.EXPLAINED

DDS & OPC-UA

Angelo Corsaro, PhD

CTO, ADLINK Tech. Inc.

Co-Chair, OMG DDS-SIG

angelo.corsaro@adlinktech.com

Motivations

Appeared in AutomationWorld
on the 23rd of July 2015

http://bit.ly/1Mzk4PV

Internet of Things' Organizational Confusion

As industry begins to better understand the Internet of Things, there remains some confusion about the role of industry organizations supporting the concept and how they relate to each other.

Speaking of OPC UA, two technology areas creating some of the greatest confusion in industry around IoT are DDS (Data Distribution Service)—often referenced by IIC—and OPC UA. The confusion surrounding these technologies stems from the fact that both are promoted as protocols enabling interoperability between devices, machines, and systems. According to [redacted], "DDS offers deterministic communication and is therefore comparable to Profinet or EtherCAT. The aim is fast data exchange within the systems. OPC focuses on interoperability—the exchange between systems. Above all, OPC UA offers security and configurable access control to interfaces and data. This is crucial for machine services."

The confusion surrounding these technologies stems from the fact that both are promoted as protocols enabling interoperability between devices, machines, and systems.

He adds that IIC members such a General Electric, Cisco, Microsoft, Oracle, and Siemens are "keen to use OPC UA" and that he is "not aware of any controllers or field devices in the automation sector that have implemented DDS" to date. However, discussions are currently underway to connect the two technologies. A meeting was recently held in Berlin "to clarify whether OPC UA should be recognized via the DDS transport layer," Hoppe says. Even though the market hasn't yet requested this connection, "experts are working on a gateway between OPC UA and DDS like we did with Sercos and EtherCAT. After all, this makes sense in order to integrate any DDS devices with OPC UA into the worldwide IoT community."

Genesis

DDS Genesis

INFORMATION SUPERIORITY

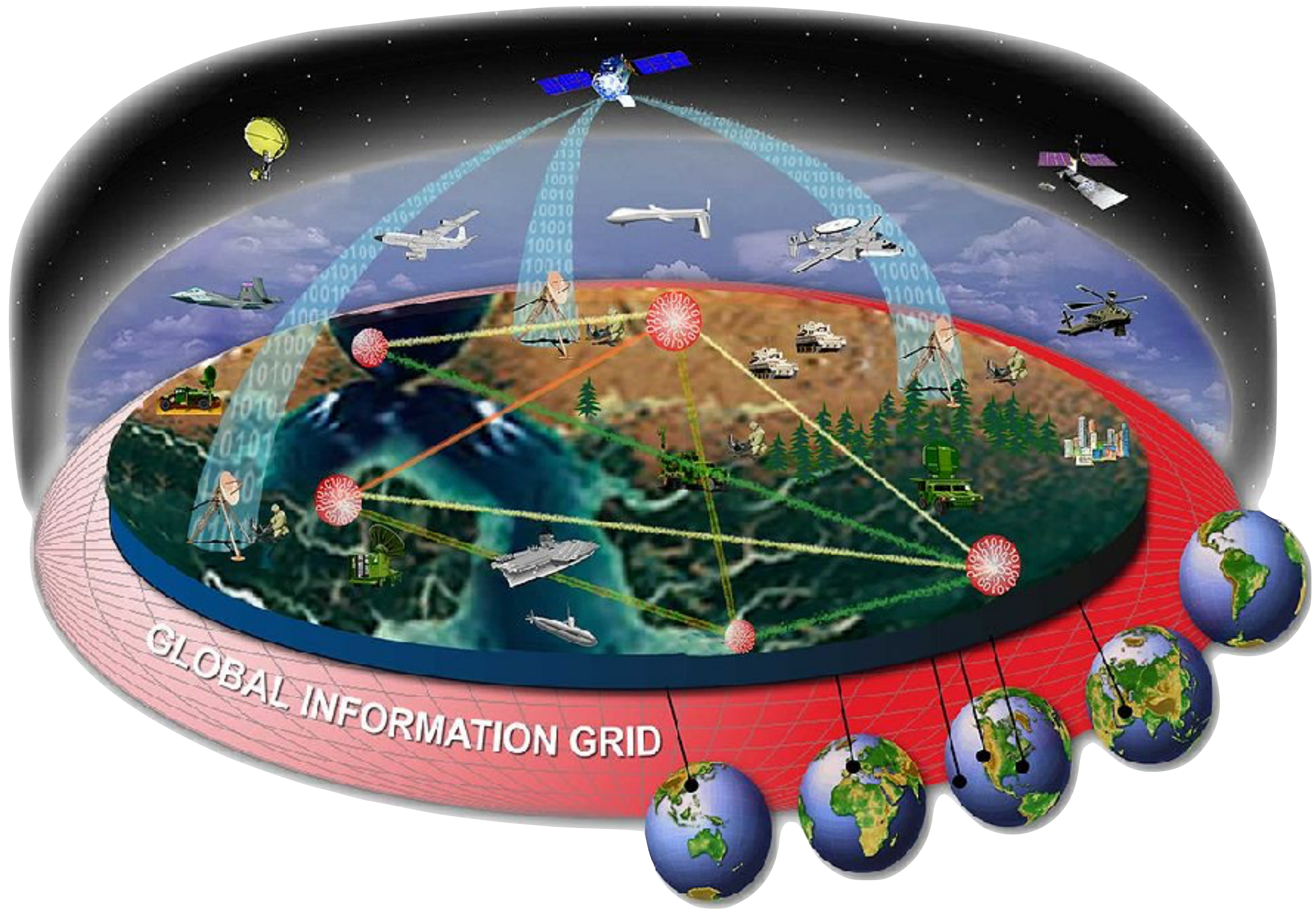
In the late '90s the DoD raised attention on the role that **Information Superiority** will play in the future of nations.

A series of operational and technological initiatives were undertaken to enable Information Superiority.

The concept of the **Global Information Grid** (GIG) was popularised.

THE GLOBAL INFORMATION GRID

A globally interconnected, end-to-end set of information capabilities for collecting, processing, storing, disseminating, and managing information on demand to warfighters, policy makers, and support personnel

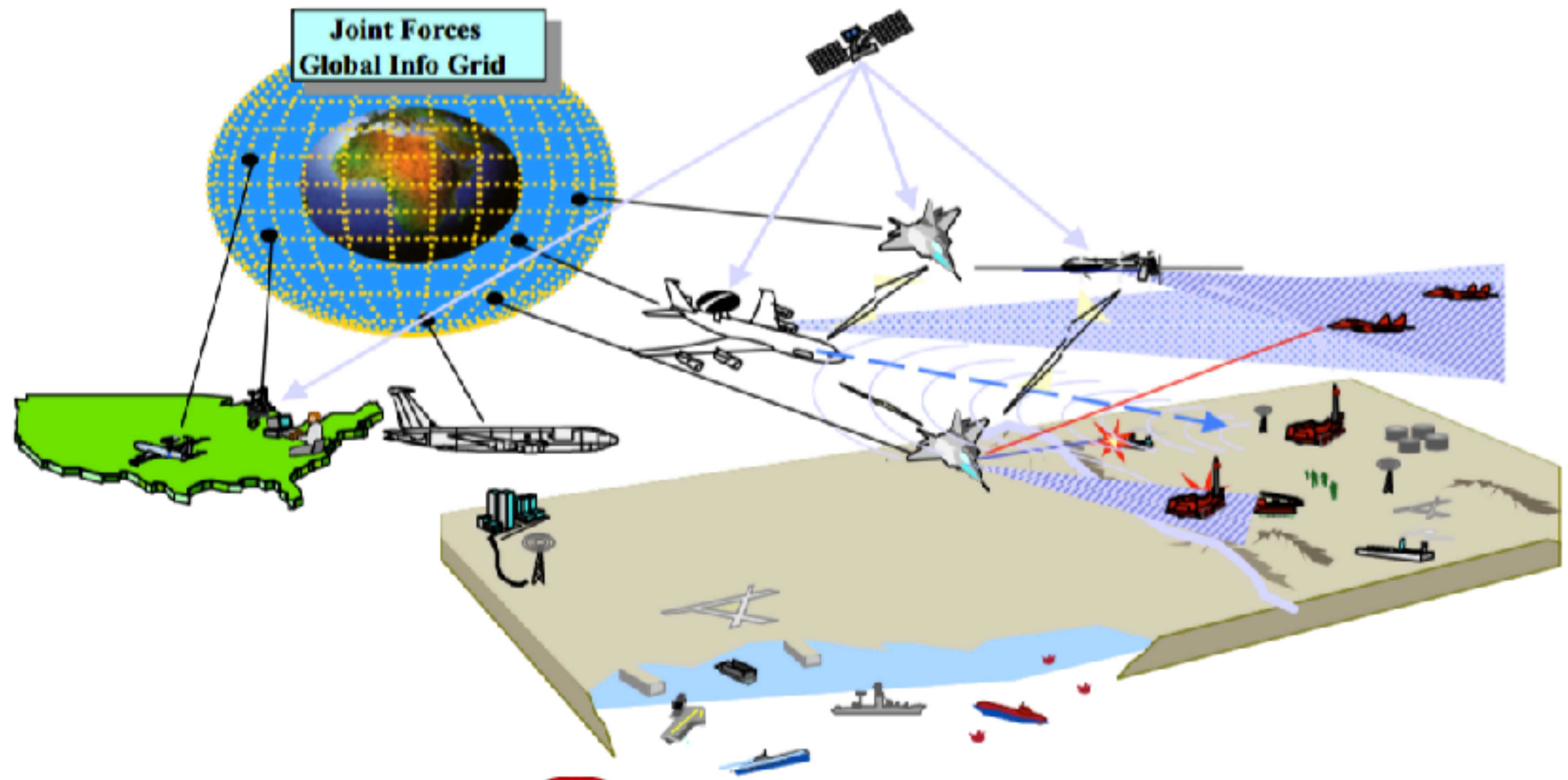


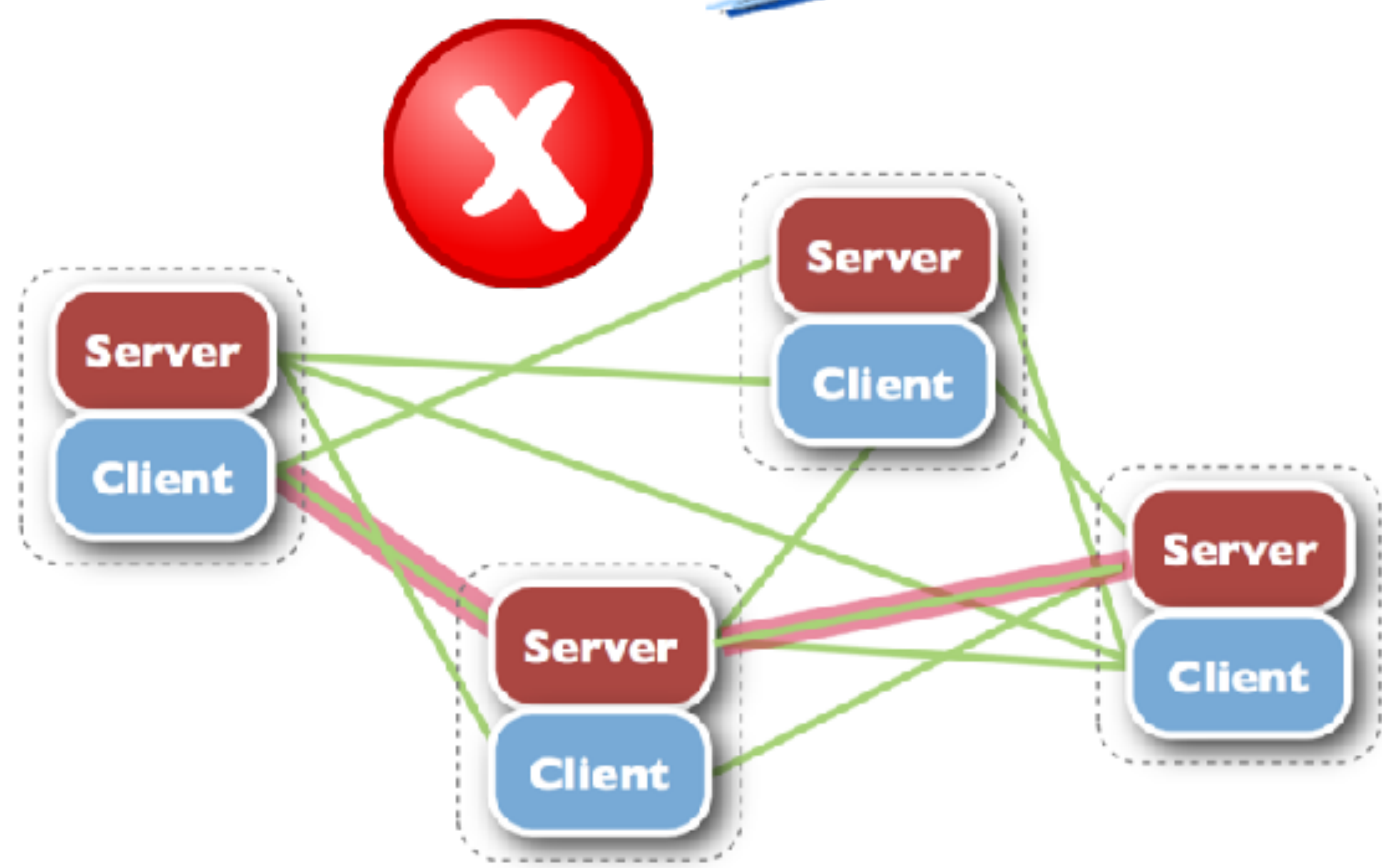
"The right data at the right place at the right time"

[Image from https://en.wikipedia.org/wiki/Global_Information_Grid]

MAKING THE GIG: CHALLENGES

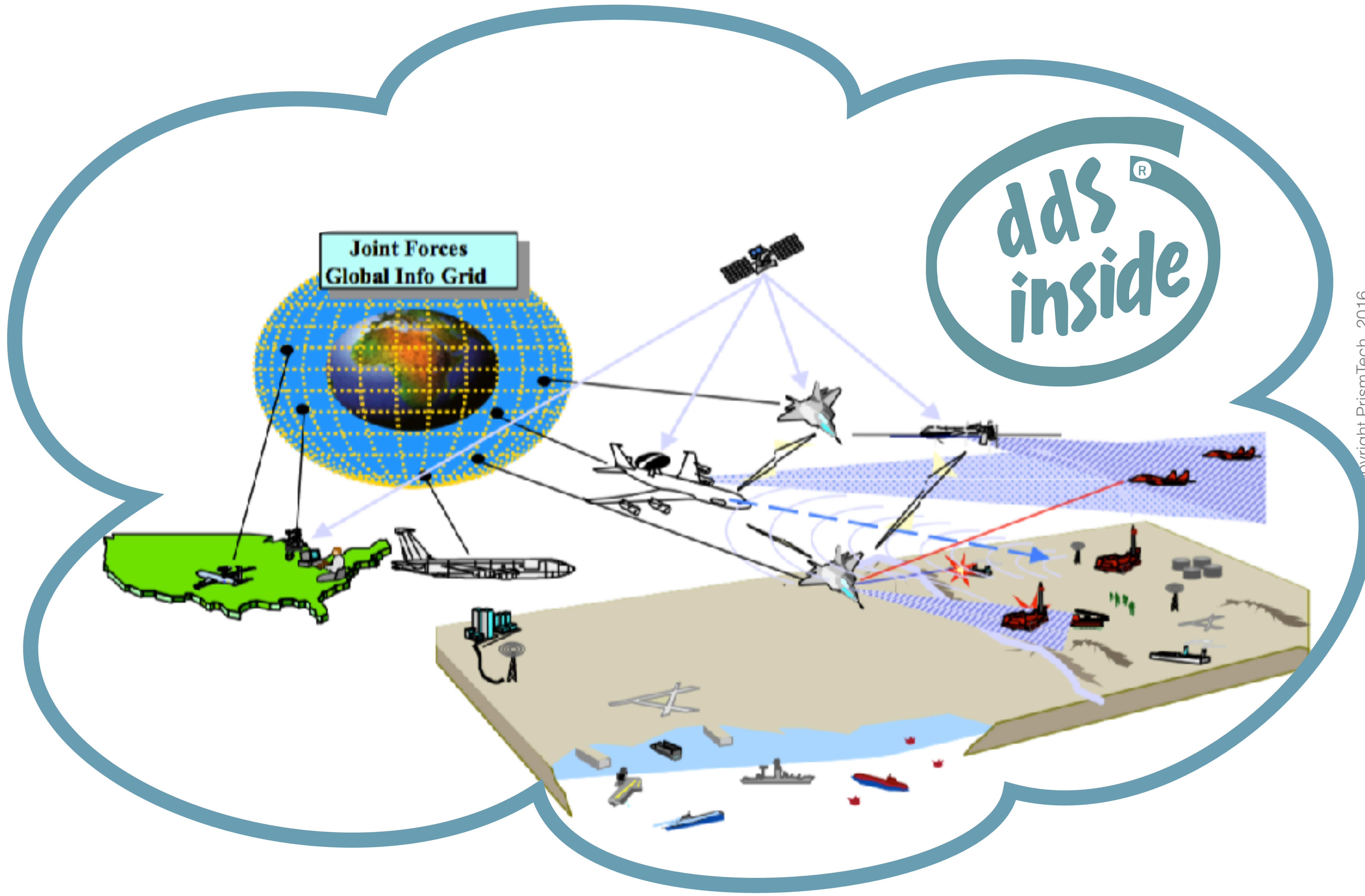
Client/Server technologies popular in the 90's such as CORBA, COM+/DCOM were not suited to implementing the GiG due to the tight coupling, high sensibility to faults, scalability and performance issues





ENABLING THE GIG: DDS

The Data Distribution Service (DDS) was introduced to **overcome** the **limitations** of **existing technologies** and **address** the data sharing **requirements** of the **GiG** for the engagement, awareness and planning grid.



RECOMMENDATION

DDS has been endorsed and recommended worldwide as the technology at the foundation for network centric systems and GIG-like functionalities

MASSIVE DEPLOYMENTS IN DEFENCE & AEROSPACE


Integrated Modular Vetronics


Training & Simulation Systems


Naval Combat Systems

Air Traffic Control & Management


Unmanned Air Vehicles


Aerospace Applications

OPC Genesis

INTEGRATION NIGHTMARE

In the early ‘90s Automation Industry there was **no standard** for **interacting** with **control hardware** and **field devices**.

As a result client applications, such as HMI, had to embed drivers and protocols for all the devices they had to interact with.

It was the age of **Integration Nightmare.**

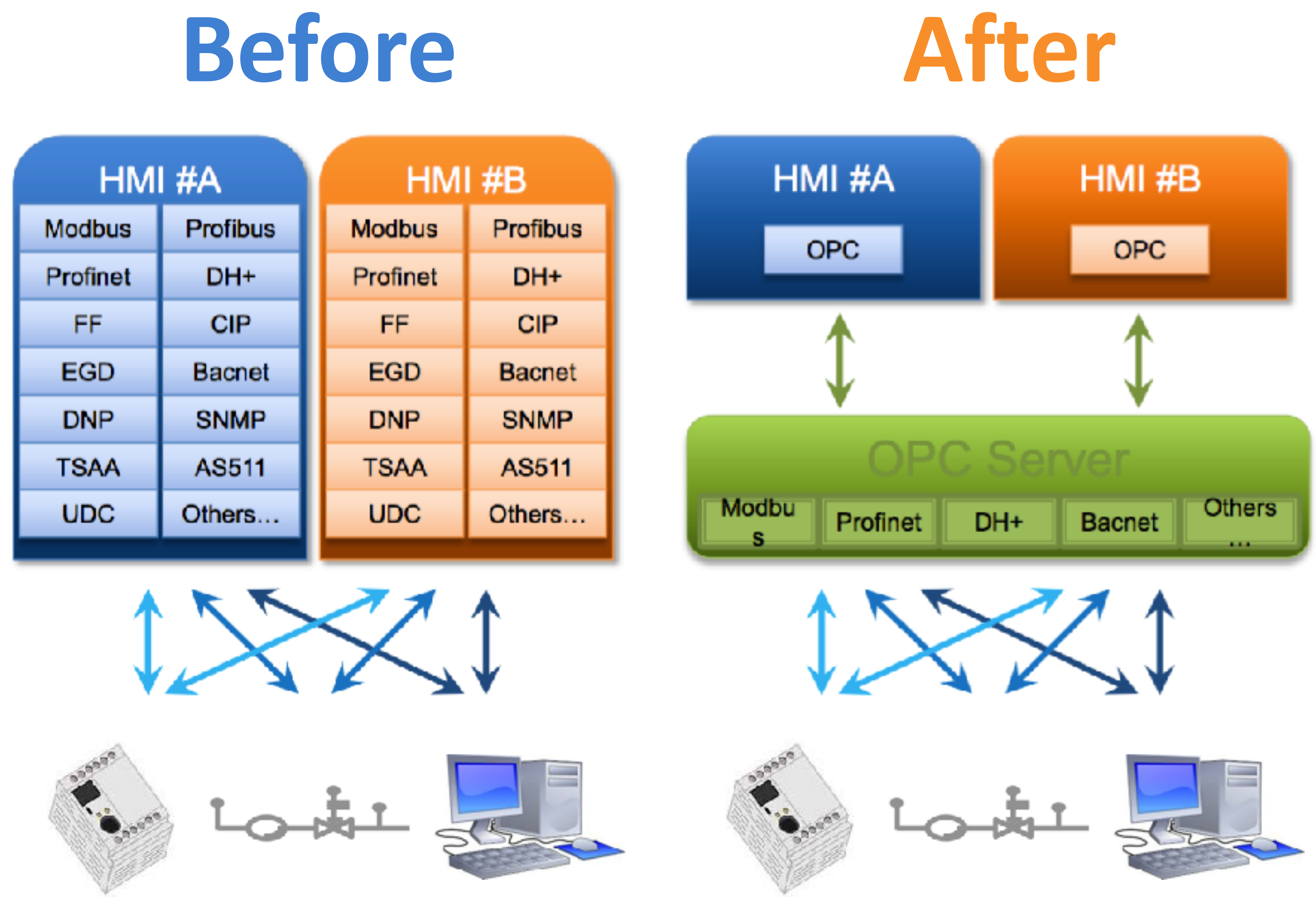
INDIRECTION

"All problems in computer science can be solved by another level of indirection, except of course for the problem of too many indirections"

—David Wheeler

OPC TO THE RESCUE

OPC (OLE for Process Control) was introduced in 1996 as a mean to **shield client applications** from the **details of the automation equipments** and providing standardised interfaces to interact with control hardware and field devices.



