



Sii-Mobility

Supporto di Interoperabilità Integrato per i Servizi al Cittadino e alla Pubblica Amministrazione

Trasporti e Mobilità Terrestre, SCN_00112

Deliverable ID: DE4.1

**Titolo: Stato dell'arte dei sistemi di acquisizione dati e
ingestion process.**

Data corrente	
Versione (solo il responsabile può cambiare versione)	0.4
Stato (draft, final)	Final
Livello di accesso (solo consorzio, pubblico)	Consorzio
WP	4
Natura (report, report e software, report e HW..)	Report
Data di consegna attesa	
Data di consegna effettiva	
Referente primario, coordinatore del documento	Paolo Nesi
Contributor	<Nome> <Cognome> <email>, <Nome> <Cognome> <email>, <Nome> <Cognome> <email>
Coordinatore responsabile del progetto	Paolo Nesi, UNIFI, paolo.nesi@unifi.it

Sommario

1	Executive Summary	3
2	Panoramica su tecnologie Data Ingestion	4
3	Data Ingestion API verso le sorgenti	5
4	Architetture per il Data Ingestion e Data Aggregation	7
5	Smart City API stato dell'arte	11
6	Algoritmi per la riconciliazione	13
6.1	SPARQL Reconciliation	15
6.2	Link discovering Based Reconciliation and comparison	16
7	Bibliografia	17
8	Acronimi	19

1 Executive Summary

Questo documento riporta lo stato dell'arte dei sistemi di acquisizione dati in ambito smart city e mobilità. Questo viene sviluppato nell'ottica del contesto di Sii-Mobility, ma escludendo tematiche che sono trattate in altri deliverable, come deliverable e algoritmi (come da specifica):

- DE2.2a – stato dell'arte e sulla ricerca effettuata nel campo del vehicle routing, M6
 - A06: Algoritmi di instradamento (veicoli e persone), Attività: 2.2.5
- DE2.4a – stato dell'arte nei problemi di percorso ottimo su grafi multimodali, multi-obiettivo, M6
 - A05: Algoritmi di ottimizzazione (percorsi con più fermate, cambi, etc.), Attività: 2.2.4
 - A07: Algoritmi per la produzione di percorsi per flotte merci, Attività: 2.2.5
- DE2.7a – stato dell'arte della ricerca nel campo dei data analytics, M6
 - A03: Algoritmi di ottimizzazione per la produzione suggerimenti per il parcheggio, Attività: 2.2.1
 - A04a: Algoritmi e strumenti per raccolta e computo flussi di persone, Attività: 2.2.1
 - A04b: Algoritmi e strumenti per raccolta e computo flussi di mezzi, Attività: 2.2.1
- DE2.9a - stato dell'arte e sulla ricerca effettuata nel campo del supporto alle decisioni, M6
 - Identificazione di correlazioni inattese
 - Identificazione di condizioni critiche, dashboard, soluzioni di firing

In questo deliverable lo stato dell'arte relativo ad algoritmi identificati in fase di specifica:

- A01: Algoritmi e strumenti di Riconciliazione, Attività: 2.2.1
- A02: Algoritmi e strumenti di Arricchimento, Attività: 2.2.1
- A08: Algoritmi di aggregazione / riconciliazione, Attività: 4.1.7

In questo scenario le aree di Data Ingestion prese in considerazione sono quelle relative a soluzioni e applicazioni per lo sviluppo del data ingestion:

- Data ingestion API verso le sorgenti,
- Architetture per il data ingestion e l'aggregazione,
- Smart City API stato dell'arte,
- Algoritmi di riconciliazione.

L'obiettivo di questo documento non è quello di presentare uno stato dell'arte completo ed esaustivo. Ma quello di presentare lo stato dell'arte attuale, mettendo in evidenza gli aspetti recenti della ricerca e dello sviluppo nei vari settori in modo da servire come punto di partenza per lo sviluppo delle ricerche specifiche.

2 Panoramica su tecnologie Data Ingestion

Negli ultimi anni sono stati molti i progetti in ambito Smart City nati dalla necessità stabilire metodologie, linee guida, prototipi per garantire l'interoperabilità e l'integrazione dei sistemi di gestione delle attività urbane e di mobilità, lo scambio e la pubblicazione di dati tipicamente messi a disposizione da Comuni, aree Metropolitane, Regioni, ma anche da enti privati che forniscono servizi ai cittadini. La grande complessità, varietà e dimensione dei dataset (Open e Private Data), la loro scarsa interoperabilità e la vasta gamma dei formati differenti in cui sono distribuiti, posiziona il problema della raccolta e aggregazione dei dati nell'ambito dei 'Big Data' [Bellini et al., 2013], [Vilajosana et al., 2013]. Per ridurre gli alti costi computazionali e fornire servizi di alta qualità, è necessario ottimizzare gli aspetti di interoperabilità ed integrazione dei sistemi [Caragliu et al., 2009], [Bellini et al., 2013]. Le maggiori aziende stanno proponendo soluzioni per rendere più intelligenti le città, concentrandosi su domini applicativi specifici, come IBM [IBM], [IBM2] sui servizi per i cittadini, le imprese, trasporti, comunicazioni, acqua ed energia; [Alcatel-Lucent] su governativo, educativo, salute, sicurezza, energia, trasporti e utility; CISCO su persone e dati [CISCO], ecc. La maggior parte di queste soluzioni presentano un'architettura multi-tier da 3 a 6 strati [Anthopoulos et al., 2014]. D'altra parte, il numero di livelli/strati è parzialmente rilevante per la trasformazione dei dati in valore per la città, e quindi in servizi per gli utenti, e in opportunità per le imprese e gli operatori interessati a creare servizi innovativi ed efficaci, sfruttando i dati della città [Filipponi, et al., 2010], [Domingo et al., 2013], [Chourabi et al., 2013]. Inoltre, i modelli per la valutazione del livello di intelligenza delle città non sono adatti a mettere in evidenza questi aspetti, poiché sono concentrati principalmente sul conteggio del numero di open data, del numero dei servizi intelligenti, e/o di soluzioni per il risparmio energetico [*Smart cities: Ranking*], [Shapiro 2006].

Le principali questioni tecniche relative alle soluzioni Smart City sono legate all'accesso ai dati, all'aggregazione, ragionamento, accesso e servizi interoperabili via Smart City API. L'obiettivo finale è servire gli utenti della città in maniera più intelligente e più efficiente, stimolando la loro partecipazione alle strategie della città. Pertanto, i dati raccolti sono utilizzati per facilitare la creazione di servizi intelligenti ed efficaci che sfruttano i dati della città e le informazioni che si possono dedurre, prime fra questi quelli della mobilità. La municipalità deve poter fornire un accesso a dati e ad alcuni servizi in modo flessibile. Ciò significa rendere efficace ed efficiente l'accesso ai dati con la loro semantica, fornendo un servizio, l'accesso a un pannello di controllo, e l'interoperabilità tramite sistemi intelligenti.

In Italia come nel mondo, i nostri comuni, le nostre città e amministrazioni pubbliche stanno pubblicando un'enorme quantità di dati aperti (open data). Questi possono essere grossolanamente aggregati per l'integrazione e l'accesso utilizzando soluzioni come CKAN [CKAN], OpenDataSoft [OpenDataSoft], ArcGIS e OpenData [ArcGIS]. Tali soluzioni per la raccolta di open data e la distribuzione sono adatti per semplici fonti di dati di indicizzazione / metadati, fornendo il supporto per i dati di navigazione e di *query*, sfruttando metadati descrittivi. In alcuni casi, essi forniscono l'accesso integrato ai data set, utilizzando strumenti di integrazione e di visualizzazione che offrono la possibilità di creare tabelle e grafici, come ad esempio distribuzioni e/o torte. Nei casi migliori, sono in grado di fornire l'accesso ai data set come Linked Data (LD), Linked Open Data (LOD), e anche tramite un endpoint RDF [RDF] per eseguire *query* SPARQL sui dati, o solo su metadati [SPARQL], sfruttando qualche ontologia. L'accesso al database RDF in termini di triple per la navigazione dati può essere eseguita utilizzando browser visuali come in [Bellini et al., 2014].

