Digital Twins for IoT Applications

A Comprehensive Approach to Implementing IoT Digital Twins

ORACLE WHITE PAPER | JANUARY 2017

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Introduction

With the growing deployments of Internet of Things (IoT) systems, the importance of the concept of a digital avatar of a physical thing has gathered significant interest in the recent years.

The most prominent example of this trend can be found in the Gartner's report titled "Top 10 Strategic Trends for 2017" published in October 2016, Digital Twins was Number 5 strategic trend for 2017 in this report. These digital proxies are expected to be built from the domain knowledge of subject matter experts as well as the real time data collected from the devices.

It is not a surprise then that most vendors of IoT Platforms have implemented some form of a digital twin. These are typically named as twins, shadows, device virtualization, etc.

In this whitepaper, we will discuss the origins of the digital twins, talk about various implementations in the industry. We will then discuss the digital twin implementation in Oracle IoT Cloud.

Origins

The term "Digital Twin" was defined by Dr. Michael Grieves at the University of Michigan around 2001-2002. He originally defined this in the context of Product Lifecycle Management. In his paper, he introduced the concept of a "Digital Twin" as a virtual representation of what has been manufactured. He promoted the idea of comparing a Digital Twin to its engineering design to better understand what was produced versus what was designed, tightening the loop between design and execution.

Dr. Grieves defined the following terms to define the digital twin:

1. **Digital Twin Prototype (DTP):** A DTP describes information to create an asset. As an example, a DTP could contain the 3D model of the asset, as well as the Bill of Materials and Bill of Processes describing how to manufacture that asset.

   Digital Twin Prototype does not refer to a specific instance of the asset. It's rather a recipe for manufacturing an asset.

2. **Digital Twin Instance (DTI):** A DTI is about a single specific physical instance of an asset. It could contain the list of exact part numbers that went into production of this specific asset, and the exact process steps that were followed in producing the given asset. The Digital Twin Instance also contains the current operational states captured from the sensors connected to the assets.

   Multiple separate physical assets could be manufactured using a single Digital Twin Prototype, and each of them will have their own Digital Twin Instances.

3. **Digital Twin Aggregate (DTA):** A DTA is simply an aggregation of multiple DTIs, and it allows for querying information about a group of assets.

Value of digital twin

The concept of the digital twin is a very powerful one. Let us briefly look at the typical benefits that this concept provides:

1. **Visibility:** The digital twin allows visibility in the operations of the machines as well as in the larger interconnected systems such as a manufacturing plant or an airport

2. **Predictive:** Using various modeling techniques (physics-based and mathematics-based), the digital twin model can be used to predict the future state of the machines

3. **What if analysis:** Through properly designed interfaces, it is easy to interact with the model and ask the what-if questions to the model to simulate various conditions that are impractical to create in real life

4. **Documentation and communication mechanism to understand and explain behaviors:** Digital twin model can be used as a communication and documentation mechanism that can be used to understand as well as explain the behaviors of an individual machine or a collection thereof

5. **Connect disparate systems such as backend business applications:** If designed correctly, the digital twin model can be used to connect with the backend business applications to achieve business outcomes in the context of supply chain operations including manufacturing, procurement, warehousing, transportation and logistics, field service, etc.

Industry implementations

Looking at the benefits listed above, it should be no surprise that most IoT vendors have taken a keen interest in this concept. Almost every IoT Platform has implemented some sort of digital twin capabilities – though there are really clear differences in terms of their maturity and vision. At a broad level, these implementations typically fall into two categories:

1. **Simplistic device models**

   These implementations typically use a JSON document that contains two primary sets of attributes:

   a. **A set of Observed or Reported values**: Typically, sensors on the devices read the current values and update these observed attributes. An example of this is the current observed speed of a machine (e.g. 1000 RPM).

   b. **A set of Desired values:** These are the values the controlling application desires to be set on the device. For example, an application can set the engine speed to 1200 RPM.

   In addition to these two primary sets of values, these implementations also store associated information like the name or serial number of the device or its current location in the JSON document.

   Essentially, these simplistic device models constitute a simple formal asynchronous communication mechanism of the state of the device over the transport protocols like MQTT/HTTP. Note that the asynchronous communication mechanism is needed because the device may be offline, or it may not be efficient to poll the device when the backend wants to communicate with it. In this sense, these models implement only the "Operational States captured from actual sensor data - current, past actual" part of the DTI concept outlined by Dr. Grieves.

2. **Industrial twins**

   These implementations are typically adopted by the Industrial IoT vendors, and these constitute information from the PLM tools on the design of a machine (similar to the DTP concept outlined by Dr. Grieves) as well as a model of one device (similar to part of the DTI concept). Some industrial vendors look at the physical properties, design information and real time data and present them in an asset/device model graph. It may be worth noting that these models are often predominantly based on the physics based properties of the machine.

