



HOME INTERNET OF THINGS (HIoT) – UN'ARCHITETTURA IOT DOMESTICA BASATA SU RETE LoRa: SERVIZI SOFTWARE

Relatori:

Chiar.mo Giuseppe Visaggio

Prof. Danilo Caivano

Laureanda:

Viviana Cianciaruso

Scenario

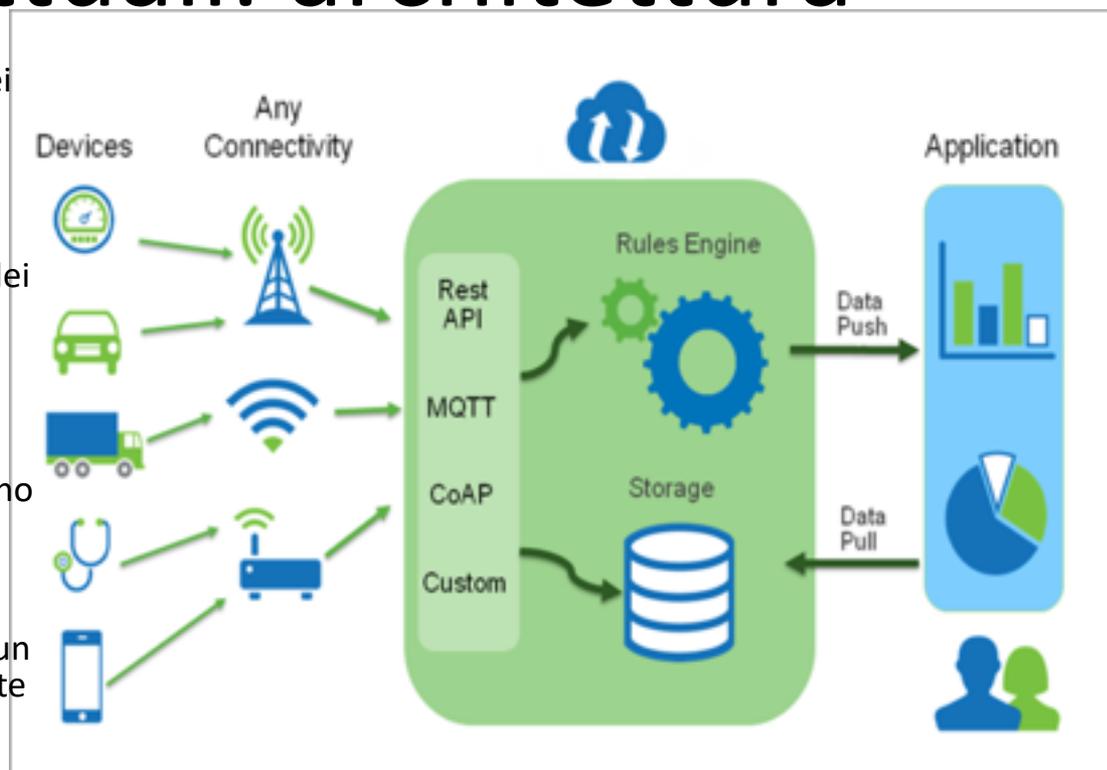
- Al giorno d'oggi esistono circa 6.4 miliardi di oggetti che sono in grado di comunicare dati appartenenti alla vita reale; si stima che nel 2020 saranno diventati più di 20 miliardi.
- Questa tecnologia chiamata Internet of Things (IoT) si è sviluppata come tipicamente orientata all'impresa, trascurando gli enormi vantaggi che si potrebbero trarre dalla stessa tecnologia utilizzata anche in ambito domestico, con architetture semplificate per tale scopo.

