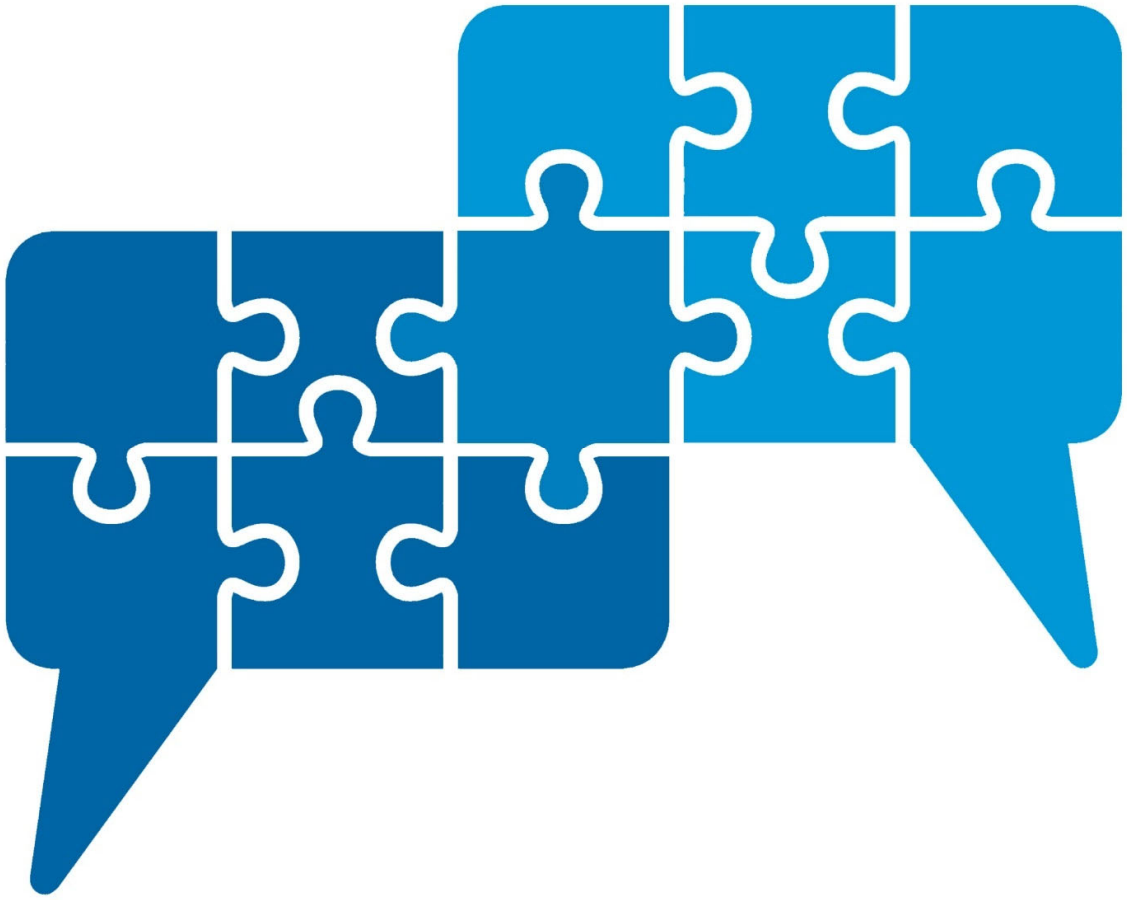
IoT Alliance Australia

Interoperability by Design

A set of Design Principles for enabling IoT success through interoperability



Published by the Internet of Things Alliance Australia
V0.7 - 25 July 2022

Authors: Dale Rankine, Bill Wood, Freddie Coertze, Dean Dobson, Aaron Watkins
Foreword by Frank Zeichner

The editorial team wish to thank the many contributors who have generously assisted in the production of this document through comment and review.

Contents

Foreword

IoT Alliance Australia (IoTAA) is the peak industry body representing the Internet of Things (IoT) in Australia. We see a thriving future for Australia and the world by connecting data, devices, people, processes and things to the Internet. It helps people make better and more informed decisions to get the best possible outcomes and ultimately helps boost Australia's future success, productivity, competitiveness, jobs, inclusiveness and the economy.

