

Thermo Fisher Scientific develops an appetite for modeling

About three years ago, Thermo Fisher Scientific in Eindhoven embarked on a journey to take model-based development to a higher level. In close collaboration, software and system architects started exploring multiple techniques. After pilots with research partners, they're now making successful steps with commercial tools from companies like Axini and Obeo.

Nieke Roos

"For our transmission electron microscopes, we have three families of product lines: entry-level, mid-range and high-end," explains Olivier Rainaut, system architecture R&D manager at the Eindhoven site of Thermo Fisher Scientific. "Across these product lines, with all the options and accessories, we have more than 1,000 active configurations. That's more than we have R&D engineers in Eindhoven."

These engineers, now almost 400 in number, aren't just doing R&D. "Our systems stay in the field for 20 years and longer. All this time, they need to have the latest versions of the control and application software, the correct network settings and the latest security patches," Rainaut points out. "Our job doesn't end once we've developed and deployed a new module; it's also our duty to keep everything alive and serviceable."

Key to keeping all the systems in the field up and running, according to Rainaut's colleague Arjen Klomp, is good interface management. "Over their lifespan, they're constantly being upgraded and updated with new hardware and software modules and applications," says the software development manager. "Not only do we need to ensure backward compatibility across many different versions, but we also need to prove that whenever we change an interface, everything keeps working correctly." To tackle this multidimensional interface management challenge, the R&D team in Eindhoven is investing heavily in model-based systems engineering (MBSE).

Software models

The MBSE push was initiated about three years ago. "We noticed that the growing complexity was making it increasingly hard for our engineers to find their way around the system and software design," Klomp recalls. "So, we decided to build an easily accessible knowledge base that people can consult when they have questions about the system. Models are very instrumental in creating such a single source of truth."

"While doing the software decomposition, we realized that we needed the in-

terfaces to be formally defined and kept under change control," continues Klomp. "At the same time, we saw many problems pop up during integration where the interface definition had remained the same but the interface behavior had become different. This made us realize that it's not enough to define the syntax of an interface; you also need to manage the semantics. Soon after, we started a project with TNO's ESI to address this."

Building on the Comma framework, the exercise with ESI, as Klomp calls it, set out to model both the syntax and



Credit: Thermo Fisher Scientific



Credit: Thermo Fisher Scientific

the semantics of subsystem interfaces. “We’ve learned that this really helps to get a better understanding of system behavior. The modeling stimulates early design discussions. It also facilitates model-based testing: instead of implementing first and then writing tests based on the implementation, you can already create tests from the interface definitions and then implement against those tests. Thus, the Comma exercise helped us uncover and fix several defects.”

This taste of modeling left Klomp and his software team with an appetite for more. “For deploying a tool on a larger scale within a company, you need two things: management support, to secure the necessary resources, and workplace buy-in, to get people to start using it. For the workplace buy-in, we looked at a couple of criteria, including tool maturity and available vendor support. After careful consideration of multiple solutions, we chose Axini for modeling and testing the subsystem interfaces. Their tool was the only one commercially available for model-based testing and the most mature, offering much more features and 7-8 more years of history than academic alternatives.”

System models

In 2020, the model-based push was taken to the system level, where Rainaut and his team did the same exercise to select the MBSE tool that best fit their purposes. From a host of options, including Sparx Enterprise Architect, IBM Rhapsody and Dassault 3DX, they picked the open-source, Eclipse-based Capella offering from Obeo. “We’re using Capella for system decomposition in a SysML-like way – hardware and software, up to interface mapping,” details Rainaut.

In advance of the tool selection, the system architects were put to work in a standard way in Visio. “They didn’t like that, as everything has to be done by hand and there’s no link between the different decomposition views,” says Rainaut, “but it forced them to structure their thoughts about the systems and cast the architecture descriptions in a digital form. Currently, we’ve covered more than 80 percent of the architecture in this way, so we’re pretty close to that single source of truth.”

The system decomposition efforts in Capella started last summer. Rainaut: “By first making them use Visio to capture everything, we’ve enforced a common way of working and we’ve decoupled the content description from the tool to be selected. In my opinion, it was a great stepping stone to MBSE. Through all kinds of pilot projects, we’re now doing more and more in Capella. The ultimate goal, of course, is to have everything in there and in a linked ecosystem.”

MBSE landscape

Using Capella, the transmission electron microscope is broken down into subsystems, ie combinations of hardware and software, interconnected by high-level interfaces. Think of a camera, a lens or a sample loader, for example. One such subsystem contains a number of software components, each having lower-level interfaces to neighboring components and realizing specific functionality. Some components process data by executing algorithms, others control the flow of execution.

