



Ricerca di Sistema elettrico

## Piattaforma ICT per la gestione dello Smart District

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

## PIATTAFORMA ICT PER LA GESTIONE DELLO SMART DISTRICT

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

Settembre 2017

### Report Ricerca di Sistema Elettrico

Accordo di Programma Ministero dello Sviluppo Economico - ENEA

Piano Annuale di Realizzazione 2016

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

Progetto: D.6 Sviluppo di un modello integrato di smart district urbano

Obiettivo:

Responsabile del Progetto: Claudia Meloni, ENEA

## Indice

SOMMARIO.....	7
INTRODUZIONE.....	9
1 DEFINIZIONE SPECIFICHE E ARCHITETTURA DI RIFERIMENTO (D6A.1) .....	11
1.1 URBAN DATASET E LIVELLI DI AGGREGAZIONE.....	13
1.2 CASI STUDIO INTER-SOLUTION.....	14
1.2.1 <i>Case Study “SmartBuilding to WebGIS”</i> .....	16
1.2.1.1 Collaborazione.....	17
1.2.1.2 Urban Dataset coinvolti.....	20
1.2.1.3 Urban Dataset “Smart Building Records” .....	20
1.2.1.4 Urban Dataset “Smart Building Anomalies” .....	21
1.2.1.5 Urban Dataset “Smart Building Average Consumption” .....	23
1.2.1.6 Formati .....	24
1.2.1.7 Trasporto .....	25
1.2.1.8 Tabella sintetica.....	26
1.2.2 <i>Case Study “DSS to Smart Building”</i> .....	28
1.2.2.1 Urban Dataset coinvolti.....	28
1.2.2.2 Tabella sintetica.....	31
1.2.3 <i>Case Study “WWTP to WebGIS”</i> .....	33
1.2.3.1 Urban Dataset Coinvolti.....	34
1.2.3.2 Tabella sintetica.....	36
1.3 CASI STUDIO “ONE WAY” .....	38
1.3.1 <i>Case Study “Smart Lighting to PELL”</i> .....	38
1.3.1.1 Urban Dataset Coinvolti.....	38
1.3.1.2 Tabella sintetica per la descrizione del caso studio .....	41
1.4 CASI STUDIO PARTICOLARI .....	43
1.4.1 <i>Case Study “Whatever”</i> .....	43
1.4.1.1 Urban Dataset Coinvolti.....	44
1.4.1.2 Tabella sintetica per la descrizione del caso studio .....	45
1.4.2 <i>Case Study “Direct Communication”</i> .....	46
1.4.2.1 Urban Dataset Coinvolti.....	47
1.4.2.2 Tabella sintetica.....	48
1.5 DALL’ANALISI CASI STUDIO AL MODELLO DATI ASTRATTO .....	49
1.5.1 <i>Analisi</i> .....	49
1.5.2 <i>Progettazione</i> .....	50
2 SVILUPPO DI UN PROTOTIPO DI PIATTAFORMA ICT DI TEST (D6A.2) .....	52
2.1 INFRASTRUTTURA ICT ENEA.....	53

2.2	ARCHITETTURA SOFTWARE.....	56
2.3	PERSISTENZA .....	59
2.4	IMPLEMENTAZIONE PIATTAFORMA .....	60
2.4.1	<i>Teoria considerata</i> .....	60
2.4.2	<i>Metodologie sperimentali utilizzate</i> .....	60
2.4.3	<i>Strumenti adottati</i> .....	60
2.4.4	<i>Tecnologie sviluppate</i> .....	61
3	CONCLUSIONI.....	64
4	APPENDICE A – SCPS CORE .....	65
5	APPENDICE B – SCPS FUNCTIONAL LEVEL.....	67
5.1	L’ARCHITETTURA DI RIFERIMENTO .....	68
5.2	CONCETTI SCPS-BASED.....	69
5.2.1	<i>Solution</i> .....	69
5.2.2	<i>Urban Dataset</i> .....	70
5.2.3	<i>Ontology</i> .....	71
5.2.4	<i>Implementazioni XML/JSON</i> .....	71
5.2.5	<i>Trasporto UrbanDataset Getaway</i> .....	71
5.2.6	<i>Registry</i> .....	72
5.3	CONCETTI NO-SCPS NECESSARI .....	72
5.3.1	<i>Sign In e Authentication</i> .....	72
5.3.2	<i>Ontology Manager</i> .....	73
5.3.3	<i>Urban Dataset Manager</i> .....	73
5.3.4	<i>Urban Dataset Database</i> .....	73
5.3.5	<i>Business Manager</i> .....	73
5.3.6	<i>GUI</i> .....	74
5.4	CONCETTI NO-SCPS OPZIONALI .....	74
5.5	UTENTI.....	74
5.6	USE CASE DIAGRAM.....	75
5.7	CONFIGURAZIONE SCP.....	76
5.7.1	<i>Creazione Istanza</i> .....	76
5.7.2	<i>Definizione Administrator</i> .....	76
5.7.3	<i>Inserimento UrbanDataset</i> .....	76
5.8	CONFIGURAZIONE SOLUTION .....	77
5.8.1	<i>Inserimento Solution</i> .....	77
5.8.2	<i>Definizione Collaboration</i> .....	77
5.9	COMUNICAZIONE INTEROPERABILE .....	77
5.9.1	<i>Produzione di UrbanDataset</i> .....	78
5.9.2	<i>Accesso a UrbanDataset</i> .....	78
5.9.3	<i>Monitoraggio</i> .....	79
5.10	CANCELLAZIONE SOLUTION .....	79

5.10.1	<i>Cancellazione UrbanDataset</i> .....	79
5.10.2	<i>Cancellazione Solution</i> .....	79
6	APPENDICE C – SCPS COLLABORATION LEVEL .....	81
7	APPENDICE D – SCPS SEMANTIC LEVEL .....	82
7.1	DESCRIZIONE DELL’ONTOLOGIA E MODELLO SEMANTICO .....	82
7.1.1	<i>Code List</i> .....	87
7.2	CONSULTAZIONE DELL’ONTOLOGIA .....	89
7.3	INTERROGAZIONE DELL’ONTOLOGIA .....	90
7.4	INSERIMENTO DI NUOVI URBAN DATASET .....	94
7.4.1	<i>Il Comitato di gestione dell’ontologia</i> .....	94
7.4.2	<i>Procedura per la richiesta d’inserimento</i> .....	94
7.4.3	<i>Modulo per la richiesta di inserimento</i> .....	95
8	APPENDICE E – SCPS INFORMATION LEVEL .....	101
8.1	IL MODELLO DATI ASTRATTO .....	101
8.1.1	<i>Modello Dati Astratto - sezione "Specification"</i> .....	103
8.1.2	<i>Modello Dati Astratto - sezione "Context"</i> .....	104
8.1.3	<i>Modello Dati Astratto - sezione "Values "</i> .....	105
8.1.4	<i>Modello Dati Astratto e Ontologia</i> .....	106
8.2	SINTASSI DI RIFERIMENTO E VALIDAZIONE .....	107
8.3	DESCRIZIONE ED ESEMPIO DI UTILIZZO DI ALCUNI ELEMENTI DEL MODELLO DATI ASTRATTO .....	110
8.3.1	<i>specificationReferences</i> .....	110
8.3.2	<i>propertyDefinition</i> .....	111
8.3.3	<i>line</i> .....	115
8.4	CREARE LE ISTANZE DEGLI URBAN DATASET .....	118
8.5	ESEMPIO: URBAN DATASET WHATEVER .....	121
8.6	RISORSE DISPONIBILI .....	126
9	APPENDICE F – SCPS COMMUNICATION LEVEL .....	127
9.1	PATTERN .....	128
9.1.1	<i>Pattern REQUEST/RESPONSE</i> .....	128
9.1.2	<i>Pattern PUSH</i> .....	129
9.2	WEB SERVICE INTERFACE .....	130
9.2.1	<i>login method interface</i> .....	130
9.2.2	<i>push method interface</i> .....	131
9.2.3	<i>basicRequest method interface</i> .....	131
9.2.4	<i>searchingRequest method interface</i> .....	132
10	APPENDICE G – SCPS TERMINOLOGIA .....	133
10.1	TERMINOLOGIA DALLO STATO DELL’ARTE .....	133
10.2	TERMINOLOGIA SMART CITY ENEA .....	134
10.3	ACRONIMI .....	137

11	APPENDICE H - SCPS POLICY PER L'USO DEGLI IDENTIFICATORI .....	139
12	APPENDICE I - RIFERIMENTI BIBLIOGRAFICI .....	140

## Sommario

L'obiettivo D6.a prevede la definizione di un set di specifiche su cui possa basarsi l'implementazione di una Piattaforma ICT per la gestione dello Smart District.

Questo si concretizza tramite due sotto-obiettivi principali:

1. un insieme di Specifiche per l'Interoperabilità: le **Smart City Platform Specification (SCPS)**;
2. l'implementazione di un **prototipo** di Piattaforma ICT, **Smart City Platform**, per la gestione dello **Smart District**, basata sulle Specifiche per l'Interoperabilità SCPS, per testare l'efficacia delle stesse.

Il lavoro del triennio, svolto per l'obiettivo D6.a, sta seguendo un percorso che può essere riassunto con le seguenti tappe intermedie, 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;
- anno 3 (PAR2017): Definizione completa delle Specifiche per l'Interoperabilità e sviluppo software per l'adesione della Piattaforma ICT alle Specifiche; il test finale prevede un servizio di monitoraggio delle diverse Solution verticali collegate alla Piattaforma ICT in ottica Smart District.

Il PAR2016, centrale nel triennio previsto, ha perseguito un duplice obiettivo: da un lato ha utilizzato il lavoro di analisi del primo anno per definire una prima versione delle Smart City Platform Specification, in particolare relative alle rappresentazioni sintattiche e ontologiche dei dati scambiati, dall'altro ha avviato lo sviluppo software di una Piattaforma ICT, sviluppo che continuerà nel terzo anno, con l'adesione completa alle Specifiche e la sperimentazione.

Il D6.a è stato di conseguenza organizzato due sotto-obiettivi:

- D6.a1: Definizione Specifiche.
- D6.a2: Sviluppo di un prototipo di Piattaforma ICT di test.

In questo rapporto saranno descritti organizzazione, svolgimento e risultati di entrambi i sotto-obiettivi, che, globalmente, hanno richiesto la collaborazione sinergica di tre gruppi:

1. il sotto-obiettivo a1, relativo alle Smart City Platform Specification, è stato affidato al laboratorio ENEA CROSS, in possesso dell'esperienza necessaria su standard e formati comuni di rappresentazione digitale dei dati e di comunicazione nello scambio dati;
2. il sotto-obiettivo a2 è stato assegnato al laboratorio ENEA ICT, da sempre impegnato nello sviluppo software e con una forte esperienza nell'ambito delle tecnologie di integrazione e interoperabilità cloud al fine di ottimizzare flessibilità operativa, affidabilità delle infrastrutture e piattaforme cloud.
3. il coordinamento, la supervisione e il lavoro di sinergia con gli altri obiettivi del D6 è stata invece affidata al laboratorio ENEA SCC, Smart Cities and Communities, avendo tale gruppo una notevole esperienza sulle tematiche relative alle Smart City.

Il risultato derivante dai due sub-task è identificabile nella prima versione del formato di interscambio dati e dell'ontologia, facenti parte delle Smart City Platform Specification, e nello sviluppo del framework del primo prototipo di Piattaforma ICT.

Il report è organizzato come segue:

- Il Capitolo 1 "Definizione specifiche e architettura di riferimento (D6a.1)" :
  - o illustra la metodologia con cui si è giunti alla definizione delle specifiche

- esamina i casi di studio che sono stati analizzati per giungere alla definizione delle specifiche
- Il Capitolo 2 “Sviluppo di un prototipo di piattaforma ICT di test (D6a.2)”:
  - espone il lavoro svolto (teoria, metodologie sperimentali utilizzate, tecnologie sviluppate, ecc.) nella fase di sviluppo del dimostratore
  - illustra i risultati ottenuti ed eventuali prodotti realizzati
- Appendici: riportano le specifiche Smart City Platform Specification, prodotte sui cinque livelli d’interoperabilità, e che sono disponibili singolarmente all’indirizzo:  
<http://smartcityplatform.enea.it/specification/>



## Introduzione

Le Specifiche architettruali per la Smart City Platform sono un elemento fondamentale nella direzione di sviluppare un modello integrato di distretto urbano intelligente.

L'obiettivo è guidare lo sviluppo di piattaforme ICT, orizzontali rispetto ai vari servizi verticali in una Smart City, in modo da fornire a essi le necessarie interfacce di scambio dati, evitando la creazione di silos non interoperabili (Figura 1).

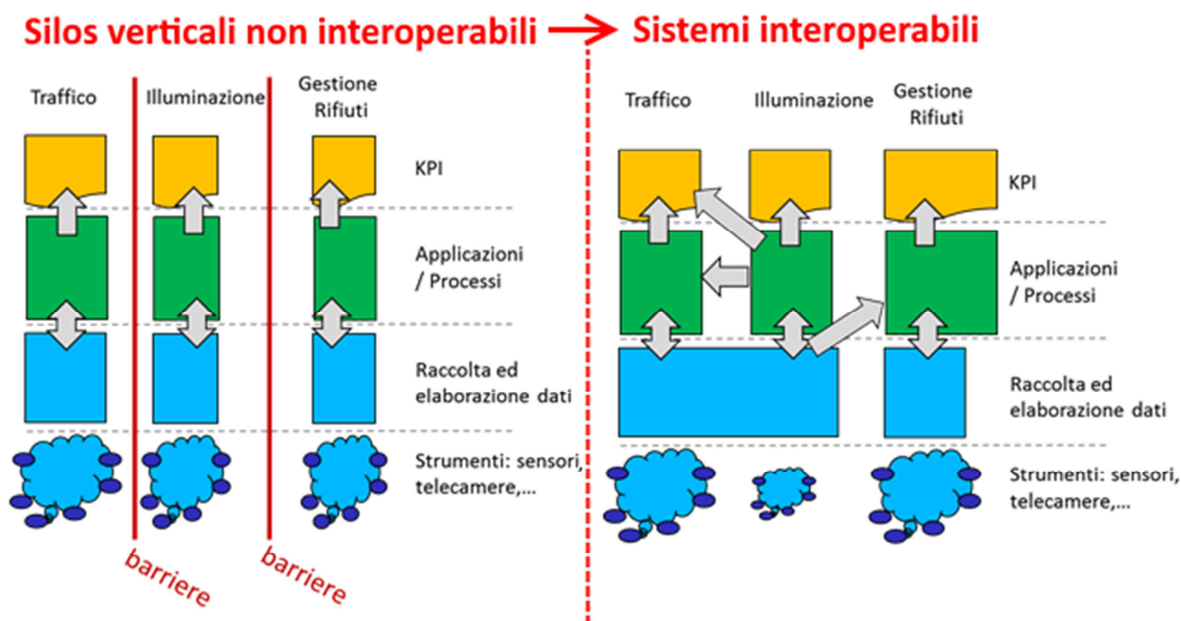


Figura 1 – Rompere i silos della Smart City

A tale scopo, nel corso del PAR2016, sulla base dei risultati del PAR2015, si è proceduto a:

- compiere un'analisi di **casi studio reali** specifici di Piattaforma e definire la **specifiche SCPS**;
- iniziare lo **sviluppo di un prototipo** di piattaforma ICT su cui testare le specifiche.

In entrambe le precedenti attività, si è fatto uso del modello di riferimento per l'interoperabilità (Figura 14 pag. 66), sviluppato nel PAR2015 e consolidato nel PAR2016. In particolare, nel modello sono evidenziati i diversi livelli di interoperabilità (si veda, APPENDICE A – SCPS Core a pag. 65).

Le Smart City Platform Specification (SCPS) sono pubbliche e suddivise nei differenti livelli di interoperabilità, individuati dal modello di riferimento (Functional, Collaboration, Semantic, Information, Communication).

Gli aspetti più innovativi del modello si pongono sui livelli *Information* e *Semantic*: è stata infatti individuata la necessità di un modello e un formato comune dei dati, in modo tale che possa essere abilitata la comunicazione con un linguaggio condiviso.

A tal fine, è stato definito il concetto di Urban Dataset, per rappresentare dati che non siano solo indicatori di performance ma anche in grado di dare un'informazione completa relativa alla città da un punto di vista energetico, sociale o logistico, su diversi livelli di aggregazione. Col termine "Urban Dataset" ci si riferisce a **collezioni di dati** (che possono andare da insiemi di dati puntuali prelevati dai sensori, eventualmente aggregati dall'applicazione verticale di dominio, fino a quelli sintetizzati sotto forma di indicatori) che raccolgono informazioni di una città, ma anche al **modello dati astratto** con cui questi vengono definiti e rappresentati nelle SCPS. Si noti che nel report del PAR2015, si è usato il termine di *Urban Key Application Indicator* (UKAI). Nel corso del PAR 2016 si è deciso di cambiare questo nome in *Urban Dataset*, in quanto di comprensione più immediata e più esatta nel voler esprimere il concetto di insiemi di dati che possono

avere livelli di aggregazione variabile, laddove il termine UKAI invece si associava solo a indicatori (e quindi dati aggregati e medie nel tempo).

Per lasciare la flessibilità adatta a livello Semantico, gli Urban Dataset vengono definiti univocamente in un'ontologia, mentre sul livello Information viene definito un formato dati ampio che possa essere adattato alle diverse esigenze informative delle diverse solution verticali.

Per fornire un insieme completo di specifiche si sono inoltre definiti: una specifica descrittiva l'architettura di riferimento, i concetti chiave e l'insieme di funzionalità supportate (Functional Level). Verranno invece completati nel PAR 2017 gli aspetti relativi alle collaborazioni tra Solution della città e SCP (Collaboration Level) le interfacce e i pattern di scambio dati (Communication Level), seppure una parte dei concetti del Communication Level sono stati affrontati già nel PAR 2016 per andare incontro alle necessità del sotto-obiettivo D6a.2.

In parallelo è stato avviato lo sviluppo per implementare e validare le specifiche per l'interoperabilità Smart City Platform Specification (SCPS), procedendo, man mano, all'implementazione di quanto previsto dalle specifiche sui diversi livelli (per esempio aderendo al formato dati definite a livello Information delle SCPS e alle interfacce Web Service previste sul livello Communication delle stesse). L'architettura del prototipo è basata cloud computing ospitata su infrastruttura ENEAGRID e utilizza soluzioni che si prestano alla scalabilità in funzione delle richieste di servizi da parte dei diversi verticali che invieranno dati. Inoltre, lo sviluppo è stato guidato anche dai diversi casi studio specifici: questo permetterà di testare le funzionalità della piattaforma software, verificare l'aderenza alle SCPS e, allo stesso tempo, di verificarne la replicabilità.

## 1 Definizione specifiche e architettura di riferimento (D6a.1)

Lo scenario di riferimento per la definizione delle SCPS è quello di un distretto o una città in cui i diversi “Contesti Applicativi” sono gestiti da apposite “Solution” verticali, laddove si definisce:

- **Contesto Applicativo:** uno specifico dominio applicativo cittadino, verticale, caratterizzato da diversi aspetti tecnologici, sociali ed economici, fra loro logicamente connessi;
- **Solution:** un insieme di tecnologie, metodologie e applicazioni software per la gestione di un Contesto Applicativo verticale.

Le Solution nascono usualmente da progetti *ad hoc*: risultano ben “integrate” verticalmente<sup>1</sup> ma non in grado di comunicare tra loro. L’ottica da cui si è guardato questo problema è quella di una pubblica amministrazione (tipicamente una municipalità) che voglia gestire tutti i Contesti Applicativi verticali cittadini tramite una **Smart City Platform (SCP)** orizzontale interoperabile. La soluzione proposta, dunque, è la definizione di Specifiche per l’interoperabilità, **aperte** e **modulari**, che non solo operino sulle interfacce fra SCP e Solution Verticali ma che vadano a coprire tutti gli aspetti necessari per instaurare una comunicazione chiara e condivisa da tutte le parti.

La prima attività per la definizione delle specifiche è stata la raccolta dei requisiti sulla base dell’analisi di **casi studio specifici di piattaforma**. Per caso studio s’intende l’analisi, a partire da un caso reale, e la descrizione formale dei relativi flussi di dati fra una Solution Verticale e la Smart City Platform o fra due Solution Verticali tramite la SCP. Un caso studio permette di individuare i requisiti necessari alla definizione delle SCPS e di predisporre l’implementazione reale dello scambio dati tramite una Smart City Platform esistente.

Tali casi studio sono stati divisi in tre categorie:

- Casi studio “**inter-solution**”: in essi la SCP fa da tramite per la comunicazione fra due Solution Verticali. In essi è descritto il flusso dei dati da una Solution di un Contesto Applicativo verticale a un’altra, attraverso la Smart City Platform.

I casi studio di questo tipo analizzati sono stati:

- “**SmartBuilding to WebGIS**”: la piattaforma Smart Building, che raccoglie i dati relativi al monitoraggio energetico di un complesso di edifici, invia i dati alla piattaforma WebGIS (sistemi informativi geografici pubblicati tramite Web), allo scopo di permettere la visualizzazione dei dati sulla mappa della città;
  - “**DSS to Smart Building**”: la piattaforma ENEA di monitoraggio territoriale (Decision Support System, DSS) fornisce i dati meteo (rilevati e previsti) a quella di gestione della rete di Smart Building, allo scopo di permettere l’adozione di politiche di risparmio energetico ottimizzate in base alla situazione meteorologica attuale e prevista;
  - “**WWTP to WebGIS**”: la piattaforma di Gestione delle acque reflue invia i dati di monitoraggio alla piattaforma WebGIS, per il monitoraggio, tramite visualizzazione sulla mappa della zona.
- Casi studio “**one-way**”: la SCP è il destinatario finale delle informazioni. Si configura, dunque, un dialogo fra Solution verticale e Smart City Platform. Si tratta di un caso particolare del precedente (di cui rappresenta la prima metà):
    - “**Smart Lighting to PELL (o SCP)**”: la piattaforma PELL (Public Energy Living Lab) è un particolare tipo di piattaforma orizzontale per la gestione energetica dell’illuminazione

---

<sup>1</sup> Come evidenziato dalla ISO 14258:1998 (16), si parla di integrazione quando tutti i sottosistemi condividono la stessa organizzazione interna dei dati, di interoperabilità quando si lavora sulle interfacce, ma i sistemi organizzano internamente i dati in modo indipendente tra loro

pubblica cittadina, che intende adottare le specifiche SCPS. Tale piattaforma deve ricevere i dati di monitoraggio dalle diverse municipalità che aderiscono all’iniziativa. Questo rappresenta un caso di adozione delle specifiche a livello nazionale e non cittadino.

- Casi studio “particolari”: sono pensati per aumentare la modularità e la generalità di applicazione delle SCPSpecification. Sono stati considerati i due seguenti:
  - “**Whatever**”: si tratta di un caso studio generico, pensato per scambiare dati non ancora registrati nella SCP. È utile, per esempio, nel caso di test preliminare per l’adozione del formato o per inviare dati generici. Il suo utilizzo implica una validazione dei dati meno rigorosa (e quindi un livello più basso di interoperabilità) ma consente in questo modo di utilizzare le Specifiche immediatamente e aderire a esse gradualmente;
  - “**Direct Communication**”: permette l’accesso diretto da una Solution verticale a un’altra. In tal caso la SCP non fa transitare i dati ma, fungendo da broker, fornisce a una prima Solution il set d’informazioni necessario alla connessione con la seconda.

Tali casi studio sono analizzati in dettaglio nel paragrafo 1.2, 1.3 e 1.4.

Questo ha permesso di identificare i dati che la Solution verticale deve scambiare con le altre Solution (casi studio inter-solution) o con la SCP (casi studio one-way) e ha portato alla definizione del concetto di **Urban Dataset**<sup>2</sup>, cioè collezioni di dati, fornite dalle applicazioni attive in una città. La natura di tali collezioni può variare, andando da raccolte dati puntuali prelevati dai sensori, fino a indicatori di sintesi (per esempio KPI).

L’analisi dei diversi casi studio elencati è stata effettuata secondo uno schema comune che si può riassumere come segue:

1. Identificazione della sequenza di passi della comunicazione (rappresentata tramite Sequence Diagram UML);
2. Analisi degli Urban Dataset scambiati (tipologia e proprietà che li caratterizzano);
3. Analisi degli aspetti relativi alla collaborazione fra le Solution e la SCP. Gli aspetti individuati come significativi in tale fase sono:
  - Periodicità con cui sono forniti / ricevuti i dati
  - Credenziali di connessione
4. Formato di scambio dati
5. Protocollo di trasporto / pattern di scambio dati

Quest’analisi ha permesso, tra l’altro, di:

- definire la struttura delle specifiche;
- individuare la struttura degli Urban Dataset (per esempio, le informazioni che devono essere sempre presenti e quelle caratteristiche di ciascun caso studio);
- individuare i principali requisiti dei protocolli di trasporto;
- elaborare le specifiche per la SCP, suddivise in sottospecifiche (una per ciascun livello di interoperabilità). Le specifiche sono state elaborate in modo da risultare il più possibile indipendenti fra loro, così che si possa aderire alle specifiche in modo graduale.

---

<sup>2</sup> Il concetto di Urban Dataset sostituisce e assorbe quello di Urban Key Application Indicator (UKAI) che era stato utilizzato nel PAR 2015. Il nome UKAI facendo riferimento solo a “indicatore” e quindi dati di sintesi, risultava poco preciso rispetto all’esigenza di inviare collezioni di dati generali che, dunque, possono anche essere di dettaglio.

Le specifiche risultanti da tale processo, a fine PAR2016, sono consultabili integralmente nelle appendici riportate nella parte finale di questo report:

- ➔ APPENDICE A – SCPS Core a pag. 65
- ➔ APPENDICE B – SCPS Functional Level a pag. 67
- ➔ APPENDICE C – SCPS Collaboration Level a pag. 81
- ➔ APPENDICE D – SCPS Semantic Level a pag. 82
- ➔ APPENDICE E – SCPS Information Level a pag. 101
- ➔ APPENDICE F – SCPS Communication Level a pag. 127
- ➔ APPENDICE G – SCPS Terminologia a pag. 133

### 1.1 Urban Dataset e livelli di aggregazione

Gli **Urban Dataset** rappresentano *collezioni* di dati, forniti dalle applicazioni attive in una città. La natura di tali collezioni può variare, andando da raccolte *dati puntuali* prelevati dai sensori, fino a *indicatori* di sintesi (per esempio KPI).

Per chiarire in quale zona di quest'ampia gamma di possibilità si collochi un Urban Dataset, quando se ne definisce uno è utile delimitare per esso sia il livello di aggregazione spaziale sia quello temporale. Tali livelli di aggregazione potranno assumere i seguenti valori

- Livello di aggregazione spaziale:
  - **not applicable**: nel caso il concetto di aggregazione spaziale non abbia senso per lo specifico Urban Dataset;
  - **item**: se i dati si riferiscono al particolare sensore o, più in generale, oggetto, al livello di aggregazione più basso (per esempio, nel caso dell'illuminazione stradale, al singolo palo);
  - **facility**: se i dati si riferiscono all'intera struttura monitorata (per esempio, l'edificio, nel caso di Smart Building, lampioni di una via, per l'illuminazione stradale, ecc.);
  - **facility aggregation**: se si riferisce a un insieme di strutture aggregate fra loro (per esempio un gruppo di Smart Building monitorati da uno stesso Aggregatore);
  - **city**: intera infrastruttura cittadina (per esempio l'intera rete di illuminazione pubblica di una città)
  - **region**: se ci si riferisce a infrastrutture che si estendono oltre la singola città, a un'intera zona/regione geografica (per esempio, impianti di depurazione delle acque)
  - **country**: infrastrutture che si estendono all'intera nazione (per esempio la rete di trasporto dell'energia elettrica)
- Livello di aggregazione temporale:
  - **not applicable**: nel caso il concetto di aggregazione temporale non abbia senso per lo specifico Urban Dataset;
  - **static**: Urban Dataset anagrafici, che definiscono le caratteristiche statiche di una struttura (per esempio localizzazione spaziale di un edificio)
  - **instantaneous**: dati istantanei (per esempio temperatura istantanea misurata da un sensore)
  - **average**: valori medi (per esempio il consumo medio di elettricità in un determinato intervallo di tempo, in un edificio)

- **total**: valori totali (per esempio il consumo totale medio di elettricità in un determinato intervallo di tempo, in un edificio)
- **forecast**: valori previsti e quindi future (per esempio previsioni meteo)

Oltre a ciò, è utile classificare gli Urban Dataset in base alla categoria di applicazioni che li produce. A tale scopo risulta utile fare riferimento alla seguente categorizzazione elaborata nell’ambito della collaborazione fra ENEA e NIST nell’iniziativa “IES-City” (1).

La classificazione è la seguente:

Categorie:	Sotto-categorie
Built environment	Smart Home Smart Building Land use and management
Water and wastewater	Water collection and management Water distribution Water consumption Wastewater management
Waste	Citizens engagement Collection and segregation Waste disposal
Energy	Energy supply Energy transmission and distribution Energy demand
Transportation	Travel demand/consumption Traffic management Surveillance
Education	Learning outcomes Learning and teaching Service management
Health	Health care systems Health care delivery Communication
Socio-economic development	E-Governance Social Innovation and Inclusion Economy and Business
Public safety, policy & Em.Res.	City surveillance and crime prevention Communication

**Tabella 1 – Categorizzazione delle applicazioni sulla base del lavoro svolto nell’ambito di IES-City**

Le classificazioni degli Urban Dataset sulla base dei livelli di aggregazione e delle categorie di applicazioni, non verranno inviate insieme ai dati, ma saranno solamente associate alle definizioni degli stessi all’interno dell’**Ontologia**, in quanto si ritiene che la loro utilità sia in fase di ricerca degli Urban Dataset disponibili nella Smart City Platform.

### 1.2 Casi Studio Inter-Solution

Questo paragrafo descrive i Casi Studio specifici per la Smart City Platform (SCP), denominati casi studio “Inter-Solution”. Tali casi studio descrivono il flusso dei dati da una Solution di un contesto applicativo

verticale a un altro, attraverso la Smart City Platform (p.es. dalla Solution Smart Building alla Solution WebGIS).

Tali casi studio sono necessari per delineare un set di requisiti sufficientemente completo per:

- definire in maniera completa le Smart City Platform Specification (D7a.1);
- implementare tutte le funzionalità necessarie al prototipo (D7a.2).

I casi studio Inter-Solution creati sono i seguenti:

1. "Smart Building to WebGIS" (visualizzazione anomalie e consumi);
2. "DSS to Smart Building" (dati meteo per logiche di gestione energetica);
3. "WWTP to DSS" (monitoraggio depuratori).

Nelle descrizioni di questi casi studio Inter-Solution, si farà riferimento a:

- componenti della SCP come "Registry", "BusinessManager" e "Ontologia";
- i dati scambiati, Urban Dataset (si veda par 1.1);
- i pattern e le interfacce di Trasporto.

Questi concetti, maturati proprio grazie all'analisi e definizione dei casi studio, sono contenuti nelle Smart City Platform Specification, definite in un documento a se stante, a cui si farà riferimento.

### 1.2.1 Case Study “SmartBuilding to WebGIS”

Il caso studio specifico in oggetto descrive il flusso dati che, tramite la Smart City Platform, avviene tra:

- la Solution verticale “Smart Building”: piattaforma software che raccoglie i dati relativi al monitoraggio energetico (elettrico e termico) di un complesso di edifici, allo scopo di monitoraggio, diagnostica e attuazione per una corretta gestione energetica;
- la Solution verticale “WebGIS”: strumento di georeferenziazione dei dati.

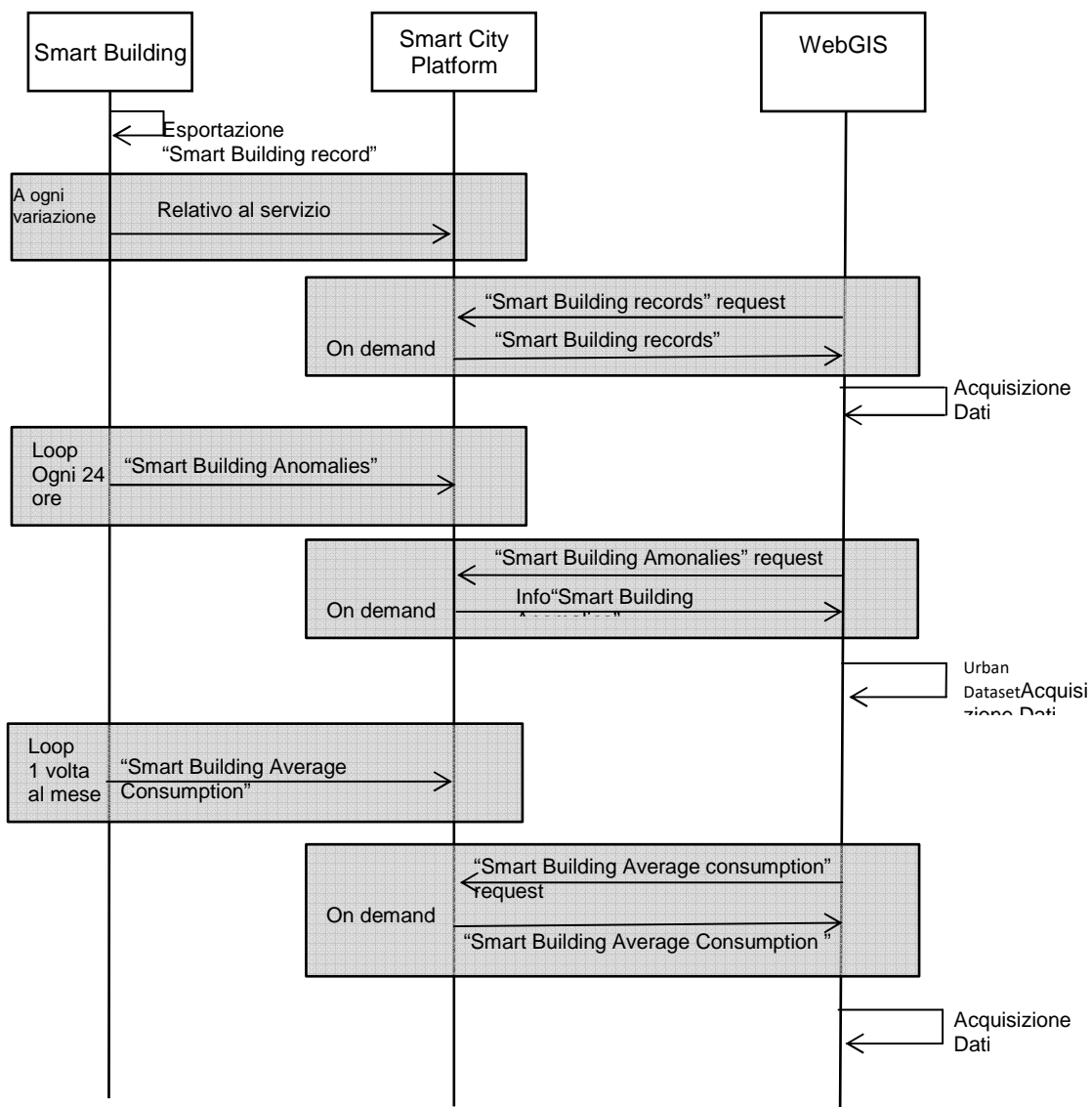
L’implementazione di questo caso studio permette di effettuare il monitoraggio energetico del contesto applicativo Smart Building tramite lo strumento WebGIS.

I passi che riassumono il passaggio di dati dalla prima Solution verticale Smart Building alla seconda Solution verticale WebGIS, attraverso la Smart City Platform, sono i seguenti:

1. Solution verticale Smart Building: generazione ed esportazione, dal proprio sistema di persistenza locale, degli Urban Dataset;
2. Solution verticale Smart Building: pubblicazione degli Urban Dataset per la Smart City Platform;
3. Smart City Platform: ricezione e acquisizione Urban Dataset;
4. Smart City Platform: persistenza degli Urban Dataset acquisiti da Smart Building;
5. Smart City Platform: pubblicazione degli Urban Dataset per il WebGIS;
6. Solution verticale WebGIS: acquisizione degli Urban Dataset Building dalla Smart City Platform;
7. Solution verticale WebGIS: importazione degli Urban Dataset e visualizzazione.

Il seguente Diagramma di Sequenza UML descrive i passi delle due collaborazioni, esplicitando i pattern di scambio dati (si veda la sotto-specifica “SCPS Communicaton Level” riportata in APPENDICE F – SCPS Communication Level) scelti per risolvere ogni step della comunicazione.





**Figura 2 – Diagramma di sequenza UML “Smart Building to WebGIS” (non sono mostrati gli eventuali Ack)**

Si noti che la comunicazione fra WebGIS e Smart City Platform avviene in maniera asincrona: periodicamente il WebGIS, perciò, dovrà fare polling verso la SCP per ricevere i dati aggiornati.

I seguenti sotto-paragrafi descrivono tutti i requisiti necessari per implementare il caso studio in oggetto dal punto di vista dell’Interoperabilità, sia in forma testuale che in forma sintetica tabellare.

### 1.2.1.1 Collaborazione

Tutti gli aspetti relativi alla collaborazione tra una Solution verticale e la Smart City Platform dovranno essere contenuti nel Registry della SCP che, in maniera centralizzata, tiene traccia delle Solution verticali coinvolte (il Registry e la relativa specifica del Collaboration Layer verranno definita nel dettaglio nel corso PAR 2017), delle relative credenziali di accesso, di quali dati saranno pubblicati, da chi, accessibili a chi, in quale modo.

Si ipotizza che la collaborazione concordata sia stata definita (attraverso un meccanismo di negoziazione o tramite bando di gara) per poi essere successivamente inserita nel Registry manualmente dall’amministratore.

Il Registry terrà traccia di tutte le collaborazioni binarie tra Smart City Platform e Solution verticale, nonché di tutti gli Urban Dataset pubblicabili/accessibili con le relative periodicità. Ipotesi:

1. la piattaforma verticale Smart Building pubblica:
  - Urban Dataset 1 (Smart Building records):
    - o una sola volta, essendo di tipo statico, a partire da 01/05/2018 ore 00:00;
    - o a ogni modifica, per un anno, fino al 01/05/2019 alle ore 00:00;
  - Urban Dataset 2 (anomalie):
    - o una volta al giorno alle ore 00:00, a partire dal 02/05/2018, per un anno, fino al 02/05/2019 alle ore 00:00;
  - Urban Dataset 3 (consumo elettrico medio giornaliero):
    - o una volta al mese alle ore 00:00, a partire dal 02/05/2018, per un anno, fino al 02/05/2019 alle ore 00:00
2. la piattaforma verticale WebGIS richiede alla piattaforma SCP Urban Dataset 1 periodicamente per sapere se vi sono aggiornamenti (p.es. una volta al giorno, alle ore 00:01 a partire dal 01/05/2018, , Urban Dataset 2 e una volta al mese per l'Urban Dataset 3, alle ore 00:01, a partire dal 01/05/2018 ore 00:00) e li memorizza.

In altre parole, il criterio di generazione dell'Urban Dataset è il seguente:

- l'Urban Dataset "Smart Building records" descrive tutti i palazzi monitorati una sola volta per tutte (salvo modifiche all'infrastruttura building);
- l'Urban Dataset "Smart Building anomalies" descrive tutte le anomalie calcolate in quella giornata su tutti i palazzi (quindi vedo solo i palazzi su cui si sono verificate anomalie ma il calcolo è fatto su tutti e su tutti i periodi, ogni giorno);
- l'Urban Dataset "Smart Building average consumption" fornisce i consumi elettrici giornalieri dell'ultimo mese.

Nel Registry, inoltre, sono indicate le credenziali di accesso con cui le Solution verticali possono connettersi alla Smart City Platform e viceversa.

Nel caso studio corrente, sono create manualmente le seguenti credenziali:

1. Smart Building
  - a. Username: [smartbuilding.admin@enea.it](mailto:smartbuilding.admin@enea.it)
  - b. Password: [\*\*\*\*\*]
2. WebGIS
  - a. Username: [webgis.admin@enea.it](mailto:webgis.admin@enea.it)
  - b. Password: [\*\*\*\*\*]

Queste credenziali di accesso possono essere utilizzate dalla Solution verticale per accedere:

- al proprio profilo;
- a una view degli Urban Dataset pubblicabili;
- a navigare gli Urban Dataset pubblicati dalla SCP ai quali si ha accesso;
- ai Web Service per pubblicare/recuperare Urban Dataset.

Anche la SCP deve avere delle credenziali di accesso, che ipotizziamo:

1. Credenziali SCP presso la piattaforma WebGIS

- a. Username: [smartcityplatform@enea.it](mailto:smartcityplatform@enea.it)
- b. Password: [\*\*\*\*\*]

Tutti i requisiti raccolti relativi alla collaborazione sono stati analizzati nei casi studio Inter-Solution e hanno permesso di avviare la definizione del Registry che sarà, nel PAR2017, contenuta nella specifica “SCPS Collaboration Level”, livello del modello di riferimento per l’Interoperabilità (si veda la sottospecifica “SCPS Core” riportata in APPENDICE A – SCPS Core).

### 1.2.1.2 Urban Dataset coinvolti

L'insieme di dati scambiati tra verticali attraverso la Smart City Platform sono denominati Urban Dataset (si veda APPENDICE G – SCPS Terminologia).

Un Urban Dataset è, in primis, registrato nell'Ontologia (si veda la sottospecifica "SCP Semantic Level" in APPENDICE D – SCPS Semantic Level), in modo tale possano essere esportate le regole di validazione nei formati utilizzati (JSON o XML). Solo l'Urban Dataset "Whatever" (par. 1.4.1) permette di inviare dati generici non ancora definiti nell'Ontologia.

Seguono le definizioni dei tre Urban Dataset individuati per il caso studio "Smart Building to WebGIS":

1. "Smart Building Records",
2. "Smart Building Anomalies",
3. "Smart Building Average Consumption".

### 1.2.1.3 Urban Dataset "Smart Building Records"

Questo primo Urban Dataset, di tipo statico, fornisce una descrizione univoca dei palazzi monitorati nel contesto applicativo verticale Smart Building e trasporta i seguenti dati:

- Proprietà di identificazione dell'Urban Dataset:
  - **UrbanDatasetId**: identificatore dell'Urban Dataset fornito dall'Ontologia;
  - **UrbanDatasetName**: Smart Building Records
  - **UrbanDatasetRef**: URI che identifica la specifica
- Contestualizzazione:
  - **Timestamp**: tempo di generazione di questo Urban Dataset
  - **Timezone**: esplicita il fuso orario dei timestamp presenti nell'Urban Dataset
  - **Coordinate**: coordinate (in formato wgs84) del centro geometrico della rete di Smart Building su cui sono stati rilevati i valori trasmessi
  - **Producer**: identifica, in modo univoco, la piattaforma/soluzione verticale che ha prodotto i dati rilevati
- Proprietà specifiche:
  - **Coordinates**: Coordinate wgs84 del centro del palazzo a cui sono riferiti i dati rilevati indicati in questa riga;
  - **BuildingId**: identificatore del palazzo monitorato e generatore di anomalie;
  - **BuildingName**: etichetta associata al palazzo monitorato e generatore di anomalie.
- Per esempio:
  - **UrbanDatasetId**: 1.0-SmartBuildingRecords;
  - **Coordinates**: 42.04 ; 12.30;
  - **BuildingId**: F40;
  - **BuildingName**: "Building F40".

Le precedenti informazioni possono essere schematizzate nella seguente tabella:

Urban Dataset	Smart Building Records		
Scopo	Anagrafica edifici: fornisce una descrizione univoca dei palazzi monitorati nel contesto applicativo verticale Smart Building.		
Aggregazione spaziale	Facility		
Categoria / Sottocategoria	Built environment / Smart Building		
Aggregazione temporale	Statico		
<b>SPECIFICAZIONE DELL'URBAN DATASET</b>			
UrbanDatasetId	1.0-SmartBuildingRecords		
UrbanDatasetUri	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingRecords">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingRecords</a>		
UrbanDatasetName	Smart Building Records		
<b>CONTESTUALIZZAZIONE DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
Timestamp	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell'Urban Dataset
Timezone	Adimensionale	String	Es. UTC, UTC+01, UTC-05
Coordinates	Adimensionale	WGS84 (World Geodetic System)	Coordinate del centro geometrico della rete di Smart Building es. "42.04" ; "12.30"
Producer	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
Coordinates	Adimensionale	WGS84 (World Geodetic System)	Coordinate per geolocalizzare il centro del palazzo (es. "42.04" ; "12.30")
BuildingID	Adimensionale	String	Identificatore del palazzo monitorato e generatore di anomalie (es. 1). Questo diventerà il codice di riferimento per tutti gli Urban Dataset associati a questo edificio
Building Name		Adimensionale	String

**Tabella 2 – Tabella di sintesi dell'Urban Data "Smart Building Records"**

#### 1.2.1.4 Urban Dataset "Smart Building Anomalies"

Questo secondo Urban Dataset, viene inviato una volta al giorno e conteggia le anomalie relative al consumo energetico, suddivise in tre categorie (low, mean, high) e associate a una particolare causa e a un determinato palazzo, calcolate su una predefinita finestra temporale.

- Specificazione dell'Urban Dataset:
  - **UrbanDatasetID**
  - **UrbanDatasetName**
  - **Description**
- Contestualizzazione:
  - **Timestamp**
  - **Timezone**
  - **Coordinates**
  - **Producer**
- Proprietà specifiche:
  - **BuildingID**
  - **BuildingName**

- **StartPeriod**
- **EndPeriod**
- **CauseID**: identificatore della causa dell’anomalia (associata a una tabella di cause predefinite, Tabella 4)
- **CauseDescription**: descrizione testuale della causa dell’anomalia (Tabella 4)
- **HighAnomalies**: numero totale di anomalie di tipo high verificatesi nel periodo
- **MeanAnomalies**: numero totale di anomalie di tipo mean verificatesi nel periodo
- **LowAnomalies**: numero totale di anomalie di tipo low verificatesi nel periodo

Le precedenti proprietà sono raccolte e sintetizzate nella seguente tabella, che rappresenta l’Urban Dataset:

Urban Dataset	Smart Building Anomalies		
<b>Scopo</b>	Conteggia le anomalie (low, mean, high) associate a una particolare causa, a un determinato palazzo, calcolate su una particolare finestra temporale.		
<b>Aggregazione spaziale</b>	Facility		
<b>Categoria / Sottocategoria</b>	Built environment / Smart Building		
<b>Aggregazione temporale</b>	Valori totali		
SPECIFICAZIONE DELL’URBAN DATASET			
<b>UrbanDatasetId</b>	1.0-SmartBuildingAnomalies		
<b>UrbanDatasetUri</b>	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAnomalies">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAnomalies</a>		
<b>UrbanDatasetName</b>	Smart Building Anomalies		
CONTESTUALIZZAZIONE DELL’URBAN DATASET			
	Unità di Misura:	Formato	Descrizione / Esempio
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell’Urban Dataset
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate del centro geometrico della rete di Smart Building es. “42.04” ; “12.30”
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
PROPRIETÀ DELL’URBAN DATASET			
	Unità di Misura:	Formato	Descrizione / Esempio
<b>BuildingID</b>	Adimensionale	String	Identificatore del palazzo monitorato e generatore di anomalie (es. 1). Questo diventerà il codice di riferimento per tutti gli Urban Dataset associati a questo edificio
<b>Building Name</b>	Adimensionale	String	Es. “F40”
<b>Start Period</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di inizio del conteggio delle anomalie (es. 26-09-2017T00:00:00)
<b>End Period</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di fine della conteggio delle anomalie(es. 26-09-2017T23:59:59)
<b>CauseID</b>	Adimensionale	String	Codice anomalia (Definito da tabella delle anomalie) Es. 1
<b>CauseDescription</b>	Adimensionale	String	Descrizione anomalia (Definita da tabella delle anomalie) Es. Luci accese in assenza di persone
<b>HighAnomalies</b>	Adimensionale	String	Numero di High Anomalies nell’intervallo di riferimento. Es. 4
<b>MeanAnomalies</b>	Adimensionale	Numero di Mean Anomalies nell’intervallo di riferimento	Numero di Mean Anomalies nell’intervallo di riferimento. Es. 3

<b>LowAnomalies</b>	Adimensionale	Definito da tabella dell'anomalie	Numero di Low Anomalies nell'intervallo di riferimento. Es. 1
---------------------	---------------	-----------------------------------	---

**Tabella 3 – Tabella di sintesi dello Urban Data “Smart Building Anomalies”**

Il parametro *BuildingId* permette di associare questo Urban Dataset al palazzo identificato tramite l’Urban Dataset Anagrafico “Smart Building Records”.

