

SAVE

ANIE
AUTOMAZIONE



Edge Computing:

un passaggio essenziale per la smart manufacturing

Damiano Di Rosa

Schneider
Electric

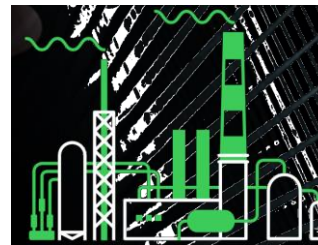
Perchè l'Edge ?

“Nessuno vuole più dipendere solo dalle comunicazioni verso il Cloud per gestire le risorse più critiche”

Jean-Pascal Tricoire

(Chairman & CEO Schneider Electric)

L'Edge si pone come soluzione per aiutare le organizzazioni a sfruttare una rapidissima crescita dei dispositivi connessi: entro il prossimo anno si avranno più di 25 miliardi di dispositivi connessi (*Cisco*)



13,5 miliardi di \$

è il potenziale disponibile per il mercato dell'Edge sino al 2025
(*BIS Research, 2018*)

Perchè l'Edge ?

“Nessuno vuole più dipendere solo dalle comunicazioni verso il Cloud per gestire le risorse più critiche”

Jean-Pascal Tricoire

(Chairman & CEO Schneider Electric)

L'Edge si pone come soluzione per aiutare le organizzazioni a sfruttare una rapidissima crescita dei dispositivi connessi: entro il prossimo anno si avranno **più di 25 miliardi di dispositivi connessi** (Cisco)



13,5 miliardi di \$

è il potenziale disponibile per il mercato dell'Edge sino al 2025
(BIS Research, 2018)

Perchè l'Edge ?

Il calcolo al livello di *edge* non solo può **ridurre l'onere dei costi di rete e di trasmissione dei dati**, ma anche **potenziare l'efficienza operativa** e migliorare la scalabilità delle organizzazioni **attraverso una basilare analisi dei dati in prossimità dei dispositivi che li originano** (*BIS Research, 2018*)

Riduzione della latenza ed ottimizzazione della banda. A livello edge è possibile analizzare e processare i dati lì dove vengono generati

Latenza e Banda disponibile

Riduzione dei costi di trasmissione evitando la trasmissione di tutti i dati grezzi al Cloud per processarli e dovendo attendere i risultati della loro analisi

Costi di trasmissione dati

Affrontare i problemi di riservatezza dei dati. A seconda della politica di riservatezza dell'azienda, i dati vengono archiviati su Cloud o Edge

Riservatezza dei dati

Collect & Connect

Il viaggio dei dati attraverso il livello Edge



1. COLLECT

Raccolta dei dati da sorgenti differenti (PLC, HMI, Enterprise IT software, Asset, Sensori...)



2. STORE & FORWARD

Storicizzazione dei dati *timestamped* a livello edge, allo scopo di creare il dataset da processare.

Forward dei dati in caso di disconnessione dalla rete



3. CONTEXTUALIZE

Contestualizzazione ed aggregazione dei dati nelle immediate vicinanze delle loro sorgenti



4. CLOUD CONNECT

Pubblicazione dei dati ottimizzati sul Cloud



Machine « Health App »

Anticipare le problematiche di una macchina monitorandone lo « stato di salute »



Machine « Health App »

Anticipare le problematiche di una macchina monitorandone lo « **stato di salute** »

Bisogni dell'utente



- Integrare facilmente gli **asset più critici** & monitorare lo **stato di ogni macchina** ovunque essa si trovi
- Ricevere **notifiche** ed informazioni dettagliate **quando la salute** della macchina **peggiora**

Soluzione Edge « Healt App »



- Acquisizione ed **aggregazione dei dati** della macchina (MIN, MAX, MEDIA, σ ...)
- Analisi del loro andamento con conseguente generazione di un **modello comportamentale** della macchina ed **attribuzione di un punteggio (%)** alla sua salute
- Rilevamento di **anomalie o peggioramenti** nel funzionamento ed invio di notifiche (mail, MQTT, telegram..)

Machine « Health App »

Quale approccio?

Trovare il
Pain Point

Definire il dataset del
Pain Point

Testare la macchina in
condizioni normali

Testare la macchina in
condizioni critiche

Scoprire il risultato
in Health App



Quali dati rappresentano il *dataset* ideale?



DATI NON IDEALI

- Dati non storicizzati
- Formati proprietari dei dati (e.s. no CSV export)
- Dati esclusivamente di stato/allarme derivanti da SCADA
- Dati raccolti con frequenza maggiore di 0.2 Hz
- Dati raccolti ogni n secondi senza informazione sullo stato
- Periodo non specificato
- Dati raccolti quando la macchina è spenta

- SCADA alarm data "Machine Error XXX" "OK" "WARN"
- Raw vibration / high frequency data
- Raw sensor values ("0x8F", "0b10110111")



