



Registro SPICCA

Architettura del Software

Versione 1.0 del 25/08/2009

Sommario

1	Introduzione	4
1.1	Scopo	4
1.2	Obiettivo	4
1.3	Riferimenti	4
1.4	Panoramica del documento.....	4
2	Rappresentazione dell'Architettura	6
3	Descrizione della architettura	8
3.1	Schema a livelli	8
4	Logical View	10
4.1	Design Packages significativi per l'Architettura	11
4.1.1	Registry.....	11
4.1.2	Gestore Registry.....	11
4.1.3	Web Application.....	11
4.1.4	Web Services	11
4.1.5	IPA	11
4.1.6	Configuratore di piattaforma	11
4.1.7	Persistence	12
4.1.8	Logger.....	12
5	Use-Case View	13
5.1	Realizzazione degli Use-Case	13
5.1.1	UC08 – InserimentoSoggettoSPCOOP.....	13
5.1.2	UC01-Inserimento AS	14
5.1.3	UC02-InserimentoParteSpecificaAS.....	15
5.1.4	UC20-ImportAS	16
5.1.5	UC23-ExportAS.....	17
6	Deployment View	19
7	Implementation View	20
7.1	Layers.....	21
7.1.1	BackEnd.....	21
7.1.2	FrontEnd.....	22
8	Data View	23

IL CONTENUTO DEL PRESENTE DOCUMENTO PUÒ ESSERE RIPRODOTTO, IN TOTO O IN PARTE, PER TUTTI GLI SCOPI FUNZIONALI ALL'ADESIONE AL SISTEMA SPICCA DI REGIONE CAMPANIA ED È ESCLUSO L'UTILIZZO A FINI DI LUCRO. PER L'UTILIZZO DI QUANTO DI SEGUITO RIPORTATO SI DOVRÀ, IN TUTTI I CASI, CITARE COME FONTE IL PRESENTE DOCUMENTO.

1 Introduzione

Questo documento è stato realizzato in collaborazione con Vitrociset S.p.a. in quanto ditta aggiudicataria dell'appalto concorso per la realizzazione del "Sistema Regionale per la Cooperazione Applicativa in Sicurezza" della Regione Campania.

Il disciplinare tecnico [BG/DT] relativo al "Sistema Regionale per la cooperazione applicativa in Sicurezza" definisce l'infrastruttura di riferimento per ottenere funzionalità di cooperazione tra servizi offerti da diversi enti della Regione Campania, preservando l'autonomia e la peculiarità dei sistemi interconnessi. Tale infrastruttura consta di diversi componenti, le cui funzionalità sono state esaminate nel documento "SPICCA – Registro SPICCA - Requisiti del software" [SPICCA-REGSPICCA-RU]. Oggetto del presente documento è l'individuazione dell'architettura delle componenti basilari per il Registro SPICCA.

1.1 Scopo

Il presente documento si inserisce nell'ambito della documentazione relativa al progetto SPICCA e costituisce la specifica degli elementi strutturali costituenti i componenti del Registro SPICCA.

1.2 Obiettivo

Obiettivo del presente documento è quello di individuare gli elementi strutturali del Registro SPICCA, specificandone le interfacce e le modalità di collaborazione, in conformità alle specifiche tecniche emesse dal CNIPA relativamente al Sistema Pubblico di Connettività e Cooperazione (SPCOOP).

1.3 Riferimenti

1. [BG] Bando di gara: appalto concorso per l'acquisizione del "Sistema Regionale per la Cooperazione Applicativa in Sicurezza" della Regione Campania – BURC 30/04/2004
2. [BG/DT] Bando di gara: appalto concorso per l'acquisizione del "Sistema Regionale per la Cooperazione Applicativa in Sicurezza" della Regione Campania – BURC 30/04/2004 – Disciplinare Tecnico
3. [SPCOOP-ARC] Sistema Pubblico di Cooperazione: Architettura. Versione 1.0, 14/10/2005
4. [SPCOOP-SPICCA] Sistema Pubblico di Cooperazione: Servizi di Registro. Versione 1.0, 14/10/2005.
5. [SPICCA-REGSPICCA-RU] SPICCA – Registro SPICCA – Requisiti del software, Versione 1.1, 6/07/2006.
6. [SPICCA-REGSPICCA-UC] SPICCA – Registro SPICCA – Specifiche degli Use Case, Versione 1.1, 6/07/2006.
7. [SPCOOP-AS2009] Struttura dell'Accordo di Servizio e dell'Accordo di Cooperazione. Versione 1.0, 05/03/2009.

1.4 Panoramica del documento



Il presente documento riporta lo schema architeturale del Registro SPICCA le specifiche SPCOOP dettate dallo CNIPA.

2 Rappresentazione dell'Architettura

Per la rappresentazione dell'architettura si seguiranno le raccomandazioni del Rational Unified Process (RUP) come dettato dalle direttive aziendali relative al ciclo di vita del software.

Il processo RUP suggerisce la rappresentazione di un'architettura software tramite diverse **viste architetturali**, ognuna delle quali pone enfasi su di un particolare aspetto dell'architettura considerata. In particolare, il predetto processo prevede le viste architetturali riportate in Figura 1 e di seguito brevemente descritte:

- **logical view** (vista logica): rappresenta l'organizzazione concettuale del sistema in termini di *layer*, *componenti*, *subsystem* e *package*, ed evidenzia le funzionalità dei principali componenti e subsystem.
- **implementation view** (vista implementativa): esprime il cosiddetto "Modello Implementativo" del RUP, ovvero rappresenta l'organizzazione del sistema in componenti "fisici", come, a titolo esemplificativo, l'organizzazione di classi in package Java, l'organizzazione di bytecode Java in file ".jar".
- **use-case view** (vista dei casi d'uso): rappresenta i casi d'uso più importanti dal punto di vista architetturale, ovvero i casi d'uso la cui realizzazione coinvolge una parte significativa dei componenti architetturali individuati;
- **process view** (vista dei processi): evidenzia i processi ed i thread costituenti il sistema software, riportando le loro responsabilità, le modalità di collaborazione;
- **deployment view** (vista di dispiegamento): rappresenta il deployment dei processi e componenti costituenti il sistema in considerazione su nodi fisici d'elaborazione.

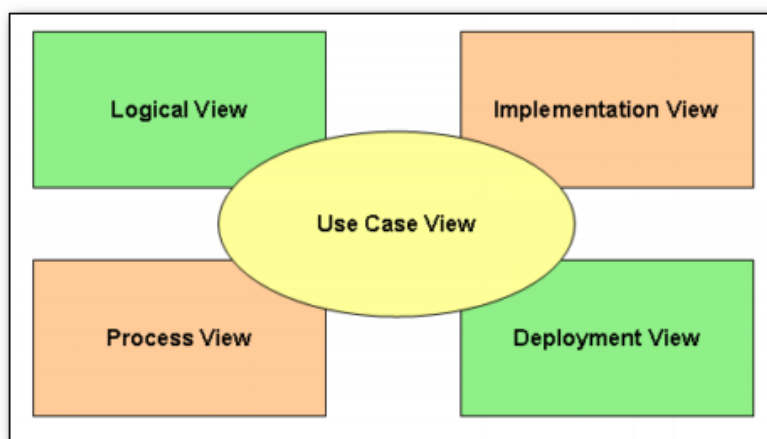


Figura 1: le 4+1 viste architetturali (P. Kruchten 1995)

Relativamente all'architettura del software del componente Registro SPICCA, nel proseguo del documento si riporteranno le viste Logical View, Use Case View, Implementation View, Deployment View e Data View.

