



Ricerca di Sistema elettrico

## Piattaforma ICT per la gestione dello Smart District: indicatori di interoperabilità e reingegnerizzazione

C. Novelli, A. Frascella, A. Brutti, N. Gessa, M. Chinnici,  
G. Ponti, G. Santomauro

## PIATTAFORMA ICT PER LA GESTIONE DELLO SMART DISTRICT: INDICATORI DI INTEROPERABILITÀ E REINGEGNERIZZAZIONE

C. Novelli, A. Frascella, A. Brutti, N. Gessa, M. Chinnici, G. Ponti, G. Santomauro (ENEA)

Dicembre 2018

### Report Ricerca di Sistema Elettrico

Accordo di Programma Ministero dello Sviluppo Economico - ENEA

Piano Annuale di Realizzazione 2018

Area: Efficienza energetica e risparmio di energia negli usi finali elettrici e interazione con altri vettori energetici

Progetto: D.7 Sviluppo di un modello integrato di smart district urbano

Obiettivo: D.7a Piattaforma ICT per la gestione dello Smart District

Responsabile del Progetto: Claudia Meloni, ENEA

## Indice

SOMMARIO.....	4
1 INTRODUZIONE.....	5
2 EVOLUZIONE E CONSOLIDAMENTO DELLE SPECIFICHE TRAMITE INDICATORI DI INTEROPERABILITÀ (D7A.1).....	6
2.1 INDICI DI INTEROPERABILITÀ PER LA SCP.....	6
2.1.1 <i>Algoritmo (indici)</i> .....	6
2.1.2 <i>Implementazione (indici)</i> .....	7
2.2 VISTE DI DATI.....	9
2.2.1 <i>Vista UrbanDataset singolo</i> .....	9
2.2.2 <i>Vista Categorie di UrbanDataset</i> .....	9
2.2.3 <i>View KPI</i> .....	11
2.3 VALUTAZIONE SPERIMENTALE DELL'INTEROPERABILITÀ RAGGIUNTA TRAMITE LE SCPS.....	12
2.4 INDICI DI VALUTAZIONE DELLE SPECIFICHE (SCPS).....	13
2.5 AGGIORNAMENTO STRATO SEMANTICO.....	17
3 PROCESSO DI REINGEGNERIZZAZIONE DEL SOFTWARE E MIGLIORAMENTO DEL LIVELLO DI SICUREZZA (D7A2) ..	18
2.6 REINGEGNERIZZAZIONE DEL SW.....	19
2.7 SPOSTAMENTO IDENTITYGATEWAY.....	20
2.8 DUPLICAZIONE DEL WS SERVER.....	20
2.9 ADOZIONE DI ELASTICSEARCH.....	23
5 CONCLUSIONI.....	25
6 RIFERIMENTI BIBLIOGRAFICI.....	26
7 ABBREVIAZIONI ED ACRONIMI.....	27
APPENDICE – SINTESI DELL'ATTIVITÀ DELL'UNIVERSITÀ DI BOLOGNA.....	28

## Sommario

Questo è il Report Tecnico finale per l'obiettivo D7.a "Piattaforma ICT per la gestione dello Smart District", relativo al PAR2018, ovvero l'estensione di tre mesi per permettere il consolidamento e l'evoluzione della piattaforma software di integrazione dati in ambito Smart City.

Il D7.a PAR2018 è organizzato in due sotto-obiettivi:

- D7.a1: "Evoluzione e consolidamento delle specifiche" (Smart City Platform Specification, SCPS);
- D7.a2: "Sviluppo di un prototipo per adesione alle Specifiche" (Smart City Platform, SCP), piattaforma ICT per la gestione di Smart District aderente alle specifiche per l'interoperabilità SCPS.

In questo report tecnico finale PAR2018, estensione di tre mesi al triennio, saranno descritti i due sotto-obiettivi separatamente così come sopra citati.

Si ritiene auspicabile un ulteriore investimento di risorse nella direzione di trasferire la SCP in contesti urbani reali, nazionali ed europei, al fine di permettere di sfruttare le potenzialità dei risultati della sperimentazione fin ora perseguiti.

## 1 Introduzione

Con il titolo “Piattaforma ICT per la gestione dello Smart District” si intende una piattaforma software di integrazione che ha lo scopo di permettere, di integrare dati eterogenei prodotti da diversi Contesti Applicativi verticali nell’ambito Smart City (smart lighting, smart building, smart home, smart street, ecc.) con un approccio basato su specifiche aperte e condivise.

- Le specifiche aperte, sviluppate nell’ambito dei precedenti PAR 2015, 2016 e 2017, sono le **Smart City Platform Specification (SCPS)**
- La piattaforma ICT che aderisce a queste specifiche prende il nome di **Smart City Platform (SCP)**

Una Smart City Platform, dunque, è una piattaforma ICT che permette anche la gestione di Smart District.

Il lavoro del triennio ha seguito un percorso basato sulle seguenti milestone, una per ogni annualità:

- anno 1 (PAR2015): Analisi di Contesti Applicativi Verticali di uno Smart District e progettazione dell’Architettura di Riferimento per la Piattaforma ICT orizzontale;
- anno 2 (PAR2016): Definizione della prima versione delle Specifiche per l’Interoperabilità (formati comuni per la rappresentazione sintattica e ontologia per l’interpretazione semantica) e primo sviluppo di Piattaforma ICT di integrazione;
- anno 3 (PAR2017): Definizione completa delle Specifiche per l’Interoperabilità e sviluppo software del prototipo di Piattaforma ICT aderente alle specifiche SCPS.

Nel PAR2018 si è mantenuta la macro-organizzazione dei due PAR precedenti nelle attività principali:

- D7.a1: Evoluzione e consolidamento delle Specifiche (tramite indicatori di interoperabilità);
- D7.a2: Sviluppo di un Prototipo per adesione alle Specifiche (tramite processo di reingegnerizzazione del software e miglioramento del livello di sicurezza);

per dare continuità al lavoro svolto dei diversi gruppi mantenendo inalterata l’organizzazione.

Essendo il PAR2018 relativo a soli 3 mesi, le risorse hanno permesso di concentrare gli sforzi su uno soltanto dei due prototipi SCP (SCP-ENEA) lasciando inalterato ma comunque funzionante e interoperabile l’altro (SCP-Giotto).

## 2 Evoluzione e consolidamento delle Specifiche tramite indicatori di interoperabilità (D7a.1)

Il sotto-obiettivo D7a1 ha recepito un'importante osservazione degli esaminatori, ovvero la mancanza di indici di prestazione che permettessero di misurare la bontà del risultato raggiunto.

Questa osservazione, inoltre, sposa l'esigenza stessa di poter controllare che i (sempre più) numerosi dati in arrivo alla Smart City Platform dalle molteplici Solution verticali fossero correttamente inviati con la frequenza concordata (configurata nella SCP).

Allo scopo di consolidare e migliorare le specifiche SCPS sono stati definiti:

- Indici di l'Interoperabilità;
- Viste di Dati.

Si è inoltre applicata una metodologia più di alto livello per la valutazione delle specifiche prodotte

Infine, è continuata la collaborazione con l'Università di Bologna allo scopo di consolidare e far evolvere gli strumenti ontology-based (si veda la sintesi di queste attività in APPENDICE).

### 2.1 Indici di Interoperabilità per la SCP

Obiettivo dell'attività è stata la progettazione e l'implementazione di uno strumento in grado di misurare quanto la Smart City Platform fosse interoperabile e quindi quanto "funzionasse bene", in particolare sul livello Collaboration, essendo il suo scopo primario l'integrazione delle diverse Solution verticali del distretto, che mettono a disposizione dell'amministratore della città, dati eterogenei da parte sistemi di fornitori diversi.

A tale scopo si è definito di un **algoritmo** in grado di ricavare, per ciascuna collaborazione Solution-SCP di produzione dati, la percentuale di UrbanDataset correttamente ricevuti dalla piattaforma SCP rispetto a quelli attesi (concordati), su un periodo di monitoraggio.

#### 2.1.1 Algoritmo (indici)

Il parametro input di riferimento è il periodo previsto per l'invio dell'UrbanDataset da parte del verticale. Su tale parametro dovrà essere considerata la seguente casistica:

- 1) L'UrbanDataset deve essere inviato con una frequenza di uno ogni N minuti:
  - Controllo nella history\_production quanti #UD questa solution abbia inviato negli ULTIMI  $N*60$  MINUTI. La percentuale è data da:  $\#UD * 100 / 60$ ;
  - Esempio:  
periodo di invio = 15 minuti => la SCP recupera il numero di UD negli ultimi  $15*60=900$  minuti e trova che #UD=58. Quindi  $58 * 100 / 60 = 96.7\%$
- 2) L'UrbanDataset deve essere inviato con una frequenza di uno ogni N ore:
  - Controllo nella history\_production quanti #UD questa solution abbia inviato nelle ULTIME  $N*24*7$  ORE. La percentuale è data da:  $\#UD * 100 / 24*7$ ;
  - Esempio:  
periodo di invio = 1 ogni ora, recupero il numero di UD nelle ultime  $1*24*7 = 168$  ore (7 giorni) e trovo che #UD=160. Quindi  $160 * 100 / 168 = 95.2\%$
- 3) L'UrbanDataset deve essere inviato con una frequenza di uno ogni N giorni:
  - Controllo nella history\_production quanti #UD questa solution abbia inviato negli ULTIMI  $N*7$ GIORNI. La percentuale è data da:  $\#UD * 100 / 7$ ;

- Esempio:  
periodo di invio = 1 ogni 2 giorni, recupero il numero di UD negli ultimi  $2*7 = 14$  giorni e trovo che #UD=14. Quindi  $14 * 100 / 7 = 200\%$
- 4) L'UrbanDataset deve essere inviato con una frequenza di uno ogni N settimane:
- Controllo nella history\_production di quanti #UD questa solution ha inviato nelle ULTIME  $N*4$  SETTIMANE. La percentuale è data da: **#UD \*100/4;**
  - Esempio:  
periodo di invio = 1 ogni 3 settimane, recupero il numero di UD nelle ultime  $3*4 = 12$  settimane e trovo che #UD=2. Quindi  $2 * 100 / 4 = 50\%$
- 5) L'UrbanDataset deve essere inviato con una frequenza di uno ogni N mesi:
- Controllo nella history\_production quanti #UD questa solution ha inviato negli ULTIMI  $N*2$  MESI. La percentuale è data da: **#UD \*100/2;**
  - Esempio:  
periodo di invio = 1 ogni 2 mesi, recupero il numero di UD negli ultimi  $2*2 = 4$  mesi, trovo che #UD=2. Quindi  $2 * 100 / 2 = 100\%$
- 6) NON sono contemplati i casi di frequenza su base annuale o su base uguale o inferiore al secondo, ovvero non sono calcolati indici su periodi di monitoraggio espressi in anni o in secondi, in quanto rispettivamente ritenuti troppo elevati o troppo piccoli e non configurati nelle collaborazioni del Registry.