[Adapted from OPC Foundation Slides OPC UA Connectivity]

Standards & Evolution

DDS

DATA DISTRIBUTION SERVICE (DDS)

The DDS specification describes a Data-Centric Publish-Subscribe (DCPS) model for **distributed application communication and integration**. This specification defines both the Application Interfaces (APIs) and the Communication Semantics (behaviour and quality of service) that enable the efficient delivery of information from information producers to matching consumers.

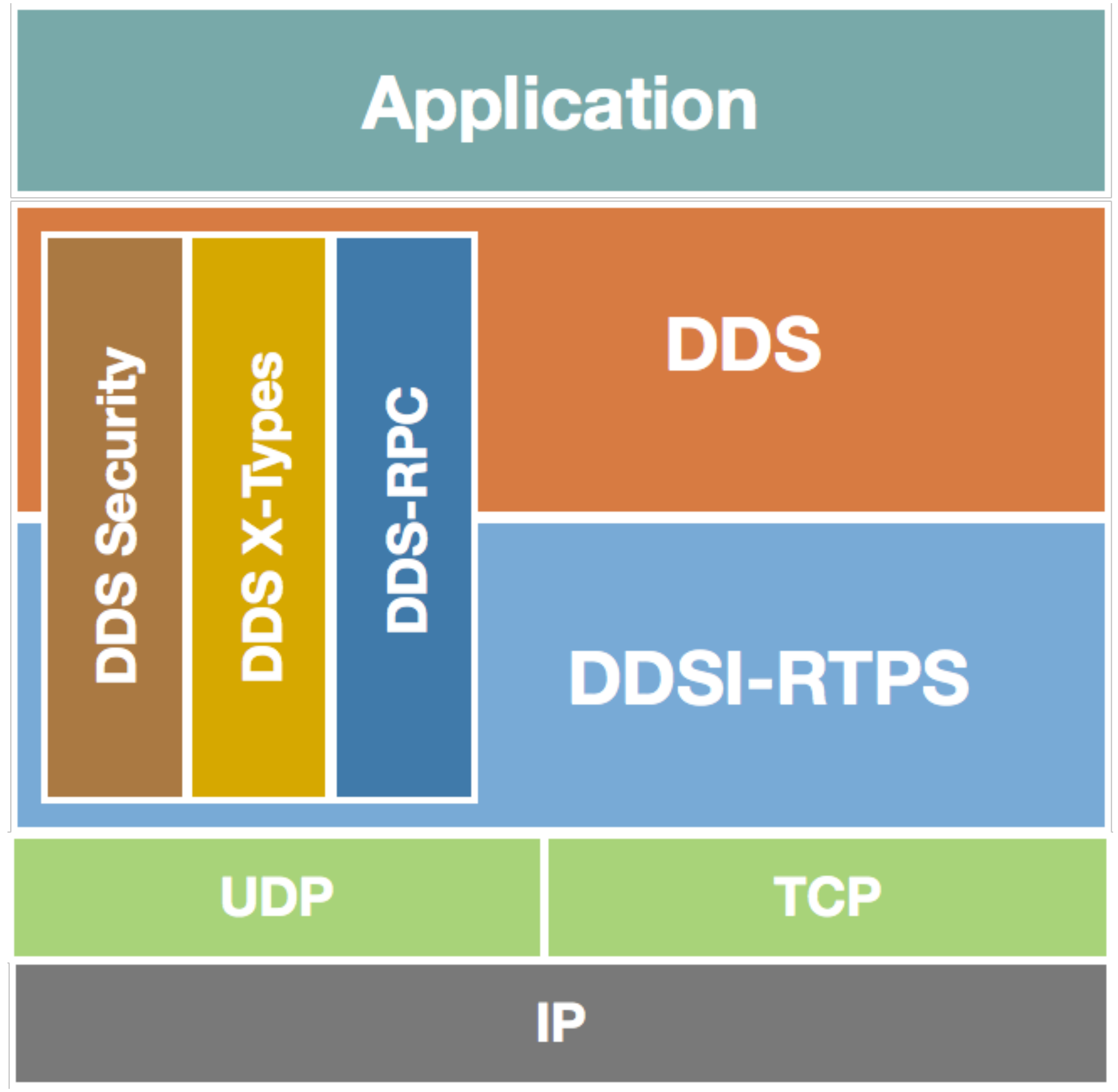
The purpose of the DDS specification can be summarised as enabling the **"Efficient and Robust Delivery of the Right Information to the Right Place at the Right Time"**.

[Extract from DDS Specification v1.4]

STANDARD STRUCTURE

DDS. Describes the semantics of the information sharing abstraction supported by DDS. Defines a nominal type system for describing DDS information models.

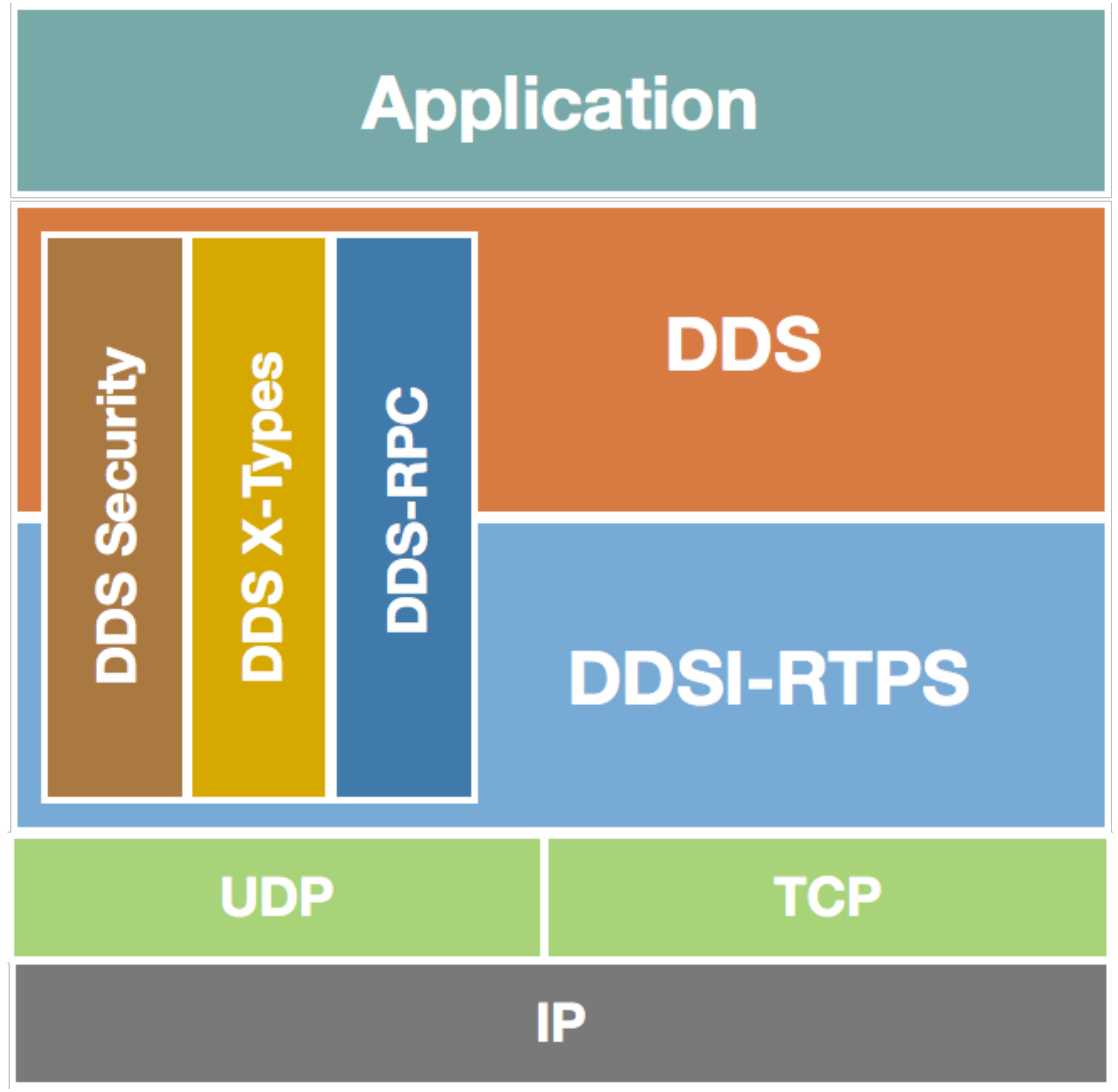
DDSI-RTPS. Defines a protocol for interoperable wire implementation of the DDS semantics.



STANDARD STRUCTURE

DDS-XTypes. Extends the DDS type system with support for structural typing as well as a dynamic type definition.

DDS-Security. Introduces information centric security in DDS for data in movement as as well as data at rest.

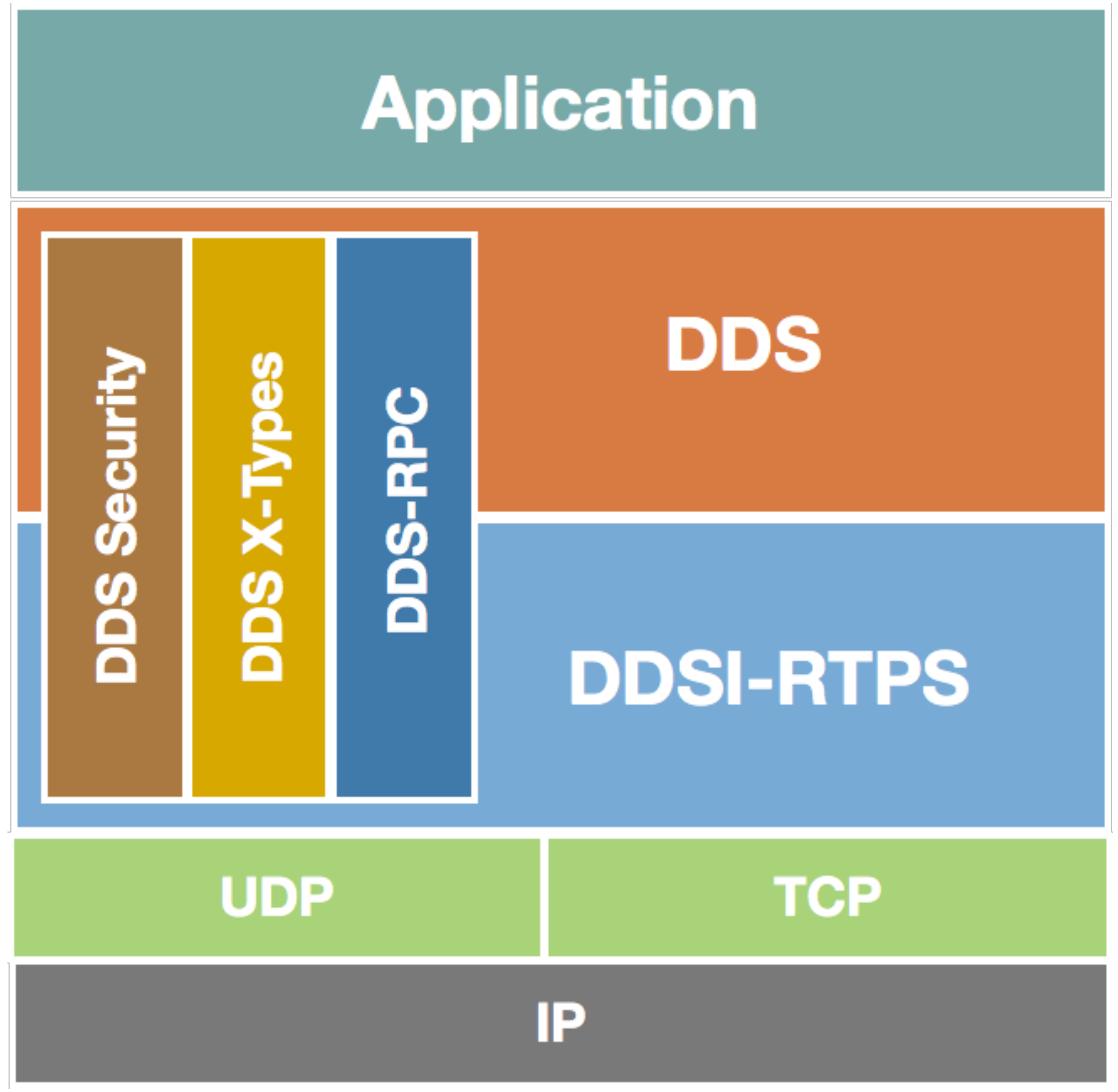


STANDARD STRUCTURE

DDS-RPC. Extends DDS with support for Remote Procedure Calls.

DDS-PSM-*. Defines highly ergonomic and optimised API mapping for specific programming languages instead of deriving those for the DDS-PSM-IDL

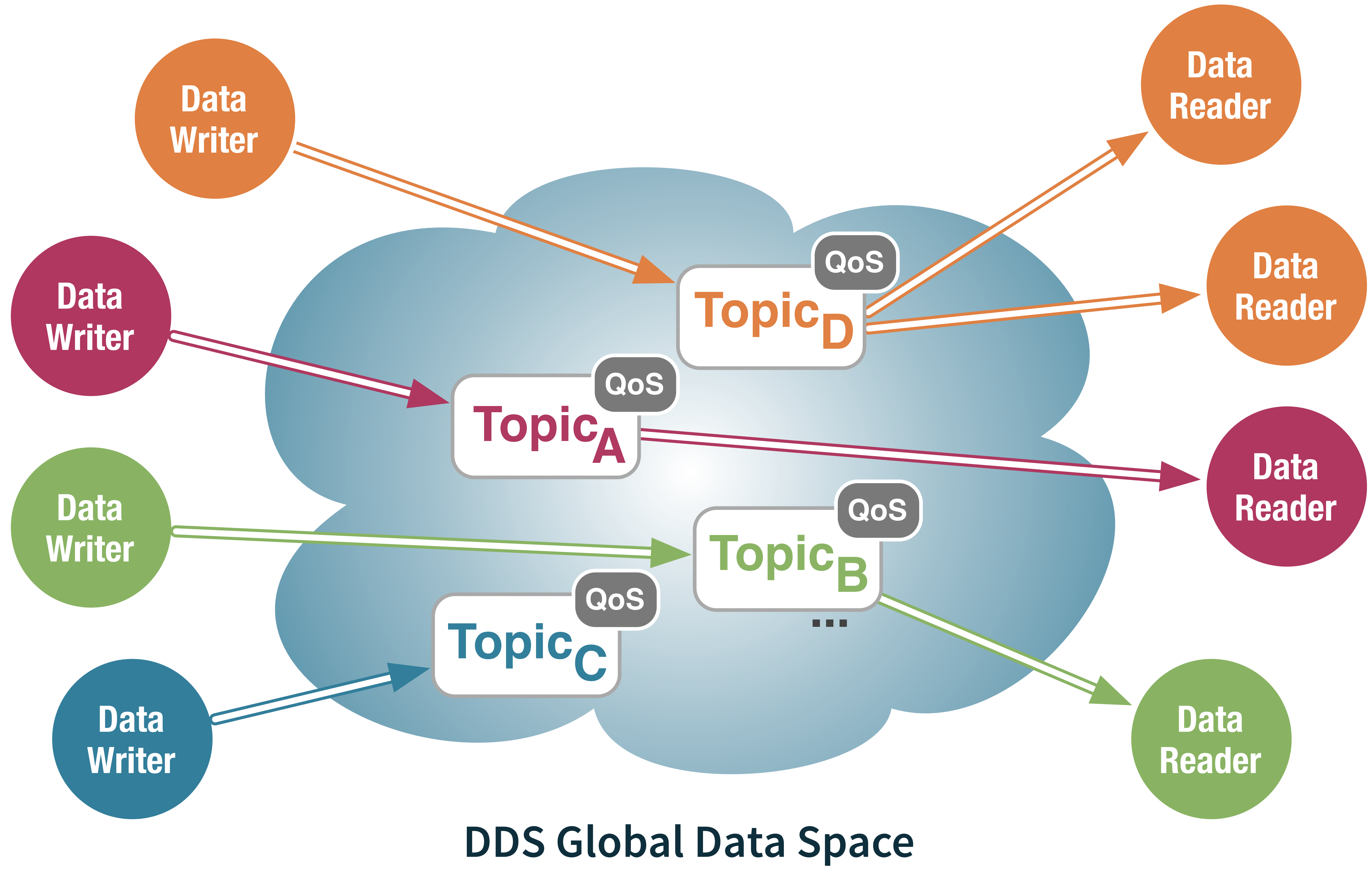
DLRL. Defines a language independent Object/Relational Mapping for DDS



DDS Abstractions

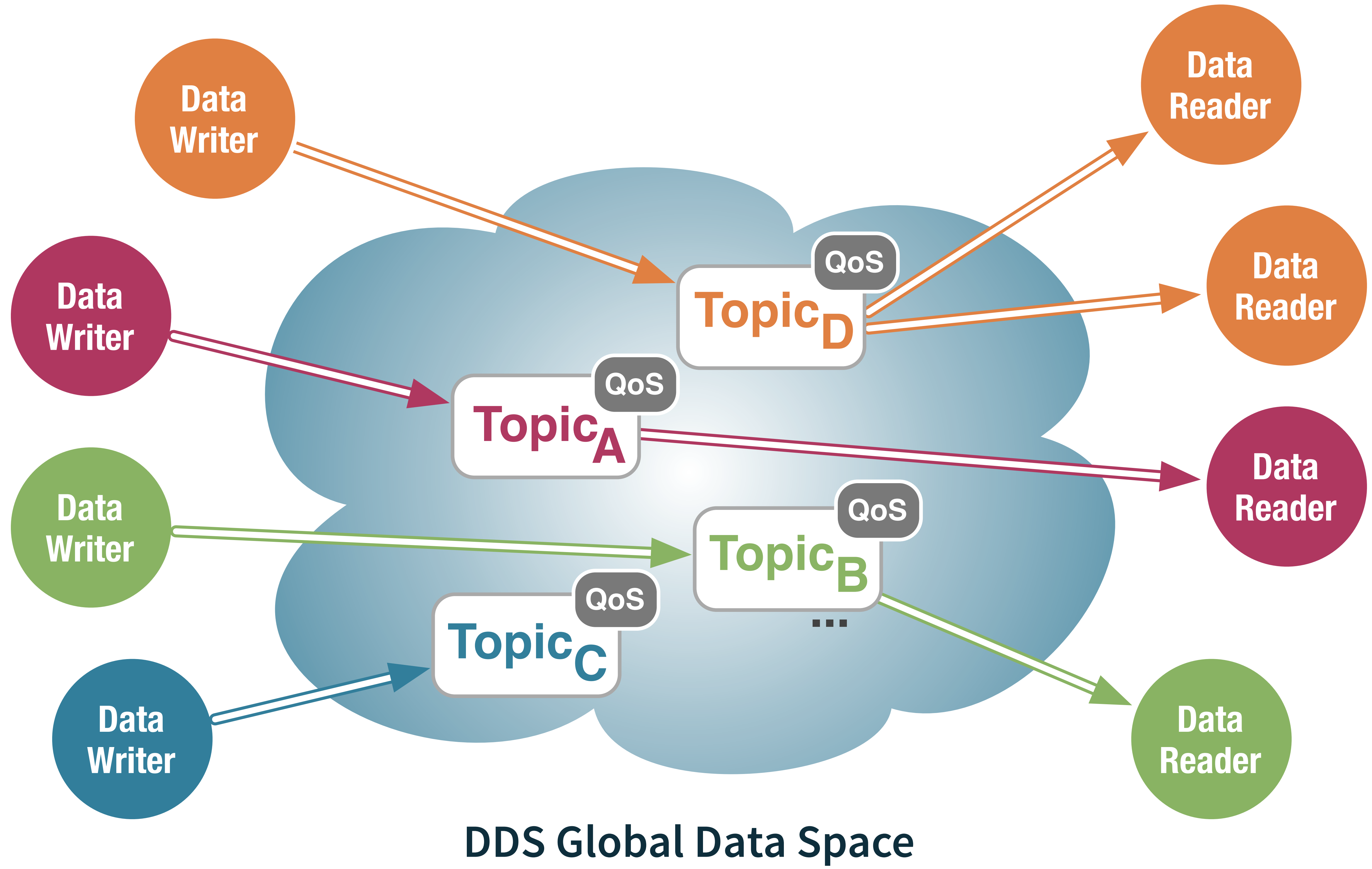
VIRTUALISED DATA SPACE

Applications can **autonomously** and **asynchronously read** and **write** data enjoying **spatial** and **temporal decoupling**



