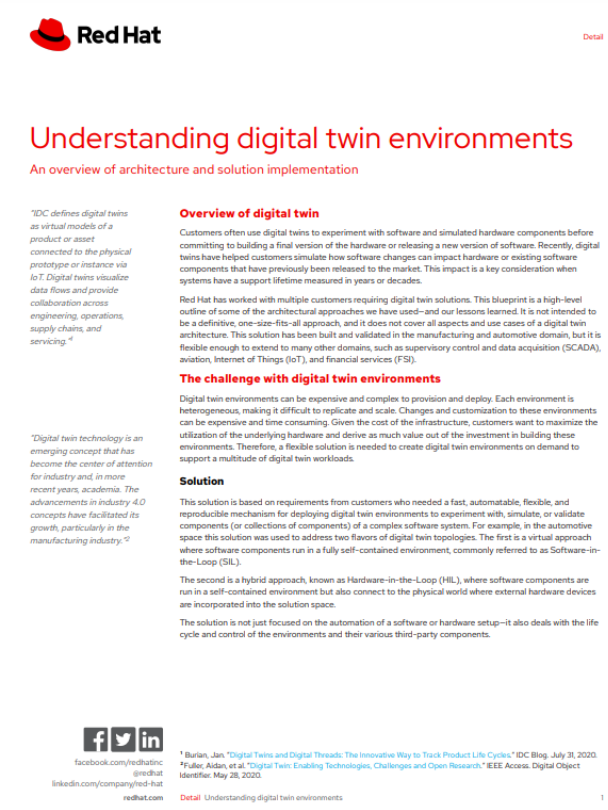
Understanding digital twin environments



Red Hat

Detail

Understanding digital twin environments

An overview of architecture and solution implementation

"IDC defines digital twins as virtual models of a product or asset connected to the physical prototype or instance via IoT. Digital twins visualize data flows and provide collaboration across engineering, operations, supply chains, and servicing."[1]

"Digital twin technology is an emerging concept that has become the center of attention for industry and, in more recent years, academia. The advancements in industry 4.0 concepts have facilitated its growth, particularly in the manufacturing industry."[2]

Overview of digital twin

Customers often use digital twins to experiment with software and simulated hardware components before committing to building a final version of the hardware or releasing a new version of software. Recently, digital twins have helped customers simulate how software changes can impact hardware or existing software components that have previously been released to the market. This impact is a key consideration when systems have a support lifetime measured in years or decades.

Red Hat has worked with multiple customers requiring digital twin solutions. This blueprint is a high-level outline of some of the architectural approaches we have used—and our lessons learned. It is not intended to be a definitive, one-size-fits-all approach, and it does not cover all aspects and use cases of a digital twin architecture. This solution has been built and validated in the manufacturing and automotive domain, but it is flexible enough to extend to many other domains, such as supervisory control and data acquisition (SCADA), aviation, Internet of Things (IoT), and financial services (FSI).

The challenge with digital twin environments

Digital twin environments can be expensive and complex to provision and deploy. Each environment is heterogeneous, making it difficult to replicate and scale. Changes and customization to these environments can be expensive and time consuming. Given the cost of the infrastructure, customers want to maximize the utilization of the underlying hardware and derive as much value out of the investment in building these environments. Therefore, a flexible solution is needed to create digital twin environments on demand to support a multitude of digital twin workloads.

Solution

This solution is based on requirements from customers who needed a fast, automatable, flexible, and reproducible mechanism for deploying digital twin environments to experiment with, simulate, or validate components (or collections of components) of a complex software system. For example, in the automotive space this solution was used to address two flavors of digital twin topologies. The first is a virtual approach where software components run in a fully self-contained environment, commonly referred to as Software-in-the-Loop (SiL).

The second is a hybrid approach, known as Hardware-in-the-Loop (HiL), where software components are run in a self-contained environment but also connect to the physical world where external hardware devices are incorporated into the solution space.

The solution is not just focused on the automation of a software or hardware setup—it also deals with the life cycle and control of the environments and their various third-party components.

facebook.com/redhatinc
@redhat
linkedin.com/company/red-hat
redhat.com

[1] Burlan, Jan. "Digital Twins and Digital Threads: The Innovative Way to Track Product Life Cycles." IDC Blog. July 31, 2020.
[2] Fuller, Aidan, et al. "Digital Twin: Enabling Technologies, Challenges and Open Research." IEEE Access. Digital Object Identifier. May 28, 2020.

Learn about digital twin environments and Red Hat's architecture and solution approach—and lessons learned from customer implementations.