Scopo

- Definizione di un'architettura semplificata per lo sviluppo dell'IoT domestico
- Realizzazione di un sistema sperimentale per tale ambito di sviluppo
- Sperimentazione della soluzione proposta

Modelli attuali: architettura

- **Sensori:** essi sono in grado di acquisire dei valori dagli oggetti di uso quotidiano, e spedirli ad un nodo centrale.
- **Nodi:** essi raccolgono i dati ottenuti dai sensori e li impacchettano secondo uno dei vari protocolli realizzati per una comunicazione efficace, tenendo conto delle scarse capacità di questi dispositivi come ad esempio una memoria grande pochi KB ed una capacità di elaborazione ridotta. Una volta impacchettati, i dati sono pronti per essere mandati ad un gateway.
- **Gateway:** esso ha il compito di ricevere correttamente i dati dal nodo attraverso un protocollo comune, e spedirli direttamente al di fuori della rete locale, attraverso Internet, verso un middleware.
- **Middleware:** Questo è uno dei più importanti layer dell'architettura IoT poiché si occupa di filtrare i dati ricevuti dai sensori, elaborarli ed inoltrare solo i dati rilevanti alle applicazioni, permettendo loro di intraprendere decisioni una volta elaborati questi dati. Spesso offre un servizio cloud.



- **Applicazione:** si occupa di reperire i dati dal middleware e renderli visualizzabili all'utente in maniera semplice ed intuitiva, visualizzando ad esempio dei grafici con l'andamento dei dati registrati.

Modelli attuali: protocolli

- L' IoT attualmente si serve di alcuni protocolli standard M2M per la comunicazione efficiente dei dati, e di middleware cloud per la gestione e visualizzazione di tali dati.
- MQTT è il protocollo di tipo publish/subscribe più utilizzato per la messaggistica dei sensori, grazie al suo basso impatto energetico e banda limitata.
- UDP è il protocollo di trasporto più usato per la sua rapidità e la sua particolare efficienza per le applicazioni leggere, supportando più client attivi.

Svantaggi



Costi traffico dati: la connessione avviene spesso via Wi-Fi o rete cellulare.

Poco scalabile: per ogni nuovo tipo di sensore aggiunto alla piattaforma è necessario portarlo ad un formato standard.

Privacy assente: i dati contenuti nei server cloud non sempre sono un vantaggio in quanto possono essere venduti per ricerche di mercato o altro.

Analisi comparativa

- Sono state esaminate diverse piattaforme per lo sviluppo IoT, valutate in base al tipo di connessione alla rete, i protocolli riconosciuti per ricevere correttamente i dati dai sensori, i protocolli utilizzati per connettersi a livello applicativo, il tipo di criptaggio messo a disposizione per la tutela della privacy, i costi di installazione.

Analisi comparativa

PIATTAFORME	TIPO CONNESSIONE HW	PROTOCOLLO HW	PROTOCOLLO SW	SICUREZZA	COSTI
TheThings Network	Antenna	Mqtt/tr50	http	Criptaggio	Open source
DeviceWISE	Sim proprietaria, antenna	Mqtt/tr50	http	Criptaggio, SSL e TSL	Open source
Hologram	Sim proprietaria	Mqtt/tr50	http	Criptaggio, PPP	A pagamento
TempoIQ		Hardware agnostic	http		A pagamento
Temboo	Sim proprietaria non	Mqtt,coap	http,rest		A pagamento
TheThingsIO	Sim proprietaria non	Mqtt, coap, udp, tcp	http, coap	SSL	A pagamento
Thingers	Sim proprietaria non	Hardware agnostic	http,rest	SSL e TSL	Open source
Sentilo	Sim proprietaria non	mqtt	http,rest		Open source

DeviceWISE è risultata la piattaforma vincente tra quelle attuali in quanto open source, con diversi livelli di criptaggio, utilizza protocolli di comunicazione standard, ma soprattutto ha duplice capacità di connessione.

Modelli attuali:



- La particolarità di questo middleware, oltre le varie funzionalità e opzioni che mette a disposizione, risiede nell'architettura particolarmente flessibile.
- Esso si compone di tre parti principali:
 - Il DeviceWISE M2M Service;
 - Il DeviceWISE Asset Gateway;
 - Il DeviceWISE Enterprise Gateway.

M2M Service



- si occupa di gestire la connessione ai dispositivi attraverso l'Asset Gateway o, alternativamente, con interfacce MQTT e HTTP(S). Inoltre esso dispone di un data warehouse built-in per storicizzare i dati ottenuti dai sensori, e di una integrazione nativa verso i sistemi di impresa come IBM, Oracle e SAP.

