



# Agilno upravljanje resursima u industrijskom procesu

Vladan Cvejić  
vladan.cvejic@ensaco.rs

Beograd, decembar 2019.

ensaco

## | Osnovni zahtevi

### Vođenje evidencije opreme (*Asset Management*)

- Održavanje, Dokumentacija
- *Record Based Process Historian*
- Real-time podaci i vizualizacija po potrebi (vendor specific, SVG, HTML 5)

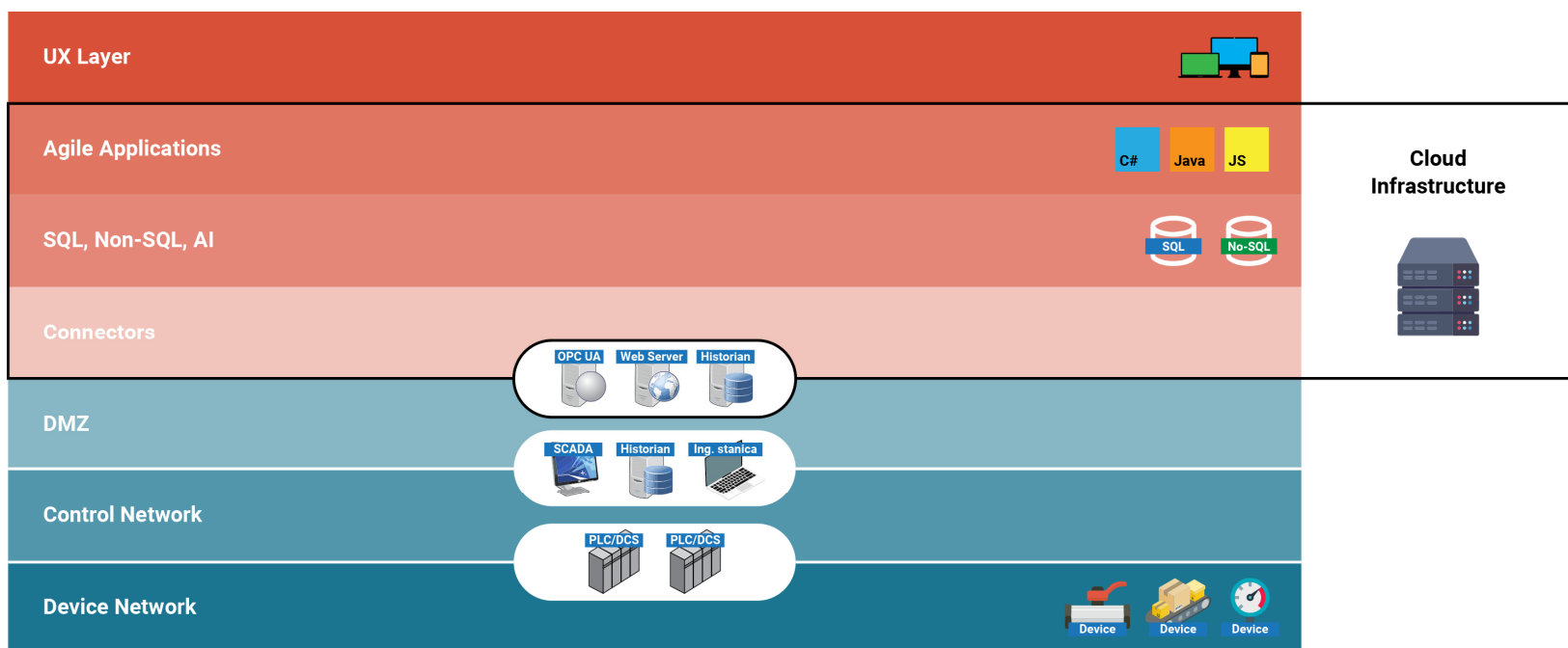
### Integracija aplikacija sa upravljačkim sistemom

- Alarmi
- Izveštaji
- Poruke

### AI/podaci – Preventivno održavanje (*Conditional Monitoring*)

- Pripremljeni podaci za ubacivanje u algoritme veštačke inteligencije

# Arhitektura sistema



## | 2 tima, 2 modela, 2 projekta?

- Inženjeri, rad na tehnološkim delovima problema
- Programeri, rad na delovima problema u poslovnim procesima
- Ne dele zajednički model podataka
- Različite dinamike izmena u sistemima:
  - tehnološki delovi sistema imaju retke zahteve za izmenama
  - poslovni delovi sistema podložni su čestim izmenama
- Konačno rešenje su dva suštinski nezavisna projekta koji su u potpunosti funkcionalno zavisni jedan od drugog

## I Problemi integracije

- Različite metodologije izrade projekta ( *Waterfall* vs. *Agile* ).
- Piramidalna struktura i različita količina podataka na različitim nivoima upravljačkog sistema komplikuje njihov način upotrebe u poslovnom delu sistema.
- Krute tehnološki definisane veze između elemenata sistema upravljanja otežavaju implementaciju različitih upita iz aplikativnog - poslovnog software-a.

