

A large industrial setting featuring several yellow KUKA brand robotic arms working on a production line. The robots are mounted on a white metal frame. The background shows a complex steel truss ceiling and other industrial equipment.

Lezione #5

Sistemi SCADA, Software ERP e MES

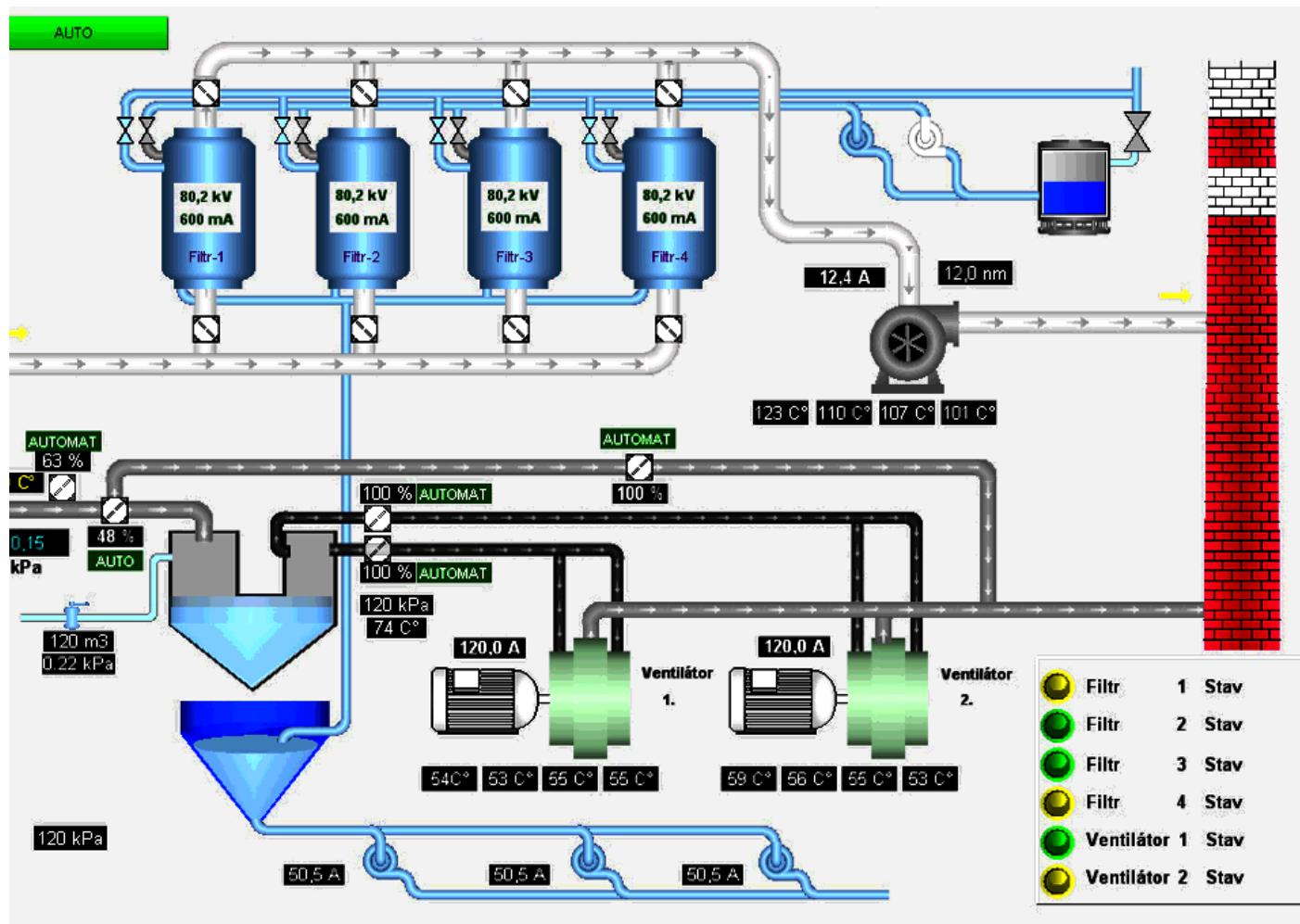
Sommario

- Introduzione ai sistemi SCADA
- Principali funzionalità
- Esempio: DTT
- Cenni ai software ERP e MES

SCADA

- Un sistema **SCADA (Supervisory Control And Data Acquisition)** è un insieme di componenti **software e hardware** che consentono di
 - acquisire dati su un processo in esecuzione
 - effettuare operazioni di supervisione su di esso in maniera **locale o remota**
- Questi sistemi in genere consentono a un **operatore** di interfacciarsi con il **processo** tramite una **HMI** sfruttando i **sistemi di controllo**

