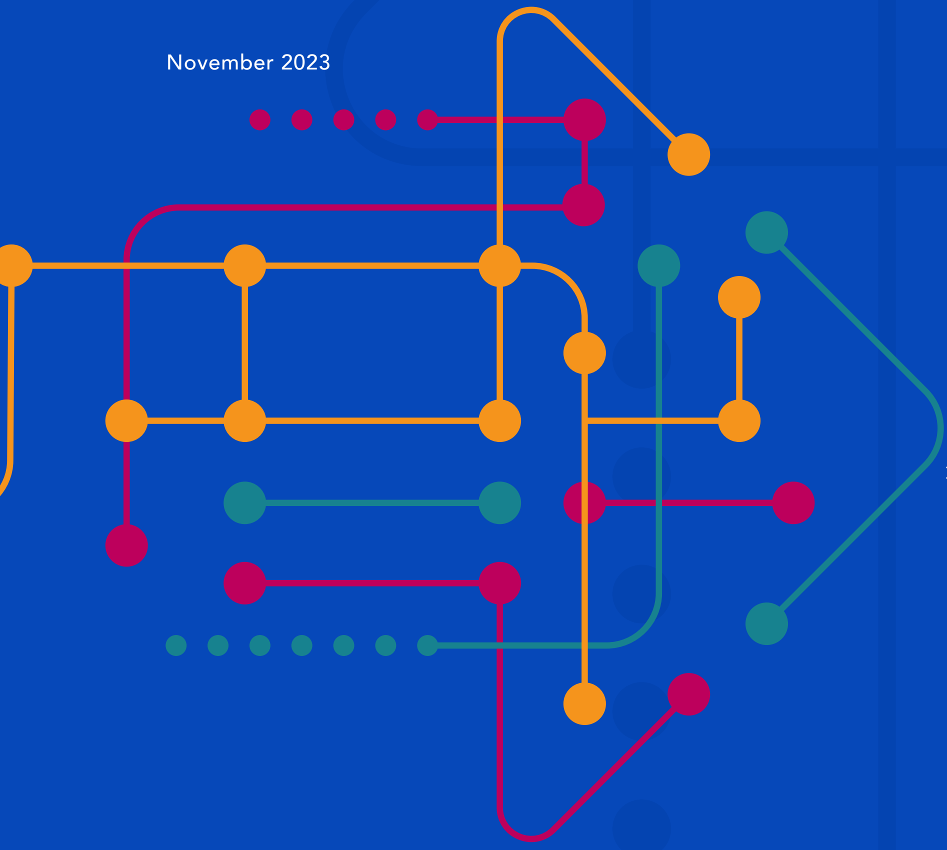
Navigating the future of ITEA

Current challenges and future directions

Throughout this year, including in the ITEA Magazine, we celebrated 25 years of ITEA by showcasing the successes achieved over the past decades and highlighting the milestones reached and the impact made. Now, as we conclude this anniversary year, it's paramount that we shift our focus towards the future of ITEA. The software innovation process is an ongoing journey in which past innovations serve as stepping stones for future breakthroughs, reinforcing the idea that software itself is a powerful tool for developing new software solutions that solve the important emerging challenges that lie ahead.



What are the new challenges?

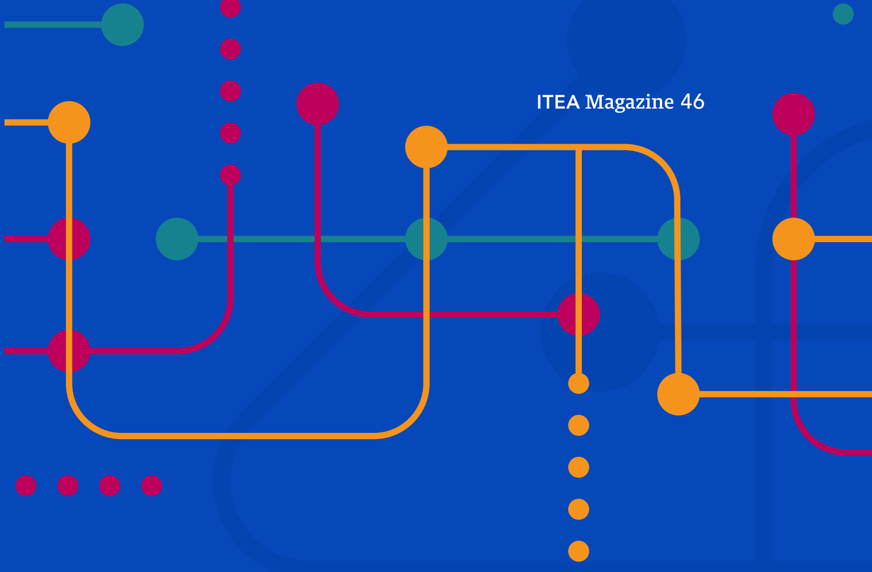
One of today's challenges comes from the abundance of potential solutions that have been developed up to now during this software innovation journey. When a digital transformation project starts, many options are available with different approaches based on standard methods or data models or a combination of both. It is often hard to make the right choice and many examples show that a wrong decision can lead to the failure of such a project. So far, the effort has been focused on developing digital technologies and not enough on how to select the right digital technology for a given problem. In addition to all of the IT methods and tools that have been or are currently being worked out, knowledge of how to use them must be developed. This knowledge is mandatory when it comes to limiting the failure of digital transformation projects. To build this knowledge, cooperation and the sharing of experience are a must.