N.B. Il calcolo può procedere SOLO SE la differenza temporale tra il primo UD arrivato da quella Solution e il momento attuale è maggiore del CAMPIONE.

### 2.1.2 Implementazione (indici)

Sulla base dei precedenti indici, è stata implementata nella GUI<sup>1</sup> della SCP, la visualizzazione degli indici di interoperabilità in forma grafica, al fine di permettere all'amministratore della piattaforma il monitoraggio delle collaborazioni con i diversi verticali.

- Gli indici vengono visualizzati dagli utenti Admin, Developer e Solution (quindi non dai non-registrati).
- Il "grafico di prestazione" riporta le diverse Solution sull'asse delle ascisse (x), per l'utente Solution vi sono solo i dati di propria competenza.
- Per ogni Solution vengono riportate tante colonne quanti sono gli UD prodotti (configurati nella sezione "production"). Quindi vi sono diversi gruppi di colonne, un gruppo per ogni Solution.

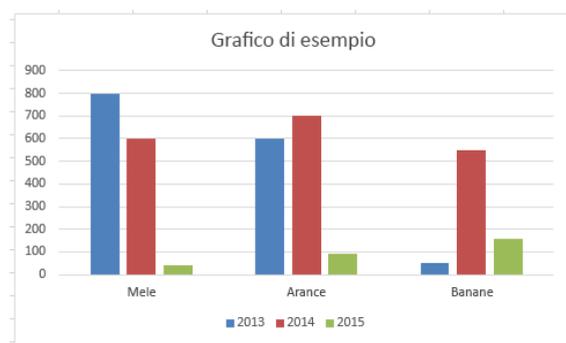


Figura 1. Grafico di esempio

<sup>1</sup> Per GUI, Graphical User Interface, si intende l'interfaccia grafica dell'applicazione

Sull’asse delle ordinate (y) viene mostrata una percentuale il cui valore è “la percentuale di quanti Urban Dataset sono stati correttamente prodotti, rispetto a quanto dichiarato nella collaborazione relativa, su una finestra temporale ben precisa”.

Inoltre sono mostrate solo le colonne calcolabili, ovvero:

- il cui tempo di campionamento è sufficiente;
- i cui parametri in input sono tutti presenti (not null).

Come mostrato nella Figura 1, gli indici sono rappresentati con degli istogrammi (posti nella home page della SCP), in cui il colore delle colonne evidenzia subito se lo scambio dati sta soddisfa le condizioni identificate a livello di Collaborazione. I colori usati sono i seguenti:

- VERDE: lo scambio dati sta rispettando al 100% l’accordo di collaborazione;
- GIALLO: alcuni UrbanDataset, meno del 10%, non sono stati ricevuti;
- ROSSO: si sta verificando una perdita di UrbanDataset superiore al 10% (questo può avvenire se la collaborazione è in fase di test oppure se si stanno verificando **problemi rilevanti**).

Inoltre, passando col puntatore del mouse sopra le colonne, viene mostrato il valore percentuale di UrbanDataset ricevuti rispetto a quelli attesi.

L’esito generale, ad oggi, è abbastanza buono e, nel tempo ogni collaboration sta raggiungendo il 100% di UrbanDataset scambiati, ovvero l’interoperabilità pienamente raggiunta sul livello Collaboration. La nuova home page con gli indici è riportata in Figura 2. Nella situazione rappresentata, si vede come la solution verticale “Smart Building Casaccia” abbia inviato il 100% degli UrbanDataset “Consumption” concordati (ovvero 1 al giorno negli ultimi 7 giorni).

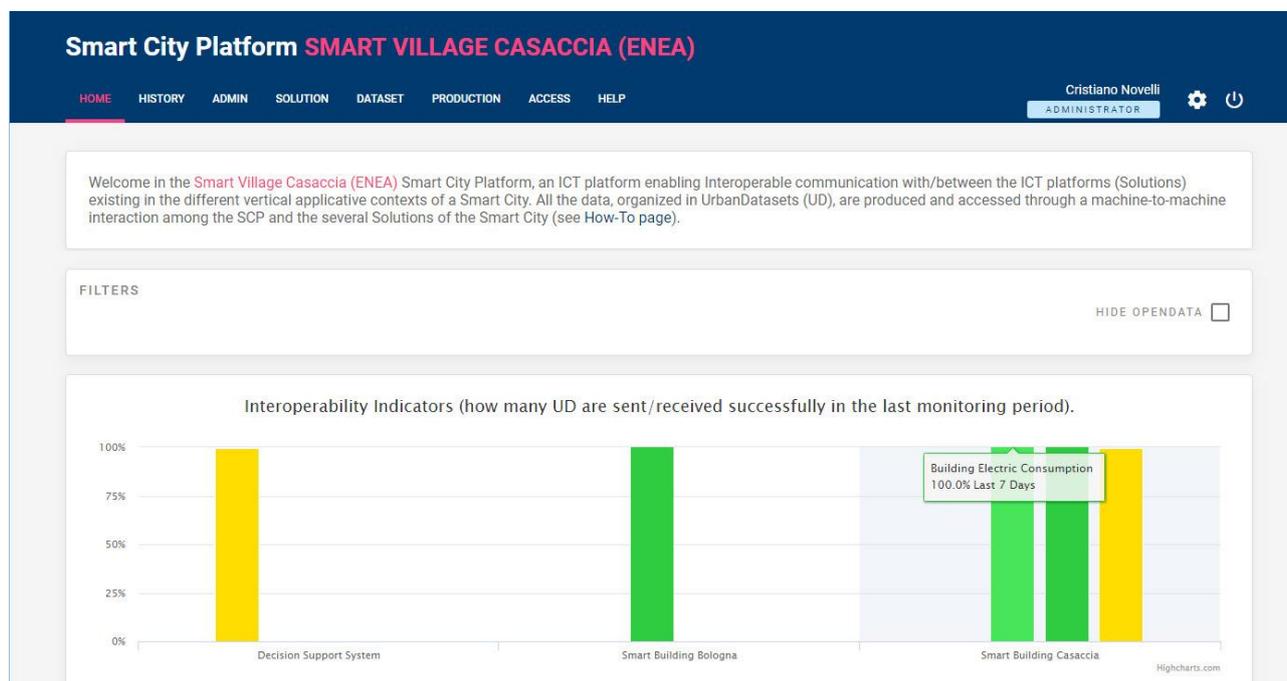


Figura 2. Diagramma degli indici di Interoperabilità

## 2.2 Viste di Dati

Oltre agli indicatori di interoperabilità sono state progettate e implementate nell'interfaccia anche alcune "viste"<sup>2</sup> di dati, recuperando i valori direttamente da un UrbanDataset o calcolando (come nel caso dei consumi elettrici) il dato su un periodo che preveda il data-fusion di più UD.

### 2.2.1 Vista "UrbanDataset singolo"

Nella sezione History, oltre alla possibilità di effettuare il download dell'UD in formato JSON, è stata implementata una vista che permette di visualizzare un UrbanDataset in maniera schematica.

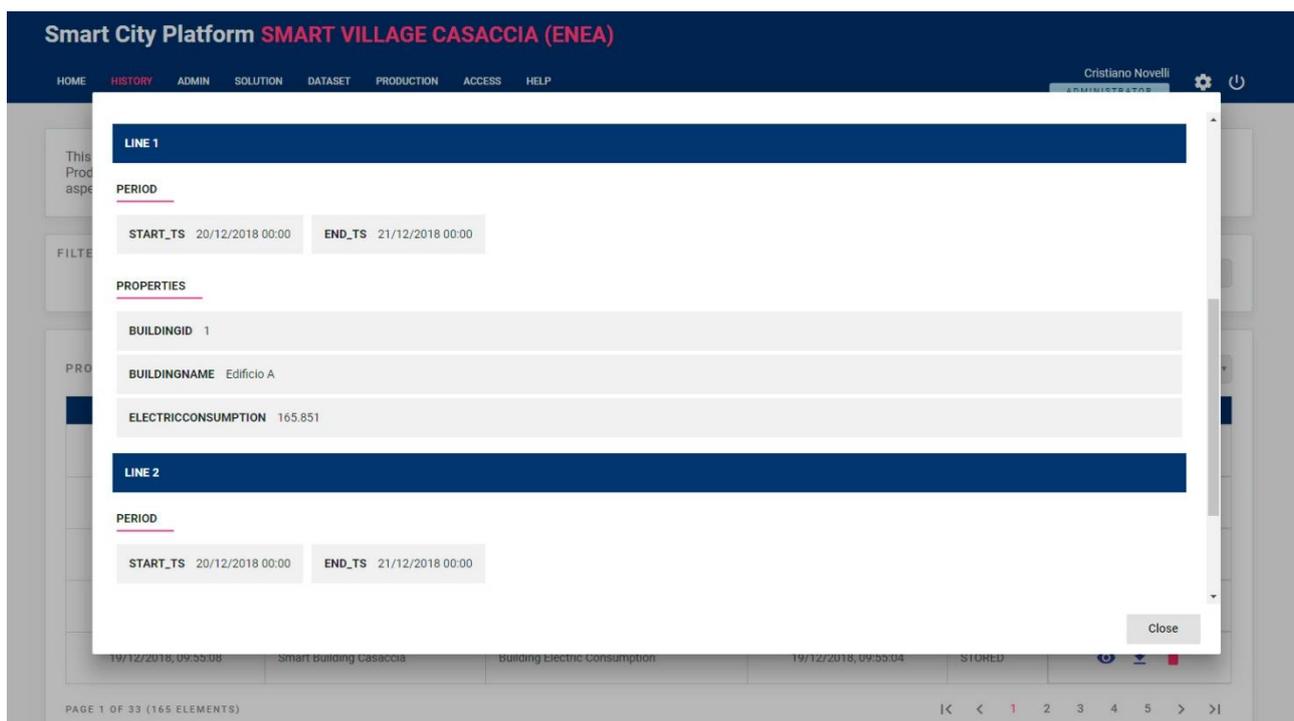


Figura 3. Visualizzazione singolo UrbanDataset

Nell'esempio riportato possiamo vedere due linee dell'UrbanDataset "Building Electric Consumption", inviato da una Solution "Smart Building"; la prima linea ci indica che l'Edificio A ha effettuato un consumo il giorno 20/12/2018 di 165.851 kWh.

### 2.2.2 Vista "Categorie di UrbanDataset"

Nella sezione "DATESET" della GUI, dove sono mostrati tutti gli UrbanDataset definiti nell'Ontologia e che sono stati configurati come supportati dall'istanza corrente di SCP, è stata migliorata l'usabilità della sezione, fornendo una visualizzazione per categorie.

Questa classificazione è contenuta nelle specifiche "SCPS Semantic Level" e, nella direzione di riutilizzarla nell'interfaccia, è stata riportata anche all'interno del Registry tramite le tabelle *dataset\_category* e *dataset\_subcategory* definendole nelle specifiche "SCPS Collaboration Level"; le tabelle in questione al momento riportano la suddivisione in categorie e sottocategorie ma non vi è ancora un meccanismo di sincronizzazione automatica con l'Ontologia (sviluppo che si auspica possa essere fatto in futuro).

