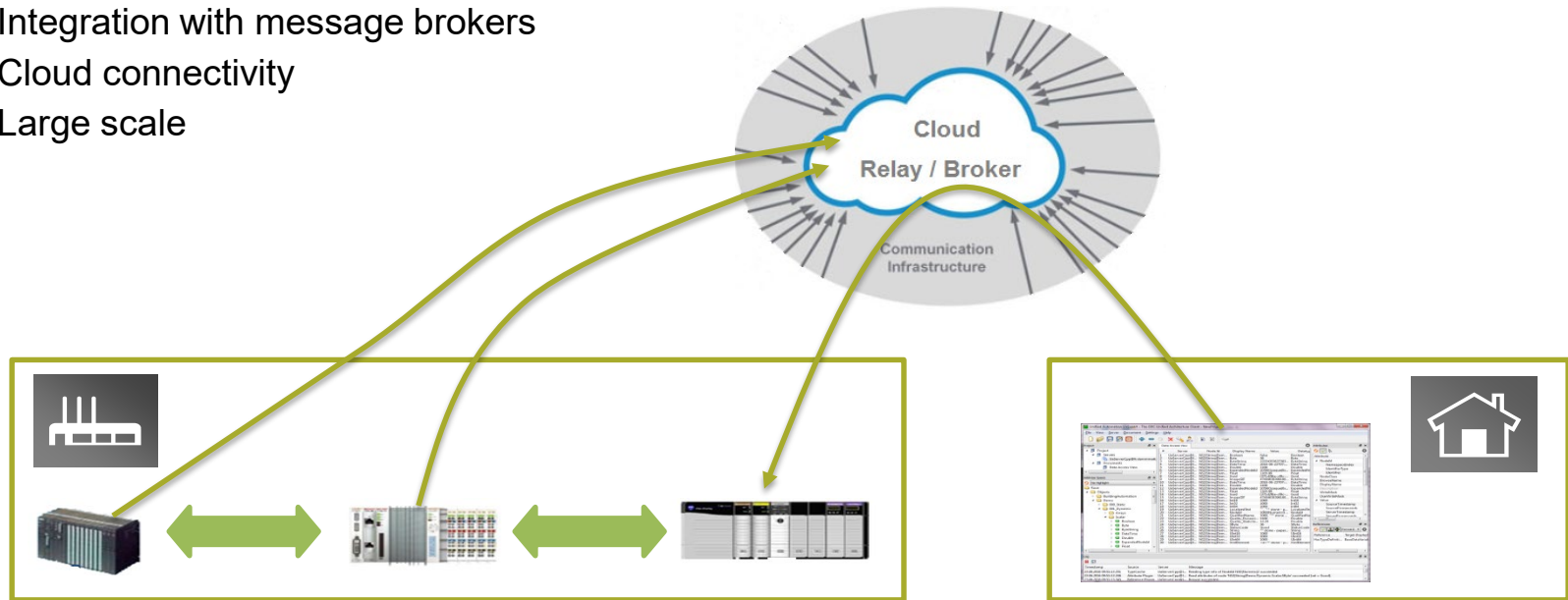


# OPC Unified Architecture

OPC UA PubSub

# PubSub Use Cases

- ▶ Publisher and Subscriber (Clients and Servers) behind firewalls (Message broker is Relay)
- ▶ Controller to controller communication
- ▶ Integration with message brokers
- ▶ Cloud connectivity
- ▶ Large scale



