mendix

IoT Solution Architecture: Essentials Guide

Introduction

The Internet of Things (IoT) is a powerful, transformative force and cornerstone for digital businesses taking advantage of the convergence of the physical and digital worlds.

The term itself was coined in 1999. Since then, we have seen the machine-to-machine era where devices began to communicate with each other, and industrial automation technologies moving from the factory into new environments like hospitals, banks, and buildings. Today, the IoT is connecting things, businesses, and people in real time and at massive scale. It is reshaping organizations across every sector of the economy.

Predictions about the proliferation of connected things vary from 20 to 50 billion devices by 2020, depending on which analyst firm you ask. While even conservative predictions are mind-boggling, we should keep in mind that IoT ultimately is not about 'things'. It's about making sense of the sensor data in order to deliver new services and experiences.

McKinsey estimates that IoT could generate $11.1 trillion a year in economic value by 2025. Understandably, IT leaders face growing pressure from the business to deliver new IoT solutions that improve operational efficiency and grow revenue via connected products and services.

So, what does it take to develop IoT solutions, and how do you establish the right architecture? IoT design is quite different from typical IT solutions in that it bridges the physical world of Operations Technology (OT) with sensors, actuators and communication devices, and the digital world of Information Technology (IT) with data, analytics, workflows, and applications. IoT presents new challenges for management, governance, and security as well.

This guide is written for Enterprise Architects who are getting involved in IoT solution development. It describes the essentials of an IoT solution, provides recommendations for its architecture, and shares best practices to get started.

> "IoT ultimately is not about 'things'. It's about making sense of the sensor data in order to deliver new services and experiences."

Table of Contents

IoT Diversity Presents Different Levels of Complexity

IoT use cases are broad and diverse, and come with different levels of complexity. The use of IoT devices ranges from connected turnstiles for monitoring event foot traffic, to sensors for rail infrastructure like signals and switches for predictive maintenance, or smart wind farms optimizing energy production of hundreds of offshore wind turbines.

Some devices simply communicate their whereabouts once an hour over a low-power wide area network, while others deliver a steady stream of data per millisecond into a time-series database that easily grows to petabytes in size.

The diversity of operational requirements creates an array of IoT Endpoints, communication protocols, data management, and analytics technologies, as well as corresponding deployment topologies.

And that's just for establishing the foundation. The real value of IoT comes from turning data into insight, and making it actionable to drive smarter operations. This fuels the need for IoT apps that empower users to act upon insight, by combining sensor data and data residing in enterprise systems such as ERP, CRM, and PLM. This may include 3rd party services such as weather forecast and traffic data. Also, IoT apps should be able to trigger actions in the physical world like turning a system on or off.

The Anatomy of IoT Solutions

Despite the diversity, there is a level of commonality across use cases that could be defined as the anatomy of IoT solutions. Taking a layered approach in describing the anatomy helps identify relevant services and technologies from the things-level all the way up to IoT apps.