3 Descrizione della architettura

Ci si pone l'obiettivo di definire un'architettura del componente Registro SPICCA che soddisfi, in conformità alle specifiche riportate nel documento [SPCOOP-SPICCA], i requisiti funzionali riportati nel documento [SPICCA-REGSPICCA-RU] e, nel contempo, i requisiti di affidabilità e prestazioni riportati nella sezione "Requisiti non funzionali" del documento [SPICCA-REGSPICCA-RU].

3.1 Schema a livelli

Di seguito, in Figura 2, viene presentato un modello dell'architettura del Registro SPICCA. In tale modello si evidenzia la presenza di una logica applicativa che possa consentire la gestione della persistenza dei dati su un DBMS con estensioni/supporto per file XML (la cui architettura interna è dettagliata in Figura 3), la necessità di esporre due tipologie di interfacce verso l'esterno ed, inoltre, inglobare l'elemento esistente IPA (Indice delle Pubbliche Amministrazioni) con una logica di sincronizzazione fra i dati del registro SPICCA e quelli dell'IPA stesso.

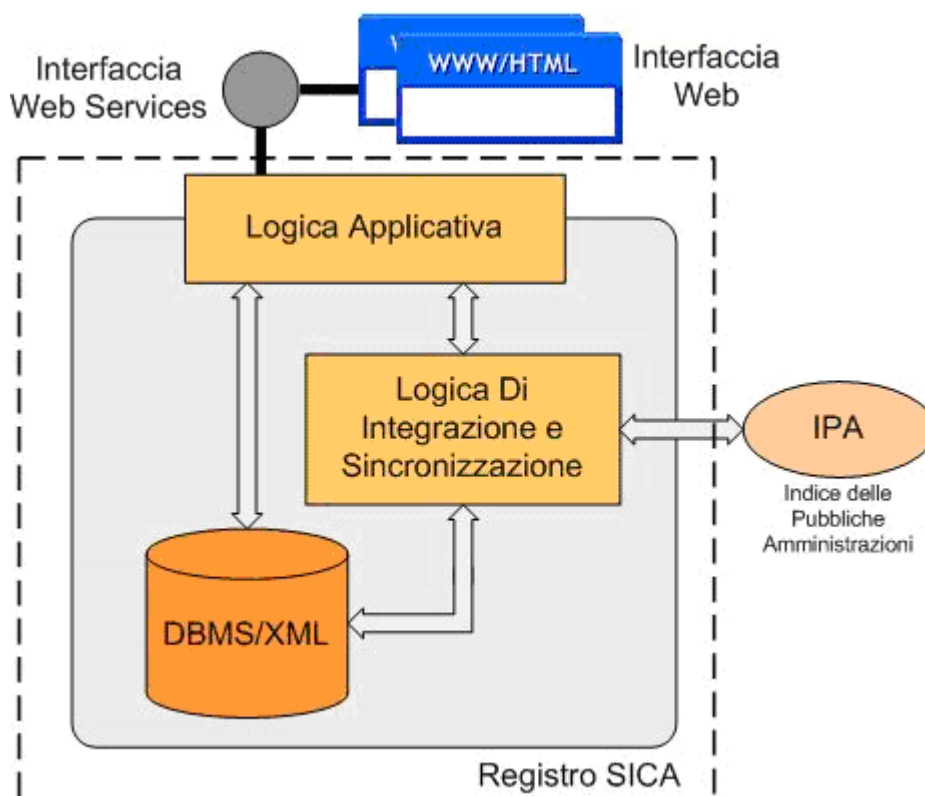


Figura 2: Architettura Registro SPICCA

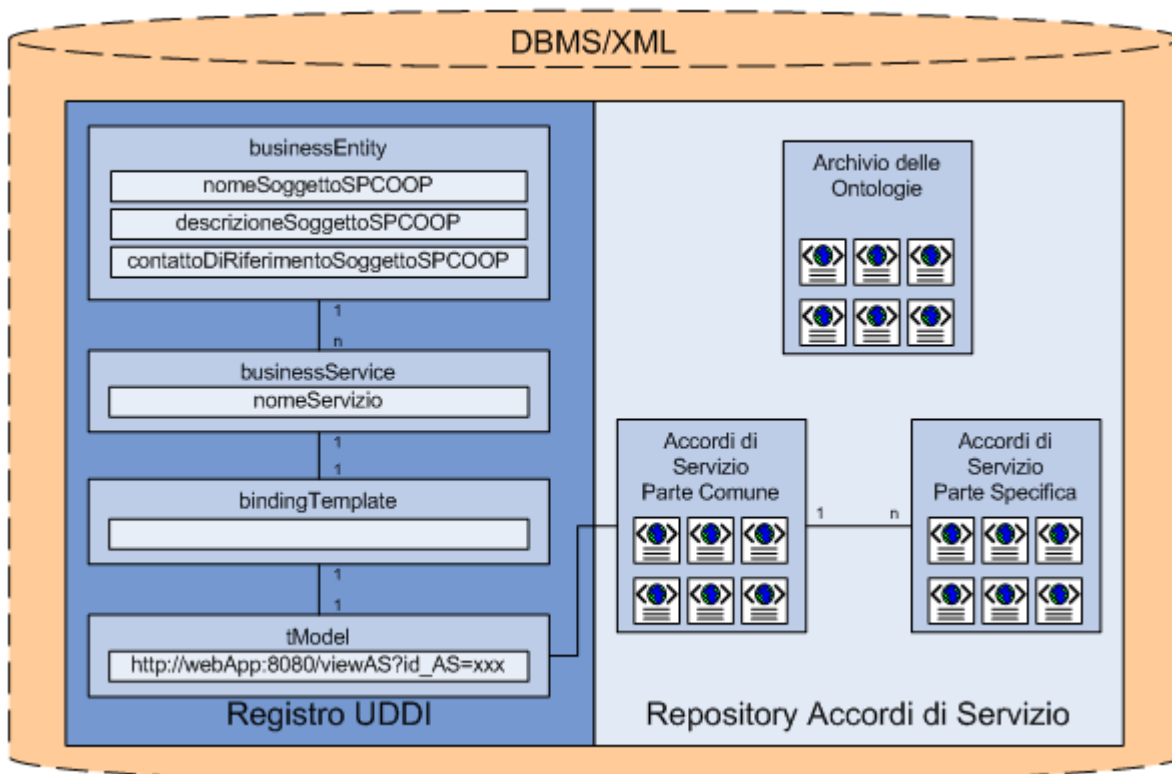


Figura 3: Architettura interna DBMS/XML

L'architettura interna del DBMS/XML evidenzia l'uso di un registro UDDI che indica un insieme di oggetti descritti in xml e accessibili via http.