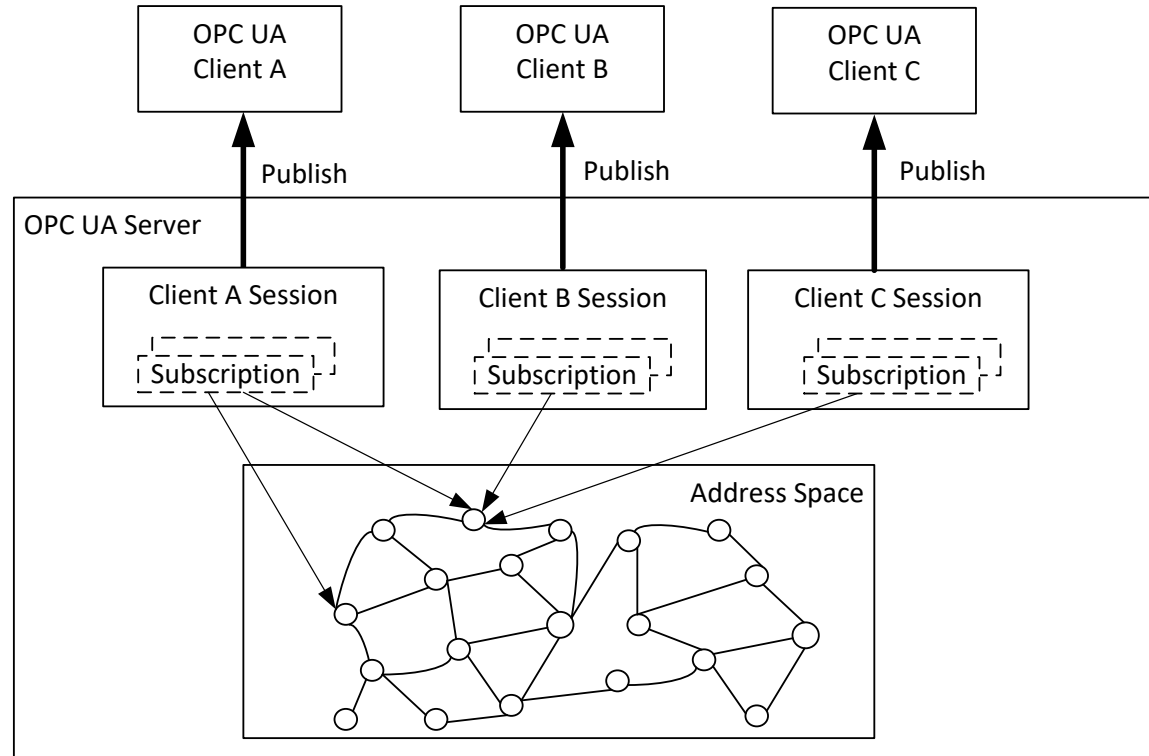
# Subscription in OPC UA Client / Server Model

## Subscription in C/S

- > Selection of data and events streams per client
- > Configuration and consumption of streams in band
- > Individual configuration per client
- > Reliable, one time data and event delivery

