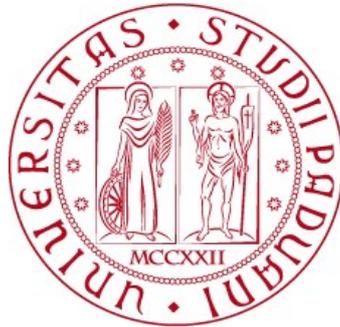


Università degli Studi di Padova

Facoltà di Ingegneria



**GESTIONE DEL WORKFLOW CLINICO:
ANALISI DEGLI STANDARD ESISTENTI
E
PROPOSTA DI NUOVE LINEE GUIDA IHE**

Laureando: Mauro Zanardini

Relatore: Prof. Giovanni Sparacino

Correlatori: Dott. Claudio Saccavini
Ing. Luca Zalunardo

Corso di laurea Magistrale in Bioingegneria

Padova, 25 Ottobre 2011

Anno Accademico 2010/2011

INDICE

1.INTRODUZIONE.....	9
1.1 La sanità digitale (eHealth).....	9
1.2 Il Consorzio Arsenal.IT.....	12
1.2.1 Presentazione generale.....	12
1.2.2 Profilo tecnico del consorzio Arsenal.IT.....	13
1.2.3 Storia del consorzio Arsenal.IT.....	14
1.2.4 Mission e valori.....	15
1.3 Contesto specifico e Open Issues.....	16
1.4 Scopo della tesi.....	17
2. IL PROGETTO DOGE: presentazione e problemi aperti di interesse.....	19
2.1 Descrizione del progetto “DOGE”.....	19
2.2 Attori e servizi nella rete MMG/PLS.....	21
2.2.1 Servizi Anagrafici.....	22
2.2.2 Servizi di condivisione documentale.....	24
2.2.3 Servizi di e-Prescription.....	25
2.2.4 Servizi di Privacy.....	26
2.3 Ciclo prescrittivo digitale.....	26
2.3.1 Inquadramento nazionale della soluzione proposta dalla Regione Veneto.....	27
2.3.2 Architettura del Sistema per la Prescrizione Elettronica.....	27
2.3.3 Gestione ePrescription Specialistica/Farmaceutica prodotta da un MMG/PLS.....	28
2.3.4 Gestione del Ciclo Prescrittivo Ospedaliero.....	32
2.3.5 Scenario funzionale: Primo caso d’uso.....	37
2.3.5.1 Creazione, pubblicazione e registrazione del documento di prescrizione.....	37
2.3.5.2 Presa in carico della e-Prescription da parte dell’Azienda Sanitaria.....	38
2.3.5.3 Schema sequenziale delle transazioni.....	40
2.3.5.4 Schema generale riassuntivo.....	40
2.3.6 Elenco dei ricettari in carico al medico.....	41
2.3.7 Open Issues nel progetto DOGE di interesse per la tesi.....	42
3. METODOLOGIA DI LAVORO: STANDARD E LINEE GUIDA ESISTENTI.....	45
3.1 Introduzione.....	45
3.2 Problematiche connesse agli standards per l’informatica sanitaria.....	45
3.3 IHE (Integrating the Healthcare Enterprise).....	46
3.3.1 I Technical Framework.....	47
3.3.2 L’Interoperabilità.....	48
3.3.3 I Messaggi.....	49
3.4 Attori e transazioni.....	50
3.5 Standard per l’informatica sanitaria: breve stato dell’arte.....	51
3.5.1 Lo standard DICOM.....	53
3.5.2 Lo standard HL7.....	55
3.5.3 Lo standard CDA.....	56
3.6 CROSS-ENTERPRISE DOCUMENT SHARING: XDS.....	57
3.6.1 Panoramica.....	57
3.6.2 Gli Attori.....	58
3.6.3 Le Transazioni.....	59
3.6.4 Tipi di documenti supportati.....	61
3.6.5 Opzioni gestite dal profilo XDS.....	62
3.6.6 EHR-CR e EHR-LR.....	62

3.6.7 Documento XDS.....	64
3.6.8 Submission Request.....	64
3.6.9 Submission Set.....	64
3.6.10 Folder.....	65
3.6.11 XDS Affinity Domain.....	65
3.6.12 Esempio di Submission Request, Submission Set e Folder.....	66
3.6.13 Modello di dati dell'XDS registry e i suoi attributi.....	67
3.6.13.1 XDS Document Registry Data Model.....	67
3.6.13.2 Attributi delle XDS Document Entries.....	67
3.6.14 DOCUMENT LIFECYCLE.....	68
3.6.14.1 Stato di disponibilità dei documenti.....	68
3.6.14.2 Relazione tra documenti.....	68
3.7 NOTIFICATION OF DOCUMENT AVAILABILITY: NAV.....	69
3.7.1 Scopo.....	69
3.7.2 Attori e Transazioni.....	70
3.7.3 Invio della notifica.....	70
3.8 CROSS COMMUNITY ACCESS: XCA.....	70
3.8.1 Introduzione.....	71
3.8.2 Attori e transazioni.....	73
3.8.3 XCA PROCESS FLOW.....	73
3.8.4 Use Cases.....	73
3.8.5 Interazioni Dettagliate.....	74
3.8.6 HomeCommuytiID defined.....	74
3.9 Processo di definizione ed approvazione di un nuovo profilo IHE.....	75

4. Analisi dei Linguaggi Standard descrittore di Workflow.....79

4.1 Introduzione.....	79
4.2 Descrittori di Workflow.....	80
4.3 WORKFLOW.....	81
4.3.1 Definizione di Workflow.....	81
4.3.2 Tipologie di Workflow.....	82
4.3.3 Benefici dei Workflow.....	83
4.3.4 Linguaggi standard per la descrizione di Workflow.....	83
4.3.4.1 WSFL (Web Services Flow Language).....	84
4.3.4.2 GSFL (Grid Services Flow Language).....	87
4.3.4.3 AGWL (Abstract Grid Workflow Language).....	88
4.3.4.4 XPD (XML Process Definition Language).....	89
4.3.4.5 SCUFL (Simple Conceptual Unified Flow Language).....	90
4.3.4.6 WSCI (Web Service Choreography Interface).....	90
4.3.4.7 MOML (Media Objects Markup Language).....	91
4.3.4.8 BPEL4WS (Business Process Execution Language for Web Services).....	92
4.3.4.9 BPEL4People (Business Process Execution Language for Web Services).....	93
4.3.4.10 BPML (Business Process Modeling Language).....	94
4.3.4.11 XFlow (XML-based document-centric Workflow).....	94
4.3.4.12 XDOC-WFMS (XML Document Centric Workflow Management System).....	99
4.3.4.13 XLR (eXchangeable Routing Language).....	99
4.3.4.14 XLANG (XML-based Language).....	100
4.4 Conclusioni sulla ricerca bibliografica.....	102

5. Profilo XDW (Cross-Enterprise Document Workflow).....105

5.1 Introduzione.....	105
5.2 Attori e transazioni.....	107
5.2.1 Raggruppamento.....	107
5.3 XDW Document.....	108
5.3.1 Introduzione.....	108
5.3.2 Prima versione dell'XDW Workflow Document (CDA-based).....	108
5.3.2.1 Struttura dell'XDW Workflow Document CDA-based.....	108
5.3.2.2 Use case: Workflow di Visita Specialistica (e-Referral).....	110

5.3.3 Seconda versione dell'XDW Workflow Document (Human Task-based).....	115
5.3.3.1 Struttura dell'XDW Workflow Document HumanTask-based.....	116
5.3.3.2 Use case: Workflow di Visita Specialistica (e-Referral).....	118
5.4 Conclusioni sull'indagine comparativa.....	120
5.5 Considerazioni sulla sicurezza di XDS.....	123

6.XDW Workflow Document basato su standard Clinical Document

Architecture (CDA).....	125
6.1 Health Level 7 (HL7).....	125
6.2 Clinical Document Architecture (CDA).....	125
6.2.1 Componenti di un documento CDA.....	128
6.2.2 Leggibilità umana e rendering dei documenti CDA.....	129
6.2.3 CDA e HL7 Messaging Standards.....	129
6.2.4 Conformità del documento CDA.....	131
6.3 XDW Workflow Document.....	133
6.3.1 Metadati dell'header.....	133
6.3.2 Metadati del CDA-body.....	142
6.3.2.1 Sezione CONTROL INFO.....	143
6.3.2.2 Sezioni INPUT e OUTPUT.....	143

7. XDW Workflow Document basato su standard Human Task.....147

7.1 BPEL4People.....	147
7.2 Standard WS-HumanTask.....	148
7.2.1 Introduzione.....	148
7.2.2 Architettura del sistema WS-HumanTask.....	149
7.3 Definizione XDW Workflow Document (HumanTask-based).....	155
7.3.1 Struttura generale del Workflow Document.....	155
7.3.2 Elementi definiti dal profilo IHE XDW.....	158
7.3.3 Elementi provenienti dallo standard OASIS WS-HumanTask.....	158
7.4 Gestione del Workflow Document.....	161
7.4.1 Gestione del Lifecycle del Workflow Document.....	161
7.4.2 Associazione tra le varie versioni di uno stesso Workflow Document.....	163
7.4.3 Creazione un nuovo Workflow.....	163
7.4.4 Aggiornamento il Workflow Document.....	164
7.4.5 Ottenere tutti i documenti associati al Workflow.....	164
7.4.6 Utilizzo del eventCodeList per gestire lo stato del Workflow Document.....	165
7.5 Metadati: XDS e XDS Folder.....	166

8. CONCLUSIONI.....169

8.1 Problemi risolti.....	171
8.2 Open issues.....	171

APPENDICE A: CDA Technical Specifications.....173

A.1 CDA header.....	173
A.2 CDA Body.....	177
A.2.1 Body non strutturato.....	177
A.2.2 Body strutturato.....	177

APPENDICE B: Technical Specifications BPEL4People.....185

B.1 Design del linguaggio.....	185
B.2 Concetti base del linguaggio.....	187

APPENDICE C: Template dell'XDW WORKFLOW DOCUMENT	
(WSHumanTask-based).....	193
C.1 Definizione del Template del XDW Workflow Document.....	193

APPENDICE D: Template dell'XDW WORKFLOW DOCUMENT	
(CDA-based).....	201
D.1 Introduzione.....	201
D.1.1 Responsabilità originator.....	202
D.1.2 Responsabilità recipient.....	202
D.2 Template dell'Header.....	203
D.2.1 Attributi.....	203
D.2.2 Partecipanti.....	205
D.2.2.1 Paziente.....	205
D.2.2.2 Autore.....	210
D.2.2.3 Custode.....	213
D.2.3 Documenti Correlati.....	215
D.3 Template del Body Strutturato.....	217
D.3.1 Primo livello: sezione CONTROL INFO.....	217
D.3.2 Secondo livello: sezioni INPUT/OUTPUT.....	223

PREFAZIONE

L'obbiettivo del lavoro di tesi è di arrivare alla definizione di un profilo IHE in grado di permettere la gestione efficace dei Workflow Clinici. In quest'ottica si è deciso di valutare gli standard presenti in letteratura individuando le caratteristiche principali dei linguaggi per la gestione di Workflow. Sulla base delle informazioni così raccolte è stata definita la struttura del Workflow Document, documento alla base delle nuove linee guida IHE proposte.

All'interno del processo IHE, come in tutte le realtà che si propongono di essere centro di coordinamento di attività di ricerca cooperativa, si instaura una forte dialettica costruttiva tra le parti coinvolte; ciò ha portato alla definizione della versione definitiva del documento partendo dallo sviluppo di una prima versione elaborata presso il consorzio Arsenal.IT. Questa proposta è caratterizzata dall'utilizzo di una composizione di standard diversi per la gestione di Workflow. Il profilo IHE così definito (che prende il nome XDW o Cross Enterprise Workflow Document) è stato approvato da IHE quale linea guida internazionale per la gestione di Workflow Clinici. Un applicativo basato su questo profilo è stato poi integrato all'interno della gestione del ciclo prescrittivo digitale del progetto regionale DOGE, permettendo il management di più di 4000 prescrizioni elettroniche al giorno all'interno dell'ULSS di Adria.

1. INTRODUZIONE

1.1 La Sanità Digitale (eHealth)

Ad oggi uno dei punti critici dei bilanci statali di tutto il mondo è costituito dall'impiego di fondi necessario per il mantenimento dei sistemi sanitari. I costi per la sanità in molti casi sono cresciuti oltre il sostenibile ed i Governi sono di conseguenza costantemente impegnati nel tentativo di contenerne il deficit e, dove possibile, di ridurlo.

Per raggiungere tale obiettivo è necessario prevedere riforme sostanziali del sistema, partendo da un riesame critico dello stato attuale dell'erogazione di prestazioni sanitarie. E' inoltre necessario considerare anche quale può essere l'apporto delle nuove conoscenze connesse all'Information & Communication Technology per ottenere livelli di razionalizzazione dei processi che erano impensabili fino a pochi anni fa. Una visione ancora più moderna dell'approccio e-Health vede le soluzioni di ICT applicate ai sistemi informativi sanitari come lo strumento necessario per:

- generare azioni efficaci per migliorare l'equilibrio strutturale dei processi;
- stimolare nuovi fabbisogni di misurazione;
- avviare il monitoraggio dell'appropriatezza dei processi di cura;
- presentare nuovi modelli organizzativi;
- definire il rapporto e l'integrazione tra cure primarie (territorio), cure delle acuzie, delle emergenze (ospedale) e lungodegenze (deospedalizzazione);

Con Sanità Digitale si vuole infatti definire il tentativo di applicazione delle tecnologie della Teoria dell'Informazione e delle Telecomunicazioni al settore sanitario, con l'obbiettivo di rendere i processi più efficienti e razionalizzabili. E-Health è quindi un termine coniato abbastanza recentemente, che ha l'obbiettivo di descrivere il complesso delle risorse, soluzioni e tecnologie informatiche di rete applicate alla salute ed alla sanità. Il termine può comprendere una vasta gamma di significati i cui limiti possono essere da una parte la medicina/healthcare e dall'altra l'informazione tecnologica. Di seguito sono presentate delle possibili applicazioni eHealth:

- Electronical Medical Record**: permette una comunicazione semplice dei dati del paziente tra le diverse figure professionali (medici di medicina generale, specialisti, care-team, farmacie).

-**Telemedicina**: include tutti tipi di cure mediche o psicologiche che non richiedono al paziente una visita diretta da parte del medico curante. Quando questo servizio funziona, i pazienti non hanno più la necessità di andare dal medico e consente al medico stesso un bacino d'utenza maggiore, con un conseguente risparmio di risorse.

-**Medicina basata su prove di efficacia**: comporta un sistema in grado di fornire informazioni circa l'appropriato trattamento delle condizioni cliniche di un paziente. Un professionista dell'healthcare può controllare se le sue diagnosi sono conformi alle ricerche scientifiche. Il vantaggio è quello di poter aggiornare i dati dei pazienti, e parallelamente disporre di uno strumento attivo nel supportare il processo di cura.

-**Consumer Health Informatics**: è stata definita quella branca dell'informazione medica che si occupa di analizzare i bisogni dei consumatori e rendere accessibili gli studi e gli sviluppi nel campo della medicina, definendo linee guida consistenti per il trattamento dei pazienti.

-**Virtual Healthcare Teams**: è un sistema composto da professionisti dell'healthcare che collaborano fra loro e condividono le informazioni sui pazienti attraverso l'utilizzo di apparecchiature digitali, permettendo un approccio più completo e specialistico nei confronti dei casi clinici specifici.

Per permettere il funzionamento dei sistemi così definiti si rende necessario assicurare l'integrazione tra sistemi informatici diversi, e garantire la consistenza dei dati memorizzati in sistemi diversi. Gli standard di interoperabilità costituiscono lo strumento essenziale per garantire queste proprietà. A livello internazionale molte risorse sono impiegate per la definizione di norme specifiche per l'informatica sanitaria, coinvolgendo enti pubblici, utenti del sistema sanitario e aziende sviluppatrici di applicativi software. In questo modo vengono definite delle interfacce attraverso le quali applicativi diversi possono interoperare. L'interazione avviene essenzialmente per due finalità distinte:

-**"gestionale"**: per permettere a due applicazioni diverse di scambiare richieste e risultati (prescrizioni di farmaci e di analisi, lettera di dimissione, prenotazione di prestazioni tramite centri unificati di prenotazione – es. CUP);

-**"clinica"**: per accedere tempestivamente al momento del bisogno alle informazioni cliniche memorizzate in applicazioni, anche remote, gestite da altri operatori sanitari o da apparecchiature sofisticate.

Secondo la American Society for Testing and Materials (ASTM), gli standard possono essere di sei

tipi diversi:

- specifiche tecniche**: la definizione di un insieme di requisiti che devono essere soddisfatti da un sistema, con le procedure per determinare se il sistema soddisfa tali requisiti;
- standard operativi**: una procedura che descrive come eseguire una o più operazioni o funzioni specifiche;
- terminologie**: un documento che comprende termini, definizioni, descrizioni, spiegazioni, abbreviazioni e acronimi;
- classificazioni**: un arrangiamento sistematico o suddivisione di prodotti, sistemi o servizi in gruppi, in base a caratteristiche similari;
- istruzioni**: una serie di opzioni o istruzioni (che non raccomandano un particolare comportamento);
- metodi di test**: procedure per identificare, misurare e valutare un materiale, un prodotto, un sistema.

Per garantire l'interoperabilità nel campo sanitario finora si è dato particolare risalto alle specifiche tecniche relative ai formati standard dei messaggi. Tali messaggi sono concepiti per essere prodotti agevolmente dal sistema che spedisce ed essere facilmente interpretabili dalle applicazioni riceventi. Inoltre devono poter trasferire in modo preciso le informazioni necessarie per un colloquio efficace tra il sistema che spedisce ed il sistema che riceve.

Come già detto i sistemi che afferiscono al medesimo sistema sanitario devono essere in grado di condividere informazioni, tuttavia, pur utilizzando protocolli standard di comunicazione, spesso non sono in grado di scambiarsi efficientemente dati, in quanto gli standard stessi possono presentare conflitti interpretativi. IHE è un'iniziativa promossa nel 1997 dalla RSNA (Radiological Society of North America) e da HIMSS (Healthcare Information and Management System Society) per promuovere l'integrazione tra i sistemi informativi sanitari, al fine di una migliore condivisione delle informazioni necessarie per l'impostazione e il monitoraggio del percorso diagnostico-terapeutico del paziente all'interno delle diverse strutture sanitarie. L'obiettivo di IHE non è quello di definire un vero e proprio standard di comunicazione, ma piuttosto coinvolgere produttori ed utenti dei sistemi informativi sanitari in un forum collaborativo per identificare e risolvere i problemi d'interoperabilità con i quali ci si confronta in ambito sanitario. IHE promuove e spinge l'adozione coordinata di standard consolidati (XML, IEEE, DICOM, IETF ecc.) attraverso la creazione e la manutenzione di documenti tecnici che contengono le linee guida per l'implementazione chiamate "Profili IHE", i quali sono pubblicati come collezione di documenti nel IHE Technical Framework e garantiscono l'efficacia e la funzionalità dei sistemi informativi.

I fornitori di tecnologie sanitarie e informatiche che condividono l'iniziativa di IHE, si sottopongono annualmente a rigorosi test di integrazione aventi per oggetto i profili definiti nei technical framework (Connect-a-thon). Durante queste sessioni viene costruita una rete tra diversi sistemi medicali, verificando che questi si integrino correttamente nello scambio di dati comuni, in quelle che vengono definite transazioni, secondo uno schema di azioni prestabilite, effettuate tra gli attori IHE e specificate preliminarmente dai gruppi IHE.

1.2 Il Consorzio Arsenà.IT

1.2.1 Presentazione generale

Il Consorzio Arsenà.IT è ormai un punto di riferimento non solo regionale, ma nazionale, sulla riflessione del ruolo dell'IT nella riorganizzazione del Servizio sanitario, come indicato sia dai piani regionali che dal piano e-Gov 2010. Il consorzio si sviluppa all'interno del panorama creato dalla sanità veneta, caratterizzato da forte dinamismo, dall'elevata efficacia e efficienza nella qualità dell'erogazione del servizio al cittadino/paziente. L'e-Health, è visto non solo come sviluppo tecnologico, ma come occasione per ridisegnare modelli organizzativi secondo logiche di rete, di sistema e d'integrazione, e volte al miglioramento del servizio di assistenza sanitaria. Il fatto che su questi aspetti si siano identificate tutte le Aziende Sanitarie del Veneto, fa del Consorzio Arsenà.IT un unicum nel panorama nazionale ed europeo. Dal 2005, anno della nascita come Consorzio Telemedicina, Arsenà.IT non ha solo promosso la cultura dell'ICT in Sanità, ma ha anche raccolto un patrimonio di conoscenze e competenze messe a disposizione di tutti i consorziati, che hanno riconosciuto al Consorzio il ruolo di Osservatorio dei sistemi di e-Health realizzati nel territorio.

Ad oggi, il Consorzio Arsenà.IT si inquadra, con un proprio modello, all'interno della nuova fase di sviluppo dell'e-Health in Italia e in Europa, caratterizzata dalla gestione strategica dei sistemi di ICT e di telemedicina, per affrontare, con risposte efficaci e coerenti, il profondo cambiamento dei bisogni e della domanda attuale di salute. Il nuovo modello di e-Health, di cui Arsenà.IT si fa ispiratore, mira al percorso integrato del paziente con i presidi territoriali, gli ospedali, lo stesso domicilio, all'analisi degli impatti delle ICT in termini di out-come clinico, alla considerazione dell'empowerment del paziente, alla verifica dell'adeguatezza dei sistemi di e-Health al tipo di servizio che si vuole erogare, ad una visione delle applicazioni di telemedicina, come un tassello strategico per la pianificazione e lo sviluppo del sistema e per la valorizzazione di tutte le possibili risorse territoriali.

Il cittadino/paziente è il punto di riferimento attorno a cui ruotano il programma di sviluppo, l'attività del Consorzio Arsenàl.IT e il suo essere Centro Studi e Osservatorio dei servizi di e-Health in Italia.

1.2.2 Profilo tecnico del Consorzio Arsenàl.IT

Arsenàl.IT, per le sue caratteristiche peculiari di Centro Studi, è chiamato ad intervenire laddove si renda necessario un sostegno tecnico per la definizione di modelli organizzativi, flussi ed architetture funzionali a supporto dei processi sanitari. Proprio per questo, il Consorzio non può esimersi dal confrontarsi continuamente con le direttive europee, nazionali e regionali, che regolamentano l'ICT in Sanità e dalla partecipazione a quei tavoli decisionali, in cui vengono trattati queste tematiche. Inoltre, data la rilevante importanza dei temi della ricerca e dell'innovazione, si rende indispensabile un continuo aggiornamento e confronto con realtà all'avanguardia, affacciandosi oltre che alle realtà venete e nazionali, anche a quelle dei board europei ed internazionali delle istituzioni che operano nel settore. Negli ultimi anni Arsenàl.IT è riuscito a posizionarsi nel board Internazionale di IHE con il ruolo di Local Healthcare IT and Consulting Company assieme ad altre organizzazioni internazionali. Da quando fa parte di IHE, Arsenàl.IT partecipa in modo collaborativo ai vari comitati internazionali nei diversi ambiti di lavoro (radiologia, laboratorio, infrastrutture, etc.), fornendo il proprio supporto e la propria esperienza e, allo stesso tempo, raccogliendo indicazioni e consigli utili sui temi d'interesse.

Data la complessa e variegata realtà veneta, l'approccio di interoperabilità maggiormente perseguito da Arsenàl.IT è quello della federazione tra diversi domini indipendenti e autoconsistenti, in grado di scambiare tra loro servizi e documenti rispettando quanto già sviluppato e implementato dalle Aziende Sanitarie ed Ospedaliere. Questo approccio rende possibile l'interfacciamento con soluzioni tecnologiche già a regime, ma anche l'interoperabilità futura con sistemi non ancora sviluppati, in una logica di estendibilità e di salvaguardia degli investimenti sostenuti in passato. In molti casi, però, non è sufficiente garantire solo l'interoperabilità tecnologica. Il Consorzio ritiene fondamentale il conseguimento dell'interoperabilità semantica, cioè della capacità di diversi sistemi informatici di condividere, oltre ai documenti, anche il significato dei dati e dei servizi scambiati. In quest'ottica, è indispensabile, a livello regionale, la definizione di insiemi di informazioni, il cui significato sia comune e condiviso (per esempio, le codifiche dei risultati e delle prestazioni). È proprio per questo, che Arsenàl.IT si sta ponendo in prima linea anche nei temi inerenti la strutturazione dei documenti clinici e la gestione dei sistemi di codifica, collaborando con il gruppo di lavoro HL7 Italia per la produzione di documenti strutturati XML-CDA2 e confrontandosi

costantemente con il gruppo di lavoro LOINC International per la definizione di codifiche comuni. Inoltre, Arsenà.IT ha costituito, nel corso del 2009, in ambito regionale, un tavolo di confronto tra i responsabili dei sistemi informativi delle aziende consorziate, denominato “Comitato Tecnico”. È questo un luogo di discussione e di condivisione delle problematiche tecniche e organizzative in un’ottica di collaborazione interaziendale. Durante gli incontri avvenuti è nata l’esigenza da parte delle aziende consorziate e della Regione di avviare due tavoli di lavoro: uno per la definizione dell’architettura del sistema informativo regionale, l’altro per la discussione degli aspetti inerenti l’anagrafica regionale dei contatti. In collaborazione con la Regione e nel rispetto di quanto già sviluppato dai consorziati, Arsenà.IT sta cercando di mettere a fattor comune le varie esperienze progettuali, di cui si occupa e di cooperare alla costruzione di una cornice omogenea capace di contenere e coniugare i servizi complementari ed interoperabili indispensabili allo sviluppo di un sistema informativo regionale all’avanguardia.

1.2.3 Storia del Consorzio Arsenà.IT

Arsenà.IT è la nuova denominazione del Consorzio Telemedicina, sorto nel 2005 in seguito ad una ricerca, avviata tre anni prima, dalla Regione Veneto sulle applicazioni di e-Health nel territorio regionale. Dall’indagine emerse che diverse aziende sanitarie avevano realizzato applicazioni di e-Health originali ed efficaci, con una tecnologia matura. Tuttavia, la varietà di approcci e di metodologie utilizzate presentava il limite dell’interoperabilità delle diverse applicazioni, pur appartenendo allo stesso ambito di trattamento clinico. L’impatto dell’applicazione tecnologica sull’organizzazione sanitaria non era stato valutato sistematicamente e si manifestavano resistenze nell’introduzione di tecnologie ICT nell’organizzazione aziendale in una parte del personale sanitario. Da questo studio prese avvio il desiderio dei Direttori Generali delle Aziende Sanitarie ed Ospedaliere della Regione Veneto di mettere a fattor comune le esperienze e le competenze per creare un Consorzio capace di governare in una logica di rete il processo di innovazione in atto. Oggi, Arsenà.IT, Centro Veneto Ricerca e Innovazione per la Sanità Digitale, ha ampliato l’ambito della sua azione oltre il ruolo di Osservatorio, per diventare Centro Studi Regionale nel settore del e-Health. La nuova denominazione Arsenà.IT, si richiama volutamente all’Arsenale di Venezia, modello di complesso industriale organizzato, nonché cuore dell’ingegneria navale di Venezia e della sua fortuna nel Mediterraneo

1.2.4 Mission e valori

Come organismo espressione di tutte e ventitré le Aziende Sanitarie ed Ospedaliere della Regione Veneto, il Consorzio Arsenà.IT ha come obiettivi primari quelli della progettazione, dello sviluppo e della valutazione di applicazioni di e-Health sovra-aziendali. Le competenze culturali e tecnologiche, che Arsenà.IT ha acquisito nella sua attività di approfondimento e confronto con gli operatori delle diverse aziende sanitarie e con quelli di organismi internazionali ed europei, con cui ha condiviso progetti di sanità elettronica e telemedicina, vengono poste a fattor comune di tutte le aziende consorziate, con attività finalizzate alla standardizzazione, all'interoperabilità dei sistemi, alla ricerca per l'innovazione, all'ingegneria dell'offerta e alla formazione. In questo senso, Arsenà.IT sviluppa modelli applicativi ed organizzativi di Sanità Elettronica, che, condivisi e promossi insieme alle aziende sanitarie regionali consorziate, costituiscono un network di conoscenza, competenza e tecnologia, unico nel panorama italiano dell'e-Health e mirano a migliorare l'efficienza dei processi organizzativi. La sua caratteristica di essere, poi, un luogo di elaborazione progettuale e di soluzione tecnologica dei problemi di governance sanitaria, nel campo dell'IT, unito all'essere Centro Studi, consente ad Arsenà.IT di mettere a disposizione dei suoi consorziati, come anche dell'intero Sistema Sanitario Nazionale, le best practices più innovative e avanzate, coerenti con la complessità gestionale ed operativa delle aziende consorziate. I valori, che sostengono l'azione del Consorzio, sono basati sul confronto e la condivisione progettuale, sullo scambio continuo di idee e suggerimenti, sull'adozione di sistemi applicativi "aperti", che permettano lo sviluppo ulteriore delle piattaforme di e-Health, in base alle esigenze di sviluppo delle aziende sanitarie venete, sulla valorizzazione delle soluzioni già presenti sul campo, fino al sostegno agli operatori, via via coinvolti nelle diverse attività progettuali, che vengono avviate. Tutto questo viene realizzato con il sovrascopo di sostenere, nel suo insieme, la crescita del Sistema Sanitario Regionale Veneto, che si presenta, oggi, in Italia, come quello tra i più avanzati ed efficienti, e di valorizzare ed ottimizzare gli investimenti assegnati per migliorare la qualità del servizio e dell'assistenza sanitaria al cittadino, aiutando a rimuovere gli ostacoli che impediscono il governo della domanda di servizi e prestazioni sanitarie e favorendo il governo dell'offerta dei livelli essenziali e uniformi di assistenza, rivolta a tutti cittadini residenti.

1.3 Contesto specifico e Open Issues

Un atto clinico complesso può essere concepito come un Workflow, configurato in modo tale da

poter perseguire un business clinico. Questo Workflow, che può essere esplicitamente definito come un Workflow Clinico, è il risultato di una composizione di servizi eterogenei, dislocati nel tempo e nello spazio, che si sviluppa passo dopo passo senza poter definire a priori lo specifico percorso seguito. Ogni Workflow clinico è caratterizzato da una propria tipologia specifica, che definisce una classe a cui afferisce e che ne determina in modo univoco delle regole di conformità che deve rispettare. Un approccio di questo tipo permette di raggiungere livelli di razionalizzazione dei processi tipici dell'approccio ingegneristico. L'applicazione della teoria dei Workflow al campo sanitario (e nello specifico di questo lavoro di tesi, al Progetto DOGE) soffre tuttavia dei forti limiti imposti sia dalla mancanza di linee guida adatte a supportare tali potenzialità. In particolare esiste un forte limite alla definizione dei Workflow Clinici connesso all'impossibilità di modificare i documenti clinici, i quali devono permanere inalterati dal momento della loro creazione. L'esempio che permette di osservare tali limiti in modo più immediato è lo sviluppo dei sistemi di prescrizione digitale. In questo caso sorge la necessità di cambiare lo stato della prescrizione durante lo sviluppo del Workflow ad essa correlato, in modo tale da poterne garantire un utilizzo corretto. Infatti, una volta che la prescrizione elettronica è stata creata, non viene più prodotto il supporto cartaceo che attesta il progresso del Workflow che si è generato. Con la prescrizione cartacea si può individuare lo stato di avanzamento del Workflow semplicemente individuando il possessore del documento. Questo non vale nel caso della prescrizione elettronica la quale, una volta prodotta, diventa disponibile ad un vasto range di farmacie per essere presa in carico. Una volta che il paziente si è recato nella farmacia di sua scelta dovrebbe essere possibile tracciare il nuovo step eseguito in modo che il documento di ePrescription non possa essere utilizzato successivamente. Tuttavia per definizione un documento clinico deve essere NON modificabile, e il metadata availabilityStatus (che può assumere solamente i valori "deprecated" o "approved") non permette di gestire la grande varietà di stati che la prescrizione può assumere durante il proprio ciclo di vita (può essere disponibile ad una farmacia per essere presa in carico, può essere presa in carico ma il medicinale può non essere dispensato oppure può essere presa in carico e il medicinale può essere dispensato ecc...). Lo stesso problema si può presentare in un vasto numero di use-cases più complessi, partendo dal Workflow correlato ad una richiesta di visita specialista fino alla gestione del Workflow originatosi a seguito di una richiesta di Teleconsulto.

1.4 Scopo della tesi

Il progetto svolto si colloca all'interno del processo di superamento dei limiti descritti nella sezione

precedente e connessi all'applicazione delle tecnologie ICT alla sanità mediante la definizione del profilo IHE XDW (Cross-Enterprise Document Workflow). In particolare l'obiettivo è quello di fornire un possibile approccio risolutivo a dei casi d'uso critici osservati all'interno del processo di sviluppo ed estensione del progetto "DOGE" e legati alla gestione del flusso dei processi clinico organizzativi. Il lavoro è stato svolto presso il Consorzio Arsenal.IT, con la finalità di definire la struttura dell'XDW Workflow Document, documento alla base della definizione delle linee guida contenute all'interno del nuovo profilo IHE XDW.

Il lavoro si è strutturato essenzialmente in due fasi distinte. In una prima fase si è deciso di valutare se tra i linguaggi XML descrittivi di Workflow presenti in letteratura fosse disponibile una soluzione in grado di ovviare ai problemi riscontrati. Una volta individuate delle soluzioni che totalmente o in parte avevano caratteristiche soddisfacenti si è proceduto in una seconda fase alla definizione di una struttura specifica del documento XDW Workflow Document.

Nello specifico sono state definite due possibili strutture per il Workflow Document, basate sull'utilizzo di standard differenti e che presentano potenzialità diverse. Per questo motivo è stata svolta anche un'indagine comparativa tra i due formati dell'XDW Workflow Document.

Di seguito viene presentata e descritta brevemente la struttura dell'elaborato.

Nel capitolo 2 si è descritto il progetto DOGE, all'interno del quale verranno applicati gli sviluppi conseguenti all'implementazione pratica del profilo XDW.

Nel capitolo 3 si è definito lo standard IHE, con particolare riguardo ai profili di interesse per il progetto DOGE, individuando quindi i limiti tutt'ora presenti per affrontare con completezza i problemi connessi alla sanità digitale.

Nel capitolo 4 è stato quindi introdotto il concetto di Workflow, con l'intenzione di estendere l'applicabilità dei linguaggi descrittivi dei Workflow specificatamente ai Workflow Clinici. L'obiettivo è stato quello di individuare l'eventuale esistenza in letteratura di standards che permettessero di tracciare la successione di steps svolti durante l'esecuzione di un'istanza di Workflow.

Partendo dalle informazioni raccolte nel capitolo 4 si sono definite nel capitolo 5 le linee guida per la definizione delle due versioni del Workflow Document, svolgendo contemporaneamente un'indagine comparativa delle due proposte, analizzandone l'applicabilità ad un semplice caso d'uso.

Nei capitoli 6 e 7 sono presentate nel dettaglio le strutture delle due versioni del documento. Infine nel capitolo 8 sono presentate le conclusioni sul progetto, ponendo particolare risalto ai problemi risolti ed ai problemi ancora aperti.

Nelle APPENDICI A e B sono state definite le specifiche tecniche dei linguaggi utilizzati per la costruzione dei due documenti, mentre nelle APPENDICI C e D sono state definite informazioni tecniche necessarie per l'implementazione effettiva dei corrispondenti Workflow Document.

2.Il Progetto DOGE:

Presentazione e problemi aperti di interesse

2.1 Descrizione del Progetto “Doge”

DOGE è un progetto regionale finalizzato alla creazione di una rete di servizi per i professionisti sanitari di primary care (medici di medicina generale e pediatri di libera scelta). DOGE si colloca nel contesto del sistema di monitoraggio della spesa sanitaria, come disciplinato dall'art. 50 della legge 326 del 2003 e ha l'obiettivo di definire un'infrastruttura di servizi applicativi per il collegamento in rete degli operatori e la trasmissione telematica al Ministero dell'Economia e delle Finanze dei dati delle ricette del Servizio sanitario Nazionale. L'iniziativa, realizzata in collaborazione con i sistemi informativi dell'Azienda ULSS 18 Rovigo, è stata affidata all'Azienda ULSS19 Adria in considerazione delle caratteristiche territoriali legate ad un ampio territorio con una bassa densità abitativa e la conseguente, elevata dispersione geografica della rete dei servizi.

La Regione Veneto con delibera n. 2322 del 28 luglio 2009, “Analisi e progettazione del modello di architettura tecnologica per lo sviluppo del SAR (Sistema di Accoglienza Regionale) ha conferito l'incarico di consulenza e progettazione della rete al Consorzio Arsenà.IT ed ha inserito questo progetto in una realizzazione più ampia per la costruzione del Fascicolo Sanitario Personale. La progettazione si traduce in uno studio di valutazione e fattibilità per individuare le caratteristiche tali da rispondere agli adempimenti previsti dalla Legge n.34/2007 “Norme in materia di tenuta, informatizzazione e conservazione delle cartelle cliniche e sui moduli di consenso informato” e sufficienti all'individuazione di requisiti minimi per la costruzione del Fascicolo Sanitario Personale Regionale.

Ad Arsenà.IT è stato assegnato anche il compito dell'analisi, progettazione, monitoraggio e supporto all'avviamento del modello tecnologico per la gestione del consenso e della privacy, all'interno delle attività previste nella DGRV 3889 del 9 dicembre 2008 in attuazione della Legge Regionale n.34/2007. Il sistema, realizzato in collaborazione con Medici di Medicina Generale (MMG) e Pediatri di Libera Scelta (PLS), consente l'interazione costante tra gli operatori di

primary care e l'Azienda Sanitaria, per rendere immediatamente disponibili agli operatori autorizzati le informazioni clinico-amministrative riguardanti la popolazione assistita e favorendo la continuità assistenziale.

La fase sperimentale del progetto nell'Azienda ULSS 19 Adria è durata due anni con l'ingresso a regime nel 2011 ed ha coinvolto circa settanta operatori.

In linea con gli orientamenti più evoluti in campo di informatizzazione dei servizi sanitari, il progetto prevede la realizzazione di una rete di servizi in grado di soddisfare le esigenze dei vari attori coinvolti nel processo di cura, che, attraverso la condivisione del dato clinico da parte degli operatori autorizzati, aumenta il livello di continuità assistenziale. In tal modo il cittadino è posto al centro dei processi di diagnosi e cura e, contemporaneamente, vengono soddisfatte, in conformità alle DPCM, le richieste di trasmissione dei dati di natura sanitaria e amministrativa al Sistema Sanitario Nazionale.

Due gli obiettivi individuati:

-Rete MMG: lo sviluppo di un sistema finalizzato a consentire un'interazione costante tra MMG/PLS ed Azienda Sanitaria, che renda immediatamente disponibili agli attori interessati le informazioni clinico-amministrative riguardanti gli assistiti, garantendo la sicurezza e riservatezza dei dati trattati. Il progetto prevede l'automazione, attraverso gli strumenti dell'ICT, dei flussi informativi relativi all'allineamento anagrafico, incluse le informazioni relative alle esenzioni e alla scelta e revoca del medico, alle notifiche di ricovero, alle prescrizioni, all'invio dei referti, ecc. in modo da aumentare significativamente la tempestività e l'accuratezza delle comunicazioni, a vantaggio sia degli assistiti che degli adempimenti verso il MEF.

-Ciclo prescrittivo: prioritario nel progetto individuare e sviluppare soluzioni che consentano di dare attuazione alle disposizioni in materia di acquisizione telematica dei dati delle ricette di cui al comma 5-bis dell'art. 50 della Legge n. 326/2006, in relazione all'opzione di adesione al comma 11 dell'art. 50 esercitata dalla Regione Veneto ai sensi dell'art. 4, comma 1, del DPCM 26 marzo 2008.

In linea con le indicazioni di architettura tecnologica formulate a livello nazionale, per minimizzare l'impatto del progetto sull'operatività corrente degli attori coinvolti, tali obiettivi sono conseguiti attraverso gli strumenti della cooperazione applicativa, e dunque rendendo interoperabili i sistemi informatizzati dell'Azienda Sanitaria e dei MMG/PLS. Ne deriva che il progetto non prevede l'introduzione di nuovi e diversi strumenti, in sostituzione di quelli già in uso, bensì definisce le regole e le modalità tecniche per l'attivazione di flussi di comunicazione adeguati tra i sistemi esistenti, idonei ad essere applicati in ogni contesto organizzativo. Grande attenzione è stata

dedicata all'attività di allineamento dei sistemi di codifica mediante l'impiego di standard riconosciuti.

Una volta analizzato il quadro delle esperienze esistenti a livello nazionale e regionale, è stata definita l'architettura tecnologica del sistema da implementare, assicurandone la compatibilità con altre esperienze in corso e la conformità agli standard riconosciuti.

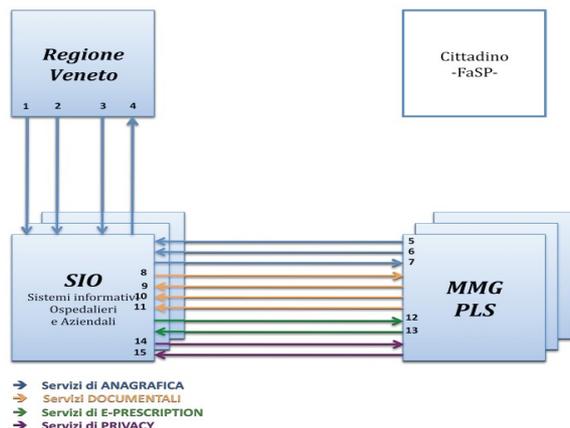
La sperimentazione ha previsto la trasmissione, direttamente da parte di MMG e PLS al presidio ospedaliero di riferimento, della prescrizione di esame diagnostico, visita specialistica e farmaceutica. La richiesta viene visualizzata ed elaborata dall'operatore del CUP. In seguito all'erogazione della prestazione MMG o PLS riceve dall'Azienda Sanitaria la notifica di pubblicazione del referto (ad esempio di laboratorio, di visita specialistica, ecc.) ed è in grado di importare il documento nel suo sistema di cartella clinica. Il progetto prevede inoltre l'attivazione di una serie di servizi di anagrafica attraverso i quali gli operatori di primary care ricevono dall'anagrafe dell'Azienda Sanitaria periodici aggiornamenti sulle variazioni dei dati anagrafici dei loro assistiti (ad esempio eventuali esenzioni, cambi di residenza, ecc.). L'implementazione di questi flussi permette un'integrazione efficiente anche con il Sistema di Anagrafe Regionale degli Assistiti già operativo nelle Aziende della provincia di Rovigo e in fase di diffusione in tutta la Regione .

