Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Progettazione e realizzazione di servizi per l'interoperabilità del Fascicolo Sanitario Elettronico mediante l'infrastruttura InFSE

Mario Ciampi, Mario Sicuranza, Angelo Esposito, Giuseppe De Pietro

RT-ICAR-NA-2015-04 **Agosto 2015**

Consiglio Nazionale delle Ricerche, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR) – Sede di Napoli, Via P. Castellino 111, I-80131 Napoli, Tel: +39-0816139508, Fax: +39-0816139531, e-mail: napoli@icar.cnr.it, URL: *www.na.icar.cnr.it*

Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Progettazione e realizzazione di servizi per l'interoperabilità del Fascicolo Sanitario Elettronico mediante l'infrastruttura InFSE

Mario Ciampi, Mario Sicuranza, Angelo Esposito, Giuseppe De Pietro

Rapporto Tecnico N: **RT-ICAR-NA-2015-04** ***Data:*** **Agosto 2015**

Progettazione e realizzazione di servizi per l'interoperabilità del Fascicolo Sanitario Elettronico mediante l'infrastruttura InFSE

Mario Ciampi, Mario Sicuranza, Angelo Esposito, Giuseppe De Pietro

Istituto di Calcolo e Reti ad Alte Prestazioni del Consiglio Nazionale delle Ricerche (ICAR-CNR)
Via Pietro Castellino, 111 – 80131 Napoli
{mario.ciampi, mario.sicuranza, angelo.esposito, giuseppe.depietro}@na.icar.cnr.it

Abstract

I documenti sanitari generati dalle strutture sanitarie devono poter confluire nel Fascicolo Sanitario Elettronico dell'assistito a valle del consenso manifestato da quest'ultimo. A tal proposito, ciascun dominio a cui afferiscono le varie strutture sanitarie deve predisporre un sistema tecnologico capace di raccogliere tali documenti in maniera tale da renderli disponibili anche agli utenti di altri domini mediante l'interoperabilità dei sistemi tecnologici. Le principali funzionalità da rendere disponibili all'esterno del dominio riguardano la ricerca dei documenti, il recupero di un documento e la comunicazione dei metadati relativi ad un documento. Questo rapporto tecnico illustra il modello funzionale e tecnico di servizi atti a realizzare tali funzionalità, rispettando la tutela della privacy e della sicurezza, utilizzando opportunamente le componenti architetturali dell'infrastruttura tecnologica InFSE. I servizi progettati sono stati realizzati attraverso le componenti software OpenInFSE, utilizzate nell'ambito di diverse sperimentazioni.

1. Introduzione

Il progetto congiunto *"Infrastruttura tecnologica del Fascicolo Sanitario Elettronico"* tra il Dipartimento per la Digitalizzazione della Pubblica Amministrazione e l'Innovazione Tecnologica (in precedenza Dipartimento per l'Innovazione e le Tecnologie, oggi confluito nell'Agenzia per l'Italia Digitale) della Presidenza del Consiglio dei Ministri ed il Dipartimento ICT (oggi confluito nel Dipartimento DIITET) del Consiglio Nazionale delle Ricerche ha avuto l'obiettivo di definire un modello architetturale di riferimento di una infrastruttura tecnologica, denominata InFSE, per la gestione dei documenti sanitari degli assistito all'interno di un Fascicolo Sanitario Elettronico (FSE).

L'architettura software dell'infrastruttura tecnologica comprende un insieme di componenti software in grado di espletare le principali funzionalità inerenti alla gestione di un FSE secondo meccanismi federati ed interoperabili. L'architettura software dell'infrastruttura InFSE ed i servizi che la compongono sono descritti in linee guida e specifiche tecniche approvate dal Tavolo di Sanità Elettronica (TSE).

Il successivo progetto *"Realizzazione di un'infrastruttura operativa a supporto dell'interoperabilità delle soluzioni territoriali di fascicolo sanitario elettronico nel contesto del sistema pubblico di connettività"* (denominato brevemente OpenInFSE) ha avuto l'obiettivo di definire e mettere a disposizione soluzioni tecnologiche e componenti software per l'interoperabilità condivise tra tutti gli attori interessati e di realizzare un'infrastruttura operativa a supporto dell'interoperabilità delle soluzioni di FSE. Nell'ambito di questo progetto, l'infrastruttura è stata integrata con i sistemi di FSE di tre regioni italiane (Calabria, Campania e Piemonte) e utilizzata per favorire l'interscambio di alcuni documenti sanitari digitali, comprendenti referti radiologici, patient summary, referti di laboratorio e prescrizioni specialistiche.

L'infrastruttura InFSE è stata anche adottata nell'ambito del progetto italiano IPSE, collegato al progetto europeo epSOS, per l'interscambio di documenti di tipo patient summary tra le piattaforme territoriali di FSE. Al progetto hanno partecipato le Regioni Abruzzo, Emilia-Romagna, Friuli-Venezia Giulia,

Lombardia, Molise, Sardegna, Toscana, Umbria, Veneto e la Provincia Autonoma di Trento. Al fine di facilitare l'integrazione delle componenti infrastrutturali con le piattaforme territoriali per realizzare gli scenari previsti dal progetto, sono stati individuati i servizi di base dell'infrastruttura da esporre e specificate le modalità di interazione con essi.

Il progetto *"Evoluzione e interoperabilità del Fascicolo Sanitario Elettronico"*, in continuità con le iniziative precedenti, ha avuto l'obiettivo di consolidare i risultati già ottenuti e supportare la diffusione sul territorio nazionale di una soluzione di interoperabilità tecnologica del FSE. Anche questo progetto ha previsto una sperimentazione delle funzionalità offerte dall'infrastruttura, la quale è consistita nella possibilità di favorire l'interscambio di documenti sanitari disponibili presso i sistemi di FSE delle Regioni Calabria e Campania, ma consultabili anche da utenti presenti in altre Regioni, quali la Toscana.

Questo rapporto tecnico illustra le scelte effettuate per la realizzazione delle prinicipali funzionali di interoperabilità tra domini differenti attraverso l'infrastruttura InFSE, generalizzando le specifiche elaborate nell'ambito delle progettualità precedenti, al fine di consentire la ricerca ed il recupero di diverse tipologie di documenti sanitari e la loro notifica presso il dominio di pertinenza dell'assistito.

Il resto del documento è strutturato come descritto di seguito. La sezione 2 descrive sinteticamente l'architettura software dell'infrastruttura tecnologica InFSE, evidenziandone i servizi di base. La sezione 3 illustra il modello funzionale dei servizi di interoperabilità dell'infrastruttura InFSE per la realizzazione delle funzionalità di ricerca e recupero dei documenti e della comunicazione dei metadati di indicizzazione. La sezione 4 illustra gli scenari di interazione tra i servizi dell'infrastruttura InFSE per l'interoperabilità tra sistemi siti in domini differenti. La sezione 5 dettaglia il modello tecnico dei servizi di interoperabilità. La sezione 6 descrive gli aspetti inerenti alla tutela della sicurezza e privacy. La sezione 0 è dedicata alla descrizione delle modalità di utilizzo delle componenti software OpenInFSE per la realizzazione delle funzionalità di interoperabilità. Infine, la sezione 8 conclude il documento.

2. Architettura software dell'infrastruttura InFSE

L'architettura software dell'infrastruttura InFSE, mostrata inFigura 1, comprende un insieme di servizi software, implementati secondo la tecnologia Web Services, dispiegati secondo un modello di tipo federato presso opportuni *nodi* siti presso i domini che hanno l'esigenza di scambiare documenti sanitari del FSE.

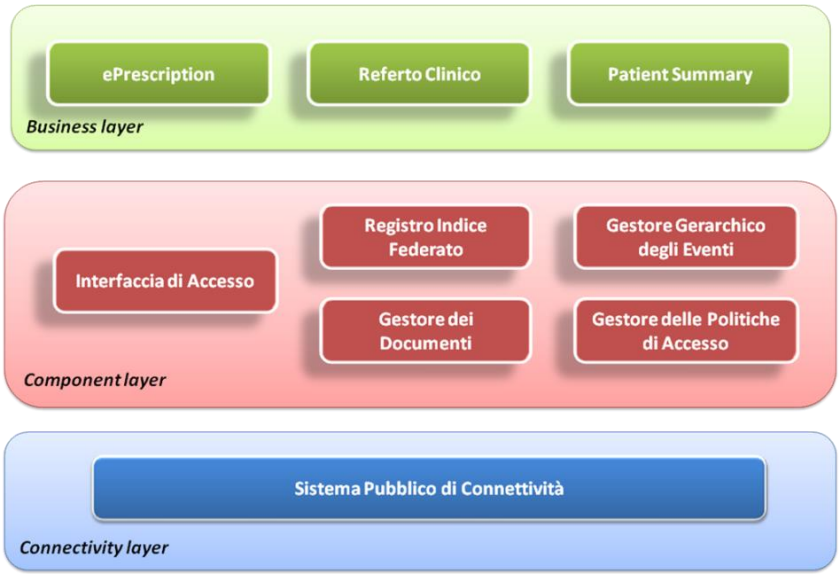

Figura 1. Architettura software dell'infrastruttura InFSE

Il modello prevede che ogni dominio sia dotato di un sistema di FSE basato sul paradigma registry/repository: il *registry* ha il compito di indicizzare i documenti sanitari di propria pertinenza, mentre il *repository* si occupa della memorizzazione di tali documenti. Ogni dominio può comprendere più repository, mentre deve esporre un unico registry, sebbene possa comprenderne altri di livello inferiore. Il cuore dell'infrastruttura è rappresentato dalla *federazione dei registry di livello superiore*. Questa soluzione permette di comunicare le informazioni di indicizzazione al sistema di FSE pertinente, per tramite dei nodi di dominio, e di ricercare i documenti di interesse inerenti ad un assistito, ovunque essi siano archiviati.

Il livello inferiore dell'architettura, *Connectivity layer*, è rappresentato dal Sistema Pubblico di Connettività (SPC) per la cooperazione applicativa tra le Pubbliche Amministrazioni.

Il livello intermedio, *Component layer*, comprende le seguenti componenti infrastrutturali:

- *Interfaccia di Accesso*: questa componente funge da interfaccia all'infrastruttura ed è dispiegata presso i nodi di dominio. Essa riceve le richieste da parte degli utenti abilitati all'accesso e dalle componenti analoghe degli altri domini e le propaga alle altre componenti dell'infrastruttura. Questa componente comprende i seguenti Web Services:
    - *IDocument*: funge da interfaccia rispetto alla componente *Gestore dei Documenti* per la pubblicazione ed il reperimento di documenti sanitari contenuti in un repository. Per quanto concerne la funzionalità di pubblicazione, interagisce anche con la componente *Registro Indice Federato* per memorizzare i metadati di un documento in un registry.
    - *IEntry*: funge da interfaccia rispetto alla componente *Registro Indice Federato* per permettere l'accesso ed il caricamento dei metadati inerenti ai documenti sanitari nei registry, l'invio di query e la gestione delle sottoscrizioni tra registry;
    - *IRegistryFederation*: funge da interfaccia rispetto alla componente *Registro Indice Federato* per gestire le federazioni di registry;
    - *IEvent*: funge da interfaccia rispetto alla componente *Gestore Gerarchico degli Eventi* per la creazione, la sottoscrizione e la notifica di eventi;
    - *IBrokerFederationMgt*: funge da interfaccia rispetto alla componente *Gestore Gerarchico degli Eventi* per la gestione delle federazioni di broker degli eventi.
- *Registro Indice Federato*: questa componente distribuita è fondata sulla federazione dei registry di livello superiore. L'obiettivo di questa componente è permettere la ricerca e la localizzazione dei documenti sanitari archiviati presso i diversi repository accessibili dall'infrastruttura. A tale scopo, essa prevede una serie di servizi che, dispiegati presso i nodi di dominio, consultano i metadati strutturati secondo un modello informativo condiviso e memorizzati nei registry. Questa componente comprende i seguenti Web Services:
    - *IMetadataMgt*: supporta le operazioni per gestire il ciclo di vita dei metadati (inserimento, aggiornamento, ecc.);
    - *IQueryMgt*: consente di sottoporre query ad uno specifico registry o ad una federazione di registry; in caso di query federata, propaga la query agli altri servizi *IQueryMgt* della federazione, aggrega i risultati e li restituisce all'utente;
    - *IEventMgt*: permette la notifica di eventi inerenti all'aggiornamento di metadati di interesse;
    - *IRegistryFederationMgt*: offre funzionalità per gestire la federazione di registry. In particolare, i registry di ogni dominio gestiscono le informazioni relative agli altri registry che fanno parte della federazione mediante opportuni metadati.
- *Gestore dei Documenti*: questa componente è dispiegata presso i repository delle strutture sanitarie. Essa consente di memorizzare e di recuperare i documenti creati da un utente autorizzato ad ogni occorrenza di un evento clinico di un assistito. Questa componente comprende il seguente Web Service:

    - *IDocumentMgt*: questo servizio permette di archiviare e recuperare un documento sanitario interagendo con uno specifico repository.