## Constrain

- > Model does not scale



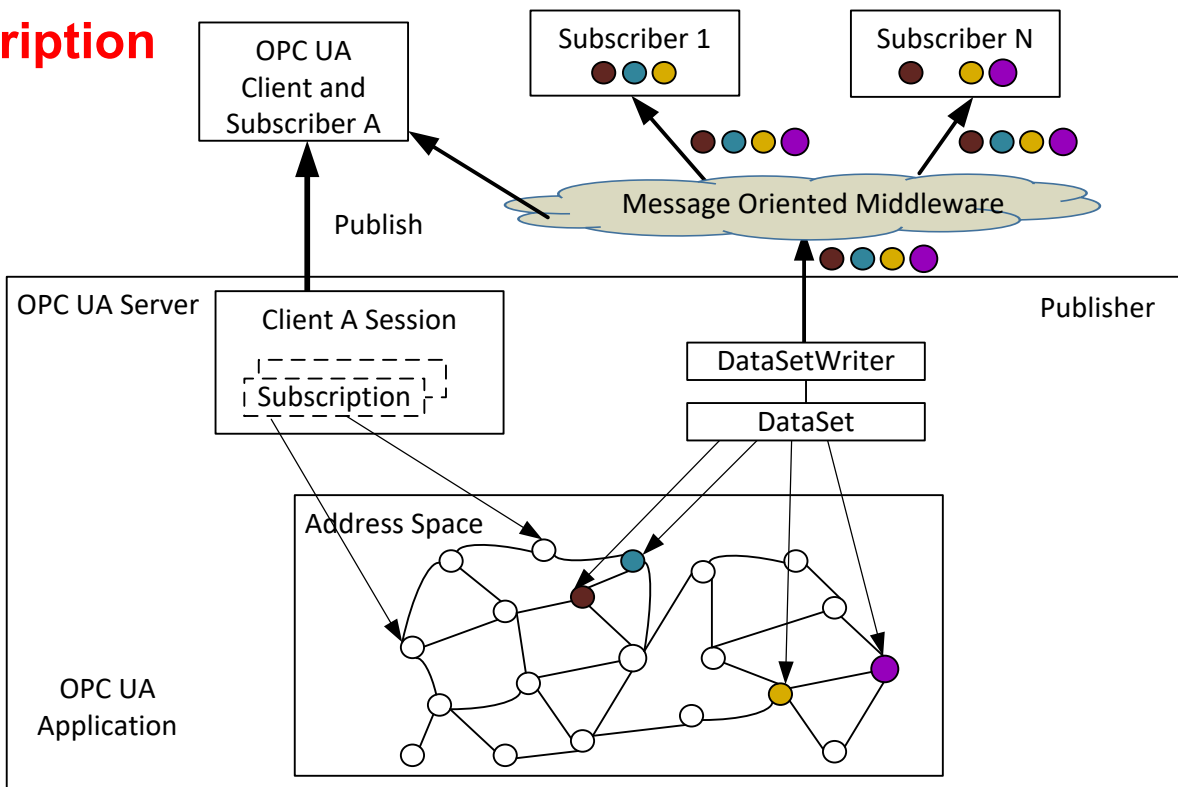
# Client / Server vs. Publish / Subscribe