Our mission is to accelerate the adoption of IoT for Australia's economic and societal benefit. To achieve this, it is essential that we facilitate IoT innovation and adoption by activating and supporting collaboration across industry, government, research and communities. Fundamental to this is one of our overarching Principles and Aims, to encourage industry interoperability and seek collaboration based on secure and open IoT solutions.

For those wishing to benefit from the great promise of IoT-based solutions, the term "interoperability" is often code for "just make it simple". Indeed simple, easy to use IoT applications are essential for user adoption. However, such simplicity can also be difficult to realise in practice - IoT-based systems and applications are often built from lots of piece parts, perhaps never designed with a particular end in mind.

Our membership is dedicated to changing this, and this document represents a first and important step in that direction. The benefits to be realised through greater interoperability across the entire IoT landscape are enormous. Interoperability is the key to scale, lower costs and greater competition. Without it, industry, government and end users cannot hope to benefit from the robust, scalable and IoT networks required to address many of our challenges.

We hope that you will find this document to be valuable in providing a starting point for an exploration of IoT interoperability as it may apply in your context – be that a vendor wishing to benefit from a dynamic and rapidly growing IoT ecosystem, or someone with the responsibility of putting IoT technology to work to address applications in smart cities, public utilities, or enterprises.

Frank Zeichner
CEO, IoT Alliance Australia

Introduction

Purpose of the document

This document represents a body of work by members of the Interoperability Workstream of the IoT Alliance Australia. It is our hope that it provides a starting point for a broader discussion and consideration of interoperability as it applies to IoT.

Our work is intended primarily for those setting out to architect and build IoT-based components and systems – be those end devices through to whole solutions and applications, and everything in between. It is not assumptive of any particular application domain, but general in nature.

We start by presenting a context for the consideration of IoT interoperability, and then laying out a rationale for pursuing such – specifically the benefits brought by interoperability across its many aspects.

There is much to be considered as one sets out on such a journey, and we acknowledge such considerations against your business needs and environment.

Moving into design and engineering, interoperability is a multi-dimensional challenge, and we present several dimensions of that challenge. Finally, we present what we believe to be the core design principles to be employed.

Intended audience

The intended audience of the document is in the first instance the membership of the IoTAA, and in particular during this formative stage of the broader interoperability discussion, IoTAA workgroups. It should help to frame and provide the various IoTAA workgroups with a language for a discussion around interoperability, providing a guideline and starting point for a deeper dive within the relevant domain. In doing so it should assist in promoting such, particularly amongst those workgroups which are more "vertical" or industry segment focussed

For individual member companies, government and industry generally, particularly the "buyer" side, it is intended to arm them with a better understanding of the importance of and benefits which can flow from interoperability. That in turn can afford them with a body of knowledge and help advance a suite of questions to be directed toward suppliers, and provide a basis for assessing the interoperability dimension during vendor selection. For the management and non-technical staff of SMEs for example, it aims to explain in "simple" terms what to look out for when taking on a journey toward digital transformation. For Industrial Automation organisations it should assist operational technology (OT) engineers in understanding considerations around industrial IoT deployments.

For large organisations, the document should help in fostering a discussion which might assist in converging various "solution islands" toward a larger and extensible outcome for those organisations.

For suppliers, it should help in the design and development process, providing a framework for working through the specification, design and adaptation of products and services, both hardware and software, ensuring they are as open and extensible as commercially warranted. It will assist in clearly articulating the extent of interoperability aspects of their products and services.

Dale Rankine
CEO & Founder, Reekoh
Co-Chair - Interoperability Workstream, IoT Alliance Australia

Bill Wood
CEO, Arid Systems
Co-Chair - Interoperability Workstream, IoT Alliance Australia

A context for interoperability

Every day and practical IoT-based solutions are often constructed from many individual building blocks, integrated in some manner and intended to work together harmoniously. Such building blocks may comprise sensing and control devices, the data generated by devices, networks and data repositories used for the storage and transmission of IoT data, functional middleware components, data analysis and presentation frameworks, and related applications.

