IBM

developerWorks®

Simplify the development of your IoT solutions with IoT architectures

Strategies for creating scalable, flexible, and robust IoT solutions

Anna Gerber

August 07, 2017

Apply these strategies when you plan data-driven IoT architectures. These strategies help you simplify development, manage complexity, and ensure that your IoT solutions remain scalable, flexible, and robust.

IoT 201: Building skills in IoT development

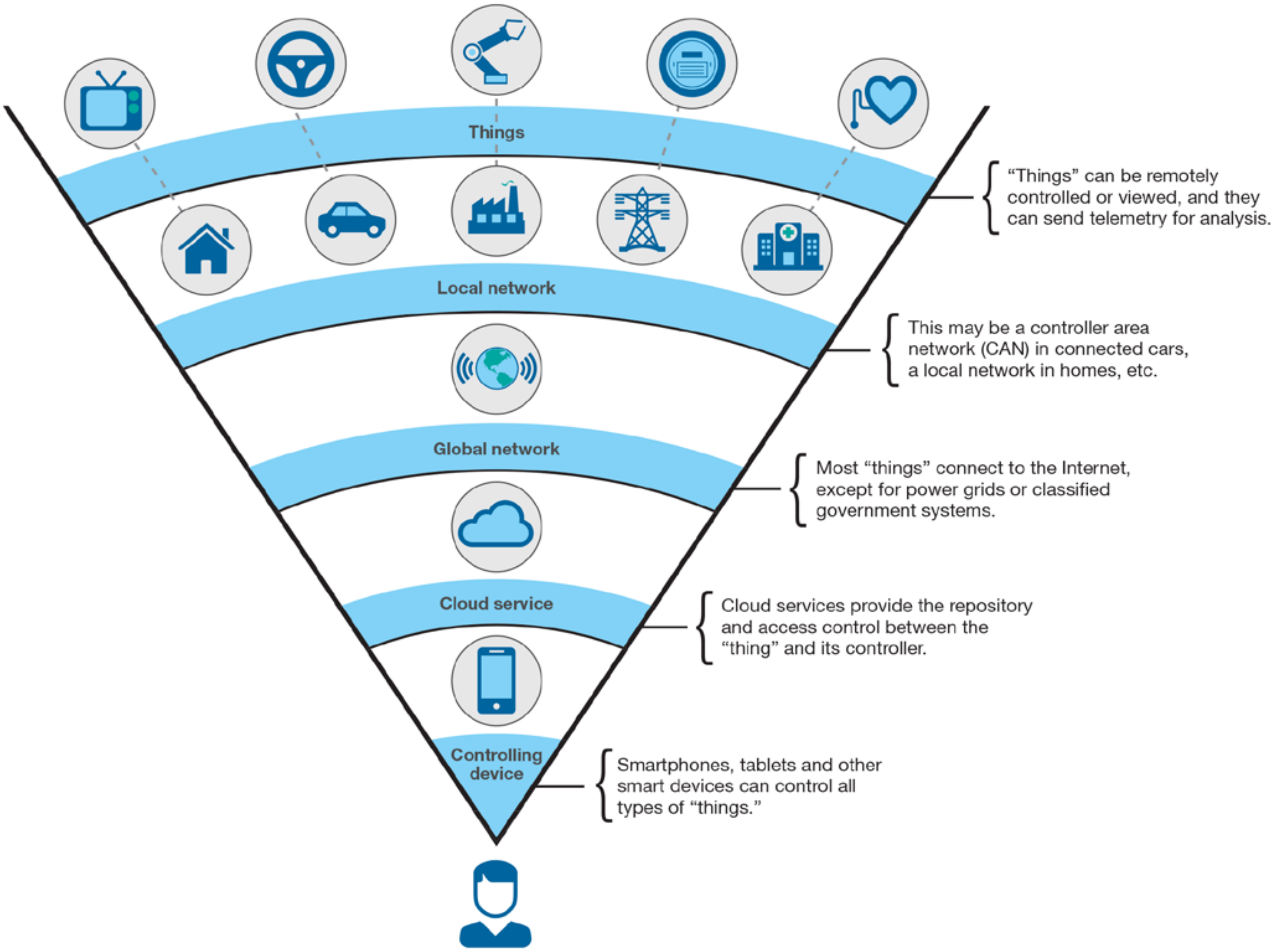
This article is part of the IoT 201 learning path, a next-steps developer guide for IoT.

- Diving deeper into IoT development
- IoT developer kits
- IoT architectures (this article)
- Large-scale IoT solution: Connected cities
- Tutorial: Develop a healthy habits tracker

One of the biggest challenges you face when you are planning Internet of Things (IoT) solutions is dealing with complexity. A typical IoT solution involves many heterogeneous IoT devices, with sensors that produce data that is then analyzed to gain insights. IoT devices are connected either directly to a network or through a gateway device to a network, which enables the devices to communicate with each other and with cloud services and applications (Figure 1).