La costruzione di questa rete di servizi, nel suo complesso, pone le basi per la realizzazione del Fascicolo Sanitario Personale (FaSP), nel quale sarà contenuta tutta la storia clinica del cittadino, in accordo con le linee guida ad oggi in fase di stesura da parte di un tavolo tran-istituzionale di cui fanno parte Regioni, Ministero della Salute e Garante della Privacy. La piattaforma cooperativa e le linee guida del Progetto DOGE risultano perfettamente in linea con il progetto Veneto ESCAPE per la digitalizzazione dei documenti clinici, curato da Arsenà.IT.

2.2 Attori e Servizi nella rete MMG/PLS

In questa sezione vengono definiti i servizi offerti e gli attori coinvolti nella rete MMG/PLS della Regione Veneto. La Figura 1.1 mostra come il progetto della rete MMG/PLS viene ad inserirsi nel disegno più generale per la realizzazione di un'architettura complessa e integrata per il Sistema Sanitario Regionale.

Fig. 2.1: Attori e servizi offerti dal SSR del Veneto



Strategia architettonica che vede coinvolti tutti i soggetti che partecipano al processo di cura: il cittadino, gli MMG/PLS, le Aziende Sanitarie e in aggiunta la Regione che non rappresenta un attore coinvolto in prima linea nel processo di cura ma che verrà ad assumere un ruolo di controllo e governo sull'intera rete e sui dati sanitari prodotti.

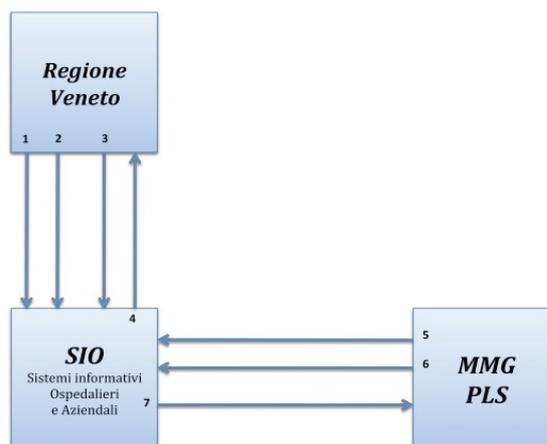
La rete degli MMG costituisce una parte importante e rilevante del disegno complessivo offrendo tutti quei servizi che andranno ad integrare tra loro Regione, SIO e MMG/PLS (cioè 3 dei 4 attori coinvolti). Il progetto è finalizzato a garantire la continuità di cura del paziente e realizzare un effettivo innalzamento qualitativo di assistenza sanitaria, migliorando il coordinamento tra i professionisti e le strutture sanitarie nella circolazione e condivisione dei dati clinici. I servizi offerti dal sistema possono essere suddivisi in quattro classi:

- SEZIONE ANAGRAFICA
- SEZIONE DOCUMENTALE
- SEZIONE E-PRESCRIPTION
- SEZIONE PRIVACY

2.2.1 Servizi anagrafici

In questa sezione vengono presentati tutti i servizi che si occupano della gestione dei dati anagrafici sia per la comunicazione tra l'anagrafe regionale e le varie anagrafiche locali dei SIO, sia tra SIO e MMG/PLS.

Figura 2.2: Schema Servizi Anagrafici



1. INSERIMENTO NUOVA POSIZIONE ANAGRAFICA: questo servizio si prende incarico di comunicare alle anagrafiche locali l'inserimento di una nuova posizione anagrafica per:

- iscrizione anagrafica (comunicazione dal comune per nascita o immigrazione da fuori Regione Veneto)
- iscrizione sanitaria (inserimento da parte dell'operatore)

2. AGGIORNAMENTO E MODIFICA DELLE INFORMAZIONI: servizio che permette:

- l'aggiornamento della posizione anagrafica dell'assistito per:
 - aggiornare un dato anagrafico dell'assistito;
 - aggiornare l'iscrizione sanitaria (SSN, ULSS);
 - aggiornare le esenzioni attive dell'assistito;
 - aggiornare un dato anagrafico del medico;
 - aggiornare il rapporto del medico con l'ULSS, inserire un nuovo rapporto in sostituzione del precedente, cancellare il rapporto
- cancellazione della posizione anagrafica dell'assistito o del medico per:
 - inserimento posizione anagrafica doppia;
 - errata iscrizione;
- Variazione e annullamento del medico curante per:
 - prima scelta del medico MMG/PLS;
 - cambio del medico MMG/PLS;
 - revoca del medico MMG/PLS;
 - cessazione del medico MMG/PLS;

3. **MERGE E UNMERGE DI DUE POSIZIONI:** servizio che permette l'unificazione di una coppia di posizioni anagrafiche/annullamento dell'unificazione di una coppia di posizioni anagrafiche (ad esempio nel caso sia stata creata una nuova posizione B e solo in un secondo momento ci si accorge che la medesima persona era già presente in anagrafe alla posizione A, in questo caso le due posizioni A e B vengono fuse).

4. **QUERY PER RICERCA DI UNA POSIZIONE ANAGRAFICA** è l'unico servizio a fluire in senso inverso cioè dal SIO alla Regione, si tratta infatti di query da parte di Azienda Sanitaria per: o ricerche di posizioni anagrafiche in anagrafe regionale o aggiornare l'anagrafe locale dei medici.

5. **QUERY PER RICERCA ANAGRAFICA DI UN SINGOLO PAZIENTE** questo servizio consente al MMG di interrogare l'azienda sanitaria di riferimento per poter avere i dati anagrafici di un singolo assistito e le sue relative informazioni.

6. **QUERY MASSIVA:** servizio che consente di interrogare l'azienda sanitaria di riferimento per poter avere tutti i dati anagrafici e ulteriori informazioni di tutti gli assistiti dell'MMG/PLS, non solo al primo avvio ma tutte le volte che è richiesto dal medico permettendo di ricaricare l'intera sua anagrafica.

7. **NOTIFICA AGGIORNAMENTI:** tramite questo servizio applicativo l'utente riceve le notifiche generate dal SIO. Queste vengono scatenate in seguito ad eventi di tipo diverso che danno luogo a differenti tipi di notifiche:

- notifiche di aggiornamenti anagrafici (aggiornamenti anagrafica assistito, nuove scelte e revoche);
- notifiche di aggiornamento per sostituzione;
- notifiche di aggiornamento per allineamento nuovo medico;
- notifiche di avvenuto ricovero;
- notifiche di dimissione;

2.2.2 Servizi di condivisione documentale

In questa classe rientrano tutti i servizi che permettono la condivisione documentale tra SIO e MMG/PLS, utilizzando il profilo XDS e le sue varianti tipo XDS-I e XCA per lo scambio interprovinciale.

Figura 2.3: Schema Servizi di condivisione documentale



8. **NOTIFICA DOCUMENTI (DSUB)**: servizio che notifica all'MMG/PLS la messa in condivisione di un documento clinico (es. referto, lettera di dimissione,..) generato dall'Azienda Sanitaria.

9. **QUERY DOCUMENTI (XDS)**: servizio invocato dall'MMG/PLS per ottenere i parametri necessari al recupero del documento.

10. **RETRIEVE DOC. (XDS)**: permette di recuperare il documento di interesse indipendentemente dalla struttura ove è in carico.

11. **SUBMIT DOC. (XDS)**: servizio che abilita l'MMG/PLS ad inviare il *Patient Summary* o altri documenti da lui prodotti alla propria Azienda Sanitaria. (ATNA in fase di definizione).

2.2.3 Servizi di e-Prescription

In questa sezione vengono presentati i servizi coinvolti nel ciclo prescrittivo di competenza degli MMG/PLS e SIO.

Figura 2.4: Schema Servizi di E-prescription



12. **CREAZIONE e-PRESCRIPTION (XDS)**: si occupa della generazione della prescrizione elettronica da parte di un MMG/PLS o di un medico operante in un'Azienda Sanitaria.

13. **NOTIFICA e-PRESCRIPTION (DSUB)**, all'MMG/PLS, per:

- Creazione di una prescrizione da parte di un medico prescrittore non MMG/PLS;
- presa in carico della prescrizione da parte di qualsiasi SIO regionale e notifiche di cambio stato della prescrizione.

2.2.4 Servizi di privacy

In quest'ultima sezione vengono presentati i servizi coinvolti nel trattamento dei consensi dati dai cittadini per la trattazione dei dati personali sia anagrafici che clinici.

Figura 2.5: Schema Servizi di Privacy



14. **RACCOLTA CONSENSI (XDS-BPPC)**: i consensi vengono raccolti dagli operatori a livello di SIO ed inviati al Repository e al Registry dove vengono raccolti, seguendo lo stesso workflow previsto dal profilo XDS per i documenti.

15. **CONSULTAZIONE CONSENSO** : la visualizzazione dei consensi da parte dei medici è possibile, attraverso l'uso della stessa architettura usata per la condivisione documentale (profilo XDS).

2.3 Ciclo Prescrittivo Digitale

All'interno degli svariati servizi offerti dalla piattaforma DOGE, riveste un ruolo fondamentale per giustificare il lavoro di tesi svolto, la gestione del ciclo prescrittivo digitale, per la quale la gestione del workflow dei documenti associati diventa fondamentale per il corretto funzionamento del

sistema stesso.

Questo capitolo presenta l'architettura studiata per la gestione dei flussi delle prescrizioni elettroniche prodotte da MMG/PLS e medici ospedalieri.

2.3.1 Inquadramento nazionale della soluzione proposta dalla Regione Veneto

Il provvedimento normativo DPCM 26 Marzo 2008 attuativo del D.L. 296/2006 art.1 comma 810 lettera C, dispone le modalità di trasmissione telematica dei dati delle prescrizioni, da parte dei singoli medici (MMG, PLS, specialisti) o da parte delle Regioni aderenti all'art 4 comma 1, al Ministero dell'economia e delle finanze e pone le basi per la ricetta elettronica.

Si colloca nel contesto del sistema di monitoraggio della spesa sanitaria, art.50 della legge 24 novembre 2003, n.326, e definisce un'infrastruttura di servizi applicativi per il collegamento in rete dei medici e per la trasmissione telematica dei dati. Tale infrastruttura, definita SAC, consiste in un sistema di accoglienza centrale per la raccolta dei dati relativi alle prescrizioni mediche inviati in modalità di cooperazione. Il SAC è raggiungibile attraverso SPC e rete internet. Il MEF mette a disposizione la propria infrastruttura di rete per consentire la connessione gratuita al SAC (Entratel). La Regione Veneto aderisce al comma 1 dell'art. 4 del DPCM 26/03/2008 di attuazione dell'art.1, comma 810, lettera c), della legge 27 dicembre 2006, n.296, in materia di regole tecniche e trasmissione dati di natura sanitaria nell'ambito del Sistema pubblico di connettività, pubblicato in G.U. n.124 del 28/05/2008. In tale ambito la Regione Veneto si fa carico della gestione del Sistema di Accoglienza Regionale (SAR) per la gestione delle impegnative.

2.3.2 Architettura del Sistema per la Prescrizione Elettronica

Il sistema di Prescrizione Elettronica va inserito oltre che nel programma per il controllo della spesa anche nel contesto generale di fornitura di servizi al cittadino inteso come agevolazione delle modalità di fruizione del servizio a partire dalla prenotazione, per passare all'erogazione, al pagamento ed al ritiro del referto o notifica all'MMG, ove previsto, secondo lo schema generale.

In tale schema l'elemento che costituisce il legame fra i vari ambiti del percorso diagnostico/terapeutico del paziente è costituito dall'identificativo della prescrizione che ne rappresenta il punto iniziale.

Assodato che per tale identificativo (NRE, Numero di Ricetta Elettronico) risulta non utilizzabile il numero prestampato sull'impegnativa, in quanto presuppone una lettura a posteriori del codice a barre che lo rappresenta e la conseguente associazione all'impegnativa digitalmente prodotta.

Operazione che sia pur tecnicamente possibile rappresenta un notevole aggravio in termini operativi del prescrittore con il pericolo di un'alta possibilità di errore.

Sulla base delle riflessioni di cui sopra, si è optato quindi, per un'assegnazione automatica dell'NRE gestita con modalità differenti nel caso si tratti di MMG o medici specialisti. Il primo caso prevede un'assegnazione diretta del NRE tra SOGEI e l'MMG, nel secondo caso è prevista invece una gestione centrale che ripercorre essenzialmente le stesse fasi ad oggi seguite per l'assegnazione dei ricettari cartacei.

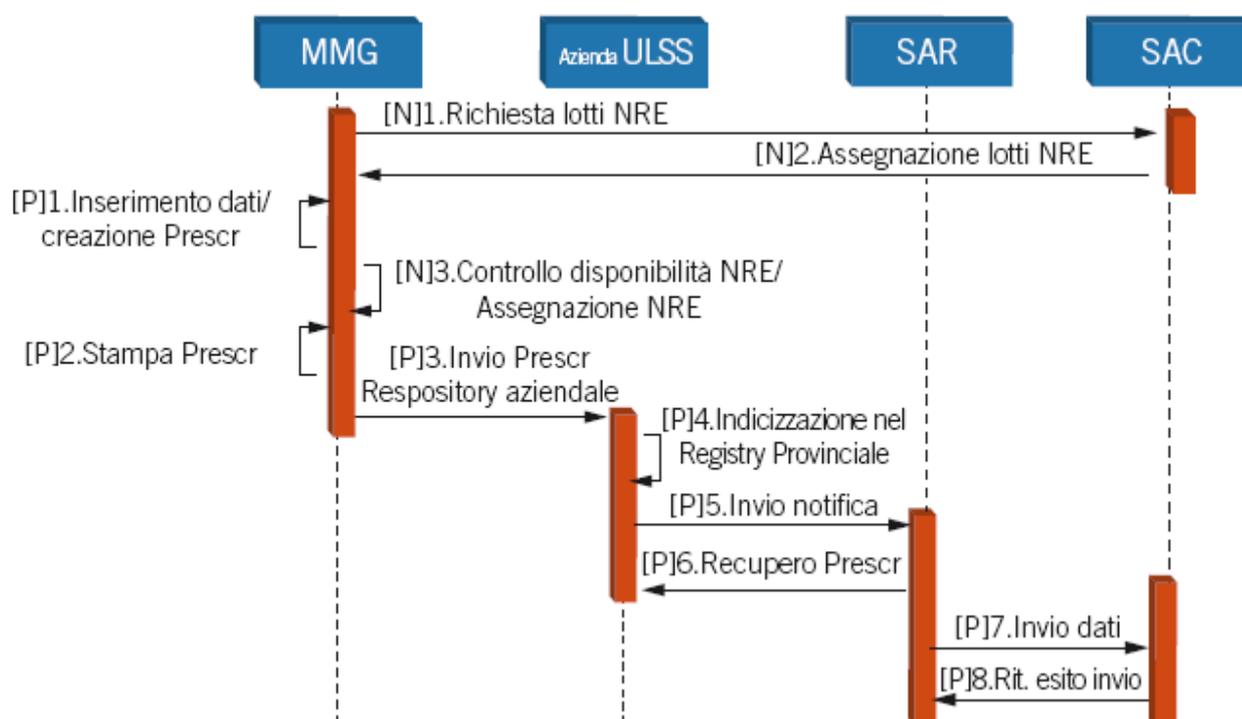
Nel primo caso è l'MMG, utilizzando il servizio messo a disposizione dal sistema centrale, a richiedere direttamente l'assegnazione di un lotto di NRE, in questo caso il lotto assegnato è automaticamente associato al Codice Fiscale dell'MMG/PLS richiedente. Nel secondo caso ci si basa invece sull'assegnazione da parte di SOGEI di un range di numeri associati a lotti, gestiti centralmente dalle Aziende Sanitarie ed Ospedaliere. Tali numeri di ricetta vengono assegnati ai prescrittori con modalità diverse, ma garantendo in ogni caso l'associazione tra NRE, Codice Fiscale del medico prescrittore e conseguente prescrizione prodotta. È garantita inoltre la comunicazione dell'associazione NRE-CF medico prescrittore al SAC attraverso l'apposito servizio.

2.3.3 Gestione ePrescription Specialistica/Farmaceutica prodotta da un MMG/PLS

Si illustra di seguito il flusso completo riguardante la gestione del ciclo prescrittivo per ePrescription specialistiche/farmaceutiche prodotte da MMG/PLS. Il workflow completo comprende sia il flusso di gestione dell'NRE, che il flusso di generazione e gestione della prescrizione digitale. Per quello che riguarda la gestione della prescrizione digitale questo capitolo si focalizza solo sul flusso di informazioni che riguardano l'invio telematico al SAC (con tutti gli step intermedi connessi), mentre per la gestione del workflow del documento digitale di prescrizione (CUP, ecc.) si rimanda al documento di progetto "Progetto DOGE".

Di seguito si riporta il Diagramma Sequenziale di gestione NRE e generazione ePrescription da parte di MMG/PLS.

Figura 2.6: Diagramma di prescrizione Elettronica prodotta da MMG/PLS



Nella tabella sottostante si descrivono le azioni riportate nella figura precedente.

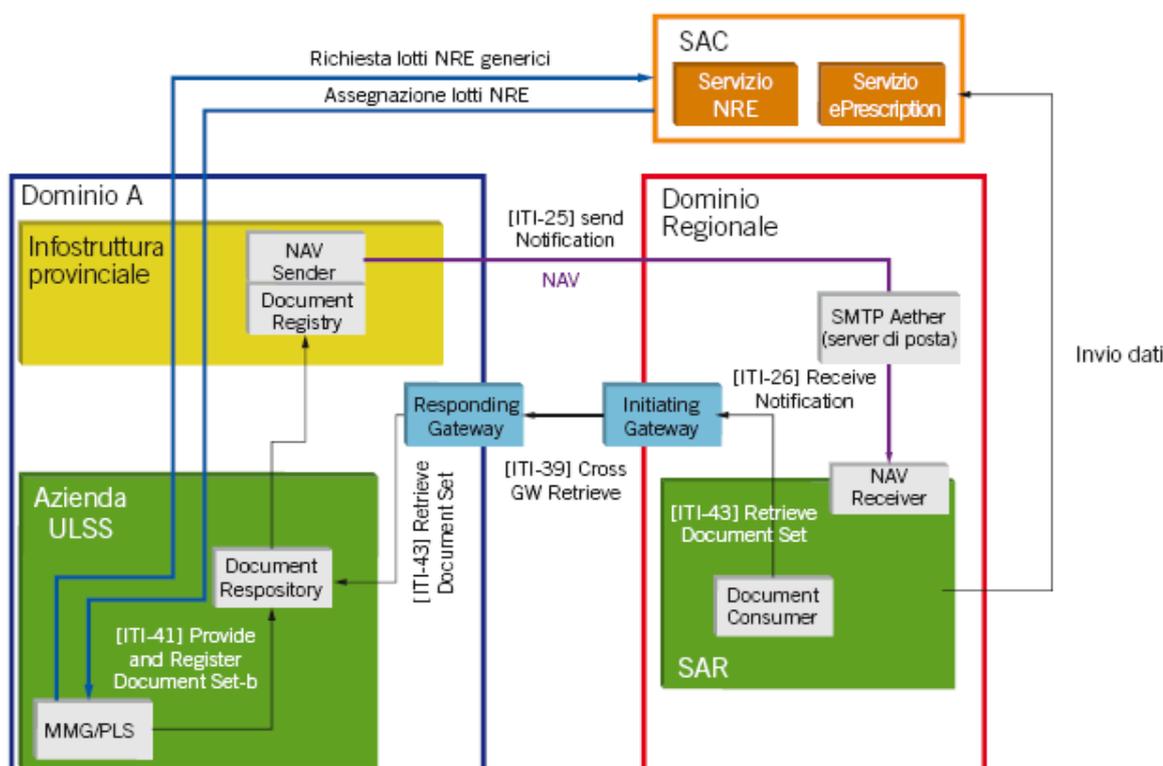
Descrizione Azioni per ePrescription prodotta da MMG/PLS		
Azione	Descrizione	Attore
[N]1. Richiesta lotti NRE	Attraverso il servizio messo a disposizione dal MEF e autenticandosi al sistema con le stesse credenziali usate per l'invio dei certificati di malattia, il MMG richiede un lotto di NRE. Parametri: -CodRegione 050 -Identificativo Lotto 2 o 3 -CF Medico	Prescrittore
[N]2. Assegnazione lotti NRE	Il SAC fornisce il codice del lotto assegnato.	SAC
[P]1. Inserimento dati/ creazione prescrizione	Funzionalità di compilazione della prescrizione e creazione del CDA2 a carico dei singoli applicativi in uso ai	Prescrittore

	<p>prescrittori.</p> <p>L'identificazione anagrafica del paziente deve avvenire in accordo con le direttive aziendali in merito all'utilizzo dell'anagrafe certificata.</p> <p>Deve implementare le regole amministrative per la corretta compilazione delle impegnative:</p> <ul style="list-style-type: none"> -Gestione esenzioni per patologia; -Gestione esenzioni per reddito; -Codifica prestazioni aziendali; -Gestione numero massimo prestazioni per singola prescrizione con vincolo della branca e del RAO se applicato; -Gestione numero massimo prodotti farmaceutici per prescrizione. 	
[N]3. Controllo disponibilità NRE/ assegnazione NRE	Verifica disponibilità all'interno della gestione del lotto di un numero NRE e associazione dell'NRE alla prescrizione.	Prescrittore
[P]2. Stampa prescrizione	L'applicativo procede alla stampa dell'impegnativa secondo il template previsto.	Prescrittore
[P]3. Invio prescrizione nel Repository aziendale	L'applicativo del prescrittore come Document Source procede attraverso la transazione [ITI-41] Provide and Register Document Set-b, alla pubblicazione del CDA2 di prescrizione nel Repository aziendale.	Prescrittore

Azione	Descrizione	Attore
[P]4. Indicizzazione del Registry provinciale	Il Repository aziendale provvede attraverso la transazione [ITI-42] Register Document Set-b, all'indicizzazione nel Registry provinciale.	Azienda (SIO)
[P]5. Invio notifica	L'evento di indicizzazione nel Registry fa da trigger per l'invio di notifica al SAR della presenza di una nuova prescrizione. Questa notifica è un messaggio smtp che conterrà una notifica completa (tutti i metadati) necessaria poi al SAR per effettuare un retrieve diretto del documento di prescrizione dal Repository aziendale dove è presente.	Azienda (SIO)
[P]6. Recupero Prescr	L'applicativo del SAR attraverso una transazione [ITI-43] Retrieve Document Set, inter-community (passando attraverso le porte Gateway con una transazione [ITI-39] Cross GW Retrieve) procede al recupero della prescrizione. A tendere questo potrà avvenire o contestualmente ad ogni evento di notifica o in modalità asincrona una volta al giorno. L'applicativo del SAR recuperato il documento di prescrizione garantisce la mappatura dei dati dal CDA2 di prescrizione al tracciato richiesto dal SAC.	Sistema SAR
[P]7. Invio dati	L'invio avviene richiamando l'apposito servizio del SAC. A regime l'invio dei dati delle prescrizioni al SAC verrà schedato secondo una ripetizione giornaliera.	Sistema SAR
[P]8. Ritorno esito invio	All'avvenuto invio dei dati il sistema SAC risponde con un messaggio di esito dell'invio e un numero di protocollo.	SAC

Nel diagramma di flusso sottostante sono rappresentati tutti gli attori che concorrono al sistema. Come si nota l'architettura interna regionale è basata sull'utilizzo di un'infrastruttura realizzata secondo lo sviluppo dei profili IHE XDS.b (Cross Enterprise Document Sharing) per le parti intra-provinciacie e il profilo IHE XCA (Cross Community Access) per l'interoperabilità inter-provinciale. I flussi informativi illustrati devono essere realizzati secondo i profili XDS.b, XCA, utilizzando altresì i seguenti profili ad essi correlati: - CT (Consistent Time) - ATNA (Audit Trail and Node Authentication) - DSUB (Document Metadata Subscription) - NAV (Notification of Document Availability) I documenti che descrivono tali profili sono di pubblico dominio e sono disponibili all'indirizzo www.ihe.net, ai quali si rimanda per ogni specifica tecnica relativa al set di profili standard qui adottati.

Figura 2.7: Diagramma di flusso Prescrizioni Elettroniche prodotta da MMG/PLS

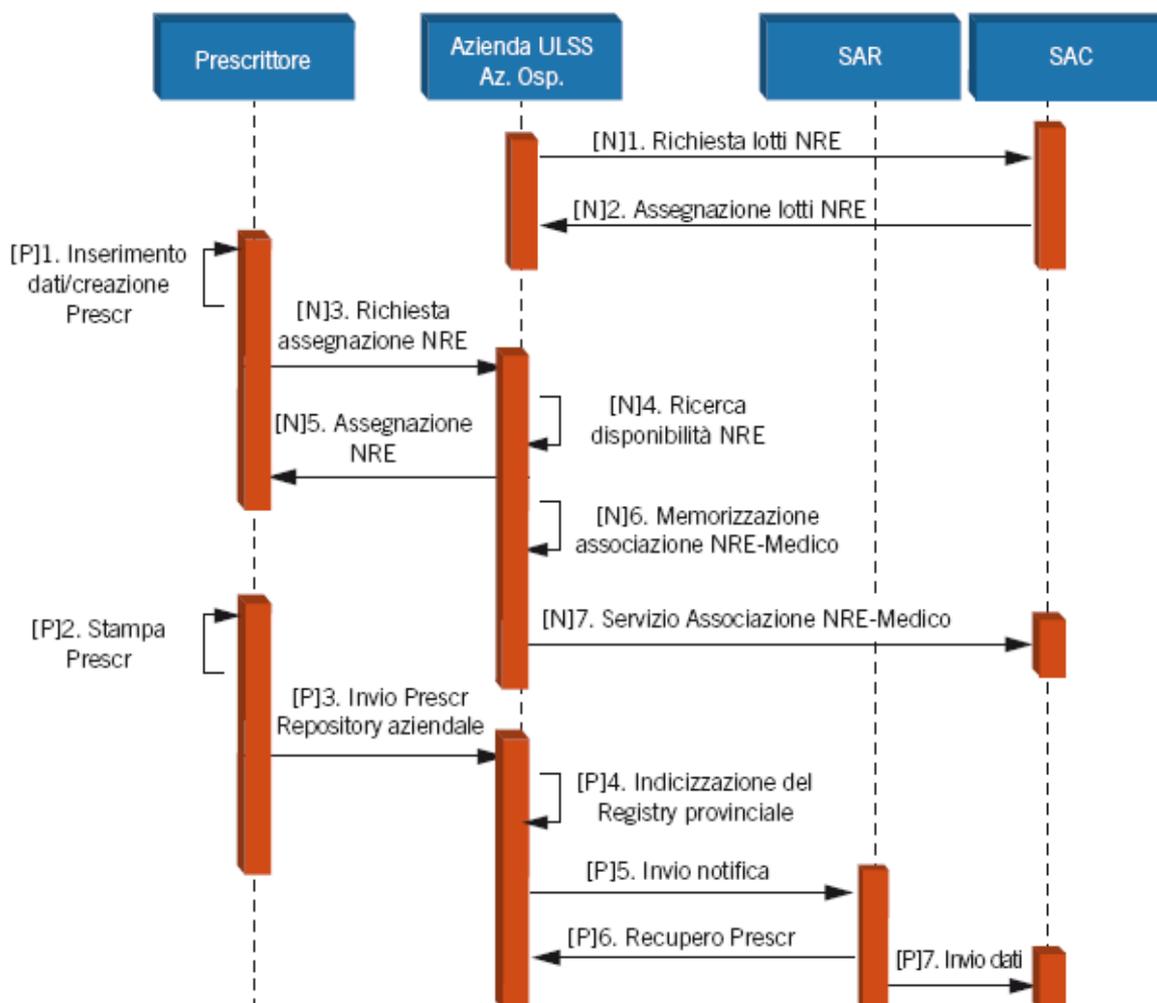


2.3.4 Gestione del Ciclo Prescrittivo Ospedaliero

Si illustra di seguito il flusso completo riguardante la gestione del ciclo prescrittivo per i medici prescrittori specialisti/interni ospedalieri. Il workflow completo comprende sia il flusso di gestione dell'NRE che il flusso di generazione e gestione della prescrizione digitale. Per quello che riguarda

la gestione della prescrizione digitale questo capitolo si focalizza solo sul flusso di informazioni che riguardano l'invio telematico al SAC (con tutti gli step intermedi connessi), mentre per la gestione del workflow del documento digitale di prescrizione (CUP, ecc.) si rimanda al documento di progetto "Progetto DOGE". 11 Di seguito si riporta il Diagramma Sequenziale di gestione NRE e generazione Prescrizione per i prescrittori interni ospedalieri.

Figura 2.8: Diagramma Prescrizione Elettronica Ospedaliera



Nel caso in cui non si ritenga di attivare la procedura di assegnazione di un lotto NRE ad un medico in quanto la specifica attività non comporta una produzione continua di prescrizioni o perché la distribuzione è gestita dall'Azienda Sanitaria od Ospedaliera si può procedere assegnando di volta in volta al medico un NRE per la specifica prescrizione. Sarà poi a carico dell'Azienda fornire al SAC l'assegnazione NRE-Medico.

Nella tabella sottostante si descrivono le azioni riportate nella figura precedente.

Descrizione Azioni per Prescrizione Elettronica Ospedaliera		
Azione	Descrizione	Attore
[N]1. Richiesta lotti NRE CodRegione 050 / CodAzienda Identificativo Lotto 3 (o 4)	Attraverso il servizio messo a disposizione dal MEF l'Azienda farà richiesta per un lotto di numeri da gestire poi internamente la distribuzione. Parametri: CodRegione 050 / CodAzienda Identificativo Lotto 3 (o 4)	Azienda (SIO)
[N]2. Assegnazione lotti NRE	Il SAC fornisce il codice del lotto assegnato	SAC
[P]1. Inserimento dati/ creazione prescrizione	Funzionalità di compilazione della prescrizione e creazione del CDA2 a carico dei singoli applicativi in uso ai prescrittori. Deve garantire l'autenticazione dell'utente tramite CNS o Carta Operatore e a tendere gestire la firma digitale. L'identificazione anagrafica del paziente deve avvenire in accordo con le direttive aziendali in merito all'utilizzo della anagrafe certificata. Deve implementare le regole amministrative per la corretta compilazione delle impegnative: -Gestione esenzioni per patologia; -Gestione esenzioni per reddito; -Accesso anagrafe prestazioni aziendali; -Accesso anagrafe farmaci; -Gestione numero massimo prestazioni per singola prescrizione con vincolo della branca e del RAO se applicato; -Gestione numero massimo prodotti	Prescrittore

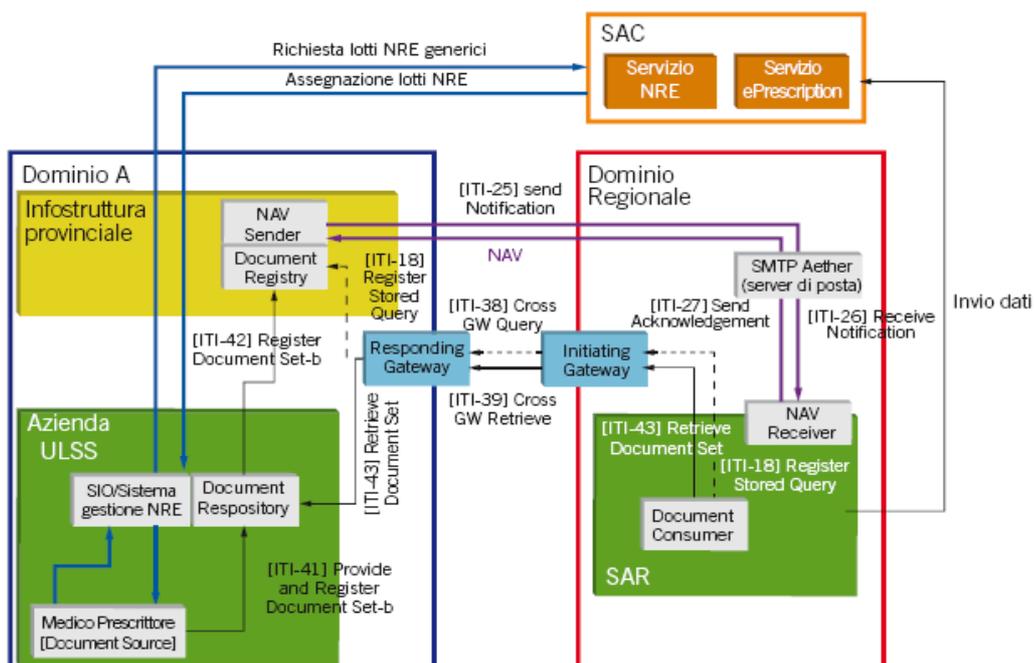
		farmaceutici per prescrizione.	
[N]3. Richiesta Assegnazione NRE		L'applicativo del prescrittore invoca il servizio Aziendale per farsi assegnare un NRE per quella prescrizione . Parametri obbligatori: -CF Medico Prescrittore	Prescrittore

Azione	Descrizione	Attore
[N]4. Ricerca disponibilità NRE	Ricerca e verifica disponibilità all'interno dell'archivio dei lotti di un numero NRE per il medico prescrittore.	Azienda (SIO)
[N]5. Assegnazione NRE	Viene assegnato dal lotto complessivo gestito dall'Azienda un NRE a quel medico richiedente.	Azienda (SIO)
[N]6. Memorizzazione associazione NRE_Medico	Viene tenuta traccia in un apposito database dell'assegnazione dei singoli NRE al CF del medico a cui sono stati assegnati.	Azienda (SIO)
[N]7. Servizio Associazione NRE-Medico	L'Azienda notifica al SAC tramite l'apposito servizio l'Associazione NRE-Medico. Parametri: -Identificativo NRE -Identificativo Lotto -CF Medico: CF del medico a cui è stato associato quell'NRE -DataAttribuzione	Azienda (SIO)
[P]2. Stampa Prescrizione	L'applicativo procede alla stampa dell'impegnativa secondo uno dei due template previsti (Template A o Template B)	Prescrittore
[P]3. Invio Prescrizione nel Repository aziendale	L'applicativo del prescrittore come Document Source procede attraverso la transazione [ITI-41] Provide and Register Document Set-b, alla pubblicazione del CDA2 di prescrizione	Prescrittore

	nel Repository aziendale	
[P]4. Indicizzazione del Registry provinciale	Il Repository aziendale provvede attraverso la transazione [ITI-42] Register Document Set-b, all'indicizzazione nel Registry provinciale	Azienda (SIO)
[P]4. Invio notifica	L'evento di indicizzazione nel Registry fa da trigger per l'invio di notifica al SAR della presenza di una nuova prescrizione. Questa notifica sarà un messaggio smtp che conterrà una fullNotification (tutti i metadati) necessaria poi al SAR per effettuare un retrieve diretto del documento di prescrizione dal Repository aziendale dove è presente.	Azienda (SIO)

Nel diagramma di flusso sottostante sono rappresentati tutti gli attori che concorrono al sistema. Come si nota l'architettura interna regionale è basata sull'utilizzo di un'infrastruttura realizzata secondo lo sviluppo dei profili IHE XDS.b (Cross Enterprise Document Sharing) per la parte intra-provinciale e il profilo IHE XCA (Cross Community Access) per l'interoperabilità inter-provinciale. I flussi informativi illustrati devono essere realizzati secondo i profili XDS.b, XCA, utilizzando altresì i seguenti profili ad essi correlati: - CT (Consistent Time) - ATNA (Audit Trail and Node Authentication) - DSub (Document Metadata Subscription) - NAV (Notification of Document Availability) I documenti che descrivono tali profili sono di pubblico dominio e sono disponibili all'indirizzo www.ihe.net, ai quali si rimanda per ogni specifica tecnica concernente il set di profili standard qui adottati.

Figura 2.9: Diagramma di flusso Prescrizione Elettronica Ospedaliera



2.3.5 Scenario funzionale: Primo caso d'uso

L'utilizzo del sistema in oggetto risponde a diversi casi d'uso. In questa sezione viene illustrato il primo e più comune. Attori coinvolti:

- un paziente che si rivolge al proprio MMG/PLS;
- l'MMG/PLS che ha in carico il paziente;
- l'Azienda Sanitaria a cui afferiscono sia il paziente che il proprio MMG/PLS.

Il paziente si rivolge al proprio MMG/PLS per farsi prescrivere una prestazione diagnostica specialistica (visita e/o esami). In questo caso il medico compila la prescrizione e la invia all'Azienda Sanitaria a cui afferisce. Una volta che il paziente si rivolge alla propria Ulss (che in questo primo caso coincide con la Ulss di appartenenza del medico) per farsi erogare la prestazione, il sistema CUP prende visione della e-Prescription creata dal medico e prenota la prestazione.

Eseguita la visita eventuali referti prodotti e relativi esiti degli esami potranno, in un secondo momento, essere pubblicati e allegati alla prescrizione di richiesta.

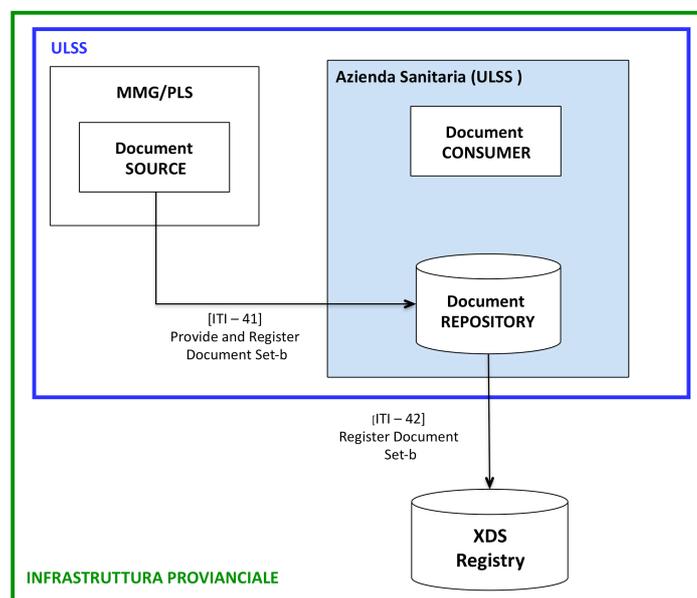
2.3.5.1 Creazione, pubblicazione e registrazione del documento di prescrizione

Il medico provvede a compilare nella sua cartella il form del documento di prescrizione.

Dopo la creazione del form, firmato digitalmente da parte del medico, il Document Source, con la transazione [ITI – 41] (Provide and Register Document Set - b) pubblica il documento appena creato nel Document Repository dell’Azienda Sanitaria di afferenza.

A questo punto l’attore Document Repository effettua la registrazione del documento nel Document Registry mediante la transazione [ITI – 42] Register Document Set – b.

Figura 2.10: Pubblicazione e registrazione della e-Prescription



2.3.5.2 Presa in carico della e-Prescription da parte dell’Azienda Sanitaria

Quando il paziente si presenta allo sportello CUP dell’Azienda Sanitaria, per farsi erogare il servizio oggetto della prescrizione del medico, il Document Consumer recupera il documento di prescrizione attraverso le transazioni:

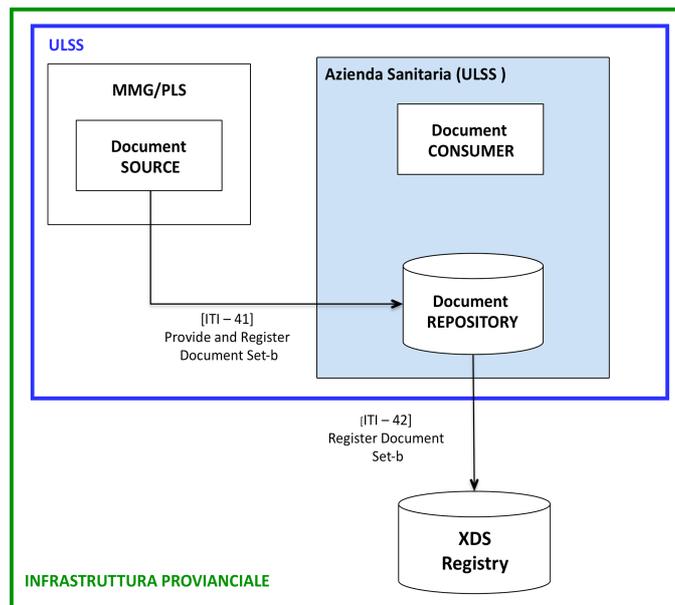
- [ITI – 18] (Registry Stored Query) relativa al Registry
- [ITI – 43] (Retrieve Document Set) relativa al Repository.

Il Document Consumer dell’Azienda Sanitaria a seguito della query, riceve dal Document Registry i metadati che contengono la lista dei riferimenti al documento che soddisfano la query, specificando l’allocazione e l’identificativo del documento trovato nel Document Repository. Ottenuti l’XDSDocumentEntry uniqueId e il Document Repository repositoryUniqueId, il Consumer

dell'Azienda Sanitaria recupera il form della prescrizione dal Document Repository attraverso la transazione [ITI_43] (Retrieve Document Set).

Una volta visualizzata la e-Prescription e presa ufficialmente in carico dall'Azienda Sanitaria in oggetto, il ciclo della prescrizione in questa prima fase risulta concluso.