## Pub-Sub = optimized Subscription

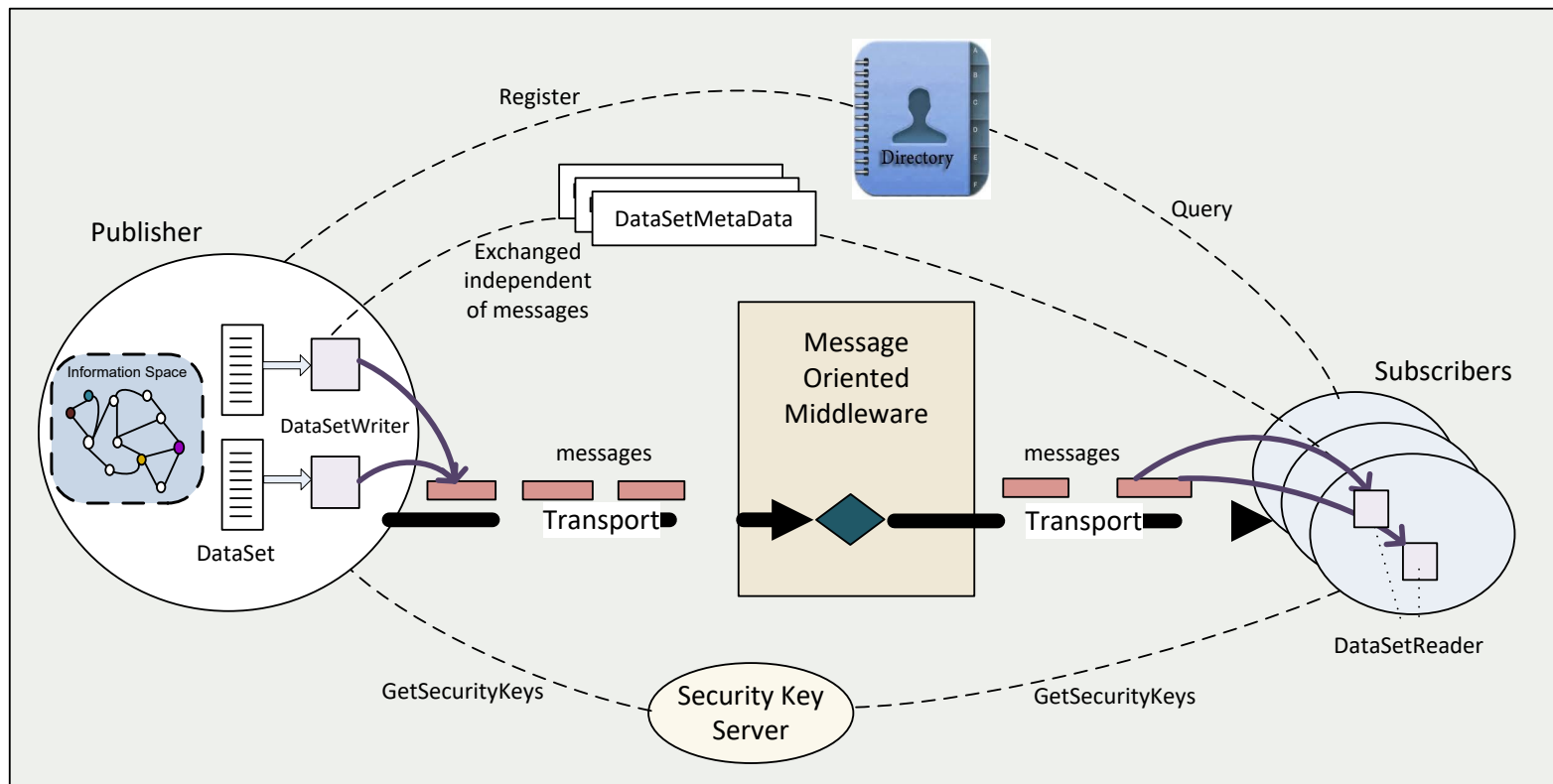
- > Best effort high speed data streaming (UDP)
- > Real-time with TSN
- > Cloud connectivity with AMQP and MQTT
- > Offloading of message distribution to broker

## Constraints

- > Only preconfigured data and event streaming
- > Configuration requires Client-Server
- > Load moved to Subscriber (UDP multicast)

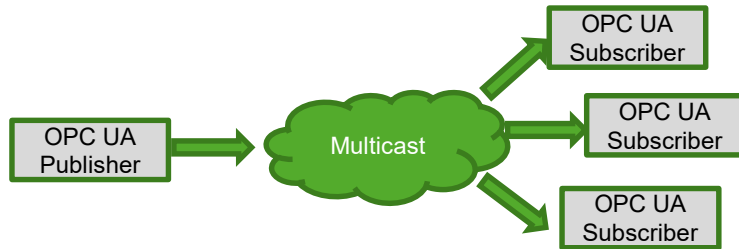


# OPC UA Publish / Subscribe – Big Picture



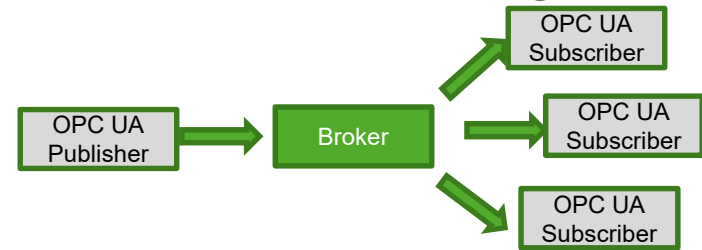
# Protocol Options

## Peer-to-peer Binding



- ▶ UADP Messages over UDP
  - Ethernet Frame, IP and UDP header
  - Standard Port 4840
  - UADP Message Mapping
- ▶ UADP Messages over Ethernet Layer 2
  - Ethernet Frame header with EtherType 0xB62C
  - UADP Message Mapping

