WHITE PAPER

An Introduction to IoT

Internet of Things or IoT is a widely-used industry buzzword offering significant transformation opportunities for businesses.

An IoT solution consists of sensors and actuators, communication channel, data collection on the cloud, and applications. It comes with its own challenges of scale, diversity, connectivity, privacy, security and regulatory compliance.

IoT platforms implement typical reference architecture comprising the Edge, Platform and Enterprise tiers and help businesses overcome technical challenges.

Introduction

Key Elements of IoT

Unique Characteristics of IoT

Typical IoT Reference Architecture

IoT Platforms & their Components

THINXTREAM®

Introduction

Internet of Things or IoT is a widely-used buzzword today across industries. IoT promises to enhance user experiences around products in the consumer, enterprise, and industrial segments.

It helps create a digital identity for a device, collects data from it and its surrounding environment, and enables its remote control. IoT offers several transformation opportunities for businesses to deliver new user experiences, provide additional revenue channels, improve process efficiency, improve product reliability and reduce costs. It also brings numerous challenges for developers of IoT platforms and IoT solutions, due to the scale, velocity and variety of data from devices.

This white paper discusses the unique technical challenges in an IoT implementation, prescribed reference architecture from the industry and how IoT platforms in the market help to overcome these challenges.

Key Elements of IoT

Any IoT solution broadly consists of four building blocks, namely sensors and actuators, communication channel, data collection on the cloud, and applications which deliver information to people and businesses.

Sensors and Actuators: Devices or sensors are responsible for gathering information either about themselves or about their environment at a "point of activity." They can be home appliances, specialized environmental sensors, wearable devices, sensors inside a machine, or any number of commonly found devices. Actuators receive commands from the applications and execute them. Examples of these include turning off the heater, closing the door or sounding an alarm.

Communication Channel: IoT devices share the collected information with a cloud-based service for subsequent processing. Devices may directly connect to the cloud over Wi-Fi® or a cellular network. Some may support only short-range communication and hence go through an intermediary field gateway to the cloud.

Data Collection on the Cloud: This is where the great value inherent in IoT is created. The data from various IoT devices is combined with other data in the cloud to provide valuable information for the end-user. This may involve one or more of these steps, namely, storing, cleaning, transforming and modeling data.

Information Delivery: The last and the most important step in the IoT chain is delivery and consumption of useful information by either end-users, devices or business systems. If it is an end-user, the goal is to provide the information in as simple and transparent a method as possible across multiple device platforms – tablets, smartphones, desktops. If it is a device or business system, then data has to be packaged into a format compatible with their interfaces.

Unique Characteristics of IoT

Having understood the basic building blocks of an IoT system, let us examine the unique challenges or characteristics of an IoT system.

Enormous Scale: IoT solution deployment entails thousands of interconnected devices, which connect to servers (on-premise or in the cloud) over near real-time networks. Server infrastructure is built on multiple interconnected services and applications from different vendors. Building such a big, complex, multivendor environment is always a challenge.

Platform Diversity: In such a diverse field, there are many software, firmware, and hardware platform variants. In addition, there are different network protocols and mechanisms for device-to-server connection. Connecting these disparate systems without standards on this scale and diversity and deriving meaningful outcomes is a significant challenge.

Software-Hardware Interconnection: An IoT solution is not limited to just the software application or the device. It's about both working together and delivering value to the end-user. Be it designing a solution or testing, this dependency between hardware and software presents few unique challenges such as compatibility issues, device availability and simulation.

Connectivity: Thousands of connected devices reporting telemetry data periodically places a significant load on the network. Challenges from unreliable network hardware and Internet connections could impact device performance and ultimately the IoT solution. Ensuring connectivity between key elements for all such real-life situations is a constant requirement throughout the lifetime of an IoT solution.

