



Industrial Automation
Automation Industrielle
Industrielle Automation



4 Access to devices

4.3 OPC (Open Process Control formerly OLE for Process Control)

4.3.1 Common elements

Prof. Dr. H. Kirrmann

Executive Summary

OPC is a standard, manufacturer-independent programming interface through which an automation application client such as a human interface can access the plant data coming from remote devices, such as programmable logic controllers, field bus devices or real-time databases.

To that effect, the manufacturer of automation devices supplies an OPC server that runs on a PC, which communicates with its devices through a proprietary protocol. An OPC Server can manage several devices of the same type. Several servers can run in parallel and each server can be accessed by several clients, which run on the same PC or in the same network. All OPC servers present the process variables in the same format to their clients as a uniform interface.

This interface consists of a set of commands collected in a software library (DLL) that can be called by client applications written in Visual Basic, C# or other Microsoft programming languages (even Excel) and which access the OPC servers.

The OPC library allows in particular to read and write process variables, read alarms and events and acknowledge alarms, and retrieve historical data from data bases according to several criteria.

Automation platforms such as ABB's 800XA platform act as OPC clients to collect data from PLCs or databases through third-party OPC servers. Several automation platforms act themselves as an OPC server to publish their data, events and historical data.

OPC is the preferred connectivity for 78% of MES, 75% of HMI / SCADA, 68% of DCS / PLC and 53% of ERP /Enterprise system level applications (according to Arc Advisory Group, 2004)"

keep on reading even if you are not an executive....

OPC Common

Overview: usage and specifications

OPC as an integration tool

Clients and Servers: configuration

Automation and Custom Interface

OPC Data Access

Overview: Browsing the server

Objects, Types and properties

Communication model

Simple Programming Example

Standard and components

OPC Alarms and Events Specification

Overview: definitions and objects

Events

Alarm Conditions

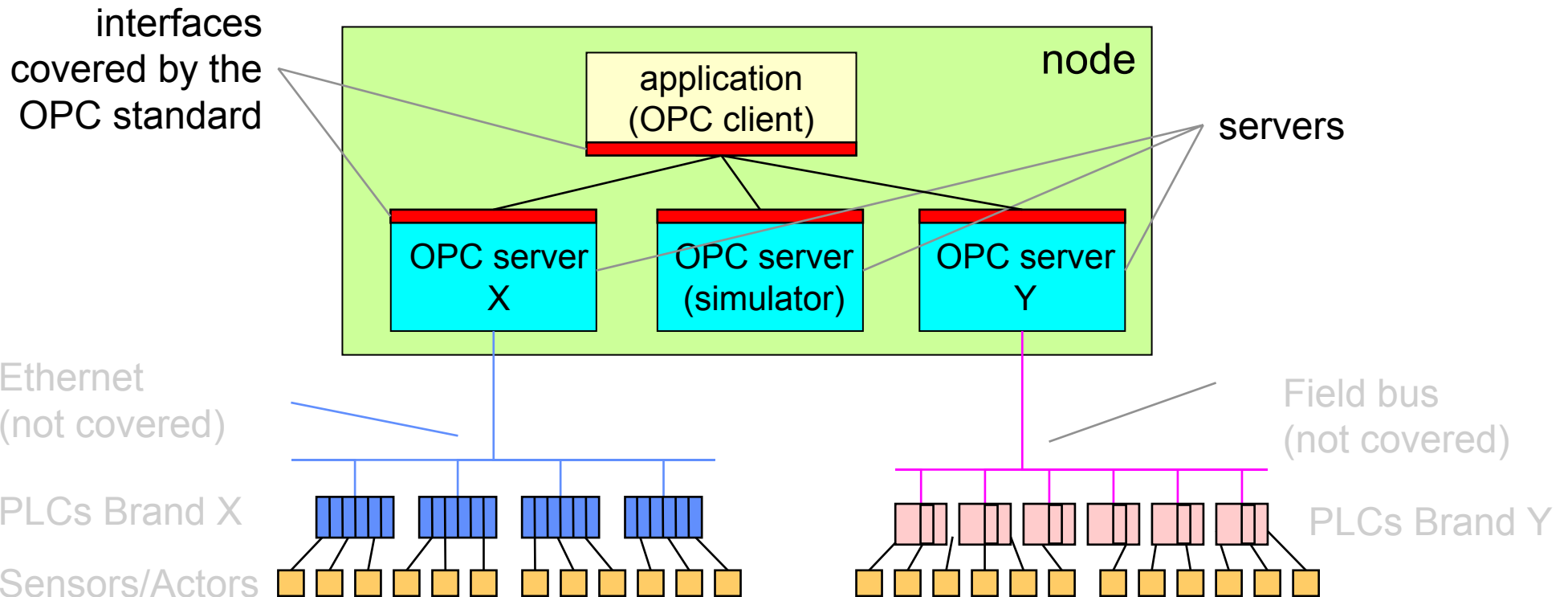
Automation Interface

OPC Historical Data Specification

Overview

What is OPC ?

OPC (formerly: "OLE¹ for Process Control", now: "Open Process Control") is an industry standard set up by the *OPC Foundation* (<http://www.opcfoundation.org/>) specifying the software interface (objects, methods) to a server that collects data produced by field devices and programmable logic controllers.



1) OLE (Object Linking and Embedding) is a Microsoft technology for connecting software components. It has since been extended by the COM / DCOM technology. It corresponds to Java Beans.

Before OPC

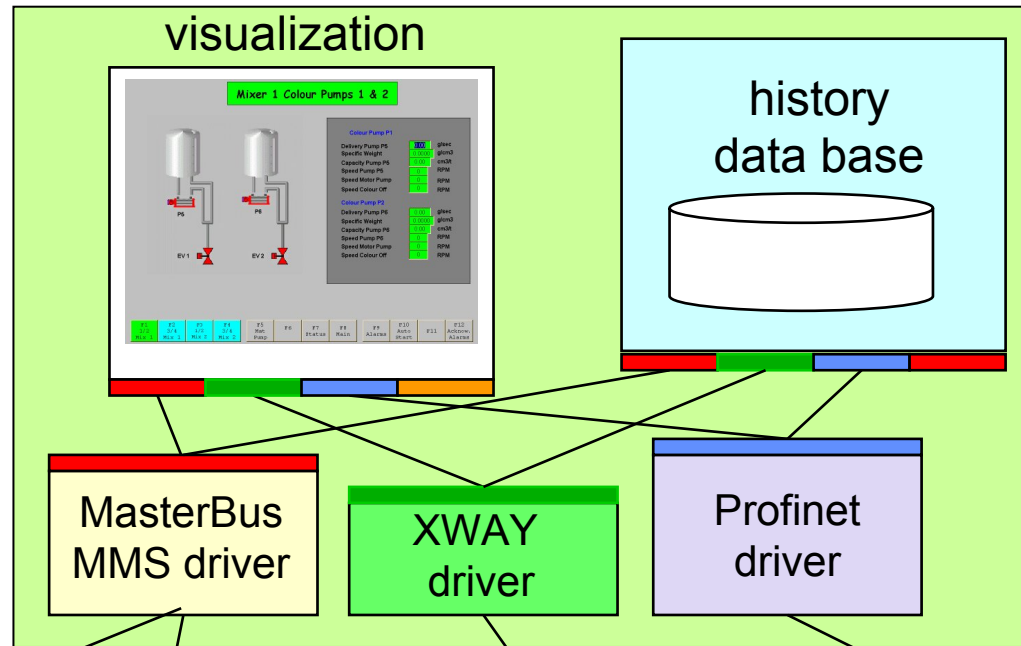
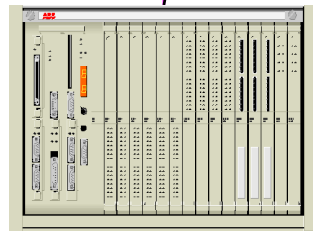


ABB PLCs



Télémécanique PLCs



Siemens PLCs

With OPC: ABB Operator Workplace Connection

application software is written independently from the type of controller

the drivers still exist, but the clients do not see them anymore

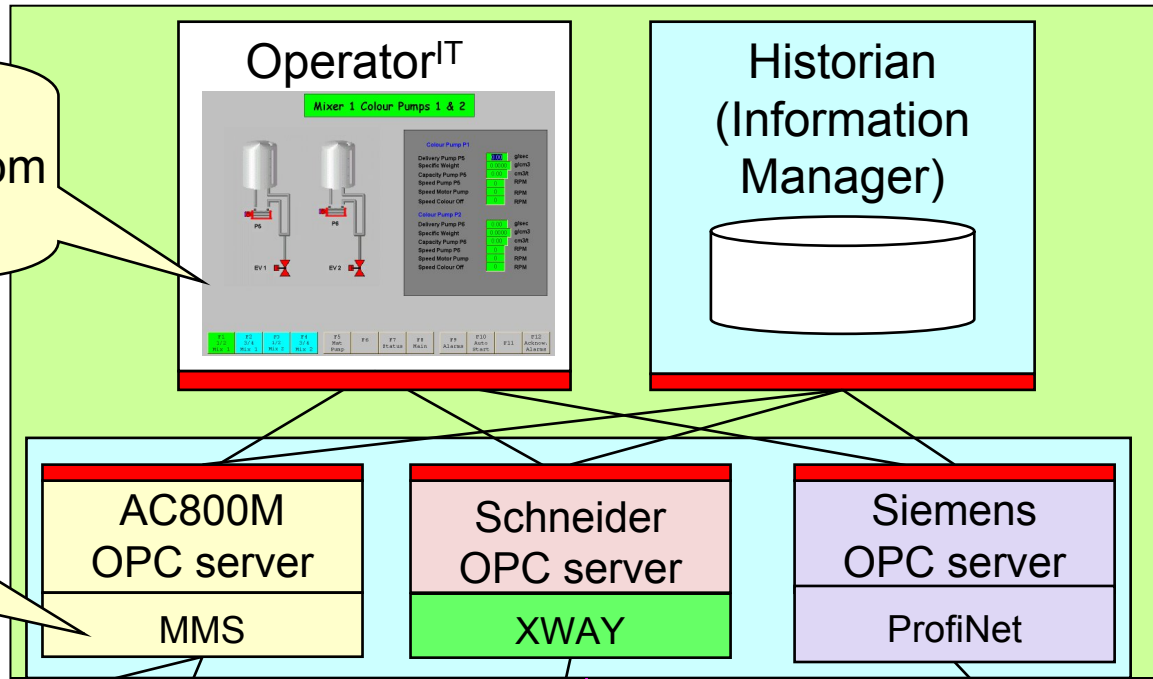


ABB AC800M



Télémécanique TSX



Siemens S7

Importance

OPC is the greatest improvement in automation since IEC 61131.

OPC is supported by the OPC foundation (<http://www.opcfoundation.org/>)

More than 150 vendors offer OPC servers to connect their PLCs, field bus devices, displays and visualization systems.