- *Gestore Gerarchico degli Eventi*: questa componente, facoltativa, effettua il routing e la notifica degli eventi clinici a tutti gli utenti interessati, adottando un modello gerarchico di classificazione degli eventi basato sul paradigma publish/subscribe. Questa componente comprende i seguenti Web Services:
    - *IPublisherRegistrationMgt*: permette la gestione delle registrazioni da parte dei produttori;
    - *ISubscriptionMgt*: permette la gestione delle sottoscrizioni da parte dei consumatori;
    - *INotificationBrokerMgt*: offre funzionalità per la gestione delle notifiche e delle strutture dati;
    - *IBrokerFederationMgt*: offre funzionalità per la gestione delle federazioni di broker degli eventi;
    - *IConsumer*: offre funzionalità che consentono ai consumatori di ricevere le notifiche.
- *Gestore delle Politiche di Accesso*: questa componente è responsabile degli aspetti generali di sicurezza. Essa consente, a valle delle fasi di autenticazione e di identificazione di un utente, di espletare la fase di autorizzazione relativa alle richieste di accesso ai documenti e ai metadati inerenti ad un dato assistito attraverso la valutazione di asserzioni di sicurezza contenute nei messaggi di richiesta e di politiche di accesso basate sul ruolo.

Il livello superiore dell'architettura, *Business layer*, comprende i servizi applicativi, quali l'ePrescription, la consultazione di referti clinici, il patient summary, ecc.

3. Modello funzionale dei servizi di interoperabilità

L'infrastruttura tecnologica InFSE offre una serie di servizi di base per la realizzazione di diverse funzionalità del FSE. Con specifico riferimento ai servizi a supporto dell'interoperabilità, le funzionalità che ogni sistema di FSE deve offrire per permettere ad un utente di un dominio di accedere al sistema di un altro dominio sono le seguenti:

- ricerca di documenti sanitari;
- recupero di un documento sanitario;
- comunicazione dei metadati di un documento sanitario.

Tali funzionalità possono essere offerte esponendo, da parte di ogni dominio, i seguenti servizi dell'infrastruttura InFSE:

- *IQueryMgt* della componente *Registro Indice Federato*, per la ricerca dei documenti: tale servizio deve essere in grado di ricevere una query, sottoporla al registry e restituire i metadati richiesti;
- *IDocument* della componente *Interfaccia di Accesso*, per il recupero di un documento: questo servizio deve essere in grado di interagire con un repository specificato (eventualmente interagendo con altri servizi) e restituire il documento richiesto.
- *IMetadataMgt* della componente *Registro Indice Federato*, per la comunicazione dei metadati: questo servizio deve permettere la trasmissione dei metadati tra domini differenti.

3.1. Ricerca di documenti sanitari

La funzionalità di ricerca dei documenti sanitari è offerta dalla componente *Registro Indice Federato* ed in particolare dal servizio *IQueryMgt*, il quale consente di ricevere una determinata query e di sottoporla ad uno specifico registry. La Tabella 1 mostra l'interfaccia dell'operazione di interesse del servizio, la cui struttura dati è conforme alle specifiche OASIS ebXML Registry V3.0.

Servizio: *IQueryMgt*		
Operazione *query*		È l'operazione che consente di ricevere una query, di sottoporla ad uno specifico registry e di ottenere i risultati.
	Parametro IN AdhocQueryRequest	È il parametro che specifica i criteri da utilizzare per la query, in maniera conforme alle specifiche ebXML.
	Parametro OUT AdhocQueryResponse	È il parametro che contiene la risposta del registry.

Tabella 1. Interfaccia del servizio *IQueryMgt*

Il parametro *AdhocQueryRequest* deve contenere il riferimento ad una stored query disponibile lato server. Il parametro *AdhocQueryResponse* contiene l'elenco dei risultati, corrispondente ad una lista di oggetti *RegistryObject*, in maniera conforme ad un modello informativo condiviso basato sullo standard ebXML Registry V3.0.

3.2. Recupero di un documento sanitario

L'interfaccia per la gestione dei documenti è *IDocument* della componente *Interfaccia di Accesso*, che permette di recuperare un documento dal FSE. In Tabella 2 è riportata una descrizione dell'interfaccia dell'operazione di interesse del servizio.

Servizio: *IDocument*		
Operazione *retrieveDocument*		È l'operazione che consente l'acquisizione di un documento.
	Parametro IN DocumentID	URN identificativo del documento richiesto.
	Parametro OUT DocumentObj	Documento richiesto in formato elettronico.

Tabella 2. Interfaccia del servizio *IDocument*

3.3. Comunicazione dei metadati di un documenti sanitario

La funzionalità di comunicazione dei metadati è offerta dalla componente *Registro Indice Federato* ed in particolare dal servizio *IMetadataMgt*, il quale consente di ricevere e memorizzare nel registry i metadati e di gestirne il ciclo di vita. La Tabella 3 mostra l'interfaccia delle operazioni di interesse del servizio, la cui struttura dati è conforme alle specifiche ebXML Registry V3.0.

Servizio: *IMetadataMgt*		
Operazione *registerEntry*		È l'operazione che consente di ricevere un insieme di metadati e di memorizzarli in uno specifico registry.
	Parametro IN SubmitObjectsRequest	È il parametro che contiene i metadati da inserire all'interno del registry.
	Parametro OUT RegistryResponse	È il parametro che contiene la risposta del registry.
Operazione *updateEntry*		È l'operazione che consente di aggiornare metadati esistenti in un registry.
	Parametro IN UpdateObjectsRequest	È il parametro che contiene i metadati da utilizzare per l'aggiornamento.
	Parametro OUT RegistryResponse	È il parametro che contiene la risposta del registry.
Operazione *deprecateEntry*		È l'operazione che consente di invalidare metadati esistenti in un registry.
	Parametro IN DeprecateObjectsRequest	È il parametro che specifica i criteri da utilizzare per l'invalidazione dei metadati.
	Parametro OUT RegistryResponse	È il parametro che contiene la risposta del registry.
Operazione *undeprecateEntry*		È l'operazione che consente di validare metadati precedentemente dichiarati invalidi in un registry.
	Parametro IN UndeprecateObjectsRequest	É il parametro che specifica i criteri da utilizzare per la validazione dei metadati.
	Parametro OUT RegistryResponse	È il parametro che contiene la risposta del registry.

Tabella 3. Interfaccia del servizio *IMetadataMgt*

Il servizio per la comunicazione dei metadati è fondamentale in un contesto in cui vi sia un dominio che ha in carico la gestione dei metadati dei documenti relativi agli assistiti di propria competenza. Infatti, nel caso in cui venga generato un documento per un dato assistito in un dominio diverso da quello di pertinenza, risulta opportuno inoltrare verso quest'ultimo i metadati di indicizzazione.

L'operazione *RegisterEntry* consente di ricevere i metadati di indicizzazione di un documento prodotto e memorizzato in un dominio differente da quello di pertinenza dell'assistito.

L'operazione *UpdateEntry* permette di aggiornare i metadati associati ad un documento.

L'operazione *DeprecateEntry* consente di invalidare i metadati relativi ad un determinato documento. Questa operazione è utile quando il dominio in cui è memorizzato il documento deve segnalare al dominio di competenza che il documento è diventato obsoleto (ad es. nel caso in cui venga prodotto un nuovo patient summary).

L'operazione *UndeprecateEntry* permette di rivalidare un documento precedentemente dichiarato invalido.

3.4. Modello concettuale dei metadati

Al fine di permettere l'invio di query in maniera condivisa, ogni registry deve indicizzare i documenti sanitari disponibili nel proprio dominio mediante metadati conformi ad un modello informativo condiviso. I principali elementi di tale modello comprendono almeno i seguenti dati:

- tipo di documento (patient summary, referto di laboratorio, ecc.);
- identificativo dell'assistito;
- stato del documento;
- data di creazione del documento;
- livello di confidenzialità del documento;
- puntamento al documento secondo la seguente tripla:
    - riferimento al dominio;
    - riferimento al repository contenente il documento all'interno del dominio;
    - identificativo locale del documento.

A valle della query, è possibile ottenere uno specifico documento sanitario archiviato presso un dominio attraverso la tripla di informazioni recuperate.

4. Scenari di interazione tra i servizi di interoperabilità

Il recupero di un documento disponibile in un dominio da parte di un utente afferente ad un altro dominio prevede le seguenti fasi:

- ricerca dei riferimenti ai documenti, soddisfacenti particolari criteri di ricerca, presso uno o più domini;
- recupero fisico di un documento presente in uno specifico dominio.

Inoltre, allo scopo di consentire ad un dominio di tenere traccia dei documenti di propria pertinenza, ciascun dominio che ha prodotto un documento deve eseguire la seguente operazione:

- invio dei metadati di indicizzazione del documento al dominio di pertinenza.

Di seguito sono descritti i possibili scenari per ogni fase.

4.1. Scenario 1.a: Ricerca diretta di documenti sanitari

In questo scenario l'utente del Dominio A effettua la richiesta al servizio *IQueryMegt* esposto dal Dominio B di pertinenza dell'assistito, come mostrato in Figura 2. L'utente ottiene in risposta i riferimenti ai documenti o un messaggio di errore, ad esempio nel caso l'utente non possegga i privilegi di accesso alla risorsa.

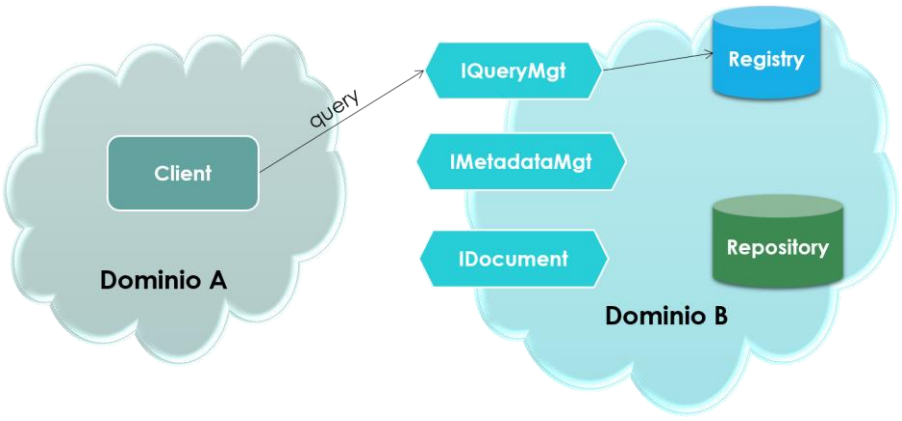

Figura 2. Flusso di interazioni per la realizzazione di una ricerca diretta

4.2. Scenario 1.b: Ricerca federata di documenti sanitari

In questo scenario l'utente del Dominio A effettua la richiesta ai servizi *IQueryMgt* esposti dai domini facenti parte della federazione, come mostrato in Figura 3. Questa operazione può essere effettuata, ad esempio, per gli usi secondari del FSE, per i quali tipicamente la ricerca non è basata su un singolo assistito. L'utente ottiene in risposta i riferimenti ai documenti o un messaggio di errore.