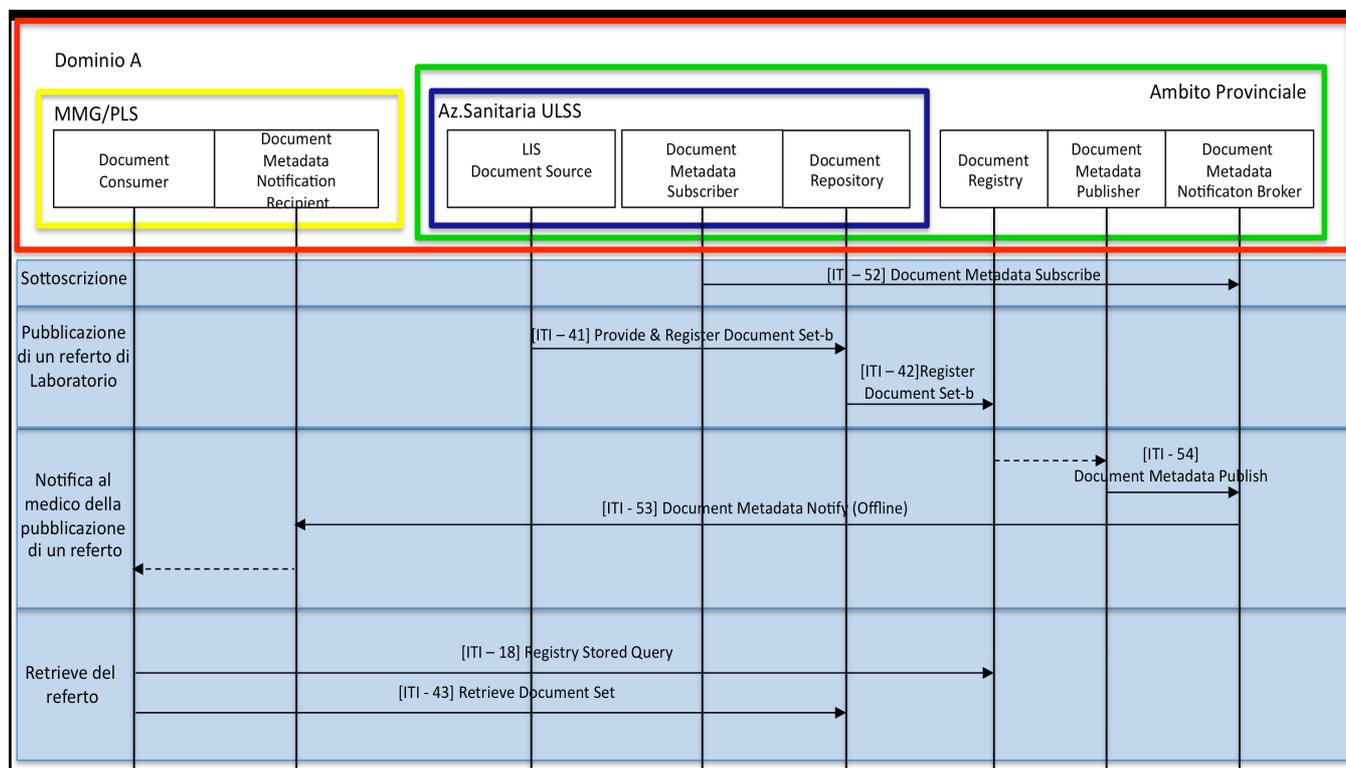
Figura 2.11: Presa in carico della e-Prescrizione da parte dell'Az.Sanitaria



2.3.5.3 Schema sequenziale delle transazioni

Di seguito viene riportato uno schema sequenziale di tutte le transazioni fin qui esposte:

Figura 2.12: Schema sequenziale delle transazioni per la pubblicazione e presa in carico di una e-Prescription



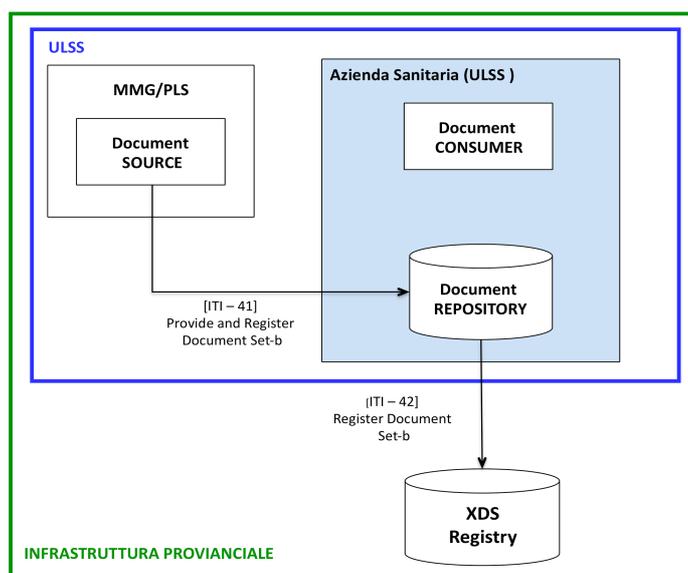
2.3.5.4 Schema generale riassuntivo

Lo schema generale riassuntivo per l'implementazione del software applicativo per la condivisione la condivisione dei documenti di prescrizione è illustrato nella seguente Figura 1.13.

Transazioni tra gli attori coinvolti, in ordine sequenziale:

1. [ITI - 41] Provide and Register Document Set-b
2. [ITI - 42] Register Document Set-b
3. [ITI - 18] Registry Stored Query
4. [ITI - 43] Retrieve Document Set-b

Figura 2.13: schema generale riassuntivo per l'implementazione del software applicativo per la condivisione la condivisione dei documenti di prescrizione



2.3.6 Elenco dei ricettari in carico al medico

Questo servizio esposto dall'azienda sanitaria fornisce l'elenco dei ricettari in carico al medico.

Attraverso questo servizio verrà fornito all'applicativo del medico un intervallo di valori corrispondenti agli attuali codici identificativi di ogni singola ricetta. Ogni singolo valore dovrà essere inserito nel documento strutturato CDA2 di prescrizione nel campo ID F.B corrispondente nel dettaglio al Codice a barre e numerico identificativo della ricetta.

Il servizio sarà esposto su web service (modalità di esposizione del servizio da parte dell'Azienda Sanitaria o della Regione ancora da definire). Campi di input obbligatori:

– pinCodeIn: Pin code del medico richiedente

Campi di output:

– array di ElencoRicettariRecord, ciascun elemento contiene i campi:

- annRicettario: anno del ricettario
- prgRicettario: progressivo del ricettario
- codReg: codice regione del ricettario
- codAsl: codice asl del medico
- codStrutt: codice struttura del medico

-codSpec: codice specializzazione del medico

2.3.7 Open Issues nel Progetto DOGE di interesse per la tesi

I problemi ancora aperti per quanto riguarda la gestione della prescrizione elettronica attraverso la piattaforma DOGE sono i seguenti:

- Profilo di integrazione Audit Trail and Node Authentication (ATNA)
- Sistema di notifica al medico di base, nel caso la prescrizione venga erogata da un altro medico: profili di integrazione Document Metadata Subscription (DSUB) e Notification of Document Availability (NAV)
- Prescrizioni farmaceutiche: connessione alla rete DOGE delle farmacie territoriali
- Documento di Workflow per tracciare il cambio di stato di una prescrizione

In particolare risulta fondamentale per lo scopo di questo lavoro di tesi il problema connesso al cambiamento di stato del documento di prescrizione. Infatti, essendo il documento di prescrizione un documento clinico, è per definizione non modificabile, in quanto l'intera documentazione associata ad un caso clinico deve persistere inalterata una volta che viene firmata e quindi legalmente autenticata. Questo motivo rende impossibile inserire all'interno del documento di prescrizione, redatto attraverso l'ausilio dello standard CDA2, un campo aggiuntivo per il quale sia possibile modificarne il contenuto man mano che il workflow della prescrizione si compia. Tuttavia è fondamentale poter tracciare lo stato della prescrizione, in modo da poter conoscere in tempo reale lo stato del documento e le operazioni possibili correlate allo stesso, evitando la possibilità che la stessa possa essere utilizzata più volte dallo stesso paziente all'interno di strutture differenti.

L'idea per risolvere questo problema nasce dall'osservazione dei numerosi linguaggi in grado di descrivere workflow. Si può infatti pensare all'iter della prescrizione (che può essere incognito a priori) come ad un particolare business che deve essere realizzato mediante l'intervento di svariati attori e agenti che progressivamente agiscono sul documento per poter raggiungere un compito predefinito. Ad ogni attore coinvolto nel workflow sono associate delle potenziali azioni che può svolgere per modificarne lo stato, e a seconda dello stato precedente, delle condizioni attuali e delle disponibilità dei servizi che dovrebbero susseguirsi nel workflow, viene compiuta una particolare azione, che determinerà, lo sviluppo di uno specifico percorso alternativo per il documento stesso. Si rende quindi necessario associare ad un istanza di workflow un documento, nel quale possano essere tracciati tutti gli step eseguiti fino ad un determinato istante, in modo da poter verificare,

attraverso un accesso a questo documento, lo stato attuale del workflow stesso. Associare per esempio un documento di questo tipo al ciclo prescrittivo significa associare al documento di prescrizione un secondo documento che permetta di identificare in che tipo di stato si trova la prescrizione stessa, per esempio se è utilizzabile, se la prescrizione è già stata utilizzata o se il servizio è stato richiesto e la prestazione è in attesa della prenotazione o della dispensazione.

Le elevatissime potenzialità di un approccio di questo tipo offrono la possibilità di applicarlo a svariati casi nel settore della sanità. Per questo motivo è stato pensato di standardizzare questo sistema, definendo un nuovo profilo di integrazione da sottoporre alla comunità internazionale, ed in particolare all'organismo internazionale IHE.

3. Metodologia di lavoro:

Lo Standard IHE

3.1 Introduzione

Il progetto Doge è stato definito basandosi sulle linee guida proposte dallo standard IHE per la condivisione dei documenti. Infatti il progetto si basa sulla possibilità di condividere informazioni cliniche all'interno della stessa Community, con l'obiettivo di estendere questi servizi anche a scenari che coinvolgono Community diverse.

In questo capitolo, viene presentato lo studio svolto sui principali standard utilizzati in Telemedicina e sui due profili, XDS e XCA, alla base della realizzazione pratica dei sistemi di Telemedicina. Ad essi sono poi associati altri profili che sono di utilità nella realizzazione del progetto DOGE:

- NAV: Notification of Document Availability

- CT: Consistent Time

- DSUB: Document Metadata Subscription

Viene inizialmente presentato IHE (Integrating Healthcare Enterprise), un'iniziativa che si propone di sviluppare l'integrazione tra i diversi sistemi informativi sanitari ; IHE può essere visto come lo sforzo di costruire un linguaggio univoco, eliminando le possibili ambiguità tra gli standard esistenti. Successivamente, vengono illustrati i due principali profili utilizzati in questo ambito:

- XDS (Cross-Enterprise Document Sharing) profilo di integrazione che si pone come obiettivo la creazione di una infrastruttura che consenta la condivisione di documenti clinici all'interno di un "affinity domain" (cioè all'interno gruppo di strutture sanitarie che stabiliscono di lavorare assieme sulla base di politiche e infrastrutture comuni);

- XCA (Cross Community Access) profilo di integrazione per la condivisione delle informazioni ad un livello superiore di quello ospedaliero e che consente l'interoperabilità sia tra Community strutturate al loro interno secondo il profilo XDS sia tra Community non aderenti ad XDS.

3.2 Problematiche di standard connesse all'informatica sanitaria

L'integrazione delle informazioni tra sistemi informativi ospedalieri di tipo gestionale, apparati elettromedicali e di diagnostica è una delle strade fondamentali da percorrere per fornire una migliore continuità di cura ai pazienti. Nel mondo della sanità l'integrazione tra sistemi è un annoso problema, per le dimensioni e per le specificità di questo settore, molto complesso, articolato e privo di sviluppo organico.

Attualmente negli ospedali esistono numerosi sistemi informativi distinti, che gestiscono, molto spesso autonomamente, i dati anagrafici e clinici del paziente e le immagini diagnostiche. Tra questi sistemi, i principali sono rappresentati dal software di gestione del CUP (centro unico prenotazioni), dalla gestione degenze (ADT), dai sistemi di cassa, cui si aggiungono sistemi dipartimentali, quali PACS e RIS (ovvero Sistema di Archiviazione e Comunicazione delle Immagini Radiologiche e Sistema Informativo Radiologico), sistemi di supporto della piattaforma amministrativa per la gestione di contabilità, approvvigionamenti, magazzino, farmacia, manutenzione, e le altre attività gestionali di un ospedale. Questi sistemi necessitano di condividere informazioni, ma, pur utilizzando protocolli standard di comunicazione, spesso non sono in grado di interoperare, in quanto gli standard possono presentare conflitti interpretativi e libera scelta di opzioni.

3.3 IHE (Integrating the Healthcare Enterprise)

IHE è un'iniziativa che si propone di sviluppare l'integrazione tra i diversi sistemi informativi sanitari. L'iniziativa IHE, nata negli Stati Uniti nel 1998 a opera di RSNA e HIMSS (Radiological Society of North America e Healthcare Information and Management Systems Society), si propone di definire in maniera chiara come gli standard esistenti (in particolare DICOM e HL7) dovrebbero essere utilizzati dai diversi sistemi informativi per realizzare tra loro un'efficace integrazione.

Lo scopo di IHE è quello di assicurare che, nella cura dei pazienti, tutte le informazioni richieste per le decisioni cliniche siano corrette e facilmente reperibili da parte delle varie figure professionali coinvolte. IHE, comunque, non vuole definire un vero e proprio standard di comunicazione, bensì quello di costruire un linguaggio univoco, eliminando le possibili ambiguità tra gli standard esistenti. Nello stesso tempo IHE vuole essere un ambiente di riferimento sui problemi d'integrazione tra i diversi sistemi informativi e le apparecchiature elettromedicali. Perciò è necessario un aggiornamento continuo e la collaborazione aperta tra produttori e utilizzatori.

Ogni anno i risultati elaborati nell'ambito di un comitato IHE vengono pubblicati nel "IHE

Technical Framework" che viene poi regolarmente aggiornato. In questo documento sono definite specifiche implementazioni degli standard esistenti in determinate aree, allo scopo di ottenere adeguate condivisioni delle informazioni mediche.

Un aspetto molto importante dell'iniziativa IHE è rappresentato dai test sperimentali che periodicamente vengono effettuati nell'ambito di specifiche sessioni, dette Connect-a-thon, cui partecipano le realtà che aderiscono a IHE. Durante un Connect-a-thon si costruisce una rete tra diversi sistemi medicali e si verifica che questi si integrino correttamente nello scambio di dati comuni, in quelle che vengono definite 'transazioni', secondo uno schema di azioni prestabilite, effettuate dagli 'attori IHE' e definite preliminarmente dai gruppi IHE. In pratica, si testa l'integrazione per una serie di situazioni tipo, detti "profili di integrazione", anche queste stabilite preliminarmente dai gruppi di lavoro IHE.

3.3.1 I Technical Framework

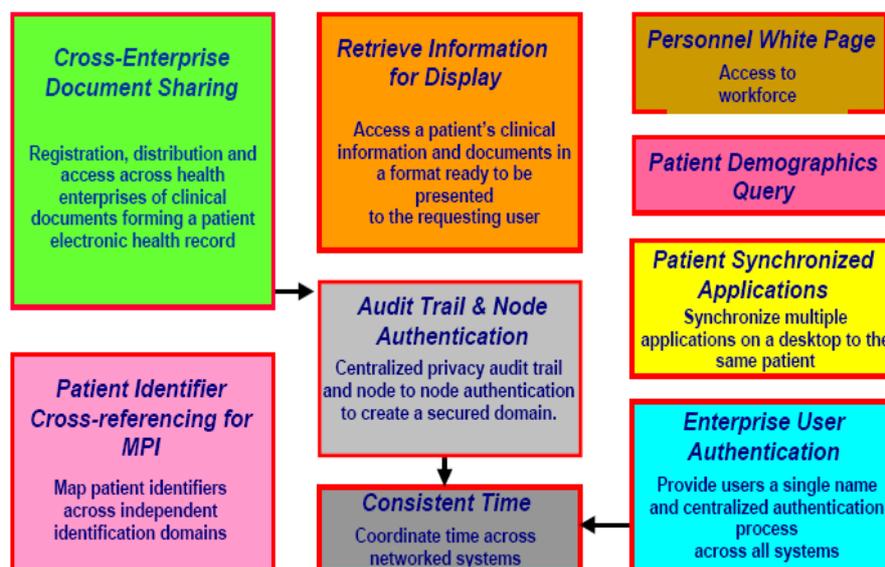
I Technical Framework (TF) identificano i domini che compongono le realtà sanitarie, denominate attori IHE, e specificano qual è il loro scopo e come devono interagire tra di loro. Descrivono queste transazioni entrando sempre più in dettaglio.

I Technical Framework sono stati divisi in undici domini di interesse:

- Cardiology
- Eye Care
- IT Infrastructure
- Laboratory
- Patient Care Coordination
- Patient Care Devices
- Anatomic Pathology
- Pharmacy
- Quality, Research and Public Health
- Radiation Oncology
- Radiology

All'interno di ogni dominio sono presenti due o più volumi: il volume 1 dà una visione generale delle funzionalità di IHE, mostrando le transazioni organizzate in unità funzionali, chiamate profili di integrazione, e sottolineando la loro capacità di soddisfare le specifiche richieste delle Infrastrutture; il volume 2 e i successivi forniscono descrizioni tecniche dettagliate di ogni transazione di IHE da utilizzare nei diversi Profili di Integrazione. In particolare il profilo XDS (che verrà discusso più dettagliatamente nel paragrafo 3.6) è presente nel Dominio IT Infrastructure. La versione corrente, è la 7.0 e specifica le transazioni IHE definite ed implementate a agosto 2010. Qui di seguito sono schematizzati i profili, compreso XDS, presenti nel Dominio IT Infrastructure.

Figura 3.1: Profili di Integrazione IHE IT



3.3.2 L'Interoperabilità

Lo scopo di IHE è far sì che i sistemi informatici sanitari siano integrati tra di essi, e per far questo propone profili di integrazione in modo da garantire l'interoperabilità tra i sistemi. Due applicazioni si dicono interoperabili, rispetto ad una specifica funzione, quando un'applicazione può accettare i messaggi inviati dall'altra e fornire la funzione richiesta in modo appropriato e soddisfacente, senza l'intervento di un operatore esterno. Perché due applicativi siano interoperabili rispetto ad una particolare funzione deve:

- esistere un sistema di trasmissione per lo scambio dei messaggi;
- essere condiviso il protocollo di comunicazione;
- essere condiviso il significato attribuito a ciascuna entità di informazione contenuta nel messaggio trasmesso;

- essere inviata una quantità di dati sufficienti per fornire la funzione richiesta;
- essere definite le modalità con cui la funzione deve essere implementata.

L'interoperabilità così definita è una proprietà unidirezionale, limitata ad una specifica funzione. In assenza di interoperabilità, la funzione richiesta può non essere disponibile od essere fornita solo in parte o con modalità diverse da quelle specificate. La piena interoperabilità può essere garantita solo se le due applicazioni rispondono conformemente per tutte le possibili funzioni richieste. Le risposte del sistema risultano conformi quando i risultati ottenuti e i metodi utilizzati per ottenerli corrispondono alle specifiche imposte.

Il problema di garantire l'interoperabilità nasce quando sistemi diversi devono essere integrati in un unico sistema complesso, superando le incompatibilità derivanti dalla presenza di applicazioni sviluppate indipendentemente o di tipo proprietario. Sono possibili diverse situazioni.

- La situazione più favorevole si presenta quando viene condiviso il significato attribuito alle entità di informazione e viene utilizzato lo stesso protocollo di comunicazione. Tale circostanza è generalmente associata ad uno standard comune implementato da entrambe le parti.
- Quando viene condiviso solo il significato attribuito alle entità di informazione, devono essere introdotte soluzioni adeguate per rendere possibile la comunicazione.
- La situazione più complessa si presenta quando vengono utilizzate entità di informazione diverse e protocolli di comunicazione incompatibili. Deve essere introdotto un modello di informazione comune, con adeguati meccanismi di conversione.

3.3.3 I Messaggi

In generale, quando si parla di messaggio si vuole indicare il contenuto di una comunicazione scritta od orale tra le persone. Nel caso di comunicazione tra due dispositivi elettronici, il contenuto del messaggio si compone di singole entità di informazione, ciascuna con una struttura chiaramente definita; per esempio il messaggio può contenere un'immagine digitale per la quale occorre specificare se i dati della matrice di pixel sono ordinati per righe o per colonne.

Un modello di informazione permette di rappresentare in modo molto efficace l'insieme di entità di informazione utilizzabili in un certo contesto, il significato di ciascuna e il tipo di relazioni esistenti. L'interoperabilità tra due sistemi esiste solo se il significato attribuito al contenuto di un messaggio non cambia tra chi invia e chi riceve. Diventa, quindi, necessario condividere il significato delle entità di informazione, mentre non risulta necessario, e in molti casi neppure realistico, condividere i modelli di semantica. Inoltre, solo la struttura utilizzata per comunicare una particolare entità di

informazione deve essere comune, mentre può essere diversa quella utilizzata per la sua memorizzazione (ad esempio non è richiesto l'impiego di uno stesso database).

Il messaggio si compone di una sequenza di bit, che, normalmente, non è progettata per essere compresa e manipolata da un utente, ma solo da un dispositivo elettronico, come i messaggi inviati ad una stampante, che possono contenere sequenze di controllo senza alcun significato testuale diretto. In alcuni casi, il messaggio può risultare facilmente accessibile anche nel suo formato elettronico, come la sequenza di bit corrispondente alla sequenza di caratteri alfanumerici di un testo.

Le diverse possibilità di accedere ad un messaggio dipendono dal suo formato ed influenzano il tipo di funzionalità che possono essere implementate.

3.4 Attori e transazioni

Nei profili di integrazione vengono definite le caratteristiche degli attori e delle transazioni. Gli attori sono i sistemi e i dispositivi che manipolano l'informazione richiesta dalle diverse procedure operative. Le transazioni sono un insieme coordinato di iterazioni tra attori per il trasferimento dell'informazione attraverso messaggi standard.

I primi attori e transazioni proposti da IHE sono indirizzati a rappresentare alcuni processi reali, relativi principalmente alla gestione delle seguenti attività: registrazione di pazienti, richiesta di esami, trasferimento delle richieste di esami; alle modalità di acquisizione, memorizzazione, archiviazione e revisione delle immagini; alla ricerca e recupero delle immagini e alla produzione di referti.

Le componenti funzionali sono gli attori, che possono svolgere in parte o completamente alcune delle funzioni tradizionalmente associate a sistemi commerciali di tipo PACS, RIS e HIS. Esempi di attori sono: i dispositivi per immagini, i sistemi di registrazione dei pazienti, i sistemi di prenotazione degli esami, i sistemi di gestione degli archivi dipartimentali, i sistemi di archiviazione delle immagini a lungo termine. Per ciascun attore vengono specificate le transazioni obbligatorie od opzionali per la dimostrazione. Le transazioni vengono specificate indicando le condizioni che devono essere rispettate nell'implementare gli standard disponibili (DICOM e HL7). Esempi di transazioni sono: la registrazione di un paziente, la richiesta di un esame, la richiesta di una lista di lavoro, la notifica del completamento di un esame, la richiesta di un'immagine. La definizione di una generica transazione include:

- la descrizione degli obiettivi della transazione;
- la definizione degli attori e dei loro ruoli;

- gli standard di riferimento per la transazione;
- il diagramma che illustra le relazioni tra attori e transazioni;
- la definizione di ciascun messaggio presente nella transazione, la sua struttura, gli eventi che lo generano e quelli che ne derivano.

L'integrazione viene progressivamente estesa a tutto il sistema organizzativo aziendale. Il progetto risulta particolarmente ambizioso.

Le direzioni di sviluppo delle prossime dimostrazioni sono duplici: in senso verticale, all'interno di domini specifici (ad esempio singoli dipartimenti di radiologia, cardiologia, patologia, ecc.), dove gli standard utilizzati sono quelli prevalenti per il particolare dominio; in senso orizzontale, per lo scambio di dati comuni a più domini (ad esempio anagrafica pazienti, elenco di medici, ecc.). Nell'integrazione orizzontale, il flusso di informazioni deve essere orientato alla terminologia specifica di ciascun dominio. Le informazioni relative a ciascuna specialità devono essere comunicate al richiedente nel formato adatto all'uso previsto. Le immagini prodotte in un dipartimento di radiologia devono essere fornite al medico richiedente nel formato utile per la sua necessità di consultazione.

3.5 Standard per l'informatica sanitaria: breve stato dell'arte

I sistemi informatizzati comunicano tra loro utilizzando prevalentemente protocolli standardizzati. Per la gestione di dati clinici e di immagini mediche, gli standard più significativi e specifici sono: HL7 CDA (Clinical Document Architecture), DICOM (Digital Imaging and Communications in Medicine) e HL7 (Health Level Seven). Ciascuno di questi standard può contribuire a migliorare il flusso di informazioni di una parte del sistema e ad aumentare l'efficienza e l'economicità del sistema complessivo. Tutti sono stati sviluppati per risolvere problemi ben definiti e sono oggetto di continue modifiche ed aggiornamenti, con l'obiettivo di rimuovere le limitazioni derivanti da un'impostazione fortemente orientata ad uno specifico settore. È auspicabile che l'informazione prodotta nei diversi formati possa essere scambiata in modo sempre più semplice, accurato e consistente. DICOM, HL7 sono stati influenzati, in diversa misura, dal modello OSI: l'Open Systems Interconnection (meglio conosciuto come Modello ISO/OSI) è uno standard stabilito nel 1978 dall'International Organization for Standardization, il principale ente di standardizzazione internazionale, (ISO), che stabilisce una pila di protocolli in 7 livelli. L'organizzazione sentì la necessità di produrre una serie di standard per le reti di calcolatori ed avviò il progetto OSI (**O**pen **S**ystems **I**nterconnection), un modello standard di riferimento per l'interconnessione di sistemi aperti. Il modello ISO/OSI è costituito da una pila (o stack) di protocolli attraverso i quali viene

ridotta la complessità implementativa di un sistema di comunicazione per il networking. In particolare ISO/OSI è costituito da strati (o livelli), i cosiddetti *layer*, che racchiudono uno o più aspetti fra loro correlati della comunicazione fra due nodi di una rete. I layers sono in totale 7 e vanno dal livello fisico (quello del mezzo fisico, ossia del cavo o delle onde radio) fino al livello delle applicazioni, attraverso cui si realizza la comunicazione di *alto livello*.

Nel momento in cui occorre acquistare dispositivi medici o sistemi informatizzati, oppure sviluppare del software o progettare un nuovo sistema, la scelta di soluzioni standardizzate, soprattutto se applicate in modo sistematico, può offrire numerosi vantaggi. Gli standard possono eliminare quasi tutti i problemi derivanti dalla presenza di sistemi proprietari. Tuttavia, occorre sottolineare che non sempre sono possibili soluzioni di tipo plug and play e che, in molti casi, anche le soluzioni standard possono richiedere interventi specifici non trascurabili per la configurazione e la personalizzazione. Sempre più frequentemente vengono introdotti dispositivi medici complessi, che incorporano sistemi avanzati di gestione dei dati, con costi crescenti per lo sviluppo e la manutenzione delle applicazioni necessarie per l'interconnessione. Soluzioni standard riducono la complessità e i costi delle interfacce e semplificano la comunicazione tra dispositivi forniti da produttori diversi. La maggiore integrazione, prodotta da soluzioni uniformi, facilita la distribuzione dell'informazione e rende più economica la gestione complessiva del sistema, a vantaggio dell'attività clinica e della salute del paziente. Con soluzioni standard, il trasferimento automatico dell'informazione diventa più semplice e riduce la necessità di interventi diretti da parte degli operatori, aumentando la loro disponibilità e diminuendo la probabilità di errori dovuti all'intervento umano. Gli standard permettono di spostare l'attenzione dai problemi tecnici alla funzionalità offerta, orientando tutto il sistema ad utilizzare l'informazione in modo strategico, per una migliore erogazione dei servizi. L'adozione sempre più diffusa di soluzioni standardizzate aumenta la competizione tra i produttori, non più protetti da soluzioni proprietarie, e favorisce lo sviluppo di dispositivi ed applicazioni conformi agli standard più diffusi, riducendone i costi di produzione ed implementazione.

3.5.1 Lo standard DICOM

Lo standard Digital Imaging and Communications in Medicine (DICOM) è stato creato dalla National Electrical Manufacturers Association (NEMA) per aiutare la distribuzione e la visione di immagini mediche, quali le CT e le MR. In sintesi, DICOM è uno standard orientato alla comunicazione di immagini mediche in formato digitale. Diffuso soprattutto nei dipartimenti di

radiologia, è caratterizzato da una struttura aperta e da ampie possibilità di estensione ed applicazione per le specialità che producono immagini mediche: patologia, endoscopia, odontoiatria, oculistica e dermatologia.

Negli anni '80, con la crescente diffusione dei sistemi di acquisizione delle immagini mediche in formato digitale (modalità) diventa indispensabile trovare soluzioni per semplificare la connessione e l'interoperabilità, a vantaggio di utenti e produttori. DICOM è la risposta a questa necessità.

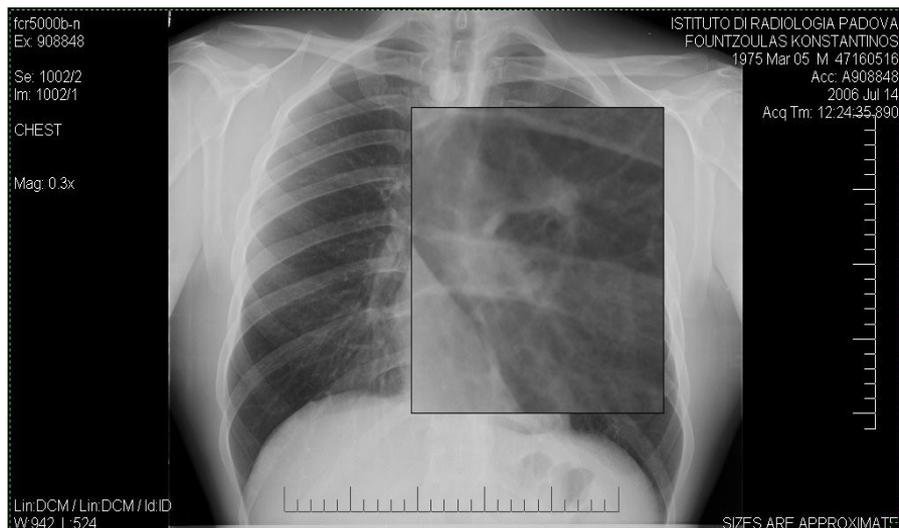
Lo standard è stato sviluppato congiuntamente da utenti e produttori di dispositivi con l'obiettivo di rendere possibile la connessione tra sistemi di produttori diversi. Possiede le componenti fondamentali per dialogare con i sistemi informatizzati di gestione delle immagini (PACS, Picture Archiving and Communication System), dell'attività ospedaliera (HIS, Hospital Information System) e, in particolare, del dipartimento di radiologia (RIS, Radiology Information System). Lo standard è in continua evoluzione, per migliorare le funzionalità indirizzate all'integrazione e per ampliare le possibilità di interfacciare sistemi diversi da quelli tradizionalmente presenti in un reparto di radiologia.

Ad esempio, a partire dal 1997 sono state definite alcune estensioni a supporto delle applicazioni di radioterapie, per consentire la rappresentazione di entità relative all'anatomia del paziente, ai piani di trattamento, alle informazioni dosimetriche, ai dati sul trattamento effettuato e alle immagini usate in radioterapia.

Un singolo file DICOM è composto da un header, che immagazzina informazioni del paziente, il tipo di scansione, la dimensione dell'immagine ed altro, e dal corpo, con i pixel dell'immagine, che può contenere informazioni in tre dimensioni.

Un'immagine DICOM può essere compressa in JPEG tramite algoritmi lossy (con perdita di informazione) o lossless (senza perdita), per ridurre la dimensione del file.

Figura 3.2: Esempio di immagine DICOM



Si riporta come esempio il “dump” (ovvero una versione “leggibile” di un messaggio DICOM codificato in realtà in binario) in formato testo di un file DICOM contenente un’immagine radiologica:

0008,0008	Image Type	CS	RIGINAL/PRIMARY
0008,0020	Study Date	DA	1999.05.24
0008,0022	Acqu. Date	DA	1999.05.24
0008,0023	Image Date	DA	1999.05.24
0008,0030	Study Time	TM	11:42:21
0008,0032	Acqu.n Time	TM	11:42:43.309512
0008,0033	Image Time	TM	11:42:48
0008,0060	Modality	CS	CT
0008,0070	Manufacturer	LO	GE SYSTEMS
0008,0080	Inst. Name	LO	Univ. di Padova
0008,1010	Station Name	SH	NP
0008,1030	Study Descr.	LO	CT CHEST/PLAN
0008,1050	Physician's Name	PN	Anonymized
0008,1070	Operators' Name	PN	Anonymized
0008,1090	Model Name	LO	HiSpeed
0010,0010	Patient's Name	PN	Anonymized
0018,0015	Body Part	CS	CHEST
0018,0090	Data Diameter	DS	500.0

0018,1000	Device Serial Number	LO	0000287.0000001
0018,1020	Software Version(s)	LO	Series 2.00
0018,1050	Spatial Resolution	DS	0.3500000
0018,1130	Table Height	DS	-212
0018,1170	Generator Power	IS	9
0018,5100	Patient Position	CS	FFS
0020,0012	Acquisition Num	IS	1
0020,0013	Image Num	IS	1
0028,0004	Photometric Int	CS	MONOCHROME2
0028,0010	Rows	US	400
0028,0011	Columns	US	512
0028,0101	Bits Stored	US	6
7fe0,0010	Pixel Data	OB	...

3.5.2 Lo standard HL7

Lo standard HL7 (Health Level Seven) deriva il suo nome dal particolare livello dello standard OSI a cui è indirizzato: il livello di applicazione o livello 7. Nasce alla fine degli anni '80 con lo scopo di uniformare e semplificare lo scambio elettronico di informazioni cliniche ed amministrative tra i diversi sistemi presenti in un'azienda sanitaria, quali HIS (Hospital Information System), RIS (Radiology Information System), LIS (Laboratory Information System). Lo sviluppo di HL7 ormai coinvolge l'intera comunità internazionale, infatti HL7 sta coordinando gli sforzi con altri organismi internazionali di standardizzazione (es. CEN/TC 251, DICOM, ISO/TC 215, W3C) per incrementare i livelli di interoperabilità dei sistemi che costituiscono i sistemi informativi sanitari. Attualmente lo standard viene sviluppato con risorse offerte volontariamente da numerose organizzazioni (industrie, agenzie governative, organizzazioni che forniscono o rimborsano l'assistenza, etc). L'attività principale di HL7 si svolge tre volte l'anno quando, circa 40 gruppi di lavoro (Technical Committees e Special Interest Groups) si riuniscono per una settimana per valutare i progressi e gli sviluppi possibili; tuttavia idee e materiali tecnici vengono elaborati con un lavoro continuo tramite e-mail, conferenze telefoniche settimanali o incontri ad hoc. Possono essere reperite maggiori informazioni relative ad attività, alla struttura organizzativa ed ai materiali prodotti in www.hl7.org.

Lo standard HL7 viene sviluppato ed aggiornato da un comitato di utenti e produttori con

l'obiettivo comune di semplificare le interfacce tra applicazioni di produttori diversi, spesso antagonisti, ed uniformare il formato e il protocollo utilizzati nello scambio di alcuni insiemi critici di dati. Definisce fundamentalmente i messaggi oggetto di scambio, le modalità di sincronizzazione degli scambi e le specifiche dei messaggi di errore. La compatibilità con lo standard non implica necessariamente la compatibilità tra i sistemi che lo hanno adottato. Infatti, relativamente ai livelli sottostanti il livello 7, lo standard HL7 assume che i protocolli di comunicazione siano compatibili, senza imporre vincoli particolari. Inoltre, non interviene sulle modalità con cui le informazioni vengono presentate o gestite ed è pertanto facilmente integrabile in diverse tipologie di soluzione software. Relativamente alla collocazione e alla condivisione dei dati, non impone condizioni particolari all'architettura complessiva del sistema e può essere adottato anche in ambienti molto eterogenei, con architetture centralizzata o distribuita tra i singoli sistemi dipartimentali. Le specifiche sono fortemente orientate a garantire un'ampia flessibilità, che tenga conto delle esigenze particolari di ciascun ambiente e di ciascun utilizzatore, con l'obiettivo, a lungo termine, di consentire l'eliminazione o la forte riduzione di interfacce proprietarie e dei costi associati di programmazione e manutenzione.

Vediamo un esempio di messaggio HL7, un messaggio ADT (Admission-Discharge- Transfer) usato per comunicare i dati di un paziente (in ingresso o uscita da una struttura sanitaria) ad altri sistemi.

```
MSH|^~\&|MY_ADT|XYZ_ADMITTING|MY_IS|XYZ_HOSPITAL||ADT^A04|101102|P|2.3.1|||||EVN||
200004211000||||200004210950PID||583020^^^ADTI||GIORGIO^BLANCHI||19980704|M|AI|
VIAD EIGATTI^PADOVA ^PD^35125|||||ITALIA|||||
PVI||E|||||5101^MARIO^ROSSI^DR|||||V1002^^^ADTI|||||200 004210950|||||
```

Tale messaggio è formato dai seguenti segmenti: MSH (Message Header, presente in ogni messaggio), EVN (Event Description, descrive l'evento), PID (Patient Identification, identificativo del paziente) e PVI (Patient Visit Information, informazioni sulla prestazione sanitaria).

3.5.3 Lo standard CDA

Il HL7 CDA è uno standard che specifica la struttura semantica da utilizzare all'interno di documenti clinici scritti in XML, esso stabilisce quindi il nome degli elementi che conterranno i dati. Un CDA comprende le seguenti caratteristiche:

- Persistenza: un documento clinico esiste sempre in uno stato inalterato, per un intervallo di tempo impostato dai requisiti dell'amministrazione pertinente;

- Amministrazione: un documento clinico è gestito da una persona o da una struttura che ha il compito di garantire l'integrità dell'informazione contenuta al suo interno;
- Autenticazione: un documento clinico è un insieme di informazioni che sono state raggruppate con l'obiettivo di essere autenticate legalmente;
- Totalità: l'autenticazione di un documento clinico si applica su tutto il documento e non su frammenti di esso;
- Leggibilità: un documento clinico deve essere facilmente leggibile da chiunque.

Un documento CDA rappresenta un oggetto di informazione completo che può contenere testo, immagine, suoni ed altri contenuti multimediali. Un documento XML costruito secondo lo standard CDA è costituito da una serie di elementi, organizzati in una struttura ad albero definita dallo schema di riferimento.

3.6 IHE CROSS-ENTERPRISE DOCUMENT SHARING: XDS

3.6.1 Panoramica

Il profilo di integrazione **IHE XDS (Cross-Enterprise Document Sharing)** si pone come obiettivo la creazione di una infrastruttura che consenta la condivisione di documenti clinici all'interno di un affinity domain, a partire da un piccolo studio privato a una grossa azienda ospedaliera. L'XDS IHE Integration Profile assume che queste realtà sanitarie appartengano ad uno o più affinity domain. Un affinity domain è costituito da un gruppo di strutture sanitarie che stabiliscono di lavorare assieme sulla base di politiche e infrastrutture comuni.

All'interno di ogni affinity domain clinico, devono essere definite determinate regole e politiche per la gestione dei dati clinici e anagrafici dei pazienti e quindi stabilire come trattare la privacy e l'accesso ai dati. Inoltre deve essere stabilito come organizzare la struttura e la rappresentazione delle informazioni cliniche.

E' importante notare che XDS non definisce queste regole, il processo che porta alla definizione di un affinity domain non è parte del profilo, sebbene XDS sia stato pensato per accomodare una vasta serie di queste linee guida. Questo per facilitare lo sviluppo delle infrastrutture basate su questi standard per la condivisione di documenti clinici dei pazienti. La struttura generale del profilo prevede quindi la possibilità di condividere e recuperare informazioni da ambienti medici anche geograficamente distinti. Tale profilo è gestito da più attori distinti e con responsabilità separate.

Uno o più Document Repository, un Document Registry, più Document Source e Consumer e un Patient Identity Source. Gli attori interagiscono tra di loro attraverso transazioni con scopi diversi. Il concetto di documento nell'XDS non è limitato a solo informazioni testuali. XDS si definisce "neutro" rispetto al contenuto dei documenti, infatti, essi possono essere di qualsiasi tipo, indipendentemente dal formato e dalla loro rappresentazione. Questo fa sì che il profilo di integrazione XDS IHE sia ugualmente adatto a semplici documenti testuali, documenti testuali formattati (ad es. HL7 CDA Release 1), immagini (ad es. DICOM) o documenti strutturati e con un determinato "vocabolario" (ad es. CDA Release 2, CCR, DICOM SR). Eventuali restrizioni su tipi di dati gestibili sono da cercarsi nell'affinity domain, nel quale possono essere specificati i formati accettati. Queste limitazioni sono necessarie per assicurare la totale interoperabilità all'interno dell'affinity domain, fra i client che visualizzeranno i documenti e le strutture che quei documenti li hanno creati.

3.6.2 Gli Attori

Document Source: è il produttore dei documenti ed il responsabile della loro pubblicazione. Deve spedire i documenti e i metadata correlati al Document Repository in modo da consentirne la pubblicazione sul Document Registry.

Document Repository: è responsabile della memorizzazione dei documenti ricevuti dal Document Source e dell'invio dei metadata corrispondenti al Document Registry. Ad ogni documento viene assegnato un URI che consente il recupero del documento stesso da parte del Document Consumer.

Document Registry: conserva le informazioni contenute nei metadata ricevuti in corrispondenza ad ogni transazione "provide and register document set" eseguita da un Document Source verso un Document Repository. I metadata conservati includono il link al documento nel Repository dove è memorizzato. Questo attore ha inoltre il compito di rispondere alle interrogazioni ricevute dai Document Consumer restituendo le informazioni che corrispondono ai criteri di ricerca ricevuti.

Document Consumer: interroga il Document Registry per ottenere le informazioni corrispondenti a determinati parametri di ricerca ed esegue il retrieve dei documenti di interesse dal repository dove sono memorizzati.

Patient Identity Source: fornisce un identificativo univoco per ogni paziente.

3.6.3 Le Transazioni

Provide and Register Document Set-b: transazione che coinvolge Document Source e Repository.

Il Document Source inoltra al Repository sia il documento che i corrispondenti metadata. A questo punto è responsabilità del Repository memorizzare i documenti ricevuti e dare il via alla transazione Register Document Set che permetta la registrazione dei metadata nel Registry.

