OPC-UA

OPC-UA - Industrial communication protocol standardized in the IEC 62541 series.

- asynchronous protocol built upon TCP, HTTP, or SOAP

OPC-UA information model is made up of nodes and ifluences between nodes.
Every node has unique node ID

* open62541 Implements the OPC-UA binary protocol stack as well as a client and server SDK.

→ Video lecture - Chikki laptop - NiVi - Unified automation - OPC UA Webnar.
Modelling Concepts

allow expose information - object oriented way.
- methods - execute commands.

Information as NODE in address space.
attributes of NODE - NodeID
NodeClass
BrowseName
DisplayName
Localized Text
} common to all Node

Asynchronous - does not require a constant bit rate.
eg. file transfer, email and world wide web

Synchronous - eg. real time streaming - IP telephony, video conferencing.

OPC-UA → platform independent

Security features - data point level
Scalable - to small devices to big devices with many access points

BASE LAYER
Information Centric Layered Architecture

Base class - OPC-UA

NODES AND REFERENCES - 8 node classes
class - extended
Base node - object - object types have attributes and properties by which we can define behaviour.

Nodes are interconnected with references (describe relation b/w nodes)

Companion Model - describes the domain information - WHAT
OPC-UA - define HOW

** OPC-UA Information Model

Object oriented concept - describes any kind of data.
Information - input data and meta data

∴ Consumer not only gets the value sent also the type of data the publisher is sending

Base Model and Modeling Rules - defined by OPC Foundation Standards.

→ Other organisation use this base model and modelling to describe their own information.
(companion specification)

2 Model of OPC-UA Transportation

Client/Server

direct connection.

Publisher / Subscriber.

pub-talk and all subscribes listening
No direct sub-connection
we don't know if all subs have received the information.

pub - upload the same info and all others are only listening.

CLIENT / SERVER COMMUNICATION
→ Connection - session for peer to peer communication.
safe transport and ack. for every message
private connection.

* Notify on change feature

Disadvantage - 1. Resource consuming - because server treats every connection individually.

Good for → cloud ~~corpo~~ communication
→ controller controller communication

SERVICES
36 services.

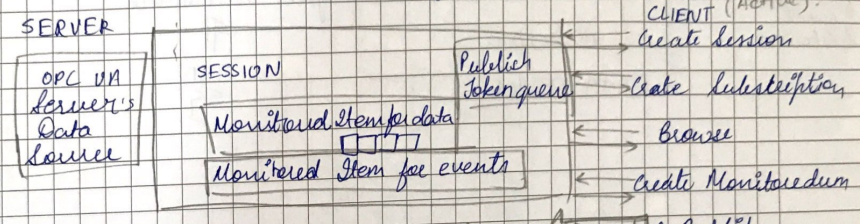
SERVICES - protocol independent

1. Secure Channel Service Set
2. Session Service Set
3. Node Management Service Set
4. ** Attribute Service set (read/write)
5. Mod Monitored item service set - service set to notify on change
6. Method Service Set (invoke)
7. Subscription Service Set

↓

Address space API to access the information in the server

Publish Service - Notify on change.



- Client start the request for session
- Server responds when session is successfully created.
- Client → subscription
- Client → browse the information to pick and choose the information he is interested in
- Transfer only when data is changed → reduce BW

Read write.

Publish/subscribe communication use case - synchronous comm. small data of fixed data

Advantage - very low resource consumption

We publish and forget about the data - publisher doesn't know if any subscribe it reading it or not.

PROTOCOLS - Client/Server

Layers → Encoding data → UA Binary, UA XML (web)

↓

Message Security → UA Secure Communication

↓

Transport → UA TCP, [SOAP 1.2, HTTP (web)]

↓

Transport Security

OPC-UA - Communication Framework