

Cooperazione di 1° livello Servizi di Confine

Orchestratore Locale

Cooperazione di 1° livello

Analisi Tecnica

Cliente	Comune di Bologna		
Progetto	ELI-CAT		
Redatto da		Data	15/04/2009
Verificato da		Data	23/04/2009
Approvato da		Data	
Stato	Definitivo		
Versione	1.3		
Pagine	20		
Nome File	EL_8A8_Servizi di confine.doc		

Storia del documento

Versione	Data	Autore	Cambiamenti apportati
1.0	20/02/2009		Prima versione

Riferimenti

N°	Riferimento	Reperibile in

Indice generale

1. Introduzione.....	5
1.1 Obiettivi del documento	5
1.2 Struttura del documento	5
1.2.1 Guida alla lettura dei <i>diagrammi di sequenza</i>	5
1.3 Termini e definizioni	5
2. <i>Dominio DCOOP1: Cooperazione di 1° Livello</i>.....	Errore. Il segnalibro non è definito.
3. Specifiche dei servizi di confine e dei messaggi interni.....	11
3.1 <i>Web Service di Pubblicazione, Delivery, Segnalazione Errori, e Forzatura del Delivery</i>	11
3.1.1 XSD.....	12
3.1.2 XSD dell'XML di uscita.....	16
3.2 <i>Messaggi inviati sulle code interne dell'orchestratore locale</i>	17
3.2.1 XSD del messaggio per la pubblicazione in tempo reale	17
3.2.2 XSD del messaggio per la pubblicazione temporizzata.....	18
3.2.3 XSD del messaggio per la notifica di pubblicazione (prossima o mancata) via posta elettronica.....	19

1. Introduzione

1.1 Obiettivi del documento

Nel presente documento vengono espresse le specifiche dei processi dell'orchestratore locale che appartengono alla cooperazione di primo livello, specifiche intese come struttura di componenti concorrenti all'esecuzione del processo stesso.

1.2 Struttura del documento

1.2.1 Guida alla lettura dei *diagrammi di sequenza*

Nei diagrammi di sequenza per il dominio DCOOP1 si sono utilizzati seguenti stereotipi per gli oggetti al fine di determinarne meglio la natura. In particolare:

- **process**: indica un processo interno all'orchestratore locale;
- **processo esterno**: indica un processo esterno all'orchestratore locale, generalmente il destinatario di qualche messaggio; nel caso DCOOP1UC1 è la sorgente di tale messaggio;
- **coda JMS**: indica che l'oggetto è una coda JMS
- **database**: indica che l'oggetto è uno storage di dati, ossia il repository dell'orchestratore locale;
- **timer**: indica che l'oggetto è un timer ed invia segnali ad intervalli regolari.

Sono stati usati stereotipi anche per etichettare i metodi nel diagramma, in modo da evidenziare quelli che hanno una natura particolare rispetto all'invocazione del metodo su un oggetto. In particolare abbiamo:

- **signal**: indica un segnale di attivazione che usualmente si manifesta come invocazione di un metodo applicativo solo dopo aver "attraversato" uno strato di software infrastrutturale (JMS, JBI etc.);
- **soap call**: indica una invocazione di un web service con SOAP;
- **soap reply**: indica la risposta di un web service con SOAP;
- **web service**: indica la chiamata ad un webservice
- **javax.mail**: indica l'invio di un messaggio di posta elettronica, effettuato appoggiandosi alle API del package javax.mail.

Si è inoltre adottata la convenzione di rappresentare il perimetro dell'orchestratore locale con un area colorata. Nel diagramma di attività relativo allo use case DCOOP1P10UC1 si è utilizzata una seconda area colorata per indicare un insieme di oggetti che vanno replicato N volte.

1.3 Termini e definizioni

Nella tabella riportata di seguito sono elencati tutti gli acronimi e le definizioni adottate nel presente documento.