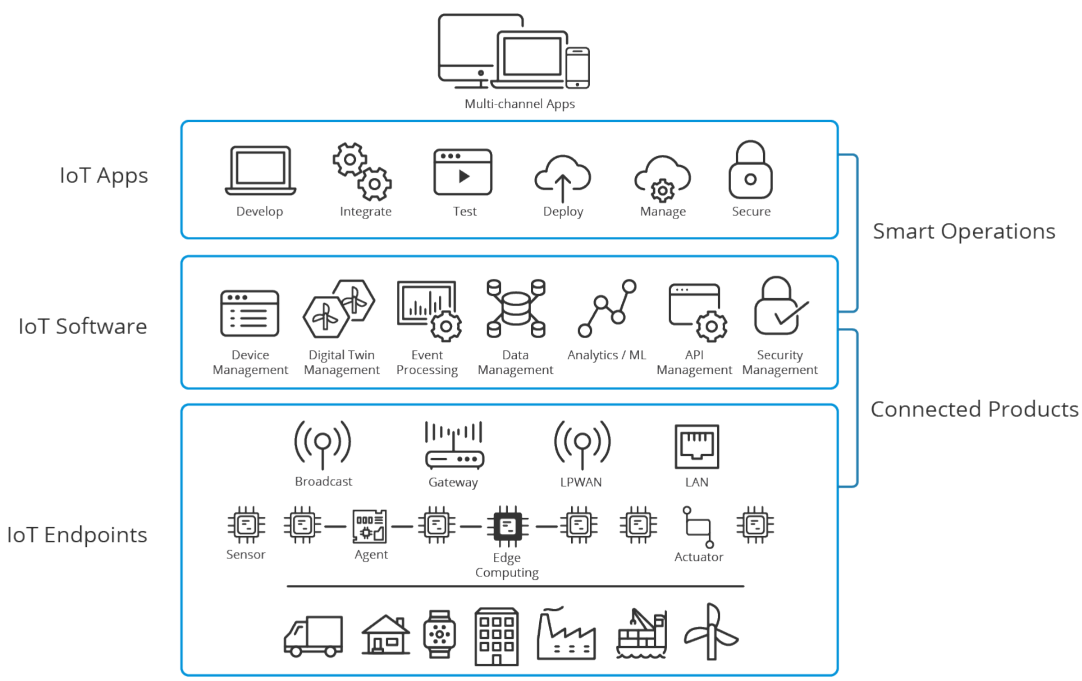


Figure 1 – IoT Solution Anatomy

IoT Endpoints

This layer covers the physical world and operational technology required to connect things and communicate:

- Things

The real endpoint for IoT is obviously the thing that should be connected. This could be a physical product like a car, truck, jet engine, packaging line, lighting system, or an HVAC unit. But it could also include 'things' such as livestock, crop, human beings, or spatial areas like rooms or outdoor space.

- Sensors

Sensors collect and report data on the actual status of things to which they're connected. Sensors could be mounted on, or embedded in, things to monitor temperature, pressure, light, motion, location, etc.

- Actuators

Actuators control the physical or logical state of a product through signals they get from IoT apps or other systems, like opening a valve, or turning a camera, motor or light on and off. This includes commands sent to embedded software e.g. to reboot or update configurations.

- Agents

Agents are components that mediate between a set of IoT devices and act as a bridge between the sensors/actuators and the cloud, deciding what data to send and when. In reverse, they also process commands and updates coming from the cloud.

- Edge Computing Device

Edge computing is a distributed architecture in which IoT data is processed at the edge of the network. Transmitting massive amounts of raw data over a network puts tremendous load on network resources. In some cases, it is much more efficient to process data near its source and send only the data that has value over the network to the cloud.

- Communication

For IoT device communication, the physical layer and communication protocols are to be distinguished. As far as the physical layer is concerned, gateways, mobile devices, mesh networks, and direct- or broadcast device communication are alternatives that may or may not be suitable depending on the IoT use case. The choice for the physical layer will determine which communication protocols are most suitable (e.g. MQTT, COAP, HPPT(S), AMQP, ZigBee, Z-Wave, etc.)

> "IT leaders face growing pressure from the business to deliver new IoT solutions that improve operational efficiency and grow revenue via connected products and services."

IoT Software

The next layer is the (cloud) platform that brings essential IoT software services together to manage the IoT endpoints securely, represent the 'digital twin' of connected things, process and analyze data, and provide APIs to consume and expose services:

- Device Management

Device management simplifies the process of configuring, provisioning, and operating the endpoint devices. It supports monitoring, testing, updating software, and troubleshooting connected devices.

- Digital Twin Management

For many IoT use cases, particularly in industrial IoT, it's valuable to define a digital twin or shadow of the connected thing. This could be as simple as a 1:1 mapping of the physical things to logical identifiers in the IoT Platform, or as sophisticated as mapping an engineering view of an asset with a hierarchical structure of components / systems to the physical devices representing that asset on an instance and class level. The digital twin is the bridge between the OT and IT worlds, and allows approaching and navigating things from a logical view vs. just seeing a 'flat' data stream.

- Event & Data Processing

Event Processing deals with event streams coming from connected devices, filtering and monitoring. In addition, services for data aggregation, data storage, and management are required.

- Analytics / Machine Learning

Analytics services are key to perform statistical analysis and apply machine learning to detect patterns on a device instance or class level for predictive maintenance, making recommendations, triggering engineering changes, etc.

- API Management

API Management is a core service to provide openness on all layers in the IoT platform for device communication, data-, service-, and backend integration, and application development.