4 Logical View

Come si evince dal component diagram riportato in Figura 4, il registro SPICCA è organizzato come segue:

- Sottosistema denominato “Registry”, composto dai componenti:
 - UDDI;
 - AccordiServizio;
- Componente denominato “Gestore Registry”;
- Componente “Web Application”;

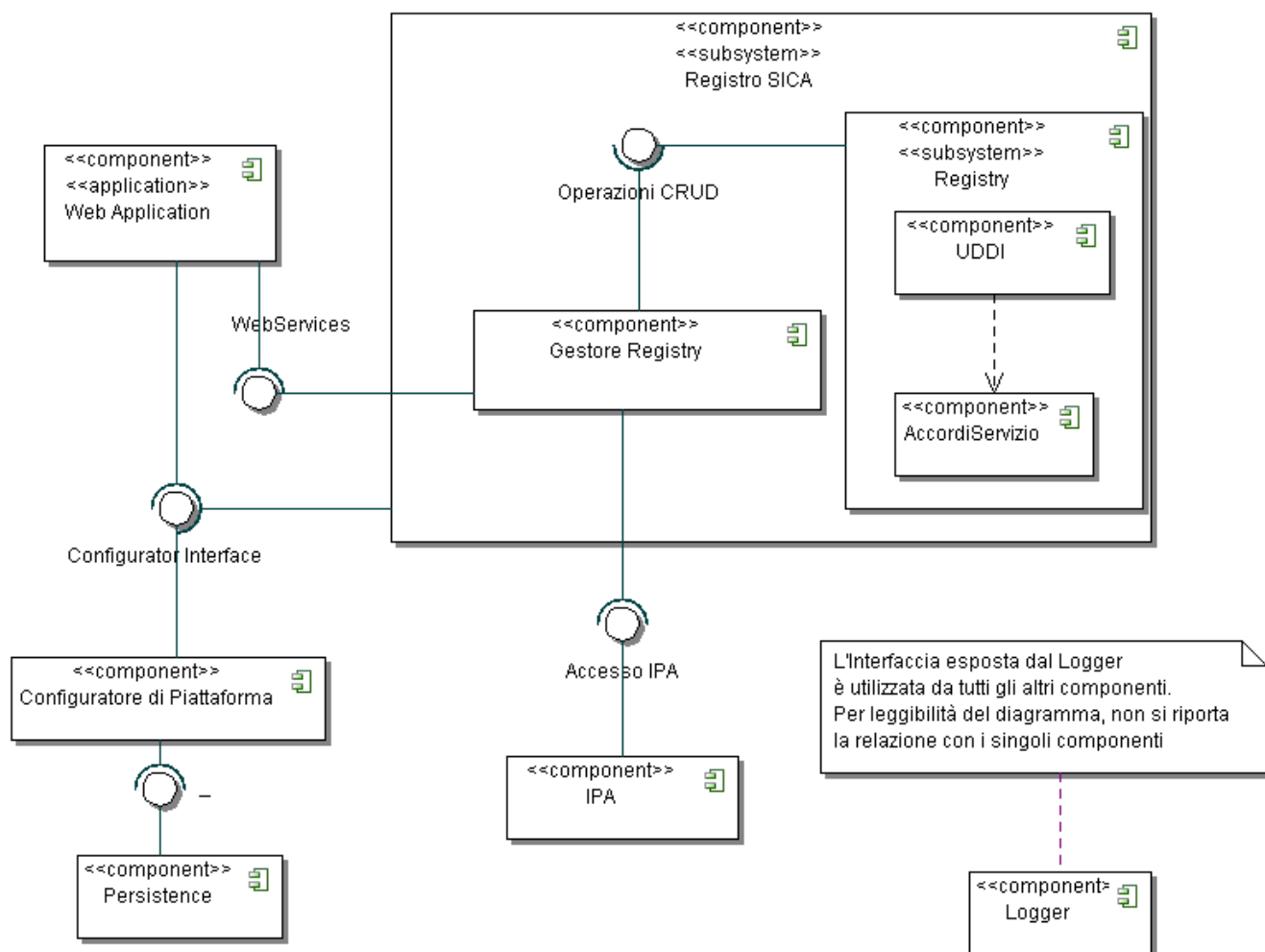


Figura 4: Component Diagram Registro SPICCA

4.1 Design Packages significativi per l'Architettura

In questa sezione saranno descritte le funzionalità dei componenti individuati.

4.1.1 Registry

Il sottosistema *Registry* è la parte del registro SPICCA di basso livello che si occupa della logica per le operazioni elementari (Create, Read, Update e Delete) di accesso ed interrogazione delle informazioni relative agli *Accordi di Servizio* ed ai *SoggettiSPCoop*.

Esso è composto dal componente *UDDI*, che è un vero e proprio registro UDDI, e dal componente *AccordiServizio* che estende le funzionalità UDDI per la realizzazione del registro SPICCA memorizzando le informazioni specifiche su un repository secondo quanto descritto dallo CNIPA per i servizi di Registro SPICCA [SPCOOP-SPICCA].

4.1.2 Gestore Registry

Il componente *Gestore Registry* è il componente che si occupa delle operazioni di gestione del Registro SPICCA. Pertanto sarà il responsabile della logica implementativa che permetterà la corretta cooperazione (e coerenza) delle operazioni di CRUD in termini di interfacciamento sia verso il componente *UDDI* che verso quello *AccordiServizio*.

4.1.3 Web Application

Il componente *Web Application* è l'applicazione web che consente ad un utente di accedere alle funzionalità del Registro SPICCA.

4.1.4 Web Services

L'interfaccia *Web Services* è il dispiegamento come web services delle funzionalità offerte dal Registro SPICCA. Dipenderà, quindi, dalle operazioni esposte dal componente *GestoreRegistry*.

4.1.5 IPA

Il componente *IPA* (Indice delle Pubbliche Amministrazioni), esterno al registro SPICCA, è un componente preesistente utilizzato dal Registro SPICCA per consentire di mantenere consistenti ed allineate le informazioni relative ai *SoggettiSPCoop*, oltre che garantire l'identità degli stessi.

4.1.6 Configuratore di piattaforma

Questo componente si occupa della gestione di tutti i parametri di configurazione di piattaforma del Registro SPICCA. Il Configuratore di piattaforma legge tutti i parametri di configurazione presenti nella base dati e li carica in memoria allo startup dell'application server. Essendo implementato secondo il pattern Singleton, sarà presente, per ogni istanza di Virtual Machine Java, un'unica istanza di configuratore a cui tutti gli altri componenti di sistema potranno attingere.

4.1.7 Persistence

Questo componente si occupa della gestione dei dati persistenti mascherando ai componenti utilizzatori il meccanismo di fatto utilizzato per la gestione dei dati persistenti.

