

---

# Utilizzo di OSGi per lo sviluppo di Digital Twin



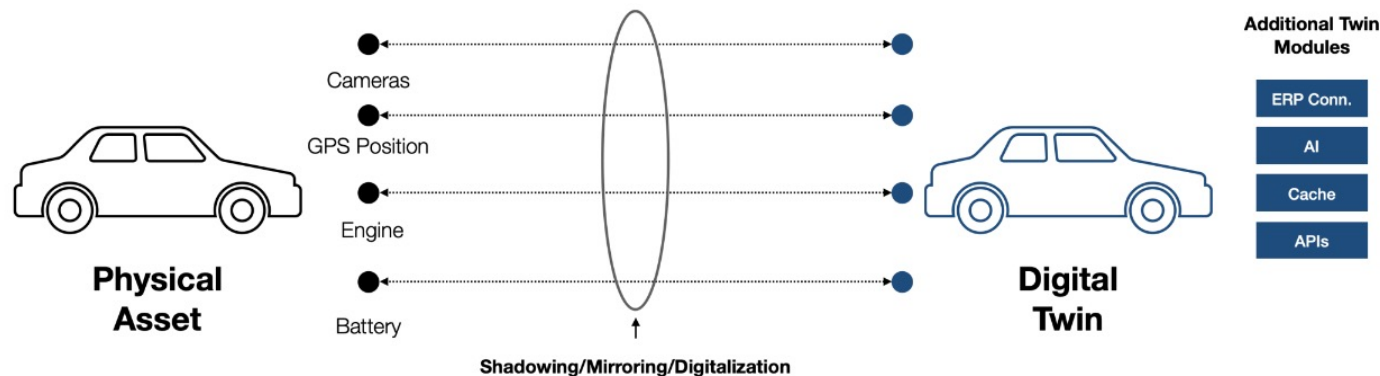
Luciano Imbimbo

RELATORI: PROF. NICOLA BIOCCHI, PROF. MARCO PICONE

ANNO ACCADEMICO 2022/2023

# Digital Twin

Rappresentazione software completa di un oggetto fisico che include le proprietà, le condizioni e i comportamenti dell'oggetto reale attraverso l'utilizzo di modelli e dati.  
Rappresenta e riflette la controparte fisica durante l'intero ciclo di vita dell'oggetto.



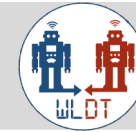
# Digital Twin

Le piattaforme e le soluzioni DT esistenti rappresentano un contributo straordinario verso la sperimentazione e l'adozione diffusa della tecnologia ma possiamo definirle solo come un punto di partenza, a causa di una serie di problematiche:

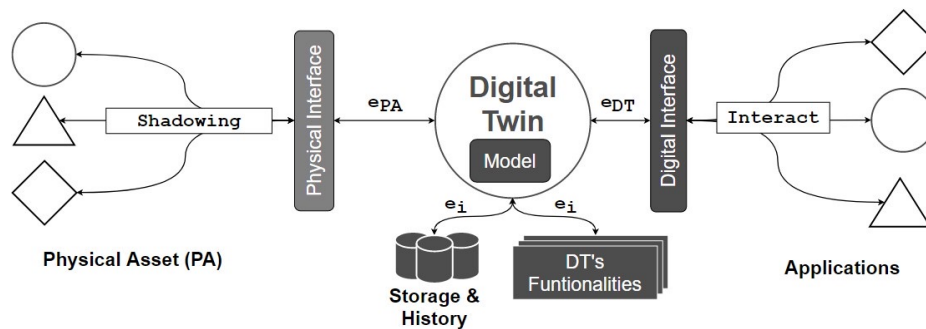
- Approcci centralizzati/monolitici in cui tutti i DT sono aggregati e distribuiti nello stesso punto (Cloud)
- I DT sono spesso "semplici" strutture dati che possono essere utilizzate per rappresentare uno scenario applicativo senza l'uso di modelli e di elementi attivi (oggetto software passivo)
- Tecnologie proprietarie costruite attorno ad un punto di controllo centrale



# White Label Digital Twin Framework

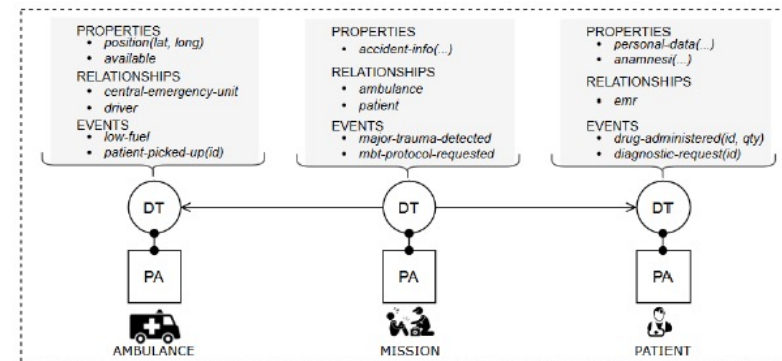


WLDT, sviluppato da Marco Picone (ricercatore e professore associato di uniMORE), intende massimizzare la modularità, la riutilizzabilità e la flessibilità per rispecchiare efficacemente oggetti fisici intelligenti nelle loro controparti digitali.