Nella maggior parte dei casi, i servizi Smart City che forniscono dati sono fortemente abilitati dalla disponibilità di dati privati, per esempio di proprietà degli operatori in città che lavorano in domini specifici: mobilità, energia, sanità, acqua, telecomunicazioni, turismo, università, ecc. Questi soggetti forniscono dati e servizi con diverse granularità e dimensioni. Ad esempio, in città, possiamo avere alcuni operatori del settore energetico con una distribuzione capillare casa per casa,

mentre gli operatori del trasporto pubblico hanno migliaia di veicoli/bus, e gli operatori di telecomunicazioni possono dispiegare in città da decine a centinaia di migliaia o milioni di sensori. Un diverso grado di granularità implica metodi diversi per la raccolta dati e per fornire l'accesso ai dati, come la pubblicazione di file di dati aperti e/o statistiche, la pubblicazione dei dati in tempo reale con i consumi, il flusso di mobilità, energia, tempo, ecc. Dati in tempo reale sono forniti da operatori della città attraverso alcune API Web Services o chiamate REST.

3 Data Ingestion API verso le sorgenti

Le **API e protocolli per fornire dati al motore di aggregatore dati** della città devono essere compatibili con standard esistenti – in modo da poter garantire l'interoperabilità - come:

- DATEX II protocolli standard per la mobilità, sistemi di trasporto intelligenti per i servizi pubblici, parcheggi, eventi di traffico da sensori;
- IETF [IETF] protocolli standard per l'Internet delle cose (IoT);
- Green Button Connect [GBC] protocolli standard per la raccolta dei dati di energia;
- ETSI, European Telecommunications Standards Institute (<http://www.etsi.org/>) che offre protocolli standard per: (i) la raccolta dati provenienti da dispositivi distribuiti sul territorio in ottica IoT (principalmente short range device); (ii) il report di eventi di traffico;
- OGC, Open Geospatial Consortium (<http://www.opengeospatial.org/>), un consorzio per la definizione di standard per applicazioni geospaziali, <http://www.opengeospatial.org/docs/is> CityGML is an open data model and XML-based format for the storage and exchange of virtual 3D city models. Fra questi anche OGC KML, OpenGIS Tracking Service Interface Standard, netCDF (Network Common Data Form) data model for array-oriented scientific data, open data model and XML schema for indoor spatial information, Sensor Model Language (SensorML), OpenGIS® Sensor Planning Service Interface Standard (SPS) defines interfaces for queries, etc.;
- CEN, European Committee for Standardization, www.cenelec.eu, www.cen.eu;
- CEN e ETSI hanno prodotto uno standard per Cooperative Intelligent Transport System, C-ITS, <https://www.cen.eu/news/brief-news/Pages/News-2014-002.aspx>
- Action Plan for deployment of ITS in Europe 2008: <http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:52008DC0886>
- EU transport: 2011 white paper, http://ec.europa.eu/transport/themes/strategies/2011_white_paper_en.htm
- The ITS Action Plan (COM (2008) 886) foresees the setup of a specific ITS collaboration platform to promote ITS initiatives in the area of urban mobility (Action 6.4) ITS Action Plan and Directive http://ec.europa.eu/transport/its/road/action_plan_en.htm
- Urban ITS http://ec.europa.eu/transport/themes/its/road/action_plan/its_for_urban_areas_en.htm http://ec.europa.eu/transport/themes/its/road/action_plan/its_for_urban_areas_en.htm
- Draft standardisation request publicly notified in April 2015 http://ec.europa.eu/growth/single-market/european-standards/notification-system/index_en.htm
- <http://www.polisnetwork.eu/uploads/Modules/PublicDocuments/mandate-for-standardisation-urban-its---ump-stakeholders-informal-meeting---2oct2015.pdf>
- EU standards http://ec.europa.eu/growth/single-market/european-standards/index_en.htm

D'altra parte, alcuni dei dati raccolti, prodotti da operatori della città e necessari alle Smart City, non sono supportati da standard, e quindi vengono adottate soluzioni personalizzate. Per esempio, per gestire lo stato di emergenza degli ospedali, lo stato di terremoti nella regione Lombardia, ecc., è stata sviluppata la soluzione E015, un ecosistema digitale con il fine di far fronte a una grande quantità di API provenienti da fonti differenti. In questo caso è stato fatto uno sforzo per la

creazione di un ambiente per la raccolta di dati disponibili tramite servizi / API e interfacce su tutto il territorio, utilizzando un semplice portale web che aggrega la documentazione [EO15]. Questo tipo di soluzione sta risolvendo solo una piccola parte dei problemi di interoperabilità di dati e servizi. E non risolve i problemi che gli sviluppatori riscontrano a sviluppare App che devono accedere a molte API per realizzare una sola applicazione innovativa, e che pertanto si devono cimentare con protocolli multipli, certificati multipli, che non sono né aggregati né semanticamente uniformi.

L'implementazione efficace dei servizi intelligenti per gli utenti città è molto spesso possibile solo sfruttando l'integrazione semantica dei dati aperti, dati privati e dati in tempo reale provenienti da amministrazioni e operatori diversi. Questo implica specifici processi di riconciliazione e l'adozione di modelli di dati unificanti e ontologie come in Km4City [Bellini et al., 2014b]. L'aggregazione semantica dei dati provenienti da diversi domini è irrealizzabile senza una ontologia comune, poiché i dati sono prodotti da diverse istituzioni / aziende, utilizzando diversi formati e finalità, diversi riferimenti a elementi geografici, e diversi standard e prodotti in diversi momenti. Così, i vari insiemi di dati sono raramente semanticamente interoperabili. Questo problema diventa esponenzialmente ingestibile quando i data set da integrare sono decine.

I dati sulla città possono presentare diversi modelli di licenza: alcuni di questi possono essere liberamente accessibili (aperti, open), mentre altri possono essere di proprietà di qualche operatore della città (cioè privati). I proprietari dei dati privati non sono interessati a rilasciarli in un ambiente non regolamentato, o con solo alcune restrizioni (ad esempio, non commerciale) che non li proteggono a sufficienza. Si vedano ad esempio i dati delle compagnie di car sharing, che sono in genere privati, della società. Per i dati aperti, così come per i dati privati vengono adottati diversi modelli di licenza [Korn et al., 2011], [Villata et al., 2011] che permettono o impediscono alcuni modelli di business, o semplicemente il loro utilizzo.

Pertanto, dati aggregati e ben riconciliati (aperti e privati) possono essere sfruttati da algoritmi opportuni per la produzione di servizi smart [Bellini et al., 2014b]: ad esempio, algoritmi di *routing* modale e multimodale e intermodalità per facilitare il parcheggio, per la realizzazione di assistenti personali sulla base del profilo di utente, per la produzione di suggerimenti e lo stimolo di comportamenti virtuosi.

SMART TUNNEL (PON R&C, in ambito Smart City 2012-2015, Salerno) è il progetto che ha sviluppato una piattaforma tecnologica modulare di supporto alla filiera logistica portuale e del trasporto su strada delle merci basata sull'integrazione di sistemi gestionali e amministrativi e di sistemi di comunicazione. Il progetto ha messo a disposizione: i) linee guida, ii) prototipo di un sistema di servizi intelligente, aperto e scalabile per la gestione ottimale dei nodi logistici (portuali e urbani) realizzato con lo scopo di supportare la partecipazione, la collaborazione e l'interoperabilità tra i vari attori della Comunità Portuale e l'accessibilità ai dati [SMTUN].

A supporto della mobilità è necessario fornire informazioni su ciò che è intorno ad una certa posizione GPS, lungo una certa traiettoria, in una certa area. A questo fine è necessaria l'integrazione di informazioni geografiche e l'integrazione di servizi geolocalizzati. La valutazione dei flussi delle persone, intesi come spostamenti possono aiutare la città a migliorare i servizi pubblici e di trasporto, fornendo suggerimenti agli utenti, e agli operatori per la progettazione della città stessa. Pertanto i dati aggregati possono essere sfruttati per implementare un gran numero di servizi e applicazioni strutturando l'Architettura Smart City e le API Smart City corrispondenti.

4 Architetture per il Data Ingestion e Data Aggregation

Per supportare l'interoperabilità, diversi tipi di architetture Smart City possono essere adottate con l'obiettivo di costruire servizi e applicazioni *smart*. Le diverse architetture forniscono diverse Smart City API e differiscono fra loro dalla strategia per trasformare i dati ai servizi per la città (dai dati alla struttura delle API), come rappresentato in Figura 1, in cui sono rappresentati tre approcci principali: a, b, e c.