La visualizzazione per categorie degli UrbanDataset supportati dalla SCP è mostrata nella seguente Figura 4.

<sup>2</sup> Con Vista (o view) intendiamo la rappresentazione tabellare di un sottoinsieme dei dati presenti nel sistema, selezionati sulla base di uno specifico criterio

Smart City Platform **SMART VILLAGE CASACCIA (ENEA)**
Cristiano Novelli  
ADMINISTRATOR

HOME HISTORY ADMIN SOLUTION **DATASET** PRODUCTION ACCESS HELP

This section presents all the UrbanDatasets (UD) defined in the Ontology and supported by the Smart City Platform (SCP). The ADMINISTRATOR and DEVELOPER users manage the DATASET through the four basic CRUD functionalities (Create, Read, Update, Delete).

MANAGE CATEGORIES
INSERT DATASET

---

**Public safety, policy & Em.Res.**

**Communication**

DATASET	DESCRIPTION	DATE	URI	
Whatever	Consentire comunicazione di base con adesione minimale alle specifiche.	13/03/2018, 15:25:20	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#Whatever	✎ 🗑

---

**Energy**

**Energy demand**

DATASET	DESCRIPTION	DATE	URI	
Counter Reading	Rilevazione del consumo elettrico di un sistema di illuminazione Pubblica con impianti trifase	26/11/2018, 10:06:10	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#CounterReading	✎ 🗑
Counter Reading Monophase	Rilevazione del consumo elettrico di un sistema di illuminazione Pubblica con impianti monofase	26/11/2018, 10:09:40	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#CounterReadingMonophase	✎ 🗑

---

**Waste**

**Citizens engagement**

DATASET	DESCRIPTION	DATE	URI	
Social Network Page Community Feedback	Inviare i dati sulle attività svolte dagli utenti relativamente ad una "Pagina" del social network	24/07/2018, 12:34:42	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SocialNetworkPageCommunityFeedback	✎ 🗑

---

**Water and waste water**

**Waste water management**

DATASET	DESCRIPTION	DATE	URI	
Treatment Plant Waste Water Performance	Consumo elettrico medio giornaliero per m3 depurato e Flusso di acqua totale per m3.	03/12/2018, 15:27:11	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#TreatmentPlantWasteWaterPerformance	✎ 🗑
Treatment Plant Waste Water Records	Anagrafica impianti di depurazione: fornisce una descrizione univoca degli impianti di depurazione monitorati.	03/12/2018, 14:15:02	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#TreatmentPlantWasteWaterRecords	✎ 🗑

---

**Built environment**

**Smart Home**

DATASET	DESCRIPTION	DATE	URI	
Home Aggregated Electric Consumption	Invio dell'energia elettrica totale consumata dalle abitazioni appartenenti all'area monitorata dall'aggregatore, o aggregatori, operanti nel distretto.	07/11/2018, 11:53:22	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#HomeAggregatedElectricConsumption	✎ 🗑
Home Aggregated Electric Production	Invio dell'energia elettrica totale prodotta dalle abitazioni appartenenti all'area monitorata dall'aggregatore, o aggregatori, operanti nel distretto.	07/11/2018, 11:55:36	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#HomeAggregatedElectricProduction	✎ 🗑
Home Aggregated Records	Invio dei dati aggregati di caratterizzazione delle abitazioni appartenenti all'area monitorata dall'aggregatore, o aggregatori, operanti nel distretto.	07/11/2018, 11:56:40	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#HomeAggregatedRecords	✎ 🗑

---

**Smart Building**

DATASET	DESCRIPTION	DATE	URI	
Building Device Anomalies	Conteggia le anomalie di uno o più sensori associati a un determinato palazzo, calcolati per una particolare causa, su una particolare finestra temporale.	07/08/2018, 11:14:20	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#BuildingDeviceAnomalies	✎ 🗑
Building Electric Consumption	Registrare il consumo energetico giornaliero di uno o più edifici, al fine di comunicarlo, mensilmente, al WebGIS e permetterne la visualizzazione sulla mappa.	01/12/2017, 16:09:06	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#BuildingElectricConsumption	✎ 🗑
Building Energy Anomalies	Conteggia le anomalie (low, mean, high) associate a una particolare causa, a un determinato palazzo, calcolate su una particolare finestra temporale.	06/04/2018, 14:14:56	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#BuildingEnergyAnomalies	✎ 🗑
Building Records	Anagrafica edifici: fornisce una descrizione univoca dei palazzi monitorati nel contesto applicativo verticale Smart Building.	01/12/2017, 16:07:34	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#BuildingRecords	✎ 🗑

---

**Land use and management**

DATASET	DESCRIPTION	DATE	URI	
Weather Condition	Fornire situazione meteo	01/10/2018, 14:39:41	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WeatherCondition	✎ 🗑

Figura 4. Vista delle categorie di UrbanDataset

### 2.2.3 Vista “KPI”

Oltre alla vista del singolo UrbanDataset e alla vista delle categorie, per mostrare (anche se in maniera molto limitata) le potenzialità derivanti dall’attingere a un ampio bacino di dati armonizzati provenienti dallo stesso tessuto urbano, si è pianificata e implementata una vista di KPI, ovvero una singola vista che contenesse UrbanDataset diversi.

Nella seguente Figura 5 è mostrato uno screenshot del risultato da cui si evince come sia possibile, agilmente, recuperare dati da diversi UrbanDataset e mostrarli in un’unica vista centralizzata.

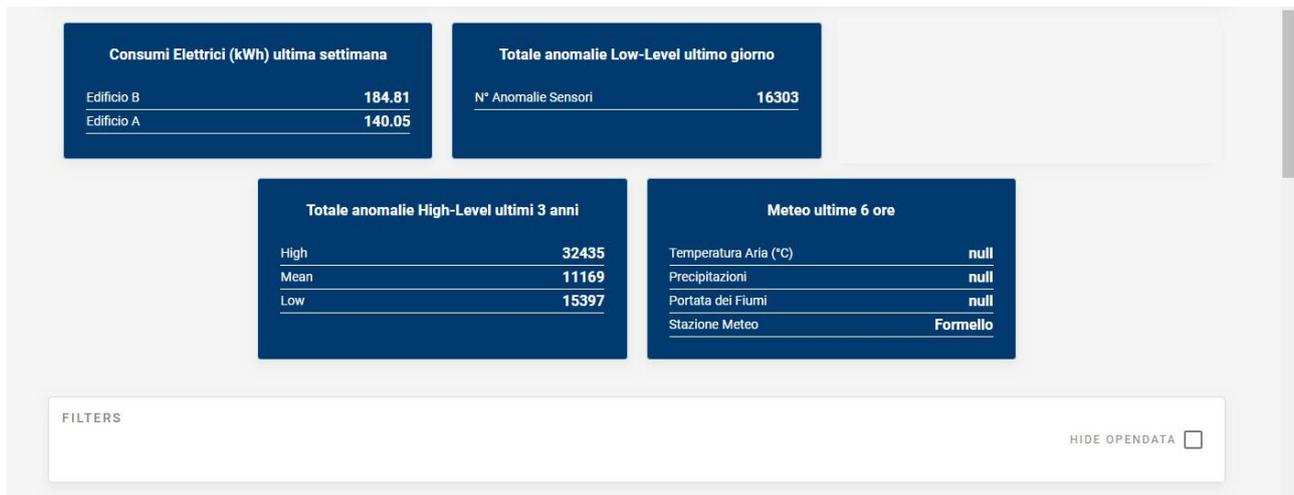


Figura 5. Vista dei KPI

Nell’esempio proposto sono visibili:

- i consumi elettrici espressi in kWh degli edifici A e B, calcolati sull’ultima settimana (sono quindi stati prelevati, tramite il metodo *UrbanDatasetGateway.searchingRequest*, i sette UD inviati quotidianamente e su di essi è stata fatta la media);
- il totale delle anomalie LowLevel (ovvero relative ai sensori) rilevate sull’ultimo giorno nell’intero distretto Smart Village (sono molte poiché molto sensori devono essere ri-attivati);
- il totale delle anomalie HighLevel (ovvero date dall’uso energetico) sugli ultimi 3 anni (questi valori sono direttamente calcolati dalla Solution verticale Smart Building);
- il meteo relativo alle ultime 6 ore (si noti che nel momento in cui si scrive i valori sono *null* poiché la sorgente dati, un web service di OPENDATA di ARPA, è al momento non funzionante e quindi la Solution verticale DSS invia l’UrbanDataset seguendo scrupolosamente l’accordo ma i valori sono non disponibili).

In questo esempio sono assenti grafici che mostrino l’andamento dei dati nel corso del tempo e mappe che mostrino i dati andando a posizionarli nello spazio (così come abbiamo visto effettuata la Solution WebGIS).

È bene far notare che non dovrebbe essere compito della Smart City Platform fornire servizi a un distretto/città le cui esigenze sono uniche e note solamente agli energy manager di quel contesto. Il compito della Smart City Platform è integrare e armonizzare i dati eterogenei provenienti dalle diverse Solution verticali dello Smart District (o Smart City) e metterli a disposizione dei produttori di Servizi.

In questo modo la SCP si conferma essere strumento per agevolare la creazione della Smart City e non (come spesso succede) strumento ICT in competizione con quelli prodotti dalle software house. La Smart City Platform, infatti, intende affiancare le piattaforme esistenti nelle città, non sostituirsi ad esse.

### 2.3 Valutazione sperimentale dell'interoperabilità raggiunta tramite le SCPS

Grazie ai precedenti indici e alla loro implementazione all'interno della SCP GUI, è stato possibile effettuare, durante il PAR2018, delle valutazioni sperimentali dell'interoperabilità raggiunta, testandole tramite l'implementazione della connessione delle diverse applicazioni verticali con la SCP:

- 6 casi studio in totale
- 8 piattaforme connesse in produzione (invio di UrbanDataset verso l'SCP) e 2 in accesso (ricezione di UrbanDataset)
- Invii e ricezioni con cadenze che vanno da invio giornaliero fino a invio orario, con diverse modalità (overritten, stored)
- Ogni caso studio gestito da un programmatore diverso
- Esistenza di due prototipi diversi di SCP, basati su tecnologie diverse, ma rispettando le SCPS per le interfacce

Ciò ha permesso di verificare che:

- Il **tempo** necessario all'implementazione della prima connessione da parte del programmatore dell'applicazione verticale, varia dai 2 giorni/uomo, nel caso si usi il client per l'invio fornito fra i tool dell'SCP, ai 7 giorni/uomo nel caso il programmatore voglia implementare il proprio tool (questi tempi sono comprensivi dello studio delle specifiche). Questo tempo si riduce a circa 2 ore/uomo di lavoro per implementazioni di altri UrbanDataset sulla stessa applicazione verticale. Questa indicazione si traduce in una valutazione positiva dell'**usabilità** delle specifiche
- Una volta implementata, la connessione funziona in modo **stabile**, com'è possibile verificare dal monitoraggio degli **indicatori di interoperabilità** e della sezione history della **SCP**. I casi di errori negli invii, subito evidenziati da tali indicatori (e verificati andando a guardare gli invii dalla pagina history), che si sono verificati durante i PAR2018 sono sempre stati da attribuire a problemi nell'applicazione verticale. In questo senso gli indicatori si sono rivelati un potente strumento di monitoraggio delle sorgenti dati.