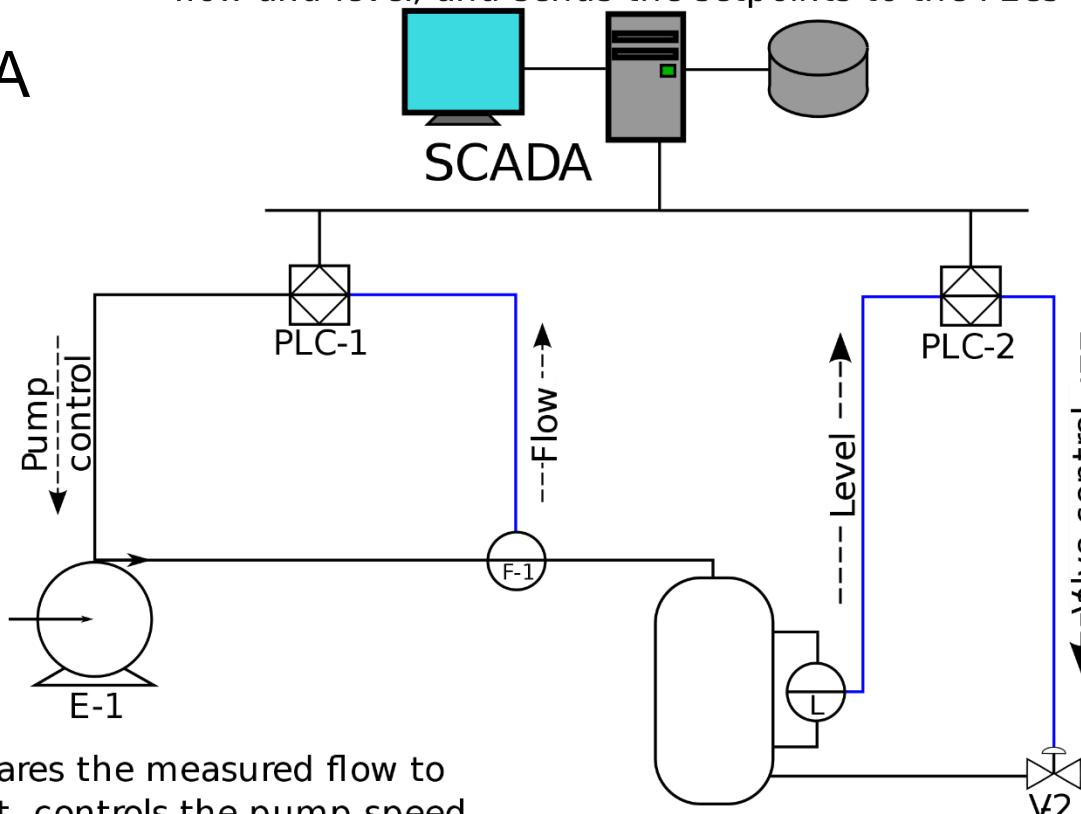
SCADA



SCADA

- Schema di un semplice SCADA

The SCADA system reads the measured flow and level, and sends the setpoints to the PLCs



PLC1 compares the measured flow to the setpoint, controls the pump speed as required to match flow to setpoint

PLC2 compares the measured level to the setpoint, controls the flow through the valve to match level to setpoint

SCADA

- **Sensori e attuatori** interagiscono con le **grandezze fisiche** di processo
- I **dispositivi di controllo** (e.g. PLC) interagiscono con sensori e attuatori, memorizzando i dati in una memoria locale e realizzando le **logiche di automazione** dell'impianto
- I dispositivi di controllo sono connessi via **rete** a uno o più **dispositivi di supervisione**
- Questi dispositivi **raccolgono e storicizzano i dati**, li **presentano all'operatore** tramite HMI e/o informazioni riassuntive, e forniscono un **supporto alla decisione** per la gestione dell'impianto

SCADA

Funzioni principali

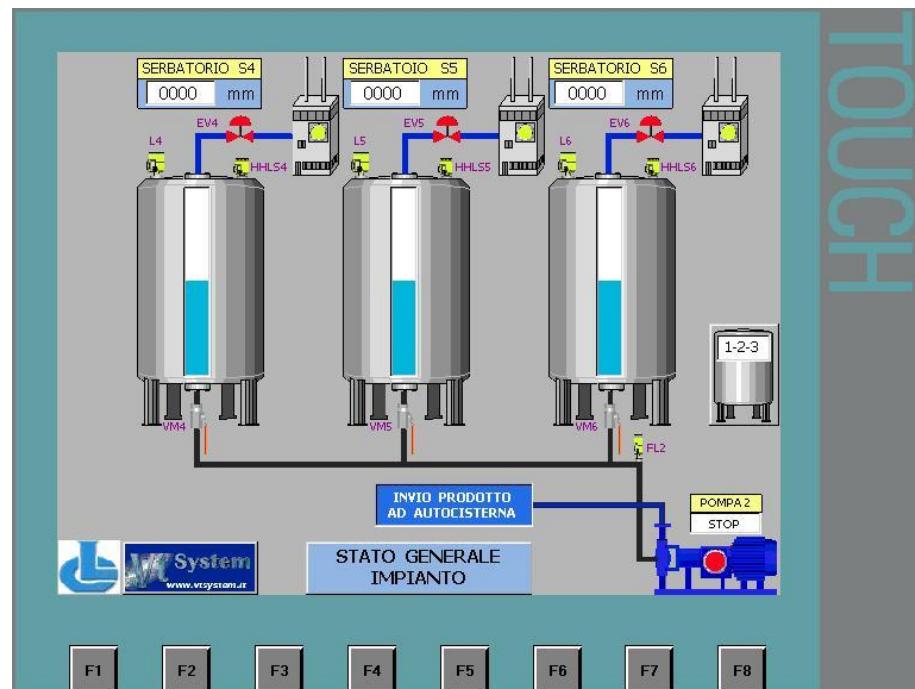
- **Acquisizione dati**
 - Per poter fornire informazioni sul processo, è necessario che il sistema **acquisisca dati** da esso mediante opportuni **driver di comunicazione**
 - Questi dati vengono raccolti in una **base dati**, che costituisce il nucleo del sistema SCADA
 - Ai dati grezzi possono essere aggiunte altre informazioni come l'identificativo del sensore, stringhe informative, unità di misura e/o conversione del valore in unità standard...
- **Controllo di supervisione**
 - Tramite uno SCADA, l'operatore può **supervisionare il processo** ed intervenire su di esso se necessario
 - Questa funzione si realizza tipicamente tramite una **Human-Machine Interface (HMI)** di tipo grafico