Anche se vi possono essere soluzioni ibride/miste le soluzioni prese in considerazione rappresentano i casi principali, vale la pena notare che, quando si parla di Web e Mobile App, intendiamo applicazioni che sfruttano l'accesso a dati per mettere servizi *smart* nelle mani degli utenti finali (utenti di città) e/o degli Operatori della città. Inoltre, negli esempi che seguono, i decisori sono le persone chiave delle amministrazioni pubbliche o degli operatori della città interessati a estrarre dai dati aggregati e integrati eventuali deduzioni, dati e/o avvisi/allarmi, notifiche, suggerimenti, etc.

Caso (a): l'InfoIntegrator della figura seguente raccoglie le informazioni sulle API e/o dati forniti da diversi fornitori di servizi (compresa la loro autenticazione e autorizzazione), e fornisce un luogo comune per sviluppatori e operatori della città per navigare e scoprire come accedere ai servizi API esposti e ai dati. Fornitori di dati/servizi, come operatori di: mobilità, energia, rifiuti, acqua, turismo, etc. Essi possono fornire alcuni dati open, altri privati, sia statici che in tempo reale. Nel caso (a), i dati/servizi non sono integrati tra loro, ogni set di API permette di accedere a dati/servizi di un unico operatore specifico. Pertanto, API e dati risultano tipicamente semanticamente non interoperabili, e i problemi dell'integrazione semantica dei dati e dei servizi sono lasciati nelle mani degli sviluppatori, che devono far fronte a diverse API di fornitori non armonizzati e ovviamente di diversi formati e di diverse modalità di autenticazione/certificazione per accedere ai dati. Gli sviluppatori devono selezionare i dati, ed integrarli ogni volta che questi cambiano. Questo implica anche che gli sviluppatori devono firmare accordi e contratti con ogni fornitore di dati e servizi.

Un esempio di questo caso (a) è la soluzione E015 per Milano [E015], in cui viene richiesto ai fornitori di API/dati di produrre informazioni strutturate attraverso dei file di Excel. Questi, con i formati, vengono pubblicati sul portale web E015, che non fornisce strumenti di ricerca (sulle descrizioni delle API), né di verifica, né di integrazione e ricerca sui dati. Questo approccio mira a condividere le informazioni ed i protocolli e potrebbe portare a regolarizzare le API degli operatori della città (anche se questo obiettivo non è perseguito), o agevolare la produzione di servizi integrati per gli utenti della città. I glossari sono definiti tramite un xml-schema e descrivono l'impostazione complessiva, le entità gestite e gli attributi che caratterizzano un servizio (la struttura del glossario NON è Open, viene concessa solo a chi aderisce al sistema) e classificano servizi e informazioni in base alle seguenti macro-categorie: Attività commerciali e Promozionali, Eventi del territorio, Itinerari del territorio, Punti di Interesse del territorio, TMC (Traffic Message Channel, un database in cui sono presenti tabelle che descrivono una rete stradale, tramite un insieme di punti e strade: un punto del TMC corrisponde ad un punto ben definito sulla rete stradale, rispetto al quale vengono riferiti gli eventi che accadono sulle strade vicine. Maggiore è il numero di punti e strade inserite nel TMC, maggiore è il dettaglio con cui possono essere fornite le informazioni sulla mobilità).

Smart City

Grazie ad E015, i soggetti partecipanti hanno realizzato soluzioni integrate di diversa tipologia, quali siti web innovativi, applicazioni per smartphone, totem multimediali e sistemi IoT. Sono state effettuate molte sperimentazioni in occasione dell'evento EXPO 2015, prevalentemente in ambito mobilità tramite: applicazioni mobile, portali web (Trenitalia), monitor stazioni ferroviarie (24 monitor tra Linate, Malpensa), pannelli informativi a bordo treno relativi allo stato dei voli, Pannelli a Messaggio Variabile (segnalazione di parcheggi di interscambio su PMV tangenziale Milano), totem informativi interattivi (nella città di Milano con info su aeroporti di Malpensa e Linate, parcheggi di interscambio, Malpensa express e Frecciarossa per le tratte di collegamento aeroporti-

città), App per la mobilità a Milano (stesse info dei totem e flussi traffico e mobilità). Per quanto riguarda i dataset messi a disposizione dal Comune di Milano, [MiOD], essi riguardano: calendario degli eventi cittadini (ITA/EN), POI (luoghi di interesse storico/artistico/culturale), Itinerari (collegati ai POI, ITA/EN).

L'approccio che consente il passaggio da "dati a servizi" attraverso un processo di aggregazione è estremamente complesso, dal momento che i dati non sono interoperabili, e l'accordo di licenza è in molteplici relazioni disposti tra ogni sviluppatore e diversi fornitori di dati. Le soluzioni appartenenti a questa categoria non soddisfano la maggior parte dei sotto-obiettivi del documento EIP sulle piattaforme urbane [EIP-SCC].

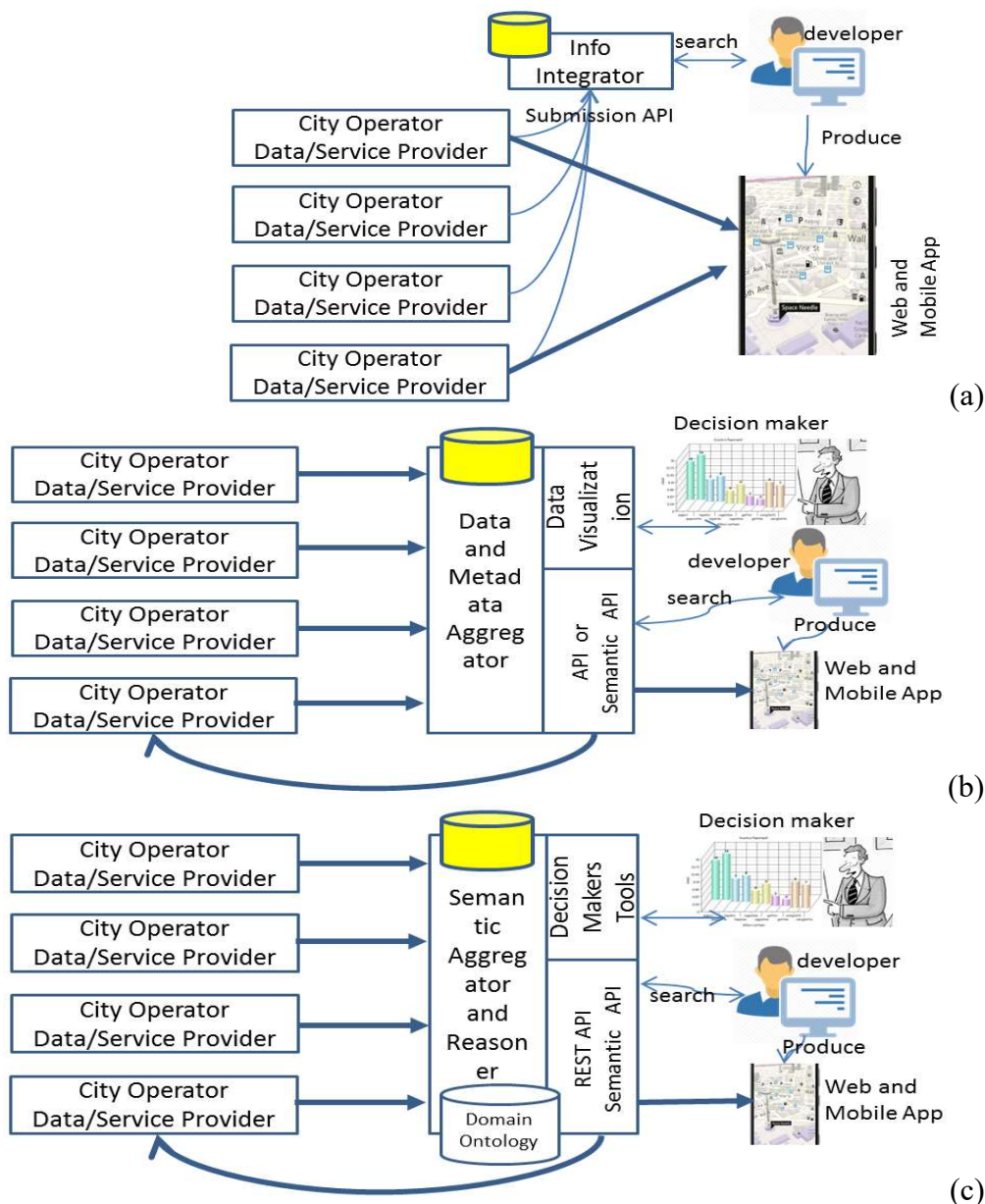


Figura 1: Da dati ai servizi, Architetture Smart City: confronto tra diverse soluzioni e generazioni. Le linee più spesse portano flussi di dati e servizi dati.