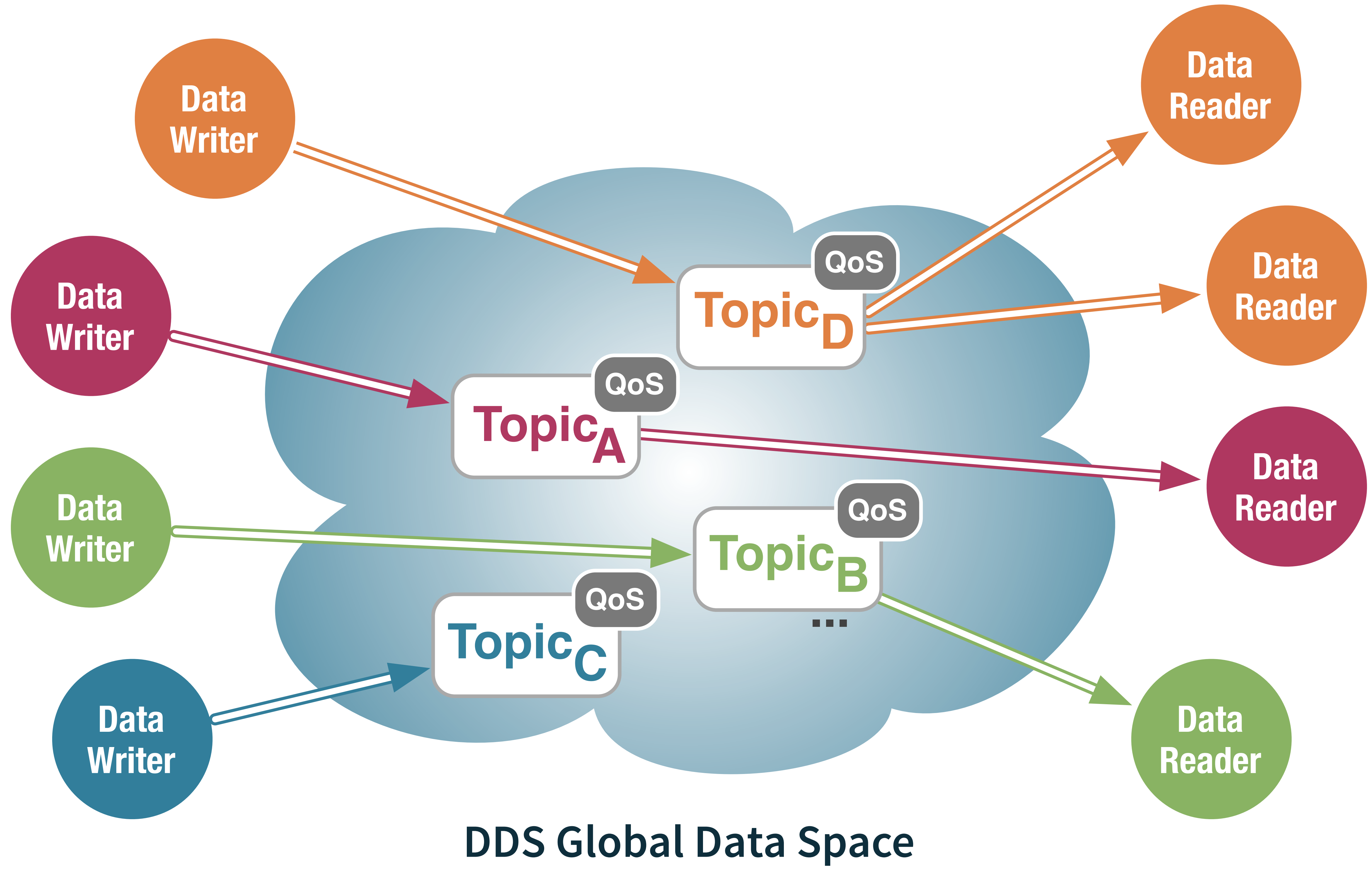
DYNAMIC DISCOVERY

Built-in dynamic discovery **isolates applications from network topology** and **connectivity** details



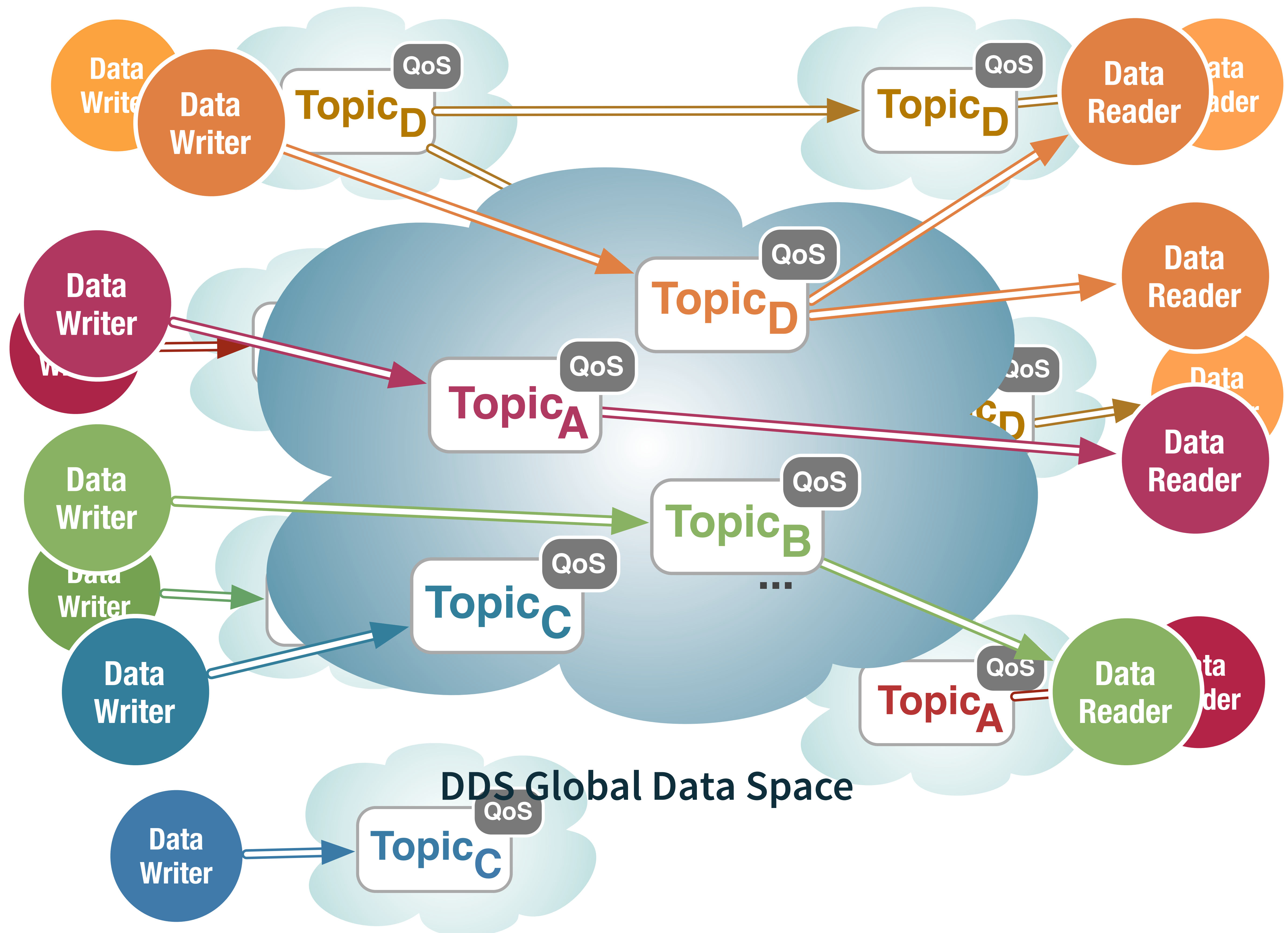
QOS ENABLED

QoS policies allow the expression of **data**'s **temporal** and **availability constraints**



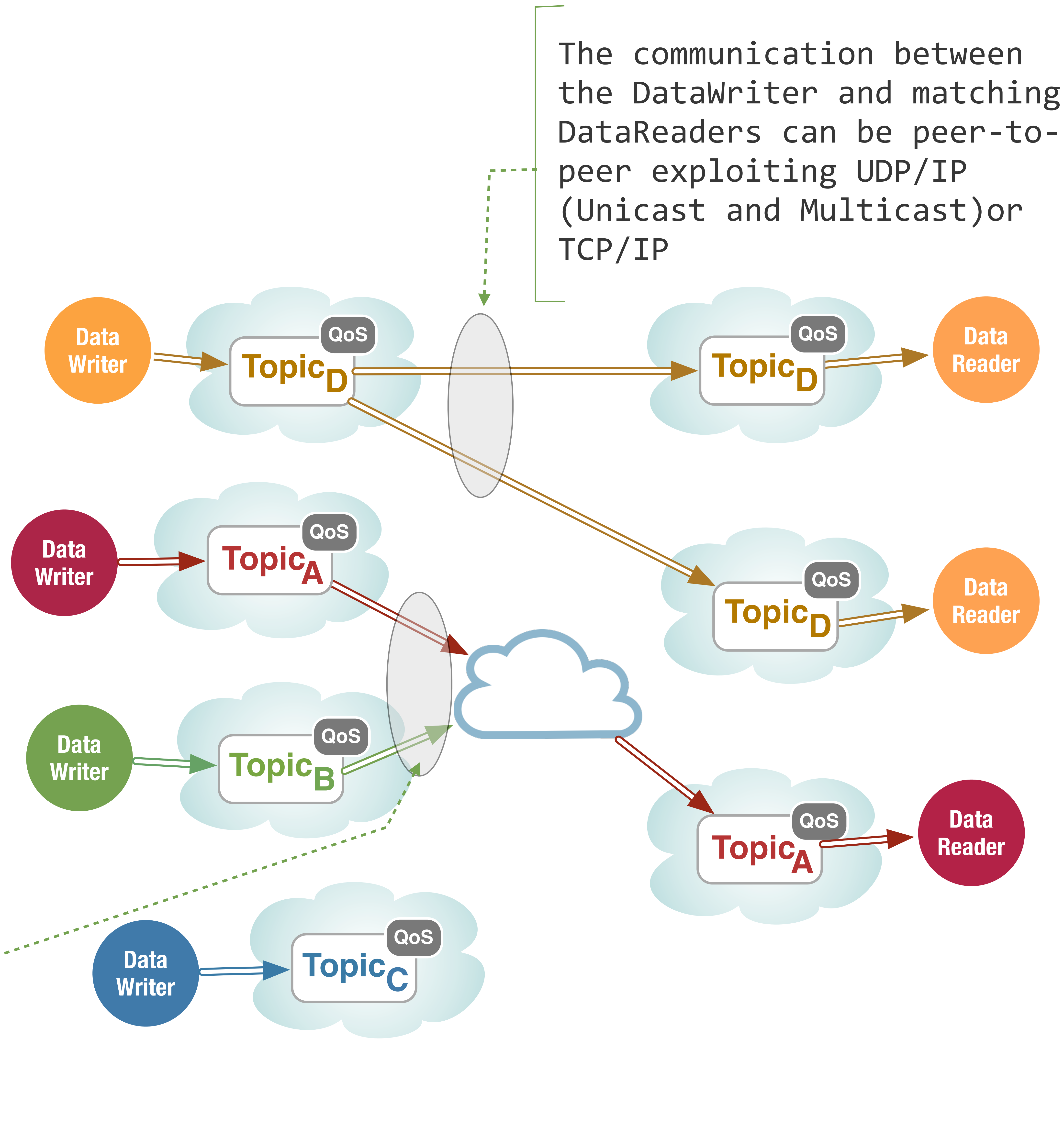
DECENTRALISED DATA-SPACE

No single point of failure or bottleneck



ADAPTIVE CONNECTIVITY

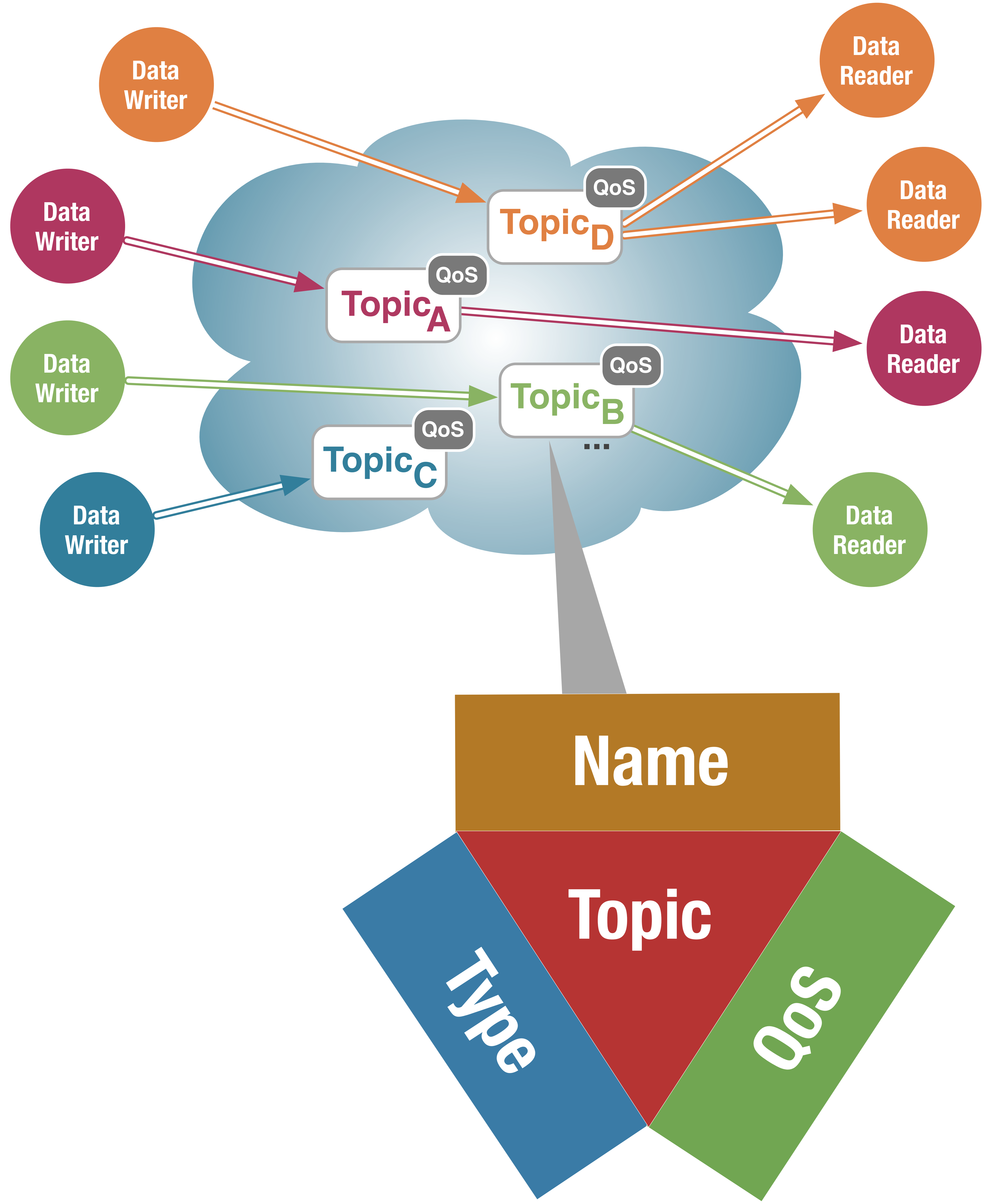
Connectivity is **dynamically adapted** to choose the most effective way of sharing data



TOPIC

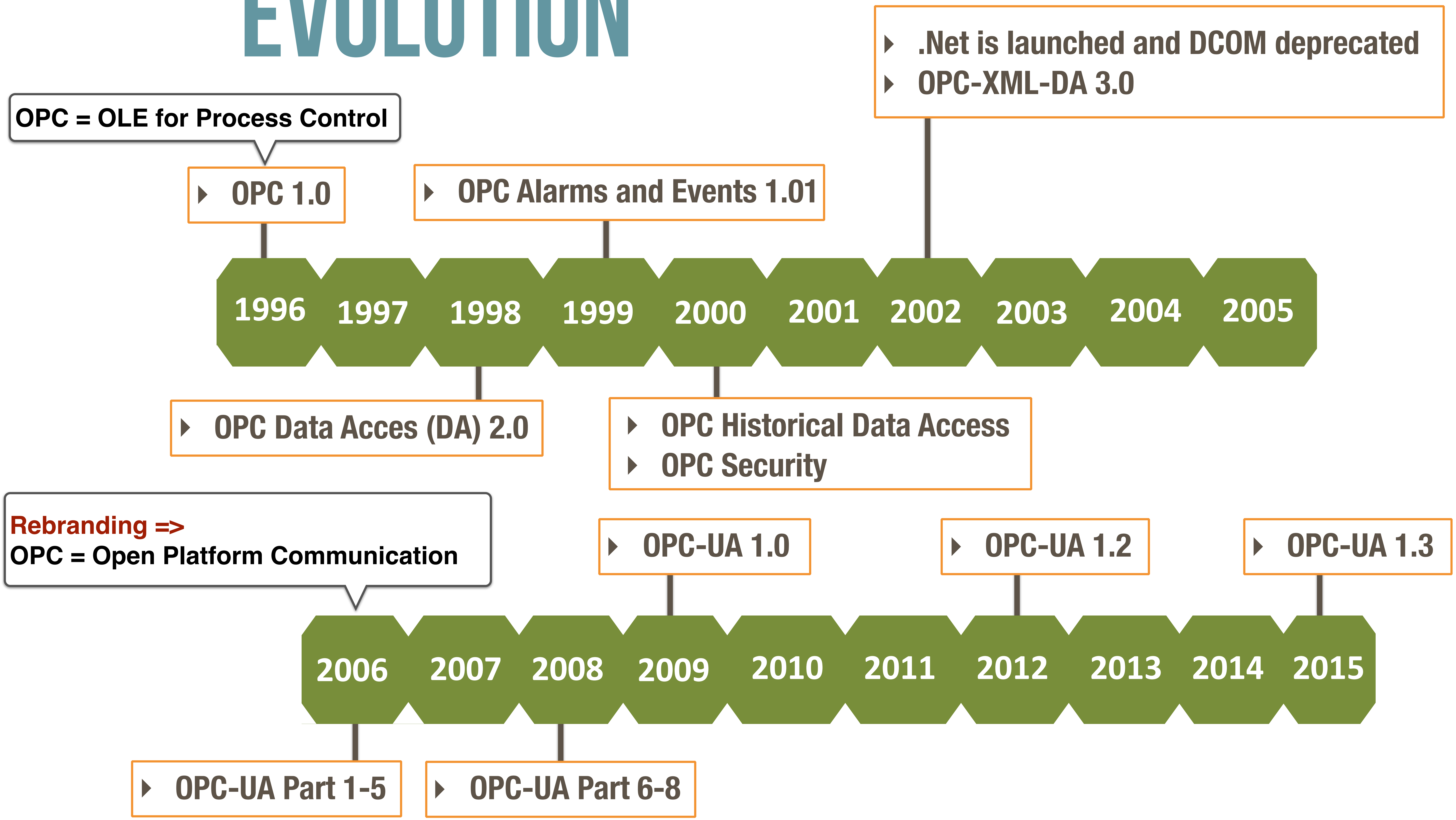
A domain-wide information's class A **Topic** defined by means of a <name, type, qos>

DDS Topics allow to expression of **functional** and **non-functional properties** of a system **information model**



OPC

OPC STANDARD EVOLUTION



OPC UA

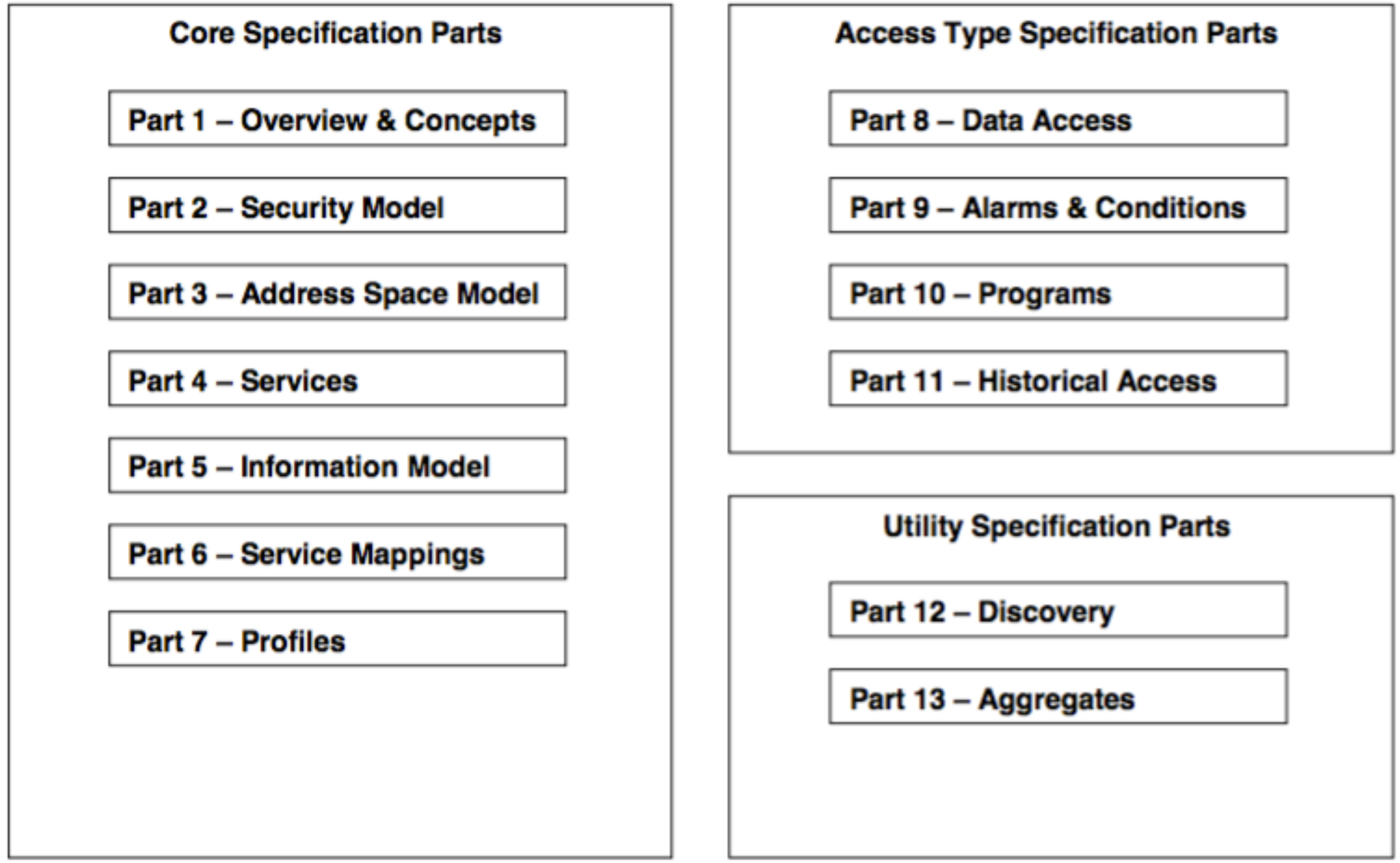
OPC UA is a **platform-independent standard** through which various kinds of systems and devices can **communicate** by sending Messages between **Clients and Servers** over various types of networks. It supports robust, secure communication that assures the identity of Clients and Servers and resists attacks. OPC UA defines sets of Services that Servers may provide […].