The IoT Reference Framework[1] document from the IoT Alliance Australia explores the IoT landscape through a layered view, encompasses these building blocks, and serves as a useful companion to this document.

Typically, the creation of IoT solutions involves the integration of several of the aforementioned building blocks, often sourced across multiple vendors. Interoperability refers to the basic ability of such building blocks to readily connect and exchange information with one another, without having been specifically designed to work together. Native interoperability is when these components are able to interoperate without specific effort and is often facilitated through standards.

Integration is the effort expended in order to achieve interoperability when devices and applications are not natively interoperable. In other words, integration often requiring customisation and the development of additional functionality specific to the requirements of the application itself.

The importance of interoperability is accentuated when building blocks have not been specifically designed to natively integrate with each other, but are required to do so. The ease of performing such integration and the resultant maintainability of the whole solution is a critically important consideration.

In the IoT context, interoperability is the ability of IoT-based building blocks, systems, services and applications to seamlessly interconnect, integrate and interact with each other as well as the non-IoT elements of a solution.

This document highlights the importance of interoperability and provides some insight into principles and considerations in achieving such. To provide some overall context, the core tenets of interoperability encompass the following.

Interconnection of components

– the manner by which system components are able to physically interconnect with each other – be that through hardware or software interfaces or networks of such. Interconnections will have physical and protocol aspects including wired and wireless interfaces, and embody protocols such as HTTPS, MQTT, UDP, CoAP, the various 5G protocol layers and so on.

Data exchange

- the process of actually exchanging information between interconnected components. An IoT solution may employ multiple data acquisition endpoints utilising different physical and protocol means, but all ultimately sending data to the same platform for analytics/AI/ML processing.

Information meaning

– the semantics of information exchanged between system components. In other words, the ability of the

[1] IoT Alliance Australia, Cyber Security Workstream - IoT Reference Framework https://www.iot.org.au/wp/wp-content/uploads/2016/12/IoT-Reference-Framework-v1.0.pdf

recipient of the information to interpret the data it has received from another component. The information may relate to elements or entities (objects and documents) and their format (text, voice, video and so on). The information may also comprise metadata.

Management and control – the overall coordination between system components such that they work in a coordinated and holistic manner and accomplish the overall result required of the solution.

Containment and avoidance of unintended consequences – the manner by which components may regulate interaction between each other so as to enable the degree of collaboration required, but manage and limit that interaction to achieve the necessary degree of security and system integrity.

Provenance and observability - information provenance explains the origin of information, how and why was it created. Information observability refers to the monitoring of the information flow/exchange with a view to detect and resolve any information issues.

Why Interoperability? The Benefits

With the rise of the digital era with connected "things", data became the main driver for our applications, resulting in a need for devices to share data more openly and easily. With the new, unique challenges this introduced, we quickly realised that not all applications can be solved by using new technology, particularly as we needed to interface with older legacy systems / architectures and different devices from different vendors.

There is also an apprehension around the possibility of vendors locking you in to a vendor specific device or protocol. Gartner identified Interoperability as one of the top three challenges preventing IoT from reaching its full potential.

Interoperability is the ability that allows for the unrestricted sharing of resources between different systems. This can refer to the ability to share data between different components or machines, both via software and hardware, or it can be defined as the exchange of information and resources between different computers through local area networks (LANs) or wide area networks (WANs). In short, interoperability is the ability of two or more components or systems to exchange information and to use the information that has been exchanged.

Here is some of the main challenges that interoperability solves:

1. Lower operational cost

Interoperability allows systems to be optimized for different functions, which means less time and manpower that needs to be spent on transferring data between machines. Interoperability also means managers and operators aren't constrained to a specific vendor, and can purchase products or services from less expensive sources. Interoperability facilitates the interworking of point solutions, also providing longer lifespan for building blocks and solutions through re-use and adaptation.