Il parametro *CauseId* fa riferimento a una lista di cause univoca, a cui questo Urban Dataset fa riferimento. Si noti che la lista sarà esterna all’ontologia e lo schema vi farà riferimento come CodeList.

Cause_Id	Diagnostica HighLevel	Algoritmo
1	Luci accese in assenza di persone	S1a OR S30
2	Luci accese in orario di lavoro con illuminazione esterna sufficiente	S1c AND S33a
3	Fancoil accesi in assenza di persone	S2a AND NOT S30
4	Malfunzionamento del timer in centrale termica	S10 OR S11
5	Scorretta gestione della distribuzione del calore	S12 OR S13
6	Valvola	S14 OR S15
7	Staratura del termostato di stanza in periodo estivo	S16
8	Anomalia impianto di raffrescamento	S17

**Tabella 4 – Tabella di decodifica delle anomalie**

Riportiamo di seguito la spiegazione dell’algoritmo relativo alla causa "Luci accese in assenza di persone" (riga 1). Tale algoritmo è descritto tramite un’espressione logica composta da due variabili unite da operatore logico “OR”, “S1a OR S30”, dove

1. S1a: Accensione di un elevato numero di luci

Il calcolo di S1 è rappresentato da questa espressione:

$$P1a \text{ OR } P1b \text{ OR } (0.7P8a + 0.3P8b)$$

Dove:

- P1a: Outlier potenza elettrica luci
- P1b: trend anomalo potenza elettrica luci
- P8a: NULL
- P8b: NULL

2. S30: % presenze alta fuori orario di lavoro

Il calcolo di S30 è rappresentato da questa espressione:

$$P51 \text{ AND } P24c$$

Dove:

- P51: Orario di lavoro
- P24c: % stanze occupate

Questa descrizione viene riportata nell’Ontologia, nella definizione dell’Urban Dataset, anche se solo come testo, per premettere a chi gestisce Solution verticale di disporre, tramite essa, di tutte le informazioni necessarie a produrre questo Urban Dataset e, a chi lo riceverà, di interpretarlo correttamente.

#### 1.2.1.5 Urban Dataset “Smart Building Average Consumption”

Il terzo Urban Dataset, anch’esso di tipo dinamico, riporta i consumi elettrici giornalieri (da mezzanotte a mezzanotte) degli edifici della rete e viene inviato una volta al mese, in modo che tali valori possano essere visualizzati sul WebGIS. Ogni linea avrà le seguenti "proprietà specifiche":

- BuildingId;

- BuildingName;
- Period (permette di risalire al giorno a cui si riferisce la singola misura)
- AverageConsumption

Le coordinate sulla singola linea saranno invece opzionali, poiché è possibile risalire a esse tramite l’Urban Dataset anagrafico.

Urban Dataset		Smart Building Average Consumption		
<b>Scopo</b>	Registrare il consumo energetico giornaliero di uno o più edifici, al fine di comunicarlo, mensilmente, al WebGis e permetterne la visualizzazione sulla mappa.			
<b>Aggregazione spaziale</b>	Facility			
<b>Categoria / Sottocategoria</b>	Built environment / Smart Building			
<b>Aggregazione temporale</b>	Valore totale			
SPECIFICAZIONE DELL’URBAN DATASET				
<b>UrbanDatasetId</b>	1.0-SmartBuildingAverageConsumption			
<b>UrbanDatasetUri</b>	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAverageConsumption">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAverageConsumption</a>			
<b>UrbanDatasetName</b>	Smart Building Average Consumption			
CONTESTUALIZZAZIONE DELL’URBAN DATASET				
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>	
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell’Urban Dataset	
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05	
<b>Coordinates</b>	Coordinate	WGS84 (World Geodetic System)	Coordinate del centro geometrico della rete di Smart Building es. “42.04” ; “12.30”	
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati	
PROPRIETÀ DELL’URBAN DATASET				
(si ripetono per ogni giorno del mese e per ogni edificio della rete)				
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>	
<b>Start Period</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di inizio della misurazione del consumo energetico (es. 26-09-2017T00:00:00)	
<b>End Period</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di fine della misurazione del consumo energetico (es. 26-09-2017T23:59:59)	
<b>BuildingID</b>	Adimensionale	String	identificatore del palazzo a cui si riferisce la misura	
<b>Building Name</b>	Adimensionale	String	Es. “F40”	
<b>AverageConsumption</b>	kWh	Double	Consumo energia elettrica dell’edificio. Es. 555,3 kWh	

**Tabella 5 - Definizione Urban Dataset “Consumo elettrico medio dell’edificio”**

### 1.2.1.6 Formati

Ogni Solution Verticale deve dichiarare, nel componente Registry della SCP, quale formato utilizzerà per poter dialogare con la Smart City Platform.

Nel corrente Case Study ipotizziamo la seguente situazione lineare:

1. Smart Building: JSON;
2. WebGIS: JSON;

dove gli Urban Dataset saranno implementati nel formato comune, parte delle specifiche.

I template dei tre Urban Dataset in oggetto sono riportati integralmente in:

<http://smartcityplatform.enea.it/casestudies/smartbuilding-webgis>.



### 1.2.1.7 Trasporto

A ogni Solution Verticale saranno associati, e memorizzati nel Registry, i protocolli di trasporto che verranno utilizzati per potersi connettere alla Smart City Platform (SCP).

Ipotizziamo la seguente situazione per il trasporto dei dati da Smart Building a SCP per i due Urban Dataset:

➤ La Solution Smart Building concorda la seguente configurazione di trasporto tra quelle supportate dalla SCP:

1. per l'Urban Dataset "Smart Building Records", di tipo statico, è stato concordato
  - Pattern PUSH: l'Urban Dataset viene inviato una sola volta (o in caso di aggiornamenti);
  - WS Rest PUSH: l'Urban Dataset viene inviato in JSON (si veda par. precedente) dalla solution Smart Building alla SCP richiamando un endpoint di questo tipo:  
*<http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={ ... }>*
2. per l'Urban Dataset "Smart Building Anomalies", di tipo dinamico, è stato concordato
  - Pattern PUSH: l'Urban Dataset viene esportato ogni giorno e inviato immediatamente e direttamente alla SCP;
  - WS Rest PUSH: l'Urban Dataset verrà inviato in formato JSON alla SCP a un endpoint di questo tipo:  
*<http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={ ... }>*

Ipotizziamo la seguente situazione per il trasporto dei dati da SCP a WebGIS:

➤ La SCP concorda la seguente configurazione di trasporto tra quelle supportate:

1. per l'Urban Dataset "Smart Building Records", di tipo statico, è stato concordato
  - Pattern Request-Response: il WebGis fa polling periodico verso la SCP e richiede l'Urban Dataset aggiornato. Questo viene inviato solo se è stato aggiornato (o in caso di prima richiesta);
  - WS Rest Basic Request: l'Urban Dataset viene inviato in JSON (si veda par. precedente) dalla SCP al WebGIS, in seguito a una richiesta effettuata dal WebGis richiamando un endpoint di questo tipo:  
*<http://www.smartcityplatform.enea.it/UrbanDatasetGateway/basicRequest?urbandataset={...}>*
2. per l'Urban Dataset "Smart Building Anomalies", di tipo dinamico, è stato concordato
  - Pattern Request-Response: il WebGis fa polling periodico verso la SCP e richiede l'Urban Dataset aggiornato. Questo viene inviato solo se è stato aggiornato;
  - WS Rest Basic Request: l'Urban Dataset verrà inviato in formato JSON al WebGIS, in seguito a una richiesta effettuata dal WebGIS richiamando un endpoint di questo tipo:  
*<http://www.smartcityplatform.enea.it/UrbanDatasetGateway/basicRequest?urbandataset={...}>*

1.2.1.8 Tabella sintetica

Caso studio:	Smart Building - WebGIS
<b>Obiettivo</b>	Monitoraggio energetico del contesto applicativo Smart Building tramite lo strumento WebGIS
<b>Attore 1</b>	Smart Building (SB)
<b>Ruolo</b>	Raccolta dei dati relativi al monitoraggio energetico di un complesso di edifici, allo scopo di monitoraggio, diagnostica e attuazione per una corretta gestione energetica; su questa base, generazione ed esportazione, dal proprio sistema di persistenza locale, degli Urban Dataset nel formato comune
<b>Attore 2</b>	WebGis
<b>Ruolo</b>	Strumento di geo-referenziazione dei dati; ricezione degli Urban Dataset e visualizzazione degli Urban Dataset su mappa
<b>Smart City Platform</b>	Recupero Urban Dataset dallo Smart Building, loro memorizzazione nel DB (persistenza) e pubblicazione degli Urban Dataset per il WebGis
<b>Urban Dataset</b>	<b>Smart Building Records</b>
Aggregazione temporale/spaziale	Statico / Facility
Policy dell'Urban Dataset	Privato
Proprietà del dato	Smart Building
Disponibilità del dato	Disponibile solo al WebGis
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingRecords">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingRecords</a>
<b>Urban Dataset</b>	<b>Smart Building Anomalies</b>
Aggregazione temporale/spaziale	Valori Totali / Facility
Policy dell'Urban Dataset	Privato
Proprietà del dato	Smart Building (Disponibile solo al WebGis)
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAnomalies">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAnomalies</a>
<b>Urban Dataset</b>	<b>Smart Building Average Consumption</b>
Aggregazione temporale/spaziale	Valori Totali / Facility
Policy Urban Dataset	Privato
Proprietà del dato	Dello Smart Building (Disponibile solo al DSS)
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAverageConsumption">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAverageConsumption</a>
<b>Start – End Collaboration</b>	Dall'01/05/2018 ore 00:00 al 02/05/2019 alle ore 00:00
	<b>Credenziali di accesso</b>
Registrazione Utente	Manuale, da parte dell'amministratore, nella versione prototipale della SCP
Credenziali Smart Building	<i>Username:</i> smartbuilding.admin@enea.it <i>Password:</i> [*****]
Credenziali WebGis	<i>Username:</i> webgis.admin@enea.it <i>Password:</i> [*****]
Credenziali SCP per accesso a Smart Building	<i>Username:</i> smartcityplatform@enea.it <i>Password:</i> [*****]
	<b>Trasporto e formato</b>
<b>Comunicazione fra SB e SCP</b>	
Pattern	Push
Autenticazione	-
Sicurezza	HTTPS o SSL
Trasporto	L'SCP espone Web Service Rest a cui l'SB invia i dati
<b>Dati</b>	<b>Urban Dataset "Smart Building Record"</b>
Periodicità	una sola volta a partire da 01/05/2018 ore 00:00 e a ogni modifica, per un anno, fino al 01/05/2019 alle ore 00:00
Qualità del servizio	Con garanzia di consegna
WS endpoint	<a href="http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={ ... }">http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={ ... }</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	<a href="http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/">http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/</a>
<b>Dati</b>	<b>Urban Dataset "Smart Building Anomalies"</b>
Periodicità	una volta al giorno alle ore 00:00, a partire dal 02/05/2018, per un anno, fino al 02/05/2019 alle ore 00:00
Qualità del servizio	Con garanzia di consegna

WS endpoint	<a href="http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={...}">http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={...}</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	<a href="http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/">http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/</a>
<b>Dati</b>	<b>Urban Dataset "Smart Building Average Consumption"</b>
Periodicità	una volta al mese alle ore 00:00, a partire dal 02/05/2018, per un anno, fino al 02/05/2019 alle ore 00:00
Qualità del servizio	Con garanzia di consegna
WS endpoint	<a href="http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={...}">http://www.smartcityplatform.enea.it/UrbanDatasetGateway/push?urbandataset={...}</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	<a href="http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/">http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/</a>
<b>Comunicazione fra SCP e WebGis</b>	
Pattern	Request Response
Autenticazione	-
Sicurezza	HTTPS
Trasporto	La SCP espone Web Service. Il WebGIS periodicamente fa polling verso la SCP per ricevere l'aggiornamento
<b>Dati</b>	<b>Urban Dataset "Smart Building Record"</b>
Periodicità	On demand, a partire dal 01/05/2018 ore 00:00
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://smartcityplatform.enea.it/UrbanDatasetGateway/basicrequest?urbandataset={...}">http://smartcityplatform.enea.it/UrbanDatasetGateway/basicrequest?urbandataset={...}</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	<a href="http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/">http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/</a>
<b>Dati</b>	<b>Urban Dataset "Anomalie"</b>
Periodicità	On demand, a partire dal 01/05/2018 ore 00:00
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://smartcityplatform.enea.it/UrbanDatasetGateway/basicrequest?urbandataset={...}">http://smartcityplatform.enea.it/UrbanDatasetGateway/basicrequest?urbandataset={...}</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	<a href="http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/">http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/</a>
<b>Dati</b>	<b>Urban Dataset "Average Consumption"</b>
Periodicità	On demand, a partire dal 01/05/2018 ore 00:00
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://smartcityplatform.enea.it/UrbanDatasetGateway/basicrequest?urbandataset={...}">http://smartcityplatform.enea.it/UrbanDatasetGateway/basicrequest?urbandataset={...}</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	<a href="http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/">http://smartcityplatform.enea.it/casestudies/dss-smartbuilding/json/templates/</a>

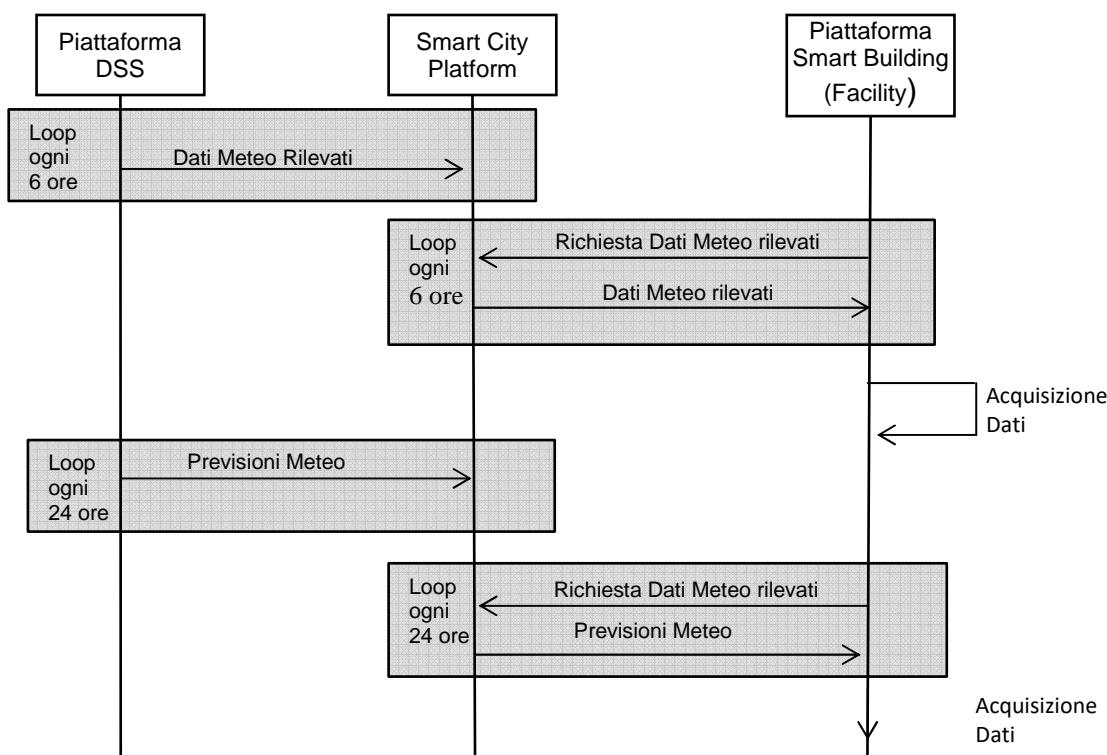
**Tabella 6 - Caso studio Smart Buiding - WebGis**

### 1.2.2 Case Study “DSS to Smart Building”

Il caso studio specifico in oggetto intende individuare i requisiti necessari per permettere il flusso dati che, tramite la Smart City Platform, dovrebbe avvenire:

- dalla Solution verticale “DSS”: sistema di supporto alle decisioni per le infrastrutture critiche
- alla Solution verticale “Smart Building”: piattaforma software che raccoglie i dati relativi al monitoraggio energetico (elettrico e termico) di un complesso di edifici, allo scopo di monitoraggio, diagnostica e attuazione per una corretta gestione energetica;

Tale flusso dati riguarda l’invio di dati meteo (rilevazione della situazione e previsione) e ha lo scopo di permettere allo Smart Building di migliorare le logiche di gestione energetica sulla loro base.



**Figura 3 - diagramma di sequenza del caso studio dati meteo per Smart Building da DSS (non sono mostrati gli eventuali Ack)**

I seguenti sotto-paragrafi mostrano tutti i requisiti necessari per implementare il caso studio in oggetto dal punto di vista dell’Interoperabilità, ovvero la struttura degli Urban Dataset (par. 1.2.2.1) e le caratteristiche dei vari stati di interoperabilità (par. 0) nella forma sintetica definita dalle tabelle mostrate nel capitolo 1.2.1 relativo allo Case Study “SmartBuilding to WebGIS”.

#### 1.2.2.1 Urban Dataset coinvolti

In questo caso studio vengono scambiati due Urban Dataset: “Dati Meteo rilevati” e “Previsione Meteo”.

L’Urban Dataset “Observed weather data” (Dati meteo rilevati) viene inviato, alla Smart Platform, ogni 6 ore e contiene dati derivanti dalle rilevazioni relative al monitoraggio ambientale delle seguenti grandezze: precipitazioni, temperature, portata idrica dei corsi d’acqua, per quello che riguarda la regione geografica di competenza della SCP.

La Smart Building Platform invia la richiesta dei dati meteo rilevati: a questa però non interessano tutti i dati disponibili, ma quelli registrati dalle centraline meteo più vicini (per esempio quelle presenti in un

raggio di 10 km). A tale scopo dovrà inviare non una semplice request, ma una “searching request” con centro e raggio della zona di interesse.

Nell’Urban Dataset fornito in risposta, a ogni *riga* (cioè ogni insieme di rilevamento forniti) verranno associate le coordinate della stazione meteo. Non è detto che ogni stazione fornisca tutti i dati (alcune potranno, per esempio, fornire solo i dati della temperatura, ecc.): in tal caso, per i dati mancanti viene comunque riportata la coppia nome - valore con valore NULL. Questo rende evidente che c’è, da parte del mittente, la consapevolezza di quel tipo di informazione, anche se in alcuni casi non la si può fornire, mentre, in caso contrario, l’opzione di non inserire il dato potrebbe essere interpretata come libertà di scegliere, a propria discrezione, se dare l’informazione o no. Inoltre, a livello di validazione, questa scelta consente di fare dei controlli in maniera più semplice e immediata.

Urban Dataset		Dati meteo rilevati	
Scopo	Fornire situazione meteo		
Aggregazione spaziale	City		
Categoria / Sottocategoria	Built environment / Land use and management		
Aggregazione temporale	Valori istantanei / totali (Nota: le precipitazioni saranno dati totali, mentre le temperatura saranno istantanee)		
SPECIFICAZIONE DELL’URBAN DATASET			
UrbanDatasetId	1.0-ObservedWeatherData		
UrbanDatasetUri	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#ObservedWeatherData">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#ObservedWeatherData</a>		
UrbanDatasetName	Observed weather data		
CONTESTUALIZZAZIONE DELL’URBAN DATASET			
	Unità di Misura:	Formato	Commenti / Esempio
Timestamp	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell’Urban Dataset
Timezone	Adimensionale	String	Es. UTC, UTC+01, UTC-05
Coordinates	Adimensionale	WGS84 (World Geodetic System)	Coordinate che identificano lo Smart Building di riferimento (a partire dal quale si calcolerà il raggio per selezionare le stazioni meteo da cui ricevere i dati)
Producer	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
PROPRIETÀ DELL’URBAN DATASET			
<i>(una riga per ogni Stazione Meteo presente nel raggio di 10 km)</i>			
	Unità di Misura:	Formato	Descrizione / Esempio
Coordinates	Adimensionale	WGS84 (World Geodetic System)	Coordinate per geolocalizzare l’area geografica a cui si riferiscono i dati, se più specifica rispetto quella macro
Start Period	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di inizio delle rilevazione dei dati (es. 26-09-2017T00:00:00)
End Period	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di fine delle rilevazione dei dati (es. 26-09-2017T23:59:59)
Meteo Station	Adimensionale	Stringa	Nome della stazione Meteo che ha fornito i dati (per es “Bracciano”)
Rainfall	mm	Double	Dati precipitazioni avvenute nell’intervallo
Temperature	°C	Double	Temperatura rilevata all’orario di endPeriod
Flow	m	Double	Portata idrica dei corsi d’acqua rilevata all’orario di endPeriod

**Tabella 7 - Definizione dell’Urban Dataset “Dati meteo”**

L’Urban Dataset “Weather forecast” (Previsioni Meteo) viene inviato, alla Smart Platform, ogni 24 ore e contiene dati di diverso tipo che il servizio meteorologico fornisce relativamente alle previsioni di pioggia a medio termine (“forecast” 6-48 ore). Il dominio territoriale di applicazione per le previsioni meteo è un’area

di 100 km intorno al punto definito dalle coordinate e vengono fornite le quantità di precipitazioni previste, come mm/h, come lista di 8 valori a intervalli di 6 ore (coprendo così le successive 48 ore).

Quindi, analogamente a quanto detto per i dati meteo rilevati, verrà inviata dallo Smart building una Searching Request con punto di interesse e raggio di 100 km attorno a esso.

Urban Dataset		Previsioni meteo	
Scopo	Fornire previsioni meteo relative alla pioggia		
Aggregazione spaziale	City		
Categoria / Sottocategoria	Built environment / Land use and management		
Aggregazione temporale	Valori previsti		
SPECIFICAZIONE DELL'URBAN DATASET			
UrbanDatasetId	scps-ontology-1.0.owl#WeatherForecast		
UrbanDatasetUri	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WeatherForecast		
UrbanDatasetName	Weather forecast		
CONTESTUALIZZAZIONE DELL'URBAN DATASET			
<i>Una riga per ognuno degli intervalli pubblicati (8 righe distanziate di sei ore l'una dall'altra)</i>			
	Unità di Misura:	Formato	Descrizione / Esempio
Timestamp	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell'Urban Dataset
Timezone	Adimensionale	String	Es. UTC, UTC+01, UTC-05
Coordinates	Adimensionale	WGS84 (World Geodetic System)	Coordinate che identificano l'area geografica macro di riferimento per tali dati <i>NOTA: le previsioni saranno valide in un intervallo di 240 x 240 Km</i>
Producer	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
PROPRIETÀ DELL'URBAN DATASET			
<i>NOTA: Ogni previsione per una coordinata specificata, sarà una lista di otto righe, che in totale copriranno le 48 ore</i>			
	Unità di Misura:	Formato	Descrizione / Esempio
Coordinates	Adimensionale	WGS84 (World Geodetic System)	Coordinate per geolocalizzare l'area geografica a cui si riferiscono i dati, se sono presenti previsioni relative a diverse aree specifica rispetto quella macro (nel qual caso, il dato presentato a livello macro sarà il centro geometrico di tali aree)
Start Period	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di inizio validità previsioni dei dati (es. 26-09-2017T00:00:00)
End Period	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data di fine validità previsioni (es. 26-09-2017T06:00:00) – 6 ore dopo la data d'inizio
Rainfall	mm/h	double	Dati precipitazioni previste

**Tabella 8 - Definizione dell'Urban Dataset "Dati meteo"**

### 1.2.2.2 Tabella sintetica

Caso studio:	Smart Building - DSS
<b>Obiettivo</b>	Gestione energetica sulla base delle condizioni meteo previste.
<b>Attore 1</b>	DSS
<b>Ruolo</b>	Acquisizione da fonte esterna e invio alla SCP dei dati monitoraggio ambientale e meteo relativi alla zona geografica d'interesse
<b>Attore 2</b>	Smart Building
<b>Ruolo</b>	Ricezione degli Urban Dataset e ottimizzazione delle logiche di gestione energetica di conseguenza
<b>Smart City Platform</b>	Recupero dei dati meteo e trasmissione allo Smart Building
<b>Urban Dataset</b>	Dati meteo rilevati
Aggregazione temporale/spaziale	Valore medio / Facility Aggregation
Policy Urban Dataset	Pubblico
Proprietà del dato	Open
Disponibilità del dato	Open
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#ObservedWeatherData">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#ObservedWeatherData</a>
<b>Urban Dataset</b>	Previsioni meteo
Aggregazione temporale/spaziale	Valore medio / Facility Aggregation
Policy Urban Dataset	Pubblico
Proprietà del dato	Open
Disponibilità del dato	Open
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#ForecastWeather">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#ForecastWeather</a>
<b>Start – End Collaboration</b>	dal 01/05/2017 ore 00:00, fino al da 01/05/2018 ore 00:00
	<b>Credenziali di accesso</b>
Registrazione Utente	Manuale, da parte dell'amministratore, nella versione prototipale della SCP
Credenziali Smart Building	<i>Username:</i> smartbuilding.admin@enea.it <i>Password:</i> [*****]
Credenziali DSS	<i>Username:</i> dss.admin@enea.it <i>Password:</i> [*****]
Credenziali SCP per accesso a Smart Building	<i>Username:</i> smartcityplatform@enea.it <i>Password:</i> [*****]
	<b>Trasporto e formato</b>
<b>Comunicazione fra SB e SCP</b>	
Pattern	Request Respone
Autenticazione	-
Sicurezza	HTTPS
Trasporto	La SCP espone Web Service Rest su cui riceve la request per entrambi gli Urban Dataset.
<b>Dati</b>	Urban Dataset "Dati Meteo rilevati"
Periodicità	una volta ogni sei ore, a partire dalle 00:00 01/05/2017, per un anno, fino al 03/05/2018 alle ore 00:00.
Qualità del servizio	Senza garanzia di consegna
WS Endpoint	<a href="http://www.smartcityplatform.enea.it/getaway/searchingrequest?urbandataset={...}">http://www.smartcityplatform.enea.it/getaway/searchingrequest?urbandataset={...}</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
<b>Dati</b>	Urban Dataset "Previsioni meteo"
Periodicità	una volta ogni 24 ore, a partire dalle 00:00 01/05/2017, per un anno, fino al 03/05/2018 alle ore 00:
Qualità del servizio	Senza garanzia di consegna
WS Endpoint	<a href="http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset={...}">http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset={...}</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
	<b>Comunicazione fra SCP e WebGis</b>
Pattern	Publish Subscribe
Autenticazione	-
Sicurezza	HTTPS

Trasporto	L'SCP espone Web Service a cui il WebGis invia I dati
<b>Dati</b>	Urban Dataset "Dati Meteo rilevati"
Periodicità	Non appena viene ricevuto un nuovo Urban Dataset
Qualità del servizio	Consegna garantita
Endpoint Urban Dataset 1	<a href="http://www.cipcastcasaccia.it/urbandataset/">http://www.cipcastcasaccia.it/urbandataset/</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
<b>Dati</b>	Urban Dataset "Previsioni meteo"
Periodicità	Non appena viene ricevuto un nuovo Urban Dataset
Qualità del servizio	Senza garanzia di consegna
WS Endpoint	<a href="http://www.cipcastcasaccia.it/urbandataset/">http://www.cipcastcasaccia.it/urbandataset/</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	

**Tabella 9- Caso studio Smart Buiding - WebGis**



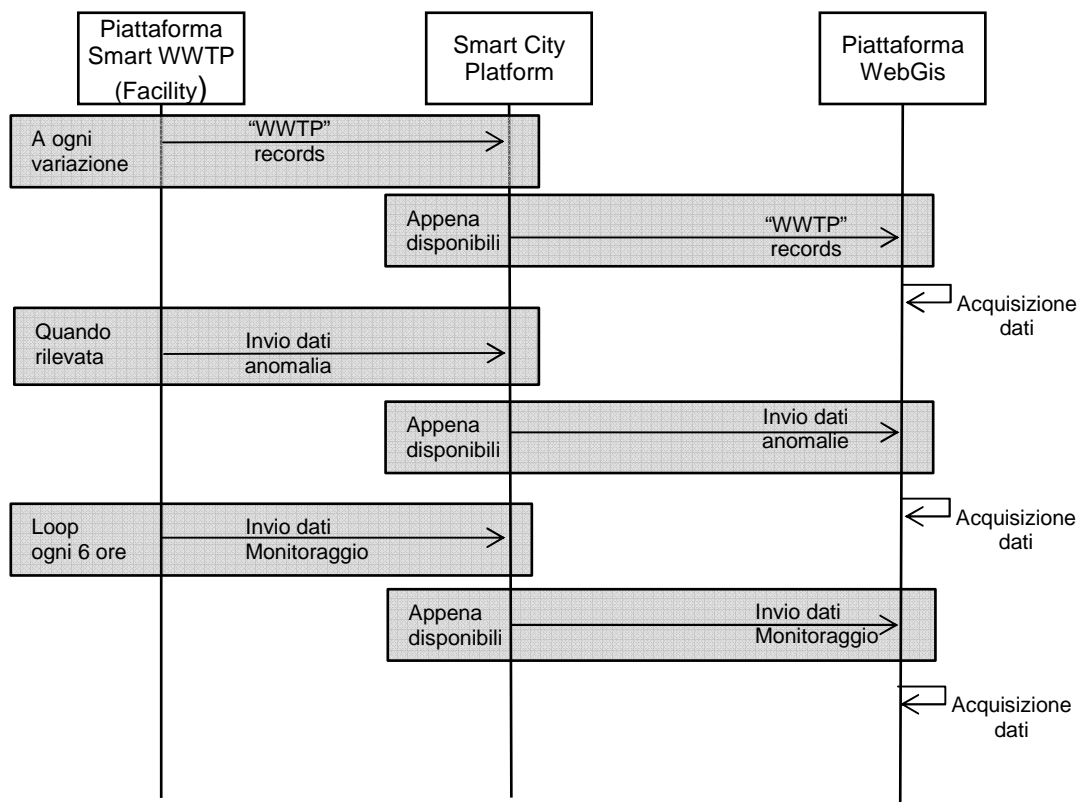
### 1.2.3 Case Study “WWTP to WebGIS”

Il caso studio specifico in oggetto intende individuare i requisiti necessari per permettere il flusso dati che, tramite la Smart City Platform, dovrebbe avvenire

- dal Contesto Applicativo “Gestione acque” (Smart WWTP – Waste Water Treatment Plant)
- allo strumento di georeferenziazione dei dati “WebGIS”.

per permettere il monitoraggio dei dati generati del contesto applicativo WWTP tramite visualizzazione su mappa geo-referenziata.

Lo Smart WWTP è un sistema di gestione e controllo di un impianto per la depurazione delle acque reflue ed è in grado di fornire la misura dei volumi trattati di acqua di scarico e dei relativi consumi energetici, nonché di fornire informazioni su situazioni di anomale che si possono verificare nell’impianto di depurazione. Una parte importante di monitoraggio e controllo deve essere residente all’interno della rete (perché, in caso di malfunzionamento, l’impianto deve continuare a funzionare), ma si può pensare di centralizzare una parte di monitoraggio, in modo che un’implementazione centralizzata possa recuperare i dati da un certo numero di depuratori nella stessa area geografica e li metta a disposizione, come Urban Dataset, alla Smart Platform. In questo caso studio, perciò, si farà l’ipotesi di essere in questo caso più generale, di Smart WWTP in grado di monitorare più di un impianto di depurazione.



**Figura 4 - Diagramma di sequenza del caso studio visualizzazione dati Gestione Acque tramite WebGis (non sono evidenziati gli Ack)**

### 1.2.3.1 Urban Dataset Coinvolti

In questo caso studio vengono scambiati tre Urban Dataset.

L’Urban Dataset “WWTP Records” è anagrafico (statico) e fornisce una descrizione univoca degli impianti di depurazione monitorati dal contesto applicativo Smart WWTP, per evitare di dover reinviare, ogni volta le coordinate dei singoli impianti. I dati trasportati sono definiti nella seguente tabella:

<b>Urban Dataset</b>	WWTP Records		
<b>Scopo</b>	Anagrafica impianti di depurazione: fornisce una descrizione univoca degli impianti di depurazione monitorati nel contesto applicativo verticale Smart Building		
<b>Aggregazione spaziale</b>	Facility		
<b>Categoria / Sottocategoria</b>	Water and wastewater / Wastewater management		
<b>Aggregazione temporale</b>	Statico		
<b>SPECIFICAZIONE DELL’URBAN DATASET</b>			
<b>UrbanDatasetId</b>	1.0-WWTPRecords		
<b>UrbanDatasetURI</b>	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WWTPRecords		
<b>UrbanDatasetName</b>	WWTP Records		
<b>CONTESTUALIZZAZIONE DELL’URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell’Urban Dataset
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate del centro geometrico della rete di Depuratori es. “42.04” ; “12.30”
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL’URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate per geolocalizzare il centro dell’impianto (es. “42.04” ; “12.30”)
<b>PlantID</b>	Adimensionale	String	Identificatore dell’impianto di depurazione monitorato (es. 1). Questo diventerà il codice di riferimento per tutti gli Urban Dataset associati a questo impianto.
<b>Plant Name</b>	Adimensionale	String	Nome dell’impianto (es. “Depuratore Casalecchio”)

Tabella 10 – Tabella di sintesi dell’Urban Data “WWTP Records”

L’Urban Dataset “Water Treatment Anomaly” (Anomalia del depuratore) viene inviato in maniera asincrona, quando eventuali anomalie vengono rilevate, quali rotture o malfunzionamenti, al fine di visualizzarle sulla mappa (l’utility che gestisce il depuratore potrebbe subito vedere gli allarmi sulla mappa).

<b>Urban Dataset</b>	Anomalie del depuratore		
<b>Scopo</b>	Anomalia verificatasi nel depuratore (per visualizzazione su Mappa)		
<b>Aggregazione spaziale</b>	Facility		
<b>Categoria / Sottocategoria</b>	Water and wastewater / Wastewater management		
<b>Aggregazione temporale</b>	Valori istantanei		
<b>SPECIFICAZIONE DELL’URBAN DATASET</b>			
<b>UrbanDatasetId</b>	1.0-WaterTreatmentAnomaly		
<b>UrbanDatasetURI</b>	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WaterTreatmentAnomaly		
<b>UrbanDatasetName</b>	Water Treatment Anomaly		
<b>CONTESTUALIZZAZIONE DELL’URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Timestamp</b>	Adimensionale	Date-Time	Tempo di generazione dell’Urban Dataset

		gg-mm-aaaaThh:mm:ss	
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate che identificano la posizione del depuratore che ha generato l'anomalia
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Start Period</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data/Ora in cui si è verificata l'anomalia
<b>End Period</b>	Adimensionale	Date - Time gg-mm-aaaaThh:mm:ss	Valore coincidente al precedente: Ora in cui si è verificata l'anomalia
<b>Cause ID</b>	Adimensionale	Int	Codice dell'anomalia (per es 2 ). La decodifica è mostrata nella Tabella 12
<b>Cause</b>	Adimensionale	String	Es. Rottura componente

**Tabella 11 – Definizione Urban Dataset “anomalie del depuratore”**

Il parametro *CauseId* fa riferimento a una lista di cause univoca, a cui questo Urban Dataset fa riferimento (riportata nella tabella successiva). Si noti che la lista sarà esterna all'ontologia e lo schema vi farà riferimento come CodeList.

Cause_Id	Anomalia
1	Rottura componentistica impianto di depurazione
2	Malfunzionamento sistema di misura
3	Malfunzionamento sistema di controllo
4	Malfunzionamento impianto elettrico
5	Evento meteorologico rilevante reale, in termini di durata e/o intensità
6	Evento meteorologico rilevante previsto

**Tabella 12 - Tabella delle anomalie WWTP**

L'Urban Dataset “Water Treatment Data” (Dati del depuratorie) contiene i dati relativi al consumo energetico e al volume di acqua trattati, nelle ultime sei ore di funzionamento. I dati vengono inviati allo scadere delle sei ore.

<b>Urban Dataset</b>	Dati del depuratore		
<b>Scopo</b>	Consumo elettrico medio giornaliero per m3 depurato e Flusso di acqua totale per m3 (per visualizzazione su Mappa)		
<b>Aggregazione spaziale</b>	Facility ( <i>singolo depuratore o rete di depuratori monitorati dallo Smart WWTP</i> )		
<b>Categoria / Sottocategoria</b>	Water and wastewater / Wastewater management		
<b>Aggregazione temporale</b>	Valori totali		
<b>SPECIFICAZIONE DELL'URBAN DATASET</b>			
<b>UrbanDatasetId</b>	1.0-WaterTreatmentData		
<b>UrbanDatasetUri</b>	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WaterTreatmentData">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WaterTreatmentData</a>		
<b>UrbanDatasetName</b>	Water Treatment Data		
<b>CONTESTUALIZZAZIONE DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell'Urban Dataset
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate che identificano la posizione del centro della rete di depuratori
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Start Period</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data/Ora in cui di inizio del periodo di 6 ore (es- 03-10-2017 08:00:00)
<b>End Perid</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data/Ora in cui di fine del periodo di 6 ore (es- 03-10-2017 14:00:00)

<b>WaterLoad</b>	mm <sup>3</sup> /giorno	Double	Flusso acqua reflua specifica. Es. 24.000
<b>ElectricConsumption</b>	kWh/m <sup>3</sup>	Double	0,4

**Tabella 13 –Definizione Urban Dataset “Dati del depuratore”**

1.2.3.2 Tabella sintetica

<b>Caso studio:</b>	<b>Smart WWTP - WebGIS</b>
<b>Obiettivo</b>	Monitoraggio del contesto applicativo Smart WWTP
<b>Attore 1</b>	Smart Waste Water Treatment Plant (Smart WWTP)
<b>Ruolo</b>	Monitoraggio di anomalie, consumi energetici e portata e, su questa base, generazione ed esportazione, dal proprio sistema di persistenza locale, degli Urban Dataset nel formato comune
<b>Attore 2</b>	WebGis
<b>Ruolo</b>	Ricezione degli Urban Dataset e visualizzazione degli Urban Dataset su mappa
<b>Smart City Platform</b>	Recupero Urban Dataset dallo Smart WWTP, loro memorizzazione nel DB (persistenza) e pubblicazione degli Urban Dataset per il WebGis
<b>Urban Dataset</b>	WWTP Records
Aggregazione temporale/spaziale	Statico / Facility
Policy dell’Urban Dataset	Privato
Proprietà del dato	Utility che gestisce la depurazione delle acque
Disponibilità del dato	Disponibile solo all’Utility
Riferimento nell’Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WWTPRecords">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WWTPRecords</a>
<b>Urban Dataset</b>	WWTP Anomalies
Aggregazione temporale/spaziale	Valori Istantanei / Facility
Policy dell’Urban Dataset	Privato
Proprietà del dato	Utility che gestisce la depurazione delle acque
Disponibilità del dato	Disponibile solo all’Utility
Riferimento nell’Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WaterTreatmentAnomaly">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WaterTreatmentAnomaly</a>
<b>Urban Dataset</b>	“WWTP Data” Dati del depuratore (Consumo elettrico e Portata)
Aggregazione temporale/spaziale	Valori Totali / Facility
Policy dell’Urban Dataset	Pubblico
Proprietà del dato	Open
Disponibilità del dato	A chiunque lo richieda
Riferimento nell’Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WaterTreatmentData">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#WaterTreatmentData</a>
<b>Start – End Collaboration</b>	dal 01/05/2018 ore 00:00, fino al da 01/05/2018 ore 00:00
	<b>Credenziali di accesso</b>
Registrazione Utente	Manuale, da parte dell’amministratore, nella versione prototipale della SCP
Credenziali Smart WWTP	Username: smartwwtp.admin@enea.it Password: [*****]
Credenziali WebGis	Username: : webgis.admin@enea.it Password: [*****]
Credenziali SCP per accesso a Smart Building	Username: smartcityplatform@enea.it Password: [*****]
	<b>Trasporto e formato</b>
<b>Comunicazione fra WWTP e SCP</b>	
Pattern	Push
Autenticazione	-
Sicurezza	HTTPS o SSL
Trasporto	L’SCP espone Web Service Rest tipo a cui lo Smart WWTP invia i dati
<b>Dati</b>	Urban Dataset “Smart WWTP Record”
Periodicità	una sola volta con garanzia a partire da 01/05/2018 ore 00:00 e a ogni modifica, per un anno, fino al da 01/05/2019 alle ore 00:00
Qualità del servizio	Con garanzia di consegna
WS endpoint	<a href="http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset=1.0-SmartWWTPRecord">http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset=1.0-SmartWWTPRecord</a>
Formato	JSON

Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
<b>Dati</b>	Urban Dataset “Water Treatment Anomaly”
Periodicità	Ogni volta che si verifica un'anomalia a partire dalle ore 00:00 del 02/05/2018, per un anno, fino al 02/05/2018 alle ore 00:00.
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset=1.0-WaterTreatmentAnomaly">http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset=1.0-WaterTreatmentAnomaly</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
<b>Dati</b>	Urban Dataset “Water Treatment Data”
Periodicità	una volta ogni 6 ore a partire dalle ore 00:00 del 02/05/2017, per un anno, fino al 02/05/2018 alle ore 00:00.
Qualità del servizio	Senza garanzia di consegna
WS Endpoint	<a href="http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset=1.0-WaterTreatmentData">http://www.smartcityplatform.enea.it/getaway/basicrequest?urbandataset=1.0-WaterTreatmentData</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
<b><u>Comunicazione fra WWTP e WebGis</u></b>	
Pattern	Push
Autenticazione	-
Sicurezza	HTTPS
Trasporto	Il WebGis espone Web Service. Quando l'SCP riceve l'aggiornamento lo comunica al WebGis
<b>Dati</b>	Urban Dataset “Smart WWTP Record”
Periodicità	Non appena viene ricevuto un nuovo Urban Dataset
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://www.webgiscasaccia.it/urbandataset/1121/">http://www.webgiscasaccia.it/urbandataset/1121/</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
<b>Dati</b>	Urban Dataset “Smart WWTP Record”
Periodicità	Non appena viene ricevuto un nuovo Urban Dataset
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://www.webgiscasaccia.it/urbandataset/1121/">http://www.webgiscasaccia.it/urbandataset/1121/</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	
<b>Dati</b>	Urban Dataset “Dati del depuratore”
Periodicità	Non appena viene ricevuto un nuovo Urban Dataset
Qualità del servizio	Senza garanzia di consegna
WS Endpoint	<a href="http://www.webgiscasaccia.it/urbandataset/1121/">http://www.webgiscasaccia.it/urbandataset/1121/</a>
Formato	JSON
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a>
Link al template	

**Tabella 14- Caso studio “WWTP to DSS”**

### 1.3 Casi Studio “One Way”

Questo paragrafo descrive i Casi Studio per la Smart City Platform in cui la piattaforma non fa da tramite per le comunicazioni fra due verticali, ma da ricevente delle informazioni (al fine di persistenza e visualizzazione su un cruscotto). Si configura, dunque, un dialogo a due fra Verticale e Smart City Platform

I casi studio particolari sono i seguenti

1. “Smart Lighting to PELL” (monitoraggio consumi elettrici in ambito PELL).
2. “Smart Home to SCP”

#### 1.3.1 Case Study “Smart Lighting to PELL”

Questo caso studio *particolare* può essere applicato a due situazioni reali molto diverse:

1. Una generica solution verticale Smart Lighting invia i consumi elettrici alla piattaforma di città SCP;
2. Una municipalità, o gestore della Illuminazione Pubblica (IP), invia i consumi elettrici alla piattaforma PELL (Public Energy Living Lab, piattaforma che raccoglie i consumi elettrici da tutte le municipalità aderenti per offrire un servizio di monitoraggio e benchmarking).

In altre parole, i ruoli del caso studio sono sempre due (a differenza dei tre visti nei precedenti casi studio) ma gli attori possono variare dipendentemente dal contesto di applicazione.

Le Smart City Platform Specification hanno dunque una duplice valenza: sono utilizzabili sia per una piattaforma ICT di Città, sia per una piattaforma nazionale come il PELL (2).

Per descrivere il caso studio in oggetto prenderemo il flusso dati che avviene tra

- una solution verticale “Smart Lighting” (o Gestore IP equivalente);
- una Piattaforma di acquisizione (PELL o Smart City Platform);

per permettere il recupero dei consumi elettrici dagli impianti di illuminazione pubblica.

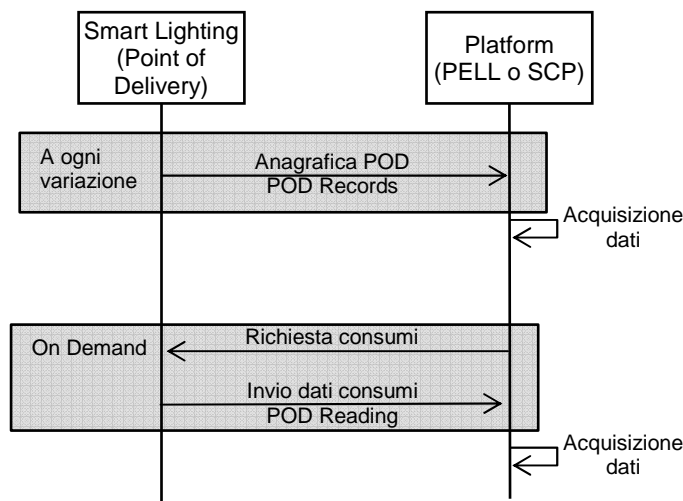


Figura 5 - Diagramma di sequenza UML del caso studio “Smart Lighting to PELL”

##### 1.3.1.1 Urban Dataset Coinvolti

In questo caso studio vengono scambiati due Urban Dataset.

L'Urban Dataset "POD List" è anagrafico (statico) e fornisce l'elenco dei Point of Delivery (POD<sup>3</sup>) che vengono monitorati, associandoli alle loro coordinate geografiche. Ciò permette, quando si inviano i dati, di inviare il solo ID del POD, senza dover, ogni volta, re-inviare le coordinate geografiche.

Si noti che l'infrastruttura del PELL prevede l'invio di una "scheda censimento," in formato Foglio di Calcolo, notevolmente complessa, che per il momento non viene trattata da questo caso studio.

Urban Dataset	POD List		
<b>Scopo</b>	Anagrafica dei punti di prelievo (POD): fornisce una descrizione univoca degli impianti di illuminazione monitorati nel contesto applicativo verticale Smart Lighting		
<b>Aggregazione spaziale</b>	Item		
<b>Categoria / Sottocategoria</b>	Energy / Energy demand		
<b>Aggregazione temporale</b>	Statico		
<b>SPECIFICAZIONE DELL'URBAN DATASET</b>			
<b>UrbanDatasetId</b>	<a href="#">1.0-PODList</a>		
<b>UrbanDatasetUri</b>	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODList">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODList</a>		
<b>UrbanDatasetName</b>	POD list		
<b>CONTESTUALIZZAZIONE DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell'Urban Dataset
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate del centro geometrico della rete di POD es. "42.04" ; "12.30"
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
<i>Nota: vi sarà una riga per ogni POD monitorato</i>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>POD_ID</b>	Adimensionale	String	Codice POD che identifica univocamente il punto di prelievo
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate per geolocalizzare il punto di prelievo (es. "42.04" ; "12.30")

**Tabella 15- Tabella di sintesi dell'Urban Data "POD List"**

L'Urban Dataset "POD Reading" contiene i dati relativi alle letture di consumo energetico dell'impianto pubblico d'illuminazione. Il set di dati in esso è stato definito all'interno dell'attività legate al PELL.