Termine	Definizione
JMS	Java Messaging Service, software infrastrutturale che offre meccanismi per inviare un messaggio ad uno o più destinatari, detti sottoscrittori
JBI	Specifica sviluppata all'interno del Java Community Process ¹ per la specifica di un approccio all'implementazione di un'architettura SOA.
SMTP	Simple Mail Transfer Protocol, protocollo standard per lo scambio di messaggi di posta elettronica tra host connessi alla rete Internet. Specificato dal documento RFC821.
MTA	Mail Transfer Agent, programma infrastrutturale preposto all'inoltro dei messaggi di posta elettronica verso utenti locali (appoggiandosi ad un Mail Delivery Agent) o remoto (consegnando il messaggio ad un altro Mail Transfer Agent)
Binding Component	Componente che interfaccia le componenti del processo con il mondo esterno utilizzando varie tecnologie e protocolli (SMTP, JMS, SOAP, HTTP). Si dividono in componenti di ingresso, che permettono all'informazione di entrare nel processo, e componenti di uscita, con cui il processo comunica i risultati al mondo esterno.

¹ Processo formalizzato che offre la possibilità di intervenire nello sviluppo della piattaforma Java.

2. Ruolo dell'Orchestratore Locale nell'architettura logica del progetto ELISA

L'orchestratore Locale definisce una vera e propria "dorsale di integrazione" per l'Ente o il CSC, vale a dire un'infrastruttura che abilita la comunicazione e la cooperazione fra i diversi sistemi.

Tale dorsale fornisce capacità di interoperabilità dei sistemi attraverso interscambi informativi incentrati sul concetto di "evento". Con questo termine si intende una rappresentazione formale, secondo grammatiche standardizzate e condivise, di "fatti" relativi ai processi organizzativi della PA che risultino significativi, dal punto di vista informativo, per diversi sistemi e applicazioni ai diversi livelli di governo, così come trasversalmente ai settori di un medesimo livello. La suddetta dorsale è costituita da un'infrastruttura atta ad ospitare processi di integrazione e dai processi stessi, realizzati ad hoc per ottemperare alle esigenze funzionali delle cooperazioni di primo, secondo e terzo livello; tale infrastruttura è costituita dalla piattaforma SOA/BPM Spagic (<http://www.spagoworld.org>). Inoltre, per facilitare le attività di configurazione e monitoraggio dell'esecuzione di tali processi, l'Orchestratore Locale è corredato da due applicazioni web, una per monitoraggio "sistemistico" (Spagic Console) ed una per le configurazioni ed il monitoraggio di "business" (OLManager). Per maggiori dettagli si rimanda ai documenti che descrivono il Modello di Dominio e le analisi dei singoli componenti.

I sistemi di area coinvolti in questo modello di cooperazione sono i seguenti:

⇒ **Sistemi interni all'Ente Locale**

- Anagrafe Comunale degli Immobili (ACI): produce eventi relativi a variazioni toponomastiche, nonché pertinenti altre entità da essa gestite (dati catastali, edifici, ecc.).
- Anagrafe Comunale Soggetti/Oggetti/Relazioni: produce eventi in base ai requisiti di implementazione del Modulo di Estensione nonché dello stesso Orchestratore Locale.
- Anagrafe della Popolazione: fornisce eventi relativi a variazioni anagrafiche dei residenti del Comune.
- Commercio: segnala nuove licenze commerciali o variazioni alle autorizzazioni esistenti.
- Sistema Informativo Tributi: fornisce eventi di variazione relativi a denunce, versamenti, atti di vario tipo, ecc.
- Edilizia privata: è produttrice di informazioni relative alle pratiche edilizie.

⇒ **Sistemi esterni all'Ente Locale**

- Agenzia del Territorio: per l'interscambio informativo dei dati di natura catastale
- Anagrafe Tributaria (SIATEL): fornisce i dati anagrafici relativi a soggetti censiti presso l'Anagrafe Tributaria del Ministero delle Finanze ed è anche la "sorgente" per forniture aggiuntive quali: utenze elettriche, gas, acqua, dichiarazioni dei redditi, bonifici relativi a ristrutturazioni, locazioni e successioni.
- Registro Imprese: fornisce eventi di variazione alle unità locali insistenti sul territorio comunale e censite in Infocamere.