DATI IDEALI

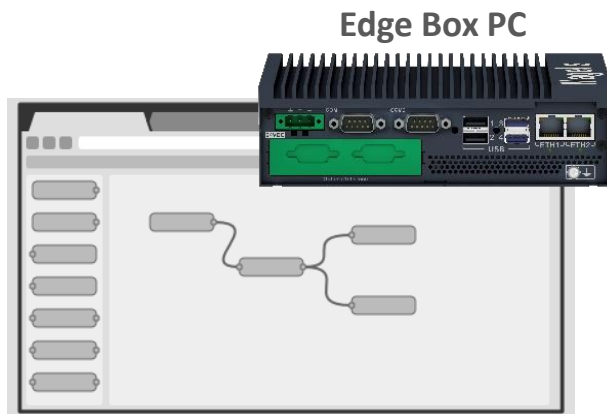
- Dati raccolti da un gateway IoT che possono essere inviati sul Cloud
- Dati raccolti durante uno stato macchina conosciuto (ad esempio uno stato misurabile, specifico e ripetibile)
- Dati aggregati (ad esempio ricavandone MIN, MEDIA, MAX e rate di cambiamento, su intervalli di 5/10 minuti)

- Current (A), Voltage (V),
- Torque (Nm), Speed (m/s)
- Pressure (Pa), Temperature (C)
- Vibration (m/s², mm/s . . .)

Machine « Health App »

Come implementare l'algoritmo? **Node-RED** è una delle scelte possibili

Node-RED è uno strumento di cablaggio IoT per connettere servizi e dispositivi attraverso un'interfaccia grafica intuitiva. Abbiamo scelto Node-RED come tecnologia per la raccolta e l'aggregazione dei dati e per l'implementazione dell'algoritmo di « Healt App »



Scalabilità

In Node-RED è semplice integrare prodotti connessi e servizi rispettando i requisiti di cybersecurity



Time to Market

Riduzione significativa dei tempi di integrazione per implementare la soluzione

Machine « Health App »

Implementazione dell'algoritmo, finalizzato a tracciare costantemente l'**Health** della macchina

Input:

fino a **8 parametri**

Input:

Istruire la macchina con i valori "corretti"

Health
App

Type: Analytics
v.1.0.0



Output:

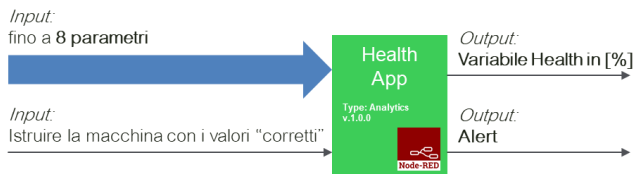
Variable Health in [%]

Output:

Alert

Machine « Health App »

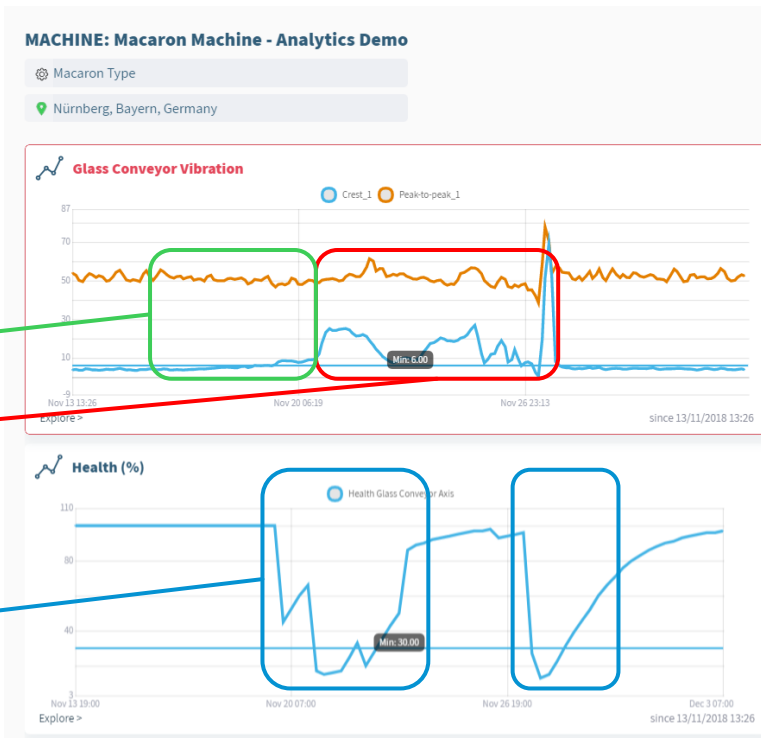
Implementazione dell'algoritmo, finalizzato a tracciare costantemente l'**Health** della macchina