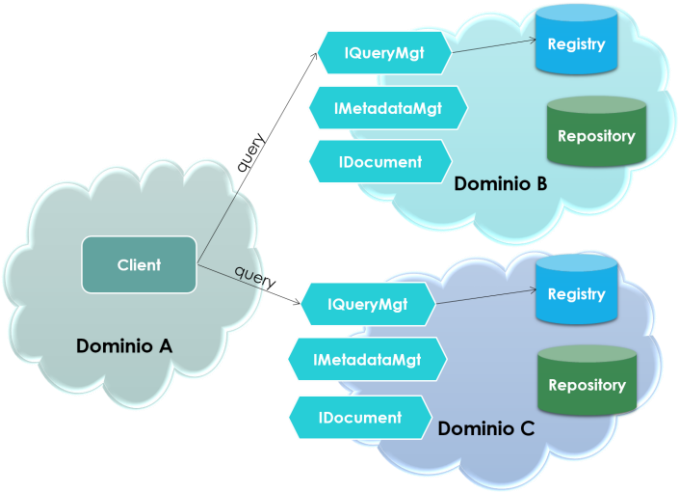

Figura 3. Flusso di interazioni per la realizzazione di una ricerca federata

4.3. Scenario 2: Recupero di un documento sanitario

Questo scenario prevede l'invio di una richiesta di recupero di un documento, da parte di un utente della Dominio A, al servizio *IDocument* del dominio contenente il documento, come mostrato in Figura 4. L'utente ottiene in risposta il documento o un messaggio di errore.

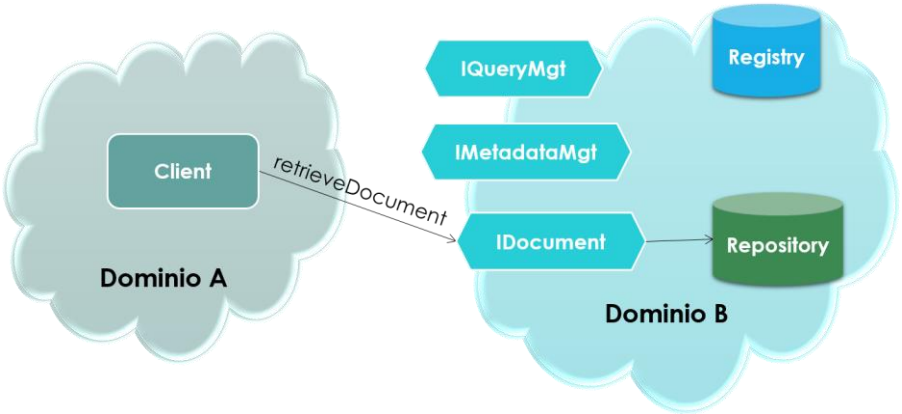

Figura 4. Flusso di interazioni per il recupero di un documento

4.4. Scenario 3.a: Comunicazione dei metadati di un documento sanitario creato

Questo scenario prevede l'invio di metadati di indicizzazione relativi ad un documento sanitario creato in un dominio diverso da quello di pertinenza dell'assistito. La richiesta deve essere inviata al servizio *IMetadataMgt*, come mostrato in Figura 5. L'utente ottiene in risposta un messaggio di notifica dell'esito dell'operazione.

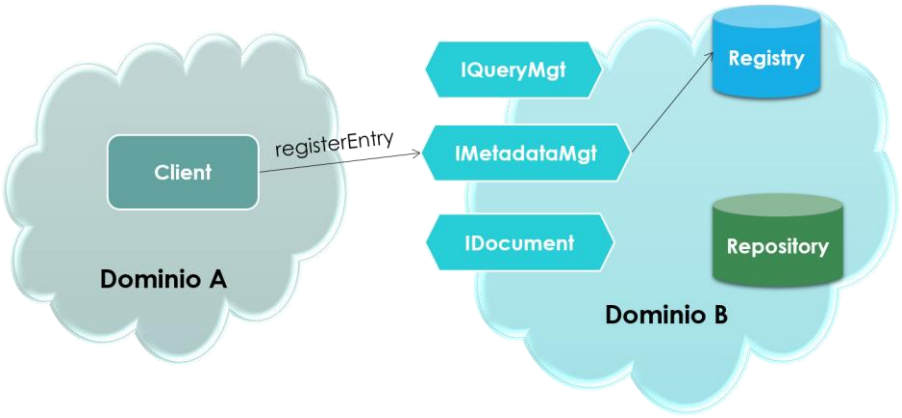

Figura 5. Flusso di interazioni per la comunicazione dei metadati di un documento sanitario creato

4.5. Scenario 3.b: Comunicazione di metadati aggiornati

Questo scenario prevede l'aggiornamento di uno o più metadati di indicizzazione relativi ad un documento sanitario, per il quale il dominio che lo contiene avefva già inviato i metadati corrispondenti. La richiesta deve essere inviata al servizio *IMetadataMgt*, come mostrato in Figura 6. L'utente ottiene in risposta un messaggio di notifica dell'esito dell'operazione.

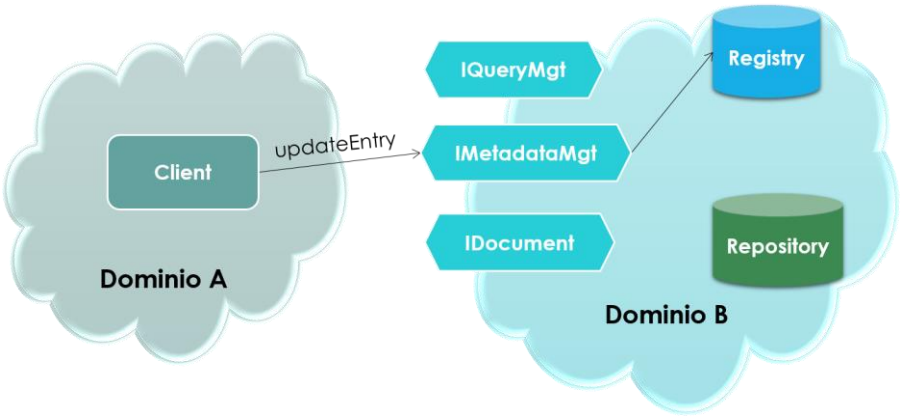

Figura 6. Flusso di interazioni per la comunicazione di metadati aggiornati

4.6. Scenario 3.c: Comunicazione dei metadati di un documento sanitario aggiornato

Questo scenario prevede l'invio di metadati di indicizzazione relativi ad un documento sanitario, creato in un dominio diverso da quello di pertinenza dell'assistito, che aggiorna un documento precedente, il quale deve essere oppurtunamente reso obsoleto. Le richieste devono essere inviate al servizio *IMetadataMgt*, come mostrato in Figura 7. L'utente ottiene in risposta messaggi di notifica dell'esito delle operazioni.

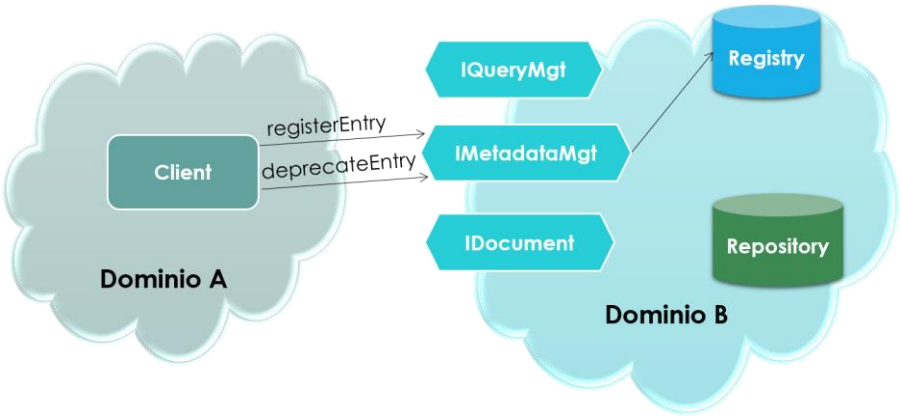

Figura 7. Flusso di interazioni per la comunicazione dei metadati di un documento sanitario aggiornato

4.7. Scenario 3.d: Rivalidazione un documento sanitario

Questo scenario prevede l'invio di una richiesta atta a rivalidare un documento reso invalido per errore. La richiesta deve essere inviata al servizio *IMetadataMgt*, come mostrato in Figura 8. L'utente ottiene in risposta un messaggio di notifica dell'esito delle operazione.

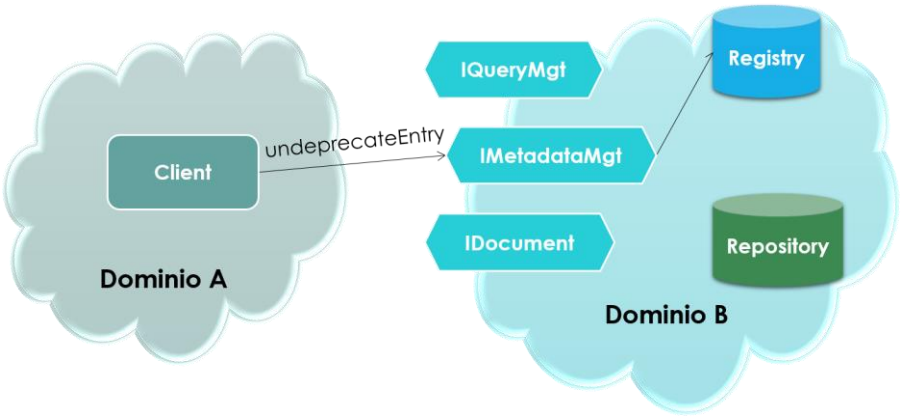

Figura 8. Flusso di interazioni per la rivalidazione di un documento sanitario

5. Modello tecnico dei servizi di interoperabilità

Questa sezione descrive il modello tecnico delle interfacce dei seguenti servizi di interoperabilità per la realizzazione delle funzionalità descritte in precedenza:

- *IQueryMgt* della componente *Registro Indice Federato*, per la ricerca di documenti sanitari indicizzati nel dominio dove è dislocato il servizio;
- *IDocument* della componente *Interfaccia di Accesso*, per il recupero di un documento sanitario disponibile in un dominio;
- *IMetadataMgt* della componente *Registro Indice Federato* per l'inoltro di richieste relative ai metadati di indicizzazione dei documenti sanitari.

5.1. Servizio IQueryMgt

Ogni dominio deve esporre il servizio *IQueryMgt*, il quale deve essere in grado di interpretare una richiesta di invocazione di una **stored query parametrizzata**, i cui parametri sono descritti di seguito.

L'operazione che deve essere invocata per l'invio della query è la seguente:

- *AdhocQueryResponse query(AdhocQueryRequest)*

La Tabella 4 mostra l'elenco dei parametri che possono essere inviati nel parametro di ingresso *AdhocQueryRequest*.

Parametro	Descrizione	Obb.
Id	Rappresenta l'identificativo della stored query (*urn:uuid:14d4debf-8f97-4251-9a74-a90016b0af0d*).	R
$XDSDocumentEntryPatientId	Rappresenta l'identificativo dell'assistito.	R
$XDSDocumentEntryStatus	Rappresenta lo stato del documento (*urn:oasis:names:tc:ebxml-regrep:StatusType:Approved*).	R
$XDSDocumentEntryClassCode	Rappresenta il tipo di documento secondo la codifica LOINC.	O
$XDSDocumentEntryCreationTimeFrom	Rappresenta il limite inferiore della data di creazione del documento.	O
$XDSDocumentEntryCreationTimeTo	Rappresenta il limite superiore della data di creazione del documento.	O

Tabella 4. Parametri della stored query (R = obbligatorio, O = opzionale)

Dopo aver ricevuto la richiesta, il servizio deve sottoporre la query al proprio registry e restituire, nel parametro di uscita *AdhocQueryResponse*, l'elenco dei metadati corrispondenti o un messaggio di errore.

5.2. Servizio IDocument

Ogni dominio deve esporre il servizio *IDocument*, il quale deve essere in grado di interpretare una richiesta di recupero di un documento.

