Common misconceptions about Industry 4.0 that manufacturers still believe

by Christian Lutz

Misunderstandings about the IIoT have made the prospect of adoption unnecessarily intimidating.

In many cases, industrial organizations avoid the most effective industry 4.0 implementations merely due to misconceptions, taking the unnecessarily long road towards digitalization. While IIoT adoption does require a new approach to managing and analyzing data collected in real-time, this isn't as difficult as many manufacturers believe.

To illustrate this point, let's take a look at four commonly believed "facts" about industrial IoT applications that are, in fact, misconceptions.

"The databases traditionally used in industrial environments, as SQL Server or Oracle, are a good option for industrial IoT projects"

This is false. The traditional databases that most industrial organizations already have in place (e.g., Microsoft SQL Server, Oracle, and so on) are inappropriate for IIoT systems, due to the tremendous data volume and complexity that these systems imply. Traditional databases are precisely that, traditional—they suit a classic manufacturing environment, not a digitalized one. There's a night-and-day difference between standalone machines and those networked within IIoT monitoring systems.

When industrial businesses implement IIoT infrastructures using traditional databases, which happens often, the problems start early on. They find that these databases are expensive to scale, unable to process the vast amount of incoming data, and incapable of handling the complex queries they need to perform. Traditional databases are simply not adequate to query huge data streams in realtime.

"NoSQL databases are the solution"

Given the fact that IIoT sensors collect massive volumes of unstructured (JSON) data, it's understandable that industrial organizations believe that to use a NoSQL database will solve all their issues.

However, using NoSQL databases for IIoT usecases will cause new problems. NoSQL databases do indeed offer distributed architectures that lend themselves to performing complex, flexible queries—but NoSQL databases come with high costs, complex infrastructures (which take intensive planning and administration to operate correctly) and engineers with specialized expertise.

Besides, nearly every IIoT system must manage both JSON and relational data, consisting of topological, firmware, ERP, or article data. If an industrial business chooses to address this need by running two databases, a relational and a non-relational database, both systems will need to be synchronized for use in parallel. This synchronization results in complex setups, unnecessarily large cloud footprints, and disadvantages when running queries.

The logical alternative comes through this more in-depth understanding of the requirements for the IIoT-database. Advanced systems are now available, offering both the ease of SQL and the flexible and scalable nature of NoSQL solutions. This combination of features is the perfect fit for IIoT implementations.

"Time-series databases are the solution"

This is also common mistake. Instead of using a NoSQL database as, for example, MongoDB, some manufacturers decide to use a time-series specialized database, as InfluxDB, for their IIoT projects.

This is a strategic error. A time-series database should not be foundational to an IIoT system, because intense parallel usage will severely limit its functionality and scalability. IIoT systems don't just need to visualize data streams: they also must perform analysis, run highly concurrent workloads, and frequently change the data model. The IIoT database must enable interactive work under heavy, real-time data loads, performing reads, writes, and executing ad-hoc queries all at once.

This processing under load is what makes it feasible to quickly and accurately identify production issues at smart factories, as well as to leverage advanced techniques such as machine learning. These capabilities are integral to IIoT systems, being very difficult to achieve them with a database designed for single-node operations and more simple datasets.

The IIoT database must also be able to adapt and extend data schemas at runtime to support agile processes. It must be possible to investigate anomalies in production using the bare sensors, ERP, quality, or other data to recognize issues. If specific jobs, materials, or suppliers are the sources of a problem, adapting the data model to examine these data types will provide the correct insights. Unfortunately for businesses using timeseries databases for their IIoT systems, making such changes to the data model would require performing a total rebuild of their databases, at a great expense of time and money.

"Artificial intelligence is out of reach due to insufficient/unclean data"

It remains common for organizations to believe that, because they don't yet have large amounts of sensor data or because their data is not perfectly clean, they aren't capable of leveraging AI systems. While a shortage of data can result in low-quality automation when driven solely by AI, it's inaccurate to assume that AI is an allornothing proposition from a data perspective.

Even with limited data, it's possible (and beneficial) to build a real-time data store, using AI and machine learning to augment and optimize IIoT decision-making while keeping the final decisions in human hands.

Developing these AI systems is a bit of a chickenand-egg situation: you have to start somewhere. Businesses that attempt to clean all their data before introducing it to their AI system will find themselves running into difficulties. Instead, a slow-and-steady incremental improvement is the correct approach. Few industrial companies have vast stores of clean data available—but by introducing systems to clean the data automatically, the AI capabilities will improve, and the opportunities for automation will increase as more data is collected.

Conclusion

Those reluctant to explore industry 4.0 implementations should become aware of the profound operational insights and benefits IIoT systems can deliver.

At the same time, organizations need to be aware of what capabilities are mandatory for success. The IIoT requires entirely new data management and analysis capabilities; to monitor, predict, and control equipment across massive pipelines call for specialized datamanagement systems.

The future value of IIoT projects also depends on achieving data-driven automation. Datamanagement systems must offer the possibility of rapid development, maintaining consistent uptime. Besides, the IT-operating costs need to be kept low when it comes to hosting, integration, and administration.