2. Easier data transfer and sharing

When using controls and products from different manufacturers, data extraction and transfer between them can be difficult. Interoperability means that data doesn't need to be moved around manually between independent products. This allows the data to be shared more easily and more efficiently between machines and across the IoT digital value chain. The ease with which this can be achieved enables the creation of new systems out of the connections between (usually disparate) smaller systems, enabling functionality not easily possible from the smaller systems alone.

3. Flexibility

The biggest benefit to implementing interoperability is the flexibility it gives operators and managers. Interoperability allows for hardware and software to be used interchangeably as long as they support each other. It also allows for transportability if a facility moves location or needs to be adjusted or remodelled. Operators and managers are no longer tied to one specific manufacturer for their plant equipment, which can also save money in the long run.

4. Scale

The ability to interconnect IoT systems, the exchange of data in particular, is the key to creation of ever larger and more valuable solutions.

5. Longer solution lifespans

Component interchangeability and greater integration flexibility can in turn lead to longer solution lifespans. The ability to more easily adapt existing solutions to changing requirements and system component end-of-life circumstances can extend the working life of a solution.

Design Considerations

Consider your needs and environment

As there are many design considerations to explore for interoperability, these need to be considered in the context of your business needs and the environment your systems exist in. The design considerations should be ranked in order of importance when mapped against the business needs and the constraints that exist in your environment, as not all considerations are equal in importance.

The 'abilities'

Scalability. An important design consideration is ensuring that the maximum transaction load between interconnected parts can be catered for within service levels expected of the overall solution. Each interconnected component of a system or application must service requests from the components with which it interacts. Ensuring that the demands of those adjoining components can be met, and that the solution is able to scale appropriately to meet overall system service levels is essential. Achieving such requires appropriate performance levels of individual components, along with the overall solution being well-architected.

Maintainability. Sustaining the operation of a solution can require monitoring, proactive identification of future issues, and the ability to update configurations and architecture without interruption of service.

Flexibility & Extensibility. IoT-based systems should be able to evolve as the need requires. The level of openness, ability to make system and component changes and the ability to extend the functionality of the system needs to be considered in system and component design.

Usability. The functionality achieved through interoperating components of a system must be fit for purpose and aligned with the business requirements.

Availability. The overall availability of integrated systems decreases when the complexity of the systems increases, and the overall availability of an integrated system is lower than the least reliable interconnected component[2]. When transaction scope encompasses several interconnected systems, the integrity of a transaction needs to be maintained and requires significant extra complexity to be built into the integration architecture so that transactions can be rolled back across multiple interconnected systems during failure of one of the systems.

Auditability. With a view to enhancing transparency and trust, auditability refers to the ability to inspect and trace who did it what and when to resolve any conflicts and to meet information audit and compliance requirements.

Reliability. The dependability of a solution to continue to function as intended under a range of operating and environmental conditions will be a function of many of the aforementioned, and must be fit for purpose.

Security

An IoT-based system is only as secure as its weakest link. Avoidance of unauthorised use and hacking mandates appropriate security be in place at all levels of a system. Designing security into components at the outset is essential, rather than a retrofit or rework after a security breach. Given the natural tension between the ease and security of an interconnection between system components, there is often a tendency to avoid implementation of achievable security.

[2] http://www.edgeblog.net/2007/in-search-of-five-9s/

The use of standards is critical in enabling interoperability as it relates to security.

Privacy

User privacy is becoming important with IoT systems since personal information will often be delivered and shared among connected things. Mechanisms are required to protect personal data and its flow across systems, and should be considered within the context of interoperability.

Design for expected service levels

Given business requirements and the system technical environment an integration is operating within, the infrastructure to support interoperability needs to be built so that it is fit for purpose and within budget constraints. For instance, a system designed for interoperability that is expected to support real-time monitoring of a critical health care function, Type 1 Diabetes Blood Glucose levels for example, will require considerable technical infrastructure (such as proactive monitoring) so as to ensure the high expectations for service levels are met, likely at a high price point. Likewise, a non-critical service utilised on an occasional basis can be made accessible relatively quickly and easily if the right tools are used.