# Projekti izrade automatizacije, upravljanja i nadzora industrijskih procesa

## Waterfall metodologija

### Prednosti

- Zahteva visoku disciplinovanost i kvalitet
- Unapred definisani zahtevi definišu kvalitetnu dokumentaciju
- Idealan za projekte u kojima je moguće precizno definisati zahteve

### Nedostaci

- Ne postoji način za povratnu informaciju korisnika u ranim fazama projekta
- Promene u kasnijim fazama projekta su spore
- Rizik od velikih troškova u slučaju nedovoljno dobro definisanih zahteva projekta



# Projekti izrade softvera za poslovne procese

## Agilna metodologija

### Prednosti

- Fokusirana na korisnike
- Veliki uticaj krajnjih korisnika za projekat
- Fokusirana na brzu izradu gotovih funkcionalnosti
- Laka proširivost funkcionalnosti

### Nedostaci

- Izrada dokumentacije nije standardan deo procesa  
*„Working software is more important than comprehensive documentation“*



## Dva modela sistema



### Upravljački model sistema

- hijerarhijski model
- podređen upravljanju procesom
- kruto definisane veze komponenata



### Agilni model sistema

- ne strukturisan model
- pogodan za mikroservisnu arhitekturu
- labave i lako promenljive veze komponenta



## | Standardna rešenja - mane

### Prilagođenje postojećih upravljačkih sistema (Bottom up)

- visoka cena izmene postojećih sistema
- rizik od mogućeg uticaja na funkcionisanje samog upravljačkog sistema
- nefleksibilno rešenje za kasnije izmene koje nisu bile predviđene u startu

### Izrada rešenja koje se naslanja na arhitekturu i modele poslovnih sistema (Top-down)

- visoka cena programera zbog zahteva za poznavanje tehnološkog procesa
- zbog metodologije rada skupe izmene

### „Brand Name“ Rešenje

- visoka cena ovakvih proizvoda
- potrebno je prilagoditi poslovne procese rešenju
- izmene zahtevaju angažovanje skupljih stručnjaka

## | Paralelni „Industrial IoT“ sistem

### Definicija

Sistem koji omogućava direktnu vezu uređaja field bus-a i kontrolera sa agilnim biznis aplikacijama – Smart Factory

### Prednosti rešenja

- lake izmene pri promeni poslovnog dela sistema
- relativna nezavisnost od postojećih upravljačkih sistema
- izmene pri promeni upravljačkog sistema nose mali rizik
- izgradnja sistema korak po korak

### Potencijalne mane rešenja

- razni nivoi „inteligencije“ opreme i upravljačkih sistema u celini
- sistem mora da bude sposoban da funkcioniše na svakom nivou arhitekture od FieldBus do Cloud okruženja
- mogućnost „Vendor Lock“ situacije zbog distribuiranosti ovakvog sistema kroz celu arhitekturu

# | Standardizacija u Industrial IoT rešenjima

## Novi modeli standarda

- pokrivaju komunikacione modele i procese projektovanja
- standardizacija modela podataka opreme istog tipa nezavisno od proizvođača
- smanjenje „*Vendor Lock*” rizika
- malo zavisni ili potpuno nezavisni od načina komunikacije (poput FDT)
- prepoznatljivi načini komunikacije za *Cloud* okruženje

## Problemi uvođenja standarda

- uvođenje standarda u proces projektovanja zahteva veće resurse u početku
- pogrešna primena standarda (...GGIO, OPC-UA kao OPC Classic...)

## | Primeri Standarda primenljivih za Industrial IoT rešenja

### OPC-UA

- u osnovi objektno orijentisani komunikacioni standard
- omogućava prenos strukture modela podataka zajedno sa samim podacima
- lako proširiv modelima drugih standarda (PLCopen, ISA88...)
- podržava transportne mehanizme prepoznatljive modernom Cloud okruženju (HTTP, JSON...)