Asset Gateway



- è un software che comunica con i vari dispositivi, processa i loro dati e manda i dati processati al M2M Service, oppure ad un'altra applicazione d'impresa attraverso l'Enterprise Gateway.

Enterprise Gateway



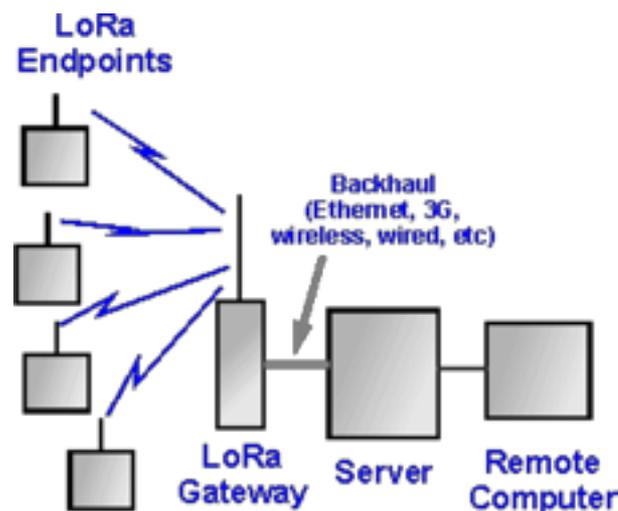
- è un software che può essere installato direttamente nel back-office aziendale e permette di mappare i dati direttamente dalla piattaforma DeviceWISE ad un'applicazione d'impresa e viceversa.

Soluzione proposta: LoRa™

- tecnologia wireless proprietaria sviluppata dalla Semtech, basata su radiofrequenze
- Lungo raggio e basso consumo di batteria
 - Più di 5km di raggio in aree urbane, più di 15km in aree suburbane
 - Più di 10 anni di vita di batteria
- Comunicazione robusta: nessuna interferenza con Wi-Fi, Bluetooth, GSM, LTE etc.
- Grande capacità di rete: supporta fino a 300k nodi