Figure 1. Human view of Internet of Things (Source: X-Force Research and Development, “IBM X-Force Threat Intelligence Quarterly 4Q 2014,” Doc

WGL03062USEN, Publish Date: Nov 2014. http://www.ibm.com/security/xforce/downloads.html)


Things
"Things" can be remotely controlled or viewed, and they can send telemetry for analysis.
Local network
This may be a controller area network (CAN) in connected cars, a local network in homes, etc.
Global network
Most "things" connect to the Internet, except for power grids or classified government systems.
Cloud service
Cloud services provide the repository and access control between the "thing" and its controller.
Controlling device
Smartphones, tablets and other smart devices can control all types of "things."

Edge computing describes the work that happens at the edges of the IoT network, where the physical devices connect to the cloud. With a focus on reducing latency, improving privacy, and reducing bandwidth costs within data-driven IoT applications, edge computing architectures are becoming increasingly common within IoT.

In this article, I discuss the following strategies that you can apply when you plan data-driven IoT architectures. These strategies help you simplify development, manage complexity, and ensure that your IoT solutions remain scalable, flexible, and robust:

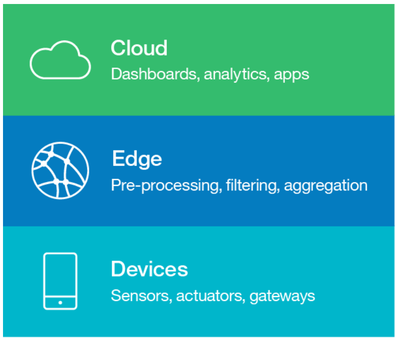
- Adopt a layered architecture
- Security by design
- Automate operations
- Design for interoperability
- Follow a reference architecture

Adopt a layered architecture

An architecture describes the structure of your IoT solution, including the physical aspects (that is, the things) and the virtual aspects (like services and communication protocols). Adopting a multi-tiered architecture allows you to focus on improving your understanding about how all of the most important aspects of the architecture operate independently before you integrate them within your IoT application. This modular approach helps to manage the complexity of IoT solutions.

For data-driven IoT applications that involve edge analytics, a basic three-tiered architecture, which is shown in Figure 2, captures the flow of information from devices, to edge services, and then out to cloud services. A more detailed IoT architecture would also include vertical layers that cut across the other layers, like identity management or data security.

Figure 2. Layers of an IoT architecture


Cloud
Dashboards, analytics, apps
Edge
Pre-processing, filtering, aggregation
Devices
Sensors, actuators, gateways

Devices layer

You can read more about the IoT device characteristics in my IoT hardware guide.

You can read more about the network technologies in my IoT connectivity guide.

The components in the device layer (shown at the bottom of Figure 2) include physical sensors and actuators that are connected to IoT devices and the IoT devices themselves. Sensors and actuators on their own are typically not considered to be "smart" devices, but sensors and actuators often connect either directly or wirelessly over technologies like Bluetooth LE or ZigBee to IoT devices that have more processing capabilities.

Some IoT devices communicate directly with related cloud services and apps. However, it is common for IoT devices to communicate upstream through gateways, which are intermediate devices that have slightly more processing power than the basic IoT devices. Although they don't always have sensors attached directly, gateway devices play an important role in the data acquisition process. They can perform basic analog-to-digital conversions, scaling, and other normalization of the raw sensor data readings.

Edge layer

The Edge layer (shown as the middle tier of Figure 2) relates to the analytics and pre-processing services that are located at the edge of the network. Edge analytics occurs in real time (or near real time) by processing the stream of data at the point where the data is collected as it comes in from the sensors. Basic pre-processing tasks like filtering and aggregation of data are performed at the edge, and then key pre-processed data is transferred upstream to cloud services and applications for further processing and analytics.

Cloud layer

After the data has been prepared, it is sent upstream for further processing, storage, and use within cloud applications, in the cloud layer (shown as the top tier of Figure 2). The cloud applications that perform the data processing are often complemented by mobile apps and web-based client applications that present the data to end users and that provide access to tools for further exploration and analysis through dashboards and visualizations.

Implement "Security by Design"

You can read more about security by design in this IBM Point of View white paper, *Internet of Things Security*.

To achieve end-to-end security within an IoT solution, security must be a priority across all of the layers of your IoT architecture. You need to think about security as a cross-cutting concern in your IoT architecture, rather than as a separate layer of your IoT architecture to be tackled in isolation at the end. With so many devices connected, the integrity of the system as a whole needs to be maintained even when individual devices or gateways are compromised. Ensure that your architecture supports multiple layers of defense. Also, ensure that your IoT solution can identify and neutralize devices that become compromised, such as by using gateways to isolate vulnerable devices and to monitor communications and usage patterns to detect anomalies.

You can read more about IoT security in this developerWorks article series.