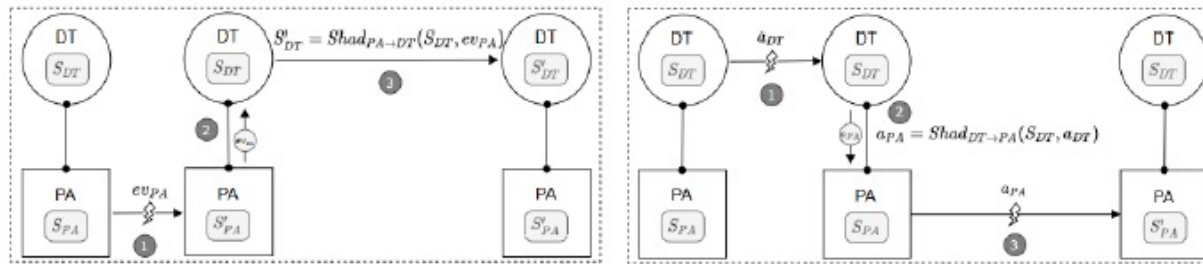
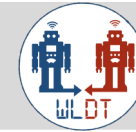


Digital Twin come composizione di moduli

Digital Twin State

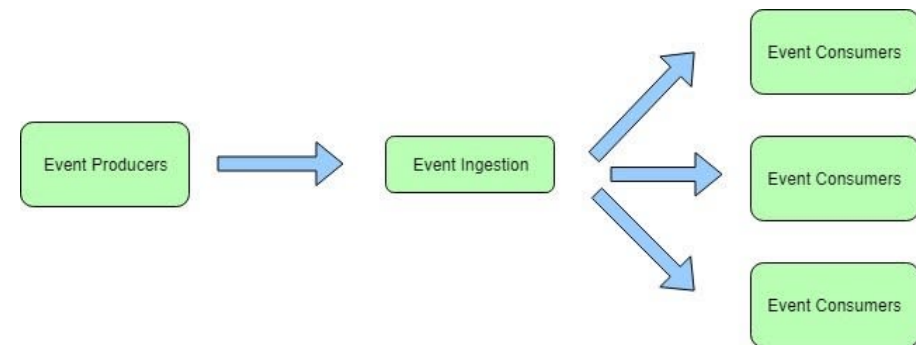


# White Label Digital Twin Framework



Processo di shadowing

Event-Driven Architecture



**Problema:** WLD T non integra al suo interno funzionalità per la gestione dinamica e runtime dei vari moduli del DT. Come risolvere?

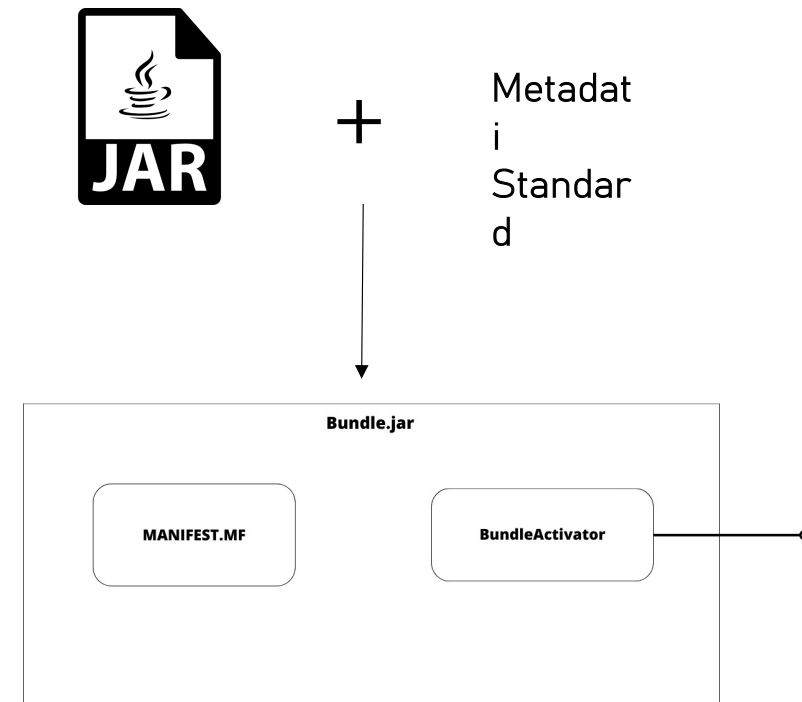
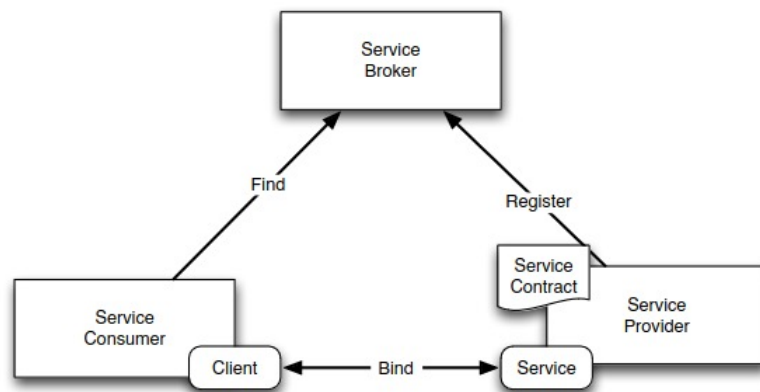