Caso (b): Il Data and Metadata Aggregator di Figura 1b raccoglie i dati e i relativi metadati per indicizzarli (principalmente dati aperti), e aggrega i dati in un modello comune in base alla struttura dei file di dati aperti in ingresso. I dati aggregati risultanti vengono resi accessibili per applicazioni

web e mobili attraverso API generate automaticamente, trascurando la semantica dei singoli data set e quindi l'interoperabilità ed i collegamenti/equivalenze tra informazioni raccolte da diversi data set, che magari fanno riferimento agli stessi elementi con attributi diversi, complementari o anche in conflitto. In alcuni casi, queste soluzioni presentano degli strumenti di visualizzazione grafica dei dati, sempre con gli stessi problemi di interoperabilità semantica. Tipicamente il modello prodotto automaticamente non presenta un servizio soddisfacente dal punto di vista della semantica interoperabile dati [Bellini et al., 2014b]. I dati non sono riconciliati tra loro, e mantengono la stessa qualità del dato originale, i record con dati mancanti sono ancora ovviamente mancanti. L'integrazione dei dati viene effettuata principalmente sulla base di forme sintattiche e lessicali di nomi dati. Il database risultante è tipicamente una serie di tabelle con funzionalità tradizionali SQL. Da questo, le API fornite non sono supportate da un'ontologia di dominio che descrive relazioni e semantica concettuale. In alcuni casi, viene fornito un modello semantico che descrive l'accesso ai dati come strategia di marketing per poter dichiarare di avere data a 5 stelle, fornendo API basate su SPARQL. Il modello ontologico per tali soluzioni viene generato in modo automatico ed è una mera rappresentazione di tabelle, non fornendo un significativo sostegno inferenziale. Ciò significa che su tali dati aggregati non si possono fare ragionamenti nel tempo e nello spazio, o almeno si hanno grosse limitazioni. Appartengono a questa categoria di soluzioni CKAN [CKAN], ArcGIS OpenData [ARCGIS], OpenDataSoft [OpenDataSoft] sulla base di ArcGIS, SOCRATA [SOCRATA] anche sulla base di ArcGIS. Le soluzioni basate su ArcGIS forniscono più funzionalità sulle query geospaziali. L'approccio conforme al Caso (b) può essere considerato come una soluzione di primo livello per l'interoperabilità dei dati, e può essere valido quando i dati da aggregare sono principalmente open, senza dati in tempo e/o dati con licenze particolari. Le soluzioni conformi al caso (b) non coprono tutti i sotto-obiettivi delle piattaforme urbane [EIP-SCC] in quanto i dati non sono armonizzati. Quando i dati aperti devono essere integrati con i dati privati e dati in tempo reale, come nei casi di servizi di mobilità urbana, la soluzione deve affrontare un certo numero di casi speciali che devono essere trattati separatamente e manualmente, per integrare tali dati con il resto dei dati statici e dinamici presenti, distruggendo i vantaggi dell'integrazione. Una soluzione a questi problemi è l'adozione di una soluzione Case (c) come descritto in seguito.

Caso (c): il Semantic Aggregatore and Reasoner (si veda Figura 1c) raccoglie i dati e servizi da parte degli operatori città, per aggregarli e integrarli in un modello unificato e semanticamente interoperabile sulla base di un'ontologia multi-dominio. Questo approccio consente di riconciliare i dati e sfruttare un modello coerente che permette di ridurre gli errori, di operare un'integrazione fra i dati che rappresentano lo stesso concetto e che provengono da data set diversi. L'utilizzo di un'ontologia multi-dominio permette l'adozione di un modello che rappresenta le relazioni di specializzazione tra le classi, di aggregazione, di associazione e similarità, e che abilitano/permittono i processi inferenziali nei database RDF Graph [Bellini et al., 2015], [Bellini et al., 2014b]. La base di conoscenza prodotta dalle riconciliazioni può essere utilizzata per la creazione di strategie per il miglioramento della qualità dei dati e per la creazione di algoritmi che traggono vantaggio dalla composizione di informazioni che provengono da più domini: mobilità e servizi, mobilità ed energia, mobilità e informazioni sulla persona. Questo vantaggio è evidente se accessibile dalle Smart City API e se può essere sfruttato dagli strumenti di supporto alle decisioni e per lo stimolo verso i *city-user*. Per lo stesso motivo, ottenuta una Knowledge Base, popolando l'ontologia con dati e deduzioni, può essere proficuamente e facilmente utilizzata per la produzione di servizi intelligenti come il routing, routing multimodale, suggerimenti su richiesta, assistenti personali, guida connessa, ecc.

Come descritto in seguito, sono coerenti parzialmente con le caratteristiche del Caso (c):

- **CitySDK** [CitySDK], che prevede alcune limitazioni ed è stato sviluppato in un progetto della Commissione Europea che coinvolge le principali città e di fornire specifiche API REST;
- **OASC** (Open & Agile Città intelligenti), che ha adottato il modello di API agnostica FIWARE NGSI [OASC] per la produzione di API Smart City sulla base CitySDK con le limitazioni corrispondenti.

Le caratteristiche del Caso (c) sono più ampiamente coperte da Km4City [Bellini et al., 2014b], [ServiceMap] adottato da altri progetti europei come RESOLUTE H2020 e REPLCIATE H2020.

Le soluzioni conformi al Caso (c) differiscono principalmente da quelle (b) per la presenza di un vero e proprio modello ontologico tra entità e concetti della città e non solo sui dati strutturali delle tabelle. Tali soluzioni (c) sono maggiormente conformi rispetto ai sotto-obiettivi delle Urban Platform della Commissione Europea [EIP-SCC] che si aspettano soluzioni in grado di coprire aspetti legati all'armonizzazione dei dati, e alla produzione di servizi intelligenti (nel nostro caso collegati alla mobilità ma non solo). La realizzazione di servizi che coinvolgono ed ingaggiano l'utente per stimolarlo in vario modo anche con servizi a valore aggiunto (sotto-obiettivo 5) è possibile solo in alcune soluzioni, come analizzato in seguito. L'ontologia può modellare domini multipli sulle entità della città e le loro relazioni, e non solo i metadati del set di dati e tabelle, come nel caso (b). Un'integrazione efficace a livello semantico del dominio dei dati consente la creazione di sistemi di supporto decisioni e algoritmi intelligenti che sfruttano la possibilità di fare ricerche semantiche su più domini, per: effettuare ragionamenti probabilistici sul supporto decisionale Bayesiano [Bartolozzi et al., 2015], per consentire la produzione di algoritmi per attuazione di routing personalizzato e assistenti personali in città, per l'evoluzione di stato del comportamento utente. Nel Caso (c) le soluzioni devono fare i conti con le prestazioni dei database a grafo (RDF Store) che raccolgono un'enorme quantità di dati. Si aprono pertanto scenari Big Data che presentano sui dati volumi rilevanti, grandi varietà, velocità anche non eccessiva, ma che necessitano grosse capacità di calcolo per le deduzioni [Bellini et al., 2013].

La Tabella seguente riassume l'analisi comparativa delle diverse architetture e soluzioni API Smart City, per consentire alla città di passare dai dati ai servizi, quindi dai dati alle applicazioni per gli utenti della città e per i decisori. Come considerazione finale, la differenza principale che può essere percepita dagli utenti della città e dai decisori è il numero di servizi intelligenti e trasversali che traggono vantaggio da più domini come nella capacità che distingue il Caso (c) dagli altri.

La categoria Case (c) della Smart City rappresenta la nuova generazione di soluzioni, ed è la direzione di sviluppo più interessante per il progetto Sii-Mobility visto che promette maggiori vantaggi rispetto agli altri casi.