OPC is also used for data exchange between applications and for accessing databases

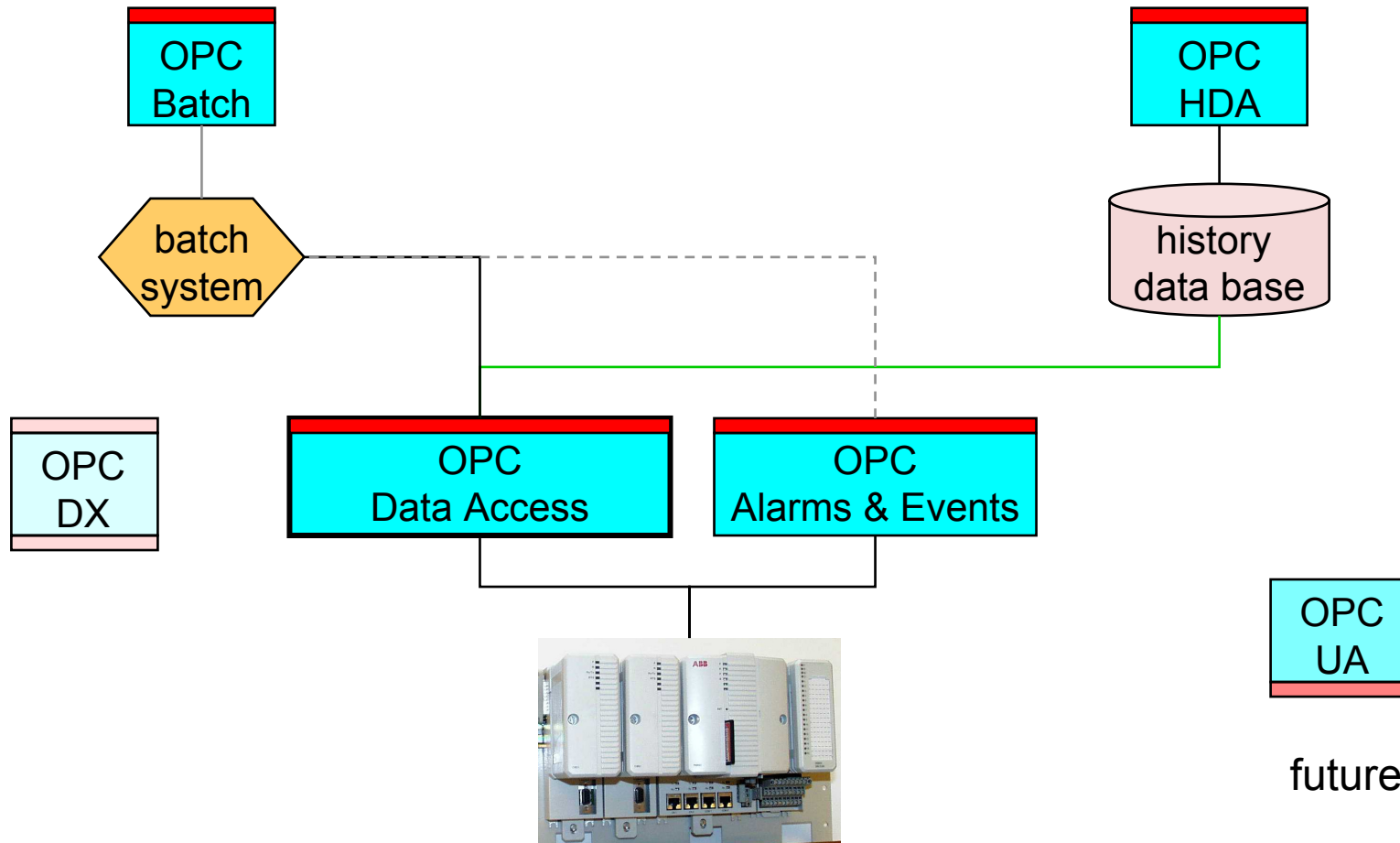
OPC is available as DLL for Automation Interface (Visual Basic,..) and Custom (C++,..)

OPC consists of three major components:

- 1) OPC - DA = Data Access (widespread, mature)
- 2) OPC - AE = Alarms and Events (not yet much used)
- 3) OPC - HDA = Historical Data Access (seldom used)

... and some profiles (batch,...)

The main OPC Specifications



Specification 1: OPC DA for Data Access

Process variables describe the plant's state, they are generated by the sensors or calculated in the programmable logic controllers (PLCs).

Process variables can be sent upon a change, on demand or when a given time elapsed.

The OPC DA (Data Access) specification addresses collecting Process Variables.

The main clients of OPC DA are visualization and (soft-) control.

The screenshot displays a complex industrial control system interface. At the top, a navigation bar includes tabs for LAC, LAB, HKW, HNA, CPB, and SYS, along with a date and time display (20:02, 21.05.97). Below this is a menu bar (User, Event, Tag, Window, Service, Help) and a search field containing 'GC-LIC-454'. The main interface is divided into several windows:

- REACTOR AREA OVERVIEW:** A central diagram showing two large blue cylindrical reactors connected by a network of pipes and valves. A red box highlights a specific component in the diagram, with an arrow pointing to the 'Batch Overview' window.
- Batch Overview:** A table listing various batches with columns for Batch ID, Lot, Campaign, Status, Batch Area, and Start Time. A table below it shows parameters for a selected batch, including 'dctHex4', 'dctHex4_start_stop', 'dctHex4', 'dctHex4c', 'dctHex4e', 'dctHex4f', 'dctHex4g', 'dctHex4h', 'dctHex4i', 'dctHex4j', 'dctHex4k', 'dctHex4l', 'dctHex4m', 'dctHex4n', 'dctHex4o', 'dctHex4p', 'dctHex4q', 'dctHex4r', 'dctHex4s', 'dctHex4t', 'dctHex4u', 'dctHex4v', 'dctHex4w', 'dctHex4x', 'dctHex4y', 'dctHex4z', 'dctHex4aa', 'dctHex4ab', 'dctHex4ac', 'dctHex4ad', 'dctHex4ae', 'dctHex4af', 'dctHex4ag', 'dctHex4ah', 'dctHex4ai', 'dctHex4aj', 'dctHex4ak', 'dctHex4al', 'dctHex4am', 'dctHex4an', 'dctHex4ao', 'dctHex4ap', 'dctHex4aq', 'dctHex4ar', 'dctHex4as', 'dctHex4at', 'dctHex4au', 'dctHex4av', 'dctHex4aw', 'dctHex4ax', 'dctHex4ay', 'dctHex4az', 'dctHex4ba', 'dctHex4bb', 'dctHex4bc', 'dctHex4bd', 'dctHex4be', 'dctHex4bf', 'dctHex4bg', 'dctHex4bh', 'dctHex4bi', 'dctHex4bj', 'dctHex4bk', 'dctHex4bl', 'dctHex4bm', 'dctHex4bn', 'dctHex4bo', 'dctHex4bp', 'dctHex4bq', 'dctHex4br', 'dctHex4bs', 'dctHex4bt', 'dctHex4bu', 'dctHex4bv', 'dctHex4bw', 'dctHex4bx', 'dctHex4by', 'dctHex4bz', 'dctHex4ca', 'dctHex4cb', 'dctHex4cc', 'dctHex4cd', 'dctHex4ce', 'dctHex4cf', 'dctHex4cg', 'dctHex4ch', 'dctHex4ci', 'dctHex4cj', 'dctHex4ck', 'dctHex4cl', 'dctHex4cm', 'dctHex4cn', 'dctHex4co', 'dctHex4cp', 'dctHex4cq', 'dctHex4cr', 'dctHex4cs', 'dctHex4ct', 'dctHex4cu', 'dctHex4cv', 'dctHex4cw', 'dctHex4cx', 'dctHex4cy', 'dctHex4cz', 'dctHex4da', 'dctHex4db', 'dctHex4dc', 'dctHex4dd', 'dctHex4de', 'dctHex4df', 'dctHex4dg', 'dctHex4dh', 'dctHex4di', 'dctHex4dj', 'dctHex4dk', 'dctHex4dl', 'dctHex4dm', 'dctHex4dn', 'dctHex4do', 'dctHex4dp', 'dctHex4dq', 'dctHex4dr', 'dctHex4ds', 'dctHex4dt', 'dctHex4du', 'dctHex4dv', 'dctHex4dw', 'dctHex4dx', 'dctHex4dy', 'dctHex4dz', 'dctHex4ea', 'dctHex4eb', 'dctHex4ec', 'dctHex4ed', 'dctHex4ee', 'dctHex4ef', 'dctHex4eg', 'dctHex4eh', 'dctHex4ei', 'dctHex4ej', 'dctHex4ek', 'dctHex4el', 'dctHex4em', 'dctHex4en', 'dctHex4eo', 'dctHex4ep', 'dctHex4eq', 'dctHex4er', 'dctHex4es', 'dctHex4et', 'dctHex4eu', 'dctHex4ev', 'dctHex4ew', 'dctHex4ex', 'dctHex4ey', 'dctHex4ez', 'dctHex4fa', 'dctHex4fb', 'dctHex4fc', 'dctHex4fd', 'dctHex4fe', 'dctHex4ff', 'dctHex4fg', 'dctHex4fh', 'dctHex4fi', 'dctHex4fj', 'dctHex4fk', 'dctHex4fl', 'dctHex4fm', 'dctHex4fn', 'dctHex4fo', 'dctHex4fp', 'dctHex4fq', 'dctHex4fr', 'dctHex4fs', 'dctHex4ft', 'dctHex4fu', 'dctHex4fv', 'dctHex4fw', 'dctHex4fx', 'dctHex4fy', 'dctHex4fz', 'dctHex4ga', 'dctHex4gb', 'dctHex4gc', 'dctHex4gd', 'dctHex4ge', 'dctHex4gf', 'dctHex4gg', 'dctHex4gh', 'dctHex4gi', 'dctHex4gj', 'dctHex4gk', 'dctHex4gl', 'dctHex4gm', 'dctHex4gn', 'dctHex4go', 'dctHex4gp', 'dctHex4gq', 'dctHex4gr', 'dctHex4gs', 'dctHex4gt', 'dctHex4gu', 'dctHex4gv', 'dctHex4gw', 'dctHex4gx', 'dctHex4gy', 'dctHex4gz', 'dctHex4ha', 'dctHex4hb', 'dctHex4hc', 'dctHex4hd', 'dctHex4he', 'dctHex4hf', 'dctHex4hg', 'dctHex4hh', 'dctHex4hi', 'dctHex4hj', 'dctHex4hk', 'dctHex4hl', 'dctHex4hm', 'dctHex4hn', 'dctHex4ho', 'dctHex4hp', 'dctHex4hq', 'dctHex4hr', 'dctHex4hs', 'dctHex4ht', 'dctHex4hu', 'dctHex4hv', 'dctHex4hw', 'dctHex4hx', 'dctHex4hy', 'dctHex4hz', 'dctHex4ia', 'dctHex4ib', 'dctHex4ic', 'dctHex4id', 'dctHex4ie', 'dctHex4if', 'dctHex4ig', 'dctHex4ih', 'dctHex4ii', 'dctHex4ij', 'dctHex4ik', 'dctHex4il', 'dctHex4im', 'dctHex4in', 'dctHex4io', 'dctHex4ip', 'dctHex4iq', 'dctHex4ir', 'dctHex4is', 'dctHex4it', 'dctHex4iu', 'dctHex4iv', 'dctHex4iw', 'dctHex4ix', 'dctHex4iy', 'dctHex4iz', 'dctHex4ja', 'dctHex4jb', 'dctHex4jc', 'dctHex4jd', 'dctHex4je', 'dctHex4jf', 'dctHex4jg', 'dctHex4jh', 'dctHex4ji', 'dctHex4jj', 'dctHex4jk', 'dctHex4jl', 'dctHex4jm', 'dctHex4jn', 'dctHex4jo', 'dctHex4jp', 'dctHex4jq', 'dctHex4jr', 'dctHex4js', 'dctHex4jt', 'dctHex4ju', 'dctHex4jv', 'dctHex4jw', 'dctHex4jx', 'dctHex4jy', 'dctHex4jz', 'dctHex4ka', 'dctHex4kb', 'dctHex4kc', 'dctHex4kd', 'dctHex4ke', 'dctHex4kf', 'dctHex4kg', 'dctHex4kh', 'dctHex4ki', 'dctHex4kj', 'dctHex4kk', 'dctHex4kl', 'dctHex4km', 'dctHex4kn', 'dctHex4ko', 'dctHex4kp', 'dctHex4kq', 'dctHex4kr', 'dctHex4ks', 'dctHex4kt', 'dctHex4ku', 'dctHex4kv', 'dctHex4kw', 'dctHex4kx', 'dctHex4ky', 'dctHex4kz', 'dctHex4la', 'dctHex4lb', 'dctHex4lc', 'dctHex4ld', 'dctHex4le', 'dctHex4lf', 'dctHex4lg', 'dctHex4lh', 'dctHex4li', 'dctHex4lj', 'dctHex4lk', 'dctHex4ll', 'dctHex4lm', 'dctHex4ln', 'dctHex4lo', 'dctHex4lp', 'dctHex4lq', 'dctHex4lr', 'dctHex4ls', 'dctHex4lt', 'dctHex4lu', 'dctHex4lv', 'dctHex4lw', 'dctHex4lx', 'dctHex4ly', 'dctHex4lz', 'dctHex4ma', 'dctHex4mb', 'dctHex4mc', 'dctHex4md', 'dctHex4me', 'dctHex4mf', 'dctHex4mg', 'dctHex4mh', 'dctHex4mi', 'dctHex4mj', 'dctHex4mk', 'dctHex4ml', 'dctHex4mm', 'dctHex4mn', 'dctHex4mo', 'dctHex4mp', 'dctHex4mq', 'dctHex4mr', 'dctHex4ms', 'dctHex4mt', 'dctHex4mu', 'dctHex4mv', 'dctHex4mw', 'dctHex4mx', 'dctHex4my', 'dctHex4mz', 'dctHex4na', 'dctHex4nb', 'dctHex4nc', 'dctHex4nd', 'dctHex4ne', 'dctHex4nf', 'dctHex4ng', 'dctHex4nh', 'dctHex4ni', 'dctHex4nj', 'dctHex4nk', 'dctHex4nl', 'dctHex4nm', 'dctHex4nn', 'dctHex4no', 'dctHex4np', 'dctHex4nq', 'dctHex4nr', 'dctHex4ns', 'dctHex4nt', 'dctHex4nu', 'dctHex4nv', 'dctHex4nw', 'dctHex4nx', 'dctHex4ny', 'dctHex4nz', 'dctHex4oa', 'dctHex4ob', 'dctHex4oc', 'dctHex4od', 'dctHex4oe', 'dctHex4of', 'dctHex4og', 'dctHex4oh', 'dctHex4oi', 'dctHex4oj', 'dctHex4ok', 'dctHex4ol', 'dctHex4om', 'dctHex4on', 'dctHex4oo', 'dctHex4op', 'dctHex4oq', 'dctHex4or', 'dctHex4os', 'dctHex4ot', 'dctHex4ou', 'dctHex4ov', 'dctHex4ow', 'dctHex4ox', 'dctHex4oy', 'dctHex4oz', 'dctHex4pa', 'dctHex4pb', 'dctHex4pc', 'dctHex4pd', 'dctHex4pe', 'dctHex4pf', 'dctHex4pg', 'dctHex4ph', 'dctHex4pi', 'dctHex4pj', 'dctHex4pk', 'dctHex4pl', 'dctHex4pm', 'dctHex4pn', 'dctHex4po', 'dctHex4pp', 'dctHex4pq', 'dctHex4pr', 'dctHex4ps', 'dctHex4pt', 'dctHex4pu', 'dctHex4pv', 'dctHex4pw', 'dctHex4px', 'dctHex4py', 'dctHex4pz', 'dctHex4qa', 'dctHex4qb', 'dctHex4qc', 'dctHex4qd', 'dctHex4qe', 'dctHex4qf', 'dctHex4qg', 'dctHex4qh', 'dctHex4qi', 'dctHex4qj', 'dctHex4qk', 'dctHex4ql', 'dctHex4qm', 'dctHex4qn', 'dctHex4qo', 'dctHex4qp', 'dctHex4qq', 'dctHex4qr', 'dctHex4qs', 'dctHex4qt', 'dctHex4qu', 'dctHex4qv', 'dctHex4qw', 'dctHex4qx', 'dctHex4qy', 'dctHex4qz', 'dctHex4ra', 'dctHex4rb', 'dctHex4rc', 'dctHex4rd', 'dctHex4re', 'dctHex4rf', 'dctHex4rg', 'dctHex4rh', 'dctHex4ri', 'dctHex4rj', 'dctHex4rk', 'dctHex4rl', 'dctHex4rm', 'dctHex4rn', 'dctHex4ro', 'dctHex4rp', 'dctHex4rq', 'dctHex4rr', 'dctHex4rs', 'dctHex4rt', 'dctHex4ru', 'dctHex4rv', 'dctHex4rw', 'dctHex4rx', 'dctHex4ry', 'dctHex4rz', 'dctHex4sa', 'dctHex4sb', 'dctHex4sc', 'dctHex4sd', 'dctHex4se', 'dctHex4sf', 'dctHex4sg', 'dctHex4sh', 'dctHex4si', 'dctHex4sj', 'dctHex4sk', 'dctHex4sl', 'dctHex4sm', 'dctHex4sn', 'dctHex4so', 'dctHex4sp', 'dctHex4sq', 'dctHex4sr', 'dctHex4ss', 'dctHex4st', 'dctHex4su', 'dctHex4sv', 'dctHex4sw', 'dctHex4sx', 'dctHex4sy', 'dctHex4sz', 'dctHex4ta', 'dctHex4tb', 'dctHex4tc', 'dctHex4td', 'dctHex4te', 'dctHex4tf', 'dctHex4tg', 'dctHex4th', 'dctHex4ti', 'dctHex4tj', 'dctHex4tk', 'dctHex4tl', 'dctHex4tm', 'dctHex4tn', 'dctHex4to', 'dctHex4tp', 'dctHex4tq', 'dctHex4tr', 'dctHex4ts', 'dctHex4tt', 'dctHex4tu', 'dctHex4tv', 'dctHex4tw', 'dctHex4tx', 'dctHex4ty', 'dctHex4tz', 'dctHex4ua', 'dctHex4ub', 'dctHex4uc', 'dctHex4ud', 'dctHex4ue', 'dctHex4uf', 'dctHex4ug', 'dctHex4uh', 'dctHex4ui', 'dctHex4uj', 'dctHex4uk', 'dctHex4ul', 'dctHex4um', 'dctHex4un', 'dctHex4uo', 'dctHex4up', 'dctHex4uq', 'dctHex4ur', 'dctHex4us', 'dctHex4ut', 'dctHex4uu', 'dctHex4uv', 'dctHex4uw', 'dctHex4ux', 'dctHex4uy', 'dctHex4uz', 'dctHex4va', 'dctHex4vb', 'dctHex4vc', 'dctHex4vd', 'dctHex4ve', 'dctHex4vf', 'dctHex4vg', 'dctHex4vh', 'dctHex4vi', 'dctHex4vj', 'dctHex4vk', 'dctHex4vl', 'dctHex4vm', 'dctHex4vn', 'dctHex4vo', 'dctHex4vp', 'dctHex4vq', 'dctHex4vr', 'dctHex4vs', 'dctHex4vt', 'dctHex4vu', 'dctHex4vv', 'dctHex4vw', 'dctHex4vx', 'dctHex4vy', 'dctHex4vz', 'dctHex4wa', 'dctHex4wb', 'dctHex4wc', 'dctHex4wd', 'dctHex4we', 'dctHex4wf', 'dctHex4wg', 'dctHex4wh', 'dctHex4wi', 'dctHex4wj', 'dctHex4wk', 'dctHex4wl', 'dctHex4wm', 'dctHex4wn', 'dctHex4wo', 'dctHex4wp', 'dctHex4wq', 'dctHex4wr', 'dctHex4ws', 'dctHex4wt', 'dctHex4wu', 'dctHex4wv', 'dctHex4ww', 'dctHex4wx', 'dctHex4wy', 'dctHex4wz', 'dctHex4xa', 'dctHex4xb', 'dctHex4xc', 'dctHex4xd', 'dctHex4xe', 'dctHex4xf', 'dctHex4xg', 'dctHex4xh', 'dctHex4xi', 'dctHex4xj', 'dctHex4xk', 'dctHex4xl', 'dctHex4xm', 'dctHex4xn', 'dctHex4xo', 'dctHex4xp', 'dctHex4xq', 'dctHex4xr', 'dctHex4xs', 'dctHex4xt', 'dctHex4xu', 'dctHex4xv', 'dctHex4xw', 'dctHex4xx', 'dctHex4xy', 'dctHex4xz', 'dctHex4ya', 'dctHex4yb', 'dctHex4yc', 'dctHex4yd', 'dctHex4ye', 'dctHex4yf', 'dctHex4yg', 'dctHex4yh', 'dctHex4yi', 'dctHex4yj', 'dctHex4yk', 'dctHex4yl', 'dctHex4ym', 'dctHex4yn', 'dctHex4yo', 'dctHex4yp', 'dctHex4yq', 'dctHex4yr', 'dctHex4ys', 'dctHex4yt', 'dctHex4yu', 'dctHex4yv', 'dctHex4yw', 'dctHex4yx', 'dctHex4yy', 'dctHex4yz', 'dctHex4za', 'dctHex4zb', 'dctHex4zc', 'dctHex4zd', 'dctHex4ze', 'dctHex4zf', 'dctHex4zg', 'dctHex4zh', 'dctHex4zi', 'dctHex4zj', 'dctHex4zk', 'dctHex4zl', 'dctHex4zm', 'dctHex4zn', 'dctHex4zo', 'dctHex4zp', 'dctHex4zq', 'dctHex4zr', 'dctHex4zs', 'dctHex4zt', 'dctHex4zu', 'dctHex4zv', 'dctHex4zw', 'dctHex4zx', 'dctHex4zy', 'dctHex4zz'. The table also includes a 'Parameters' section with fields for 'Message', 'RPC', and 'Duplicate'.
- Unit Overview:** A table listing various units with columns for Unit, Batch ID, Status, Operator, and Operator Status. The table includes units such as 6A1, 6A2, 6A3, 6B1, 6B2, 6B3, 6C1, 6C2, 6C3, 6C4, 6C5, 6C6, 6C7, 6C8, 6C9, 6C10, 6D1, 6D2, 6D3, 6D4, 6D5, 6D6, 6D7, 6D8, 6D9, 6D10, 6E1, 6E2, 6E3, 6E4, 6E5, 6E6, 6E7, 6E8, 6E9, 6E10, 6F1, 6F2, 6F3, 6F4, 6F5, 6F6, 6F7, 6F8, 6F9, 6F10, 6G1, 6G2, 6G3, 6G4, 6G5, 6G6, 6G7, 6G8, 6G9, 6G10, 6H1, 6H2, 6H3, 6H4, 6H5, 6H6, 6H7, 6H8, 6H9, 6H10, 6I1, 6I2, 6I3, 6I4, 6I5, 6I6, 6I7, 6I8, 6I9, 6I10, 6J1, 6J2, 6J3, 6J4, 6J5, 6J6, 6J7, 6J8, 6J9, 6J10, 6K1, 6K2, 6K3, 6K4, 6K5, 6K6, 6K7, 6K8, 6K9, 6K10, 6L1, 6L2, 6L3, 6L4, 6L5, 6L6, 6L7, 6L8, 6L9, 6L10, 6M1, 6M2, 6M3, 6M4, 6M5, 6M6, 6M7, 6M8, 6M9, 6M10, 6N1, 6N2, 6N3, 6N4, 6N5, 6N6, 6N7, 6N8, 6N9, 6N10, 6O1, 6O2, 6O3, 6O4, 6O5, 6O6, 6O7, 6O8, 6O9, 6O10, 6P1, 6P2, 6P3, 6P4, 6P5, 6P6, 6P7, 6P8, 6P9, 6P10, 6Q1, 6Q2, 6Q3, 6Q4, 6Q5, 6Q6, 6Q7, 6Q8, 6Q9, 6Q10, 6R1, 6R2, 6R3, 6R4, 6R5, 6R6, 6R7, 6R8, 6R9, 6R10, 6S1, 6S2, 6S3, 6S4, 6S5, 6S6, 6S7, 6S8, 6S9, 6S10, 6T1, 6T2, 6T3, 6T4, 6T5, 6T6, 6T7, 6T8, 6T9, 6T10, 6U1, 6U2, 6U3, 6U4, 6U5, 6U6, 6U7, 6U8, 6U9, 6U10, 6V1, 6V2, 6V3, 6V4, 6V5, 6V6, 6V7, 6V8, 6V9, 6V10, 6W1, 6W2, 6W3, 6W4, 6W5, 6W6, 6W7, 6W8, 6W9, 6W10, 6X1, 6X2, 6X3, 6X4, 6X5, 6X6, 6X7, 6X8, 6X9, 6X10, 6Y1, 6Y2, 6Y3, 6Y4, 6Y5, 6Y6, 6Y7, 6Y8, 6Y9, 6Y10, 6Z1, 6Z2, 6Z3, 6Z4, 6Z5, 6Z6, 6Z7, 6Z8, 6Z9, 6Z10.