Dati della macchina in condizioni normali

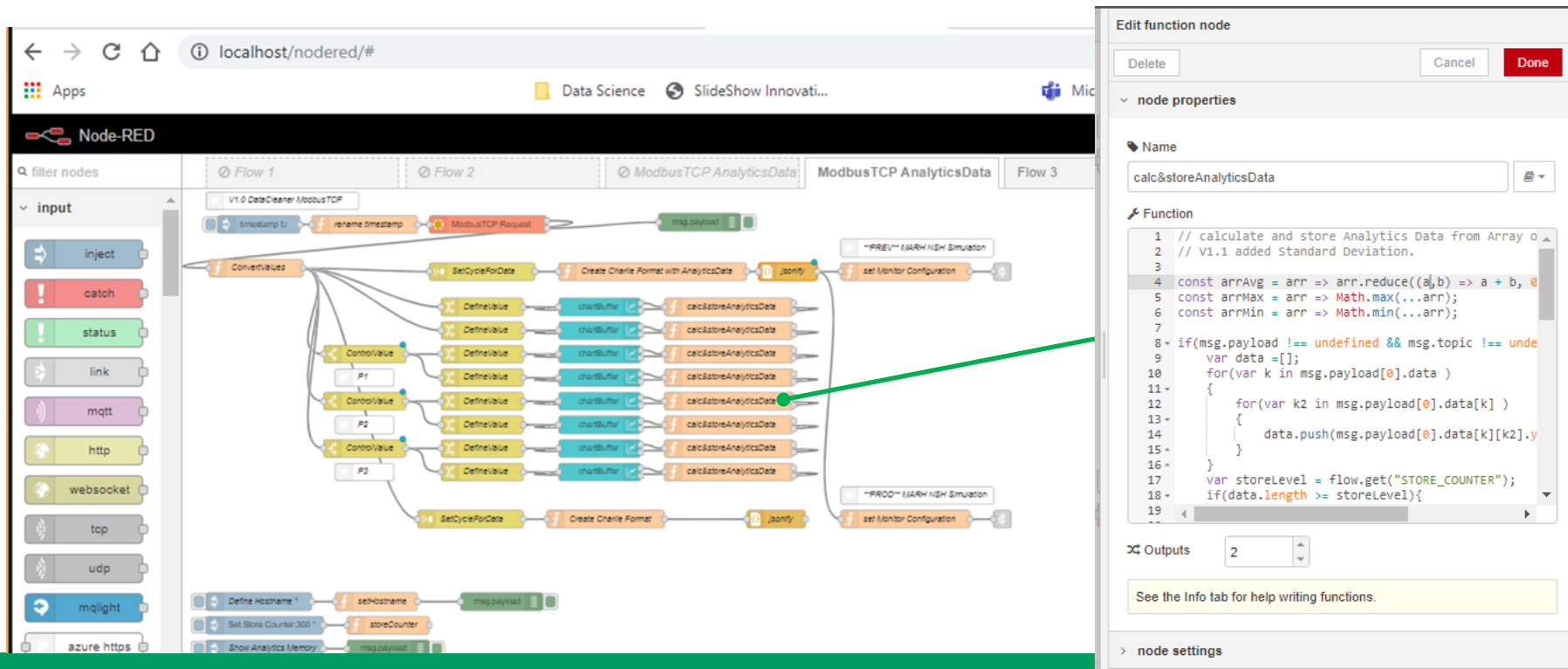
Dati della macchina in condizioni anomale

Parametro **Health** basso in corrispondenza di un'anomalia



Machine « Health App »

Flow in Node-RED e dettaglio funzione per l'aggregazione dei dati



The image displays the Node-RED web interface in a browser window at localhost/nodered/#. The main workspace shows a flow titled 'ModbusTCP AnalyticsData' with several nodes including 'V1.0 DataCleaner ModbusTCP', 'timestamp to', 'rename timestamp', 'ModbusTCP Request', 'msg.payload', 'ConvertValues', 'SetCycleForData', 'Create Charlie Format with AnalyticsData', 'jsonify', 'set Monitor Configuration', and 'PREVIEW IASRH NISH Simulation'. A green arrow points from the 'calc&storeAnalyticsData' function node in the flow to its detailed view in the 'Edit function node' panel on the right.

The 'Edit function node' panel shows the following code for the 'calc&storeAnalyticsData' function:

```

1 // calculate and store Analytics Data from Array 0
2 // V1.1 added Standard Deviation.
3
4 const arrAvg = arr => arr.reduce((a,b) => a + b, 0)
5 const arrMax = arr => Math.max(...arr);
6 const arrMin = arr => Math.min(...arr);
7
8 if(msg.payload !== undefined && msg.topic !== unde
9   var data = [];
10   for(var k in msg.payload[0].data )
11     {
12       for(var k2 in msg.payload[0].data[k] )
13         {
14           data.push(msg.payload[0].data[k][k2].y
15         }
16     }
17   var storeLevel = flow.get("STORE_COUNTER");
18   if(data.length >= storeLevel){
19

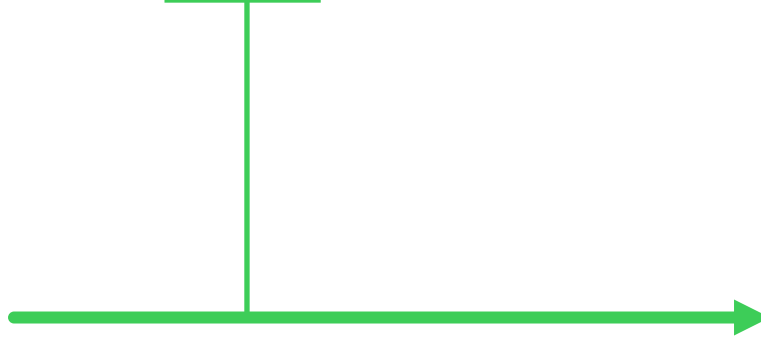
```

The 'node settings' section at the bottom of the panel shows 'Outputs' set to 2 and a message: 'See the Info tab for help writing functions.'