L'operazione che deve essere invocata per il recupero di un documento è la seguente:

- *DocumentObj retrieveDocument(DocumentID)*

La Tabella 5 mostra l'elenco dei parametri che devono essere inviati nel parametro di ingresso *DocumentID*, i quali sono recuperati a valle della query.

Parametro	Descrizione
IDocumentUrl	Rappresenta il riferimento al dominio contenente il documento. È utilizzato dall'utente per invocare il servizio *IDocument* esposto dal dominio di interesse. Corrisponde al metadato *regionalServiceEndpoint*.
IDocumentMgtUrl	Rappresenta il riferimento, utilizzato dal servizio *IDocument*, allo specifico repository della struttura sanitaria che contiene il documento. Corrisponde al metadato *repositoryServiceEndpoint*.
documentID	Rappresenta l'identificativo di un documento, che permette di identificare uno specifico documento in un repository. Corrisponde al metadato *repositoryitem*.

Tabella 5. Parametri per la richiesta di recupero di un documento

Dopo aver ricevuto la richiesta, il servizio deve interagire con il repository specificato e restituire il documento richiesto nel parametro di uscita *DocumentObj*, o un messaggio di errore. In particolare, il parametro di uscita *DocumentObj* deve contenere gli elementi indicati nella Tabella 6.

Parametro	Descrizione
document	Rappresenta una struttura Base64 contenente il documento in uno specifico formato (ad es. HL7-CDA2).
documentName	Rappresenta il nome del documento nel formato utilizzato per l'elemento *document*.
documentType	Rappresenta la descrizione del tipo di documento (ad es. patient summary).
styleSheet	Rappresenta una struttura Base64 contenente il documento in uno specifico formato (ad es. PDF/A o lo stylesheet del documento HL7-CDA2).
styleSheetName	Rappresenta il nome del documento nel formato utilizzato per l'elemento *stylesheet*.

Tabella 6. Elementi restituiti dal servizio *IDocument*

5.3. Servizio IMetadataMgt

Ogni dominio deve esporre il servizio *IMetadataMgt*, il quale consente di ricevere i metadati relativi ai documenti sanitari prodotti in altri domini.

Le operazioni di interesse sono le seguenti:

- *RegistryResponse registerEntry(SubmitObjectsRequest)*
- *RegistryResponse updateEntry(UpdateObjectsRequest)*
- *RegistryResponse deprecateEntry(ApproveObjectsRequest)*
- *RegistryResponse undeprecateEntry(DeprecateObjectsRequest)*

La Tabella 7 e la Tabella 8 mostrano i parametri di ingresso/uscita delle operazioni.

Parametro	Descrizione
SubmitObjectsRequest	Rappresenta l'elenco dei metadati da registrare.
UpdateObjectsRequest	Rappresenta l'identificativo dell'insieme dei metadati ed i nuovi valori dei campi da aggiornare.
ApproveObjectsRequest	Rapresenta l'identificativo dell'insieme dei metadati da invalidare.
DeprecateObjectsRequest	Rappresenta l'identificativo dell'insieme dei metadati da rivalidare.

Tabella 7. Parametri per le richieste relative alla comunicazione di metadati

Parametro	Descrizione
RegistryResponse	Rappresenta un messaggio contenente l'esito dell'operazione.

Tabella 8. Elemento restituito dal servizio *IMetadataMgt*

5.4. Modello informativo dei metadati

Il modello informativo dei metadati che l'infrastruttura InFSE è in grado di gestire è piuttosto esteso. Questo paragrafo descrive l'insieme di metadati obbligatori che ogni dominio deve utilizzare per l'indicizzazione dei documenti disponibili presso i propri repository. È comunque possibile prevedere l'utilizzo di ulteriori metadati.

L'elenco dei metadati obbligatori è il seguente:

1. stato del documento;
2. tipo di documento;
3. data di creazione del documento;
4. identificativo dell'assistito;
5. riferimento al documento;
6. codice di confidenzialità del documento;
7. elenco dei ruoli ammessi all'accesso.

La struttura dei metadati deve essere conforme allo standard ebXML Registry Information Model (ebRIM) V3.0. La Tabella 9 mostra la rappresentazione di ogni metadato.

Id	Nome elemento	Tipo elemento	Descrizione
0	**ClinicalDocument**	*ExtrinsicObject*	Rappresenta il documento indicizzato, il quale contiene l'elenco di tutti i metadati riportati nella tabella.
1	**status**	*ExtrinsicObject.status*	Rappresenta lo stato del documento. Possibili valori: *Approved*, *Deprecated*, *Submitted*, *Withdrawn*.

2	**code**	*Classification*	Rappresenta il tipo di documento secondo la codifica LOINC.
3	**effectiveTime**	*ExtrinsicObject.Slot*	Rappresenta la data di creazione del documento, il cui formato di codifica è YYYYMMddhhmmss+|-ZZzz.
4	**XDSDocumentEntry.patientId**	*ExternalIdentifier*	Rappresenta l'identificativo dell'assistito rappresentato dal codice fiscale.
5.1	**regionalServiceEndpoint**	*ExtrinsicObject.Slot*	Primo parametro per l'identificazione di un documento. Corrisponde al nome simbolico del dominio contenente il documento.
5.2	**repositoryServiceEndpoint**	*ExtrinsicObject.Slot*	Secondo parametro per l'identificazione di un documento. Corrisponde al nome simbolico del repository contenente il documento.
5.3	**repositoryitem**	*ExtrinsicObject.Slot*	Terzo parametro per l'identificazione di un documento su scala nazionale. Corrisponde all'identificativo locale del documento.
6	**confidentialityCode**	*Classification*	Rappresenta il codice che specifica il livello di confidenzialità del documento.
7	**role**	*Classification*	Rappresenta l'elenco dei ruoli degli utenti che possono accedere al FSE dell'assistito.

Tabella 9. Struttura dei metadati secondo il modello ebRIM

6. Sicurezza e privacy

La componente *Gestore delle Politiche di Accesso* ha il compito di verificare l'autorizzazione all'accesso ai servizi InFSE. La fase di autorizzazione consiste nel determinare se l'utente di un dominio che invia una richiesta ad un altro dominio possiede i diritti ad accedere alle risorse richieste. L'autorizzazione è fornita o negata dopo la verifica di una serie di attributi specificati in un portafoglio di asserzioni di sicurezza, firmate digitalmente, sulla base delle politiche del dominio erogatore e del consenso dell'assistito a cui le risorse fanno riferimento.

Il portafoglio di asserzioni che deve essere costruito deve essere basato sullo standard OASIS SAML V2.0.

Le asserzioni sono riportate di seguito:

1. *asserzione di identità*: contiene l'attestazione relativa all'identificazione effettuata dall'utente che intende usufruire dei servizi;
2. *asserzione di attributo*: provvede a garantire la qualifica e/o gli attributi dell'utente;
3. *asserzione applicativa*: contiene eventuali informazioni atte ad associare l'utente ad un soggetto terzo per la delega all'accesso a determinate risorse, informazioni relative alla destinazione d'uso associata all'identità dell'utente, il contesto in cui l'utente si trova ad operare e che per tali motivi richiede l'accesso alla risorsa;
4. *asserzione di autorizzazione*: contiene l'autorizzazione ad accedere ad un particolare documento identificato a valle di una query.

6.1. Asserzioni di sicurezza

6.1.1. Asserzione di identità

Nell'asserzione di identità è presente l'attestazione relativa all'identificazione effettuata dall'utente che intende usufruire di servizi. Il sistema che provvede all'identificazione (Issuer) si fa garante del *level of assurance* ed è per questo responsabile di quanto asserito. Un'asserzione di identità valorizza sia un *AuthnStatement* con le informazioni del sistema responsabile dell'identificazione dell'utente e dell'istante di identificazione, sia un *AttributeStatement* contenente l'attributo riportato in Tabella 10. L'intera asserzione è poi firmata dall'*Identity Provider* che in questo modo si assume la responsabilità di quanto asserito.

Attribute ID	URN	Descrizione
SubjectId	*urn:oasis:names:tc:xacml:1.0:subject:subject-id*	Identificativo dell'utente.

Tabella 10. Struttura dell'asserzione di identità

6.1.2. Asserzione di attributo

L'asserzione di attributo provvede a garantire la qualifica e/o gli attributi dell'utente ed è emessa dall'ente che è in grado di certificare tale informazione per l'utente registrato. L'asserzione di attributi utente valorizza un *AttributeStatement* valorizzata con gli attributi riportati in Tabella 11. L'intera asserzione è poi firmata dall'*Attribute Authority* dell'ente che in questo modo si assume la responsabilità di quanto asserito.

Attribute ID	URN	Descrizione
SubjectId	*urn:oasis:names:tc:xacml:1.0:subject:subject-id*	Identificativo dell'utente.
OrganizationId	*urn:oasis:names:tc:xspa:1.0:subject:organization-id*	Identificativo del dominio presso cui l'utente è in carico.
Organization	*urn:oasis:names:tc:xspa:1.0:subject:organization*	Descrizione del dominio presso cui l'utente è in carico.
Role	*urn:oasis:names:tc:xacml:2.0:subject:role*	Ruolo dell'utente.

Tabella 11. Struttura dell'asserzione di attributo

6.1.3. Asserzione applicativa

Nell'asserzione applicativa sono contenute:

- eventuali informazioni atte ad associare l'utente ad un soggetto terzo per la delega all'accesso a determinate risorse. Come le altre tipologie di asserzioni, ha sempre una validità temporale e, inoltre, può essere in ogni momento revocata dall'utente;
- informazioni relative alla destinazione d'uso relative all'accesso al servizio richiesto;
- il contesto in cui l'utente si trova ad operare che, per tali motivi, richiede l'accesso alla risorsa;
- il consenso dell'assistito (nell'accezione di annotazione del medico, ossia quest'ultimo attesta che l'assistito ha fornito il suo consenso puntuale alla consultazione di specifici documenti sanitari).

L'asserzione applicativa valorizza un *AttributeStatement* con gli attributi riportati in Tabella 12. Utilizzando la propria smartcard, l'utente firma l'intera asserzione e, in questo modo, si assume la responsabilità di quanto asserito.

Attribute ID	URN	Descrizione
SubjectId	*urn:oasis:names:tc:xacml:1.0:subject:subject-id*	Identificativo dell'utente.
OrganizationId	*urn:oasis:names:tc:xspa:1.0:subject:organization-id*	Identificativo del dominio presso cui l'utente è in carico.
Organization	*urn:oasis:names:tc:xspa:1.0:subject:organization*	Descrizione del dominio presso cui l'utente è in carico.
EnvironmentLocality	*urn:oasis:names:tc:xspa:1.0:environment:locality*	Posizione dalla quale l'utente opera (ospedale, studio medico, casa dell'assistito).
SubjectRole	*urn:oasis:names:tc:xacml:2.0:subject:role*	Ruolo dell'utente.
SubjectPurposeOfUse	*urn:oasis:names:tc:xspa:1.0:subject:purposeofuse*	Destinazione d'uso della richiesta.
ResourceType	*urn:oasis:names:tc:xspa:1.0:resource:hl7:type*	Tipo della risorsa a cui si intende accedere (es. patient summary).
ResourceId	*urn:oasis:names:tc:xacml:1.0:resource:resource-id*	Identificativo dell'assistito la cui risorsa si intende accedere.
PatientConsent	*urn:oasis:names:tc:xspa:1.0:resource:patient:consent*	Manifestazione del consenso puntuale ricevuto dall'assistito.
ResourceAction	*urn:oasis:names:tc:xacml:1.0:action:action-id*	Azione che l'utente intende effettuare sulla risorsa.

Tabella 12. Struttura dell'asserzione applicativa

6.1.4. Asserzione di autorizzazione

L'asserzione di autorizzazione provvede a concedere o negare l'accesso ad un determinato servizio richiesto. È emessa dall'amministrazione erogante sulla base di politiche prestabilite. Un'asserzione di autorizzazione valorizza un *AttributeStatement* contenente gli attributi riportati in Tabella 13. L'intera asserzione è poi firmata dal Policy Enforcement Point dell'ente che in questo modo si assume la responsabilità di quanto autorizzato.

