

The Essence of Software Engineering: Applying the SEMAT Kernel

Scott Ambler interviews Ivar Jacobson

[The Essence of Software Engineering: Applying the SEMAT Kernel](#) from Addison-Wesley Professional was released in January 2013. Written by Ivar Jacobson, Pan-Wei Ng, Paul E. McMahon, Ian Spence, and Svante Lidman this book describes the work to date of the Software Engineering Method and Theory (SEMAT) community. The goal of the SEMAT community is, as expressed in the SEMAT Call for Action, to “refound software engineering based on a solid theory, proven principles and best practices.” In this article I interview Ivar Jacobson about this fascinating work.

Scott Ambler: In 100 words or less, what is “the essence” of software engineering?

Ivar Jacobson: The software world has a huge number of methods. I believe there are more than 100,000 methods out there. Some have a brand such as Rational Unified Process (RUP), Scrum, Extreme Programming (XP), but most of them are homegrown with some ideas picked up from textbooks or magazines. Underneath all these methods we can find a common ground, or a kernel, of things we always have and things we always do when developing software. These things form the essence of software engineering. A kernel has been developed within the SEMAT community and is now likely becoming adopted as a standard by Object Management Group (OMG).

Scott: With all the other software engineering books out there, why did you feel this book was needed?

Ivar: This book is fundamentally different from any other book in software engineering. It describes this kernel, and it teaches software development based on this kernel in a method-independent way. This is very much needed in education – both at universities and in the industry. Instead of teaching different methods separately, you can now teach the kernel and then specialize this to teach specific methods. Once you have learnt the universal kernel, it becomes significantly easier to learn any specific approach, for instance Scrum, Lean, Kanban. They have a lot in common which is not easily visible today, but the kernel clearly reveals that.

Scott: You describe software engineering in terms of alphas. What are they and why are they important?

Ivar: The alpha concept is a powerful construct, the value of which has been proven over that last 7-8 years in many organizations - clients of my company. It incorporates several practical ideas, the most important one being that teams can measure the progress and health of their endeavors in a practice-independent way. At any moment in time the team can identify where it is now and where it is going next. It makes the team result focused instead of document driven or activity centric. The alpha concept is also fundamental to implementing the idea of a method being a composition of practices built on top of a universal kernel. It is the alpha concept that enables us to be able to create a robust, extensible, intuitive common ground.

Scott: Can you provide a few examples of alphas?

Ivar: We have identified seven top-level alphas: Requirements, Software System, Work, Team, Way of Working, Opportunity and Stakeholders. Take for example Software System, which has the following potential states: Architecture Selected, Demonstrable, Usable, Ready, Operational and Retired. To achieve a particular state there is a set of checkpoints to fulfill.

Apart from the top-level alphas, the kernel can be extended by adding lower level alphas--sub-alphas to the top-level alphas. These are typically added by the practices that the team adopts. For instance Requirements can have sub-alphas, which represent the individual requirement items--for instance a user story, a use case slice, or a feature.

Scott: How do the ideas you capture in this book relate to agile? To lean? To RUP?

Ivar: The beauty of the approach is that it is agnostic to any particular method. You can apply the ideas of the book with agile, lean or any other legacy or new method. Since agile and lean today are the dominant ways of working, I will give two examples of how we make these better: 1) we can measure progress and health systematically at any point in time, and 2) it is easier to select and adopt the most appropriate set of agile practices to be used. We know that we will find better practices as we learn more about software development. The SEMAT approach allows the team to make major changes to their way of working in an evolutionary way instead of doing what they have done in the past: throw out the old way of working and start all over with a new way of working.

Scott: Many organizations are now trying to apply agile at scale. What insights does this book provide for doing so?

Ivar: Agile at scale is what my company has been doing for many years, and with that background many of the ideas behind SEMAT have been developed. The kernel idea became evident when we tried to help large companies to govern their methodologies, usually many but with no common ground. Large companies will use the kernel as a base and allow teams to select practices from a practice library to create their diverse methods. It helps the organization to become a learning one, since practices (much smaller than methods) can be adopted and improved across the organization. Practices will be improved, taught and mix-and-matched across the organization. Practices are by definition designed to be lean so the methods applied in the organization will naturally become lean.

Scott: In the book you indicate that there is more work to do on this topic. So, how has this work evolved since then and where will do you hope it will go in 2013 and 2014?