Figura 3.3: Diagramma Provide and Register Document Set



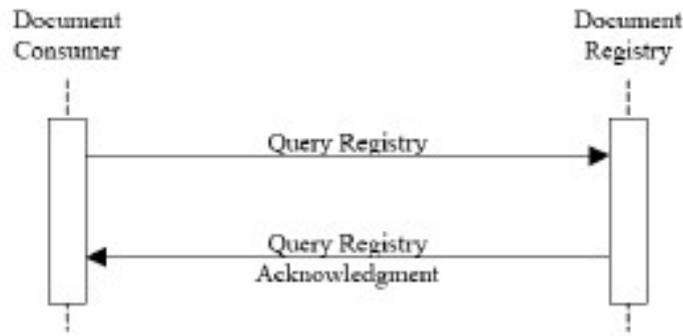
Register Document Set-b: transazione tra Document Repository e Registry. Permette la registrazione delle informazioni relative ai documenti giunti al Repository. In corrispondenza ad ogni documento viene generata una XDS Document Entry nel Registry. Questa transazione comprende una serie di validazioni sui metadata ricevuti dal Registry. In caso di fallimento di una di queste validazioni, anche relativa ad uno solo dei documenti contenuti in una stessa submission, verrà ritornata una risposta negativa.

Figura 3.4: Diagramma Register Document Set



Registry Stored Query: è la transazione che consente al Document Consumer di interrogare il Registry. In base ai criteri di ricerca ricevuti, il Registry fornisce le informazioni necessarie ad eseguire il retrieve dei documenti che corrispondono al risultato della query.

Figura 3.5: Diagramma Query Registry



Retrieve Document Set: transazione utilizzata dal Document Consumer per accedere ai documenti nei diversi Document Repository. Questi ultimi devono fornire al consumer i documenti richiesti.

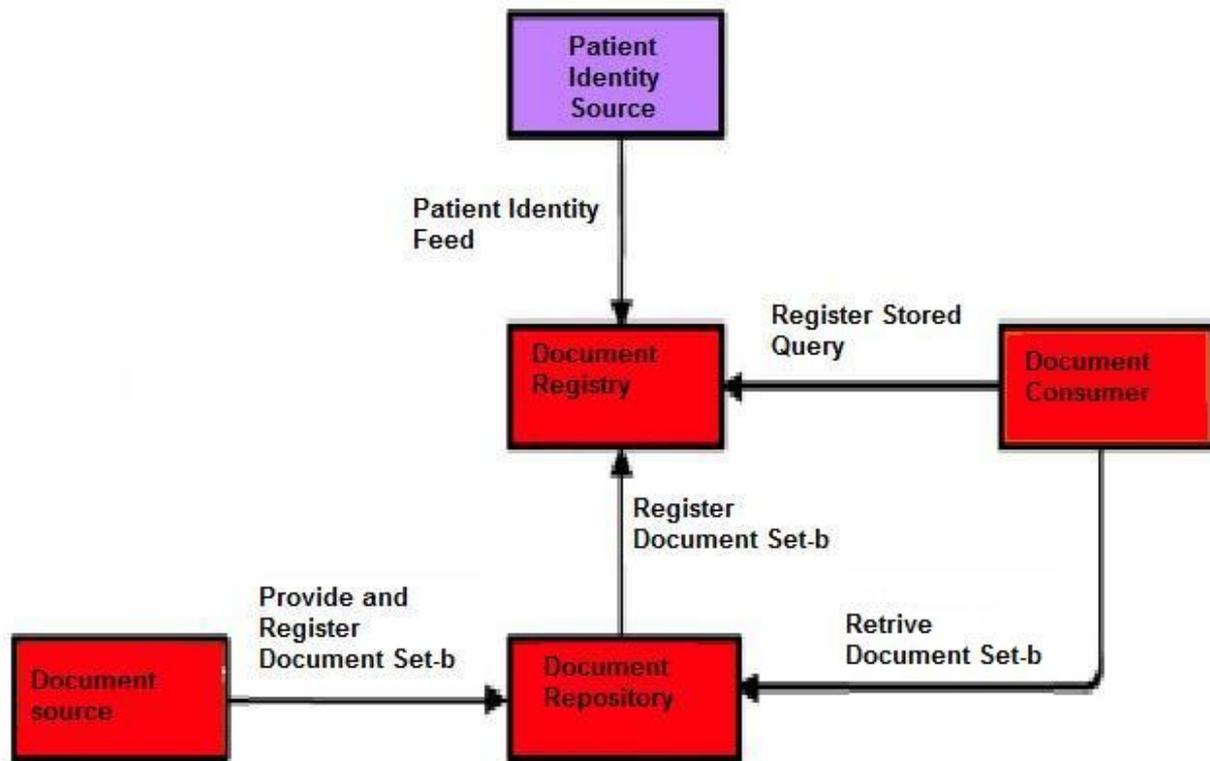
Figura 3.6: Diagramma Retrive Document



Patient Identity Feed: trasmette le informazioni riguardanti l'identificazione dei pazienti (identificativi e dati anagrafici). Consente la modifica e il merge dei dati in questione. Tale transazione, generata dal Patient Identity Source è rivolta al Registry al fine di consentire la popolazione di quest'ultimo con i dati di tutti i pazienti registrati nell'affinity domain.

Di seguito è riportato uno schema rappresentativo degli attori e le transazioni necessarie per il corretto funzionamento del profilo XDS.

Figura 3.7: Attori/Transazioni XDS



3.6.4 Tipi di documenti supportati

Nella tabella seguente vengono elencati i tipi di documenti supportati dai Profili di Integrazione IHE: XDS e XDS-I. XDS è utilizzato per condividere referti medici o comunque documenti testuali, XDS-I per immagini radiologiche.

Dominio di IHE Technical Framework	Nome del Profilo di Integrazione	Tipo di documento supportato
Patient Care Coordination	Cross-Enterprise Sharing of Medical Summaries	Referti medici in formato HL7 CDA
Radiologia	Cross-Enterprise Document Sharing for Imaging (XDS-I)	Referti radiologici diagnostici in formato testuale o PDF
		Riferimenti a una collezione di DICOM SOP Instances in un manifest nel formato DICOM Key Object Selection

3.6.5 Opzioni gestite dal profilo XDS

Oltre alle funzionalità base il Document Source permette di gestire transazioni più complesse rispetto alla condivisione di singoli documenti. Infatti si rende spesso necessario gestire scenari più complessi, come la creazione e manipolazione di Folder (con cui si identificano delle aggregazioni logiche, non necessariamente fisiche, di documenti riferiti allo stesso paziente o al medesimo caso clinico) o la modifica dei documenti presenti nel repository. In particolare offre la possibilità di:

-inviare più documenti in un'unica Submission Request;

-gestire degli stati del documento:

1. inviare un documento che vada ad aggiungersi ad un altro documento già presente nel registry/repository;
2. inviare un documento che vada a rimpiazzare un altro documento già presente nel registry/repository;
3. inviare un documento che vada a trasformare un altro documento già presente nel registry/repository;

-gestire il Folder:

1. creare un Folder;
2. aggiungere uno o più documenti ad un Folder.

Attore	Opzioni
Document Source	Transazione off-Line
	Invio multiplo di documenti
	Gestione degli stati del documento
	Gestione del folder
Document Repository	Transazione off-Line
Document Registry	Non ci sono opzioni definite
Document Consumer	Non ci sono opzioni definite
Patient Identity Source	Non ci sono opzioni definite

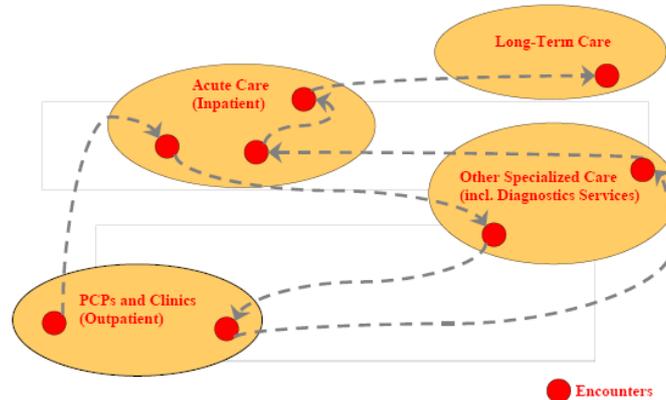
3.6.6 EHR-CR e EHR-LR

L'EHR-CR (Electronic Health Record - Care-delivery Record) è il contenitore documentale di tutto ciò che è stato prodotto in un episodio di cura (cartella clinica); è organizzata sulle informazioni del paziente/contatto (encounter) e viene gestita dal provider (organizzazione) che eroga il servizio (Azienda Ospedaliera che come mission esegue ricoveri e gestisce la cartella). L'XDS permette di

condividere i dati contenuti nelle cartelle private in un qualcosa al di sopra delle organizzazioni stesse (Aziende Ospedaliere).

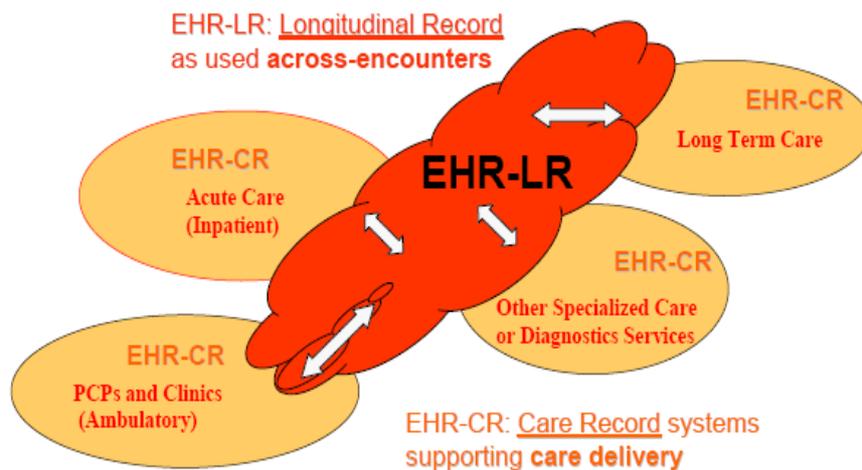
L'organizzazione ha in possesso un certo tipo di dati ma ad un certo punto decide di condividerli con un'altra organizzazione.

Figura 3.8: Storia clinica di un paziente in diverse strutture



EHR-LR (Electronic Health Record – Longitudinal Record) è l'insieme di documenti XDS costituenti la storia degli eventi sanitari di un paziente. L'EHLR è un contenitore di dati condivisi dalle EHR-CR (come la condivisione in rete di cartelle o di stampanti nel proprio computer).

Figura 3.9: Invio e condivisione di documenti relativi ad un paziente



3.6.7 Documento XDS

Un documento XDS è la più piccola unità di informazione che può essere inviata ad un attore Document Repository. Un documento XDS è un insieme di informazioni cliniche contenenti osservazioni e servizi per poter essere interscambiati. Un documento XDS deve essere leggibile (human readable) con un'appropriata applicazione e deve essere strutturato secondo gli standard definiti da IHE.

Solo il Document Source e il Document Consumer possono accedere alle informazioni del documento. L'attore Document Source si occupa della produzione del metadata e lo invia al Document Registry per creare una XDS Document Entry che verrà utilizzata per le query richieste dal Document Consumer. Il Source è l'unico attore ad avere la responsabilità del documento e deve quindi rimpiazzarlo nel caso fosse errato. Il documento inoltre deve essere identificato da un codice univoco.

3.6.8 Submission Request

L'XDS Submission Request è il messaggio fondamentale che sta alla base della condivisione dei documenti. Viene utilizzata dal:

- Document Source nella transazione *Provide and Register Document Set* al Document Registry;
- Document Repository nella transazione *Register Document Set Transaction* verso il Document Registry.

L'XDS Submission Request contiene informazioni necessarie per la corretta registrazione dei documenti. E' composto da:

- un set di Metadata;
- una Submission Set;
- zero o più *Folder*;
- documenti da aggiungere ad un *Folder* precedentemente creato;
- zero o più documenti.

3.6.9 Submission Set

Una XDS Submission Set è l'insieme di documenti XDS (nuovi o copie/link a preesistenti) registrati insieme e associati ad un evento sanitario di un singolo paziente. Deve essere creata in ogni

Submission Request. E' legata ad un unico Document Source ed è presente nella transazione *Provide & Register Document Set* o nella *Register Document Set*.

Lo stesso documento, inizialmente registrato come parte di prima *Submission Set*, può essere successivamente referenziato da una seconda *Submission Set*. Questo permette di associare documenti di un paziente presenti nell'unità sanitaria con *Submission Sets* più recenti.

3.6.10 Folder

Lo scopo di un XDS Folder è di offrire un meccanismo per i Document Sources di raggruppare i documenti per svariati motivi, ad es. si vuole raggruppare esami eseguiti in un certo periodo di tempo, o per un determinato caso clinico e offre la possibilità al Document Consumer di trovare tutte le Document Entries all'interno di una stesso Folder.

I seguenti concetti vengono applicati al Folder XDS:

- un Folder raggruppa un set di XDS Document legati ad un singolo paziente;
- uno o più Document Source possono inviare documenti in una data Folder;
- il contenuto di una Folder è definito da un elenco di codici/significati;
- i Document Consumer possono trovare Folders esistenti interrogando il Document Registry;
- il Document Registry conosce permanentemente il Folder creato;
- i Folder XDS non possono essere annidati;
- lo stesso documento può essere presente in più Folder;
- il Folder è identificato unicamente.

3.6.11 XDS Affinity Domain

Un XDS Affinity Domain è una struttura amministrativa composta da attori Document Sources, Document Repositories, Document Consumers e un singolo Document Registry che aggrega documenti clinici per poi condividerli.

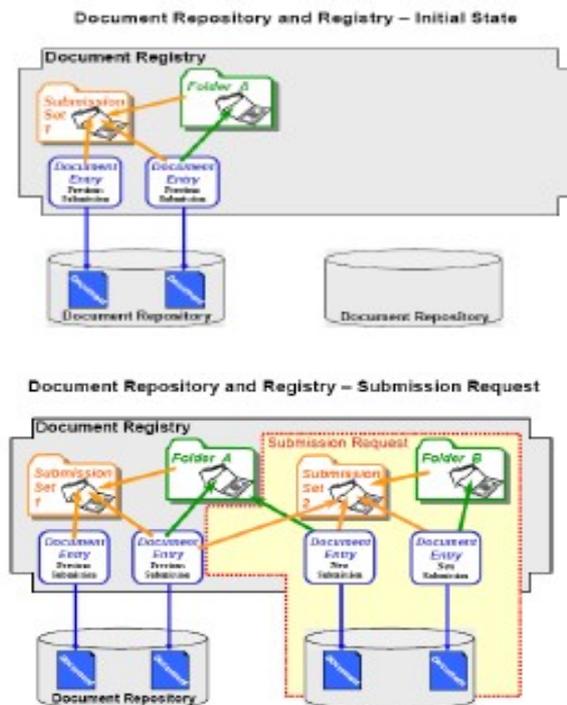
Per assicurare un'effettiva interoperabilità tra Document Sources e Consumers, è necessario stabilire delle linee guida all'interno dell'Affinity Domain. Alcune linee includono:

- quale formato per i documenti può essere registrato;
- quali sono i valori e gli schema che si possono usare per la submission del metadata del documento, per la submission set e per la registrazione dei folder.

3.6.12 Esempio di Submission Request, Submission Set e Folder

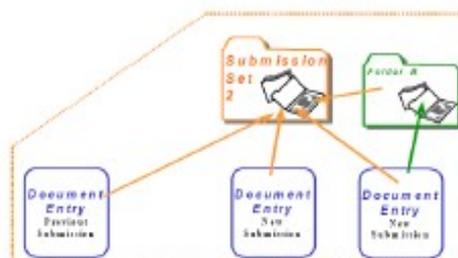
La sequenza di figure visualizzate qui di seguito mostrano un esempio di una *Submission Request* che include due nuovi documenti, un riferimento a un documento già esistente e l'uso di due *Folders*. La prima figura rappresenta lo stato iniziale di un *Document Registry* nel quale sono stati inviati due documenti, dove uno è associato al *Folder A*. La seconda figura mostra un *Submission Request* che aggiunge due nuovi documenti, inserendone uno in un *Folder* già esistente e l'altro in un *Folder B*.

Figura 3.10: Esempio di una Submission Request



Il contenuto della Submission Set del precedente esempio è visualizzato qui sotto. Le Document Entries associate alla Submission Set sono parti logiche della Submission Set.

Figura 3.11: Contenuto logico di una Submission Set



3.6.13 Modello di dati dell’XDS registry e i suoi attributi

Il profilo di interazione XDS fornisce al Document Source un mezzo per inviare documenti al repository, ma anche inviare informazioni (metadata) di questi documenti al Document Registry. Un metadata ben formulato permette al Document Consumer di reperire documenti clinici di interesse, così come un catalogo di una libreria aiuta il lettore a trovare il libro che vuole.

3.6.13.1 XDS Document Registry Data Model

Le seguenti entità sono usate nel XDS Document Registry Data Model:

- XDS Document Entry:** Informazione gestita dal Document Registry che descrive le maggiori caratteristiche del documento e un link al Document Repository per ritrovarlo.
- XDS Folder:** Un contenitore logico che raggruppa uno o più XDS Document Entries il cui scopo è quello di organizzare i Documenti.
- XDS Submission Set:** I documenti, quando vengono registrati dal Document Source, devono essere inclusi in una e solo una Submission Set. Una XDS Submission Set raggruppa zero o più documenti e ha riferimenti a documenti già registrati per assicurare la persistenza.
- XDS Submission Request:** Una Submission Request include una e una sola Submission Set, zero o più Folder nuovi e documenti che possono essere associati a Folder esistenti o nuovi.

3.6.13.2 Attributi delle XDS Document Entries

XDS definisce un set di attributi primari che soddisfano le esigenze della maggior parte degli attori coinvolti nel sistema, per recuperare i documenti più rilevanti (Vedi tabella qui di fianco). Oltre a questi, sono presenti attributi secondari come ad esempio il tipo di documento, l’autore e l’ente che l’ha prodotto, e il “ciclo di vita” (Life Cycle).

Figura 3.12: Attributi primari del XDS Document

Patient Id
Service Start and Stop Time
Document Creation Time
Document Class Code and Display Name
Practice Setting Code and Display Name
Healthcare Facility Type Code and Display Name
Availability Status (Available, Deprecated)
Document Unique Id

3.6.14 DOCUMENT LIFECYCLE

3.6.14.1 Stato di disponibilità dei documenti

Ad ogni documento XDS, contenuto nell’XDS Document Registry, deve essere assegnato uno stato di disponibilità:

- Approved: Disponibile,
- Deprecated: Obsoleto ma comunque reperibile.

Un documento XDS è settato come “Approved” dopo che il Repository e il Registry hanno processato la submission con successo. Un documento di tipo “Approved” può essere cambiato in “Deprecated” solo dal Source che l’ha creato. Un documento “Deprecated” rimane disponibile alle queries del Document Consumer.

3.6.14.2 Relazione tra documenti

I Documenti XDS possono essere legati a documenti precedenti i tre modi:

- Replacement
- Addendum

-Transformation

Queste relazioni tra i documenti vengono tracciate nel Document Registry. Un documento originale non può essere legato a nessun altro documento e di conseguenza non è presente il parent Id e il parent relationship. Il Registry deve rifiutare i documenti che contengono relazioni con documenti che non sono registrati o sono stati settati come “deprecated”.

1. Un documento rimpiazzato (*replacement*) è una nuova versione di un documento esistente. Il nuovo documento ha un nuovo document Id, un attributo che lo associa con l’Id del vecchio documento e un attributo con il codice “RPLC”. Lo stato del vecchio documento passa quindi a “deprecated”.

2. Un documento in aggiunta (*addendum*) è un documento separato che fa riferimento ad un documento esistente, e può estendere o alterare le osservazioni del documento precedente. Modifica il documento originale, ma quest’ultimo rimane comunque nello stato “approved”. Il nuovo documento ha il document Id del documento precedente e un attributo con il codice “APND”.

3. Un documento viene trasformato (*transformation*) nel caso in cui la modalità cambia il formato del documento. Ad esempio un documento in formato CDA può essere convertito in una DICOM SR. Il nuovo documento ha il document Id del documento precedente e un attributo con il codice “XFRM”.

3.7 NOTIFICATION OF DOCUMENT AVAILABILITY: NAV

Il Profilo di Integrazione NAV definisce un meccanismo di notifica punto a punto tra sistemi e utenti all’interno di un XDS Affinity Domain. In questo profilo per inviare le notifiche viene utilizzato il protocollo SMTP e altri standard.

Il Profilo NAV definisce il formato, il contenuto, la codifica e la trasmissione dei messaggi di notifica e l’acknowledgements tra gli attori NAV.

Queste notifiche servono per indicare che sono presenti nel Registry i metadata di uno o più documenti nuovi o esistenti. In questo profilo non è definito il modo in cui il Consumer utilizza questa notifica per permettere diversi approcci.

3.7.1 Scopo

Questo Profilo definisce il formato e il contenuto dell'e-mail di notifica tra gli attori dell'Affinity Domain. Il messaggio può essere trasmesso tramite un tradizionale programma di posta elettronica, mentre il software per ricevere il messaggio deve essere integrato nell'XDS Consumer. Il Consumer, una volta letto il contenuto dell'e-mail, può procedere con l'acquisizione dei documenti o del metadata associato.

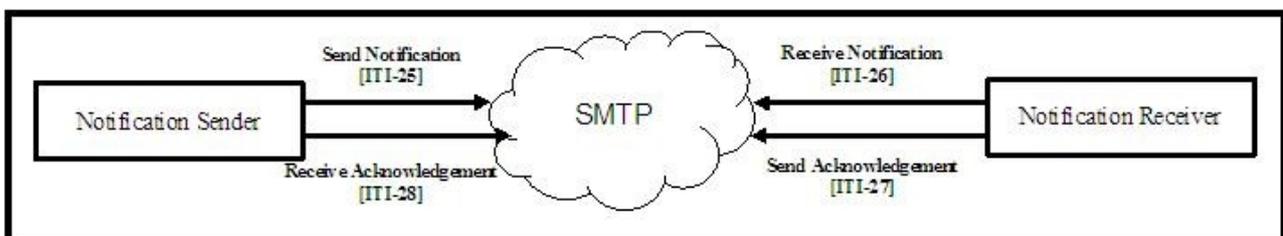
Il Profilo è stato progettato con l'intento di inviare e-mail con le minime informazioni del paziente (PHI7); il Consumer può ottenerle solo dopo aver interrogato il Registry. I messaggi contengono solo:

- l'UUIDs dei documenti che sono stati resi disponibili;
- l'indirizzo del Registry dove recuperare I documenti;
- un testo libero con delle istruzioni per il ricevente;
- l'indirizzo e-mail del mittente;
- L'indirizzo e-mail del ricevente.

3.7.2 Attori e Transazioni

La figura 3.14 mostra gli attori direttamente coinvolti nel Profilo di Integrazione NAV e le transazioni tra di essi. Non sono necessariamente presenti attori che possono essere coinvolti indirettamente alla transazione perché presenti in altri profili. L'attore che riceve il messaggio deve supportare almeno uno tra i protocolli SMTP, POP38 e IMAP9.

Fig. 3.14: Attori e transazioni profilo NAV



3.7.3 Invio della notifica

-Eventi Trigger: Un Registry XDS usa questa transazione per inviare una notifica quando:

1. vengono registrati nuovi documenti
2. è bisogno di notificare documenti esistenti.

-Semantica del messaggio: Il mittente deve costruire un messaggio mail di tipo multipart/mixed conforme al RFC 822 e RFC 1521 e la spedizione deve avvenire tramite protocollo SMTP.

La prima parte del messaggio indica lo scopo per cui è stato inviato il messaggio e come utilizzarlo; tutto ciò per permettere la corretta interpretazione da parte del ricevente. Inoltre sarà presente un link di tipo mailto: URL se il mittente implementa la ricevuta di ritorno (acknowledgement).

La seconda parte è un allegato di tipo application/xml con un nome del tipo 'IHEXDSNAV-UUID.xml', dove UUID è un identificatore univoco del messaggio.

3.8 CROSS COMMUNITY ACCESS: XCA

3.8.1 Introduzione

Il profilo di integrazione **XCA (Cross Community Access)** definisce attori e transazioni per la condivisione delle informazioni ad un livello superiore di quello ospedaliero, si pone come obiettivo la creazione di una infostruttura che consenta di richiedere e ricevere dati medici posseduti da altre communities.

Nel profilo XCA viene definito il concetto di *Community* come un insieme di strutture sanitarie che sottoscrivono delle policies e adottano protocolli di comunicazione condivisi con lo scopo di condividere documentazione clinica. Ciascuna Community viene identificata da un codice univoco (homeCommunityId). Ciascuna struttura sanitaria può far parte anche di più community.

Attualmente il profilo XCA detta linee guida concrete principalmente per lo scambio documentale ma è in fase di sviluppo anche l'XCA-I (Cross Community Access for Imaging) che si occupa dello scambio di immagini.

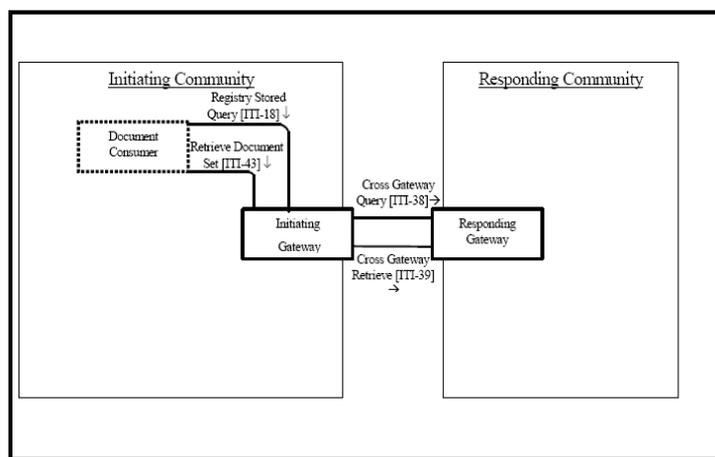
Un grosso vantaggio derivante dall'adozione del profilo XCA è che consente l'interoperabilità sia tra Community strutturate al loro interno secondo il profilo XDS che tra e con Community non aderenti ad XDS o sviluppate secondo principi legacy.

3.8.2 Attori e transazioni

In questo profilo vengono definiti due nuovi attori: *Initiating Gateway* e *Responding Gateway*. Il

primo si prende carico di tutte le transazioni in uscita dalla Coommunity locale, e si preoccupa di inoltrarle alle altre Community. Il Responding Gateway, invece, gestisce tutti i flussi in ingresso dalle altre Community e li recapita agli attori afferenti alla Community locale.

Figura 3.15: XCA New Actor diagram

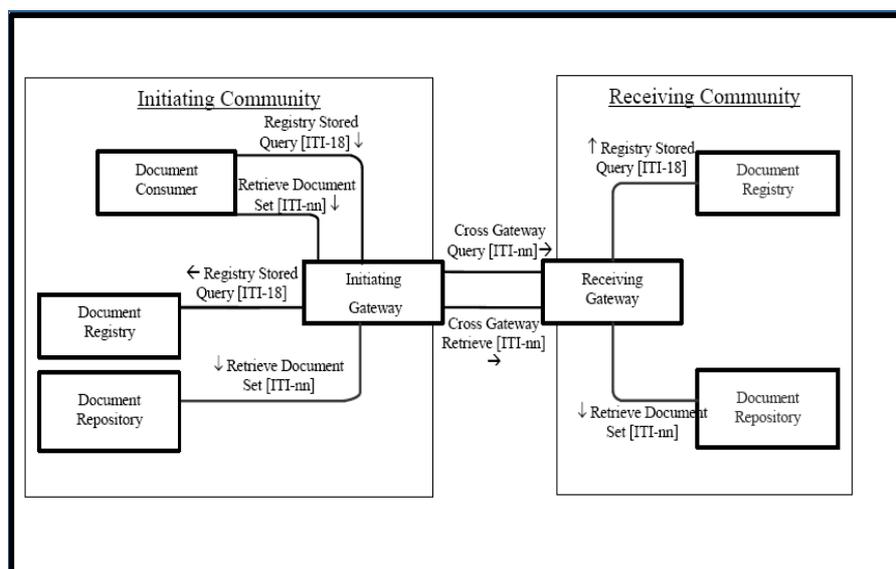


La figura mostra i due nuovi principali attori direttamente coinvolti nell' XCA Integration Profile e le loro rispettive transazioni.

Actors	Transactions	Optionality	Section in Vol. 2
Initiating Gateway	Cross Gateway Query	R	3.38
	Cross Gateway Retrieve	R	3.39
	Registry Stored Query	O	3.18
	Retrieve Document Set	O	3.43
Responding Gateway	Cross Gateway Query	R	3.38
	Cross Gateway Retrieve	R	3.39

Nella tabella sono riportate le transazioni per ogni attore direttamente coinvolto nel profilo XCA. Sono etichettate con "R"(required) quelle richieste obbligatoriamente nell'implementazione, con "O"(optional) quelle opzionali. Accanto ad ognuna si trova il rispettivo capitolo a cui far riferimento nell' IT Infrastructure Technical Framework Volume2.

Figura 3.16: XCA Transaction diagram



-DOCUMENT CONSUMER: Supporta le transazioni Registry Stored query and Retrieve Document Set, il nuovo requisito è la divulgazione dell' homeCommunityId richiesto.

-INITIATING GATEWAY: Per conto degli attori interni, si prende carico di iniziare tutte la transazioni in uscita dalla community. Riceve le transazioni Registry Stored query and Retrieve Document Set provenienti dal Consumer. Inoltre le transazioni al Document Registry e Repository, se appropriate. Si interfaccia con le altre communities attraverso le transazioni Cross Gateway Query and Cross Gateway Retrieve.

-RECEIVING GATEWAY: E' l'unico punto di contatto per tutte le transazioni entranti in una community. Propaga le transazioni in ingresso agli attori locali.

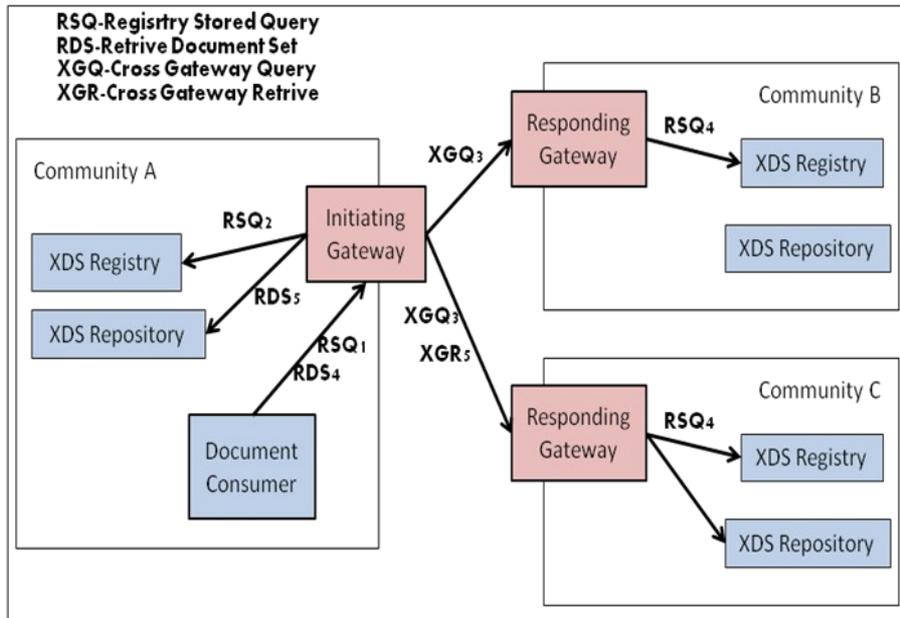
-NEW TRANSACTION

-*Cross Gateway Query:* dall' Initiating Gateway al Receiving Gateway, mima la Registry Stored Query;

-*Cross Gateway Retrieve:* dall' Initiating Gateway al Receiving Gateway, mima Retrieve Document Set.

3.8.3 XCA PROCESS FLOW

Figura 3.17: XCA Process Flow



3.8.4 Use Cases

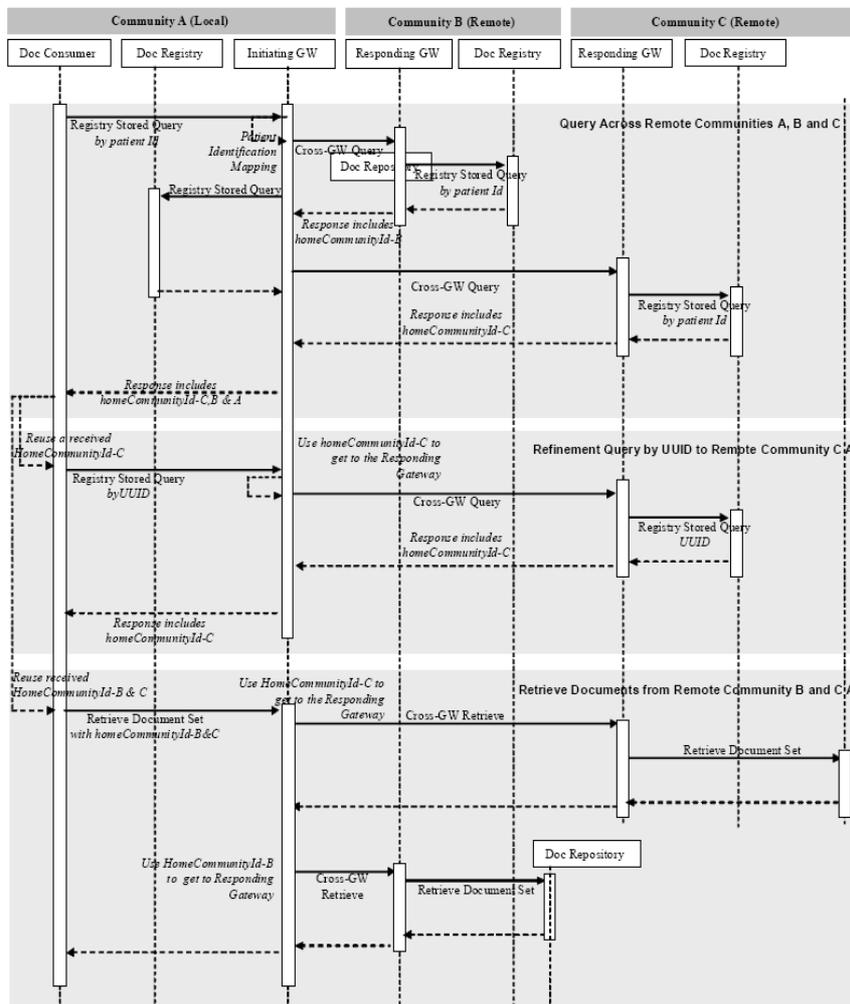
Assumiamo all'interno di un dato dominio, come può essere la Regione Veneto, di avere diverse communities. Una a Padova basata sull'IHE-XDS. Una a Vicenza basata su una diversa infrastruttura per la condivisione dei dati clinici (non segue profilo XDS). Un'altra a Treviso basata anche questa su IHE-XDS. Un paziente X, che viaggia frequentemente, ha ricevuto cure mediche presso ognuna di queste communities. Il paziente X viene ricoverato nell'ospedale di Padova. Il medico curante usa il suo sistema informatico ospedaliero per domandare informazioni mediche sul paziente anche agli altri ospedali e quindi attraverso più domini. Una volta trovate le informazioni, i riferimenti ai dati del paziente fuori dal dominio locale vengono memorizzati localmente per una consultazione futura più semplice.

3.8.5 Iterazioni Dettagliate

A titolo di esempio nel diagramma seguente viene presentata una visuale ad alto livello delle iterazioni tra gli attori quando entrambe le communities, initiating e responding, sono XDS Affinity

Domain.

Figura 3.18: XCA Detailed Interaction



3.8.6 HomeCommunityId defined

Il profilo XCA fa uso di un valore (homeCommunityId) che è un identificatore unico e globale per una community ed è utilizzato per ottenere l'endpoint Web Server del services che provvede ad accedere ai dati di quella community. L' HomeCommunityId :

- viene ritornato con la risposta attraverso le transazioni Cross Gateway Query and Cross Gateway Retrieve per indicare l'associazione dell'elemento di risposta con una community;
- è un parametro opzionale per la Registry Stored Query e per la Retrieve DocumentSet, non è obbligatorio in queste due transazioni indicare a quale community è diretta la richiesta;
- è un parametro opzionale nel caso di richieste che non includono un parametro patientId

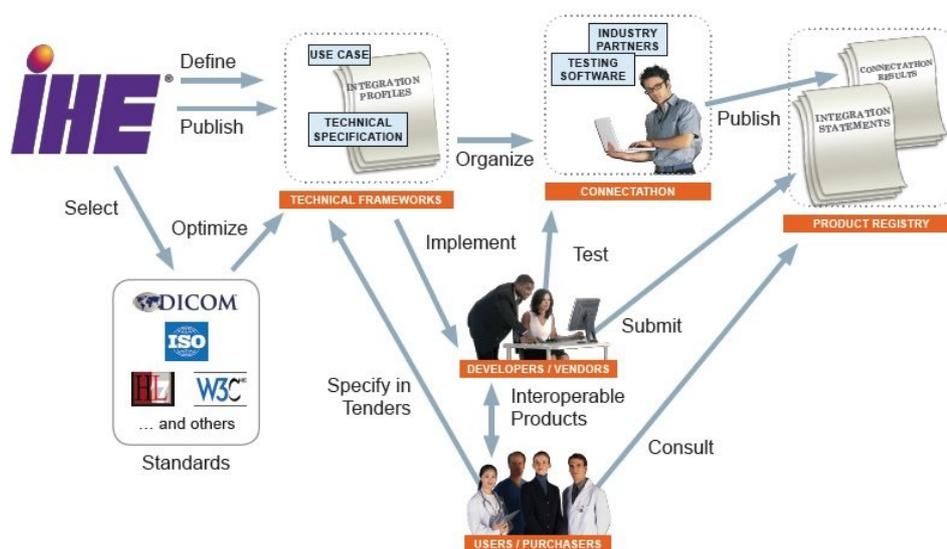
-è usato dall'Initiating Gateways per indirizzare le risposte alla community dove è stata originata la richiesta.

3.9 Processo di definizione ed approvazione di un nuovo profilo IHE

IHE unisce tra loro utilizzatori e sviluppatori di healthcare information technology (HIT), al fine di analizzare problematiche esistenti e arrivare alla definizione di linee guida in materia:

1. Esperti clinici e tecnici definiscono i casi d'uso critici per lo scambio e la condivisione di informazioni;
2. Tecnici esperti creano specifiche dettagliate per la comunicazione tra sistemi per risolvere i casi d'uso proposti selezionando ed ottimizzando degli standard già definiti;
3. Aziende produttrici di applicativi software implementano le specifiche così definite (che vengono chiamati profili IHE) all'interno di sistemi HIT;
4. IHE testa i sistemi proposti dalle varie aziende in eventi accuratamente programmati e supervisionati chiamati Connect-a-thons.

Figura 3.19: Processo completo IHE



Ogni commissione segue questo processo annuale a quattro steps per raggiungere l'interoperabilità nei vari domini clinici.

IHE organizza anche dimostrazioni dell'applicabilità dei sistemi IHE, funzionanti all'interno di scenari del mondo clinico reale, invitando clinici e tecnici esperti per gli specifici domini a collaborare con le commissioni IHE, o inviando la documentazione prodotta per essere commentata pubblicamente. Infatti, più di 200 organizzazioni in tutto il mondo (tra le quali agenzie governative,

fornitori, compagnie HIT ed altri) sono diventati membri di IHE e condividono le iniziative proposte.

Durante gli eventi Connect-a-thons tenuti regolarmente a livello internazionale in diverse località, dei tecnici esperti testano i sistemi proposti dai venditori, attraverso l'utilizzo di software avanzati progettati da IHE e da dei partner. L'IHE Product Registry costituisce un registro che contiene le informazioni utili ad amministratori IT e responsabili esecutivi per la scelta degli applicativi specifici che hanno superato questi test IHE e che devono essere integrati all'interno dei sistemi informatizzati in fase di progetto.

Tenendo presente che la missione dei domini IHE è di andare incontro alle necessità specifiche di ogni dominio, diventa fondamentale per le IHE Regions avere l'opportunità di influenzare i processi di sviluppo degli standards. I vari domini possono sollecitare l'introduzione di nuovi profili attraverso le numerose iniziative Regionali e Nazionali organizzate da IHE.

Il processo di approvazione di un nuovo profilo IHE segue un iter ben preciso. In particolare le nuove proposte devono essere inizialmente valutate da una Planning Committee e quattro distinte fasi in cui la proposta è valutata da delle Technical Committee, dopo le quali il profilo può essere approvato in via temporanea in trial implementation.

-La richiesta, in particolare, deve essere inoltrata a:

- Alla mailing list della Technical Committee e Planning Committee del dominio specifico.

- Alla mailing list della commissione coordinatrice in modo che il responsabile del dominio possa valutare se la richiesta fatta al dominio specifico è appropriata

- Alla mailing list del responsabile regionale, in modo da anticiparne la visione della proposta per una successiva comunicazione.

-La richiesta deve essere effettuata almeno trenta giorni prima della data di scadenza, per permettere ai responsabili di organizzare la propria agenda con anticipo.

-La richiesta deve essere formulata utilizzando uno specifico template proposto, all'interno del quale devono essere presenti:

- Il problema che il profilo vuole affrontare o la necessità che vuole soddisfare.

- Come deve risolvere il problema

- Un potenziale standard al quale può essere applicabile

-La Planning Committee rivede la breve proposta formulata e seleziona una lista di proposte sulle quali è necessario sviluppare le future considerazioni.

-Le proposte che non passano questa prima fase di analisi possono essere risottoposte alla commissione negli anni successivi.