A second challenge is related to the integration of new objectives within the design process of software-based systems. There are at least two dimensions that are of the upmost importance and are not yet completely well-managed: security and sustainability.

- **Security** is clearly a must as more and more critical infrastructures are software-based. The consequences of cyber security attacks are becoming more and more serious for society and the economy. At the same time, the surface of attacks has increased as these critical infrastructures are more and more connected. Both the severity of the attacks and the higher risk of attacks must be addressed. The challenge is to integrate a security-by-design approach, to develop new protection features and to offer degraded operation modes to maintain at least a minimal service level for these critical systems.
- **Sustainability** is a second objective that is now a priority when designing a system. The problem scope is large as the design process, the manufacturing process, the operation phase and potentially the recycling phase must be considered. For each of these steps, it is important to minimise the resources (human resources, energy, raw materials, etc.) needed by the system in order to reduce the environmental footprint and create a sustainable system. During the design, for example, it is important to reuse previous sub-systems to save effort or to design in a way that will allow the new system to be reused. The manufacturing process also needs to minimise the raw materials and the energy needed. For software systems, energy consumption needs to be optimised when running; this is a complex task, particularly in cloud-based infrastructures. Finally, the maintenance process needs to balance the lifespan of the system and the resources needed to repair it, while the raw material used by the system must potentially be recycled at the end of life of the system. Collectively, these objectives make the design of sustainable systems a huge challenge.

These two new objectives in the design of complex systems come in parallel with the functional objectives. The problems are therefore mainly multi-criteria optimisations in a complex and sometimes partially unknown environment. The objectives can be conflicting as security, for example, can need more resources and thus negatively impact the sustainability of the system. This shows how complex this challenge is and that cooperation and experience sharing are again mandatory to progression.

A third challenge to continue to progress in the software innovation journey is the creation of ecosystems that will be able to seize some of the new opportunities offered by digital technologies. So far, only the low-hanging fruits have been picked, mostly the ones for which a single organisation can manage the development of the software-based system. There are additional opportunities offered by digital technologies, but they will require several organisations to come together and jointly work on digital technology



solutions that will unlock new services. One example is the 'smart road' concept in which car suppliers, local authorities in charge of roads and service providers should cooperate to develop safer, smoother and more environmentally friendly usage of the roads. There are many other domains for which only cooperation between several organisations can unlock all of the opportunities offered by digital technologies. We have already experimented with cooperation that has been mutually beneficial, such as standardisation efforts. These are mainly 'horizontal' efforts, and the challenge is to go a step further and to find a win-win structure that will enable cross-organisation research and development in order to propose future digital services.