Oltre a queste, uno SCADA moderno in genere assolve ad altre funzionalità

Rappresentazione dei dati

Spesso lo SCADA presenta i dati all'operatore tramite una HMI, che in genere comprende:

- **Quadri sinottici**, con elementi statici e dinamici (a seconda che l'aspetto grafico vari in base al dato)
- **Pannelli di controllo**, consentono all'operatore di interagire con l'impianto in maniera intuitiva (manopole, pulsanti, slider...)



SCADA

Funzioni aggiuntive

- In genere per programmare le HMI è necessario del **software dedicato**
 - Siemens
 - Allen-Bradley
 - Schneider
 - GE Digital

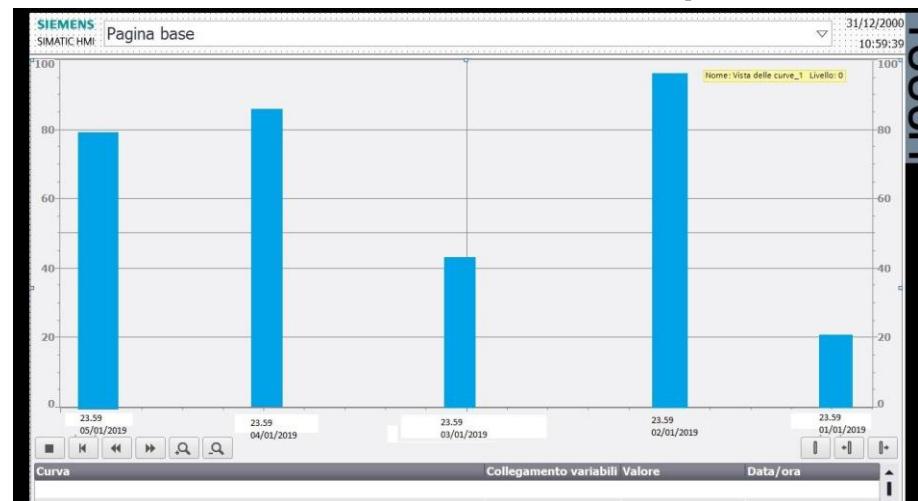


FactoryTalk[®]



Storizzazione dei dati

- Spesso, gli SCADA possono anche restituire **serie storiche di dati** per fare analisi di tipo gestionale e tenere traccia dei ritmi di produzione (efficientamento del processo, gestione di criticità...)
- Possono anche generare automaticamente dei *report* a intervalli prefissati



Gestione di allarmi

- Lo SCADA consente di programmare la condizione che causa l'insorgenza di un allarme
- All'insorgere di una condizione di allarme, lo SCADA provvede a **informare l'operatore** (lampeggianti, sirene, e-mail, SMS...)



Gestione di allarmi

- Tipi di allarmi:
 - **A insorgenza**
Permangono fino a una reinizializzazione del sistema
 - **A insorgenza e riconoscimento**
Permangono fino a quando l'operatore non ne comanda la disattivazione (*riconoscimento*)
 - **A insorgenza, riconoscimento e rientro**
Permangono finché non avviene il riconoscimento e la condizione di attivazione viene meno



Gestione di ricette

- Gli SCADA possono eseguire particolari sequenze di operazioni programmate dall'utente (dette *ricette*, *lotti* o *batch*)
- Tali ricette possono essere eseguite:
 - Al verificarsi di una **scadenza temporale** prefissata
 - All'insorgere di **eventi** particolari (ad esempio in risposta a un allarme)
 - Su **richiesta** dell'operatore



Supporto alla manutenzione

- La manutenzione può essere **correttiva** o **preventiva**
- Lo SCADA può fornire all'utente dei **piani di manutenzione**
- Può anche aiutare a coordinare tali operazioni, che potrebbero dover essere eseguite su un **numero elevato** di dispositivi



Interazione con sistemi di livello superiore

- Agli SCADA moderni è spesso richiesto anche di interfacciarsi con altri sistemi
- Esempi:
 - *Domain Controller* per la gestione degli accessi e l'autenticazione degli utenti
 - Sistemi MES (*Manufacturing Execution System*) per gestire il dispaccio di ordini, le scorte di magazzino e in generale per gestire e controllare la produzione
 - Software gestionali ERP (*Enterprise Resource Planning*)

Evoluzione dei sistemi SCADA

- Nel corso del tempo, i sistemi SCADA hanno subito una significativa evoluzione
- I primi SCADA erano spesso realizzati su mainframe o minicomputer con sistemi di comunicazione proprietari
- Oggi si va nella direzione di sistemi **distribuiti**, spesso **web-based**
- La connessione dei sistemi SCADA a reti aziendali o a internet ha aumentato i **rischi per la sicurezza** in tali sistemi (e.g. Stuxnet)

SCADA vs DCS

- Nella lezione precedente, abbiamo parlato di DCS (Distributed/Decentralized Control System) come di un paradigma che descrive un sistema di automazione ‘a livelli’
- In alcuni casi però, la dizione DCS è utilizzata anche per indicare dei sistemi di controllo e supervisione integrati simili a uno SCADA