Ivar: Thus far SEMAT has focused on getting a widely agreed upon kernel. The next step of SEMAT is to create a practice library described on top of the kernel. There are today a set of 25 or more such practices defined on top of an earlier version of the kernel, so we are not starting from scratch. Tools are being developed to support practice authoring and practice exchange. Courses are being written to

teach the kernel and practices. And more of similar nature. However, we are also working on a different area, an underlying theory for software engineering, which has got a lot of attention from the academic world.

Scott: Who is working on the SEMAT effort?

Ivar: SEMAT was founded by Bertrand Meyer, Richard Soley and me. Today we serve as an advisory board. SEMAT has an executive committee of 7-8 people who lead the work in the community. More than 20 people have been working together for a long time to get where we now are: a widely agreed upon kernel. SEMAT has also local chapters in several countries, in China, South Africa, Latin America, Russia and under formation in Japan and South Korea.

Getting to this kernel was the very first milestone of SEMAT. Now the work scales up, and many more people are joining the effort. We have just appointed a new chairman of the executive committee-- professor June Park--with experience from both the academic world as a professor at the University of Iowa in Iowa City and from the industry as CTO of Samsung SDS.

We will be working in primarily two areas: 1) the practice area and 2) the theory area.

The practice area will work on creating a practice library of well-proven practices such as user stories, use-case 2.0, Scrum, architecture design, etc., which can be mixed and matched to create specific methods.

The theory area has been very active this last year and successful in creating a significant interest in the academic world. The objective is to come up with a general theory in software engineering. The SEMAT kernel is interesting as a base for such a theory – it works as a definition of what software engineering is.

Scott: Do you have any parting thoughts to share with us?

Ivar: SEMAT represents a paradigm shift in the way we look upon ways of working. The kernel is a simple thinking framework to help teams understand the progress and health of their endeavor. As with every paradigm shift there is a threshold to step over before being able to see the fundamental differences with what you had before. Once over, you get a very good basis for discussing and comparing practices, you can easier upgrade your existing way of working in small controllable steps, and you can systematically learn from other teams inside or outside your company.

Dr. Ivar Jacobson is a father of components and component architecture, use cases, the Unified Modeling Language and the Rational Unified Process. He has contributed to modern business modeling and aspect-oriented software development. Lately, Ivar has been working on how to deal with methods and tools in an agile and lean way. He is one of the leaders of SEMAT discussed in this interview. In 2004, Ivar received the Gustaf Dalen medal from Chalmers Institute Of Technology, Gothenburg Sweden. He is an international honorary advisor at Peking University, Beijing, and he is an honorary doctor at San Martin de Porres University, Peru. He is the principal author of six influential and best-selling books, including Object-Oriented Software Engineering, Business Process Reengineering with Objects, Software Reuse: Architecture, Process and Organization for Business Success, The Road to the Unified Software Development Process, and The Unified Software Development Process. He co-authored two UML books with Grady Booch and James Rumbaugh. Ivar is also a founder of Ivar Jacobson International, operating in seven countries around the world. His company's home page is ivarjacobson.com.

Scott W. Ambler is a Senior Consulting Partner of Scott Ambler + Associates, working with organizations around the world to help them to improve their software processes. He provides training, coaching, and mentoring in disciplined agile and lean strategies at both the project and organizational level. Scott is the founder of the Agile Modeling (AM), Agile Data (AD), Disciplined Agile Delivery (DAD), and Enterprise Unified Process (EUP) methodologies. He is the (co-)author of several books, including Disciplined Agile Delivery, Refactoring Databases, Agile Modeling, Agile Database Techniques, The Object Primer 3rd Edition, and The Enterprise Unified Process. Scott is a senior contributing editor with Dr. Dobb's Journal and his company's home page is ScottAmbler.com.



About Ivar Jacobson International

IJI is a global services company providing high quality consulting, coaching and training solutions for customers implementing enterprise-scale agile software development.

IJI improves the performance of software development teams by introducing new practices, and removing barriers to their wider adoption.

Through the provision of high calibre people, innovative practices, and proven solutions, we ensure that our customers achieve strong business/IT alignment, high performing teams, and projects that deliver.

www.ivarjacobson.com

Sweden

+46 8 515 10 174

info-se@ivarjacobson.com

Netherlands

+31(0) 20 654 1878

info-nl@ivarjacobson.com

United Kingdom

+44 (0)207 953 9784

info-uk@ivarjacobson.com

Asia

+8610 82486030

info-asia@ivarjacobson.com

Americas

+1 703 338 5421

info-usa@ivarjacobson.com