## Broker Binding

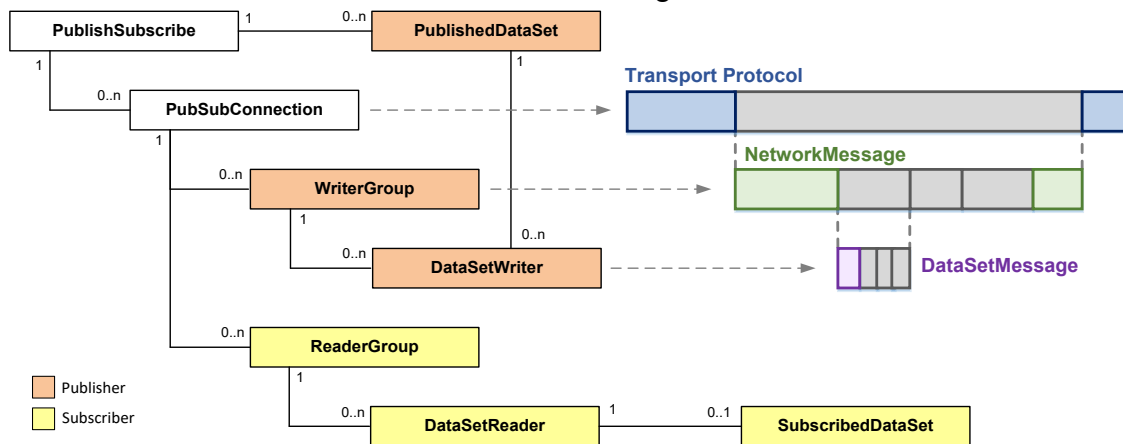


- ▶ JSON Messages over AMQP
- ▶ UADP Messages over AMQP
- ▶ JSON Messages over MQTT
- ▶ UADP Messages over MQTT

# What Information gets Published?

## Conceptual Model and Configuration Model

Client / Server Model can be used to configure Publisher / Subscriber



### PubSubConnection:

Network addressing setting

Protocol selection

### WriterGroup:

Combines several DataSet Messages to one Network Message

Timing settings for the creation of Network Messages

Security Settings (Security managed per group)

MessageRepeatCount (Datagram)

- How often one message is send out

### DataSet:

List of named data values (DataSet Fields)

### DataSetMetaData:

Message contract between Publisher and Subscriber

- Meta data for data values (Name, DataType, Properties)
- ConfigurationVersion (if configuration changes)
- DataSetClassId (Standardized DataSets)

### Published DataSet:

Configuration of data acquisition

- List of variables
- Event filter
- Application-specific messages

DataSetMetaData

### DataSetWriter:

Content Mask

- Status (per value / message / none)
  - Timestamp (per value / message / none)
  - Raw values (fixed size without type information)
- Encoding of Messages (JSON or UADP)

KeyFrameCount

- Cyclic or exception-based communication

RequestedDeliveryGuarantee (Broker)