Information is conveyed using OPC UA-defined and vendor-defined **data types**, and Servers define **object models** that Clients can dynamically discover. Servers can provide access to both **current and historical data**, as well as **Alarms and Events to notify Clients of important changes**.

[Extract from OPC-UA Overview and Concepts v1.03

STANDARD STRUCTURE

OPC-UA is organised in Core, Access Type and Utility specifications

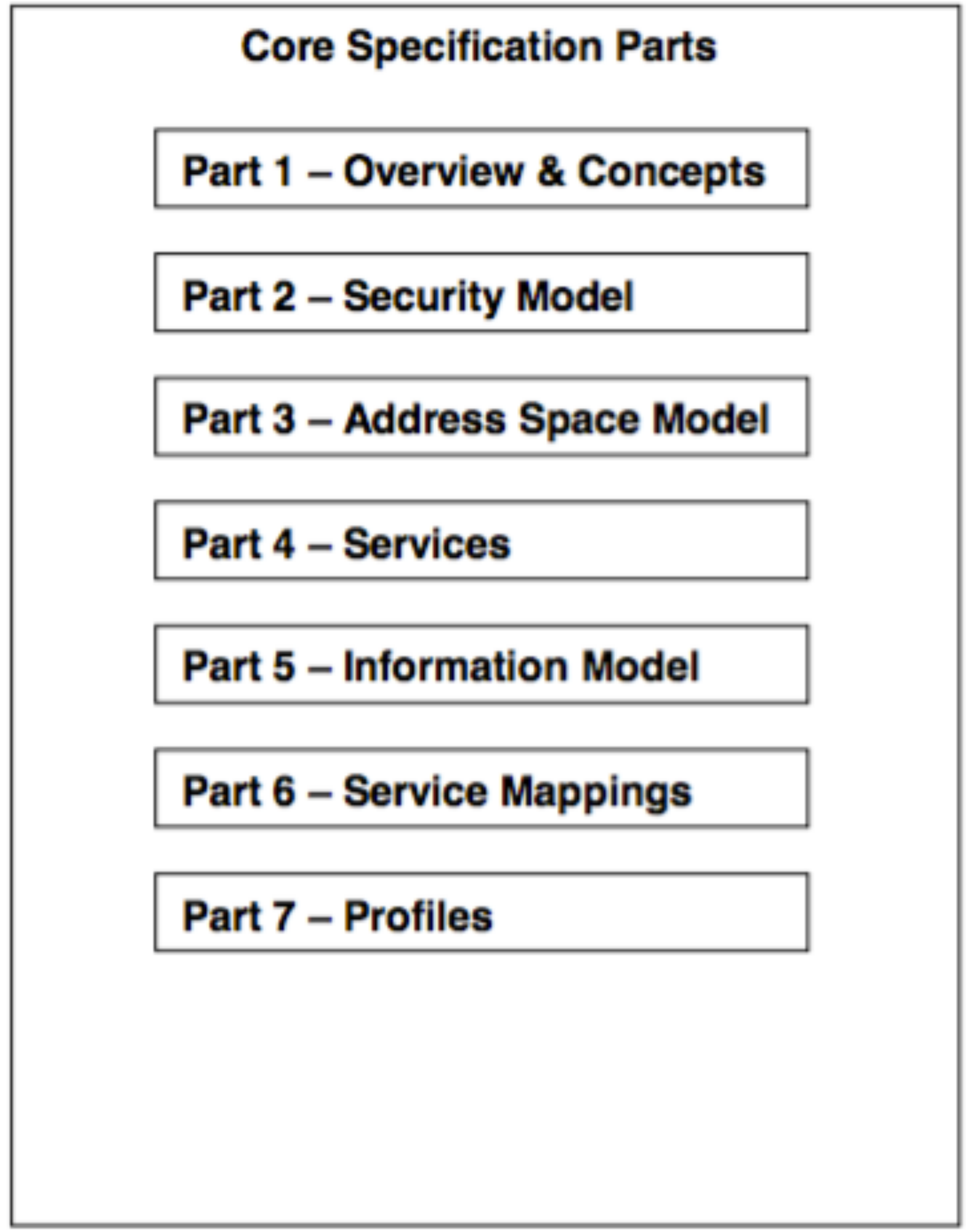


CORE SPECIFICATIONS

Security Model. Defines how communication between OPC-UA Clients and Servers should be secured

Address Space Model. Describes the contents and structure of the Server's AddressSpace.

Services. Defines the services provided by OPC UA Servers.

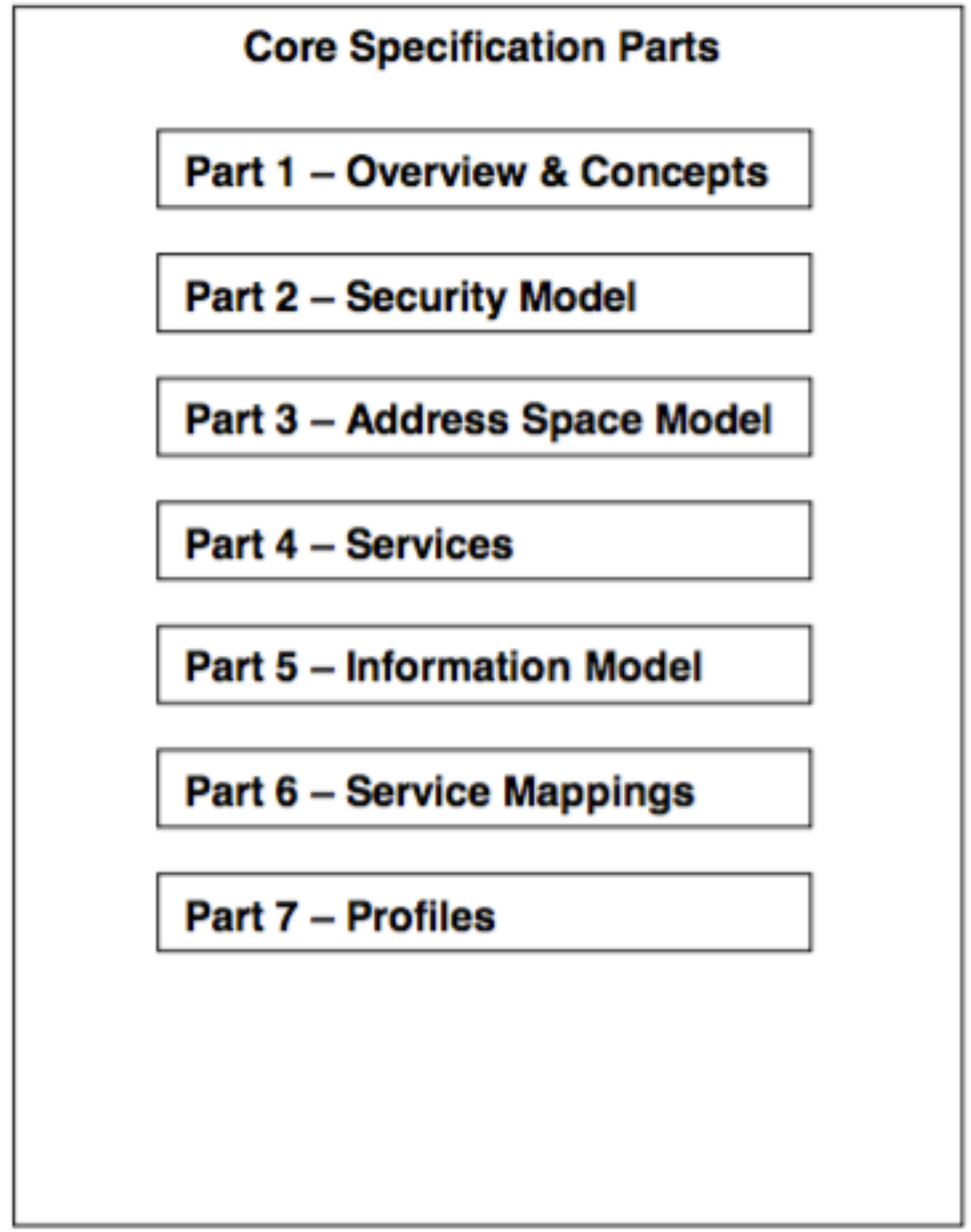


CORE SPECIFICATIONS

Information Model. Specifies the types and their relationships defined for OPC UA Servers.

Service Mapping. Specifies the mappings to transport protocols and data encodings supported by OPC UA.

Profiles. Specifies the Profiles that are available for OPC Clients and Servers. These Profiles provide groups of Services or functionality that can be used for conformance level certification.



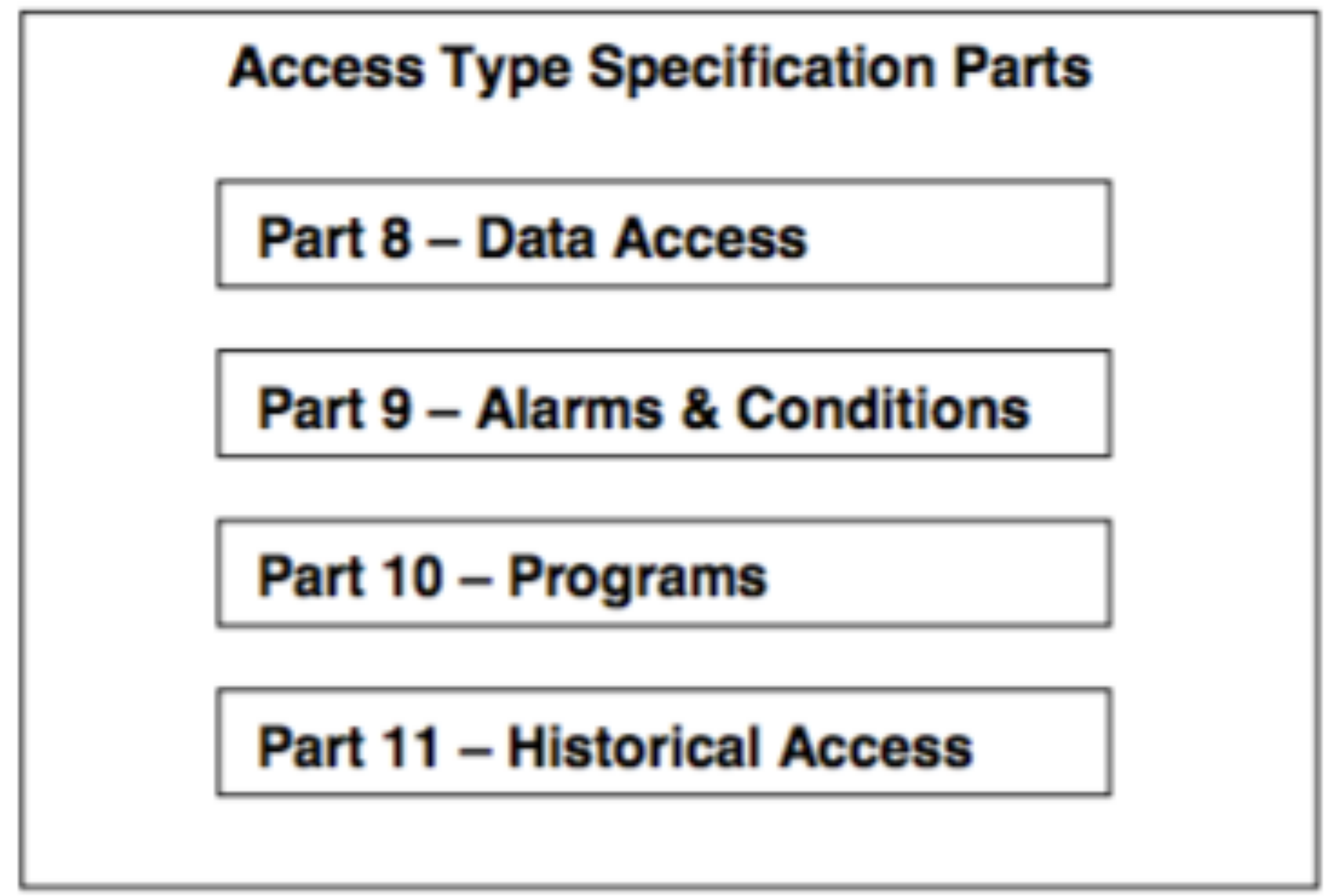
ACCESS TYPE SPECIFICATIONS

Data Access. Specifies the use of OPC UA for data access.

Alarms & Conditions. Specifies the use of OPC UA support for access to Alarms and Conditions.

Programs. Specifies OPC UA support for access to Programs.

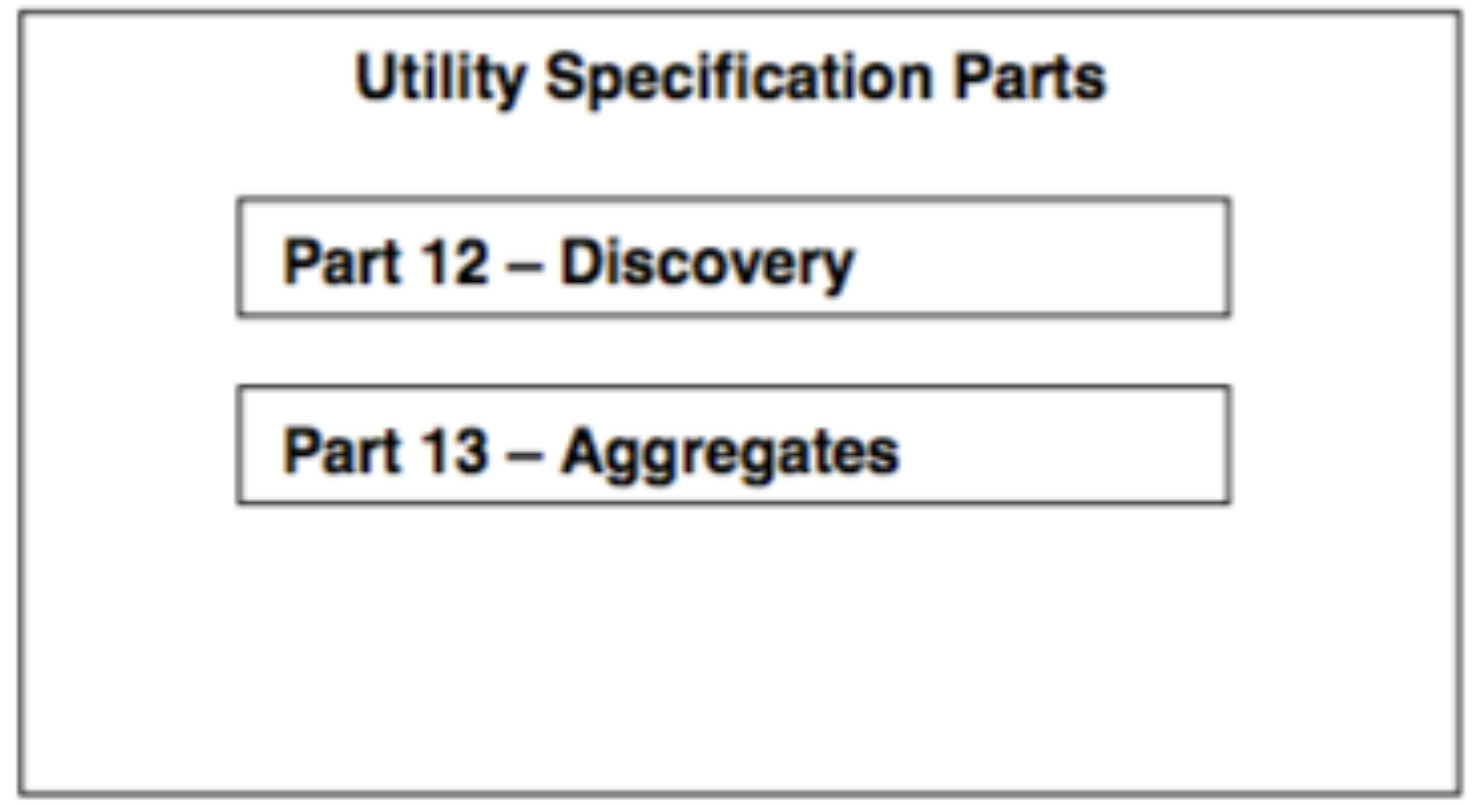
Historical Access. Specifies use of OPC UA for historical access. This access includes both historical data and historical Events.



UTILITY SPECIFICATIONS

Discovery. Specifies how Discovery Servers operate in different scenarios and describes how UA Clients and Servers should interact with them.

Aggregates. specifies how to compute and return aggregates like minimum, maximum, average etc. Aggregates can be used with current and historical data.

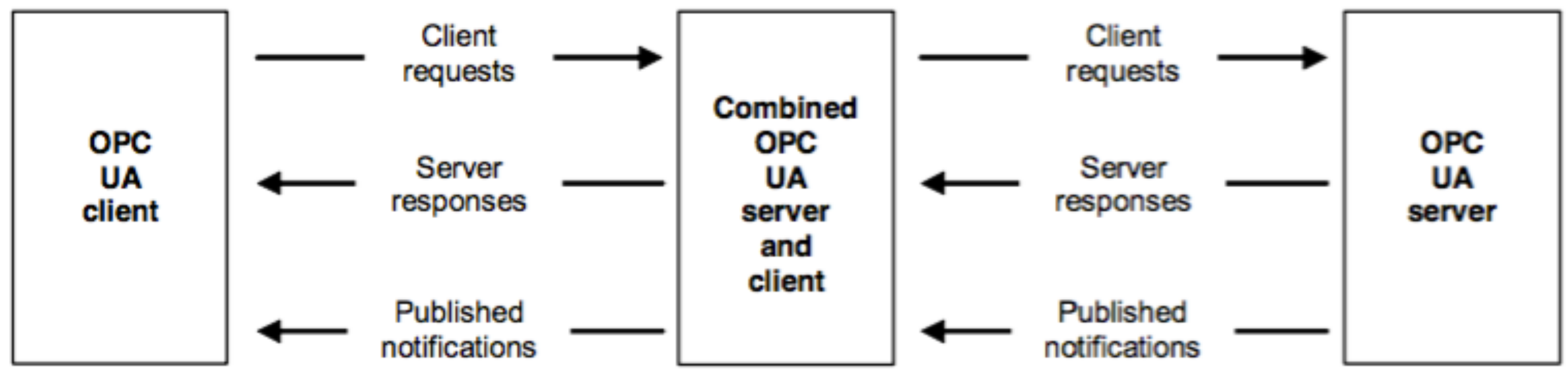


OPC-UA Abstractions

CLIENT/SERVER

OPC UA is rooted in the **client-server model**.

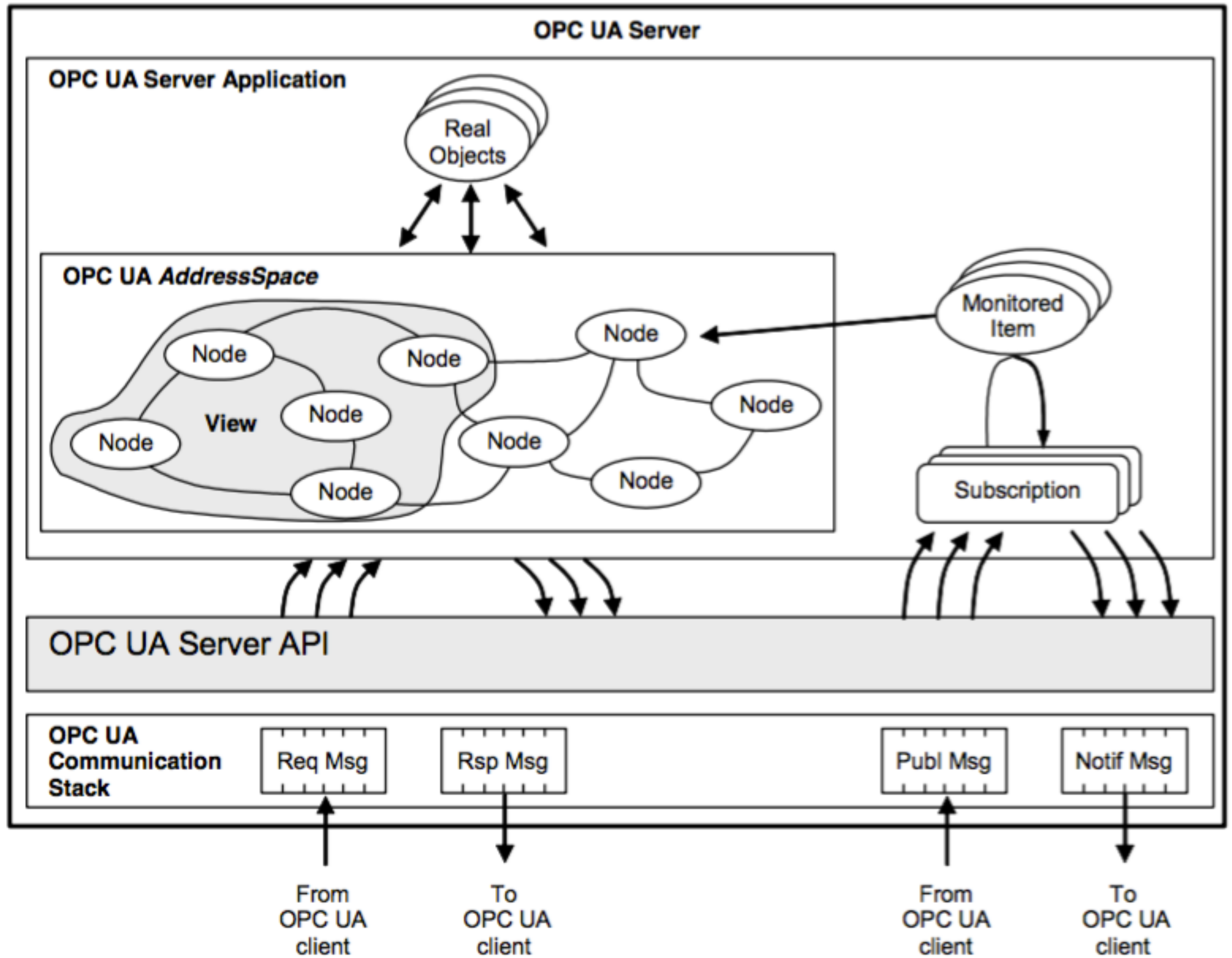
Clients interact with the server and are coupled in time, e.g. client and server must be running at the same time for anything useful to happen.