Iniziano a questo punto le Technical Committee. In particolare inizia il lavoro della prima

commissione:

-In questa fase viene attuata la prima valutazione tecnica della proposta focalizzandosi sull'analisi della fattibilità tecnica. Prevede che gli autori del profilo preparino una nuova proposta più dettagliata che in aggiunta alla versione precedente mostri:

- degli use-case più dettagliati che mostrino l'utilità e l'applicabilità del profilo proposto
- maggiori dettagli relativi agli standard più rilevanti
- una rapida descrizione di cosa voglia includere l'IHE supplement proposto (se si tratta di nuovi attori, nuovi profili o nuove transazioni).
- Una lista di potenziali rischi che verrebbero gestiti dalla nuova proposta
- Se possibile, una stima degli effetti ipotizzabili legati al superamento dei problemi descritti e che la proposta permetterebbe di superare

-La Planning Committee invia la proposta dettagliata alla Technical Committee .

-La Technical Committee valuta la proposta dettagliata focalizzandosi specificatamente sulle sezioni che riguardano:

- stima degli sviluppi potenziali del profilo e dell'implementabilità del prodotto
- editore del profilo candidato
- individuazione delle risorse allo sviluppo del profilo (chi aiuterà nella stesura del profilo stesso) e individuazione delle risorse informative (il responsabile alle comunicazioni relative al profilo con la Technical Committee)
- Chiarificazioni e raccomandazioni riguardanti lo scopo del profilo
- Il budget necessario per permettere lo sviluppo del profilo
- Risposte alle questioni aperte dalla Planning Committee . La Technical Committee e la Planning Committee lavorano congiuntamente per scegliere e valutare le proposte.

-La proposta così aggiornata viene rimandata alla Planning Committee

-La Planning Committee valuta le proposte che possono avanzare alla fase finale di selezione

- I voti non vengono assegnati alla singola proposta ma ad una collection di proposte (certe combinazioni di proposte creano evidentemente dei gruppi di lavoro che interagiscono strettamente per rendere il processo di definizione più efficiente e proficuo)
- Viene presa in considerazione anche la IHE Pipeline Impedence Matching

-Vengono comunicate le proposte scelte per i successivi sviluppi.

-Le successive analisi ad opera delle Technical Committee sono volte a seguire dettagliatamente gli sviluppi del profilo. In particolare dopo la terza Technical Committee il profilo passa un periodo di public Comment, durante il quale ogni membro di IHE ed ogni ente competente con le specifiche tematiche affrontate può commentare la proposta fatta.

-A questo punto durante l'ultima Technical Committee vengono analizzati i commenti fatti, apportate le modifiche opportuna e quindi, se viene approvato, il profilo viene pubblicato e iniziano le fasi di trial implementation.

4. Analisi dei Linguaggi Standard per la gestione di Workflow

4.1 Introduzione

Un processo clinico può essere definito come una sequenza di operazioni svolte dai vari attori che partecipano al processo. In quest'ottica, questo flusso di operazioni può essere pensato come un Workflow clinico, all'interno del quale i vari attori del sistema sanitario operano per raggiungere un determinato business. Data la complessità di questi sistemi, le linee guida IHE non permette di gestire correttamente i cambiamenti di stato, e lo scambio di informazioni che possono avvenire, ed in particolare, risulta necessario tenere traccia dello stato di avanzamento di ogni Workflow clinico per poterne avere uno sviluppo seguente corretto. In questa sezione, quindi, viene presentata la ricerca bibliografica eseguita con lo scopo di ritrovare in letteratura degli standard descrittori di workflow in grado di tenere traccia degli step eseguiti, e trarre da questi informazioni utili per progettare un documento in grado di svolgere lo stesso compito nel caso dei Workflow clinici. Non ci si è focalizzati sull'indagine dei soli linguaggi per i quali è definito un esplicito utilizzo in sanità, preferendo un'analisi a più ampio spettro che includesse il più elevato numero possibile di standards. L'utilizzo dei linguaggi descrittori di workflow è infatti molto diffuso in svariati ambiti, ed in particolare svolge un ruolo fondamentale in tutte quelle applicazioni automatizzabili che prevedono il coinvolgimento di servizi distribuiti ed eterogenei. E' proprio in queste realtà infatti che risultano evidenti le potenzialità associate al tracking del workflow. Nel caso clinico inoltre, vi è il forte limite correlato all'impossibilità di modificare i documenti associati ad un particolare Workflow, impedendo in questo modo di memorizzare all'interno degli stessi i cambiamenti di stato subiti. Questo risulta particolarmente evidente i diversi esempi pratici quali per esempio il workflow associato alla prescrizione digitale farmaceutica, od al workflow associato al processo di teleconsulto neurochirurgico. Nel primo caso infatti, è fondamentale poter tenere traccia dello stato di avanzamento del workflow del documento di e-Prescription, per poter stabilire le operazioni possibili associate allo stesso (prenotazione, dispensazione, ecc..), impedendo il riutilizzo del documento. Nel secondo caso invece, risulta fondamentale tenere traccia dello stato del teleconsulto, per poter stabilire con precisione l'istante in cui si è verificato un evento e l'autore che

l'ha eseguito.

All'interno di questo capitolo, in particolare, dopo aver definito cosa è un Workflow e un linguaggio descrittore di Workflow, si sono analizzate le principali funzionalità connesse a questo tipo di approccio per lo svolgimento di specifici business. L'ultima parte della sezione è dedicata ad un'analisi dettagliata dei principali linguaggi analizzati, facendo particolare attenzione alla potenziale tracciatura degli step di un'istanza del Workflow stesso.

4.2 Descrittori di Workflow

L'affermarsi dei Web Services, e la crescente necessità di gestire sistemi e applicazioni sempre più complesse hanno contribuito allo sviluppo di numerosi dialetti XML, ognuno dei quali rispondente a specifiche necessità e riferito ad uno specifico campo di applicazione. In modo immediato è possibile osservare l'elevato numero di linguaggi standard proposti, legato alla grande eterogeneità dei problemi che possono essere trattati mediante questo approccio.

I Grid Workflow possono essere pensati come una collection di attività che devono essere portate a termine in un ordine ben preciso. Ogni attività deve poter essere eseguita su risorse eterogenee e non vi è alcun controllo centralizzato del processo di esecuzione (non si tratta semplicemente di una lista sequenziale di attività la cui esecuzione viene lanciata dall'esterno). Strutturare i processi in questo modo permette di superare i problemi di integrazione tra parti del sistema eterogenee e di ottenere così sistemi meno costosi e più flessibili rispetto a soluzioni monolitiche e caratterizzati da un elevato grado di interoperabilità.

Secondo la definizione data dal World Wide Web Consortium (W3C) un Web Service (servizio web) è: *“un sistema software progettato per supportare l'interoperabilità tra diversi elaboratori su di una medesima rete”*. La caratteristica fondamentale di un Web Service è quella di offrire un'interfaccia software (descritta in un formato automaticamente elaborabile quale, ad esempio, il Web Services Description Language) utilizzando la quale altri sistemi possono interagire con il Web Service stesso attivando le operazioni descritte nell'interfaccia tramite appositi "messaggi" inclusi in una "busta" (la più famosa è SOAP). Tali messaggi sono solitamente trasportati tramite il protocollo HTTP e formattati secondo lo standard XML. Grazie all'utilizzo di standard basati su XML e attraverso un'architettura basata sui Web Services (chiamata, con terminologia inglese, Service oriented Architecture - SOA) applicazioni software scritte in diversi linguaggi di programmazione e implementate su diverse piattaforme hardware possono quindi essere utilizzate, tramite le interfacce che queste "espongono" pubblicamente e mediante l'utilizzo delle funzioni che sono in grado di effettuare (i "servizi" che mettono a disposizione) per lo scambio di informazioni e l'effettuazione di

operazioni complesse. L'utilizzo dei Web Services permette di ottenere un maggior livello di interoperabilità tra applicazioni basandosi sugli standards del web. Questo approccio è più oneroso della semplice adozione di linguaggi standard, in quanto si ha bisogno di modelli di integrazione comuni dei processi, in modo da fare comunicare tra loro i servizi.

Il processo fondamentale per arrivare all'applicazione di un linguaggio standard è la definizione di un modello virtuale di risorse eterogenee, permettendo così una visione globale delle risorse stesse e una gestione unificata di autenticazioni, autorizzazioni e account per gli utenti finali. Questo è ottenuto tramite il Grid computing. Le realtà più importanti a livello normativo e di comunità di sviluppo nell'ambito dei linguaggi descrittivi di Workflow sono essenzialmente tre:

-World Wide Web Consortium (W3C): sviluppa tecnologie che garantiscono interoperabilità, agendo da forum di informazioni comunicazioni e attività comuni. (L'ufficio italiano è W3C-IT);

-Global Grid Forum (GGF): sponsorizza presso la comunità internazionale l'adozione del Grid computing sia in ambito di ricerca che industriale, considerandolo come migliorativo per la progettazione e implementazione di progetti e decisivo per la riduzione dei costi di sviluppo delle tecnologie IT;

-Workflow Management Coalition (WfMC): organizzazione internazionale no-profit di aziende, utenti, analisti, università, gruppi di ricerca rivolti ai workflow. Il suo scopo è quello di promuovere e sviluppare l'utilizzo dei workflow attraverso standard per la terminologia, l'interoperabilità e la connettività a livello software tra prodotti basati su workflow.

4.3 WORKFLOW

4.3.1 Definizione di Workflow

Può essere fornita una definizione per il termine Workflow:

“Automatizzazione di un processo (parziale o completa) in cui dei compiti passano da un partecipante all'altro all'interno del gruppo di lavoro, conformemente ad un insieme di regole predefinite. Un sistema di workflow definisce, crea e gestisce l'esecuzione di tali processi.”

Un Workflow è quindi una struttura che racchiude le attività da svolgere, che coordina e

supervisiona la loro esecuzione.

I Workflow si basano sull'applicazione di tecnologie standard utilizzate nella definizione dei Web Services: XML e WSDL.

XML: (eXtensible Markup Language) è un metalinguaggio in grado di modellare agevolmente una rilevante quantità di aspetti del mondo informatico. E' un linguaggio marcatore che definisce un meccanismo sintattico che consente di estendere o controllare il significato di altri linguaggi marcatori (ogni utente è libero di creare il proprio linguaggio da zero o estendendo schemi già esistenti).

WSDL: (Web Service Description Language) è il linguaggio di descrizione delle interfacce dei servizi ed è pensato per l'automatizzazione delle interazioni. E' una specifica che utilizzando una grammatica XML indica come descrivere un web service. Si può così risalire a diverse informazioni riguardanti il servizio a cui il file appartiene. Permette di descrivere:

- operazioni che il web service espone al mondo esterno;
- il formato dei dati che rappresentano l'input e l'output delle funzionalità offerte;
- informazioni riguardanti il protocollo di trasporto utilizzato;
- modo di localizzare il servizio.

4.3.2 Tipologie di workflow

Esistono diverse tipologie di Workflow, utilizzabili in funzione del livello di conoscenza del sistema da strutturare e degli obiettivi del processo stesso.

Workflow collaborativo: gestisce processi meno rigidi e prevede una relazione tra gli utenti più vicina a quella della collaborazione tra gruppi di lavoro. In pratica si tratta di un'evoluzione dei prodotti groupware integrati con motori di workflow;

Workflow strutturato: gestisce attività strutturalmente ben definite e ripetibili, che possono essere specificate attraverso una serie di regole;

Workflow ad Hoc: viene realizzato utilizzando sistemi più leggeri che lasciano all'utente il compito di eseguire di sua iniziativa passi significativi della procedura ogni volta sia richiesta una modifica

dinamica al processo di esecuzione.

4.3.3 Benefici dei Workflow

L'approccio ad un problema mediante l'utilizzo dei Workflow consente, oltre al grandissimo vantaggio di poter raggiungere elevatissimi livelli di interoperabilità, di ottenere numerosi benefici direttamente connessi al processo modellato. In particolare:

1. **automazione del processo**: riduzione dei tempi e degli errori, miglioramento della qualità del servizio offerto e miglioramento nell'organizzazione e nel controllo dei processi.
2. **esecuzione**: supporto alla comunicazione, analisi e comprensione dei fenomeni legati al ciclo di vita dei processi.
3. **monitoraggio del processo**: misura performance e supporto decisionale
4. **analisi del processo**.

4.3.4 Linguaggi standard per la descrizione di Workflow

Possono essere individuate due generazioni di linguaggi standard per la descrizione di workflow. La prima generazione (SOAP WSDL e UDDI) consentono la connettività e la comunicazione tra consumer e provider del servizio, ma non permettono lo sviluppo di soluzioni solide che garantiscano sicurezza, efficienza e qualità del servizio. Per questo motivo sono state sviluppate una serie di specifiche di seconda generazione volte a superare questi limiti. Ogni linguaggio preso in esame (posto che tutti si basano sullo schema XML) presenta caratteristiche comuni sia tra di loro, sia con i più diffusi linguaggi di programmazione ad alto livello: strutture di controllo del flusso di esecuzione, condizionali e di iterazione, oltre alla possibilità di gestire degli errori, e le eccezioni. Ciò che invece li distingue è principalmente il livello di astrazione con cui presentano allo sviluppatore gli strumenti a disposizione: maggiormente standard ed integrato il WSFL, più orientato al computing distribuito il GSFL, più votato alla parallelizzazione l'AGWL, più standardizzato ed estensibile l'XPDL, più flessibile lo SCUFL, più descrittivo il WSCI, maggiore generalità per il MOML, più supportati da grossi nomi commerciali e volti all'enterprise business il BPEL4WS ed il BPML. In particolare i più importanti linguaggi descrittivi di Workflow analizzati sono:

-WSFL (Web Services Flow Language)

- GSFL (Grid Services Flow Language)
- AGWL (Abstract Grid Workflow Language)
- XPDL (XML Process Definition Language)
- SCUFL (Simple Conceptual Unified Flow Language)
- WSCCI (Web Service Choreography Interface)
- MOML (Media Objects Markup Language)
- BPEL4WS (Business Process Execution Language for Web Services)
- BPEL4People (Business Process Execution Language for People)
- BPML (Business Process Modeling Language)
- XFlow (XML-based document-centric Workflow)
- XLR (eXchangeable Routing Language)
- XDOC-WFMS (XML Document Centric Workflow Management System)
- XLANG (XML-based Language)

4.3.4.1 WSFL (Web Services Flow Language)

E' un linguaggio basato sull' XML, progettato da IBM. Utilizza WSDL per descrivere le interfacce dei servizi ed i relativi protocolli.

WSFL attiva Web Services come implementazioni delle attività di un business process, quindi ogni attività è associata al service provider responsabile dell'esecuzione di ciascun passo del processo. Questa relazione definisce l'associazione tra le attività presenti nell'applicazione e le operazioni offerte dal service provider. Graficamente il linguaggio pensa di modellare il problema come un grafo orientato. Il principale vantaggio offerto da questo tipo di linguaggio è quello di offrire un controllo diretto sul lifecycle del processo.

WSFL considera due possibili tipologie di composizione di Web Services:

- il primo modo specifica il più appropriato *usage pattern*, ovvero individua la composizione di servizi in modo tale da realizzare una specifica funzione e tipicamente il risultato di questa composizione è la descrizione del business process.
- il secondo tipo invece specifica il pattern di interazioni tra Web Services e in questo caso il risultato è la descrizione delle interazioni che avvengono tra i partner.

Nel primo caso la composizione è creata descrivendo come utilizzare le funzionalità offerte dalla collection di WS ed è noto come *flow composition*. WSFL modella questa composizione specificando la sequenza di esecuzione delle funzionalità dei WS coinvolti. L'ordine di esecuzione è specificato dalla definizione dei flussi di dati e di controllo tra i WS. Per questo motivo questo tipo di approccio viene definito Modello di flusso.

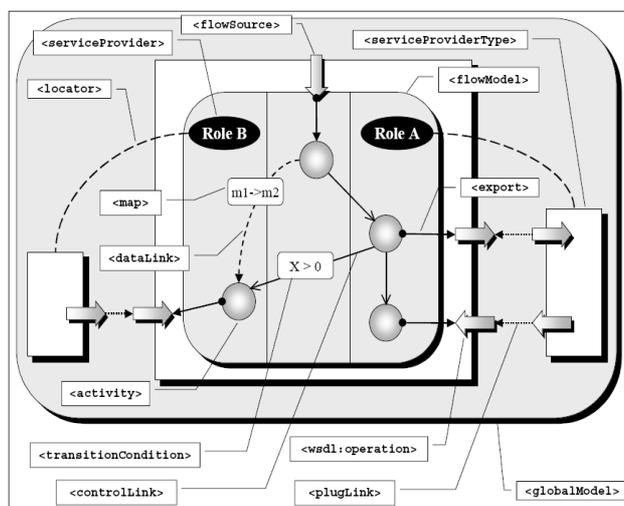
Nel secondo caso non viene fornita nessuna sequenza di esecuzione. Dunque la composizione permette di descrivere come i vari WS interagiscono tra loro. Queste interazioni sono modellate tramite legami tra punti di uscita delle interfacce associate ai vari WS (ogni link corrisponde ad una interazione tra un WS e l'operazione di un interfaccia di un altro WS). A causa della natura decentralizzata e distribuita di queste interazioni si utilizzerà la definizione di Modello Globale.

Il linguaggio WSFL permette l'utilizzo ricorsivo delle composizioni di WS, in modo tale che una qualsiasi composizione di WS (sia Global Model che Flow Model) possa essere considerata nella sua totalità come un WS. Questa proprietà permette di rappresentare il sistema in modo più aggregato, fornendo maggior leggibilità al linguaggio stesso. Il linguaggio WSFL supporta anche diversi modelli di interazioni tra i partner coinvolti nel business process. In particolare sono possibili interazioni gerarchiche (che in genere sono stabili e a lungo termine) e relazioni peer-to-peer stabilite dinamicamente e definite solo per la durata dell'istanza che ha generato l'interazione.

Per definire un'applicazione del linguaggio è necessario stabilire:

- business process;
- business rules che permettono di passare a step successivi;
- flussi di informazioni tra i vari step del processo.

Figura 4.1: Sistema WSFL

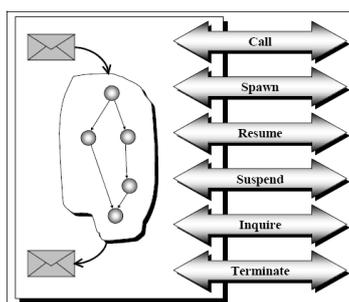


Il flow model descrive la struttura del business process: le attività (rappresentate con cerchi) rappresentano i vari step del processo, i dati e i controlli rappresentano i flussi di informazione e le regole per passare da uno step all'altro. Per ogni attività:

- va individuato il service provider responsabile dell'esecuzione dello step del processo
- vanno individuate le associazioni tra le attività e le operazioni offerte dal service provider (elementi export e plug-in)

Utilizzando WSDL si può definire l'interfaccia tramite la quale un consumer accede al flow model e l'interfaccia attraverso cui il provider accede al flow model. Per ogni attività ci sarà un'operazione eseguita dall'interfaccia esterna del flow model che permetterà di interagire con il service provider che implementa quell'attività stessa. In questo modo il sistema può essere pensato come un repository di servizi ai quali accedono due parti: chi vuole fornire il servizio attraverso il flow model e chi vuole essere il consumatore del servizio implementato dal flow model. Questo tipo di interazione è descritta dal global model. Il flow model impone un vincolo nel sequenziamento delle attività. Vengono definiti tutti i provider, dei quali vengono specificate dei legami vincolanti, che definiscono i servizi che vanno utilizzati una volta che il modello viene istanziato (questi legami possono essere statici oppure dinamici e quindi variare a seconda delle condizioni in cui viene interpellato il provider). Ogni attività è associata ad un provider, e l'ordine di precedenza con cui vengono eseguite le attività è definito dai collegamenti tra le attività stesse. Esistono due tipologie di possibili collegamenti tra le attività: *data connection* o *control connection*. La prima connette il completamento di un'attività con l'esecuzione di un'altra; mentre la seconda rappresenta uno scambio di dati tra due attività. Questa distinzione è una grande potenzialità del linguaggio perché permette di eseguire un'operazione senza scambio di informazioni ma solo al completamento di un'altra attività. E' possibile gestire direttamente il lifecycle delle istanze, intervenendo in fase di esecuzione. Ogni modello di flusso è associato sempre a delle porte che permettono la gestione del lifecycle di un'istanza del flow model stesso attraverso specifiche operazioni. Permette per esempio di eseguire un'istanza del modello, sospenderla e poi riprendere l'esecuzione. Un'ulteriore possibilità è quella di interrogare l'applicazione per individuarne lo stato (questo rappresenta l'unico modo in cui risulta possibile tracciare il workflow dell'applicazione). Queste porte sono implementate mediante l'utilizzo dello standard WSDL. Ogni istanza è caratterizzata da un specifico ID (che viene restituito dall'operazione di spawn), il quale permette di applicare operazioni alla particolare istanza che identifica.

Figura 4.2: Porte di accesso al sistema



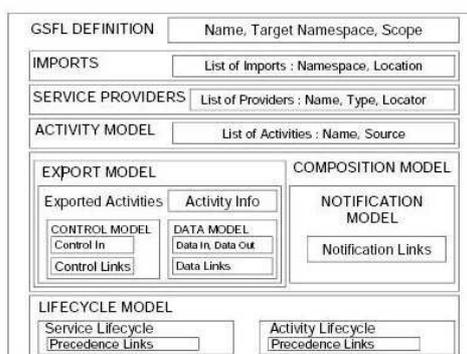
Il linguaggio non offre la possibilità di tracciare la sequenza di attività svolte all'interno di un istanza. L'unico modo per valutare le prestazioni di un esecuzione è quello di appoggiarsi ad un apposito Web Service adatto a monitorare i vari step del processo.

4.3.4.2 GSFL (Grid Services Flow Language)

GSFL è uno strumento per l'analisi dei workflow per Web Services destinati ad impiego in ambienti Grid. GSFL è fortemente orientato al calcolo distribuito, ed è stato progettato per sopperire alla mancanza di un unico linguaggio che potesse sfruttare al meglio le risorse che Grid mette a disposizione in questo ambito. Le principali caratteristiche del linguaggio GSFL sono la composizione ricorsiva dei Grid Services, la comunicazione peer-to-peer effettiva tra i servizi e la gestione individuale a livello di servizio del Lifecycle.

Nonostante la possibilità di agire direttamente sul lifecycle il linguaggio non offre la possibilità di memorizzare gli step eseguiti dall'applicazione.

Figura 4.3: Struttura del sistema GSFL



4.3.4.3 AGWL (Abstract Grid Workflow Language)

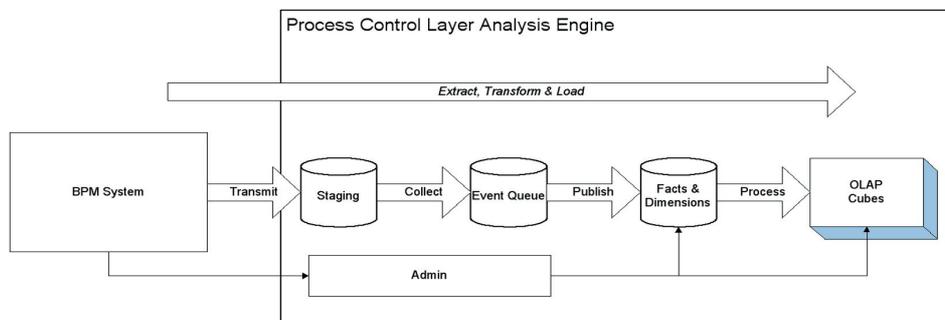
AGWL è un linguaggio per workflow potente ed user-oriented, pensato per applicazioni Grid in ambito scientifico e performance-oriented; inoltre fornisce strumenti avanzati per la parallelizzazione, la modularizzazione e il riuso dei workflow. Proprietà e vincoli possono essere specificati per ottimizzare l'esecuzione delle applicazioni in base al middleware Grid sottostante. Inoltre AGWL è un dialetto XML utilizzato per descrivere applicazioni di Grid Workflow ad un alto livello di astrazione: è stato progettato per permettere all'utente di concentrarsi non tanto sulla struttura della rappresentazione del servizio, ma sulla descrizione dell'applicazione scientifica a cui lavora. L'utilizzatore non deve perciò preoccuparsi del modo in cui i dati di input e output delle varie attività vengono passati tra un blocco e il successivo e di come vengono invocate e terminate le attività componenti il servizio o dell'infrastruttura Grid sottostante. AGWL permette la creazione di workflow che presentino un vero parallelismo nell'analisi dei dati. AGWL è basato su XML ed utilizza una rappresentazione a grafo dove le attività, collegate tra loro da transizioni, rappresentano computational tasks o interazioni definite dall'utente. Come detto tali attività sono in relazione una all'altra tramite transizioni manipolabili, strutture di controllo o direttamente attraverso i dati. Gli sviluppatori del AGWL hanno progettato il linguaggio in modo che supporti l'orchestrazione di applicazioni di Grid workflow attraverso un ampio set di costrutti quali sequenze di attività, sottoattività, meccanismi di controllo del flusso di esecuzione (*exclusive choice, sequential loops*), meccanismi di controllo dei dati (porte di input/output) e data repositories. Inoltre AGWL cerca di andare oltre i linguaggi esistenti proponendo costrutti avanzati quali l'esecuzione parallela di attività, la possibilità di imporre pre e post-condizioni, cicli paralleli, meccanismi di sincronizzazione degli eventi e selezione di attività in base allo stato di determinati parametri. In aggiunta, AGWL permette all'utente la definizione ad alto livello di vincoli e proprietà delle attività e dei dati facenti parte dell'applicazione.

Il file AGWL può essere realizzato in ambiente friendly o può essere direttamente scritto con linguaggio XML. Attraverso un compilatore il file AGWL viene trasformato in un file che rappresenta un workflow eseguibile e infine attraverso l'ausilio di un applicazione (es. ASKALON) viene eseguita l'applicazione corrispondente al workflow prodotto. Queste applicazioni esterne possono fornire indicazioni sulla particolare istanza eseguita, anche tracciando il workflow effettivo.

4.3.4.4 XPDL (XML Process Definition Language)

L'XML Process Definition Language (XPDL) è un linguaggio proposto dalla Workflow Management Coalition (WfMC) per la definizione di processi in un ampio spettro di ambiti differenti; la sua sintassi è basata su XML e specificata da un XML Schema. Lo scopo del XPDL è mettere a disposizione una *lingua franca* nell'ambito delle applicazioni workflow per la modellazione di problematiche reali, consentendo un utilizzo più efficace della grande varietà di tools già a disposizione per affrontare tali casistiche. Il nucleo del XPDL è un insieme minimo di costrutti presenti nella maggior parte dei sistemi di gestione dei workflow, per cui non vi è diretto supporto ad alcuni dei più comuni pattern workflow; a questo scopo l'XPDL possiede specifiche estensioni per adattarsi ai possibili casi d'uso, pur mantenendo una generalità di base. XPDL modella lo stesso problema da differenti prospettive, fornendo un file in formato XML da utilizzare come punto di interscambio comune tra le parti di un processo. XPDL ha un linguaggio di strutturazione a grafo basata su processi e attività (nodi) e transizioni (archi). La definizione del processo non può essere a sua volta racchiusa da altre definizioni e le transizioni vengono gestite attraverso la specificazione dei rapporti e dello scambio di messaggi tra le attività. Le condizioni associate a queste transizioni determinano a runtime quali attività verranno eseguite e quindi l'ordine di esecuzione. XPDL focalizza la sua attenzione sull'importanza della distribuzione del carico di lavoro; ogni attività ha un attributo che specifica la o le risorse necessarie alla propria esecuzione. Il file XML alla base della definizione di un processo secondo il linguaggio standard XPDL (versione 2.1), non è utilizzato solo per realizzare un lavoro attraverso la sua esecuzione o per modellare compiti, ma costituisce anche le informazioni base da fornire ai BAM (Business Activity Monitoring) reporting tools e soprattutto determina le dimensioni e i membri per OLAP (OnLine Analytical Processing) cube reporting technologies.

Figura 4.4: Sistema per tracciatura del Workflow



Per monitorare l'istanza il linguaggio propone di utilizzare un servizio Administrator che invia il processo definito tramite il modello XPDL ad un motore di analisi, e trasmette contemporaneamente il flusso di log events che descrivono i dettagli della particolare esecuzione del processo. Il motore di analisi struttura il database e gli OLAP cubes (web services per l'analisi di grandi moli di dati che si basano sulla definizione di un database che permette di associare tra loro i dati in record multidimensionali che permettono query veloci) in base a come è stato definito il processo, ai partecipanti e alla coda di informazioni che vengono fornite.

4.3.4.5 SCUFL (Simple Conceptual Unified Flow Language)

SCUFL è un linguaggio di alto livello per la descrizione del workflow. Il formalismo sfrutta informazioni riguardanti l'ambiente per ridurre la quantità di dati da inserire esplicitamente all'interno della definizione del workflow (da cui il prefisso Simple). Anche in fase realizzativa, il linguaggio cerca di tenere i dettagli implementativi al minimo e di far concentrare l'utente sulla modellazione concettuale di un problema. SCUFL si sta affermando come uno dei più diffusi standard della comunità, con lo scopo di coordinare tra loro diversi Web Services unificando tecniche e strumenti a disposizione. SCUFL è supportato dal workflow enactment engine FreeFluo e dal workbench Taverna per l'editing ed il testing di workflow SCUFL. Taverna si appoggia al Web in due modi: prendendo codice SCUFL dinamicamente da URIs (ad esempio URIs per descrizioni di interfacce in WSDL) o scandagliando il Web alla ricerca di potenziali processori che i suoi utenti possono inserire nel loro workflow. SCUFL aderisce agli standards XML, su cui è basato; per ottenere le descrizioni degli elementi che esso modella è necessario utilizzare un parser.

Il vantaggio di essere ad alto livello non consente di fornire le indicazioni su come sono avvenuti gli accessi ai WS durante un'esecuzione, quindi non viene fornito all'utente la possibilità di tracciare il flusso delle attività eseguite.

4.3.4.6 WSCI (Web Service Choreography Interface)

Il WSCI (Web Services Choreography Interface) è un linguaggio basato sullo standard XML per descrivere il flusso dei messaggi scambiati tra Web Services, focalizzandosi sugli aspetti di interazione e comunicazione; fornisce un punto di vista preferenziale sul comportamento del sistema ed è stato sottoposto al W3C da figure quali l'Intalio, BEA, SAP e Sun Microsystems.

Sostanzialmente WSCI descrive il comportamento osservabile di un Web Service in termini di dipendenze logiche e temporali dai messaggi scambiati dal sistema (come regole sulla sequenza di tali messaggi, correlazione, exception handling e transazioni). Quindi si può dire che WSCI consente di ottenere una *visione message-oriented* dell'intero sistema. Scendendo più nel dettaglio, WSCI non si preoccupa di analizzare a fondo l'implementazione interna del processo che guida lo scambio dei messaggi, piuttosto il suo scopo è quello di permettere ai progettisti ed ai tools di sviluppo di comprendere dinamicamente l'evolversi del sistema. Le descrizioni ottenute con il WSCI possono essere validate con i più comuni strumenti a supporto del mondo XML.

I Web Services giocano un ruolo chiave nell'architettura emergente del Web-based Grid computing: uno stack di standard, il *Web Services Computing Stack* si sta affermando per assicurare interoperabilità tecnica e semantica tra i Web Services. Tale stack, sviluppato dal W3C Web Services Description Working Group. Parallelamente, altri standards stanno costruendo una semantica per i processi in ambito enterprise business, interoperabilità nelle collaborazioni ed approccio top-down. Si assume che lo stack proposto dal W3C e questi nuovi standard debbano in qualche modo interfacciarsi con uno strato comune posto al centro. Si necessita quindi di un'architettura flessibile, che renda possibile la comunicazione in entrambi i sensi e il linguaggio WSCI costituisce il primo passo per raggiungere questo obiettivo. WSCI è sì un componente dello stack, ma al tempo stesso anticipa un'integrazione con i più alti livelli di astrazione e collaboratività. Il metamodello su cui si basa permette di fornire una visione del processo message-oriented, descrivendo le interazioni tra i Web Service ma senza definire le proprietà e il funzionamento del WS stesso.

Questo linguaggio soffre di certi limiti: innanzitutto WSCI non definisce il middleware di appoggio che connette i Web Services coinvolti nell'applicazione, non è un linguaggio per la definizione di processi esecutivi in quanto non definisce come il Web Service gestisce il processo al suo interno e non definisce alcun meccanismo per identificare il servizio target.

Questo linguaggio non garantisce le proprietà richieste nell'ambito healthcare in quanto non focalizza l'attenzione sulle attività che possono essere svolte dai vari attori che partecipano al workflow, ma definisce solo i messaggi scambiati tra vari soggetti. Questo linguaggio si applica nel caso in cui risulti necessario descrivere le relazioni tra WS in un workflow predefinito.

4.3.4.7 MOML (Media Objects Markup Language)

Il Media Object Markup Language (MOML) è un linguaggio di modellazione basato su XML modulare ed estensibile. Il linguaggio base definisce un set di primitive, gli strumenti per gestirle e

per specificare come interagiscono tra loro. MOML è un linguaggio il cui campo di applicazione è molto più ampio rispetto alla maggior parte dei linguaggi basati su XML e tale generalità si traduce ovviamente in un basso supporto per ambienti specifici come quello della modellazione dei workflow. Gli sviluppatori possono utilizzare la versione base di MOML o le sue estensioni a piacimento per creare oggetti su misura per le loro applicazioni e ciò ha permesso ai laboratori dell'Università di Berkeley, di produrre gli strumenti che prestino supporto alla creazione di applicazioni workflow. Essendo uno strumento così generico, una maggiore difficoltà ad implementare funzioni o costrutti rispetto a linguaggi molto specializzati viene compensata dalla possibilità di creare strutture ad hoc, altrimenti difficilmente implementabili.

MOML, come altri linguaggi simili, rappresenta i workflow con una struttura a grafo in cui i blocchi sono le attività e le connessioni tra essi sono il passaggio di dati tra l'output dell'attività precedente e l'input della successiva.

Il linguaggio focalizza l'attenzione sulla possibilità di implementare costrutti logici complessi come le iterazioni. Non viene proposta la possibilità di tracciare il flusso delle attività svolte.

4.3.2.8 BPEL4WS (Business Process Execution Language for Web Services)

Diverse aziende informatiche hanno sviluppato un vivo interesse verso gli aspetti relativi ai workflow e alle loro applicazioni. Il risultato più importante dei lavori svolti in quest'area è lo standard BPEL, esteso in seguito, per trattare con i Web Services, in BPEL4WS. BPEL4WS , ovvero Business Process Execution Language for Web Services, è stato sviluppato da IBM e supportato da Microsoft, SAP, BEA e Siebel: rappresenta una convergenza dei concetti presenti nelle specifiche degli standard XLANG (Web Services for Business Process Design) e WSFL (Web Services Flow Language), rispettivamente di proprietà Microsoft ed IBM ed interagisce con Web Services di standard WSDL (v. 1.1). BPEL4WS è una estensione XML che definisce delle strutture utili per gestire l'orchestrazione di un insieme di Web Service permettendo alle aziende l'utilizzo, la gestione e l'integrazione di questi ultimi senza occuparsi minimamente dell'infrastruttura informatica alla base del sistema. In pratica, BPEL4WS consente ad un processo di utilizzare vari servizi offerti da dei partner, permettendo l'automatizzazione di alcuni cicli aziendali. Le sue origini sono da ricercarsi nei tradizionali modelli di flusso e nei linguaggi di programmazione strutturata, dai quali prende in prestito diversi concetti. Tutto questo si basa sulle tecnologie standard nel campo dei Web Service: XML e WSDL. Il BPEL4WS è un linguaggio ad alto livello e deve essere sottoposto a parsing con tool quali l'IBM BPEL4WS Parser (facente parte della IBM BPEL4WS

SDK sviluppata dalla IBM medesima e disponibile come open source), il MSXML (Microsoft XML) o il Xerces Java Parser entrambi liberi. L'estensione BPEL4People permette di implementare all'interno delle applicazioni modellate anche human tasks, diventando così più funzionale nelle applicazioni healthcare dove devono essere modellati come attori del sistema non solo dei WS ma anche persone coinvolte nel processo.

4.3.2.9 BPEL4People (Business Process Execution Language for Web Services)

L'estensione BPEL4People permette di implementare all'interno delle applicazioni modellate anche human tasks, diventando così più funzionale nelle applicazioni healthcare dove devono essere modellati come attori del sistema non solo dei WS ma anche persone coinvolte nel processo.

Questo linguaggio offre la possibilità di memorizzare la storia delle attività svolte. In particolare, per ogni evento task vengono memorizzate una serie di importanti informazioni correlate:

- event_id
- event_time
- task_id
- user (principal)
- event_type
- event_data
- fault_name
- startOwner
- endOwner

Inoltre certi eventi possono avere dati associati all'evento stesso, ed essere associati alla task stessa. Ogni evento viene memorizzato in un database, che può essere interrogato per ricavare le informazioni relative alla storia di una particolare istanza dell'applicazione. E' possibile indagare il database ottenendo in risposta solo la descrizione degli eventi, oppure anche i dati associati ad ogni evento (questo permette di ottenere, se necessario, ricerche più veloci):

- può essere ritornata lista di tutti gli eventi con dati associati;
- può essere ritornata lista di tutti gli eventi senza dati associati;
- può essere ritornato un sottoinsieme di eventi svoltisi in un certo intervallo;

-può essere ritornata una lista filtrata di eventi.

4.3.2.10 BPML (Business Process Modeling Language)

BPML è stato sviluppato da Intalio, con la collaborazione di Sun, SAP e la sponsorizzazione della BPMI (Business Process Management Initiative), un'organizzazione no-profit che incoraggia le compagnie e le industrie a sviluppare servizi che sfruttano la rete Internet; più in particolare, la BPMI cerca di incentivare lo sviluppo del *Business Process Management* attraverso la creazione di standard per il process design. Il linguaggio BPML, le cui specifiche sono libere, fornisce numerosi strumenti per la definizione logica di processi in ambito business, coprendo un gran numero di workflow patterns; supporta la descrizione dei servizi in WSDL e condivide quasi totalmente la sintassi con il BPEL4WS ed essendo un linguaggio ad alto livello il suo utilizzo necessita di parser compatibili, come ad esempio il MSXML (Microsoft XML) o lo Xerces Java Parser, entrambi software open source.

Il principale vantaggio fornito da questo linguaggio è quello di poter rappresentare processi di enterprise business di ogni complessità. Inoltre supporta XML Schema, WSDL ed Xpath ed è compatibile con il WSCI, di cui condivide quasi totalmente la sintassi. E' un meta-linguaggio e le istanze dei suoi processi vengono gestite tramite il Simple Object Access Protocol (SOAP).

BPML definisce un modello astratto per esprimere modelli generici di processi, garantendo quindi elasticità nell'affrontare problematiche. Il flusso di esecuzione viene controllato dettagliatamente, ma non viene tenuta traccia delle attività svolte.

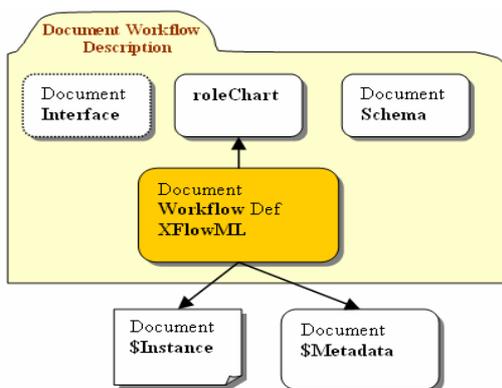
4.3.2.11 XFlow (XML-based document-centric Workflow)

XFlow appartiene al sottoinsieme dei descrittori di workflow documentali. In questo caso l'obbiettivo è quello di descrivere una sequenza di step che un particolare documento deve seguire per essere elaborato visionato e scambiato tra i vari attori coinvolti nel processo.

Per tenere traccia della storia dell'istanza di un documento (comprendendo anche gli agenti che lo hanno manipolato) e dello stato del documento in ogni tratto del suo percorso, è necessario associare al documento stesso rispettivamente un documento *Log*, che contiene la storia pregressa, ed un documento *Metadata*, che rappresenta lo stato corrente del documento. In ogni fase del

processo di elaborazione del documento quest'ultimo cambia il proprio stato, quindi il *Metadata* viene prima salvato nel *Log* e poi aggiornato (il Document Workflow engine realizza automaticamente queste due operazioni per ogni step). Per ognuno delle due componenti viene definito uno specifico linguaggio dichiarativo, utilizzando XML. Di conseguenza un documento che afferisce al Document Workflow durante tutto il proprio life-cycle è associato a 6 documenti:

Figura 4.5: Documenti che definiscono il workflow

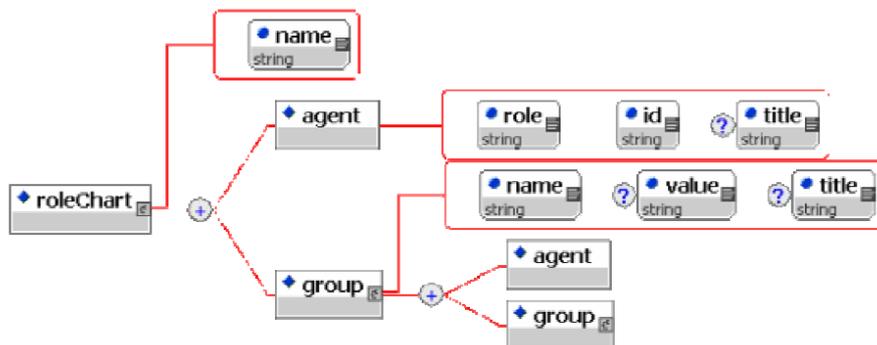


-Document Schema (XML Schema): Document schema descrive la struttura e i tipi di dati dei documenti che partecipano al flusso. Viene definito tramite XML schema.

-Document Interface (XForms, Web Service): Questo documento descrive per ogni agente che lo dovrà manovrare l'interfaccia che il documento mostra (attraverso WSDL o Web Services).

-Role Chart (Agent Role Declaration): E' un documento XML contenente la descrizione di tutti gli agenti che partecipano al workflow. Ogni attore ha un ruolo e un identificatore univoco. I ruoli sono organizzati gerarchicamente in *chart* e un agente può partecipare al workflow attraverso più ruoli e quindi può partecipare a più role chart.