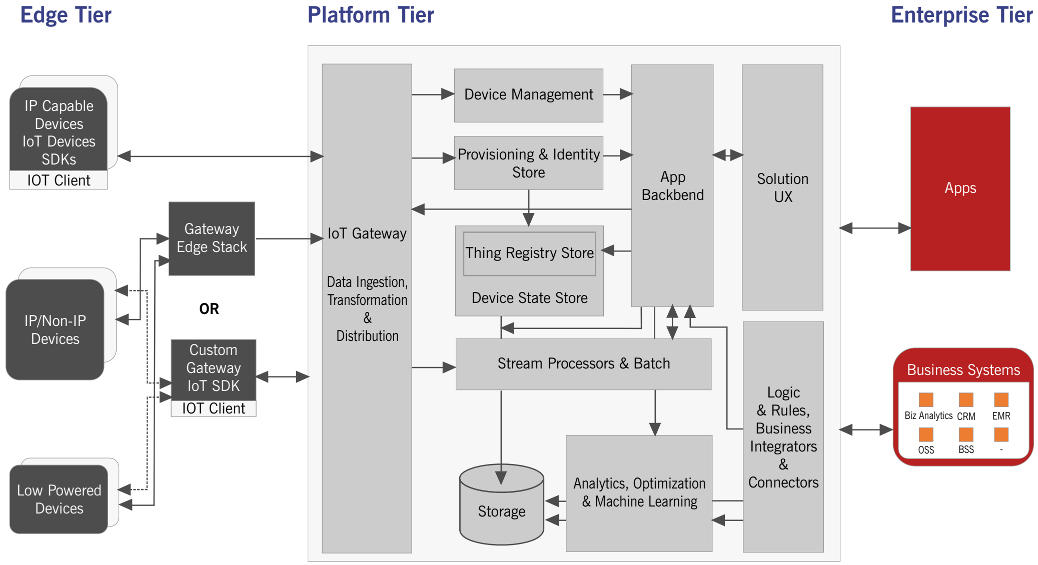
The Rate of Changes: The state of certain devices changes rapidly, e.g., in the case of a wearable device – connected/disconnected, walking, stopping, sleeping and waking up the context of devices including location and speed. Such changes are aggregated at the cloud level and could result in enormous data.

Security and Privacy: Networked devices and applications exposed on the public Internet are always vulnerable to being hacked. Conforming device and applications against the prescribed security standards is vital. As IoT grows, hackers are constantly trying to find system weaknesses. Constant security upgrades and testing is a must in any IoT deployment.

Analytics Challenges: The real value of an IoT solution comes from analyzing the collected IoT data and deriving actionable insights. But its sheer size, accuracy, completeness, and outliers can pose a big challenge. For instance, incorrect data sent and recorded on the cloud could lead to inaccurate analytics which may impact decision making.

Regulation: Collecting data pertaining to an individual, enterprise, or a city requires thoughtful consideration on security, privacy, law of land and civil rights to name a few. And, when there is a conflict between these aspects due to either the application or the implementation, it makes the task of meeting all regulatory requirements very challenging. For e.g., laws which prevent data from going out of the country, the conflict between law enforcement, surveillance and civil rights. Legal liability for unintended uses, security breaches or privacy lapses could be a huge negative factor for IoT adaptability.

Typical IoT Reference Architecture



Copyright © 2018, Microsoft Corp.

Figure 1. Typical 3-tier IoT reference architecture

A typical IoT reference architecture comprises 3 broad tiers, viz., Edge, Platform and Enterprise.

The Edge

The Edge tier collects data from the Edge nodes, using the proximity network and send them to the cloud IoT gateway using WAN protocols. The Edge may be resident on the device itself if it is network-enabled or on a different device, often referred to as field gateway, which is network-enabled. The Edge tier consists of the following set of common functions.

Sensors measure the readings of key parameters such as temperature, pressure, humidity, vibration, noise, etc. Actuators receive commands from the applications and execute them. Examples of these include turning off the heater, closing the door or sounding an alarm.

Controllers are connected to Sensors and Actuators through proximity networks such as Bluetooth®, Zigbee®, UART, USB, etc. Controllers are housed either directly on the device itself, or an IoT Edge device. In either case, they are connected to the cloud over the common TCP/IP WAN mode.