I sistemi di trasporto terrestri presentano scenari che sono sempre più spesso colpiti da situazioni di congestione a causa del fatto che le situazioni specifiche non possono essere facilmente previste utilizzando i dati classici dei sistemi di trasporto intelligenti tradizionali (ITS), come per esempio: flussi, tempistiche e code ai semafori, storico dei tempi di percorrenza, etc., cioè solo dati relativi alla mobilità e non al contesto, all'energia, al comportamento delle persone in relazione agli eventi in città, al meteo, all'ambiente, al giorno della settimana, etc. Il sistema di trasporto locale presenta costi molto elevati e connessi disagi per i cittadini rispetto alle soluzioni di mobilità disponibili. Ciò è dovuto principalmente alla scarsa interoperabilità e alla scarsa intelligenza tra i sistemi di gestione e di trasporto di monitoraggio, i servizi per la mobilità, servizi e sistemi per il trasporto merci, ordinanze e servizi pubblici (come ad esempio ospedali, centri, musei), eventi, trasporti privati, il trasporto ferroviario, parcheggi e spostamento di persone, a causa della limitata capacità del sistema di incorporare e reagire alle variazioni della città e cittadini.

Tabella 1: Confronto di diverse soluzioni Smart City per passare da dati a servizi.

(Y) significa sì ma con capacità limitate, spesso a causa della limitata ontologia nel rappresentare gli aspetti di entità di dati della città in: tempo, spazio, più domini, strutture, servizi e relazioni

	Case (a) Info Integrator	Case (b) Data and Metadata Aggregator	Case (c) Semantic Aggregator and Reasoner
Addressing Open Data	Y	Y	Y
Addressing Private Data	Y	Y	Y
Addressing Real Time Data	Y	Y	Y
Addressing Services	Y	N	Y
Providing Data Search	N	Y	Y
Providing Metadata Search	Y	Y	Y
Providing Space Reasoning	N	(Y)	Y
Providing Time Reasoning	N	(Y)	Y
Providing Integrated Authenticated Access to data	Only metadata	Y	Y
Providing Syntactic Interoperable Data/Services	N	Y	Y
Providing Semantic Interoperable data/Services	N	N	Y
Independent from the Data model changes	N	N	Y
Providing REST API on data	N	Y	Y
Providing SPARQL API on data	N	(Y)	Y
Providing inference support on Data	N	N	Y
Providing Data Visualization	N	Y	Y
Providing Decision Maker Support	N	(Y)	Y

Smart City

5 Smart City API stato dell'arte

Le principali funzionalità delle Smart City API per gli aggregatori dati di tipo (c) possono essere classificate in pochi gruppi che includono funzionalità generali e specifiche strettamente connesse con il dominio (di ricerca, la mobilità, l'ambiente, la partecipazione degli utenti e la consapevolezza, intelligente interoperabilità della città, ecc.).

CitySDK è un kit di sviluppo per le città e gli sviluppatori che mira ad armonizzare le API attraverso le città [CitySDK]. L'architettura SOA di CitySDK è tipicamente strutturata su database SQL. CitySDK fa trasformazioni su ogni data set al fine di ottenere e gestire dati uniformi. Non ci sono relazioni semantiche tra i dati raccolti dai fornitori di dati; solo alcuni dei legami possono essere facilmente stabiliti, per esempio tra un evento e il punto di interesse a cui si riferisce. Il lavoro in materia d'interoperabilità semantica del dato è limitata a livello di API, e la redazione di linee guida e standard per i fornitori di dati e gli utenti API. Il progetto ha coinvolto 8 città europee: Helsinki, Barcellona, Amsterdam, Manchester, Lamia, Istanbul, Lisbona e Roma. Tali città hanno collaborato per la realizzazione di una interfaccia web unificata e la messa a disposizione di un toolkit per lo sviluppo di servizi digitali, che comprende interfacce aperte e interoperabili, linee guida e standard di usabilità.

ECIM (European Cloud Marketplace for Intelligent Mobility) <http://ecim-iminds.rhcloud.com>, (FP7 CIP-ICT-PSP-2013-2017) è una soluzione derivata dal modello EPIC [ECIM], [EPIC], ed è stata sviluppata per coprire le esigenze delle amministrazioni pubbliche e delle piccole imprese per i cittadini. La soluzione ha come utenti finali: le Pubbliche Amministrazioni (mobilità), le piccole imprese (accesso a servizi e strumenti per creare nuove opportunità di lavoro) e i cittadini (possibilità di fare uso di servizi di trasporto efficienti usando il proprio smartphone). L'approccio

ECIM aggrega servizi pubblici e privati per la creazione e la validazione di nuovi. È stato pilotato su Bruxelles, Parigi, Barcellona, nell'ambito di un progetto finanziato dalla Commissione Europea. In questo caso, non vi sono descrittori semantici relativi ai servizi offerti, né ai set di dati. Le informazioni provenienti da fornitori di dati sono in genere convertite dal loro formato nativo a JSON, e inserite in un database comune MySQL. La soluzione ECIM differisce dalla soluzione E015 [EO15] poiché ECIM opera un'integrazione API REST parziale <http://platform.ecim-cities.eu>. I servizi offerti possono essere aperti o al pagamento. Il progetto ECIM si concentra sulla mobilità, e le azioni di partecipazione sono previste con lo scopo di creare nuovi servizi che soddisfano le esigenze dei cittadini.

Transport.API [Transport.API] <http://www.transportapi.com> è un servizio per la fornitura di dati aperti aggregati in UK. Questo servizio mette a disposizione, tramite un sistema di API REST un rilevante numero di data set che integrano dati statici e in tempo reale, soprattutto per quanto riguarda gli aspetti di mobilità. Per questa ragione, può essere parzialmente classificata come parziale soluzione di tipo (c). I suoi limiti principali sono legati alla bassa integrazione con altri dati del dominio e con il modello di licenza fornito. Essendo un servizio nato solo per profitto è inadatto ad essere adottato dalle pubbliche amministrazioni che intendono adottare servizi Smart City.

Navitia.io è un progetto open source sfruttando **OpenDataSoft** per l'aggregazione dei dati aperti in Francia [Navitia.io]. L'interoperabilità Navitia.io è a livello di API; i set di dati sono raccolti da vari domini, aggregate in un unico database, classificati per grandi categorie (una tassonomia applicata ai metadati che descrivono ogni set di dati). Non esiste un modello semantico uniforme che descrive i dati e di interconnessione delle serie di dati stessi: meccanismi di riconciliazione o di miglioramento della qualità sono completamente mancanti. Per queste ragioni, può essere parzialmente classificata come parziale soluzione (c). Inoltre, il servizio non è ospitato su un server potente, o non è scalabile. Gli utenti sono invitati a limitare le loro richieste: al massimo una al secondo. Il progetto ha sviluppato un sistema di API Rest (sul modello HATEOAS) e i dati sono restituiti nei formati: JSON, csv, Excel.

KISIO digital, <http://www.canaltp.fr>, è un'azienda che offre alcuni servizi in modalità Open Source (sotto specifiche licenze d'uso: <http://www.canaltp.fr/Open>), mentre altri (con maggiori funzionalità) sono a pagamento. In generale i dataset gestiti e messi a disposizione tramite API sono numerosi, e si situano prevalentemente nell'ambito della mobilità, con la possibilità di collegare anche dataset relativi ad eventi e POI delle varie città. Inoltre sono in grado di calcolare itinerari in base alle informazioni in tempo reale sulla mobilità (ritardi dei mezzi pubblici, user profiling, vincoli, etc.).

FiWare IOT Management, <https://www.fiware.org> è una suite di componenti software standardizzato. Il progetto ha lo scopo di essere utilizzato come base per creare applicazioni e servizi avanzati. È uno dei risultati del partenariato pubblico-privato dell'UE su "Future Internet". FIWARE fornisce una serie di funzionalità e librerie di base in modalità Open Source Software. Questi componenti, denominati "Generic Enablers", forniscono una serie di *API (Application Programming Interface) che per connettersi a dispositivi IoT (Internet of Things), dati di processo e mezzi di comunicazione in tempo reale su grande scala, consentono di eseguire analisi su grandi quantità di dati o incorporare funzionalità avanzate per permettere agli sviluppatori (e agli utilizzatori finali delle App) di interagire con i componenti gestiti (e.g. sensori, attuatori, etc.). Nel dettaglio:

- Tutte le risorse IoT, sono registrate e trattate nel Sistema FiWare come entità NGSI (Next Generation Service Interfaces). Gli sviluppatori devono quindi conoscere solo le API NGSI (REST).