Use existing recognised standards and frameworks

When considering the interfaces and interactions between interconnected systems, utilise existing standards and frameworks as a starting point at least where possible since others with similar interoperability needs are likely to have previously pursued the best approaches to interoperability. Look for open industry standards, rather than proprietary "standards" which can create lock-in, reducing flexibility and increasing cost.

Utilising an open standard or framework allows you to leverage the experiences and efforts of others (for example, via an SDK), accelerating your interoperability journey, as well as making the integration with third party systems considerably easier. It also enables your customers to derive value from your products for longer, considering that the need to replace old technology will be less if it is standards compliant.

Industry specific standards and frameworks are even more useful than generic high-level standards and frameworks, as the interoperability services are more relevant contextually, and usually richer in functionality. Examples include for instance

- the System Interoperability Framework (SIF) for the education sector;
- HL7 for the healthcare sector;
- Haystack for the standardisation of semantic data models and web services for homes, buildings, factories, and cities.
- Sparkplug B, a specification building upon the MQTT messaging protocol.

When selecting a standard or framework, look for:

- Open standards (proprietary standards only if there are limited / inappropriate open standards available).
- Adoption. Standards / frameworks that are widely adopted are likely to be best in class and widely supported, with skills of these standards / frameworks available in the employment marketplace.
- Documentation and SDKs. Ensure that the standard / framework chosen has adequate documentation and tools available to help with the implementation.
- Currency. Adoption of recent standards that are well supported with an active community ensures a good future for the standard / framework. The converse is that adoption of a standard / framework that is too new will mean limited support and incomplete specifications and tools available.
- Fit. Ensure the features of the standard / framework are a good match to your business needs now and for the future.

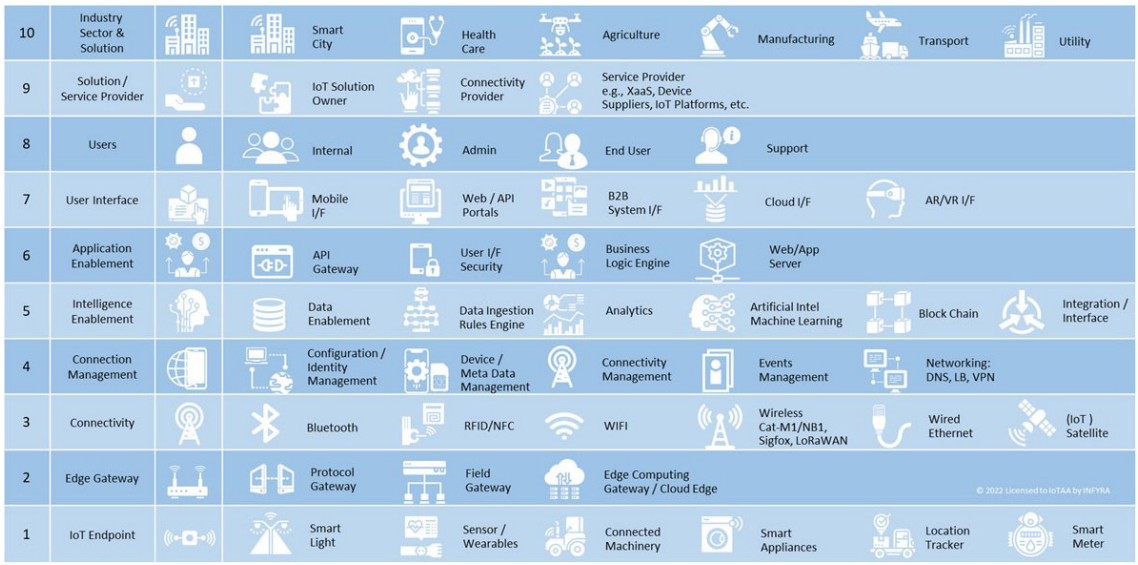
Dimensions of interoperability

The IoT Reference Framework

For IoT-based applications, interoperability is a multi-dimensional challenge. While each IoT-based solution is different, the IoT Reference Framework from the IoT Alliance Australia provides a useful lens through which to consider interoperability.