The Edge also does common asset management functionalities like onboarding a sensor, actuators and connected devices, configuring them, managing software upgrade of itself as well as connected devices, and more recently hosting edge analytics, data aggregation, and machine learning models locally.

Platform Tier

The Platform tier receives, consolidates and analyzes data flows from the Edge tier and issues control command on behalf of the Enterprise tier to the Edge tier. It also provides services such as data query and analytics and management functions for devices and assets. This tier comprises of multiple functions which are broadly operational in nature. They are usually hosted on a remote infrastructure such as the cloud or datacenter.

The operations responsibilities include provisioning, management, monitoring and optimization of the systems in the Edge tier.

Provisioning brings assets online remotely, securely and at scale. After provisioning, this layer communicates with them at the asset level as well as the fleet level.

Device Management enables the platform to issue management commands to the Edge or the device and receive responses for these commands from the Edge or the recipients.

Analytics comprises a set of functions for cleaning, enriching and modeling data. This could be done in either streaming mode or in batch mode. The analytics system could also use historical data of asset operation and performance to predict maintenance needs and carry out optimization by reducing energy consumption for instance.

Monitoring and Diagnostics is responsible for processing collected health data so that it can diagnose the real cause of a problem, and then provide alerts on future, abnormal conditions and deviations.

Enterprise Tier

The Enterprise tier implements business-specific applications and provides interfaces to end-users and operation specialists. It receives data from the Platform tier, does high-level business-specific processing/presentation and issues control commands to the Platform tier which is finally executed by the Edge tier. It comprises two major blocks namely domain-specific applications and connectors to business systems.

- **Applications:** The Application block comprises two parts. The first part implements specific functionality that is required for the use-case under consideration, like rules, models, engines, activity flows, etc.
- **Connectors to Business Systems:** The Business Connectors enable end-to-end operations of the IoT systems by integrating the underlying functionality of the Platform tier with business functions including Enterprise Resource Planning (ERP), Customer Relationship Management (CRM), Product Lifecycle Management (PLM), Manufacturing Execution System (MES), Human Resource Management (HRM), asset management, service lifecycle management, billing and payment, work planning and scheduling systems.

IoT Platforms & their Key Components

Because IoT is a system of systems, finding an organization with the required level of technical expertise across all relevant areas is often difficult. IoT platforms enable businesses to overcome technical challenges without needing to figure it out by themselves. They provide the following functionalities:

- Connect devices
- Handle different communication protocols
- Provide security and authentication for devices and users
- Collect, visualize, and analyze data
- Integrate with other Web services

Currently, IoT platforms from AWS® and Microsoft Azure® cloud service providers are the most popular ones in the market. They implement the reference architecture described in the previous section either in its entirety or to a large extent and reduce the barriers for IoT adoption. Here are the few fundamental components of these platforms.

The IoT Gateway is a scalable message broker which can ingest all the events and data sent from millions of devices quickly, reliably and securely. In addition to providing a path to receive events and data from the device, these gateways also provide a reverse channel to send commands and notifications back to the device.

A Rules Engine acts as an extension to the IoT Gateway and can be configured with rules to receive messages from the IoT Gateway published on a specific topic from a device. It executes an action related to the rule on meeting the condition specified in the rule. Actions could be storing the device data in SQL/NoSQL database services, running scalable code (popularly known as “Functions”) for extracting useful data or triggering business workflows.

IoT devices communicate to the IoT Gateway using standard protocols like HTTP, MQTT, and WebSocket. These protocols provide mechanisms for addressing devices and provide constructs to push/pull data, and publish/subscribe for notifications in a secure, reliable and efficient way.

Devices and IoT Gateway authenticate each other using certificates and all messages which flow between them are encrypted using well-known cryptography algorithms. They implement mutual authentication as means of identifying each other.

Device Provisioning is a process of onboarding a new device either singly or in bulk into the IoT Gateway. Devices are configured with device IDs and certificates which are either provided by the IoT Platforms or are provisioned using certificates provided by device manufacturers.