2.1 Modalità di attivazione dei processi di cooperazione

Premesso che il modello PUBLISH/SUBSCRIBE adottato dall'Orchestratore Locale è per definizione asincrono, in questo paragrafo si vuole descrivere le possibili tipologie di attivazione delle orchestrazioni da parte dei sistemi esterni:

- **Sincrona:** la generazione dell'evento ed il suo corrispondente invio all'Orchestratore Locale avviene contestualmente al cambiamento dell'informazione all'interno del sistema di area stesso.
- **Asincrona:** la generazione dell'evento ed il suo corrispondente invio all'Orchestratore Locale avviene ad intervalli di tempo specifici per ogni sistema di area; in questo caso ciascun "evento" corrisponde ad un sottoinsieme di unità logiche distinte e ad un intervallo temporale di riferimento che può essere definito (giornaliero, settimanale, mensile, come ad esempio potrebbe avvenire nel caso dell'Anagrafe della Popolazione) o indefinito (come accade per le forniture dell'Agenzia delle Entrate, in cui l'evento nasce quando la fornitura diventa disponibile).

2.2 Livelli di cooperazione

All'interno del modello proposto, si individuano tre livelli di cooperazione distinti:

- **1° livello:** "produttore" e "consumatore" si scambiano "messaggi grezzi", vale a dire non mediati dall'Anagrafe Comunale SOR. Il primo livello di cooperazione prevede la possibilità, da parte di ciascun sistema di area interno al Comune (compresa Anagrafe SOR), di sottoscrivere il servizio di notifica delle variazioni pubblicate da un qualsiasi altro sistema di area interno o esterno all'Ente o al CSC. In questo scenario ciascun sistema dichiara la modalità di ricezione adottata che può essere "in tempo reale" o "temporizzata" (con dichiarazione dell'intervallo temporale richiesto). L'Orchestratore Locale implementa appositi meccanismi per "sincronizzare" come necessario le sorgenti con le destinazioni, ad es. immagazzinando in opportune code i messaggi di input sincroni ricevuti ed inviandoli in blocco ad un sistema che si sia dichiarato interessato a riceverli. Questo stesso modello di cooperazione è quello usato ad esempio da ACSOR per acquisire periodicamente le variazioni dai sistemi di area.
- **2° livello:** "produttore" e "consumatore" si scambiano "messaggi evoluti", per il tramite dell'Anagrafe SOR, ove il destinatario non riceve semplicemente "eventi generici", ma messaggi contestualizzati al proprio dominio, in quanto espressi in termini delle chiavi identificative interne relative alle proprie entità (soggetti, oggetti). Il secondo livello di cooperazione, più sofisticato, vede l'Anagrafe SOR agire come vero e proprio produttore di informazioni. Le classi di eventi gestite sono due:
 - **notifiche di variazioni anagrafiche relative ai soggetti**
 - **notifiche di variazioni toponomastiche, relative sia a soggetti che oggetti**

In entrambi i casi le variazioni sono pubblicate in termini delle chiavi interne di ciascun sistema satellite che abbia effettivamente partecipato all'integrazione dei dati nell'Anagrafe Comunale SOR.

- **3° livello:** a questo livello si ha una vera e propria orchestrazione di processi complessi, comprendenti diversi sistemi, rispetto alla stretta applicazione del modello PUBLISH/SUBSCRIBE. Tale livello di cooperazione, infatti, prevede l'implementazione di veri e propri processi trasversali tra singoli di sistemi area, che sfruttano la "conoscenza" dei sistemi esterni coinvolti, delegando ad essi il ruolo di attori primari per le corrispondenti responsabilità di "business" ed agendo da collante per fornire servizi complessi dati dalla somma delle competenze dei singoli.