CLIENT/SERVER

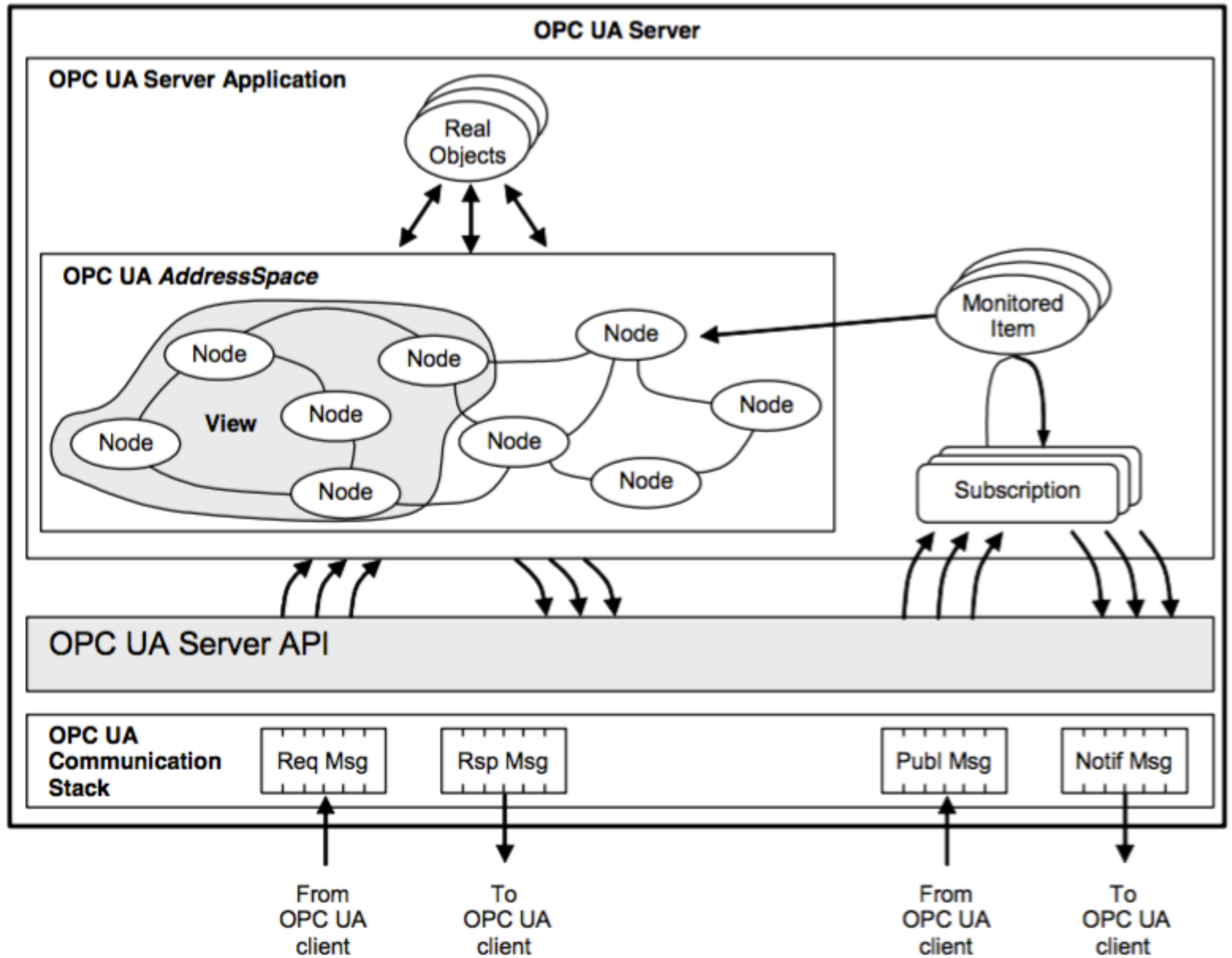
The OPC UA server provides, through a standard API, access to its address space.

The address space is organised as a graph of nodes. A server can decide to expose a specific view.



CLIENT/SERVER

Clients can register subscriptions with the server and eventually be notified by a server when data changes or some event occurs.



Checkpoint

DDS & OPC-UA SIMILARITIES

For the objectives stated from both the **DDS** and **OPC-UA** specifications it emerges how both standards **address** the problem of **information management in distributed systems**.

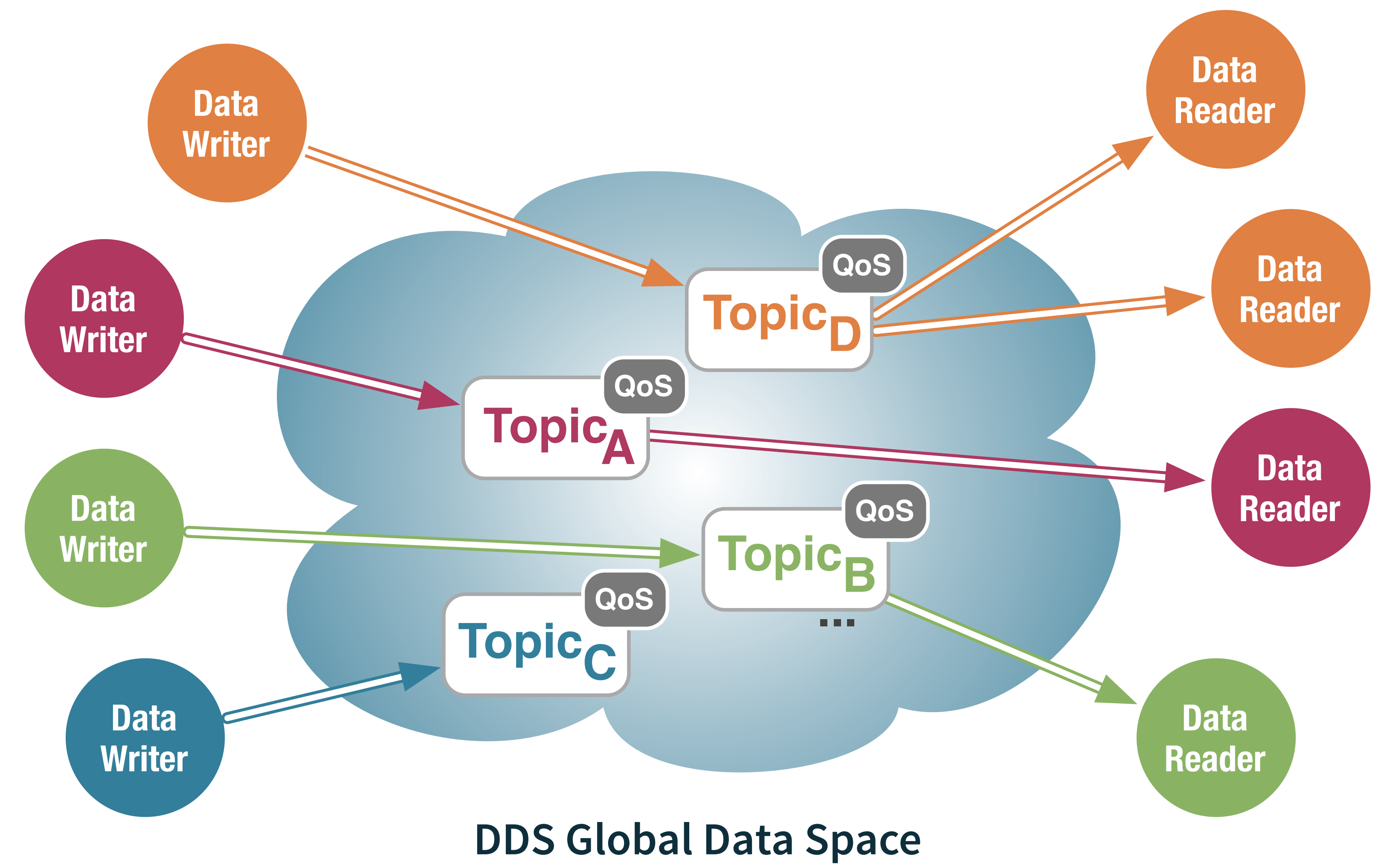
Both DDS as well as OPC UA provide support for **Information Modelling**.

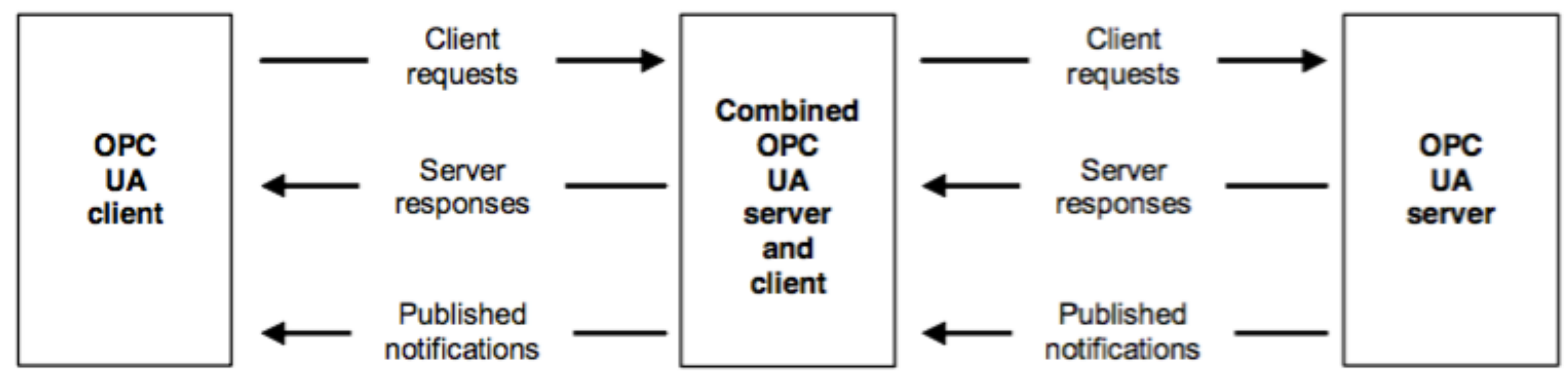
DDS through **relational** data modelling while **OPC UA** via **Object Oriented** modelling.

DDS & OPC-UA DIFFERENCES

DDS abstraction is centred around a Decentralised Data Space that decouples applications in time and space.

OPC UA abstraction is centred around client-server.

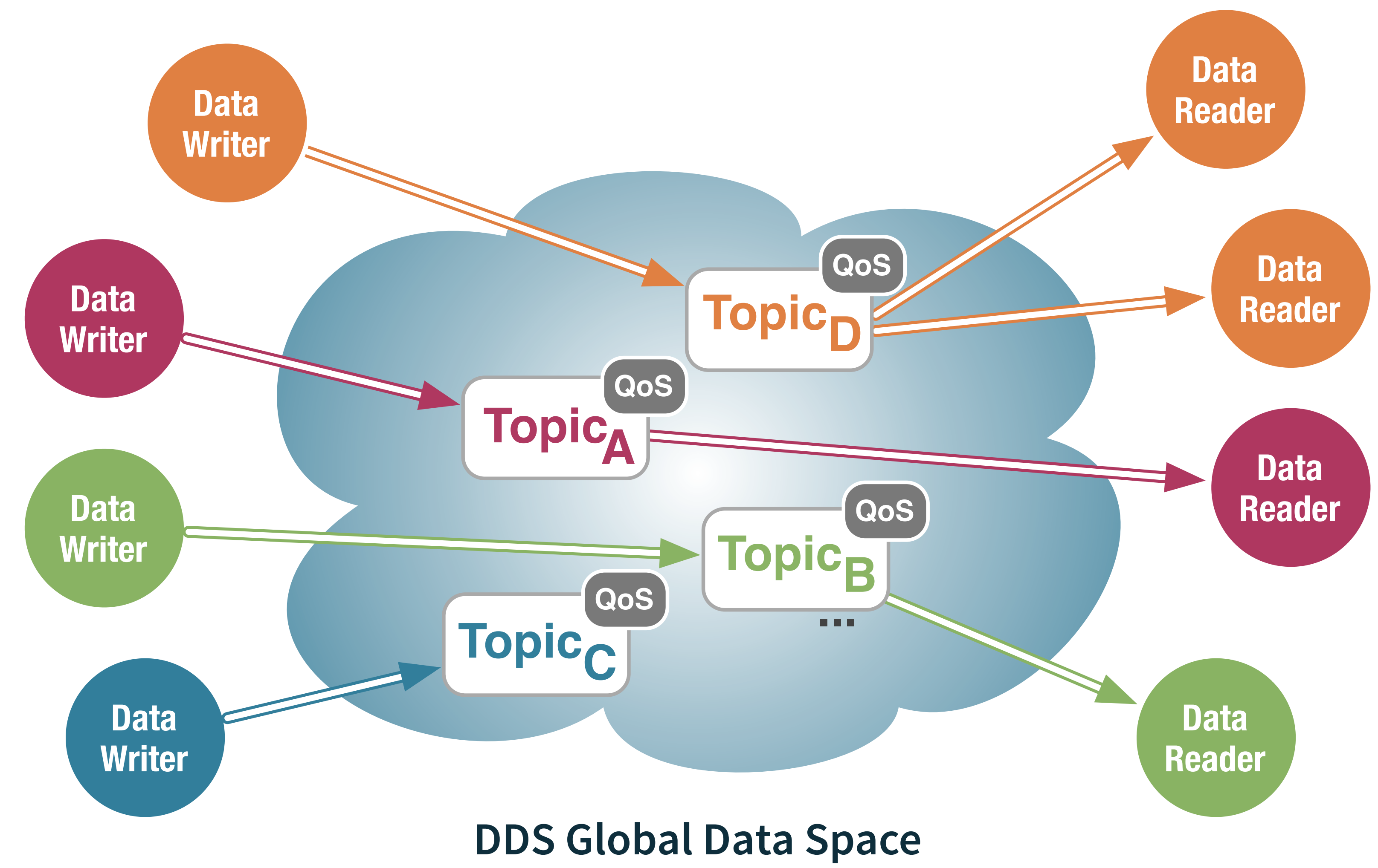


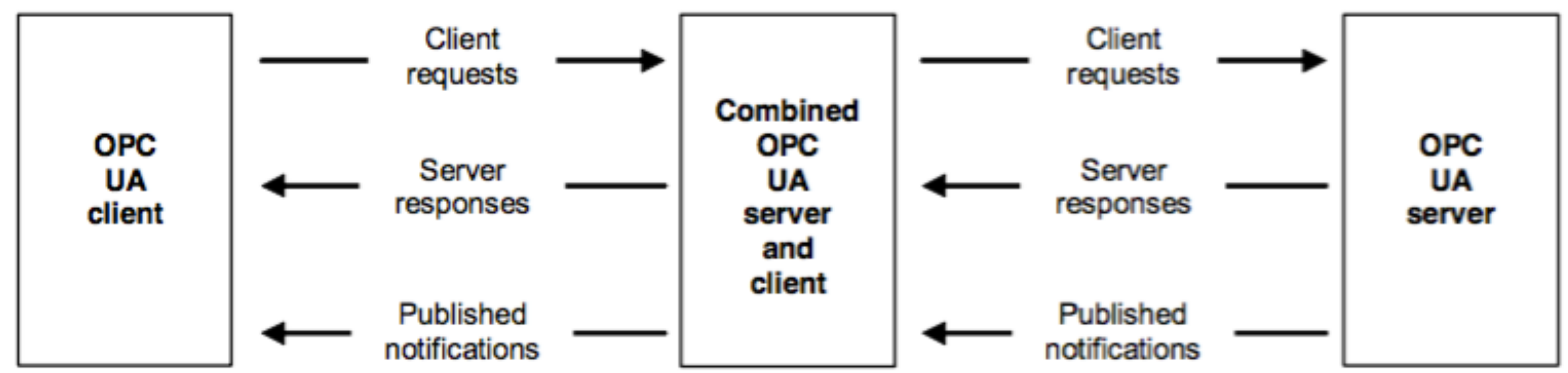


DDS & OPC-UA DIFFERENCES

DDS applications interact by anonymously and asynchronously reading and writing data in the global data space.

OPC UA applications interact by invoking requests on one or more UA servers.

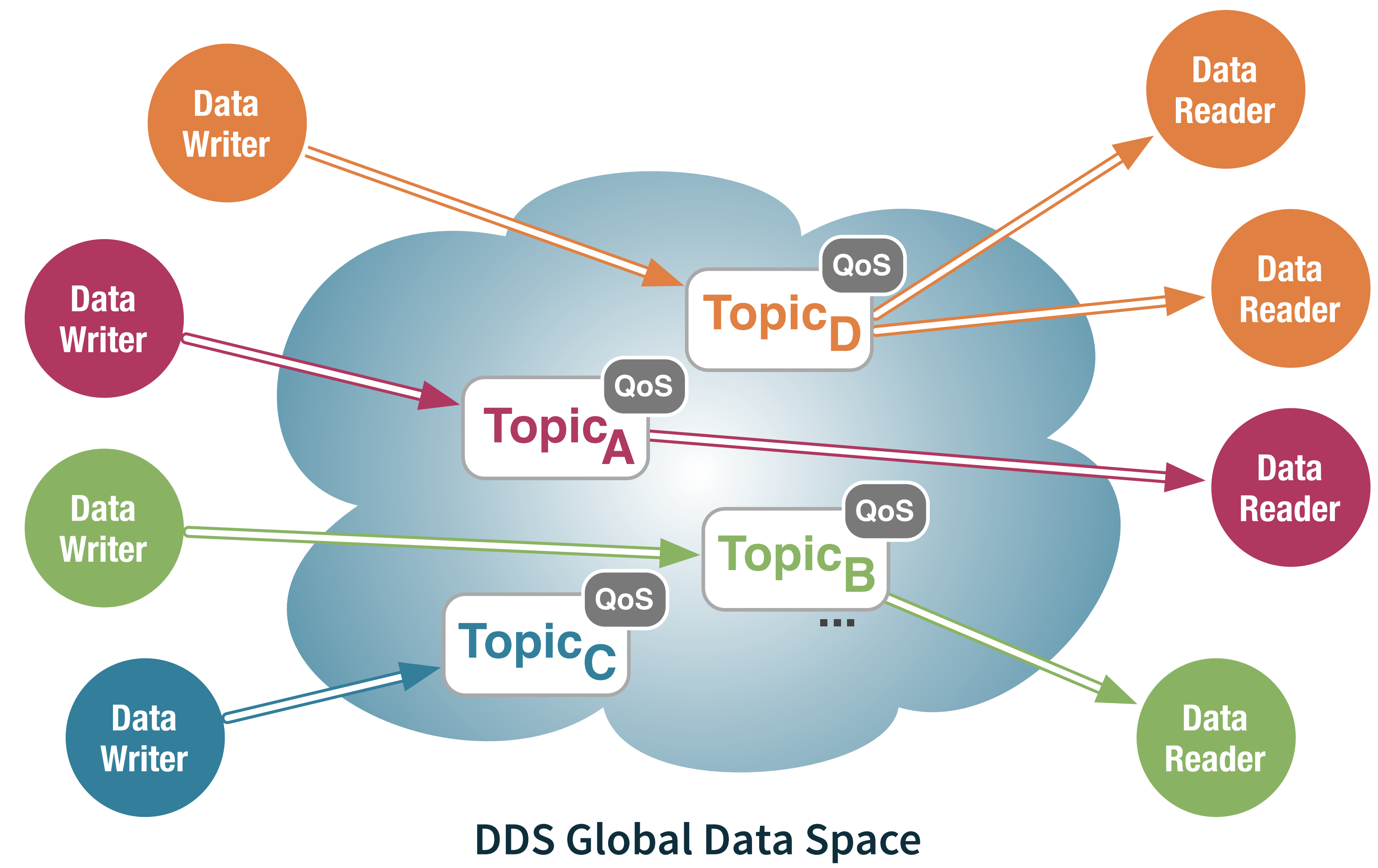


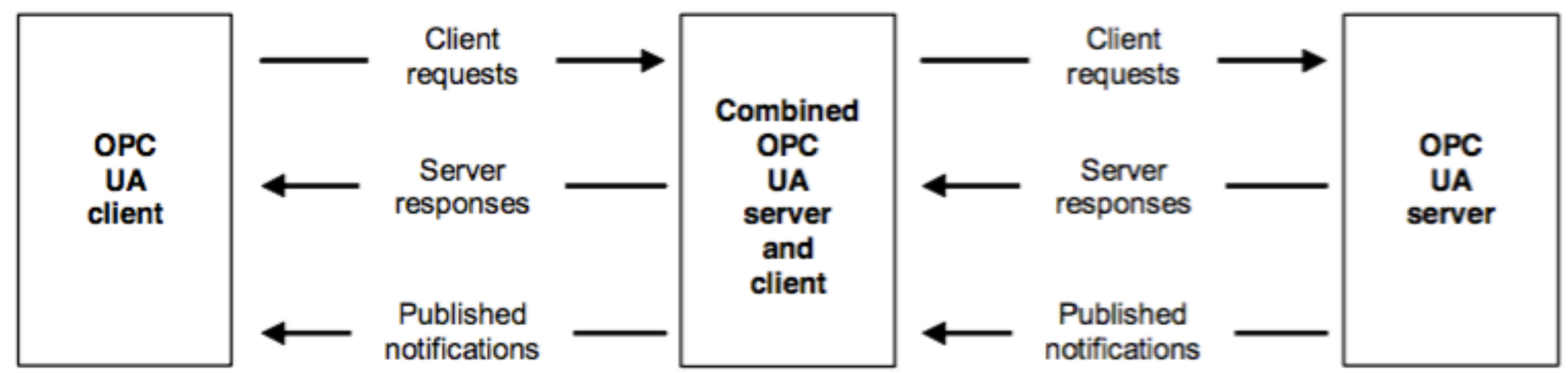


DDS & OPC-UA DIFFERENCES

DDS favours resource-oriented and declarative programming style, i.e., you express how things should be.