- Security Management

Security Management is vital to ensure that IoT endpoints do not expose security threats due to the increased attack surface IoT creates. IoT devices generate sensitive information about operations transmitted over the internet. Also, devices themselves are vulnerable to hacks that could cause serious business damage. Security services should include (certificate-based) device attestation, network connectivity, software upgrades, authentication, identity and access management, and data loss prevention.

> "The real value of IoT comes from turning data into insight, and making it actionable to drive smarter operations."

IoT Apps

The Apps layer is where IoT solutions are brought to life, turning data into actionable insight, putting it in the hands of business users, customers and partners. This is the layer where integrations with existing back-ends and 3rd party services are established and workflows are defined to act upon insight. Core services in the apps layer include:

- Integrated Development Environment (IDE)

A design time environment is required to develop IoT apps. This could be a traditional IDE for coding in a specific programming language or a model-driven environment for visual development of IoT apps. In addition, core services for software configuration management and branching & merging are needed for development teams to commit their work, and create builds and application packages. Finally, the IDE should guide developers to apply the right patterns and best practices for IoT app development.

- Multi-channel Apps

In today's world of web and mobile apps, the IDE ideally supports development of cross-platform, responsive and multi-channel apps, optimized for specific form factors, using device features and supporting gestures with minimal overhead.

- Integration

Integration is the lifeblood of IoT apps. Apps should have access to IoT endpoints (via the digital twin) for reading the full history of a 'thing' after receiving an alert, or triggering an actuator. They should be able to leverage various IoT software services (e.g. time series data and machine learning algorithms) and weave these services into IoT apps. Last but not least, integration with enterprise backends and 3rd party services is needed for managing workflows and making IoT apps contextual e.g. by creating a dashboard for a service engineer, enhanced with engineering and customer support data.

- Testing

Testing & quality assurance are essential disciplines in IoT app development projects. Test automation on various levels (unit test, integration, functional test) helps minimize the test burden relative to (iterative) development cycles.

- Deployment

Staged deployment to target environments and automated provisioning of application resources (web server, OS, database, file storage) helps DevOps engineers to efficiently manage IoT apps. Ideally, there's flexibility to deploy at a cloud of choice—for instance, close to where core IoT services that the application is using are running.

- Management

User management, application management, monitoring, and self-service options for horizontal / vertical scaling and configuring high-availability are important to manage IoT apps. Specifically, support for elasticity backed by a stateless application architecture is essential to deal with variable load and volume.

- Security

Like for the IoT platform layer, security on an app level is vital. This concerns both the application runtime environment and the security settings for the apps themselves (e.g. access and authentication).

The elements that define the anatomy of an IoT solution may come across as overwhelming. As mentioned before, the type and level of sophistication of the IoT Solution will determine how many of the elements and services described are needed to create an end-to-end solution.

Nevertheless, it's clear that the diverse set of endpoints, network technologies, protocols, IoT software, and application development services pose a challenge for enterprises planning to adopt IoT to transform their business operations. The question is: How do you make IoT solution development manageable? The answer lies in adopting a platform approach.

Adopt Platforms to Simplify IoT Solution Deployment

Adopting platforms for the IoT Software and IoT Apps layer will significantly simplify the process of connecting, managing, getting insight from, and building apps for IoT-enabled products and services.

IoT Software Platforms

There's an emerging category of IoT Software Platforms that offer an integrated set of IoT software services in the cloud. Some vendors like Amazon Web Services (AWS), Microsoft (Azure) and IBM (Bluemix) have expanded general purpose cloud platforms with a rich set of IoT services. Others like GE (Predix), Siemens (Mindsphere) and PTC (Thingworx) offer specific Industrial IoT Platforms or platforms focused on managing device connectivity, such as Cisco (Jasper).

Irrespective of background and focus, these platforms essentially offer an abstraction layer that eliminates a lot of the complexity of IoT Solution development.

For IoT endpoints there's no standard 'platform' due to the immense variety and variability of industry-specific sensors, agents, actuators, and gateways. In selecting the IoT Endpoint infrastructure, it's important to ensure that devices:

- Support standard communication protocols

- Can be managed remotely through an IoT Software Platform.

