

Státnicové okruhy pro předmět Průmyslová informatika a internet

Richard Šusta, verze 0.5 Beta

Dokument obsahuje vybrané části z mých přednášek, které podle mého názoru souvisejí s jednotlivými státnicovými okruhy. Vznikl pro usnadnění přípravy na státnice a jedná se pouze o pomůcku, nikoliv o kompletní učebnici, která by byla závaznou normou pro zkušební komisi.

Obsah

Státnicové okruhy pro předmět Průmyslová informatika a internet.....	1
A. okruhy I.	2
1. Architektura distribuovaných síťových aplikací.....	2
2. Struktura sítě, aplikační vrstvy, její hlavní části.....	4
3. Nejběžnější databázové servery v průmyslových aplikacích a jejich management.	4
4. Webové aplikace v distribuovaných řídicích systémech.	5
5. Webové komponenty pro vývoj průmyslových internetových aplikací.	8
6. Zabezpečení, přístupová práva, veřejné klíče, hashovací kódy, digitální podpisy.	8
7. Spolehlivost sítí a řešení vhodná pro kritické aplikace.....	8
B. Nové okruhy	9
8. Klasické PLC, tagově orientované PLC (ControlLogix).	9
9. Průmyslové sítě PROFINET a ControlNet.....	10
10. OPC (OLE for Process Control).....	13
11. Systémy SCADA a MES (Manufacturing Execution Systems) - principy, organizace, činnost a sběr dat.....	14

A. okruhy I.

1. Architektura distribuovaných síťových aplikací.

Co je distribuovaná síťová aplikace? Nazývá se tak aplikace složená z více autonomních jednotek (počítačů) vzájemně komunikujících po sítích. Programy se pro ni musí/mohou rozdělit na několik částí běžících simultánně, takže svým způsobem jde o paralelně prováděné operace, spuštěné na různých počítačích.

Není-li síťové propojení deterministické, pak distribuované síťové aplikace pracují v nehomogenním prostředí, kde se musí počítat s různými latencemi, poruchami ve spojení či závady jednotlivých počítačů.

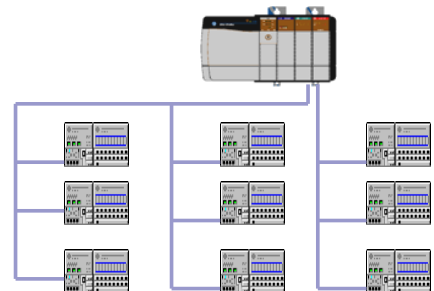
Pozn. některé průmyslové sítě se chovají deterministicky, ale většina sítí nikoliv.

Výhody centralizovaných aplikací

- Všechny programy máme na jednom místě
- Lépe se řeší systémové problémy
- Stejně rychlý přístup na vnější jednotky

Nevýhody centralizovaných aplikací

- Velké programy
- Výkon závisí na počtu I/O
- Závada jednoho počítače znamená zastavení celého programu

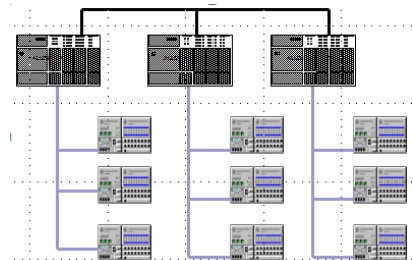


Výhody distribuovaných

- Rozdělení programu podle úloh
- Snáze se hledají lokální problémy
- Zvýšení výkonu na lokálních sítích

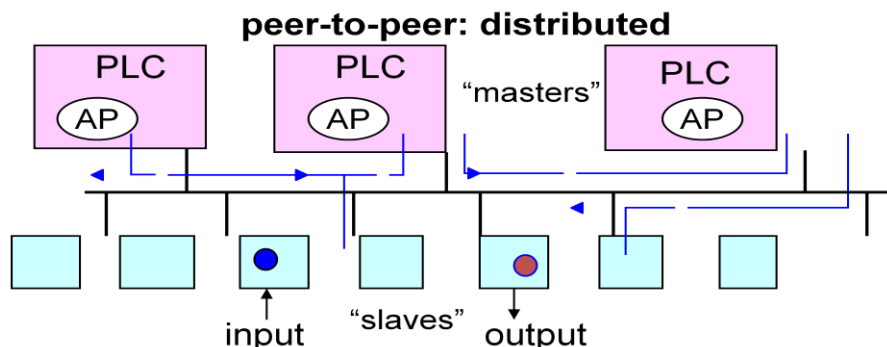
Nevýhody distribuovaných

- Dostup na některá zařízení přes další uzly
- Systémové problémy se hůře hledají
- Vyšší cena
- Udržujeme několik programů



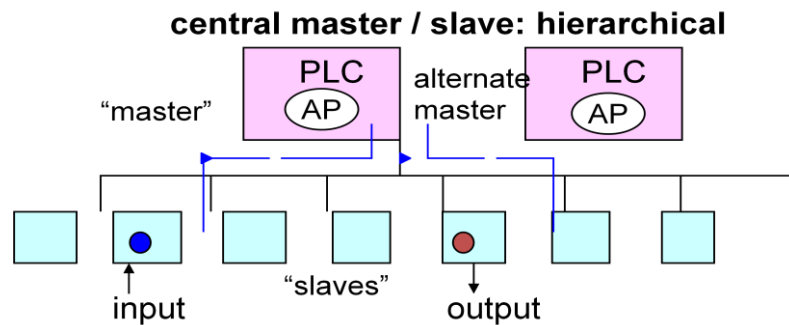
Podle architektury lze distribuované systémy rozdělit do těchto kategorií:

- **Peer-to-peer** architektura – zde neexistuje žádný ústřední prvek, který by poskytoval služby nebo spravoval síťové zdroje; všechno je rovnoměrně rozděleno mezi všechny uzly (peers). Peers mohou sloužit jak servery tak jako klienti.



- **Klient-server** – klientský program kontaktuje server, aby získal data, ty pak formátuje a prezentuje uživateli. Vstup od uživatele se odesílá zpět na server, který dle svého

programu zareaguje – odezvou poslanou klientovi, uložením vstupu, či jeho ignorováním.