### IEC 61850

- pokriva sve procese od projektovanja do komunikacije
- definiše standardne modele podataka za tipove opreme
- specijalizovan za energetiku

### IEC 62453 (FDT)

- obezbeđuje nezavisnost opreme od proizvođača
- nezavistan od protokola komunikacije (ProfiBus, DeviceNet, Modbus, OPC-UA...)
- definiše celu vertikalnu protoku podataka od senzora do Cloud okruženja

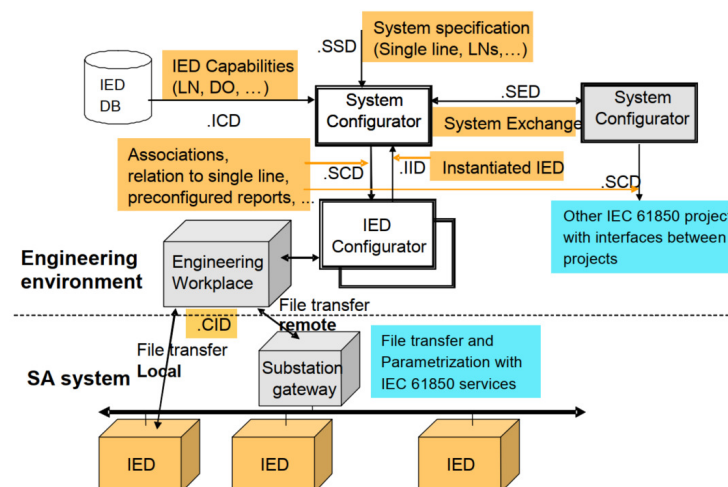
# IEC 61850

## Projektovanje

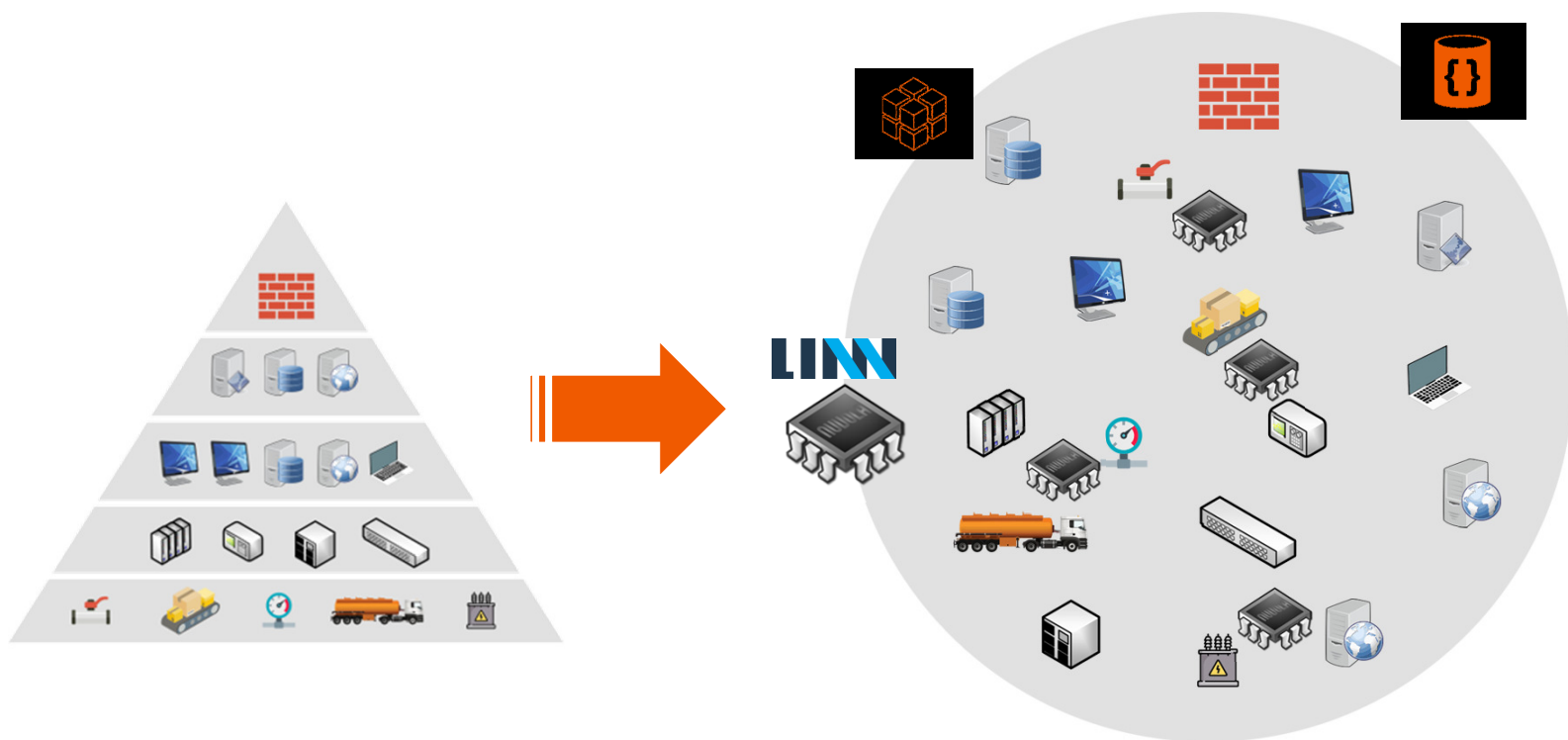
- Standardizovano projektovanje koji je „*Vendor Independent*“
- Definiše *System Configuration Language* SCL – (XML) jezik za definisanje i inženjering sistema
- Interoperabilnost – uređaji različitih proizvođača međusobno razmenjuju informacije i čine funkcionalnu celinu
- CILJ: potpuna digitalizacija

