

PROGETTO TOO(L)SMART

OUTPUT AZIONE 2 - O.2.e

Codice Output	O.2.e
Denominazione	Documento di Analisi di sostenibilità economica e business model innovativi per la gestione di reti aperte IoT
Unità di Misura	Numero
Valore Target	1
Enti coinvolti	Ente Responsabile: Comune di Torino Enti Partecipanti: tutti
Descrittivo:	
<p>Durante il Progetto, è stato coinvolto il Consorzio TOP-IX (TOriNO Piemonte Internet eXchange), partner dell'Open Lab Torinese denominato "Torino City Lab" e ingaggiato in attività di assistenza tecnica al laboratorio tematico di testing IoT/IoD.</p> <p>Topix è un Consorzio nato con l'obiettivo di costruire e gestire l'infrastruttura di rete per fornire i servizi tipici di un Internet Exchange e per promuovere e supportare, attraverso il Development Program, progetti di innovazione tecnologica e/o di business basati sull'utilizzo di Internet a banda larga).</p> <p>Nell'ambito di Too(l)smart hanno condotto uno studio metodologico per confrontare le piattaforme orizzontali (generaliste) per la gestione ed analisi di dati provenienti da <i>device IoT</i>.</p> <p>Sono state presi in considerazione sedici fra i principali software disponibili</p> <p>Le piattaforme sono state divise equamente fra Software Open Source e Software Proprietari.</p> <p>Le prime (Open source) utilizzano un software libero e gratuito: avendo accesso al codice sorgente, chiunque può utilizzarlo e modificarlo a piacimento. Le seconde, sono proprietarie di determinati produttori.</p> <p>L'analisi e il confronto delle piattaforme operato nello studio si poneva come obiettivo quello di orientare nella scelta della piattaforma più adatta alla gestione di reti IoT e dei dati provenienti dai <i>device</i> di tali reti, passando in rassegna le caratteristiche dei due tipo di piattaforme e gli aspetti da considerare per individuare la preferibilità di una o dell'altra soluzione a seconda delle situazioni e delle necessità.</p> <p>L'analisi approfondisce anche aspetti quali le differenze tra soluzioni "verticali" (strutturate per "ambito", ad es. applicazioni Internet of Things per l'assistenza sanitaria, mobilità, domotica) e "orizzontali" (trasversali) e l'Edge computing. Chiude l'analisi una riflessione economica comparativa in grado di orientare le PA nella scelta del modello di piattaforma IoT più adatta per esigenze, competenze del personale interno e capacità di esposizione finanziaria. Tutti elementi questi che informano una adeguata analisi di sostenibilità economico-finanziaria a supporto delle PA, anche nella scelta o meno di procedere con il riuso della BP Too(L)smart.</p>	
Allegati:	
IoT Platform	
Guida alla Tabella confronto Piattaforme IoT	

Guida alla
“Tabella confronto piattaforme IoT”

Indice

1. Considerazioni generali
2. Glossario colonne
3. Piattaforme Open Source e Proprietarie
4. Soluzioni Open e Soluzioni Proprietarie
5. Soluzioni Orizzontali e Soluzioni Verticali
6. Edge Computing
7. Valutazione economica comparativa

1.Considerazioni generali

Nella tabella proposta per il confronto delle piattaforme orizzontali (generaliste) per la gestione ed analisi di dati provenienti da device IoT, sono state prese in considerazione sedici fra i principali software disponibili.

Le piattaforme sono divise equamente fra Software Open Source e Software Proprietari.

La scelta finale non sarà certamente semplice: molte delle piattaforme analizzate sono molto simili tra loro in termini di funzionalità e protocolli di comunicazione gestiti.

Questo documento intende dare alcune chiavi di lettura per una maggiore comprensione della tabella allegata.

2.Glossario colonne

Tipologia

La tipologia del software ovvero Open Software o Proprietario.