In particolare i problemi rilevati sono stati di due tipi:

- 1) L'indicatore di interoperabilità mostra che un UrbanDataset non viene ricevuto secondo le tempistiche previste dal livello collaborazione.

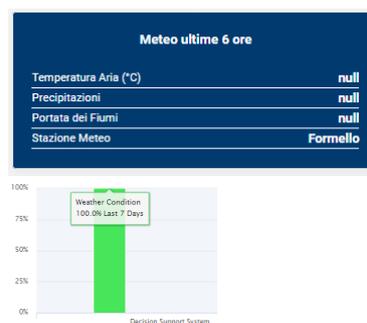
I casi di questo tipo che si sono verificati erano dovuti a malfunzionamenti nel client che inviava i dati (tipicamente, il processo si era fermato, per ragioni derivanti dall'infrastruttura del server in cui risiede)



- 2) I KPI mostrano che, pur arrivando regolarmente gli UrbanDataset, i dati sono vuoti.

In questo caso, pur funzionando lo strato di interoperabilità è la centralina meteo che presenta una rottura e non sta rivelando correttamente i dati.

Questo caso si evidenzia l'utilità degli indicatori KPI implementati nella GUI



- Inoltre, nel corso del PAR 2017 era già stato verificato come le solution verticali riescano a comunicare indifferentemente con entrambe le SCP, semplicemente cambiando la loro configurazione e senza sforzo ulteriore di sviluppo software. Questo ci dà una valutazione positiva dell'**interoperabilità** delle specifiche, ma anche della **replicabilità** delle SCP e mette in evidenza le potenzialità derivate da una diffusione dell'uso delle specifiche.

## 2.4 Indici di valutazione delle Specifiche (SCPS)

La valutazione di un set di specifiche per l'interoperabilità è stata affrontata, per gli standard relativi alle Smart Grid, all'interno del progetto europeo ELECTRA IRP (1). La metodologia messa a punto in quel caso faceva leva sulle tipologie di applicazioni relative alla Smart Grid, sul modello SGAM (Smart Grid Architecture Model) per lo sviluppo e l'analisi di architetture per la Smart Grid (2) e sulla metodologia CAMSS (Common Assessment Method for Standards and Specifications) per la scelta degli standard da parte delle pubbliche amministrazioni (3) e aveva un obiettivo più ampio (classificazione di un ampio insieme di standard e comparabilità fra essi) di quello che ci si è posti qui (effettuare una valutazione delle SCPS). Permutando però l'approccio e i parametri principali della metodologia si può definire una griglia di valutazione applicabile alle SCPS.

I parametri individuati per la valutazione dell'interoperabilità di un set di specifiche, sono dunque i seguenti:

1. **Maturità:** si può valutare sulla base del numero delle *realise* esistenti, manutenibilità delle specifiche, aggiornamenti recenti e sulla stabilità delle tecnologie su cui esse si basano. Al fine di valutare la maturità, i parametri da valutare sono:
  - Numero di release: una specifica che non è alla sua prima versione avrà avuto il tempo di confrontarsi con implementazioni reali e aggiustare problemi presenti all'inizio
  - Presenza di aggiornamenti recenti: indicano vitalità del gruppo di lavoro sulle specifiche e dunque una specifica che non è abbandonata a se stesso
  - Data di inizio dello sviluppo: capire da quanto tempo il gruppo di lavoro ci sta lavorando dà l'idea di quanto essa sia stata consolidata
  - Presenza di meccanismi di retro-compatibilità: indicano l'attenzione del gruppo di lavoro a non lasciare indietro utenti con l'avanzare delle versioni
  - La maturità delle tecnologie su cui esse si basano
2. **Diffusione:** si valutano le implementazioni note delle specifiche e i prodotti che lo implementano, ma anche l'esistenza di gruppi di utenti e di alternative che potrebbero erodere il terreno, tramite i seguenti parametri:
  - Ampio uso delle specifiche presso vendor e fornitori
  - Uso delle specifiche in prodotto con ampia diffusione sul mercato
  - Esistenza di uno o più user group di supporto alle specifiche
  - Esistenza di altre specifiche sullo stesso dominio: se la concorrenza è bassa è maggiore lo spazio disponibile alla diffusione
  - Eventuali processi di armonizzazione in corso o esistenti con altre specifiche concorrenti
3. **Usabilità:** si valuta la disponibilità di documentazione di supporto, di strumenti per la sua implementazione, ma anche di gruppi di utenti che forniscano supporto e strumenti aggiuntivi.

I parametri da valutare sono i seguenti:

- Esistenza di linee guida e documentazione di supporto all'adozione delle specifiche
- Esistenza di tutorial o esempi per l'adozione delle specifiche
- Esistenza di un'implementazione di riferimento

- Esistenza di strumenti software di supporto all'adozione
- Esistenza di user group: questi possono fornire sia la possibilità supporto reciproco che sviluppare tool di uso comune).

4. **Gestione dell'Ambiguità**: si valutano i gradi di libertà possibili nell'applicare le specifiche, guardando anche alla chiarezza della documentazione e all'implementazione di meccanismi di test di conformità e interoperabilità. L'ambiguità infatti può far sì che implementazioni conformi alle specifiche, non risultino interoperabili fra loro. Si valuta guardando i seguenti parametri:

- Gestione esplicita dell'ambiguità: cioè esistono meccanismi per tenerne conto?
- Sufficiente livello di esplicitazione dei requisiti funzionali: questo limita la possibilità di comprensione erronea da parte di chi adotta le specifiche
- Esistenza di meccanismi e tool di test di conformità alle specifiche: laddove per conformità si intende l'adesione del documento scambiato rispetto alle specifiche
- Esistenza di meccanismi e tool di test di interoperabilità: per validare l'interoperabilità, oltre al processo bisogna validare il messaggio che lo trasporta e il processo, come avviene con i precedenti indici di interoperabilità
- Livello di completezza e dettaglio delle specifiche: questo limita i casi in cui le specifiche possano essere implementati con insufficiente livello di univocità

In proposito di come si sia realizzata la gestione dell'ambiguità senza perdere flessibilità, nelle specifiche, si veda (4)

5. **Sicurezza/Privacy**: si valuta l'implementazione di aspetti relativi a sicurezza e privacy. I parametri sono i seguenti:

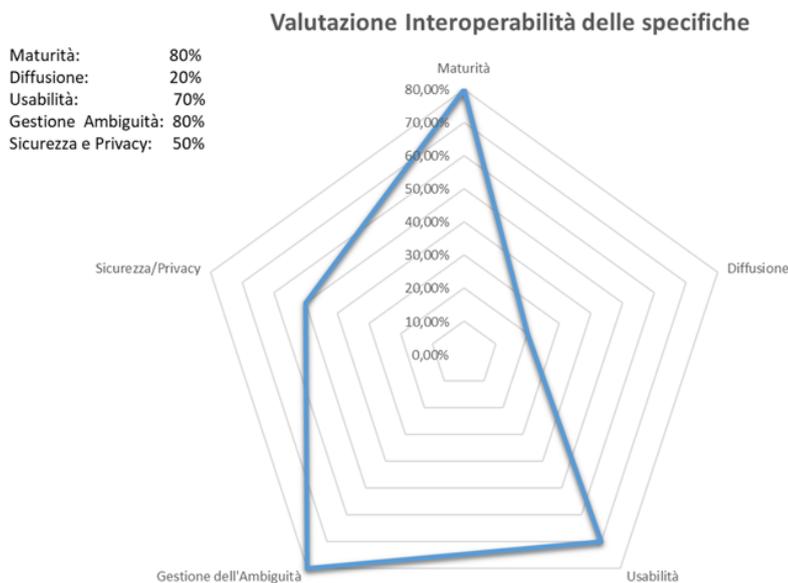
- Esistenza di *meccanismi per la sicurezza*
- Esistenza di *meccanismi per la privacy*
- Documento di *specifica dedicato alla sicurezza*
- Documento di *specifica dedicato alla privacy*

Il risultato di tale valutazione è mostrato analiticamente in Tabella 1 e sinteticamente in Figura 6.

**Tabella 1 - Valutazione dei parametri teorici di interoperabilità per le SCPS**

<b>Maturità</b>						
Parametri	Numero di release	Ultimo aggiornamento sito	Data inizio dello sviluppo	Esistono meccanismi di retrocompatibilità	Le tecnologie su cui si basa sono stabili e ben definite?	TOTALE
	1	dic-18	ott-15	sì	sì	
Scala	0% per una sola release 10% due release 20% > due release	20% se è passato meno di un anno 10% se è passato più di un anno e meno di tre 0% se sono passati più di tre anni	0% se è passato meno di un anno 10% se è passato più di un anno e meno di tre 20% se sono passati più di tre anni	20% in caso positivo 0% in caso negativo	20% in caso positivo 0% in caso negativo	Da 0% a 100%
Valutazione	0%	20%	20%	20%	20%	<b>80%</b>
<b>Diffusione</b>						
Parametri	Specifiche usate in diverse implementazioni presso vendor/supplier	I prodotti che usano le specifiche hanno una quota di mercato significativo	Specifiche supportate da uno o più user group	Numero di specifiche esistenti sullo stesso dominio	Processo di armonizzazione in corso con altre specifiche sullo stesso dominio	TOTALE
	poche implementazioni al momento	quota non significativa	nessuno	Una ( <i>all'interno di FIWARE</i> )	Avviato ( <i>NOTA: si sta guardando all'armonizzazione con alcune delle specifiche presenti in FIWARE</i> )	
Scala	0% nessuna /poche 10% numero significativo 20% larghissima diffusione	0% quota non significativa 10% quota significativa 20% quota maggioritaria	0% nessun user group 10% un user group 20% più di uno	0% diverse specifiche concorrenti 10% una specifica concorrente 20% nessun concorrente	0% nessuna armonizzazione 10% armonizzazione avviata/in corso 20% armonizzazione effettuata	Da 0% a 100%
Valutazione	0%	0%	0%	10%	10%	<b>20%</b>
<b>Usabilità</b>						
Parametri	Disponibilità di linee guida e documentazioni	Disponibilità di tutorial e/o esempi	Disponibilità di una implementazione di riferimento	Disponibilità di strumenti software per il supporto all'implementazione	Specifiche supportate da uno o più user group	TOTALE
	Sì	Sì	Sì (la SCP ENEA)	Sì	nessuno	
Scala	20% in caso positivo 0% in caso negativo	20% in caso positivo 10% solo esempi 0% in caso negativo	20% in caso positivo 0% in caso negativo	20% in caso positivo 0% in caso negativo	0% nessun user group 10% un user group 20% più di uno	Da 0% a 100%
Valutazione	20%	10%	20%	20%	0%	<b>70%</b>
<b>Gestione dell'Ambiguità</b>						
Parametri	L'ambiguità è esplicitamente affrontata?	Requisiti funzionali e non funzionali ben esplicitati	Esistono meccanismi e tool di test di conformità alle specifiche	Esistono meccanismi e tool di test di interoperabilità	Le specifiche hanno un buon livello di dettaglio, completezza e coerenza?	TOTALE
	Sì	Sì	Sì	Sì	Sì	
Scala	20% in caso positivo 0% in caso negativo	20% in caso positivo 0% in caso negativo	20% in caso positivo 0% in caso negativo	20% in caso positivo 0% in caso negativo	20% in caso positivo 0% in caso negativo	Da 0% a 100%
Valutazione	20%	20%	20%	20%	20%	<b>100%</b>