On the left side, a photograph of an ABB PLC rack is shown. Two arrows point from the rack to the software interface: one points to the 'REACTOR AREA OVERVIEW' window, and the other points to the 'Batch Overview' window.

Specification 2: OPC AE for Alarms and Events

Events are changes in the process that need to be logged, such as "production start"

Alarms are abnormal states in the process that require attention, such as "low oil pressure"

OPC AE (Alarms and Events) specifies how alarms and events are subscribed, under which conditions they are filtered and sent with their associated messages.

The main clients of OPC AE are the Alarms and Event loggers.

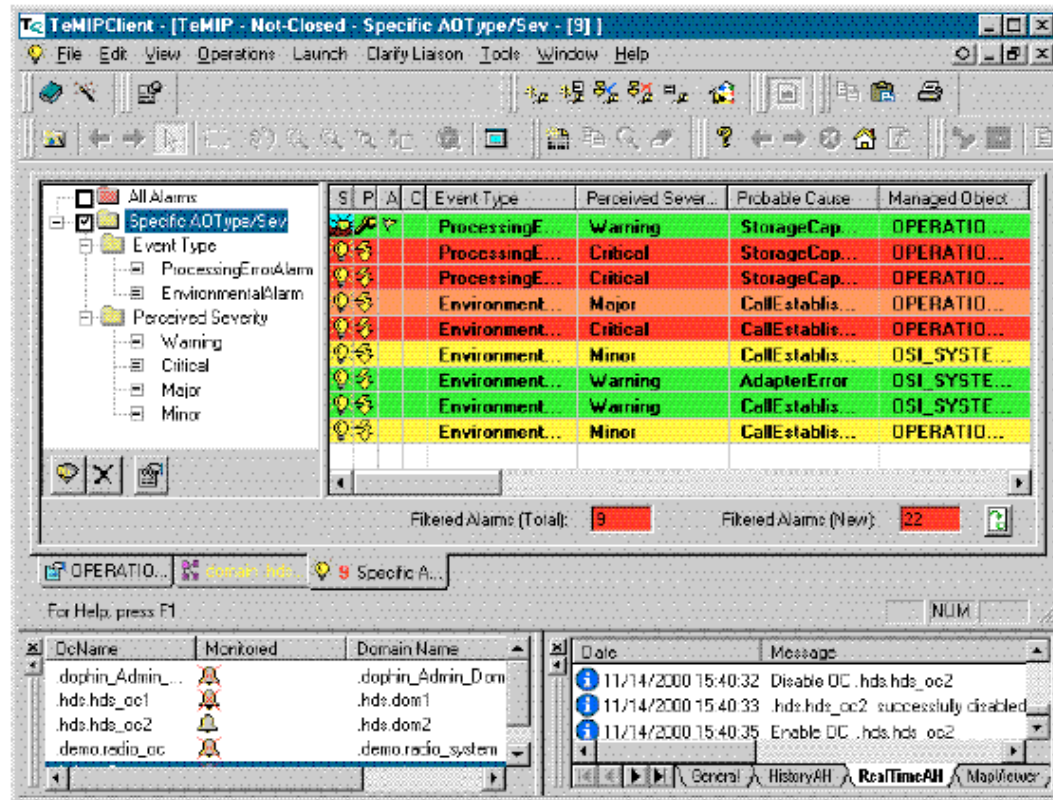
determine the exact time of change
(time stamping)

categorize by priorities

log for further use

acknowledge alarms
(events are not acknowledged)

link to clear text explanation

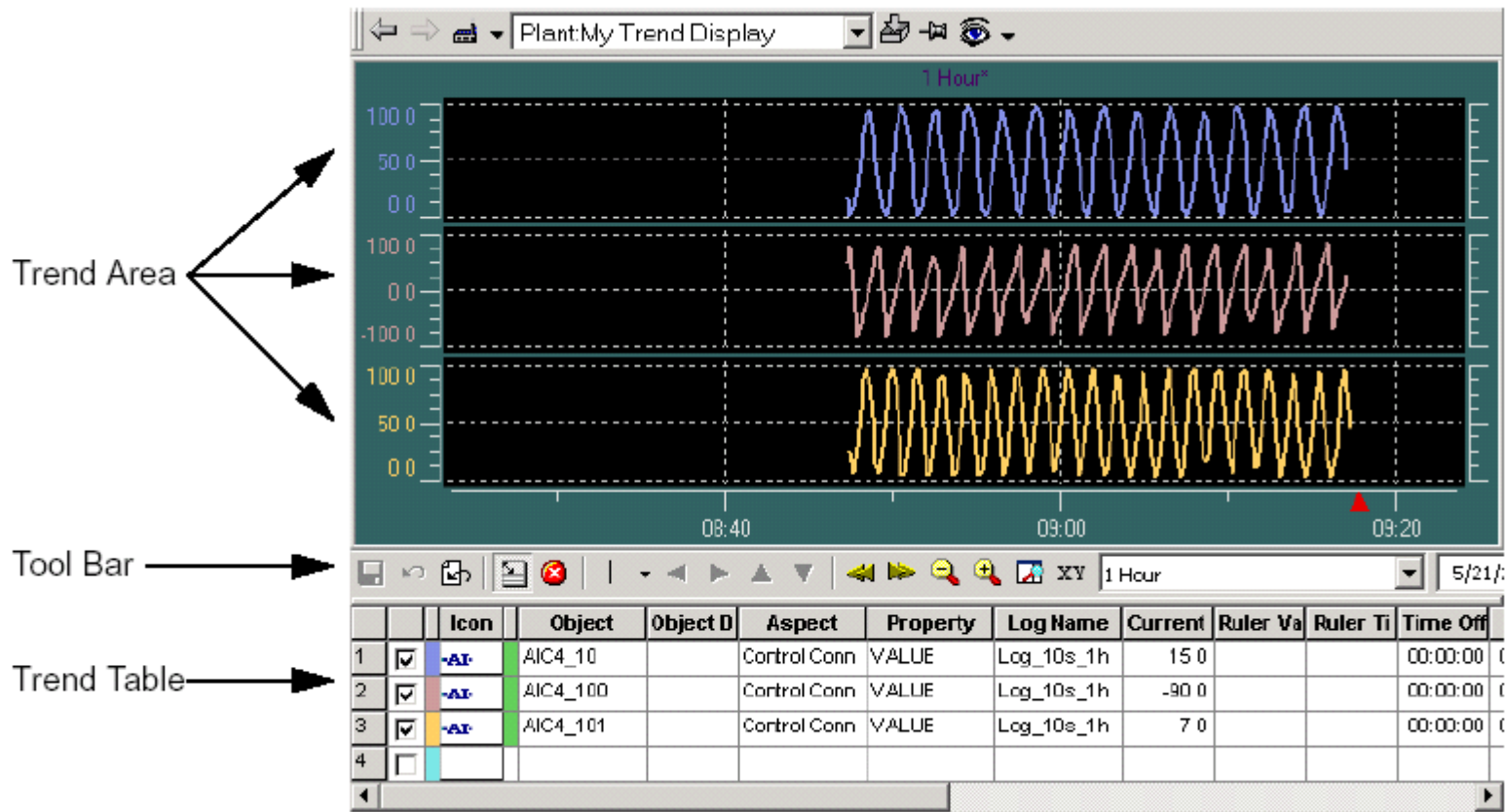


Specification 3: HDA for Historical Data Access

Historical Data are process states and events such as: process variables, operator actions, recorded alarms,... that are stored as logs in a long-term storage for later analysis.

OPC HDA (Historical Data Access) specifies how historical data are retrieved from the logs in the long-term storage, filtered and aggregated (e.g. compute averages, peaks).

The main client of OPC HDA are Trend Displays and Historians.



Specification 4: OPC Batch

based on:

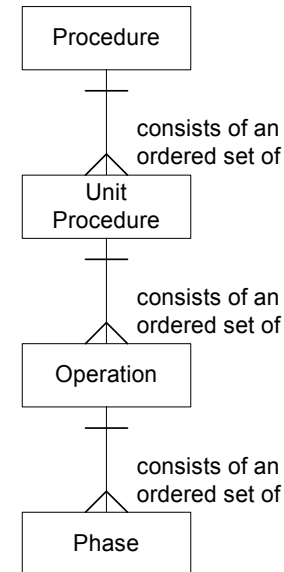
IEC 61512-1 Batch Control – Part 1: Models and Terminology

(ANSI/ISA S88.01 1995)

ISA-dS88.02-2000 draft 17 of May 2000

allows to access:

- equipment capabilities,
- current operating conditions,
- historical and
- recipe contents



Beyond Microsoft: OPC UA

In a move to get more independence from Microsoft and use web technology, a new specification called " Unified Architecture" (formerly. OPC XML) that uses web services for all kinds of transactions: query, read, write, subscribe,...

The classical OPC DA, AE and HDA are implemented with XML / SOAP /WSDL this allows encryption and authentication of process data.

This does not only standardize the interfaces, but also the transmitted data.

OPC as an integration tool

OPC Common

Overview: usage and specifications

OPC as an integration tool

Clients and Servers: configuration

Automation and Custom Interface

OPC Data Access

Overview: Browsing the server

Objects, Types and properties

Communication model

Simple Programming Example

Standard and components

OPC Alarms and Events Specification

Overview: definitions and objects

Events

Alarm Conditions

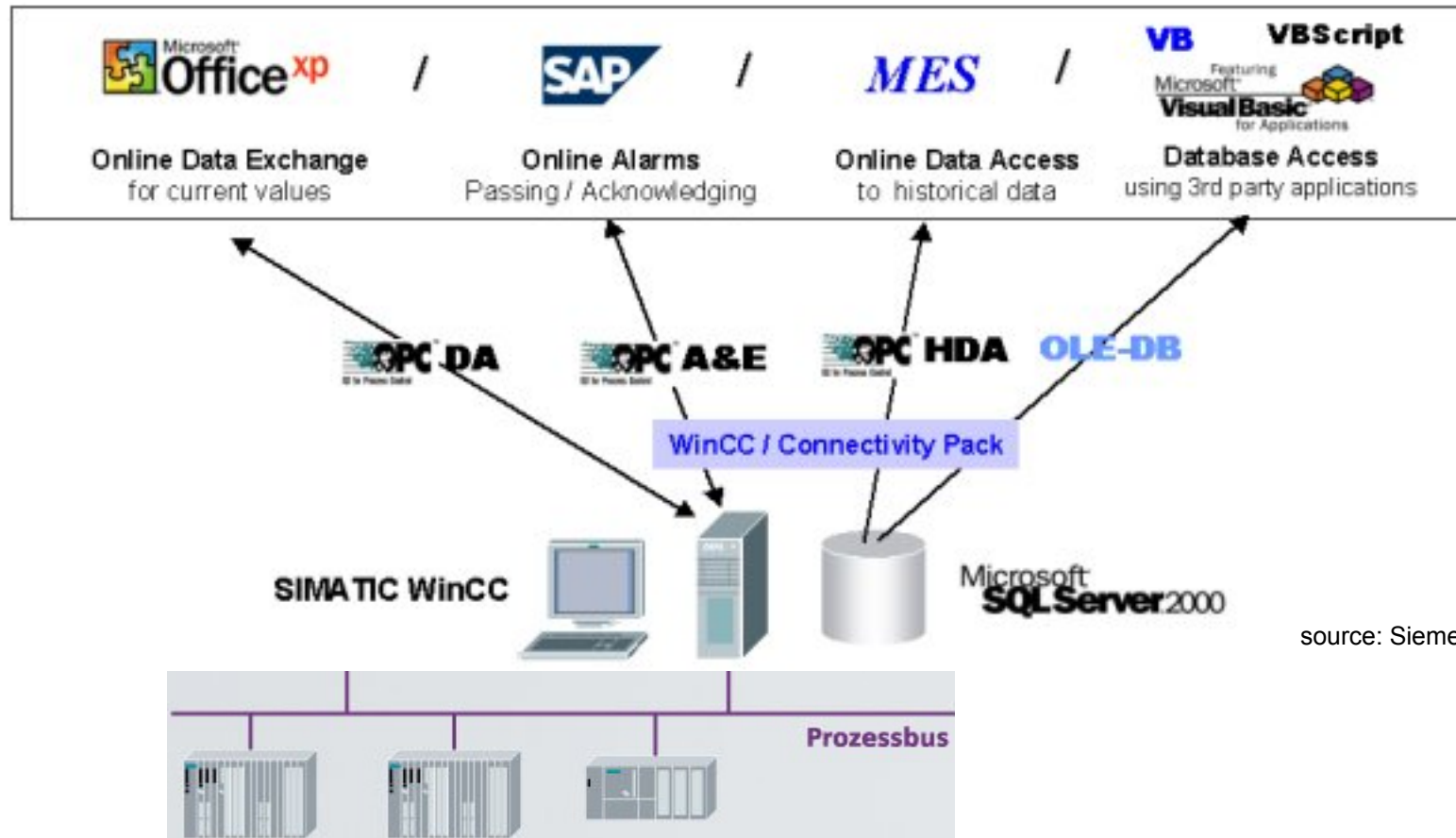
Automation Interface

OPC Historical Data Specification

Overview

OPC as a hub

OPC variables is also a convenient way to exchange data between applications on the same machine. OPC data can be easily read in any Microsoft Office application