- L'architettura IoT supporta l'attuazione di due scenari principali: i) connessione di pochi dispositivi, con la messa a disposizione di protocolli semplici; ii) gestione di reti di oggetti, che necessitano invece di protocolli più complessi e strutturati.
- Gli IoT GEs si classificano in: IoT Backend e IoT Edge.

Nel dettaglio, l'architettura sviluppata da FiWare per la gestione dei sensori è caratterizzata dalle seguenti componenti:

<https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/FIWARE.ArchitectureDescription.IoT.Backend.DeviceManagement>

I protocolli usati da Fiware per connettere e gestire i sensori al Sistema back-end (si tratta dei protocolli degli IoT Agents, si veda fig. Architettura generale) sono i seguenti:

- UL2.0/HTTP (Api REST UL2.0) Ultralight2.0: è una specifica per la semantica relativa ai dispositivi. IoT Agent che gestisce dispositivi tramite connessioni HTTP/IPv4;
- UL2.0/MQTT: IoT Agent che gestisce dispositivi tramite il protocollo MQTT;
- LWM2M/CoAP: IoT Agent che gestisce dispositivi in accordo alle specifiche ETSIM2M e OneM2M tramite il protocollo REST IETF CoAP;
- Generic IoT Agent: IoT Agent generico, creato per far fronte ad eventuali nuovi standard e/o protocolli.

6 Algoritmi per la riconciliazione

Nonostante il gran lavoro svolto dalle Pubbliche Amministrazioni (PA) sulla produzione di dati aperti, questi non sono in genere semanticamente interoperabili tra loro, né con i molti dati privati disponibili in città. I dati aperti della PA contengono tipicamente informazioni statistiche sulla città (ad esempio i dati sulla popolazione, incidenti, alluvioni, voti, amministrativi contabili, consumo di energia, presenze in musei, ecc.), la posizione e la descrizione di punti di interesse sul territorio (tra cui, musei, attrazioni turistiche, ristoranti, negozi, alberghi, ecc.), i principali servizi GOV, dati ambientali, lo stato del tempo e previsioni, cambiamenti nelle regole del traffico per gli interventi di manutenzione, ecc. Inoltre, un ruolo rilevante nella città è coperto dai dati privati provenienti da mobilità e trasporti, come quelli creati dagli Intelligent Transportation Systems (ITS), per la gestione del bus, e soluzioni per la gestione e il controllo di aree di parcheggio, auto e bike sharing, il flusso di auto in generale, servizi che forniscono accessi a traffico limitato Zone, ZTL, ecc.

I dati aperti e privati possono includere dati in tempo reale, come la misura del flusso di traffico (e l'indicazione sui Level of Service), la posizione dei veicoli (autobus, car sharing, moto, taxi, netturbini, ecc.), la posizione delle stazioni, gli orari e lo stato dei treni per quanto riguarda l'arrivo, stato di aree tramite sistemi di monitoraggio Bluetooth per il monitoraggio dei movimenti di persone o auto tramite i loro telefoni cellulari o client Bluetooth, sensori ambientali, telecamere per la valutazione di flussi e per la sicurezza.

Le PA come gli operatori della mobilità hanno grandi difficoltà nell'elaborazione e nell'aggregazione di tali dati per fornire nuovi servizi, anche se questi possono avere una forte rilevanza nel miglioramento della qualità della vita e dei servizi dei cittadini. Pertanto, le nostre città non sono così intelligenti come potrebbero esserlo sfruttando una base di conoscenza semanticamente interoperabile sui dati disponibili.

La situazione dei dati e la loro complessità è una condizione presente anche in città sopra i 100.000 abitanti attive sulla pubblicazione di dati aperti, come Firenze. Firenze in Italia è considerata uno delle migliori città per la produzione di dati aperti ed è ad un ottimo livello anche se consideriamo il livello Europeo.

Pertanto, la variabilità, la complessità, la varietà e la dimensione di questi dati rendono il processo di data ingestion e aggregazione complesso, rientrando nei problemi di Big Data affrontati in

[Bellini et al., 2013], [Vilajosana et al., 2013]. La varietà e la variabilità dei dati è dovuta alla presenza di diversi formati e di scarsa (o non esistente-) interoperabilità semantica dei singoli campi e delle diverse serie di dati. Al fine di ridurre il costo di ingestione e integrazione, ottimizzando servizi e gestione delle informazioni integrata a livello di qualità necessario, è richiesta una interoperabilità migliore e integrazione tra sistemi [Caragliu et al., 2009], [Bellini et al., 2013]. Questo problema può essere parzialmente risolto utilizzando processi di riconciliazione specifici per rendere questi dati interoperabili con altri data set e con quelli già aggregati. La velocità dei dati è legata alla frequenza di aggiornamento dei dati, e permette di gestire in modo distinto sia dati statici che dinamici. I dati statici sono raramente aggiornati, ad esempio una volta al mese/anno, invece i dati dinamici vengono aggiornati da una volta al giorno a ogni minuto, fino ad arrivare a dati in tempo reale (*stream*), che comunque hanno sempre un ritardo di trasmissione.

Quando questi modelli di dati vengono analizzati e poi elaborati per diventare semanticamente interoperabili, possono essere utilizzati per creare una base di conoscenza integrata che può essere alimentata da istanze di dati corrispondenti (con dati statici, quasi-statici e in tempo reale). D'altra parte, questo approccio non risolve il problema poiché istanze diverse che si riferiscono ad una stessa entità possono non essere interoperabili da permettere di collegarli insieme. Ad esempio, i nomi delle strade provenienti da due diverse fonti (data set) possono identificare fisicamente la stessa strada ma possono essere scritti in modo diverso creando un Semantic miss-Link. Questi problemi devono essere risolti con processi di riconciliazione.

Quanto descritto può essere risolto con la costruzione di un modello di conoscenza fino a creare con successo una grande base di conoscenza semanticamente interoperabile che può essere utilizzata per fornire servizi ad applicazioni di terze parti, della pubblica amministrazione e/o di imprese. Queste applicazioni possono sfruttare la base di conoscenza composta dai tanti data set, open e private data, statici e in tempo reale per effettuare interrogazioni e produrre servizi innovativi: per esempio la ricerca di servizi intorno ad un certo punto GPS, la ricerca di un'area in cui non sono disponibili ristoranti, l'individuazione e previsione di condizioni critiche, o anche dare la possibilità di effettuare suggerimenti per il miglioramento dei servizi al cittadino sulla base di dati statistici, deducendo causalità (relazioni di causa effetto). Questi servizi possono essere contestualizzati e utilizzati da diversi operatori in città, quali amministrazioni pubbliche, operatori della mobilità, e spot pubblicitari. Inoltre, applicazioni specifiche per allertare sulle condizioni meteorologiche e/o situazioni critiche stanno per essere prodotte per fornire un servizio agli amministratori pubblici.

Per collegare i servizi, i punti di interesse alle informazioni stradali nel *repository* si ha bisogno di un'azione di riconciliazione in più fasi. Per esempio, la notazione usata dalla Regione Toscana in qualche Open Data riferisce alla guida stradale, ma in generale non sempre i riferimenti coincidono con quelli utilizzati all'interno di Open Data in relazione ai diversi punti di interesse. In sostanza, le diverse amministrazioni pubbliche stanno pubblicando dati aperti che non sono semanticamente interoperabili.

I problemi tipici sui dati possono essere correlati a: (i) la bassa qualità dei dati, (ii) mancanza di dati che dovrebbero arrivare in tempo reale, (iii) cambiamenti nel modello dei dati set da una versione alla successiva, e nei dati stessi, (iv) modifiche e aggiornamenti nei set di dati (questo problema potrebbe generare un cambiamento nel modello ontologico e quindi si deve attivare un intervento umano per la revisione del modello), ecc. A tal fine, sono necessari processi di verifica e validazione periodici da eseguire mediante la definizione di un insieme di interrogazioni SQL/SPARQL sulla base di conoscenza al fine di individuare incongruenze e incompletezza, e verificando lo stato corretto del modello.

Queste query di verifica sono eseguite periodicamente, per eseguire un test di regressione ogni volta che viene eseguito un nuovo aggiornamento nel processo di ingestione dei dati, e quando i dati in tempo reale arrivano nello RDF store finale. Il processo di validazione può permettere di

identificare i problemi relativi alle classi ed alle istanze di classi. A tal fine, una quarta informazione associata a ciascuna tripla (facendola diventare una quadrupla) permette di individuare le problematiche e i processi che devono essere corretti.