Sicurezza /Privacy					
Parametri	Sono previsti meccanismi di sicurezza	Sono previsti meccanismi per la privacy	C'è un documento di specifica esplicitamente dedicato alla sicurezza	C'è un documento di specifica esplicitamente dedicato alla privacy	TOTALE
	Sì	Sì	No	No	
Scala	25% sì 0% no	25% sì 0% no	25% sì 0% no	25% sì 0% no	Da 0% a 100%
Valutazione	25%	25%	0%	0%	<b>50%</b>



**Figura 6. Valutazione teorica delle specifiche SCPS**

Su tale valutazione si possono effettuare alcune osservazioni e commenti:

- Non è una sorpresa che il livello di **diffusione** sia basso: si tratta di specifiche appena nate, alla loro prima versione e al termine della loro fase di sperimentazione. Da questo però discende una considerazione sulla prosecuzione delle attività: dopo questa fase di implementazione delle specifiche avvenuta nell’ambito dei PAR di questo triennio, bisognerà lavorare alla diffusione dell’uso delle specifiche. A tal fine si stanno già individuando una serie di casi reali in cui iniziare a sperimentare le specifiche fuori dall’ambito di laboratorio.
- Il livello di **maturità** delle specifiche è già soddisfacente, grazie allo sviluppo effettuato in questi tre anni. Essendo una specifica giovane avrà bisogno di essere testata sul campo in modo più intensivo per correggere possibili imperfezioni che daranno luogo a nuove versioni. Dunque il processo di lavoro sulla diffusione, di cui si parlava nel punto precedente, permetterà di raggiungere anche un livello di maturità ottimale
- Dal punto di vista di **usabilità** e **gestione delle ambiguità**, questi requisiti sono stati quelli che hanno guidato il gruppo, sulla base sia dell’ampia esperienza sull’interoperabilità che dei requisiti individuati. Per tale motivo ci si può ritenere soddisfatti del lavoro effettuato
- L’ultimo parametro, infine, quello della sicurezza e della privacy è stato affrontato tramite l’uso di specifiche già esistenti (per esempio OAuth). Non essendo questo il core delle SCPS, ma un requisito aggiuntivo, ci si può ritenere soddisfatti del valore di tale parametro. Ciò non toglie che sulla base delle esperienze dei prossimi anni si valuterà la necessità effettiva di avere dei documenti aggiuntivi

esplicitamente dedicati a questi due requisiti, con conseguente incremento anche di quest'ultimo parametro.

## 2.5 Aggiornamento strato semantico

In collaborazione con l'università di Bologna è stato aggiornato il pool di applicazioni basate sull'ontologia. In

Figura 7 è mostrato lo stato finale di questo set di applicazioni software, evidenziando le modifiche rispetto alla versione rilasciata alla fine del PAR 2017.

In particolare si può notare che:

- I generatori automatici di Documentazione, Template e Schematron degli UrbanDataset sono stati aggiornati
- È stata implementata la nuova applicazione di validazione semantica e sintattica
- È stata aggiornata l'applicazione di conversione fra UrbanDataset in formato JSON e formato XML
- È stata aggiornata l'applicazione Web di navigazione dell'ontologia.

Per i dettagli si vedano l'Appendice (Sintesi dell'attività dell'Università di Bologna) e il report RdS/PAR2018/021.

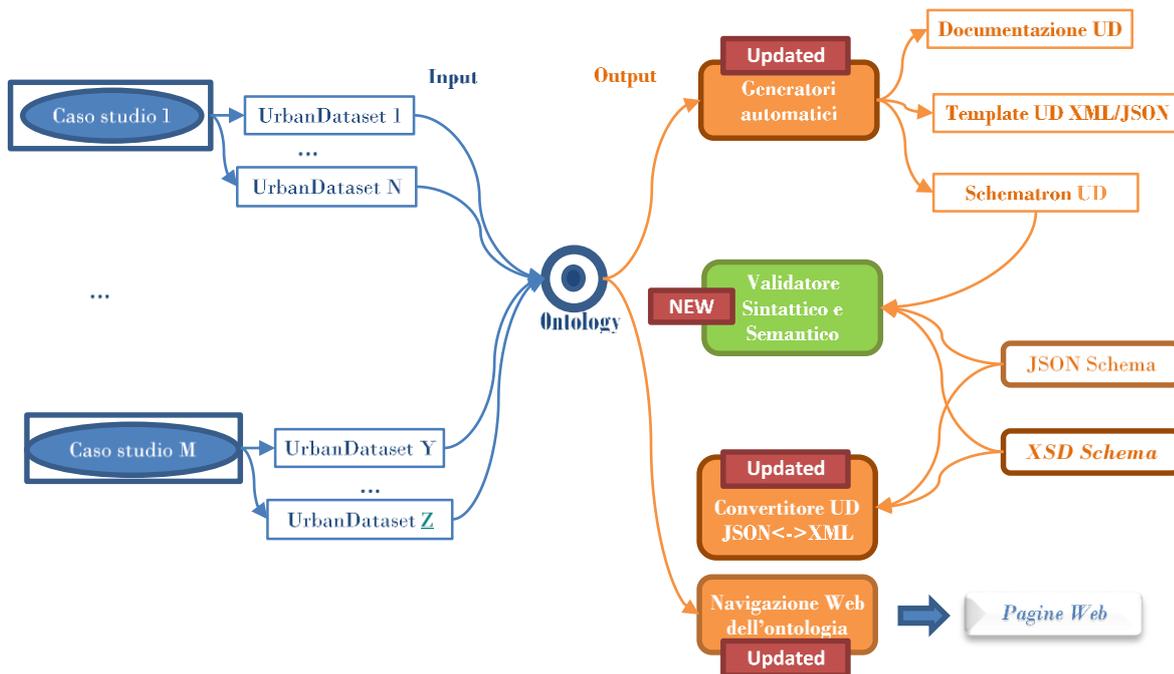


Figura 7. Strumenti SCPS Ontology-based

### 3 Processo di reingegnerizzazione del software e miglioramento del livello di sicurezza (D7a2)

Durante i tre mesi del PAR2018 sono state eseguite sessioni di test sui vari moduli che costituiscono il Prototipo della Smart City Platform (SCP) al fine di validare l'adesione alle specifiche e correggere, attraverso una procedura di bug fixing, eventuali fenomeni inaspettati. Inoltre, è stata portata avanti l'attività di sviluppo software per migliorare, attraverso un processo di reingegnerizzazione che ha coinvolto sia il codice che l'architettura, la replicabilità della Smart City Platform e la garanzia d'interoperabilità della stessa in diversi contesti applicativi.

Al termine del PAR2017 l'architettura software è stata progettata e implementata nel prototipo SCP com'è descritto nella rappresentazione schematica dell'organizzazione delle macchine qui di seguito riportata. Si noti che tale schema non è parte delle SCPS; ciò significa che potrebbero esistere altri prototipi SCPS-based, e quindi interoperabili, ma con un'organizzazione interna completamente differente.

In altre parole, lo schema architetturale in Figura 8 della SCP è solo una delle possibili implementazioni che si può ottenere partendo dalle specifiche SCPS. Infatti, organizzando diversamente le macchine virtuali e distribuendo su di esse le diverse componenti funzionali (ma sempre rispettando le specifiche) si ottiene una SCP interoperabile con le altre che implementano le stesse specifiche. Ciò è avvenuto, ad esempio, nel corso del PAR2018, durante il quale sono state eseguite dell'operazione quali: lo spostamento di alcuni componenti, la duplicazione di alcune Macchine Virtuali e l'introduzione di una nuova tipologia di database: tali modifiche non hanno avuto ripercussioni sull'interoperabilità della SCP e ciò è dimostrato dal fatto che non è stato necessario effettuare alcuna modifica alle applicazioni verticali che inviano dati verso di essa. Queste modifiche della versione precedente della SCP sono illustrate in Figura 9. Nel seguito verranno descritte con più dettaglio le attività svolte.

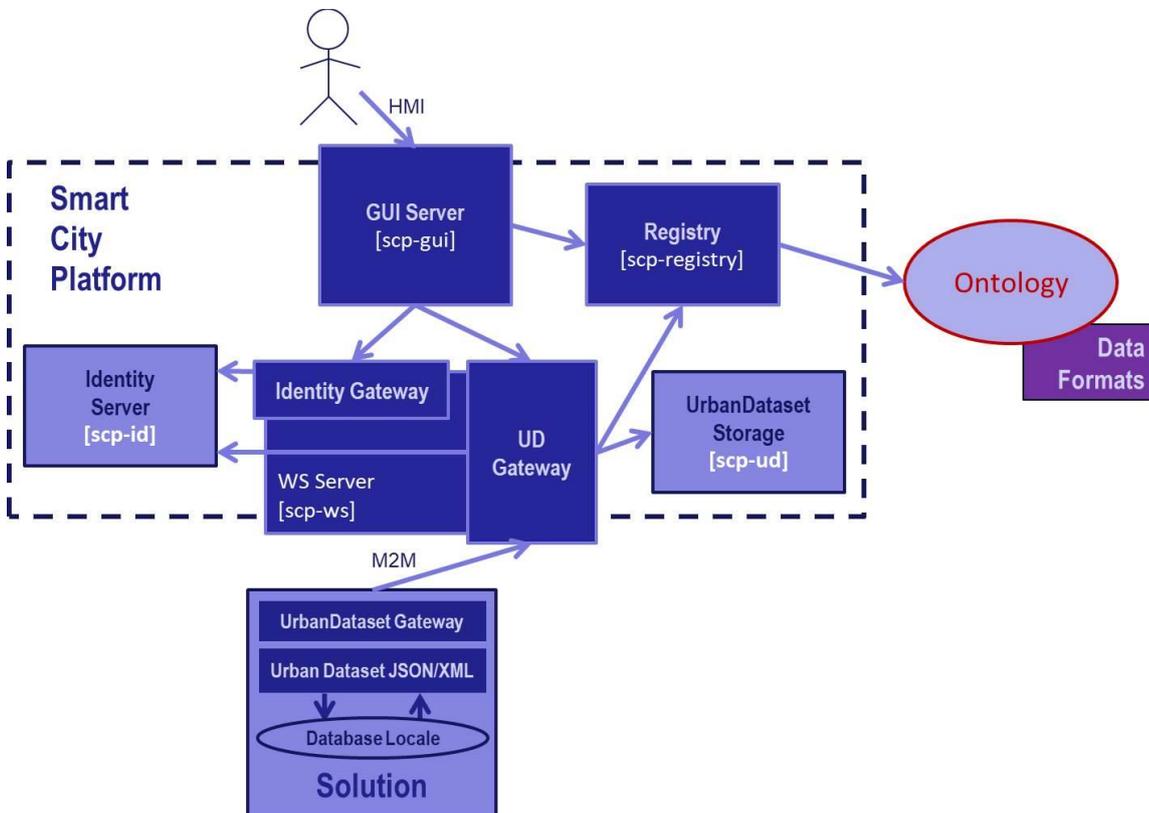


Figura 8. SCP - Architettura Software (PAR2017)