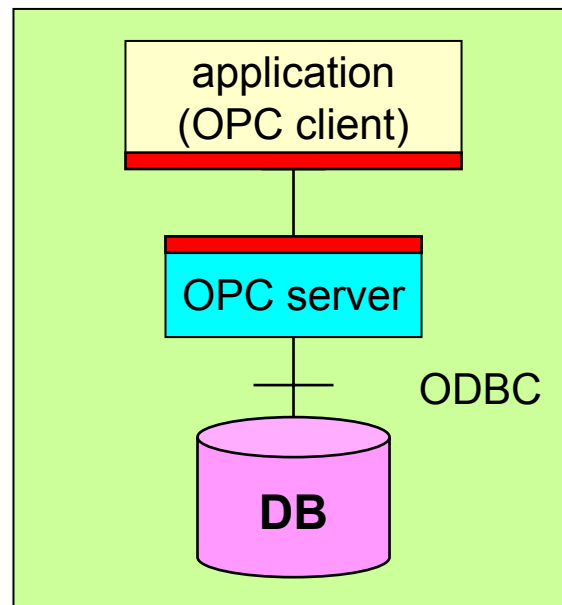
source: Siemens WinCC

OPC connection to databases

Tools such as LifeWire's allow to build an OPC DA interface to any ODBC - equipped database.

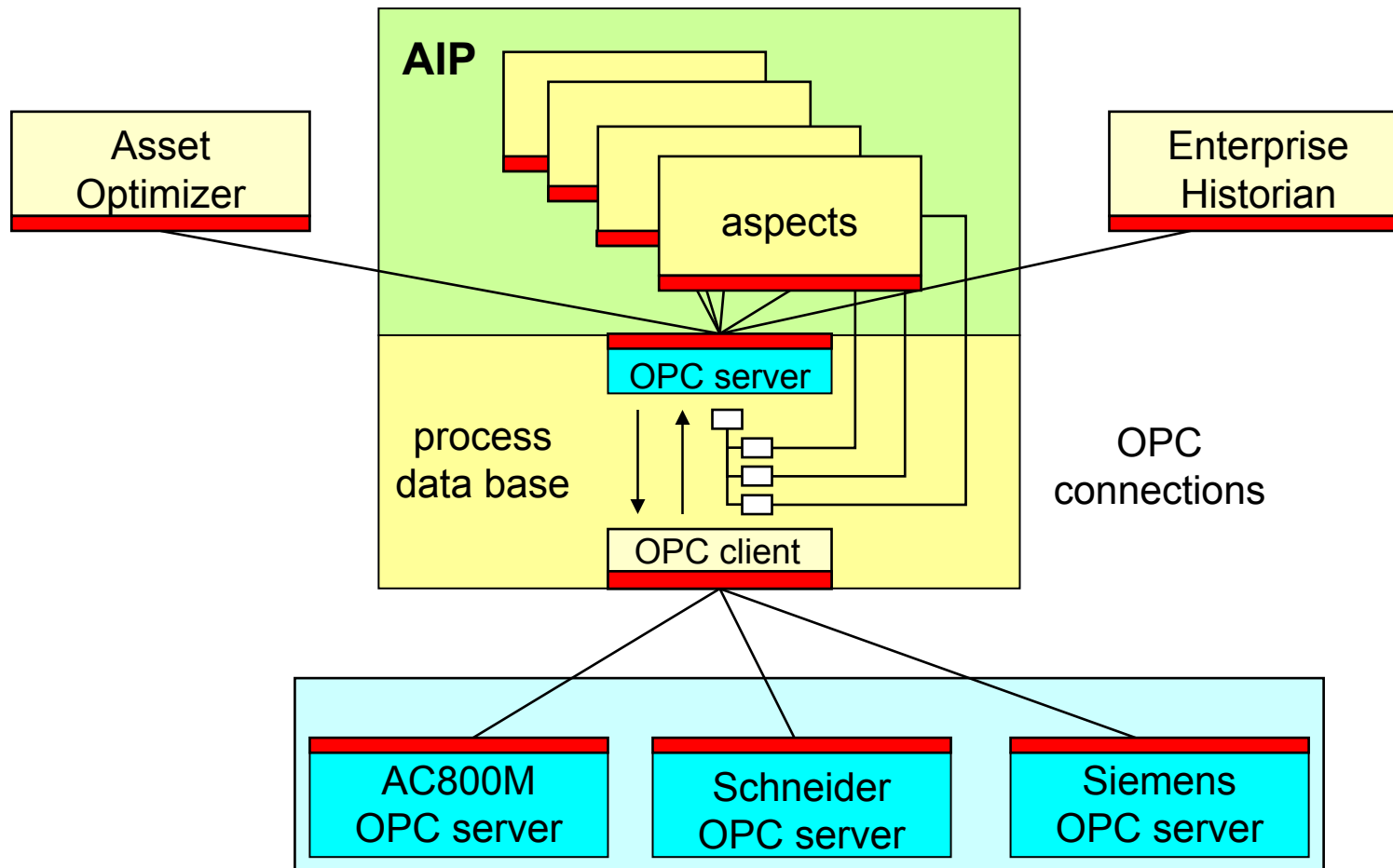
The database internal structure (exposed through queries) is reflected as a hierarchy of OPC items.

This allows to give a unified access to simple items.

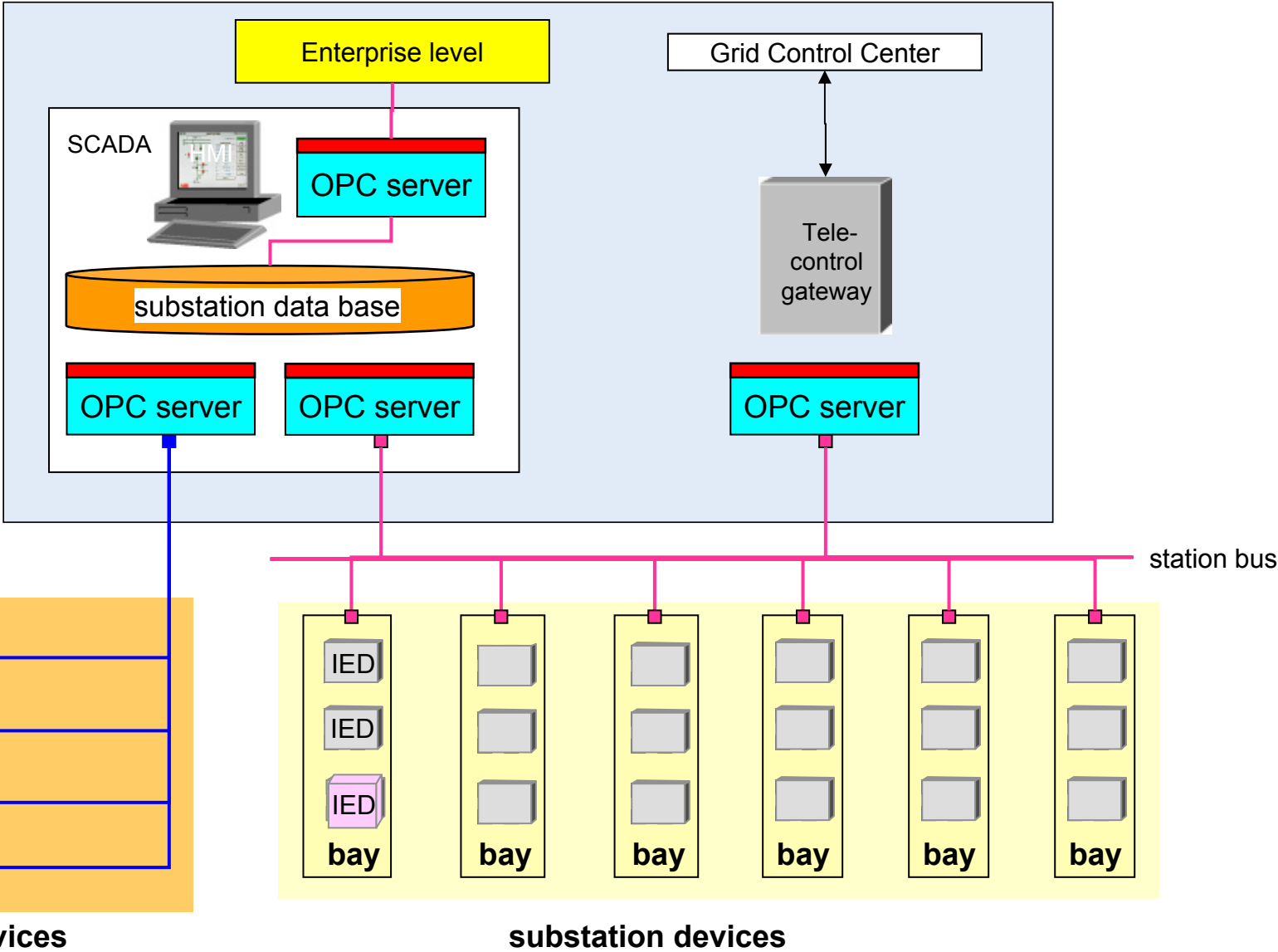


OPC for internal communication: ABB's 800xA as example

ABB's Integration Platform (800xA) is at the same time OPC server and OPC client. Components (aspects) within AIP expose their properties as OPC objects. Internal (within AIP) and external communication takes place over OPC.