Attribute ID	URN	Descrizione
SubjectId	*urn:oasis:names:tc:xacml:1.0:subject:subject-id*	Identificativo dell'utente che ha inviato la richiesta.
Role	*urn:oasis:names:tc:xacml:2.0:subject:role*	Ruolo dell'utente che ha inviato la richiesta.

Tabella 13. Struttura dell'asserzione di autorizzazione

6.2. Verifica delle asserzioni

La gestione delle asserzioni di sicurezza è effettuata dalla componente *Gestore delle Politiche di Accesso* dell'infrastruttura InFSE. Essa si compone di due sotto-componenti in conformità allo standard XACML: il Policy Enforcement Point (PEP) ed il Policy Decision Point (PDP).

Il PEP, dislocato presso ogni servizio, ha il compito di intercettare ogni richiesta per effettuare le seguenti verifiche:

- analizzare la correttezza della struttura delle asserzioni;
- verificare la firma digitale di ogni asserzione;
- verificare la coerenza dell'asserzione applicativa con il messaggio applicativo;
- verificare se il ruolo dell'utente ed il purpose of use sono ammissibili;
- verificare il consenso dell'assistito, interagendo con il sistema di gestione del consenso utilizzato nello specifico dominio.

Se le verifiche non vanno a buon fine, il PEP restituisce un messaggio di errore. In caso contrario, esso invoca la sotto-componente PDP. Quest'ultima ha l'obiettivo di autorizzare o meno l'accesso al servizio consultando opportune politiche di accesso basate sul ruolo.

6.2.1. Verifica delle asserzioni per la ricerca dei documenti

Il portafoglio di asserzioni contenente le prime tre asserzioni deve essere spedito dall'utente contestualmente all'invio della query al servizio *IQueryMgt*, per permettere al PEP dispiegato presso questo servizio di effettuare le opportune verifiche. In particolare, per quanto concerne la verifica della coerenza dell'asserzione applicativa con il messaggio applicativo, il PEP deve controllare che l'identificativo dell'assistito indicato nell'asserzione coincide con quello presente nella query contenuta nel messaggio applicativo. Nel caso in cui le verifiche vadano a buon fine, il PEP, dopo aver ottenuto esito positivo dal PDP, invoca il servizio *IQueryMgt*, il quale sottopone la query al registry. Se i documenti ricercati esistono, il PEP restituisce all'utente, contestualmente al messaggio di risposta del servizio, opportune asserzioni di autorizzazione all'accesso ai documenti, che saranno utilizzate nella richiesta successiva di recupero del documento. Questa tipologia di asserzione deve contenere l'identificativo del documento e deve essere valida per uno specifico periodo temporale.

6.2.2. Verifica delle asserzioni per il recupero di un documento

L'asserzione di autorizzazione ottenuta dalla componente PEP, dislocata presso il servizio *IQueryMgt* invocato, deve essere spedita dall'utente contestualmente all'invio della richiesta di recupero di un documento. Tale messaggio è spedito al servizio *IDocument* per permettere alla componente PEP dispiegata presso questo servizio di effettuare le opportune verifiche. In particolare, per quanto concerne la verifica della coerenza dell'asserzione di autorizzazione con il messaggio applicativo, la componente PEP deve

controllare che l'identificativo del documento indicato nell'asserzione coincide con quello presente nella richiesta contenuta nel messaggio applicativo. Nel caso in cui le verifiche vadano a buon fine, il PEP invoca il servizio *IDocument*, il quale richiede l'interrogazione del repository indicato allo scopo di restituire il documento richiesto. Considerato che la valutazione della decisione da parte del PDP viene svolta nella fase di ricerca dei documenti, non si rende necessaria una ulteriore verifica dal parte di quest'ultima componente per questa funzionalità.

6.2.3. Verifica delle asserzioni per la comunicazione dei metadati

Il portafoglio di asserzioni contenente le prime tre asserzioni deve essere spedito dall'utente contestualmente alla chiamata del servizio *IMetadataMgt*, per permettere al PEP dispiegato presso questo servizio di effettuare le opportune verifiche. In particolare, per quanto concerne la verifica della coerenza dell'asserzione applicativa con il messaggio applicativo, il PEP deve controllare che l'identificativo dell'assistito indicato nell'asserzione coincide con quello presente nei metadati inoltrati. Nel caso in cui le verifiche vadano a buon fine, il PEP, dopo aver ottenuto esito positivo dal PDP, invoca il servizio *IMetadataMgt*, il quale sottopone la richiesta al registry. Il dominio che riceve i metadati da parte di un altro dominio deve verificare che esso risulta essere il dominio autorizzato a gestire la richiesta.

6.3. Struttura dei messaggi SOAP

Questo paragrafo descrive la struttura dei messaggi SOAP che deve essere rispettata per l'interazione tra un applicativo sito in un dominio ed i servizi *IQueryMgt*, *IDocument* e *IMetadataMgt* dislocati presso un altro dominio.

In conformità allo standard WS-Security, l'header di ogni messaggio SOAP deve contenere le asserzioni di sicurezza, mentre il body deve trasportare le richieste e le risposte dei servizi.

In generale, il formato dei messaggi deve rispettare le strutture riportate in Figura 9, Figura 10 e Figura 11.

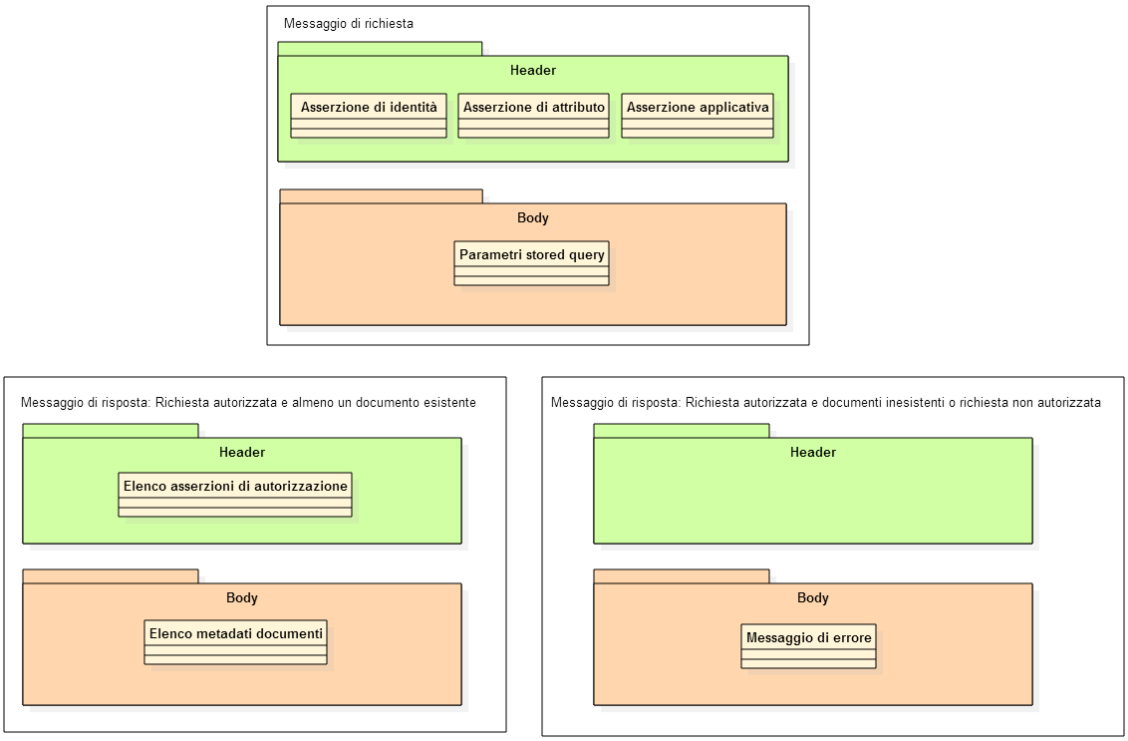

Figura 9. Struttura dei messaggi SOAP per il servizio *IQueryMgt*

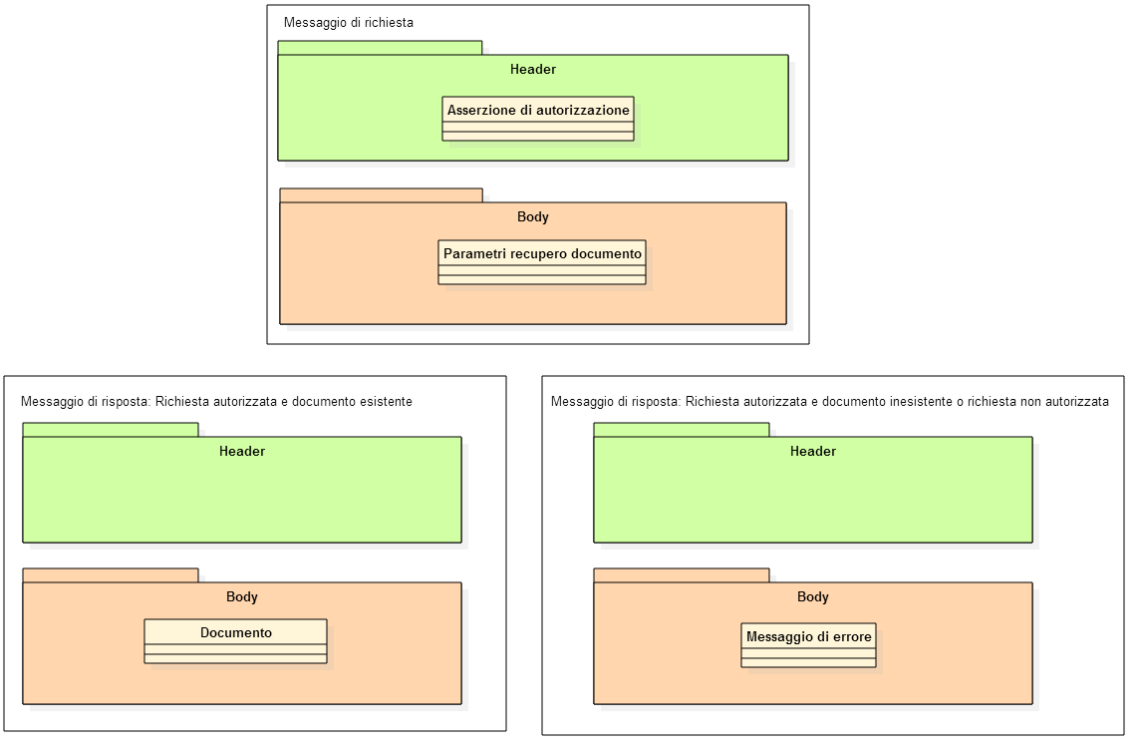

Figura 10. Struttura dei messaggi SOAP per il servizio *IDocument*

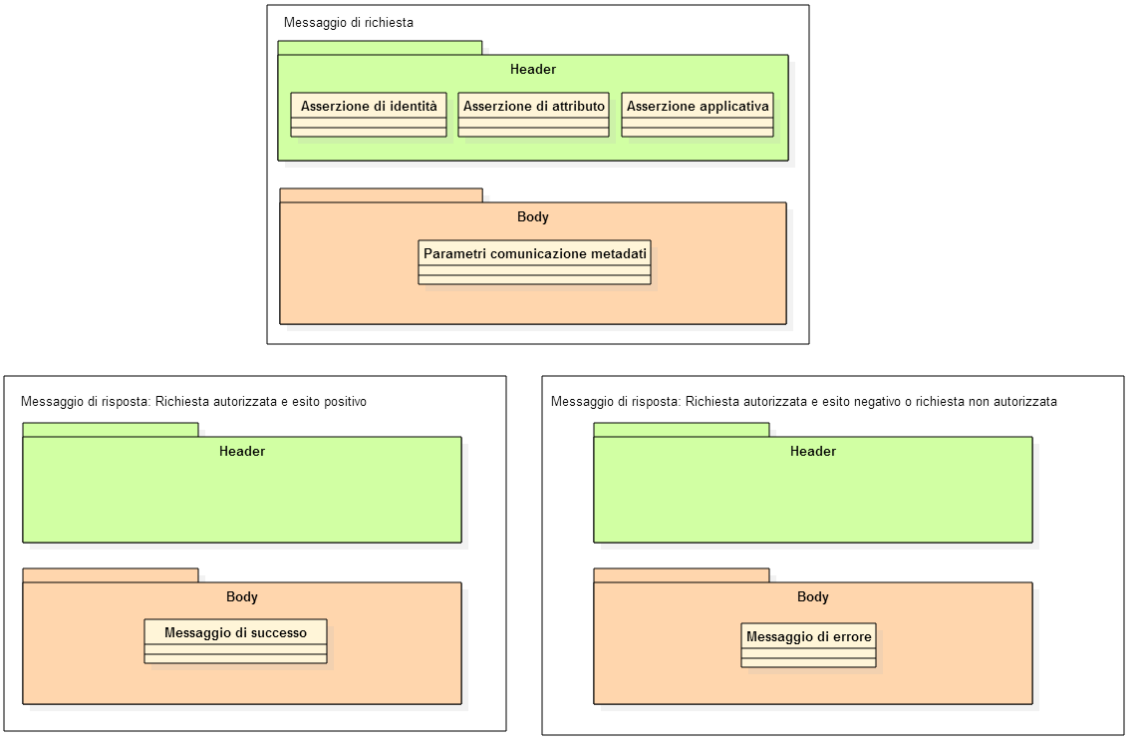

Figura 11. Struttura dei messaggi SOAP per il servizio *IMetadataMgt*

7. Implementazione dei servizi mediante le componenti OpenInFSE

Il progetto OpenInFSE ha avuto tra i suoi obiettivi lo sviluppo ed il rilascio di componenti software conformi al modello architetturale InFSE.

