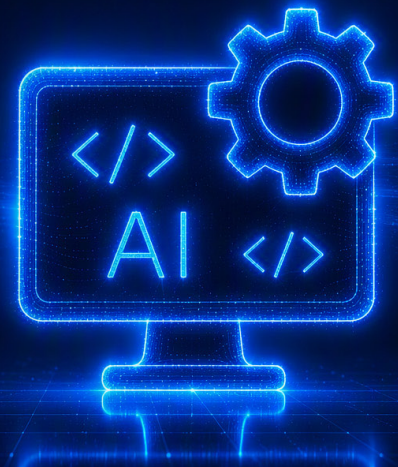


AI engineering as a competitive advantage

Interview with Alexander Casall



AI Engineering is not only changing software development. Above all, it is changing the economics of innovation.

Alexander Casall

Chief Commercial Officer at ZEISS Digital Innovation, explains why AI Engineering is far more than Copilot support for developers and why companies need to rethink their software strategy to remain competitive.

Mr. Casall, everyone is talking about AI. Why should CEOs be particularly interested in AI Engineering?

Alexander Casall: Because AI Engineering directly influences a company's speed of innovation. Many decision-makers still see AI primarily as an efficiency topic. That view is too narrow. The real leverage lies in evaluating more solution options earlier and identifying the best business fit faster. The question is no longer simply, "Can we develop this solution?" It is, "Which solution creates the greatest business value?"

Does AI Engineering also change the traditional make-or-buy decision for software?

Yes. Standard software was long considered almost automatically more economical than a custom solution. However, if AI Engineering lowers the effort required for development and, in particular, further development, this calculation needs to be reassessed. Custom software can become economically viable where companies need to differentiate, expect frequent changes, or want greater control over critical business processes.

Make-or-buy remains a strategic decision, but cost, flexibility, time to market, technological sovereignty, and long-term adaptability take on new weight. In volatile markets in particular, this sovereignty becomes an advantage: companies can make decisions more independently, adapt faster, and evolve their software in closer alignment with their own business model.

Does this also make legacy software modernization more economically attractive?

Yes. Many modernization initiatives have been postponed because analysis, refactoring, testing, and migration appeared too costly or risky. AI Engineering can reduce this effort and make gradual modernization more economical. The business impact lies in reducing technical debt faster, lowering

operational risks, and enabling existing systems to support new products, processes, and business models

To put it provocatively: Will AI soon make offshore software development obsolete?

"Obsolete" would be overstated. But the traditional cost argument is becoming weaker. If AI Engineering can potentially reduce project costs by 35 to 45 percent, depending on the context, the entire business case changes. Teams closer to the customer then gain importance through domain knowledge, cultural proximity, shorter decision-making paths, and faster iterations.

Are we therefore seeing a renaissance of onshoring?

Yes, at least for certain tasks and industries. AI can offset part of the cost advantage of traditional offshore models. At the same time, business knowledge, domain expertise, market understanding, and customer proximity become more valuable. When business owners and engineering teams collaborate more closely, assumptions can be tested faster and solutions can be developed around actual needs.

Many companies report productivity gains, yet the business impact is often missing. Why?

Because speed alone does not create value. Accelerating an unsuitable process often only shifts the bottleneck. The key question is therefore: What would this business process look like if we designed it from the ground up as AI-native today?

Where does the greatest strategic value lie?

In creating additional options for action. Many ideas previously failed not because they were technically infeasible, but because of effort, time, or risk. When the effort required for implementation and validation decreases, companies can test more variants, learn earlier, and pursue applications that were previously not economically viable. The real revolution is not that software is created faster. It is

that we can now economically realize solution options that would never previously have had a viable business case.

Can you illustrate this with a specific project?

Yes. In one customer project, human-led AI Engineering enabled us to reduce development effort by 51 percent compared with a traditional team setup. After just five weeks, the weekly value contribution was three times higher. For the customer, this meant more feasible options, lower investment risk, and the opportunity to implement competitively relevant software ideas economically that would not previously have been prioritized.

In another highly regulated medical technology project, we accelerated UI and code implementation by a factor of three, and testing and documentation by a factor of ten. The business impact was lower development costs, shorter time to market, and faster innovation cycles, even in regulated and business-critical environments.

Critics say AI produces code, but not quality.

That is true when AI is used as an uncontrolled code generator. This is why we speak of AI Engineering, not merely AI Coding. Particularly in regulated or business-critical environments, requirements, architecture, security, verification, and approval remain under human responsibility. AI accelerates analysis and implementation, but people remain accountable for intent, risk, and release.

Many companies are concerned about IP protection and data security. Are those concerns justified?

Absolutely. Governed AI Engineering requires controlled toolchains, clearly separated project contexts, defined access rights, approved models, and technical safeguards. Customer data and intellectual property must not flow uncontrolled into training environments

or third-party contexts. In business-critical environments, this is not an optional feature but a prerequisite.

Will AI Engineering replace developers?

It primarily changes the focus of their work. There will be less manual routine and more framing, specification, architecture, context, verification, and review. AI shifts effort rather than eliminating engineering responsibility. Teams that combine technology, domain knowledge, and decision-making capability become particularly valuable. AI Engineering is not a productivity tool for IT. It is a new operating model for innovation. The strategic opportunity lies not only in faster coding, but in shorter learning cycles, more economically testable solution options, and controlled implementation. Particularly in volatile markets, adaptability becomes a measurable competitive advantage.

What should CEOs do now?

CEOs should reassess the economics of their solution portfolio: Which previously unprofitable ideas become viable through AI Engineering? Where do make-or-buy decisions shift in favor of custom software? And where do the limitations of standard software or legacy systems justify a new solution? The next step is to derive prioritized business cases with measurable value contributions, clearly defined risks, and a robust scaling perspective.

This brings the argument full circle: AI Engineering changes not only how software is created, but also which innovations are economically possible in the first place. Software Engineering thus becomes a strategic lever for growth, differentiation, and competitiveness.

AI Software Engineering raises new strategic questions. We would be pleased to share our experience, decision models, and insights into specific courses of action.

For more information, visit:
zeiss.com/digital-innovation