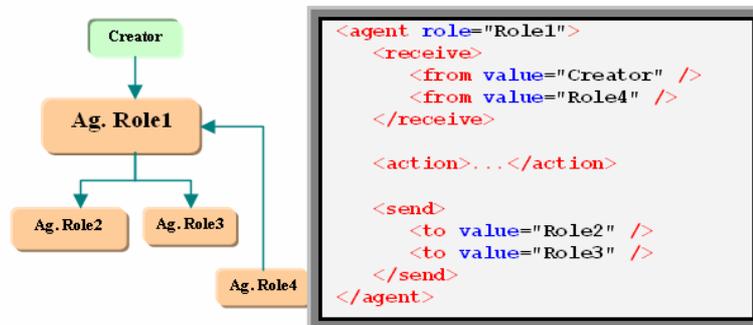
Figura 4.6: Struttura di un elemento roleChart



-XFlow: Document Workflow Definition (XflowML): Per definire un possibile linguaggio in grado di descrivere il complesso flusso documentale ci possono essere più approcci sintattici.

Si possono usare soluzioni simili ai correnti linguaggi utilizzati per descrizione di Workflow, utilizzando nozioni di nodi fork e join per descrivere il flusso. Oppure si può descrivere il flusso dal punto di vista degli agenti, ovvero per descrivere un flusso documentale basta descrivere in modo accurato tutti gli agenti e tutte le operazioni che questi possono eseguire sull'istanza del documento. L'idea degli autori di rappresentare il flusso tramite un documento XML ha permesso agli stessi di scegliere la seconda opzione, in quanto il linguaggio XML si presta meglio alla descrizione di liste gerarchiche piuttosto che di grafi. Questa scelta comporta la necessità di descrivere in modo molto dettagliato gli agenti indicando ogni possibile interazione con altri agenti.

Figura 4.7: Gerarchie nei ruoli



-il documento Metadata: Per ogni istanza di documento che partecipa ad uno specifico DW vengono fornite delle informazioni aggiuntive (metadata) che vengono memorizzate insieme ad altre informazioni con l'obiettivo di poter ricostruire la storia del documento (Log), la quale consiste in tutte le transazioni subite dal documento stesso durante il suo processing, incluse le informazioni relative agli agenti coinvolti. Il Log permette di annullare operazioni. Ogni volta che un documento è manipolato da un agente le differenze tra la versione precedente del documento vengono generate e salvate all'interno del *Metadata*. Solo una volta che il life-cycle del documento è completato e il documento viene archiviato tutti questi dati vengono cancellati. La creazione del document system Metadata avviene ad opera del Document Workflow Engine, che costituisce il supporto run-time del DW. Esso implementa gli agenti interni, il supporto alle attività degli agenti e certi moduli di sistema che gli agenti esterni utilizzano per interagire con il Document Workflow system. I campi contenuti all'interno del documento *Metadata* sono i seguenti:

-<ID>: contiene un identificatore univoco

- <Sender>: contiene il nome dell'agente mittente del documento
- <Receiver>: contiene il nome dell'agente destinatario del documento
- <Time>: contiene l'ora in cui è stato inviato il documento.
- <Date>: contiene la data in cui è stato inviato il documento.
- <Deadline>: contiene la data e l'ora entro la quale il documento deve completare il proprio percorso. Può essere utile per gestire delle politiche di priorità.
- <Duplicate>: (yes/no) indica se il documento è stato duplicato.

Il documento Log: Il documento Log è un documento XML che contiene la lista di tutte le operazioni che vengono effettuate sul documento e le firme degli agenti. Grazie a questo documento è possibile:

- monitorare semplicemente il ciclo di vita di un documento;
- verificare l'integrità e l'autenticità dei dati inseriti;
- recuperare il flusso in caso di crash;
- effettuare statistiche sui documenti.

La struttura del documento di Log è molto semplice, infatti questo viene costruito in maniera sequenziale dai vari agenti ogni volta che questi effettuano l'operazione di submit di un documento. Per ogni documento esiste un unico Log che cresce mano a mano che il documento viene manipolato lungo il suo iter. Quando un agente effettua l'operazione di submit il sistema esegue le seguenti operazioni: aggiunge al documento Log un elemento <Log> specificando nell'attributo *agentname* il nome dell'agente; in seguito inserisce all'interno dell'elemento le seguenti informazioni:

- Il *Metadata* ricevuto insieme al documento;
- il frammento di documento XML che è accessibile all'Agente;
- la digital signature relativa solo al frammento;
- le informazioni relative al tempo di invio e ai destinatari del documento.

Un agente firma solo un frammento del documento (la parte accessibile) perché l'intero documento potrebbe contenere molte altre informazioni compilate da altri agenti. Quando un documento viene duplicato vengono creati tanti Header, uno per ogni istanza di documento. Il documento Log invece

resta unico. Le informazioni temporali presenti all'interno del Log permettono di monitorare il flusso del documento ed individuare eventuali problemi come ad esempio dei colli di bottiglia.

Il documento Form: Il documento Form viene utilizzato dal sistema per fornire agli agenti umani una rappresentazione grafica del documento e permette di descrivere gli elementi grafici tramite i quali si possono modificare i campi del documento. Il documento XML viene in questo modo visualizzato come un modulo con dei campi da compilare, dei bottoni ed altri elementi grafici. I tag usati per descrivere i vari controlli grafici sono quelli usati nella tecnologia Xforms. Il documento Form non viene preso in considerazione dagli agenti software. Un documento Form deve avere come radice un elemento `<Form>`, all'interno del quale si trova una descrizione di tutti i controlli associati ai vari campi dell'istanza del documento. In questo modo si riescono a definire i vari tipi di controlli che dovranno essere visualizzati per interagire con i campi dell'istanza.

Nel documento Form sono descritti tutti gli elementi grafici del modulo; gli agenti però devono visualizzare solo alcuni di questi controlli perché nella descrizione del flusso sono espressi per ciascun agente dei vincoli sulla visibilità dei campi. Per rendere la rappresentazione grafica più chiara e piacevole è possibile assegnare dei fogli di stile al documento specificandone l'URI con un attributo `stylesheet` dell'elemento `<Form>`. Il documento Form potrebbe essere generato in maniera automatica dallo schema del documento e poi modificato da chi progetta il flusso per migliorare la visualizzazione dei documenti.

-Il documento: La classe di documenti viene definita attraverso un XML Schema, in cui si forniscono informazioni precise riguardo la struttura del documento e soprattutto sul tipo dei campi che permettono di fornire il link ai documenti associati al documento stesso. L'istanza deve avere come radice un elemento `<Instance>` con un insieme di attributi, quali:

-*Name*: identifica in modo univoco il documento;

-*Schema*: indica l'URI dello schema xml che definisce la classe di documenti XML a cui il documento appartiene;

-*Header (Metadata)*: indica l'URI del documento XML che contiene tutte le informazioni sullo stato attuale del documento.

-*Log*: indica l'URI del documento XML dove sono salvate tutte le operazioni effettuate sul documento e le firme dei vari agenti;

-*XFlow*: indica l'URI del documento xml che descrive il flusso;

-*Form*: è un attributo opzionale che indica l'URI del documento xml contenente alcune informazioni relative alla visualizzazione tramite Xforms.

Una possibile istanza di documento potrebbe essere la seguente:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<Instance
Name = "Document.xml"
Schema = "Document.xsd"
Header = "Header.xml"
Log = "Log.xml"
XFlow = "XFlow.xml"
Form = "Form.xml">

<Esempio>
<Campo1>Prova</Campo1>
<Campo2>777</Campo2>
<Campo3>>true</Campo3>
</Esempio>
```

4.3.2.12 XDOC-WFMS (XML Document Centric Workflow Management System)

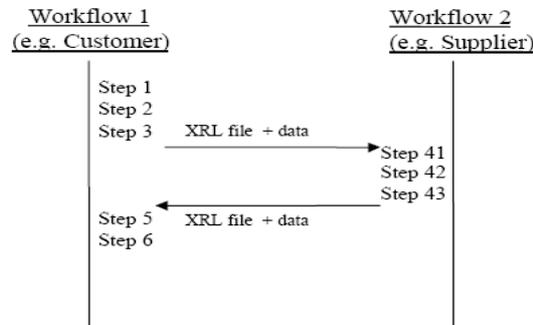
XdoC-WFMS è un linguaggio progettato per specificare, per eseguire e per controllare workflows. Anche in questo caso si tratta di un linguaggio volto a descrivere attività che prevedono di accedere e manipolare una molteplicità di documenti. Ogni attore coinvolto all'interno del processo viene definito come un *micro-agent*. Ogni micro-agente permette di accedere, aggiornare e trasferire documenti.

La particolarità di questo linguaggio è legata alla possibilità di memorizzare all'interno di un repository centrale tutte le operazioni svolte all'interno dell'esecuzione di un'istanza. Ad ogni step viene associato un identificatore dell'istanza, un identificatore dell'agente coinvolto, un identificatore dell'istante in cui l'operazione è stata eseguita. Questo repository può poi essere interrogato al termine dell'esecuzione per ottenere un filtraggio sulle azioni svolte.

4.3.2.13 XLR (eXchangeable Routing Language)

Un altro sottoinsieme dei linguaggi descrittivi di Workflow è rappresentato dai linguaggi dinamici e adattativi. Si tratta dei linguaggi che non prevedono a priori delle regole definite per lo svolgimento di un workflow, il processo infatti evolve in funzione delle scelte prese durante il life-cycle stesso. Ogni volta che viene attivato un nuovo workflow (ovvero un sotto-workflow che si genera a partire dal workflow originario) non vengono inviati solo i dati associati all'esecuzione svolta, ma anche

l'XRL file che viene aggiornato così ad ogni passo (viene tenuta traccia di ciò che è fatto e in base a questo posso modificare il percorso futuro del processo) quando lo restituisco al main WorkFlow vengono ritornati i dati e il file XRL aggiornato.



Infatti utilizzare un WorkFlow adattativo necessita di modificare progressivamente l'XRL file di partenza. All'interno del file è presente l'elemento *state* che tiene memoria dello stato in cui si trova l'istanza di WorkFlow.

Nonostante questa attenzione per l'andamento dell'esecuzione dell'istanza non viene tenuta traccia di tutto lo stato passato ma solo dello stato attuale e, nel caso in cui si sia realizzato un sub-WorkFlow, viene tenuta traccia di che punto ha dato il via al set di operazioni per poi ritornarvi alla fine dell'esecuzione.

La potenzialità è legata alla possibilità di modificare a livello di istanza il file che descrive il WorkFlow fornendo quindi un controllo diretto a livello di flusso e non solo di classi definite in modo statico.

4.3.2.14 XLANG (XML-based Language)

XLANG è un'estensione del linguaggio WSDL, Web Service Definition Language. Esso fornisce sia il modello di una collection di servizi utilizzati per svolgere un compito, nonché per definire le interfacce che i servizi stessi mostrano agli attori del processo stesso. XLANG, come BPML, sono stati progettati espressamente fondandosi su una teoria di calcolo esplicito. Le azioni svolte dagli attori coinvolti sono le entità principali che definiscono un processo descritto tramite XLANG. Alle quattro tipologie di azione contemplate per WSDL (richiesta/risposta, sollecitare risposta e notifica,) si aggiungo altre due tipologie: timeout (termine e durata di un processo) e lancio delle eccezioni. Infatti il timeout non viene considerato come una proprietà di un azione specifica, ma un azione

propria che viene considerata come l'evento di invio dell'evento timeout.

Un processo viene definito all'interno della definizione dei servizi interpellati, e la definizione del processo specifica le proprietà dei servizi interpellati. Le operazioni in ingresso ed in uscita di un servizio rappresentano le interazioni di quest'ultimo con altri servizi, quindi una sequenza di operazioni equivale ad un'orchestrazione di una serie di servizi, all'interno del quale è sempre ben definito l'inizio e la conclusione della specifica interazione.

Possono essere attivati contemporaneamente più servizi, e lo stesso servizio può inizializzare diverse istanze del processo, sulla base della richiesta di più clients. Questa inizializzazione può avvenire in modo esplicito (attraverso un'azione diretta di richiesta) o in modo implicito, se l'accesso al servizio è contemplato all'interno della sequenza di operazioni che viene eseguita.

XLANG specifica anche la nozione di correlazione tra i messaggi. Un istanza di un servizio tipicamente intrattiene una o più conversazioni con altre istanze di servizio rappresentanti gli altri partecipanti (user, partner o altri sistemi) coinvolti nelle interazioni. E' possibile che un sistema o un partner non siano configurati con un protocollo di comunicazione che tiene traccia della conversazione, quindi non sarebbe possibile risalire alle informazioni correlate ad una particolare istanza. Tuttavia grazie alla correlazione tra messaggi, all'interno dello stesso servizio vengono definiti dei gruppi di correlazione. XLANG implementa la correlazione dei messaggi fornendo un meccanismo molto generico per specificare i gruppi correlati di operazioni all'interno di un'istanza del servizio. Un insieme di correlazione può essere specificato come un insieme di proprietà condivise da tutti i messaggi nel gruppo correlati. Il "gruppo di correlazione" rappresenta l'unico modo per tenere traccia del ciclo di vita di un'istanza:

- Gruppi di correlazione vengono inizializzati e terminati nell'ambito della loro istanza di servizio.
- Gruppi di correlazione possono essere definiti tra diverse istanze.
- La creazione di un gruppo di correlazione è provocata da una operazione di contrassegno speciale.
- Il life-cycle del gruppo correlazione è determinato dalla durata del suo contesto o servizio.

Come BPML, XLANG si basa su un flusso di dati XML, che viene alimentato dal flusso di messaggi scambiati tra i servizi e supporta le decisioni di controllo sul flusso stesso. XLANG presuppone che i tipi di documento XML siano specificati con XML Schema.

XLANG permette di fornire un controllo diretto sul flusso dell'istanza, tramite elementi base simili a quelli forniti da BPML: <sequence>, <switch>, e <all>. Inoltre fornisce il supporto per il looping

con l'elemento <while>, che specifica che un frammento della definizione del processo viene eseguito fino a quando una determinata condizione non è più verificata. Come BPML, anche XLANG fornisce una semantica per specificare le eccezioni e gestori di eccezioni, con il costrutto <pick>. Inoltre spesso esiste un problema quando si specificano gli indirizzi delle porte in uscita: raramente è possibile conoscere in anticipo l'indirizzo del messaggio in uscita. XLANG permette di specificare che l'indirizzo associato a queste porte in uscita verrà fornito dinamicamente.

4.4 Conclusioni sulla ricerca bibliografica

Dall'indagine svolta risulta evidente come, nonostante la grandissima varietà di linguaggi descrittivi di workflow esistenti, e la forte adattabilità degli stessi a numerosissimi campi di applicazione, non esistano soluzioni in grado di soddisfare totalmente le necessità richieste per il documento in grado di tracciare un workflow clinico. In particolare, sebbene diversi linguaggi siano stati definiti appositamente per la gestione dei complessi workflow che si possono generare in ambito sanitario, si tratta spesso di linguaggi adatti a descrivere in modo appropriato solamente le componenti amministrative e gestionali di un sistema sanitario. A seguito di ciò, e delle ridotte complicità legate al particolare svolgimento di tali attività, il tracciamento della particolare istanza di workflow non comporta sensibili vantaggi, e per questo motivo viene trascurata. La potenzialità comunque di memorizzare gli step eseguiti durante un particolare business è comunque presente, affidando il compito a web services esterni. Il secondo limite, osservato nella maggior parte dei linguaggi analizzati, consiste nell'impossibilità di gestire workflow le cui condizioni di sequenzialità variano in fase di esecuzione. In particolare solo la sotto-categoria dei linguaggi adattativi e dinamici consente di modificare a livello di istanza le condizioni che permettono al workflow di evolvere attraverso un certo pattern di step. In questo caso si osserva che il tradizionale documento che descrive il workflow prestabilito per la particolare istanza diventa parte stessa del flusso di dati e informazioni scambiati tra agenti e servizi, e viene manipolato dagli stessi per modificarne il contenuto in funzione delle condizioni attuali del workflow e di altri parametri esterni di interesse (come per esempio il livello di utilizzo delle linee di comunicazione tra attori che partecipano al workflow). L'assenza di tale proprietà di adattabilità è quindi una condizione fortemente limitante per quanto riguarda i workflow clinici, per i quali non può sempre essere definita a priori la sequenza di step che verrà seguita nella particolare istanza.

Parallelamente l'analisi bibliografica ha permesso di ricavare numerosi spunti di riflessione, che

hanno poi condizionato la progettazione del documento che permette la tracciatura del workflow clinico. Innanzitutto è importante osservare come sia utile avere a disposizione dei parametri in grado di poter permettere una valutazione dell'efficienza del workflow, come per esempio il tempo impiegato per portare a compimento un determinato business, il numero di attori coinvolti o il numero di step compiuti. Per garantire la responsabilità dei vari attori coinvolti ogni step deve essere associato alla firma dell'attore che ha manipolato dati e informazioni in quella determinata fase. Inoltre è indispensabile poter garantire la possibilità, in caso del verificarsi di errori o eccezioni, di poter ritornare ad uno stato consistente del workflow, evitando l'interruzione del workflow incompleto.

Sulla base delle seguenti considerazioni ed altre legate più strettamente al campo di applicazione specifico è stato definito il profilo **XDW** (Cross-Enterprise Document Workflow).

5. Profilo XDW (Cross-Enterprise Document Workflow)

5.1 Introduzione

Il profilo Cross-Enterprise Document Workflow (XDW) permette di gestire lo stato dei documenti coinvolti in un workflow. Infatti, il metadato “availabilityStatus” consente di specificare due stati (“approved” o “deprecated”) ma non di gestire i differenti steps del workflow clinico. Questo profilo permette di risolvere il problema mediante l'utilizzo di un documento specifico, il Workflow Document, dove vengono tracciati i differenti steps di tutti i documenti correlati allo stesso evento clinico, memorizzando le proprietà specifiche (“chi” e “quando”) connesse al cambiamento di stato legato all'azione di un professionista coinvolto nel processo sanitario.

I partecipanti ad ambienti multi-organizzativi possono così tracciare gli step correlati a workflow paziente-centrici, memorizzando come essi hanno gestito le proprie attività. XDW è progettato sullo scambio di documenti clinici prodotti dagli altri profili IHE, come per esempio XDS, definendo il processo di associazione di un documento ad un workflow paziente-specifico. XDW definisce una infrastruttura comune e interoperabile sulla quale un vasto range di workflow possono essere definiti. XDW è progettato per supportare l'enorme complessità dei servizi sanitari, garantendo la massima flessibilità per adattarsi durante l'evoluzione del workflow stesso.

Il profilo definisce una struttura dati condivisa per tracciare il workflow, chiamata “Workflow Document”, il quale memorizza gli step passati del workflow, mantenendo in modo consistente i riferimenti alle informazioni ed ai dati clinici forniti in input e output ad ogni step specifico. Queste informazioni scambiate relative allo stato del workflow permettono ai vari partecipanti del sistema di: essere a conoscenza della storia di ognuno dei workflow associati al paziente, accedere allo stato corrente del workflow e di mantenere il coordinamento dello stesso aggiornando il documento con i nuovi steps che vengono eseguiti.

XDW è pensato per offrire una struttura comune per garantire l'interoperabilità in modo workflow-indipendente con questi obiettivi:

-fornire una piattaforma sulla quale un vasto range di workflow specifici possano essere definiti,

attraverso una specializzazione del contenuto, attraverso minimi oneri di implementazione e specificazione. Questi verranno chiamati XDW-Based Content Profiles.

-essere utilizzabile da molti profili sia clinici che non-clinici, impedendo approcci competitivi che riducano l'interoperabilità dei sistemi.

-Aumentare la consistenza dell'interoperabilità tra workflow, e permettere lo sviluppo di applicazioni interoperabili per la gestione dei workflow dove sia minimizzata la personalizzazione workflow-specifica.

-Facilitare la definizione e l'evoluzione dei workflow per e da professionisti della sanità, minimizzando le conoscenze tecniche richieste.

-Facilitare l'integrazione di workflow multi-organizzativi mediante la varietà di sistemi di gestione di workflow per le organizzazioni.

In modo più specifico XDW supporta workflow che:

-sono paziente-centrici;

-hanno una specificazione clinica o operativa che è definita al di fuori di XDW. La definizione del business del workflow è esterna a XDW, e deve essere conosciuta solo dai sistemi partecipanti al workflow stesso, e non dai sistemi che costituiscono l'infrastruttura;

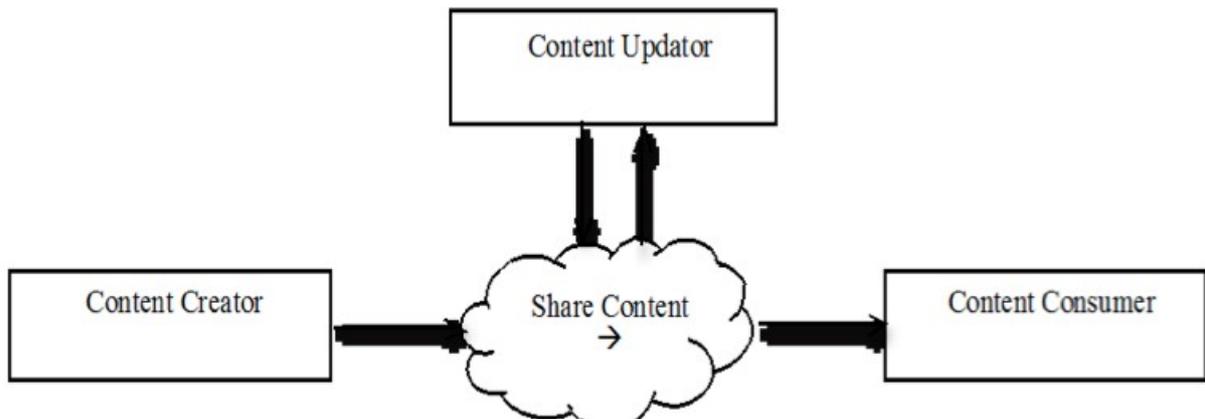
-necessitano di essere tracciati in ambienti distribuiti. Quindi i partecipanti del workflow necessitano di sapere i più recenti step eseguiti, e quale è la storia degli step più remoti. Solo informazioni relative alla storia più lontana del workflow sono scambiate, quindi non si rende necessaria per XDW la presenza di un sistema di gestione del workflow centralizzato;

-mantengono una rigida separazione tra XDW, che traccia il workflow, e le informazioni cliniche e amministrative, che sono gestite attraverso puntatori verso i documenti condivisi come definito dallo standard XDS.

5.2 Attori e transazioni

Il profilo XDW è basato su tre attori il Content Creator, il Content Consumer ed il Content Updater. Il contenuto del documento (il “Content”) viene creato dal primo attore, mentre viene utilizzato dal Consumer. Lo scambio e la condivisione del contenuto ad opera di un attore è permessa dall'utilizzo dagli specifici profili IHE. Il Content Updater può agire su un esistente Workflow Document consumandolo e sostituendolo con una nuova versione. Un attore XDW Content Creator deve essere in grado di creare un nuovo workflow (cioè creare un nuovo XDW Workflow Document e posizionarlo in una nuova cartella).

Figura 5.1: Diagramma degli attori del profilo XDW



5.2.1 Raggruppamento

Un XDW Content Creator o Content Consumer deve essere raggruppato con appropriati attori del profilo XDW, in modo da poter scambiare tra soggetti gli XDW Workflow Document. Il metadato inviato mediante la condivisione o lo scambio di documenti ha una relazione o dipendenza specifica (chiamata “binding” o “legame”) con il contenuto del documento clinico.

Un XDW Content Creator deve essere raggruppato con:

- Un XDS.b Document Source che permette sia il Document Replacement Option e la Folder Management Option.

Un XDW Content Consumer deve essere raggruppato con:

- Un XDS.b Document Consumer

Un XDW Content Updater deve essere raggruppato con:

-Un XDS.b Document Consumer

-Un XDS.b Document Source

5.3 XDW Document

5.3.1 Introduzione

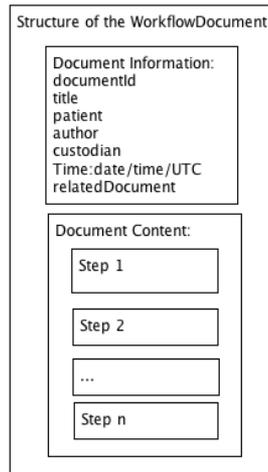
L'implementazione del XDW Content Profile è basata sulla gestione di un documento: l'XDW Workflow Document. In questa sezione si vuole proporre un'indagine comparativa tra due possibili strutturazioni del Workflow Document. Nel primo caso, che utilizzerà come standard di riferimento HL7 CDA, ad ogni azione è associato uno step all'interno del Workflow Document, permettendo una strutturazione semplice per ogni tipo di utilizzo. Nel secondo caso, invece, la struttura del Workflow Document è basata sullo standard Human Task, il quale permette di strutturare il documento in una serie di task, all'interno delle quali possono essere tracciati più stati diversi della stessa task mediante l'utilizzo del costrutto Task Event. Di seguito nella sezione verranno presentati i due diversi approcci, con particolare riguardo all'analisi di un semplice caso d'uso (e-Referral). In questo modo sarà possibile un confronto tra i limiti e il potenziale di ognuno dei due approcci.

5.3.2 Prima versione dell'XDW Workflow Document (CDA-based)

5.3.2.1 Struttura dell'XDW Workflow Document (CDA-based)

Il Workflow Document è usato per tracciare gli steps di un workflow che descrive la gestione dei documenti correlati ad un workflow clinico ed i suoi cambiamenti che avvengono step dopo step. L'XDW Workflow Document è un documento strutturato, basato sullo standard CDA, e contenente le informazioni condivise relative al workflow tracciato e più specificatamente le informazioni relative ad ogni step già realizzatosi all'interno del workflow stesso. Come mostrato in figura 5.2, l'XDW Workflow Document è strutturato in due parti: la prima parte contiene le informazioni generali relative al documento, la seconda parte è strutturata in sottosezioni che descrivono i differenti steps del Workflow.

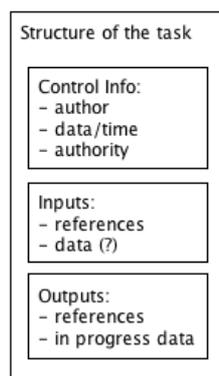
Figura 5.2: Struttura dell'XDW Workflow Document basato su CDA



Ogni azione o cambiamento di stato di un documento deve essere descritto tramite uno step tracciato all'interno del documento. Ogni step possiede la medesima struttura, basata su tre sezioni foglia che contengono rispettivamente: le informazioni di controllo relative allo step, il riferimento ai documenti forniti in ingresso allo step, i riferimenti ai documenti prodotti durante lo step:

- Control Information:** descrive lo step specifico e l'autore dello step
- Input:** contiene le informazioni relative ai documenti forniti in ingresso allo step e necessarie per accedere agli stessi documenti
- Output:** contiene le informazioni relative ai documenti prodotti durante lo step e necessarie per accedere agli stessi documenti

Figura 5.3 Struttura della task



In ogni istante, se un partecipante al Workflow sceglie di produrre un avanzamento del Workflow per uno specifico paziente, deve necessariamente creare uno o più nuovi steps, aggiungendo un elemento task all'istanza più recente del documento di Workflow. Questa nuova versione del

documento viene pubblicata nel repository come un “replacement” dell'istanza precedente. Tutte le versioni “deprecated” del Workflow Document, come anche la versione più recente dello stesso documento, sono contenute all'interno della stessa cartella, quindi lo uniqueId della cartella costituisce un riferimento stabile ad una particolare istanza del workflow, mentre lo uniqueId associato al documento è differente per ogni versione del documento stesso. Non è necessario che tutti i documenti i cui riferimenti sono contenuti all'interno del workflow document siano posizionati all'interno della stessa cartella. La necessità e la frequenza con cui si effettuano ricerche all'interno della lista dei documenti associati ad un particolare workflow è connessa alla particolare specificazione del workflow stesso. Se si rende necessario un “workflow back-link”, cioè la possibilità di risalire partendo da un documento clinico al Workflow Document al quale esso appartiene, può essere realizzato, all'interno del contesto XDS, attraverso diversi approcci:

-Query per i documenti di workflow basata sul contenuto degli attributi associati a particolari metadati e filtrando l'insieme di documenti così ottenuti ottenendo i soli documenti non-deprecated. All'interno di ogni documento così ottenuto vengono poi selezionati gli Id dei documenti i cui riferimenti sono presenti all'interno dei vari steps.

-Se tutti i partecipanti all'interno del Workflow sono parte dello stesso Affinity Domain XDS si può richiedere che tutti i documenti contenuti come riferimento all'interno dello specifico Workflow Document siano posizionati all'interno della stessa cartella che contiene il Workflow Document.

5.3.2.2 Use case: Workflow di Visita Specialistica (e-Referral)

-Descrizione del caso d'uso: Questo caso d'uso si focalizza sulla gestione del Workflow che si genera quando un medico indirizza un paziente ad un altro ente sanitario per eseguire un esame specialistico (e-Referral). In questo caso è necessario tracciare il workflow associato al documento e-Referral. Inoltre sono coinvolti diversi attori e diversi sistemi, che possono produrre oggetti e documenti che vengono scambiati all'interno del workflow stesso per migliorare il processo clinico descritto e facilitare le decisioni dei professionisti coinvolti. Il workflow è lineare con chiari e definiti steps, tuttavia vi è la possibilità di associare ad ogni step diversi documenti in ingresso.

-Flusso dei processi descritti: Un paziente si reca dal proprio medico di base per avere un consulto relativo ad uno specifico problema clinico. Il medico esamina il paziente ed eventuali reports clinici relativi a questo problema. Dopo la visita il medico prescrive un consulto con uno specialista,

creando un documento e-Referral come risultato dello studio condotto. Da questo momento l'e-Referral è disponibile per un vasto range di providers ed il paziente ha la possibilità di rivolgersi alla struttura sanitaria che preferisce per eseguire la visita. La struttura scelta verifica che lo stato dell'e-Referral sia disponibile e prenota la visita. Lo specialista consulta il documento e-Referral ed esamina il paziente e le eventuali informazioni riportate relative al caso clinico in esame (come per esempio l'eReferral stesso e il riferimento ai reports che il medico di base ha considerato importanti per caratterizzare il problema clinico). Dopo aver eseguito il proprio esame lo specialista genera un report clinico dell'esame, il riferimento allo stesso viene riportato all'interno del Workflow Document che viene così aggiornato con il nuovo step processato. Questo report può essere notificato al medico di base e il workflow associato alla richiesta specialistica viene così completato. In ogni momento il medico di base può accedere ai documenti associati al caso clinico e prodotti durante l'intero workflow. Questo viene eseguito tramite un semplice query and retrieve realizzato dal software di cartella clinica del medico di base verso il Registry ed il Repository. E' possibile anche monitorare il progresso del workflow mediante l'utilizzo di un sistema di notifiche definito tramite il profilo Document Metadata Subscription (DSUB) o il profilo Notification of Document Availability (NAV). Questo caso d'uso è stato molto semplificato rispetto al caso reale tuttavia, attraverso l'utilizzo di altri profili IHE, è possibile aumentarne la ricchezza e la complessità. Questo è in linea con la principale proprietà di XDW ovvero l'assoluta interoperabilità con gli altri profili.

Di seguito viene presentata nel dettaglio la cronologia degli step processati in relazione all'XDW Workflow Document:

-Un paziente richiede un consulto con il proprio medico di base per uno specifico problema clinico