Urban Dataset	POD Reading		
<b>Scopo</b>	Monitorare il consumo elettrico di un sistema di Illuminazione Pubblica		
<b>Aggregazione spaziale</b>	City		
<b>Categoria / Sottocategoria</b>	Energy / Energy demand		
<b>Aggregazione temporale</b>	Valori totali		
<b>SPECIFICAZIONE DELL'URBAN DATASET</b>			
<b>UrbanDatasetId</b>	<a href="#">1.0-PODReading</a>		
<b>UrbanDatasetUri</b>	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODReading">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODReading</a>		
<b>UrbanDatasetName</b>	POD reading		
<b>CONTESTUALIZZAZIONE DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell'Urban Dataset

<sup>3</sup> Col termine "punto di prelievo" o "point of delivery" (POD), generalmente ci stiamo riferendo ai punti fisici di prelievo e di energia elettrica dalla rete elettrica, per l'alimentazione di illuminazione pubblica. Si noti che, in Italia, a ogni punto di prelievo è associato un **codice POD** univoco che lo identifica

<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate che identificano la posizione del centro dell'insieme di POD monitorati
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>POD_ID</b>	Adimensionale	String	Codice POD che identifica univocamente il punto di prelievo
<b>Start Perid</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data/Ora in cui di fine del periodo di rilevazione (es- 03-10-2017 14:00:00)
<b>End Perid</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Data/Ora in cui di fine del periodo di rilevazione (es- 03-10-2017 14:30:00)
<b>Active Energy</b>	kWh	Double	Energia attiva consumata
<b>Voltage Phase1</b>	V	Double	Tensione della fase R
<b>Voltage Phase2</b>	V	Double	Tensione della fase S
<b>Voltage Phase3</b>	V	Double	Tensione della fase T
<b>Current Line1</b>	A	Double	Corrente della linea 1
<b>Current Line2</b>	A	Double	Corrente della linea 2
<b>Current Line3</b>	A	Double	Corrente della linea 3
<b>Active Power Phase 1</b>	kW	Double	Potenza attiva fase S
<b>Active Power Phase2</b>	kW	Double	Potenza attiva fase R
<b>Active Power Phase3</b>	kW	Double	Potenza attiva fase T
<b>Total Active power</b>	kW	Double	Potenza attiva totale
<b>Power Factor Phase 1</b>	Adimensionale	Double	Fattore di potenza R
<b>Power Factor Phase 2</b>	Adimensionale	Double	Fattore di potenza S
<b>Power Factor Phase 3</b>	Adimensionale	Double	Fattore di potenza T
<b>Reactive Power Phase 1</b>	kVAR	Double	Potenza reattiva R
<b>Reactive Power Phase 2</b>	kVAR	Double	Potenza reattiva S
<b>Reactive Power Phase 3</b>	kVAR	Double	Potenza reattiva T
<b>Total Reactive Power</b>	kVAR	Double	Potenza reattiva totale
<b>Apparent Power Phase 1</b>	kVA	Double	Potenza apparente R
<b>Apparent Power Phase 2</b>	kVA	Double	Potenza apparente S
<b>Apparent Power Phase 3</b>	kVA	Double	Potenza apparente T
<b>Total Apparent Power</b>	kVA	Double	Potenza apparente totale
<b>Reactive Energy</b>	kVARh	Double	Energia reattiva (inductiveEnergy + capacitiveEnergy)

Tabella 16 - Tabella di sintesi dell'Urban Data "POD reading"



### 1.3.1.2 Tabella sintetica per la descrizione del caso studio

Caso studio:	Smart Lighting to PELL
<b>Obiettivo</b>	Invio dei consumi elettrici (POD Reading) da Smart Lighting a Piattaforma PELL (o SCP)
<b>Attore 1</b>	Smart Lighting (SL)
<b>Ruolo</b>	Acquisizione dei consumi elettrici ed esportazione, dal proprio sistema locale, tramite formato comune Urban Dataset
<b>Attore 2</b>	Platform (piattaforma PELL o SCP)
<b>Ruolo</b>	Ricezione dei consumi elettrici per servizi di monitoraggio e benchmarking.
<b>Urban Dataset</b>	POD Records
Aggregazione temporale/spaziale	Statico / Item
Policy dell'Urban Dataset	Privato
Proprietà del dato	...
Disponibilità del dato	...
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODList">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODList</a>
<b>Urban Dataset</b>	POD Reading
Aggregazione temporale/spaziale	Valori medi / Item
Policy dell'Urban Dataset	Privato
Proprietà del dato	Smart Lighting
Disponibilità del dato	
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODReading">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#PODReading</a>
<b>Start – End Collaboration</b>	Dall'01/05/2018 ore 00:00, fino al da 02/05/2019 ore 00:00
Registrazione Utente	<b>Credenziali di accesso</b> Nel caso PELL esiste un meccanismo di registrazione alla Piattaforma e di adesione tramite invio di una Scheda Censimento descrivente l'infrastruttura IP (3). A questa fase di registrazione e validazione dell'infrastruttura IP segue l'invio periodico dell'Urban Dataset "POD Reading" relativo ai consumi elettrici quotidiani. Nel caso SCP la registrazione è manuale: avviene da parte dell'amministratore nella versione prototipale della SCP, tramite dashboard apposita, a valle di un bando di gara o accordo formale esterno tra le parti. In entrambi i casi, vi saranno le credenziali di accesso user+password per utilizzare il Web Service per l'invio dell'Urban Dataset "POD Reading".
Credenziali SL	<i>Username:</i> smart.lighting@comune-x.it <i>Password:</i> *****
Credenziali PELL (o SCP)	<i>Username:</i> pell@enea.it <i>Password:</i> [*****]
<b>Comunicazione fra SL e PELL (o SCP)</b>	<b>Trasporto e formato</b>
Pattern	REQUEST/RESPONSE
Autenticazione	Accesso con credenziali e token di accesso
Sicurezza	https e SSL/TLS?
Trasporto	
<b>Dati</b>	Urban Dataset "POD List"
Periodicità	una sola volta con garanzia a partire da 01/05/2018 ore 00:00 e a ogni modifica, per un anno, fino al da 01/05/2019 alle ore 00:00
Qualità del servizio	Con garanzia di consegna
WS endpoint (esempio)	<a href="http://www.smartlighting.comunex.it/ws/rest/UrbanDatasetGetaway">http://www.smartlighting.comunex.it/ws/rest/UrbanDatasetGetaway</a>
Formato	XML
Link allo schema XML	<a href="http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd">http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd</a>
Link al template XML	
<b>Dati</b>	Urban Dataset "POD Reading"
Periodicità	una volta al giorno alle ore 12:00
Qualità del servizio	Con garanzia di consegna
WS endpoint (esempio)	<a href="http://www.smartlighting.comunex.it/ws/rest/UrbanDatasetGetaway">http://www.smartlighting.comunex.it/ws/rest/UrbanDatasetGetaway</a>
Formato	XML
Link allo schema XML	<a href="http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd">http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd</a>

Link al template XML

[urbandataset-schema-1.0.xsd](#)  
<http://smartcityplatform.enea.it/casestudies/pell/xml/templates/PODReadingTemplate.xml>

**Tabella 17- Smart Lighting - PELL**

## 1.4 Casi Studio Particolari

Questo paragrafo descrive i Casi Studio per la Smart City Platform che, differendo sostanzialmente da quelli visti nel precedente paragrafo, sono definiti come “particolari”.

I casi studio particolari sono i seguenti

1. “Whatever” (trasmissione di dati generici non ancora formalizzati);
2. “Direct Communication” (per abilitare la comunicazione diretta tra due solution verticali).

### 1.4.1 Case Study “Whatever”

I casi studio che abbiamo visto finora riguardano specifiche solution verticali.

Il principio con cui sono costruite le Specifiche per la Smart City Platform è che possano essere flessibili e adottabili in maniera parziale/incrementale.

Risulta utile, perciò, fornire un caso studio generico, in cui vengono scambiati dati, non ancora formalizzati in uno specifico Urban Dataset documentato nell’Ontologia (Urban Dataset Reference Specification), dichiarandone uno generico. Questo Urban Dataset lascia, perciò, la libertà di inviare un insieme di dati, non interpretabili semanticamente, rispettando il rigore della struttura sintattica base dell’Urban Dataset di primo livello, cioè definita dall’unico schema generale, ma non del secondo livello, che, invece, definisce tutti i dati che compongono l’Urban Dataset in oggetto.

Il suo utilizzo implica una validazione più debole (e quindi un livello più basso di interoperabilità) e, dunque, un’adesione di livello “base” alle specifiche. Idealmente si pone come uso iniziale delle stesse, che poi dovrebbe/potrebbe evolvere verso l’uso di un Urban Dataset registrato nell’Ontologia.

Anche in questo caso, la persistenza dei dati sarà decisa in fase di definizione della collaborazione nel Registry ma si deve tenere ben presente che, essendo l’UD non registrato, i dati potrebbero essere inconsistenti e quindi è suggerito, per questo tipo di UD, l’assenza di persistenza presso la SCP.

N.B. questo Urban Dataset è utile nel caso di test preliminare per l’adozione del solo formato o per inviare dati generici, senza la necessità di una definizione semantica, o anche in assenza dell’Ontologia.

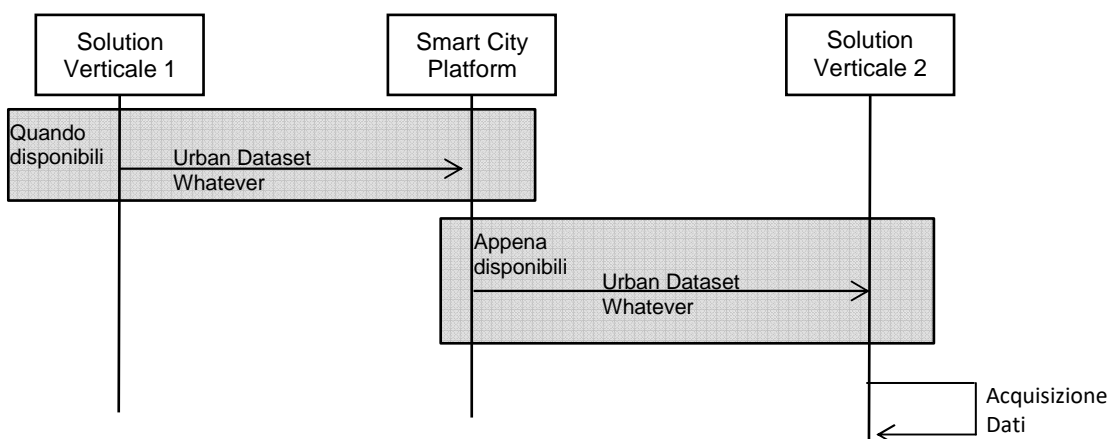


Figura 6 – Caso studio Whatever

1.4.1.1 Urban Dataset Coinvolti

<b>Urban Dataset</b>	Whatever		
<b>Scopo</b>	Consentire comunicazione di base con adesione minimale alle specifiche		
<b>Aggregazione spaziale</b>	Da definire per il singolo caso implementativo		
<b>Categoria / Sottocategoria</b>	Da definire per il singolo caso implementativo		
<b>Aggregazione temporale</b>	Da definire per il singolo caso implementativo		
<b>SPECIFICAZIONE DELL'URBAN DATASET</b>			
<b>UrbanDatasetId</b>	1.0-Whatever		
<b>UrbanDatasetUri</b>	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#Whatever		
<b>UrbanDatasetName</b>	Whatever		
<b>CONTESTUALIZZAZIONE DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell'Urban Dataset
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate del centro geometrico del sistema a cui si riferisce l'Urban Dataset
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Valore</b>	Da specificare	Da specificare	

Tabella 18 – Definizione Urban Dataset “Whatever”

### 1.4.1.2 Tabella sintetica per la descrizione del caso studio

Caso studio:	Whatever
<b>Obiettivo</b>	Scambio di dati non certificati nell'Ontologia utilizzando l'adesione "base" al formato Urban Dataset delle specifiche SCP
<b>Attore 1</b>	Solution Verticale 1 (SV1)
<b>Ruolo Attore 1</b>	Inviare un dato non codificato fra gli Urban Dataset presenti nell'Ontologia
<b>Attore 2</b>	Solution Verticale 2 (SV2)
<b>Ruolo Attore 2</b>	Ricevere un dato non codificato fra gli Urban Dataset presenti nell'Ontologia
<b>Smart City Platform</b>	Trasferire il dato fra le due solution
<b>Urban Dataset</b>	Whatever
Aggregazione temporale/spaziale	Non applicabile / Non applicabile
Policy dell'Urban Dataset	Accesso per il destinatario Solution verticale 2
Proprietà del dato	Da definire per il singolo caso implementativo
Disponibilità del dato	Da definire per il singolo caso implementativo
Riferimento nell'Ontologia	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#Whatever">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#Whatever</a>
<b>Start – End Collaboration</b>	Da definire per il singolo caso implementativo
	<b>Credenziali di accesso</b>
Registrazione Utente	Manuale
Credenziali SV1	<i>Username:</i> Da definire per il singolo caso implementativo <i>Password:</i> Da definire per il singolo caso implementativo
Credenziali SV2	<i>Username:</i> Da definire per il singolo caso implementativo <i>Password:</i> Da definire per il singolo caso implementativo
Credenziali SCP per accesso a SV2	<i>Username:</i> smartcityplatform@enea.it <i>Password:</i> MyPassword3
	<b>Trasporto e formato</b>
<b>Comunicazione fra PV1 e SCP</b>	
Pattern	PUSH
Autenticazione	-
Sicurezza	HTTPS
Trasporto	L'SCP espone Web Service Rest per la ricezione dell'Urban Dataset
<b>Dati</b>	Urban Dataset "Whatever"
Periodicità	Quando disponibile
Qualità del servizio	Da definire nel singolo caso implementativo
WS Endpoint	<a href="http://www.smartcityplatform.enea.it/ws/rest/getaway/">http://www.smartcityplatform.enea.it/ws/rest/getaway/</a>
Formato	XML o JSON (Da definire per il singolo caso implementativo)
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a> (JSON SCHEMA) <a href="http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd">http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd</a> (XML SCHEMA)
Link al template	
<b>Comunicazione fra WWTP e PV2</b>	
Pattern	Push
Autenticazione	-
Sicurezza	HTTPS
Trasporto	SV2 espone Web Service
<b>Dati</b>	Urban Dataset "Whatever"
Periodicità	Non appena viene ricevuto un nuovo Urban Dataset
Qualità del servizio	Da definire nel singolo caso implementativo
WS Endpoint	Da definire per il singolo caso implementativo
Formato	XML o JSON (Da definire per il singolo caso implementativo)
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a> (JSON Schema) <a href="http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd">http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd</a> (XML Schema)
Link al template	

Tabella 19- Caso studio Whatever

### 1.4.2 Case Study “Direct Communication”

Il caso studio particolare “Direct Communication” descrive la situazione in cui una prima Solution Verticale SV1 vuole accedere direttamente a una Solution Verticale SV2, senza inviare i dati alla SCP, ma richiedendo alla piattaforma centrale solo le informazioni necessarie a stabilire la comunicazione.

In alcuni casi, infatti, può non essere necessario utilizzare la SCP come “intermediario” per immagazzinare i dati ma è sufficiente instaurare lo scambio dati diretto tra le due solution verticali.

I passi fondamentali per riassumere questo caso studio particolare sono i seguenti:

- la Solution Verticale 1 carica sulla SCP l’Urban Dataset “Communication Information” che contiene il set di informazioni per connettersi direttamente al servizio offerto da SV2;
- la SCP immagazzina l’Urban Dataset di SV1 e lo rende disponibile alla ricerca, catalogandolo opportunamente (si ipotizza in una sezione “Direct Communication”);
- la Solution Verticale 2 interroga la SCP come fosse un “semplice” Registry e decide di recuperare le informazioni relative alla comunicazione diretta offerta da SV1;
- la SCP restituisce a SV2 l’insieme di dati necessari per instaurare la comunicazione con SV1;
- SV2 instaura la comunicazione diretta con SV1 effettuando le chiamate e ricevendo dati.

Si rende necessario, a questo punto, formalizzare un Urban Dataset “Communication Information” per descrivere l’insieme di dati necessari a instaurare la comunicazione, che si ipotizzano al momento essere i seguenti:

1. Denominazione Solution Verticale (di cui si sono richiesti i dati);
2. Descrizione testuale del dato offerto;
3. Pattern implementato per la comunicazione diretta;
4. Endpoint del Web Service;
5. URL dello Schema del Formato con cui si invieranno i dati;
6. Altri parametri liberi.

Questi dati devono essere precedentemente caricati da SV1 tramite apposito Urban Dataset “Communication Information” (si veda il seguente diagramma UML).

Chiunque avrà l’accesso all’Urban Dataset, potrà accedere al servizio offerto da SV2.

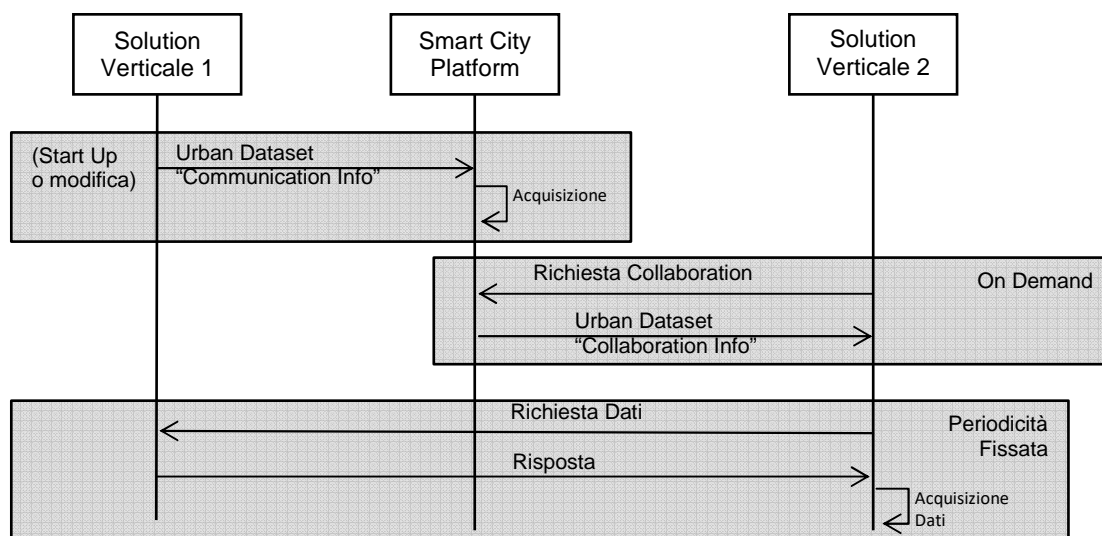


Figura 7 – Caso studio “Direct Communication”

### 1.4.2.1 Urban Dataset Coinvolti

<b>Urban Dataset</b>	Communication Information		
<b>Scopo</b>	Descrivere il set di parametri necessari per consentire di instaurare una comunicazione diretta tra due solution verticali senza usare la SCP come intermediario		
<b>Aggregazione spaziale</b>	Not applicable		
<b>Categoria / Sottocategoria</b>	Not applicable		
<b>Aggregazione temporale</b>	Statico		
<b>SPECIFICAZIONE DELL'URBAN DATASET</b>			
<b>UrbanDatasetId</b>	1.0-CommunicationInformation		
<b>UrbanDatasetUri</b>	http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#CommunicationInformation		
<b>UrbanDatasetName</b>	Communication Information		
<b>CONTESTUALIZZAZIONE DELL'URBAN DATASET</b>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Timestamp</b>	Adimensionale	Date-Time gg-mm-aaaaThh:mm:ss	Tempo di generazione dell'Urban Dataset
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate del centro geometrico del sistema/solution di cui si sta chiedendo la connessione
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
<i>Nota: vi sarà una riga per ogni POD monitorato</i>			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Vertical Solution Name</b>	Adimensionale	String	Es. "Casaccia Smart Village"
<b>Data description</b>	Adimensionale	String	
<b>Communication Pattern</b>	Adimensionale	Enumeration	La lista predefinita è quella dei pattern individuate al momento: "Push", "Request Response", "Publish Subscribe".
<b>WS Endpoint</b>	Adimensionale	String	URL dell'endpoint del Web Service a cui si vuole accedere
<b>Data Format</b>	Adimensionale	String	URL dello Schema del Formato con cui si invieranno i dati
<b>Valore</b>	Da specificare	Da specificare	Parametro libero

**Tabella 20 – Definizione Urban Dataset "Communication Information"**

1.4.2.2 Tabella sintetica

Caso studio:	Direct Communication
<b>Obiettivo</b>	Attivazione di una comunicazione diretta fra due solution verticali
<b>Attore 1</b>	Solution Verticale 1 (PV1)
<b>Ruolo</b>	Fornisce una descrizione di un proprio servizio web al quale altre solution verticali potranno accedere instaurando comunicazioni dirette
<b>Attore 2</b>	Solution Verticale 2 (PV2)
<b>Ruolo</b>	Recupera la descrizione del servizio web offerto da SV1 e vi accede instaurando una comunicazione diretta
<b>Smart City Platform</b>	Invia i dati per la connessione diretta
<b>Urban Dataset</b>	Communication Information
Aggregazione temporale/spaziale	Statico / non definito
Policy dell'Urban Dataset	Pubblico (chi accede a questo Urban Dataset accede al servizio descritto)
Proprietà del dato	Solution Verticale 1
Riferimento nell'Ontologia	<a href="http://www.smartcityplatform.enea.it/smartcity-ontology#UrbanDataset14">http://www.smartcityplatform.enea.it/smartcity-ontology#UrbanDataset14</a>
<b>Start – End Collaboration</b>	Da definire per il singolo caso implementativo
	<b>Credenziali di accesso</b>
Registrazione Utente	Da definire per il singolo caso implementativo
Credenziali SV 1	<i>Username</i> : Da definire per il singolo caso implementativo
	<i>Password</i> : Da definire per il singolo caso implementativo
Credenziali SV 2	<i>Username</i> : Da definire per il singolo caso implementativo
	<i>Password</i> : Da definire per il singolo caso implementativo
	<b>Trasporto e formato</b>
<b>Comunicazione fra SV1 e SCP</b>	
Pattern	PUSH
Autenticazione	-
Sicurezza	HTTPS
Trasporto	L'SCP espone Web Service Rest tipo PUT
<b>Dati</b>	Urban Dataset "Communication Information"
Periodicità	Alla prima connessione
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://www.smartcityplatform.enea.it/ws/rest/getaway/">http://www.smartcityplatform.enea.it/ws/rest/getaway/</a>
Formato	XML o JSON (Da definire per il singolo caso implementativo)
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a> (JSON Schema)
	<a href="http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd">http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd</a> (XML Schema)
Link al template	
<b>Comunicazione fra SCP e SV2</b>	
Pattern	Request Response
Autenticazione	-
Sicurezza	
Trasporto	L'SCP espone Web Service per pattern Request Response
<b>Dati</b>	Urban Dataset "Communication Information"
Periodicità	Non appena viene ricevuto un nuovo Urban Dataset
Qualità del servizio	Con garanzia di consegna
WS Endpoint	<a href="http://www.smartcityplatform.enea.it/ws/rest/getaway/">http://www.smartcityplatform.enea.it/ws/rest/getaway/</a>
Formato	XML o JSON (Da definire per il singolo caso implementativo)
Link allo schema	<a href="http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json">http://smartcityplatform.enea.it/specification/information/1.0/json/schemas/scps-urbandataset-schema-1.0.json</a> (JSON Schema)
	<a href="http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd">http://smartcityplatform.enea.it/specification/information/1.0/xml/schemas/scps-urbandataset-schema-1.0.xsd</a> (XML Schema)
Link al template	

Tabella 21- Caso studio "Direct Communication"



## 1.5 Dall'analisi Casi Studio al Modello Dati Astratto

Da questa prima analisi si è giunti a una prima considerazione: ogni singolo step del flusso dati ha bisogno necessariamente di passare per tre macro-fasi che caratterizzano l'utilizzo della Smart City Platform. Le macrofasi sono le seguenti:

1. Configurazione della Smart City Platform: in essa verranno definiti
  - a. gli Urban Dataset, tramite un'Ontologia, in modo da averne una definizione condivisa (Urban Data Reference Specification);
  - b. un Registro delle Piattaforme Specifiche Verticali che possono accedere alla Collaborazione;
2. Configurazione della Collaborazione Piattaforma Verticale-Smart City Platform: in essa verranno decisi
  - a. gli aspetti relativi alle informazioni, ovvero quali Urban Dataset saranno scambiati;
  - b. le modalità di collaborazione (connessione, pattern di scambio, disconnessione);
  - c. le modalità di trasporto per permettere lo scambio dati tra Piattaforma Specifica Verticale e la Smart City Platform.
3. Scambio dei dati
  - a. Esportazione/Pubblicazione Urban Dataset;
  - b. Recupero/Importazione Urban Dataset;

La gestione di queste fasi richiede di definire nel livello funzionale:

- un componente (Business Manager, paragrafo 5.3.5, pag. 73) che, quando un Urban Dataset è ricevuto, scorre la lista degli accessi a quell'Urban Dataset e spedisce o pubblica
- un database (Registry, paragrafo 5.2.6, pag. 72) che memorizza le caratteristiche della collaborazione fra piattaforma

nonché il set d'informazioni e le funzionalità relativi a essi.

Inoltre, il Modello Dati Astratto (si veda APPENDICE E – SCPS Information Level a pag. 101, per l'esposizione dettagliata di tale modello dati) è stato progettato sulla base dei requisiti individuati nel corso dell'analisi dei casi studio, come illustrato nei paragrafi seguenti.

### 1.5.1 Analisi

I requisiti per il modello sono stati identificati tenendo conto delle esigenze che la sottospecifica "SCPS Information Level" deve soddisfare, ovvero:

- garantire l'interoperabilità con le eventuali soluzioni esistenti e non vincolare le implementazioni a specifiche tecnologie
- essere applicabile a qualsiasi città, indipendentemente dalle sue specificità e dai Contesti Applicativi esistenti
- essere modulare, poter quindi essere utilizzata anche in scenari in cui le altre specifiche SCPS non sono utilizzate.

Nello specifico, dall'analisi di queste esigenze, i **requisiti** individuati sono i seguenti:

- il modello deve essere indipendente dalla sintassi utilizzata per la creazione dei documenti scambiati
- il modello deve essere indipendente da qualsiasi protocollo/servizio per lo scambio di documenti elettronici
- il modello deve poter essere utilizzato per rappresentare Urban Dataset rilevati da qualsiasi Contesto Applicativo

- il modello deve essere scalabile nel tempo, ovvero supportare lo scambio di Urban Dataset non noti al momento della sua definizione.

In termini di **caratteristiche**, invece, il modello è stato progettato in modo tale che esso possa supportare la struttura degli Urban Dataset, sulla base dei flussi dati individuati nell'analisi dei casi studio, quindi:

- la descrizione formale dell'Urban Dataset utilizzato (p.es.: il riferimento alla specifica a cui aderisce)
- informazioni per contestualizzare i dati rilevati (p.es.: il sistema che li ha calcolati, il fuso orario dei timestamp, il momento in cui i dati sono stati aggregati, ...)
- i dati rilevati, ovvero i valori delle proprietà che compongono l'Urban Dataset e la cui natura, quindi, dipende dal suo tipo (p.es.: il tempo di permanenza di un'auto in un tratto di strada, la temperatura interna di un appartamento, ...)
- proprietà eterogenee.

È stato inoltre stabilito che un'istanza del modello dati possa contenere un solo tipo di Urban Dataset.

### 1.5.2 Progettazione

Il requisito di avere un unico modello per qualsiasi tipo di Urban Dataset, la numerosità degli Urban Dataset, l'impossibilità di conoscerli tutti al momento della progettazione e l'eterogeneità delle proprietà che li compongono hanno richiesto la definizione di una struttura generica e flessibile, priva di *elementi* non attribuibili a qualsiasi Urban Dataset.

Il modello, quindi, è stato concepito come una sorta di contenitore in cui le proprietà non sono predefinite, ma devono essere specificate insieme ai dati rilevati.

Spiegheremo l'approccio adottato con il seguente esempio; consideriamo due Urban Dataset (*UD*), aventi proprietà molto diverse:

1. *UD1* relativo ai consumi elettrici di un edificio in un certo periodo volto a fornire le seguenti informazioni:
  - istante di misurazione
  - periodo di rilevamento
  - identificatore dell'edificio a cui deve essere attribuito il consumo rilevato
  - consumo elettrico rilevato
2. *UD2* relativo al monitoraggio delle auto in transito in una strada in un certo periodo volto a fornire le seguenti informazioni:
  - istante di misurazione
  - periodo di rilevamento
  - nome della strada monitorata
  - numero di auto

Delle ipotesi di modelli dati specifici per ciascuno dei due UD potrebbero essere quelle illustrate in Tabella 22 e Tabella 23, in cui ci sono elementi attribuibili a qualsiasi UD (*Timestamp*, *Inizio periodo*, *Fine periodo*) ed elementi specifici (*Edificio* e *Consumo elettrico* per l'*UD1*, *Strada* e *Numero auto* per l'*UD2*):

Elemento del modello	Esempio di valore
Timestamp	2017-07-15 12:0:00
Inizio periodo	2017-07-14 12:0:00
Fine periodo	2017-07-15 12:0:00

<b>Edificio</b>	<i>ENEA-PaIA-BO</i>
<b>Consumo elettrico</b>	<i>105 kWh</i>

**Tabella 22** Modello dati specifico per l'UD1

<b>Elemento del modello</b>	<b>Esempio di valore</b>
Timestamp	<i>2017-07-11 12:0:00</i>
Inizio periodo	<i>2017-07-10 12:0:00</i>
Fine periodo	<i>2017-07-11 12:0:00</i>
<b>Strada</b>	<i>Via Bianchi</i>
<b>Numero auto</b>	<i>46</i>

**Tabella 23** Modello dati specifico per l'UD2

Adottando un approccio generico, invece, si potrebbe definire un solo modello capace di supportare pienamente entrambe gli UD 1 e 2, ma anche altri aventi proprietà molto diverse poiché le proprietà specifiche degli UD non sarebbero esplicitate come elementi del modello, ma verrebbero gestite come se fossero dei parametri da indicare insieme ai valori rilevati su di esse. Un esempio di questo approccio è illustrato da Tabella 24 e Tabella 25.

<b>Elemento del modello</b>	<b>Esempio di valore</b>
Timestamp	<i>2017-07-15 12:0:00</i>
Inizio periodo	<i>2017-07-14 12:0:00</i>
Fine periodo	<i>2017-07-15 12:0:00</i>
<b>Proprietà</b>	<i>Edificio</i>
<b>Valore della proprietà</b>	<i>ENEA-PaIA-BO</i>
<b>Proprietà</b>	<i>Consumo elettrico</i>
<b>Valore della proprietà</b>	<i>105 kWh</i>

**Tabella 24** Modello dati generico applicato all'UD 1

<b>Elemento del modello</b>	<b>Esempio di valore</b>
Timestamp	<i>2017-07-11 12:0:00</i>
Inizio periodo	<i>2017-07-10 12:0:00</i>
Fine periodo	<i>2017-07-11 12:0:00</i>
<b>Proprietà</b>	<i>Strada</i>
<b>Valore della proprietà</b>	<i>Via Bianchi</i>
<b>Proprietà</b>	<i>Numero auto</i>
<b>Valore della proprietà</b>	<i>46</i>

**Tabella 25** Modello dati generico applicato all'UD 2

L'approccio generico è appunto quello adottato per la definizione del Modello Dati Astratto.

## 2 Sviluppo di un prototipo di piattaforma ICT di test (D6a.2)

L'obiettivo D6.a2 consiste nella progettazione e implementazione sull'infrastruttura ENEA, di un primo prototipo di piattaforma ICT denominata Smart City Platform (SCP) per la gestione di Smart District. Il lavoro svolto in questa seconda annualità è stato sviluppato sulla base dei risultati ottenuti nel PAR2015, in particolare quelli riguardanti la definizione delle specifiche architetture volte all'interoperabilità del prototipo di piattaforma software, e in stretta collaborazione con il sotto-obiettivo D6.a1.

Nel dettaglio, sono state implementate e validate le cinque sotto-specifiche riguardanti i livelli d'interoperabilità, definite nel D6.a1.

Nell'elenco sottostante sono riportati i progressi implementativi fatti per ognuno dei livelli d'interoperabilità con le relative specifiche (Figura 14), e in grassetto le sotto-specifiche implementate nel D6.a2.

1. **Functional:** design dell'Architettura di Riferimento della SCP e delle funzionalità;
2. **Collaboration:** definizione del Registry e delle GUI di base per la gestione delle collaborazioni;
3. **Semantic:** definizione dell'Ontologia per l'interpretazione centralizzata;
4. **Information:** modello dei dati gestiti dalla SCP e formati XML e JSON per rappresentare i dati;
5. **Communication:** definizione dei pattern e delle interfacce supportate per il trasporto dei dati.

L'architettura proposta nel sotto-obiettivo D6.a2, per la SCP è flessibile e affidabile; è basata su Cloud Computing ospitata su infrastruttura ENEAGRID e utilizza soluzioni che si prestano alla scalabilità in funzione delle richieste di servizi da parte dei diversi verticali che invieranno dati. Inoltre, lo sviluppo è stato guidato anche dai diversi casi studio (per dettagli sulla definizione dei casi studio si veda il D6.a1).

Il riferimento ai casi studio ha permesso, da un lato, di testare le funzionalità della piattaforma software, e, dall'altro, di verificarne la replicabilità; infatti, i moduli sono progettati sulla base di specifiche e gli stessi sono creati per essere indipendenti ma cooperanti.

Gli aspetti d'interoperabilità sono garantiti e incentrati sul formato dei dati scambiati nella SCP.

Durante quest'annualità, gli aspetti chiave che hanno permesso la produzione di un ambiente smart in ENEA sono stati:

1. Definizione e integrazione di una piattaforma capace di offrire **servizi cloud-based** all'interno di ENEAGRID;
2. Erogazione di **soluzioni personalizzabili e flessibili** per la gestione di problematiche legate ad ambienti smart;
3. Garantire **scalabilità** e capacità di gestione dei diversi casi studio;
4. Progettazione di un **modello di dati versatile**, ma allo stesso espressivo per i contesti eterogenei di tipo smart;
5. Definizione di protocolli di comunicazione per il trasporto dati da e verso la SCP;
6. Progettazione d'**interfacce di scambio dati** e di esportazione di servizi;
7. Aspetti di **sicurezza e privacy** della SCP, con gestione centralizzata delle utenze dei servizi convogliati.

La progettazione del prototipo di Smart City Platform è stata fatta individuando le risorse hardware e software disponibili sull'infrastruttura ENEAGRID e in particolare sono state utilizzate le risorse presenti Centro Ricerche ENEA di Portici. Questo, infatti, è il centro che ospita la maggior parte delle risorse di calcolo in ENEA, con particolare riferimento ai più recenti ed evoluti cluster CRESCO<sup>4</sup>. In tale sito, è presente anche un'infrastruttura di Cloud Computing basata su tecnologia VMWare, prodotto leader nel campo della virtualizzazione. È stato necessario espandere l'infrastruttura di cloud per far fronte alle esigenze della SCP e di tutti i servizi virtuali da essa erogati.

---

4 <http://www.cresco.enea.it/>

Per rispondere alle esigenze delle attività svolte nel PAR 2016, sono state definite una serie di **macchine virtuali (VM)**, ognuna delle quali svolge un ruolo ben preciso nella SCP, vale a dire:

1. Gestione dei servizi e del trasporto dei dati (lato servizi);
2. Gestione della persistenza dei dati (lato database);
3. Gestione degli accessi e delle identità (lato sicurezza e privacy).

Grazie alla definizione delle specifiche dell'architettura di riferimento per la SCP, eseguita nella prima annualità e ripresa in questa seconda (D6.a1) si è passati alla fase implementativa che ha coperto la seconda annualità. Fondamentale è stata, in questa annualità, l'interazione e integrazione dei due sotto-obiettivi D6.a1 e D6.a2, per la fase operativa della SCP.

Il risultato derivante dal sotto-obiettivo D6.a2 è identificabile in una prima sperimentazione dell'implementazione basata in particolare sulle sotto-specifiche SCP Information Level, SCP Communication Level in ENEAGRID. In particolare, si è voluto assicurare un primo test relativo a un'implementazione completa della catena dei servizi offerti dalla SCP. Per tale test, è stato sufficiente validare le soluzioni progettate riguardo un singolo caso studio e un singolo pattern di trasporto dei dati, gestendo anche la persistenza degli stessi una volta modellati nella SCP.

Il lavoro fatto in questa seconda annualità è fondamentale per quello in previsione nella terza e ultima annualità; infatti, quest'ultima sarà incentrata sull'implementazione del prototipo in termini di completa aderenza e verifica delle specifiche Smart City Platform presentate nel sotto-obiettivo a.1. Inoltre, il D6.a2 prevede nell'ultima annualità una fase dedicata all'integrazioni di due o più contesti applicativi verticali (descrittivi nei casi studio inter-solution del secondo anno) relativi allo contesto Smart Village Casaccia, contesto reale che ben approssima quello di uno Smart District urbano. Quest'ultimo aspetto permetterà di completare lo sviluppo di una Smart Platform in grado di raccogliere e integrare tra loro i dati dai diversi ambiti applicativi.

Nei prossimi paragrafi si documenta il lavoro svolto nel sotto-obiettivo D6.a2.

Il paragrafo 2.1 offre una panoramica sintetica dell'infrastruttura ENEA utilizzata per lo sviluppo e l'implementazione del prototipo della SCP. Nel paragrafo 2.2 viene invece presentata l'architettura software di riferimento per l'implementazione delle specifiche. Il paragrafo 2.3 affronta la tematica della persistenza dei dati con focus particolare sulla progettazione e implementazione del Data Model Registry e alle scelte tecnologiche adottate per i database. Nel paragrafo 2.4 sono invece discusse le modalità di accesso ai dati e autenticazione, trasporto tramite WS Rest, importazione/esportazione del formato JSON.

## 2.1 Infrastruttura ICT ENEA

L'ENEA sviluppa e gestisce una complessa architettura ICT che mette a disposizione dell'utenza sistemi avanzati di calcolo, modellistica e visualizzazione tridimensionale dei dati, attraverso un utilizzo estensivo delle tecnologie GRID interconnesse grazie alla rete GARR. L'infrastruttura ICT annovera al momento 6 poli (Frascati, Portici, Bologna, Casaccia, Trisaia, Brindisi), ognuno dotato di infrastrutture di calcolo e visualizzazione 3D e di competenze specialistiche operanti su diverse tematiche applicative. In particolare presso il Centro Ricerche di Portici è operativo un sistema di supercalcolo, che si colloca ai primissimi posti nel panorama Nazionale, denominato *CRESCO - Centro Computazionale di RicErca sui Sistemi COmplessi*. Attualmente, l'infrastruttura raggiunge una potenza di circa 150 TFlops, che raggiungerà una potenza di 700 Tflops a dicembre 2017 con la messa in funzione delle nuove macchine. I cluster HPC sono basati principalmente su processori multicore convenzionali INTEL Xeon e processori accelerati dedicati basati sia su Intel Xeon/PHI che GPU/Nvidia. I sistemi di storage si basano su sistemi RAID per un totale di circa 1.8 PB in configurazione SAN/Switched e SRP/Infiniband. I files system sono disponibili sia su rete geografica che in cluster GPFS paralleli a elevato throughput .

I cluster CRESCO sono utilizzati per attività di Ricerca e Sviluppo in alcuni dei settori di punta dell'ENEA. Il progetto CRESCO si inquadra nell'ambito delle attività da molti anni avviate per lo sviluppo della griglia computazionale ENEA e dell'inserimento di questa nel sistema di GRID computing nazionale ed europeo. L'ENEA opera dal 1998 su questa tematica, riuscendo a pervenire a soluzioni tecnologiche originali per la

condivisione in GRID di piattaforme di calcolo eterogenee e la remotizzazione di strumenti scientifici di grande complessità, quale per esempio il microscopio elettronico situato nel Centro di Brindisi. ENEA ha partecipato al più grande progetto di GRID finanziato dal Programma Quadro (Progetto EGEE) a livello europeo, nell'ambito del quale sono stati ulteriormente sviluppati alcuni risultati riguardanti l'interoperabilità delle piattaforme di calcolo, ed è membro dell'associazione IGI (Italian GRID Infrastructure) che ha come obiettivo, la costituzione dell'organismo nazionale di governo della GRID. "sistema virtuale" delle risorse informatiche distribuite per il calcolo scientifico e grafica avanzata dell'ENEA. L'infrastruttura consente all'utente di utilizzare, in modo completamente automatico, qualunque calcolatore del sistema per eseguire i propri programmi ed elaborare i propri dati. È compito del sistema risolvere tutte le operazioni legate alla localizzazione fisica dei calcolatori, alla loro architettura e ai diversi sistemi operativi. Si ha quindi a disposizione un ambiente unico, dove la scelta della stazione di lavoro da parte dell'utente è dipendente unicamente dalla portabilità dell'applicazione. L'infrastruttura CRESCO è una "enabling technology" che consente a laboratori e gruppi di ricerca appartenenti a tutti i Dipartimenti dell'Agenzia di affrontare problemi tecnico-scientifici fino a oggi preclusi e di accedere a finanziamenti promossi dai programmi Europei e nazionali.

La possibilità di accedere alle risorse disponibili sull'infrastruttura di supercalcolo CRESCO dell'ENEA consente di ottenere soluzioni per la gestione e la visualizzazione di grandi moli di dati ottenuti sia dalle attività di modellistica e simulazione che da impianti sperimentali complessi.

I sistemi di calcolo scientifico che operano nell'infrastruttura ENEA condividono uno spazio dati che fa uso anche del file system geografico AFS, accessibile in modo trasparente da qualsiasi sistema di elaborazione, ovunque situato sia nei centri di calcolo dei vari centri ENEA che all'esterno.

I super calcolatori CRESCO, installati nei siti di Portici, Frascati, Casaccia e Brindisi costituiscono la risorsa di calcolo più importante. Il Centro ENEA di Portici ospita la parte più consistente dei sistemi HPC-CRESCO.

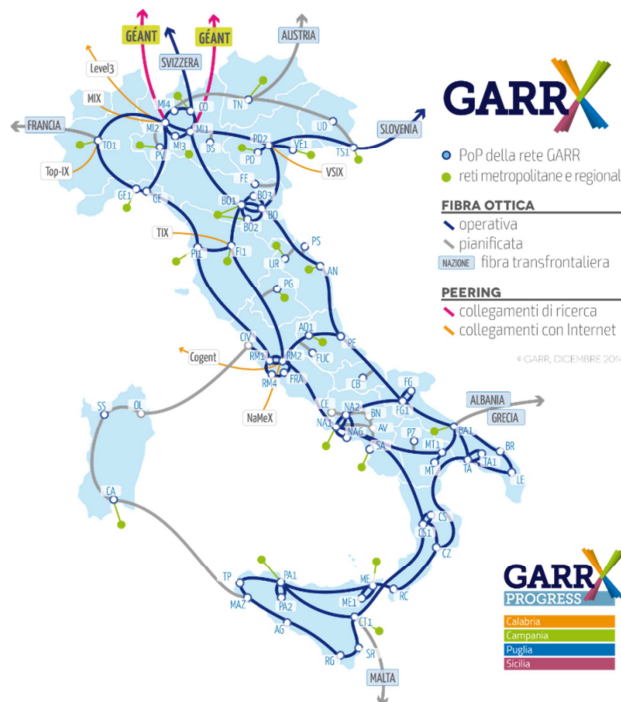


Figura 8 – Infrastruttura ENEA inserita nella rete italiana GARR.

Nel 2015 l'ENEA e il CINECA, il maggior centro di supercalcolo nazionale, hanno siglato una convenzione, della durata di 5 anni, in cui l'ENEA riveste ufficialmente un ruolo di TIER\_1 per le infrastrutture HPC e insieme al CINECA, che riveste il ruolo nazionale di TIER\_0 per l'HPC, possono presentare progetti congiunti di Ricerca & Sviluppo e di realizzazione di infrastrutture HPC a valere su bandi nazionali ed europei.

Nello sviluppo di questa convenzione, l'ENEA sta installando presso il centro di calcolo CRESCO un nuovo sistema HPC: CRESCO6 dalla Potenza di picco di circa 700 TFlops basati sui nuovi processor Intel SkyLake a 24 cores. CRESCO6 è basato sulla nuova tecnologia di rete a bassa latenza OmniPath di Intel, capace di sostenere una banda di 100 Gbps.

La seguente tabella riporta le caratteristiche dei sistemi HPC, basati su CPU Intel Xeon convenzionali, del Centro di Calcolo CRESCO installati o in corso di installazione (CRESCO6 entro fine 2017).

Logical Name	CRESCO3	CRESCO 4	CRESCO 5	CRESCO 6
Model	SuperMicro	SuperMicro	SuperMicro	Lenovo NeXtScale
Architecture	Linux Cluster	Linux Cluster	Linux Cluster	Linux Cluster
Processor	AMD Opteron 6234TM 12c 2.4 GHz	Intel Xeon E5-2670 (SandyBridge) 4c 2.6 GHz	Intel Xeon E5-2630-v3 (Haswell) 8c 2.4 GHz	Intel Xeon 8160 (SkyLake) 24c 2.1 GHz
#core	1536	4864	640	10368
#node	64	304	40	216
#rack	4	4	2	4
RAM per node	64 GB	64 GB	64 GB	196 GB
Interconnection	Infiniband 4xQDR	Infiniband 4xQDR	Infiniband 4xQDR	Intel Omni-Path
Operating System	Centos 6.4	Centos 6.4	Centos 6.4	Centos 7.4
Peak Performance	13 TFlops	100 TFlops	24 TFlops	700 TFlops

**Tabella 26 – Caratteristiche principali dei sistemi HPC del C.R. Portici.**

Ulteriori risorse aggiuntive installati nel centro di Calcolo CRESCO sono relative ai cluster speciali HPC e ai sistemi di storage ad alte prestazioni, riportati nella Tabella 27:

Risorsa	caratteristiche
HPC cluster Linux large memory	5 nodi a grande memoria 768 GB RAM, IvyBridge 3.5GHz
HPC cluster Linux cooprocessori	4 nodi con Intel PHI e Nvidia K40
HPC GPU	2 nodi GPU S2050
RAID Storage System	2xDDN SFA7700 540+540 TB SATA disks
RAID Storage System	DDN S2A9900 600 TB SATA disks
Tape Library disaster recovery	IBM TS3500 2.5 PB LTO 4 tapes

**Tabella 27 – Caratteristiche aggiuntive riferite ai cluster speciali HPC e ai sistemi di storage ad alte prestazioni presenti nel C.R. Portici.**

Le aree di storage totale distribuite per i vari centri ENEA sono:

- AFS filesystem distribuito della cella ENEA.IT: 30 TB
- GPFS filesystem parallelo in configurazione multi-site: 1.8 PB
- Cloud storage dedicato distribuito in 7 centri: 1PB

Le risorse Hardware e Software distribuite nei vari centri ENEA sono gestite da IBM Platform LSF un Resources Management System che permette di ottimizzare l'utilizzo dei vari clusters HPC e di controllarne la disponibilità. L'accesso alle risorse è garantito in sicurezza da autenticazione Single-Signed-On(SSO) basata su Kerberos e da autorizzazioni definite da ACL (Access Control List). L'accesso dell'utenza alle applicazioni scientifiche e tecniche (come Ansys, Comsol, Nastran, Fluent) è fornito anche attraverso portali dedicati (Laboratori Virtuali) basati su tecnologie come NX per ottimizzare l'accesso remoto ai dati.

In generale l'intera infrastruttura ICT dell' ENEA include i seguenti servizi:

- Generali: rete, SSO su Active Directory, E-Mail – dominio enea.it
- Videoconferenze e Voip
- Gestionale: servizi amministrativi (retribuzioni, time-sheets, missioni...)
- Cloud Computing: VMware e OpenStack
- Cloud Storage: OwnCloud storage (eneabox e E3S)
- Web site basati su architetture Plone/WordPress su piattaforme LAMP
- Gestione Remota di laboratori ed esperimenti scientifici

L'utilizzo del cluster avviene facendo il login su uno dei nodi di front-end (il sistema di autenticazione: Kerberos 5). I nodi di front-end servono semplicemente per il lancio delle applicazioni tramite LSF, per editare i propri script di lancio o per le compilazioni. Il sistema presente per il monitoring è il sistema Zabbix.

L'accesso ai front-end di ENEA-GRID dall'esterno può avvenire via web dalla pagina di CRESCO ([www.cresco.enea.it](http://www.cresco.enea.it)) mediante le interfacce web per utente:

- FARO/NX.
- Jobrama: stato dei job e accounting.

I file system disponibili su CRESCO4 sono:

- AFS/Open AFS: il file system geograficamente distribuito comune a tutta ENEAGRID.
- GPFS: il file system di IBM ad alte prestazioni per l'I/O parallelo, anche su WAN.