While IoT Software Platforms offer strong support for core IoT services, the capabilities for application development and lifecycle management are often limited to offering an SDK for coding IoT apps. There are some serious drawbacks, however, taking a traditional programming approach.

The Challenges of IoT App Development

IoT is a nascent technology domain, and many companies have yet to start their IoT journey to drive smarter business operations, create new products and change how they interact with their customers. To help the business unlock the value of IoT, IT teams need a way to experiment quickly and cost effectively, so they can turn new ideas into value-driving IoT applications. Many organizations, however, struggle with:

- Lack of Agility
IoT application development requires rapid experimentation, frequent iteration and close collaboration between business and IT. Traditional development approaches don't facilitate the required speed and agility, not to mention active involvement from the business.

- Technical Complexity
Teams often struggle to master the complexity of app development and integrate the heterogeneous mix of endpoints, IoT services, back-end systems, and data required for IoT solutions.

- Skills Shortage
IoT solutions require scarce, hard-to-find skills spanning IoT technology, integration, data management, analytics, and app development. According to Gartner, 49 percent of CIOs expecting to experience IT skills shortages in the next 12 months.

As a next abstraction layer on top of the IoT Software Platforms, companies should consider adopting a high productivity application platform that seamlessly leverages the underlying IoT technologies.

High Productivity Platforms Speed IoT App Development

High Productivity Application Platform as a Service (hpaPaaS) is a fast-growing technology market uniquely suited to help IT leaders and their teams address these challenges and facilitate rapid, iterative development essential to IoT success.

According to Gartner, hpaPaaS "provides application development, deployment and execution in the cloud using declarative, model-driven application design and one-step deployment." In other words, hpaPaaS abstracts away from the technical complexity associated with traditional software development, enabling a broader range of users to rapidly build applications, including, but not limited to, IoT solutions.

Here's a brief overview of some key high productivity platform capabilities supporting IoT application development:

- Visual Development
Helps close the talent gap by enabling both professional and citizen developers to build IoT applications without needing to code. They can visually model their data models, UIs and logic, leveraging a marketplace of building blocks to kick-start development.

- IoT Connectors
Minimizes technical complexity by offering out-of-the-box connectors enabling developers to leverage services from common IoT Software Platforms (e.g. AWS, IBM Watson and Microsoft Azure) in their applications, simply by dragging and dropping IoT services in visual application models.

- Business-IT Collaboration
Facilitates rapid experimentation and iteration by enabling developers and business stakeholders to collaborate across the complete application lifecycle, from ideation and requirements gathering to team development and feedback management.

- Scalability and High Availability
Enables your team to start small with IoT innovation and scale easily, without needing to rework the application architecture. Platforms leveraging underlying technologies like Cloud Foundry or Docker offer built-in application resilience and high availability.

- Multi-Cloud Deployment
Offers the flexibility to deploy IoT apps in the same cloud as the IoT Software Platform, which helps to minimize latency and leverage other services in the same public or (virtual) private cloud accessible under the customer's account and plan.

- Hybrid Cloud Support
Provides the possibility to deploy and manage IoT apps across clouds while centrally maintaining coordination and orchestration. This approach creates an IoT application hub, allowing companies to use IoT software services from various vendors and optimize for parameters like cost or functional coverage.