4.1.8 Logger

Questo componente espone delle API per tutte le operazioni di logging e tracing. Il Logger astrae il logger Log4J offrendo una serie di helper per un log facilitato a tutti i componenti di piattaforma. Attraverso il configuratore di piattaforma, sarà possibile utilizzare un oggetto LoggerUtil che fornirà un riferimento al logger desiderato. Tutti i logger saranno configurati in base dati e caricati in memoria dal configuratore di sistema.

5 Use-Case View

In questo paragrafo vengono illustrati i casi d'uso, per il Registro SPICCA, più importanti dal punto di vista architetturale; ovvero i casi d'uso la cui realizzazione coinvolge una parte significativa dei componenti architetturali individuati. I requisiti che hanno portato alla loro identificazione sono riportati nel documento "Requisiti del Software" ([SPICCA- REGSPICCA -RU]).

Gli Use Case Diagram (UCD) descrivono il comportamento del sistema dal punto di vista di un osservatore esterno al sistema stesso. Nei casi d'uso, l'attenzione è centrata su quello che il sistema fa, piuttosto che sul come lo realizza.

Un attore rappresenta persone o sistemi (terzi rispetto quello in analisi) che interagiscono con il sistema che si sta descrivendo, scatenando gli eventi coinvolti nel caso d'uso.

I casi d'uso sono descritti nel documento "Specifiche degli Use Case" ([SPICCA-REGSPICCA-UC]), che presenta gli UCD in ordine decrescente di astrazione; la descrizione inizia con i casi d'uso più generali per passare poi a quelli più dettagliati. Ciascun caso d'uso illustrato nell'UCD ha una sezione dedicata in cui s'illustra il flusso base degli eventi e, qualora necessario, i principali scenari alternativi ed il loro punto di estensione, le pre-condizioni e le post-condizioni.

5.1 Realizzazione degli Use-Case

In questo paragrafo viene descritta la realizzazione di alcuni casi d'uso particolarmente significativi. Al fine di presentare le modalità di collaborazione dei diversi componenti architetturali per la realizzazione dei casi d'uso, si utilizzeranno i Sequence Diagram UML che evidenzieranno, tra l'altro, l'ordine temporale delle diverse azioni eseguite dai componenti stessi.

5.1.1 UC08 – InserimentoSoggettoSPCOOP

In Figura 5 è riportato il sequence diagram dell'inserimento di un SoggettoSPCoop. L'utente richiede, mediante la web application di interfaccia, di eseguire l'operazione di inserimento di un SoggettoSPCoop; il Gestore Registry esegue una verifica dei dati da inserire, in particolare prima cerca il soggetto all'interno del Registry e nel caso in cui non lo trova, verifica che il soggetto che si cerca di inserire è presente nell'IPA. Nel caso in cui il soggetto da inserire non è nel Registry ed è presente nell'IPA, il Gestore Registry scrive i dati relativi al soggetto nel Registry; in caso contrario, mediante la web application, mostra all'utente un messaggio di errore di inserimento.

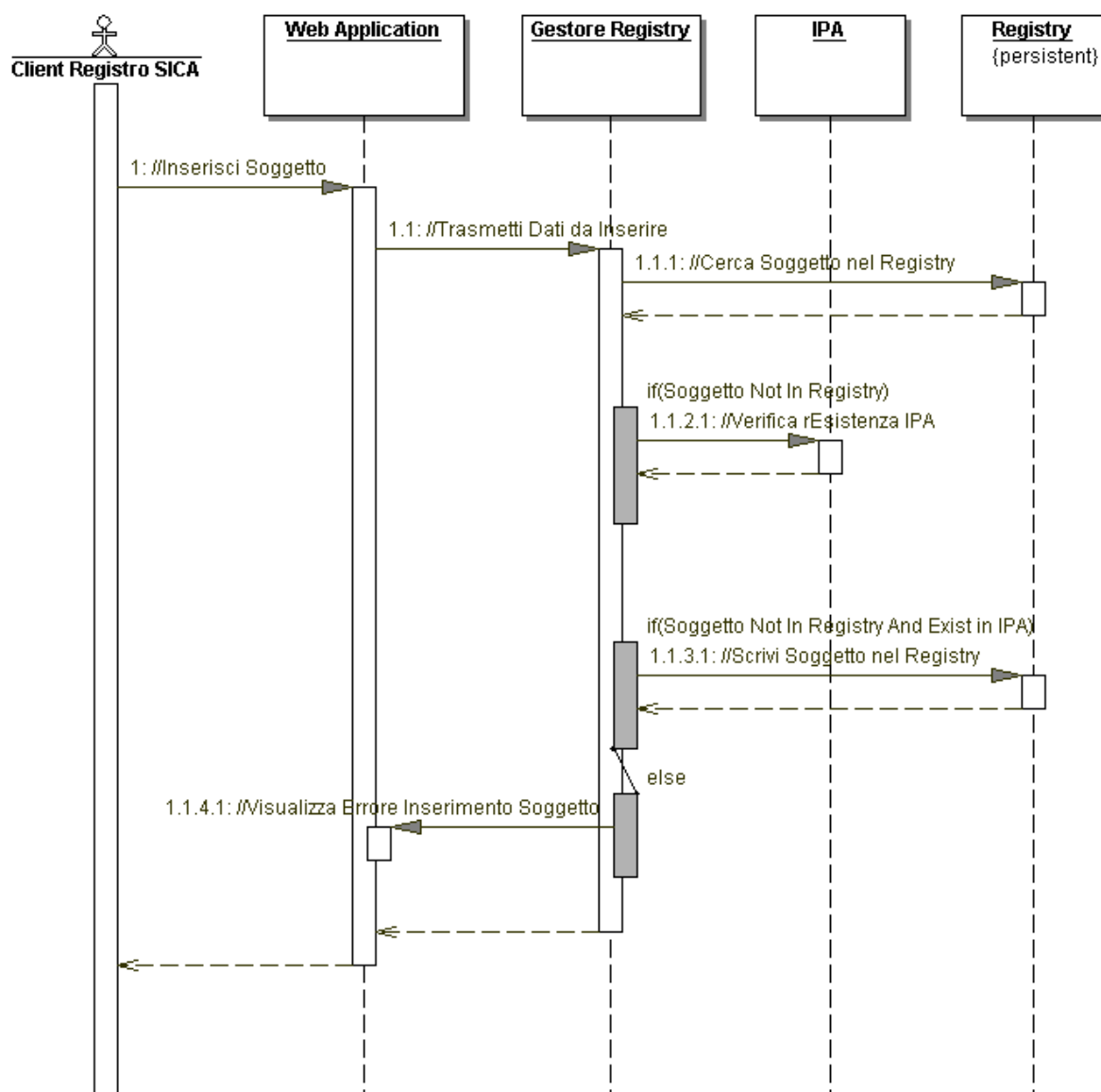


Figura 5: Registro SPICCA - Inserimento SoggettoSPCoop

5.1.2 UC01-Inserimento AS

In Figura 6 è riportato il sequence diagram dell'inserimento della parte comune di un Accordo di Servizio. L'utente richiede, mediante la web application di interfaccia, di eseguire l'operazione di inserimento della parte comune di un Accordo di Servizio; la web application mostra l'elenco dei SoggettiSPCoop e l'elenco delle Ontologie presenti nel registry, recuperati attraverso il Gestore Registry. L'utente seleziona il Soggetto SPCoop per il quale deve inserire l'accordo di servizio; seleziona l'ontologia a cui appartiene il servizio da inserire. La web application mostra la form di inserimento dati, relativa all'ontologia selezionata, e l'utente inserisce i dati relativi all'accordo. Il Gestore Registry verifica l'integrità di tali dati e nel caso di errore viene mostrato all'utente un messaggio di errore di inserimento; altrimenti lo stesso scrive i dati nel Registry.