**UNIMORE**  
UNIVERSITÀ DEGLI STUDI DI  
MODENA E REGGIO EMILIA

# Soluzione: OSGi

Framework Java che consente agli sviluppatori di creare applicazioni altamente modulari, utilizzando i seguenti concetti:

1. **Bundle**: più piccola unità di distribuzione in OSGi.
2. **Servizi**: il mezzo principale di comunicazione e collaborazione tra i bundle in OSGi.
3. **OSGi Framework**: ambiente runtime che gestisce il ciclo di vita e le interazioni tra i bundle. Fornisce meccanismi per l'installazione, l'avvio, l'arresto, l'aggiornamento e la disinstallazione dei bundle.



# Apache Felix



Implementazione della piattaforma OSGi Framework and Service sotto licenza Apache

Maven Build Plugin



Gogo Shell e File install

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tutti i diritti riservati.

Installa la versione più recente di PowerShell per nuove funzionalità e miglioramenti. https://aka.ms/PSWindows

PS C:\Users\lucia> cd ..
PS C:\Users> cd ..
PS C:\> cd .\Felix\
PS C:\Felix> java -jar .\felix-framework-7.0.5\
Error: Unable to access jarfile .\felix-framework-7.0.5\
PS C:\Felix> java -jar .\felix-framework-7.0.5\bin\felix.jar
PS C:\Felix> cd .\felix-framework-7.0.5\
PS C:\Felix\felix-framework-7.0.5> java -jar .\bin\felix.jar
Starting to listen
Exl: Service of type org.osgi.service.event.EventAdmin registered.
Exl: Service of type java.lang.Object registered.