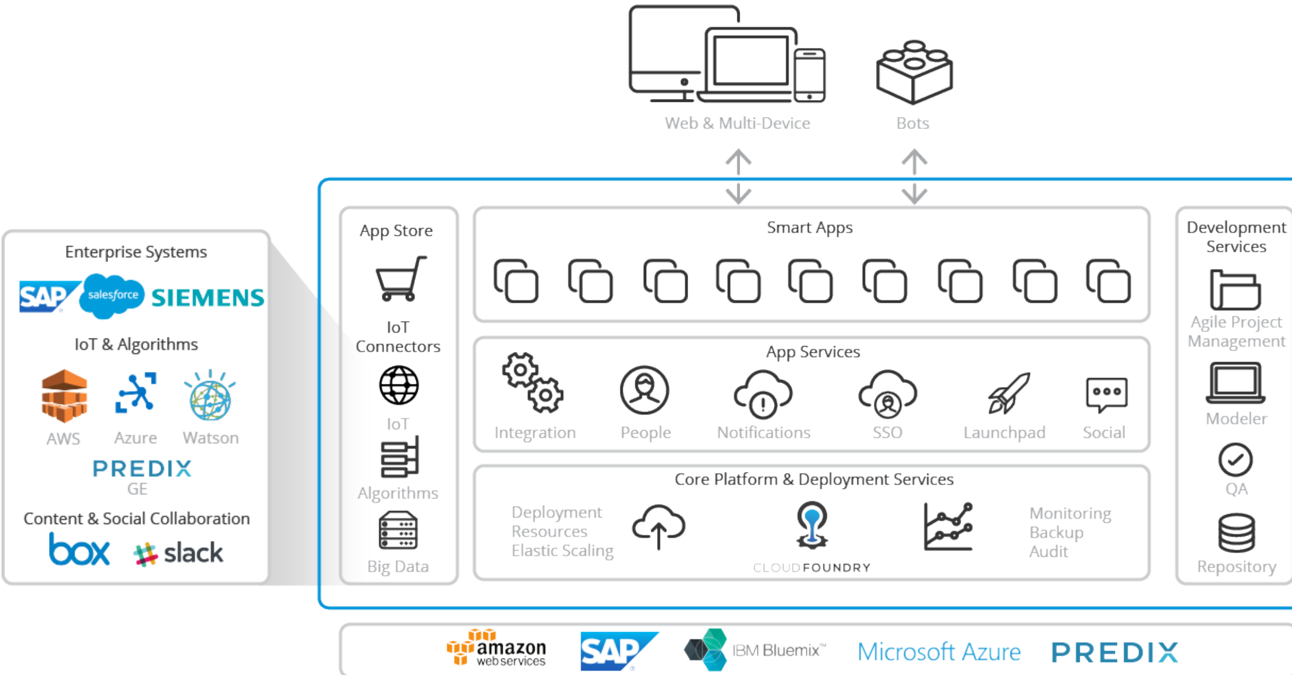


Figure 2 – Core capabilities of a high productivity platform

An integrated high productivity platform accelerates each step of the application lifecycle, helping your organization to realize the shortest time to value. The power of high productivity platforms is that they are not only ideal for IoT solutions, but a broad range of applications to digitize your operations, customer experiences and business models. Paired with leading IoT Software Platforms that offer a broader set of cloud services, high productivity platforms become an essential layer in an organization's innovation stack.

Key Architectural Recommendations

The importance of defining and adhering to a solid IoT solution architecture cannot be underestimated. As many IoT initiatives are initiated in the business, enterprise architects should actively seek to get involved. If not, there's a looming threat of 'quick-and-dirty' IoT solutions, with a myriad of point-to-point integrations, a lack of scalability, a future maintenance nightmare and vulnerable security model.

Assuming a layered approach, and corresponding platforms that abstract technology and reduce complexity are adopted, the following recommendations are a guide for defining a future-proof IoT architecture:

- Plan for diversity

In the absence of industry standards, and considering the broad range of use cases and supporting OT and IT technologies, explicitly plan for, and manage, heterogeneity. Accept the fact that you'll have to work with a variety of protocols and standards and select the right combination on a per project basis.

- Implement an API strategy

Ensure that each layer in the IoT stack is API-based and can expose and consume web services (REST, SOAP, JSON, OData). This will reduce the integration burden significantly and provide the foundation for flexibility. As IoT use cases often trigger additional creative ideas, make sure services can be enhanced easily to leverage data for multiple consumers of IoT data.

- Adopt an event-driven architecture

IoT solutions should be designed for the ability to process large and varying volumes of events in real time. This calls for a loosely coupled, event-driven architecture across the IoT software and apps layer, with capabilities such as in-memory processing to handle load.

- Define a data management policy

As the portfolio of IoT applications grows, data management will become a challenge, as the things that produce sensor data are represented in a variety of systems (PLM, ERP, CRM) across departments and business units. Managing data discovery, reconciliation, and governance becomes a precondition for scaling IoT initiatives.

- Embrace flexibility

No platform delivers everything needed to build differentiating IoT solutions. Be prepared to adopt best-of-breed technologies and domain-specific services, such as predictive maintenance algorithms and visualization techniques that help the business create solutions for competitive advantage.