Una Smart City Platform (o Piattaforma ICT per Distretto/Città) deve permettere due interazioni:

- **Human-Machine Interface (HMI)**: è l'interazione umana tramite interfaccia utente, che nell'ambito di una Piattaforma ICT per Smart City serve a permettere la configurazione delle collaborazioni, nonché la possibilità di verificare il traffico dati; questa interazione è definita nelle specifiche "SCPS Collaboration Level" dove gli utenti fondamentali definiti sono: *Developer, Administrator, Solution Manager, Cittadino*;
- **Machine To Machine (M2M)**: è l'interazione automatica tra sistemi, nel nostro particolare caso è l'interazione tra Solution verticali produttrici di dati dalla città verso la Smart City Platform e, viceversa, l'interazione tra Smart City Platform verso le Solution verticali fruitrici di dati per la città.

Rispettivamente, queste due interazioni sono rappresentate nella parte alta e bassa dello schema nella figura precedente. Inoltre, nella stessa figura si nota che, internamente alla SCP, vi sono i seguenti **componenti** implementati in altrettante **macchine virtuali [VM]**:

- GUI [*scp-gui*]
- Registry [*scp-registry*]
- UrbanDataset Gateway [*scp-ws*]
- Identity Server [*scp-id*]
- UrbanDataset Storage [*scp-ud*]

Nel corso del PAR2018 si è modificata e aggiornata l'architettura software per aumentare il livello di sicurezza e rendere la SCP più robusta.

In particolare si sono portate a termine quattro importanti sotto-attività di seguito elencate.

1. Reingegnerizzazione del software per facilitare la replicabilità;
2. Spostamento dell'*IdentityGateway* dalla VM WS Server alla VM IdentityServer;
3. Duplicazione del WS Server;
4. Adozione di ElasticSearch.

Nel seguito, le attività sopra riportate saranno oggetto di approfondimento.

## 2.6 Reingegnerizzazione del sw

Al fine di consolidare il sistema di replicabilità della SCP sono state adottate solo soluzioni open source e i componenti sono state sviluppati in maniera tale da poterli reinstallare su nuovo pool di macchine virtuali.

In particolare sulle diverse VM sono state parametrizzate quelle parti del software che dipendono dalla specifica installazione (ad es. il nome delle macchine virtuali o particolari credenziali di accesso ai vari sistemi) e salvare questi parametri in opportuni file di configurazione.

In questo modo la replica dell'intera SCP (ma anche dei singoli componenti) si traduce in una semplice clonazione della SCP (o del componente) ed il successivo aggiornamento dei relativi parametri nei file di configurazione.

A tal proposito, nell'implementazione del componente *UrbanDatasetGateway* sulla macchina dedicata ai servizi web (*scp-ws*) si è previsto l'uso di un web server open source della famiglia *Apache Tomcat* (versione 8.5) e sviluppato lo strato software (i servizi) utilizzando la tecnologia JAVA.

I servizi presenti sull'*UrbanDatasetGateway* si possono suddividere in due categorie: in una sono raggruppati tutti i servizi che gestiscono l'accesso degli utenti (*login, isAlive, logout*) e nell'altra quelli che propriamente gestiscono le produzioni e gli accessi degli *UrbanDataset* (*push, basicRequest, specificRequest, searchingRequest, deepSearchingRequest*)

Il processo di reingegnerizzazione di questo componente ha comportato, ad esempio, il trasferimento di tutte quella parte di codice JAVA che memorizzava le costanti della SCP (non a caso la classe JAVA nella quale esse erano definite si chiama *Constants*) in un opportuno file di configurazione esterno al codice ma accessibile in lettura solo da quest'ultimo.

Un processo di reingegnerizzazione del tutto analogo a quello appena descritto è stato svolto per il componente *IdentityGateway*.

Inoltre, in un'ottica di generalizzare la capacità della SCP di utilizzare diversi sistemi di archiviazione dei dati, è stato eseguito sull'*UrbanDatasetGateway* un ulteriore processo di reingegnerizzazione che ha visto la creazione di un layer intermedio tra la parte di codice nella quale sono definiti i servizi che si occupano della produzione e degli accessi degli *UrbanDataset* e quella parte di codice che interagisce propriamente con il server database (*MySql*) dove essi sono archiviati.

In generale, il sistema modulare del codice JAVA sviluppato e, in particolare, la generazione di questo wrapper intermedio, ha permesso, come verrà spiegato in seguito, l'aggancio di un ulteriore sistema di database (*ElasticSearch*) senza notevoli sforzi se non quello di scrivere, come era stato fatto in precedenza con *MySql*, i nuovi metodi di comunicazione con il nuovo DB *ElasticSearch* attraverso le API che vengono fornite assieme ad esso.

## 2.7 Spostamento *IdentityGateway*

Con lo scopo di rendere il sistema di gestione delle utenze più sicuro e indipendente dalla scelta del particolare prodotto software installato, è stata creata un'interfaccia intermedia tra il *Identity Provider* e l'utente, nella interazione HMI. In particolare, questa interfaccia che prende il nome di *IdentityGateway* si compone di metodi che si espongono come interazione esclusiva tra l'*Identity Server* e il *GUI Server*. L'*IdentityGateway* raccoglie le richieste degli utenti e, in funzione anche del ruolo dell'utente stesso, interroga il *Identity Provider* attraverso opportune chiamate.

Avendo previsto una parametrizzazione e un confinamento delle parti variabili della SCP in opportuni file di configurazioni, la replica e quindi lo spostamento di uno o più componenti è risultato abbastanza agevole e indolore. Un esempio, che è nato dall'esigenza di aumentare la sicurezza e accorpate le parti più simili tra loro, è stato quello di spostare il componente *IdentityGateway* dalla macchina *scp-ws* alla macchina *scp-id*.

I servizi esposti dall'*IdentityGateway* centralizzano i metodi per la gestione delle identità degli utenti sulla piattaforma ed in particolare permettono sia interventi da *admin* come:

- la registrazione di una nuova utenza (*signup*);
- la cancellazione di un'utenza (*delete*);
- l'aggiornamento della password (*updatePassword*);

sia operazioni da semplice *user* come:

- l'autenticazione tramite username e password con il rilascio di un JWT-Token (*login*);
- la verifica della validità del JWT-Token (*isAlive*);
- il rilascio del JWT-Token prima della sua prevista scadenza temporale (*logout*).

È sembrato quindi del tutto naturale eseguire questo spostamento, per mezzo del quale si sono separate, non solo logicamente ma anche fisicamente, le parti della SCP che gestiscono gli accessi alla piattaforma (vale a dire l'*Identity Server* che era già installato sulla *scp-id* e adesso anche l'*IdentityGateway*) dal resto delle componenti funzionali.

## 2.8 Duplicazione del *WS Server*

Inizialmente, la Smart City Platform ha utilizzato per le interazioni M2M e HMI il servizio *UrbanDatasetGateway* implementato sulla VM *scp-ws*. Questo web service permette alle Solution verticali di inviare (e accedere a) gli *UrbanDataset* della SCP, tramite un gateway la cui interfaccia di comunicazione rispetta le specifiche *SCPS Communication Level*, per garantire piena interoperabilità.

La Smart City Platform provvede l'interazione M2M attraverso il servizio *UrbanDatasetGateway* implementato sulla VM *scp-ws*. Questo web service permette alle Solution verticali di inviare (e accedere a) gli UrbanDataset della SCP, tramite un gateway la cui interfaccia di comunicazione rispetta le specifiche *SCPS Communication Level*, per garantire piena interoperabilità.

L'implementazione "interna" del *UrbanDatasetGateway* richiede venga effettuata una serie di controlli e verifiche quando il servizio prende in carico una richiesta. A titolo di esempio, riportiamo di seguito un elenco dei principali controlli che vengono effettuati per eseguire l'inserimento di un *UrbanDataset* inviato da una Solution, tramite il metodo *push*, sul prototipo SCP.

#### Metodo *push*:

Questo metodo

- riceve in input 3 parametri: *token*, *resource\_id*, *dataset*;
- e restituisce in output un JSON contenente: *code*, *message*.

A livello interno della Vm *scp-ws* avvengono le seguenti operazioni:

- 1 Viene inviata una richiesta di verifica della validità del *token* ricevuto al server *scp-id* che gestisce le utenze;
- 2 Se la richiesta di autenticazione non va a buon fine viene restituito un codice e un messaggio di autenticazione fallita (ad es., code: "12", message: "Token Invalid");
- 3 Se la richiesta di autenticazione va a buon fine, viene estratta la *username* relativa al *token*;
- 4 Si verifica la coerenza del *resource\_id* ricevuto in input, ovvero si controlla che la collaborazione Solution-SCP relativa sia abilitata nel *Registry* (database sulla VM *scp-registry*) per produrre quel particolare Urban Dataset.
  - 4.1 Se la verifica non va a buon fine viene restituito un codice e un messaggio di collaborazione non valida (ad es., code: "20", message: "Unknow Collaboration");
  - 4.2 Se la verifica va a buon fine si recupera dal *Registry* la modalità di persistenza prevista per quella particolare collaborazione (ad es., append, overwritten) e si passa a convalidare e a persistere il parametro *dataset*;
    - 4.2.1 Anzitutto si verifica che la dimensione del *dataset* non sia superiore ad una certa soglia;
    - 4.2.2 Se la verifica non va a buon fine si restituisce un codice e un messaggio di errore (ad es., code: "35", message: "UrbanDataset too long");
    - 4.2.3 Altimenti si passa a validare la struttura dell'UrbanDataset secondo lo schema previsto;
    - 4.2.4 Se la validazione non va a buon fine viene restituito un codice e un messaggio di dataset non valido secondo il formato JSON (ad es., code: "31", message: "UrbanDataset Invalid against JSON Schema");
    - 4.2.5 Se la validazione da esito positivo si passa alla persistenza dell'UrbanDataset;
      - 4.2.5.1 Segue scrittura sul database dell'UrbanDataset seconda modalità di persistenza prevista e si fa restituire il numero di linee inserite e il timestamp del context;
      - 4.2.5.2 Si aggiunge un nuovo record alla tabella *history\_production* del *Registry* con il numero di linee e il timestamp appena ottenuti;
      - 4.2.5.3 Viene inviato un codice e un messaggio di successo (code: "02", message: "Push Successful").