The Identity Registry stores all information about provisioned devices. This information isn't device metadata but is related to identity and authentication. It provides monitoring information like connection status (connected or disconnected) and last activity time; you are able to enable, disable, update, and delete the devices using the registry. It also maintains connection status and last activity time.

Device State Store often referred to as “Shadow” or “Digital Twin” by commercial IoT platforms, stores and retrieves state information pertaining to the device. The last reported state sent from device is stored in the State Object and an Application can request a change to the device state (named desired state) by writing to the Device State Store so that the desired state is replicated to the real device by the message broker; after changing its internal state, the device replies with the new state that is stored as new reported state in the State Object. Devices are able to report their state by publishing messages to the message broker through topics; the broker delivers received messages to all clients subscribed on the specific topics.

The Thing Registry contains device-related information such as custom attributes that are part of the device's metadata e.g., manufacturer, serial number, model, etc. IoT platforms provide APIs to interact with the Thing Registry.

IoT SDK for devices/gateway provides functions for accessing the message broker i.e., publishing and subscribing messages to topics and all the operations related to Device State Store for updating, retrieving and deleting them.

IoT platforms provide a lot of other out-of-the-box-services like Stream Analytics which are often coupled with the IoT Gateway to receive all data and execute real-time analysis on them to raise an alert or generate messages for other systems like Service Bus. The Machine Learning service processes data and executes predictive analysis with either predefined or application specific models. The Charting and Dashboarding tools provides a great way to visually display information to end-users in few clicks.

For business system connectivity, these platforms also provide out-of-the-box line-of-business application integration for SAP®, Oracle®, Salesforce®, SQL Server®, and WebSphere® MQ through Service Bus, workflows, API Gateways, and primitive integration points like mails, queues, and notification channels.

Conclusion

IoT is the latest industry buzzword and is fueling the next major technology revolution after the recent explosion in mobile and cloud IT use. However, IoT comes with its own challenges of scale, diversity, connectivity, privacy, security and regulatory compliance.

Many IoT platforms have recently emerged, which implement reference architectures, provide the required foundation and address the most common challenges. IoT platforms offer provisioning, data ingestion, security, rule processing, stream analytics, storage, machine learning to connectors for enterprise business applications. “Pay-per-use” business models enable the rapid implementation of low-cost, proof of concept IoT solutions without writing code, and easy extension to production quality and scalable solutions, which can connect to millions of devices.

As an IoT services provider, Thinxtream has delivered quality IoT solutions based on both AWS IoT and Azure IoT Hub, as well its home-grown IoT platform DeviceMaestro® Smart™. It is well-versed in the IoT challenges that come from scale, diversity, connectivity, privacy, security and regulatory compliance. By leveraging the IoT expertise built over a decade, Thinxtream ensures cost-effective, quality and timely delivery of IoT solutions.

References

- https://www.pddnet.com/blog/2015/03/4-key-elements-iot
- https://blog.apnic.net/2015/10/20/5-challenges-of-the-internet-of-things/
- https://smartnet.niua.org/sites/default/files/resources/t-tut-smartcity-2016-2-pdf-e.pdf
- https://www.iiconsortium.org/IIC_PUB_G1_V1.80_2017-01-31.pdf

Thinxtream Technologies is a global software company with a portfolio of innovative software platforms, products, components, solutions, patents, competences and services for Internet of Things (IoT) across several industry verticals and applications, successfully enabling leading customers, including Fortune 500 companies, meet their application, product and business goals.

Interested in learning more? For more information contact:

Thinxtream Technologies Pte. Ltd.
220 Orchard Road #05-01
Midpoint Orchard
SINGAPORE 238852
Phone: +65 66358625
Email: info@thinxtream.com

Thinxtream Technologies, Inc.
10260 SW Greenburg Road
Suite 400 Portland, OR 97223,
U.S.A
Phone: +1 503 293-3598
Email: info@thinxtream.com

www.thinxtream.com

LinkedIn/thinxtream

All prices, specifications and characteristics set forth in this publication are subject to change without notice.

THINXTREAM®