You should adopt standards and best practices for these aspects of your IoT infrastructure:

- Device, application and user identity, authentication, authorization, and access control
- Key management
- Data security
- Secure communication channels and message integrity (by using encryption)
- Auditing
- Secure development and delivery

Automate operations

Ensure that your IoT architecture supports automation and orchestration across all of the layers. Plan to use these automation features when you roll out your IoT solution so that you can support fast and easy development and deployment. For example, a microservices architecture at the Edge or Cloud Layer can be implemented by using container technologies and orchestrated by using tools such as Kubernetes, provided by your IoT platform. These features make operations like setting up a new device or gateway or deploying a new instance of a cloud application to process the device data less error prone. Avoiding manual configuration ensures that operations are repeatable, which is essential for being able to scale up to IoT solutions that involve thousands or even millions of connected devices.

Design for interoperability

The diversity of devices, network protocols, and data formats that you adopt in your IoT solutions presents one of the greatest architectural challenges for IoT. If you intend to adopt more than one IoT platform in your IoT solution, you'll need to consider whether the technologies that are used within each IoT platform can be integrated into a cohesive solution.

One of the best strategies for maintaining interoperability in IoT is adopting standards. Standards provide you with flexibility to switch out or introduce additional components, as long as the new components conform to the same standards that you already adopted.

Reference architectures also provide guidelines to assist you in planning your IoT architecture. They are usually standards-based and encapsulate design patterns and best practices. Adopting a reference architecture, and then selecting IoT platforms that implement them by following the guidelines described in the reference architecture, is a reliable strategy for maintaining interoperability within your IoT architecture.

Follow reference architectures

There are many initiatives currently working towards standardizing IoT architectures to improve interoperability. IoT platform vendors and research partners collaborate through these initiatives to define IoT reference architectures. Reference architectures act as an architectural foundation, describing high level building blocks that are used within IoT solutions and establishing a shared terminology for key architectural concepts. These initiatives draw on a broad range of existing solutions to highlight effective design patterns and best practices.

Some widely referenced IoT reference architectures include:

- Internet of Things – Architecture (IoT-A): The IoT-A reference model and architecture was developed through an EU lighthouse project in 2013. IoT-A was designed to be built upon in order to develop concrete architectures that are applicable across a range of domains.
- IEEE P2413 - Standard for an Architectural Framework for the Internet of Things (IoT): This ongoing IEEE standardization project aims to identify commonalities across IoT domains, including manufacturing, smart buildings, smart cities, intelligent transport systems, smart grid, and healthcare.
- Industrial Internet Reference Architecture (IIRA) – IIRA was developed specifically for industrial IoT applications by the Industrial Internet Consortium, which was founded in March 2014 by AT&T, Cisco, General Electric, IBM, and Intel.

Reference architectures can be used as a template for developing IoT solutions. The architectures listed above describe IoT architectural components and their functions in high-level terms, but they can be made more concrete by mapping abstract requirements onto specific technologies or technology stacks.

Components of an IoT Reference Architecture

The details of reference architectures differ depending on the application domain; however, most IoT reference architectures describe at least the following capabilities:

- Managing devices and their data
- Connectivity and communication
- Analytics and applications

In addition, reference architectures usually also describe mechanisms to address non-functional requirements such as flexibility, reliability, quality of service, interoperability, and integration.

Managing devices and their data

The device management aspect of a reference architecture is concerned with managing devices, their identity, and their lifecycles. Reference architectures describe:

- Device onboarding
- Updating device firmware
- Applying new configurations
- Triggering remote operations like disabling, enabling, or decommissioning devices

Connectivity and communication

Managing connectivity and bidirectional communication between devices, between devices and gateways, and between gateways and cloud services and applications is another key capability that is often described within IoT reference architectures. For edge computing, event-driven architectures are a good choice, with communication between devices and services using publish/subscribe protocols and message brokers.

Analytics and applications

To derive value from the data that comes from IoT devices, cloud applications provide visualization and analytics tools that operate over streams or batches of data to identify actionable insights. Depending on the use case, decision management and business process tools can trigger alerts or carry out actions in response.

Concrete Reference Architectures

The patterns and guidelines that are often provided by a reference architecture can be drawn from within a specific IoT domain, such as Industrial IoT, or can be generalized from solutions across a range of domains. High-level generalized architectures can be used as templates to create more concrete domain-specific or platform-specific architectures.

You can read more about IoT platforms in my IoT platform guide.

General-purpose IoT platform vendors often provide more applied reference architectures, along with implementation guides for developing IoT solutions that conform to their reference architectures using the tools and software agents that are provided by the platform. Some IoT platform-centric reference architectures include:

- IBM IoT reference architecture
- Intel IoT platform Reference Architecture
- Microsoft Azure IoT Architecture
- Amazon Web Services Pragma Architecture

IBM Industrie 4.0 reference architecture is an example of a domain-specific reference architecture, which is designed for industrial IoT applications, based on the IIRA reference architecture and the IBM IoT Reference Architecture.

Conclusion

Designing data-driven IoT solutions is complex due to the scale and heterogeneousness of the devices and connectivity involved. In this article, I've outlined some strategies for designing secure, flexible, and scalable IoT architectures.

Related topics

- IoT hardware guide
- IoT connectivity guide
- IoT platforms guide
- IBM Point of View white paper, Internet of Things Security