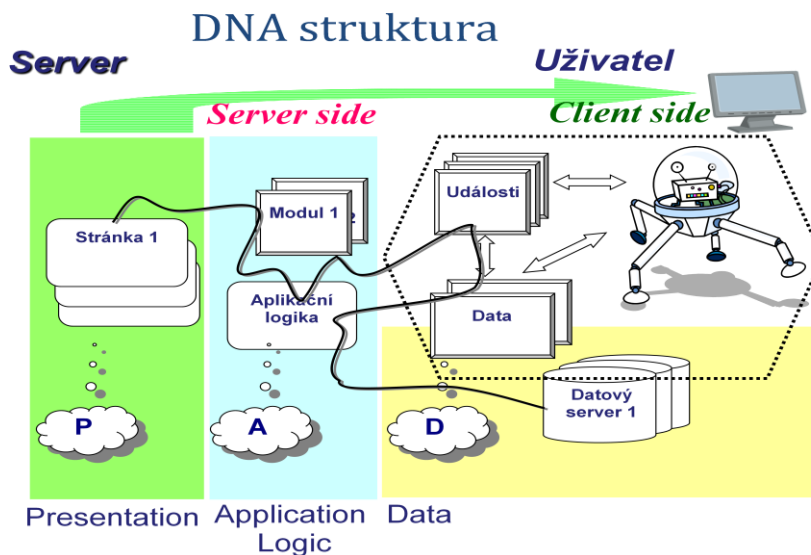


- **Třístupňová architektura** – klientská aplikace má spojení serverovou aplikací, která využívá databázový server. Aplikaci lze tak rozčlenit na tři části.
Jedno z možných označení 3-vrtvé aplikační architektury je DNA (komerční název Microsoftu = Distributed Network Architecture)

- P - **Presentation** dálkové monitorování, zadávání povelů, kde se hodí klasické webové aplikace
- A - **Application Logic** nebo **Business Logic** zpracování údajů z datových zdrojů
- D - **DATA** vlastní datové zdroje

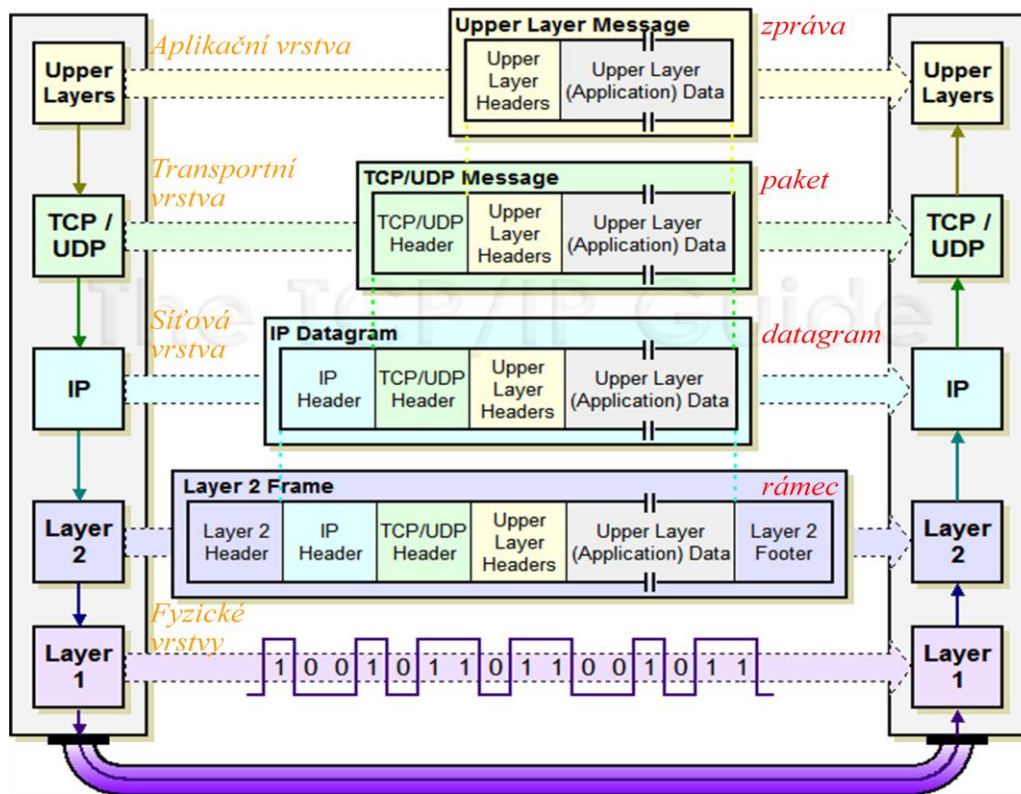
Poznámky

- *Jde především o organizační členění. Jednotlivé části aplikace mohou pracovat na různých nebo na stejných prostředcích.*
- *Spojení mezi moduly se provádí datovými přenosy, např. komunikací mezi procesy, přes internet nebo průmyslové sítě, nebo i jinak*



- **N-stupňová architektura** – architektura rozšířená o další vrstvy, které používá serverová aplikace.
- **Clusterová architektura** – seskupení strojů pracujících vedle sebe a pracujících paralelně na nějakém procesu. Úkol je rozdělován na části, které jsou vykonány zvlášť jednotlivými počítači, a finální výsledek dostaneme složením jednotlivých částí.
- **Space based** – síťová infrastruktura vytvářející iluzi jednotného adresového prostoru. Data jsou transparentně replikována podle potřeb aplikace.

2. Struktura sítě, aplikační vrstvy, její hlavní části.



Některé protokoly

Aplikační vrstva

- **FTP** (File Transfer Protocol) - služba FTP
- **SMTP** (Simple Mail Transfer Protocol) - pro E-mail
- **POP3** (Post Office Protocol) - pro službu E-mail
- **HTTP** (Hypertext Transfer Protocol) - pro službu WWW

Transportní vrstva

- **TCP** (Transmission Control Protocol) - kladně potvrzovaný přenos dat
- **UDP** (User Datagram Protocol) - nepotvrzovaný přenos dat

Síťová vrstva

- **IP** (Internet Protocol) - přenos paketů, směrování přes IP adresy
- **RARP** (Reverse Address Resolution Protocol) - automatické přidělení IP adresy připojenému počítači
- **ICMP** (Internet Control Message Protocol) - přenos služeb. informací

3. Nejběžnější databázové servery v průmyslových aplikacích a jejich management.

V průmyslových aplikacích se používají databázové servery jako např. SQL Server, Oracle, a jiné

– ty zajišťují

- Správu uživatelů a jejich skupin a vazby na doménu;
- Autorizaci uživatelů přes lokální nebo doménové účty;
- Definice modelu řízeného zařízení;
- Společné číselníky všech modulů -Materiály, měrné jednotky, výčtové typy používané výrobcem; ...
- Centrální správu zálohování celého systému;

- Správu hlášení definovaných událostí s možností přeposílání na e-mail/sms uživatelů.

Batch moduly nad databázemi - zajišťují řízení dávkových (šaržových výrobních procesů) podle mezinárodního standardu ANSI/ISA 88.