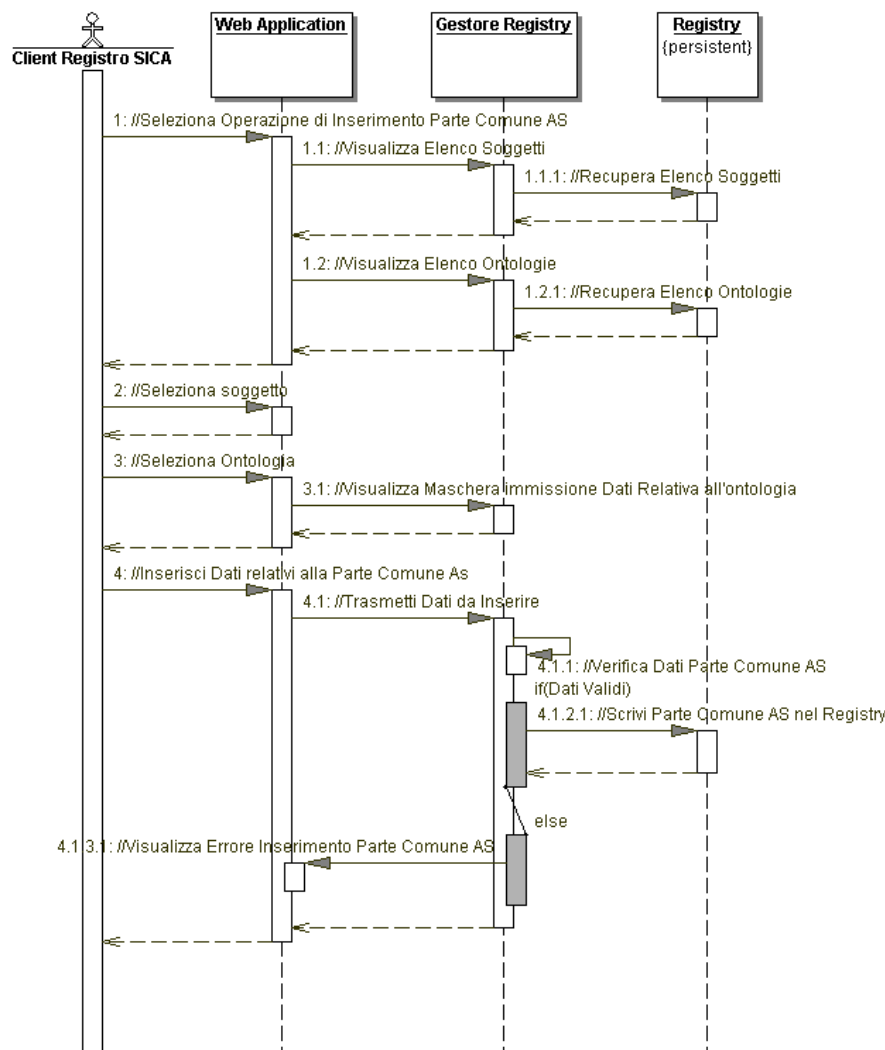


Figura 6: Registro SPICCA - Inserimento Accordo di Servizio (parte comune)

5.1.3 UC02-InserimentoParteSpecificaAS

In Figura 7 è riportato il sequence diagram dell'inserimento della parte specifica di un Accordo di Servizio. L'utente richiede, mediante la web application di interfaccia, di eseguire l'operazione di inserimento della parte comune di un Accordo di Servizio; la web application mostra l'elenco dei SoggettiSPCoop presenti nel registry, recuperati attraverso il Gestore Registry. L'utente seleziona il Soggetto SPCoop per il quale deve inserire la parte specifica dell'accordo di servizio e la web application mostra l'elenco delle parti comuni degli accordi di servizio relativi al SoggettoSPCoop selezionato, presenti nel Registry, che sono stati recuperati dal Gestore Registry. L'utente seleziona l'accordo di servizio per il quale deve inserire la relativa parte specifica, la web application mostra la form di inserimento dati, relativa all'ontologia a cui appartiene il servizio, e l'utente inserisce i relativi dati. Il Gestore Registry verifica l'integrità di tali dati e nel caso di errore viene mostrato all'utente un messaggio di errore di inserimento; altrimenti lo stesso scrive i dati nel Registry.