“The high-level subsystem interfaces we model and increasingly also test with Axini,” elaborates Klomp. “By cre-

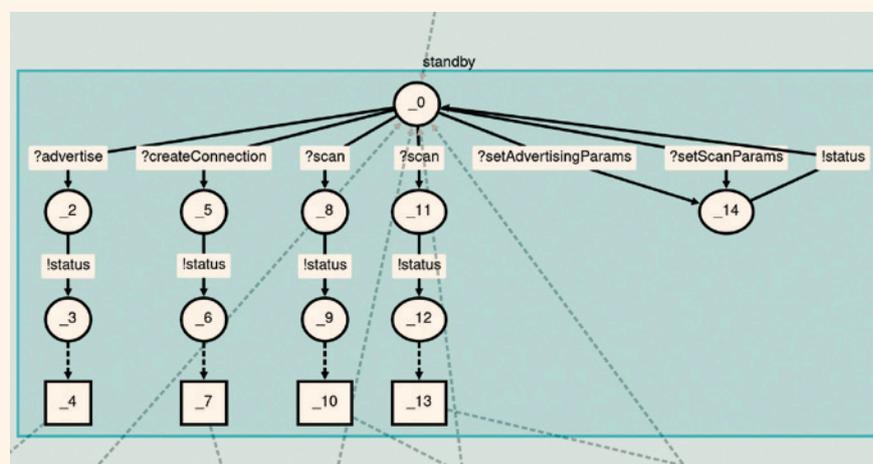
ating a so-called test clam around a subsystem, we can model both the provided interfaces and their relation with the required interfaces. This makes it possible to do error injection, allowing us to test good-weather behavior but also, by manipulating the required interfaces, induce bad and sad weather and verify whether the provided interfaces give the expected response.”

“With Verum’s Dezyne, we model the control components within the subsystems and check whether they’re doing what they’re supposed to do, ie whether they adhere to their interface specifications,” continues Klomp. “Note that we’re talking about discrete state control here. For continuous physical control, eg to move the sample loader, but also for other kinds of data processing, Mathworks-like modeling would be the way to go.”

While Axini and Verum are sometimes seen as competitors, Klomp actually views them as complementary. “Axini is more like a system-level interface verification tool,” he observes. “As we move down the MBSE landscape, adding more and more details through refinement, at a certain moment, we arrive at the software component level and we enter the realm of Dezyne.”