SCADA vs DCS

- In questa accezione, SCADA e DCS assolvono a funzionalità simili:
 - **integrare** più sottosistemi
 - acquisire ed elaborare **dati**
 - scambiare **informazioni con il campo**
 - il tutto in **architettura distribuita**
- In alcuni casi la distinzione tra i due non è ben definita

SCADA vs DCS

- In genere si indicano con
 - SCADA i sistemi che comunicano tramite **reti WAN** (negli ultimi anni si sta diffondendo Ethernet + TCP/IP)
 - DCS sistemi che raggruppano numerosi nodi controllati in una rete **LAN**

SCADA vs DCS

- Una differenza importante è nel modo in cui interagiscono con il processo
 - Uno **SCADA** fornisce un'indicazione, che va poi gestita dai dispositivi di controllo ('cosa fare', ma non 'come fare')
 - I dispositivi dovrebbero essere in grado di svolgere il compito anche se lo SCADA si disconnette o ha un guasto
 - Possono eventualmente ritornare un codice di errore, che viene tradotto in un **allarme**
 - In un **DCS** l'interazione tra dispositivi di controllo e di supervisione è più stretta
 - Spesso il supervisore nei DCS è implementato su un RTOS
 - L'utilizzo di reti locali consente di garantire i tempi di comunicazione

SCADA vs DCS

- L'interazione tra operatore e sistema di controllo avviene in entrambi i casi tramite HMI
- In genere per programmare le HMI in SCADA è necessario del **software dedicato**
- Spesso i DCS hanno dei tool di visualizzazione **integrati**
 - Questo semplifica anche il mapping delle variabili tra programma di controllo e interfaccia (i tag sono automaticamente disponibili)

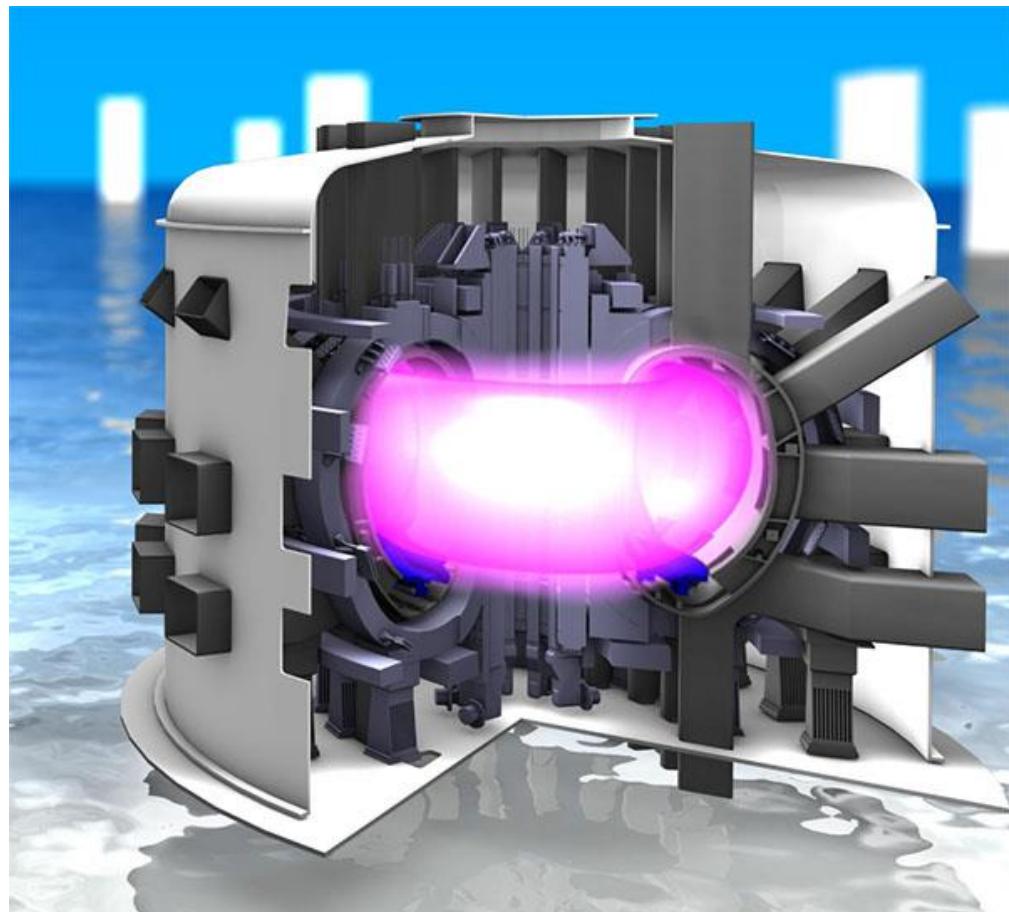
- In generale i DCS implicano **costi più onerosi**
 - reti dedicate
 - sistemi real-time
 - gestione di malfunzionamenti tramite ridondanze
- Solitamente garantiscono maggiore **affidabilità e sicurezza**

Esempio: DTT

Il **Divertor Tokamak Test facility** (DTT) è un prototipo di reattore a fusione nucleare che dovrebbe essere costruito nei prossimi anni nei laboratori di ENEA Frascati

I **Tokamak** sono macchinari complessi, in cui le funzioni di controllo e monitoraggio svolgono un ruolo critico

Nell'ultimo Interim Design Report è incluso anche il design concettuale di un sistema SCADA



Esempio: DTT

DICS: DTT Instrumentation and Control System

“

DTT Instrumentation and Control System, DICS, is a combination of systems and subsystems that collectively allow automating the operation of the DTT device.