Machine « Health App »

Test su macchina reale (robot Delta P4): inizializzazione

- **Teaching Phase:** Selezionati 14 giorni di dati “sani” già nel database
- **Variables choice:** Dataset da monitorare [Asse A: **Vel** (Max,Avg,Min) | **FdbCur** (Max,Avg,StdDev,Min)]
- **Hardware setup:** Leggera modifica dell’aggancio di una molla



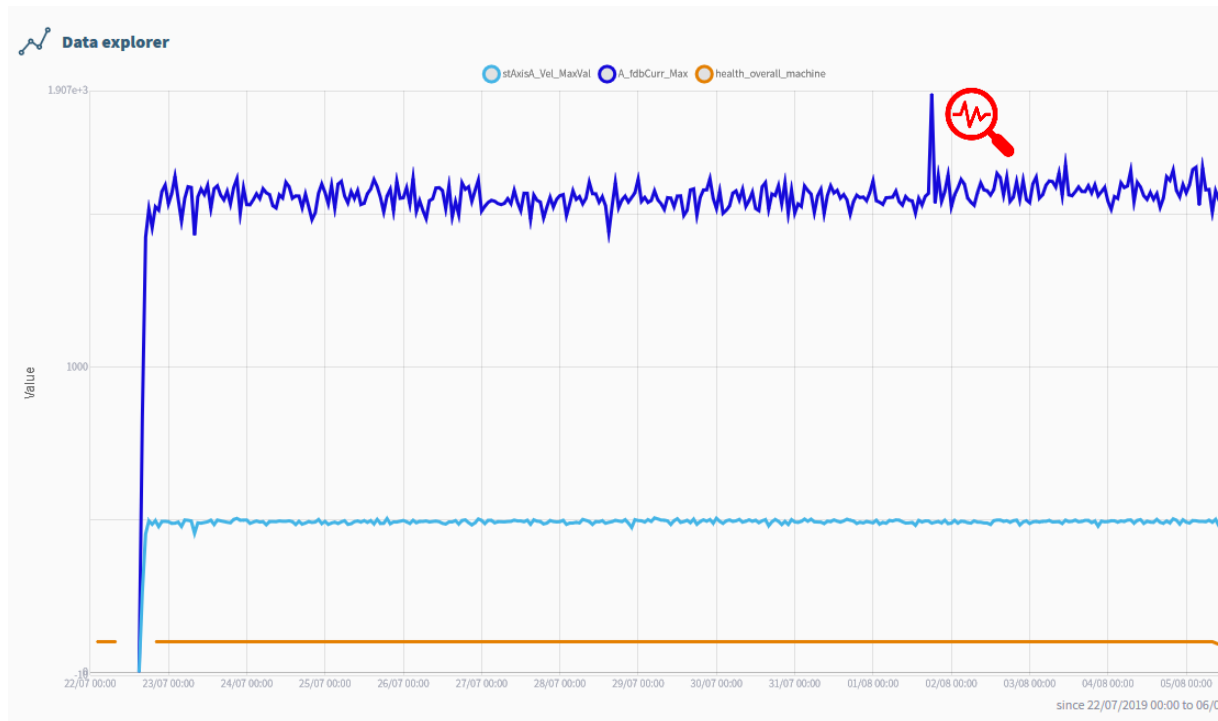
Machine « Health App »

Test su macchina reale (robot Delta P4): monitoraggio

- Robot continuamente attivo per i primi 16gg
- Dopo circa 10 giorni si verifica un picco da parte della *A_fdbCurr_Max* (5208)
- Fino al 15° giorno la variabile *health_overall_machine* ha valore 100%