Obsahují správu

- a) Fází,
- b) Operací
- c) Receptur a předpisů v životním cyklu

Umožňují

- Plánování výrobních dávek
- Řízení průběhu výrobní dávky
- Ukládání a prohlížení historie výrobní dávky
- Tisk protokolu

Historické databáze – dovolují ukládat data o historii výroby. Jde o **nejdůležitější součást** průmyslových systémů, v nichž se požaduje **trasovatelnost dat** pro tvorbu rodokmenu (genealogie) produktu a ukládání procesních a kvalitativních parametrů k materiálovým lotům. Historické databáze pracují se dvěma typy dat

a) Trendy

b) Alarmy a hlášení

pro ty zajišťují jejich sběr, dlouhodobou archivaci, prezentaci a analýzy

Základní charakteristikou historických databází je schopnost ukládat velké množství dat, řádově až desítky miliony záznamů za vteřinu – zde se zpravidla využívá přímý zápis do diskových polí ve speciálních formátech.

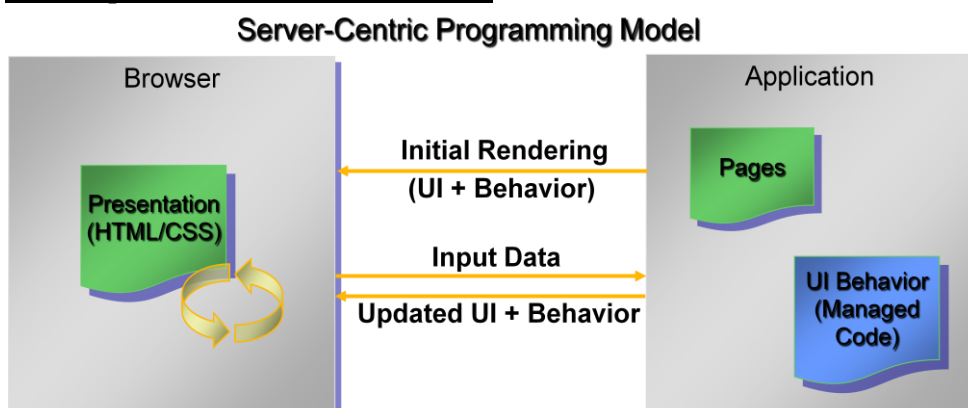
Příklady historických databází např.

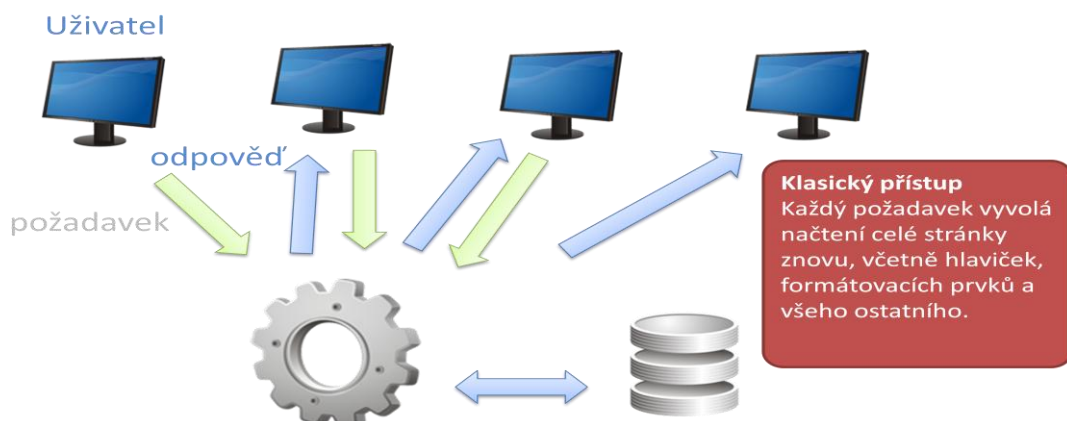
- Enterprise Historian,
- OSISoft,
- GE-Fanuc Intellution iHistorian,
- Rockwell Automation (RSBizWare) Historian,
- **Historian COMES** – český výrobek

4. Webové aplikace v distribuovaných řídicích systémech.

Zde lze zmínit **3-vrstvou strukturu DNA** (zmíněno u otázky Architektura distribuovaných síťových aplikací) jako příklad webové aplikace v distribuovaných řídicích systémech, kde se v části prezentace využívá webové rozhraní.