Allo scopo di sviluppare una versione completa delle funzionalità, sono stati individuati un registry ed un repository di riferimento. La scelta è ricaduta sull'implementazione open-source del registry/repository ebXML OMAR v3.1 rilasciata da freebXML. Il registry OMAR è utilizzato dai servizi della componente *Registro Indice Federato* e della componente *Gestore dei Documenti*.

Con riferimento alla componente *Registro Indice Federato*, i servizi permettono di memorizzare metadati in un registry, gestire il loro ciclo di vita, effettuare query ad un singolo registry o a più registry. In particolare, l'invio di query federate non comporta un'interazione diretta tra i registry, ma piuttosto tra i servizi InFSE. È possibile realizzare opportuni adapter capaci di far interagire i servizi della componente *Registro Indice Federato* con un registry preesistente. L'adapter deve permettere essenzialmente di sottoporre una query al registry secondo il modello informativo condiviso esponendo verso l'esterno l'interfaccia del servizio *IQueryMgt*. La registrazione di metadati conformi al modello informativo nel registry preesistente può essere effettuata mediante servizi preesistenti.

Con riferimento alla componente *Gestore dei Documenti*, il servizio interagisce con OMAR per la memorizzazione ed il recupero di documenti. Anche in questo caso, è possibile implementare un adapter in grado di permettere l'interazione tra la componente ed un repository preesistente.

Nel seguito del documento sono illustrate le scelte effettuate per consentire la relizzazione delle funzionalità di interoperabilità secondo quanto descritto in precedenza.

7.1. Ricerca dei documenti

La ricerca, da parte del Dominio A, di documenti presenti nel Dominio B, prevede:

- il dispiegamento del servizio *IEntry* dell'*Interfaccia di Accesso*, presso il Dominio A;
- il dispiegamento del servizio *IQueryMgt* del *Registro Indice Federato* e della componente *Gestore delle Politiche di Accesso*, presso il Dominio B.

La componente *Gestore delle Politiche di Accesso* prevede un filtro installato a monte del servizio *IQueryMgt*, in grado di verificare le asserzioni di sicurezza e autorizzare o meno l'accesso al servizio.

Di seguito sono descritte le due fasi: verifica delle asserzioni e accesso al servizio.

7.1.1. Verifica delle asserzioni

La Figura 12 mostra l'architettura delle componenti per la verifica delle asserzioni.

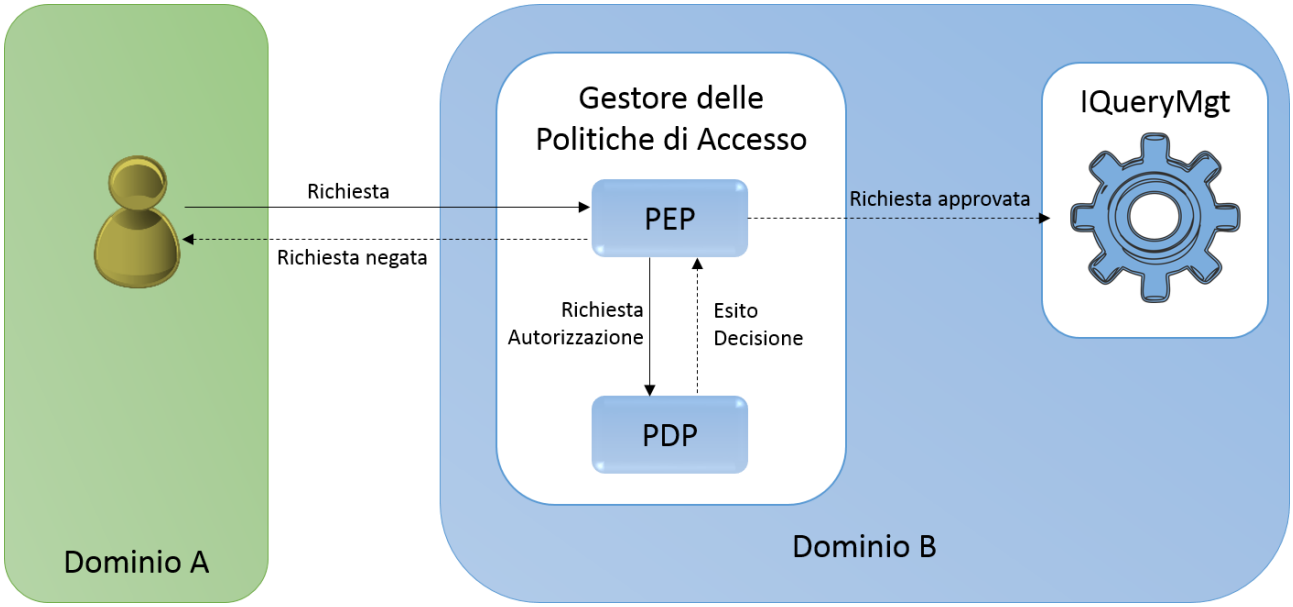

Figura 12. Verifica delle asserzioni per il servizio OpenInFSE *IQueryMgt*

Di seguito sono riportati i passi del flusso di interazione:

1. l'applicativo invia il messaggio di richiesta contenente una query al servizio *IQueryMgt*, nel cui header è presente il portafoglio delle asserzioni;
2. il messaggio è intercettato dalla componente PEP, che effettua i seguenti controlli:
    a. **controllo asserzione di identità:** la componente verifica che sia presente l'attributo previsto nell'asserzione di identità;
    b. **controllo asserzione di attributo:** la componente verifica che siano presenti gli attributi previsti nell'asserzione di attributo;
    c. **controllo asserzione applicativa:** la componente verifica che siano presenti gli attributi previsti nell'asserzione applicativa, che il ruolo presente nell'attributo *SubjectRole* e che il purpose of use presente nell'attributo *SubjectPurposeOfUse* siano compresi tra i valori ammissibili. Il PEP deve inoltre verificare la corrispondenza tra l'identificativo dell'assistito, presente nell'attributo *ResourceId* nell'asserzione applicativa, con l'identificativo dell'assistito presente nel body del messaggio;
    d. **controllo firma asserzioni:** la componente verifica la correttezza delle firme digitali delle asserzioni;
    e. **controllo consenso:** la componente verifica se l'assistito ha fornito il consenso necessario per l'accesso al documento. Questa verifica è strettamente dipendente dal sistema infrastrutturale e dalle politiche adottate dai singoli domini;
3. il PEP, effettuati i controlli, inoltra la richiesta al PDP, che, in funzione delle politiche di accesso basate sullo standard XACML, permette che tale richiesta sia inoltrata o meno al servizio *IQueryMgt*;
4. in caso di autorizzazione fornita, il servizio *IQueryMgt* risponde alla richiesta con un messaggio contenente le informazioni necessarie per il recupero del documento;
5. il PEP restituisce un messaggio, nel cui header sono presenti le asserzioni di autorizzazione per l'accesso ai documenti e nel cui body vi è la risposta ricevuta dal servizio *IQueryMgt.*

7.1.2. Accesso al servizio per l'esecuzione della query

Dopo l'analisi delle asserzioni di sicurezza, il PEP inoltra la richiesta al servizio *IQueryMgt*.

Nel caso di ricerca diretta, i passi da eseguire per l'invio di una query da parte di un applicativo nel Dominio A ad un servizio *IQueryMgt* del Dominio B, come mostrato in Figura 13, sono i seguenti:

1. un utente del Dominio A sottopone una query al servizio *IEntry* del proprio nodo, specificando il riferimento al servizio *IQueryMgt* del Dominio B da invocare;
2. il servizio *IEntry* del Dominio A propaga la query al servizio *IQueryMgt* del Dominio B;
3. il servizio *IQueryMgt* del Dominio B sottopone la query al proprio registry;
4. il registry fornisce i risultati al servizio *IQueryMgt* del Dominio B;
5. il servizio *IQueryMgt* del Dominio B restituisce i risultati al servizio *IEntry* del Dominio A;
6. il servizio *IEntry* del Dominio A fornisce l'elenco dei risultati all'utente.

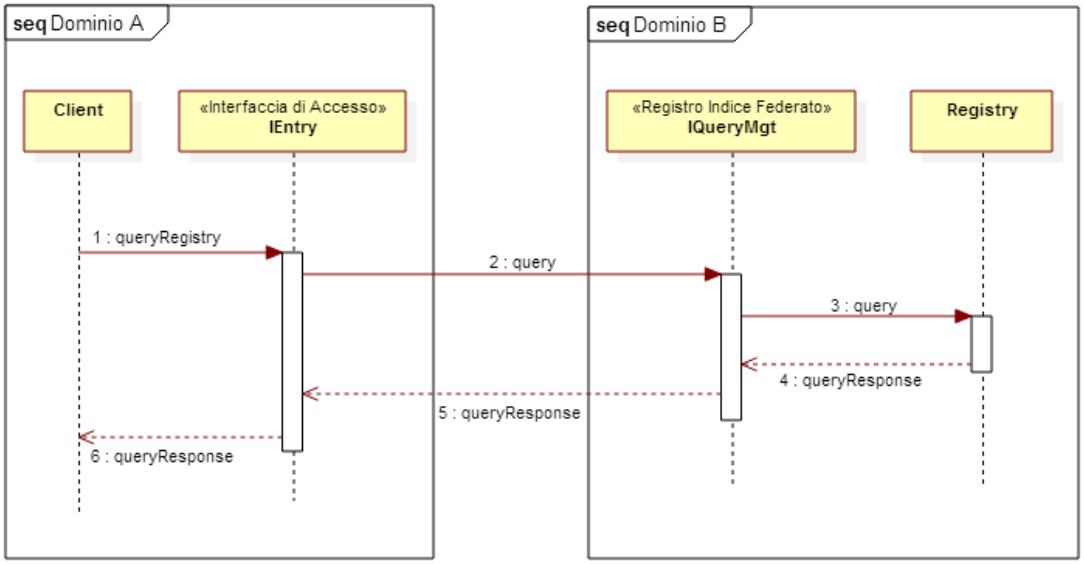

Figura 13. Ricerca diretta medianti i servizi OpenInFSE

Nel caso di query federata, i passi da eseguire per l'invio di una query da parte di un applicativo nel Dominio A verso i servizi *IQueryMgt* che fanno parte della federazione, come mostrato in Figura 14, sono i seguenti:

1. un utente del Dominio A sottopone una query federata al servizio *IEntry* del proprio nodo;
2. il servizio *IEntry* del Dominio A sottopone la query federata al servizio *IQueryMgt* del Dominio A;
3. il servizio *IQueryMgt* del Dominio A richiede al proprio registry l'elenco dei riferimenti ai servizi *IQueryMgt* degli altri domini facenti parte della federazione;
4. il registry fornisce al servizio *IQueryMgt* del Dominio A l'elenco dei riferimenti;
5. il servizio *IQueryMgt* del Dominio A propaga la query (in modalità non federata) ai servizi *IQueryMgt* degli altri domini;
6. i servizi *IQueryMgt* degli altri domini sottopongono la query al proprio registry;
7. i registry forniscono i risultati ai servizi *IQueryMgt* corrispondenti;
8. i servizi *IQueryMgt* degli altri domini offrono i risultati al servizio *IQueryMgt* del Dominio A;

9. il servizio *IQueryMgt* del Dominio A sottopone la query al proprio registry;
10. il registry fornisce i risultati al servizio *IQueryMgt* del Dominio A;
11. il servizio *IQueryMgt* del Dominio A aggrega tutti i risultati;
12. il servizio *IQueryMgt* del Dominio A fornisce i risultati al servizio *IEntry* del Dominio A;
13. il servizio *IEntry* del Dominio A restituisce l'elenco dei risultati all'utente.