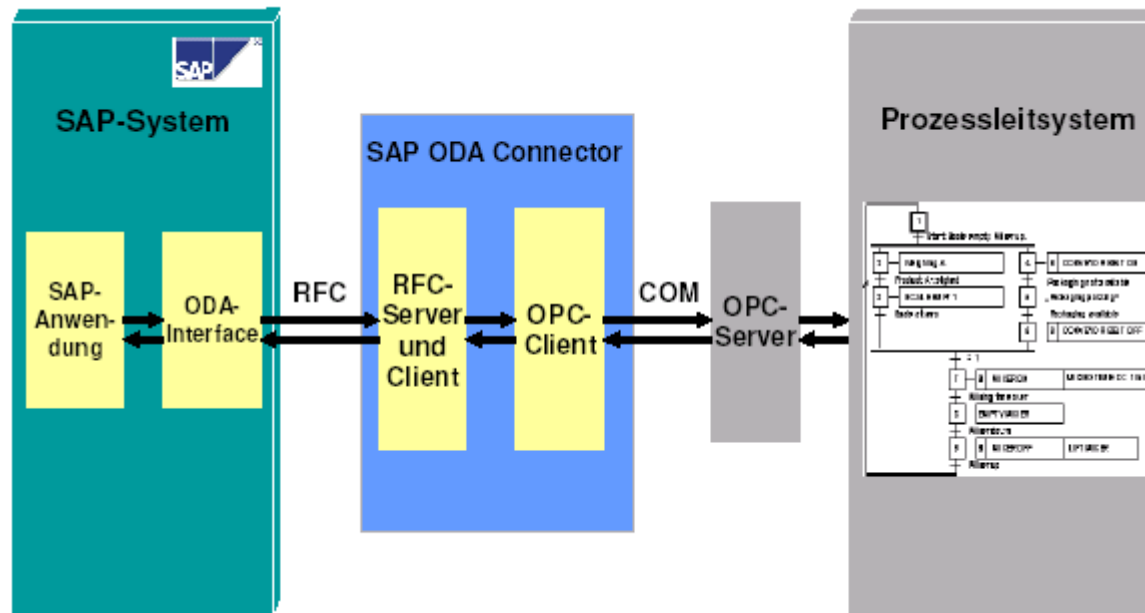


Example substation automation



OPC Connection to Enterprise Resource Planning

Direct connection to SAP (BAPI) is provided by tools such as Matrikon's or Intellution's



Simulators and Explorer: which helps are available

Explorer:

Several tools are available on the market to browse OPC servers, especially:

- Matrikon OPC Explorer (no source code)
- TopServer Client (source code in VB available)

Simulator:

OPC data should be simulated before commissioning the real plant.

To this effect, commercial simulation servers allow to create, observe and change variables by hand or according to time functions (ramp, random,...).

Most PLC servers have also a simulation mode.

Freeware servers such as Matrikon have only limited number of variables

These explorers and simulators work with OPC DA, AE is yet seldom.

Client and Servers

OPC Common

Overview: usage and specifications

OPC as an integration tool

Clients and Servers: configuration

Automation and Custom Interface

OPC Data Access

Overview: Browsing the server

Objects, Types and properties

Communication model

Simple Programming Example

Standard and components

OPC Alarms and Events Specification

Overview: definitions and objects

Events

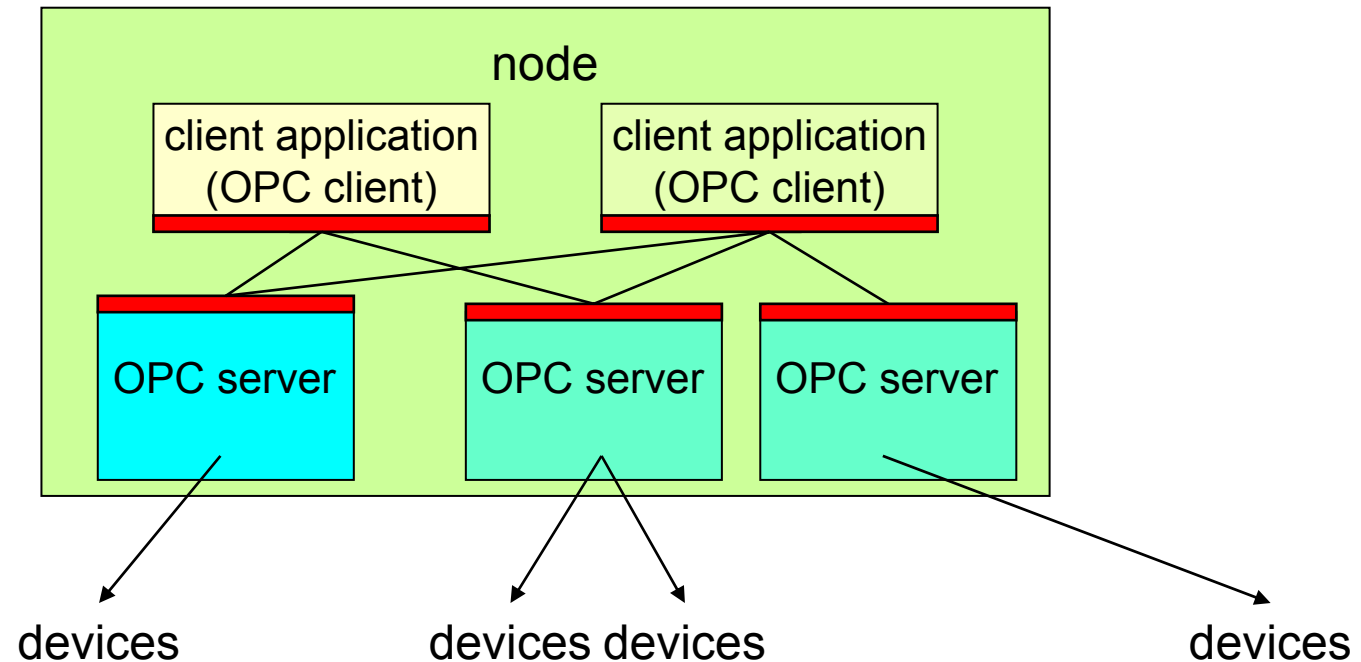
Alarm Conditions

Automation Interface

OPC Historical Data Specification

Overview

Server(s) and Client(s) in the same node

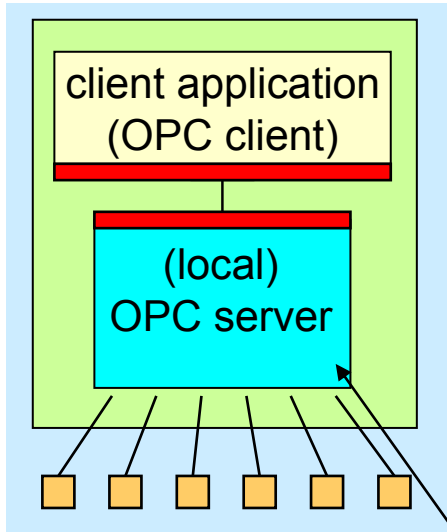


Clients and servers run as parallel processes

The OPC specification defines the interface between client and server in the form of objects and methods.

Direct and Fieldbus access

direct connection

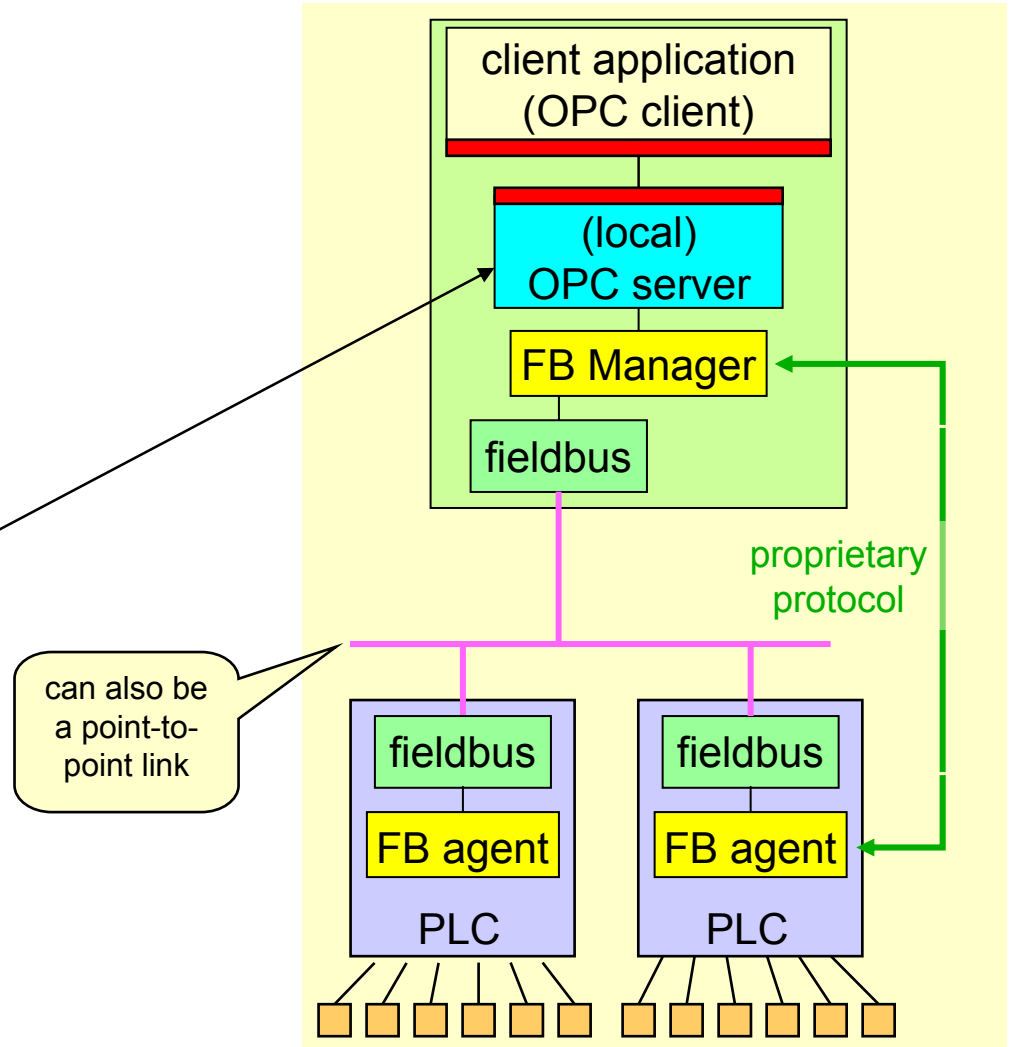


clients and servers run as parallel processes

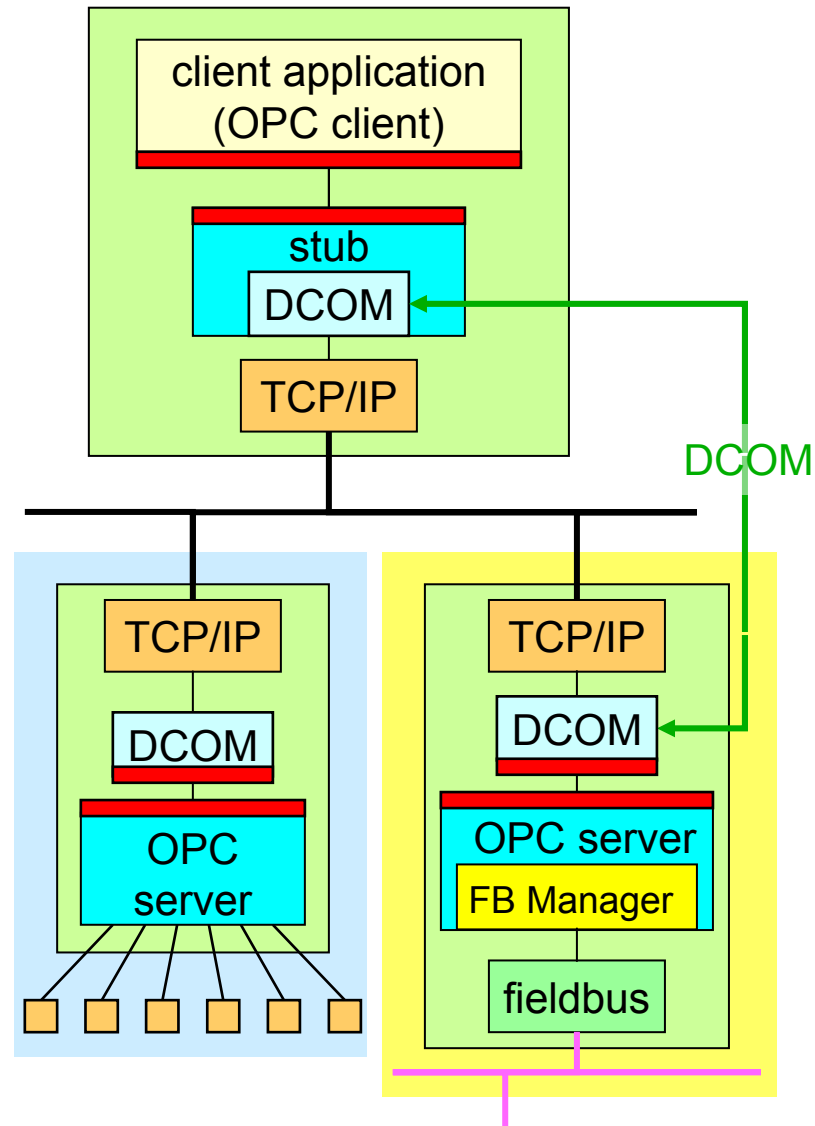
I/O devices

The OPC server is running all the time, as soon as at least one client is present