The IoT Reference Framework is applicable to any IoT solution architecture, and takes into consideration end-to-end solution aspects, including IoT endpoints, connectivity, network, user, platform, application, solution and ecosystem. It serves the following purposes. It

- provides a 'common language' to facilitate discussions amongst the IoT community;
- provides factual, vendor-neutral information on end-to-end IoT building blocks; and
- assists organisations to clearly and simply document and articulate their IoT solution requirements.



IoT Reference Framework v1.2 – IoT Alliance Australia

Connecting the pieces

Network connectivity is the most basic level of interoperability, be that at endpoints implementing different data acquisition technologies perhaps, or the interconnection of subsystems. For example, a solution may consist of LoRaWAN, LTE, LTE Cat M1, NB-IoT, Bluetooth, or even RFID endpoints, all collecting data in different 'radio environments', but ultimately all sending data to the same data platform for processing. In industrial cases, SCADA and the use of OPC Unified Architecture (UA) among others are at play, providing a multitude of connectivity options

An orderly exchange of data

With network connectivity established between interconnected components and systems, data must be transacted in an orderly manner. A higher-level application protocol is required to provide an orderly transfer of the data itself. Software architectures often include the three most popular IoT protocols - REST (over HTTP/HTTPS), MQTT (a publish-subscribe protocol) and the Constrained Application

Protocol (CoAP) for devices with limited resources. Such protocols control data exchange ensuring that each party is in a position to receive data from the other party, is able to request data when required, can acknowledge receipt and so on.

The interpretation of data

The purpose of physically interconnecting system components via a network with appropriate protocols is, of course, the transference of data. However, that data is useless unless it can be interpreted and understood. There are two aspects to be considered.

1) Data serialisation. This is the "encoding" of data such that it adheres to a known format, analogous to language in human communication. When not proprietary, common standard serialisation methods include JSON, XML and CSV. For example, a number of variables pertaining to a temperature sensor in a transport application may include the current temperature, maximum and minimum recorded values, sensor location and so on. While these are often sent as key/value pairs, how those are formatted in a block of characters is a matter of serialisation.

2) Data meaning and semantics. This gets to the heart of being able to interpret and make use of exchanged data. In the human communication analogy, what do the words and phrases actually mean? Transformation of payload data may be required, so being able to decipher the semantics or meaning of the payload is critical. In our transport temperature sensor example, which variables must be present, which are optional? What are the units of measurement (degrees F or C)? How to account for sensor versions as applications evolve? A "schema" if often used to specify the expected structure of data. These are verbose but machine-readable specifications of the expected data structure. This becomes particularly important in a future scenario where AI is used to assist with IoT device auto-discovery, and auto-categorisation.

3) Meta data and schema standards. While an accompanying data schema is critical to interpreting any given data, the creation and evolution of standards around these schemas is becoming increasingly important. The transport industry for example is greatly aided if standards for temperature sensor data exist and are followed. A great deal of work is going into defining standard schemas for all manner of devices across a wide array of industry verticals, and creating public repositories of such standards.

The security, privacy and ability to trust the data

The implementation of secure end-to-end applications requires attention to security at all levels. It is critical that components brought together to interoperate as part of a system are able to do so in a protected manner, and in control of what is shared. The preservation of required privacy policies must also be maintained end-to-end. A breach at any level or interconnect may jeopardise maintenance of data privacy by the system as a whole.

For example, how does one component be assured that another is actually who it claims to be, and in so doing preserve trust in the data exchanged by users? In this context, authentication is used to validate identity. Having authenticated the identity of an interconnected device or system, the information made available to that component must be managed. The process of verifying and managing what is able to be shared is known as authorization.

Achieving Interoperability - Core Design Principles

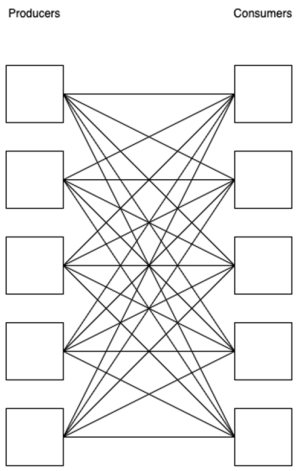
The best way to circumvent integration challenges is to prepare your IoT devices, networks and platforms for interoperability from the start. The following design principles represent elements that permit interoperability to be baked in from the start.

