



Understanding digital twin environments



Understanding digital twin environments

An overview of architecture and solution implementation

10C defines digital twin as virtual models of a product or asset, accessible to the physical counterpart or related via all digital twin available data flows and provide capabilities across engineering, operations, supply chains, and servicing.

Overview of digital twin

Customers often use digital twin 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 creating digital twin solutions. This is applied in 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 operations control and data ingestion (OCDA), avionics, Internet of Things (IoT), and financial services (FS).

The challenges with digital twin environments

Digital twin environments can be extensive and complex to provision and deploy. Each environment is heterogeneous, making it difficult to migrate and use. Changes and customization to these environments can be expensive and time-consuming. Given the cost of infrastructure, customers need to maximize the utilization of the underlying hardware and software as much as possible and all 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 world, this solution was used to simulate new features of digital car components. The architecture is a virtual approach where software components run in a fully self-contained environment, commonly referred to as Software on the Edge (SOE).

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

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



*Based on "Digital Twin and Digital Thread: The Roadmap to Full Product Lifecycle (PLC) Mfg. July 6, 2020" by Mike Adams at Digital Twin Factory Technology (https://www.digitalfactory.com/2020/07/06/digital-twin-roadmap/).
Revised: May 28, 2021

© Red Hat, 2021. Understanding digital twin environments.

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