3. Dominio DCOOP1: Cooperazione di 1° Livello

Il dominio "Cooperazione di 1° Livello" include le funzioni di pubblicazione degli eventi e la conseguente consegna a tutti i sottoscrittori per tale evento.

Include pertanto l'operazione di pubblicazione, messa a disposizione di chi ha il compito di pubblicare gli eventi.

Al dominio appartengono anche una serie di processi automatici preposti a:

- la consegna degli eventi in tempo reale ai sottoscrittori destinatari che hanno scelto questa forma di consegna;
- la consegna temporizzata degli eventi;
- l'invio ai publisher notifiche riguardo alle pubblicazioni da effettuare a breve o che non sono avvenute nei tempi previsti, accumulando ritardo;
- per la segnalazione di anomalie da parte dei subscriber;
- per la nuova sottomissione dei delivery la cui trasmissione è andata in errore;
- per la forzatura dell'invio immediato di un delivery in attesa.

4. Specifiche dei servizi di confine e dei messaggi interni

Passiamo ora a definire i web service di confine con cui, nell'ambito della cooperazione di primo livello, l'orchestratore locale contatta i sistemi esterni e viene a sua volta contattato. Di seguito descriveremo il formato dei messaggi interni che vengono inviati attraverso le code JMS.

4.1 Web Service di Pubblicazione, Delivery, Segnalazione Errori, e Forzatura del Delivery

Il web service di pubblicazione è quello esposto dall'orchestratore locale sia per il processo DCOOP1P1 ("Pubblicazione") che per il processo DCOOP1P10 ("Trasmissione"). un XML con i dati ausiliari del servizio, mentre il corpo dell'evento viaggia come allegato.

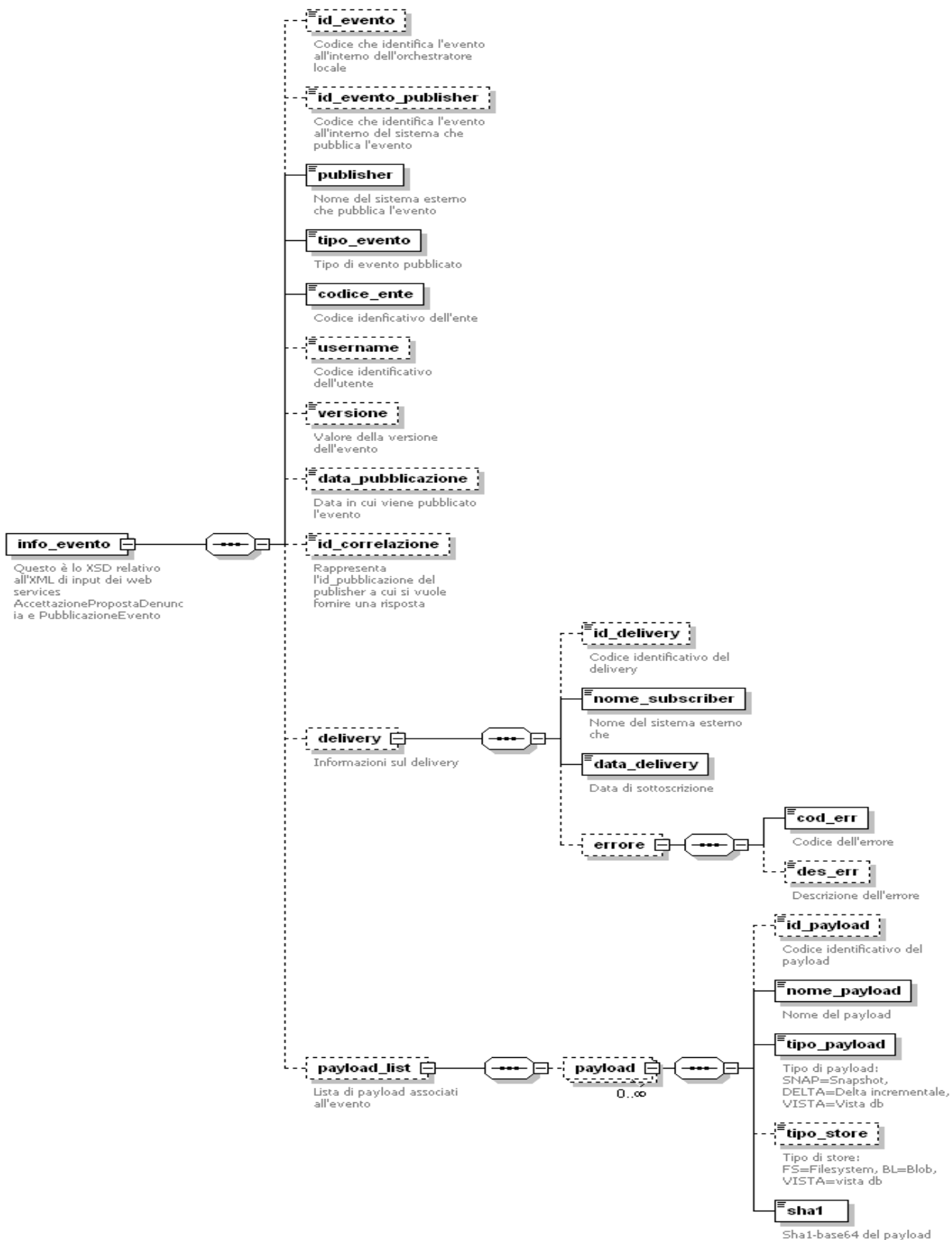
La risposta è ancora un messaggio XML di risposta con codice di errore (0 se tutto OK, maggiore di zero altrimenti) ed il relativo messaggio: "OK" se tutto è andato bene, una descrizione chiara di una situazione di anomalia applicativa oppure il messaggio associato ad un errore inatteso dell'infrastruttura.