OPC UA favours an imperative programming style, i.e. you express how things should be done.

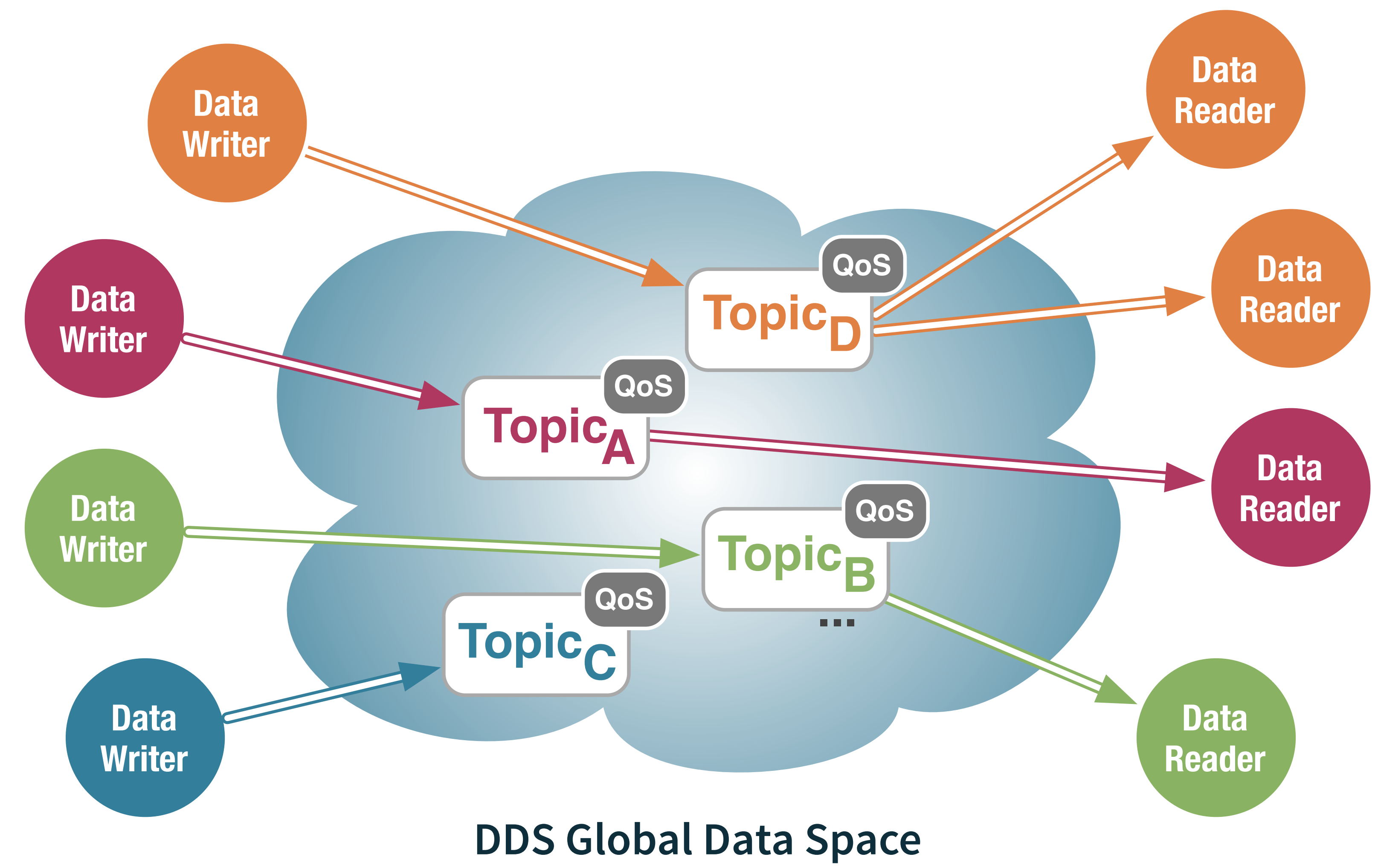


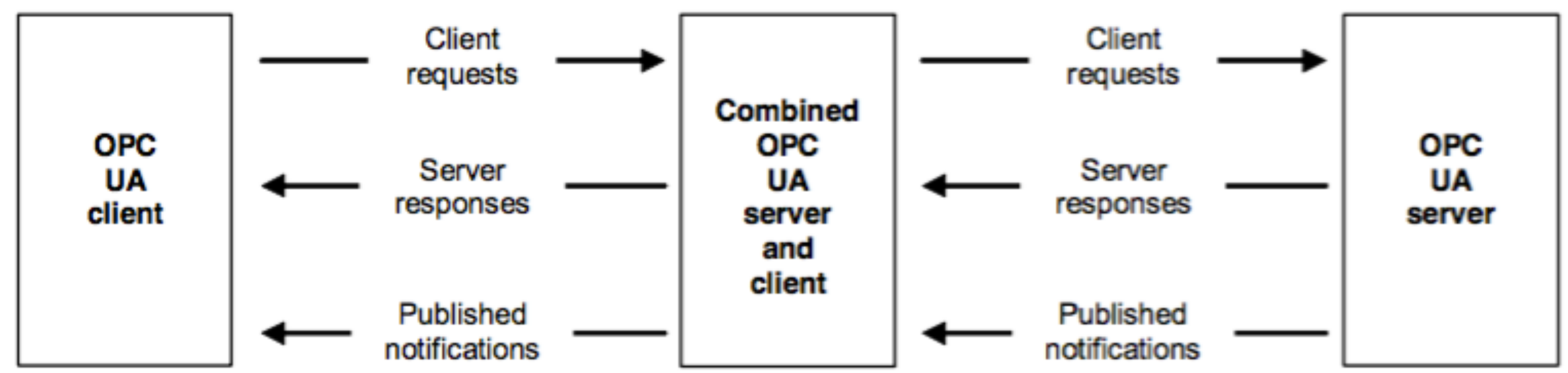


DDS & OPC-UA DIFFERENCES

DDS applications enjoy complete location transparency. Data gets automatically where there is interest.

OPC UA applications have to undergo a two step resolution process, first they need to look-up servers and then browse data in its address space.

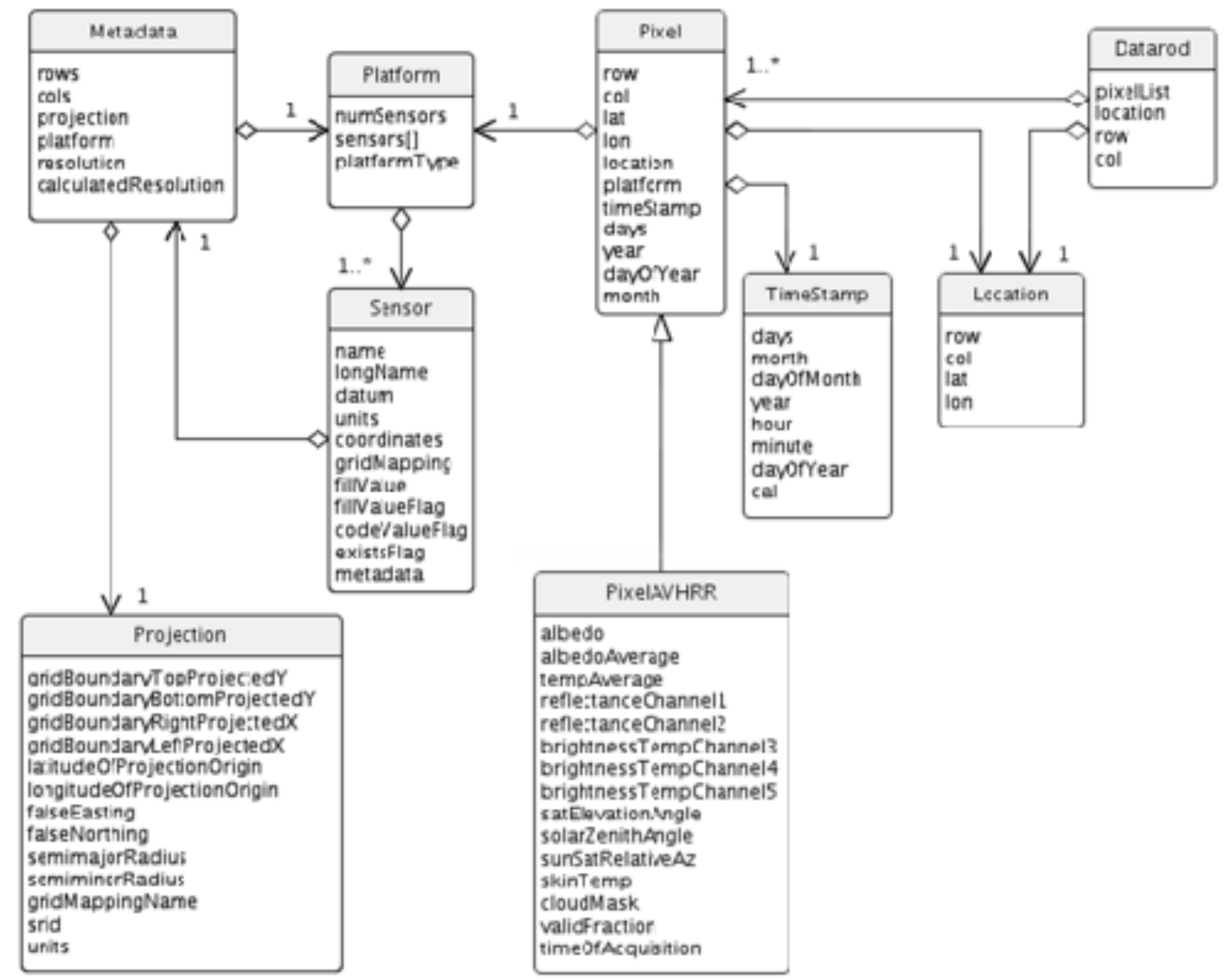




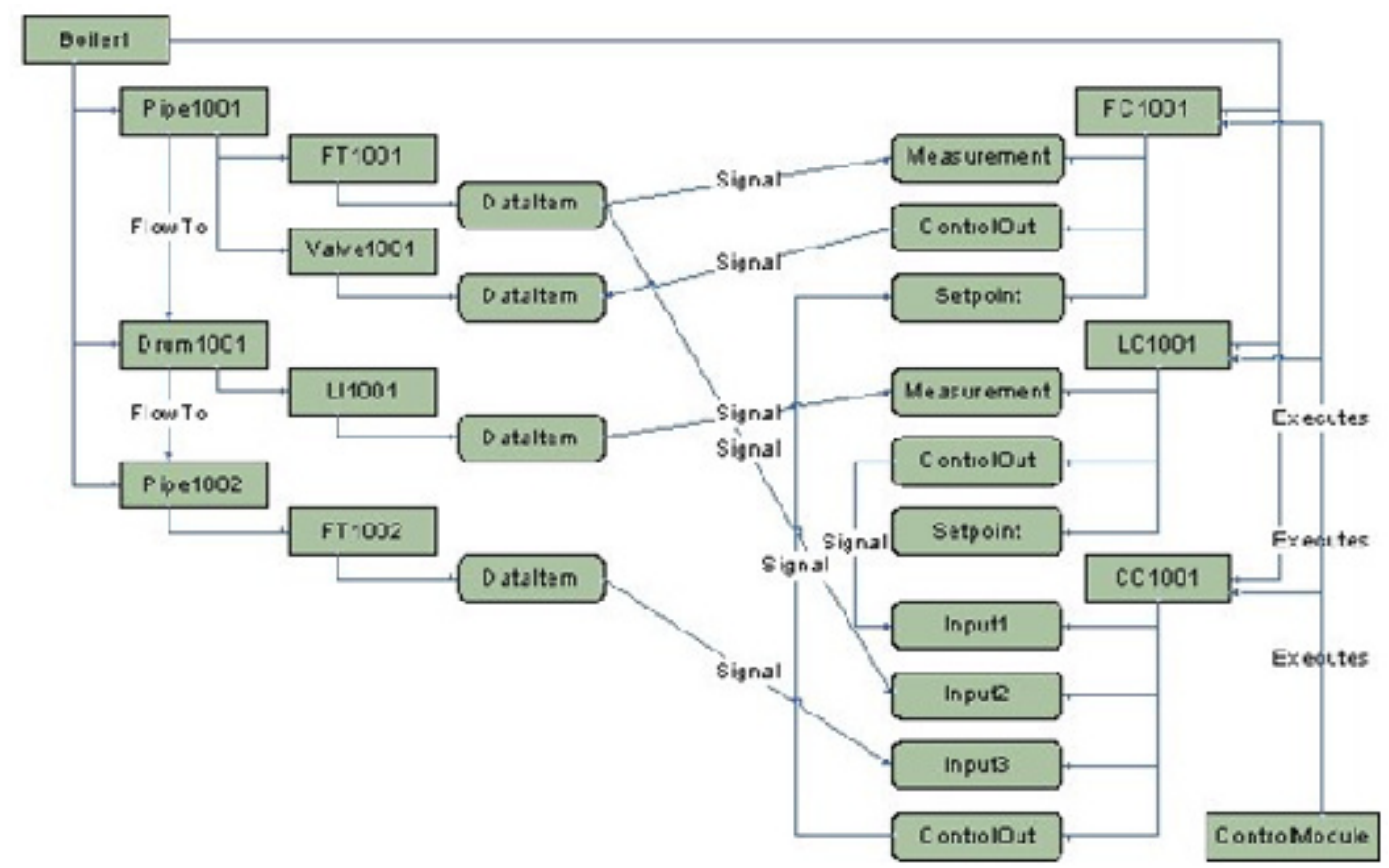
DDS & OPC-UA DIFFERENCES

DDS data modelling is relational. DDS Information model can be queried joined and projected

OPC UA data modelling is Object Oriented. **OPC UA** can be browsed and queried when servers support this extension.



DDS data model

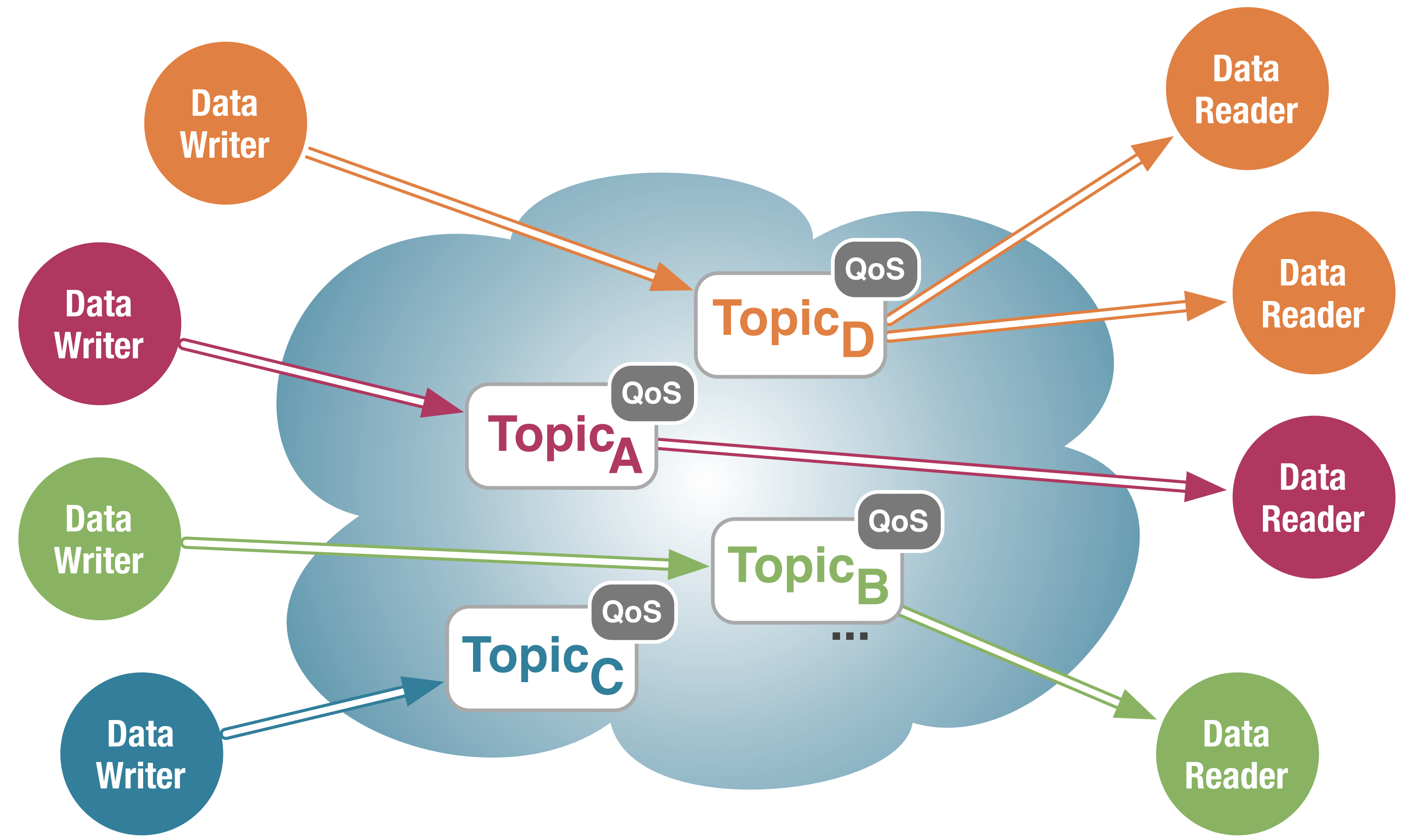


OPC-UA data model

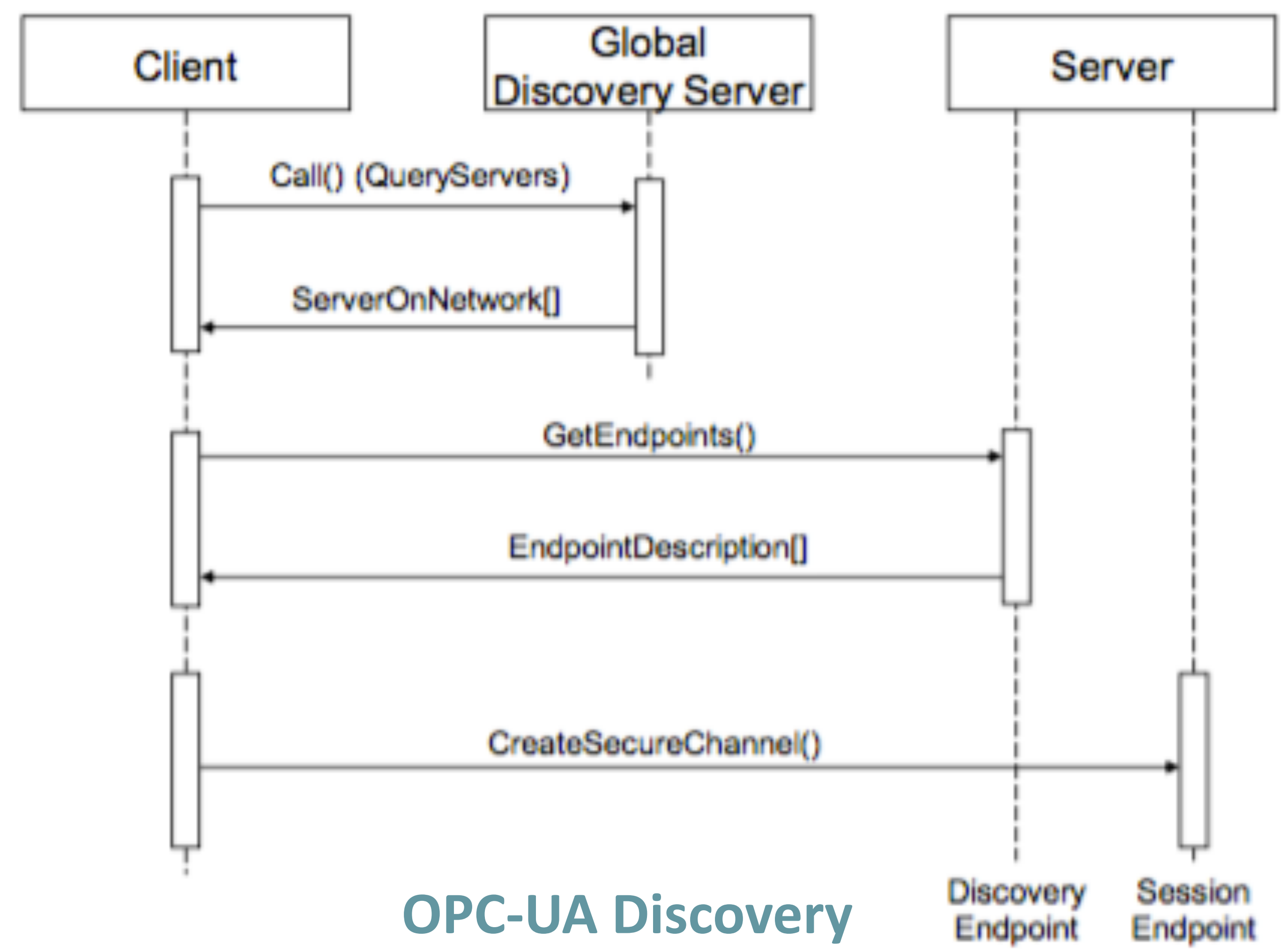
DDS & OPC-UA DIFFERENCES

DDS' Dynamic Discovery allows for very dynamic systems in which applications and data are discovered automatically. Applications are notified of relevant information discovered. DDS has out-of-the box a **plug-and-play** nature.

OPC UA applications have to explicitly "search" for things. That means that supporting plug and play behaviours requires programmatic effort.