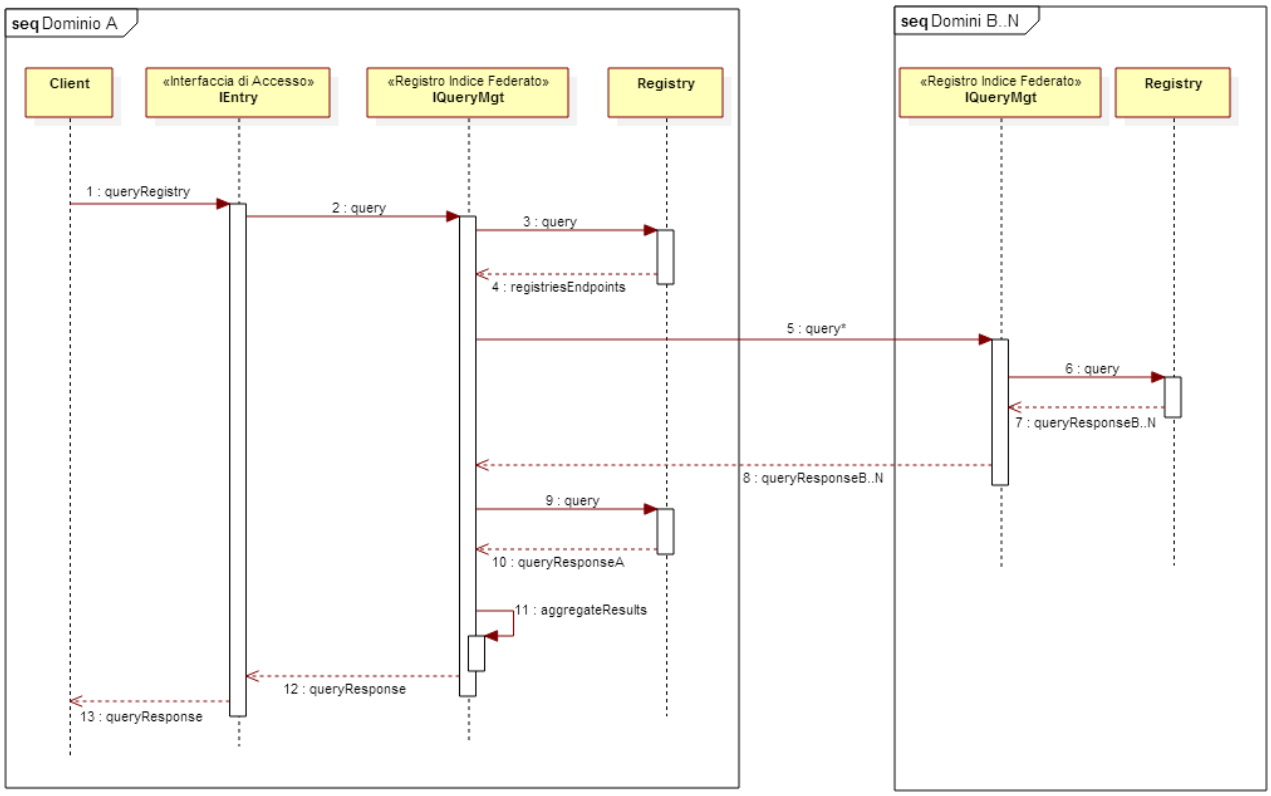

Figura 14. Ricerca federata medianti i servizi OpenInFSE

7.2. Recupero di un documento

Il recupero di un documento disponibile nel Dominio B da parte del Dominio A, prevede:

- il dispiegamento del servizio *IDocument* dell'*Interfaccia di Accesso*, presso il Dominio A;
- il dispiegamento del servizio *IDocument* dell'*Interfaccia di Accesso*, della componente *Gestore delle Politiche di Accesso* e del servizio *IDocumentMgt* della componente *Gestore dei Documenti*, presso il Dominio B.

La componente *Gestore delle Politiche di Accesso* prevede un filtro installato a monte del servizio *IDocument* del Dominio B, in grado di verificare le asserzioni di sicurezza e autorizzare o meno l'accesso al servizio.

Di seguito sono descritte le due fasi: verifica delle asserzioni e accesso al servizio.

7.2.1. Verifica delle asserzioni

La Figura 15 mostra l'architettura delle componenti per la verifica delle asserzioni.

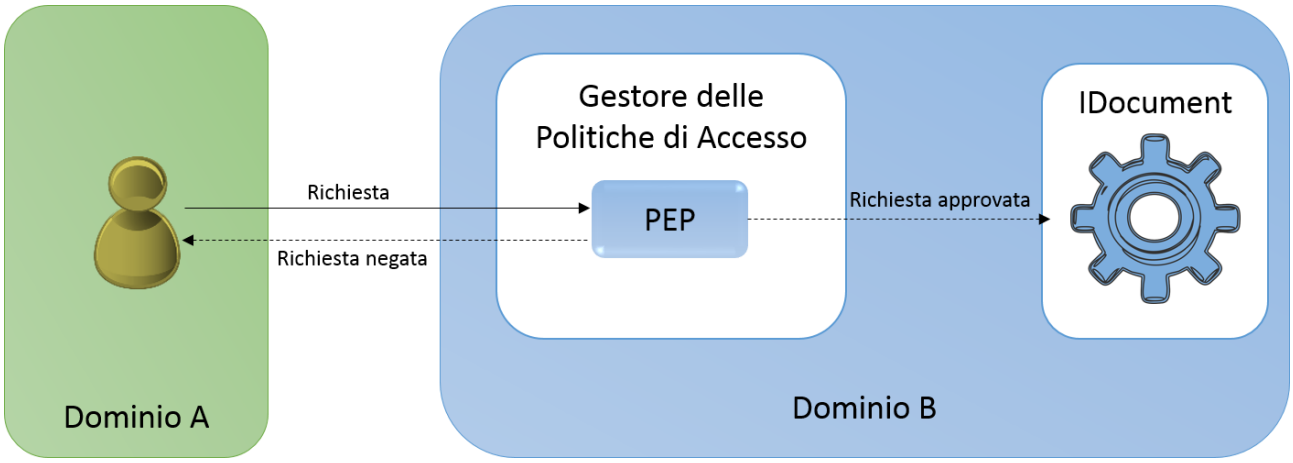

Figura 15. Verifica delle asserzioni per il servizio OpenInFSE *IDocument*

In maniera analoga a quanto descritto per la verifica delle asserzioni per la ricerca dei documenti, di seguito sono descritti i principali passi del flusso di interazione per il recupero di un documento:

1. l'applicativo invia il messaggio di richiesta di recupero di un documento al servizio *IDocument*, nel cui header è presente l'asserzione di autorizzazione;
2. il messaggio è intercettato dalla componente PEP, che effettua i seguenti controlli:
    a. **controllo asserzione di autorizzazione:** la componente verifica che siano presenti gli attributi previsti dall'asserzione di autorizzazione e la corrispondenza tra l'identificativo del documento, presente nell'asserzione, con l'identificativo del documento presente nell'elemento *documentID* del body del messaggio;
    b. **controllo firma asserzioni:** la componente verifica la correttezza della firma digitale dell'asserzione;
    c. **controllo consenso:** la componente verifica se l'assistito ha fornito il consenso necessario per l'accesso al documento. Questa verifica è strettamente dipendente dal sistema infrastrutturale e dalle politiche adottate dai singoli domini;
3. il PEP, effettuati i controlli, permette che tale richiesta venga inoltrata o meno al servizio *IDocument*;
4. in caso di autorizzazione fornita, il servizio *IDocument* risponde alla richiesta con un messaggio contenente il risultato (il documento o un messaggio di errore);
5. il PEP restituisce un messaggio contenente nel body la risposta ricevuta dal servizio *IDocument.*

7.2.2. Accesso al servizio per il recupero di un documento

Dopo l'analisi delle asserzioni di sicurezza, il PEP inoltra la richiesta al servizio *IDocument*. In particolare, il servizio *IDocument* interagisce con il servizio *IDocumentMgt* in grado di interfacciarsi con un repository OMAR. Una soluzione alternativa consiste nell'implementare un servizio, conforme all'interfaccia *IDocument*, in grado di interfacciarsi direttamente con un repository preesistente secondo i meccanismi dipendenti dalla logica interna del dominio.

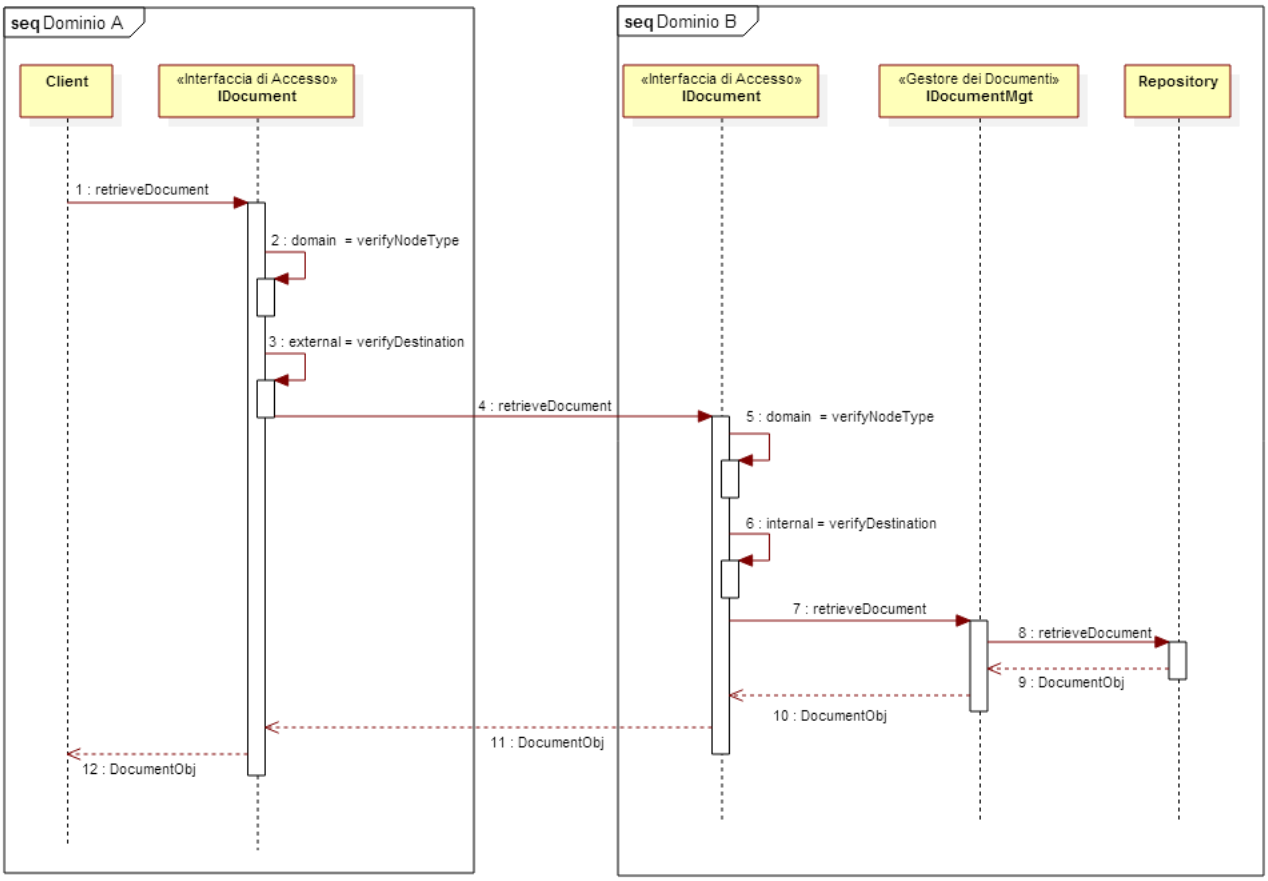

Figura 16. Recupero di un documento medianti i servizi OpenInFSE

Di seguito sono descritti i passi da eseguire per la richiesta di recupero di un documento da parte di un applicativo del Dominio A al servizio *IDocument* del Dominio B, come mostrato in Figura 16:

1. un utente del Dominio A invoca il servizio *IDocument* del proprio nodo specificando una tripla di informazioni (riferimento al dominio contenente il documento, riferimento al repository contenente il documento, identificativo del documento), ottenuta a valle della query;
2. il servizio *IDocument* del Dominio A verifica che è dislocato presso il nodo di dominio (e non presso un nodo di livello inferiore);
3. il servizio *IDocument* del Dominio A verifica che il documento richiesto è disponibile in un altro dominio (analizzando il riferimento del dominio contenente il documento);
4. il servizio *IDocument* del Dominio A invoca il servizio *IDocument* del Dominio B contenente il documento al riferimento specificato, opportunamente risolto;
5. il servizio *IDocument* del Dominio B verifica che è dislocato presso il nodo di dominio (e non presso un nodo di livello inferiore);
6. il servizio *IDocument* del Dominio B verifica che il documento richiesto è disponibile presso il proprio dominio analizzando il riferimento del dominio contenente il documento;
7. il servizio *IDocument* del Dominio B invoca il servizio *IDocumentMgt* al riferimento specificato, risolto a partire dal riferimento al repository;
8. il servizio *IDocumentMgt* richiede il documento (individuato dall'identificativo specificato) al repository;
9. il repository fornisce al servizio *IDocumentMgt* il documento richiesto, se disponibile;

10. il servizio *IDocumentMgt* restituisce il risultato al servizio *IDocument* del Dominio B;
11. il servizio *IDocument* del Dominio B restituisce il risultato al servizio *IDocument* del Dominio A;
12. il servizio *IDocument* del Dominio A fornisce il risultato all'utente.

7.3. Comunicazione dei metadati

La comunicazione dei metadati da parte del Dominio A al Dominio B prevede:

- il dispiegamento del servizio *IEntry* dell'*Interfaccia di Accesso*, presso il Dominio A;
- il dispiegamento del servizio *IMetadataMgt* del *Registro Indice Federato* e della componente *Gestore delle Politiche di Accesso*, presso il Dominio B.

La componente *Gestore delle Politiche di Accesso* prevede un filtro installato a monte del servizio *IMetadataMgt*, in grado di verificare le asserzioni di sicurezza e autorizzare o meno l'accesso al servizio.

Di seguito sono descritte le due fasi: verifica delle asserzioni e accesso al servizio.

7.3.1. Verifica delle asserzioni

La Figura 17 mostra l'architettura delle componenti per la verifica delle asserzioni.

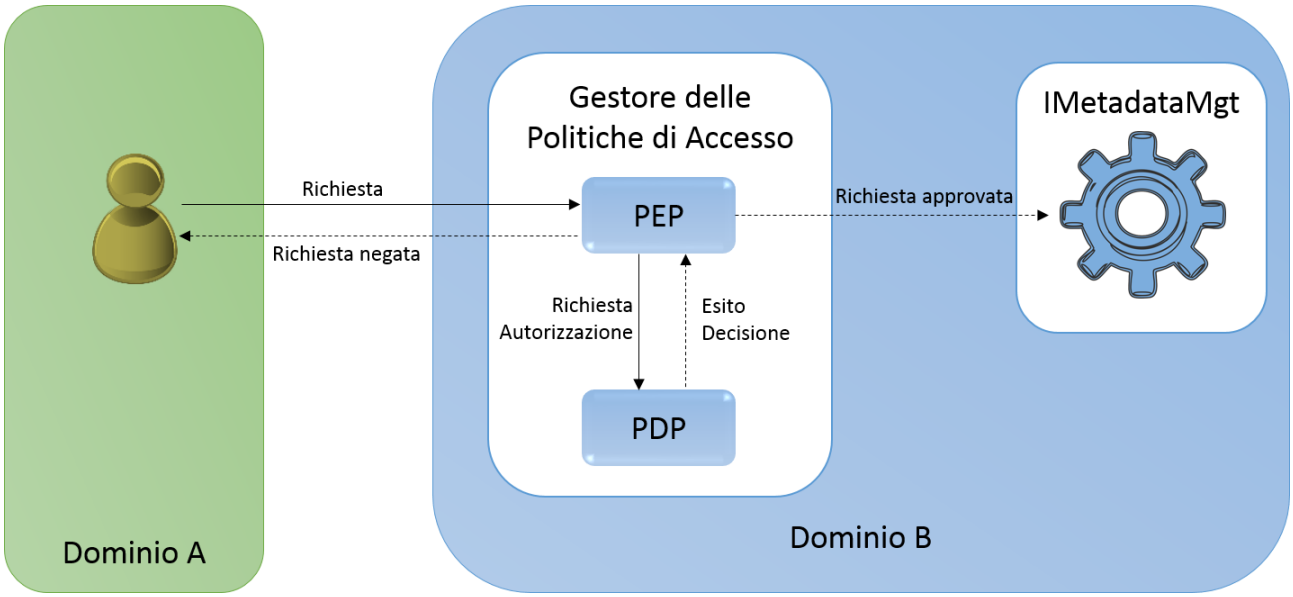

Figura 17. Verifica delle asserzioni per il servizio OpenInFSE *IMetadataMgt*

Di seguito sono riportati i passi del flusso di interazione:

1. l'applicativo invia il messaggio di richiesta relativa ai metadati al servizio *IMetadataMgt*, nel cui header è presente il portafoglio delle asserzioni;
2. il messaggio è intercettato dalla componente PEP, che effettua i seguenti controlli:
    a. **controllo asserzione di identità:** la componente verifica che sia presente l'attributo previsto nell'asserzione di identità;

b. **controllo asserzione di attributo:** la componente verifica che siano presenti gli attributi previsti nell'asserzione di attributo;

c. **controllo asserzione applicativa:** la componente verifica che siano presenti gli attributi previsti nell'asserzione applicativa, che il ruolo presente nell'attributo *SubjectRole* e che il purpose of use presente nell'attributo *SubjectPurposeOfUse* siano compresi tra i valori ammissibili. Il PEP deve inoltre verificare la corrispondenza tra l'identificativo dell'assistito, presente nell'attributo *ResourceId* nell'asserzione applicativa, con l'identificativo dell'assistito presente nel body del messaggio;

d. **controllo firma asserzioni:** la componente verifica la correttezza delle firme digitali delle asserzioni;

e. **controllo consenso:** la componente verifica se l'assistito ha fornito il consenso necessario per l'alimentazione del FSE. Questa verifica è strettamente dipendente dal sistema infrastrutturale e dalle politiche adottate dai singoli domini;

3. il PEP, effettuati i controlli, inoltra la richiesta al PDP, che, in funzione delle politiche di accesso basate sullo standard XACML, permette che tale richiesta sia inoltrata o meno al servizio *IMetadataMgt*;
4. in caso di autorizzazione fornita, il servizio *IMetadataMgt* risponde alla richiesta con un messaggio contenente un messaggio di successo o di errore;
5. il PEP restituisce un messaggio, nel cui body vi è la risposta ricevuta dal servizio *IMetadataMgt.*

7.3.2. Accesso al servizio per la memorizzazione dei metadati

Dopo l'analisi delle asserzioni di sicurezza, il PEP inoltra la richiesta al servizio *IMetadataMgt*. Di seguito sono descritti i passi da eseguire per la trasmissione di metadati da parte di un applicativo nel Dominio A ad un servizio *IMetadataMgt* del Dominio B, come mostrato in Figura 18:

1. un utente del Dominio A sottopone una richiesta al servizio *IEntry* del proprio nodo, specificando il riferimento al servizio *IMetadataMgt* del Dominio B da invocare;
2. il servizio *IEntry* del Dominio A propaga la richiesta al servizio *IMetadataMgt* del Dominio B;
3. il servizio *IMetadataMgt* del Dominio B memorizza i metadati nel proprio registry;
4. il registry fornisce un messaggio al servizio *IMetadataMgt* del Dominio B;
5. il servizio *IMetadataMgt* del Dominio B restituisce il messaggio al servizio *IEntry* del Dominio A;
6. il servizio *IEntry* del Dominio A fornisce il messaggio all'utente.

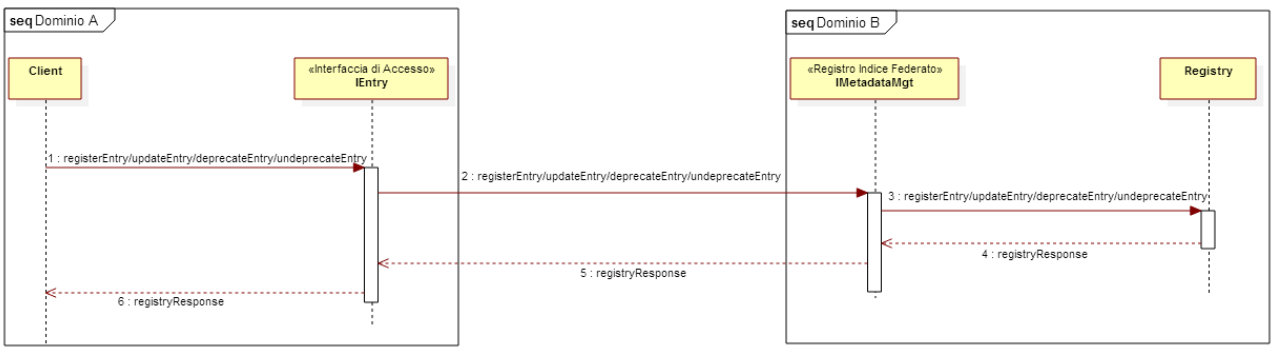

Figura 18. Trasmissione di metadati medianti i servizi OpenInFSE

8. Conclusioni

Questo rapporto tecnico ha illustrato un insieme di servizi opportunamente progettati a supporto di funzionalità di ricerca di documenti sanitari, recupero di un documento sanitario e comunicazione di metadati relativi ad un documento sanitario in regime di interoperabilità tra sistemi di FSE di domini differenti. Allo scopo, è stata utilizzata l'infrastruttura tecnologica InFSE, elaborata nell'ambito di progetti congiunti tra il DDI ed il CNR e approvata dal TSE. Per ciascun servizio, il rapporto tecnico descrive il modello funzionale, gli scenari di interazione ed il modello tecnico. Particolare importanza è stata fornita agli aspetti inerenti alla sicurezza delle informazioni scambiate, garantita mediante la conformità agli standard OASIS SAML e XACML. Infine, i servizi progettati sono stati realizzati a partire dalle componenti OpenInFSE, che costituiscono l'implementazione di riferimento dell'infrastruttura InFSE.

Riferimenti

[1] DDI – CNR, Linee guida e specifiche tecniche dell'infrastruttura tecnologica InFSE

[2] SPC, http://www.agid.gov.it/agenda-digitale/infrastrutture-architetture/sistema-pubblico-connettivita

[3] OASIS ebXML Registry V3.0, http://docs.oasis-open.org/regrep/v3.0/regrep-3.0-os.zip

[4] OASIS SAML V2.0, https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security

[5] OASIS XACML V3.0, https://www.oasis-open.org/committees/xacml/