Il web service di delivery **non** è esposto dall'orchestratore locale, ma è esposto dai vari subscriber ed **invocato** dall'orchestratore locale. Il servizio esposto dal subscriber deve presentare la medesima interfaccia del web service di pubblicazione.

Il web service di "Segnalazione Errori" è anche questo esposto dall'orchestratore locale così come quello di "Forzatura Delivery", questi ultimi non prevedono allegati.

Tutti questi web service presentano lo stesso WSDL e rispondono con un XML schematizzato dal medesimo XSD.

4.1.1 XSD



```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <xs:element name="info_evento">
    <xs:annotation>
      <xs:documentation>Questo è lo XSD relativo all'XML di input dei web services
AccettazionePropostaDenuncia e PubblicazioneEvento</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="id_evento" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>Codice che identifica l'evento all'interno dell'orchestratore
locale</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="id_evento_publisher" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>Codice che identifica l'evento all'interno del sistema che pubblica
l'evento</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="publisher" type="xs:string">
          <xs:annotation>
            <xs:documentation>Nome del sistema esterno che pubblica l'evento</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="tipo_evento" type="xs:string">
          <xs:annotation>
            <xs:documentation>Tipo di evento pubblicato</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="codice_ente" type="xs:string">
          <xs:annotation>
            <xs:documentation>Codice identificativo dell'ente</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="username" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>Codice identificativo dell'utente</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="versione" type="xs:string" minOccurs="0">
          <xs:annotation>
            <xs:documentation>Valore della versione dell'evento</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
</xs:annotation>
</xs:element>
<xs:element name="data_pubblicazione" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Data in cui viene pubblicato l'evento</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="id_correlazione" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Rappresenta l'id_pubblicazione del publisher a cui si vuole fornire
una risposta</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="delivery" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Informazioni sul delivery</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="id_delivery" type="xs:string" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Codice identificativo del delivery</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="nome_subscriber" type="xs:string">
        <xs:annotation>
          <xs:documentation>Nome del sistema esterno che </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="data_delivery" type="xs:string">
        <xs:annotation>
          <xs:documentation>Data di sottoscrizione</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="errore" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="cod_err" type="xs:string">
              <xs:annotation>