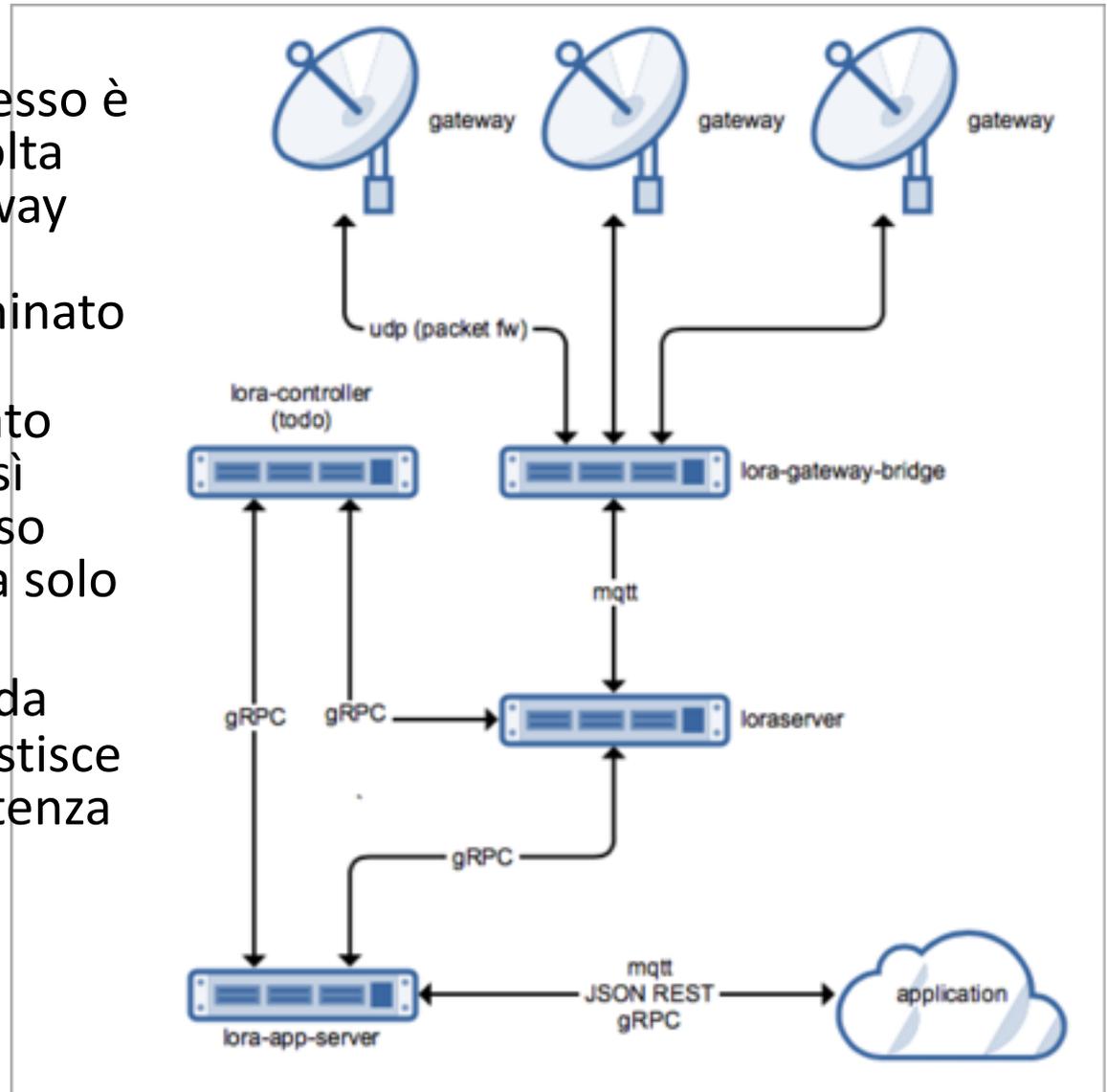
Architettura hardware

- End-Point: essi sono gli elementi della rete LoRa che effettuano il rilevamento o il controllo, e sono normalmente situati in remoto.
- LoRa Gateway: Questa parte della rete LoRa può essere Ethernet, cellulare o qualsiasi altro collegamento di telecomunicazione cablata o wireless. I gateway sono collegati al server di rete utilizzando connessioni IP standard. In questo modo i dati utilizzano un protocollo standard, ma può essere collegato a qualsiasi rete di telecomunicazioni pubblica o privata.
- Network server: esso agisce per eliminare i pacchetti duplicati, gli orari di riconoscimento, e si adatta velocità di trasmissione dati. Grazie al modo semplice in cui esso può essere distribuito e collegato, rende molto semplice implementare una rete LoRa.
- Computer remoto: un computer remoto può controllare le azioni degli endpoint o raccogliere dati da loro.



Architettura software

- LoRa gateway bridge: esso è un servizio che, una volta ricevuti i dati dai gateway LoRa in un particolare protocollo UDP denominato packet forwarder, li standardizza nel formato MQTT, e li spedisce così standardizzati attraverso Internet. Esso funziona solo con Linux.
- LoRa server: funziona da network server che gestisce i dati in arrivo o in partenza da o verso i gateway.





- Grazie all'architettura flessibile di entrambe le componenti, combinare le due tecnologie porta ad enormi vantaggi:
 - Nessun costo di connessione
 - Privacy migliorata grazie all'uso di radiofrequenze
 - Conversione dei dati dei sensori in formato standard MQTT
 - Unica antenna per connettere migliaia di dispositivi
 - Lungo raggio di azione entro circa 15km
 - Nessun costo di installazione
 - Privacy migliorata attraverso vari sistemi di criptaggio
 - Possibilità di migrazione dati su database locali
 - Compatibilità con tutti i dispositivi MQTT
 - App dedicata
 - Visualizzazione personalizzata dei dati