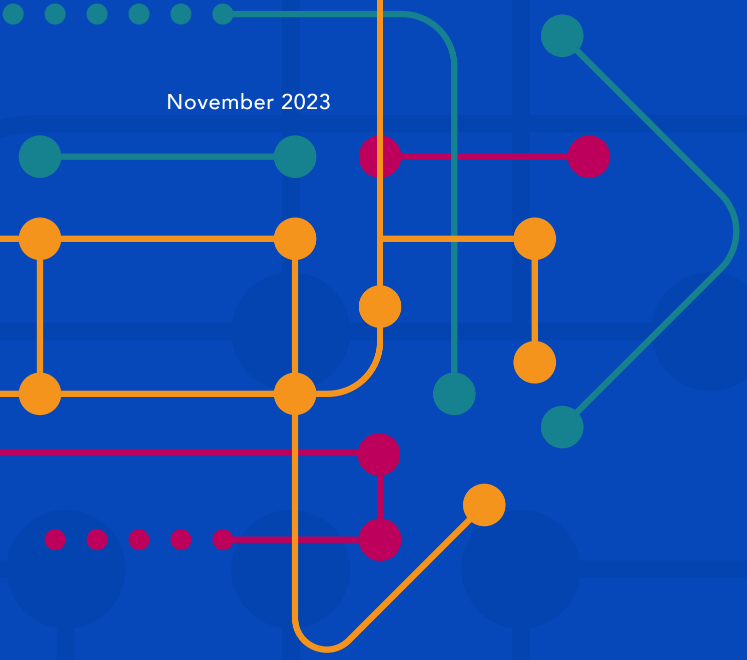
There are other challenges that could be mentioned, of course, but the three that have been presented here represent the synthesis of the concerns that ITEA has collected through its customer-oriented organisation and approach. ITEA is an association with more than 15 Board companies representing various industrial sectors that contribute to the shaping of a vision of the software innovation journey. It has also established Advisory Boards in the domains of smart cities and cyber security with members that help to identify the current needs and the real-life challenges. Regular customer workshops are organised to collect the end-users' pains and to brainstorm on how to solve concrete problems. The challenges presented in this section are the high-level expression of the feedback given by all these sources.

What are the future directions?

To continue this software innovation journey and to engage with more challenging but highly valuable software applications, some research directions can be highlighted.

- First, in the domain of data exploitation, there are opportunities in aggregating more sources to reach a new level of insights that can be built from these data. The recent success of the ChatGPT and DALL.E 2 OpenAI generative tools for text and image generation shows the power of aggregating a lot of data, organising them, training huge neural networks and developing easy-to-use interfaces. These tools would not be so powerful if they were based on a limited amount of data. The aggregation of different data sources can really make a qualitative difference in many domains. The health sector is a good example. The aggregation of patient data – of medical analysis data and treatment data – would empower the generation with new knowledge. This example also shows that the problem is not only technical but also organisational (how to federate all of the data owners) and ethical (how to protect privacy, how to control the use of such a powerful tool). There are many domains for which the aggregation of data would deliver valuable insights that would be unreachable otherwise. The question of whether the initiative to aggregate the data should be public or private is open and may have different answers depending on the domain. In any case, the aggregation to the next level of data sources can really pay off and is a good direction for research that will have to deal with technical, organisational and ethical problems.

- A second important research direction is related to the next generation of software applications that we want to develop, run and control. Most of the time, the applications are highly distributed: starting at the sensor and actuator level with modules at the edge and other components in the cloud or even in large supercomputing facilities. At each level, the IT resources are quite diverse (from very tiny microcontrollers to highly specialised hardware like graphical processing units or neural processing units). The IT infrastructure is dynamic, with nodes that can be switched off or integrated in applications. It would also be good for a software application to optimise its energy consumption and to detect potential cyber security attacks. In other words, the software application must be aware of its behaviour and be able to negotiate with the IT infrastructure and certain quality service agreements. There is a need for new technologies that will enable the dialogue and some contract agreements between the applications and the underlining infrastructures. In addition, new solutions are mandatory, not only to code the application but to develop the layer that will control the application and will give the software the ability to adapt to a dynamic environment and optimise its operation. Some progress in this direction will enable the development of highly distributed software applications that will be robust, dependable and energy efficient. This will be a way to continue the software innovation journey with a new class of



applications that are much more powerful than what is currently achievable.

- The third direction aims to address the design and the operation of complex systems. As explained earlier, we want to be able to tackle more ambitious functional specifications in the design phase and to add new objectives to them, such as dependability, security and sustainability. In most cases, we want to simultaneously co-develop the system itself and an environment that will optimise its operation (digital twins, monitoring tools, maintenance tools, etc.). All of these objectives come at a time when the workforce that is able to develop software is limited. Most of the Eureka countries have not chosen to target programming in standard educational courses. Even if the digital native generations are experts in using digital technologies, most people will not be able to actively develop them. They will know how to ‘read’ but will need some help to ‘write’. Due to this limited workforce and the challenging targets, progress in the design tools and in the methodologies to create these complex systems is mandatory. To facilitate the involvement of non-software engineers, a priority is the development of low-code or no-code approaches. This will allow more people to actively take part in the digital transformation. The new technologies that should be developed will aim to increase the productivity in software-based system design and to decrease the failure rate of digital transformation projects.

Again, more directions for the continuation of the software innovation journey could be presented. Nevertheless, the three which have been highlighted here show that this journey is not yet at its end. As ITEA sets sail into its next chapter, these efforts and directions will serve as the compass guiding our voyage towards a brighter and more innovative software-driven future.