- DTT Interim Design Report, 2019

Esempio: DTT

Componenti:

- **Central control**
 - Central Command and Control System (**CCCS**)
 - Central Machine Protection Systems (**CMPS**)
 - Central Occupational Safety Systems (**COSS**)
- **Local control**
Plant System Controllers, organizzati in Plant System Groups (**PSGs**)
- **Distributed control**
Real Time Control Systems (**RTCS**)
- **Reti di interconnessione**

Esempio: DTT

Central Control System

CCCS

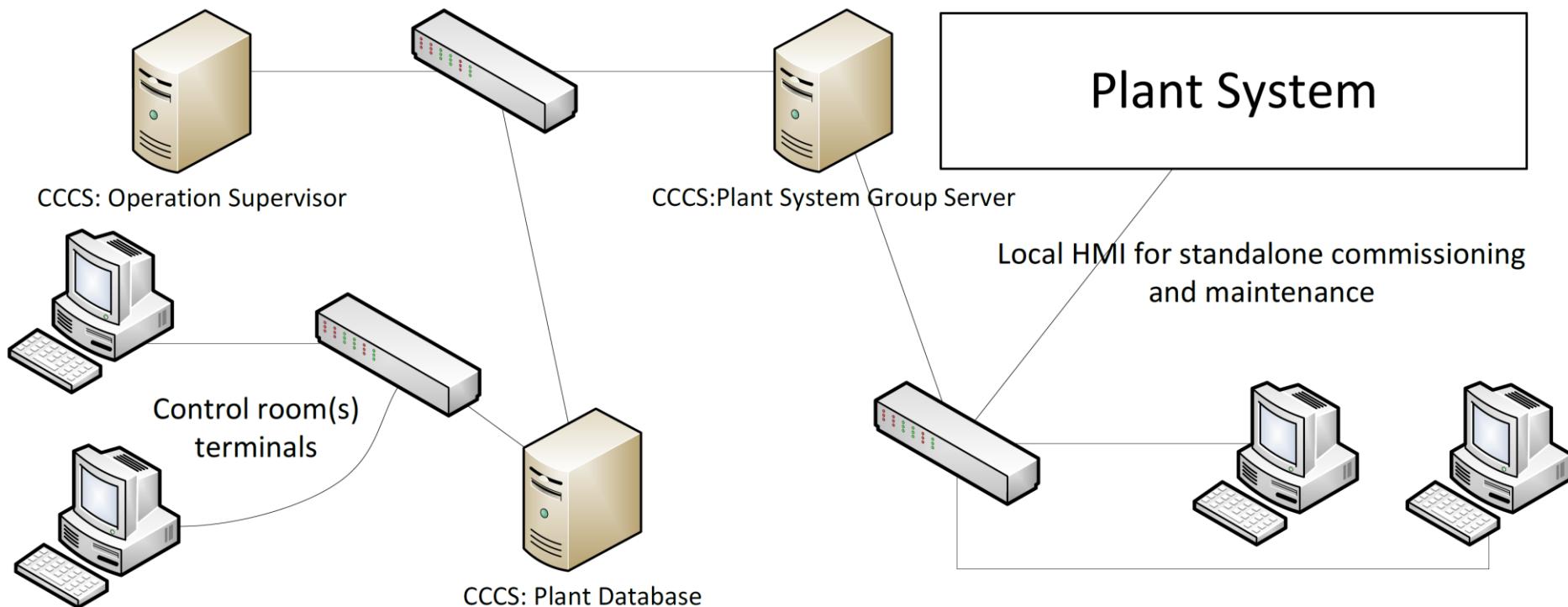
- Funzioni di controllo e coordinazione **non critiche e non real-time**
- Interfacce per gestire le operazioni, terminali utente, server per gestire il database dell'impianto e le ricette, meccanismi di comunicazione agli altri sistemi

CCCS

- Database
- Interfacce operatore e quadri sinottici
- Programmazione degli esperimenti
- Accesso ai dati sperimentali e Historical Experiment Database (**HED**) (→ MDS+)
- Coordinazione automatica dei sottosistemi
(Operation Supervisor)

Esempio: DTT Central Control System

CCCS



Esempio: DTT

Central Control System

CMPS

- Funzioni di controllo e coordinazione **critiche e real-time**
- Gestisce guasti e malfunzionamenti per minimizzarne gli effetti negativi
- Gestisce anche il clock, le reti di comunicazione RT e le reti per la Machine Protection
- Le strategie di gestione effettive sono implementate a livello dei *Plant Systems*, e vengono coordinate dal CMPS