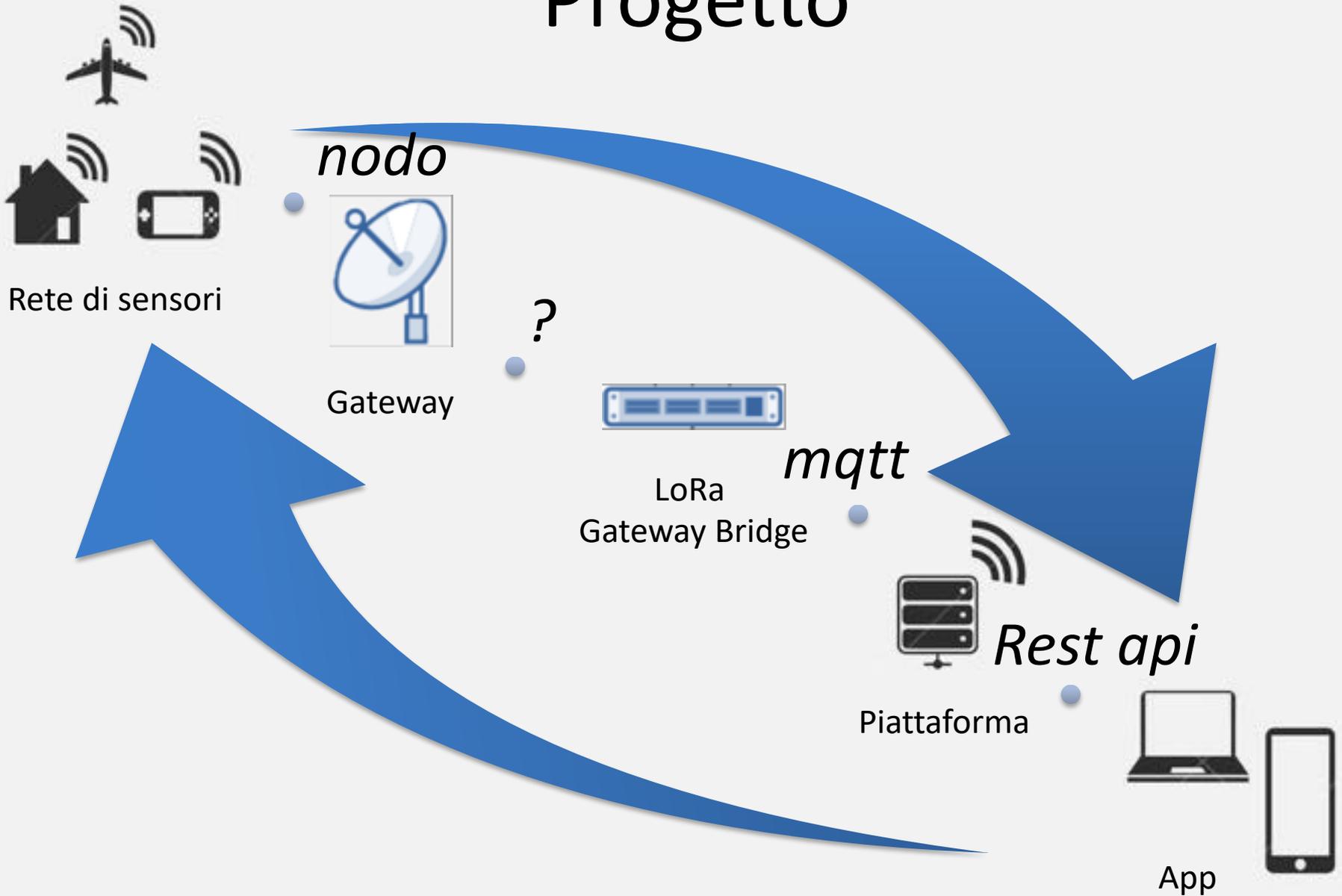
Scenari applicativi



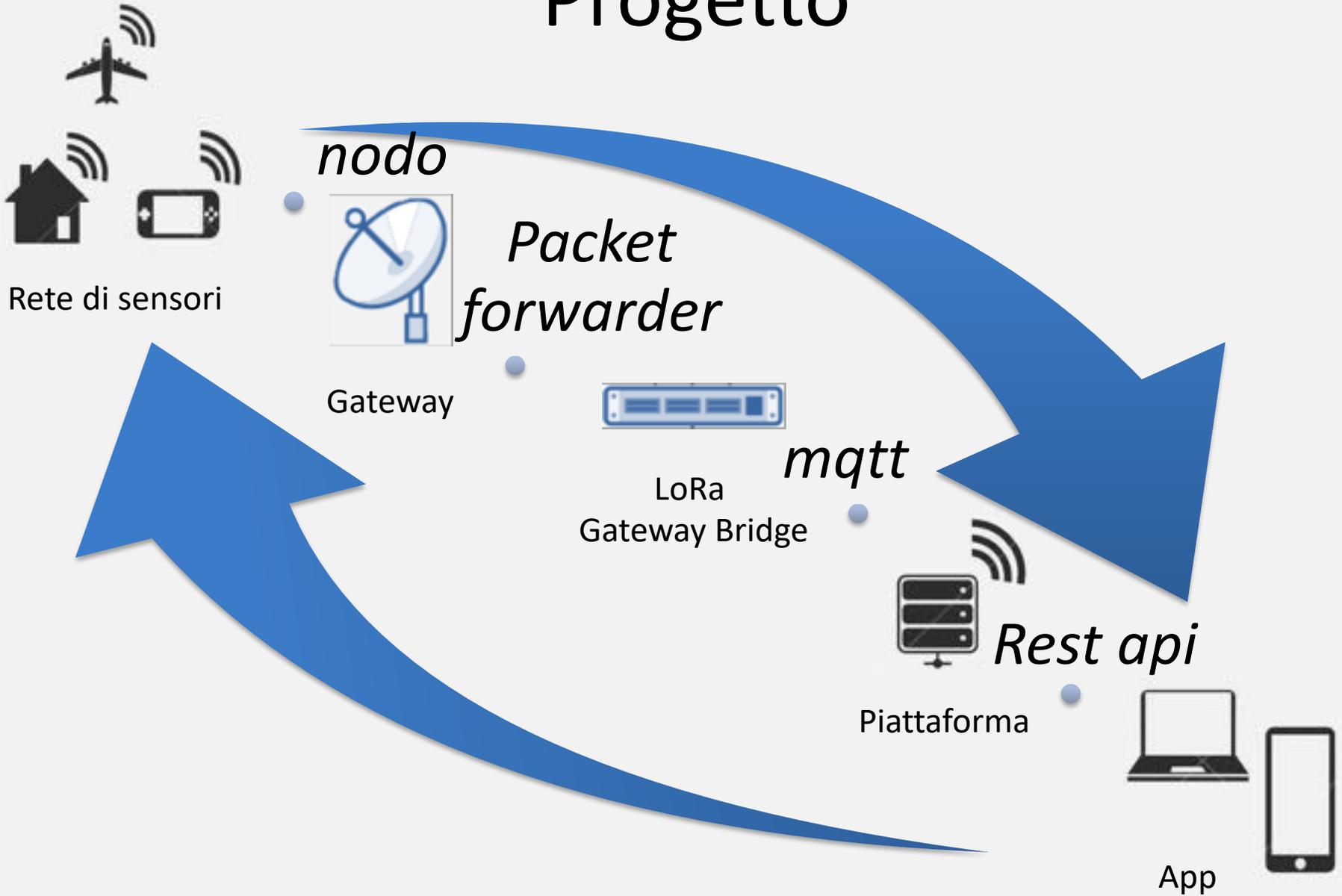
Limiti

- La tecnologia LoRa impone alcune restrizioni: è possibile utilizzare il gateway bridge solo se il nodo che desidera inviare ad esso i dati rispetta un particolare protocollo UDP, denominato packet forwarder, di cui l'alleanza LoRa è proprietaria, anche se open source. Questo rende onerosa in termini di tempi e costi l'integrazione del bridge con dispositivi che, seppur LoRa-like, non rispettano l'intera configurazione software.

Progetto



Progetto



Conclusioni

- La soluzione proposta si è rivelata efficace nel corso della sperimentazione condotta
- L'applicazione della tecnologia LoRa ha permesso di rendere l'architettura abilitante più scalabile, sicura e facilmente integrabile con molti dispositivi
- In sviluppi futuri, sarà possibile testare la stessa architettura con dispositivi LoRa in molteplici campi di applicazione, senza perdere di efficacia