Welcome to Apache Felix Gogo

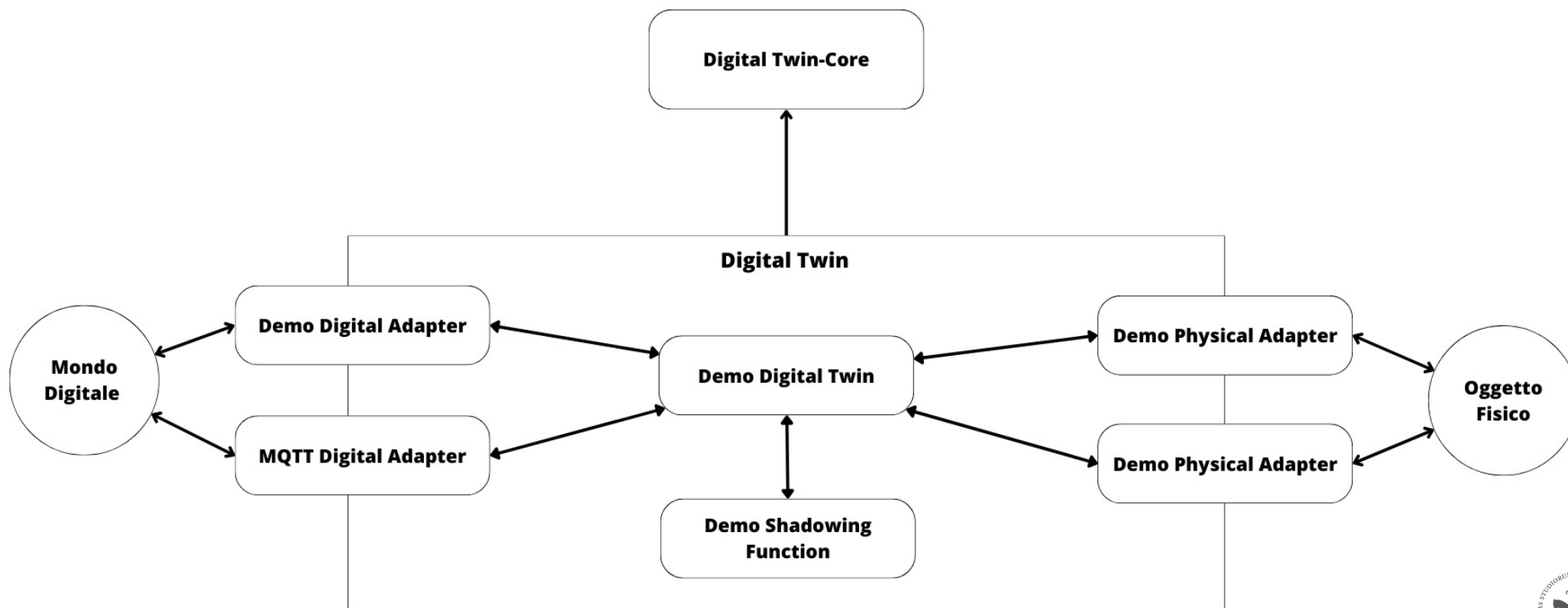
g! lb
START LEVEL 1
ID|State|Level|Name
0|Active|0|System Bundle (7.0.5)|7.0.5
1|Active|1|Jansi (1.18.0)|1.18.0
2|Active|1|JLine Bundle (3.13.2)|3.13.2
3|Active|1|Apache Felix Bundle Repository (2.0.10)|2.0.10
4|Active|1|Apache Felix Gogo Command (1.1.2)|1.1.2
5|Active|1|Apache Felix Gogo JLine Shell (1.1.8)|1.1.8
6|Active|1|Apache Felix Gogo Runtime (1.1.4)|1.1.4
26|Active|1|Client (1.0.0.SNAPSHOT)|1.0.0.SNAPSHOT
57|Active|1|Apache Felix EventAdmin (1.6.4)|1.6.4
68|Active|1|DigitalTwinCore (2.0.0)|2.0.0
69|Installed|1|demoDigitalTwin (2.0.1)|2.0.1
70|Installed|1|demoPhysicalAdapterTemperatura (2.0.0)|2.0.0
71|Installed|1|demoPhysicalAdapterUmidità (2.0.0)|2.0.0
72|Installed|1|demoDigitalAdapter (2.0.0)|2.0.0
77|Active|1|org.eclipse.paho.client.mqttv3 (1.2.5)|1.2.5
78|Installed|1|demoDigitalAdapterMQTT (2.1.1)|2.1.1
79|Installed|1|demoShadowingFunction (2.0.0)|2.0.0