Pertanto, a questo fine è stato definito un processo di *workflow* iterativo. Durante la validazione possono essere trovati casi, come quello delle previsioni del tempo, in cui non era presente un collegamento tra i dati a causa di una diversa codifica del nome del comune. Per questo motivo a sostegno del processo di riconciliazione deve essere creata una tabella che contenga il codice ISTAT di ciascun comune, che verrà utilizzato tutte le volte che i nuovi dati meteo sono disponibili. Tali dati verranno quindi automaticamente completati con il codice ISTAT corretto, sostenendo in tal modo la ricerca per l'istanza della classe PA a cui collegare le previsioni del tempo.

Un processo rilevante per il miglioramento dei dati e per l'interoperabilità semantica è relativo all'applicazione delle riconciliazioni tra le entità associate a posizioni come strade, numeri civici e località. A questo proposito, esistono diversi tipi di incongruenze all'interno dei vari set di dati integrati, quali:

- errori di battitura;
- Mancanza del numero civico, o la sostituzione con uno zero "0" o (acronimo italiana che significa senza numero civico) "SNC";
- Comuni senza un nome ufficiale (ad esempio Vicchio / Vicchio del Mugello);
- I nomi delle strade con caratteri non comuni (-, /, °, Ang,,?.);
- Numeri civici con caratteri strani (-, /, °, Ang, (,)?.);
- Nome della strada con le parole in un ordine diverso dal solito (ad esempio Via Petrarca Francesco, lo scambio di nome e cognome);
- Numero erroneamente scritto (ad esempio, 34 / AB, 403D, 36INT.1);
- Numeri civici rosso (in alcune città, numeri civici possono avere un colore in modo che una strada può avere 4 / nero e 4 / rosso, rosso è il sistema di numerazione per i negozi.);
- Numeri romani nel nome della strada (ad esempio, via Papa Giovanni XXIII).
- Etc.

Il processo di riconciliazione può essere eseguito con l'obiettivo di trovare elementi che identificano la stessa entità pur presentando differenti URI. Così le riconciliazioni identificate sono risolte creando una tripla owl:sameAs vero il toponimo (la posizione) selezionato. Il rilevamento della mancanza di una riconciliazione può essere effettuato effettuando (i) una serie di query SPARQL specifiche, (ii) strumenti per quello che viene chiamato RDF "link discovery". A tal fine, sono stati proposti linguaggi dichiarativi per il link discovery come SILK [Isele, Bizer, 2013] e LIMES [Ngomo, Auer, 2011]. Sia la produzione di query in SPARQL, sia la programmazione di algoritmi di link discovery implica la conoscenza della struttura ontologica dello store RDF da confrontare/analizzare.

6.1 SPARQL Reconciliation

La metodologia che viene tipicamente utilizzata per la riconciliazione SPARQL consiste nel cercare di collegare ogni servizio al numero civico, e quindi, eseguire la riconciliazione a livello stradale [Bellini et al., 2014b]. Il primo passo di riconciliazione consiste in una ricerca esatta del nome della via associati a ciascun servizio integrato. Ad esempio, per riconciliare il servizio che si trova in "via della Vigna Nuova 40 / R-42 / R, Firenze", è necessaria una query SPARQL, per la ricerca di tutti gli elementi della classe di strada collegata al comune di "FIRENZE" (attraverso il ObjectProperty inMunicipalityOf), che hanno un nome che corrisponde esattamente a "Via della Vigna Nuova" (controllando entrambi i campi: nome ufficiale, nome alternativo). Il risultato della query deve

essere filtrato di nuovo, imponendo che un'istanza della classe StreetNumber esiste e corrisponde al numero civico "40" o "42", con il codice della classe R Rosso. Un problema molto frequente per ricerca esatta è l'esistenza di molteplici modi per esprimere qualificatori come il toponimo (ad esempio Piazza e P.zza) o parti del nome proprio della strada (come Santa, o S. o S o S.ta): grazie al supporto di tabelle, all'interno del quale sono inseriti la possibile modifica della notazione per ogni singolo caso identificato, una seconda fase di riconciliazione è stata eseguita, in base alla ricerca esatta del nome della strada, che ha permesso di aumentare il numero di riconciliati servizi a numero civico livello. Il terzo passo di riconciliazione si basa sulla ricerca dell'ultima parola all'interno del campo v: Street-Indirizzo di ciascuna istanza della classe di servizio, perché, statisticamente, per un'alta percentuale di nomi di strade, questa parola è la chiave per identificare univocamente un incontro.

Le suddette tre fasi sono effettuate senza tener conto del numero civico, e quindi per ottenere una riconciliazione su un solo livello di ogni singolo servizio. Un'ulteriore fase di correzione può essere effettuata anche manualmente (i) alla ricerca di servizi e le posizioni incongruenti tramite il servizio di ricerca web come Google, (ii) per ripulire il numero civico, (iii) per l'accettazione e l'esecuzione di match non trovati tenendo conto della lista dei candidati probabili suggeriti dai risultati della query.

6.2 Link discovering Based Reconciliation and comparison

La riconciliazione tramite link discovering consiste nello scrivere algoritmi in linguaggio SILK specifici per la ricerca e la scoperta di link/URI. Questi algoritmi consentono di scoprire questi link tramite la stesura di algoritmi che vanno nella struttura del grafo e che utilizzano algoritmi specifici per il calcolo delle distanze tra elementi e grafi tramite metriche di similarità tra modelli e relazioni principalmente sulla base di string matching e misure di distanza (euclidea, modelli pesati, distanze degli alberi e/o patter, match di stringa, tassonomica, e algoritmi di distanza come quelli di Jaro, Jaro-Winkler, Leveisthein, Dice, Jaccard, etc.) [Isele, Bizer, 2013].

7 Bibliografia

- [Alcatel-Lucent] Alcatel-Lucent Market and Consumer Insight team, "Getting Smart about Smart Cities Understanding the market opportunity in the cities of tomorrow", Oct. 2013
- [Anthopoulos et al., 2014] Anthopoulos, Leonidas, and Panos Fitsilis. "Exploring architectural and organizational features in smart cities." Advanced Communication Technology (ICACT), 2014 16th Int. Conference on. IEEE, 2014.
- [ArcGIS] ArcGIS OpenData: <http://opendata.arcgis.com/>
- [Bartolozzi et al., 2015] M. Bartolozzi, P. Bellini, P. Nesi, G. Pantaleo and L. Santi, "A Smart Decision Support System for Smart City", IEEE International Conference on Smart City/SocialCom/SustainCom together with DataCom 2015 and SC2, December 2015, Cina, IEEE press. <http://smartds.disit.org>
- [Bellini et al., 2013] P. Bellini, M. Di Claudio, P. Nesi, N. Rauch, "Tassonomy and Review of Big Data Solutions Navigation", as Chapter 2 in "Big Data Computing", Ed. Rajendra Akerkar, Western Norway Research Institute, Norway, Chapman and Hall/CRC press, ISBN 978-1-46-657837-1, eBook: 978-1-46-657838-8, july 2013, pp.57-101, DOI: 10.1201/b16014-4
- [Bellini et al., 2014] P. Bellini, P. Nesi, A. Venturi, "Linked Open Graph: browsing multiple SPARQL entry points to build your own LOD views", <http://log.disit.org> International Journal of Visual Language and Computing, Elsevier, 2014, DOI information: <http://dx.doi.org/10.1016/j.jvlc.2014.10.003>
- [Bellini et al., 2014b] P. Bellini, M. Benigni, R. Billero, P. Nesi and N. Rauch, "Km4City Ontology Building vs Data Harvesting and Cleaning for Smart-city Services", International Journal of Visual Language and Computing, Elsevier, 2014, <http://dx.doi.org/10.1016/j.jvlc.2014.10.023>
- [Bellini et al., 2015] P. Bellini, I. Bruno, P. Nesi, N. Rauch, "Graph Databases Methodology and Tool Supporting Index/Store Versioning", publication on JVLIC, Journal of Visual Languages and Computing, Elsevier, 2015
<http://www.sciencedirect.com/science/article/pii/S1045926X15000750>
- [Caragliu et al., 2009] Andrea Caragliu, Chiara DelBo, Peter Nijkamp. "Smart cities in Europe", in 3rd Central European Conference in Regional Science – CERS, Kosice (sk), 7–9 Ottobre2009.
- [Chourabi et al., 2013] Chourabi, Hafedh, et al. "Understanding smart cities: An integrative framework." System Science (HICSS), 2012 45th Hawaii International Conference on. IEEE, 2012.
- [CISCO] CISCO http://www.cisco.com/c/dam/en_us/about/ac79/docs/ps/motm/Smart-City-Framework.pdf <http://www.cisco.com>,
http://www.cisco.com/c/dam/en_us/solutions/industries/docs/gov/everything-for-cities.pdf
- [CitySDK] CitySDK, <http://www.citysdk.eu/> , <http://www.citysdk.eu/about-the-project/deliverables-2/>
- [CKAN] CKAN: <http://ckan.org>
- [DATEX] DATEX II: http://www.datex2.eu/sites/www.datex2.eu/files/Datex_Brochure_2011.pdf
- [Domingo et al., 2013] Domingo, A.; Bellalta, B.; Palacin, M.; Oliver, M.; Almirall, E., "Public Open Sensor Data: Revolutionizing Smart Cities," in Technology and Society Magazine, IEEE , vol.32, no.4, pp.50-56, winter 2013, doi: 10.1109/MTS.2013.2286421
- [ECIM] ECIM, European Cloud Marketplace for Intelligent mobility, <http://ecim-iminds.rhcloud.com>, (FP7 CIP-ICT-PSP-2013-2017).
- [EIP-SCC] Requirements Specification For Urban Platforms, EIP Project, version 2.2, 2016, European Innovation Partnership for Smart Cities & Communities (EIP_SCC).
- [EO15] E015 digital ecosystem, <http://www.e015.expo2015.org/>
- [EPIC] EPIC, European Platform for Intelligent Cities, <http://www.epic-cities.eu>, ICT PSP (2011-2013)
- [ETSI] <http://www.etsi.org/technologies-clusters/technologies/internet-of-things>