                <xs:documentation>Codice dell'errore</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="des_err" type="xs:string" minOccurs="0">
              <xs:annotation>
                <xs:documentation>Descrizione dell'errore</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="payload_list" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Lista di payload associati all'evento</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="payload" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="id_payload" type="xs:string" minOccurs="0">
              <xs:annotation>
                <xs:documentation>Codice identificativo del payload</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="nome_payload" type="xs:string">
              <xs:annotation>
                <xs:documentation>Nome del payload</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="tipo_payload" type="xs:string">
              <xs:annotation>
                <xs:documentation>Tipo di payload: SNAP=Snapshot, DELTA=Delta incrementale,
VISTA=Vista db</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="tipo_store" type="xs:string" minOccurs="0">
              <xs:annotation>
                <xs:documentation>Tipo di store: FS=Filesystem, BL=Blob, VISTA=vista
db</xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="sha1" type="xs:string">
              <xs:annotation>
                <xs:documentation>Sha1-base64 del payload</xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

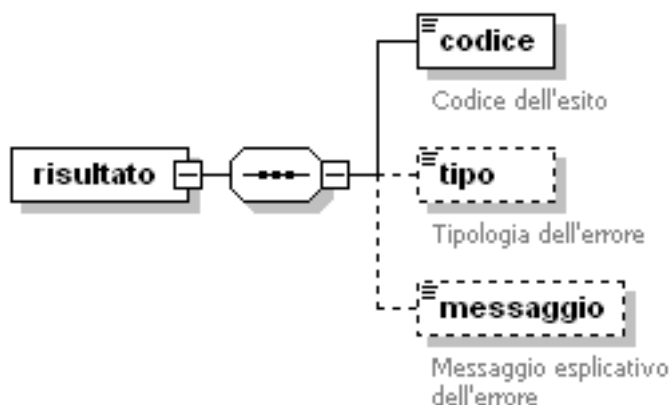
```

```

    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

4.1.2 XSD dell'XML di uscita



```
<?XML version="1.0"?>
```

XSD per lo XML ritornato dai web service implicati nelle collaborazioni di primo e terzo livello

```

<risultato>
  <codice>foo - 0 nessun errore, diverso da 0 codice di
    errore</codice>
  <tipo>Tipo dell'errore (i.e. classe dell'eccezione, Applicativo se
    errore applicativo - da ignorare e non mettere se non c'è
    errore</tipo>
  <messaggio>Messaggio di errore (OK per nessun errore)</messaggio>
</risultato>
-->

```

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.eng.it/PluginPEC"
  targetNamespace="http://www.eng.it/PluginPEC"
  elementFormDefault="qualified">
  <xs:element name="risultato">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="codice" type="xs:string">
          <xs:annotation>
            <xs:documentation>

```



```

        Codice dell'esito
    </xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="tipo"
              type="xs:string"
              minOccurs="0">
    <xs:annotation>
        <xs:documentation>
            Tipologia dell'errore
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element name="messaggio"
              type="xs:string"
              minOccurs="0">
    <xs:annotation>
        <xs:documentation>
            Messaggio esplicativo dell'errore
        </xs:documentation>
    </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

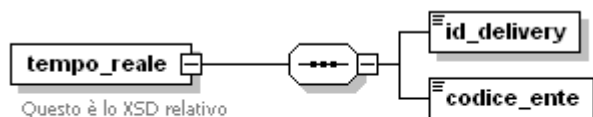
```

4.2 Messaggi inviati sulle code interne dell'orchestratore locale

Di seguito esponiamo gli XSD dei messaggi XML inviati sulle code interne dell'orchestratore locale che, ricordiamo, sono una per ciascuna delle tre attività:

- delivery in tempo reale
- delivery temporizzato
- notifica di pubblicazione via e-mail