1. Use de-facto and formal standards where available and appropriate
2. Plan to be part of an open eco-system
3. Make it easy for users to interact with your system
4. Design with forwards compatibility in mind
5. Decouple your interface from your implementation

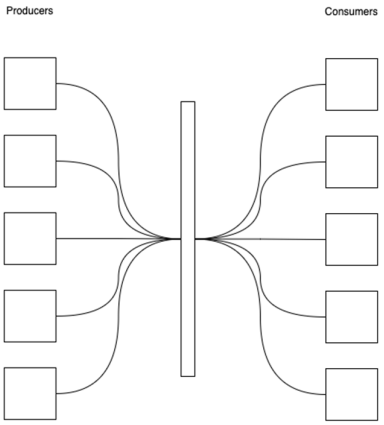
The following sections elaborates on these core principles.

Use de-facto and formal standards where available and appropriate

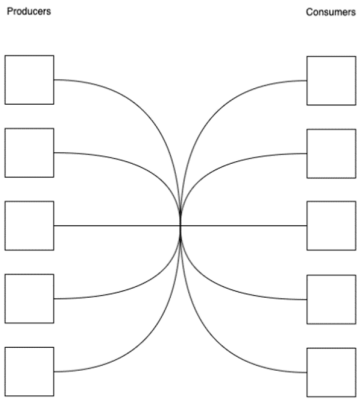
Standardisation provides a simple manner in which to gain the benefits of the network effect. This is true whether the standard used is an informal end-user standard, a commonly accepted format, for example Sparkplug B (an open source software specification for industrial IoT) or Haystack (an open source initiative to streamline working with IoT data), or a formal industry standard.



Consider an example situation where there are five different classes of assets producing data and five different consuming systems that wish to utilise that data. If there is no standardisation of the payload, data transformation is required for each producer -> consumer integration, resulting in a total of twenty-five different payload transformations.



However, where some form of standard is available, transformation is only required in changing from producer formats to the standard and from the standard to the consuming formats, resulting in a reduction to a total of ten different payload transformations.



When the producers are capable of producing data directly in the standardised format and the consumers are able to consume data directly from the standardised formats, we have a natively interoperable system and no transformation is required.

Whilst the example shown above considers data payload formats, it is important to realise that integration efforts exist at multiple levels – transmission protocols, networks, data formats, authentication methods and error handling approaches. Common implementation methods at each layer reduces overall inefficiencies and brings benefit to end users.

Plan to be part of an open eco-system

Plan to be part of an open eco-system rather than forcing your users to interact with your system in the way that you feel is best. Unexpected use of a system can lead to new innovations, which in turn can lead to increased use of a product.

As a supply-side participant, provide open and well documented points of integration where possible. Also ensure that these integration points utilise methods that are in common use (e.g. encrypt traffic using SSL, communicate via HTTPS). Providing such integration points encourages transparency and constancy and allows others to build reliable products, services or solutions with you as part of their value chain.

As an end user or consuming application, insist upon open and extensible components to form the building blocks of your solution. Doing so reduces the risk of backwards incompatibility from your vendors and allows you to benefit from a growing portfolio of competing products and services. Over time, such demands encourage cross vendor support and interoperability.

The more proprietary and closed off a system is, the higher the cost associated with utilising the data in a way differing to that originally envisaged by the system creator.

Make it easy for users to interact with your system

Reducing barriers to adoption (or abandonment) of your system is a critical step in the path to interoperability. As an end user, select products that are easy to work with. They should use existing networks, standardised payload formats as well as readily available encryption methods where they're available. Using existing components in the market helps to ensure stability as well as improving the availability of skilled professionals capable in the area and make it easier to change vendors with minimal impact.

Quality documentation is critical to reducing the effort associated with adopting a system, so it is important that considerable effort be expended in this area. It should be noted that when utilising existing components and technologies, there will likely be voluminous documentation available including best practices to adoption, thereby limiting required documentation to the specific areas of value creation associated with the system in question.