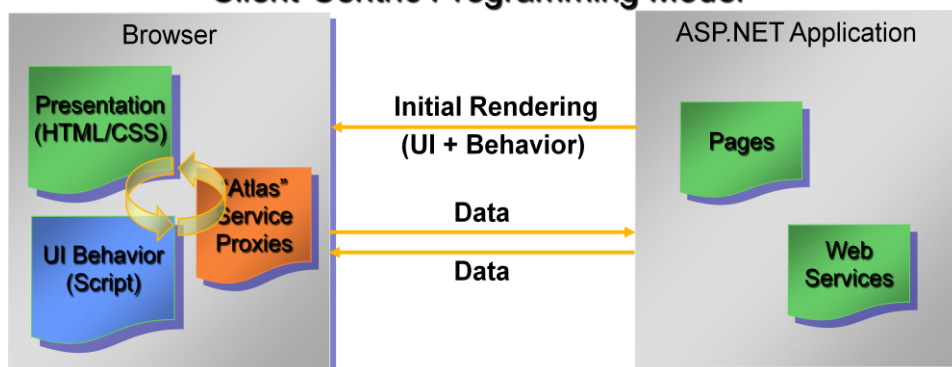
Členění podle komunikace se serverem



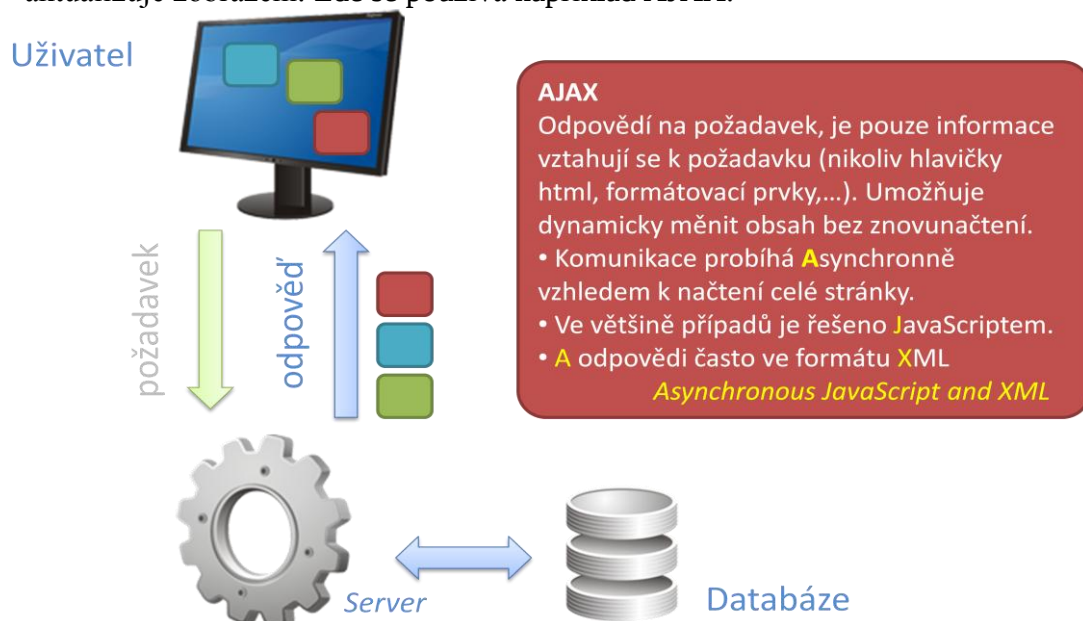


Server-centric model soustřeďuje veškeré operace na server. Prohlížeč načte výchozí údaje. Pokud uživatel provede nějakou akci, po níž se musí aktualizovat zobrazená data, pošle se požadavek na server, ten vrátí nový výsledek jako celou stránku.

Client-Centric Programming Model



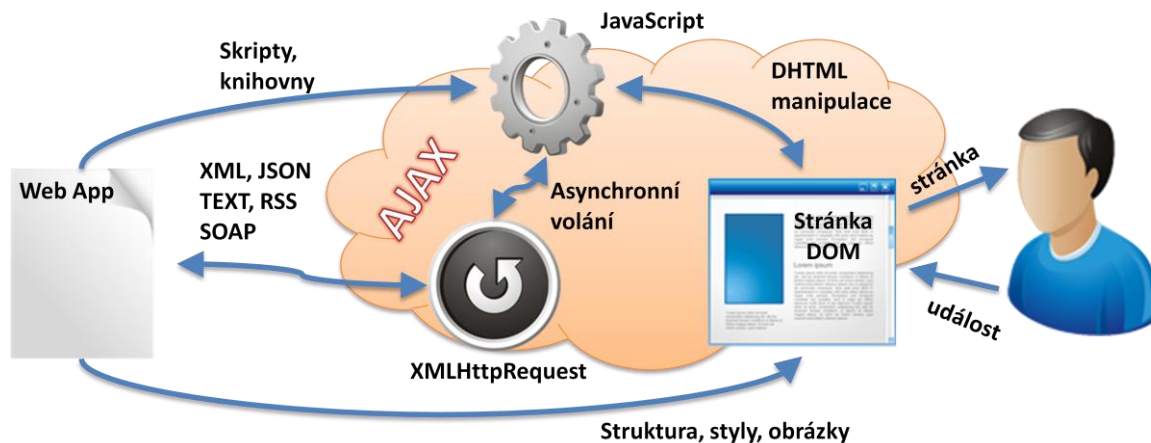
Client-centric model soustřeďuje větší část operací na klienta, serveru zůstává role zdroje dat. Prohlížeč načte výchozí údaje. Pokud uživatel provede nějakou akci, po níž se musí aktualizovat zobrazená data, klient si vyžádá potřebné nové údaje od serveru a sám si aktualizuje zobrazení. Zde se používá například AJAX.



AJAX není ale technologie, ale použití několika různých technologií za určitým cílem.

- (x)HTML a CSS pro prezentaci informací
- JavaScript a DOM (Document Object Model) pro manipulaci s obsahem

- XMLHttpRequest pro komunikaci se serverem
- XML, JSON, text pro přenos dat



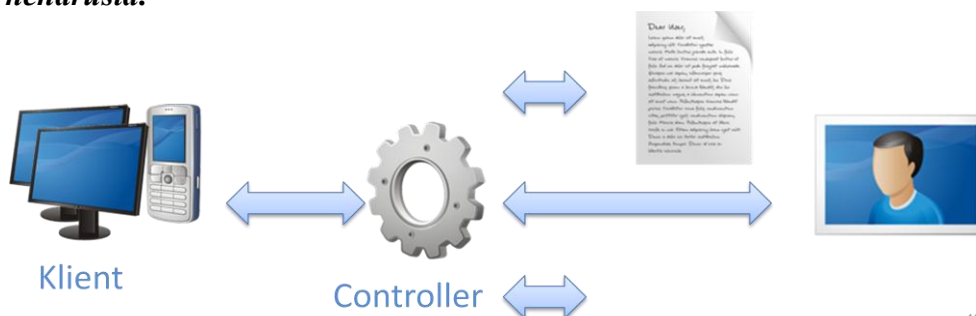
Dělení webových aplikací podle struktury stránek:

Model 1 - Klient přistupuje k jednotlivým stránkám přímo, na základě URL. Vhodné pro decentralizované řízení aplikace. Každá stránka zpracovává své vlastní vstupy.

Jednodušší - hodí se pro menší stránky, ale jeho složitost roste s počtem stránek

Model 2 - Klient přistupuje k centrálnímu prvku „Controller“. Vhodné pro centralizované řízení aplikace. Výstup je vrácen na základě: URL, parametrů i stavu aplikace. Struktura sítě, aplikační vrstvy, její hlavní části.

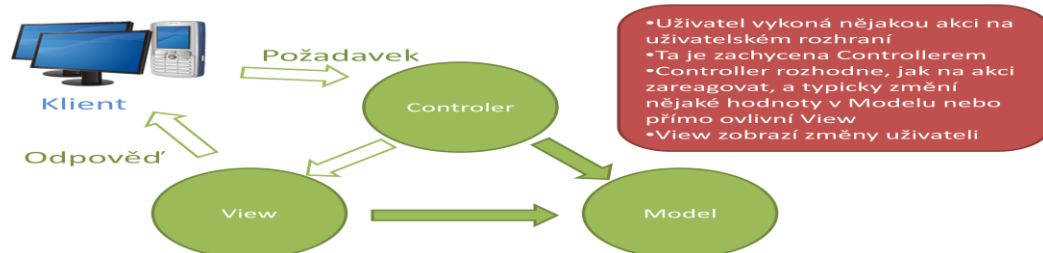
Složitější – větší úsilí pro vytvoření základu, ale s rostoucím počtem stránek již tolik nenarůstá.