4.2.1 XSD del messaggio per la pubblicazione in tempo reale



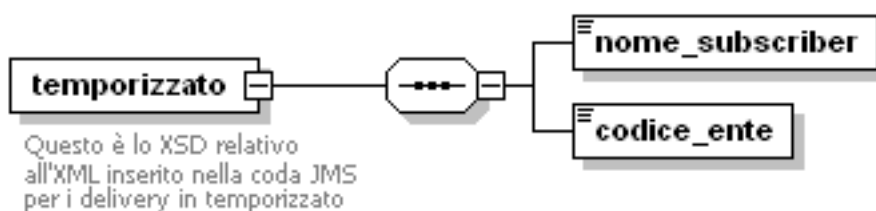
Questo è lo XSD relativo all'XML inserito nella coda JMS per i delivery in tempo reale.

I messaggi inviati sulla coda JMS per il delivery in tempo reale hanno come elemento radice 'tempo_reale'. Questo elemento a sua volta contiene cinque elementi:

- 'id_evento', di tipo intero. L'id dell'evento nella base dati del repository.
- 'id_subscriber', di tipo intero. L'id del sottoscrittore del delivery nella base dati del repository.
- 'codice_ente', di tipo string. Codice ente di riferimento.

```
<?XML version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="tempo_reale">
    <xs:annotation>
      <xs:documentation>Questo e' lo XSD relativo all'XML inserito
        nella coda JMS per i delivery in tempo reale.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="id_delivery" type="xs:integer"/>
        <xs:element name="codice_ente" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

4.2.2 XSD del messaggio per la pubblicazione temporizzata



I messaggi inviati sulla coda JMS per il delivery in tempo reale hanno come elemento radice 'temporizzato'. Questo elemento a sua volta contiene due elementi:

- 'nome_subscriber', di tipo string, Il nome del sottoscrittore cui inviare l'evento
- 'codice_ente', di tipo string. Codice ente di riferimento.

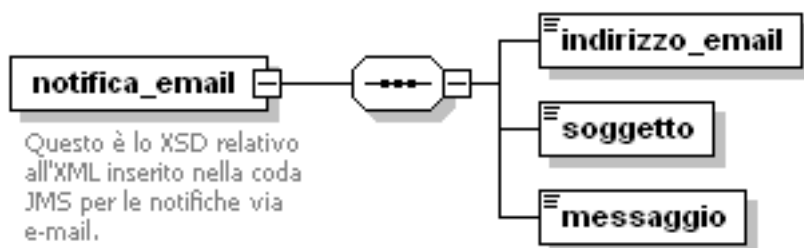
```
<?XML version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="temporizzato">
```

```

<xs:annotation>
  <xs:documentation>Questo è lo XSD relativo all'XML
    inserito nella coda JMS per i delivery in temporizzato
    reale.</xs:documentation>
</xs:annotation>
<xs:complexType>
  <xs:sequence>
    <xs:element name="nome_subscriber" type="xs:integer"/>
    <xs:element name="codice_ente" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

4.2.3 XSD del messaggio per la notifica di pubblicazione (prossima o mancata) via posta elettronica



I messaggi inviati sulla coda JMS per le notifiche di pubblicazione prossima o scaduta hanno come elemento radice 'notifica_email'. Questo elemento a sua volta contiene due elementi:

1. 'indirizzo_email', di tipo string. L'indirizzo e-mail cui inviare la notifica.
2. 'soggetto', di tipo string. L'oggetto del messaggio.
3. 'messaggio', di tipo string, Il messaggio da inviare al publisher destinatario della notifica.

```

<?XML version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <xs:element name="notifica_email">
    <xs:annotation>
      <xs:documentation>Questo è lo XSD relativo all'XML
        inserito nella coda JMS per le notifiche via
        e-mail.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="indirizzo_email"

```

```
                                type="xs:string"/>
    <xs:element name="soggetto"
                                type="xs:string"/>
    <xs:element name="messaggio"
                                type="xs:string"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```