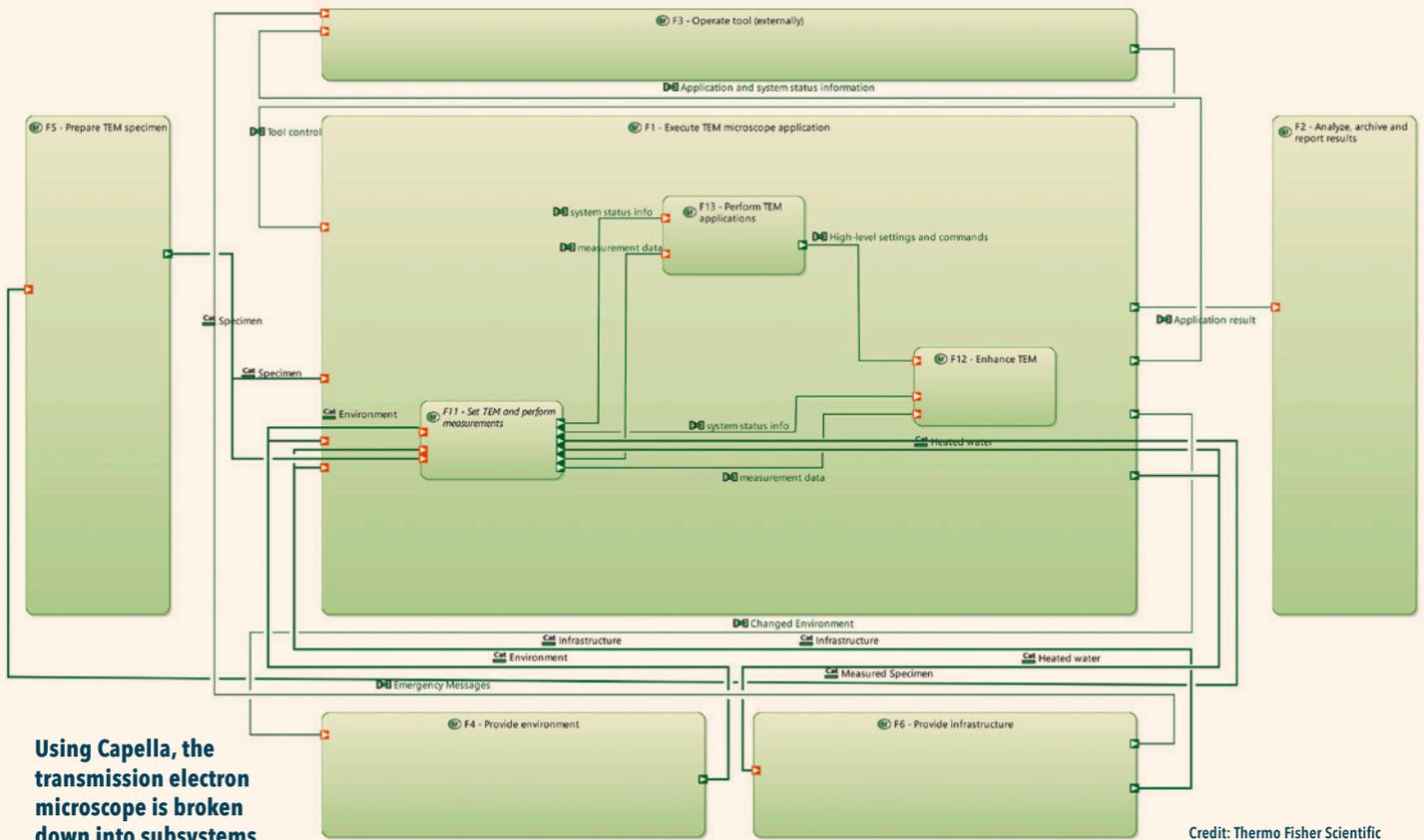
Paying off

Klomp is very happy to see the MBSE push bear fruit. “The modeling forces people to do the hard thinking upfront. It’s no longer possible to resort to Powerpoint architecting and then leave it to the engineers to solve the problems. The tools clearly show where a design doesn’t work and it’s up to the architects to find solutions. In the beginning, those unfamiliar with the approach were really



The high-level subsystem interfaces are modeled and increasingly also tested with Axini.

Credit: Axini



Using Capella, the transmission electron microscope is broken down into subsystems.

Credit: Thermo Fisher Scientific

reluctant to change their ways, but the mindset has shifted. Because they're actually finding and fixing errors, there's no denying anymore that it works."

Modeling will also be highly instrumental in tackling one of the big testing challenges in high tech today: the scarcity of hardware to test on. "That's becoming more and more of a problem for us," acknowledges Klomp. "Being big and costly, there aren't that many physical test systems to begin with and as the need for testing grows, securing machine time for testing is only getting more difficult. Also, having these 1,000+ active configurations, we can't possibly test them all physically. That's why we're running more and more of our tests in a virtual environment – using modeling tools like Axini, which allows us to test interfaces without all the components having to be available."

The decision to move to a single source of truth, ultimately in Capella, with a common way of working is paying off in other ways as well. Rainaut: "Previously, when everybody was free to choose their own tool for architecture description, it was impossible to build a complete overview and very time-consuming to retrieve system specifics. Now, with the models, safeguarded by change control boards, we have this common basis. It allows us to have better discussions about the architecture and find informa-

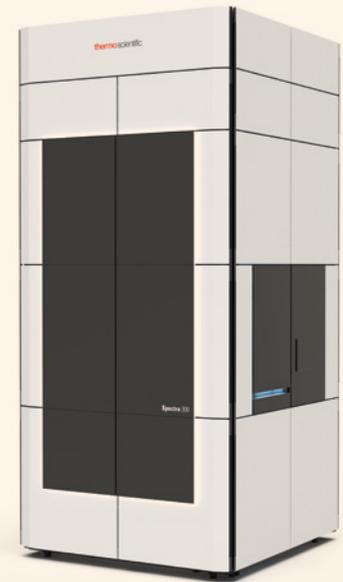
tion and perform decompositions much more quickly. It has also become much easier to bring new employees up to speed, which is all the more important as the R&D team in Eindhoven is growing rapidly – last year alone, we hired over a hundred people."

External collaboration

And there are more benefits on the horizon. "Performance prediction," gives Rainaut as an example. "Time to resolution, ie how long it takes the electron

microscope to achieve the desired picture detail, is an important metric for our customers. Historically, we'd do all kinds of calculations in Matlab or even Excel to get an estimate for that. Now, we could start doing tolerance budgeting based on architecture models made in Capella, for instance, or connect these architecture models to a performance analysis tool like ESI's POOSL and run simulations to check whether a design is meeting our target for time to resolution. This gives us a solid quantitative basis for design improvements. One of our focal points for 2022 is linking the architecture models to the simulation models."

Next to boosting internal development, model-based interface management can also streamline external collaboration. "When communicating and integrating technologies with partners and suppliers, it really helps to have a clearly defined list of interfaces," finds Rainaut. "Aligning both sides is the big challenge there. Another focal point for this year, therefore, is further integrating our system models with theirs." Klomp has a similar priority for his software team: "In 2022, supported by Axini, Verum and our other tool partners, I want to look into how we can extend our model-based technologies to our supply base."



Credit: Thermo Fisher Scientific

This article was written in close collaboration with Axini.