## Komunikacija

- Specificira informacioni model i servise
- Mapira se na moderne komunikacione protokole - *TCP/IP* i *Ethernet*
- Modularan i *Future proof* (model i servisi se mogu mapirati na nove protokole)
- standard sadrži mapiranje na MMS protokol
- standardni načini mapiranja drugih protokola (Modbus, DNP3, IEC 60870-5-104...)



## Korisnički slučaj – Integracija - 1



## | Korisnički slučaj – Integracija - 2



### Industrijska IoT platforma

- potpuna skalabilnost za sve nivoe u arhitekturi (od embedded do Cloud okruženja)
- laka podrška za ubacivanje protokol stekova
- soft-logic podrška za primarnu obradu podataka



### Inženjerski alat

- konfigurator platforme koji sadrži inženjerima prepoznatljive alate
- generisanje koda za više programske jezike (C#, Java, Python...)



### Mikroservisi – platforma za mikro servise

- mogućnost rada na Linux, Windows sistemima kao i u „Cloud“-u
- funkcionalnosti prepoznatljive agilnim aplikacijama



### NoSQL Baza Podataka

- podaci dostupni i u prepoznatljivom formatu za agilne aplikacije

## | Korisnički slučaj – Integracija - 3

### Proces implementacije

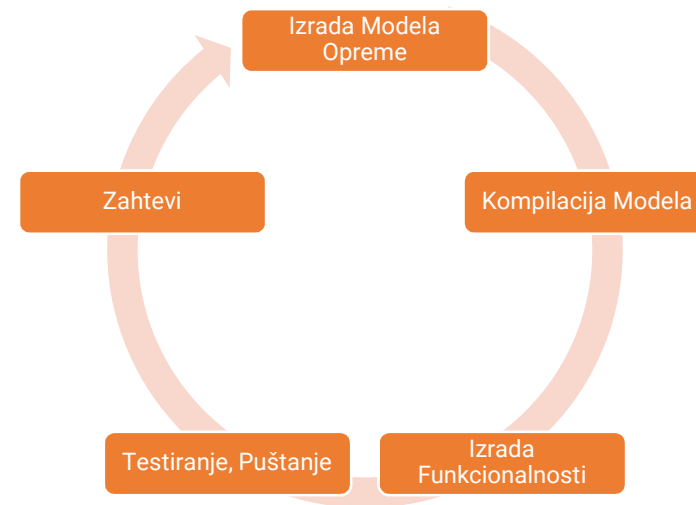
- IIoT platforma ima za cilj dovođenje inteligencije uređaja različitih generacija na jednak nivo
  - Ili se koristi postojeća komunikaciona infrastruktura i uređaji koji zadovoljavaju zahteve
  - Ili se ubacuju interfejs elementi koji „obmotavaju“ uređaj i funkcionalno ga nadograđuju
- Platforma ima dva nivoa konfigurisanja:
  - Procesno konfigurisanje
    - ✓ Izrada modela podataka opreme
    - ✓ Konektovanje na izvore podataka u procesu
  - Priprema za biznis aplikacije (API)
    - ✓ Postavljanje platforme u „*Cloud*„ okruženje
    - ✓ Kompilacija apstrakcija u više programske jezike (C#, Java, Python...)
- Razvoj biznis aplikacija po agilnoj metodi
  - Izrada funkcionalnosti aplikacije po agilnoj metodi
  - Po potrebi uspostavljanje novih zahteva procesnom nivou za izmene modela



## | Korisnički slučaj – Integracija - 4

### Proces implementacije

- Iterativni metod izrade
- deljenje zajedničkih apstrakcija i alata
- direktno uključivanje inženjera u proces izrade





| Hvala Vam na pažnji.

**ensaco**

Palackova 8/IV/7  
11080 Belgrade, Serbia  
[Info@ensaco.rs](mailto:Info@ensaco.rs)