g! _
```



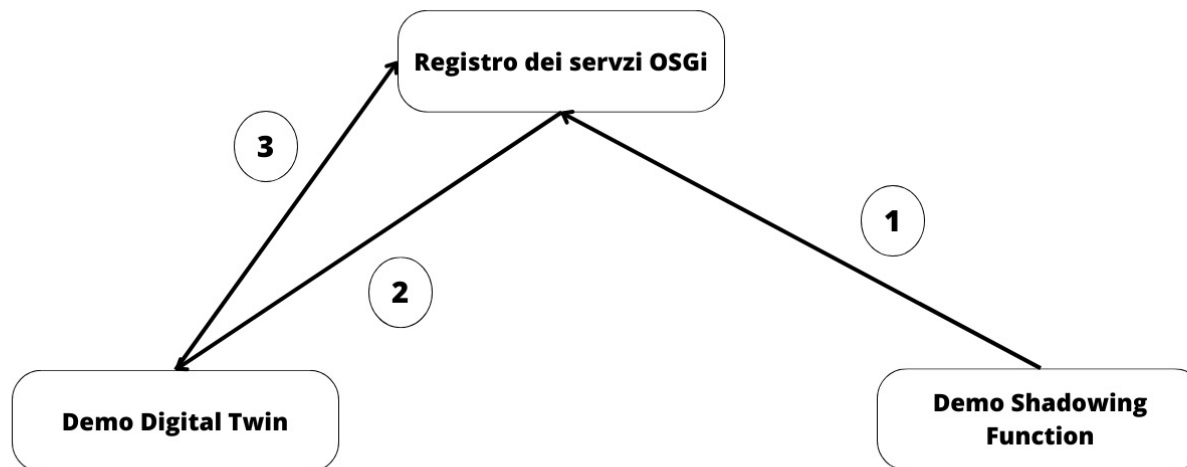
**UNIMORE**  
UNIVERSITÀ DEGLI STUDI DI  
MODENA E REGGIO EMILIA

# Il progetto



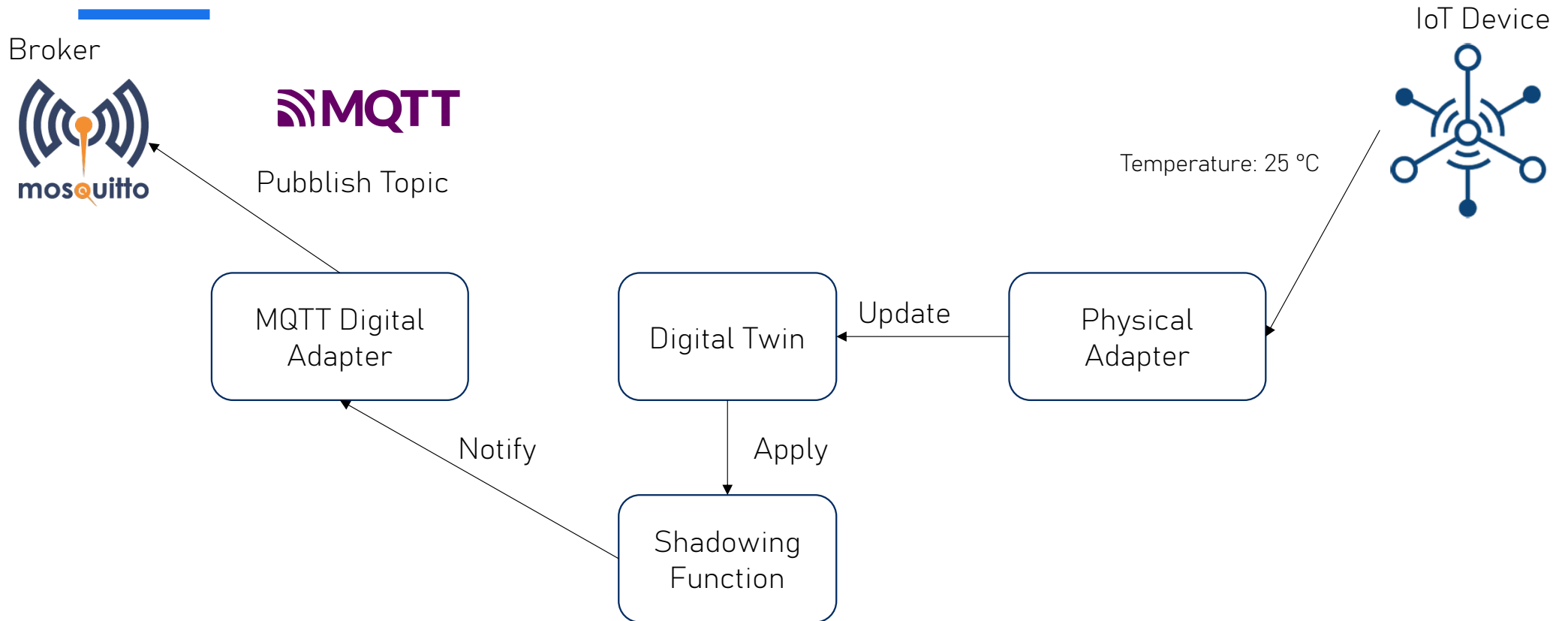


# Digital Twin & Shadowing Function



1. Il bundle della Shadowing Function si rende visibile registrando un servizio nel registro del framework
2. Il DT riceve una notifica e inizializza il suo stato interno
3. Il DT si rende visibile agli altri component (interfacce fisiche e digitali)

# Interfacce fisiche e Digitali



# Funzionamento

```
PowerShell X +
PS C:\> cd .\Felix\felix-framework-7.0.5\
PS C:\Felix\felix-framework-7.0.5> java -jar .\bin\felix.jar
Starting to listen
Ex1: Service of type org.osgi.service.event.EventAdmin registered.
Ex1: Service of type java.lang.Object registered.

Welcome to Apache Felix Gogo

g! _
```



Grazie per l'attenzione.

Luciano Imbimbo

RELATORI: PROF. NICOLA BIOCCHI, PROF. MARCO PICONE

ANNO ACCADEMICO 2022/2023