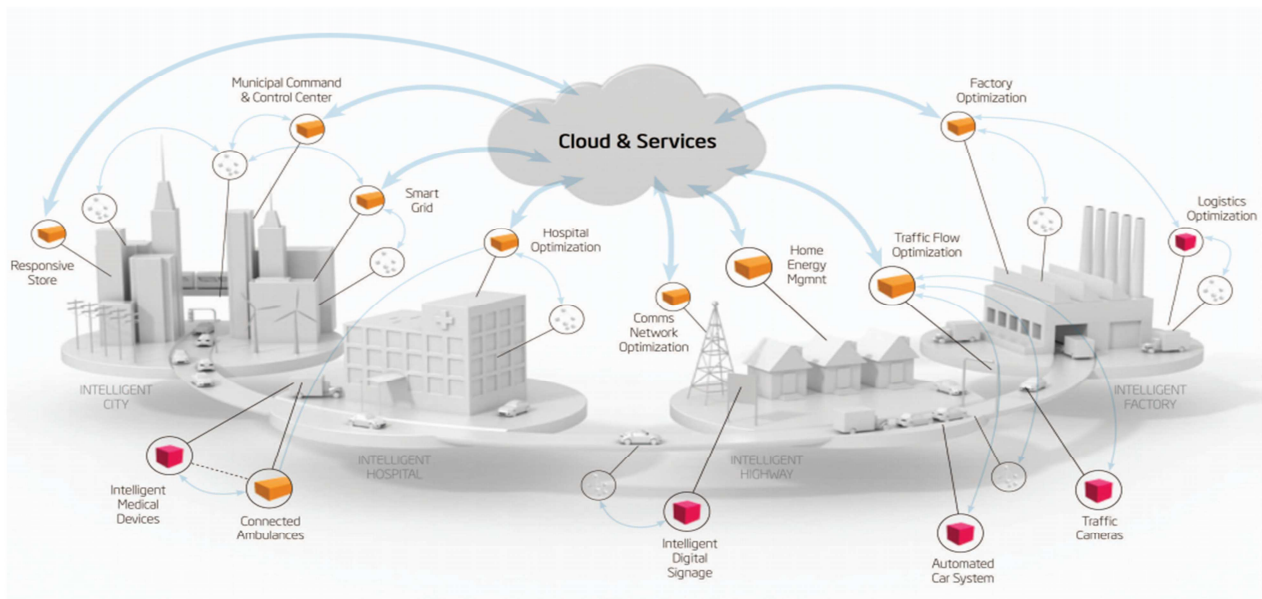
L'infrastruttura ENEA, grazie alle sue funzionalità, ha reso possibile la progettazione e l'implementazione delle specifiche architetture - presentate nella prima parte di questo report – per lo sviluppo di un prototipo di smart city platform (SCP).

## 2.2 Architettura Software

L'emergere dell'Internet of Things (IoT) (4) e la continua evoluzione della digitalizzazione, offre opportunità per creare un sistema di sistemi (SoS) (5) oltre a quelli individuali (sistemi isolati di monitoraggio e controllo). L'approccio SoS offre l'opportunità di collegare i singoli sistemi in uno più grande, così da creare nuovi modelli di business e quindi maggiori opportunità di raccogliere dati, analizzarli e sviluppare nuovi servizi. Questi ultimi sulla base, ovviamente, della disponibilità dei dati fornita da un ampio spettro di sistemi. Sistemi diversi ma interoperabili e compatibili nella prospettiva del consumatore/utente: affidabilità del servizio e difesa della portabilità dei dati.

La progettazione e lo sviluppo di un prototipo-ENEA di piattaforma ICT sono posti nell'ottica di sviluppare un SoS e quindi un modello integrato di distretto urbano in Figura 9. Per comprendere come debbano essere governati i processi di standardizzazione e interoperabilità dei servizi che sono alla base degli Smart District è tuttavia, necessario comprenderne la natura. Si tratta di servizi di IT/Information Communication Technology (ICT) che possono assumere specifiche connotazioni di Cloud Computing, Internet of Things, Big Data, ecc.





**Figura 9 - Esempio di approccio Sistemi di Sistemi (SoS) per Smart City (6).**

Nel caso dei distretti e nello specifico della Smart District Platform uno degli obiettivi-ICT del progetto è utilizzare la potenza e la flessibilità del cloud computing per integrare diversi sistemi di monitoraggio e controllo locali al fine di ottenere servizi energetici su scala distrettuale. In quest’ottica, è stato seguito un approccio ICT-SoS based per la progettazione e l’implementazione del prototipo di piattaforma Smart City. Nel dettaglio, per l’architettura di riferimento, le considerazioni sull’ICT sono state focalizzate sui seguenti temi:

- Scalability: utilizzo delle tecnologie Cloud su infrastruttura ENEA.
- Data aggregation: Scelte tecnologiche dei DBs e data model registry per l’accesso ai dati.
- Data interoperability: formato e trasporto dei dati.
- Security&Privacy: in relazione all’accesso ai dati, per evitare i rischi di lock-in e cattura dei dati

Con riferimento alla scala di distretto sono state sviluppate le principali funzionalità del prototipo sulla base delle specifiche architetture fornite dal D6.a1. Prima di descrivere l’architettura del prototipo, si rimanda allo schema che riassume il concetto alla base della Smart City Platform di Figura 15 (pag. 68), il cui principale obiettivo è quello di sviluppare un sistema di piattaforma orizzontale rispetto ai vari servizi verticali (silos) presenti in una città. La piattaforma orizzontale sarà in grado di fornire a ogni verticale le necessarie interfacce di scambio dati garantendo così l’interoperabilità dei silos.

Lo sviluppo delle funzionalità del prototipo è stato fatto seguendo le specifiche di riferimento presentate in appendice con lo scopo di rendere attuare l’interoperabilità alle diverse scale: livelli architetture (functional, collaboration, information, ...), tra i silos verticali (building, lighting,...), livello di aggregazione dei dati (piattaforma smart city, piattaforma locale, sensori).

La progettazione e lo sviluppo delle funzionalità di prototipo-piattaforma ICT cloud-based su infrastruttura ENEA si può riassumere nei seguenti punti:

- Progettazione delle funzionalità del prototipo di piattaforma per la gestione di grandi moli di dati:
  - Particolare attenzione è stata data allo studio e all’analisi delle problematiche inerenti la persistenza di Big Data e alle tecnologie Open Source da adoperare per la loro gestione sull’infrastruttura ENEA (paragrafo 2.3).
- Sviluppo delle funzionalità del prototipo e architettura di riferimento. Partendo dal modello architetture di riferimento verranno definiti i vari livelli di interoperabilità del prototipo (paragrafo 2.4).

La piattaforma prototipale Smart City Platform (SCP), come già detto precedentemente, è stata installata sull'infrastruttura ENEA presso l'ambiente del C.R. Portici.

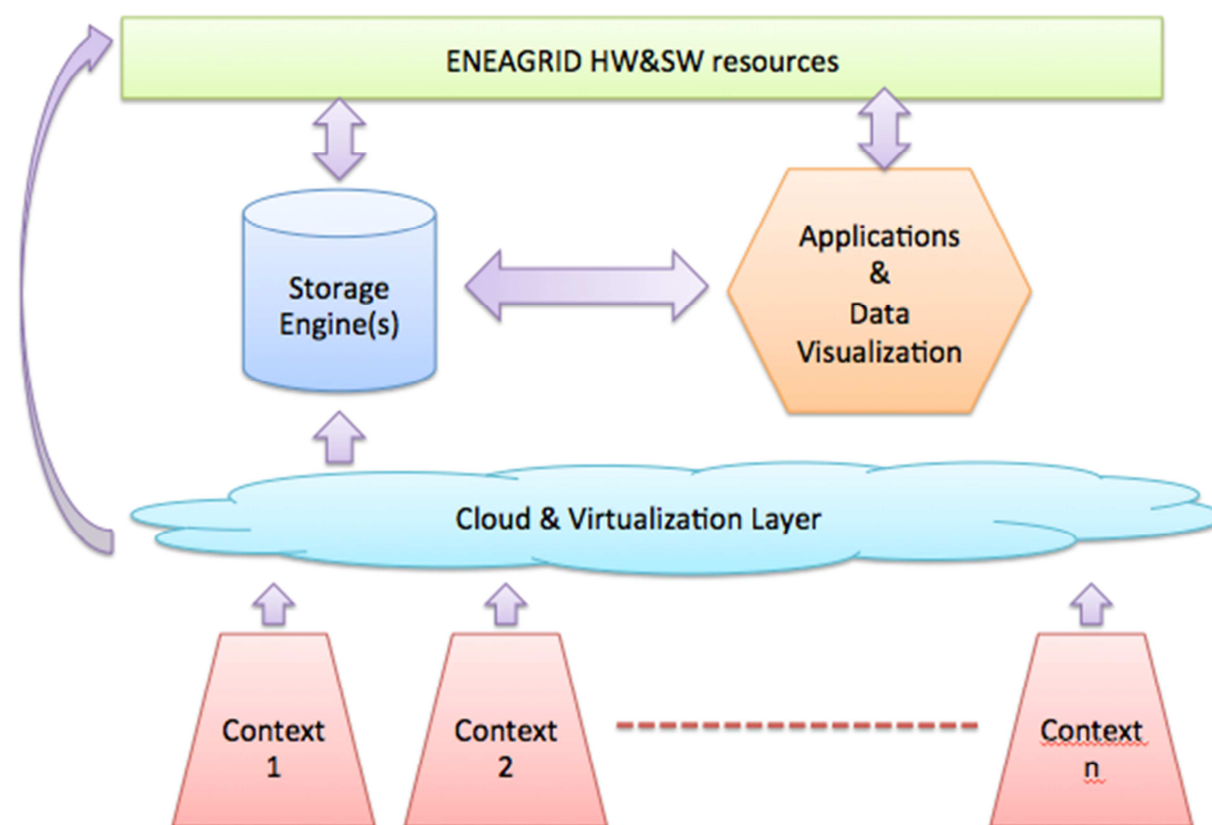
La creazione di un ambiente customizzato ENEA-piattaforma cloud-based è possibile grazie alle tecnologie presenti sulla griglia ENEA, e soprattutto la tecnologia VMWARE che consente di ospitare i servizi di gestione, di trasporto ed elaborazione dei dati del progetto, garantendo la loro disponibilità e la protezione nel tempo. Grazie alle funzionalità di un ambiente customizzato ENEA-piattaforma è possibile reperire i dati energetici provenienti dall'ambiente test Casaccia e da altre fonti esterne di dati.

Nel dettaglio, il cluster di server che ospita il progetto è basato su tecnologia **VMWARE** e si compone di **n.8 server** dislocati fisicamente nel centro di Calcolo della sede ENEA-C.R Portici

Ognuno dei server della cloud farm VMWare ha le seguenti caratteristiche:

- 2 socket AMD® con processori Opteron® Opteron 6234TM 12 cores 2.4 GHz.
- 64 GB/RAM.
- Due interfacce Gethernet.
- Supporto BMC/IPMI 2.0 e software per la gestione remota della console.

In Figura 10, si riporta il diagramma di architettura tecnica del prototipo Smart City Platform ospitato dall'infrastruttura ENEA.



**Figura 10 – Schema di riferimento dell'architettura del prototipo di Smart City Platform**

L'architettura presentata presenta la peculiarità della **scalabilità sia orizzontale che verticale**; pertanto, il numero delle VM e/o delle risorse elaborative può essere variato in funzione del volume e del throughput dei dati immessi nel sistema. Inoltre, la flessibilità garantita da questa soluzione di cloud unita all'integrazione con le risorse dell'ENEAGRID permette di sfruttare le caratteristiche più rilevanti di entrambi i contesti e di avere quindi una SCP particolarmente potente e funzionale.

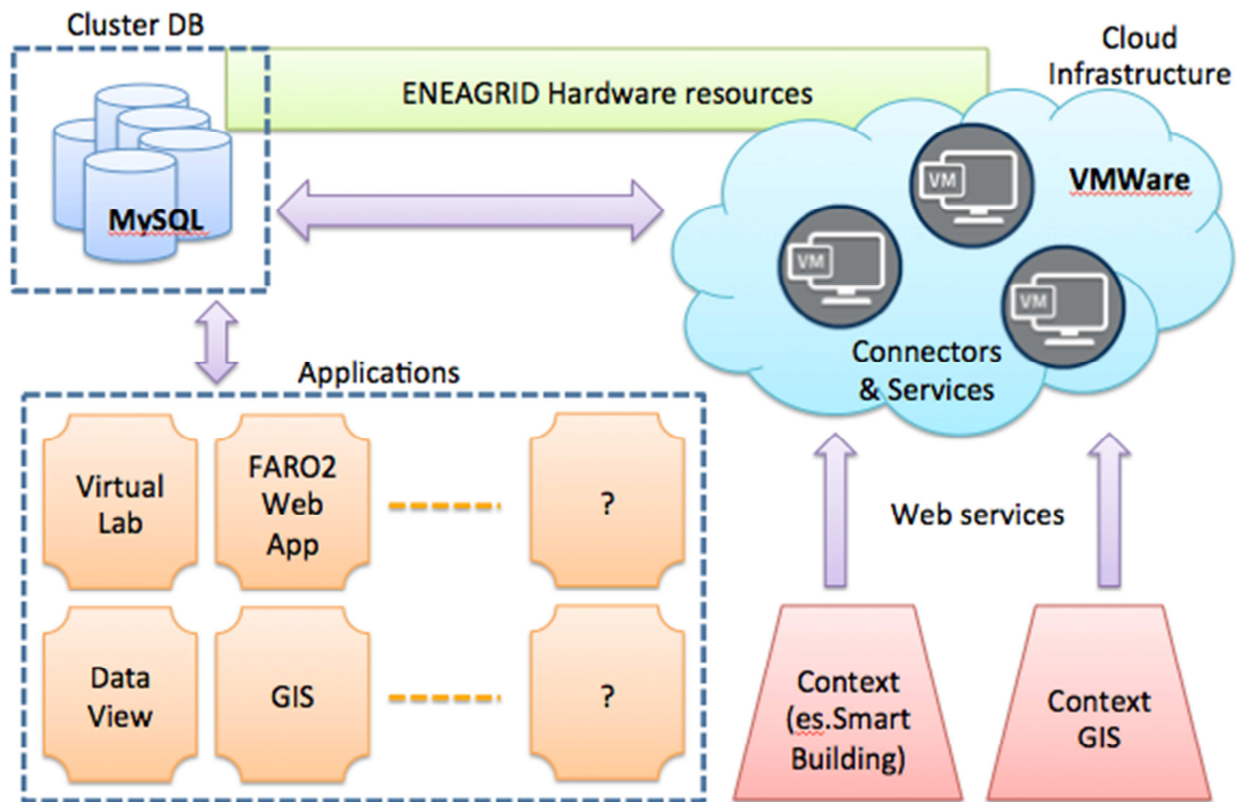


Figura 11 – Architettura del prototipo Smart City Platform: dettagli implementativi.

Nella Figura 11, sono riportati i dettagli implementativi dell'architettura utilizzata per lo sviluppo del prototipo di piattaforma Smart City già presentata in Figura 10.

Nel dettaglio, nel diagramma in Figura 11 sono presentati i seguenti componenti:

1. Infrastruttura di cloud VMWare
  - Prodotto leader del settore.
  - Live migration delle vm e backup.
  - Fault tolerance.
  - Flessibilità.
  - Connector dedicati ai verticali.
2. Cluster DB
  - MySQL/MariaDB cluster.
  - Scalabile ed estendibile con altri DB engine (e.g., PostgreSQL).
3. View e application interface
  - Laboratorio Virtuale per community e tool di support.
  - FARO2-like web app per accesso dedicato ad applicative.
  - Sezione di Data Esplorazione & Presentation.
  - Interfacce di visualizzazione dati (e.g: Tableau).
  - Visualizzazione dati GIS attraverso ArcGIS.

### 2.3 Persistenza

L'ambito di persistenza si concentra sull'archiviazione e il recupero di dati efficienti con riferimento alla Smart City Platform. Tale ambito nel modello di riferimento corrisponde al livello di piattaforma locale (Figura 14).

La piattaforma locale contiene la persistenza dei dati prelevati dal campo, eseguirà l'aggregazione e la rielaborazione dei dati nonché la messa a disposizione degli indicatori sintetici verso il livello superiore.

Per quanto riguarda i dati provenienti dai vari verticali (i.e., Urban Dataset), è stato progettato un modello di persistenza dei dati che si basa sulle specifiche del formato concordato, strutturando i dati in arrivo dalle solution verticali in un modello relazionale.

Il flusso logico segue i seguenti passi:

1. La comunicazione tra il verticale e la SCP avviene mediante Web Service (WS).
2. Il dato viene inviato al web service in formato JSON o XML.
3. Il dato ricevuto viene verificato in modo che, prima di essere processato, si attesti che sia conforme alle specifiche di formato stabilite.
4. Il dato in formato testuale viene mappato in un oggetto software per poter essere gestito e manipolato.
5. I campi dell'UD vengono estratti dall'oggetto software e scritti su un database relazionale appositamente creato per la persistenza degli UD.

A questo punto, una volta strutturati i dati nel database, questi possono essere interrogati da altre applicazioni ed elaborati ulteriormente o visualizzati in interfacce grafiche.

## 2.4 Implementazione Piattaforma

A seguire sarà fornita una breve descrizione del lavoro svolto in termini d'implementazione del trasporto dei dati (Urban Dataset) da e verso la Smart City Platform (SCP). In particolare saranno illustrate: la teoria considerata, le metodologie sperimentali utilizzate, gli strumenti adottati e le tecnologie sviluppate.

### 2.4.1 Teoria considerata

L'architettura della SCP è organizzata in moduli funzionali, a ognuno dei quali è demandato lo svolgimento di una particolare funzione. Il tutto ha il fine ultimo di creare un ambiente che si interfacci con l'esterno permettendo così lo scambio dati con le Solution verticali che sono registrate presso la SCP stessa. L'organizzazione in moduli permette di separare lo sviluppo dei diversi componenti (dati, funzioni e interfacce) e di garantire al tempo stesso l'espandibilità e la scalabilità del framework.

### 2.4.2 Metodologie sperimentali utilizzate

Poiché uno degli obiettivi principali della SCP è di interfacciarsi con Solution verticali esterne, si è dotato l'ambiente Smart City Platform di una serie d'interfacce verso l'esterno, che sono il punto focale delle specifiche per l'interoperabilità, e, in generale, verso le Solution verticali che inviano e ricevono dati dalla stessa SCP. Questo si traduce nella definizione di una serie di servizi web raggiungibili tramite un indirizzo web pubblico.

Il servizio web, sviluppato su protocollo sicuro HTTPS con architettura REST (Representational State Transfer), prevede l'invio e la ricezione di due tipologie di formato dati: il formato **JSON** e il formato **XML**.

Infine, per garantire la sicurezza della trasmissione dei dati in maniera tale da accertare l'identità tra il mittente e, il ricevente è stato preso in considerazione **OpenId Connect** come opportuno protocollo di riconoscimento.

### 2.4.3 Strumenti adottati

A livello *hardware*, per ognuna delle attività della Smart City Platform, è stata configurata in maniera opportuna una macchina server. Nello specifico, i tre server sono nodi dell'infrastruttura di Cloud Vmware di ENEAGRID. Una prima macchina virtuale (**ws-scp**) funge da endpoint della SCP ed è dedicata alle operazioni RESTful, al controllo dell'identità digitale e alla lettura/scrittura sul DB. Una seconda macchina (**db-scp**) è configurata per ospitare il database. Infine su una terza e un'ultima macchina (**id-scp**) è installato il servizio di certificazione dell'identità. Da notare che questa configurazione a tre server è replicabile in funzione del numero di verticali che si devono gestire, a prova ancora una volta della modularità e della scalabilità della soluzione adottata.

A *livello software*, sulla ws-scp è stato installato e configurato un server Tomcat (versione 8.5.15) che espone i suoi servizi verso l'esterno della rete ENEA. Sulla macchina id-scp è stato invece installato Connect2id (versione 6.9.1) che funge da ID Provider e anche questi è eseguito dal server Tomcat (versione 8.5.6). Infine sulla macchina db-scp è stato installato un database relazionale MySql (versione 14.14, distribuzione 5.7.19) per registrare gli UrbanDataset e i dati riguardanti le utenze autorizzate all'accesso.

I dettagli delle 3 MV sono riportati in Tabella 28.

Macchine Virtuali (Cloud Vmware-ENEA)	Nome	Tipologia/Operazioni	Caratteristiche Software delle MV
Gestione dei servizi e del trasporto dei dati	ws-scp	Servizi (Endpoint), operazioni effettuate: <ul style="list-style-type: none"> <li>• RESTful;</li> <li>• controllo dell'identità digitale;</li> <li>• lettura/scrittura sul DB.</li> </ul>	Server Tomcat (Versione 8.5.15)
Gestione della persistenza dei dati	db-scp	Database, operazioni effettuate: <ul style="list-style-type: none"> <li>• registrare gli UrbanDataset;</li> <li>• registrare i dati riguardanti le utenze autorizzate all'accesso.</li> </ul>	Database relazionale MySql (Versione 14.14, distribuzione 5.7.19)
Gestione degli accessi e delle identità	id-scp	Sicurezza & privacy, operazioni effettuate: <ul style="list-style-type: none"> <li>• certificazione dell'identità.</li> </ul>	- Connect2id (versione 6.9.1) che funge da ID Provider; - Server Tomcat (versione 8.5.6).

**Tabella 28 – Tabella descrittiva delle Macchine Virtuali della Smart City Platform.**

#### 2.4.4 Tecnologie sviluppate

Per il server ws-scp, è stato implementato un pacchetto di classi JAVA (versione 1.8+) che, una volta esportato come modulo Web application ARrchive (WAR), viene erogato dal server Tomcat.

I principali metodi sviluppati in questo modulo si distinguono in funzione alle operazioni che eseguono:

- Il **controllo dell'identità** (metodo di login) – Figura 12. Il Web Service riceve le credenziali di autenticazione di un sender e, dopo avere eseguito una verifica dell'identità, sulla macchina id-scp, gli restituisce un token di autenticazione, che ha una validità temporale. A quel punto il richiedente potrà utilizzare il token, per l'invio degli UrbanDataset in modalità certificata;
- La **ricezione dell'Urban Dataset** (metodo push) – Figura 13. Il Web Service, una volta ricevuti il token di autenticazione e l'Urban Dataset, esegue dapprima un controllo del token interrogando la macchina id-scp, e in caso di successo scrive il dato su un'opportuna tabella del database installato sul server db-scp.

Sulla macchina id-scp sono state eseguite delle opportune configurazioni affinché il database sul quale immagazzinare le informazioni degli utenti sia proprio quello usato dalla SCP per gestire tutte le altre informazioni. Più precisamente, si è collegato il server di Connect2id sulla macchina id-scp con il database MySql erogato del server db-scp.

Infine, sul server MySql della macchina db-scp è stato creato lo schema Registry. Quest'ultimo consta di due principali componenti: nella prima risiedono le tabelle adibite propriamente alla registrazione e alla gestione degli UrbanDataset, nella seconda invece si trovano le tabelle che utilizza appunto Connect2id.

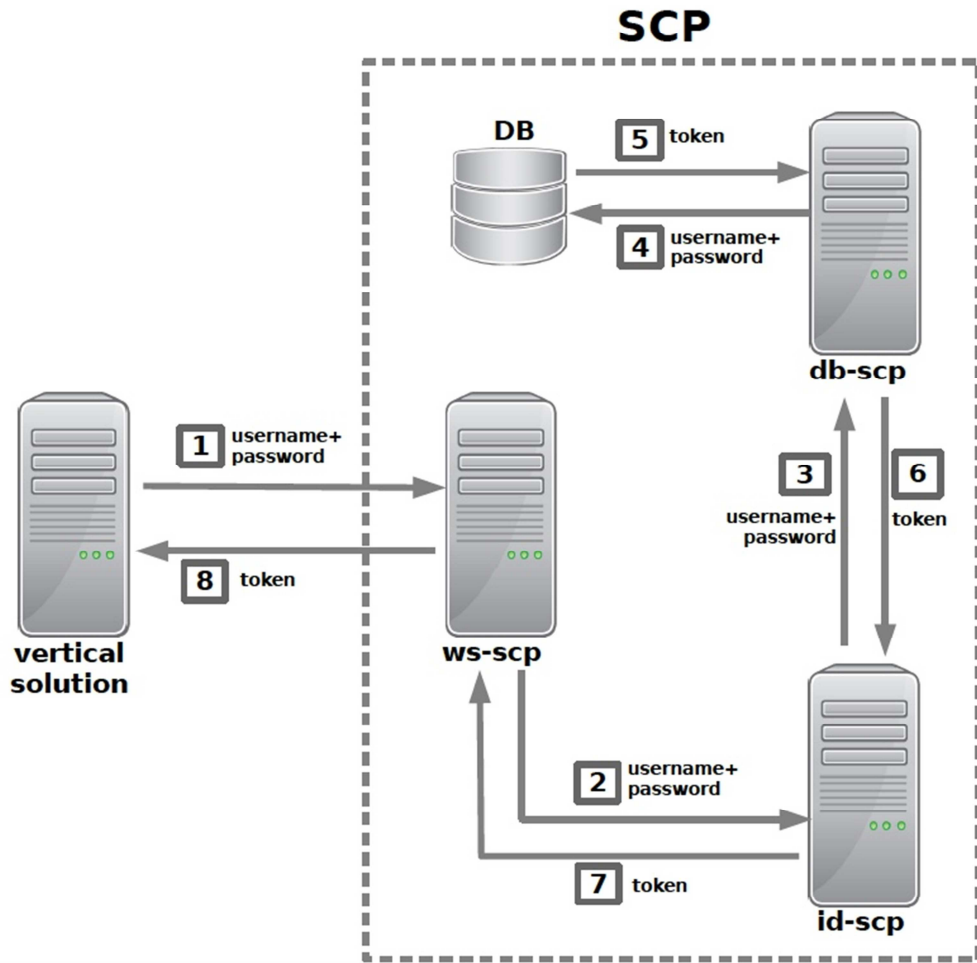


Figura 12 - La figura rappresenta il metodo login sviluppato relativo al controllo dell'identità. In dettaglio il diagramma rappresenta i flussi in entrate e uscita dal Web Service.

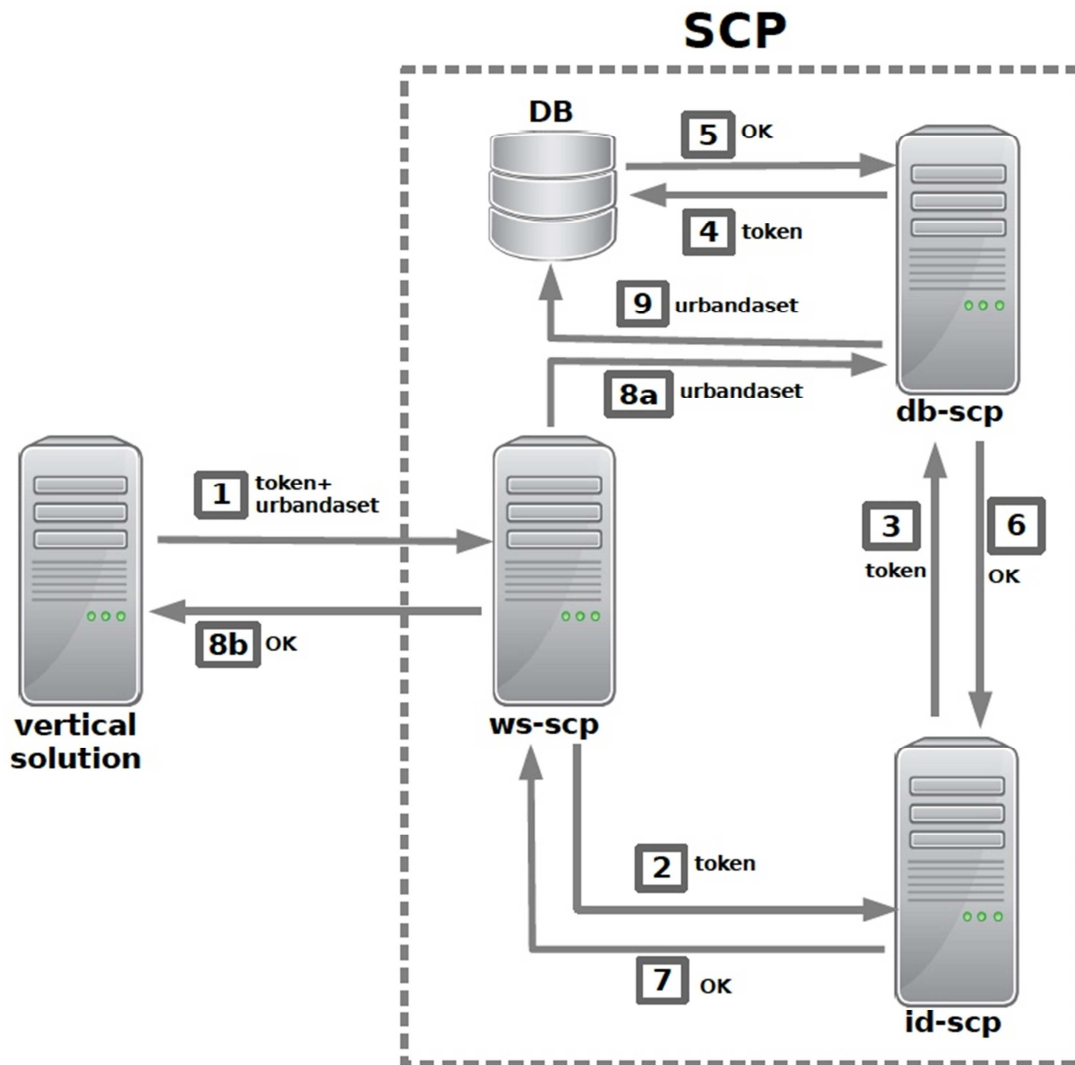


Figura 13 – La figura rappresenta il metodo push sviluppato e inerente la ricezione dell’Urban Dataset. In dettaglio il diagramma rappresenta i flussi in entrate e uscita dal Web Service.

### 3 Conclusioni

Nella terza e ultima annualità del triennio, le Smart City Platform Specification raggiungeranno la loro prima versione stabile e conseguentemente, il prototipo potrà aderire a esse completamente e potrà essere utilizzato nella sperimentazione. Questa sperimentazione prevede l'implementazione di almeno uno dei casi d'uso specifici di piattaforma definiti, per mettere in comunicazione due contesti applicativi verticali.

In particolare

- Per il D6a.1 sono state definite in maniera completa le specifiche per gli strati Information e Semantic e si è iniziato a definire parti importanti degli altri livelli che sono a buon punto. Inoltre sono stati definiti una serie di casi di studio fra cui saranno scelti quelli su cui testare le specifiche. Nell'ultima annualità, le specifiche saranno rifinite, completate e testate, alimentando gli input per la fase implementativa (sotto-obiettivo a.2) e integrando i feedback ricevuti in ritorno per consolidare ognuna delle 5 specifiche.

In particolare, sullo stato information si prevede la progettazione e lo sviluppo di **artefatti per la validazione della semantica** delle istanze degli Urban Dataset (per esempio Schematron, ...), sia generici che specifici per tipo di Urban Dataset, utilizzando le informazioni contenute nell'ontologia.

Gli artefatti generici avranno l'obiettivo di effettuare delle verifiche sulla qualità/coerenza del contenuto delle istanze (per esempio, la coerenza tra il timestamp di inizio e il timestamp di fine periodo: il secondo non può essere antecedente il primo).

Gli artefatti specifici avranno l'obiettivo di verificare la conformità delle istanze alla specifica dell'Urban Dataset a cui aderiscono (questo sarà possibile solo per gli Urban Dataset dichiarati nell'ontologia)

- per il D6a.2, Il lavoro fatto in questa seconda annualità è fondamentale per quello in previsione nella terza e ultima annualità; infatti, quest'ultima sarà incentrata sull'implementazione del prototipo in termini di completa aderenza e verifica delle specifiche Smart City Platform presentate nel sotto-obiettivo a.1. Inoltre, il D6.a2 prevede nell'ultima annualità una fase dedicata alle integrazioni di due o più contesti applicativi verticali (descrittivi nei casi studio inter-solutions del secondo anno) relativi allo contesto Smart Village Casaccia, contesto reale che ben approssima quello di uno Smart District urbano. Quest'ultimo aspetto permetterà di completare lo sviluppo di una SCP in grado di raccogliere e integrare tra loro i dati dai diversi ambiti applicativi.



## 4 APPENDICE A – SCPS Core

Le SCPS Core introducono l'utente delle specifiche SCPS alla comprensione delle stesse.

Le Smart City Platform Specification hanno come obiettivo la definizione di un set di Specifiche che siano punto di riferimento per le Piattaforme ICT in ambito Smart City che vogliono garantire Interoperabilità tra i diversi Contesti Applicativi gestiti.

Per **Interoperabilità** s'intende la capacità di un sistema di scambiare informazioni con altri sistemi differenti, senza errori e senza ambiguità, sia dal punto di vista sintattico e semantico.

Le Smart City sono, in effetti, caratterizzate dalla presenza di dati eterogenei collezionati da una moltitudine di Contesti Applicativi, sulle tematiche energetiche e sociali più disparate, per esempio:

- Smart Building Network: gestione energetica di reti di edifici;
- Smart Lighting: gestione intelligente dell'illuminazione pubblica;
- Smart Street: mobilità abbinata alla gestione energetica dell'illuminazione pubblica;
- Smart Home Network: gestione energetica di reti di case e/o appartamenti tramite domotica;
- Smart Community: coinvolgimento del cittadino tramite i canali sociali;
- Smart Mobility: monitoraggio e gestione del traffico cittadino; ecc.

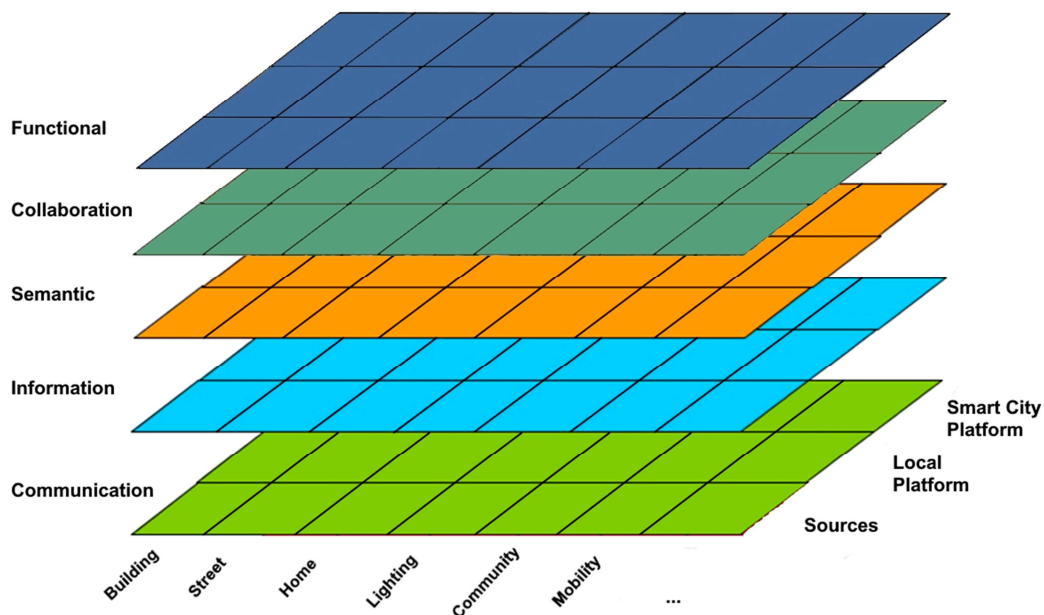
Scopo delle Smart City Platform Specification è permettere a una Piattaforma ICT di città (o distretto) di adottare un approccio comune per comunicare con i diversi Contesti Applicativi del tessuto urbano in modo interoperabile; in particolare, una Piattaforma ICT basata sulle SCPS punta a due importanti traguardi:

- 1) fornire uno strumento alla municipalità, basato su specifiche aperte, per monitorare e gestire i diversi contesti applicativi della città, svincolandosi da soluzioni proprietarie e chiuse;
- 2) divenire ponte di comunicazione interoperabile tra le diverse piattaforme preesistenti della città.

La realizzazione delle SCPS ha richiesto un'approfondita analisi del problema dell'interoperabilità da tre differenti prospettive:

- gli aspetti relativi all'Interoperabilità organizzati e raggruppati su cinque diversi livelli logici;
- la Smart City vista come sistema di sistemi verticali eterogenei (Building, Lighting, SmartHome, ecc.)
- il livello di aggregazione dei dati (Sensori/Campo, Piattaforma ICT Locale, Piattaforma Smart City).

Il risultato dell'analisi è un **Modello di Riferimento per l'Interoperabilità**, rielaborazione dello Smart Grid Architecture Model<sup>5</sup>.



**Figura 14- Modello di Riferimento per l'Interoperabilità**

La figura fornisce un modello di riferimento essenziale al fine di organizzare i molteplici aspetti, riguardanti l'Interoperabilità dello scambio di dati, nei seguenti livelli:

1. **Functional:** concetti chiave e componenti da un punto di vista delle funzionalità architetture;
2. **Collaboration:** set di informazioni necessarie a esprimere le collaborazioni tra i diversi attori in gioco, rappresentante la configurazione della comunicazione interoperabile;
3. **Semantic:** semantica del linguaggio comune in modo tale che il significato dei dati scambiati sia condiviso e l'interpretazione degli stessi priva di ambiguità;
4. **Information:** sintassi del linguaggio comune, ovvero il modello dei dati di riferimento e i formati comuni di rappresentazione digitale da utilizzare;
5. **Communication:** protocolli di trasporto e definizione dell'interfaccia di scambio dati.

Le Smart City Platform Specification sono state definite e organizzate basandosi su questi cinque Livelli, circoscrivendo i diversi insiemi di aspetti dal punto di vista dell'Interoperabilità.

Le SCPS sono quindi modulari e, in generale, possono essere utilizzate in maniera indipendente una dalle altre, permettendone un'adozione graduale.

Le seguenti appendici completano le SCPS Core:

- A. Terminologia e Acronimi (APPENDICE G – SCPS Terminologia);
- B. SCPS Policy per l'uso degli identificatori (APPENDICE H - SCPS Policy per l'uso degli identificatori);
- C. Bibliografia (APPENDICE I - Riferimenti bibliografici).

<sup>5</sup> CEN-CENELEC-ETSI Smart Grid Coordination Group, "Smart Grid Reference Architecture", November 2012

## 5 APPENDICE B – SCPS Functional Level

Le Specifiche del Functional Level (o “Livello Funzionale”) descrivono l’insieme di funzionalità che permettono di raggiungere due principali obiettivi:

1. permettere alla Smart City Platform di recuperare i dati provenienti dalla città/quartiere;
2. abilitare la comunicazione interoperabile tra Solution di Contesti Applicativi diversi.

Per **Smart City Platform** si intende una Piattaforma ICT di integrazione dati che abbia aderito alle specifiche Smart City Platform Specification. Aderendo alle Smart City Platform Specification si predispongono l’infrastruttura a una comunicazione interoperabile tra i diversi Contesti Applicativi urbani.

L’utente ideale della Smart City Platform è l’amministrazione cittadina (o la multi-utility delegata).

Un **Contesto Applicativo** è un insieme concettuale di aspetti tecnologici, sociali ed economici, logicamente connessi, che permette, tramite il monitoraggio e il successivo utilizzo dei dati, di ottenere un miglioramento in termini di gestione efficiente dell’energia e delle risorse e/o in termini di miglioramento della qualità della vita. Per esempio, sono Contesti Applicativi: smart lighting, smart street, smart home, mobility, gestione rete idrica, ecc.

Ogni Contesto Applicativo è gestito usualmente tramite una o più Solution.

Una **Solution** è un insieme di tecnologie e metodologie atte a implementare una o più azioni per ottenere concretamente un miglioramento in termini di gestione dei consumi energetici e di risorse e/o un miglioramento della qualità della vita. Queste azioni consistono tipicamente nell’acquisizione di dati provenienti da una o più reti applicative e dal loro conseguente trattamento (monitoraggio, integrazione, trasformazione, pubblicazione).

Tipicamente una Solution corrisponde alla Piattaforma Locale per la gestione di un singolo Contesto Applicativo e, per questo motivo, si farà riferimento a essa proprio tramite il termine “Solution”.

Per esempio il Contesto Applicativo “Smart Street” che prevede il monitoraggio del traffico per un uso intelligente dell’illuminazione pubblica, potrebbe essere gestito tramite una Solution “Smart Street Platform” corrispondente a una Piattaforma Locale a quel particolare Contesto Applicativo.

Le specifiche Smart City Platform Specification (SCPS) intendono permettere la comunicazione interoperabile tra una Smart City Platform e una o più Solution.

Tale comunicazione interoperabile prevede l’adesione a cinque sotto-specifiche SCPS, corrispondenti ai cinque livelli del modello di riferimento per l’interoperabilità (si veda la “SCPS Core”).

La presente prima specifica, SCPS Functional Level, descrive le funzionalità necessarie a implementare la comunicazione interoperabile da quattro differenti punti di vista:

- l’Architettura di Riferimento;
- i Concetti Chiave (relativi ai Componenti della Smart City Platform);
- gli Utenti;
- una Panoramica delle Funzionalità organizzata per casi d’uso.

Le definizioni date in questa prima parte sono riprese dalla Terminologia (riportata integralmente nel “SCPS Core”) che contiene la descrizione dei termini e dei concetti chiave più importanti.

### 5.1 L'Architettura di Riferimento

Al fine di spiegare le funzionalità della Smart City Platform che, aderendo alle Smart City Platform Specification, garantisce la comunicazione interoperabile con le Solution della città, è indispensabile fornire una descrizione dell'architettura di riferimento.

Si assume, per convenzione, una rappresentazione schematica che prevede:

- una Smart City Platform orizzontale;
- una serie di Solution verticali;

dove, idealmente, i dati raccolti dalla città "saliranno" verticalmente grazie alla Solution nel proprio contesto applicativo di gestione, fino ad arrivare alla Piattaforma ICT di integrazione.

Lo schema seguente rappresenta l'Architettura di Riferimento della Smart City Platform.

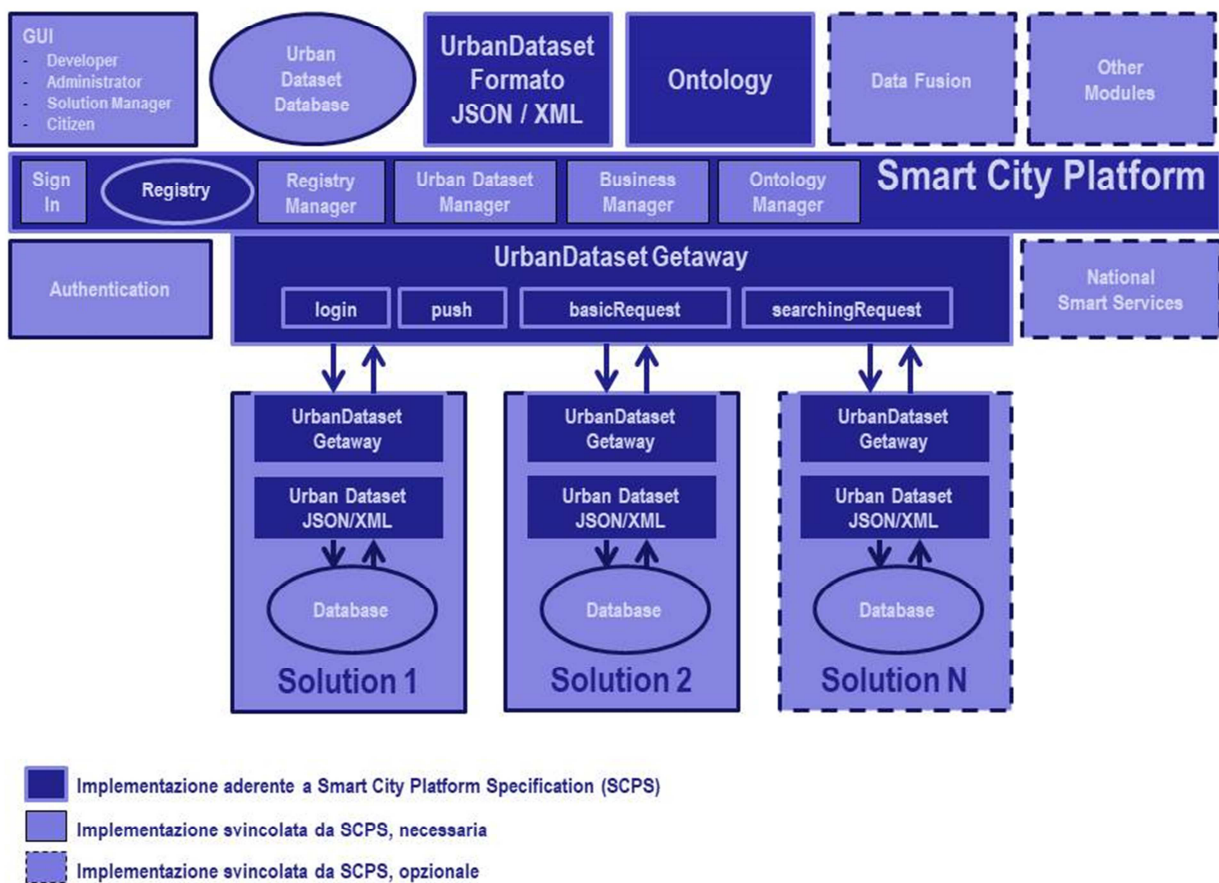


Figura 15 – Architettura di Riferimento della Smart City Platform

La presente sotto-specifica (SCPS Funcional Level) fornisce una spiegazione dei concetti chiave e delle funzionalità che devono essere abilitate per garantire l'interoperabilità nella comunicazione.

Si noti che non tutti i componenti dell'architettura di riferimento devono aderire alle SCPS o, in altre parole, per garantire l'interoperabilità si sono evidenziati i componenti che devono rispettare i requisiti delle Smart City Platform Specification (così come descritto chiaramente nella legenda della figura).

## 5.2 Concetti SCPS-based

I Concetti Chiave sono relativi a componenti che devono obbligatoriamente aderire alle Smart City Platform Specification (SCPS) e verranno introdotti in questo paragrafo facendo riferimento allo schema dell'Architettura di Riferimento (Figura 15).

### 5.2.1 Solution

Una **Solution** verticale corrisponde a una Piattaforma locale a un particolare Contesto Applicativo nella città o quartiere (per esempio, in un contesto "Smart Building Network", si può considerare Solution verticale una "Smart Building Platform" che permetta di gestire il monitoraggio e la diagnostica su reti di edifici).

Ogni Solution verticale attua internamente le proprie politiche di gestione, in modo indipendente dalla Smart City Platform ma, sebbene differenti Solution gestiscano contesti e dati molto diversi tra loro, è evidente come sia invece comune una serie di macro-funzionalità che le caratterizza e che può essere rappresentata schematicamente tramite un modello comune di Solution verticale.



Figura 16 - Modello comune di Solution

Il modello di Solution descrive un insieme di caratteristiche comuni che permettono di individuare processi di gestione dei dati, che sono identici nei diversi Contesti Applicativi e, di conseguenza, individuare un approccio sistematico per implementare l'integrazione basata su una comunicazione interoperabile.

Il modello di Solution verticale, schematizzato nella figura precedente, prevede:

- l'Acquisizione di dati dalle Sorgenti Dati (p.es. rete di sensori per rilevare i consumi elettrici);
- lo Storage (o memorizzazione) di questi dati "grezzi" nel relativo database di acquisizione;
- il Monitoraggio dei dati acquisiti al fine di osservare i flussi dati del Contesto Applicativo in oggetto e prendere consapevolezza sulle azioni migliori da compiere ai fini della gestione;
- l'Elaborazione dei dati grezzi, e in particolare l'Aggregazione di questi dati, per poter astrarre l'informazione e generare anche indicatori che permettano di comprendere agevolmente il Contesto Applicativo.

Volendo portare un esempio, sulla ipotetica Solution "Smart Building Platform":

- l'Acquisizione dei dati avviene andando ad acquisire i consumi elettrici dai quadri elettrici e i consumi termici dalle valvole predisposte a questo utilizzo;
- i dati vengono acquisiti all'interno della Solution e immagazzinati nel DB locale (Storage);
- avviene il Monitoraggio ai fini diagnostici per individuare possibili anomalie;
- l'Elaborazione dei dati quotidiani permette di effettuare Aggregazioni su base temporale e/o spaziale, andando a definire, per esempio, il numero di anomalie settimanale per un particolare edificio monitorato;

- questo Indicatore permette così capire, nel tempo, se la gestione effettuata a livello di singolo palazzo e a livello di reti di edifici è efficace e permette effettivamente una più corretta gestione dell'energia.