DDS Global Data Space

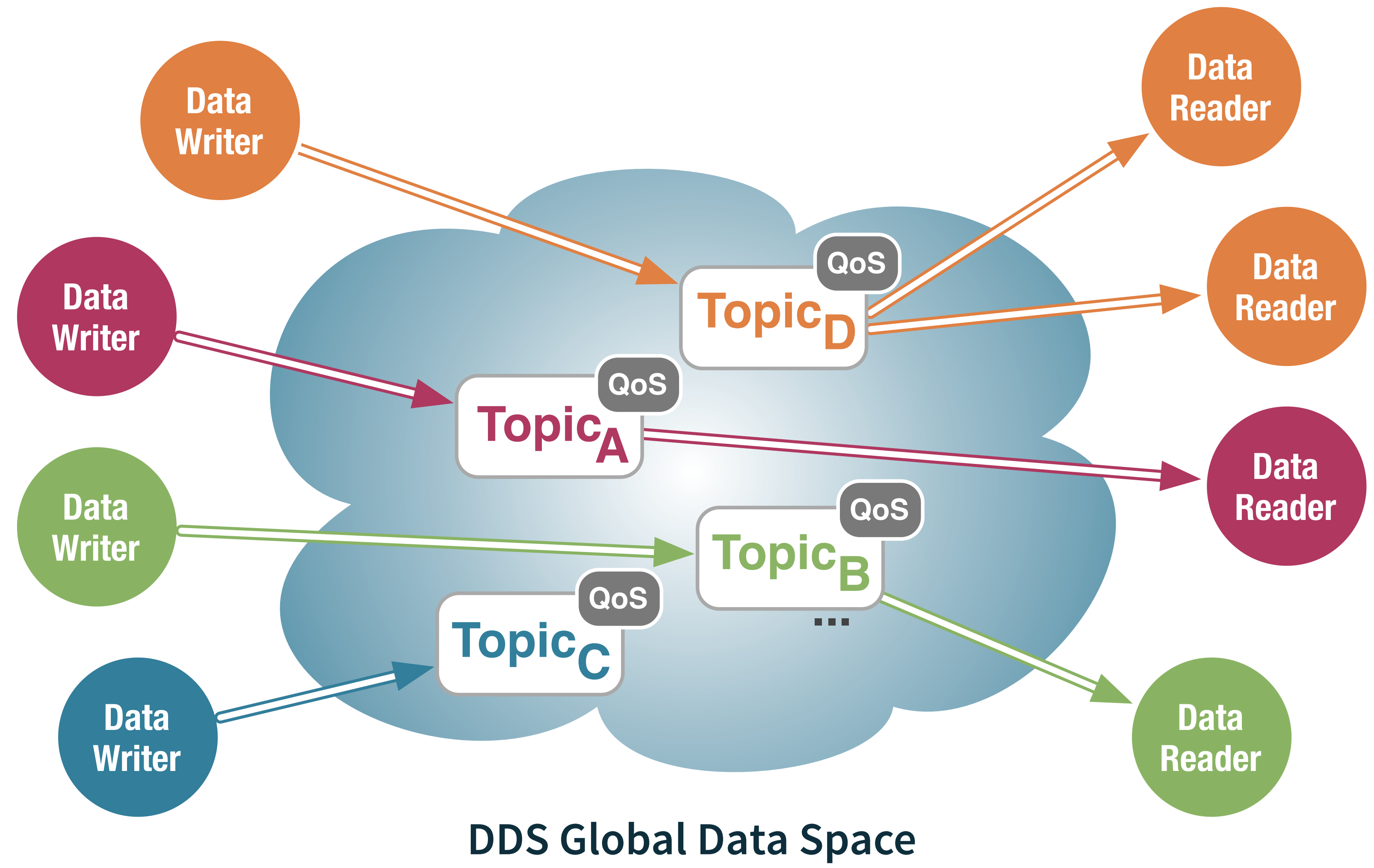


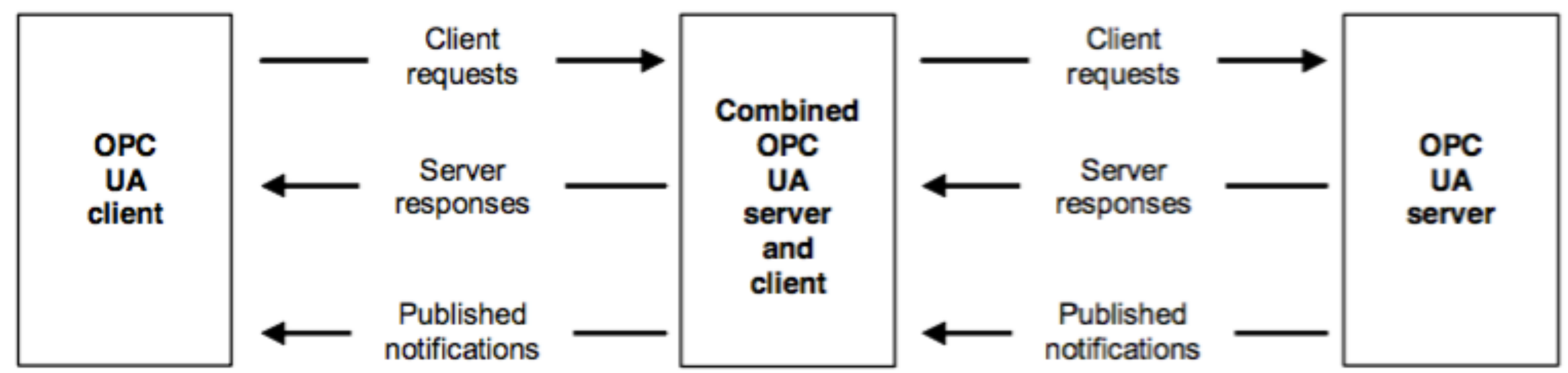
OPC-UA Discovery

DDS & OPC-UA DIFFERENCES

DDS allows information to be annotated with QoS so to capture non-functional properties.

OPC UA does not support QoS specification.

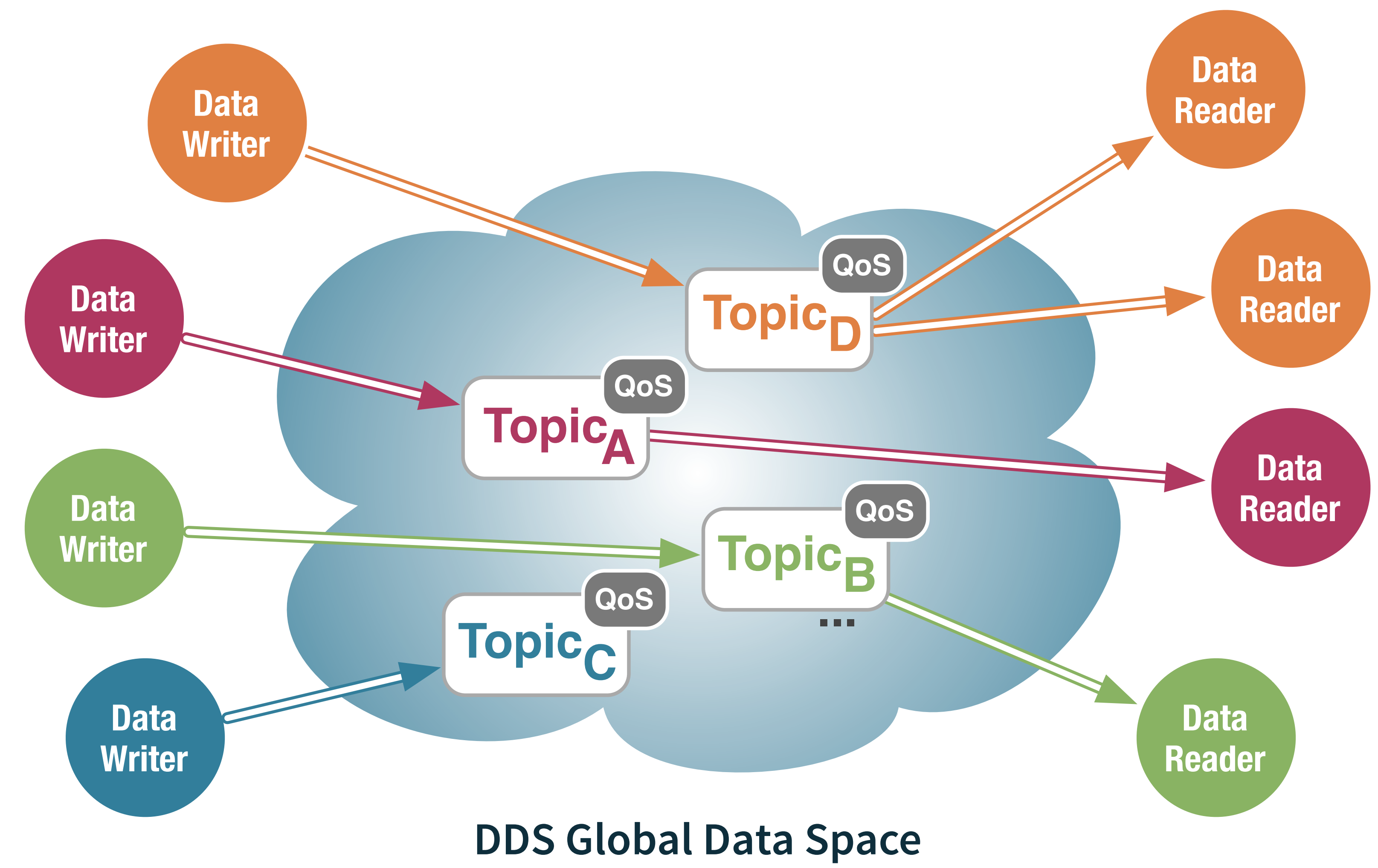


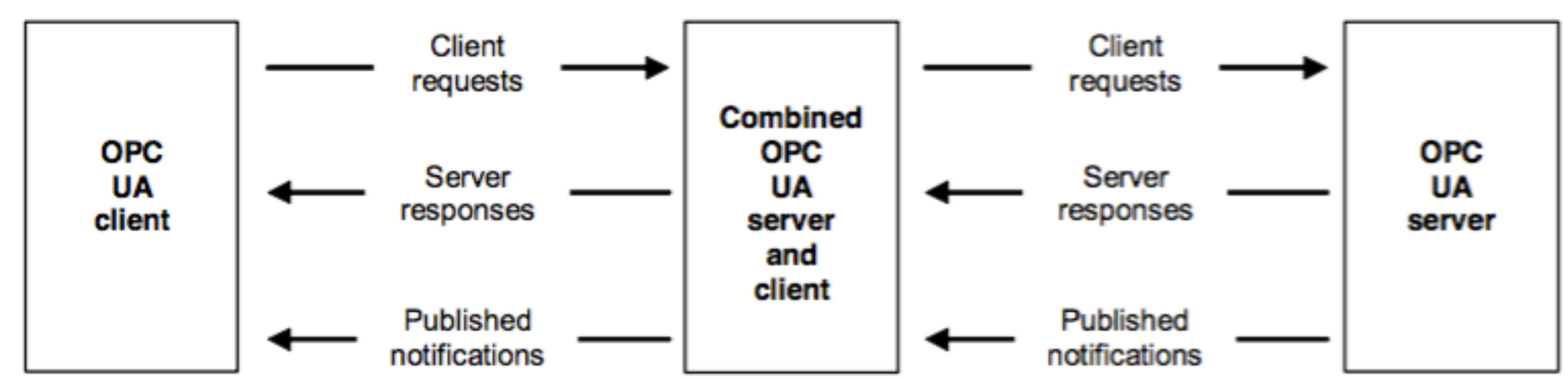


DDS & OPC-UA DIFFERENCES

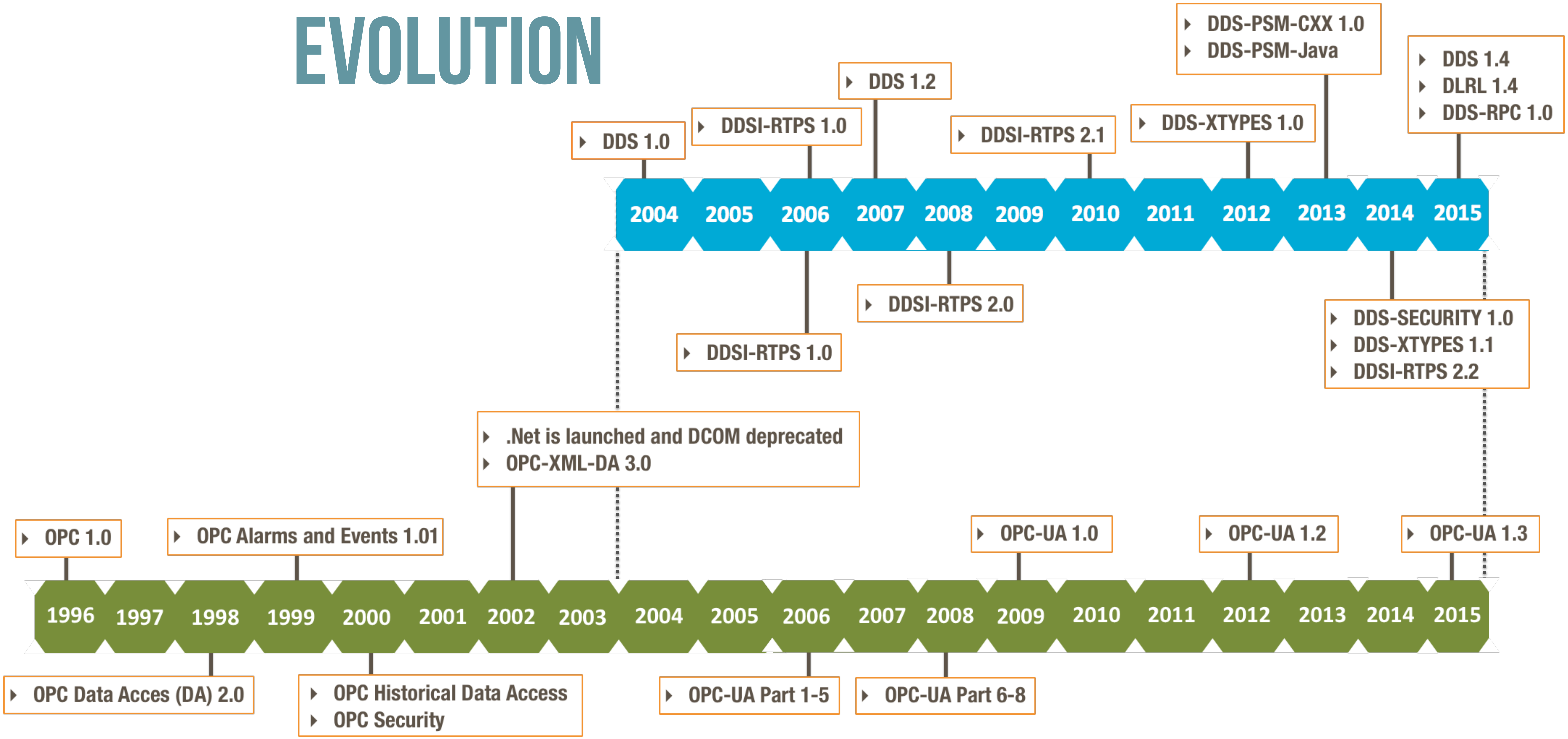
DDS security addresses data in motion as well as data at rest. Additionally DDS security provides pluggable Authentication, Access Control, Cryptography and Logging

OPC UA security focuses on establishing secure channels between client and servers.





DDS / OPC STANDARD EVOLUTION



STANDARD "MAPPING"

Part 2. Security Model	DDS-Security
Part 3. Address Space Model	DDS
Part 4. Services	There is no server concept in DDS. Negotiation is done via the discovery services.
Part 5. Information Model	DDS, DDS-XTypes
Part 6. Service Mappings	Platform Specific Models of each specification define implementation when necessary.
Part 7. Profiles	Each Specification has its own conformance points.
Part 8. Data Access	DDS, DDS-XTypes
Part 9. Alarm & Conditions	Alarms / Events are represented as topics in DDS.
Part 10. Programs	DDS-RPC
Part 11. Historical Access	DDS, DDSI-RTPS, DDS provides built-in support for history. Beside this vendor support seamless integration with time series stores.
Part 12. Discovery	DDS, DDSI-RTPS. DDS Has built-in decentralised discovery.
Part 13. Aggregates	DDS promotes micro-service architectures, thus aggregates are typically provided by analytics, or similar applications.

Misconceptions

Internet of Things' Organizational Confusion

Appeared in AutomationWorld on the 23rd of July 2015

http://bit.ly/1Mzk4PV

As industry begins to better understand the Internet of Things, there remains some confusion about the role of industry organizations supporting the concept and how they relate to each other.

Speaking of OPC UA, two technology areas creating some of the greatest confusion in industry around IoT are DDS (Data Distribution Service)—often referenced by IIC—and OPC UA. The confusion surrounding these technologies stems from the fact that both are promoted as protocols enabling interoperability between devices, machines, and systems. According to [redacted] "DDS offers deterministic communication and is therefore comparable to Profinet or EtherCAT. The aim is fast data exchange within the systems. OPC focuses on interoperability—the exchange between systems. Above all, OPC UA offers security and configurable access control to interfaces and data. This is crucial for machine services."

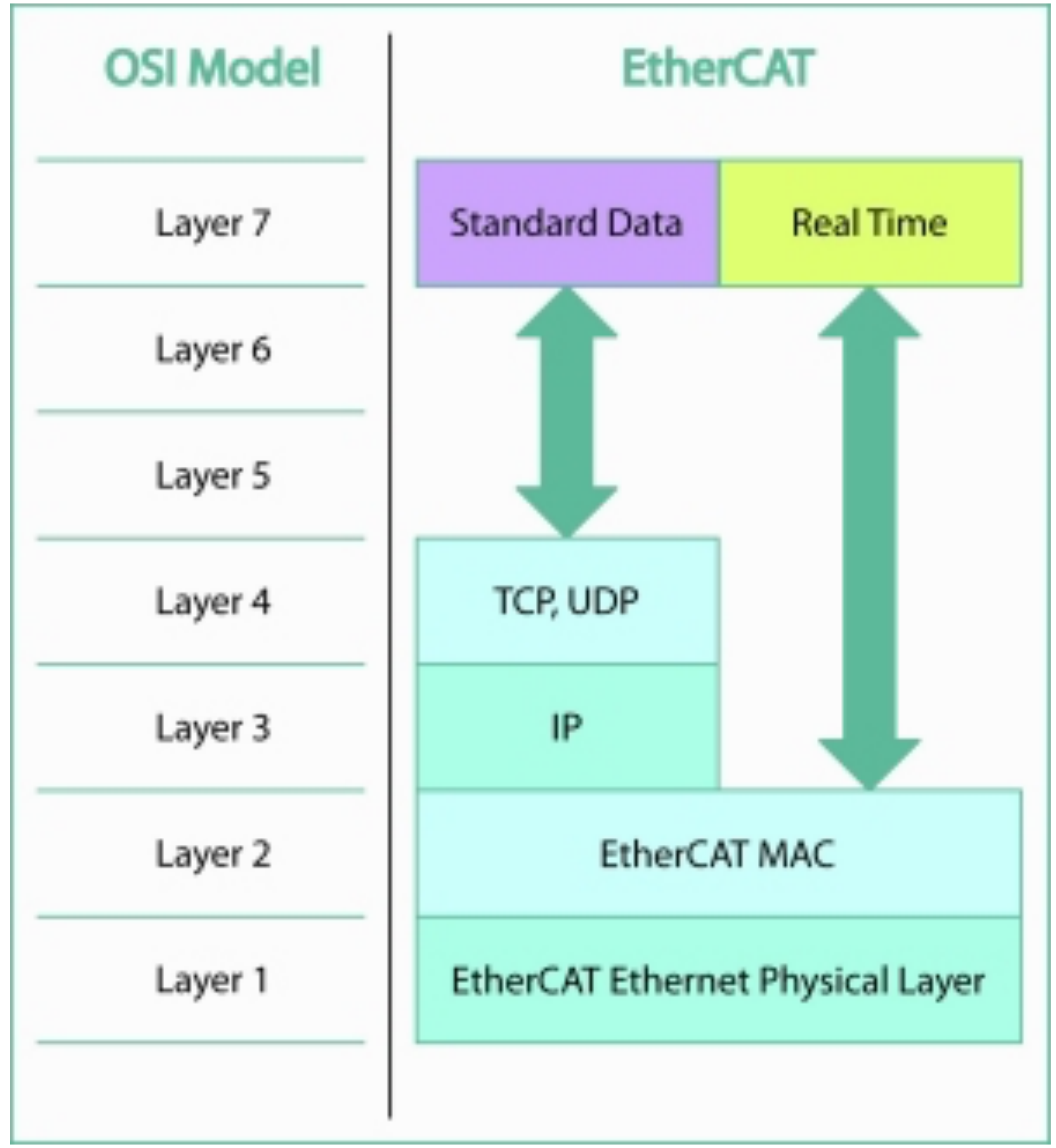
The confusion surrounding these technologies stems from the fact that both are promoted as protocols enabling interoperability between devices, machines, and systems.

He adds that IIC members such a General Electric, Cisco, Microsoft, Oracle, and Siemens are "keen to use OPC UA" and that he is "not aware of any controllers or field devices in the automation sector that have implemented DDS" to date. However, discussions are currently underway to connect the two technologies. A meeting was recently held in Berlin "to clarify whether OPC UA should be recognized via the DDS transport layer," Hoppe says. Even though the market hasn't yet requested this connection, "experts are working on a gateway between OPC UA and DDS like we did with Sercos and EtherCAT. After all, this makes sense in order to integrate any DDS devices with OPC UA into the worldwide IoT community."