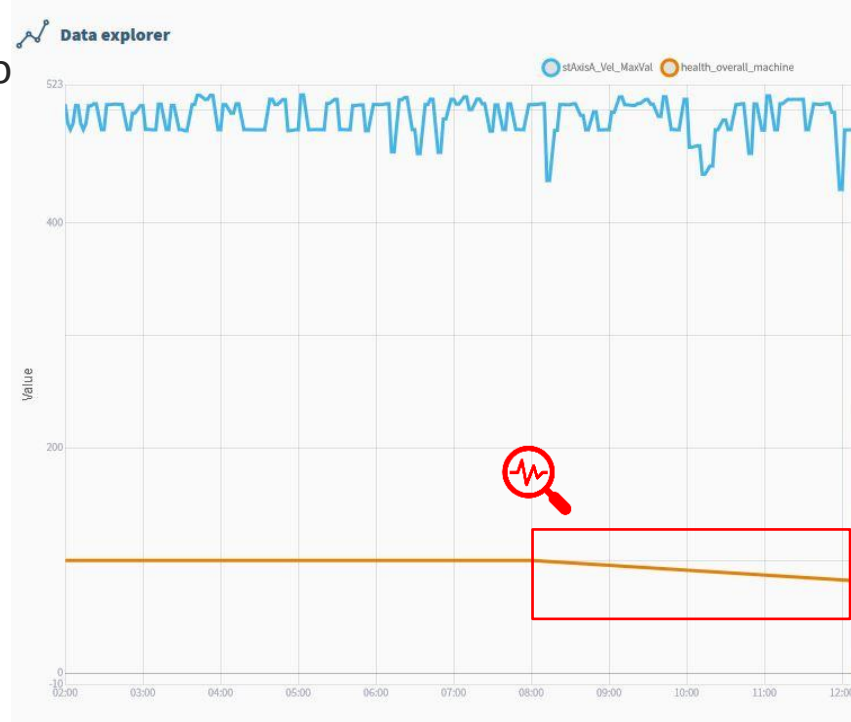
Stato salute macchina: **ottima**



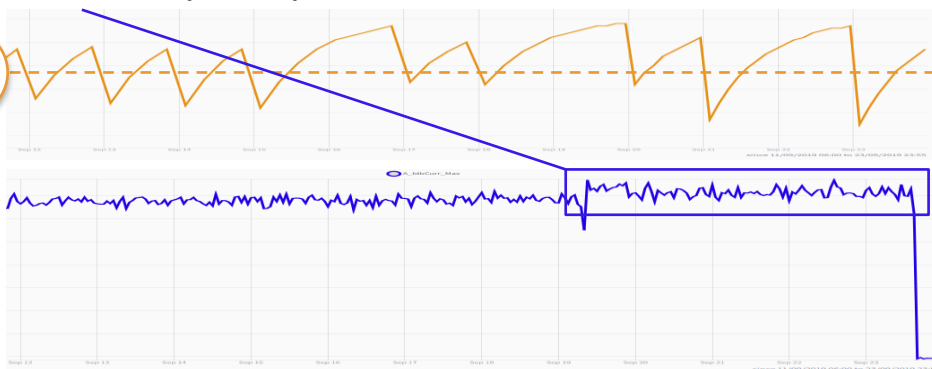
Machine « Health App »

Test su macchina reale (robot Delta P4): monitoraggio

- Il 16° giorno il valore di *health_overall_machine* inizia a calare sino a scendere, in sole 4h, al **75%** 
- e si assesta in **quest'intorno** per altri 14 giorni di funzionamento, durante i quali viene effettuata una manutenzione ordinaria (lubrificazione) e la *corrente* sposta più in alto il suo intervallo di lavoro:



75



Machine « Health App »

Test su macchina reale (robot Delta P4): esito finale

- Il 31° giorno il robot cede
- Errore: **Excessive Tracking Deviation / Limit X Exceeded**

Template: ExceptionList

List of all active Machine Exceptions

udlNumberOfActiveExceptions: 2

Extended 1

Extended 2

#	udlModuleId	limReactionTimeStamp	eDiag	eDiagExt	sMsg
1	1010101	TOD#15:29:51.510	12	ExcessiveTrackingDeviationLimitExceeded	Excessive tracking deviation limit X exceeded.
2	1010101	TOD#15:29:51.520	2	DriveWarning	DiagCode: 8132 Msg: Tracking deviation limit exceeded Ext.
3	0	TOD#00:00:00	0	Ok	
4	0	TOD#00:00:00	0	Ok	
5	0	TOD#00:00:00	0	Ok	
6	0	TOD#00:00:00	0	Ok	
7	0	TOD#00:00:00	0	Ok	
8	0	TOD#00:00:00	0	Ok	
9	0	TOD#00:00:00	0	Ok	
10	0	TOD#00:00:00	0	Ok	

Active Reactions:

24:

<

1

2

3

4

5

6

7

8

9

10

>

For test only!

Create an exception with reaction:

AsyncStop

SyncStopEL

StopEndOfCycle

MainsContactorOff

SyncStopEH

Diag Out

Control Panel

Machine Control

Modules

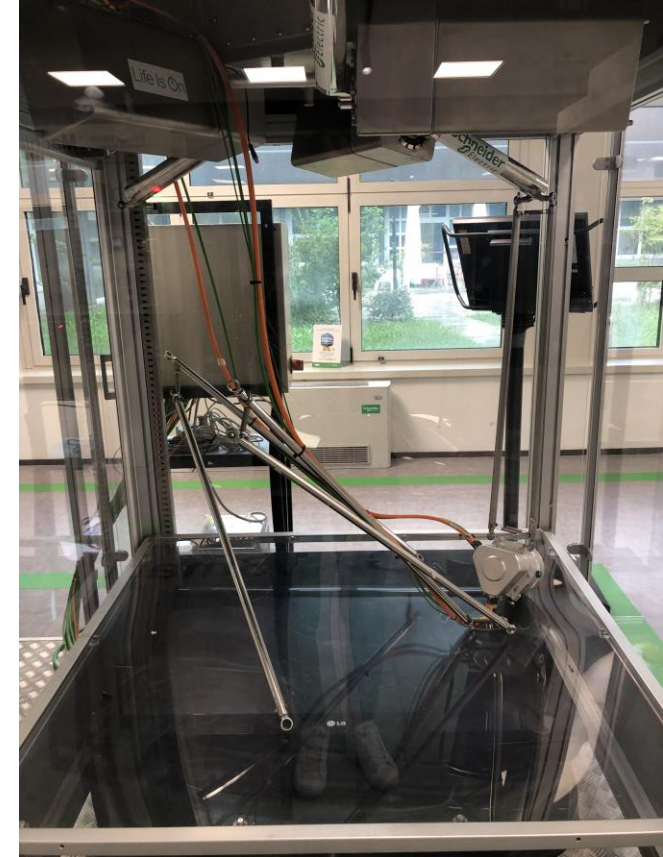
Robotic Module

Exception List

LogData List

Vis Navigation

Enable Vis



Machine « Health App »

Test su macchina reale (robot Delta P4): osservazioni finali

- L'implementazione a livello Edge di un efficiente algoritmo di aggregazione ed analisi dei dati di macchina (Health App) ha permesso di **tracciare e rilevare** in pochissimo tempo **un peggioramento (seppur lieve) della « salute »** della macchina stessa
- Ignorando quindi i segnali che **hanno continuato ad indicare un livello di salute più basso del normale** (dovuto ad un comportamento non corretto rispetto al modello comportamentale di riferimento) la macchina si è effettivamente danneggiata...