In questa direzione (dal basso verso l'alto), salendo dallo specifico Contesto Applicativo verticale verso la Smart City Platform orizzontale, i dati possono essere inviati e utilizzati a livello di quartiere o città, per avere una panoramica sullo stesso tessuto urbano da diversi punti di vista e per riutilizzare gli stessi dati all'interno di un diverso Contesto Applicativo verticale.

Senza questo modello, per ogni Solution di ogni Contesto Applicativo, si dovrebbe pensare a un apposito meccanismo di comunicazione dei dati verso la piattaforma ICT di distretto e a un sistema di integrazione apposito implementato ad hoc.

Questo modello di Solution, invece, è fondamentale per la definizione di un meccanismo di integrazione standard (definito tramite le specifiche SCPS) per la comunicazione interoperabile tra Solution e Smart City Platform e che può essere condiviso e utilizzato per tutte le Solution dei diversi Contesti Applicativi Verticali. I prossimi concetti sono la chiave per la comprensione di tale meccanismo, definito per abilitare la comunicazione interoperabile.

### 5.2.2 Urban Dataset

La comunicazione interoperabile tra sistemi, oggetto di queste specifiche, è basata sullo scambio di dati eterogenei in ambito Smart City: trattasi di set di dati provenienti da Contesti Applicativi diversi, che sono talmente variegati tra loro e in una quantità così elevata che risulterebbe impossibile creare una descrizione omogenea, valida per tutti, per contenerli tutti. Si pensi, per esempio, a dataset contenenti i dati relativi al flusso del traffico, oppure a dataset descrittivi i dati meteo, o i consumi elettrici o termici di una rete di edifici, oppure i dati relativi alle anomalie della rete idrica, ecc..

Benché questi dati siano così diversi tra loro, è altrettanto evidente che abbiano aspetti comuni:

- sono relativi allo stesso tessuto urbano, classificabili in categorie e sotto-categorie;
- sono esprimibili mediante dataset (tabelle di dati);
- sono caratterizzati sempre da una collocazione spazio-temporale.

Sebbene questi aspetti non vadano a toccare il dato specifico nel particolare Contesto Applicativo verticale in cui è stato creato, sono sufficienti per permettere di definire un approccio alla comunicazione che permetta di convergere verso una descrizione condivisa che consenta la comunicazione interoperabile.

Questo approccio trova il suo fulcro nel concetto di "Urban Dataset".

Un **Urban Dataset** (UD) è la rappresentazione del dato scambiato tra Solution verticali della città e Smart City Platform, secondo le specifiche SCPS, ed è caratterizzato:

- da una descrizione semantica univoca e centralizzata, tramite l'Ontologia;
- da un formato definito da:
  - o Modello Dati Astratto, che definisce il contenuto in modo indipendente dalla sintassi;
  - o Implementazione Sintattica di tale modello, al momento fornito secondo due sintassi di riferimento (JSON e XML);
- da un servizio di trasporto, l'UrbanDataset Gateway, basato su Web Service, REST e SOAP.

Questi tre aspetti verranno spiegati dettagliatamente nelle specifiche SCPS, rispettivamente, nei livelli Semantic, Information e Communication.

Si noti che l'UrbanDataset Gateway suddetto utilizza nella propria interfaccia, come riferimento di UrbanDataset, non l'identificatore di UD, ma l'identificatore della **collaborazione**, inteso come identificatore della produzione di un particolare UD, da parte di un particolare Solution, in una specifica città/quartiere, in uno specifico momento temporale.

In ambito SCPS, quindi, il termine “*Collaboration*” non è sinonimo di “*UrbanDataset*”: è qualcosa di ancora più specifico e non ambiguo, perché si riferisce non solo a un particolare *UrbanDataset* ma anche a chi quell’ *UrbanDataset* l’ha prodotto in un periodo specifico.

### 5.2.3 Ontology

L’**Ontologia** (o *Ontology*) è un componente che si occupa di definire la struttura sintattica e semantica degli *Urban Dataset*, nonché classificarli in categorie e sotto-categorie.

L’*Ontologia* è un componente indipendente, esterno alla *Smart City Platform*, per consentirne un uso condiviso con altre piattaforme SCPS-based, in modo tale che un processo di convergenza sia alla base di ogni comunicazione. Un team di esperti (ENEA) si occupa della manutenzione di questo componente.

Essendo un componente che fornisce l’interpretazione semantica centralizzata dei dati, l’*Ontologia* ricopre un ruolo di primaria importanza la cui definizione è contenuta in un’apposita sotto-specifica “*Smart City Platform Specification (SCPS) – Semantic Level*”.

### 5.2.4 Implementazioni XML/JSON

Per scambiare gli *Urban Dataset*, le specifiche SCPS mettono a disposizione due **implementazioni**, in XML e JSON.

Non vi sono altri modi considerati validi per rappresentare gli *Urban Dataset*.

Si noti che un *Urban Dataset* ha una sola definizione (nell’*Ontologia*) e che tale modello dati può essere rappresentato da uno dei due implementazioni messi a disposizione, appunto XML o JSON.

Scegliere un’implementazione piuttosto che un’altra non cambia il contenuto delle informazioni, che rimangono le stesse: sebbene XML fornisca uno strumento più potente e rigoroso per la rappresentazione formale dei dati nonché della validazione che è possibile effettuare, JSON risulta sicuramente più leggero.

Una volta scelta l’implementazione ritenuta più efficace, tra XML e JSON, ogni *Solution* potrà esportare dal proprio database locale l’*Urban Dataset* da inviare o potrà importare l’*Urban Dataset* ricevuto.

La *Smart City Platform*, invece, riceverà gli *Urban Dataset* dalle *Solution* a scopo di monitoraggio e ne permetterà il recupero, abilitando o meno l’accesso, per consentire una comunicazione inter-*Solution*.

Le implementazioni XML e JSON sono, di fatto, il linguaggio formale per la rappresentazione dei dati secondo le specifiche SCPS, la loro definizione è contenuta in un’apposita sotto-specifica:

“*Smart City Platform Specification (SCPS) – Information Level*”.

### 5.2.5 Trasporto *UrbanDataset Gateway*

Il servizio di **trasporto** dei dati, che consente di inviare e ricevere *Urban Dataset*, è definito nelle specifiche SCPS tramite la definizione di un *Web Service: UrbanDataset Gateway*.

Il *Web Service* viene fornito nelle SCPS tramite la definizione di:

- tre pattern: push, request/response, publish/subscribe;
- un’interfaccia comune che prevede i metodi: login, push, basicRequest, searchingRequest;
- un’implementazione REST dell’interfaccia che permette la trasmissione del formato JSON;
- un’implementazione WDSL dell’interfaccia che permette la trasmissione del formato XML.

Si noti che, implementando il *web service* sia lato *Smart City Platform* che lato *Solution*, si abilitano i diversi pattern di trasporto in entrambe le direzioni, permettendo di configurare ogni collaborazione secondo le proprie esigenze.

L’ “*UrbanDataset Gateway*” corrisponde al servizio di trasporto dei dati ed è descritto dettagliatamente nella specifica “*Smart City Platform Specification – Communication Level*”.

### 5.2.6 Registry

Il **Registry** è il componente della Smart City Platform che permette di gestire:

- le Solution registrate;
- gli Urban Dataset definiti (nell'Ontologia);
- gli Urban Dataset che devono essere prodotti e da quali Solution;
- gli Urban Dataset che devono essere acceduti e da quali Solution;
- le policy dei dati;
- le modalità di comunicazione supportate e configurate (in termini di Formato e Trasporto);
- altre informazioni complementari alle precedenti.

Il Registry è, fondamentalmente, un database la cui funzione è quella di essere strumento per gestire le informazioni relative alle collaborazioni tra Smart City Platform e Solution, dal punto di vista di chi produce e di chi accede agli Urban Dataset.

Il Registry corrisponde, quindi, al servizio di persistenza delle collaborazioni e sarà descritto dettagliatamente nella specifica "Smart City Platform Specification – Collaboration Level".

## 5.3 Concetti No-SCPS necessari

I "Concetti No-SCPS necessari" sono relativi a componenti la cui implementazione non trova particolari vincoli nelle SCPS per l'Interoperabilità ma le cui funzionalità devono essere previste poiché necessarie al corretto utilizzo della Smart City Platform, utilizzandone tutte le potenzialità.

### 5.3.1 Sign In e Authentication

La **registrazione** (Sign In) di una Solution nel sistema della Smart City Platform può avvenire in due modi:

1. tramite inserimento dell'utente Administrator;
2. tramite registrazione della Solution da parte del Solution Manager e successiva accettazione da parte dell'Administrator.

In entrambi i casi, le informazioni inserite sono le stesse e corrispondono a:

- Anagrafica della Solution e del referente (o dei referenti);
- Credenziali di accesso alla Smart City Platform;
- Credenziali di accesso alla Solution, nel momento in cui esponga il web service previsto;
- Urban Dataset che verranno prodotti;
- Urban Dataset che verranno acceduti;
- Configurazione del Trasporto;

(ognuno di questi gruppi di informazioni trova l'esatta ubicazione nelle tabelle che saranno descritte nella sotto-specifica "SCPS Collaboration Level").

Una volta inserite tutte le informazioni relative alla Solution, l'utente Administrator potrà convalidarle e renderle effettive, fino a quel momento la comunicazione non potrà essere accettata (par.0).

Le "Credenziali di accesso alla Smart City Platform" preparate per la Solution, consentono di:

- accedere ai servizi utente (GUI, si veda par. seguente);
- accedere al web service per inviare o recuperare Urban Dataset.

**Authentication** è il componente che si occupa di autenticare l'utente nel momento in cui esso accede al sistema. L'interfaccia di accesso è strettamente relativa alle specifiche "SCPS Communication Level"



(tramite username e password e l'utilizzo di un token per le chiamate successive). Come il sistema di autenticazione sia implementato internamente, invece, non è vincolato alle Specifiche SCPS.

### 5.3.2 Ontology Manager

Internamente alla Smart City Platform, il componente Ontology Manager avrà il compito di gestire la connessione con l'Ontologia, provvedendo a:

1. la sincronizzazione degli Urban Dataset nel momento in cui questi vengano creati, modificati o cancellati dall'Ontologia;
2. il recupero delle definizioni semantiche e sintattiche degli Urban Dataset, quando necessario;
3. classificare e ricercare gli Urban Dataset utilizzando la tassonomia predefinita;
4. navigare gli Urban Dataset definiti nell'Ontologia per decidere quali supportare nella SCP.

Queste funzionalità verranno definite nel livello SCPS Semantic Level che descriverà in dettaglio le interfacce per interagire con l'Ontologia.

### 5.3.3 Urban Dataset Manager

L'**UrbanDataset Manager** è un componente che si occupa di gestire gli UrbanDataset, in arrivo e in uscita, eseguendo le seguenti trasformazioni:

1. trasformazione da formati JSON/XML a struttura dati (unmarshalling) fedele al modello comune;
2. persistenza della struttura dati degli UrbanDataset nell'UrbanDataset Database;
3. recupero degli UrbanDataset dall'Urban Dataset Database verso struttura dati comune;
4. trasformazione da struttura dati comune al formato JSON o XML (marshalling).

Si noti che queste funzionalità:

- prevedono l'uso del modello comune di UrbanDataset delle specifiche SCPS Information Level;
- utilizzano una struttura dati, che serve a gestire il modello comune, che può essere sviluppata in qualunque linguaggio di programmazione software.

### 5.3.4 Urban Dataset Database

L'UrbanDataset Database è la banca dati che si occupa di memorizzare tutti i dataset provenienti dalle diverse Solution. Non vi sono particolari restrizioni su questo componente se non che debba permettere:

1. la persistenza del modello comune di uno specifico Urban Dataset;
2. il recupero di uno specifico Urban Dataset;
3. di effettuare ricerche spaziali, temporali e di categoria.

### 5.3.5 Business Manager

Il componente **Business Manager** è un componente che si occupa di abilitare la comunicazione inter-solution: in altre parole permette, ricevuto un Urban Dataset da una prima Solution, di abilitarne l'accesso per una seconda Solution.

Il Business Manager è quindi paragonabile a un trigger che rimane in ascolto di nuovi Urban Dataset in arrivo e, una volta ricevuto il dato, attiva una funzionalità che si riassume nei seguenti passi:

1. Validazione dell'UrbanDataset rispetto alla Solution1 producer (la collaborazione relativa a questo UD è correttamente specificata per questa Solution1?)

2. Verifica, nella lista degli accessi, che a questo particolare UrbanDataset, prodotto da questa particolare Solution1 (ovvero “collaboration”) vi sia una o più Solution2 aventi il permesso di recuperarlo;
3. Per ogni Solution2, avente l’accesso all’UD sopraggiunto, se è impostata nella comunicazione un pattern di tipo “PUSH”, verrà inviato l’UD suddetto, altrimenti sarà possibile a quella Solution2 recuperarlo, con pattern “REQUEST/RESPONSE”.

Questa funzionalità è spiegata in maniera più dettagliata nel paragrafo “Panoramica Funzionalità”.

### 5.3.6 GUI

Le **Interfacce Utente** (o Graphical User Interface, **GUI**) permettono alle diverse tipologie di utente di poter accedere alla propria Area di Gestione.

Questo è un componente la cui definizione è strettamente relativa all’implementazione: le presenti specifiche “SCPS Functional Level” introducono questo componente dal punto di vista architettonico, mentre la sotto-specifica “SCPS Collaboration Level” fornirà una descrizione delle GUI fondamentali dettagliata nonché il Data Model del database Registry su cui sviluppare ognuna delle funzionalità previste per ogni tipologia di utente.

## 5.4 Concetti No-SCPS opzionali

I “Concetti No-SCPS opzionali” sono relativi a componenti la cui implementazione non deve rispettare nessuna sotto-specifica SCPS per l’interoperabilità e che non è strettamente necessaria per il corretto funzionamento della Smart City Platform ma permette di aggiungere nuove funzionalità.

Nell’Architettura di Riferimento si è ritenuto interessante specificare i seguenti componenti “opzionali”:

- **Data Fusion module:** modulo avente i permessi per accedere a una serie di Urban Dataset, a fine di fondere questi dati per creare nuova conoscenza ed eventualmente memorizzare l’output risultante come nuovi UrbanDataset da utilizzare in appositi servizi per la città; i nuovi dati generati dagli Urban Dataset potrebbero per esempio essere KPI per misurare le prestazioni del distretto/città (cosa di cui non si occupano, al momento, le SCPS);
- **National Smart Services:** con questa dicitura si è voluto sottolineare come una singola smart city platform potrebbe, unendosi a una rete nazionale o europea di raccolta KPI (p.es. il Public Energy Living Lab, PELL), partecipare e usufruire a un monitoraggio energetico su vasta scala, a scopo di benchmarking e trasmissione di know-how sulle esperienze Smart City;

**Other Modules:** si è voluto affermare, andando a specificare questo componente generico, la possibilità, da parte della Smart City Platform, di prestarsi a essere estesa con nuovi moduli software che possano accrescere le funzionalità della piattaforma in modo scalabile.

## 5.5 Utenti

Gli Utenti (User) previsti come utilizzatori della Smart City Platform sono quattro:

- 1) **Developer:** si occupa di amministrare la piattaforma dal punto di vista tecnico e dei componenti software che costituiscono la piattaforma, in particolare della prima configurazione (par.0);
- 2) **Administrator:** si occupa di amministrare la piattaforma orizzontale della città (tipicamente è un rappresentante della municipalità o della multi-utilities che gestisce la città/distretto) e quindi gestisce le collaborazioni tra Smart City Platform e le diverse Solution verticali;
- 3) **Solution Manager:** responsabile della piattaforma verticale che si conetterà alla Smart City Platform, accede ai dati descrittivi la situazione della propria Solution (anagrafica, credenziali di accesso, Urban Dataset da produrre o accessibili, parametri della comunicazione e dati inviati fino a quel momento);

4) **Citizen:** utente Cittadino generico che accede ai servizi offerti dalla piattaforma (se presenti).

Per ogni Utente è prevista un'apposita Area di Gestione con visualizzazione dei dati di propria competenza.

Particolarmente importante è l'utente Administrator in quanto gestisce gli Urban Dataset supportati dalla Smart City Platform, le registrazioni delle Solution verticali e le diverse associazioni Solution-UrbanDataset, sia dal punto di vista della produzione che dal punto di vista dell'accesso.

Si noti che le Interfacce Utente (o Graphical User Interface, GUI) saranno introdotte nel par. 5.3.6 e descritte dettagliatamente nella specifica "SCPS Collaboration Level" e saranno organizzate e definite su questi possibili utenti.

## 5.6 Use Case diagram

Dopo avere presentato l'Architettura di Riferimento, gli Utenti e i Concetti Chiave, si intende ora fornire una panoramica delle funzionalità tramite alcuni use case diagram.

Questo percorso ripercorre tutte le fasi inerenti la Smart City Platform e tutte le azioni che i diversi Utenti devono compiere per interagire con questo sistema.

L'obiettivo che si prefigge questo percorso di esempio è di dimostrare come, attraverso la Smart City Platform e, nel particolare, per mezzo dei componenti presentati e delle funzionalità relative, si possano raggiungere due macro-obiettivi:

1. Permettere all'utente Administrator di effettuare il monitoraggio di ogni Solution;
2. Permettere a due utenti Solution Manager di impostare una comunicazione di dati da una Solution verticale 1 a una Solution verticale 2.

Le fasi inerenti la Smart City Platform sono le seguenti:

- Configurazione SCP per Città/Quartiere;
- Configurazione di Solution;
- Avvio Comunicazione Interoperabile;
- Cancellazione Solution.

Si ricorda che l'Ontologia è un componente architettonico indipendente dalla SCP che sarà curato da un team di esperti. L'intersezione tra Ontologia e SCP è la definizione e certificazione di UrbanDataset. Una piattaforma SCP farà riferimento all'Ontologia, preesistente, senza poter effettuare modifica su di essa.

Tutti i riferimenti alle Interfacce Utente (o GUI, Graphical User Interface) saranno approfonditi nelle SCPS Collaboration Level, dove sono descritte in maniera dettagliata (nel PAR2017) le GUI fondamentali per poter implementare le funzionalità basilari della SCP.

## 5.7 Configurazione SCP

L'installazione di una Smart City Platform basata su specifiche SCPS prevede la creazione di una nuova istanza di Smart City Platform.

Nel momento in cui viene creata, deve essere effettuata la prima configurazione della Smart City Platform.

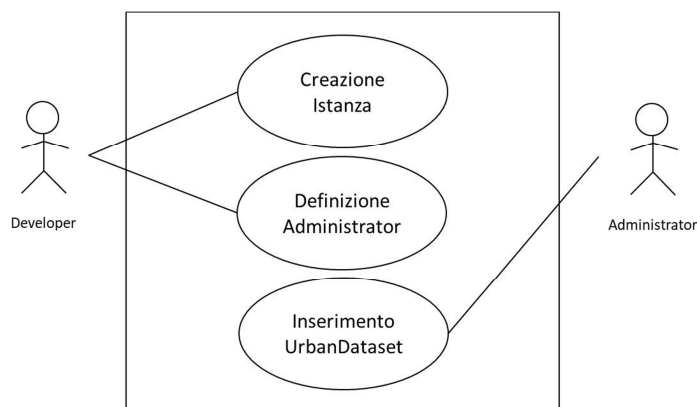


Figura 17 – Fase1: Configurazione SCP

### 5.7.1 Creazione Istanza

Il Data Model del Registry sarà fornito con le specifiche “SCPS Collaboration Level” già configurato con le impostazioni iniziali della Smart City Platform ma tale Data Model dovrà essere istanziato per una particolare città/quartiere, andando a inserire/modificare le informazioni presenti nella tabella *smart\_city\_administration*. L'utente Developer si occupa di istanziare una nuova SCP.

### 5.7.2 Definizione Administrator

Nessun utente “Administrator” è definito dopo aver istanziato una SCP. L'unico utente presente è l'utente “Developer” che DEVE definire, in prima istanza, quello che sarà l'utente “Administrator” (uno o più). Questa funzionalità è permessa dall'interfaccia utente (GUI) definita per l'utente Developer.

### 5.7.3 Inserimento UrbanDataset

Al fine di inserire gli UD da supportare attraverso la SCP, E' RACCOMANDATO che l'Administrator decida quali UrbanDataset inserire tra quelli già definiti nell'Ontologia.

Questa funzionalità è implementata dall'OntologyManager (par. 5.3.2), tramite un sistema di navigazione degli UD definiti e successivo inserimento, ma PUO' essere operazione manuale effettuata dall'utente Administrator tramite l'apposita GUI.

A questo punto la prima configurazione della SCP è completa.

L'utente Administrator, a questo punto, vedrà:

- nella vista degli Urban Dataset, tutti gli UD inseriti;
- nella vista delle Solution registrate, una lista vuota.

Di default, solamente un UrbanDataset è definito nella SCP: l'UD “Whatever”.

Gli aspetti legati alla definizione degli UrbanDataset sono descritti nelle specifiche “SCPS Semantic Level”.

## 5.8 Configurazione Solution

Per poter abilitare la comunicazione tra Smart City Platform e Solution, deve essere effettuata la configurazione di una o più Solution.

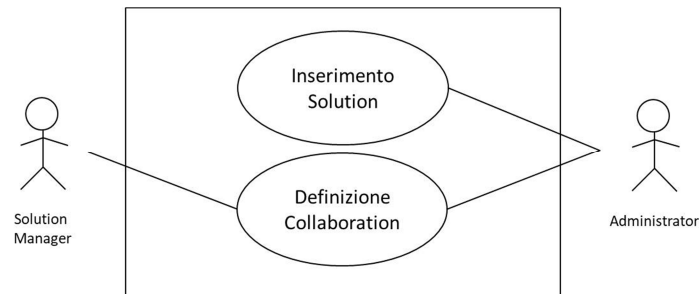


Figura 18 – Fase2: Configurazione Solution

### 5.8.1 Inserimento Solution

La configurazione di una nuova Solution può avvenire in due modi:

1. l'utente Administrator inserisce manualmente i dati relativi alla Solution (OBBLIGATORIA);
2. la Solution si registra e attende la moderazione dell'utente Administrator (OPZIONALE).

### 5.8.2 Definizione Collaboration

Una volta che una Solution è registrata, l'Administrator può abilitare su quella Solution la produzione di un particolare UD (denominata "collaboration", si veda par.5.2.2) o permettere l'accesso a un UD.

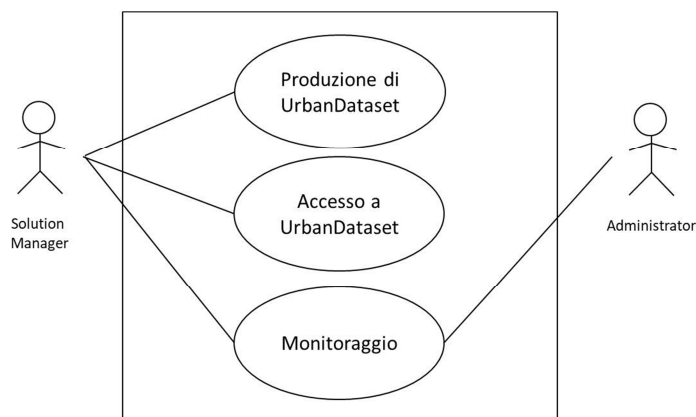
Questa associazione Solution-UD, in produzione e accesso, può essere definita

1. in maniera diretta e immediata da parte dell'Administrator;
2. in maniera indiretta e moderata, permettendo agli utenti la navigazione degli UD presenti nella SCP, potendo poi effettuare richiesta di produrre o accedere a un particolare UD, attendendo la successiva abilitazione dell'utente Administrator.

Tutti gli aspetti legati alla configurazione della collaboration saranno descritti nelle specifiche "SCPS Collaboration Level" nel PAR2017, sia dal punto di vista delle interfacce GUI utilizzate, sia dal punto di vista del Registry utilizzato per memorizzare ogni informazione.

## 5.9 Comunicazione Interoperabile

Una volta che le configurazioni della Smart City Platform e di una o più Solution sono state ultimate, è possibile avviare la fase di comunicazione interoperabile vera e propria; essa si basa principalmente sul formato di rappresentazione dell'UrbanDataset (si veda "SCPS Information Level") e sul trasporto dei dati tramite l'UrbanDataset Gateway (si veda "SCPS Communication Level").



**Figura 19 – Fase3: Comunicazione Interoperabile**

### 5.9.1 Produzione di UrbanDataset

Ogni qualvolta una generica Solution verticale è stata configurata, presso la SCP, per produrre un particolare UrbanDataset, viene specificato anche il periodo (*TimestampStart*, *TimestampEnd*) entro il quale dovrà produrre l'UD.

Quando il momento *TimestampStart* per far partire la produzione dell'UD è raggiunto, la Solution lo renderà disponibile alla SCP pubblicandolo e/o inviandolo, nel formato comune deciso e tramite il canale di trasporto e frequenza configurati, fino al momento *TimestampEnd*.

Nel dettaglio, la produzione di UrbanDataset da parte di una Solution è riassumibile in questa serie di passi:

1. esportazione dal proprio database locale delle informazioni che si devono inviare verso la SCP, aggregandole nell'UrbanDataset configurato;
2. rappresentazione dell'UrbanDataset secondo il formato comune JSON o XML;
3. utilizzo del web service *UrbanDatasetGateway*, SOAP o REST, per l'invio dell'UrbanDataset alla SCP (secondo il pattern e il protocollo configurati);
4. la SCP riceve l'UrbanDataset e lo immagazzina nella propria bancadati.

Tutti gli aspetti legati alla rappresentazione dell'UrbanDataset in formato JSON o XML sono descritti nelle specifiche "SCPS Information Level", mentre il web service *UrbanDatasetGateway*, SOAP o REST, è descritto nelle specifiche "SCPS Communication Level".

### 5.9.2 Accesso a UrbanDataset

Ogni qualvolta una generica Solution verticale deve accedere a un particolare UrbanDataset, viene specificato anche il periodo (*TimestampStart*, *TimestampEnd*) entro il quale potrà accedere all'UD.

Quando il momento *TimestampStart* per l'accesso dell'UD è raggiunto, la Solution può accedere alla SCP recuperandolo/ricevendolo, nel formato comune deciso e tramite il canale di trasporto e frequenza configurati, fino al momento *TimestampEnd*.

Nel dettaglio, l'accesso a un UrbanDataset da parte di una Solution è riassumibile in questa serie di passi:

1. la SCP esporta l'UrbanDataset dalla propria bancadati nel formato JSON o XML configurati;
2. la Solution tramite l'utilizzo del web service *UrbanDatasetGateway*, SOAP o REST, richiede o riceve l'UrbanDataset dalla SCP (secondo il pattern e il protocollo configurati);
3. importazione nel proprio database locale delle informazioni contenute nell'UD.

Tutti gli aspetti legati alla rappresentazione dell'UrbanDataset in formato JSON o XML sono descritti nelle specifiche "SCPS Information Level", mentre il web service *UrbanDatasetGateway*, SOAP o REST, è descritto nelle specifiche "SCPS Communication Level".

### 5.9.3 Monitoraggio

Gli UrbanDataset inviati e acceduti sono soggetti a monitoraggio:

- l'utente Solution Manager può vedere tutti gli UrbanDataset che la propria Solution ha prodotto verso la Smart City Platform o ha recuperato dalla Smart City Platform;
- l'Administrator può vedere tutti gli UrbanDataset di tutte le Solution, sia in fase di produzione che in fase di accesso, e ha quindi una fotografia completa dei dati gestiti dalla SCP.

Tutti gli aspetti legati alla definizione dei servizi utente tramite GUI saranno descritti nelle specifiche "SCPS Collaboration Level" nel PAR2017.

## 5.10 Cancellazione Solution

Una volta che le collaborazioni tra una Solution e SCP hanno raggiunto un loro naturale termine, sia in fase di produzione sia in fase di accesso ai dati, è possibile cancellare la Solution dal sistema, andando a gestire anche i dati prodotti da quella Solution.

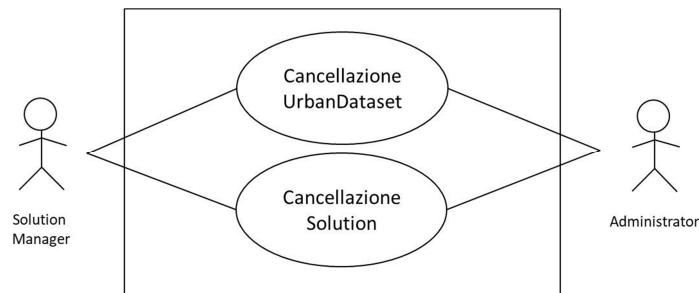


Figura 20 – Fase4: Cancellazione Solution

### 5.10.1 Cancellazione UrbanDataset

Prima di cancellare una Solution dal sistema è bene capire quale sarà il destino riservato agli UrbanDataset prodotti da quella Solution.

Si prospettano diverse ipotesi, dipendenti dalla policy sugli UrbanDataset prodotti:

1. Se i dati sono di proprietà della Solution, essa può decidere di eliminarli;
2. Se i dati sono di proprietà della SCP, essa può decidere di eliminarli o conservarli;
3. Se i dati sono di proprietà di una terza parte, verranno conservati e sarà la terza parte a decidere.

Quindi, si noti bene che, una Solution può aver prodotto UrbanDataset che potrebbero rimanere presenti nella SCP anche dopo la cancellazione dal sistema della Solution suddetta.

Per approfondimento sulla policy degli UrbanDataset e sui servizi utente (GUI) per gestire le associazioni SCP-Solution, si farà riferimento alle specifiche "SCPS Collaboration Level" nel PAR2017.

### 5.10.2 Cancellazione Solution

Una volta gestita la cancellazione degli UD prodotti da una Solution, è possibile cancellare la Solution.

Anche in questo caso si prospettano diverse ipotesi:

1. Se non sono presenti più UrbanDataset prodotti da questa Solution e l'utente Administrator o l'utente Solution Manager decide di rimuovere questa Solution definitivamente, allora la Solution e tutti i dati a essa connessi vengono rimossi;

2. Nel caso vi siano collaboration attive, in produzione, o vi siano dati di proprietà della Solution, l'operazione non viene permessa e la Solution rimane nel sistema;
3. A discrezione dell'utente Administrator la Solution può essere disabilitata (sarà ancora presente nel sistema assieme ai dati a essa connessi, ma non potrà né produrre né accedere a dati della SCP).

Tutti gli aspetti legati alla definizione delle GUI saranno descritti nelle specifiche "SCPS Collaboration Level" e rilasciate a fine PAR2017.



## 6 APPENDICE C – SCPS Collaboration Level

Le specifiche del **Collaboration Level** (o “Livello Collaborazione”) descrivono l’insieme di informazioni che caratterizzano la collaborazione tra la piattaforma Smart City Platform (SCP) e una qualsiasi Solution.

L’architettura di riferimento, dove viene descritta la relazione tra Smart City Platform orizzontale e Solution verticali, nonché una spiegazione dei concetti chiave, sono presentati nelle SCPS Functional Level.

Le informazioni relative alle collaborazioni tra Smart City Platform e Solution:

- sono gestite tramite le interfacce utente (Graphical User Interface) che la piattaforma mette a disposizione per le diverse tipologie di utente, e
- vengono immagazzinate nella banca dati Registry.

Le **Graphical User Interface (GUI)** sono le interfacce web che gli utenti devono utilizzare per interagire con la piattaforma SCP. Nel capitolo 1 si darà una descrizione dettagliata delle funzionalità fondamentali, che trovano puntuale riscontro nella struttura del Registry, per ogni tipologia di utente considerato.

Il **Registry** è un DataBase che contiene le caratteristiche relative alle collaborazioni tra Smart City Platform e Solution verticali, dal punto di vista di produzione e accesso agli Urban Dataset, nonché le informazioni relative a tutti gli utenti. Nel capitolo 2 si darà una descrizione dettagliata di ogni tabella SQL prevista.

Si sottolinea che un’implementazione basata sulla Smart City Platform Specification “Collaboration Level” non è vincolata a una particolare scelta tecnologica per la gestione della persistenza ma DEVE implementare le funzionalità descritte per le GUI e DEVE implementare il Data Model del Registry che accompagna questa specifica sottoforma di dump SQL e la cui ultima versione è disponibile online.

Le specifiche SCPS Collaboration Level sono in via di definizione e oggetto di sviluppo e consolidamento, nonché utilizzate nell’implementazione di piattaforma ICT, nel corso del PAR2017, alla fine del quale ne verrà rilasciata la prima versione.

## 7 APPENDICE D – SCPS Semantic Level

Le Specifiche del **Semantic Level** (o “Livello Semantico”) definiscono la semantica comune a cui aderire per permettere che lo scambio di dati (UrbanDataset) tra una Smart City Platform e le Solution verticali sia non ambiguo e quindi permetta di raggiungere una comunicazione interoperabile tra i sistemi.

Il core del Livello Semantico è l’**ontologia**: essa definisce gli Urban Dataset disponibili e ne rappresenta il riferimento principale, dato che tali definizioni, per ragioni di flessibilità, non possono essere documentate in maniera univoca dagli schemi definiti all’Information Level.

Per tale ragione l’ontologia è conservata e gestita in maniera centralizzata rispetto a una o più Smart City Platform e divenire, in questo modo, punto condiviso di convergenza semantica.

Per lo stesso motivo, il suo essere accentratore delle definizioni semantiche degli Urban Dataset, rende l’ontologia anche un nodo critico, la cui modifica e aggiornamento devono essere fatti con attenzione, attraverso un processo rigoroso e controllato.

Per tale ragione le specifiche definiscono:

- un Comitato di gestione dell’ontologia;
- una procedura formale per sottoporre la richiesta d’inserimento al Comitato;
- un template per sottoporre la richiesta

L’ontologia è un **file OWL** e può essere navigato e consultato tramite programmi esterni (siano essi desktop come Protégé o visualizzatori on line come WebVOWL). Inoltre, per facilitarne l’interrogazione, viene fornita una libreria software che permette di consultarla senza dover conoscere SPARQL.

In questo livello, inoltre, si definiscono anche le **Code List** che associano liste di codici utilizzate dagli Urban Dataset al loro significato. Questi sono rappresentati con il formato standard OASIS genericode (7). Si noti che il modulo di richiesta di nuovi Urban Dataset contiene anche un foglio per l’inserimento di nuove code list.

### 7.1 Descrizione dell’ontologia e modello semantico

L’ontologia pone centralmente il concetto di **UrbanDataset (UD)**.

**Tale definizione di UD** è definita come una sottoclasse di **Entity** della Prov Ontology<sup>6</sup> (8)

Con tale concetto si riferisce ogni dato, aggregato o meno, che un contesto applicativo è in grado di elaborare a partire dai dati raccolti nello Smart District. Questo è il nodo nevralgico di tutta la comunicazione e l’informazione principale che deve essere scambiata all’interno dell’infrastruttura.

Ontologie già esistenti, definite e testate in altri contesti, sono state utilizzate per associare, quando possibile, ai concetti e dati di Urban Dataset altre informazioni quali, per esempio, le unità di misura utilizzate per i valori raccolti dai sensori. In questo modo si hanno diversi vantaggi. Uno di questi è l’uso di una risorsa già testata e che quindi ha superato la validazione dovuta all’uso di tale strumento in un’altra applicazione. Un altro vantaggio è l’uso di un riferimento già noto ad altri sistemi automatici. L’utilizzo di una rappresentazione comune di un dominio permette di condividere anche tra umani la conoscenza di dominio facilitando la comprensione della struttura dei dati e favorisce il riuso di tali conoscenze anche in altre applicazioni e in contesti diversi.

Di seguito sono elencate le varie parti (che verranno riprese e chiarite anche più avanti) che compongono l’ontologia e che ruotano attorno al concetto principale di **UrbanDataset**. Il tutto è accompagnato da grafici che ne mostrano le relazioni (in tali immagini le parti in grassetto sono state definite per l’occasione mentre le altre sono pezzi di ontologie importate che completano e aiutano la sua definizione).

---

<sup>6</sup> la raccomandazione dle W3C che fornisce un insieme di classi, proprietà e restrizioni per rappresentare le informazioni generate da sistemi diversi

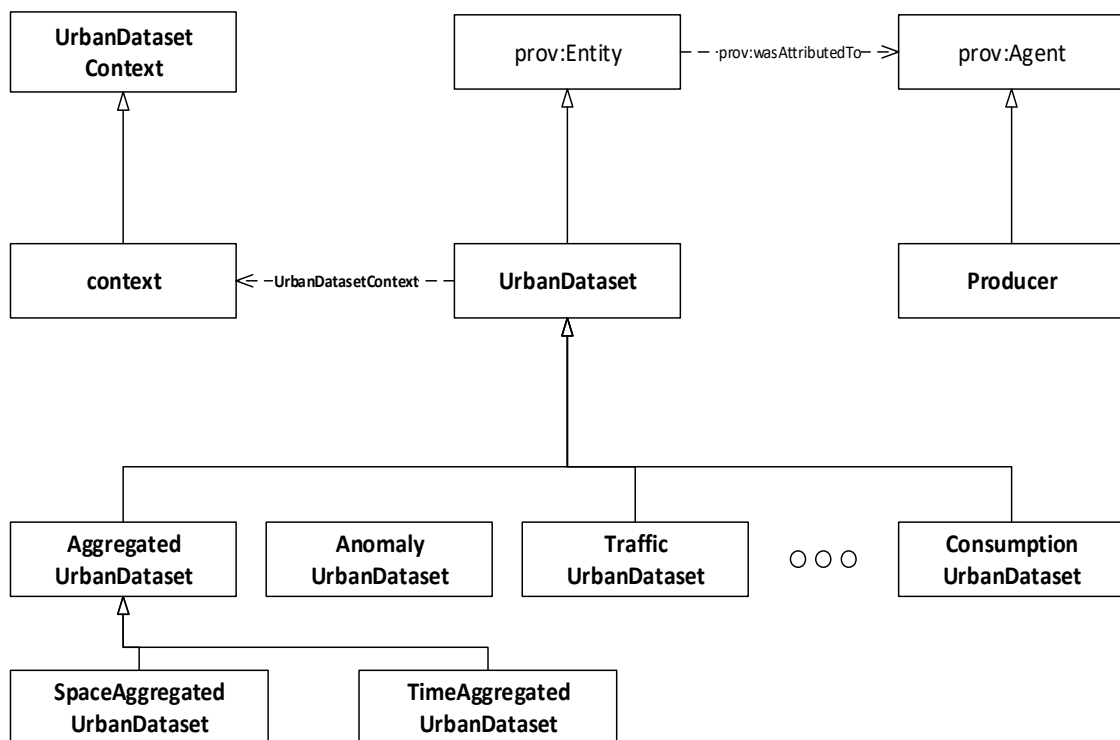


Figura 21 – Schema di alcuni dei principali concetti relativi agli Urban Dataset

- **UrbanDatasetContext:** ogni Urban Dataset ha un suo contesto che serve per descrivere meglio le informazioni e per caratterizzarle, ovvero informazioni come posizione geografica a cui fanno riferimento i dati, lingua usata per le descrizioni, tempo di creazione e fuso orario. Le informazioni di un contesto sono modellate come proprietà.
- **Property:** un UrbanDataset, quando viene generato, viene associato ad altre informazioni (sia specifiche che di contesto) che lo descrivono, come il tempo di creazione, la posizione del sensore da cui è prodotto, la sua descrizione o il suo identificativo e la versione. Tali informazioni sono definite come proprietà dell'UrbanDataset o di un contesto.
- **Producer:** un aspetto importante di un set di dati è la sua provenienza, cioè chi è il fornitore dei dati. PROV Ontology, introdotta in precedenza, in questo caso svolge questa funzione. Nel caso specifico, *Producer*, sottoclasse di "Agent", (Figura 25) descrive l'ente a cui è demandata l'attività di creazione del dato per cui può essere usata, come proprietà per riferire il produttore, *wasAttributedTo* di PROV-O.
- **DataType:** nel caso si faccia riferimento a quantità fisiche misurabile, si è deciso di aggiungere un riferimento all'ontologia OM per collegarla alle unità di misure del sistema internazionale. Per gestire ed elencare i tipi di dati si è invece creata la classe **DataType**.
- **ApplicationContext:** oltre alle specifiche proprietà, all'Urban Dataset è anche associato un campo che indica il contesto applicativo a cui si riferisce (es. SmartBuilding per il caso delle anomalie in un edificio) e un campo per indicare il produttore del dato.

### Meccanismi di aggregazione

Un Urban Dataset può essere catalogato in funzione del suo livello di aggregazione temporale o spaziale. Questo permette sia per chiarire meglio il significato dell'Urban Dataset che per effettuare delle ricerche mirate su determinate categorie.

Tali livelli di aggregazione potranno assumere i seguenti valori

- Livello di aggregazione spaziale:
  - **not applicable**: nel caso il concetto di aggregazione spaziale non abbia senso per lo specifico Urban Dataset;
  - **item**: se i dati si riferiscono al particolare sensore o, più in generale, oggetto, al livello di aggregazione più basso (per esempio, nel caso dell’illuminazione stradale, al singolo palo);
  - **facility**: se i dati si riferiscono all’intera struttura monitorata (per esempio, l’edificio, nel caso di Smart Building, lampioni di una via, per l’illuminazione stradale, ecc.);
  - **facility aggregation**: se si riferisce a un insieme di strutture aggregate fra loro (per esempio un gruppo di Smart Building monitorati da uno stesso Aggregatore);
  - **city**: intera infrastruttura cittadina (per esempio l’intera rete di illuminazione pubblica di una città)
  - **region**: se ci si riferisce a infrastrutture che si estendono oltre la singola città, a un’intera zona/regione geografica (per esempio, impianti di depurazione delle acque)
  - **country**: infrastrutture che si estendono all’intera nazione (per esempio la rete di trasporto dell’energia elettrica)
  
- Livello di aggregazione temporale:
  - **not applicable**: nel caso il concetto di aggregazione temporale non abbia senso per lo specifico Urban Dataset;
  - **static**: Urban Dataset anagrafici, che definiscono le caratteristiche statiche di una struttura (per esempio localizzazione spaziale di un edificio)
  - **instantaneous**: dati istantanei (per esempio temperature istantanea misurata da un sensore)
  - **average**: valori medi (per esempio il consumo medio di elettricità in un determinato intervallo di tempo, in un edificio)
  - **total**: valori totali (per esempio il consumo totale medio di elettricità in un determinato intervallo di tempo, in un edificio)
  - **forecast**: valori previsti e quindi future (per esempio previsioni meteo)

A tale scopo sono state definite le classi **SpaceAggregatedUrbanDataset** e **TimeAggregatedUrbanDataset**: un UrbanDataset con diversa aggregazione è definito come istanza di queste sottoclassi e riferisce, attraverso una proprietà, l’aggregazione usata e definita come istanza della classe **Aggregation** (Figura 22).

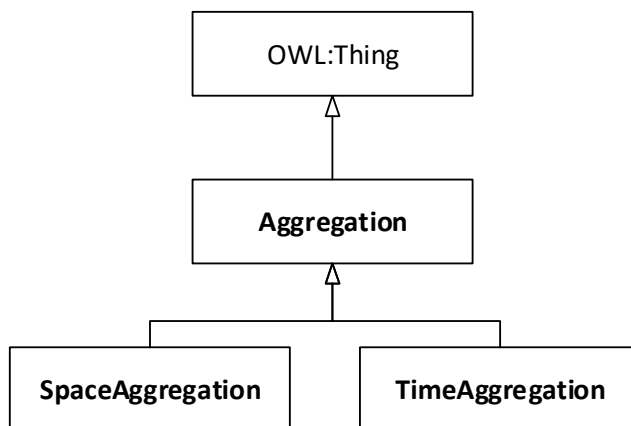


Figura 22 – Schema dei concetti per definire il tipo di aggregazione

## Proprietà degli Urban Data Set

Le singole proprietà dei diversi **UrbanDataset** sono a loro volta delle istanze di sottoclassi di **Property** (Figura 3). Si noti che queste proprietà rispecchiano il contenuto che dovranno avere le implementazioni sintattiche degli Urban Dataset (siano esse in sintassi XML o JSON) per rappresentare i dati che devono essere scambiati e, in tal senso, ne forniscono il riferimento *normativo*.

Property ha due sottoclassi:

- **ContextProperty**: raggruppa tutte le istanze di proprietà usate per descrivere il contesto (es. coordinate, lingua, tempo in cui è stato raccolto e inviato il dato, ecc.);
- **UrbanDatasetProperty**: raggruppa tutte le istanze di proprietà usate per descrivere proprietà specifiche degli Urban Dataset (es. il numero di anomalie, il numero di parcheggi occupati, l'assorbimento energetico medio, ecc.).

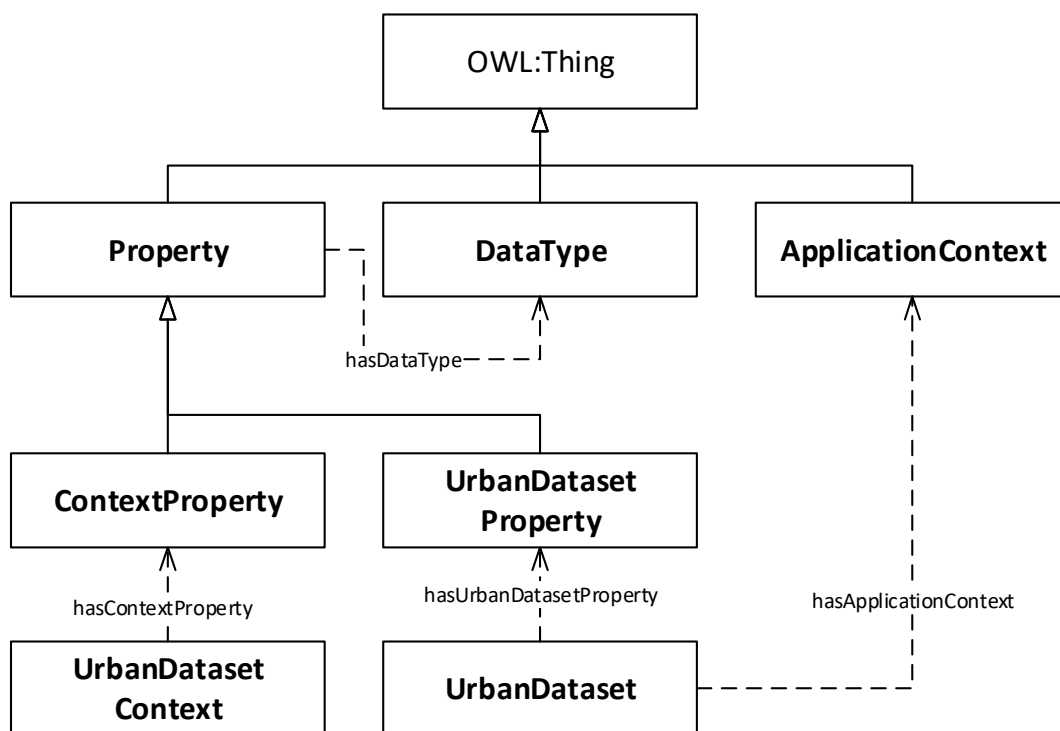


Figura 23 – Schema dei principali concetti relativi agli Urban Dataset

L'ontologia descrive anche proprietà composte, ovvero è possibile che una proprietà sia composta a sua volta da altre proprietà. Ciò è descritto semplicemente creando un'istanza di proprietà che ha uno o più riferimenti ad altre istanze di proprietà tramite le object property *hasContextProperty* e *hasUrbanDatasetProperty*.

A ogni proprietà è stato associato il **DataType**, ovvero il tipo del dato usato per rappresentare l'informazione, ovvero un numero intero, un numero reale, una string o un timestamp, tramite la object property *hasDataType*. Istanze di **DataType** sono per esempio: **double**, **integer**, **string**. Oltre a questo è stata prevista l'associazione con un'unità di misura che qualifichi il dato. A questo scopo è stata usata l'ontologia OM (Ontology of units of Measure) descritta meglio in seguito.