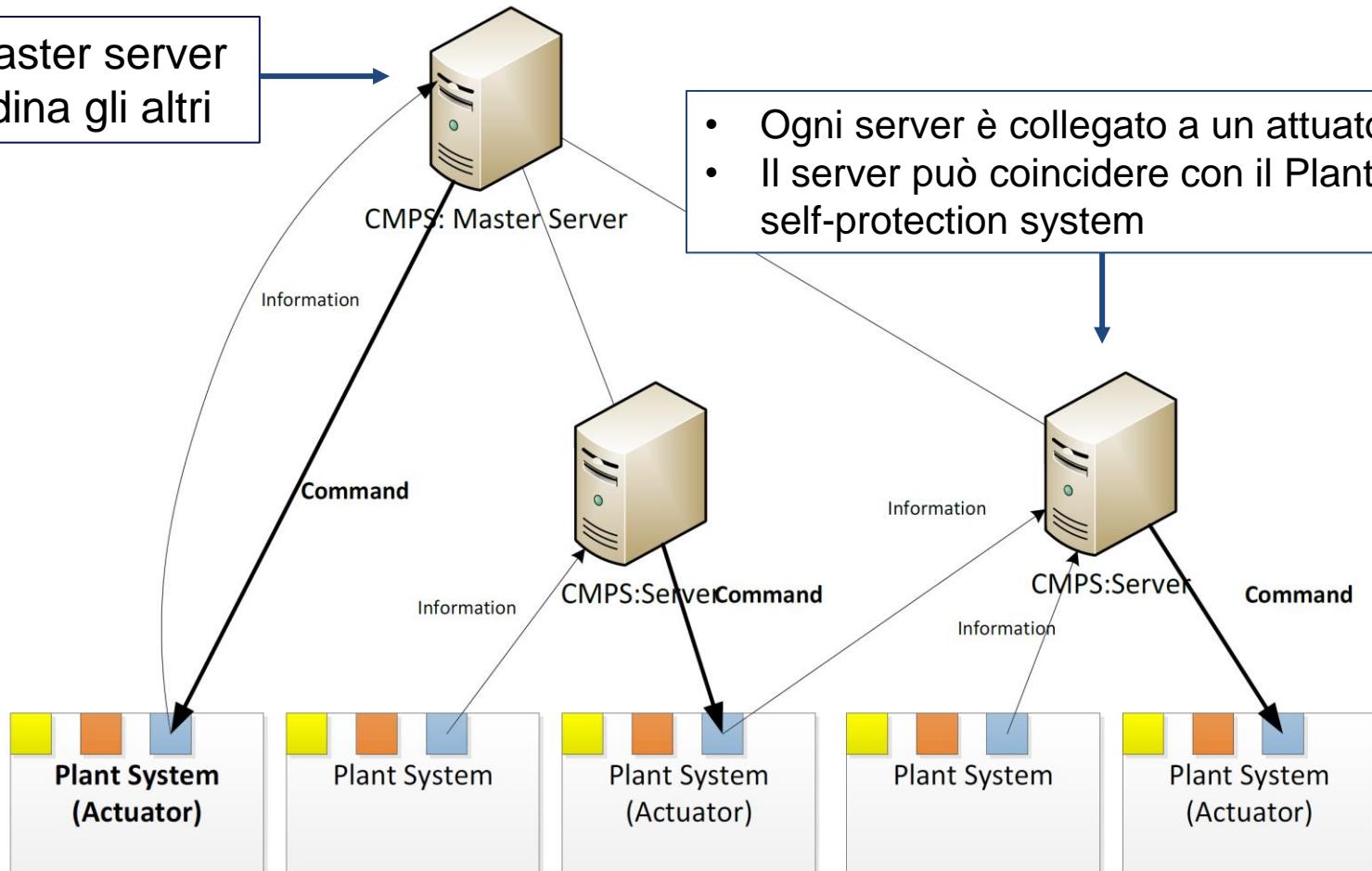
CMPS

Critical vs non-critical protection functions

- sono considerate critiche le condizioni che possono causare gravi ritardi nelle operazioni o costi di manutenzione/riparazione elevati
- *quality-control* necessario su tutte le componenti coinvolte nella gestione di queste evenienze

Esempio: DTT Central Control System

Un master server coordina gli altri



Il numero e i compiti di ogni server sono scelti in modo da
minimizzare la complessità del sistema!

COSS

- Funzioni di controllo e coordinazione **critiche e real-time**
- Gestisce gli eventi che possono causare danno a persone in prossimità dell'impianto
- Include il controllo degli accessi ad aree potenzialmente pericolose, funzioni di *interlock* (ad esempio per sorgenti di alta tensione...), ecc.

Plant Systems

- **Standard Industrial Control Platform: PLC**
 - Main+backup
 - Possibili framework per lo sviluppo di SW in-house sono UNICOS e ITER Industrial Control Library
 - *'Whenever a tender will be placed, some level of contractual pressure will be placed so that tenderers will adopt this platform'*

Plant Systems

- **Standard Computer Platform:** Linux
 - Distro standard per il CCCS
 - Installazioni minimali con kernel semplificati per i Plant System (disponibilità ↑ - vulnerabilità ↓)
 - Real-Time dove necessario
 - Framework [MARTE2](#) per lo sviluppo di applicazioni di controllo RT

Plant Systems

- **Data Acquisition Hardware**
 - PLC remote I/O cards per performance «moderate» (<1kHz)
 - National Instruments C-RIO + Linux PC nei casi in cui sono necessarie performance più spinte (con l'uso di FPGA)
 - Piattaforma per prestazioni elevate **ancora da definire**

Plant Systems

- **Interfaccia col CCCS - soluzioni «preferite»:**
 - Native interface to the chosen Supervisory Control and Data Acquisition (SCADA)
 - [OPC-UA](#)
 - [RESTful](#)

Esempio: DTT

Distributed Control System

RTCS

- Funzioni perlopiù **non critiche**
- Controllori, sistemi di diagnostica e attuatori specifici per l'applicazione considerata
- Framework MARTe2 + Linux per lo sviluppo