Data Storage

Dove vengono memorizzati i dati provenienti da import manuale o automatico e dai dispositivi IoT

Linguaggi/SDK

I linguaggi di programmazione utilizzabili per poter analizzare i dati presenti nella piattaforma e se sono presenti degli SDK. Il Software Development Kit (SDK, traducibile in italiano come "pacchetto di sviluppo per applicazioni"), indica genericamente un insieme di strumenti per lo sviluppo e la documentazione di software. - *Wikipedia*

Protocolli Output

I protocolli di comunicazione disponibili per poter comunicare con la piattaforma ed estrarre i dati su di essa memorizzati.

Device Management

Indica la possibilità di gestire i vari dispositivi IoT collegati alla piattaforma.

Il termine "gestione" è molto generalista e può essere inteso come:

- Update del software a bordo del device
- Modifica dei parametri di configurazione del device
- Indicazione della location fisica del device
- Gestione dell'intelligenza a bordo del dispositivo (ove disponibile)

Protocolli Device

I protocolli di comunicazione con cui i device IoT comunicano i dati misurati alla piattaforma. Alcuni protocolli (es. REST API) sono delle fotografie dei dati in un preciso istante di tempo mentre altri (es. MQTT) possono essere dei flussi di dati.

Sicurezza

Il tipo di crittografia con i quali vengono comunicati i dati da e verso la piattaforma.

Real Time

La possibilità di poter accedere ad una dashboard più o meno customizzabile per la visualizzazione dei dati raccolti in tempo reale.

Edge Computing

La possibilità di avere software o hardware dedicato per l'utilizzo dell'Edge Computing e di conseguenza la possibilità di fare dell'attuazione periferica (vicino al device IoT).

Analytics

Gli strumenti/software/prodotti utilizzati dalla piattaforma per fare Data Analytics.

Export/Data Visualization

Le modalità di esportazione dei dati in formato interpretabile e/o la generazione automatica di widget per visualizzazione grafica.

Network Device

I protocolli di Livello 2 della Pila ISO-OSI con i quali la piattaforma è in grado di dialogare.

Budget

Link dove reperire informazioni sui costi e prezzi delle singole piattaforme.

3. Piattaforme Open Source e Proprietarie

Le piattaforme prese in considerazione si possono dividere in due grosse macro categorie:

- Open Source
- Proprietarie

Open Source	Proprietarie
<ul style="list-style-type: none">• KAA/KAAIOT• Mainflux• SiteWhere• Stack4Things• Thinger.io• Thingsboard• Yucca SmartData Platform• WSo2	<ul style="list-style-type: none">• Amazon AWS IoT• Bosch IoT Suite• Carriots• Eurotech Everyware Cloud• HP Enterprise• IBM Watson• Microsoft Azure IoT Hub• TheThings.io

4. Soluzioni Open Source e Soluzioni Proprietarie

Il software Open Source è **libero e gratuito** per definizione: avendo accesso al codice sorgente, chiunque può utilizzarlo e modificarlo a piacimento. Non sempre esistono versioni “ready to use” del software Open Source, ovvero versioni che possono essere semplicemente installate e utilizzate. In alcuni casi è necessario “costruire” l’applicazione a partire dal codice sorgente e quindi avere competenze tecniche in ambito di sviluppo software.

Se l’uso del software Open Source è gratuito, di solito non lo è l’assistenza tecnica.

La risoluzione dei problemi difatti è demandata alla “community” che può fornire una prima assistenza tramite forum e guide online.

In alcuni casi invece può essere messa a disposizione a pagamento dagli sviluppatori del software offrendo servizi per problematiche più delicate e personalizzazioni avanzate.

Molto spesso il software Open Source dà origine a due filoni dello stesso prodotto: la **versione Community** e la **versione Enterprise**. Come suggerisce il nome, la prima è completamente gratuita e gode del solo supporto della community che contribuiscono alla sua crescita ed evoluzione. La seconda è, invece, orientata ad aziende che necessitano di strumenti robusti, facili da usare e con un supporto continuo 24/7 che le metta al riparo da eventuali perdite di tempo o blocchi del loro “core business”.

I fattori da considerare nella scelta degli strumenti da utilizzare sono fondamentalmente il **tempo necessario** per realizzare la soluzione desiderata e i **costi da sostenere** per realizzarla e garantirne il mantenimento.