## Contesti applicativi degli Urban Dataset

Un **UrbanDataset** contiene un'aggregazione di dati provenienti dalle piattaforme locali della Smart City (per es. applicazioni per la gestione dell'illuminazione cittadina, per il traffico, ecc.) e perciò tali informazioni sono ascrivibili a un particolare contesto applicativo. Per questo motivo è stata prevista nell'ontologia

l'esistenza di una proprietà che metta in relazione un'istanza di *UrbanDataset* con un contesto applicativo. Per fornire un'idea, in Figura 24 sono definite le istanze (indicate con testo sottolineato) che al momento un *ApplicationContext* può assumere attraverso la relazione *hasApplicationContext*. Inoltre, è specificato dall'ontologia che un *UrbanDataset* può essere associato esclusivamente a un solo *ApplicationContext* esplicitato tramite la dicitura *only*.

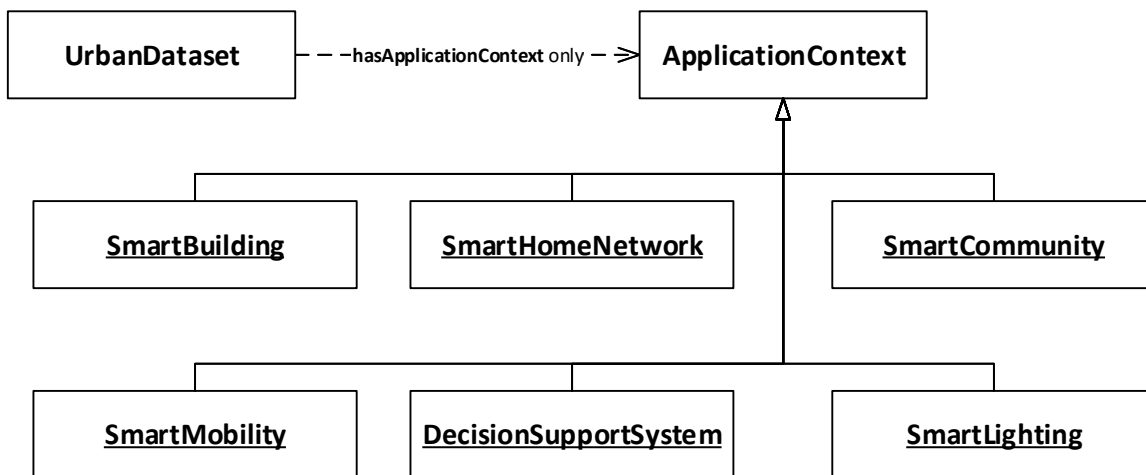


Figura 24 – Associazione tra *UrbanDataset* e *ApplicationContext*

**Uso di ontologie esterne**

L'ontologia definita utilizza altre due ontologie esterne per una maggiore integrazione e compatibilità con l'esterno. La prima è PROV-O (Figura 25) che, come già detto, ha come scopo l'indicazione della provenienza di un dato. In particolare vengono utilizzate le classi:

- **prov:Entity**: una entità è un qualcosa di fisico, digitale, concettuale o altro con alcuni aspetti fissi; le entità possono essere reali o immaginarie;
- **prov:Agent**: un agent è qualcosa che possiede una qualche forma di responsabilità per un'attività, per l'esistenza di un'entità o per l'attività di un altro agent.

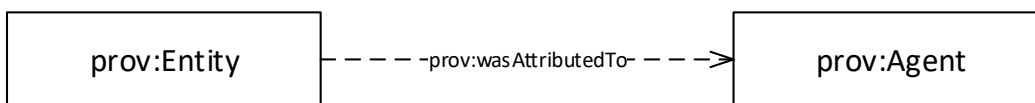


Figura 25 – Uso di PROV-O

Le unità di misura sono rappresentate come istanze in modo che sia possibile associarle alle istanze di *UrbanDataset*. L'ontologia individuata che meglio soddisfa questa condizione è OM, che prevede un sistema di organizzazione delle unità di misura multiple e sottomultiple.

Essa contiene un'estesa lista delle unità di misura, del Sistema Internazionale (SI) e non, definite come istanze della classe **unit**. Ha adottato un sistema che permette di definire multipli e sottomultipli per dieci delle diverse unità di misura fornendo una struttura per aggiungere anche multipli e sottomultipli non ancora presenti ma possibilmente utili in futuro per specifiche unità di misura. Utilizza per fare ciò una sottoclasse della classe **unit** detta **PrefixedUnit** e associa a ogni istanza di questa classe il prefisso usato (es. kilo o milli) e l'unità di misura base del SI. Le istanze dei prefissi sono definite staticamente come istanze della classe *Prefix*.

In Figura 26 è mostrata la struttura completa dell'ontologia OM. Nel nostro caso non viene usata tutta, ma solo la parte relative alle singole unità di misura, ovvero le istanze della classe *Unit*.

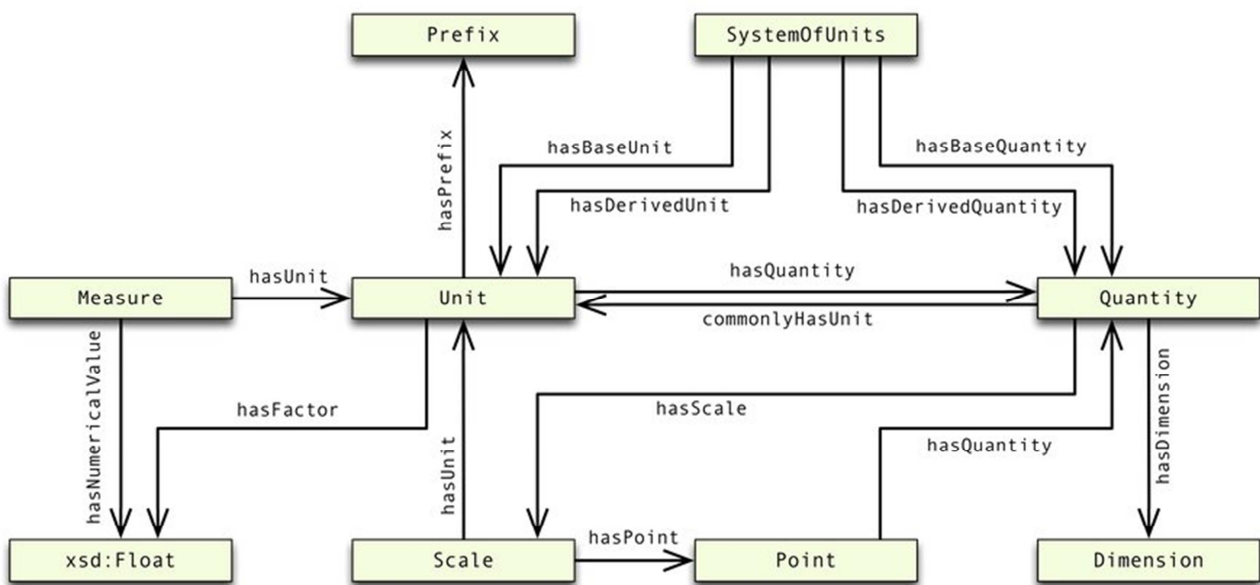


Figura 26 – Struttura delle classi dell’ontologia OM

### Informazioni sui produttori dei dati

I produttori dei dati possono essere diversi, dato che le diverse piattaforme, sorgenti dei dati, possono essere gestite da enti diversi. Diventa necessario tenere traccia di chi è il responsabile dei diversi Urban Dataset ed è quindi utile avere informazioni dettagliate al riguardo. A questo scopo esiste già una ontologia, il cui nome è FOAF [17], definita a partire dal formato vCard con lo scopo gestire e descrivere le informazioni di contatto di persone e organizzazioni. In questo modo, alle istanze della classe Producer possono essere aggiunte informazioni strutturate già comprensibili ai sistemi in grado di analizzare il formato FOAF.

#### 7.1.1 Code List

In generale, un flusso di informazioni è composto da tre tipologie di dati:

- dati fattuali (per es. i valori assunti dalle proprietà di un Urban Dataset)
- testi (per es. la descrizione di un Urban Dataset)
- codici (per es. l'unità di misura delle proprietà degli Urban Dataset).

È buona pratica vincolare a dei vocabolari controllati i valori che possono essere assunti dagli elementi del flusso che ricadono nella terza tipologia. Tali vocabolari, infatti, possono essere pensati come insiemi predefiniti di valori in cui ogni valore rappresenta una semantica concordata tra le parti che lo usano; essi, quindi, favoriscono innanzitutto l'interoperabilità fra le applicazioni che li adottano, inoltre, supportando il multilinguismo, consentono a utenti di comunità diverse di meglio cogliere il significato dei valori che rappresentano (un esempio in Tabella 29).

Codice	Descrizione italiano	Descrizione inglese
MTR	metro	metre
KWH	kilowattora	kilowatt hour
KWT	kilowatt	kilowatt
INH	pollice	inch

Tabella 29 – Esempio - Sottoinsieme della Code List "Unit Of Measure" dell'UNECE

L'istanza di un vocabolario controllato costituisce una **Code List** (o **Lista di Codici**); le Code List sono caratterizzate dalla *versione* che consente di tracciare le variazioni che, nel corso del tempo, può subire l'insieme dei valori che compongono il vocabolario implementato (per es. aggiunta di valori, modifica al significato di valori esistenti,...).

Le Code List possono essere espresse in modo formale attraverso lo standard **OASIS Genericode** (7).

Al fine di favorire l'interoperabilità semantica tra le applicazioni che aderiscono alle specifiche per la Smart City Platform, è stato deciso di definire, a livello di sottospecifica semantica, sia quali degli elementi che compongono un Urban Dataset devono essere vincolati a Code List, sia il vocabolario da utilizzare (quindi le liste di valori e relativa semantica).

Alcune delle Code List definite sono valide per tutti gli Urban Dataset (Tabella 30); altre sono strettamente legate allo specifico Urban Dataset (Tabella 31). Alcune delle Code List indicate nell'ontologia sono state formalizzate anche in formato Genericode (si veda la cartella <http://smartcityplatform.enea.it/specification/semantic/1.0/gc/>).

Code List <sup>7</sup>	Descrizione <sup>8</sup>	Genericode <sup>9</sup>	Elemento <sup>10</sup>	Urban Dataset <sup>11</sup>
DataTypeCode	Tipi di dato ammessi per le proprietà	DataTypeCode.gc	dataType	Tutti
unitOfMeasure	Unità di misura	<i>non disponibile</i>	unitOfMeasure	Tutti
TimezoneCode	Fusi orari	TimezoneCode.gc	timeZone	Tutti
FormatCode	Formati coordinate geografiche	FormatCode.gc	format	Tutti
LanguageCode	Lingue	LanguageCode.gc	language	Tutti

**Tabella 30 – Code List valide per qualsiasi Urban Dataset**

Code List	Descrizione	Genericode	Elemento	Urban Dataset
BuildingAnomalyCause	Cause anomalie in un edificio	AnomalyCauseCode.gc	val con elemento <i>property="CauseID"</i>	1.0- SmartBuildingAnomalies
PlantAnomalyCause	Cause anomalie in un impianto di depurazione	PlantAnomalyCause.gc	val con elemento <i>property="CauseID"</i>	1.0-WaterTreatmentData

**Tabella 31 – Code List valide per specifici Urban Dataset**

<sup>7</sup> Code List: nome che identifica la Code List

<sup>8</sup> Descrizione: descrizione della Code List

<sup>9</sup> Genericode: nome del file genericode che formalizza la Code List (tutti i file sono disponibili all'url: <http://smartcityplatform.enea.it/specification/semantic/1.0/gc/>)

<sup>10</sup> Elemento: elemento del Modello Dati Astratto per il quale è richiesto l'uso di questa Code List

<sup>11</sup> Urban Dataset: nome dell'Urban Dataset che richiede l'uso della Code List indicata



## 7.2 Consultazione dell'ontologia

L'ontologia è stata formalizzata nel linguaggio OWL e resa disponibile online e scaricabile al link:

<http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl>

L'ontologia importa alcune ontologie esterne, che vengono caricate dal web. Nel caso l'importazione non andasse a buon fine, devono essere pre-caricate e rese disponibili.

L'ontologia può essere visualizzata attraverso i normali programmi per l'editing dei file OWL. In particolare può essere visualizzata attraverso programmi desktop come Protégè (<https://protege.stanford.edu/>). In questo caso il programma deve prima essere installato sul computer e poi eseguito per poter caricare e visualizzare il file. Protégè può essere scaricato al link <https://protege.stanford.edu/products.php#desktop-protege>.

La figura seguente riproduce la visualizzazione dell'ontologia tramite Protégè.

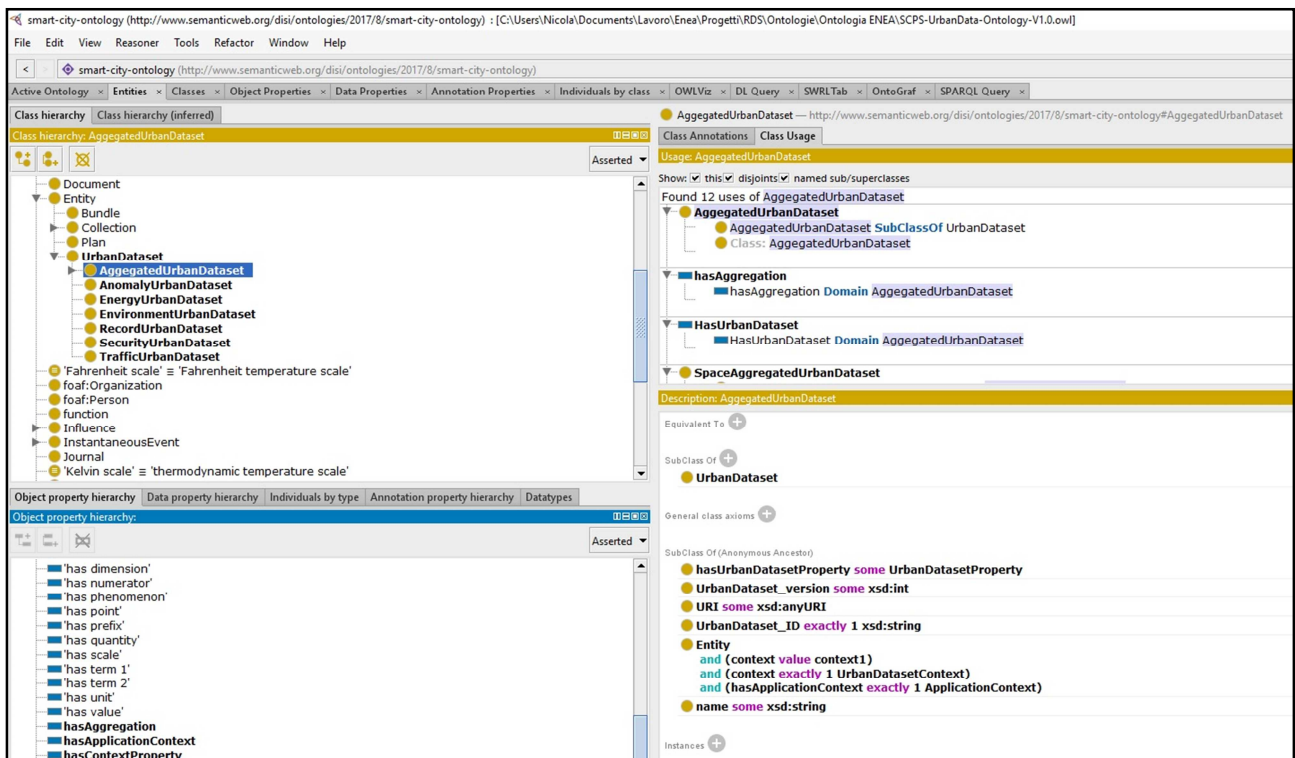
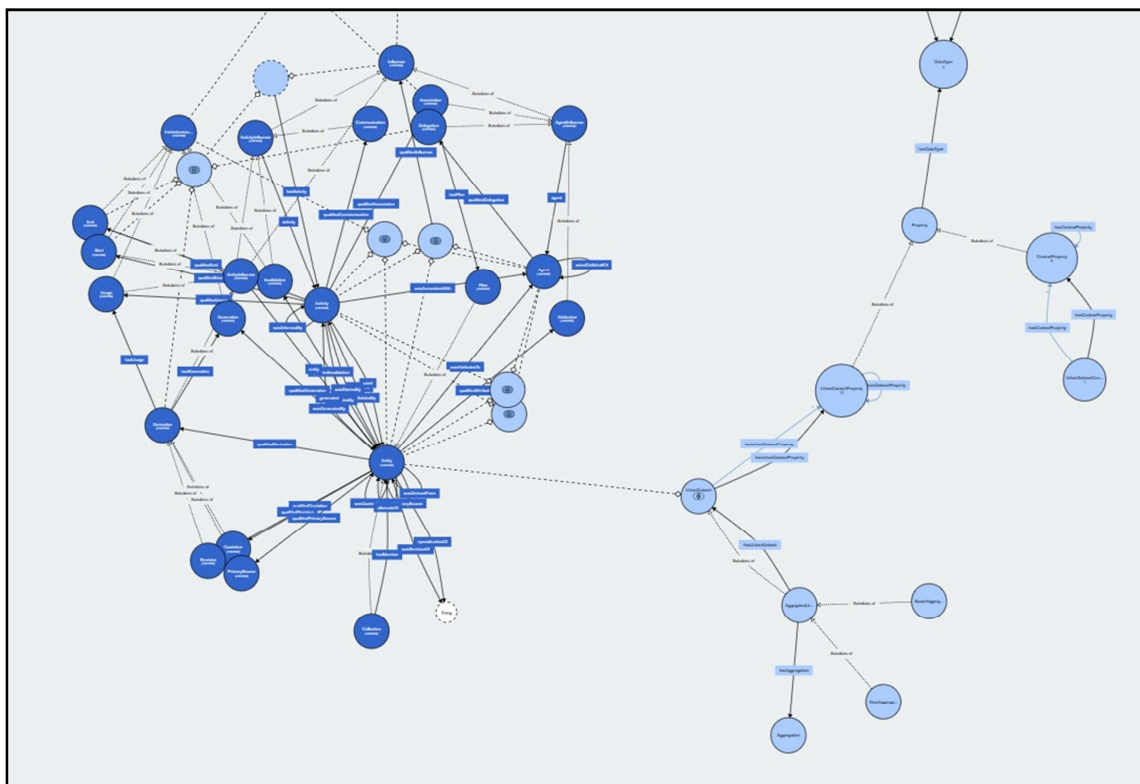


Figura 27 – Visualizzazione dell'ontologia con Protégè

Altro modo per visualizzare l'ontologia è ricorrere a visualizzatori online come WebVOWL, disponibile al link <http://vowl.visualdataweb.org/webvowl.html>. In questo caso l'applicazione online consente di indicare dal menù in basso nella pagina web un file locale o un indirizzo web da cui caricare i dati dell'ontologia per poterla poi visualizzare. La Figura 28 riproduce la visualizzazione dell'ontologia tramite WebVOWL.



**Figura 28 – Visualizzazione dell’ontologia con WebVOWL**

L’ontologia, in questo caso, è visualizzata come grafo nel quale sono rappresentati graficamente concetti e relazioni fra di essi e dal quale è possibile avere rapidamente una visione di insieme. Selezionando e cliccando sui vari oggetti del grafo, l’applicazione fornisce tutte le informazioni disponibili sull’entità selezionata.

### 7.3 Interrogazione dell’ontologia

Per facilitare e migliorare l’adozione e uso dell’ontologia, oltre ad aver ipotizzato delle query di accesso alle informazioni contenute nell’ontologia, si è provveduto a sviluppare una libreria software che permetta di interagire con l’ontologia senza dover conoscere in maniera fine la sua struttura o il linguaggio di interrogazione. In questo capitolo è mostrata l’architettura della libreria realizzata.

Per interrogare un’ontologia sviluppata con le tecnologie del web semantico è necessario conoscere il linguaggio di interrogazione SPARQL. Dal momento che una stessa interrogazione è molto probabile che sia rivolta più di una volta, ha senso avere delle query predeterminate che possono essere richiamate con un comando più semplice preso da una lista definita a priori.

Questa modalità di funzionamento non obbliga chi vuole accedere alle informazioni contenute nell’ontologia a imparare un nuovo linguaggio di interrogazione e per questo risulta più facile integrare queste query all’interno di software più grandi e con uno scopo diverso ma che necessitino delle informazioni presenti nell’ontologia.

Un altro obiettivo è quello di fornire una copertura completa per l’accesso alle informazioni presenti nell’ontologia. A partire dalla struttura dell’ontologia definita, si è partiti con l’ideazione di una serie di richieste che permettessero di accedere a tutte le informazioni presenti senza conoscere la struttura e da punti di partenza diversi (es. conoscendo il soggetto o l’oggetto di una relazione). La libreria, quindi, deve permettere di recuperare le informazioni conoscendo solo in parte ciò che si vuol sapere e per questo risulta molto importante un buon grado di facilità di esplorazione a partire dalle interfacce fornite.

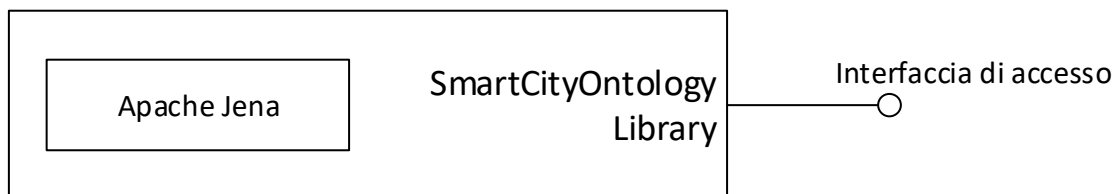
Un requisito necessario riguarda la possibilità di accedere alle informazioni dell'ontologia in remoto per cui è necessario che la libreria sviluppata sia in grado di raggiungere su http il server che contiene l'ontologia.

Un altro vincolo importante riguarda l'utilizzo di Java come linguaggio di programmazione. Si prevede che questa libreria sia sviluppata per essere integrata all'interno di altre applicazioni. Java è uno dei linguaggi più diffusi e conosciuti perciò per garantire una facilità di utilizzo della libreria si è optato per questa scelta.

### **Architettura**

Partendo dalle caratteristiche appena descritte si è arrivati alla definizione dell'architettura. Per poter recuperare le informazioni presenti nell'ontologia, questa deve essere inserita all'interno di un server che accetti richieste di interrogazione ed essendo tecnologie sviluppate dal W3C, le richieste vengono portate al server attraverso protocollo web http. Per affrontare tale vincolo e ottenere già un supporto alle astrazioni dei linguaggi SPARQL e RDF si è deciso di utilizzare una libreria già sviluppata per tali scopi. A questo proposito si è scelto di utilizzare Apache Jena<sup>12</sup>. Questa libreria fornisce una serie di strumenti utili per interagire con un server SPARQL e fornisce un supporto per memorizzare ed elaborare in locale dataset in formato RDF. Il software in questione è sviluppato da diversi anni ed è ancora supportato e in evoluzione. Questo garantisce che il prodotto abbia una buona maturità e affidabilità e una garanzia di evoluzione e supporto futuri sufficiente, oltre che necessaria per l'utilizzo in un sistema complesso e duraturo come quello oggetto di tali specifiche. Un'ulteriore garanzia deriva dalla diffusione che ha Apache Jena, infatti risulta essere una delle più usate in ambiente Java per questo tipo di attività.

La libreria sviluppata, denominata **SmartCityOntology Library**, si interfaccia, quindi, con Apache Jena fornendo un accesso semplificato alle sue operazioni. Tale accesso è costruito in maniera specifica per l'ontologia definita in precedenza. La Figura 29 mostra, attraverso la rappresentazione grafica, che la libreria definita ha come compito quello di nascondere la complessità dell'accesso a un sistema SPARQL e per farlo sfrutta le operazioni messe a disposizione da Apache Jena.



**Figura 29 – Architettura della libreria**

La libreria deve svolgere una funzione piuttosto semplice, ovvero prevedere un set di query pronte da usare per interrogare la base di dati. Un approccio in cui si preparano solamente delle query dentro dei metodi potrebbe essere veloce all'inizio ma difficile da estendere. Per cui si è pensato di organizzare la sua architettura su più livelli in modo da poter riusare le funzionalità dei livelli sottostanti per creare più facilmente interrogazioni e funzionalità più complesse.

---

<sup>12</sup> <https://jena.apache.org/>

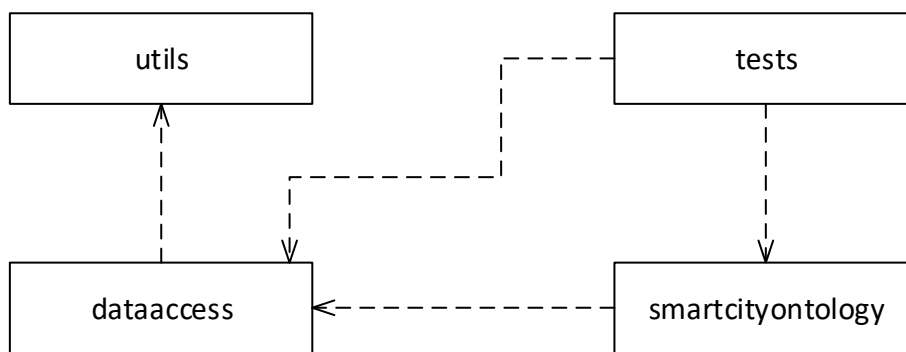


Figura 30 – Architettura della libreria

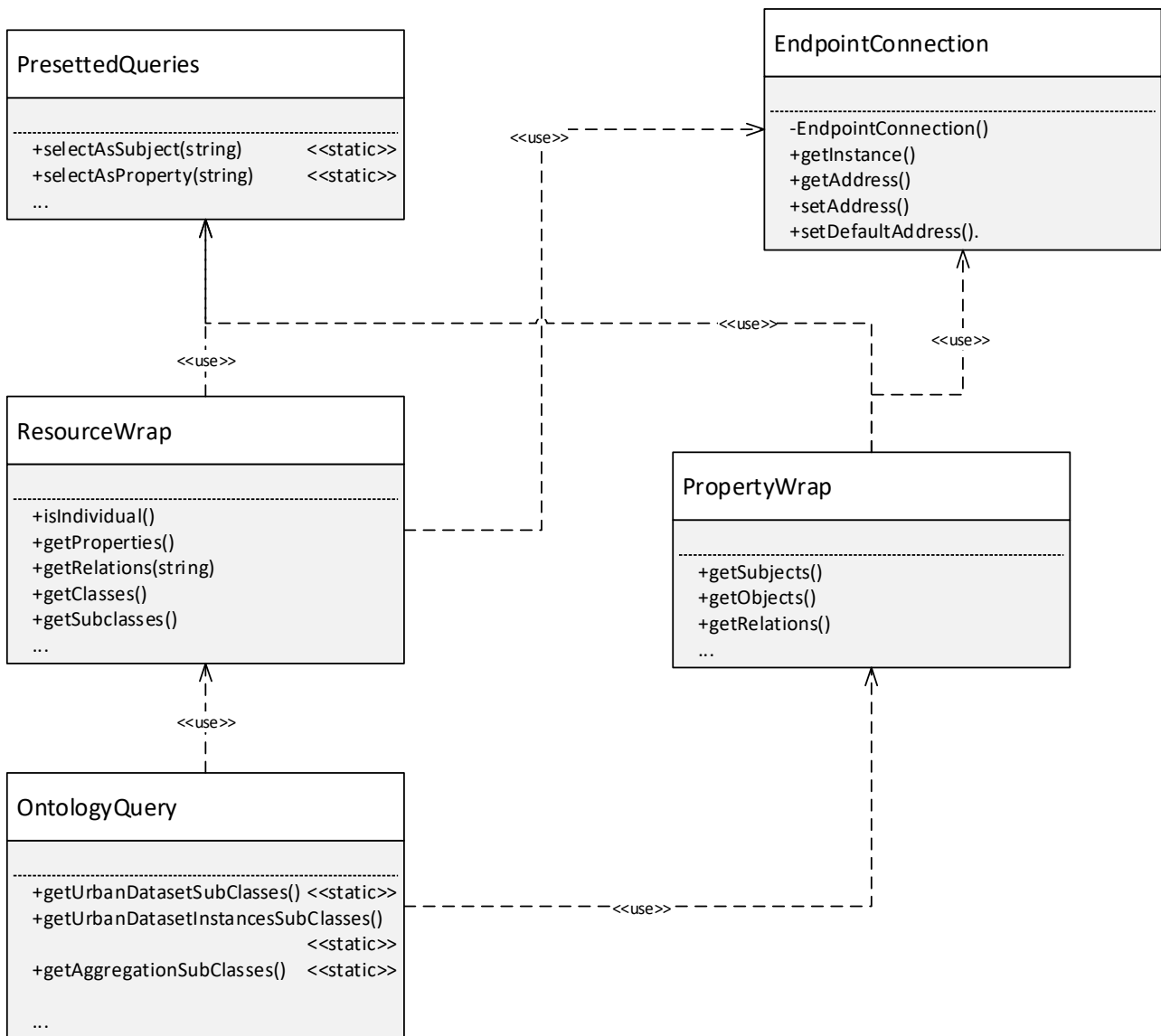
In Figura 30 è mostrato il diagramma dei 3 package della libreria:

- nel package **utils** sono state inserite le informazioni per la configurazione della libreria stessa (classe *Configs*, che legge il file *config.properties*) e sono stati creati degli scheletri di query per costruire dinamicamente l'interrogazione (classe *PresettetQueries*);
- nel package **dataaccess** sono state inserite le classi per effettuare la connessione al sistema remoto (classe *EndpointConnection*) e per costruire interrogazioni che, appoggiandosi sulle funzionalità del package *utils*, recupera le informazioni di una generica ontologia a partire da parametri specifici. A questo scopo sono presenti le due classi *PropertyWrap* e *ResourceWrap*. In particolare, in questo package sono presenti metodi per poter ricostruire la struttura dell'ontologia e per questo potrebbe essere utilizzata anche per altre ontologie diverse dalla nostra. Naturalmente questo è ottenuto nascondendo della complessità ma allo stesso tempo limitando i gradi di libertà;
- il package **smartcityontology**, infine, è quello che contiene le operazioni per recuperare le informazioni specifiche per l'ontologia definita in precedenza. Tali operazioni sono possibili attraverso la classe *OntologyQuery* che consiste di metodi statici che operano sull'ontologia SCPS-UrbanDataset (esempi di questi metodi sono: *getUrbanDatasetInstancesSubClasses*, *getUrbanDatasetPropertyInstances*, ...). Sono stati implementati anche una serie di test automatici per verificare con il comportamento della libreria sia corretto.

Al momento dell'uso della libreria viene creato in automatico la connessione verso il server definito all'interno di un file di configurazione. Naturalmente è possibile modificare il server remoto anche durante l'utilizzo, ma è possibile interrogare solo un server alla volta. Questa scelta si è basata sul fatto che, essendo pensata per una specifica ontologia, si presuppone che venga usata per interrogare l'ontologia presente su un particolare server che fornisce la funzione di indice per i dataset.

Un'altra scelta è stata quella di avere un sistema che costruisce le query utilizzando più livelli per rendere più facile l'estensione nel caso l'ontologia dovesse subire delle modifiche come delle aggiunte di nuovi concetti o una variazione della struttura.

Sono state previste anche delle query che permettono di limitare il numero di risultati recuperati dal server. Si possono creare filtri, attraverso la libreria o da parte dell'utilizzatore, sui dati recuperati dal server in caso non sia necessaria la lista completa. Questo, però, potrebbe causare dei problemi di sovraccarico del server e rallentare il recupero delle informazioni. Infatti, nel caso in cui la query dovesse ottenere come risposta una lista molto lunga di dati, causerebbe un rallentamento nel recupero e nell'invio degli stessi, per cui si è riportato anche nella documentazione la preferenza a richiedere che sia il server stesso a limitare l'output, invece che il software che integra la libreria. In questo modo si può ottenere un carico minore sul server e riuscire a gestire più richieste contemporanee.



**Figura 31 – Diagramma delle classi**

In Figura 31 è mostrato un diagramma parziale delle classi della libreria. Sono mostrati solamente alcuni dei metodi presenti. Di seguito sono elencate e descritte - alcune classi

- *OntologyQuery*: definisce le query principali specifiche per l'ontologia, utilizzabili attraverso dei metodi get che consentono di estrarre particolari dati
- *ResourceWrap* e *PropertyWrap* sono delle classi pensate per fungere da wrapper rispettivamente per le classi *Resource* e *Property* di Apache Jena. Aggiungono metodi per verificare se una risorsa è una istanza o meno, recuperare le sue sottoclassi o le sue istanze e le relazioni in cui sono coinvolte all'interno dell'ontologia. Queste due classi mantengono il riferimento al server remoto nascondendolo all'esterno. Naturalmente il riferimento al server può essere recuperato dall'esterno per modificare l'indirizzo del server durante l'esecuzione.
- La classe *EndpointConnection* gestisce la connessione alla prima esecuzione prendendo l'indirizzo da un file di configurazione che può quindi essere modificato dall'utente prima dell'esecuzione ed evita di cablare nel codice l'indirizzo del server che gestisce le richieste garantendo flessibilità.

Infine è stata creata una documentazione (in formato html) che descrive le diverse classi e i metodi della libreria per sapere le operazioni che si possono effettuare, e per facilitarne l'uso e l'integrazione all'interno di altri software.

## 7.4 Inserimento di nuovi Urban Dataset

### 7.4.1 Il Comitato di gestione dell'ontologia

Al fine di mantenere l'uniformità e la coerenza semantica, fondamentali per l'interoperabilità, anche nel caso di implementazioni autonome e indipendenti della piattaforma, si ritiene importante che l'ontologia sia gestita come risorsa centralizzata.

L'accesso all'ontologia è libero: per cui è possibile scaricarla, modificarla e installarla presso i propri server. In questo caso, però, interoperabilità e uniformità semantica non sono più garantite. Per tale ragione, implementazioni della piattaforma non basate sull'ontologia centrale potrebbero risultare non conformi alla sottospecifica livello semantico.

Manutenzione, gestione e aggiornamento dell'ontologia sono a cura dal **Comitato di Gestione** dell'ontologia della Smart City Platform.

In particolare, tale Comitato ha il compito di:

- formalizzare e seguire la procedura per la richiesta di inserimento di nuovi Urban Dataset
- interfacciarsi con gli utenti che vogliono effettuare tale richiesta
- esaminare le richieste di inserimento di nuovi Urban Dataset
- aggiornare l'ontologia, quando necessario (sia in caso di input esterni, che di decisioni interne)

Il Comitato è costituito da personale del laboratorio DTE-SEN-CROSS di ENEA e del Dipartimento di Informatica - Scienza e Ingegneria (DISI) dell'Università di Bologna.

### 7.4.2 Procedura per la richiesta d'inserimento

Al fine di avviare la procedura per la richiesta di un nuovo Urban Dataset, l'utente deve compilare l'apposito modulo (che viene spiegato in dettaglio nel paragrafo successivo) e inoltrarlo al Comitato nelle inviando una mail all'indirizzo [smartcityplatform.project@enea.it](mailto:smartcityplatform.project@enea.it), con oggetto "Richiesta inserimento nuovo Urban Dataset: [NOME URBAN DATASET]".

L'invio, per ottimizzare la procedura, dovrebbe avvenire in seguito a un'attenta analisi, da parte dell'utente, dell'ontologia esistente al fine di verificare se l'Urban Dataset richiesto non esista già o se siano già definite, in essa, proprietà utilizzabili per costruire il modello del nuovo Urban Dataset<sup>13</sup>.

Si noti che i nomi delle proprietà da inserire nell'ontologia devono essere scritti **in inglese**, per lasciare valenza internazionale alla stessa.

In seguito alla richiesta il Comitato:

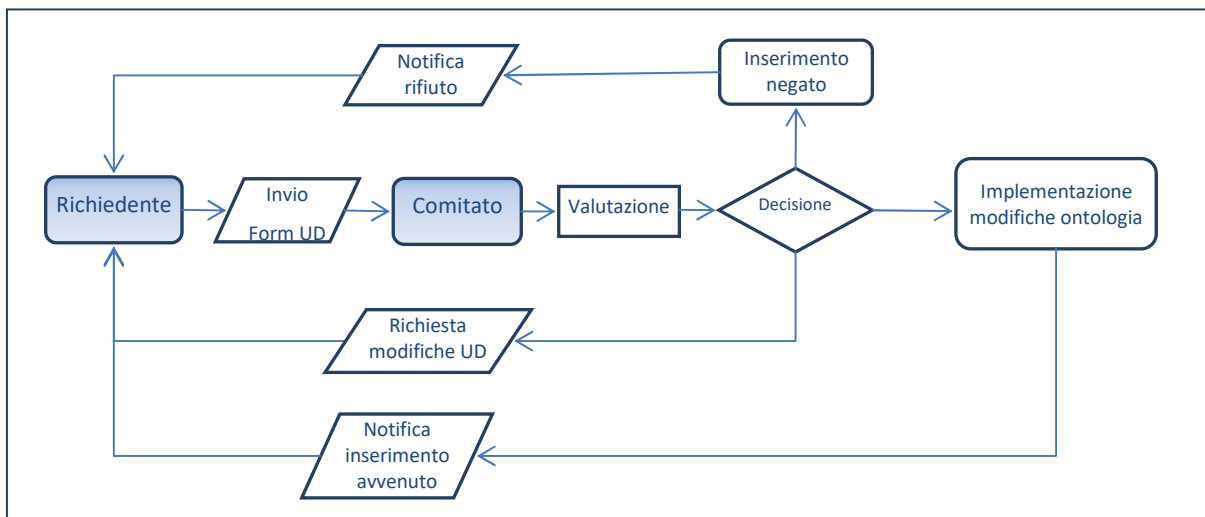
- valuta
  - coerenza dell'Urban Dataset con i fini e con la struttura delle Smart Platform Specification
  - coerenza sintattica delle proprietà richieste
- esamina l'ontologia per verificare che
  - non esista già un Urban Dataset equivalente a quello richiesto
  - alcune fra le proprietà in esso presenti, non siano già presenti nell'ontologia, seppure espresse in modo diverso
  - le eventuali liste di codici richieste, da associare a proprietà dell'Urban Dataset non siano già presenti, seppure parzialmente, nell'ontologia.

---

<sup>13</sup> Per esempio, nel caso che esista già la proprietà "electric consumption", è inutile crearne un'altra chiamata "consumption of electricity" da inserire nel proprio Urban Dataset

- in seguito al precedente esame, decide se:
  - inserire l'Urban Dataset, così come è stato richiesto
  - richiedere modifiche alla struttura dell'Urban Dataset richiesto
  - rifiutare, motivando il rifiuto, l'inserimento dell'Urban Dataset. Motivazioni possono essere: la presenza di un Urban Dataset equivalente nell'ontologia, che può essere utilizzato in luogo di quello richiesto, oppure forti incoerenze con le specifiche, che non siano risolvibili tramite delle modifiche
- aggiorna l'ontologia, effettuando l'inserimento dell'Urban Dataset e producendone una nuova versione
- invia notifica, al richiedente e a tutti gli utenti registrati, della presenza di una nuova release dell'ontologia

La procedura è illustrata nello schema a blocchi della figura successiva:



**Figura 32 – Schema a blocchi della procedura di richiesta di inserimento di un nuovo Urban Dataset**

### 7.4.3 Modulo per la richiesta di inserimento

Per fornire, al Comitato, la proposta di inserimento di un nuovo Urban Dataset, si suggerisce di utilizzare il modulo SchedaPropostaUD Excel disponibile nella cartella:

<http://smartcityplatform.enea.it/specification/semantic/1.0/update/>

Il modulo è costituito da tre parti:

1. **Caso d'uso:** contiene alcune informazioni che permettono ai valutatori di inquadrare meglio il caso d'uso in cui si colloca l'utilizzo dell'Urban Dataset proposto.

In particolare vanno inseriti, nel presente modulo:

- Il nome del caso d'uso
- L'attore che produce l'Urban Dataset (inteso come tipologia di piattaforma: per esempio Smart Building, Smart Lighting, ecc)
- L'attore che riceve l'Urban Dataset (anche in questo caso inteso come tipologia (per esempio ESCO)
- Il nome dell'Urban Dataset

- Il formato che si intende usare (XML, JSON o altro nel caso si utilizzi un formato diverso, quale CSV)
- La Policy da applicare all’Urban Dataset: sarà trattato come Open Data oppure l’accesso è ristretto (privato)?

Il modulo, disponibile all’indirizzo:

- <http://smartcityplatform.enea.it/specification/semantic/1.0/update/OF224-008-v1-SchedaPropostaUD.xlsx>

è mostrato nella seguente figura.

1. INFORMAZIONI SUL CASO D’USO			
Input	Descrizione	Esempio	
Nome del Caso d'uso	Inserire nome del caso d'uso per cui si richiede l'Urban Dataset	Electric consumption	
Attore 1 - Produttore	Inserire la tipologia di piattaforma che produce l'Urban Dataset	Smart Home	
Attore 2 - Ricevente	Inserire a tipologia di piattaforma che riceve l'Urban Dataset	ESCO platform	
Nome Urban Dataset	Inserire il nome dell'UD		
Formato prescelto	Formato o formati prescelti		
Policy dell'UD	Si tratta di Open Data o il suo avverso è limitato?		Open Data / privato

Figura 33 – Prima pagina del foglio di calcolo per la richiesta di un nuovo Urban Dataset: descrizione del caso d’uso

2. **Urban Dataset:** il modulo per l’inserimento degli Urban Dataset è quello più complesso e si divide in tre parti: categorizzazione, contestualizzazione e proprietà dell’Urban Dataset.

Nel primo riquadro (**categorizzazione**) vengono inseriti, oltre al nome dell’Urban Dataset, alcuni dati che permettono di categorizzarlo nell’ontologia. Si tratta di:

- Livello di aggregazione spaziale, che può assumere i seguenti valori:
  - **Not applicable:** se il concetto di aggregazione spaziale non è applicabile all’Urban Dataset che si sta definendo
  - **Item:** se i dati si riferiscono al particolare sensore o, più in generale, oggetto, al livello di aggregazione più basso (per esempio, nel caso dell’illuminazione stradale, al singolo palo)
  - **Facility:** se i dati si riferiscono all’intera struttura monitorata (per esempio, l’edificio, nel caso di Smart Building, lampioni di una via, per l’illuminazione stradale, ecc.)
  - **Facility Aggregation:** se i dati si riferiscono a un insieme di strutture aggregate fra loro (per esempio un gruppo di Smart Building monitorati da uno stesso Aggregatore)
  - **City:** se i dati si riferiscono a un'intera infrastruttura cittadina (per esempio l’intera rete di illuminazione pubblica di una città)
  - **Region:** se i dati si riferiscono a infrastrutture che si estendono oltre la singola città, a un’intera zona/regione geografica (per esempio, impianti di depurazione delle acque)
  - **Country:** i dati si riferiscono a infrastrutture che si estendono all’intera nazione (per esempio la rete di trasporto dell’energia elettrica)
- Livello di aggregazione temporale, che può assumere i seguenti valori:
  - **Not applicable:** se il concetto di aggregazione temporale non è applicabile all’Urban Dataset che si sta definendo
  - **Static:** per Urban Dataset anagrafici, che definiscono le caratteristiche statiche di una struttura (per esempio localizzazione spaziale di un edificio)



- **Instantaneous:** nel caso di dati istantanei (per esempio temperature istantanea misurata da un sensore)
- **Average:** per valori medi (per esempio il consumo medio di elettricità in un determinato intervallo di tempo, in un edificio)
- **Total:** valori totali (per esempio il consumo totale medio di elettricità in un determinato intervallo di tempo, in un edificio)
- **Forecast:** valori previsti e quindi futuri (per esempio previsioni meteo)
- Categoria e sottocategoria delle principali applicazioni coinvolte. Viene utilizzata la categorizzazione definita insieme al NIST, nell’iniziativa “IES City”<sup>14</sup>:

<b>Categorie:</b>	<b>Sotto-categorie</b>
<b>Built environment</b>	Smart Home
	Smart Building
	Land use and management
<b>Water and wastewater</b>	Water collection and management
	Water distribution
	Water consumption
	Wastewater management
<b>Waste</b>	Citizens engagement
	Collection and segregation
	Waste disposal
<b>Energy</b>	Energy supply
	Energy transmission and distribution
	Energy demand
<b>Transportation</b>	Travel demand/consumption
	Traffic management
	Surveillance
<b>Education</b>	Learning outcomes
	Learning and teaching
	Service management
<b>Health</b>	Health care systems
	Health care delivery
	Communication
<b>Socio-economic development</b>	E-Governance
	Social Innovation and Inclusion
	Economy and Business
<b>Public safety, policy &amp; Emergency Response</b>	City surveillance and crime prevention
	Communication

**Tabella 32 – Categorie e sottocategorie di applicazioni, in base al lavoro fatto in IES-City**

<sup>14</sup> <https://pages.nist.gov/smartcitiesarchitecture/>

Il modulo di inserimento dati è mostrato nella figura sottostante:

CATEGORIZZAZIONE DELL'URBAN DATASET	
Valore	Descrizione
<b>Urban Dataset</b>	<i>Inserire il nome</i>
<b>Aggregazione spaziale</b>	<i>Livello di aggregazione spaziale dell'Urban Dataset</i>
<b>Aggregazione temporale</b>	<i>Livello di aggregazione temporale dell'Urban Dataset</i>
<b>Categoria</b>	<i>Categoria principale di applicazioni coinvolte</i>
<b>Sottocategoria</b>	<i>Sottocategoria principale di applicazioni coinvolte</i>

Figura 34 – Modulo di inserimento Urban Dataset: categorizzazione

Nel secondo riquadro (**contestualizzazione**) sono previsti una serie di campi che rappresentano proprietà comuni a tutti gli Urban Dataset (Timestamp di generazione, Timezone a cui si riferiscono tutti i timestamp dell'Urban Dataset, Coordinate del centro geometrico del sistema che produce gli Urban Dataset<sup>15</sup>, identificatore univoco del sistema che ha prodotto i dati). Tali campi non possono essere modificati e sono riportati nella scheda solo per rendere il proponente consapevole della presenza di questi dati.

CONTESTUALIZZAZIONE DELL'URBAN DATASET <i>[Dati obbligatori: sempre presenti]</i>				
	Unità di Misura	Raccomandazioni	Tipo	Descrizione
<b>Timestamp</b>	Adimensionale	Esprimere il timestamp come: gg-mm-aaaa hh:mm:ss	Date-Time	Tempo di generazione dell'Urban Dataset
<b>Timezone</b>	Adimensionale	Esprimere il timestamp come offset, espresso come + oppure meno il numero di ore e minuti di distanza dall'ora UTC: +/-hh:mm (ISO 8601)	String	Fuso orario di riferimento per tutti i timestamp del messaggio
<b>Coordinate</b>	Adimensionale	Esprimere come WGS84 (World Geodetic System)	double	Coordinate del centro geometrico del sistema che produce gli Urban Dataset
<b>Producer</b>	Adimensionale		String	Identificatore univoco del sistema che ha prodotto i dati

Figura 35 – Modulo d’inserimento Urban Dataset: contestualizzazione

L’ultima parte del modulo (**proprietà**) si divide a sua volta in due riquadri.

Il primo riporta una serie di proprietà comuni agli Urban Dataset (timestamp della riga, coordinate del sistema che ha generato lo specifico set di dati riportati nella riga e periodo a cui fanno riferimento i valori). L’unica modifica che può essere effettuata in tale scheda è la scelta di inserire o meno timestamp e coordinate (entrambi opzionali a questo livello).