Design with forwards compatibility in mind

As a producer, it is easy to be focused on the value you wish to add right now. However, give some thought as to the potential directions your system may evolve over time. Will early adopters be able to benefit from such changes or will they encounter backwards incompatible changes? Some example approaches that can be considered:

- Modular hardware capable of being switched and upgraded independently
- Including version numbers inside APIs or only adding fields and never removing
- Adopting some form of standard within your design where possible, as these are less likely to evolve in an incompatible manner due to wider implementation

Decouple your interface from your implementation

We have previously mentioned how reducing barriers to adoption is a critical step in the path to interoperability, as is designing with forwards compatibility in mind. A simple (to express) design principle that aids in both of these is to ensure that anything that is an internal implementation decision of your system remains internal and is not exposed to the customer.

If you use a particular network technology, a particular programming language or a particular database model, do not expose that directly as your interface. All of these change over time (although perhaps at different paces) and can represent both barriers to adoption (what if that customer does not use Java) or barriers to future change and innovation (how many customer implementations will break because your database schema changed).

A well-defined abstraction layer requires some effort up front and effort to maintain, but will reward all involved.

Glossary

TERM	MEANING
AI	Artificial Intelligence
API	Application Programming Interface, a type of software interface enabling computer programs to communicate with each other
CoAP	Constrained Application Protocol, a specialized Internet application protocol for constrained devices
CSV	Comma Separated Variables, a file format and extension
HTTPS	Hypertext Transfer Protocol Secure, an extension of the Hypertext Transfer Protocol (HTTP) used for secure communication over a computer network
IoT	Internet of Things
IoTAA	IoT Alliance Australia, the peak industry body representing the Internet of Things (IoT) in Australia
JSON	JavaScript Object Notation, an open standard file and data interchange format
LAN	Local Area Network, a computer network that interconnects computers within a limited area
LoRaWAN	A low power wide area networking protocol (from Long Range Wide Area Network)
LTE Cat M1	Long-Term Evolution Cat M1, a low power wide area network cellular radio technology standard
ML	Machine Learning, a field of computer science
MQTT	MQ Telemetry Transport, a lightweight, publish-subscribe, machine to machine network protocol
NB-IoT	Narrowband Internet of Things
OT	Operational Technology, the practice of using hardware and software to control industrial equipment
REST	Representational State Transfer, a software architectural style that describes a uniform interface between decoupled components in a Client-Server architecture
RFID	Radio Frequency Identification, a means of identifying and tracking tags attached to objects using electromagnetic fields
SCADA	Supervisory Control and Data Acquisition, a control system architecture for high-level supervision of machines and processes
SDK	Software Development Kit, a collection of software development tools

SME	Small (and) Medium-sized Enterprise
SSL	Secure Sockets Layer, a standard security technology for establishing an encrypted link between a server and a client
UDP	User Datagram Protocol, a network communications method
WAN	Wide Area Network, a telecommunications network that extends over a large geographic area
XML	Extensible Markup Language, a markup language and file format for storing, transmitting, and reconstructing arbitrary data

Further Reading

Standards Australia

AS ISO/IEC 21823.1:2020
Internet of things (IoT) - Interoperability for internet of things systems, Part 1: Framework

AS ISO/IEC 21823.2:2021
Internet of things (IoT) – Interoperability for IoT systems – Part 2: Transport interoperability

International Organisation for Standardisation/International Electrotechnical Commission

ISO/IEC 21823-3:2021
Internet of Things (IoT) - Interoperability for IoT systems - Part 3: Semantic interoperability

ISO/IEC 21823-4:2022
Internet of Things (IoT) - Interoperability for IoT systems - Part 4: Syntactic interoperability

ISO/IEC 30161-1
Internet of things (IoT) – Data exchange platform for IoT services – Part 1: General requirements and architecture

Industry IoT Consortium
The Industrial Internet of Things Connectivity Framework
https://www.iiconsortium.org/wp-content/uploads/sites/2/2022/06/IIoT-Connectivity-Framework-2022-06-08.pdf