Vylepšená model 2 pak používá 3 prvky Model, View a Controller

Model 2 – View – Controller (MVC)

- Architekturu popsal Trygve Reenskaug (1979)



5. Webové komponenty pro vývoj průmyslových internetových aplikací.

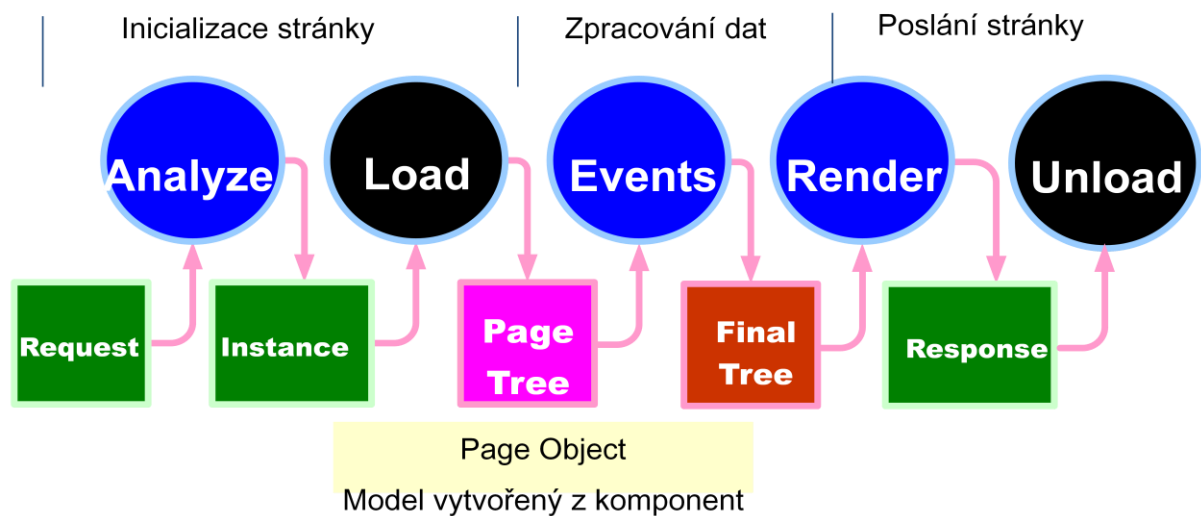
Dva typy komponent (názvosloví podle ASP.NET)

- *HtmlControls* – třídy základních prvků ve webové stránce
- *WebControls* – třídy vyšší abstrakce

HtmlControls i *WebControls* žijící na serveru, jejich instance vzniknou při zpracování stránky, poté se změní pomocí zpracování událostí (požadavků poslaných klientem - event handling), poté vykreslí svoji podobu do html

- pracuje s vlastnostmi a metodami objektů
- zpracovává události a podobně jako v „okénkovém“ programování

Princip objektového chování webového serveru



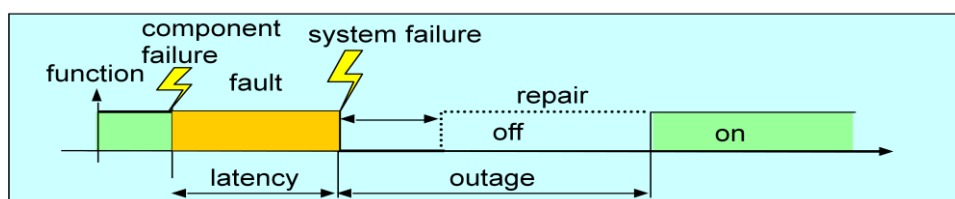
Zpracování události (PostBack) v server centric modelu



6. Zabezpečení, přístupová práva, veřejné klíče, hashovací kódy, digitální podpisy.

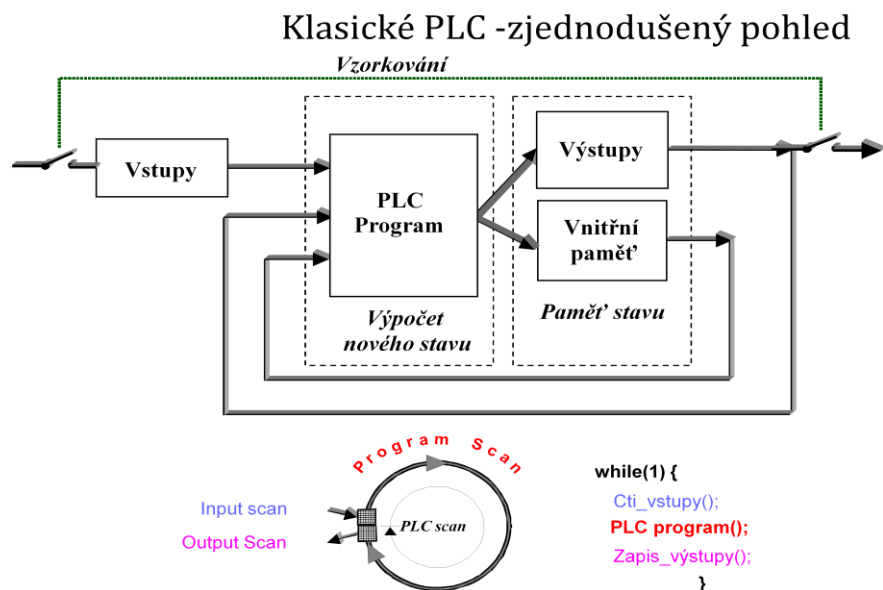
7. Spolehlivost sítí a řešení vhodná pro kritické aplikace.