# NetworkMessage with DataSetMessages

## UADP Message Mapping

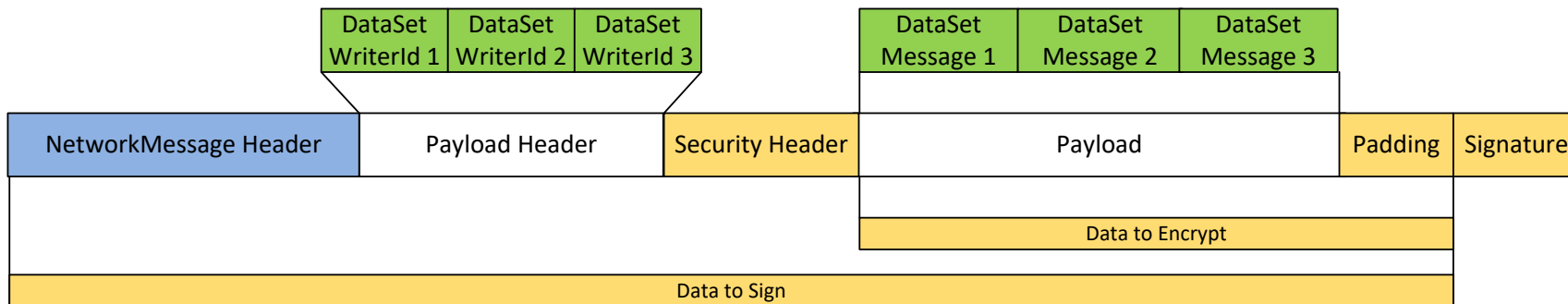
UA Binary encoded  
Message security

## NetworkMessage

Contains list of DataSetMessages (Payload)  
Payload can be encrypted

## DataSetMessage

Data in encrypted Payload  
DataSetWriterId in unencrypted header



## JSON Message Mapping

Has similar layout of information  
in JSON encoding  
No message security



# OPC UA Publish / Subscribe Security

## > Client-Server Security

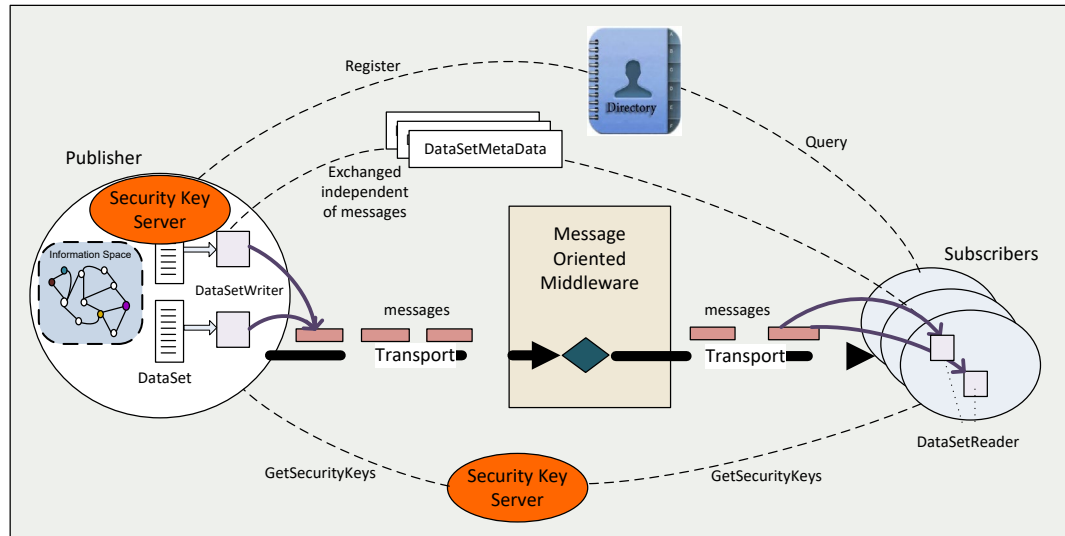
- > PKI and asymmetric algorithms to exchange session keys
- > Session keys are used for communication with symmetric algorithms
- > Session keys are frequently rotated

## > Pub-Sub Security (end to end security)

- > Session keys must be shared between Publishers and Subscribers
- > Keys are managed for a security group
- > Messages are sent in the context of a security group
- > Key distribution is done with OPC UA Client-Server security
- > Authentication and Authorization during access to security group at key server