Oracle Strategy (Oracle Digital Twin in Oracle Internet of Things Cloud)

The following diagram describes Oracle's approach to Digital Twins implemented in Oracle Internet of Things Cloud:

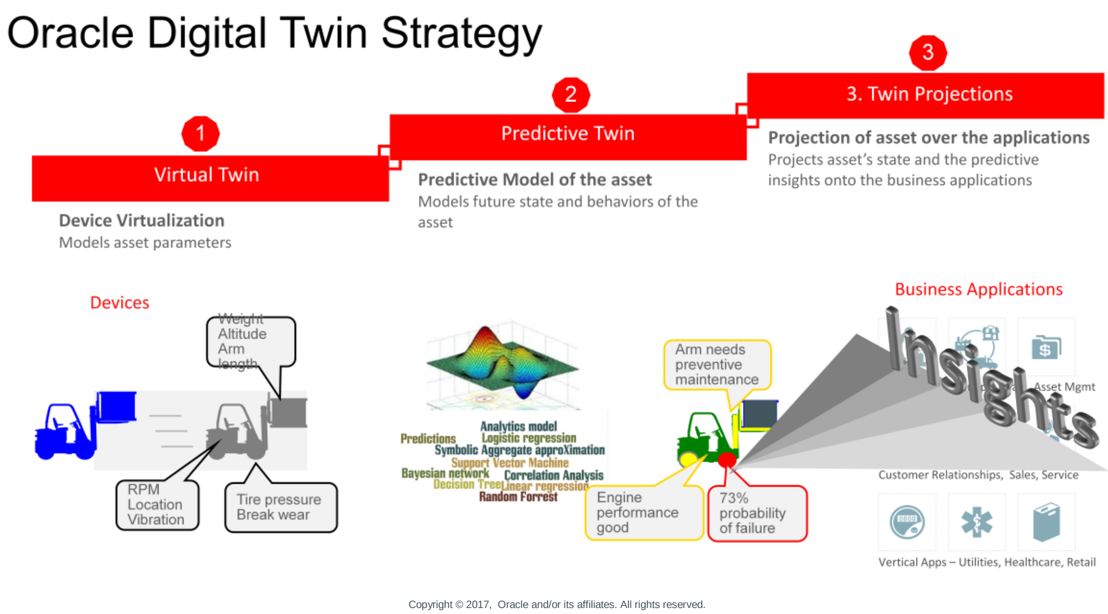

Figure 1: Oracle's Digital Twin Strategy

The figure above describes the elements of a comprehensive approach to Digital Twins. There are three pillars to this approach.

Virtual Twin:

Oracle's Device Virtualization refers to creating a virtual representation of a physical asset or a device in the cloud. This is needed for several reasons. First, the physical asset may not always be connected to the applications. For example, a connected car could be travelling through a tunnel and could lose connectivity momentarily. It is important for the rest of the backend software to be able to interrogate the last known status or to control the operating parameters even when the device is not online/connected. Second, the devices are connected over a large variety of protocols and connectivity methods. Business applications such as an ERP system should not be encumbered with this complexity. Device virtualization offers an abstraction for secure bidirectional communication between the world of business applications and that of the devices.

In addition to the basic models similar to those based on simple JSON documents with observed and desired values, Oracle IoT Cloud delivers Device Virtualization using a powerful semantics model. This semantics model offers several benefits. One notable advantage of this semantics model is that it allows specification of the normal operating ranges for the device attributes. This simplifies the implementation of Edge Computing/Fog significantly. In a typical implementation, in order to detect the threshold violations of a given parameter (e.g. the temperature is too high), users have to write a separate gateway application that processes this and then manage the lifecycle (deployment, upgrades, security, etc.) of that application. With Oracle Device Virtualization, the device model is intrinsically intelligent enough to detect the abnormalities and generate appropriate alerts without requiring the user to write and deploy an edge-computing program. Further, the business rules that are declaratively defined on top of complex event processing (CEP) engine in the Oracle IoT Cloud Service can also be automatically instantiated at the edge of the IoT network.

In addition, Oracle Device Virtualization technology can significantly optimize the volume of the network traffic as well as optimize the delivery mechanism due to the semantic awareness built into the device model. While most of the prevailing implementations have focused on incremental approaches by using efficient protocols such as MQTT to wrestle with the cost of network bandwidth, our revolutionary approach of automated statistical modeling at the edge based on the semantic model drives down the network traffic by orders of magnitude. In order to illustrate this concept, let's look at an example of monitoring operating parameters of a fleet of connected automobiles. Based on the semantics model, the edge is intelligent enough to decide when the operating parameters are in the 'normal' range and when they are not. Further, it knows which messages are urgent (e.g. a check-engine light or a break-failure notification), which messages are important (e.g. low tire pressure), and which messages are routine (e.g. the oil viscosity is slowing deteriorating towards an unacceptable range). The semantics based model is able to automatically optimize the frequency of messages as well as determine the best delivery mechanism (e.g. whether to send an alert message over cellular network versus download the data when the vehicle is connected to a WiFi network at the end of a shift).

Predictive Twin:

Once we have implemented the Device Virtualization, we get a functional abstraction for interacting with the device. For example, we can interrogate the device or control it through the virtualization abstraction. Using the model, we can react to the current status of the device.