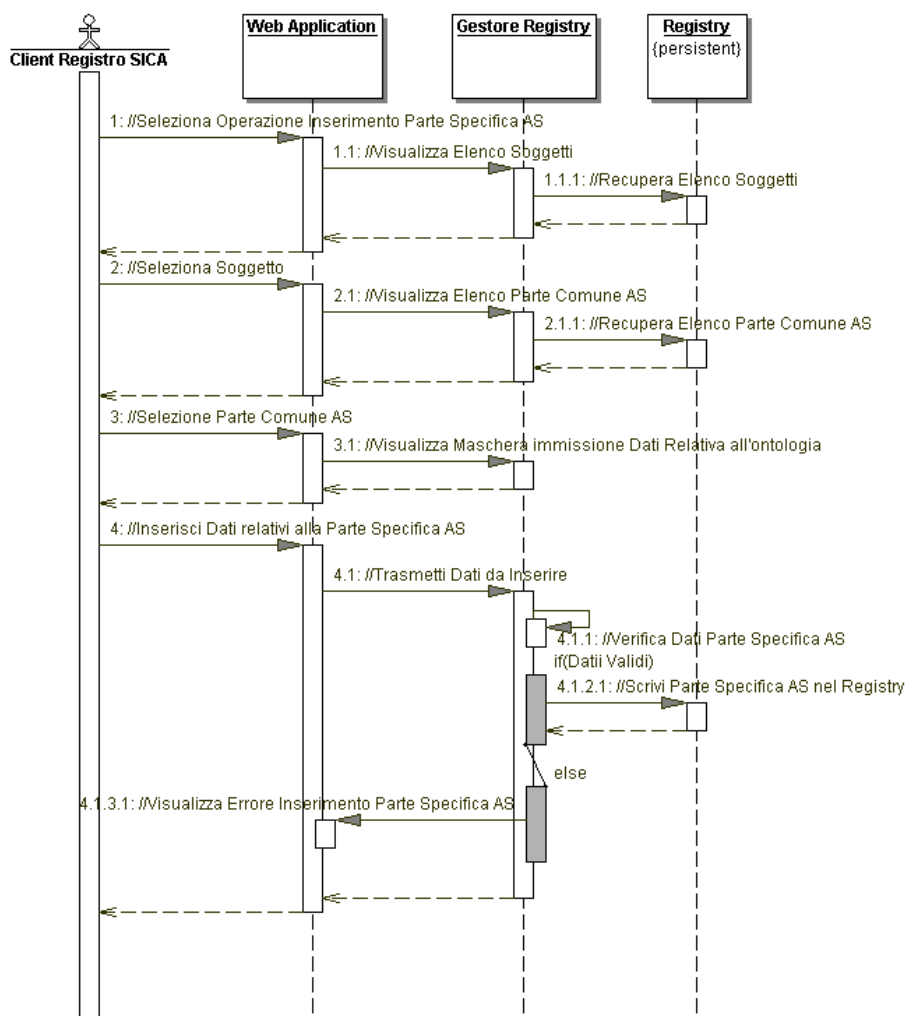


Figura 7: Registro SPICCA - Inserimento Accordo di Servizio (parte specifica)

5.1.4 UC20-ImportAS

In Figura 8 è riportato il sequence diagram dell'Import della Parte Comune e della Parte Specifica di un Accordo di Servizio. L'utente, che deve essere loggato nel sistema come Amministratore, richiede, mediante la web application di interfaccia, di eseguire l'operazione di Import di un Accordo di Servizio; la web application mostra la possibilità di selezionare la modalità di Import, di caricare i file contenenti la Parte Comune e quella Specifica di un Accordo di Servizio, l'elenco dei Soggetti Fruitori e l'elenco dei Profili di Comunicazione, presenti nel Registry, recuperati attraverso il Gestore Registry. L'utente seleziona la modalità di Import "Import Accordo di Servizio Completo", il Soggetto Fruitore e il Profilo di Comunicazione ed importa i file con estensione apc e aps, contenenti rispettivamente la Parte Comune e la Parte Specifica di un Accordo di Servizio. Il Gestore Registry verifica l'integrità di tali dati, effettuando nello specifico un'operazione di validazione sui documenti relativi alle Specifiche dei Livelli di Servizio e di

Sicurezza secondo gli standard esposti in [SPCOOP-AS2009] e, in caso di errore, fornisce all'utente un messaggio di errore di import, altrimenti scrive i dati nel Registry.

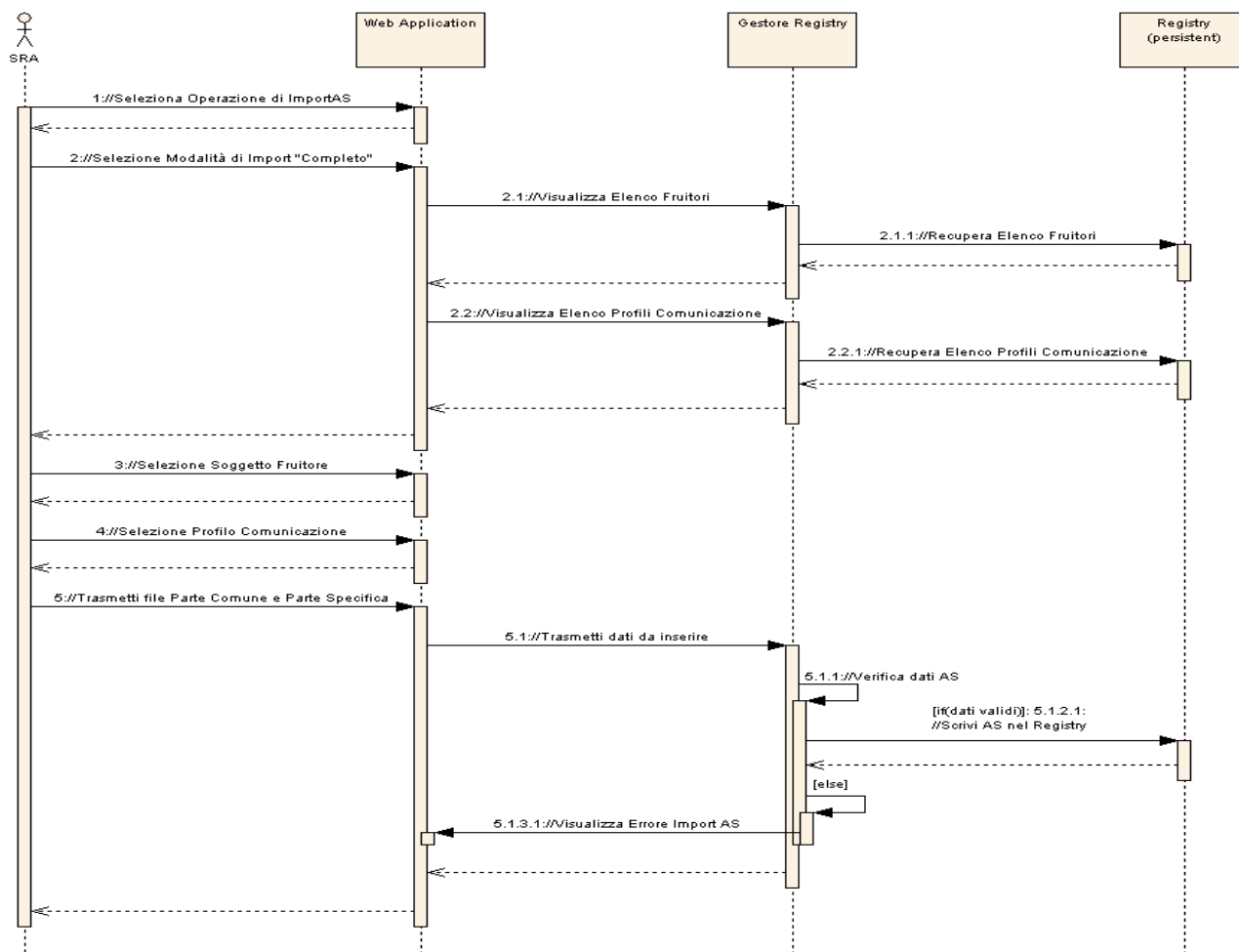


Figura 8: Registro SPICCA – Import di un Accordo di Servizio Completo

5.1.5 UC23-ExportAS

In Figura 9 è riportato il sequence diagram dell'Export di un Accordo di Servizio. L'utente richiede, mediante la web application di interfaccia, di eseguire l'operazione di Export di un Accordo di Servizio; la web application mostra l'elenco degli Accordi di Servizio presenti nel Registry, recuperati attraverso il Gestore Registry. L'utente seleziona l'Accordo di Servizio per il quale effettuare l'operazione di Export; l'utente, inoltre, può decidere se effettuare l'operazione di Export della Parte Comune o della Parte Specifica dell'Accordo di Servizio selezionato. Il Gestore Registry preleva i dati richiesti dal Registry e li trasmette all'utente mediante la Web Application.