fieldbus connection

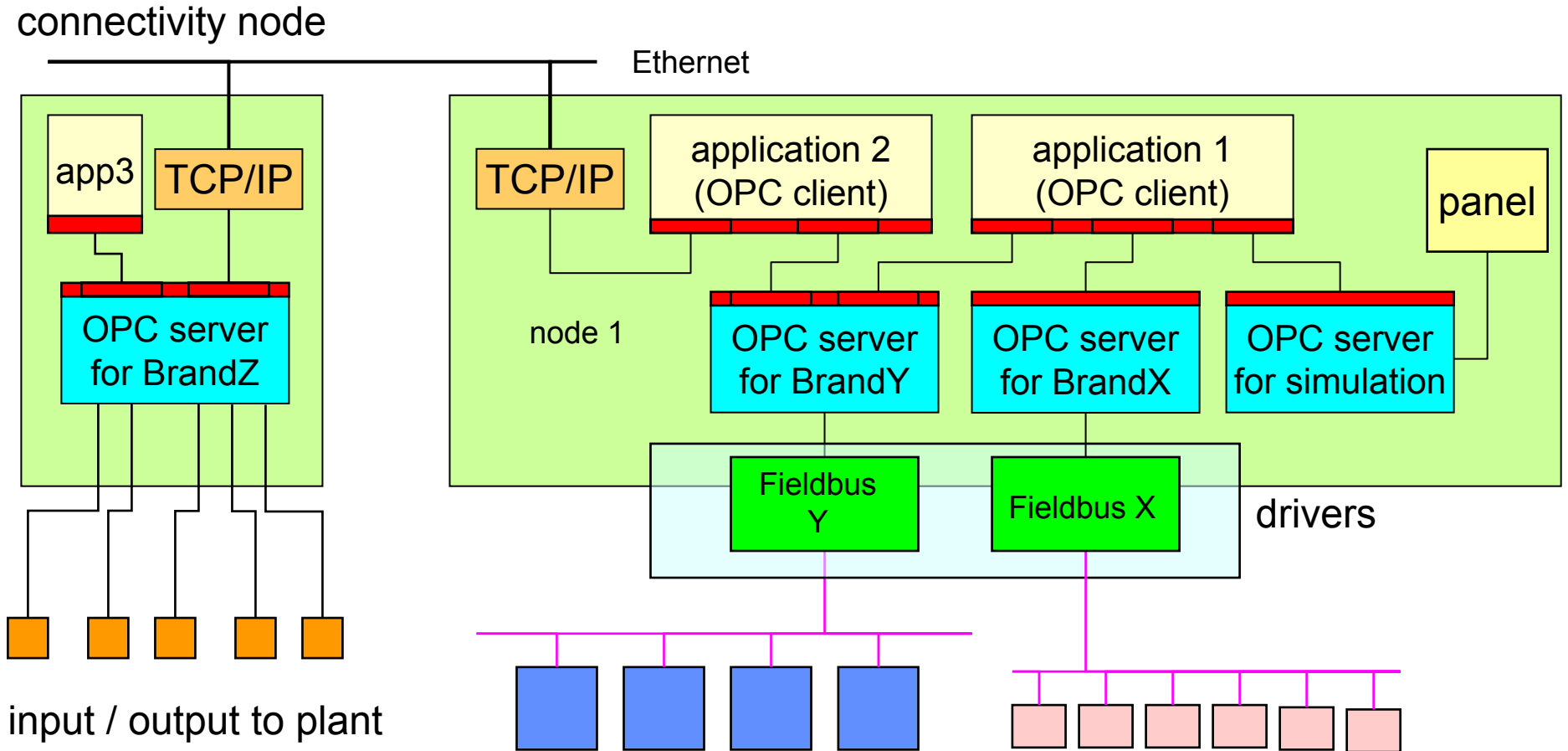


Accessing a server in another node



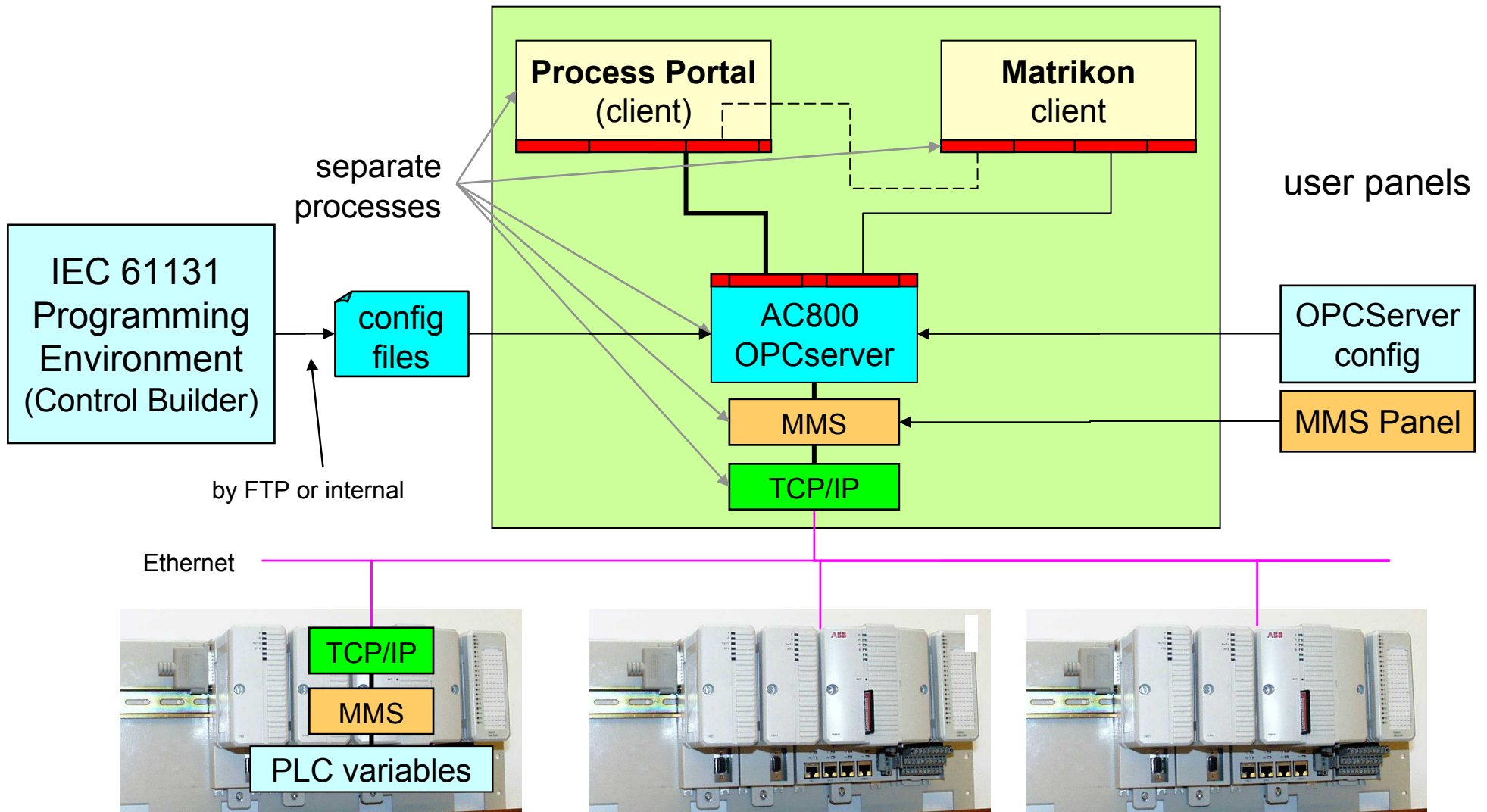
Limitation:
does not work over firewalls.
Solution:
OPC UA (see later)

Full-fledged COM/DCOM across multiple nodes



The OPC servers support multiple clients and servers on the same, or on remote nodes. They run as separate processes (as soon as at least one client is requesting them)

Example: ABB AC800 OPC Server based on MMS



The variables are defined in the server, not in the PLC.