However, merely 'reacting' to the situation is neither sufficient nor it is optimal. For example, just knowing that a machine has developed a problem is good. But knowing that a machine is likely to develop a problem in future is even more important as it gives the user time to deal with the problems before they occur.

Behavioral and Predictive Modeling can be done in two ways:

1. **Physics-based approach:** A model can be created using physics-based approach utilizing the knowledge of the exact design as well as manufacturing parameters of the asset. Techniques such as finite element analysis are often used to create fairly accurate models that typically answer the 'what if' questions. For example, using such models, users can estimate the stress patterns on various parts of a machine for a given set of loading conditions.

   In practice, building these models takes significant effort by the team that designed these products to create models with reasonable fidelity.

   Oftentimes, the math involved in creating the FEM models tends to be fairly complex, and hence these models tend to be fairly static and they don't adapt to the complex and continuously changing environment. But the biggest drawbacks of these models include (a) it often takes the original designers of the machines to create these models, and a customer who buys an assembled product cannot put together a model that suits their needs, and (b) while these models can model answer questions about performance on various load conditions, these models don't provide guidance for fixing the issues.

2. **Analytical/Statistics based models:** A predictive model can be built using the machine learning techniques without necessarily involving the original designers. A data analyst can create a predictive model merely based on external observations of the machine. This option proves to be far more practical because it provides creation of various models based on the end customer's needs.

   Another significant aspect that makes these models extremely popular is the fact that they take into account the 'whole system'. We call it the contextual data. Take an example of a manufacturing operation. The Ishikawa diagram suggests that to identify the problems, you need to look at the famous 5Ms: Man, Machine, Method, Material, and Management.

   Using the built-in integrations with Oracle as well as non-Oracle applications, Oracle IoT Cloud can bring the contextual data in the backend business applications in addition to the traditional machine data streaming through the IoT system. This allows creation of models that far superior in their effectiveness and usability than the Physical models above.

   Not all predictive models are created equal. There is a spectrum of complexity based on what you are trying to solve for. The simples of these models are typically based on using trends and patterns in the data. For these, Oracle Stream Explorer (a Complex Event Processing engine with a declarative business-user level UI) that is included in the IoT Cloud is sufficient. Slightly more complex models can be created using Apache Spark based analytics engine in the IoT Cloud. In addition to the standard libraries packaged with an Apache Spark distribution, Oracle IoT cloud offers additional libraries for dealing with Time Series data. More complex models are typically developed by Data Scientists using Oracle R Advanced Analytics for Hadoop (ORAAH). These R models can then be executed in the IOT data pipeline. Business users can use simpler interfaces that are provided by the Oracle Big Data Discovery product. Oracle offers a rich set of tools for solving business problems of various complexities.

Twin Projections:

The predictive models can generate predictions and offer insights into the operation of machines. These insights are pretty useless unless these become integral parts of your existing business processes. This requires 'projecting' the insights onto your backend application infrastructure so that the business applications can freely interact with the IoT system to create an intelligent system.

The purpose of Twin Projection is:

- Integration of insights generated in the IoT system with business processes
- Triggering appropriate remedial business processes/workflows
- Accessing contextual or transactional data for decision support from business apps
- Allowing business apps visibility into current and predicted machine states and environment

Oracle IoT Cloud supports the following integrations:

- Native pre-built integrations with Oracle applications like Oracle ERP (Supply Chain, Manufacturing, Maintenance applications) and CX (Service)
- Integration with over 150 applications through the Oracle Integration Cloud Service
- REST APIs for integration

Summary

Digital Twin is an important concept that is going to be strategic to your business operations as IoT deployments proliferate through your organization.

Oracle IoT Cloud offers the most comprehensive implementation of the Digital Twin through:

a) Virtual Twin – through Device Virtualization that goes above and beyond the simplistic JSON documents enumerating observed and desired values

b) Predictive Twin – through analytics models built using a variety of techniques to suit the complexity of the problem you are trying to solve

c) Twin Projections – projecting insights generated by the twin on to your backend business applications to make IoT an integral part of your business infrastructure

References:

1. Cearley, David (Oct 2016). Gartner's Top 10 Strategic Technology Trends For 2017. Retrieved from http://www.forbes.com/sites/gartnergroup/2016/10/26/gartners-top-10-strategic-technology-trends-for-2017/#dcb32f704437

2. Dr. Grieves, Michael and Vickers, John (2001). Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems (Excerpt). Retrieved from http://research.fit.edu/camid/documents/doc_mgr/1221/Origin%20and%20Types%20of%20the%20Digital%20Twin.pdf

References to Oracle Products mentioned in this paper

1. Oracle IoT Cloud: https://cloud.oracle.com/iot
2. Oracle R Advanced Analytics for Hadoop (ORAAH): http://www.oracle.com/technetwork/database/database-technologies/bdc/r-advanalytics-for-hadoop/overview/index.html
3. Oracle Big Data Discovery: https://www.oracle.com/big-data/big-data-discovery/index.html

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

- blogs.oracle.com/oracle
- facebook.com/oracle
- twitter.com/oracle
- oracle.com

Integrated Cloud Applications & Platform Services

White Paper Title
January 2017