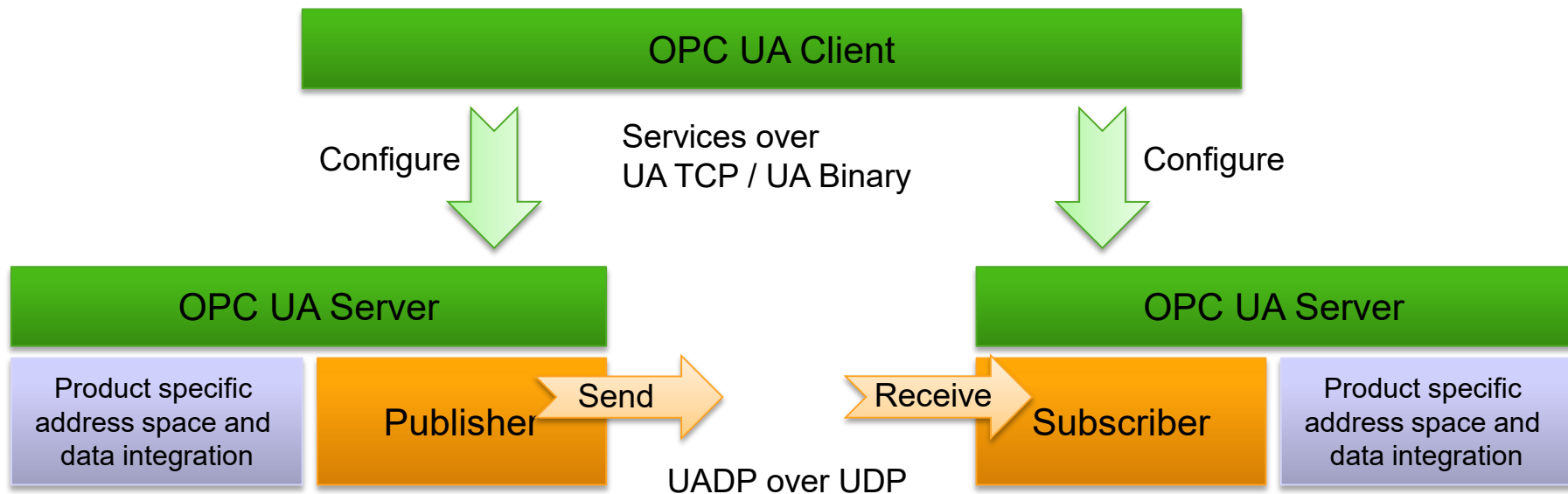
## Options:

- > Publisher is Key Server
- > Central Key Server



# Configured Controller to Controller Communication

- ▶ Existing OPC UA Server can be extended
- ▶ Configuration through OPC UA Clients



# Controller to Controller Real-time Communication

- ▶ UADP over UDP provides
  - Thin and efficient protocol stack for message handling
  - Allows cyclic data exchange
  - Base for device side real-time handling
- ▶ Standard Ethernet is not real-time capable today
- ▶ TSN (Time Sensitive Network) can solve this
  - TSN work in IEEE 802 working group – will be part of standard Ethernet
  - IEEE 802.1AS-Rev/D2.0 Timing & synchronization for time sensitive applications
  - IEEE 802.1Qbv Enhancement for scheduled traffic
  - IEEE 802.1CB Frame Replication and Elimination for Reliability
  - IEEE 802.1Qbu Frame preemption
  - Directly benefit from Ethernet speed enhancements (Gb+)
- ▶ Open Issues to be solved
  - System wide configuration of endpoints and switches

# Conclusion on OPC UA PubSub

## PubSub offers

- Large Scale: Small publisher for large amount of subscribers
- Integration into existing message infrastructure
- Cloud connectivity
- Controller-to-Controller connectivity

## Characteristics

- Different protocol options
  - Broker: AMQP and MQTT
  - Broker-less: UDP or Ethernet Layer 2 (with TSN)
- Different encodings
  - UADP: High performance and message-based security
  - JSON: Interoperable to OPC-UA-unaware AMPQ/MQTT applications
- Different settings
  - Exception-based or cyclic
  - QoS settings for broker, repeating messages for broker-less

## Options for Companion Specifications

- Reference to OPC UA profiles to require specific PubSub settings
- DataSetClassId for standardized DataSets