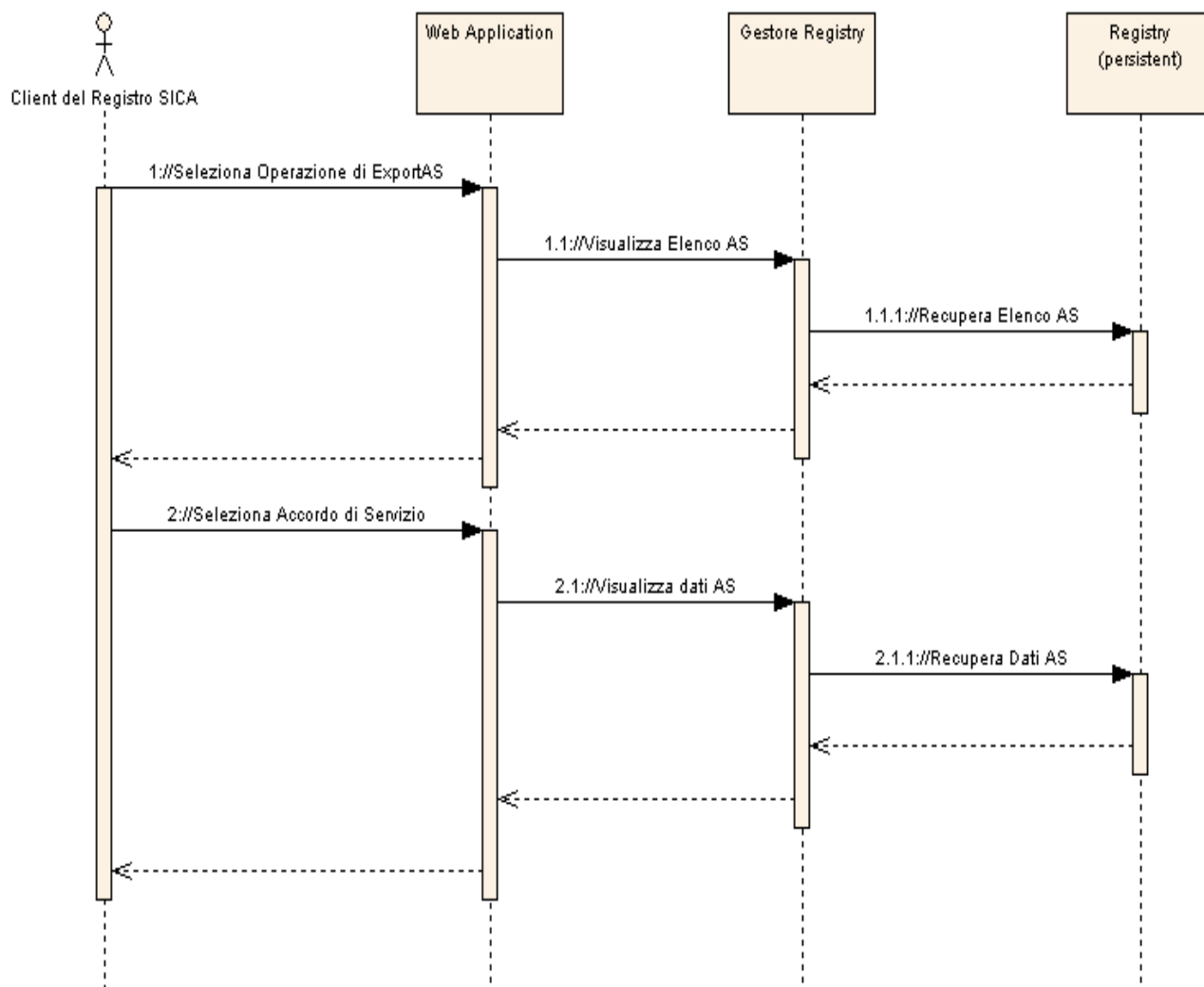


Figura 9: Registro SPICCA – Export di un Accordo di Servizio

6 Deployment View

In questo paragrafo viene riportata la configurazione, su cui il Registro SPICCA sarà rilasciato, mediante il Deployment Diagram di Figura 10. Il SOAP Server esporrà le funzionalità del Registro SPICCA come web services, il WEB Server conterrà l'applicazione web per l'accesso web ai servizi del Registro SPICCA, il DB Server ospiterà i dati relativi agli accordi di servizio e ai SoggettiSPCoop.

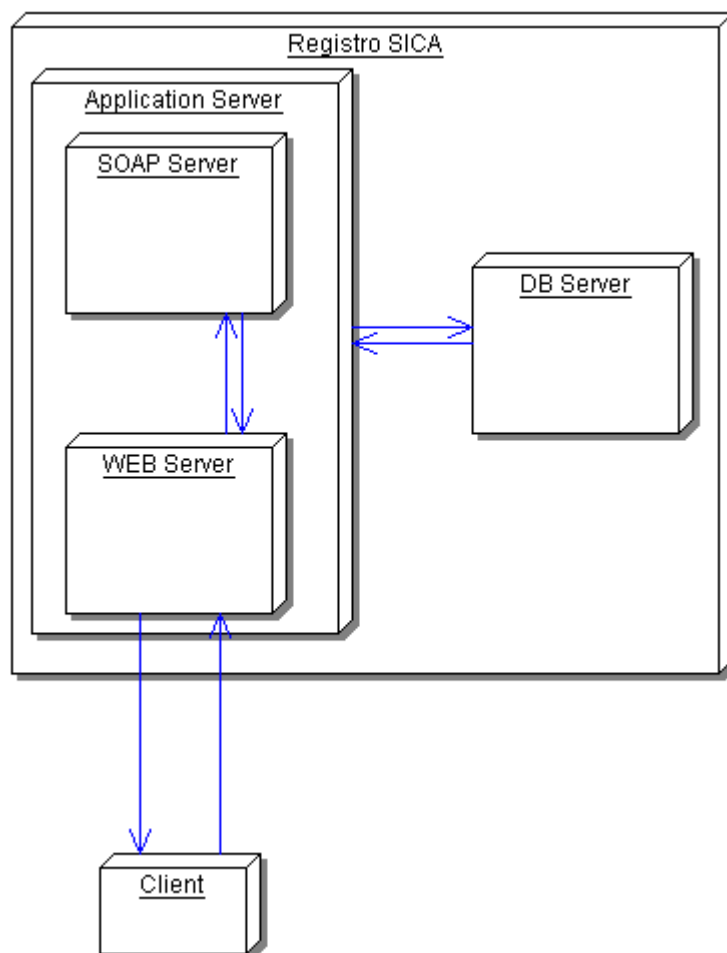


Figura 10: Deployment Diagram Registro SPICCA

7 Implementation View

In questo paragrafo viene riportata la struttura complessiva del modello di implementazione, la decomposizione del software in sottosistemi e ogni componente significativo per l'architettura.

In Figura è riportato il class diagram dei componenti di "backend" del Registro SPICCA.