OPC Technology

OPC Common

Overview: usage and specifications

OPC as an integration tool

Clients and Servers: configuration

OPC Technology, client and custom interface

OPC Data Access

Overview: Browsing the server

Objects, Types and properties

Communication model

Simple Programming Example

Standard and components

OPC Alarms and Events Specification

Overview: definitions and objects

Events

Alarm Conditions

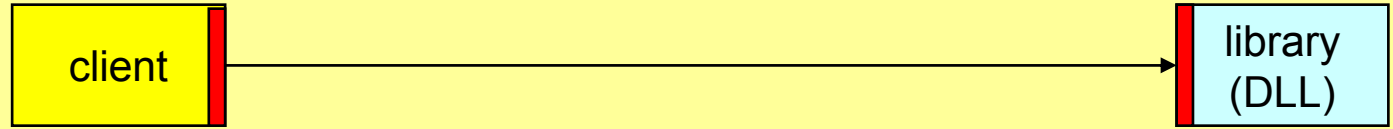
Automation Interface

OPC Historical Data Specification

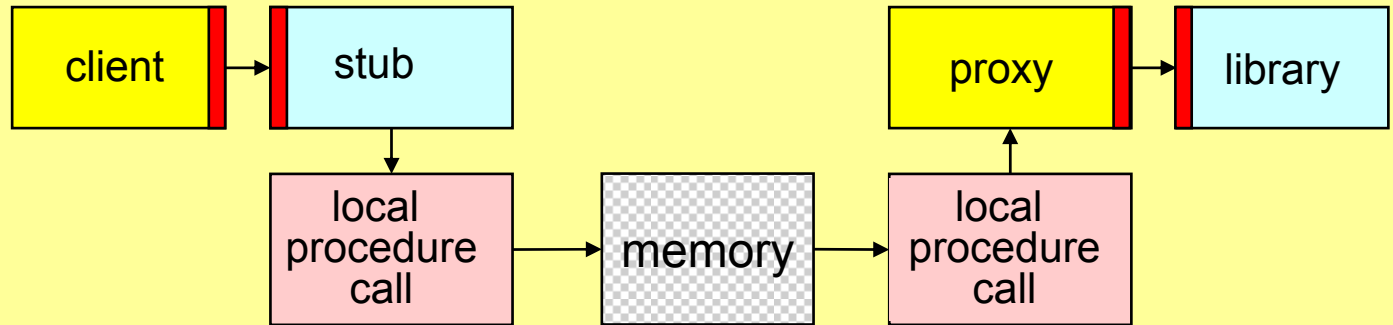
Overview

COM/DCOM quick intro

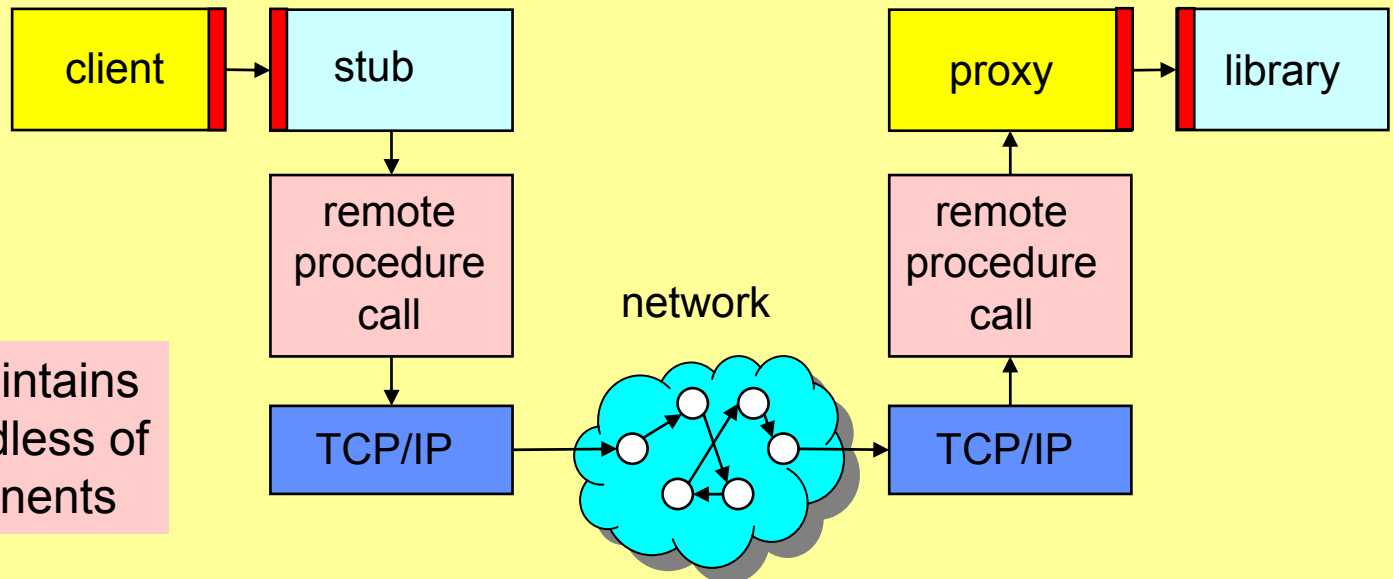
1) same process



2) different processes
same computer

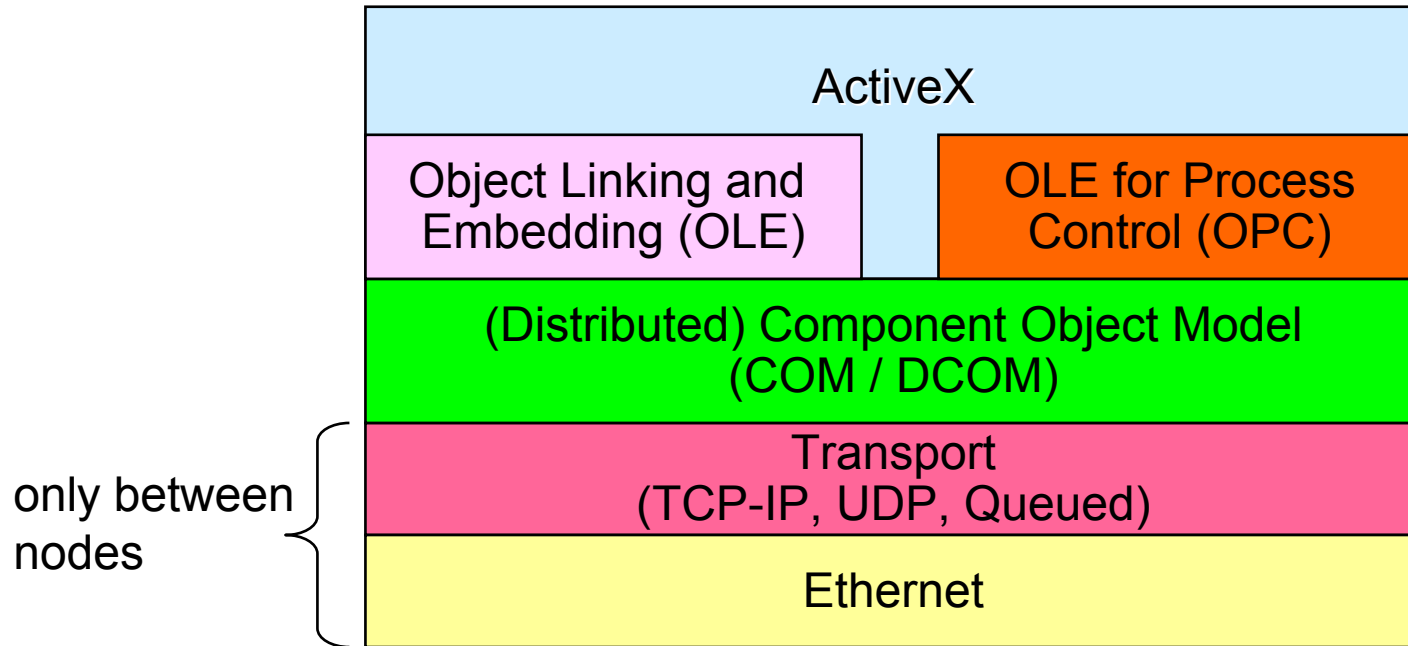


3) different computers



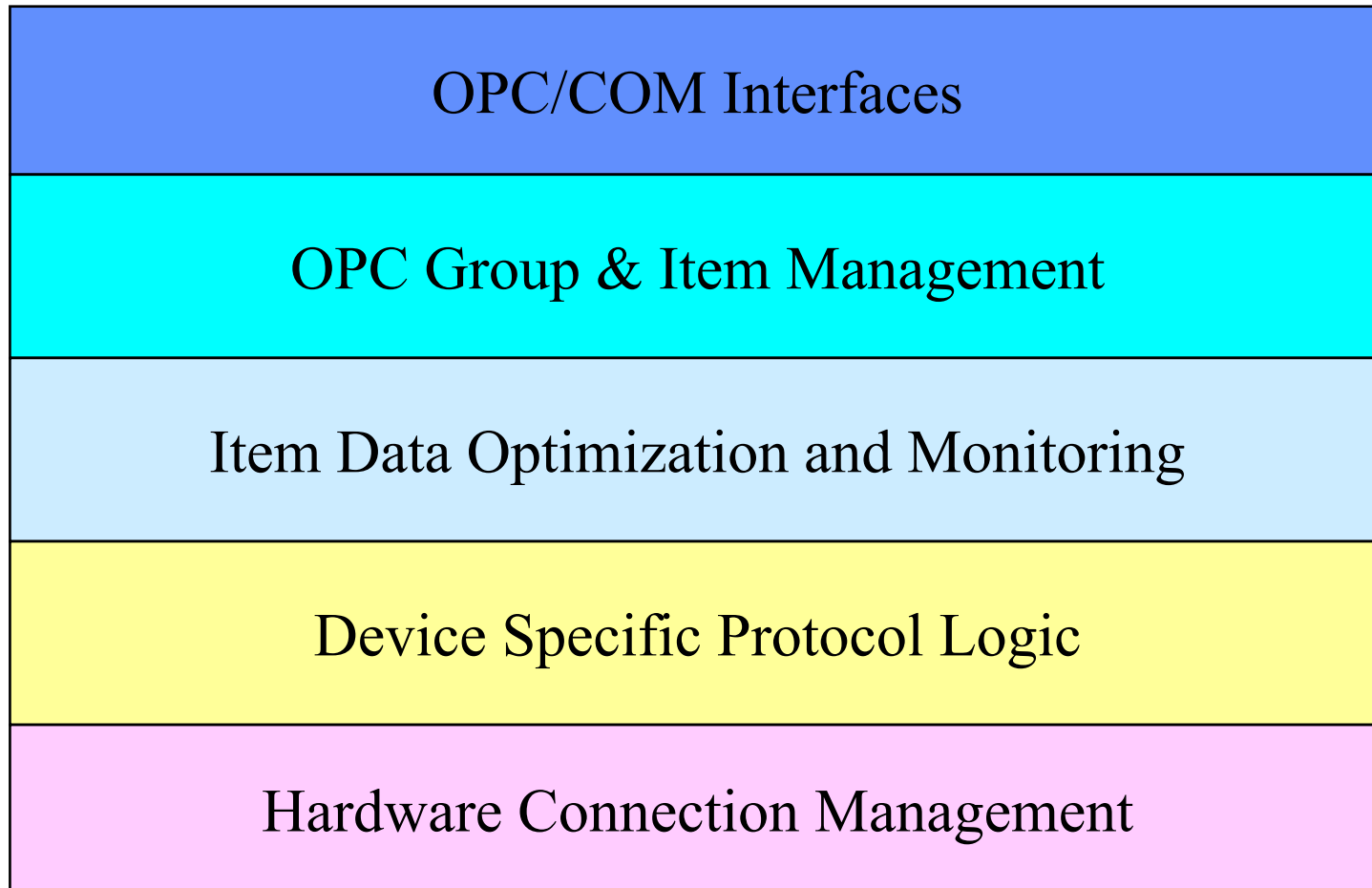
COM/DCOM (COM+) maintains the same interface regardless of the location of the components

OPC technologies



OPC bases on Microsoft's COM/DCOM technology (i.e. it only works on Windows platforms). Effort to port it to other platforms (Linux) and web transport protocols (XML) are in progress. Advantages are the direct integration into all applications, such as Excel.

Structure of an OPC server



“Automation” vs. “Custom” interface

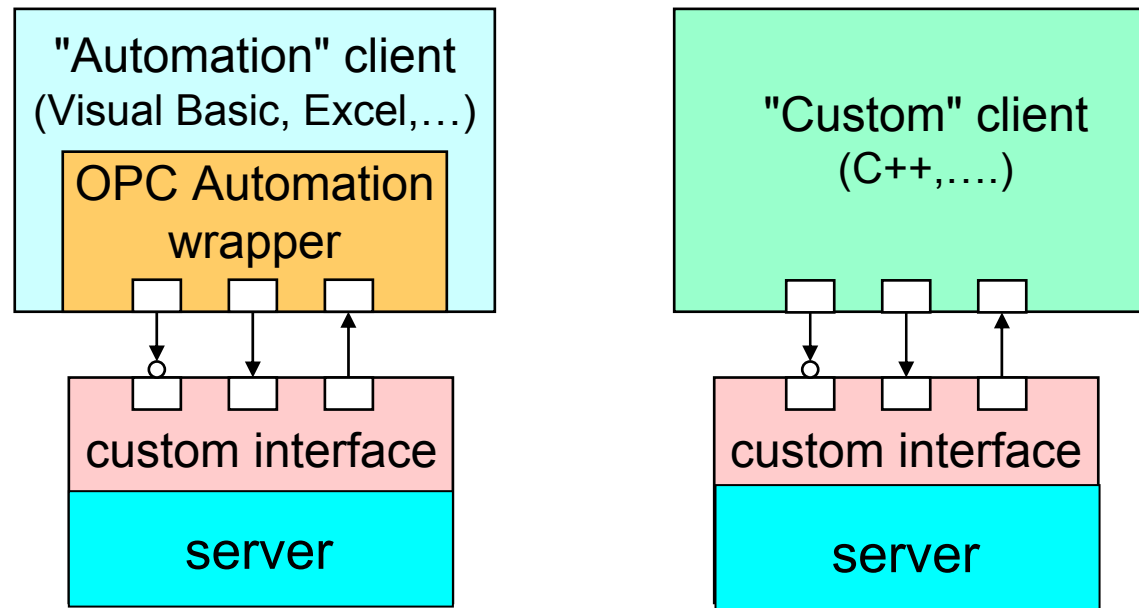
The OPC specifications define two interfaces: "custom" and "automation".

“custom” is the native C++ interface of COM.

“automation” is the interface offered in Visual Basic, used in Word, Excel,.....

The interface is defined by a Type Library (distributed by the OPC Foundation)

Functionality is roughly the same in both models, “automation” is easier to use, but "custom" gives a more extended control.



Assessment Common

What is the objective of OPC ?

On which technology does OPC rely ?

What is an OPC Server ?

Which are the main OPC specifications ?

What are the components of the OPC DA Automation Interface ?

How does an automation platform use the OPC interfaces ?

Three-tiers Active-X components