Definitions: Failure, Fault, Error

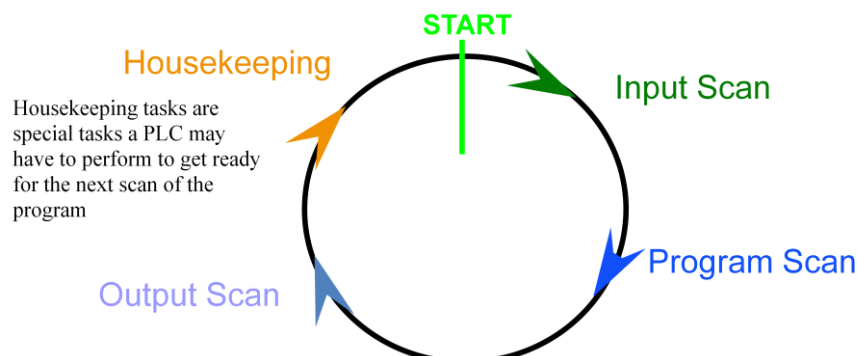


B. Nové okruhy

8. Klasické PLC, tagově orientované PLC (ControlLogix).

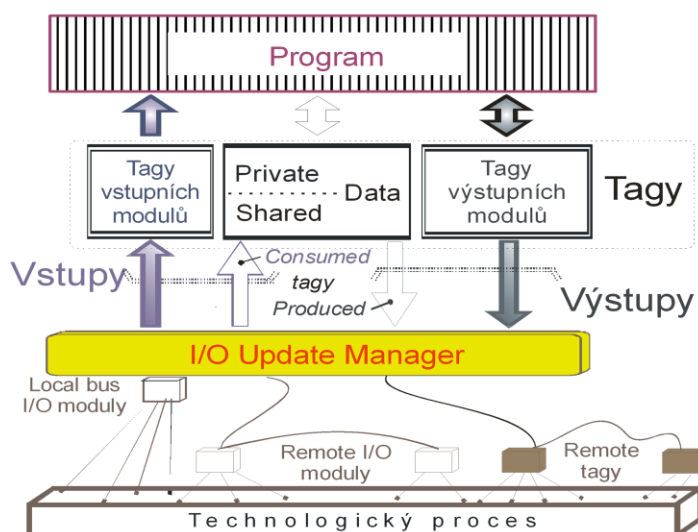


Přesnější obrázek pro PLC Program Scan



- Klasický PLC program se vykonává periodicky. Nepřístupuje se v něm k vstupům a výstupům přímo, ale pracuje s obrazy jejich dat, uložených ve dvou pamětech – v obraze vstupů a v obraze výstupů. Vzniká tak periodický cyklus složený ze tří kroků:
 - **Vzorkování vstupů - scan vstupů** (*input scan*) - načtení hodnot vstupů ze vstupních modulů do paměti zvané obraz vstupů (*input image*).
 - **Výpočet programu - scan programu** (*program scan*) - vykonání celého programu, výpočet nových hodnot výstupů a jejich uložení do paměti zvané obraz výstupů (*output image*).
 - **Zápis výstupů - scan výstupů** (*output scan*) - zápis obrazu výstupů do výstupních modulů.
 - **Režie, vnitřní údržba** (*housekeeping*) – PLC provádí komunikaci, přípravu na další scan....

Tagově orientovaná PLC

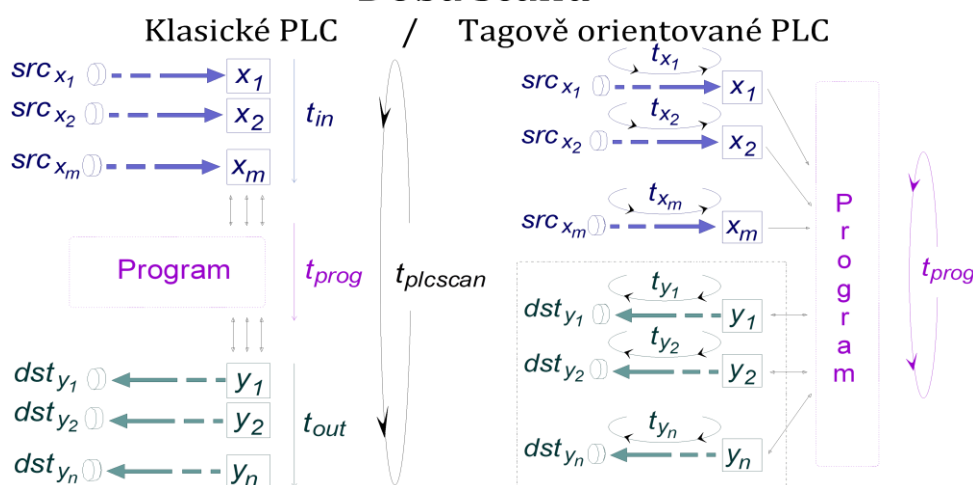


Tagově orientovaná PLC nemají synchronizovaný I/O scan s programem. nenačítají všechna data, ale pouze údaje definované tag.

Tag je obecně pojmenované označení kousku (části) informace – zde formát dat, odkud a po jaké síťové cestě se získávají, v jakých časových intervalech se aktualizují.

Výhodou tagově orientovaného PLC je kratší doba scanu.

Doba scanu -

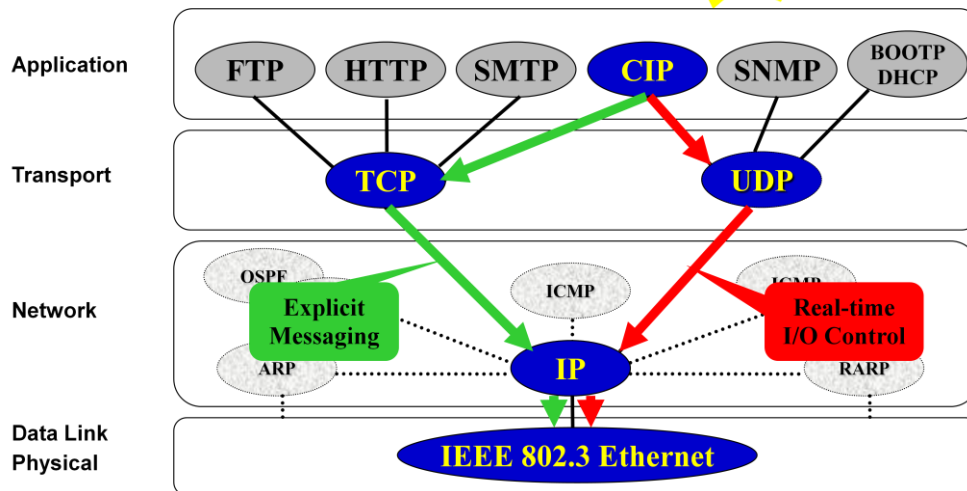
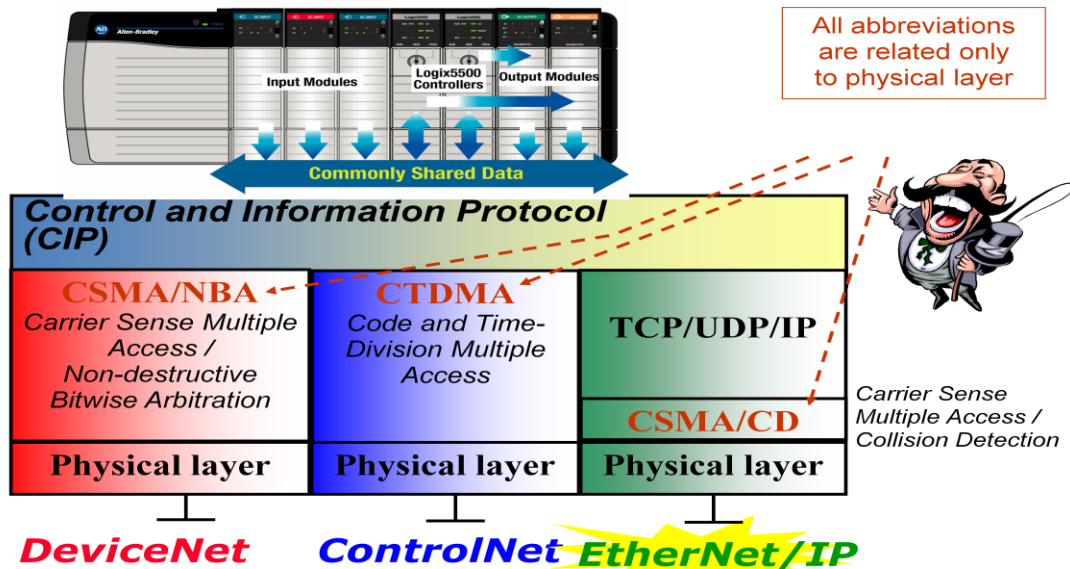