ETHERCAT

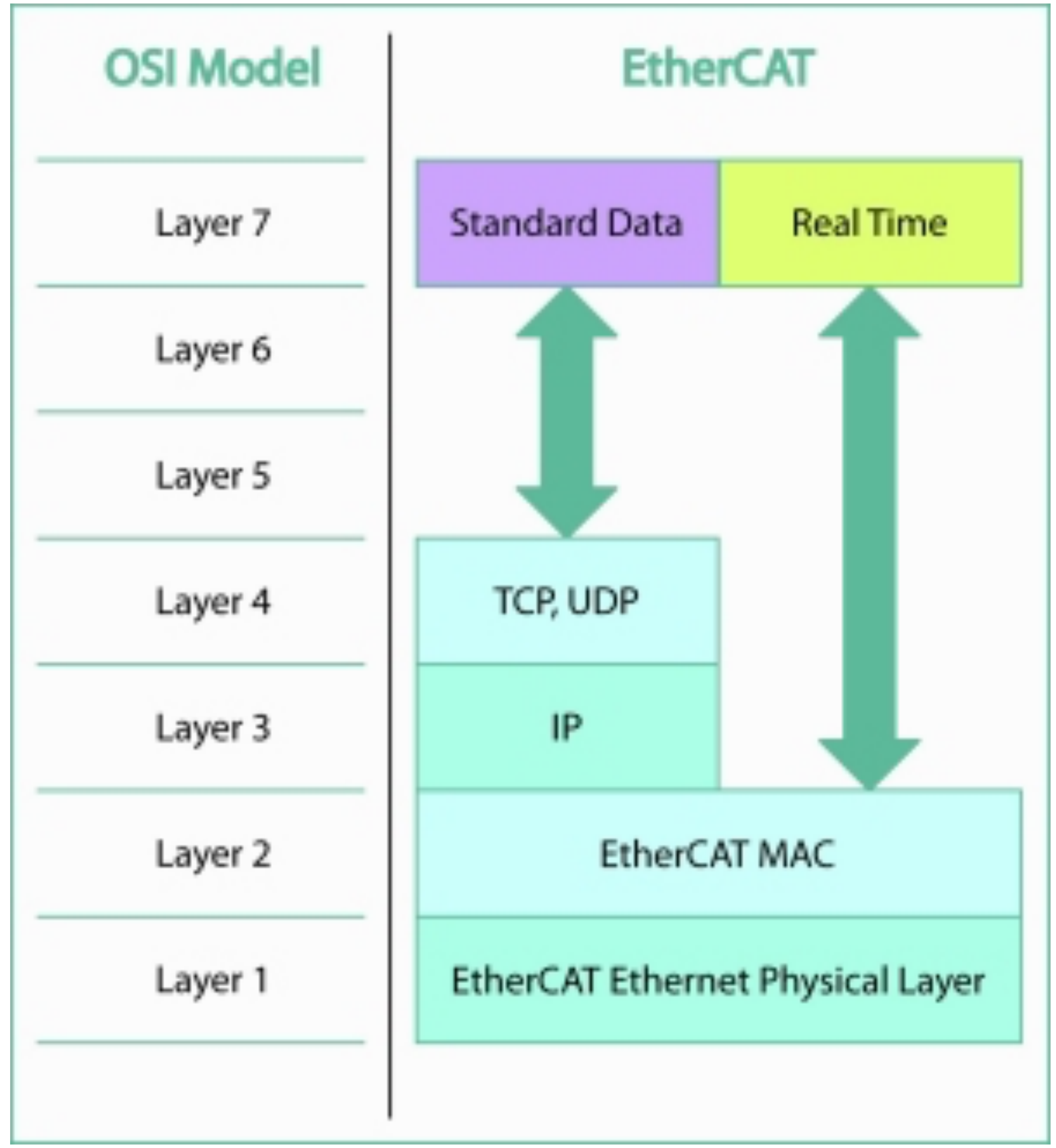
EtherCAT - Ethernet for Control Automation Technology - is an **Ethernet-based fieldbus system**.

The protocol is standardised in IEC 61158 and is suitable for both hard and soft real-time requirements in automation technology.



ETHERCAT

The goal during development of EtherCAT was to apply Ethernet for automation applications requiring short data update times (also called cycle times; ≤ 100 μs) with low communication jitter (for precise synchronisation purposes; ≤ 1 μs) and reduced hardware costs.



OSI Model Refresher

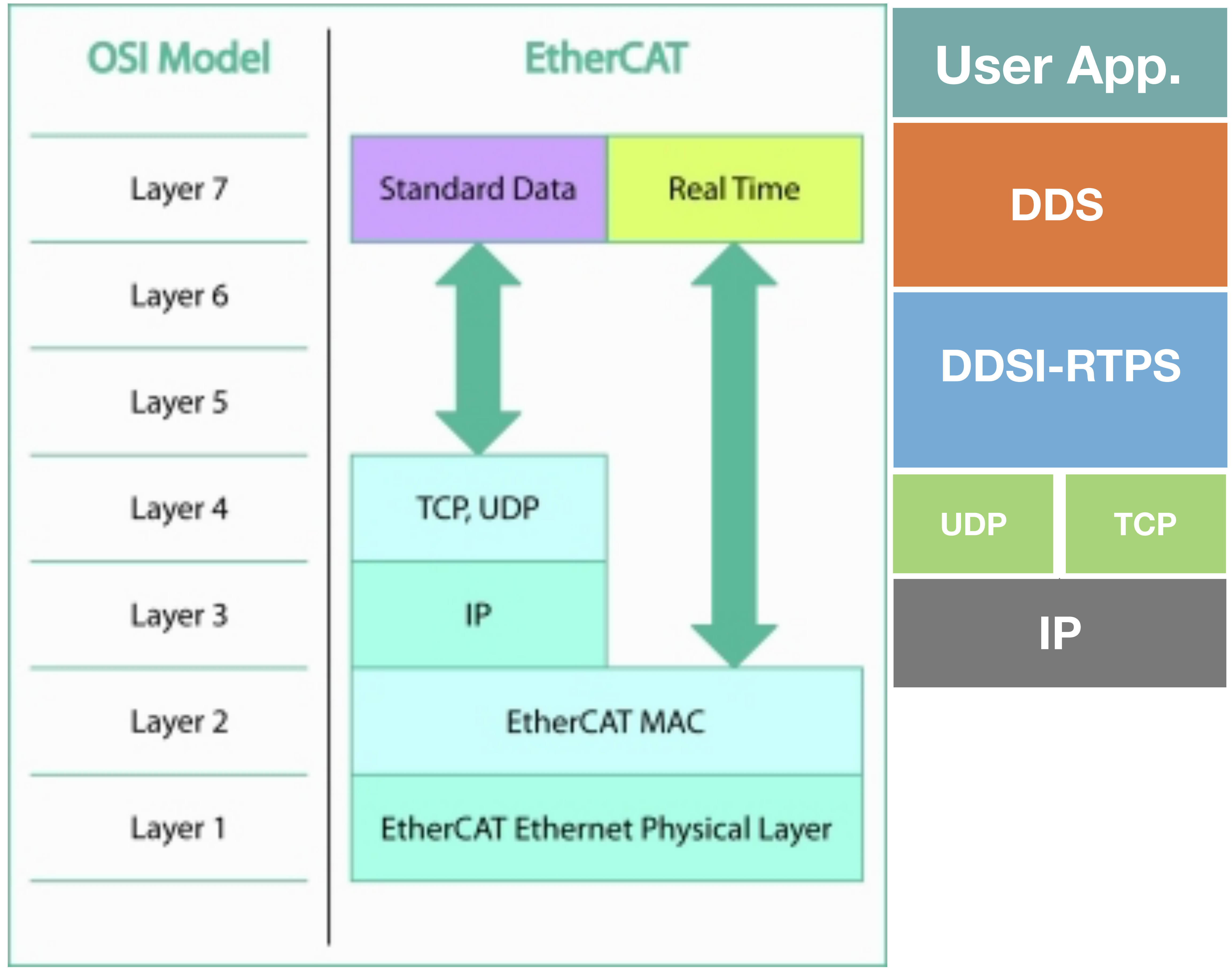
Notice that not all communication stack follow literally the OSI Model. As an example the IP stack deviates in many respects.

7. Application	**High-level APIs**, including resource sharing, remote file access, directory services and virtual terminals
6. Presentation	**Translation of data between a networking service and an application**; including **character encoding**, data compression and **encryption/decryption**
5. Session	Managing **communication sessions**, i.e. continuous exchange of information in the form of multiple back-and-forth transmissions between two nodes
4. Transport	Reliable **transmission of data segments** between points on a network, including segmentation, acknowledgement and multiplexing
3. Network	Structuring and managing a multi-node network, including **addressing**, **routing** and **traffic control**
2. Data Link	Reliable **transmission of data frames between two nodes** connected by a physical layer
1. Physical	**Transmission and reception of raw bit streams** over a physical medium

DDS VS ETHERCAT

DDS provides the features defined as part of the Session and Presentation ISO/OSI Model as part of the DDSI-RTPS and DDS layer and layers and sits on-top the Network layer.

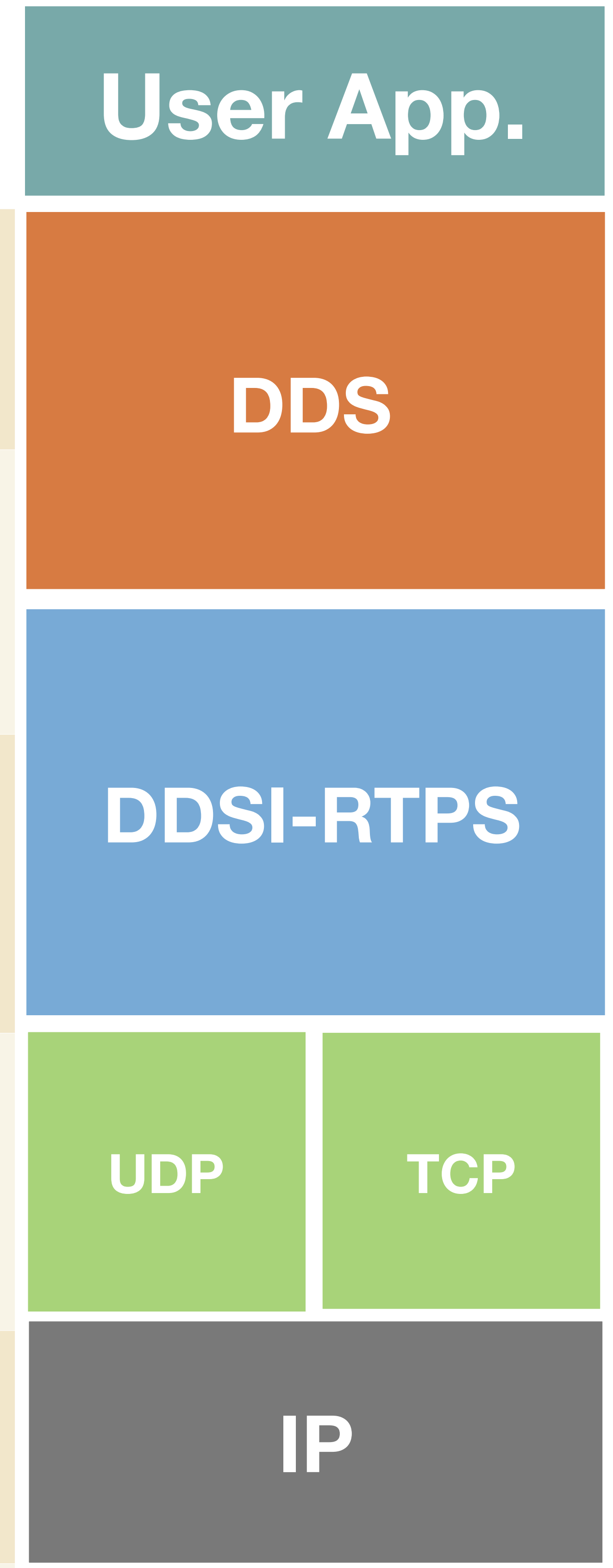
EtherCat is implemented at the Physical and Data Link Layers, this it is way below the stack in terms of abstractions.



DDS IS MORE THAN A MESSAGING PROTOCOL

DDS is in reality a coordination abstraction for distributed computations. It leverages a protocol, namely DDSI-RTPS but it is far more than a protocol.

7. Application	**High-level APIs**, including resource sharing, remote file access, directory services and virtual terminals
6. Presentation	**Translation of data between a networking service and an application**; including **character encoding**, data compression and **encryption/ decryption**
5. Session	Managing **communication sessions**, i.e. continuous exchange of information in the form of multiple back-and-forth transmissions between two nodes
4. Transport	Reliable **transmission of data segments** between points on a network, including segmentation, acknowledgement and multiplexing
3. Network	Structuring and managing a multi-node network, including **addressing**, **routing** and **traffic control**
2. Data Link	Reliable **transmission of data frames between two nodes** connected by a physical layer
1. Physical	**Transmission and reception of raw bit streams** over a physical medium



DDS ISN'T JUST FOR THE EDGE

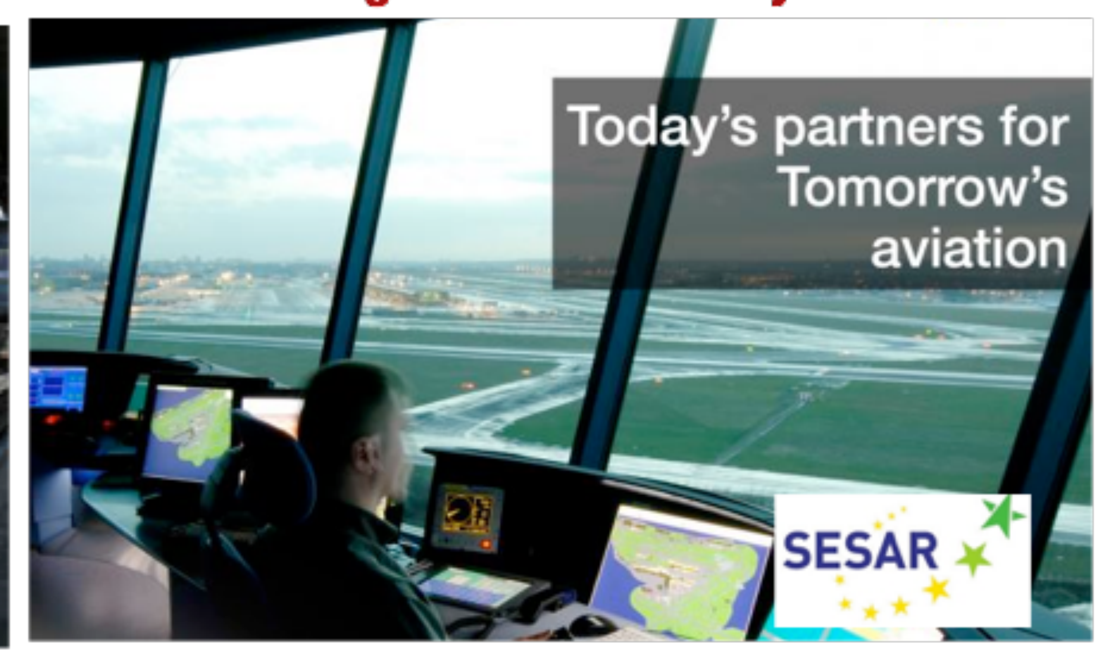
DDS integrates at European Scale in Air Traffic Control Centres.

DDS is at the foundation of several large scale systems such as Smart Cities, Transportation Management Systems, Energy Production, Ground Satellite Control Centres, Connected Vehicles etc.

Smart Cities

Large Scale SCADA Systems

Train Control Systems



Air Traffic Control & Management

DDS in IIoT

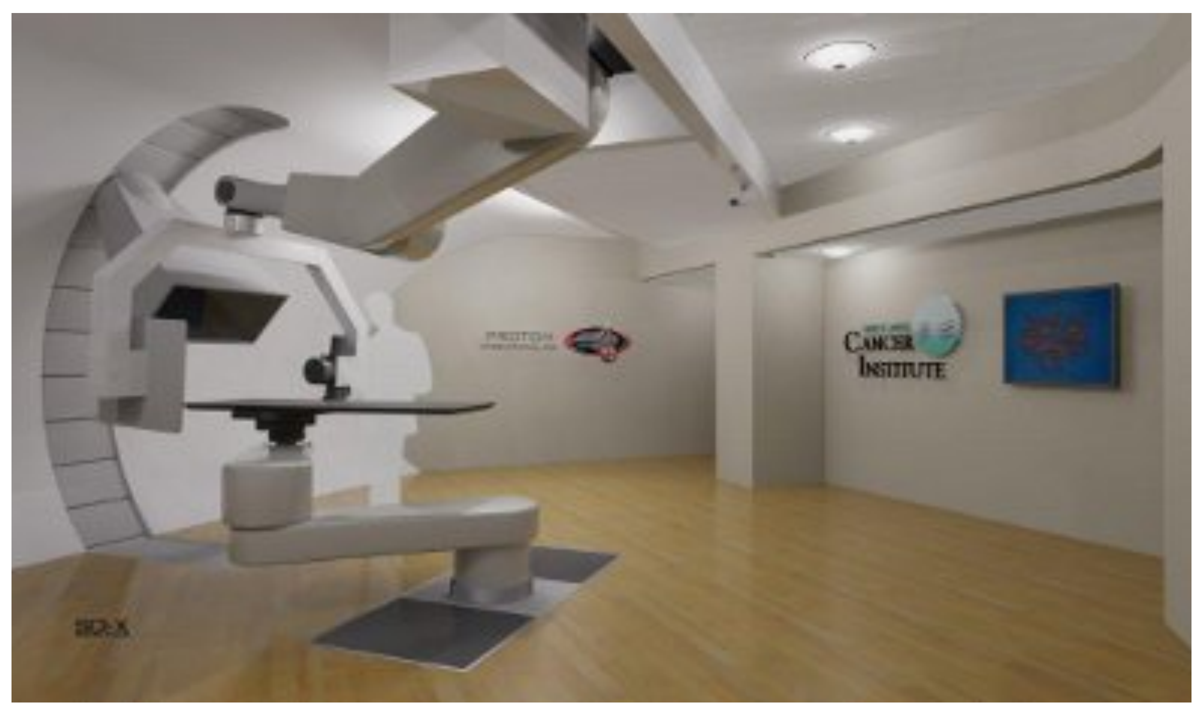
SOME DDS DEPLOYMENTS IN IOT/IIOT

Agricultural Vehicle Systems

Large Scale SCADA Systems

Smart Cities

Train Control Systems



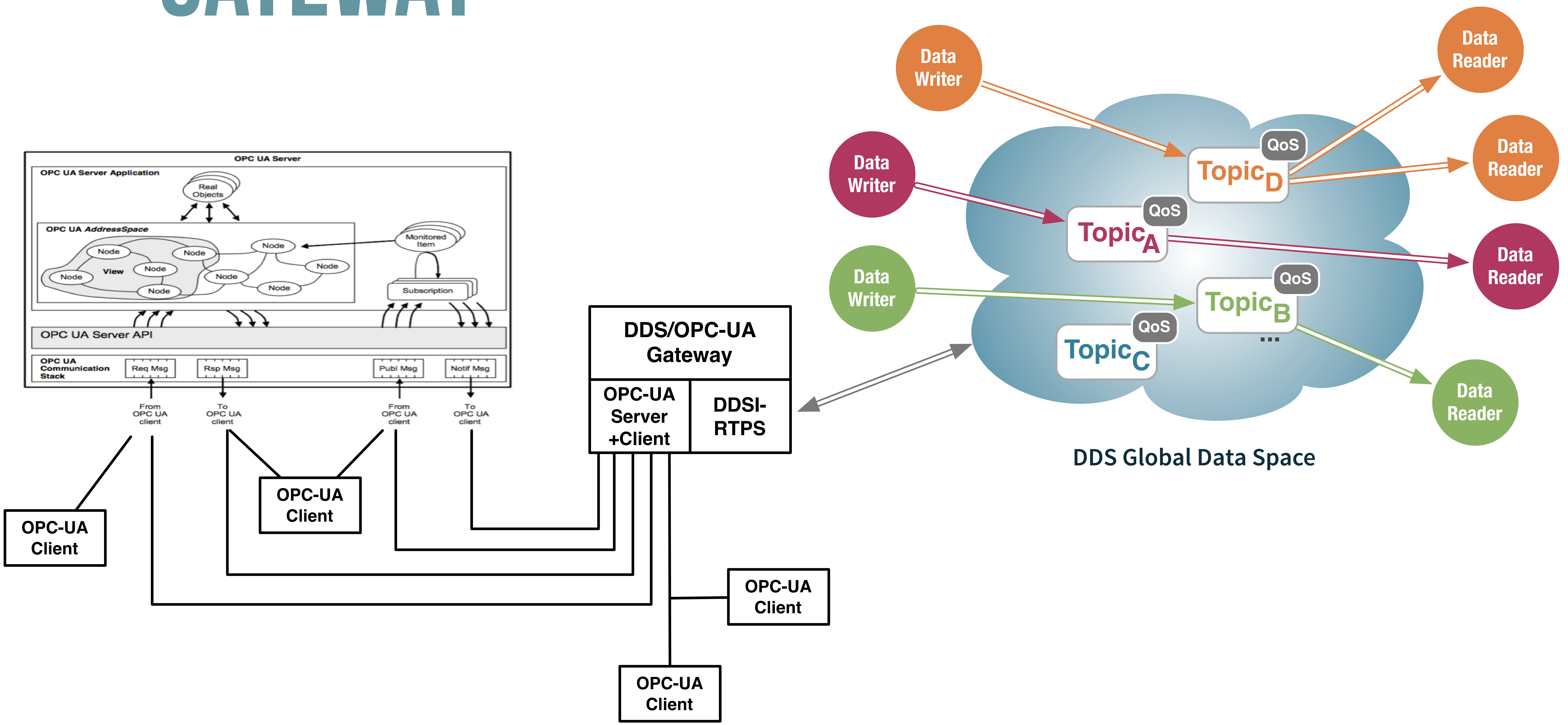
Complex Medical Devices

High Frequency Auto-Trading

DDS / OPC-UA Interworking

OPC-UA/DDS GATEWAY

Upcoming OMG standard defining a gateway to automatically bridge data from OPC-UA to DDS and vice-versa.



Concluding Remarks

SUMMING UP

DDS and OPC UA address a similar problem in a very different way.

DDS was designed since its inception to support GIG-like systems, thus naturally matches IoT/IIoT requirements.

OPC has emerged from the Automation industry, thus its ecosystem is its biggest strength.