<sup>15</sup> Per esempio, nel caso di una rete di Smart Building, sarebbe il centro geometrico degli insiemi di edifici

PROPRIETA' DELL'URBAN DATASET <i>[Qui vanno definite le righe dell'UD]</i>							
Proprietà comuni	Unità di Misura	Raccomandazioni	Formati	Descrizione	Esempio	Inserire?	
<b>Timestamp</b>	Date-Time	Esprimere il timestamp come: gg-mm-aaaa hh:mm:ss	String	Tempo di generazione della "riga" (singolo set set di valori) [opzionale]	26-09-2017 00:00:00		
<b>Coordinate</b>	Adimensionale (Coordinate)	Esprimere come WGS84 (World Geodetic System)	String	Coordinate del centro geometrico del sistema che ha prodotto la "riga" (singolo set set di valori) [opzionale]	42.04 ; 12.30		
<b>Start Period</b>	Date-Time	Esprimere il timestamp come: gg-mm-aaaa hh:mm:ss	String	Data di inizio del rilevamento dei dati	25-09-2017 00:01:00	Obbligatoria	
<b>End Period</b>	Date-Time	Esprimere il timestamp come: gg-mm-aaaa hh:mm:ss	String	Data di finr del rilevamento dei dati (se dati istantanei, sarà uguale al valore di start)	26-09-2017 00:00:00	Obbligatoria	

Figura 36 – Prima parte del modulo relativo alle proprietà: proprietà comuni

Il secondo permette l’inserimento delle proprietà specifiche che caratterizzano l’Urban Dataset. I campi da riempire sono:

- Nome della proprietà
- Unità di misura
- Raccomandazioni: qui devono essere inserite raccomandazioni specifiche, relative alla proprietà, quali il riferimento a uno specifico standard per rappresentarli (per es. il WGS84 per le coordinate) o l’uso di liste di codici (nel caso di formato enumeration). Il contenuto di tali liste sarà specificato nei fogli successivi
- Formato dati: scelto (fra quelli standard, quali String, Int, Enumeration ecc.)
- Descrizione
- Esempio
- Annidata: il modello dei dati prevede la possibilità di definire proprietà annidate. Se la proprietà è annidata, qui andrà inserito il nome delle proprietà che si annidano qui dentro (tali nomi devono corrispondere ai nomi altre proprietà definite nella tabella.
- Esiste già nell’ontologia?: è un valore booleano che dice che quella particolarità è già stata definita (in un altro Urban Dataset precedente) e quindi può essere riutilizzata

Il numero di righe predisposte per l’inserimento di nuove proprietà, è limitato a 23, ma può essere aumentato (togliendo la protezione al foglio, che non richiede password e copiando e inserendo altre righe di proprietà), se risulta necessario.

Proprietà specifich	Unità di Misura	Raccomandazioni	Formati	Descrizione	Esempio	Annidata	Esiste già nell'ontologia?
Nome proprietà 1							
Nome proprietà 2							
Nome proprietà 3							

Figura 37 – Seconda parte del modulo relativo alle proprietà: proprietà specifiche

### 3. Liste di codici (code-list)

L’ultima pagina del foglio di calcolo permette di inserire una lista di codici, nel caso sia stata introdotta nella colonna raccomandazioni della pagina Urban Dataset, in corrispondenza di un tipo Enumeration.

All’inizio della pagina si inserisce nome della lista di codici e l’eventuale nome alla colonna codici (se necessario). Quindi è possibile elencare i codici ed esprimerne il significato in italiano e inglese.

Nel caso sia necessario definire più di una lista di codici, basterà copiare la pagina e ripeterla il numero di volte necessaria per esprimere tutte

<b>3 CODE LIST</b>		
<i>Nota: Nel caso sia necessaria più di una lista di codici, replicare questo foglio</i>		
Nome della Code-List:	inserire nome	
Nome generale della colonna Codice	inserire nome	Es. <i>Causeld</i>
Codice	Descrizione Italiana	Descrizione inglese

## 8 APPENDICE E – SCPS Information Level

La sottospecifica del livello Information definisce il **formato** che i sistemi che aderiscono all'Architettura di Riferimento per la Smart City Platform devono adottare per scambiare **Urban Dataset**, ovvero informazioni significative su aspetti rilevanti della Smart City.

Il formato è definito da:

- un **Modello Dati Astratto** che, in modo indipendente dalla sintassi, rappresenta e definisce il contenuto che deve avere un documento utilizzato per scambiare Urban Dataset
- l'**implementazione sintattica** del Modello. Le implementazioni sintattiche possono essere molteplici. La specifica fornisce due sintassi di riferimento: XML (eXtensible Markup Language ) e JSON (JavaScript Object Notation).

Il Modello Dati Astratto determina la semantica del formato (concetto diverso dalla Semantica degli Urban Dataset che è gestita dalla "Sottospecifica Livello Semantico"). Le implementazioni sintattiche definiscono le sintassi supportate dalla Smart City Platform.

Insieme, Modello Dati Astratto e implementazioni sintattiche di riferimento, garantiscono l'interoperabilità con la piattaforma a *Livello Information*.

Questa sottospecifica è organizzata nelle seguenti parti:

- descrizione del Modello Dati Astratto definito per rappresentare i documenti;
- artefatti per l'implementazione del Modello Dati Astratto nelle due sintassi (XML e JSON);
- procedura per la creazione delle istanze degli Urban Dataset;
- descrizione ed esempio di utilizzo di alcuni elementi del Modello Dati Astratto;
- un esempio di un Urban Dataset "Whatever";
- risorse a supporto dell'implementazione della specifica SCPS Information;
- sintesi e possibili ulteriori sviluppi della ricerca

### 8.1 Il Modello Dati Astratto

Il Modello Dati Astratto ha l'obiettivo di definire, in modo indipendente dalla sintassi, il contenuto e il significato dei documenti utilizzati per scambiare set di dati rilevati sulle proprietà che compongono un Urban Dataset.

Il suo obiettivo è facilitare la mappatura tra le implementazioni sintattiche utilizzate dalle applicazioni esistenti e quelle supportate dalla Smart City Platform.

È stato progettato sulla base di una serie di requisiti e caratteristiche individuati analizzando sia le esigenze che questa sottospecifica deve soddisfare, sia la struttura degli Urban Dataset.

Il Modello Dati Astratto è descritto per mezzo di tabelle e si compone delle seguenti sezioni:

- **Specification:** contiene le informazioni che descrivono l'Urban Dataset utilizzato (per esempio il riferimento alla specifica a cui aderisce e le proprietà che lo compongono)
- **Context:** fornisce le informazioni che contestualizzano i valori trasmessi (per esempio il fuso orario dei timestamp)
- **Values:** contiene i dati rilevati sulle proprietà che compongono l'Urban Dataset, raggruppati in righe.

#### Come leggere le tabelle che rappresentano il Modello Dati Astratto

Le tabelle che rappresentano il Modello Dati Astratto sono composte dalle seguenti colonne:

- **Elemento:** etichetta scelta per un'informazione contenuta nel modello.
- **Descrizione:** descrizione testuale dell'informazione.
- **Occorrenze:** numero di ripetizioni ammesse (cardinalità) per l'informazione. È espresso come coppia di valori: il primo valore indica il numero minimo di ripetizioni (se valore minimo >0 l'informazione è obbligatoria), il secondo valore indica il numero massimo di ripetizioni ammesse.
- **Tipo:** tipo di dato ammesso per l'informazione (p.es. intero, stringa,...).
- **Esempio:** valore di esempio per l'informazione.

Il colore delle righe delle tabelle indica il livello di aggregazione delle informazioni:

- **riga con sfondo bianco:** indica una informazione elementare (es.: "data", "temperatura", ...) o, se il nome dell'elemento è preceduto dal simbolo @, una informazione che qualifica un'altra informazione elementare. Per queste due tipologie di informazione sono ammessi solo tipi primitivi (stringa, intero, booleano,...);
- **riga con sfondo grigio:** indica una informazione aggregata, composta da più informazioni elementari (p.es.: "periodo", "source", ...). Le informazioni aggregate sono sempre di tipo aggregato.

Si consideri, per esempio, il seguente frammento estratto da una riga del blocco *Context* del modello:

Elemento	Descrizione	Occorr.	Tipo
<b>coordinates</b>	Coordinate WGS84 del centro geometrico della rete applicativa su cui sono stati rilevati i valori trasmessi.	0..1	aggregato
@ <i>format</i>	Formato WGS84 in cui sono espresse le coordinate. Opzioni possibili: – Gradi, Minuti, Secondi (WGS84-DMS) – Gradi Decimali (WGS84-DD)	0..1	string <i>codelist</i>
<b>longitude</b>	Longitudine.	1..1	double
<b>latitude</b>	Latitudine.	1..1	double
<b>altitude</b>	Altitudine.	0..1	double

la sua interpretazione è la seguente:

*Il blocco "Context" può contenere un elemento "coordinates", di tipo aggregato, che contiene i dati che identificano il centro geometrico della rete applicativa su cui sono stati rilevati i valori trasmessi; questo è opzionale ("coordinates" ha cardinalità minima=0) e non può essercene più di uno (massima cardinalità=1). I dati che occorre indicare sono:*

- *opzionalmente, il formato WGS84 in cui sono espresse le coordinate; se questo elemento è presente, il suo valore deve essere una stringa e appartenere ad una lista di codici predefinita*
- *longitudine e latitudine (obbligatori poiché "longitude" e "latitude" hanno cardinalità minima=1) di tipo double*
- *opzionalmente, l'altitudine di tipo double.*

Il valore di alcune delle informazioni presenti nel modello dati astratto sarà vincolato dall'ontologia (vedere *Paragrafo 0*).

L'uso degli identificatori, invece, è descritto nella specifiche "SCPS\_Core\_e\_Terminology" all'interno del livello Functional.

### 8.1.1 Modello Dati Astratto - sezione "Specification"

La sezione **Specification** è composta dalle informazioni indicate nella seguente tabella:

Elemento	Descrizione	Occorr.	Tipo	Esempio
<b>specificationReferences</b>	Insieme di riferimenti che consentono l'identificazione della specifica dell'Urban Dataset utilizzato.	1..1	aggregato	
<b>@version</b>	Versione della specifica.	0..1	string	01
<b>id</b>	Codice identificativo della specifica.	1..1	string	UD0000000000
<b>@schemeID</b>	Identificativo dello schema di identificazione secondo il quale è stato definito l'identificatore.	0..1	string	SCPS
<b>name</b>	Nome associato all' Urban Dataset.	1..1	string	Smart Building Average Consumption
<b>uri</b>	URI che identifica la specifica (può essere un urn indicate il dominio della specifica, o un url).	1..1	string	http://ontologia.esempio.it/UD0000000000
<b>propertyDefinition</b>	Questo elemento consente di inserire, all'interno del messaggio stesso, la descrizione di una delle proprietà che compongono l' Urban Dataset.  <i>Per la struttura del blocco, si veda i paragrafo propertyDefinition.</i>  <i>L'elemento deve essere ripetuto per ciascuna proprietà che compone l'Urban Dataset.</i>	0..n	aggregato	

#### L'elemento aggregato **propertyDefinition**

Il Modello Dati prevede la possibilità di definire sia **proprietà elementari** (p.es.: *magnitudo di un evento sismico, latitudine, longitudine, consumo elettrico di un edificio, consumo termico di un edificio*), sia **proprietà aggregate**, ovvero proprietà composte da più proprietà elementari (p.es.: *epicentro di un evento sismico, che è composto da latitudine e longitudine, o consumo totale di un edificio, che è composto da consumo elettrico e consumo termico*).

La definizione di una **proprietà elementare** deve essere data dalle seguenti informazioni:

<b>propertyDefinition</b>	Struttura dell'elemento in caso di <b>proprietà elementare</b> .			
---------------------------	--	--	--	--

<b>propertyName</b>	Nome della proprietà.	1..1	string	<i>AverageConsumption</i>
<b>propertyDescription</b>	Descrizione della proprietà.	0..1	string	<i>Consumo elettrico medio di un edificio, calcolato su uno specifico periodo.</i>
<b>dataType</b>	Tipo di dato con cui viene espresso il valore della proprietà.	1..1	string	<i>double</i>
<b>unitOfMeasure</b>	Unità di misura in cui è espresso il valore.	0..1	string	<i>kWh</i>

La definizione di una **proprietà aggregata** deve fornire le seguenti informazioni:

<b>propertyDefinition</b>	Struttura dell'elemento in caso di <b>proprietà aggregata</b> .	0..n	aggregato	
<b>propertyName</b>	Nome della proprietà.	1..1	string	<i>TotAverageConsumption</i>
<b>propertyDescription</b>	Descrizione della proprietà.	0..1	string	<i>Consumo energetico (elettrico e termico) elettrico medio di un edificio, calcolato su uno specifico periodo.</i>
<b>subPropertyRef</b>	Riferimento (nome) di una proprietà che compone la proprietà indicata nell'elemento <i>propertyName</i> . <i>Anche la proprietà riferita deve essere definita, seguendo la struttura propertyDefinition.</i>	2..n	string	<i>AverageConsumption</i>
				<i>ThermalConsumption</i>

### 8.1.2 Modello Dati Astratto - sezione "Context"

La sezione **Context** è composta dalle informazioni indicate nella seguente tabella:

Elemento	Descrizione	Occorr.	Tipo	Esempio
<b>producer</b>	Dati del sistema (Piattaforma/Soluzione) che ha prodotto i dati rilevati.	1..1	aggregato	
<b>id</b>	Identificatore univoco del sistema.	1..1	string	<i>Smart Home Test</i>
<b>@schemeID</b>	Identificativo dello schema di identificazione secondo il quale è stato definito l'identificatore.	0..1	string	<i>SCPS</i>
<b>timeZone</b>	Fuso orario di qualsiasi marca temporale presente nel documento.	1..1	string	<i>UTC</i>
<b>timestamp</b>	Istante di generazione di questo Urban Dataset (ovvero il momento in cui i dati sono stati aggregati).	1..1	dateTime	<i>2017-01-30T10:52:50</i>
<b>coordinates</b>	Coordinate WGS84 del centro	1..1	aggregato	



	geometrico della rete applicativa su cui sono stati rilevati i valori trasmessi.			
<i>@format</i>	Formato WGS84 in cui sono espresse le coordinate. Opzioni possibili: - Gradi, Minuti, Secondi (WGS84-DMS) - Gradi Decimali (WGS84-DD)	0..1	string	<i>WGS84-DD</i>
<b>longitude</b>	Longitudine.	1..1	double	<i>49.5215</i>
<b>latitude</b>	Latitudine.	1..1	double	<i>43</i>
<b>altitude</b>	Altitudine.	0..1	double	<i>3.0987</i>
<b>language</b>	Lingua in cui sono espressi i campi testo dell'istanza.	0..1	string	<i>IT</i>

### 8.1.3 Modello Dati Astratto - sezione "Values "

La sezione **Values** è composta di **una o più righe (1 riga = 1 gruppo di proprietà)**, ognuna contenente le informazioni indicate nella seguente tabella:

Elemento	Descrizione	Occorr.	Tipo	Esempio
<b>id</b>	Numero riga.	0..1	integer	<i>1</i>
<b>description</b>	Descrizione testuale dei valori rilevati.	0..1	string	<i>Consumo elettrico giornaliero dell'unità abitativa.</i>
<b>timestamp</b>	Data e ora (marca temporale) di generazione dei dati riportati in questa riga. <i>Se coincide con il timestamp di generazione dell'Urban Dataset, può essere omissso.</i>	0..1	dateTime	<i>2017-01-30T10:52:50</i>
<b>coordinates</b>	Coordinate WGS84 dell'oggetto a cui sono riferiti i dati rilevati indicati in questa riga. <i>Se questo elemento è presente, specifica ulteriormente la posizione del dato rispetto alle coordinates della sezione Context.</i>	0..1	aggregato	
<i>@format</i>	Formato WGS84 in cui sono espresse le coordinate. Opzioni possibili: - Gradi, Minuti, Secondi (WGS84-DMS) - Gradi Decimali (WGS84-DD)	0..1	string	<i>WGS84-DD</i>
<b>longitude</b>	Longitudine.	1..1	double	<i>49.52</i>
<b>latitude</b>	Latitudine.	1..1	double	<i>43</i>

	<b>altitude</b>	Altitudine.	0..1	double	3.09
	<b>period</b>	Periodo durante il quale sono stati rilevati i dati. <i>Nel caso di grandezza istantanea, start_timestamp e end_timestamp coincidono.</i>	0..1	aggregato	
	<b>start_timestamp</b>	Marca temporale indicante l'inizio del periodo.	1..1	dateTime	2014-01-01T10:52:50
	<b>end_timestamp</b>	Marca temporale indicante la fine del periodo.	1..1	dateTime	2014-02-28T10:52:50
	<b>property</b>	Proprietà che compone l'Urban Dataset. <i>Per la struttura dell'elemento, si veda la sezione property. L'elemento deve essere ripetuto per ciascuna proprietà che compone l'Urban Dataset.</i>	1..n	aggregato	

L'elemento aggregato **property**

Il modello dati stabilisce come definire sia i valori delle **proprietà elementari**, sia i valori delle **proprietà aggregate**.

Una **proprietà elementare** deve essere espressa conformemente alla seguente struttura:

	<b>property</b>	Struttura del blocco in caso di <b>proprietà elementare</b> .			
	<b>name</b>	Nome identificativo della grandezza rilevata.	1..1	string	<i>Electric Consumption</i>
	<b>value</b>	Valore rilevato.	1..1	string	123

Una **proprietà aggregata** deve essere espressa conformemente alla seguente struttura:

	<b>property</b>	Struttura del blocco in caso di <b>proprietà aggregata</b> .			
	<b>name</b>	Nome identificativo della proprietà rilevata.	1..1	string	<i>Flat Consumption</i>
	<b>property</b>	Proprietà elementare che compone la proprietà aggregata.	2..n		

8.1.4 **Modello Dati Astratto e Ontologia**

Il Modello Dati Astratto è stato definito in modo tale da essere generico e, per questo motivo, non impone vincoli né fornisce indicazioni sulle proprietà da specificare in fase di creazione delle istanze degli Urban Dataset.

Questo livello di interoperabilità riguarda la **semantica degli Urban Dataset** ed è gestito attraverso l'utilizzo di un'ontologia (si veda la "Sottospecifica Livello Semantico").

A livello semantico sono gestite anche le liste di codici (*Code List*) ammesse per gli elementi per i quali il Modello Dati Astratto le richiede, ovvero:

- timeZone
- language
- unitOfMeasure
- format
- dataType

## 8.2 Sintassi di riferimento e validazione

La sottospecifica relativa al livello Information fornisce la definizione formale dell'implementazione del Modello Dati Astratto nelle due sintassi supportate dalla Smart City Platform: XML (9) e JSON (10).

Il formalismo utilizzato per la sintassi XML è **XML Schema 1.0** (11); XML Schema (XSD) è un linguaggio di descrizione del contenuto di un file XML; definisce:

- gli elementi e gli attributi che possono apparire in un documento XML
- la struttura del documento (quindi la sequenza e il numero di occorrenze di elementi e attributi)
- i tipi di dato di elementi e attributi
- eventuali valori di default e limiti sui valori ammessi per elementi e attributi.

In Figura 38 è riportato uno screenshot che mostra un sottoinsieme dell'XML Schema che implementa il Modello Dati Astratto.

Il formalismo utilizzato per la sintassi JSON è **JSON Schema**<sup>16</sup> (12); JSON Schema è un linguaggio di descrizione del contenuto di un file JSON; definisce:

- gli elementi che possono apparire in un documento JSON
- la struttura del documento (quindi la sequenza e il numero di occorrenze di elementi)
- i tipi di dato di elementi
- eventuali limiti sui valori ammessi per gli elementi.

In Figura 39 è riportato uno screenshot che mostra un sottoinsieme del JSON Schema che implementa il Modello Dati Astratto.

Gli Schema, XML e JSON, costituiscono il riferimento per la costruzione dei documenti utilizzati per scambiare Urban Dataset; consentono, inoltre, di **validare la sintassi e la struttura** dei documenti scambiati.

A tal fine possono essere utilizzati sia per realizzare propri strumenti di validazione, sia per alimentare tool di validazione realizzati da terze parti (per esempio validatori disponibili su web o editor XML).

Gli Schema sono disponibili al link: <http://smartcityplatform.enea.it/specification/information/1.0/>

Invece, per la **validazione della semantica degli Urban Dataset** (quindi, per esempio, per verificare che le proprietà specificate siano effettivamente quelle che compongono l'Urban Dataset indicato) è possibile creare degli artefatti di validazione (per esempio XML Schema customizzati, Schematron (13), ...), generici o specifici per tipo di Urban Dataset, utilizzando le informazioni contenute nell'ontologia.

---

<sup>16</sup> JSON Schema è una specifica draft; l'Urban Dataset JSON Schema definito in questa sottospecifica basa sulla versione *draft 06*.

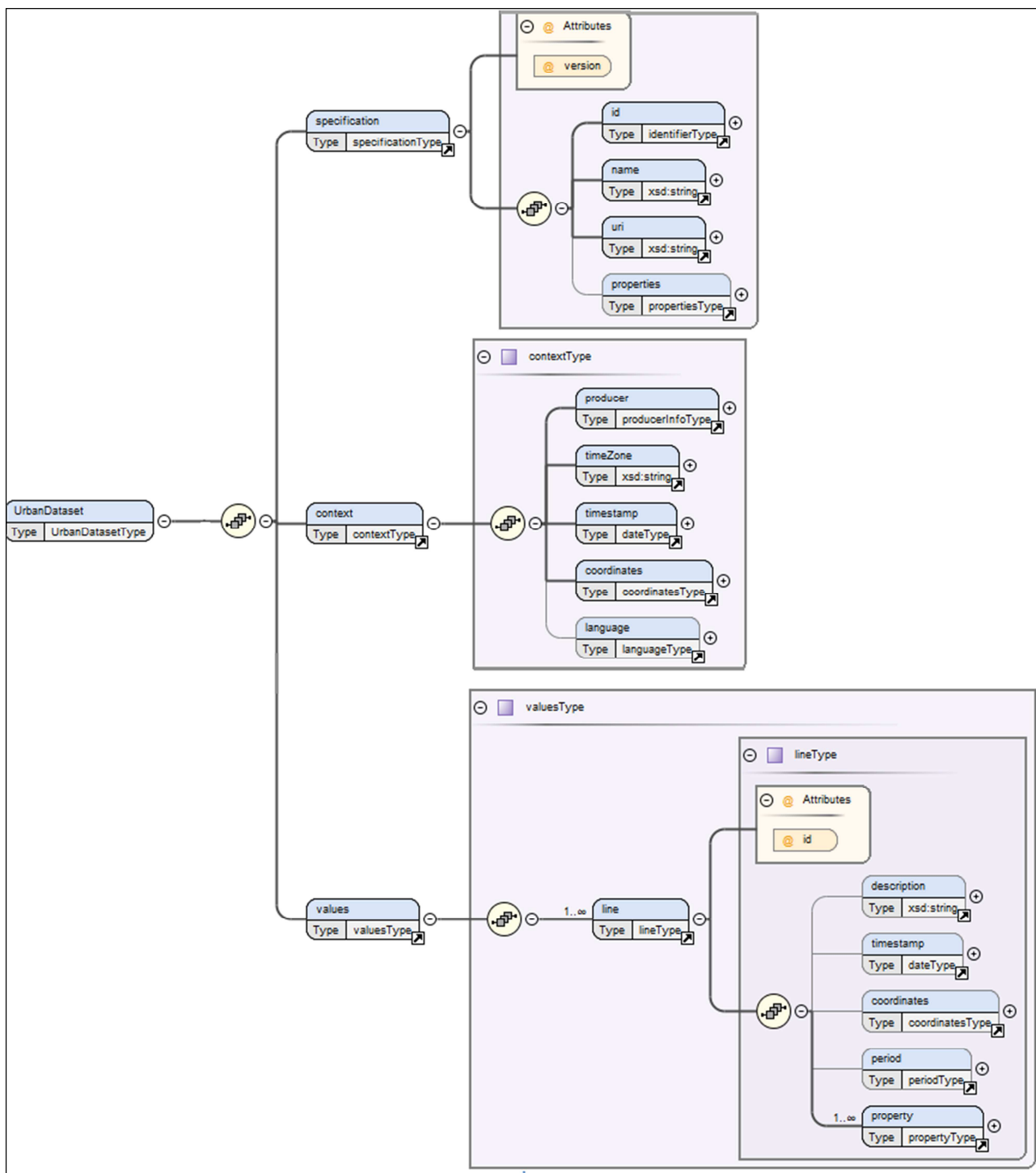


Figura 38 – Screenshot Urban Dataset XML Schema

```

    ▼ UrbanDataset {3}
      type : object
      ▼ properties {4}
        ▼ specification {4}
          type : object
          ▼ properties {5}
            ▶ version {1}
            ▶ id {3}
            ▶ name {1}
            ▶ uri {1}
            ▶ properties {4}
            ▶ required [3]
            ▶ additionalProperties {1}
          ▼ context {4}
            type : object
            ▼ properties {5}
              ▶ producer {4}
              ▶ timeZone {1}
              ▶ timestamp {1}
              ▶ coordinates {4}
              ▶ language {1}
            ▶ required [4]
            ▶ additionalProperties {1}
          ▼ values {4}
            type : object
            ▼ properties {1}
              ▼ line {3}
                type : array
                minItems : 1
                ▼ items {3}
                  type : object
                  ▼ properties {6}
                    ▶ id {1}
                    ▶ description {1}
                    ▶ timestamp {1}
                    ▶ coordinates {4}
                    ▶ period {4}
                    ▶ property {3}
                  ▶ required [1]
                ▶ required [1]
                ▶ additionalProperties {1}
              ▶ additionalProperties {1}
            ▶ required [3]
  
```

Figura 39 – Screenshot Urban Dataset JSON Schema

### 8.3 Descrizione ed esempio di utilizzo di alcuni elementi del Modello Dati Astratto

#### 8.3.1 specificationReferences

L'elemento *specificationReferences* è composto da un insieme di elementi che consentono di indicare la specifica a cui aderisce l'Urban Dataset.

L'elemento *specificationReferences* e gli elementi che lo compongono sono stati resi obbligatori poiché, al fine di favorire l'interoperabilità, si richiede che l'Urban Dataset abbia sempre una specifica di riferimento; a tale scopo si raccomanda l'uso dell'Ontologia ma, poiché l'Architettura di Riferimento può essere adottata a livelli, sono ammesse anche altre modalità (p.es. un documento di testo in formato pdf), che però rendono l'implementazione non conforme all'intera SCPS.

È invece opzionale l'indicazione della versione della specifica adottata (attributo *version*); se non indicata, si assume che sia la più recente versione disponibile.

È opzionale anche l'indicazione dello schema di identificazione (attributo *schemeID*) a cui appartiene il codice identificativo della specifica. Se la specifica dell'Urban Dataset è definita nell'Ontologia, se ne raccomanda l'indicazione e il valore da utilizzare è "SCP" (che identifica le specifiche registrate nella Smart City Platform).

I casi di valorizzazione del blocco *specificationReferences* sono sostanzialmente due:

- caso 1: Urban Dataset registrato nell'Ontologia, in cui rientra anche il caso particolare dell'Urban Dataset Whatever
- caso 2: Urban Dataset non registrato nell'Ontologia.

Di seguito alcuni esempi di valorizzazione del blocco *specificationReferences* nelle sintassi XML e JSON.

Il primo esempio, Figura 40 e Figura 41, mostra il caso di un Urban Dataset registrato nell'Ontologia; tutti gli elementi, anche quelli opzionali, sono valorizzati.

```
<specification version="1.0">
  <id schemeID="SCPS"> 1.0- SmartBuildingAnomalies</id>
  <name>Smart Building Anomalies</name>
  <uri> http://smartcityplatform.enea.it/...</uri>
</specification>
```

Figura 40 – Esempio di implementazione XML del blocco *specificationReferences*

```
"specification": {
  "version": "1.0",
  "id": {
    "value": " 1.0-SmartBuildingAnomalies",
    "schemeID": "SCSP"
  },
  "name": "Smart Building Anomalies",
  "uri": " http://smartcityplatform.enea.it/..."
}
```

Figura 41 – Esempio di implementazione JSON del blocco *specificationReferences*

Il secondo esempio, Figura 42 e Figura 43, mostra il caso di un Urban Dataset **non** registrato nell'Ontologia la cui specifica potrebbe essere un file pdf contenente un testo come quello illustrato Figura 44; gli elementi opzionali non sono stati valorizzati.

```
<specification >
<id>MyUD010</id>
<name>Parking monitoring</name>
<uri>http://myDomain.it/ MyUD010.pdf</uri>
</specification>
```

```
"specification": {
  "id": {
    "value": "MyUD010"
  },
  "name": "Parking monitoring ",
  "uri": "http://myDomain.it/ MyUD010.pdf "
}
```

Figura 42 – Esempio di implementazione XML del blocco `specificationReferences`

Figura 43 – Esempio di implementazione JSON del blocco `specificationReferences`

### Specifica dell'Urban Dataset "Parking monitoring"

L'Urban Dataset "Parking monitoring" ha l'obiettivo di monitorare le auto presenti in un parcheggio.

*Identificatore:* MyUD010

*Nome:* Parking monitoring

*Riferimento di questa specifica:* <http://myDomain.it/ MyUD010.pdf>

*Proprietà che compongono l'Urban Dataset*

- ParkID: identificatore parcheggio, tipo stringa
- CarN: numero auto presenti nel parcheggio in un certo istante, tipo intero

Figura 44 – Esempio specifica di un Urban Dataset non registrato

### 8.3.2 `propertyDefinition`

L'elemento `propertyDefinition` consente di dichiarare e descrivere nell'istanza dell'Urban Dataset le proprietà che lo compongono, rendendo così il messaggio autodescrittivo.

È opzionale: se ne raccomanda l'utilizzo nei casi di specifica dell'Urban Dataset non formalizzata nell'ontologia, o di Urban Dataset "Whatever"; negli altri casi il suo utilizzo è a discrezione degli implementatori, ricordando che le proprietà dichiarate attraverso le diverse occorrenze dell'elemento `propertyDefinition` devono comunque essere coerenti con la specifica dell'Urban Dataset.

In Figura 47e Figura 48, per esempio, le implementazioni XML e JSON del blocco `propertyDefinition` di un Urban Dataset non presente nell'Ontologia; è stato ripreso l'esempio illustrato nella precedente Figura 44.

Per un esempio di definizione di proprietà aggregata, si veda il *Capitolo 8.4*.

```

<specification >
  <id>MyUD010</id>
  ...
  <properties>
    <propertyDefinition>
      <propertyName>ParkID </propertyName>
      <propertyDescription>Identificatore parcheggio.</propertyDescription>
      <dataType>string</dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>CarN</propertyName>
      <propertyDescription>Numero auto presenti nel parcheggio in un certo istante.</propertyDescription>
      <dataType>integer</dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
  </properties>
</specification>

```

**Figura 45 – Esempio di implementazione XML del blocco *propertyDefinition***

```

"specification": {
  "id": {
    "value": "MyUD010"
  },
  ...,
  "properties": {"propertyDefinition": [
    {
      "propertyName": "ParkID ",
      "propertyDescription": "Identificatore parcheggio.",
      "dataType": "string",
      "unitOfMeasure": "adimensionale"
    },
    {
      "propertyName": "CarN ",
      "propertyDescription": "Numero auto presenti nel parcheggio in un certo istante.",
      "dataType": "integer",
      "unitOfMeasure": "adimensionale"
    }
  ]}
}

```

**Figura 46 – Esempio di implementazione JSON del blocco *propertyDefinition***

In Figura 47 e Figura 48, per esempio, le implementazioni XML e JSON del blocco *propertyDefinition* per l'Urban Dataset "Smart Building Anomalies" definito nell'Ontologia.



```

<specification version="1.0">
  <id schemeID="SCPS"> 1.0-SmartBuildingAnomalies</id>
  ...
  <properties>
    <propertyDefinition>
      <propertyName>BuildingID</propertyName>
      <propertyDescription>Identificatore del palazzo monitorato e generatore di anomalie.</propertyDescription>
      <dataType>string</dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>BuildingName</propertyName>
      <propertyDescription>Etichetta associata al palazzo monitorato e generatore di anomalie.
      </propertyDescription>
      <dataType>string</dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>CauseID</propertyName>
      <propertyDescription>Causa a cui si riferiscono le anomalie individuate dalla piattaforma smart building,
      espressa come codice.</propertyDescription>
      <dataType>integer</dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>CauseDescription</propertyName>
      <propertyDescription>Causa a cui si riferiscono le anomalie individuate dalla piattaforma smart building,
      espressa come testo.</propertyDescription>
      <dataType>integer </dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>HighAnomalies</propertyName>
      <propertyDescription>Numero delle anomalie individuate con priorità high.</propertyDescription>
      <dataType> integer </dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>MeanAnomalies</propertyName>
      <propertyDescription>Numero delle anomalie individuate con priorità mean.</propertyDescription>
      <dataType>integer </dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>LowAnomalies</propertyName>
      <propertyDescription>Numero delle anomalie individuate con priorità low.</propertyDescription>
      <dataType>integer </dataType>
      <unitOfMeasure>adimensionale</unitOfMeasure>
    </propertyDefinition>
  </properties>
</specification>

```

Figura 47 – Esempio di implementazione XML dell'elemento *propertyDefinition*

```

"specification": {
  "version": "1.0",
  "id": {
    "value": "1.0-SmartBuildingAnomalies ",
    "schemeID": "SCPS"
  },
  ...
  "properties": {"propertyDefinition": [
    {
      "propertyName": "BuildingID",
      "propertyDescription": "Identificatore del palazzo monitorato e generatore di anomalie.",
      "dataType": "string",
      "unitOfMeasure": "adimensionale"
    },
    {
      "propertyName": "BuildingName",
      "propertyDescription": "Etichetta associata al palazzo monitorato e generatore di anomalie.",
      "dataType": "string",
      "unitOfMeasure": "adimensionale"
    },
    {
      "propertyName": "CauseId",
      "propertyDescription": "Causa a cui si riferiscono le anomalie individuate dalla piattaforma smart building,
espressa come codice.",
      "dataType": "integer",
      "unitOfMeasure": "adimensionale"
    },
    {
      "propertyName": "CauseDescription",
      "propertyDescription": "Causa a cui si riferiscono le anomalie individuate dalla piattaforma smart building,
espressa come testo.",
      "dataType": "integer",
      "unitOfMeasure": "adimensionale"
    },
    {
      "propertyName": "HighAnomalies",
      "propertyDescription": "Numero delle anomalie individuate con priorità high.",
      "dataType": "integer",
      "unitOfMeasure": "adimensionale"
    },
    {
      "propertyName": "MeanAnomalies",
      "propertyDescription": "Numero delle anomalie individuate con priorità mean.",
      "dataType": "integer",
      "unitOfMeasure": "adimensionale"
    },
    {
      "propertyName": "LowAnomalies",
      "propertyDescription": "Numero delle anomalie individuate con priorità low.",
      "dataType": "integer",
      "unitOfMeasure": "adimensionale"
    }
  ]}
}

```

Figura 48 – Esempio di implementazione JSON dell'elemento "specification"

### 8.3.3 line

Un'istanza di Urban Dataset deve avere almeno un elemento *line*, ovvero almeno un set di valori rilevati sulle proprietà che compongono l'Urban Dataset.

Ai valori deve essere associato l'istante di generazione; se esso coincide con l'istante di creazione dell'Urban Dataset, può essere omissso a livello di *line*, altrimenti deve essere indicato (elemento *timestamp* in *line*). Se non indicato a livello di *line*, si assume che il timestamp della riga sia uguale al timestamp dell'Urban Dataset (*timestamp* nel blocco *context*).

Opzionalmente, nell'elemento *line* è inoltre possibile indicare:

- il numero della riga (elemento *id*);
- una descrizione testuale del set di valori riportati nella riga (elemento *description*);
- le coordinate WGS84 dell'oggetto a cui sono riferiti i valori indicati nella riga (elemento *coordinates* in *line*); se non indicate a livello di *line*, si assume di poter associare all'oggetto le coordinate WGS84 del centro geometrico della rete applicativa su cui sono stati rilevati i valori trasmessi (*coordinates* nel blocco *context*);
- il periodo durante il quale sono stati rilevati i valori. Il periodo è espresso come coppia di marche temporali, una di inizio (elemento *start\_timestamp*) e una di fine periodo (elemento *end\_timestamp*) che devono essere indicate entrambe, anche in caso di misurazione istantanea (in questo caso coincidono). Se l'istante di misurazione coincide con l'istante di generazione dei valori (*timestamp* in *line*), il periodo può essere omissso.

È invece obbligatorio indicare almeno una coppia proprietà-valore (elemento *property*) poiché si assume che un Urban Dataset abbia almeno una proprietà (elementare o aggregata).

Per un esempio di riga con proprietà aggregata, si veda il *Capitolo 8.4*.

```

<values>
  <line id="1">
    <description>Anomalie negli ultimi tre anni.</description>
    <timestamp>2017-08-02T11:22:15</timestamp>
    <period>
      <start_ts>2014-08-02T00:00:00</start_ts>
      <end_ts>2017-08-02T00:00:00</end_ts>
    </period>
    <property name="BuildingID">
      <val>1</val>
    </property>
    <property name="BuildingName">
      <val>F40</val>
    </property>
    <property name="CauseID">
      <val>1</val>
    </property>
    <property name="CauseDescription">
      <val>Luci accese in assenza di persone</val>
    </property>
    <property name="HighAnomalies">
      <val>301</val>
    </property>
    <property name="MeanAnomalies">
      <val>115</val>
    </property>
    <property name="LowAnomalies">
      <val>280</val>
    </property>
  </line>

  <line id="2">
    <description>Anomalie negli ultimi tre anni.</description>
    <timestamp>2017-08-02T11:22:15</timestamp>
    <period>
      <start_ts>2014-08-02T00:00:00</start_ts>
      <end_ts>2017-08-02T00:00:00</end_ts>
    </period>
    <property name="BuildingID">
      <val>4</val>
    </property>
    <property name="BuildingName">
      <val>F66</val>
    </property>
    <property name="CauseID">
      <val>1</val>
    </property>
    <property name="CauseDescription">
      <val>Luci accese in assenza di persone</val>
    </property>
    ...
  </line>
</values>

```

Figura 49 – Esempio di implementazione XML dell'elemento "line"

```

{"line": [
  {
    "id": 1,
    "description": "Anomalie negli ultimi tre anni.",
    "timestamp": "2017-08-02T11:22:15",
    "period": {
      "start_ts": "2014-08-02T00:00:00",
      "end_ts": "2017-08-02T00:00:00"
    },
    "property": [
      { "name": "BuildingID",
        "val": "1" },
      { "name": "BuildingName",
        "val": "F40" },
      { "name": "CauseId",
        "val": "1" },
      { "name": "CauseDescription",
        "val": "Luci accese in assenza di persone" },
      { "name": "HighAnomalies",
        "val": "301" },
      { "name": "MeanAnomalies",
        "val": "115" },
      { "name": "LowAnomalies",
        "val": "280" }
    ]
  },
  {
    "id": 2,
    "description": "Anomalie negli ultimi tre anni.",
    "timestamp": "2017-08-02T11:22:15",
    "period": {
      "start_ts": "2014-08-02T00:00:00",
      "end_ts": "2017-08-02T00:00:00"
    },
    "property": [
      { "name": "BuildingID",
        "val": "4" },
      { "name": "BuildingName",
        "val": "F66" },
      { "name": "CauseId",
        "val": "1" },
      { "name": "CauseDescription",
        "val": "Luci accese in assenza di persone" },
      { "name": "HighAnomalies",
        "val": "55" },
      { "name": "MeanAnomalies",
        "val": "54" },
      { "name": "LowAnomalies",
        "val": "88" }
    ]
  }
]}

```

Figura 50 – Esempio di implementazione JSON dell'elemento "line"

### 8.4 Creare le istanze degli Urban Dataset

Il percorso che gli implementatori dovrebbero svolgere per realizzare un'applicazione capace di abilitare lo scambio di Urban Dataset tra un Contesto Applicativo e la Smart City Platform dovrebbe prevedere i seguenti step:

#### 1. Identificazione o definizione della specifica dell'Urban Dataset che si vuole implementare

Il primo step consiste nell'analizzare gli Urban Dataset esistenti (nell'ontologia e nei casi studio resi disponibili) al fine di individuare un Urban Dataset capace di supportare lo scambio di informazioni che si vuole attivare. Se nessun Urban Dataset risulta adeguato, decidere se utilizzare la specifica dell'Urban Dataset "Whatever" o attivare la procedura per la richiesta di inserimento di un nuovo Urban Dataset nell'ontologia (si veda la "Sottospecifica Livello Semantico"); in questo secondo caso, la specifica da utilizzare sarà quella rilasciata al termine della procedura (la definizione dell'Urban Dataset "Whatever" può essere anche il primo passo per iniziare a testare l'uso del formato oppure per scambiare i dati nell'attesa che l'Urban Dataset richiesto venga approvato. In generale non si dovrebbe considerare l'uso del "Whatever" come la soluzione definitiva).

#### 2. Analisi del Modello Dati Astratto e della specifica individuata

Il contenuto che devono avere i messaggi utilizzati per scambiare Urban Dataset è definito dal Modello Dati Astratto; quindi, prima di passare all'implementazione vera e propria, occorre analizzarlo e confrontarlo con la specifica dell'Urban Dataset selezionato, al fine di stabilire quali proprietà specificare e quali dei suoi elementi opzionali utilizzare (in generale, l'uso degli elementi opzionali nei blocchi "Specification" e "Context" è a discrezione degli implementatori; l'uso degli elementi opzionali nel blocco "Values" è guidato dalla specifica dell'Urban Dataset).

Come descritto in precedenza, infatti, il Modello Dati Astratto è stato concepito per supportare qualsiasi tipo di Urban Dataset. Per questo motivo alcuni suoi elementi sono stati definiti come opzionali in modo da poter essere utilizzati o meno a seconda dei casi e, per la stessa ragione, non impone vincoli né fornisce indicazioni sulle proprietà da specificare in fase di creazione delle istanze degli Urban Dataset.

Entrambe questi aspetti, però, sono definiti sia negli Urban Dataset disponibili nei casi studio, sia nell'ontologia che, in questo step, devono essere analizzate e confrontate con il Modello.

Si consideri, per esempio, la specifica dell'Urban Dataset "Consumo elettrico medio dell'edificio" (Tabella 33):

Urban Dataset	Smart Building Average Consumption		
Scopo	Registrare il consumo energetico giornaliero di uno o più edifici, al fine di comunicarlo, mensilmente, al WebGis e permetterne la visualizzazione sulla mappa.		
Aggregazione spaziale	Facility		
Categoria / Sottocategoria	Built environment / Smart Building		
Aggregazione temporale	Valore totale		
<b>SPECIFICAZIONE DELL'URBAN DATASET</b>			
UrbanDatasetId	1.0-SmartBuildingAverageConsumption		
UrbanDatasetUri	<a href="http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAverageConsumption">http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#SmartBuildingAverageConsumption</a>		
UrbanDatasetName	Smart Building Average Consumption		
<b>CONTESTUALIZZAZIONE DELL'URBAN DATASET</b>			
Timestamp	Unità di Misura:	Formato	Descrizione / Esempio
	Adimensionale	Date-Time	Tempo di generazione dell'Urban

		gg-mm-aaaaThh:mm:ss	Dataset
<b>Timezone</b>	Adimensionale	String	Es. UTC, UTC+01, UTC-05
<b>Coordinates</b>	Adimensionale	WGS84 (World Geodetic System)	Coordinate del centro geometrico della rete di Smart Building es. "42.04" ; "12.30"
<b>Producer</b>	Adimensionale	String	Identificatore del Sistema che ha prodotto i dati
<b>PROPRIETÀ DELL'URBAN DATASET</b>			
(si ripetono per ogni giorno del mese e per ogni edificio della rete)			
	<b>Unità di Misura:</b>	<b>Formato</b>	<b>Descrizione / Esempio</b>
<b>Start Period</b>	Adimensionale	Date-Time gg-mm-aaaaT hh:mm:ss	Data di inizio della misurazione del consumo energetico (es. 26-09-2017T00:00:00)
<b>End Period</b>	Adimensionale	Date-Time gg-mm-aaaaT hh:mm:ss	Data di fine della misurazione del consumo energetico (es. 26-09-2017T23:59:59)
<b>BuildingID</b>	Adimensionale	String	identificatore del palazzo a cui si riferisce la misura
<b>BuildingName</b>	Adimensionale	String	Es. "F40"
<b>AverageConsumption</b>	kWh	Double	Consumo energia elettrica dell'edificio. Es. 555,3 kWh

**Tabella 33 - Definizione Urban Dataset "Consumo elettrico medio dell'edificio"**

La specifica indica che l'Urban Dataset:

- non ha proprietà aggregate
- non richiede l'indicazione delle coordinate a livello di riga
- non richiede l'indicazione del timestamp a livello di riga
- richiede l'indicazione del periodo a livello di riga
- è composto dalle proprietà *BuildingID*, *BuildingName* e *AverageConsumption* di cui definisce anche tipo ed eventuale unità di misura.

Per implementare questo Urban Dataset, quindi, nel blocco *Values* è sufficiente supportare il sottoinsieme del Modello Astratto illustrato in Tabella 34; l'utilizzo degli elementi opzionali nei blocchi *Specification* e *Context* è invece a discrezione degli implementatori.

Elemento	Occorr.	Tipo	Valori ammessi (nel range ammesso dal Tipo indicato)
<b>id</b>	0..1	integer	<i>qualsiasi</i>
<b>description</b>	0..1	string	<i>Qualsiasi descrizione coerente con lo scopo dell'Urban Dataset</i>
<b>timestamp</b>	0..1	dateTime	
<b>coordinates</b>	0..1	aggregato	
@format	0..1	string	
<b>longitude</b>	1..1	double	
<b>latitude</b>	1..1	double	
<b>altitude</b>	0..1	double	
<b>period</b>	10..1	aggregato	
<b>start_timestamp</b>	1..1	dateTime	<i>qualsiasi</i>

<b>end_timestamp</b>	1..1	dateTime	<i>qualsiasi</i>
<b>property</b>	1..#3	aggregato	
<b>name</b>	1..1	string	<i>BuildingID, BuildinName, AverageConsumption</i>
<b>value</b>	1..1	string	<i>qualsiasi</i>

Tabella 34 – Blocco "Values" dell'Urban Dataset "Consumo elettrico medio dell'edificio"

### 3. Mapping tra Modello Dati e sistema

Questo step ha l'obiettivo di definire come saranno popolati i messaggi creati a partire dal sottoinsieme del Modello Dati Astratto individuato dallo step precedente; a tal fine occorre identificare l'eventuale sorgente dati (per es. un database) che fornirà i dati per generare gli Urban Dataset e mappare ogni elemento del Modello in un elemento/campo della sorgente.

### 4. Implementazione

L'ultimo passaggio riguarda la scelta della sintassi per i messaggi da scambiare, l'analisi del relativo schema (che è l'unico riferimento normativo per la sintassi e la struttura dei messaggi) e la predisposizione del sistema per la generazione dei messaggi nel formato scelto.

Strumenti/documentazione di supporto per questo step:

- Guida all'implementazione del Modello Dati Astratto (che definisce la corrispondenza tra gli elementi dell'intero Modello gli elementi degli schemi XML e JSON)
- sottoinsieme del Modello Dati individuato allo step 2 e mappatura definita allo step 3 che, insieme, indicano quali elementi dello schema utilizzare e come valorizzarli.

In Figura 51 è illustrato un esempio di mappatura tra sorgente dati e sottoinsieme del Modello Dati e tra quest'ultimo e sintassi scelta per l'implementazione.