9. Průmyslové sítě PROFINET a ControlNet.

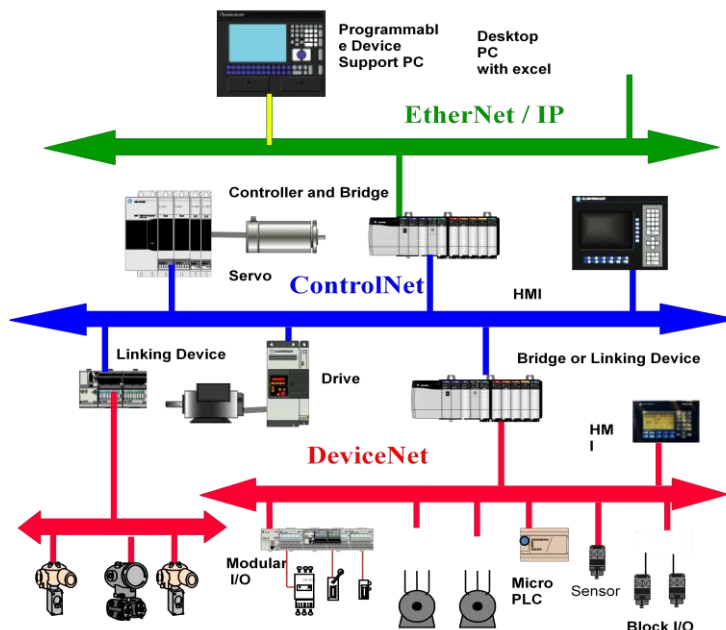
- ControlNet je otevřená síť spravovaná ControlNet International a používá se pro přenos časově kritických a i časově nekritických dat mezi rovnocennými zařízeními.
- Přenesených dat má pevnou rychlost 5 Mbit/s, kde se data přenášejí v cyklech NUT (Network Update Time).
- Časově nekritická data se přenášejí v neplánovaných přenosech.
- Producenti dat zasílají údaje. Způsob přenosu je definován parametry RPI (Request Packed Interval) a COS (Change Of State) v konfiguračních datech. RPI určuje interval, po jehož uplynutí se získávají nová data. Možnost COS umožňuje posílat data pouze v případě, že nastala specifikovaná změna vstupního stavu.

Similarity DeviceNet, ControlNet, Ethernet/IP

Producer/Consumer I/O Model

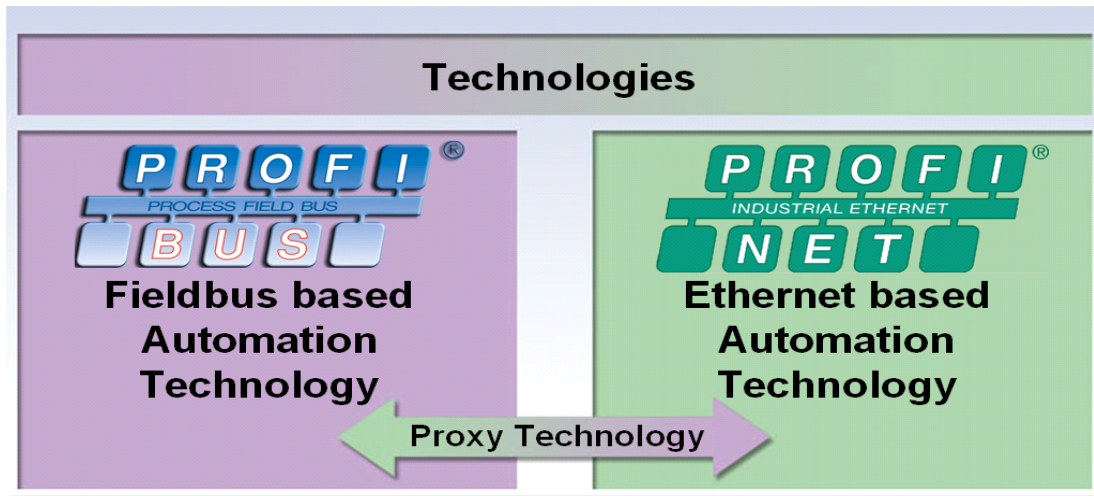


Průmyslové síť Rockwell (Allen-Bradley)