...l'evento di rottura è certamente correlato al livello di salute della macchina (mai ritornato alla normalità)



*Quale può essere il miglior modo per
acquisire dati da un'architettura esistente
(brownfield) ?*

Secondary sensing

...ovvero l'aggiunta (ad un'architettura esistente) di una rete di **rilevamento secondaria**

Aggiunta di un II livello di sensoristica **non intrusiva**

Energy Monitoring

Performance Monitoring

Machine Status Monitoring

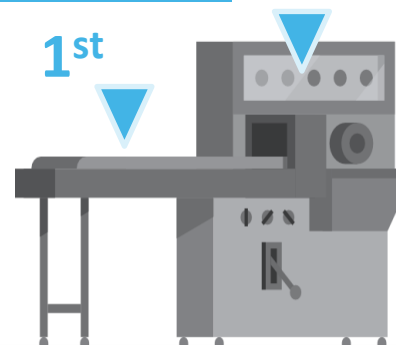
Macchina dotata dei propri sensori ed attuatori, cablati

1st

Comunicazione wireless a corto raggio (applicazioni *indoor*)

2nd

Livello di *Secondary sensing* per acquisire le informazioni necessarie



Secondary sensing

Semplice monitoraggio della propria applicazione attraverso **prodotti connessi *self-powered***

Finecorsa wireless (self-powered)



Uso: Posizione di un oggetto
Posizione di un carter
Conteggio prodotti

Benefici:

- Installazione semplificata

Pulsanti wireless (self-powered)



Uso: Call to action
Location

Benefici:

- Riduzione costi di installazione
- Plug & Play

Misuratori di Energia (self-powered)



Uso: Monitoraggio di energia e
potenza

Benefici:

- Monitoraggio degli asset
- (1) Installazione immediata in quadri pre-esistenti
- (2) Nessun cablaggio necessario

Sensori di Temperatura, umidità, vibrazioni (self-powered)



Uso: Monitoraggio di parametri
ambientali e meccanici

Benefici:

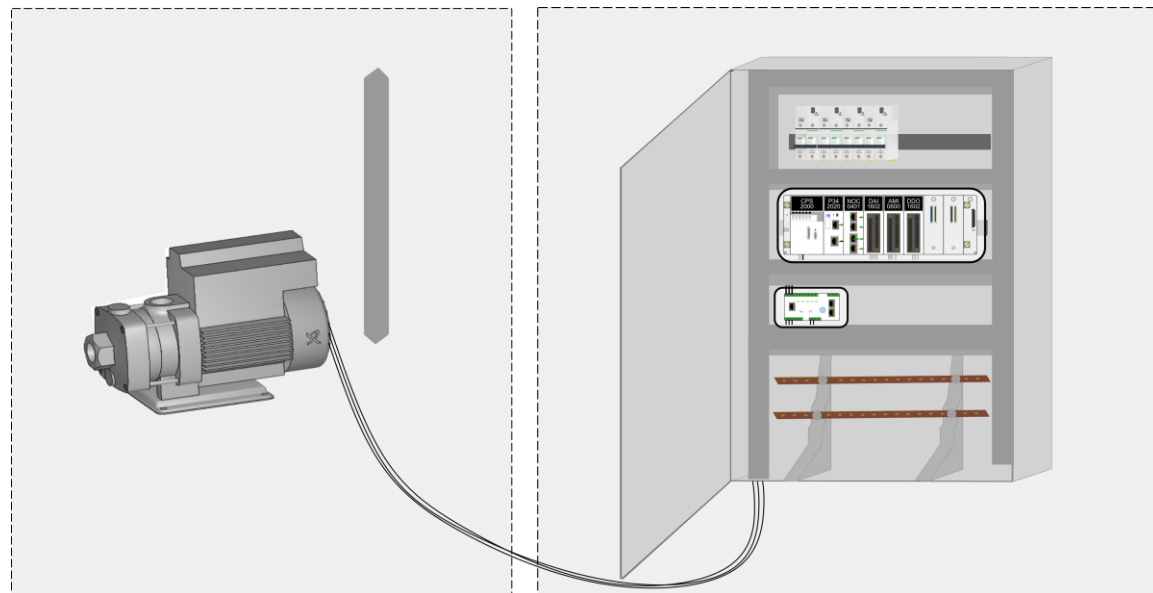
- Installazione semplificata
- Ready-to-use

