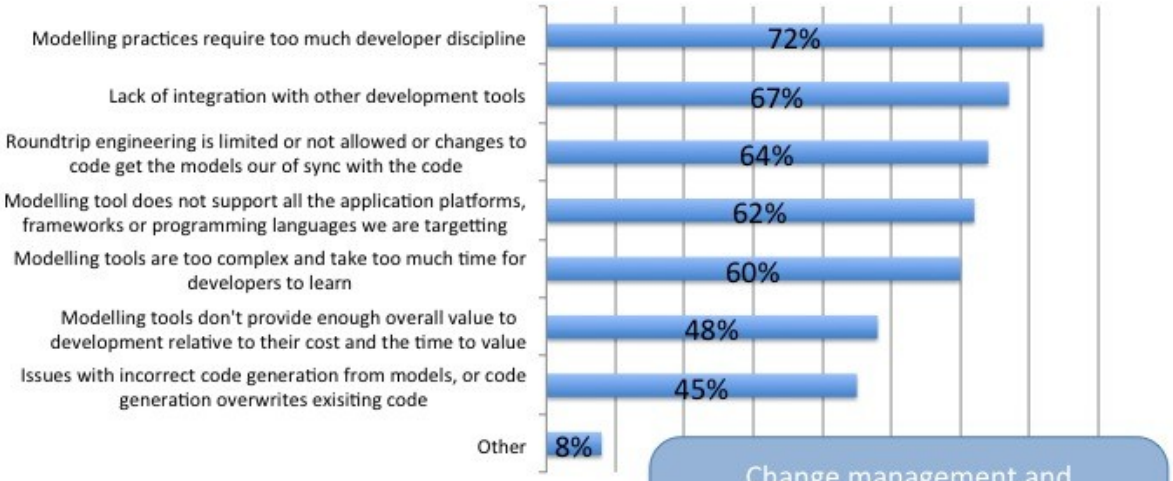
The output formats that Mendix supports ensure easy comprehension, making sense to both parties, ultimately leading to the development of software that better represents the business processes and requirements. Mendix also allows business users to view, feedback on and participate in software development.

The Forrester survey demonstrated that companies with a good integration and implementation of MDD tools had the following results:

- Improved programmer productivity – (41%)
- Enhanced ability to make design changes and update applications (38%)
- Improved documentation quality (38%).

This clearly demonstrates the improvements that businesses can feel from the use of MDD, with companies who develop a greater understanding of the inherent principles of this type of development gaining the most benefits.

Main challenges users are facing with tools are both soft and hard issues

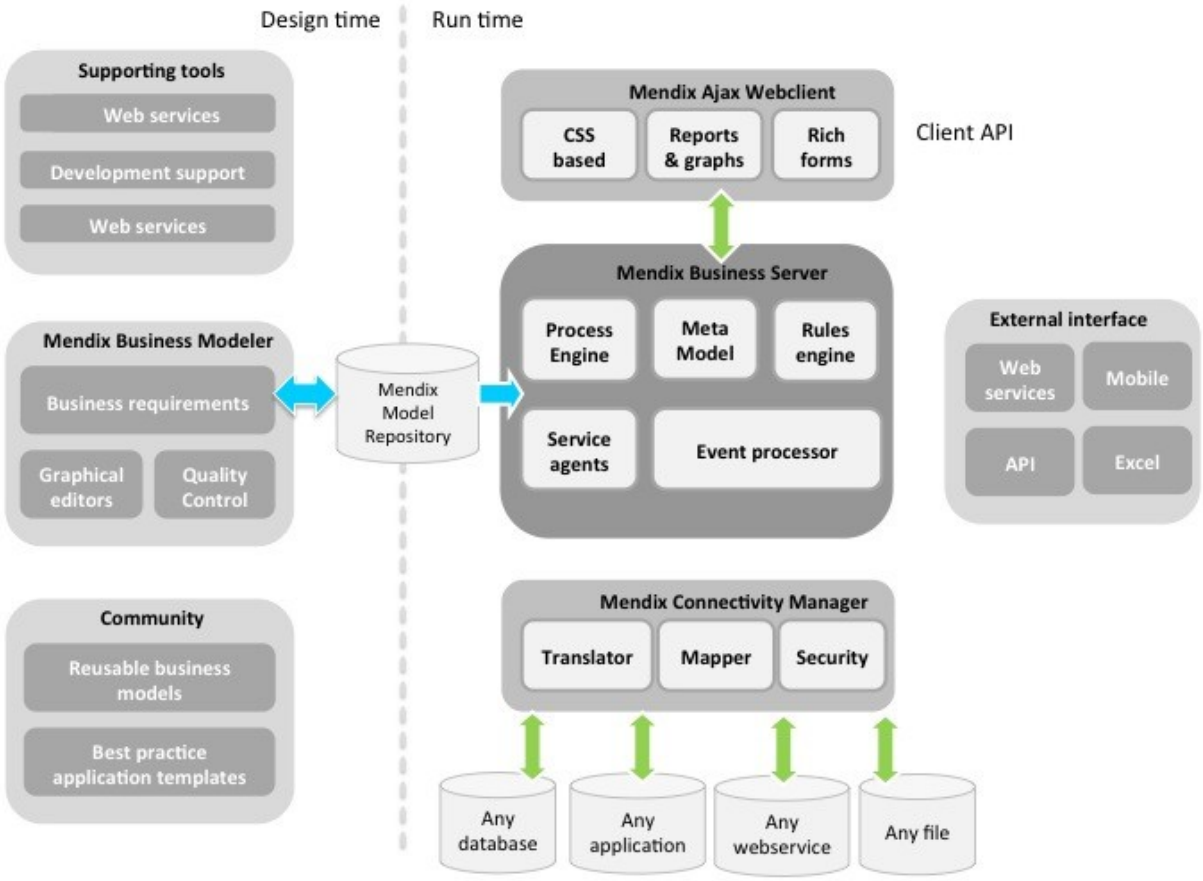


Source: Unisys Study, Q208

Change management and governance are the main challenges. Integration of modelling tools with other tools is the next main concern

Mendix Model-driven Platform Suite

The Mendix Model-driven Platform provides a complete suite consisting of the several "core components" and "additional development tools". The platform is built on an open, standards-based and extensible architecture.



Top level overview of the Mendix Architecture

This overview represents a top-level view of the technical architecture, showing the composition of the core components of the Mendix framework. As illustrated a clear distinction can be made between a design-time and run-time environment.

With the Mendix Business Modeller you define the specs of your service-oriented business application (SOBA) using graphical models that are automatically executed (using the "1 click deploy" feature) by the Mendix Business Server. The Mendix Business Server retrieves the data from underlying data stories (either new or pre-existing) through the Connectivity Manager and provides services to its clients. One of these clients is the Mendix WebClient that generates a Rich Internet (Ajax) User Interface for end users. As shown in the diagram, the presentation, logic and data are clearly separate from one another applied in a layered structure within the run-time environment.