Networks

- **Command and control network**
 - Ethernet-based
 - Comunicazione tra CCCS e server PSG, terminali in control-room, partecipanti remoti
 - *Field network* basata su un mix di IP e *fieldbus*
- **Data Acquisition Network**
 - TCP/IP-based
 - Usata per trasmissioni a frequenze >100Hz
 - I dati non vengono conservati localmente nei Plant System

Networks

- **Real Time Network**

- Protocollo Point-to-multipoint (MULTICAST) UDP
- Traffico stazionario: tutti i nodi contribuiscono con **carichi noti** e con **pattern periodici** sincronizzati al *central timing system*
- Topologia: **switched multi-star**

- **RTN Simulation Support**

- La rete RT deve anche consentire test *system-in-the-loop* per testare combinazioni di Plant System usando input sintetici
- Simulazioni offline per testare il software real-time dei Plant System su piattaforme PC (in **Simulink**)

Esempio: DTT

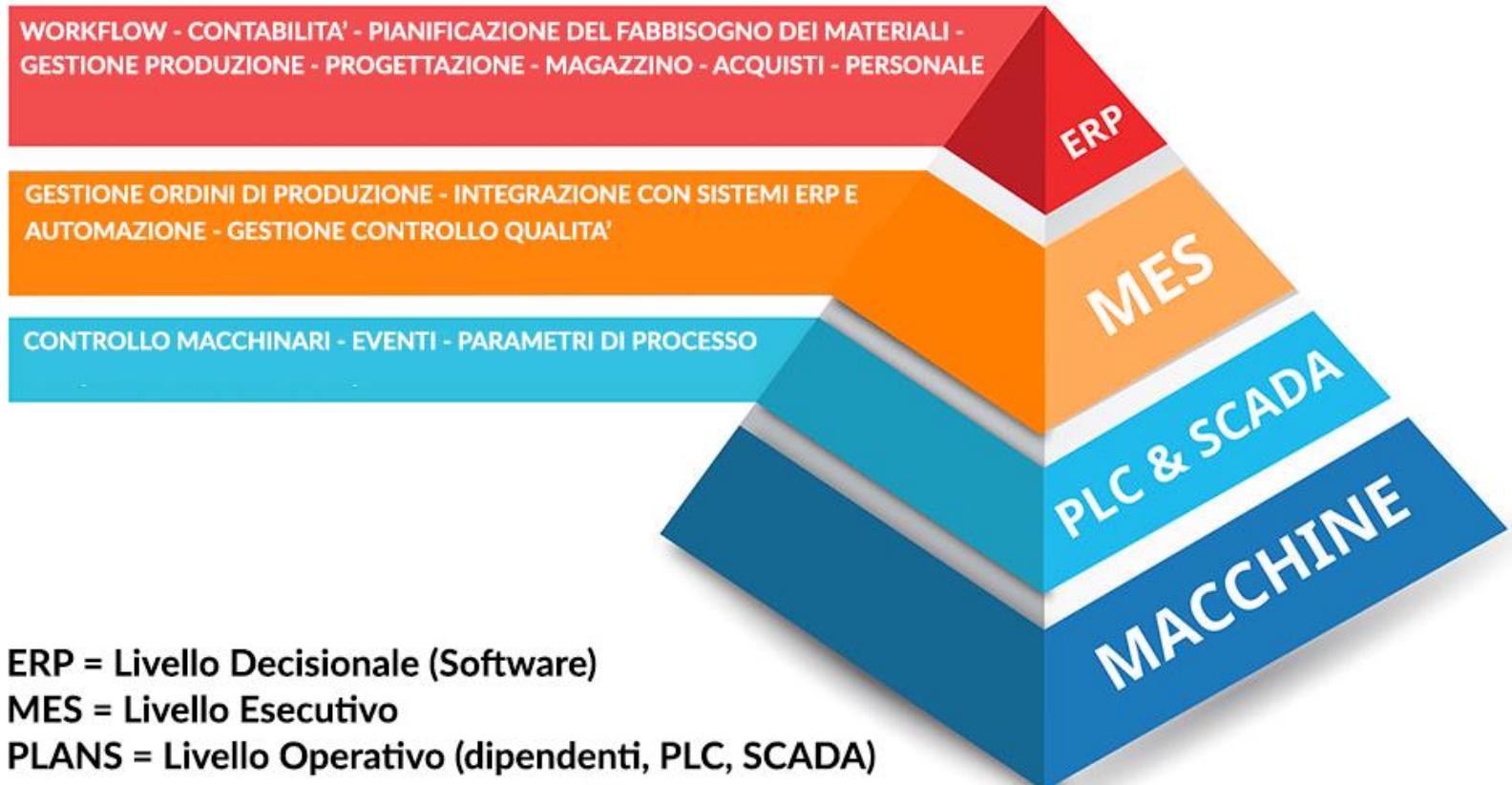
Local Control System

- La tecnologia SCADA da adottare è in fase di definizione
- Preferenza per sistemi **industriali e scalabili** che possano essere facilmente integrati con altri prodotti (e.g. WINCC-OA)

ERP, MES

- Al di sopra dei sistemi di supervisione e acquisizione dati, nel modello CIM troviamo i livelli di stabilimento e azienda
- Anche questi livelli sono caratterizzati dai loro sistemi SW

ERP, MES



da entersoftware.it

“

Il MES è un sistema che acquisisce e distribuisce informazioni che consentono l'ottimizzazione delle attività produttive dal lancio dell'ordine al prodotto finito.