Secondary sensing

Esempio di architettura

Prodotti utilizzati

- **Prodotti connessi** (eg. ZigBee)
 - Sensori di Temperatura, Umidità, Vibrazioni
 - Energy meter
 - Pulsanti e finecorsa
- **Edge**
 - PLC
 - Gateway segnali (eg. ZigBee → ModBus)
 - Edge Box PC (eg. NodeRED)
- **Cloud**
 - App, Analytics e Servizi Cloud

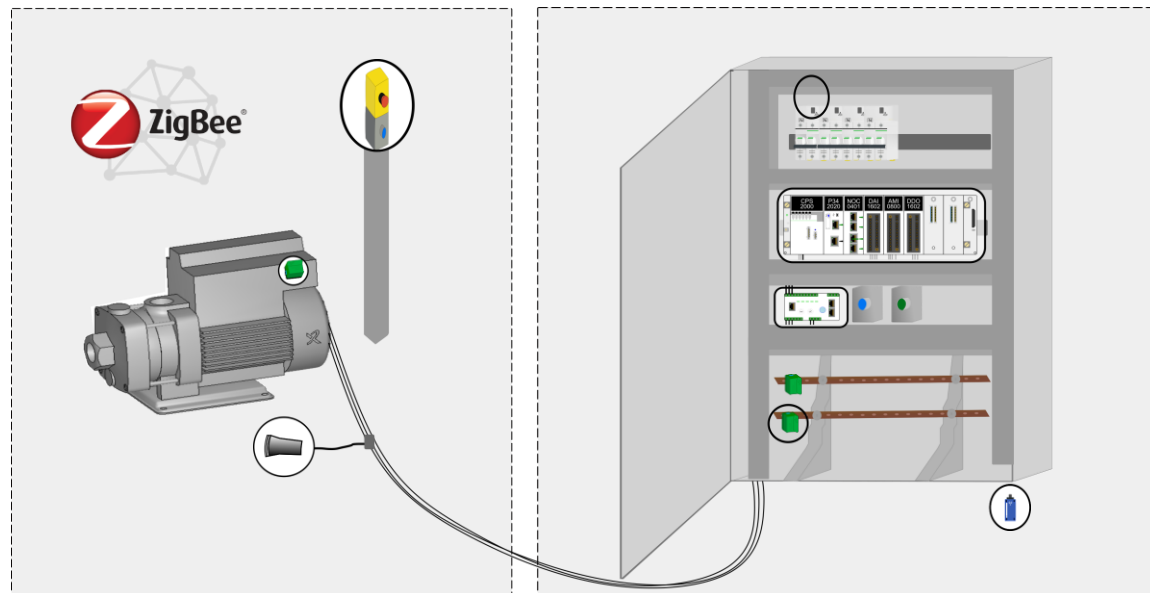


Secondary sensing

Esempio di architettura

Prodotti utilizzati

- **Prodotti connessi (eg. ZigBee)**
 - Sensori di Temperatura, Umidità, Vibrazioni
 - Energy meter
 - Pulsanti e finecorsa
- **Edge**
 - PLC
 - Gateway segnali (eg. ZigBee → ModBus)
 - Edge Box PC (eg. NodeRED)
- **Cloud**
 - App, Analytics e Servizi Cloud