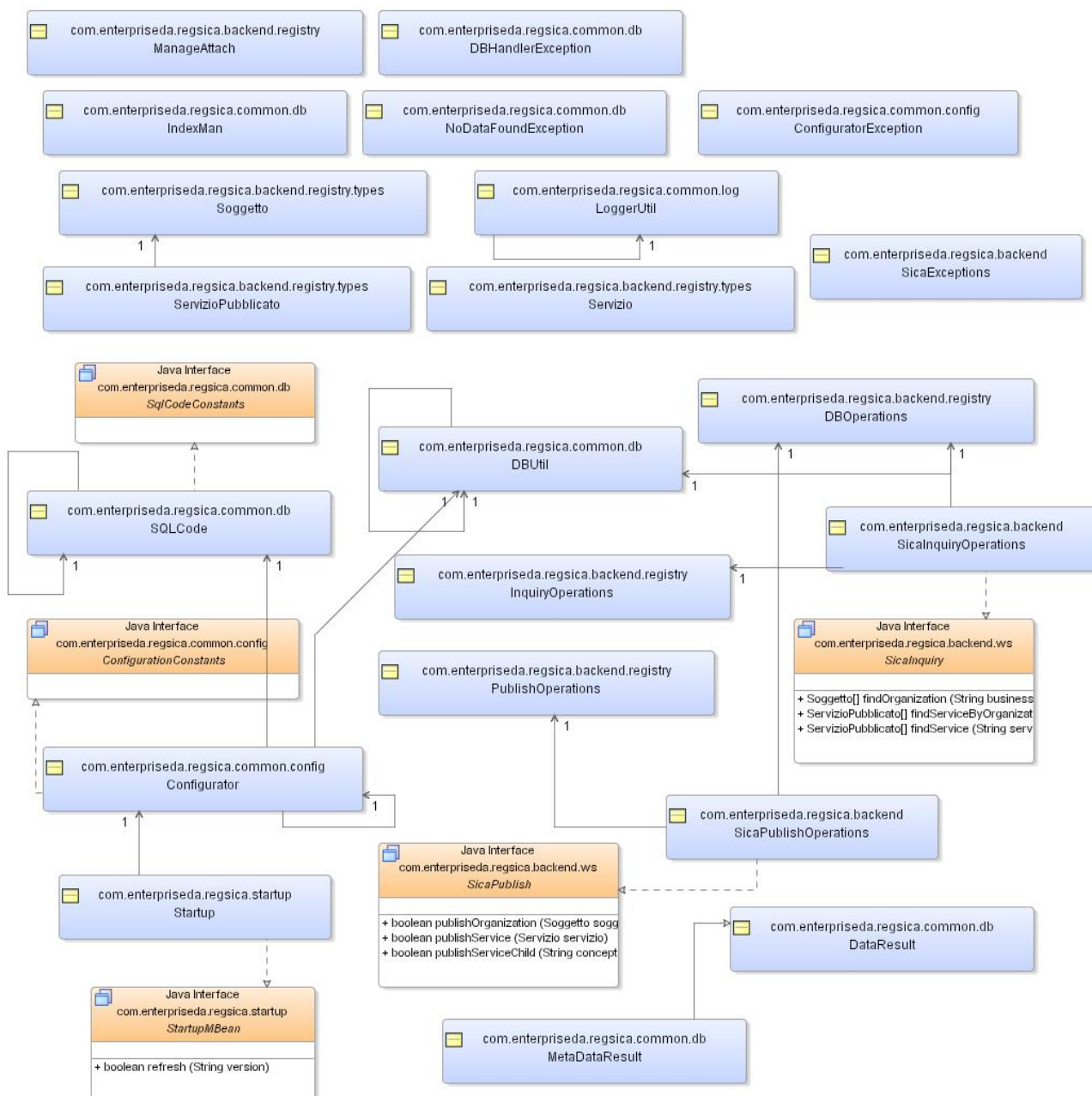


Figura 11: Class Diagram Registro SPICCA "backend"

7.1 Layers

7.1.1 BackEnd

La componente di BackEnd è stratificata nei Layer riportati in Figura 82.

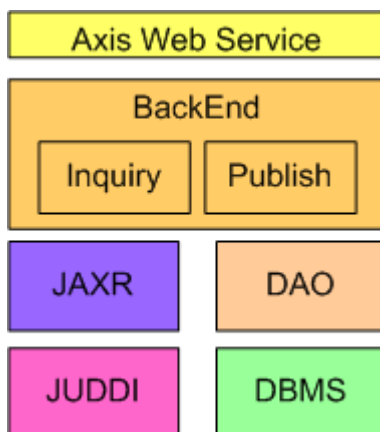


Figura 82: Vista Layer Registro SPICCA "backend"

7.1.1.1 Axis Web Service

Esposizione dell'interfaccia dei servizi di BackEnd sottoforma di Web Service pubblicati su server AXIS.

7.1.1.2 BackEnd

Layer principale dove è implementata la logica di gestione delle operazioni di ricerca (Inquiry) e pubblicazione (Publish) sul Registro SPICCA. Entrambe le sottocomponenti Inquiry e Publish dovranno interagire sia con il sottosistema JAXR che con quello DAO.

7.1.1.3 JAXR

Java API for XML Registries (JAXR) è il Layer che interagisce con il sottosistema JUDDI per la ricerca/pubblicazione di informazioni su un generico registro UDDI. La scelta di questo interfacciamento è dettata da un ovvio disaccoppiamento dal sistema sottostante che potrebbe essere, ad esempio, anche un'implementazione ebXML di un registro UDDI.

7.1.1.4 DAO

Layer di classi generiche che interagiscono con un qualsiasi DBMS.

7.1.1.5 JUDDI

Layer di memorizzazione delle informazioni relative ai servizi pubblicati secondo lo standard UDDI.

7.1.1.6 DBMS

Layer di memorizzazione delle informazioni relative agli accordi di servizio gestite, secondo quanto dettato dalle specifiche CNIPA, mediante un DBMS.

7.1.2 FrontEnd

La componente di FrontEnd è stratificata nei Layer riportati in Figura 93: dall'immagine si evince che i Layer rispecchiano il pattern MVC del framework Jakarta Struts.

E' da evidenziare che il Layer Controller interagisce direttamente con il Layer Axis Web Service per accedere alle funzionalità di ricerca/pubblicazione sul Registro SPICCA esposte dal componente di BackEnd.

7.1.2.1 DBMS

Layer di memorizzazione delle informazioni di supporto alla componente FrontEnd mediante l'uso di un DBMS.

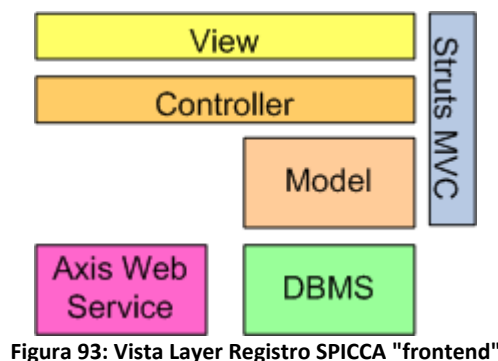


Figura 93: Vista Layer Registro SPICCA "frontend"