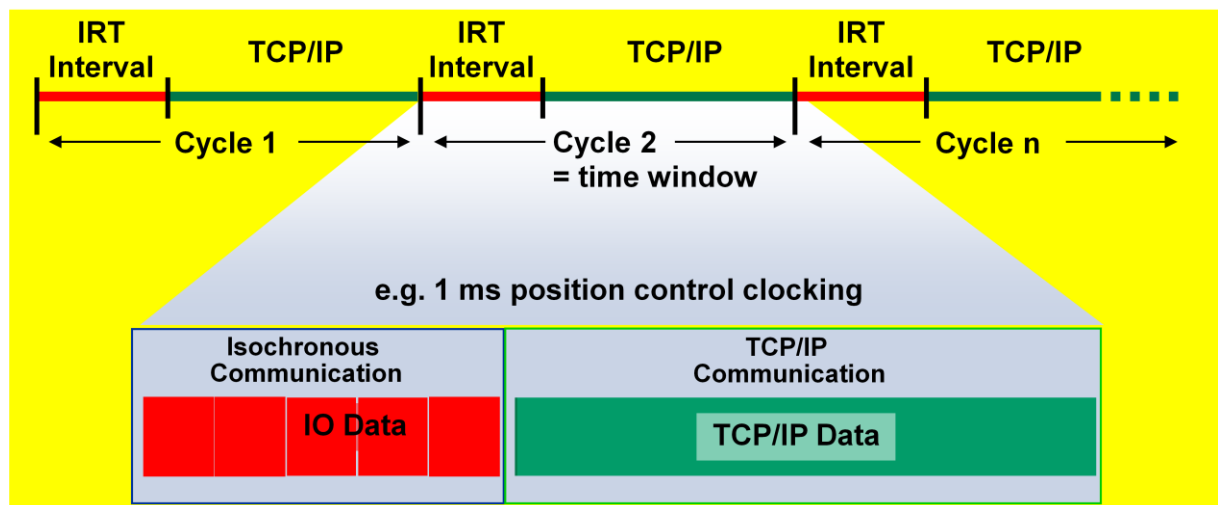


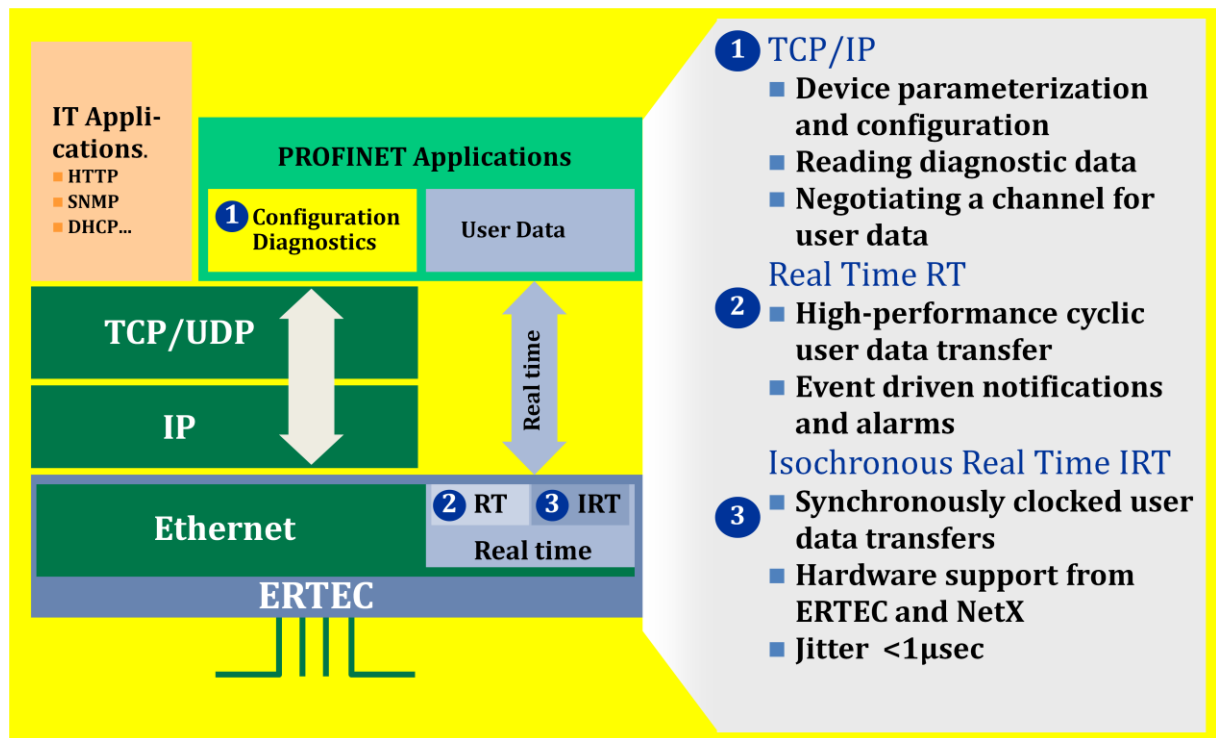
PROFINET

The PROFIBUS User Organization presented PROFINET in 2000 as its solution for industrial automation based on Industrial Ethernet.

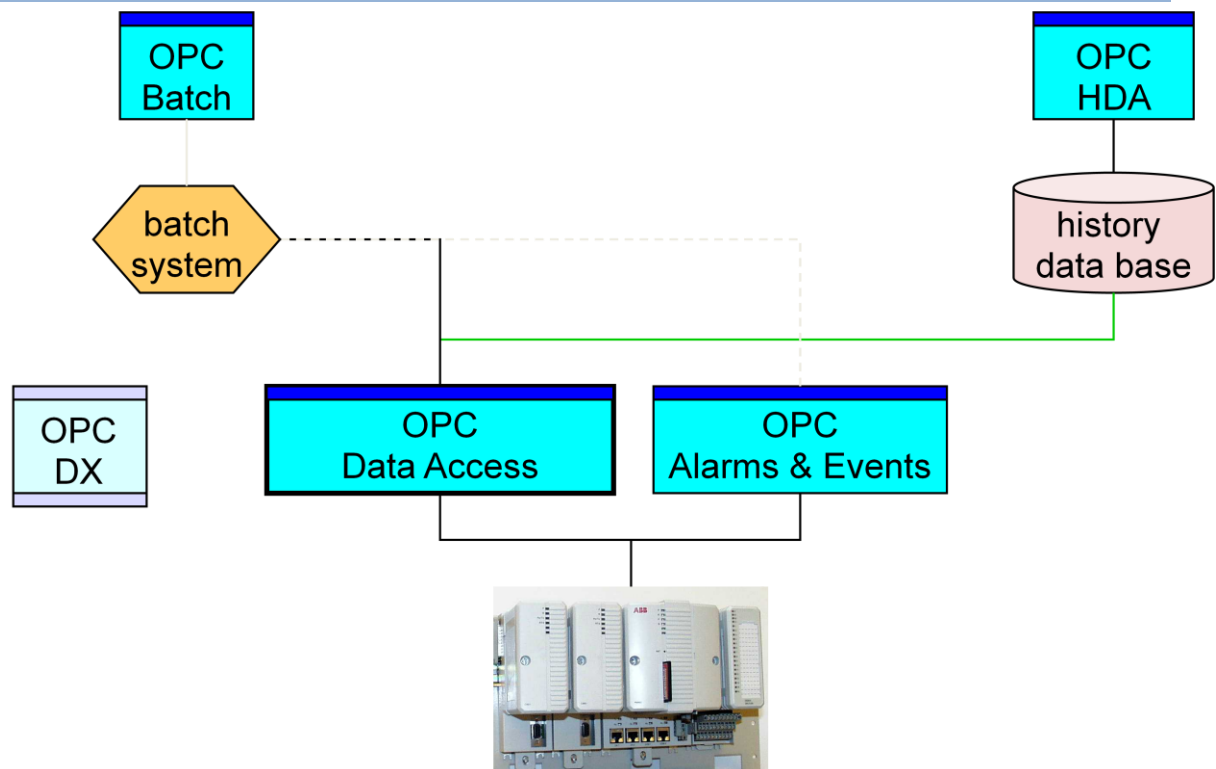


- PROFINET can reserve bandwidth for PROFINET IO and use synchronously clocked data transfers. Bandwidth is reserved by allocating separate time slots for real-time communication with the aid highly cascaded switches -> Precisely timed data transfer
- Note: Basic technology for synchronously clocked applications. RT applications use different asynchronous cycles for applications, data transfers and field devices. RT application are not generally synchronous





10. OPC (OLE for Process Control).



OPC DA for Data Access - The OPC DA (Data Access) specification addresses collecting Process Variables. The main clients of OPC DA are visualization and (soft-) control. Process variables can be sent upon a change, on demand or when a given time elapsed.

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.

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 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.

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.

11.Systémy SCADA a MES (Manufacturing Execution Systems) - principy, organizace, činnost a sběr dat.

Funkčnosti MES podle MESA normy