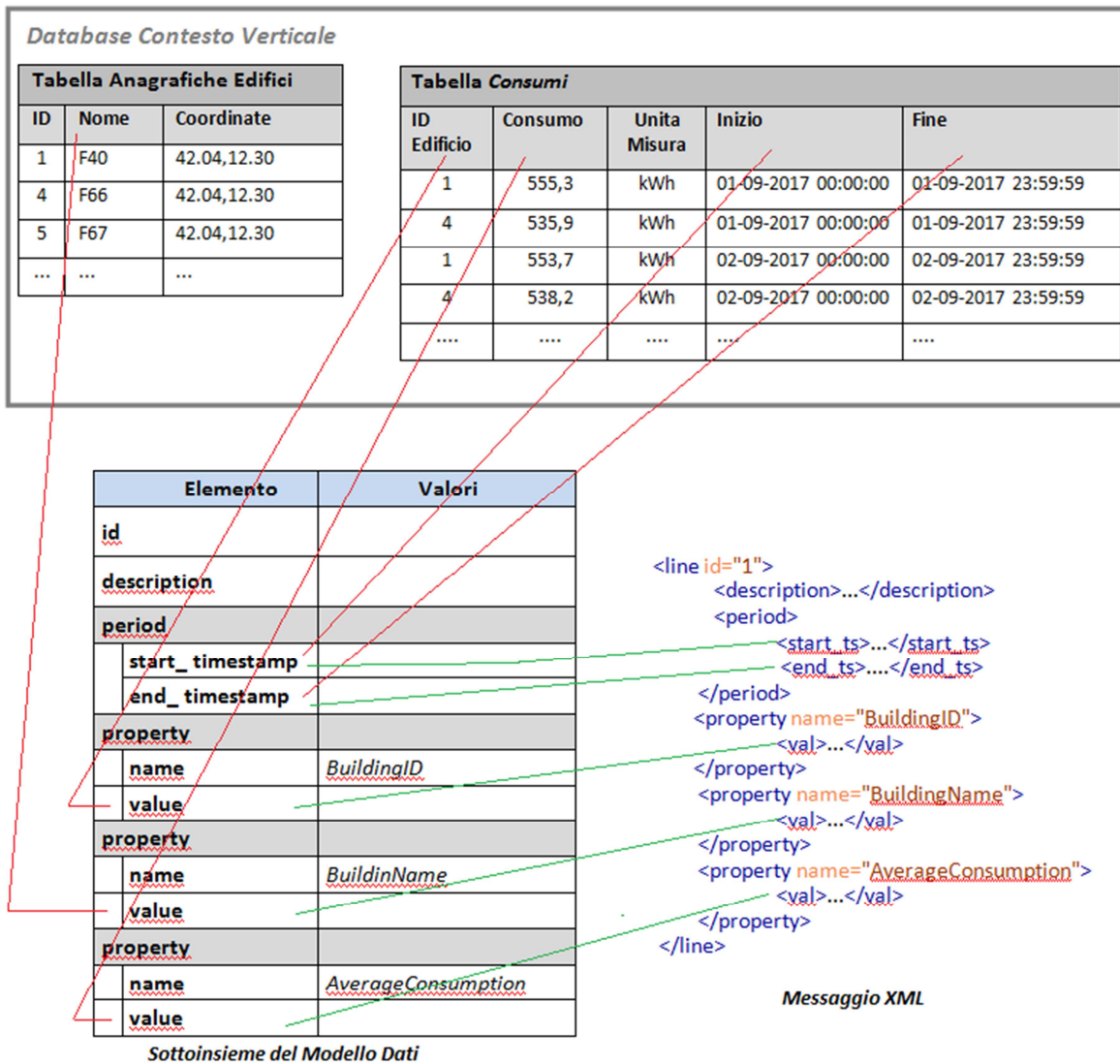


Figura 51 – Esempio di mappatura

### 8.5 Esempio: Urban Dataset Whatever

Figura 52 e Figura 53 mostrano un esempio di Urban Dataset "Whatever" in sintassi XML e JSON; ricordiamo che Urban Dataset "Whatever" può essere utilizzato per inviare set di dati relativi a proprietà non ancora definite nell'Ontologia.

Questo esempio mostra anche come definire e valorizzare una proprietà aggregata.

L'esempio riguarda l'invio di dati relativi a eventi sismici rilevati in Italia; per ogni evento vengono indicati l'istante temporale in cui si è verificato, l'epicentro e la magnitudo.

Si noti che:

- nel blocco *context* sono indicate le coordinate di riferimento utilizzate per l'Italia, visto che l'Urban Dataset fa riferimento a eventi che si verificano in questa zona geografica;
- a livello di riga non sono riportate le coordinate poiché il contesto geografico è uguale per tutte le righe ed è quello indicato a livello di Urban Dataset;
- per ogni riga è indicato il timestamp, poiché l'istante di generazione dei valori riportati varia di riga in riga ed, quindi, diverso dall'istante di generazione dell'Urban Dataset;
- a livello di riga non è, invece, indicato il periodo poiché si tratta di misurazioni istantanee che coincidono con l'istante di generazione dei valori indicati.

```

<UrbanDataset>
  <specification>
    <id schemeID="SCPS" > 1.0-Whatever </id>
    <name>Whatever</name>
    <uri> http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#Whatever
  </uri>
  <properties>
    <propertyDefinition>
      <propertyName>Epicentre</propertyName>
      <propertyDescription>Epicentro Evento Sismico</propertyDescription>
      <subProperties>
        <propertyName>Latitude</propertyName>
        <propertyName>Longitude</propertyName>
      </subProperties>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>Latitude</propertyName>
      <propertyDescription>Distanza angolare di un punto dall'Equatore in Gradi decimali.</propertyDescription>
      <dataType>double</dataType>
      <unitOfMeasure> WGS84-DD </unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>Longitude</propertyName>
      <propertyDescription>Distanza angolare di un punto dal Meridiano 0 in Gradi decimali.</propertyDescription>
      <dataType>double</dataType>
      <unitOfMeasure> WGS84-DD </unitOfMeasure>
    </propertyDefinition>
    <propertyDefinition>
      <propertyName>Magnitude</propertyName>
      <propertyDescription>Dimensione dell'evento sismico.</propertyDescription>
      <dataType>double</dataType>
      <unitOfMeasure>ML</unitOfMeasure>
    </propertyDefinition>
  </properties>
</specification>

```

```

<context>
  <producer>
    <id>INGV-ISIDe</id>
  </producer>
  <timeZone>UTC</timeZone>
  <timestamp>2017-11-19T13:50:00</timestamp>
  <coordinates>
    <longitude>12.56</longitude>
    <latitude>41.87</latitude>
  </coordinates>
  <language>IT</language>
</context>
<values>
  <line id="1">
    <description>Dati evento sismico in Italia.</description>
    <timestamp>2017-11-19T13:37:44</timestamp>
    <property name="Epicentre">
      <property name="Latitude">
        <val>46.06</val>
      </property>
      <property name="Longitude">
        <val>10.03</val>
      </property>
      <property name="Magnitude">
        <val>4.4</val>
      </property>
    </line>
    <line id="2">
      <description>Dati di un evento sismico in Italia.</description>
      <timestamp>2017-11-19T13:42:17</timestamp>
      <property name="Epicentre">
        <property name="Latitude">
          <val>44.65</val>
        </property>
        <property name="Longitude">
          <val>10.07</val>
        </property>
        <property name="Magnitude">
          <val>2.4</val>
        </property>
      </line>
      <line id="3">
        <description>Dati di un evento sismico in Italia.</description>
        <timestamp>2017-11-19T13:49:41</timestamp>
        <property name="Epicentre">
          <property name="Latitude">
            <val>44.68</val>
          </property>
          <property name="Longitude">
            <val>10.01</val>
          </property>
          <property name="Magnitude">
            <val>2.1</val>
          </property>
        </line>
      </values>

```

Figura 52 – Esempio di implementazione XML di un Urban Dataset Whatever

```

{"UrbanDataset": {"specification": {
  "id": {"value": " 1.0-Whatever "},
  "name": "Whatever",
  "uri": " http://smartcityplatform.enea.it/specification/semantic/1.0/ontology/scps-ontology-1.0.owl#Whatever
",
  "properties": {"propertyDefinition": [
    {
      "propertyName": "Epicentre",
      "propertyDescription": "Epicentro Evento Sismico",
      "subProperty": [
        {"propertyName": "Latitude"},
        {"propertyName": "Longitude"}
      ]
    },
    {
      "propertyName": "Latitude",
      "propertyDescription": "Distanza angolare di un punto dall'Equatore in Gradi decimali.",
      "dataType": "double",
      "unitOfMeasure": " WGS84-DD "
    },
    {
      "propertyName": "Longitude",
      "propertyDescription": "Distanza angolare di un punto dal Meridiano 0 in Gradi decimali.",
      "dataType": "double",
      "unitOfMeasure": " WGS84-DD "
    },
    {
      "propertyName": "Magnitude",
      "propertyDescription": "Dimensione dell'evento sismico.",
      "dataType": "double",
      "unitOfMeasure": "ML"
    }
  ]
},
  "context": {
    "producer": {
      "id": "INGV-ISIDe"
    },
    "timeZone": "UTC",
    "timestamp": "2017-11-19T13:50:00",
    "coordinates": {
      "longitude": 12.56,
      "latitude": 41.87
    },
    "language": "IT"
  },
  "values": {"line": [
    {
      "id": 1,
      "description": "Dati evento sismico in Italia.",
      "timestamp": "2017-11-19T13:37:44",

```

```

"property": [
  {
    "name": "Epicentre",
    "property": [
      { "name": "Latitude",
        "val": "46.06"
      },
      { "name": "Longitude",
        "val": "10.03"
      }
    ]
  },
  { "name": "Magnitude",
    "val": "4.4"
  }
]
},
{
  "id": 2,
  "description": "Dati evento sismico in Italia.",
  "timestamp": "2017-11-19T13:42:17",
  "property": [
    {
      "name": "Epicentre",
      "property": [
        { "name": "Latitude",
          "val": "44.65"
        },
        { "name": "Longitude",
          "val": "10.07"
        }
      ]
    },
    { "name": "Magnitude",
      "val": "2.4"
    }
  ]
},
{
  "id": 3,
  "description": "Dati evento sismico in Italia.",
  "timestamp": "2017-11-19T13:49:41",
  "property": [
    { "name": "Epicentre",
      "property": [
        { "name": "Latitude",
          "val": "44.68"
        },
        { "name": "Longitude",
          "val": "10.01"
        }
      ]
    },
    { "name": "Magnitude",
      "val": "2.1"
    }
  ]
}
]
}
}
}

```

Figura 53 – Esempio di implementazione JSON di un Urban Dataset Whatever

## 8.6 Risorse disponibili

Le risorse disponibili per il Livello Information possono essere suddivise in due macro-categorie:

- *Specification*
- *Case Studies Resources*.

La categoria **Specification** include le specifiche di riferimento del formato Urban Dataset, quindi:

- gli schemi che implementano il Modello Dati Astratto nelle due sintassi di riferimento:
  - XML Schema (*scps-urbandataset-schema-1.0.xsd*)
  - JSON Schema (*scps-urbandataset-schema-1.0.json*)
- del materiale a supporto dell'implementazione; in particolare:
  - un **JSON** e un **XML Campione**: sono istanze con dati fittizi che hanno lo scopo di illustrare la struttura delle istanze Urban Dataset
  - una **Guida all'implementazione** in formato XML dell'Urban Dataset (nei formati .pdf e .html, equivalenti). La guida fornisce una rappresentazione "human-friendly" dell'XML Schema dell'Urban Dataset
  - una **Guida all'implementazione** in formato JSON dell'Urban Dataset (in formato .pdf). La guida fornisce una rappresentazione "human-friendly" del JSON Schema dell'Urban Dataset.

La categoria **Case Studies Resources** include del materiale di supporto (esempi, template,...) realizzato per i casi studio analizzati volto a facilitare e illustrare l'adozione delle specifiche in questi scenari. Per esempio:

- istanze XML e JSON di esempio relative ai casi studio supportati
- materiale a supporto dell'implementazione dei casi studio supportati:
  - **JSON e XML Campione** (sono delle istanze in cui stati valorizzati solo gli elementi e attributi che hanno valore fisso per l'Urban Dataset del caso studio a cui si riferiscono, per esempio i "propertyName"; gli altri elementi e attributi, per esempio i "val", non sono valorizzati)
  - **Guide all'implementazione** dell'Urban Dataset (in formato XML e JSON) specifiche per caso studio
  - liste di codici, espresse in formato Genericode, utilizzate solo in specifici casi studio.

In futuro includerà anche artefatti per la validazione delle istanze (per es. schemi customizzati, schematron,...).

## 9 APPENDICE F – SCPS Communication Level

Le Specifiche del Communication Level (o “Livello Comunicazione”) descrivono il protocollo di Trasporto per lo scambio dati che abilita la comunicazione tra Smart City Platform (SCP) orizzontale e una o più Solution verticali (si veda cap.1, “Functional Level”) allo scopo di raggiungere due principali obiettivi:

3. permettere alla Smart City Platform di recuperare i dati da ogni Solution;
4. abilitare la comunicazione tra Solution verticali (inter-solution communication).

Prima di descrivere il protocollo di trasporto definito, riprendiamo alcuni concetti espressi nella descrizione dell’Architettura di Riferimento (livello Functional, capitolo 1):

- ogni Solution verticale è una piattaforma software locale a un Contesto Applicativo specifico della Smart City, e si occuperà della comunicazione con la Smart City Platform;
- la comunicazione è abilitata presso la SCP tramite una serie di informazioni concordate e immagazzinate nel Registry centrale (Collaboration Level);
- ogni Solution è producer e/o consumer di set di dati, denominati Urban Dataset;
- gli Urban Dataset hanno una definizione semantica e sintattica centralizzata (livello Semantic, capitolo 3) e sono rappresentati nel formato comune (livello Information, capitolo 4).

Il protocollo di trasporto delle specifiche SCPS si può riassumere in due azioni principali che una Solution verticale può decidere di intraprendere per comunicare con la Smart City Platform:

1. Produzione di Urban Dataset: esportazione, da parte di una Solution verticale (p.es. Smart Building platform), di dati che saranno recuperati dalla Smart City Platform;
2. Accesso a Urban Dataset: accesso ai dati che una Solution verticale (p.es. WebGIS platform) attua presso la Smart City Platform che li ha precedentemente pubblicati.

Se la prima azione abilita il recupero di dati da Solution a Smart City Platform, e la seconda azione abilita l’accesso ai dati da Smart City Platform a Solution, l’insieme coordinato delle due azioni permette lo scambio di dati da una Solution verticale a un’altra tramite la Smart City Platform.

La comunicazione può quindi avvenire in diversi modi, dipendentemente dal pattern di scambio dati scelto, dal protocollo di trasporto, dal web service implementato: la SCPS Communication Level descrive un insieme di possibili configurazioni per attuare tale comunicazione.

Nei prossimi paragrafi verrà descritto il protocollo di trasporto che permette di abilitare questa comunicazione, in termini di:

- Pattern;
- Interfaccia Web Service.

## 9.1 Pattern

Nei seguenti paragrafi vengono presentati alcuni pattern architetturali che sono delineati nelle specifiche “Smart City Platform Specification - Communication Level” e che quindi devono essere implementati da una Smart City Platform che aderisce a tali specifiche.

Ricordiamo che nell'ambito dell'ingegneria del software:

- Un “design pattern” può essere definito come "una soluzione progettuale generale a un problema ricorrente". Si tratta di una descrizione o modello logico da applicare per la risoluzione di un problema che può presentarsi in diverse situazioni durante le fasi di progettazione e sviluppo del software, ancor prima della definizione dell'algoritmo risolutivo della parte computazionale.
- Un “pattern architetturale” esprime uno schema per l’organizzazione strutturale di sistemi software.

Dunque, i design pattern sono orientati agli oggetti e mostrano relazioni e interazioni fra classi o oggetti; i pattern architetturali, invece, si pongono a un livello più alto e descrivono un pattern complessivo adottato dall’intero sistema.

In questo paragrafo si presenteranno 3 pattern architetturali:

1. Request/Response;
2. Push;
3. Publish/Subscribe;

che potranno essere utilizzati per implementare la comunicazione tra Solution Verticali e Smart City Platform, dipendentemente dalle necessità insite nel caso da gestire.

### 9.1.1 Pattern REQUEST/RESPONSE

“REQUEST/RESPONSE” è uno dei pattern più classici, è lo stesso che è usato, per esempio, nella navigazione delle pagine web: un Client digita un URL e in risposta ottiene la pagina web desiderata dal Server.

Nelle chiamate Web Service su HTTP, il pattern REQUEST/RESPONSE viene implementato in modo sincrono: una volta effettuata la richiesta dal componente Client, si mantiene una connessione aperta fino a quando la risposta del componente Server viene consegnata, oppure scade un timer (dunque è bloccante).

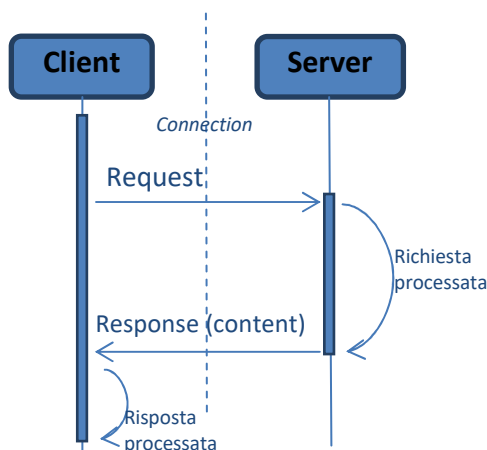


Figura 54 – Interazione Client-Server nel pattern Request/Response

Adottando il pattern REQUEST/RESPONSE per abilitare la comunicazione basata su SCPS, si intende la seguente implementazione di servizio:



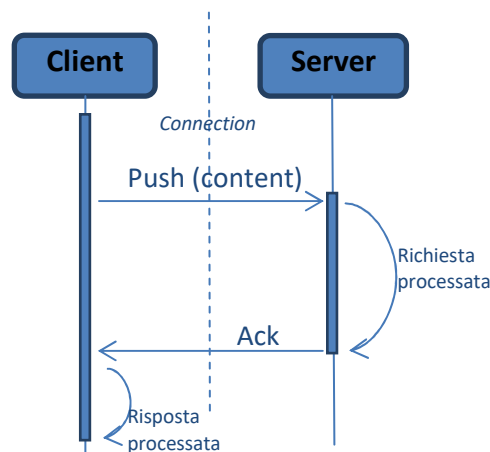
- 1- Il componente Client effettua una REQUEST per RICHIEDERE un determinato Urban Dataset;
- 2- Il componente Server restituisce la RESPONSE contenente l'Urban Dataset richiesto.

Si noti che, dipendentemente dalla situazione, Client e Server possono essere ruoli interpretati da una Solution Verticale e dalla Smart City Platform e viceversa.

### 9.1.2 Pattern PUSH

Il pattern "PUSH" è molto al pattern "request/response" ma il Client, invece di richiedere qualcosa, invia nella sua chiamata un contenuto che vuole sia ricevuto dal Server, che potrà rispondere con un acknowledgment (si veda in proposito il paragrafo **Errore. L'origine riferimento non è stata trovata.** "Consegna affidabile").

Nelle chiamate Web Service su HTTP, il pattern PUSH viene implementato tipicamente in modo sincrono: una volta effettuata la chiamata dal componente Client, si mantiene una connessione aperta fino a quando l'ack del componente Server viene consegnato, oppure scade un timer (dunque è bloccante).



**Figura 55 – Interazione Client-Server nel pattern Push**

Adottando il pattern PUSH per abilitare la comunicazione basata su SCPS, si intende la seguente implementazione di servizio:

- 1- Il componente Client effettua una chiamata PUSH per INVIARE un determinato UrbanDataset;
- 2- Il componente Server riceve l'UrbanDataset, lo processa e risponde con un Ack.

Si noti che, dipendentemente dalla situazione, Client e Server possono essere ruoli interpretati da una Solution Verticale e dalla Smart City Platform e viceversa.

## 9.2 Web Service Interface

In questo paragrafo viene descritta l'interfaccia del web service *UrbanDatasetGateway*; l'implementazione di questo web service è necessaria per aderire alle specifiche SCPS Communication Level.

Il web service *UrbanDatasetGateway* è sviluppato su protocollo HTTP/HTTPS con architettura REST (REpresentational State Transfer), e implementa i pattern architetturali presentati nel par. **Errore. L'origine riferimento non è stata trovata.:**

- REQUEST/RESPONSE: per consentire a un client di richiedere un Urban Dataset a un server in ascolto (p.es. la Smart City Platform, come client, richiede un Urban Dataset alla Solution verticale che espone il servizio, come server, il quale risponde con l'Urban Dataset richiesto);
- PUSH: per consentire a un client di inviare direttamente un Urban Dataset a un destinatario in ascolto (p.es. una Solution verticale, come client, invia un Urban Dataset alla Smart City Platform che espone il servizio, come server, e riceve l'Urban Dataset);
- PUBLISH/SUBSCRIBE: per consentire la trasmissione di dati tra una Solution verticale "publisher" e N Solution verticali "subscriber", tramite la Smart City Platform che fungerà da "broker"; questo particolare pattern potrà essere implementato tramite il riutilizzo del pattern "PUSH".

Viene definito un singolo servizio web che, implementato sia lato Smart City Platform, sia lato Solution verticale, permette la comunicazione in entrambe le direzioni secondo i pattern PUSH e REQUEST/RESPONSE.

N.B. nella descrizione dei PATTERN viene descritto il meccanismo di ricezione affidabile tramite re-invio del messaggio, numero di tentativi e timeout, con conseguente gestione del messaggio non consegnato.

Ciò deve essere supportato anche nell'implementazione del servizio di trasporto.

I parametri di configurazione del servizio devono essere immagazzinati:

- Nel Registry lato SCP;
- In un file di configurazione lato Solution verticale.

Le seguenti tabelle descrivono l'interfaccia del Web Service e prescindono dall'implementazione.

Servizio	UrbanDatasetGateway
Metodi	login (username, password)
	push (token, dataset)
	basicRequest (token, collaboration_id)
	searchingRequest (token, collaboration_id, from_utc_datetime*, to_utc_datetime*, coordinate_center*, radius*)

\*valori opzionali

N.B. il metodo searchingRequest con tutti i parametri opzionali mancanti, richiama la basicRequest

### 9.2.1 login method interface

Metodo	login
Parametro	username tipo: string/string descrizione: username fornito dal client chiamante il metodo
Parametro	password tipo: string/password descrizione: password fornita dal client
Valore di Ritorno	token tipo: string/auth-token descrizione: se l'autenticazione ha avuto successo viene ritornato un token (minimo 10 caratteri) che permetterà il successivo accesso.

	Valore di Ritorno	error tipo: string/code descrizione: codice di 3 cifre in formato stringa. Possibili valori per questo metodo: 100 autenticazione fallita
--	-------------------	--

### 9.2.2 push method interface

	Metodo	push
	Parametro	token tipo: string/auth-token descrizione: il token di autenticazione per l'accesso, viene rilasciato a ogni client che effettua la login con successo e ha una validità temporale
	Parametro	dataset tipo: string/dataset descrizione: rappresentazione dell'Urban Dataset secondo specifiche SCPS in XML o JSON
	Valore di Ritorno	success tipo: string/code descrizione: codice di 3 cifre in formato stringa. Possibile valore in caso di <i>success</i> : 000 di transazione avvenuta con successo
	Valore di Ritorno	error tipo: string/code descrizione: codice di 3 cifre in formato stringa. Possibile valore in caso di <i>error</i> : 101 token invalido 102 permesso invalido per la produzione/accesso all'urban dataset 103 dataset invalido secondo lo schema 104 system error 105 bad request

### 9.2.3 basicRequest method interface

	Metodo	basicRequest
	Parametro	token tipo: string/auth-token descrizione: il token di autenticazione per l'accesso, viene rilasciato a ogni client che effettua la login con successo e ha una validità temporale
	Parametro	collaboration_id tipo: string/collaboration_id descrizione: identifica univocamente un Urban Dataset prodotto da una specifica Solution producer nell'ambito di una specifica collaborazione, in una specifica Smart City Platform.  Se il web service è pubblicato sulla Smart City Platform, il valore del <i>collaboration_id</i> può essere recuperato da una Solution registrata che ha accesso a tale UrbanDataset.  Se il web service è pubblicato sulla Solution verticale, il valore del <i>collaboration_id</i> coincide con l'identificatore dell'UrbanDataset.
	Valore di Ritorno	dataset tipo: string/dataset descrizione: rappresentazione dell'Urban Dataset secondo specifiche SCPS in XML o JSON
	Valore di Ritorno	error tipo: string/code descrizione: codice di 3 cifre in formato stringa. Possibile valore in caso di <i>error</i> : 101 token invalido 102 permesso invalido per la produzione/accesso all'urban dataset 103 dataset invalido secondo lo schema 104 system error 105 bad request

9.2.4 searchingRequest method interface

Metodo	searchingRequest
Parametro	<p>token</p> <p>tipo: string/auth-token</p> <p>descrizione: il token di autenticazione per l'accesso, viene rilasciato a ogni client che effettua la login con successo e ha una validità temporale</p>
Parametro	<p>collaboration_id</p> <p>tipo: string/collaboration_id</p> <p>descrizione: identifica univocamente un Urban Dataset prodotto da una specifica Solution producer nell'ambito di una specifica collaborazione, in una specifica Smart City Platform.</p> <p>Se il web service è pubblicato sulla Smart City Platform, il valore del <i>collaboration_id</i> può essere recuperato da una Solution registrata che ha accesso a tale UrbanDataset.</p> <p>Se il web service è pubblicato sulla Solution verticale, il valore del <i>collaboration_id</i> coincide con l'identificatore dell'UrbanDataset.</p>
	<p>from_utc_datetime</p> <p>tipo: string/UTC_DATETIME</p> <p>descrizione: data e ora da cui si vuole specificare la partenza di un intervallo temporale. Nel caso sia assente, la finestra temporale include tutti i valori possibili.</p>
	<p>to_utc_datetime</p> <p>tipo: string/UTC_DATETIME</p> <p>descrizione: data e ora da cui si vuole specificare la fine di un intervallo temporale. Nel caso sia assente, la finestra temporale include tutti i valori possibili da <i>from_utc_datetime</i> fino al momento in cui viene effettuata la chiamata. Nel caso sia assente <i>from_utc_datetime</i> ma presente <i>to_utc_datetime</i> viene ritornato un errore 105 "bad request".</p>
	<p>coordinate_center_x</p> <p>tipo: double/COORDINATE_WGS84</p> <p>descrizione : Coordinata X del centro della ricerca spaziale. Nel caso sia assente, la finestra spaziale include tutti i valori possibili. Nel caso sia assente <i>coordinate_center_x</i> ma presente <i>coordinate_center_y</i> o <i>radius</i> viene ritornato un errore 105 "bad request".</p>
	<p>coordinate_center_y</p> <p>tipo: double/COORDINATE_WGS84</p> <p>descrizione : Coordinata Y del centro della ricerca spaziale. Nel caso sia assente, la finestra spaziale include tutti i valori possibili. Nel caso sia assente <i>coordinate_center_y</i> ma presente <i>coordinate_center_x</i> o <i>radius</i> viene ritornato un errore 105 "bad request".</p>
	<p>radius</p> <p>tipo: double/DISTANCE_WGS84</p> <p>descrizione: raggio del cerchio con centro <i>coordinate_center</i>. Nel caso sia assente, la finestra spaziale include tutti i valori possibili. Nel caso sia presente <i>radius</i> ma assente <i>coordinate_center_x</i> e/o <i>coordinate_center_y</i> viene ritornato un errore 105 "bad request".</p>
Valore di Ritorno	<p>dataset</p> <p>tipo: string/dataset</p> <p>descrizione: rappresentazione dell'Urban Dataset secondo specifiche SCPS in XML o JSON</p>
Valore di Ritorno	<p>error</p> <p>tipo: string/code</p> <p>descrizione: codice di 3 cifre in formato stringa. Possibile valore in caso di <i>error</i>:</p> <ul style="list-style-type: none"> <li>101 token invalido</li> <li>102 permesso invalido per la produzione/accesso all'urban dataset</li> <li>103 dataset invalido secondo lo schema</li> <li>104 system error</li> <li>105 bad request</li> </ul>

## 10 APPENDICE G – SCPS Terminologia

### 10.1 Terminologia dallo Stato dell'Arte

Termine	Definizione	Fonte
<b>Architettura di Riferimento</b>	Un'Architettura di Riferimento è un <b>modello di progetto architeturale</b> che indica come un insieme astratto di relazioni si concretizzi in un insieme di requisiti. Lo scopo principale di un'Architettura di Riferimento è fornire una guida allo sviluppo di architetture concrete. Da uno stesso <u>Modello di Riferimento</u> si possono derivare diverse architetture di riferimento	OASIS (14)
<b>Cloud Computing</b>	<p>Il Cloud Computing è un modello di business in cui l'utente non compra il prodotto, ma la possibilità di utilizzare quel prodotto senza disporre fisicamente. Si tratta sostanzialmente dell'uso di servizi software ospitati su risorse hardware e software localizzate su Internet.</p> <p>Questo può avvenire in tre modalità:</p> <ul style="list-style-type: none"> <li>➤ Infrastructure as a Service (IaaS): si noleggia l'infrastruttura (computer virtuali) in cui far girare le proprie applicazioni. In pratica si offrono all'utente server e spazio di archiviazione;</li> <li>➤ Platform as a Service (PaaS): rispetto all'IaaS si offre anche una piattaforma software collegata per costruire le proprie applicazioni;</li> <li>➤ Software as a Service (SaaS): applicazioni complete disponibili nel cloud.</li> </ul>	
<b>Data fusion</b>	Si definisce Data Fusion il processo di analisi dei segnali, provenienti da un insieme vario di sensori, eseguito da un Sistema Informatico, al fine ottenere risultati omogenei e utilizzabili.	
<b>Internet delle Cose (IoT)</b>	Internet delle cose (o Internet of Things) è neologismo riferito all'estensione di Internet al mondo degli oggetti e dei luoghi concreti.	<i>AUTO-ID Center (consorzio di ricerca con sede al MIT)</i>
<b>Interoperabilità</b>	Capacità di un sistema di scambiare informazioni con altri sistemi differenti, senza errori e senza ambiguità, sia dal punto di vista sintattico che semantico	
<b>Middleware</b>	Termine generale per indicare sistemi software che servono a mediare fra due programmi separati e, di solito, già esistenti.	
<b>Modello di Riferimento (Architeturale) o Framework</b>	Struttura astratta che permette di comprendere le relazioni fra le entità di un determinato ambiente. Esso permette, inoltre, lo sviluppo di specifiche architetture di riferimento. Un Modello di Riferimento consiste di un <b>insieme minimale di concetti unificanti, assiomi e relazioni</b> .	OASIS (14)
<b>Model Driven Engineering</b>	Metodologia di sviluppo software che si focalizza sulla creazione di modelli astratti orientati ai concetti di dominio	

<b>Sistema Ciberfisico (CPS)</b>	Sistema informatico in grado di interagire in modo continuo con il sistema fisico in cui opera. Il sistema è composto da elementi fisici dotati ciascuno di capacità computazionale e riunisce strettamente le cosiddette "tre C": capacità computazionale, comunicazione e capacità di controllo	NIST (15)
<b>Sistema di acquisizione dati</b>	Qualsiasi sistema in grado di rilevare e memorizzare grandezze analogiche e/o digitali	
<b>Standard Aperto</b>	Standardizza formati dati e documento ai fini dell'interoperabilità del Software.  Per essere definito tale, deve possedere i seguenti requisiti: <ul style="list-style-type: none"> <li>- processo di decisione collaborativo e trasparente;</li> <li>- definito da un organizzazione di standardizzazione riconosciuta;</li> <li>- supportato dai mercati e indipendente dai vendor;</li> <li>- disponibile pubblicamente a un costo nullo o, comunque, accessibile;</li> <li>- diritti licenziati senza royalty.</li> </ul>	

## 10.2 Terminologia Smart City ENEA

Termine	Definizione
<b>Acquisizione Dati</b>	Per Acquisizione Dati s'intende l'insieme di tecnologie e metodologie (protocolli e formati) che consentono di ottenere dati digitali provenienti da una o più <u>Reti Applicative</u> e immagazzinarli in un <u>Database di Acquisizione</u> . ➤ Esempio: Nel caso dei quadri elettrici della Rete Applicativa dell'illuminazione pubblica l'Acquisizione Dati si riferisce ai Point Point of Delivery (POD) provenienti.
<b>Area di Gestione</b>	Convenzione schematica per separare e organizzare il Ciclo di Vita dei Dati in tre parti distinte: <ol style="list-style-type: none"> <li>1. Area Campo: area che descrive il <u>Flusso Dati</u> dalla <u>Rete Applicativa</u> in oggetto fino all' <u>Acquisizione Dati</u> e allo <u>Storage Dati</u> nel <u>Database Centrale</u>;</li> <li>2. Area Piattaforma Locale: area che descrive l'<u>Integrazione e Trasformazione Dati</u> allo scopo di eseguire la <u>Diagnostica</u>, di generare i <u>KPI</u> e di creare dei <u>Servizi</u> per l'<u>Utente finale</u>;</li> <li>3. Area Smart District Platform: area che descrive il <u>Flusso Dati</u> da ogni <u>Piattaforma Locale</u> verticale alla <u>Smart District Platform</u>.</li> </ol>
<b>Caso d'Uso (o Use Case)</b>	Un Caso d'Uso è una specifica formale, definita tramite diagrammi e descrizioni testuali, che permette di rappresentare un insieme di attività logicamente connesse, che sono una parte del sistema in oggetto. L'insieme dei casi d'uso permette di individuare i requisiti necessari all'implementazione.
<b>Caso Studio</b>	Un Caso Studio è l'analisi, a partire da un caso reale, e la descrizione formale dei relativi flussi di dati fra una Solution Verticale e la Smart City Platform o fra due Solution Verticali tramite la SCP. Un caso studio permette di individuare i requisiti necessari alla definizione delle SCPS e di predisporre l'implementazione reale dello scambio dati tramite una Smart City Platform esistente. ➤ Esempio: il Caso d'Uso "Smart Eye", relativo all'Area Campo, è una descrizione del flusso dati dal sensore ottico Smart Eye, che è usualmente applicato sui pali dell'illuminazione pubblica, al Database Centrale per l'Acquisizione Dati sul traffico.
<b>Code list (o lista di codici)</b>	Istanza di un vocabolario controllato, ovvero insieme predefinito di valori in cui ogni valore rappresenta una semantica concordata tra le parti che lo usano. Favorisce l'interoperabilità fra le applicazioni che lo adottano e, supportando il multilinguismo,

consente a utenti di comunità diverse di meglio cogliere il significato dei valori che rappresenta.

<b>Conformità</b>	Aderenza a una specifica senza violazione dei vincoli da essa imposti e con il supporto di tutti i requisiti richiesti.
<b>Contesto Applicativo</b>	<p>Una Contesto Applicativo è un insieme concettuale di aspetti tecnologici, sociali ed economici, logicamente connessi, che permette, tramite il monitoraggio e il successivo utilizzo dei dati, di ottenere un miglioramento in termini di gestione efficiente dell'energia e delle risorse e/o in termini di miglioramento della qualità della vita.</p> <ul style="list-style-type: none"> <li>➤ p.es. Sono Contesti Applicativi: smart lighting, smart home, mobility, smart community.</li> </ul> <p>Un Contesto Applicativo è gestito usualmente tramite una o più <u>Solution</u> e può essere associato a una o più <u>Reti Applicative</u>.</p> <ul style="list-style-type: none"> <li>➤ p.es. In un Contesto Applicativo smart street, potrei utilizzare la tecnologia Smart-Eye, integrando i dati provenienti dalle Reti Applicative relative a lighting e mobility, gestendo i dati tramite una Solution Smart Street Platform.</li> </ul>
<b>Customizzazione</b>	Specializzazione di una specifica che ne riduce le ambiguità e libertà interpretative e meglio supporta le esigenze di uno specifico dominio/contesto pur restando conforme alla specifica.
<b>Ciclo di Vita dei Dati</b>	Il Ciclo di Vita dei Dati è la descrizione del <u>Flusso dei Dati</u> , provenienti da una o più <u>Reti Applicative</u> , che sono sottoposti all' <u>Acquisizione</u> , <u>Integrazione</u> , <u>Trasformazione Dati</u> , fino alla <u>Pubblicazione Dati</u> tramite <u>Servizi</u> finali per l' <u>Utente</u> e alla condivisione con la <u>Smart District Platform</u> .
<b>Database di Acquisizione (o Database Centrale)</b>	Database (o DB, o Banca Dati) preposto all'acquisizione e memorizzazione ( <u>Storage Dati</u> ) dei dati digitali provenienti da una o più <u>Reti Applicative</u> . Il Database può essere organizzato in uno o più schemi distinti, per organizzare le diverse fasi di elaborazione del dato ( <u>acquisizione</u> , <u>trasformazione</u> , <u>integrazione</u> , <u>pubblicazione</u> ).
<b>Diagnostica</b>	La Diagnostica è l'azione di <u>Monitoraggio Dati</u> volta all'individuazione di anomalie tramite l'applicazione di specifici algoritmi.
<b>Flusso (o Flow)</b>	<p>Un Flusso è una transizione (di persone, di energia, di beni, di fenomeni ambientali, o d' informazioni) da uno <u>Stato</u> sorgente a uno Stato destinazione, tramite passaggio per eventuali stati intermedi.</p> <ul style="list-style-type: none"> <li>➤ p.es.: è un flusso il passaggio della corrente elettrica tra due centraline o la variazione della temperatura esterna o le informazioni relative a una rete sociale.</li> </ul>
<b>Flusso Dati</b>	<p>Un Flusso Dati è un particolare caso di Flusso relativo alle informazioni digitali.</p> <ul style="list-style-type: none"> <li>➤ p.es.: è un Flusso Dati il passaggio di informazioni in formato digitale tra un sensore di rilevamento della variazione di temperatura e il <u>Database di Acquisizione</u>.</li> </ul>
<b>Integrazione Dati (o Data Integration)</b>	<p>Per Integrazione Dati s'intende un processo che, applicato ai dati provenienti da due o più Reti Applicative,</p> <ul style="list-style-type: none"> <li>– fornisce all'utente una visione unificata dei dati</li> <li>– permette creazione di nuova conoscenza grazie alla loro combinazione, combinazione, in una <u>Logica Smart</u>.</li> <li>➤ Esempio: è un'Integrazione Dati, la nuova conoscenza che viene generata combinando i dati forniti da Reti applicative meteo con quelli di consumo energetico di uno Smart Building, ai fini di ottimizzare il comportamento energetico di chi abita in quell'edificio</li> </ul>

<b>KPIs (o Key Performance Indicators)</b>	I Key Performance Indicators sono indici di prestazione, ricavati dalla <u>Trasformazione Dati</u> e utilizzati per avere una conoscenza più ampia di un <u>Contesto Applicativo</u> (per esempio su un periodo di tempo).
<b>Logica Smart</b>	Una Logica Smart consiste in un'azione il cui scopo è un miglioramento inteso sia in termini di consumi energetici e di risorse e/o sia in termini di qualità della vita. La Logica Smart avviene tramite l' <u>Acquisizione di dati</u> provenienti da una o più <u>Reti Applicative</u> e dal loro conseguente trattamento ( <u>monitoraggio</u> , <u>integrazione</u> , <u>trasformazione</u> , <u>pubblicazione</u> ) <ul style="list-style-type: none"> <li>➤ Esempio: il monitoraggio della mobilità per la gestione energetica dell'illuminazione pubblica è una <u>Logica Smart</u>.</li> </ul>
<b>Monitoraggio Dati</b>	Per Monitoraggio Dati s'intende l'analisi dei Dati recuperati da una <u>Rete Applicativa</u> al fine di ricavare nuova conoscenza che può essere utilizzata in una <u>Logica Smart</u> . <ul style="list-style-type: none"> <li>➤ p.es. E' un Monitoraggio Dati quello relativo all'analisi dei Dati Recuperati dalla Rete Applicativa dell'Illuminazione Pubblica ai fini della <u>Diagnostica</u></li> </ul>
<b>Piattaforma Locale</b>	Una Piattaforma Locale è una piattaforma ICT che permette di gestire uno specifico <u>Contesto Applicativo</u> . <ul style="list-style-type: none"> <li>➤ P.es. la Piattaforma Locale del contesto applicativo Smart Building, permette di gestire le informazioni relative a una rete di building monitorati all'interno dell'ENEA Casaccia (Facility).</li> </ul>
<b>Pubblicazione Dati</b>	Per Pubblicazione Dati s'intende l'azione di rendere disponibili i Dati (provenienti dall' <u>Acquisizione Dati</u> , dall' <u>Integrazione Dati</u> e dalla <u>Trasformazione Dati</u> ) sotto forma di <u>Servizio</u> , agli <u>Utenti</u> finali.
<b>Rete Applicativa</b>	Una Rete Applicativa, fisica o virtuale, distribuita sul Distretto, Città, Regione o territorio nazionale, identifica un <u>Flusso</u> di persone, di energia, di beni, di mutamenti ambientali o di informazioni. <p>Ogni Rete Applicativa è sorgente di uno specifico <u>Flusso Dati</u> digitale.</p> <ul style="list-style-type: none"> <li>➤ Esempio: sono Reti Applicative: la <i>rete dell'illuminazione pubblica</i>, la <i>rete sociale su internet</i>, la <i>rete della mobilità</i>, la <i>rete idrica</i>.</li> </ul>
<b>Servizio Utente</b>	Un Servizio è un'applicazione software attraverso la quale un <u>Utente</u> può accedere a Dati digitali, risultato dell'applicazione di una <u>Logica Smart</u> , al fine di ottenere un beneficio in termini di efficienza energetica e/o miglioramento della qualità della vita.
<b>Smart District Platform</b>	Con "Smart District Platform" s'intende, nell'ambito delle specifiche, una Piattaforma ICT di <u>integrazione dati</u> che abbia aderito alle specifiche "Smart City Platform Specification" sui cinque livelli di Interoperabilità. <p>Aderendo alle Smart City Platform Specification si predispongono una comunicazione interoperabile tra i diversi <u>Contesti Applicativi</u> urbani.</p>
<b>Solution (o Soluzione)</b>	Insieme di tecnologie e metodologie atte a implementare una o più azioni per ottenere concretamente un miglioramento in termini di gestione dei consumi energetici e di risorse e/o un miglioramento della qualità della vita. Queste azioni consistono tipicamente nell'acquisizione di dati provenienti da una o più <u>reti applicative</u> e dal loro conseguente trattamento (monitoraggio, integrazione, trasformazione, pubblicazione). <p>Tipicamente una Solution corrisponde alla <u>Piattaforma Locale</u> per la gestione di un singolo <u>Contesto Applicativo</u> e, per questo motivo, si farà riferimento a essa proprio tramite il termine "Solution".</p>
<b>Stato (o Nodo)</b>	Descrizione statica che identifica un dato istante e/o un dato luogo geografico.



<b>Storage Dati</b>	Immagazzinamento, permanente o temporaneo, dei Dati digitali che provengono dall' Acquisizione, dall'Integrazione e dalla Trasformazione Dati. Lo Storage Dati utilizza come mezzo fisico un <u>Database di Acquisizione</u> .
<b>Trasformazione Dati</b>	Applicazione di un algoritmo che prende in input un insieme di dati, e restituisce in output un insieme di dati diverso. <ul style="list-style-type: none"> <li>➤ p.es. E' una Trasformazione Dati la generazione di <u>KPI</u></li> </ul>
<b>Use case</b>	➔ Si veda: <i>Caso d'Uso</i>
<b>Utente</b>	Il fruitore finale di un <u>Servizio Utente</u> da cui ricava un beneficio in termini economici, sociali e/o di miglioramento della qualità della vita.
<b>Urban Dataset</b>	Rappresentazione del dato scambiato tramite la Smart City Platform, caratterizzato: <ul style="list-style-type: none"> <li>- da una descrizione semantica univoca e centralizzata, tramite l'Ontologia;</li> <li>- da due formati di rappresentazione, JSON e XML;</li> <li>- da un servizio di trasporto, l'UrbanDataset Gateway, basato su Web Service REST e SOAP.</li> </ul>

### 10.3 Acronimi

Acronimo	Significato	Note
API	Application Programming Interface	
CEN	Comitato Europeo di Normazione	
CENELEC	Comitato europeo di normazione elettrotecnica	
CPS	Cyber Physical System	
ETSI	European Telecommunications Standards Institute	
FI-PPP	Future Internet Publica Private Partership	
IaaS	Infrastructure as a Service	Tipologia di Cloud
ICS	International Classification of Standard	Struttura gerarchica che contiene 40 campi nel primo livello, che si dettagliano in 392 gruppi nel secondo livelli e 909 sottogruppi al livello 2
ICT	Information and Communication Technologies	
IES-City	IoT Enabled Smart City framework	Iniziativa che vede coinvolti, tra gli altri NIST ed ENEA
IoT	Internet of things	
ITU-T	International Telecommunication Union – Telecommunication Standardization Bureau	
JSON	JavaScript Object Notation	Formato di testo per lo scambio di dati tra applicazioni. È basato su un sottoinsieme del linguaggio JavaScript, ma non è un linguaggio di programmazione ed è completamente indipendente da qualsiasi linguaggio di programmazione.
JSON Schema	Schema per documenti JSON	Vocabolario che consente di descrivere e validare documenti JSON.
KPI	Key Performance Indicator	
MDE	Model Driven Engineering	
NIST	National Institute of Standards & Technology (US)	

<b>PaaS</b>	Platform as a Service	Tipologia di Cloud
<b>SaaS</b>	Software as a Service	Tipologia di Cloud
<b>SC</b>	Smart City	
<b>SCP</b>	Smart City Platform	
<b>SCPS</b>	Smart City Platform Specification	Nel contesto di riferimento Smart City –ENEA
<b>SG</b>	Smart Grid	
<b>SGCG</b>	Smart Grid Coordination Group	Coinvolge CEN, CENELEC ed ETSI
<b>SGAM</b>	Smart Grid Architecture Model	Intelaiatura di analisi dei requisiti di interoperabilità dei casi d’uso, sviluppata dallo SGCG
<b>UD</b>	Urban Dataset	Nel contesto di riferimento Smart City –ENEA
<b>WGS84</b>	World Geodetic System 1984	Sistema di coordinate geografiche geodetico, mondiale, basato su un ellissoide di riferimento elaborato nel 1984. Dal punto di vista geometrico, è un particolare sistema terrestre convenzionale (CTS Conventional Terrain System), ovvero un sistema di riferimento cartesiano usato per descrivere la Terra.
<b>XML</b>	eXtensible Markup Language	Formato di testo per la definizione di linguaggi di scambio di dati tra applicazioni. È un metalinguaggio di markup che consente di definire linguaggi di markup per la descrizione di dati di qualsiasi tipo e di documenti strutturati.
<b>XSD</b>	XML Schema Definition Language	Vocabolario che consente di definire e validare la struttura di documenti XML.

## 11 APPENDICE H - SCPS Policy per l'uso degli identificatori

Nel corso del PAR 2017, nell'ambito delle specifiche saranno definite sia le regole di denominazione per la creazione degli identificatori assegnati dalla SCP (i.e. lo schema di identificazione SCPS), sia una lista di schemi di identificazione diversi da quello della SCPS ma da essa ammessi e riconosciuti.

## 12 APPENDICE I - Riferimenti bibliografici

1. **NIST.** *International Technical Working Group on IoT-Enabled Smart City Framework (IES-City)*. [Online] [Riportato: 11 9 2017.] <https://pages.nist.gov/smartcitiesarchitecture/>.
2. **Laura Blaso, Arianna Brutti, Angelo Frascella, Nicoletta Gozo e Cristiano Novelli.** Architetture e Piattaforme di Interoperabilità per le Smart City. *Energia, Ambiente e Innovazione (ENEA Magazine)*. Gennaio-Marzo 2017, p. 34-39.
3. **Maurizio Esitini, Mariangela Merrone e Nicoletta Gozo.** *Schede Censimento – Uno strumento per incentivare i processi d'efficientamento energetico della rete di pubblica illuminazione*. s.l. : Ricerca di Sistema elettrico, 2014.
4. **Yan, L., Zhang, Y., Yang, L. T., & Ning, H.** *The Internet of Things: From RFID to the Next-Generation Pervasive Networked Systems*. s.l. : Auerbach Publications, 2008. ISBN 9781420052817.
5. *System-of-systems—the meaning of. System-of-systems Engineering.* **Boardman, J., Sauser, B.** Los Angeles, CA, USA : s.n., 2006. IEEE/SMC International Conference.
6. *Smart cities e società inclusiva. Il ruolo della standardizzazione.* **Valeria Falce, Elena Di Maggio.** Bologna : s.n., 2015. La linfa della smart city: dati e informazioni. Verso un nuovo paradigma culturale e giuridico.
7. **OASIS.** Code List Representation (Genericode) Version 1.0. [Online] 2007. <http://docs.oasis-open.org/codelist/cs-genericode-1.0/doc/oasis-code-list-representation-genericode.html>.
8. **W3C.** PROV-O: The PROV Ontology. [Online] 2013. <https://www.w3.org/TR/prov-o/>.
9. —. Extensible Markup Language (XML). *W3C*. [Online] 11 10 2016. <https://www.w3.org/XML/>.
10. **IETF.** The JavaScript Object Notation (JSON) Data Interchange Format (RFC 7159). [Online] 14 10 2015. <https://datatracker.ietf.org/doc/rfc7159>.
11. **W3C.** XML Schema (XSD). [Online] W3C, 05 04 2012. <https://www.w3.org/standards/xml/schema>.
12. **JSON-Schema-org.** JSON Schema specification - draft 07. [Online] 19 11 2017. <http://json-schema.org/documentation.html>.
13. **schematron.com.** Schematron. [Online] 2017. <http://schematron.com/>.
14. **OASIS.** *Reference Model for Service Oriented Architecture 1.0*. s.l. : OASIS, 2006.
15. **NIST.** *Framework for Cyber-Physical Systems - Release 0.8*. s.l. : NIST, 2015.
16. **ISO.** *ISO 14258:1998 Industrial automation systems -- Concepts and rules for enterprise models*. s.l. : ISO, 1998.