

Why do our clients build models?

RAPID DEVELOPMENT

Development can proceed at top speed without having to wait for details of the platform design to be settled. In fact, separating out other concerns, such as the user interface technology, makes things go even faster.

FOCUS ON THE REQUIREMENTS

Models that exclude platform specific features can be focused exclusively on the application logic. So you have a clearer picture of the essential requirements and can evaluate, communicate and validate them without distraction. And this also contributes to rapid development.

STEADY PROGRESS

When, not if, the platform changes, only the markup files need to be updated. Our clients change the models only when the application itself changes. So small changes to the platform don't require reworking the models.

FLEXIBILITY IN DESIGN

And this means that our clients can freely experiment with platform design alternatives without having to worry about breaking the application logic. So they deliver superior applications on superior platforms.

SUPPORT FOR MULTIPLE PLATFORMS

The same models can be redeployed to multiple platform variations without having to change the models. Only the markup files change.

SAFETY AND CONSISTENCY

Once our client's application logic is modeled, it can be deployed on multiple platforms. This means that the same rules and constraints are deployed on each platform.

SUPPORT FOR CONCURRENCY

Concurrency is increasingly important. Most development languages introduce arbitrary sequencing. So when added concurrency becomes available on the platform, it is not obvious how to leverage it without tearing apart the application logic. Executable UML captures your application logic without embedding any unnecessary sequencing. So the models can be easily mapped to a wide variety of concurrency schemes without ever having to change the models themselves.

Our Clients

Our clients build complex high performance, mission critical systems that must operate safely.



We work with large organizations who want to reduce the cost of software development and deliver new features rapidly while ensuring safety and reliability.

Their computing platforms can be highly specialized with unique interfaces and multiple processors. The platform design typically evolves and eventually results in varied configurations to support different levels of features and spin-off products.

But even if the hardware is fixed, frameworks, network protocols, data storage formats, and other relevant software technologies are ever changing.

So it makes sense to separate the application logic, our client's valuable intellectual property, from the implementation details as much as possible.

That's why our clients preserve their application logic in platform independent models. These models are built, debugged, tested, documented and maintained separately from the platform. During implementation, the models are combined with markup files that map model elements to particular design patterns and platform specific implementation features.

What we do

We help our clients model large and complex systems using Executable UML by itself, or integrated with other tools, modeling languages, libraries and software services.

Model based systems engineering (MBSE) should deliver clear and compelling results, not just in the long term, but on a daily basis. The important ideas should be communicated, documented and validated. Hard problems should be identified and solved early on.

Valuable intellectual property should be preserved for fast redeployment on varied and evolving computing platforms. And the models should lead directly to efficient, reliable and maintainable production code.

But this is all easier said than done. It is easy for teams to get distracted modeling the wrong things at inconsistent levels of detail. Models can be built in the wrong order, leading to unnecessary rework or poor designs. Opportunities for parallel development can be missed, wasting valuable time and resources. Key model reuse patterns can be overlooked resulting on large scale wasted effort and needless extra code.

We won't let that happen to you.

Our expert training and consulting services will give your systems engineers, developers and project leaders the skills, patterns and techniques necessary to ensure success on your MBSE project.

Hands On Training

We deliver engaging courses packed with challenging exercises and to-the-point lecture. We don't hire 'instructors'. Each class is led by a highly seasoned expert actively involved in their topic on a day to day basis. Since we teach what we do, we are always ready to tackle your team's most challenging questions.

Team members will come away from our courses with a new perspective, fresh enthusiasm and practical skills.

Onsite Problem Solving

Training is an important step, but the real learning happens on the job. We roll up our sleeves and work intensively with your team on a day to day basis to understand all the subtle aspects of your problem and deliver detailed solutions.

As the work progresses, team members will see what it takes to perform solid analysis and build worthwhile models. In time, they can share those skills with other team members. And the provided solutions yield useful examples that can be incorporated into internal training guides for your other engineers.

Management Assistance

Managers and project leaders don't need to be modelers. But the work products, review criteria and agile processes on an MBSE project are unique and need to be carefully understood.

It is also important for managers to know exactly why each model type is necessary and to understand the specific benefits of modeling as opposed to coding. They also need to spot any signs of counterproductive modeling.

We offer brief presentations and training geared toward project leaders, managers and executives so that they can get the results they want and provide the support the developers need.

High Performance Models

We help teams build executable models that solve hard problems on challenging platforms.