- [Filipponi et al., 2010] Filipponi, L.; Vitaletti, A.; Landi, G.; Memeo, V.; Laura, G.; Pucci, P., "Smart City: An Event Driven Architecture for Monitoring Public Spaces with Heterogeneous Sensors," in Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on , vol., no., pp.281-286, 18-25 July 2010
- [GBC] Green Button Connect: <http://www.greenbuttonconnect.com/>
- [HOMER] <http://homerproject.eu/>
- [IBM] IBM Smart City <http://www-935.ibm.com/services/us/gbs/bus/html/smarter-cities.html>
- [IBM2] IBM Institute for Business Value, "How Smart is your city? Helping cities measure progress", [online]. Available: http://www.ibm.com/smarterplanet/global/files/uk_en_uk_cities_ibm_sp_pov_smartcity.pdf , Oct. 2013
- [IETF] IETF: <https://www.ietf.org>
- [Isele, Bizer, 2013] R. Isele, C. Bizer. "Active learning of expressive linkage rules using genetic programming". Web Semantics: Science, Services and Agents on the World Wide Web 23 (2013): pp.2-15.
- [Korn et al., 2011] Korn, N., Oppenheim, C.. "Licensing Open Data: A Practical Guide". In: Discovery [online]. June 2011 [cit. 2012-02-20]. Retrieved from http://discovery.ac.uk/files/pdf/Licensing_Open_Data_A_Practical_Guide.pdf
- [LisboaOD] Lisbona Open Data, <http://www.lisboaparticipa.pt/pages/newApps.php>
- [MiOD] Datasets Comune di Milano, <http://www.turismo.milano.it/>
https://www.comune.milano.it/wps/portal/?urile=wcm:path:ist_it_contentlibrary/sa_sitecontent/sfoglia_news/notizie_primo_piano/tutte_notizie/lavoro_sviluppo_ricerca/adesione_comune_pia_ttaforma_e015
- [Navitia.io] Navitia.io, <https://navitia.opendatasoft.com>
- [Ngomo, Auer, 2011] A. Ngomo, S. Auer. "LIMES: a time-efficient approach for large-scale link discovery on the web of data". Proc. of the 22nd int. joint conf. on Artificial Intelligence, Vol.3. AAAI Press, 2011.
- [OASC] OASC: <http://oascities.org/> NGSI OASC: http://wiki.fiware.org/FI-WARE_NGSI_Open_RESTful_API_Specification
- [Open311] Open311, A collaborative model and open standard for civic issue tracking. <http://www.open311.org/>
- [OpenDataSoft] OpenDataSoft: <https://www.opendatasoft.com/>
- [Rancourt et al., 2013] Marie-Eve Rancourt, Jean-François Cordeau and Gilbert Laporte, Long-Haul Vehicle Routing and Scheduling with Working Hour Rules (2013), in: Transportation Science, 47:1(81-107)
- [RDF] RDF <https://www.w3.org/RDF/>
- [ServiceMap] ServiceMap: <http://servicemap.disit.org> , <http://www.disit.org/6597>
- [Shapiro 2006] Shapiro, Jesse M. "Smart cities: quality of life, productivity, and the growth effects of human capital." The review of economics and statistics 88.2 (2006): 324-335.
- [Smart cities: Ranking] Smart cities: Ranking of European medium-sized cities. Centre of Regional Science, Vienna University of Technology, 2007.
- [SMTUN] http://www.ponrec.it/open-data/risultati/smart-cities/pon04a2_g/
- [SOCRATA] SOCRATA: <https://www.socrata.com/>
- [SPARQL] SPARQL: <https://www.w3.org/TR/rdf-sparql-query/>
- [Transport.API] Transport.API, <http://www.transportapi.com>
- [Vilajosana et al., 2013] Ignasi Vilajosana, Jordi Llosa, Borja Martinez, Marc Domingo-Prieto, Albert Angles, "Bootstrapping smart cities through a self-sustainable model based on big data flows". Commun. Mag.51 (6) (2013) IEEE51 (6) (2013).

[Vilajosana et al., 2013] Vilajosana, I. ; Llosa, J. ; Martinez, B. ; Domingo-Prieto, M. ; Angles, A.,
"Bootstrapping smart cities through a self-sustainable model based on big data flows",
Communications Magazine, IEEE, Vol.51, n.6, 2013

8 Acronimi

- API: Application Program Interface
- AVL: Automatic vehicle location
- AVM: Automatic Vehicle Monitoring
- BDaaS: Big Data as a Service
- CAP principle: Consistency Availability Partition Tolerance principle
- CBB: Content Based Billing
- CBB: Content Based Billing
- CEN: European Committee for Standardization
- DBMS: database management system
- FCD: Floating Cellular Data
- GPRS: General packet radio service
- GPS: Global positioning System
- GSM: Global System for Mobile
- ICT: Information and Communication Technologies
- ITS: Intelligent Transport Systems
- LCD: liquid-crystal display
- LOD: linked open data
- MC: Mobile Collector
- MMS: Multimedia Messaging Service
- NLP: Natural Language Processing
- NoSQL: no SQL database
- OD: open data
- OD: Open Data
- OGC: Open Geospatial Consortium
- OWL: Web Ontology Language
- PA: Pubblica Amministrazione
- PMI: Piccola e Media Impresa
- PMS: Private Mobile Systems
- POS: part-of-speech
- RDF: Resource Description Framework
- RFID: Radio Frequency IDentification o Identificazione a radio frequenza
- RTTI: Real-time Travel & Traffic Information
- SDI: Spatial Data Infrastructures
- SII: sistema di interoperabilità integrato
- SIMONE: progetto Simone
- SMS: Short Message Service
- SN: social networking, oppure sensor network
- SOA: Service Oriented Architecture
- SOAP: Simple Object Access Protocol
- SSAMM: Agenzia per la Mobilità Metropolitana strumenti di supporto, TOSCANA
- TPEG: Transport Protocol Experts Group
- TPL: gestore trasporto pubblico locale
- UML: Unified Modeling Language
- UMTS: Universal Mobile Telecommunications System
- UTC: Urban Traffic Control
- UUDI: Universal Description Discovery and Integration

- V2I: Vehicle-to-Infrastructure
- V2V: vehicle-to-vehicle
- VMS: Variable Message Sign
- VWSN: Vehicular Wireless Sensor Networks
- W3C: World Wide Web Consortium
- WSD: Word Sense Disambiguation
- WSDL: Web Services Description Language
- WSN: Wireless Sensor Networks
- XMI: XML Metadata Interchange standard di OMG
- XML: Extensible Markup Language
- ZTL: Zona a Traffico Limitato