Soluzioni Open Source

Una soluzione Open Source è preferibile nelle seguenti situazioni:

- si vuole realizzare velocemente un **prototipo economico** e sono sufficienti dei template grafici semplici;
- la soluzione realizzata è ad uso interno dell'azienda oppure si hanno a disposizione il tempo e le **competenze tecniche** necessarie per modificare i template messi a disposizione dalla piattaforma, per trasformare il prototipo nella soluzione finale;
- si vuole avere pieno controllo sulla **scelta delle tecnologie** da utilizzare; si vogliono utilizzare tecnologie all'avanguardia difficilmente disponibili su piattaforme proprietarie;
- si dispone di un **partner tecnologico** in grado di fornire il supporto tecnico necessario.

Alcuni aspetti da considerare nella scelta di soluzione Open Source sono i seguenti:

- verificare la "vitalità" della community di sviluppatori e possibilmente scegliere una piattaforma sponsorizzata da qualche azienda. Questo dà maggiori garanzie sulla longevità del software open source individuato e sulla disponibilità di aggiornamenti continui;
- pur avendo a disposizione l'intera piattaforma, spesso alcune funzionalità sono implementate tramite plugin che potrebbero essere o diventare a pagamento. Porre attenzione nella scelta dei plugin e limitarli al minimo indispensabile;
- verificare l'attenzione della community ai problemi legati all'upgrade della piattaforma, come la realizzazione di guide o tool che aiutino in questa operazione delicata;
- In caso di utilizzo di sistemi operativi o hardware di proprietà, considerare l'obsolescenza di tali sistemi nel medio e lungo periodo rispetto alla piattaforma scelta.

Soluzioni Proprietarie

Una soluzione Proprietaria è preferibile nelle seguenti situazioni:

- è fondamentale realizzare in **tempi brevi** una **soluzione robusta e semplice** da usare;
- **non è importante poter personalizzare** in modo estremo l'aspetto della soluzione realizzata;
- non è necessario avere il pieno controllo delle tecnologie utilizzate;
- non è richiesta l'integrazione di tecnologie all'avanguardia;

- si vuole essere al riparo da problemi legati a bug, aggiornamenti e sicurezza del sistema (assistenza 24/7).

Alcuni aspetti da considerare nella scelta di soluzione Proprietaria sono i seguenti:

- scegliere un produttore solido e affermato, che garantisca supporto a lungo termine. Le aziende nascono e muoiono e la migrazione da una piattaforma proprietaria ad un'altra non sempre è possibile o agevole;
- assicurarsi di avere pieno accesso ai dati caricati nella piattaforma per poterli recuperare in qualsiasi momento;
- verificare l'attenzione del produttore all'uso di nuove tecnologie e all'integrazione con sistemi esterni.

5. Soluzioni Orizzontali e Soluzioni Verticali

L'equilibrio tra piattaforme orizzontali e ambienti verticali è uno degli elementi più critici per l'evoluzione del mercato Internet of Things in Italia nei prossimi 10 anni. Infatti, mentre proliferano soluzioni orientate verticalmente (ad es. applicazioni Internet of Things per l'assistenza sanitaria, mobilità, domotica), i fornitori di soluzioni IoT sviluppano piattaforme orizzontali che abilitano, mediante servizi core, le diverse soluzioni verticali.

I leader del mercato Internet of Things che sono già attivi su questo percorso di sviluppo sono:

- Amazon Web Services IoT;
- HP Enterprise;
- IBM Watson;
- Microsoft Azure IoT.

In un prossimo futuro le piattaforme orizzontali diventeranno il minimo comune denominatore in un universo sempre più connesso di "smart environment".

Pensiamo ad esempio alle Smart Urban Infrastructures (SUI); quelle infrastrutture di secondo livello che erogano ad esempio servizi LPWA abilitanti soluzioni IoT per le Smart City e che fanno riferimento sia a frequenze radiomobili su banda licenziata che banda non licenziata.

6. Edge Computing

Tra le tante caratteristiche che una piattaforma IoT deve avere – sul piano computazionale, applicativo, della connettività, della capacità di analisi e della sicurezza – per essere performante, il corretto posizionamento delle risorse di calcolo nella rete è certamente tra quelle fondamentali.