Come detto, l'*UrbanDatasetGateway* è usato sia nell'interazione M2M di scambio UD tra piattaforme (Solutions e SCP) sia nell'interazione HMI (umana) per recuperare gli UD via interfaccia web.

L'aggiornamento dell'architettura software ha comportato, oltre allo spostamento dell'*IdentityGateway* descritto in precedenza, anche la duplicazione del WS Server in modo tale da ottenere:

- *UrbanDatasetGateway* “privato” [*scp-ws*]
- *UrbanDatasetGateway* “pubblico” [*scp-ws-broker*]

In altre parole, la macchina virtuale *scp-ws* è stata clonata per ottenere infine due VM identiche *scp-ws* e *scp-ws-broker*, rispettivamente privata e pubblica.

La clonazione della macchina è stata agevolata dalla reingegnerizzazione del software descritta in precedenza. Infatti, la scelta di parametrizzare e convogliare le costanti che distinguono i due *UrbanDatasetGateway* in singoli file esterni al componente (e non più integrati nel codice sorgente di deployment) ha permesso di eseguire facilmente una copia della VM seguita poi da una semplice sostituzione del file contenente la nuova configurazione.

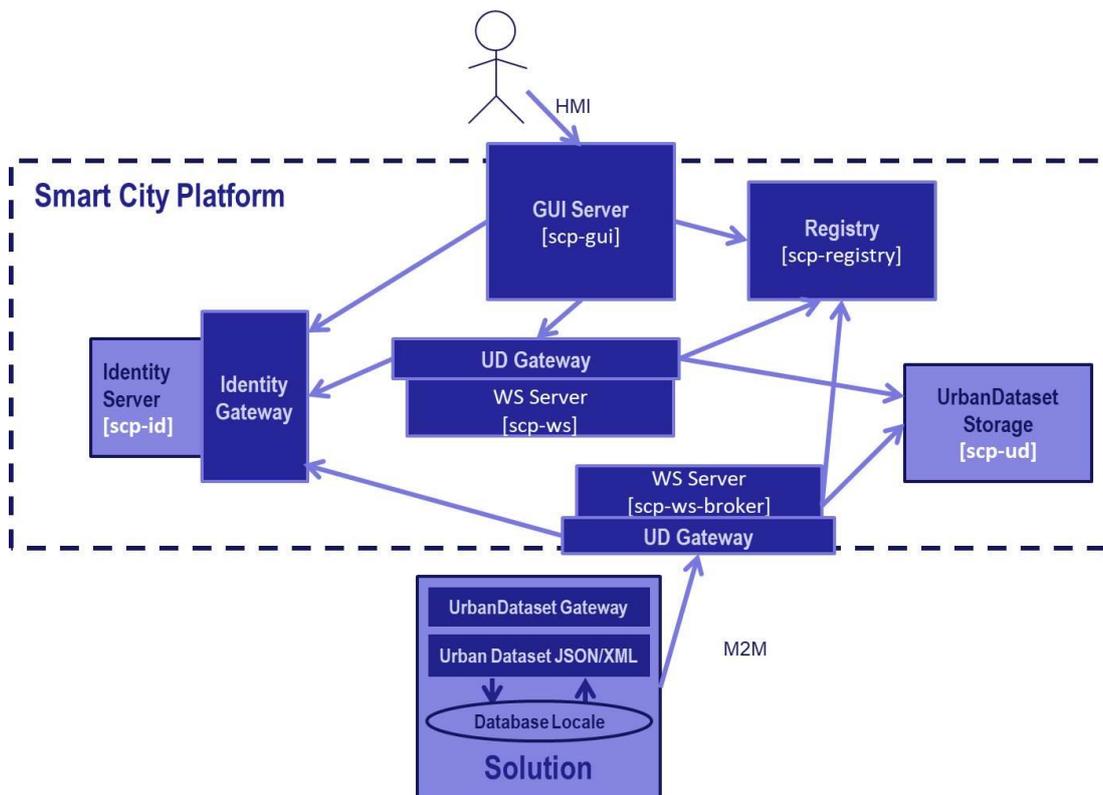


Figura 9. SCP - Architettura Software (PAR2018)

La duplicazione dell’*UrbanDatasetGateway* su due macchine distinte ha permesso di separare fisicamente i due canali di comunicazione (M2M e HMI) che la SCP fornisce per interagire con il database dei *UrbanDataset*. Inoltre, con questa nuova organizzazione dei componenti, nel prototipo SCP-ENEA le VM *scp-ws* e *scp-ws-broker* essendo distinte, rendono in generale l’architettura più robusta e sicura. Ciò che potrebbe accadere, infatti, è che, in seguito a un attacco malevolo al gateway pubblico predisposto per l’interazione M2M, l’intera piattaforma venga compromessa (compreso l’accesso tramite interfaccia web dell’interazione HMI). La clonazione della macchina (*scp-ws*) e la pubblicazione di una sola delle due (ovvero il broker) garantisce che, anche in caso di attacco all’access point pubblico, sia mantenuto il funzionamento per l’accesso tramite GUI (così come si evince dallo schema in figura seguente).

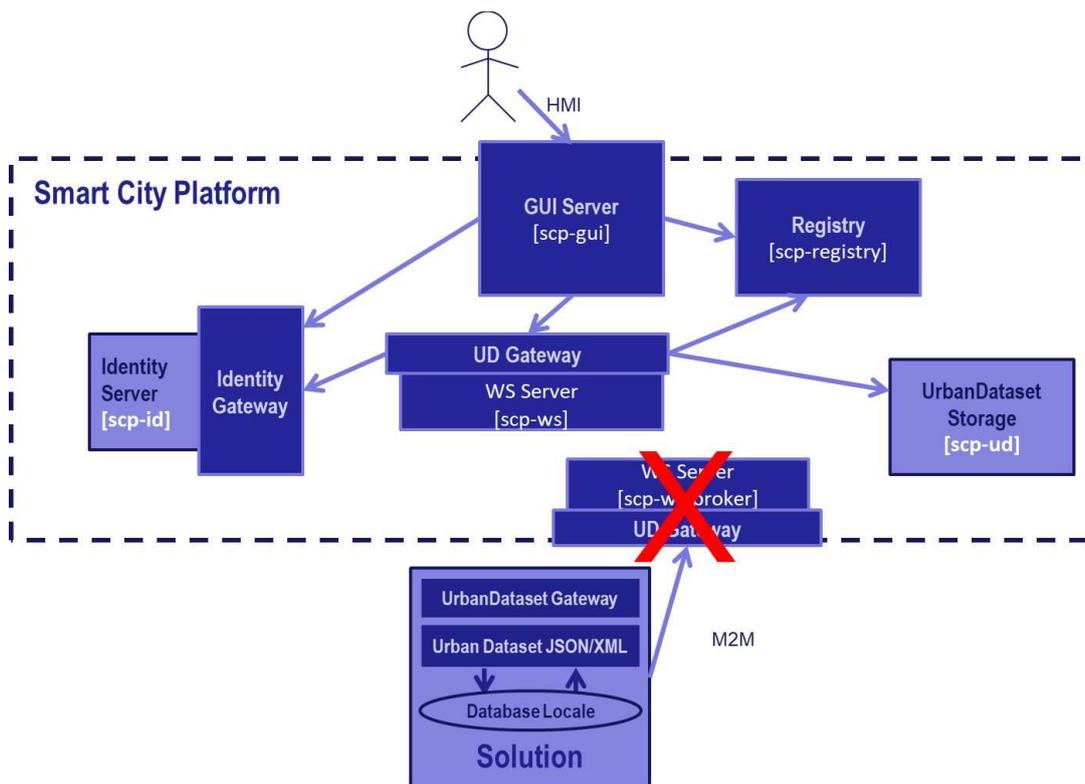


Figura 10. SCP – Ipotesi di attacco al Gateway pubblico

## 2.9 Adozione di ElasticSearch

In generale, una piattaforma Smart City Platform si basa sulle specifiche SCPS che garantiscono la piena interoperabilità nella comunicazione con le Solution verticali della Smart City.

Tra le funzionalità fondamentali (livello Functional) previste dalle specifiche SCPS vi è la persistenza degli UD a carico dell’“UrbanDataset Database”. Non vengono fornite linee guida particolari dalle specifiche SCPS per effettuare questa implementazione che è lasciata libera, a chi volesse implementare una SCP, in quanto non interferisce con l’interoperabilità della Piattaforma ICT.

Il prototipo *SCP-ENEA* è stato sviluppato in origine utilizzando un sistema di persistenza dei dati di tipo **relazionale** (*MySQL*). La piattaforma, in questo caso, riceve un *UrbanDataset* in ingresso e lo trasferisce, dopo un’opportuna conversione (prima in un oggetto JAVA e poi in dato strutturato tabellare) al database *MySQL*. Le stesse trasformazioni, nell’ordine inverso, avvengono in fase di recupero di un *UrbanDataset* dal database, al fine di inviarlo alla *Solution* che ne ha fatto richiesta. Questa procedura, che in fase di sperimentazione ha fornito buoni risultati in termini di efficienza e robustezza di persistenza, è stata affiancata da una nuova tecnologia di archiviazione dei dati basata su un sistema di tipo **non relazionale** (si vedano in tal proposito, le valutazioni effettuate nella tabella 5 del Report RdS/PAR2017/040).

La scelta di utilizzare una nuova base di dati di tipo non relazionale è ricaduta sul prodotto *ElasticSearch* (per brevità *ES*) e per i motivi di seguito descritti. Questo tipo di database si basa su un meccanismo di acquisizione dei dati di tipo documentale ed è molto performante sulla gestione di grosse moli di dati per via della possibilità di replica del dato su diversi nodi (clustering). Inoltre *ES* riconosce nativamente la struttura dati di tipo JSON (qual è appunto l’*UrbanDataset* quando viene inviato su un canale RESTful), garantendo un sistema di mapping e di riconoscimento dei principali tipi di dato primitivi (ad es. string, number, timestamp e coordinates) delle coppie chiave-valore e fornendo funzioni di gestione e ricerca molto veloci.

Attualmente, per una prima fase di testing e sperimentazione, la versione 6.4 di *ElasticSearch* è stata installata e configurata sul nodo *scp-ws*. La configurazione è stata eseguita in maniera tale da indicizzare con

un'opportuna tecnica tutti gli *UrbanDataset* in input e rendere la loro ricerca temporale e georeferenziata molto performante.

In particolare, in fase di inserimento (metodo *push*) di un nuovo *UrbanDataset* (UD) su *ElasticSearch*, si genera un identificativo univoco (*specific\_ud\_id*) composto dalla concatenazione della stringa *collaboration\_id* (che si utilizza per verificare che la solution produttrice del dato sia autorizzata all'invio) e la stringa *timestamp* di produzione estratta all'interno del documento stesso. Mappando i valori che la chiave *specific\_ud\_id* assume con un tipo di dato *keyword* allora è possibile applicare delle espressioni regolari (*regexp*) che facilitano la ricerca. Un esempio di *specific\_ud\_id* è il seguente:

"SCP-1\_SmartBuildingCasaccia-3\_BuildingElectricConsumption-1.0\_20180125120000\_2017-01-01T00:00:00".