Utilizzando dati real-time, attuali ed accurati, il MES guida, risponde e informa sulle attività dello stabilimento e dei reparti produttivi così come e quando esse accadono.

- MESA international

Che funzioni svolge un sw MES?

- Gestione degli **ordini**
 - avanzamento
 - quantità prodotte/scartate
 - date d'inizio e fine lavorazione
 - stato dell'ordine
 - identificazione tramite barcode

Che funzioni svolge un sw MES?

- **Gestione dipendenti**
 - ruoli
 - turni
 - presenze, entrate/uscite
 - operazioni a cui sono abilitati
 - operatori che hanno preso in carico una lavorazione o un guasto

Che funzioni svolge un sw MES?

- **Gestione delle risorse**
 - macchinari disponibili/guasti
 - gestione di allarmi e fermi macchina

Che funzioni svolge un sw MES?

- **Reportistica**
 - scheda di lavorazione di un ordine
 - rilevamento tempi di produzione
 - quadratura presenze/produzione
 - report su guasti e fermi
 - rilevamento tempo medio di elaborazione
 - identificazione dei ‘colli di bottiglia’ nella produzione

Domande a cui risponde un MES

- Il cliente vuole sapere se, raddoppiando il suo ordine, la fornitura può ancora essere garantita nei tempi previsti
- Il cliente vuole sapere a che punto è la lavorazione del suo ordine
- Il cliente richiede che ogni prodotto consegnato sia corredata da informazioni dettagliate sulla produzione
- Una macchina si guasta. Come riprogrammare gli ordini per consentire la manutenzione garantendo però la puntualità di consegna al cliente?
- Qual è il lead time di produzione? Che cosa “rallenta” la produzione?
- Un prodotto viene ritirato dal commercio perché presenta delle non conformità, quali lavorazioni ha subito e quando le ha subite? Qual è l'esatta provenienza delle materie prime?
- Il cliente vuole il documento di collaudo con i dettagli dei controlli qualità eseguiti per la certificazione del prodotto
- L'utilizzo dei materiali (materie prime, macchine) è ottimizzato?
- Quanto è efficace ed efficiente il processo produttivo?

da [Qualitas Informatica](#)

- Con **Enterprise Resource Planning** si indicano in genere **sw gestionali** in grado di gestire i processi di *business* rilevanti per un'azienda
- In genere si tratta di sw **modulari**, tipicamente con un **database comune** che garantisce che i dati siano corretti e aggiornati
- La struttura modulare consente di inserirli *progressivamente* in un'azienda quando questa cresce al punto da sentirne il bisogno

- Contabilità
- **Material Requirements Planning**
(gestione e pianificazione dell'acquisto dei materiali in base a *lead time* di produzione, scorte attuali, distinta base, previsione di vendita)
- Gestione acquisti e magazzini
- Gestione del personale (*Human Resources*)
- Gestione degli impianti e della manutenzione
- Gestione delle vendite e della distribuzione
- ecc.

Riferimenti e Risorse

- [3] Cap. 12
- [What is SCADA?](#) @RealPars 
- [Modbus](#) @RealPars (→ ScadaBR) 
- [Lista di software SCADA](#) open source
- [Note su SCADA](#) (S. Bimbo e E. Colaiacovo)
- [ERP, MES, PLM](#)
- [DTT Interim Design Report](#)
- [ERP, MES, PLM](#) @dmsolutions
- Altro su [MES](#) @Qualitas informatica
- [What is ERP](#) @Oracle
- [MES](#) @Entersoftware



Fine Lezione #5

Sistemi SCADA

Cos'è un sw ERP (Oracle)

- ERP systems are designed around a single, defined data structure (schema) that typically has a common database. This helps ensure that the information used across the enterprise is normalized and based on common definitions and user experiences. These core constructs are then interconnected with business processes driven by workflows across business departments (e.g. finance, human resources, engineering, marketing, operations), connecting systems and the people who use them. Simply put, ERP is the vehicle for integrating people, processes, and technologies across a modern enterprise.

Cos'è un sw ERP (Oracle)

- For example: consider **a company that builds cars** by procuring parts and components from **multiple suppliers**. It could use an **ERP system to track the requisition and purchase of these goods and ensure that each component across the entire procure-to-pay process uses uniform and clean data** connected to enterprise workflows, business processes, reporting, and analytics. When ERP is properly deployed at this automotive manufacturing company, **a component**, for example, “front brake pads,” **is uniformly identified by part name, size, material, source, lot number, supplier part number, serial number, cost, and specification**, along with a plethora of other descriptive and data-driven items. Since data is the lifeblood of every modern company, ERP makes it easier to collect, organize, analyze, and distribute this information to every individual and system that needs it to best fulfill their role and responsibility.

Cos'è un sw ERP

ERP also ensures that these data fields and attributes roll up to the correct account in the company's general ledger so that all costs are properly tracked and represented. If the front brake pads were called "front brakes" in one software system (or maybe a set of spreadsheets), "brake pads" in another, and "front pads" in a third, it would be tough for the automotive manufacturing company to figure out how much is spent annually on front brake pads, and whether it should switch suppliers or negotiate for better pricing.

Cos'è un sw ERP

A key ERP principle is the central collection of data for wide distribution. Instead of several standalone databases with an endless inventory of disconnected spreadsheets, ERP systems bring order to chaos so that all users—from the CEO to accounts payable clerks—can create, store, and use the same data derived through common processes. With a secure and centralized data repository, everyone in the organization can be confident that data is **correct, up-to-date, and complete.**