Nelle prime concezioni dell'architettura IoT si immaginava che i dispositivi alla periferia della rete si sarebbero limitati a raccogliere dati.

Il modello generico di Edge Computing prevede che i dispositivi e i sensori installati alla estrema periferia della rete comunichino con un device intermedio - il gateway - e non direttamente con il cloud.

Il gateway esegue una prima analisi dei dati raccolti e spesso anche una selezione di quali inviare a monte, verso il cloud. In questo modo è possibile tra l'altro gestire reazioni immediate ai dati raccolti localmente, evidenziare subito connessioni causali fra eventi e garantire un minimo funzionamento del sistema anche in assenza della connessione al cloud centrale.

Non tutte le piattaforme di analytics pensate per le applicazioni IoT supportano elementi di Edge Computing. Non tutte cioè sono in grado di bilanciare l'elaborazione dei dati tra centro e periferia, considerando i due tipi di elaborazione all'interno di una medesima architettura. Ci sono approcci più cloud-centrici e altri che danno una maggiore autonomia alla periferia.

Un esempio della prima concezione viene da Microsoft, dove il punto di partenza per l'estensione dell'intelligenza alla periferia della rete è Azure Stream Analytics (ASA), la piattaforma cloud per l'analisi dei dati IoT. In estrema sintesi, la piattaforma permette di concentrare nel cloud i flussi dati in tempo reale e di applicarvi funzioni di event processing descritte con un linguaggio simile a SQL.

ASA è una soluzione tipicamente cloud ma Microsoft ne ha sviluppato anche una versione da edge computing. Il principio è quello di installare una piattaforma software specifica (Azure IoT Edge) su dispositivi perimetrali basati su Windows o Linux. Non devono necessariamente essere PC: la piattaforma è aperta anche a sistemi embedded. In questo modo molte delle funzioni di ASA sono eseguite in locale.

Ci sono soluzioni che puntano su un approccio edge computing più semplice ma anche più trasversale, nel senso che si può implementare su dispositivi anche più semplici dei gateway. Questi non offrono funzioni di analytics sofisticate ma eseguono soprattutto operazioni di valutazione e filtro dei dati raccolti. Tipicamente, queste funzioni servono ad attivare il dialogo verso il cloud solo quando i dati raccolti sono diversi da quelli previsti, cioè sintomo di anomalie nel funzionamento del sistema.

Un esempio di questo approccio è la parte Edge Analytics della Watson IoT Platform, che si basa sul progetto Apache Edgent. Edgent è un modulo che permette di eseguire funzioni di valutazione e filtro su piattaforme Java e Android, ad esempio su device semplici quanto il Raspberry Pi. I dati "normali" restano in locale e solo quelli che soddisfano certe condizioni sono trasmessi a monte.

L'edge computing non sostituisce il cloud. Le infrastrutture cloud continueranno a gestire enormi quantità di dati, ma si tratta di dati che, per la maggior parte, non richiedono risposte in tempo reale. Il cloud centralizza l'elaborazione e la memorizzazione dei dati, l'Internet of Things richiede di attivare infrastrutture a livello di edge per consentire il processo decisionale in tempo reale.

7.Valutazione economica comparativa

Una comparazione economica fra le diverse soluzioni proprietarie e non è esercizio estremamente complicato visti i differenti business model proposti da ciascuno di esse.

I parametri che spesso entrano in gioco per tentare tale valutazione sono:

- Numero di device
- Numero di chiamate API alla piattaforma
- Spazio disco occupato
- Moduli integrativi da installare
- Numero di istanze per servizio da accendere (se in cloud)
- Numero di utenti abilitati alla piattaforma

Oltre a queste variabili è utile considerare che se si opta per una soluzione Open Source da installare in una propria Server Farm bisogna aggiungere i costi di:

- Acquisto dei server e housing degli stessi (caso 1)
- Acquisto di macchine virtuali ed loro hosting in cloud provider (caso 2)
- Personale dedicato all'installazione e personalizzazione del software
- Personale dedicato allo sviluppo di nuove funzionalità (se necessario)