A titolo di esempio, sfruttando la wildcard "\*" è possibile effettuare ricerche per indice sulla collezione di tutti gli UD prodotti:

- per la SmartCityPlatform "SCP-1" ("SCP-1\_\*");
- per la SmartCityPlatform "SCP-1" dalla Solution "SmartBuildingCasaccia-3" (SCP-1\_SmartBuildingCasaccia\_\*);
- per la SmartCityPlatform "SCP-1" dalla Solution "SmartBuildingCasaccia-3" nell'ambito della tipologia "BuildingElectricConsumption" (SCP-1\_SmartBuildingCasaccia\_BuildingElectricConsumption\_\*);
- per la SmartCityPlatform "SCP-1" dalla Solution "SmartBuildingCasaccia-3" nell'ambito della tipologia "BuildingElectricConsumption" che prevedono un tempo di inizio della collaborazione dal "2018-01-25T12:00:00" (SCP-1\_SmartBuildingCasaccia\_BuildingElectricConsumption\_20180125120000\_\*);
- per la SmartCityPlatform "SCP-1" dalla Solution "SmartBuildingCasaccia-3" nell'ambito della tipologia "BuildingElectricConsumption" che prevedono un tempo di inizio della collaborazione dal "2018-01-25T12:00:00" e con timestamp di produzione corrispondente a "2019-01-01T00:00:00" (SCP-1\_SmartBuildingCasaccia\_BuildingElectricConsumption\_20180125120000\_2017-08-08T00:00:00);

Assieme allo *specific\_ud\_id* sono state generate altre due chiavi: una utilizzata per le ricerche su range temporali (*specific\_ud\_timestamp*) e una utilizzata per le ricerche su domini spaziali (*specific\_ud\_location*). Questi due campi, *specific\_ud\_timestamp* e *specific\_ud\_location*, acquisiscono, rispettivamente, il valore *timestamp* e il valore *coordinates* definiti all'interno della sezione *context* dell'*UrbanDataset*.

Sviluppando sull'*UrbanDatasetGateway* una opportuna classe JAVA che utilizza le API fornite da *ES* per interrogare il server DB e tenendo presente la particolare tecnica di indicizzazione appena descritta, è stato naturale considerare la ricerca per *specific\_ud\_id* come base di partenza per implementare i metodi *specificRequest* e *basicRequest*, analogamente la ricerca combinata per *specific\_ud\_timestamp* e *specific\_ud\_location* è stata invece utilizzata per definire il metodo *searchingRequest*.

## 5 Conclusioni

Con il PAR2018 si è concluso l'obiettivo D7a che prevedeva, la definizione di un set di specifiche (SCPS) e la progettazione e l'implementazione di una Piattaforma ICT (SmartCityPlatform, SCP) per la gestione di Smart District: risultati raggiunti e pubblicati in rete.

La piattaforma ICT è stata testata con dati reali provenienti dalle Solutions verticali del centro ENEA Casaccia (Smart Village) che ben approssima un distretto reale (sebbene rimanga un ambiente sperimentale controllato).

Al termine di questo trimestre inoltre è stato possibile:

- Definire KPI relativi alle specifiche e definirne un sottoinsieme implementabile nella piattaforma software. L'analisi di questi KPI ha portato a delle valutazioni positive del lavoro fatto e a delle criticità (legate soprattutto alla diffusione delle specifiche) che indicano la strada da fare per portare avanti e valorizzare i risultati raggiunti
- Completare il set di strumenti incentrati sull'ontologia
- Rinforzare l'architettura dell'implementazione in modo da renderla più sicura e più stabile

Nonostante il termine del triennio del progetto, la sperimentazione del prototipo continuerà anche in futuro, andando a integrare altre Solution verticali che il centro ENEA Casaccia avrà via via a disposizione, in modo da divenirne strumento centralizzato per la gestione energetica.

Un futuro trasferimento tecnologico del prototipo SCP in un contesto reale, invece, garantirebbe sia un consolidamento delle specifiche SCPS (che sono nate per essere utilizzate nei bandi di gara delle municipalità ma non hanno ancora una dimostrazione su contesto urbano reale) e sia un'evoluzione del prototipo stesso verso quello che sarebbe sua naturale destinazione: la gestione energetica di città/distretti tramite un approccio basato sull'interoperabilità.

Alla dimostrata replicabilità di Solution dovrebbe conseguire la replicabilità di SCP in diversi contesti urbani: si ritiene che la diffusione delle specifiche da una parte, un utilizzo della SCP-ENEA su territorio nazionale e la possibilità di renderla interoperabile con altre piattaforme per smart city, anche in contesti internazionali (Horizon2020 insiste sull'uso di specifiche condivise), dall'altra aprirebbro nuovi scenari di studio e sviluppo.

In conclusione, sebbene l'obiettivo D7a abbia raggiunto i risultati pianificati, in un ambiente sperimentale quale è lo Smart Village Casaccia, le potenzialità delle specifiche SCPS e del prototipo SCP sono molto alte e per questo meritevoli di un ulteriore investimento di risorse per poter essere trasferite e dimostrate in contesti urbani reali in diretta collaborazione con le municipalità.

## 6 Riferimenti bibliografici

1. A. Frascella, J. Swiderski, G. Proserpio, E. Rikos, A. Temiz, A. Babs, M. Uslar, S. Branchetti, G. Graditi, Looking for the unified classification and evaluation approach of SG interface standards for the purposes of ELECTRA IRP. USB Proceedings 2015 International Symposium on Smart Electric Distribution System and Tech, Vienna 2015, p. 318-323.
2. CEN-CENELEC-ETSI Smart Grid Coordination Group. Smart Grid Reference Architecture, 2012.
3. European Commission. CAMSS: Common Assessment Method for Standards and. 2010.
4. A. Brutti, A. Frascella, N. Gessa, P. De Sabbata, C. Novelli, Interoperability in the Smart City: A Semantic Approach for Merging Flexibility with Strictness, Proceedings 2018 IEEE International Conference on Smart Computing, Taormina 2018.

## 7 Abbreviazioni ed acronimi

Lista degli acronimi usati nel testo

API	Application Programming Interface
CAMSS	Common Assessment Method for Standards and Specifications
DB	Database
ES	Elastic Search
GUI	Graphical User Interface
HMI	Human Machine Interaction
ICT	Information and Communication Technology
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
M2M	Machine to Machine
SCP	Smart City Platform
SCPS	Smart City Platform Specification
SGAM	Smart Grid Architecture Model
UD	UrbanDataset
VM	Virtual Machine
WS	Web Service
XML	eXtensible Markup Language

## Appendice – Sintesi dell'attività dell'Università di Bologna

Il lavoro svolto durante il periodo di estensione dal Dipartimento di Informatica – Scienza e Ingegneria (DISI) dell'università di Bologna consiste nella progettazione e sviluppo di nuovi artefatti software costruiti intorno all'ontologia degli UrbanDataset e miglioramento di quelli già definiti durante il periodo di progetto precedente, con lo scopo di concludere il lavoro e fornire una serie di strumenti funzionanti e perfezionati sulla base delle esigenze che sono apparse sempre più chiare durante la fase di realizzazione precedente.

Il primo intervento effettuato è stato un'analisi sull'operazione di generazione dei template e trasformazione tra JSON e XML dei documenti contenenti la definizione degli UrbanDataset al fine di rendere più facile l'adattamento del codice sviluppato in seguito ad eventuali modifiche dei formati XML e JSON. In precedenza, non si era tenuto conto della possibilità di effettuare delle variazioni sui formati XML e JSON; conseguentemente a questa valutazione, è stato ritenuto opportuno modificare l'applicazione in modo da minimizzare il più possibile le modifiche da dover apportare al codice in caso di modifiche ai formati.

In questa nuova versione, quindi, una buona parte del codice viene generata automaticamente a partire dai file di schema XSD e JSONSchema, e solo sulla restante parte è necessario fare dei piccoli aggiustamenti spesso guidati dagli strumenti di sviluppo che segnalano eventuali errori. Inoltre, oltre ad aggiornare l'applicazione, è stata definita la procedura necessaria, in caso di modifiche alla versione degli XML schema delle specifiche, per tenere costantemente aggiornata questa applicazione con le ultime versioni.

Oltre a ciò, è stata progettata e sviluppata una nuova applicazione che offre la possibilità di validare un file contenente un UrbanDataset con i file di schema. Nello specifico, l'applicazione valida i file in formato XML rispetto allo schema XSD e allo Schematron (SCH) relativo allo specifico UrbanDataset, mentre valida rispetto al JSONSchema i file in formato JSON.

Questo sistema è stato anche integrato nell'applicazione per la trasformazione tra formati XML e JSON. Durante una trasformazione tra XML e JSON (e viceversa) vengono quindi eseguite tutte le validazioni necessarie per verificare la correttezza dei documenti da trasformare.

L'applicazione per la validazione può supportare tutte le versioni degli schemi XSD e JSON sviluppate nel tempo, per dare modo agli utenti di validare sempre i propri documenti e set di dati rispetto alle versioni che stanno utilizzando. L'applicazione è pesantemente basata su file di configurazione molto dettagliati, approccio che favorisce una semplice manutenzione del codice in caso di modifiche agli schema.

Durante questo periodo sono stati anche migliorati i messaggi che le applicazioni generano e mostrano all'utente durante l'esecuzione. In particolare, sono stati gestiti tutta una serie di errori in modo da mostrare in maniera chiara all'utente i motivi di questi errori, e facilitarlo nella comprensione degli errori stessi e nell'individuazione delle azioni necessarie per correggerli e utilizzare correttamente gli applicativi. La gestione degli errori è stata fatta a valle di una fase intensa di testing e validazione delle applicazioni, per cercare di fare emergere il più possibile tutte le problematiche irrisolte e le lacune nella gestione di problemi come: file di configurazione errati, file in input non validi o mancanti, ecc.

Infine, sulla pagina web di navigazione dell'ontologia sono state integrate le librerie per la validazione e per la trasformazione dei formati dei documenti contenenti UrbanDataset. In questo modo è possibile per gli utenti del sito e per gli utilizzatori dei formati degli UrbanDataset, tramite upload del file da trasformare, ottenere la traduzione del formato del documento sia da XML verso JSON che viceversa. Oltre a questo, è stato integrato un sistema per la generazione di documentazione sulle informazioni dell'ontologia basato su trasformazioni XSL ed è stata effettuata una riorganizzazione di alcune pagine. Nel sito web è stata fatta un'analisi per la organizzazione delle informazioni all'interno delle form web, andando a migliorare e perfezionare la parte grafica e la presentazione delle informazioni. Le stesse pagine web create sono state verificate e sono stati corretti vari errori che si sono riscontrati nella fase di test.