- Demand openness and portability

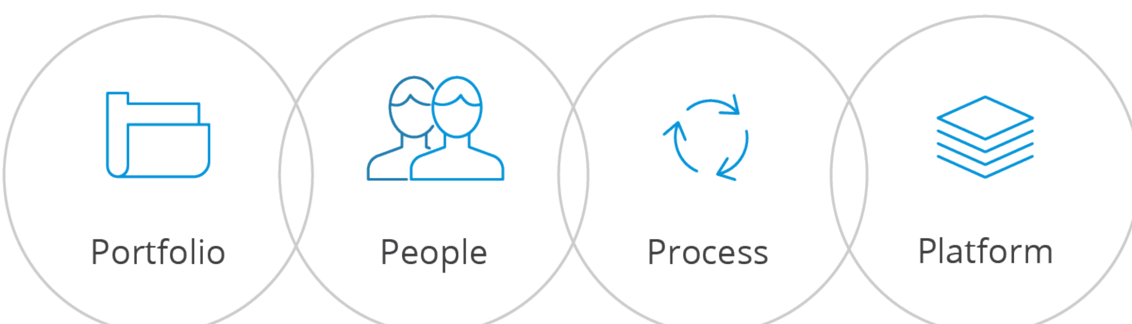
Since the IoT space is emerging, it's likely that today's choices for platforms and technologies will be reconsidered in a few years. To protect current investments in IoT app development it's critical to plan for portability and to use open standards / specifications to prevent lock-in.

- Ensure end-to-end security

Do not compromise security, especially since IoT endpoints and communication introduce new areas of vulnerability. Assess end-to-end security for IoT solutions and make sure that vendors have the right certifications in place and can prove that their platforms are secure and frequently audited.

Define an Incremental IoT Strategy

The Internet of Things opens countless opportunities for businesses to drive smarter operations or even change their business models. But it takes more than selecting technology to achieve success. It requires managing and balancing four key aspects of digital execution:



Early adopters of IoT have begun to unlock significant business value by employing incremental IoT strategies. These organization start small, using IoT to optimize current operations, and gradually expand to deliver more fundamental business model transformation. With that idea in mind, companies need the right set of tools and processes to foster low-cost, high-value experimentation.

The following best practices have proven to foster rapid IoT experimentation:

1. Build an IoT SWAT team
Build a cross-functional team that includes representatives from the business and professionals from OT and IT. The purpose is to bring together people with ideas and the technical aptitude to bring them to life. It's critical to have an Enterprise Architect in the team to oversee the technical implementation holistically.

2. Identify pilot projects
Work with the business to identify IoT pilot projects. Allocate time and resources for this type of experimentation to familiarize yourself with the domains and prove the potential value to your organization.

3. Adopt agile, visual development
Adopt model-driven development to create a common language between business and IT to foster collaboration and faster experimentation. Use agile development (SCRUM) to develop and test a minimum viable product (MVP) early in the process to ensure the ability to change direction with minimal risk based on what you learn.

4. Implement DevOps
Integrate the development, release and ops processes across the chain to enable continuous integration and continuous delivery (CI/CD) of IoT solutions.

5. Create a feedback loop
It is important to have a mechanism to continuously capture feedback from users to feed the process of continuous innovation.

To summarize the essence: think big, start small, learn fast and minimize time-to-value.

> "Think big, start small, learn fast and minimize time-to-value."

Conclusion

The Internet of Things is a driving force behind digital business, and an opportunity to drive smart operations and business model innovation. The IoT domain is nascent and supports a huge variety of use cases across industries impacting business and consumers.

The technology landscape is diversifying rather than consolidating, which makes it difficult for IT leaders and architects to get a grip on complexity and implement a future-proof architecture. There's an emerging category of platforms that offer an integrated set of IoT software services that simplify device management, communication, data management, analytics, and security, thus abstracting technologies away from developers.

Similarly, high productivity platforms for apps running on top of IoT Software Platforms offer a complete set of application lifecycle management services that speed IoT app development and hide complexity so that less technical developers can build sophisticated solutions.

The best way to discover the value of IoT is to start experimenting in close collaboration between business and IT. Involvement of Enterprise Architects is critical to ensure that solutions are secure, can scale and follow key architectural principles.