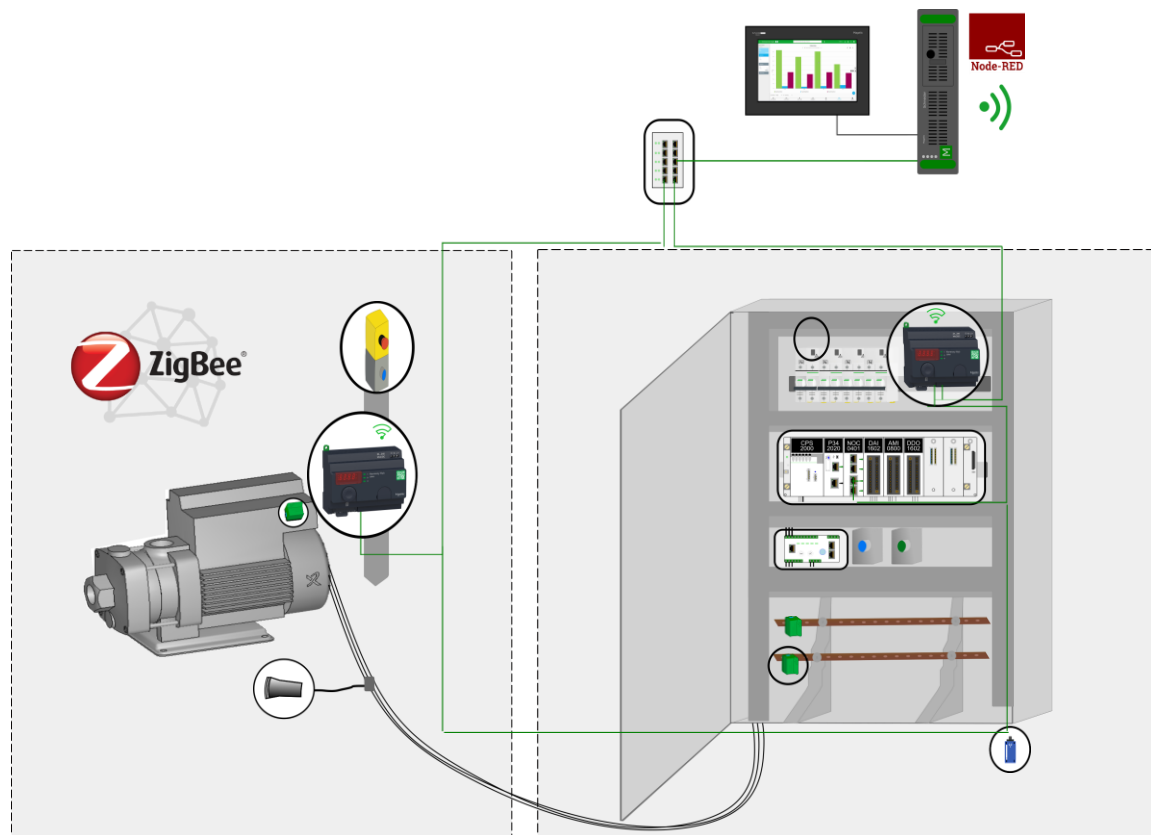


Secondary sensing

Esempio di architettura

Prodotti utilizzati

- **Prodotti connessi (eg. ZigBee)**
 - Sensori di Temperatura, Umidità, Vibrazioni
 - Energy meter
 - Pulsanti e finecorsa
- **Edge**
 - PLC
 - **Gateway segnali** (eg. ZigBee → ModBus)
 - **Edge Box PC** (eg. NodeRED)
- **Cloud**
 - App, Analytics e Servizi Cloud

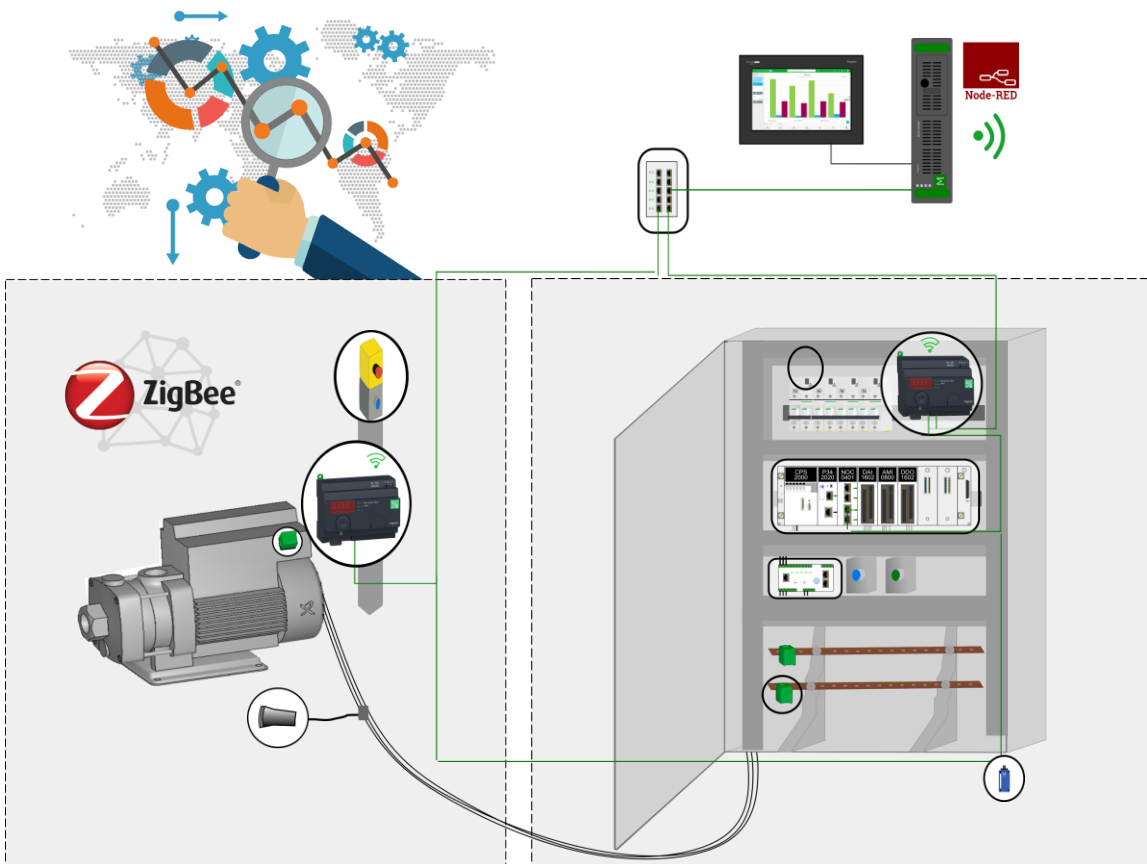


Secondary sensing

Esempio di architettura

Prodotti utilizzati

- **Prodotti connessi** (eg. ZigBee)
 - Sensori di Temperatura, Umidità, Vibrazioni
 - Energy meter
 - Pulsanti e finecorsa
- **Edge**
 - PLC
 - Gateway segnali (eg. ZigBee → ModBus)
 - Edge Box PC (eg. NodeRED)
- **Cloud**
 - App, Analytics e Servizi Cloud





GRAZIE