-Il dottore esamina il paziente e i reports correlati al problema clinico. Alla fine della visita produce un report e un e-Referral (1), all'interno del quale viene ordinato un esame specialistico. Per fornire una panoramica completa del caso clinico il medico di base decide di associare oltre alla richiesta anche altri esami informativi per lo specialista che eseguirà l'esame, che verranno tracciati nella sezione di input del XDW Workflow Document (2). In questo istante il Workflow Document è costituito da una sola task che descrive il primo step: produzione dell'e-Referral. Questa task è caratterizzata da tre elementi: una sezione di "input" dove il medico di base traccia i riferimenti al report della visita, ed ad un insieme di reports che contengono informazioni utili a definire il motivo della richiesta di visita specialistica; una sezione "control info" dove sono memorizzate le informazioni relative allo step; una sezione di "output" dove tutti i documenti prodotti in relazione

allo step vengono tracciati (documento e-Referral) e possibili suggerimenti per gli steps successivi.

-Il software del medico di base produce due oggetti: l'eReferral e il Workflow Document per tracciare il workflow clinico dell'e-Referral stesso. Questi due elementi sono referenziati utilizzando il meccanismo delle cartelle XDS. La scelta dell' XDS Folder mechanism non è vincolante ma suggerita per raccogliere documenti correlati allo stesso caso clinico e tracciati mediante lo stesso Workflow Document.

-Il Content Creator produce una cartella di documenti per raggruppare tutti i documenti correlati allo stesso Workflow Document.

-Il Content Creator produce l'e-Referral e il Workflow Document e li invia al document Repository (3,4)

-Da questo momento l'e-Referral è disponibile per un vasto range di provider della sanità

-Il paziente si reca dal provider di propria scelta per prenotare l'esame

-L'ufficio per le prenotazioni richiede la tessera sanitaria al paziente per il Retrieve del Workflow document e l'e-Referral.

-Il Content Consumer interroga e ritira il Workflow Document e l'e-Referral.

-L'ufficio prenota la visita. Una volta prenotata la visita il Workflow Document viene sostituito con la versione aggiornata che contiene un nuovo step che permette di definire il nuovo stato dell'e-Referral che passa dallo stato "ordinato" allo stato "schedulato".

-Il Content Updater sostituisce il vecchio Workflow Document con la versione aggiornata

-Quando il paziente è ammesso in ospedale, l'ufficio prenotazioni consulta l'e-Referral e il Workflow Document associato e verifica lo stato della prenotazione stessa.

-Il Content Consumer interroga e ritira il documento di Workflow e l'e-Referral.

-Se il documento e-Referral si trova nello step corretto e l'esame può essere eseguito l'ufficio di prenotazione sostituisce il Workflow Document con la versione aggiornata che contiene tutte le informazioni dei documenti precedenti e una nuova task che descrive lo step corrente. In questo istante l'e-Referral passa dallo stato "schedulato" allo step "in-progress" e l'esame può essere eseguito

-Il Content Updater sostituisce il vecchio Workflow Document con la versione aggiornata.

-L'esame viene eseguito ed il medico specialista analizza tutti i documenti di interesse correlati al workflow. Una volta che l'esame viene effettuato il medico produce un report dell'esame stesso e Workflow Document viene aggiornato con un nuovo step, passando dallo stato "in-progress" allo stato "completato".

-Il Content Consumer interroga e ritira il Workflow Document, l'e-Referral e tutti i

documenti di interesse.

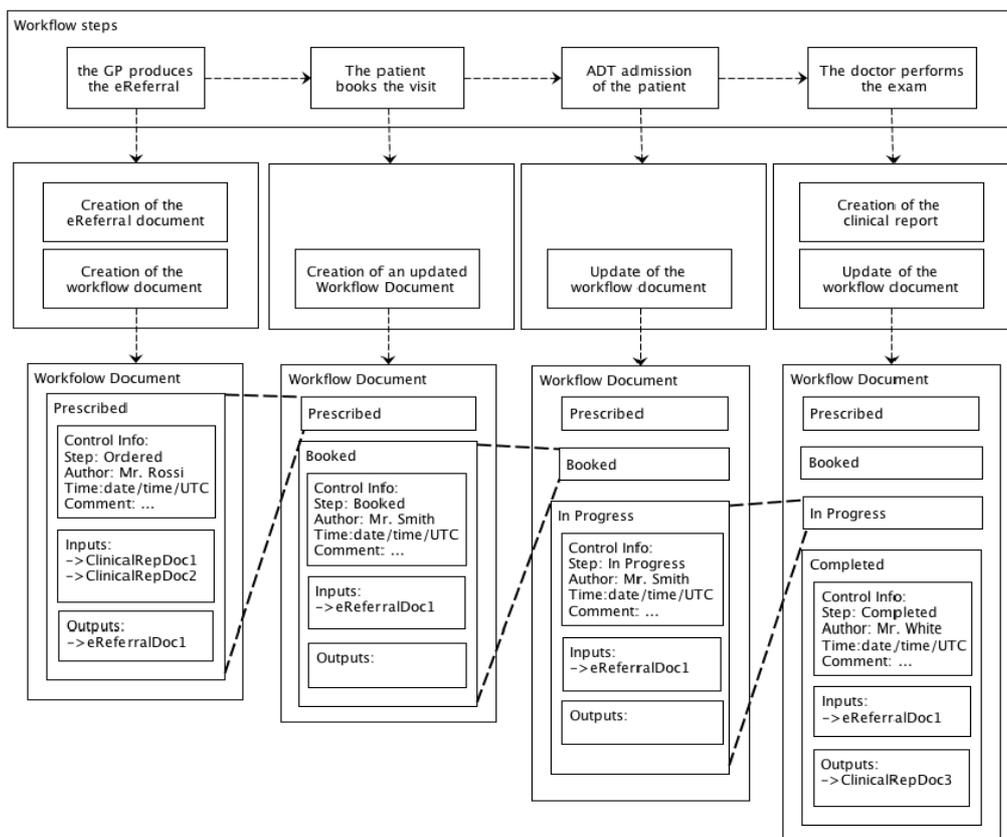
-Il Content Updater sostituisce il vecchio Workflow Document con la versione aggiornata

-Il Content Creator produce i report della visita.

In ogni istante, durante questo processo, se il paziente richiede un ulteriore consulto al proprio medico di base quest'ultimo può consultare il Workflow Document e tutti i documenti correlati allo stesso, visionando lo stato di avanzamento del Workflow stesso. Questo è permesso da una semplice interrogazione del registry. E' possibile anche gestire un sistema di sottoscrizione e di notifica che permetta di comunicare in tempo reale lo stato di avanzamento del workflow, attraverso l'utilizzo dei profili (Document Metadata Subscription) DSUB e Notification of Document Availability (NAV).

-Diagramma di composizione della struttura del XDW Workflow Document: il diagramma seguente (fig. 5.6) rappresenta la progressiva creazione del Workflow Document durante lo svolgimento degli steps del workflow nel caso d'uso in esame. Per ogni step del Workflow il diagramma rappresenta una colonna all'interno della quale vi è un blocco che contiene le azioni tecniche associate allo step e la struttura e il contenuto dell'XDW Workflow Document in questo istante.

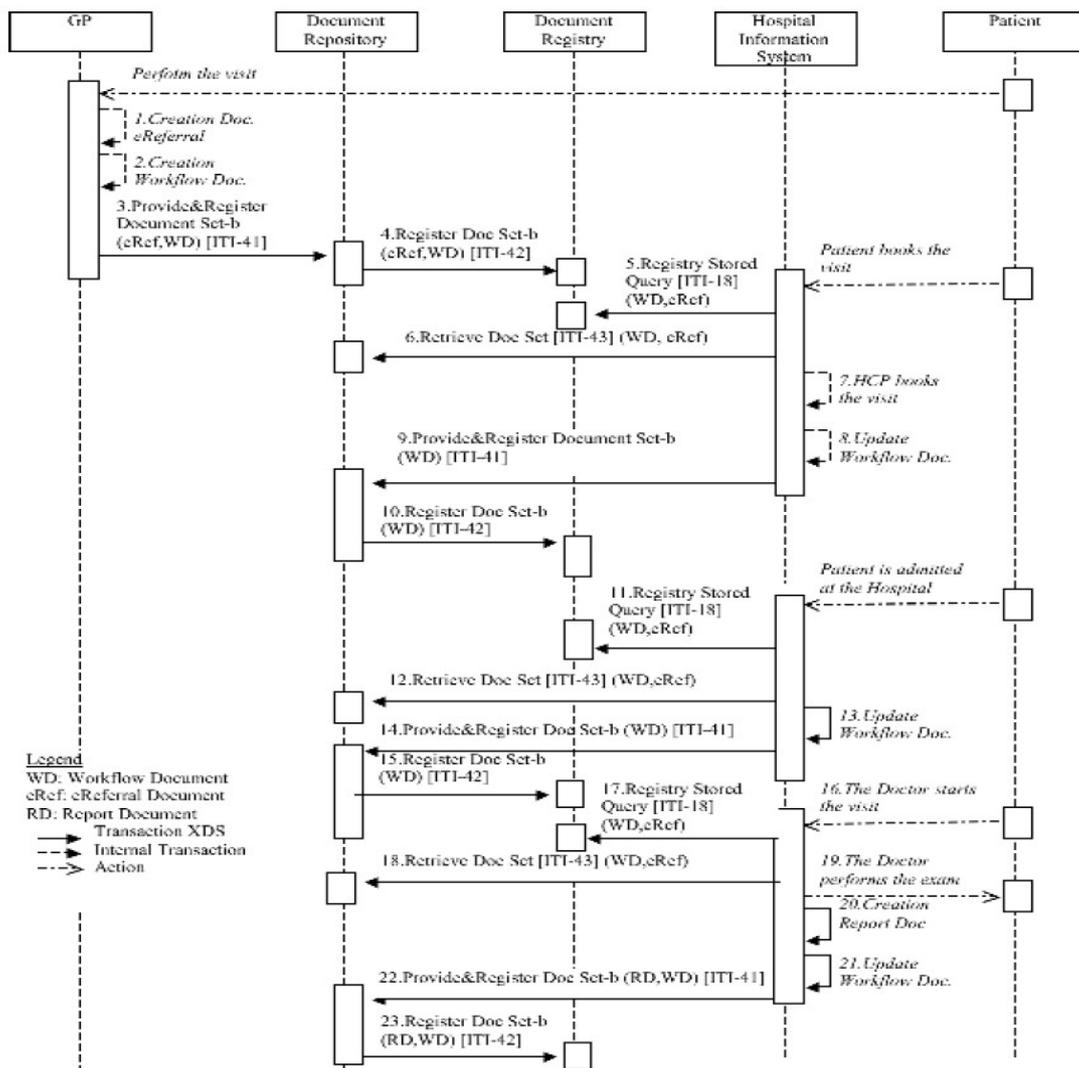
Figura 5.6: Diagramma di composizione della struttura del Workflow Document CDA-based. Caso d'uso e-Referral



-Diagramma delle azioni: il diagramma seguente (figura 5.7), rappresenta la sequenza di dati scambiati tra gli attori del sistema coinvolti nel caso d'uso dell'e-Referral. Questo è rappresentato da un esempio di applicazione di XDS. In questo caso il Content Creator è raggruppato con il Document Source XDS ed il Content Consumer è raggruppato con il Document Consumer XDS. Lo scambio di dati e informazioni tra questi due attori avviene tramite l'utilizzo delle transazioni XDS.

Nel diagramma sia il medico di base che lo specialista rappresentano un Content Creator e un Content Consumer incorporato.

Figura 5.7: Grafico del flusso di processi nel profilo XDW, caso d'uso eReferral



5.3.3 Seconda versione dell'XDW Workflow Document (Human Task-based)

La definizione di un Workflow è ottenuta tramite un set di tasks e regole logiche o cliniche. Ogni definizione di task descrive un'attività o un gruppo di attività che devono essere realizzate dal proprietario della task stessa. Le regole così definite assicurano che i vari partecipanti al Workflow operino congiuntamente per avanzare all'interno delle tasks e muovendosi da una task all'altra in modo consistente.

Viene definita un'architettura del sistema a quattro strati. In questa architettura del workflow:

-Il primo layer supporta lo scambio o la condivisione di documenti. Questi fondamenti di interoperabilità sono garantiti dal set di profili IHE già esistenti (come per esempio XDS, XDM, XDR).

-Il secondo layer definisce una generica struttura dati chiamata Workflow Document condivisa tra i partecipanti al workflow utilizzando il primo layer. Nello stesso modo i documenti forniti in input e quelli prodotti in output ad una task del workflow sono gestiti dal profilo XDW sono scambiati utilizzando lo stesso primo layer di questa architettura.

-Il terzo layer introduce la definizione semantica del workflow, che può essere compreso ed eseguito dai sistemi e dalle applicazioni partecipanti. La realizzazione di workflow specifici permette ai partecipanti di condividere una conoscenza comune associata alle specifiche tasks, le dipendenze esistenti tra le diverse tasks e un certo numero di regole che controllano l'esecuzione del workflow stesso. I dettagli per l'esecuzione sono trasmessi attraverso l'XDW Workflow Document definito dal secondo layer dell'architettura. Le specifiche per la definizione del terzo layer sono invece distinte dal profilo XDW e sono generalmente gestite in modo migliore attraverso l'utilizzo del linguaggio naturale.

-Il quarto layer di questa architettura contiene le applicazioni eseguite dai sistemi partecipanti al workflow. Queste applicazioni permettono di fornire il collegamento tra il workflow gestito da XDW ed il workflow gestito localmente. Molti dei dettagli dei workflows locali gestiti da ogni specifica applicazione possono essere nascosti e incapsulati in singole tasks più complesse, le quali possono essere esposte attraverso XDW, in modo tale da poter tenere nascosti certi dettagli di non utilità. La definizione dei Workflows del terzo layer deve contenere quindi solo tasks ad alta granularità, le quali richiedono la coordinazione attraverso legami tra le organizzazioni coinvolte.

5.3.3.1 Struttura dell'XDW Workflow Document (Human Task-based)

Come detto nelle sezioni precedenti attraverso la creazione del Workflow Document il profilo XDW permette di tracciare l'esecuzione di un workflow paziente-specifico, organizzando la struttura in Task ed in TaskEvents.

-Una **task** descrive un'attività, o un gruppo di attività, che devono essere eseguite o sono state eseguite. Ad ogni task sono associati degli attributi:

- il tipo di task;

- il possessore della task;

- lo stato corrente della task;

- i riferimenti a documenti utilizzati come input o prodotti come output della task;

- la storia di passati task Events per la specifica task, che permettono di documentare il progredire degli step della task fino al momento attuale.

Quando una persona o un'organizzazione viene assegnata come possessore della task, essa viene posta in esecuzione (passando dallo stato “created” o “ready” allo stato “in-progress”). Quando le attività attese sono completate con successo la task passa in uno stato “completed”, in caso contrario in uno stato “failed”.

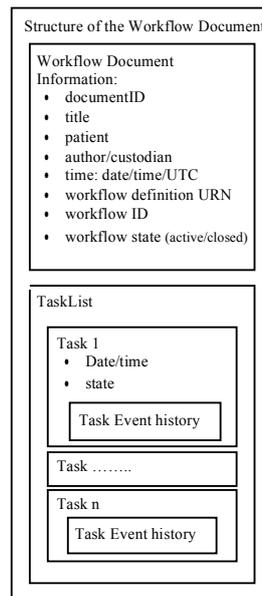
-un **Task Event** rappresenta il possibile cambiamento di stato di una task. La task Event List è l'insieme dei possibili task Events per una determinata task.

Come mostrato in figura 5.9 l'XDW Workflow Document è strutturato in due parti:

- La prima parte contiene le informazioni generali associate al documento di Workflow

- La seconda parte contiene l'insieme delle differenti tasks completate o non completate del workflow, e per ogni task contiene l'insieme die Task Event che tracciano il progresso della task stessa. In particolare la sintassi utilizzata per definire Task e Task Event è presa dalle specifiche offerte dallo standard OASIS Human Task.

Figura 5.9 Struttura del Workflow Document



Task e Task Event includono al loro interno i riferimenti a documenti clinici ed amministrativi forniti in input o prodotti come output agli stessi:

-L'attributo **Input** contiene i riferimenti ai documenti rilevanti per i partecipanti al workflow che performano la task. Può anche contenere il riferimento al Workflow “padre” che ha generato l'attuale Workflow “figlio”;

-L'attributo **Output** contiene i riferimenti ai documenti che sono stati prodotti come risultato di una task eseguita. Può anche contenere il riferimento al Workflow figlio che si è generato durante lo svolgimento della task.

In ogni istante se un partecipante sceglie di aggiornare il Workflow di un paziente, egli deve creare necessariamente una nuova task, o aggiornare una precedente task con un nuovo task Event. Ogni aggiornamento del Workflow Document comporta la creazione di un nuovo Workflow Document che viene memorizzato come una nuova istanza attiva che sostituisce la versione precedente, la quale rimane comunque disponibile nel repository acquisendo lo stato “deprecated”.

Quando un nuovo documento di Workflow è creato, esso viene posizionato all'interno di una Workflow Context Folder. Tutte le versioni successive dello stesso documento sono contenute nella stessa cartella, quindi il Workflow Context Folder UniqueId costituisce un riferimento stabile ad una particolare istanza di Workflow, mentre il Workflow Document UniqueId è diverso per ogni versione del Workflow Document stesso. Nel caso in cui si stia operando in un contesto XDS è sempre possibile posizionare all'interno della stessa cartella Workflow Context Folder anche tutti i documenti clinici ed amministrativi contenuti come riferimento nel Workflow Document. Questo

approccio non è applicabile al contesto definito da XCA, in cui i documenti possono appartenere a community distinte.

5.3.3.2 Use case: Workflow di Visita Specialistica (e-Referral)

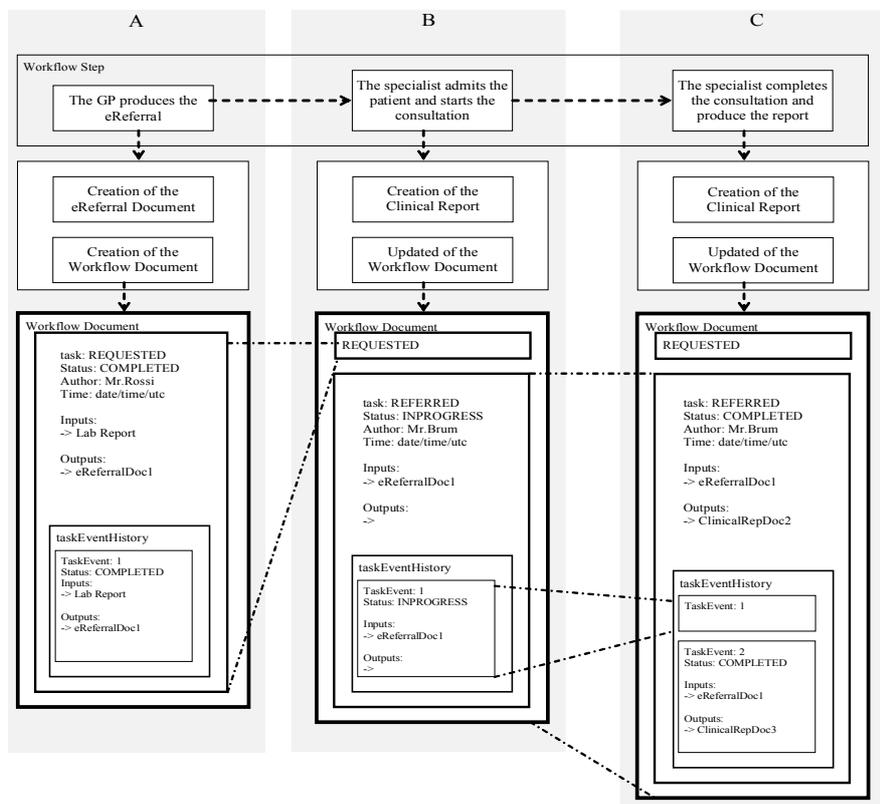
Il workflow associato alla prescrizione di consulto specialistico (e-Referral) si costituisce di tre distinti step:

- A. Il medico di base indirizza il proprio paziente verso un altro provider sanitario per un consulto specialistico
- B. Lo specialista inizia il proprio consulto che può prevedere più visite
- C. Lo specialista conclude il proprio consulto producendo un report.

Ogni step può essere descritto dal punto di vista clinico o dal punto di vista puramente tecnico.

-Flusso dei processi descritti: Di seguito viene presentata in modo dettagliato la sequenza cronologica degli steps eseguiti, con particolare attenzione alle modalità di creazione ed aggiornamento del Workflow Document (figura 5.10).

Figura 5.10: Diagramma di composizione del Workflow Document



-A. Durante questa task il medico di base esamina il paziente e controlla i dati clinici più recenti associati allo stesso. Il medico decide di indirizzare il paziente verso uno specialista, creando un documento e-Referral e associando alla task il riferimento di tutti i dati clinici ed i report utili a giustificare la richiesta. Il software del medico di base, in qualità di Content Creator, produce sia il documento e-Referral che un Workflow Document che permette di tracciare il workflow clinico dell'e-Referral stesso. In questo momento il Workflow Document contiene un'unica task ("requested") con queste caratteristiche:

- stato "completed", che permette di indicare che la task è completata e quindi non potranno essere definiti ulteriori task event.

- come input alla task sono forniti tutti i documenti clinici ed i report di interesse utili per giustificare la richiesta del consulto specialistico.

- come output della task il riferimento al documento e-Referral prodotto

Per poter condividere e scambiare i documenti prodotti durante la task, il software del medico di base, in quanto Content Creator raggruppato con l'XDS Document Source, invia l'e-Referral Document e il Workflow Document all' XDS Document Repository.

-B. In questa task il paziente si reca dallo specialista prescelto o suggerito dal medico di base. Lo specialista consulta il documento e-Referral e il Workflow Document associato per poter comprendere la tipologia di task che è necessario generare. Lo specialista accede al documento utilizzando il proprio software personale, che rappresenta un raggruppamento tra un Content Updater e un XDS Document Consumer, per interrogare e ritirare il Workflow Document e l'e-Referral Document. Se non vi sono problemi di consistenza con la definizione del workflow presente nel Workflow Document, lo specialista accetta il paziente e aggiorna il Workflow Document, in modo che nessun altro specialista possa iniziare un altro consulto. Il Workflow Document viene aggiornato con una nuova versione nella quale una nuova task "referred" è stata aggiunta al contenuto della versione precedente. Questa task è caratterizzata dalle seguenti informazioni:

- uno stato "in-progress", che permette di definire lo stato associato alla task Event descritta

- come input della task il riferimento al documento e-Referral prodotto dal medico di base

Il software dello specialista, in qualità di Content Updater e di XDS Document Source, realizza

l'aggiornamento del Workflow Document con la nuova versione del documento e invia il documento all'XDS Document Repository/Registry attraverso un replace del documento stesso.

-C. Lo specialista termina il proprio consulto e produce il proprio report. In questa task il software dello specialista, in qualità di Content Updater, aggiorna il Workflow Document cambiando lo stato della task “referred”. Questa task è caratterizzata da:

- stato “completed”, in quanto il nuovo task Event prodotto determina il completamento del consulto specialistico

- come input della task è fornito il riferimento all'e-Referral prodotto dal medico di base, immaginando che lo specialista non utilizzi i report ai documenti clinici referenziati dal medico di base alla prima task.

- come output della task viene fornito il riferimento al report del consulto prodotto dallo specialista.

La storia dei cambiamenti di stato della task è tracciato all'interno della task stessa, attraverso una lista chiamata taskEventHistory. Il software dello specialista in quanto Content Updater e Docuemnt Source, realizza l'aggiornamento del Workflow Document inviando la nuova versione dello stesso all'XDS Document Repository, mediante un replace della vecchia versione del Workflow Document.

In ogni istante il medico di base può visionare il Workflow Document e i documenti prodotti e associati a questo workflow. Questo può essere realizzato attraverso una query ed un retrieve eseguito dal software del medico di base stesso al Repository ed al Registry.

Anche se non descritto in questo caso d'uso, è sempre possibile utilizzare i profili di notifica DSUB o NAV per poter gestire un sistema di notifiche al medico di base.

5.4 Conclusioni sull'indagine comparativa

Nei capitoli 6 e 7 verranno poi analizzati in modo specifico i dettagli implementativi per entrambe le versioni dell'XDW Workflow Document. Tuttavia è già possibile definire le due principali differenze che contraddistinguono le soluzioni proposte. La prima riguarda la struttura scelta per descrivere un evento del processo. La seconda e più evidente differenza risiede nei diversi linguaggi standard utilizzati per la strutturazione del documento, la quale comporta un gran numero

di conseguenze pratiche legate alla particolare architettura dei sistemi definiti tramite Human Tasks ed alle potenzialità del linguaggio.

Per quanto riguarda il primo spunto di analisi, infatti, si può osservare come nel caso CDA-based si sia deciso di strutturare il body del documento con una serie di steps identici, associando ogni azione svolta all'interno del Workflow ad uno step ben definito. In questo modo i cambiamenti di stato dei documenti o degli step stessi vengono tracciati come step indipendenti. Questo tipo di approccio fornisce linearità al linguaggio adottato e elevata semplicità nella strutturazione del documento, ma soffre dei limiti legati alla descrizione specifica di ogni step. Non è sempre possibile infatti definire degli step specifici per ogni cambiamento di stato.

Attraverso l'approccio offerto dallo standard Human Task è stato invece possibile strutturare il documento in task. Ad ogni task sono associati uno o più Task Event che permettono di tracciare in una lista i cambiamenti di stato legati alla task stessa. Non si tratta di vere e proprie sotto-task, che tuttavia il linguaggio Human Task renderebbe disponibili, ma un modo per monitorare l'evolvere di un processo che non è stato ultimato (descrivendo le varie fasi di un processo più complesso). In questo modo dunque, se non è possibile definire una particolare task per un evento che comporta solo il cambiamento di stato di una task precedente, l'approccio offerto permette di modificare lo stato della task aggiungendo un elemento task Event. Non essendo tuttavia definite in modo specifico quali attività debbano essere descritte tramite una task e quali siano riconducibili ad un task Event, questo secondo approccio potrebbe portare alla perdita di consistenza del linguaggio perché documenti strutturati in modo diverso potrebbero descrivere lo stesso processo. Vengono quindi fornite parallelamente al profilo delle indicazioni che suggeriscono, ovunque sia possibile, di utilizzare task e non task Event. Attraverso questo approccio possono essere definite task anche per compiti in programma e non ancora eseguiti, gestendo i vari stati che può assumere la task stessa (creato, preso in gestione, in programma di esecuzione, in-progress, fallito, ultimato ecc...).

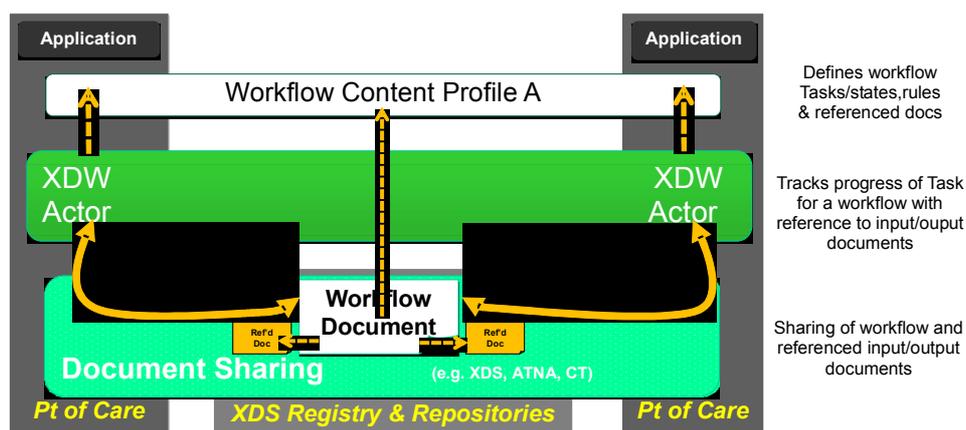
Per quanto riguarda invece le differenze imposte dalla scelta del linguaggio standard, usato per la strutturazione del documento, si possono fare diverse considerazioni. Innanzitutto in entrambi gli approcci non si è deciso di utilizzare tutte le funzionalità dei linguaggi utilizzati. In particolare il linguaggio CDA è progettato per strutturare documenti clinici, e permette di definire un template specifico per ogni tipo di informazione clinica descritta. Tuttavia una caratteristica fondamentale del XDW Workflow Document prevede che il documento non debba contenere informazioni cliniche, ma solamente riferimenti esterni ai documenti correlati allo specifico workflow clinico. Di conseguenza viene impedito l'utilizzo di tutti quei metadati che forniscono informazioni cliniche.

Per quanto riguarda lo standard Human Task, invece, si è deciso di adottare per semplicità una struttura semplificata per le task, che vengono definite "lean tasks". Questa scelta permette in

particolare di non fornire all'interno delle task indicazioni per il rendering delle informazioni e impedisce la definizione di sotto-tasks. Entrambe le scelte permettono di migliorare l'interoperabilità delle applicazioni che si scambiano i Workflow Document.

Il linguaggio Human Task è concepito all'interno di un architettura di sistema organizzata su più livelli. In particolare è possibile definire una sovrastruttura esterna al documento stesso, sulla quale far riferimento per poter individuare non solo la tipologia di workflow in esecuzione ma anche le particolari regole di sviluppo del workflow stesso, definendo espressamente: gli stati possibili, le possibili evoluzioni del flusso da una qualsiasi configurazione specifica e documenti che possono essere referenziati (figura 5.11). XDW costituisce il nucleo centrale del sistema necessario per garantire l'interoperabilità, e fornisce una piattaforma sulla quale possono essere definiti un gran numero di Workflow specifici, definiti tramite una “content specialization” che richiede risorse minime per la specificazione e l'implementazione.

Figura 5.11: Architettura del sistema XDW



Questo approccio non può essere utilizzato nel sistema definito tramite il linguaggio CDA, anche se è possibile definire il business del workflow valorizzando appositi campi codificati all'interno del documento strutturato. In questo modo possono essere definite le regole per la successione degli step, il tipo dei documenti che possono essere referenziati, e i valori che possono assumere determinati campi. Altri workflow invece, possono essere referenziati semplicemente attraverso un identificatore univoco. Questo approccio lascia la gestione dello sviluppo del workflow al personale sanitario. Gli step futuri non sono gestiti direttamente tramite XDW, ma vengono considerati solo ad un livello superiore di XDW, dove, le eventuali regole di svolgimento del workflow vengono considerate come parte integrante ai passi precedenti. Questa scelta comunque non comporta la definizione di regole di conformità per la creazione del documento stesso, le quali possono essere

definite solo attraverso il riferimento ad uno specifico template per la particolare tipologia di Workflow.

Il linguaggio Human Task permette inoltre al documento di essere gestito direttamente da tutte le organizzazioni che utilizzano lo standard BPEL per la definizione dei workflow. In particolare un'organizzazione che utilizza questo standard può direttamente estrarre informazioni e le task complete dal Workflow Document una volta che la task deve essere processata, posizionarle all'interno del Workflow Management System per essere eseguite e aggiornare direttamente il documento una volta che il lavoro è stato completato.

5.5 Considerazioni sulla sicurezza di XDS

Il profilo XDW non include nessuna politica di sicurezza o controllo sulla privacy. Infatti, XDW è un documento amministrativo, il quale non deve contenere informazioni cliniche ma solo informazioni amministrative, le quali tuttavia possono essere di critica importanza tanto quanto le informazioni cliniche. L'XDW Workflow Document verrà redatto da differenti organizzazioni, le quali dovranno sostituire la versione precedente del documento. Per un documento clinico non è pensabile che un'organizzazione sostituisca un documento creato da un'altra organizzazione. Generalmente infatti un documento clinico è totalmente prodotto da un individuo od un'organizzazione. Esistono quindi delle regole che limitano le operazioni di Replace solo per documenti per i quali l'autore è il medesimo sia nella vecchia versione che nella nuova. Questa regola deve essere abbandonata per poter permettere il funzionamento di XDW. Questo cambiamento deve essere monitorato e controllato, quindi il trasporto per XDW deve essere raggruppato con i profili ATNA e XUA. Questo comporta delle regole di sicurezza aggiuntive rispetto a quelle necessarie per gli usuali report clinici.

6. XDW Workflow Document

CDA-based

6.1 Health Level 7 (HL7)

HL7 è un organismo internazionale di standard afferente all'ANSI (American National Standards Institute) e formato da specialisti del mondo sanitario che ha come mission la predisposizione di standard per lo scambio, la gestione e l'integrazione in formato elettronico delle informazioni sanitarie, promuovendone l'uso all'interno delle diverse organizzazioni assistenziali.

Nel corso degli anni HL7 si è affermato come standard universalmente adottato a livello internazionale per la messaggistica e per i documenti elettronici (CDA Rel. 2.0) in ambito sanitario ed è adottato da tutti i più rilevanti programmi nazionali di sanità elettronica sia in ambito europeo che extraeuropeo. HL7 ha una vasta rete di organizzazioni affiliate a livello nazionale responsabili per la localizzazione.

Lo standard HL7 versione 3 è basato su un approccio Model Driven ed è fondato su unico RIM (Reference Information Model) che è il sorgente di tutti i modelli informativi definiti nei diversi domini coperti da HL7v3, compreso lo stesso standard CDA Rel. 2.0. Il RIM è, inoltre, riconosciuto come standard ISO. Il RIM definisce un modello astratto per la rappresentazione dell'insieme delle informazioni sanitarie così come veicolate tra attori ed organizzazioni del mondo sanitario. Il RIM è specificato tramite un modello UML, che definisce l'insieme delle classi che rappresentano entità, ruoli e partecipazioni ammessi da HL7v3 e dai quali vengono derivati tutti i modelli dei diversi domini attraverso un processo formale (tipico di un approccio Model Driven) di trasformazione e raffinazione ("refinement").

6.2 Clinical Document Architecture (CDA)

Clinical Document Architecture è uno standard che specifica la struttura e la semantica di documenti clinici per lo scambio all'interno del dominio sanitario. Un documento CDA è un oggetto informativo strutturato in grado di contenere testi, immagini, suoni ed altri contenuti multimediali. Esso è composto da differenti blocchi informativi che veicolano informazioni relative

al paziente, al medico, alla struttura sanitaria, all'autore del documento, al firmatario del documento, agli eventi clinici, alle osservazioni o alle procedure mediche a cui il documento si riferisce. Ogni documento deve essere, come prescrive lo standard stesso, "human readable" e quindi in grado di essere visualizzato in maniera algoritmica dal ricevente del documento senza la necessità di conoscerne le specificità. Da un punto di vista tecnico, la struttura dei documenti CDA, deriva in modo formale dal Reference Information Model (RIM) di HL7 versione 3 e ne utilizza, quindi, i relativi datatype.

L'utilizzo formale del RIM di HL7v3 garantisce la flessibilità necessaria anche in relazione alle future evoluzioni dello standard fornendo un modello per l'implementazione di documenti strutturati. Lo standard CDA si presta alla rappresentazione di diverse tipologie di documenti clinici fornendo peraltro un elevato grado di flessibilità nelle modalità di rappresentazione di concetti. In tale contesto è quindi necessario, anche in funzione delle specificità del paese nella gestione delle informazioni e processi sanitari, adattare lo standard in relazione ai singoli oggetti informativi che si vuole rappresentare (ad es.: prescrizione farmaceutica, prescrizione specialistica, referto, lettera di dimissione,...) fornendo per ciascuno di questi il dettaglio relativo alla modalità di rappresentazione in CDA dei concetti, delle informazioni e delle codifiche in essi contenute. Lo standard HL7-CDA Rel. 2.0 fornisce in tale ambito delle specifiche norme che tale processo di adattamento deve ad ogni modo seguire (HL7 Refinement and Localization) per garantire, durante il processo di localizzazione della specifica (definizione dei vocabolari, codifiche, estensioni), la completa conformità allo standard. Un documento CDA viene trasmesso all'interno di un messaggio HL7 definito per trasferire documenti clinici, tuttavia la specificazione del linguaggio adottato per questi messaggi è al di fuori degli scopi di CDA.

Un documento clinico contiene osservazioni e descrizioni di servizi, e deve possedere le seguenti caratteristiche:

- PERSISTENZA: il documento clinico deve continuare ad esistere inalterato per un determinato periodo definito dai regolamenti locali.
- AMMINISTRAZIONE: il documento clinico è conservato da un organizzazione specifica che ne è responsabile.
- AUTENTICAZIONE: il documento clinico è un assemblaggio di informazioni che devono essere legalmente autenticate.
- CONTESTO: il documento clinico stabilisce il contesto di default per il suo contenuto.
- INTERESSA: l'autenticazione del documento clinico si applica all'intero documento e non si applica a porzioni del documento senza il contesto completo del documento.

-LEGGIBILITA' UMANA: Il documento clinico deve essere comprensibile alla lettura umana.

Un documento CDA non specifica la creazione o la gestione dei documenti, ma solamente la marcatura per lo scambio del documento tra entità diverse garantendone l'interoperabilità.

Gli obiettivi di un documento CDA sono:

- dare la priorità al processo di cura del paziente
- supportare lo scambio di documenti leggibili tra persone anche con diversi livelli di conoscenza tecnica
- promuovere il mantenimento nel tempo di tutte le informazioni codificate
- abilitare un vasto range di applicazioni che processano i documenti dopo lo scambio.
- compatibilità con un vasto numero di applicazioni che creano documenti
- realizzare il progetto in tempi ragionevolmente veloci
- permettere un controllo dei requisiti delle informazioni contenute da parte degli enti specifici incaricati

Da questi obiettivi segue un certo numero di principi di progettazione del linguaggio:

- l'architettura deve essere compatibile con XML e HL7 RIM
- le barriere all'utilizzo dell'architettura devono essere minime
- l'architettura specifica lo gli schemi necessari per effettuare lo scambio di documenti
- l'architettura deve richiedere il minimo numero di regole e requisiti sulla struttura del documento e sui contenuti necessari per garantire lo scambio
- l'architettura deve essere scalabile per favorire markup molto dettagliati
- il documento prodotto deve poter alloggiare informazioni e requisiti necessari a vari aspetti: professionale, commerciale e normativo
- il documento, per essere leggibile da soggetti umani, deve essere compatibile con i browser XML comunemente adottati e con i comuni drivers di stampa.
- devono essere utilizzati standard aperti

6.2.1 Componenti di un documento CDA

Le componenti principali di un documento CDA sono mostrate di seguito:

```
<ClinicalDocument>
... CDA Header ...
<StructuredBody>
  <section>
    <text>...</text>
    <Observation>...</Observation>
    <Observation>
      <reference>
        <ExternalObservation>...</ExternalObservati
on>
      </reference>
    </Observation>
  </section>
  <section>
    <section>...</section>
  </section>
</StructuredBody>
</ClinicalDocument>
```

Un documento CDA è racchiuso all'interno di due elementi `<Clinical Document>` e contiene un header ed un body. L'header si sviluppa dall'elemento `<Clinical Document>` all'elemento `<Structured Body>` dove viene classificato e identificato il documento stesso, oltre a fornire informazioni relative all'autenticazione, al paziente, ai provider del servizio e all'incontro tra paziente e provider. Il body invece contiene la relazione dell'episodio clinico descritto nel documento. Il body può essere in forma non strutturata o strutturato mediante un linguaggio di markup, in quest'ultimo caso il body deve essere contenuto all'interno di due elementi `<Structured Body>` e può essere suddiviso in sezioni che sono sistemabili in modo ricorsivo. Ogniuna di queste sezioni è racchiusa tra due elementi `<Section>` e può contenere un singolo blocco narrativo (Section Narrative Block) ed un qualsiasi numero di accessi (Entry Acts) e di riferimenti esterni. Il narrative block è racchiuso all'interno di due elementi `<text>` posti all'interno del medesimo blocco section, e costituisce uno slot per le informazioni leggibili che devono essere mostrate. Sempre all'interno di una section del body sono contenute le eventuali entries, che costituiscono invece la parte strutturata dei dati il cui contenuto è quindi machine-readable. Anche i riferimenti esterni concorrono all'interno del contesto delle CDA entries, e sono

racchiusi tra due elementi <reference>. Questi riferimenti devono puntare ad oggetti che esistono al di fuori del documento CDA, come per esempio immagini, suoni, video o altri documenti. La CDA entry che racchiude il riferimento ad un oggetto esterno può essere utilizzata per codificare una specifica porzione del riferimento esterno che è indirizzata nel narrative block.

6.2.2 Leggibilità umana e rendering dei documenti CDA

La necessità di un documento CDA di essere leggibile garantisce al destinatario di un documento CDA che possa, attraverso un approccio algoritmico, mostrare il contenuto clinico all'interno di un browser standard per il web.

Per garantire questa potenzialità ci sono delle necessità che influenzano la struttura del linguaggio CDA:

- deve esistere un modo deterministico per un recipient di un documento CDA per visualizzare il contenuto del documento stesso
- la leggibilità non comporta la necessità di trasmettere uno specifico style-sheet. Deve essere quindi possibile visualizzare tutti i documenti CDA con un unico style-sheet e con un applicativi generici per la presentazione del contenuto
- la leggibilità deve valere per il contenuto autenticato del documento, ci possono dunque essere delle informazioni aggiuntive espressamente dedicate al machine-processing, che non sono autenticate e quindi non è necessario visualizzarle.
- quando del contenuto strutturato viene derivato dal contenuto narrativo devono essere espressamente definiti i meccanismi tramite i quali avviene questa derivazione.
- quando la porzione narrativa è derivata dal contenuto strutturato deve essere identificato il processo che porta a questa derivazione.

Queste necessità hanno comportato lo sviluppo attuale del linguaggio CDA, per il quale il materiale che deve essere visualizzato è posizionato all'interno del campo Section.text. Il contenuto di questo campo è specificatamente aggiunto a mano dall'operatore che realizza il documento. Le osservazioni strutturate possono riferirsi al contenuto del campo Section.text.

6.2.3 CDA e HL7 Messaging Standards

Un documento CDA è oggetto di informazione definito e completo, il quale esiste anche al di fuori del contesto di messaggi tra partner di un Workflow, oppure può essere codificato all'interno di un

messaggio HL7. Di conseguenza CDA integra le specifiche della messaggistica HL7.

Dal punto di vista di HL7 V2 o V3 message un documento CDA è un oggetto multimediale che può essere scambiato come un pacchetto Multipurpose Internet Mail Extensions (MIME, RFC 2046) codificato come un tipo di dato incapsulato. Ogni strategia per lo scambio di documenti CDA deve soddisfare le seguenti specifiche:

- non esiste la necessità di cambiare i riferimenti interni al documento CDA quando viene creato il pacchetto di scambio
- non esiste la necessità di cambiare i riferimenti interni al documento CDA quando viene aperto il pacchetto di scambio
- tutte le componenti del documento CDA che sono parte integrante del documento (come per esempio un body non strutturato secondo linguaggio XML) sono incluse nel medesimo pacchetto.
- non ci sono restrizioni sulla struttura del pacchetto dei riceventi, e possono posizionare le componenti del documento CDA in cartelle a loro scelta.

In HL7 V3 i documenti CDA possono essere scambiati all'interno di ogni messaggio che permette lo scambio di documenti (per esempio HL7 V3 Medical Records Messages). L'attributo Act.text contiene il pacchetto MIME, codificato come un tipo di dato incapsulato. Dato che HL7 V3 Medical Records Messages e CDA derivano da un modello comune, molti campi all'interno del messaggio si sovrappongono in significato con i campi del documento CDA. La seguente tabella mostra le corrispondenze tra questi campi:

HL7 V3 Medical Records Component	CDA Component
ClinicalDocument	ClinicalDocument
authenticator	authenticator
legalAuthenticator	legalAuthenticator
dataEnterer	dataEnterer
encounterPerformer	encounterPerformer
responsibleParty	responsibleParty

HL7 V3 Medical Records Component	CDA Component
custodian	custodian
participant	participant
informationRecipient	informationRecipient
recordTarget	recordTarget
author	author
subject	subject
relatedDocument / ParentDocument	relatedDocument / ParentDocument
documentationOf / Event	documentationOf / Event
inFulfillmentOf / Order	inFulfillmentOf / Order

Un documento clinico può essere rivisto ed aggiornato, e può essere aggiunto ad un documento già esistente. Idealmente, un documento aggiornato dovrebbe auto-dichiararsi obsoleto, e contenere automaticamente un puntatore verso la più recente versione del documento stesso. Questo dovrebbe permettere ai provider della sanità di basare le proprie scelte medico-diagnostiche su dati completi ed aggiornati. In pratica, tuttavia, è impossibile garantire un esplicito puntatore in avanti verso la nuova versione del documento. Senza un processo che tracci la catena di custodi dei documenti clinici e delle loro copie, non esiste la possibilità di garantire che un documento clinico che viene visualizzato non sia stato successivamente revisionato e corretto con una nuova versione.

Per minimizzare dunque il rischio di visualizzare informazioni obsolete, esiste una critica interdipendenza tra documenti clinici e i sistemi di gestione dei documenti. Se un documento viene visualizzato all'esterno del contesto del sistema di gestione dei documenti, non è possibile conoscere con certezza l'attendibilità dei dati contenuti nel documento stesso. I messaggi HL7 che gestiscono l'invio di documenti CDA garantiscono una corretta visualizzazione delle informazioni cliniche.

6.2.4 Conformità del documento CDA

Un documento è conforme allo standard CDA se rispetta il CDA Schema, e utilizza valori per i

campi che appartengano alla lista dei vocaboli codificati all'interno del vocabolario del dominio specificato. Tuttavia la maggior parte delle caratteristiche del linguaggio CDA non possono essere validate a livello macchina, in particolare i requisiti correlati alla leggibilità umana del documento.

Documenti CDA possono essere prodotti da un applicazione (che prende il nome di Document Originator), o ottenuti tramite una conversione da altri formati come un output diretto di un applicazione autorizzata. Questo Document originator è spesso responsabile sia della comunicazione con l'attore responsabile della memorizzazione persistente del documento attraverso l'utilizzo di messaggistica HL7, sia della conformità del documento con la sua specificazione.

Un document recipient è un'applicazione che riceve aggiornamenti di stato e documenti prodotti dai document originator ed è responsabile di garantire che il documento CDA sia conforme alle specifiche. Il formato CDA è uno standar progettato per lo scambio di documenti, di conseguenza non esistono dei particolari requisiti per la memorizzazione persistente. Tuttavia la gestione documentale è fondamentale anche per la specifica del linguaggio CDA. L'attore responsabile della memorizzazione permanente del documento (che può dunque avvenire anche sotto forma di un altro formato) è il custodian ed è definito all'interno dell'Header del CDA.

Le responsabilità del recipient:

- Assegnare i valori di default (dove definiti) per le istanze per le quali non è stato specificato un valore
- Analizzare i blocchi dell'header e interpretarne il significato.
- Analizzare i blocchi del body e interpretare il significato in modo sufficienter tale da poterlo visualizzare (non tutte le entries devono essere interpretate).

Le responsabilità dell'originator:

- Deve garantire che la porzione del body sia strutturata in modo tale che il recipient possa interpretarne il significato
- Non è richiesto di codificare l'intera porzione narrativa in entries

6.3 XDW Workflow Document

Per la progettazione del Workflow document si è deciso di utilizzare la struttura offerta dallo standard HL7 CDA Rel. 2.0. Il documento strutturato è presentato nell'appendice A. La scelta dei metadati e dei valori da associare ad essi è presentata di seguito.

6.3.1 Metadati dell'header

-<realmCode>: è l'elemento obbligatorio che definisce il dominio di appartenenza del documento. Indica l'esistenza di un insieme di restrizioni applicate per il dominio italiano al profilo HL7-CDA Rel. 2.0.

Attributo	Tipo	Valore	Dettagli
code	ST	"IT"	Definisce l'id di contesto per l'Italia

-<typeId>: è l'elemento obbligatorio che indica che il documento è strutturato secondo le specifiche HL7-CDA Rel 2.0, e più precisamente dallo schema definito da "CDA Release Two Hierarchical Description".

Attributo	Tipo	Valore	Dettagli
root	OID	"2.16.840.1.113883.1.3"	Object identifier di HL7 per i modelli registrati
extension	ST	"POCD_HD000040"	Codifica identificativa del CDA Release Two Hierarchical Description. Che rappresenta lo schema che contiene la gerarchia delle classi CDA

-<templateId>: elemento obbligatorio che indica il template di riferimento per il documento CDA. I template possono essere utilizzati per individuare, in funzione della tipologia di documento espresso da <code>, un insieme di restrizioni e linee guida da applicare all'intero documento o ad una sua porzione. Per la normativa italiana può essere applicato un unico template all'intero documento (eventuali altre specificazioni nelle sottosezioni possono essere trascurate). Il

documento non viene identificato tramite templateId ma tramite code. <templateId> identifica solo la specifica versione del template che deve essere utilizzata dal document consumer per la validazione del documento corrente.

Attributo	Tipo	Valore	Dettagli
root	OID	((“2.16.840.1.113883.2.9.10.X” HL7 italia template CDA))	OID dell'ente che mantiene gli schemi di template del documento per la sanità elettronica
extension	ST	Proposta??? “IT_WF_DOCUMENT_001”	Possibile valore di template

-<id>: è l'elemento obbligatorio che identifica univocamente l'istanza del documento CDA (rappresenta la IUD: Identificativo univoco del documento). ID non deve essere un codice parlante ma solo un codice univoco nel dominio di riferimento, per questo motivo le applicazioni non devono poter utilizzare l'id per ricavare informazioni sul documento. La codifica proposta suggerisce di valorizzare l'attributo root con l'OID assegnato ad ogni ASL/AO distribuita sul territorio nazionale. L'attributo extension, invece, riporterebbe una codifica univoca per un particolare sotto-dominio così composta:

<ID STRUTTURA>.<ID OPERATORE>.<TIMESTAMP>.<RANDOM SEED>

Nel dettaglio:

-<ID STRUTTURA> è il campo (o una serie di campi separati dal carattere “.”) che identifica la struttura finale che assegna l'identificativo <ID OPERATORE>.

-<ID OPERATORE> è l'ID univoco assegnato dalla struttura competente ad ogni attore in grado di interagire con il sistema.

-<TIMESTAMP> è la data alla quale viene creato il documento, nella forma YYYYMMDDHHmmSS.

-<RANDOM_SEED> è un codice casuale generato al momento della creazione dell'ID (5 caratteri alfanumerici).

Attributo	Tipo	Valore	Dettagli
root	OID	“[OID_ASL/AO]”	OID assegnato ad ogni ASL /AO distribuita sul territorio
extension	ST	[IUD]	Codifica univoca
assigningAuthorityName	ST	“[Nome ASL/AO]”	Nome del dominio di identificazione dei documenti

-<setId> e <versionNumber>: il primo ha valore costante tra le varie versioni del documento, mentre il secondo cambia al variare della revisione. Tutte le nuove versioni del documento devono avere un univoco <id> e diverso dalle precedenti ed un <setId> uguale a quello definito nel primo documento pubblicato. Alla prima creazione <setId> e <id> sono valorizzati nello stesso modo, e progressivamente viene aggiornato solo l'UID del <id>.

Gli attributi di <setId>:

Attributo	Tipo	Valore	Dettagli
root	OID	“OID di Arsenal”	Identificativo univoco del dominio di indetificazione dei documenti
extension	ST	[IURD]	Codifica univoca
assigningAuthorityName	ST	“Arsenal.IT”	Nome del dominio di identificazione dei documenti

Gli attributi di <versionNumber>:

Attributo	Tipo	Valore	Dettagli
value	INT	[versione del documento]	Ad ogni versione successiva del documento deve essere incrementato di un'unità

-<code>: elemento obbligatorio che specifica la tipologia del documento. Il valore deve fare riferimento al sistema di codifica LOINC o, in assenza di codici specifici, ad un'ulteriore codifica condivisa.

Attributo	Tipo	Valore	Dettagli
codeSystem	OID	“2.16.840.1.113883.6.1”	Identificativo univoco per la codifica LOINC
code	CS	“12345-0”	Codice LOINC relativo alla tipologia di documento XDW Workflow Document
codeSystemName	ST	“LOINC”	Nome del vocabolario
codeSystemVersion	ST	“2.19”	Versione del vocabolario

displayName	ST	“XDW Workflow Document”	Descrizione del codice
-------------	----	-------------------------	------------------------

-<effectiveTime>: si tratta di un campo obbligatorio che rappresenta la data e l'ora di creazione del documento. Espresso nel formato di dato TS (time Stamp):

[YYYYMMddhhmmss+|-ZZzz]

Dove vengono espressi anno (YYYY), mese (MM), giorno (dd), ora (hh), minuti (mm), secondi (ss) e l'offset rispetto all'orario di Greenwich (GMT Greenwich Mean time), e per l'Italia può valere +0100 nel caso di ora solare e +0200 nel caso di ora legale.

-<languageCode>: è un elemento obbligatorio che indica la lingua in cui è redatto il documento. Rappresenta un codice conforme alle specifiche dell'IETF (Internet Engineering Task Force) RFC 3066 (OID: 2.16.840.1.113883.6.121). Nel caso del workflow document devo valorizzare il campo in questo modo:

Attributo	Tipo	Valore	Dettagli
code	ST	“en-GB”	Identificativo del nome della lingua

-<confidentialityCode>: è un campo obbligatorio che specifica la riservatezza del documento. Al workflow Document può essere associato un livello di confidenzialità Normal (N), in quanto il documento non deve contenere informazioni mediche riservate.

Attributo	Tipo	Valore	Dettagli
codeSystem	OID	2.16.840.1.113883.5.25	OID della codifica
code	ST	N	Valore di confidenzialità
codeSystemName	ST	HL7 Confidentiality	Nome della codifica
displayName	ST	Normal	Nome del valore di confidenzialità

-<recordTarget>: è un elemento obbligatorio che identifica il soggetto della prestazione. E' un elemento costituito da un ruolo <patientRole> e svolto da un'entità identificata dall'elemento <patient>.

-<patientRole>: Normalmente l'elemento <patientRole> deve prevedere al suo interno almeno un elemento di tipo <id>, destinato ad accogliere la chiave identificativa del paziente, secondo gli schemi assegnati da ogni singola regione e, eventualmente, un ulteriore elemento di tipo <id>, destinato ad accogliere le informazioni relative al codice fiscale.

Esistono tuttavia diverse casistiche possibili , che dipendono dalla tipologia di soggetto in esame:

- **Soggetti assicurati da istituzioni estere:** <patientRole> deve riportare almeno due elementi di tipo <id> contenenti:

1. il numero di identificazione della Tessera Sanitaria (il numero seriale della Tessera TEAM (Tessera Europea di Assicurazione Malattia) per i soggetti assicurati da istituzioni estere);

Attributo	Tipo	Valore	Dettagli
root	OID	"2.16.840.1.113883.2.9.4.3.1"	OID della tessera TEAM estera (verificare che non sia stato cambiato spostandolo da HL7 IT a HL7 international)
extension	ST	[STATO ESTERO] “.“ [NUMERO SERIALE]	Sigla dello stato che rilascia la tessera (standard ISO 3166-1 a tre caratteri) + numero seriale della carta
assigningAuthorityName	ST	[ISTITUZIONE ESTERA]	Nome dell'ente che gestisce gli identificativi

2. il numero di identificazione personale dell'assistito (ad es. il codice fiscale).

Opzionalmente.

Attributo	Tipo	Valore	Dettagli
root	OID	"2.16.840.1.113883.2.9.4.3.3"	HL7 OID per l'identificazione personale TEAM per igli Stati esteri
extension	ST	[STATO ESTERO] “.“ [NUMERO IDENTIFICAZIONE PERSONALE]	Sigla dello stato che rilascia la tessera (standard ISO 3166-1 a tre caratteri) + numero di identificazione personale
assigningAuthorityName	ST	[ISTITUZIONE ESTERA]	Nome dell'ente che gestisce gli identificativi

3. <patientRole> può riportare un ulteriore elemento di tipo <id> atto a identificare il

paziente mediante un codice anagrafico regionale.

Attributo	Tipo	Valore	Dettagli
root	OID	“2.16.840.1.113883.2.9.” + [REGIONE] + “.4.1”	OID dello schema di identificazione regionale
extension	ST	[CODICE IDENTIFICATIVO]	Codice anagrafica regionale

- Stranieri Temporaneamente Presenti (STP): Può contenere fino a due identificativi :

1. deve contenere obbligatoriamente un id contenente il codice identificativo STP.

Attributo	Tipo	Valore	Dettagli
root	OID	“2.16.840.1.113883.2.9.2 [REGIONE].4.1”	OID dello schema di identificazione regionale delle persone
extension	ST	“STP” + [CODICE IDENTIFICATIVO STP ASSEGNATO]	Codice STP di 16 caratteri assegnato allo straniero temporaneamente presente. Deve iniziare con la stringa “STP”. Il codice STP può essere assegnato anche dalla ASL.
assigningAuthorityName	ST	[NOME REGIONE / ASL]	Nome Regione / nome ASL

2.<patientRole> può riportare un ulteriore elemento di tipo <id> atto a identificare il paziente mediante un codice anagrafico regionale.

Attributo	Tipo	Valore	Dettagli
root	OID	“2.16.840.1.113883.2.9.” + [REGIONE] + “.4.1”	OID dello schema di identificazione regionale
extension	ST	[CODICE IDENTIFICATIVO]	Codice anagrafica regionale

- Cittadino italiano o straniero residente (iscritto SSN): viene descritto mediante due id:

1. Un codice facoltativo assegnato dall'anagrafica regionale

Attributo	Tipo	Valore	Dettagli
root	OID	“2.16.840.1.113883.2.9.” +	OID dello schema di identificazione

		[REGIONE] + “.4.1”	regionale
extension	ST	[CODICE IDENTIFICATIVO]	Codice anagrafica regionale

2. L'identificativo obbligatorio che contiene il Codice fiscale del paziente

Attributo	Tipo	Valore	Dettagli
root	OID	“2.16.840.1.113883.2.9.4.3.2”	OID del codice fiscale definito da Ministero Economia e Finanze
extension	ST	[CODICE FISCALE]	Codice fiscale del paziente
assigningAuthorityName	ST	“Ministero Economia e Finanze”	Ministero dell'economia e delle finanze

-<addr>: elemento opzionale con cardinalità massima illimitata. Permette di definire uno o più indirizzi del paziente, specificando per ogni indirizzo memorizzato il tipo di utilizzo utilizzando il vocabolario HL7 “Address use code” definito per il progetto australiano NETHA che prevede le seguenti possibilità:

CODICE	DESCRIZIONE
WP	Work place
PST	Postal address
H	Home address
TMP	Temporary address
omitted	Not stated

La specifica del paese di residenza viene espressa utilizzando il vocabolario ISO 3166-1 a due o tre caratteri.

-<telecom>: elemento opzionale con cardinalità massima illimitata. Permette di memorizzare i recapiti telefonici del paziente, associando ad ogni elemento il tipo di utilizzo dello stesso. I codici per definire il tipo di utilizzo sono mappati nel vocabolario “HL7 v3 TelecommunicationAddressUse”:

CODICE	VALORE
H	Home
HP	Primary home
HV	Vacation home
WP	Workplace
AS	Answering service
EC	Emergency contact
MC	Mobile contact
PG	Pager

-<patient>: L'elemento obbligatorio <patient> contiene i dettagli anagrafici relativi al paziente. Riporta alcuni sotto-elementi obbligatori con l'indicazione dei dati anagrafici, quali in particolare il nominativo del paziente, attraverso l'elemento <name> (ed i sottoelementi <family> e <given>) ed il sesso, attraverso l'elemento <administrativeGenderCode>. E' inoltre facoltativo inserire il luogo di nascita nell'elemento <birthplace>. Nel caso di Referti per i quali sia prevista la possibilità di anonimato, in ottemperanza a quanto previsto dall'art. 87 nella nuova disciplina sulla Privacy (D.Lgs. 30 giugno 2003 n. 196), gli elementi anagrafici <name> e <birthplace>, qualora presenti, vanno riportati sprovvisti di valori, ma devono ambedue essere decorati con l'attributo **nullFavour="MSK"** per permetterne la comprensione al document consumer.

-<custodian>: E' l'elemento obbligatorio che identifica l'organizzazione incaricata della custodia del documento originale. Tale organizzazione è solitamente la struttura di cui fa parte colui che ha creato il documento. L'elemento <custodian> è composto da un ruolo, rappresentato dall'elemento nominato <assignedCustodian>, svolto da un'entità rappresentata dall'elemento <representedCustodianOrganization>.

L'elemento <representedCustodianOrganization> deve contenere al suo interno un elemento <id> che riporta l'identificativo della struttura che ha la responsabilità della conservazione del documento:

Attributo	Tipo	Valore	Dettagli
root	OID	"2.16.840.1.113883.2.9.4.1.1"	Identificativo del dominio di identificazione delle ASL
extension	ST	[ID_ASL]	Identificativo dell'ASL all'interno del dominio di identificazione definito dall'attributo root
assigningAuthorityName	ST	"FLS11"	Dominio di identificazione delle ASL

--<author>: E' l'elemento che identifica il soggetto che ha creato il documento. Esso può essere una persona o una macchina. L'autore può essere identificato da uno o due elementi <id>. L'elemento <author> deve contenere un sotto-elemento <time>, con l'indicazione dell'ora di produzione del documento. La valorizzazione deve essere effettuata attraverso un tipo Time Stamp (TS). La sezione può opzionalmente contenere un elemento <assignedAuthor>/<name> che riporti i dati relativi al nome del soggetto in esame all'interno delle sezioni opportune, così come sezioni dedicate alla memorizzazione di indirizzo, recapiti telefonici, ecc. (sezioni <addr>, <telecom>, ...). L'autore è identificato tramite due <id>:

1. <id> obbligatorio contenente il codice fiscale dell'autore

Attributo	Tipo	Valore	Dettagli
root	OID	"2.16.840.1.113883.2.9.4.3.2"	OID del codice fiscale definito da Ministero dell'Economia e delle Finanze
extension	ST	[CODICE FISCALE]	Codice fiscale dell'autore
assigningAuthorityName	ST	"Ministero Economia e Finanze"	Ministero dell'economia e delle finanze

2. <id> facoltativo che contiene un codice regionale

Attributo	Tipo	Valore	Dettagli
root	OID	"2.16.840.1.113883.2.9. [REGIONE].3.2"	OID dello schema di identificazione regionale degli operatori
extension	ST	[CODICE FISCALE]	Codice anagrafica regionale degli

			operatori.
assigningAuthorityName	ST	“identificativi operatori Regione [REGIONE]”	Autorità definita dall'OID contenuto in root

-<assignedAuthoringDevice>: elemento che permette di definire il device utilizzato dall'autore per redigere il documento. (<softwareName>Workflow Management System v1.0<softwareName>)

-<representedOrganization>: rappresenta l'id dell'organizzazione a cui appartiene l'autore (id root="[OID DELL'ORGANIZZAZIONE DI APPARTENENZA]")

-<relatedDocument>: Elemento opzionale che viene utilizzato per gestire la versione del documento corrente. La nuova versione del Workflow Document è sempre un Replace del documento originario (typeCode="RPLC"). Il sotto-elemento <parentDocument> permette di referenziare il documento originario da cui è derivato il documento attuale.

Attributo	Tipo	Valore	Dettagli
root	OID	“OID di Arsenal”	OID assegnato ad ogni ASL /AO distribuita sul territorio
extension	ST	[IUD]	Codifica univoca
assigningAuthorityName	ST	“Arsenal.IT”	Nome del dominio di identificazione dei documenti

6.3.2 Metadati del CDA-body

Il body del Workflow Document deve essere strutturato (<structuredBody>). Il linguaggio HL7 prevede che il dettaglio di un documento CDA strutturato possa raggiungere fino a tre differenti livelli di specializzazione attraverso gli elementi <section>. Ogni task costituisce una section del body, e all'interno di ogni task le sezioni INPUT e OUTPUT sono definite tramite altre due section. La sezione CONTROL INFO, invece, appartiene al primo livello di specializzazione del linguaggio CDA in quanto contiene le informazioni che riguardano l'intera task in esame e permette di estendere le sovrascritture degli elementi <author>, <effectiveTime> all'interno dell'intera sezione che definisce la task stessa.

Si è deciso di definire uno specifico templateId che identifichi la struttura standard che deve avere ogni task, in modo da poter definire nuove regole di conformità che devono essere verificate da originator e recipient dei documenti. Nello stesso modo si è definito un templateId per le sezioni di INPUT e OUTPUT (le quali sono caratterizzate da una struttura del tutto identica), in modo da poter imporre ulteriori vincoli anche sulla struttura delle due sottosezioni specifiche.

6.3.2.1 Sezione CONTROL INFO

-<code>: è un elemento obbligatorio, che permette di memorizzare il codice identificativo dello step che viene descritto nella task, mediante l'utilizzo del vocabolario specifico del dominio che utilizza il profilo XDW.

Attributo	Tipo	Valore	Dettagli
codeSystem	OID	"2.16.840.1.113883...."	Identificativo univoco per la codifica specifica
code	CS	"12345"	Codice descrittivo della task
codeSystemName	ST	"[VOCABOLARIO_DOMINIO]"	Nome del vocabolario specifico
codeSystemVersion	ST	"1.0"	Versione del vocabolario
displayName	ST	"[NOME_CODICE]"	Nome associato al codice specifico

-<author>: elemento obbligatorio che permette di identificare l'autore della task (identificato con un <id> obbligatorio contenente il codice fiscale e un <id> opzionale che contiene un'identificazione dell'anagrafica regionale). Attraverso un soggetto del tipo <assignedAuthor> è possibile definire indirizzo e recapiti dell'autore, individuare il device utilizzato per redarre il documento e individuare l'organizzazione di appartenenza (nello stesso modo in cui è gestito l'elemento <Author> nell'header del documento). Questa scelta è permessa dalla capacità del linguaggio CDA di sovrascrivere e aggiornare all'interno di ogni sottosezione i valori associati a dei particolari metadati. I nuovi valori valgono all'interno del dominio definito dagli elementi <component> <section>.

6.3.2.2 Sezioni INPUT e OUTPUT

Le sezioni di INPUT e OUTPUT sono sezioni opzionali, che possono figurare con cardinalità

massima pari ad 1 per ogni task. Sono caratterizzate da un codice che descrive la tipologia di sottosezione. Non esiste tuttavia un codice adatto a descrivere una sezione predisposta ad accogliere semplicemente i documenti ed i riferimenti ad oggetti di utilità clinica che sono stati forniti in ingresso al processo che si è svolto mediante la task descritta, e nemmeno un codice per descrivere la sezione predisposta ad accogliere i documenti prodotti all'interno del atto clinico descritto. Per questo motivo si è deciso di definire un nuovo vocabolario che rappresenti le sezioni interne del WorkflowDocument:

Codice	Nome_codice
1	“INPUT”
0	“OUTPUT”

Questo vocabolario può essere referenziato all'interno della codifica OID, ottenendo uno specifico codice di riferimento. In questo modo è possibile completare i campi dell'elemento <code> che identifica la sezione di input o output in questo modo:

Attributo	Tipo	Valore	Dettagli
codeSystem	OID	“[OID DEL VOCABOLARIO]”	Identificativo univoco per il vocabolario di codifica proposto
code	CS	“1” o “0”	Codice descrittivo della sezione input o output (come definito nel vocabolario)
codeSystemName	ST	“[NOME VOCABOLARIO]”	Nome del vocabolario definito
codeSystemVersion	ST	“1.0”	Versione del vocabolario definito
displayName	ST	“INPUT” o “OUTPUT”	Nome associato al codice

--<title>: elemento obbligatorio che definisce il titolo da associare alla sezione di riferimento nel caso di rendering del documento stesso. Può assumere due valori “Section INPUT” o “Section OUTPUT”.

--<text>: elemento obbligatorio che definisce la porzione human readable del body. Si è deciso di strutturarlo come una lista (<list>) di oggetti (<item>), alla quale è associata l'etichetta visualizzabile definita dall'elemento <caption>. Attraverso l'elemento caption può essere definito un ID per il particolare documento di riferimento, in modo da poter realizzare una corrispondenza

tra il contenuto del text e il contenuto delle entries. L'elemento caption permette anche di definire un'etichetta visualizzabile che viene associata all'elemento referenziato.

-<entry>: è l'elemento che permette di definire la porzione machine processable della section. Ogni entry è caratterizzata da un elemento <act> al quale è associato un codice che rappresenta la tipologia di atto descritto. Si è deciso anche in questo caso di definire un vocabolario in grado di descrivere le varie tipologie di atti di interesse per il caso del documento di Workflow. In questo caso esistono due tipologie di atti:

-fornire documenti di utilità alla task

-produrre documenti come risultato della task una volta completata

Al dizionario così definito sarà associato un nome e un OID.

CODICE	NOME CODICE	DESCRIZIONE
00	providing_document	Atto che descrive la fornitura di documenti in ingresso alla task
11	Producing_document	Atto che descrive la produzione di documenti in uscita dalla task

Attributo	Tipo	Valore	Dettagli
codeSystem	OID	"OID d"el dizionario	Identificativo univoco per la codifica
code	CS	"codice dell'atto descritto"	Codice descrittivo dell'atto
codeSystemName	ST	"documental_act "	Nome del vocabolario
codeSystemVersion	ST	"1.0"	Versione del vocabolario
displayName	ST	"[NOME_ATTO]"	Nome associato al codice dell'atto

-<reference>: è un sotto-elemento della entry, che permette di definire eventuali riferimenti ad oggetti esterni. In particolare l'elemento <originalText> permette di correlare il contenuto della porzione narrativa con la porzione processabile, associando all'attributo reference.value il valore del campo della porzione narrativa content.ID. Per esempio se al campo Component/section/text/list/item/content/@ID viene associato il valore "doc1", allora deve essere associato al campo Component/section/entry/act/reference/originaltext/reference/@value il valore "#doc1" in modo da poter correlare l'informazione contenuta nella parte narrativa con quella

processabile.

-<ExternalDocument>: è l'elemento che permette espressamente di memorizzare il riferimento al documento esterno, attraverso l'utilizzo di tre distinti <id>:

-**documentId**: id che definisce l'identificativo del documento

-**repositoryId**: id che definisce l'identificativo del repository in cui è memorizzato il documento

-**HomeCommunityId**: id che definisce l'identificativo della community a cui appartiene il repository.

Ad ogni documento è associato un ulteriore codice, ricavato dal vocabolario LOINC che permette di individuare la tipologia di documento clinico referenziato:

Attributo	Tipo	Valore	Dettagli
codeSystem	OID	"2.16.840.1.113883.6.1"	Identificativo univoco per la codifica LOINC
code	CS	"12345-0"	Codice LOINC descrittivo del documento clinico
codeSystemName	ST	"LOINC"	Nome del vocabolario LOINC
codeSystemVersion	ST	"2.19"	Versione del vocabolario LOINC
displayName	ST	"[NOME_DOCUMENTO_CLINICO]"	Nome associato al codice LOINC del documento clinico

7. XDW Workflow Document

Human Task-based

7.1 BPEL4People

WS-BPEL introduce un modello per descrivere i processi di lavoro basato sui Web Services. Un processo BPEL è costituito dunque da un orchestrazione di WS che interagiscono tra loro. Il linguaggio permette inoltre di descrivere pattern complessi di azioni di controllo esercitate sul processo, includendo gestione degli errori e azioni di compensazione. Nella pratica, tuttavia, risulta necessario poter descrivere e gestire anche interazioni umane all'interno di processi complessi. Una definizione di processo deve poter incorporare sia persone che altre tipologie di partecipanti, in quanto entrambi possono svolgere un ruolo attivo e influenzare lo svolgersi dei processi stessi. In quest'ottica viene definita un'estensione al linguaggio BPEL, che permetta di considerare le interazioni umane come di prima classe: BPEL4People. Esso definisce un nuovo tipo di attività di base che vengono implementate come Human tasks. Le persone spesso partecipano all'esecuzione di un business process comportando l'introduzione di nuovi aspetti nell'analisi del sistema, come per esempio le interazioni tra il processo e le interfacce dei partecipanti e il dover tenere in considerazione i comportamenti umani.

La specificazione del linguaggio BPEL4People introduce un set di nuovi elementi che estendono lo standard BPEL e permettono la modellazione corretta delle interazioni umane, le quali possono variare dal semplice atto di approvazione a scenari molto più complessi. Le attività delle persone sono definite come una nuova tipologia di attività di base, permettendo la definizione dell'interazione umana all'interno di un processo in un modo molto più semplice e diretto. L'implementazione di un'attività umana può essere ottenuta attraverso una definizione diretta in-linea o attraverso una definizione autonoma dei compiti, ottenuta mediante l'utilizzo della specificazione WS-Human Task, che comprende: diagramma di stato delle attività, sintassi, e protocolli di coordinazione per favorire l'integrazione delle human tasks. La specificazione introduce anche le funzioni XPath, necessarie per accedere al contesto del processo.

Gli obiettivi del linguaggio BPEL4People sono essenzialmente due:

-portabilità: l'abilità di prendere artefatti creati in fase di progettazione da fornitori che

lavorano in un certo ambiente e renderlo utilizzabile da altri fornitori che lavorano in un altro ambiente.

-interoperabilità: la capacità di più componenti diversi di interagire tra loro scambiando informazioni utilizzando messaggi ben definiti e protocolli standard. Questo permette di connettere sistemi progettati da fornitori diversi all'interno dello stesso processo di esecuzione.

Lo standard BPEL4People non definisce come i processi che contengono interazioni umane vengano monitorati. Spesso l'introduzione di un ente umano all'interno del processo avviene attraverso l'esecuzione di una query ad una cartella che ha un proprio modello organizzativo.

7.2 Standard WS-HumanTask

7.2.1 Introduzione

Human tasks, o brevemente tasks, permettono di integrare esseri umani all'interno di applicazioni orientate ai servizi. Human tasks sono servizi implementati dalle persone. Una Human task ha due interfacce: una espone il servizio offerto o l'approvazione di un servizio. La seconda interfaccia permette alle persone di interagire con queste tasks.

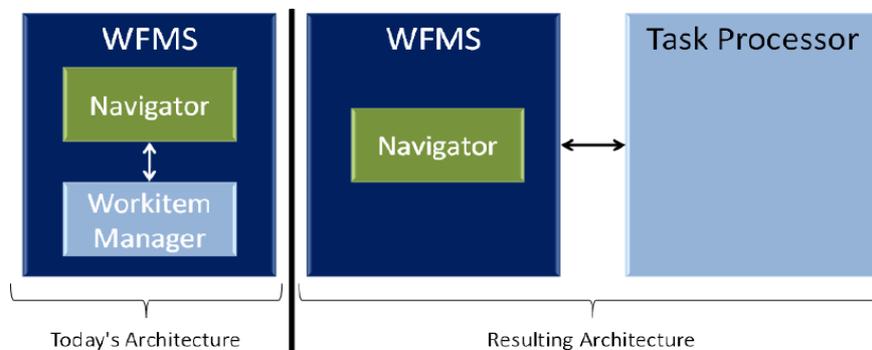
Una human task ha delle persone associate ad essa. Questa assegnazione permette di definire chi è abilitato a svolgere un particolare ruolo all'interno della task stessa. Human tasks devono essere assegnate alle persone in un ordine ben definito, che può essere sequenziale o mediante assegnazioni parallele o qualsiasi combinazione di queste possibilità.

Human tasks possono anche specificare quali metadati della task devono poter essere accessibili ai diversi device o applicazioni che partecipano al processo, rendendo il linguaggio portabile e interoperabile. Human tasks sono definite per reagire a timeout, eventi di trigger e azioni di reazione prestabilite a particolari eventi. Questo è verificato anche per le notifiche, le quali sono speciali tipi di task che permettono l'invio di informazioni riguardanti informazioni utili per la gestione del business. Le notifiche sono task one-way, ovvero vengono inviate a prescindere dal fatto che vengano ricevute dal destinatario. Il linguaggio è focalizzato sull'obiettivo di raggiungere la massima interoperabilità e portabilità. Non è obiettivo della specifica WS-HumanTask definire come le task e le notifiche vengono monitorate e sviluppate. Generalmente l'assegnazione dei ruoli avviene attraverso la realizzazione di una query all'interno di una directory dotata di un certo modello di organizzazione interno. Le modalità di assegnazione dei ruoli e la definizione della struttura delle directory vengono descritte dallo standard BPEL4People.

7.2.2 Architettura del sistema WS-HumanTask

Uno dei motivi per cui lo standard WS-HumanTask ha aumentato notevolmente la sua importanza all'interno dei linguaggi definiti per la descrizione di Workflow è stata la necessità di supportare l'abilità di permettere ad ogni applicazione di creare human tasks in una modalità orientata ai servizi. Tradizionalmente human tasks erano considerate in modo marginale all'interno dei principali Workflow Management Systems. In particolare non era possibile gestire o influenzare l'interezza del life-cycle di un'istanza di Workflow al di fuori della gestione centralizzata del WFMS e quindi non era possibile definire human tasks in grado di influenzare il processo stesso.

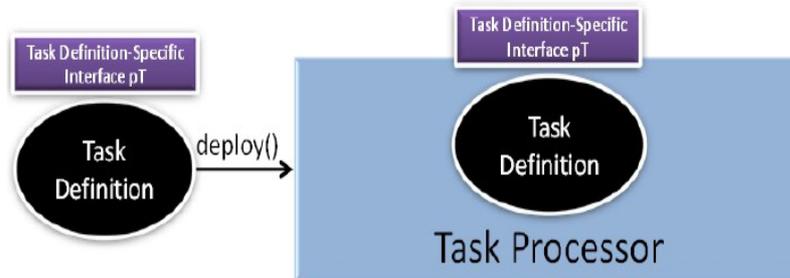
Figura 7.2: Impatto delle Human Tasks sull'architettura del WFMS



Il componente all'interno del WFMS tipicamente responsabile per la gestione del life-cycle della task è chiamato Workitem Manager. Utilizzando l'approccio introdotto dalle Human Tasks, il WFMS non incorpora più il Workitem manager, ma finisce per interagire invece con il Task Processor. In questa architettura il Task Processor è un componente a se stante, separato ed esposto come un servizio, il quale permette la creazione di tasks e l'interazione con altre tasks in risposta all'azione di qualsiasi richiedente interno al sistema. E' il task processor che permette di gestire il life-cycle delle tasks e allo stesso tempo di renderle effettivamente attive all'interno del processo. Separando il Task Processor dal WFMS rende anche possibile l'utilizzo delle Human Tasks anche al di fuori del contesto creato dal WFMS, garantendo l'interoperabilità con qualsiasi altra applicazione WS-HumanTask (anche riferita col nome di Task Parent). Le tasks definite dallo standard WS-HumanTask hanno un interfaccia, rappresentata come una porta applicazione-dipendente, e definita come Task Definition specific interface. Per creare istanze di tasks, gestite da uno specifico Task Processor, viene implementata il tipo di porta necessaria per permetterne la gestione da parte del Task Processor prima che esso venga invocato. In figura 7.3 viene rappresentato il processo di

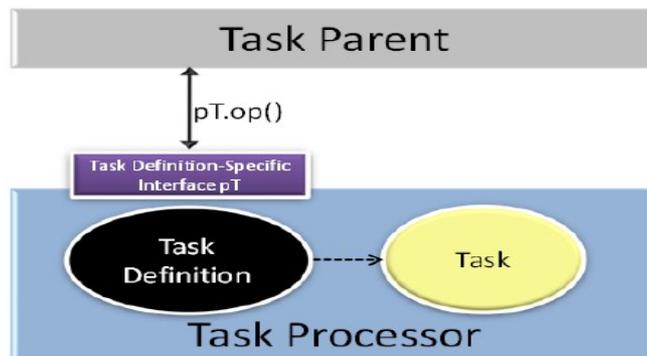
definizione della task e di associazione mediante la porta di interfaccia al corrispondente Task Processor.

Figura 7.3: Definizione della task e associazione al Task Processor



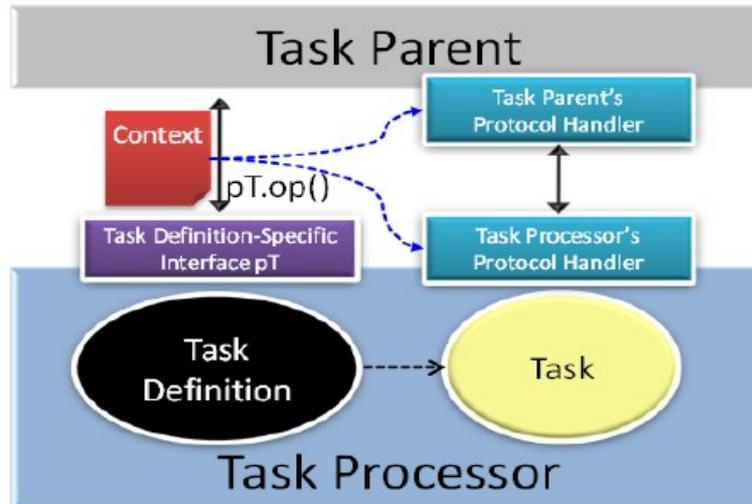
Una volta che la task è disponibile all'interno di un task processor, un qualsiasi requestor può creare un'istanza di task ed interagire con essa. Il requestor che crea un istanza della task è definito come Task Parent. Un'istanza di task è creata invocando un'operazione attraverso l'interfaccia della task che deve essere creata. In genere ogni interfaccia espone solamente una possibile operazione attraverso la propria interfaccia.

Figura 7.4: Processo di creazione di un'istanza di task



Nei sistemi di Workflow in genere il life-cycle di una task è dipendente anche dal WFMS. Infatti se si genera la condizione per cui il processo viene terminato a seguito di un errore il WFMS deve essere in grado di terminare ogni istanza di task attiva nel processo. Per automatizzare il processo che permette di accoppiare strettamente il life-cycle di un Task Parent e di tutte le istanze inizializzate dallo stesso, lo standard WS-HumanTask utilizza il WS-Coordination specification come sistema di coordinazione. Questo richiede la definizione di un protocollo di coordinazione presentato in figura 7.5.

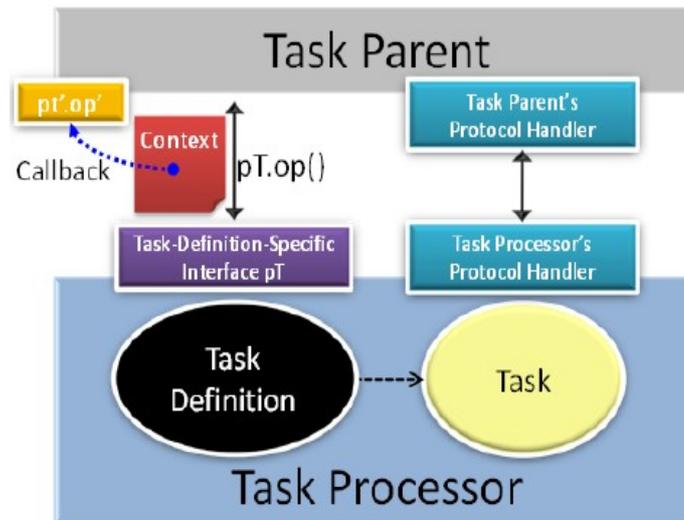
Figura 7.5: Definizione protocollo di coordinazione



Quando un task Parent crea una task utilizzando una specifica operazione `op()` attraverso una porta del tipo `pT`, delle informazioni associate al contesto di coordinazione del life-cycle sono passate dal Task Parent al sistema di gestione, mediante l'utilizzo della stessa porta. Come qualsiasi gestore di life-cycle, le informazioni passate contengono il riferimento esplicito all'end-point dell'istanza, che viene utilizzato dal recipient per creare il corrispondente dato di coordinazione. Una volta che il recipient riceve il riferimento il contesto di coordinamento dell'istanza esso invierà le stesse informazioni rielaborate all'interno del proprio protocollo di gestione del life-cycle al coordinatore del processo. In questo modo è stato definito un canale di coordinazione tra Task Parent e Task Processor, in modo che i due soggetti possono scambiarsi messaggi di protocollo che permettono di accoppiare il life-cycle delle istanze di task con il proprio Task Parent.

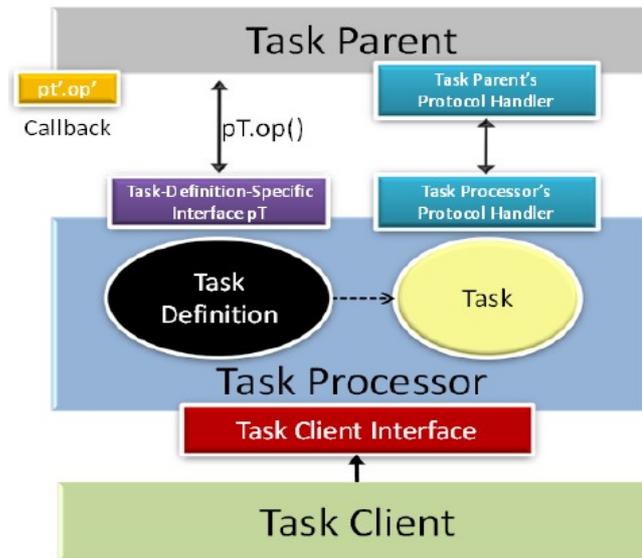
Spesso le tasks sono già in esecuzione all'esterno del processo, e vengono invocate dal sistema in modo asincrono. Perciò il Task Parent inizializza la task e aspetta il risultato prodotto dalla stessa per ritornarlo come informazione utile per il proseguimento del processo. Per permettere la restituzione dei risultati il Task Processor necessita di conoscere dove inviare questi risultati. Per questo motivo il contenuto informativo è esteso con metadati aggiuntivi che specificano il riferimento all'end-point della task, che individua anche il punto dove inviare i risultati, come se l'invio di risultati fosse l'operazione associata all'end-point stesso e eseguita dal Task Processor. In figura 7.6 viene presentata questa funzionalità dello standard, mostrando che le informazioni scambiate contengono anche un riferimento alla porta del tipo "pt" e specificano il nome dell'operazione "op" utilizzata dalla porta stessa per ritornare i risultati.

Figura 7.6: Passaggio di informazioni per garantire la continuazione del processo



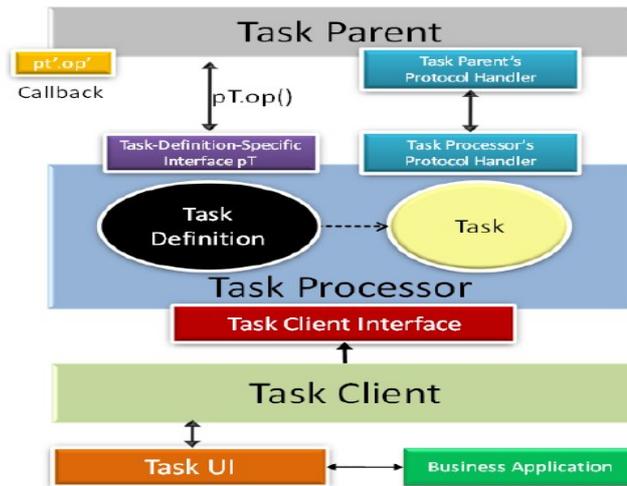
Infine, l'applicazione Task Parent invocando l'operazione implementata dalla task è abilitato a passare dati addizionali parallelamente al messaggio di richiesta. Questi dati sono chiamati Human task context e fornisce la possibilità di sovrascrivere certi elementi che fanno parte della Task Definition. Nello stesso modo anche altre informazioni sono restituite parallelamente all'invio del messaggio di risposta, determinando una propagazione di informazione dalla task completata all'applicazione del Task Parent, come per esempio la definizione delle persone attualmente assegnate agli specifici ruoli, o il risultato della task stessa. Una volta che la task è stata creata essa può essere presentata ai propri (potenziali) possessori per essere manipolata. Per questo motivo, un'altra applicazione chiamata Task Client è tipicamente utilizzata. Un Task Client presenta ad ogni utilizzatore le tasks a cui esso può accedere. Gli utilizzatori possono così rivendicare il possesso di una task e portare a termine il lavoro associato ad essa. Altre funzionalità offerte dal Task Client sono la possibilità di saltare una task, di aggiungere commenti o informazioni accessorie alla task e di nominare altri utilizzatori per realizzare la task. Per permettere al Task Client di svolgere queste funzioni sulle tasks, lo standard WS-HumanTask specifica una task client interface, che deve essere implementata dal Task Processor. In figura 7.7 viene mostrata l'architettura risultante dall'introduzione dei Task Client

Figura 7.7: Introduzione del Task Client e corrispondente interfaccia



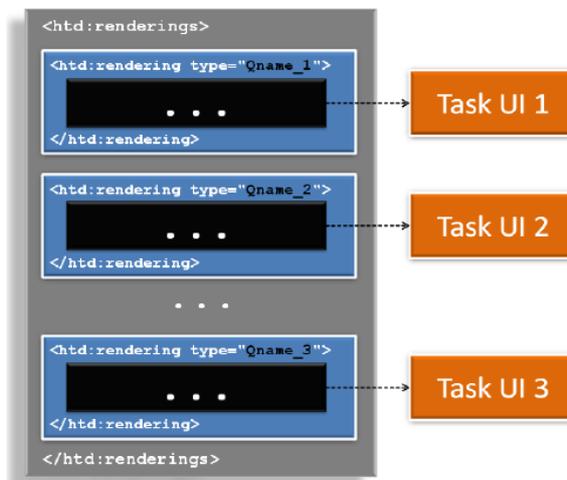
Una volta che un utilizzatore seleziona una task con il proprio Task Client, l'interfaccia dell'utilizzatore associata alla task viene mostrata, permettendo all'utilizzatore stesso di visualizzare le informazioni applicazione-specifiche pertinenti alla task. WS-HumanTask non specifica la struttura di visualizzazione di queste interfacce, ma permette di mostrarne il significato utilizzando un container che permette di definire dei suggerimenti per il rendering ai Task Client. Il Task Client in risposta utilizza le informazioni per costruire o iniziare la costruzione dell'interfaccia dell'utilizzatore della task (i dettagli di come questo viene effettuato sono al di fuori degli scopi di WS-HumanTask). Nel caso di Lean Tasks, questo rendering è generato dal Task Processor. Dal punto di vista del Task Client, il fatto che la task sia una Lean Task non è evidente. Inoltre, la task richiede l'utilizzo di una business application per completare la task stessa (anche l'utilizzo di queste applicazioni è fuori dagli scopi di WS-HumanTask, anche se è comunque fondamentale per la definizione della struttura complessiva dell'architettura, come mostrato in figura 7.8)

Figura 7.8: Architettura complessiva dell'infrastruttura Human Task



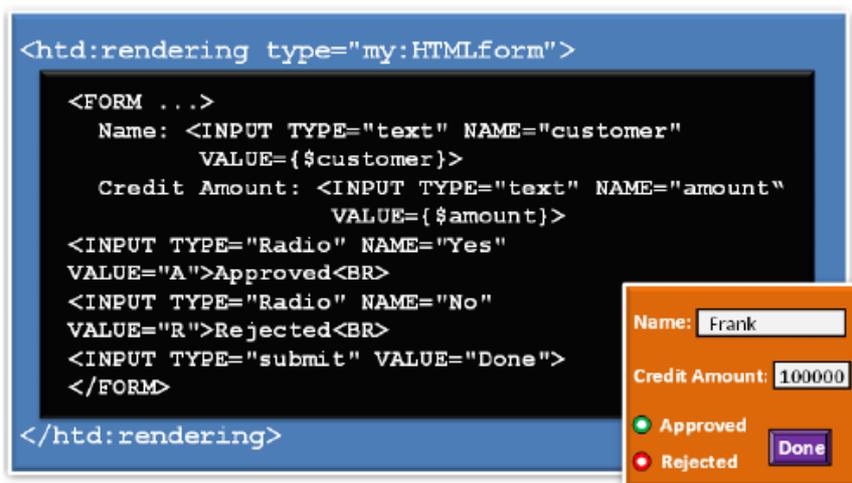
Il container descritto per il rendering delle informazioni associate ad una task, è creato mediante l'elemento <rendering>. Un elemento di questo tipo, specifica il proprio tipo, espresso come un QName che descrive il meccanismo da utilizzare per realizzare il rendering necessario per generare l'interfaccia dell'utilizzatore per la particolare task. Tutte le informazioni necessarie per creare l'interfaccia dell'utilizzatore sono fornite dagli elementi annidati all'interno dell'elemento rendering, come mostrato in figura 7.9. Questi elementi annidati permettono anche di fornire informazioni riguardanti la business application necessaria per completare la task ed altri parametri corrispondenti.

Figura 7.9: Potenzialità di rendering di una task



Per esempio in figura 7.10 viene mostrato il rendering del tipo `my:HTMLform`. Questo QName denota che per il rendering dell'interfaccia corrispondente alla task sia necessaria la capacità di processare il formato HTML. L'elemento annidato contiene il riferimento al contenuto in formato HTML. L'esempio assume che il processore sia in grado di comprendere la notazione `{$...}` per acquisire i valori presentati come input della task.

Figura 7.10: esempio di rendering di una task



Una task può avere diversi rendering associati ad essa. Questo fornisce l'abilità ad una task di essere interpretata attraverso diversi meccanismi di accesso, adattandosi alle preferenze dell'utilizzatore.

7.3 Definizione XDW Workflow Document (HumanTask-based)

7.3.1 Struttura generale del Workflow Document

In questa sezione verrà definita la struttura dell'XDW Workflow Document fornendo uno schema descrittivo e spiegandone l'utilizzo. Come definito nelle sezioni precedenti, questo documento non deve contenere informazioni cliniche riguardanti il paziente. Deve infatti contenere solamente informazioni necessarie ad organizzare e definire tasks di lavoro. Tutte le informazioni cliniche riguardanti qualsiasi task eseguita, devono essere fornite attraverso documenti separati dal Workflow Document, e inserite come riferimento nelle sezioni di input o output dello stesso. Per la definizione della struttura del documento si sono adottati due diversi standard: HL7 CDA Release 2.0 e WS-HumanTask Specification Version 1.1. Infatti l'XDW Workflow Document è un documento che incorpora elementi provenienti da entrambe le strutture.

Il Workflow Document esiste per coordinare le attività di molteplici persone che collaborano all'interno di svariate organizzazioni. Queste organizzazioni spesso hanno per le proprie attività un sistema di gestione interno e sono dotate di un proprio insieme di regole amministrative per dirigere l'interazione tra persone e compiti. In generale è permesso lo scambio di queste regole attraverso documenti per garantire la trasmissione di informazioni di gestione dei lavoro. Spesso la condivisione di documenti è uno step molto semplice da realizzare e può essere svolto come parte del processo di cura. L'XDW Workflow Document permette di condividere e le informazioni associate al processo in atto nello stesso modo in cui vengono condivise le le informazioni cliniche del paziente. Non è possibile integrare diversi Workflow Management Systems interni alle organizzazioni in quanto sono caratterizzati generalmente da regole amministrative indipendenti.

Un Workflow XDW utilizza l' XDS lifecycle management tool per coordinare gli aggiornamenti del workflow document invece di richiedere l'integrazione dei diversi sistemi di gestione delle differenti organizzazioni coinvolte.

Un XDW Workflow Document deve soddisfare i requisiti imposti dal XDW XML Schema, il quale comprende elementi provenienti dallo standard CDA e elementi provenienti dallo standard WS-HumanTask. In figura 7.11 viene rappresentato il livello principale di strutturazione del Workflow Document. E' possibile dividere la struttura principale del documento in quattro parti:

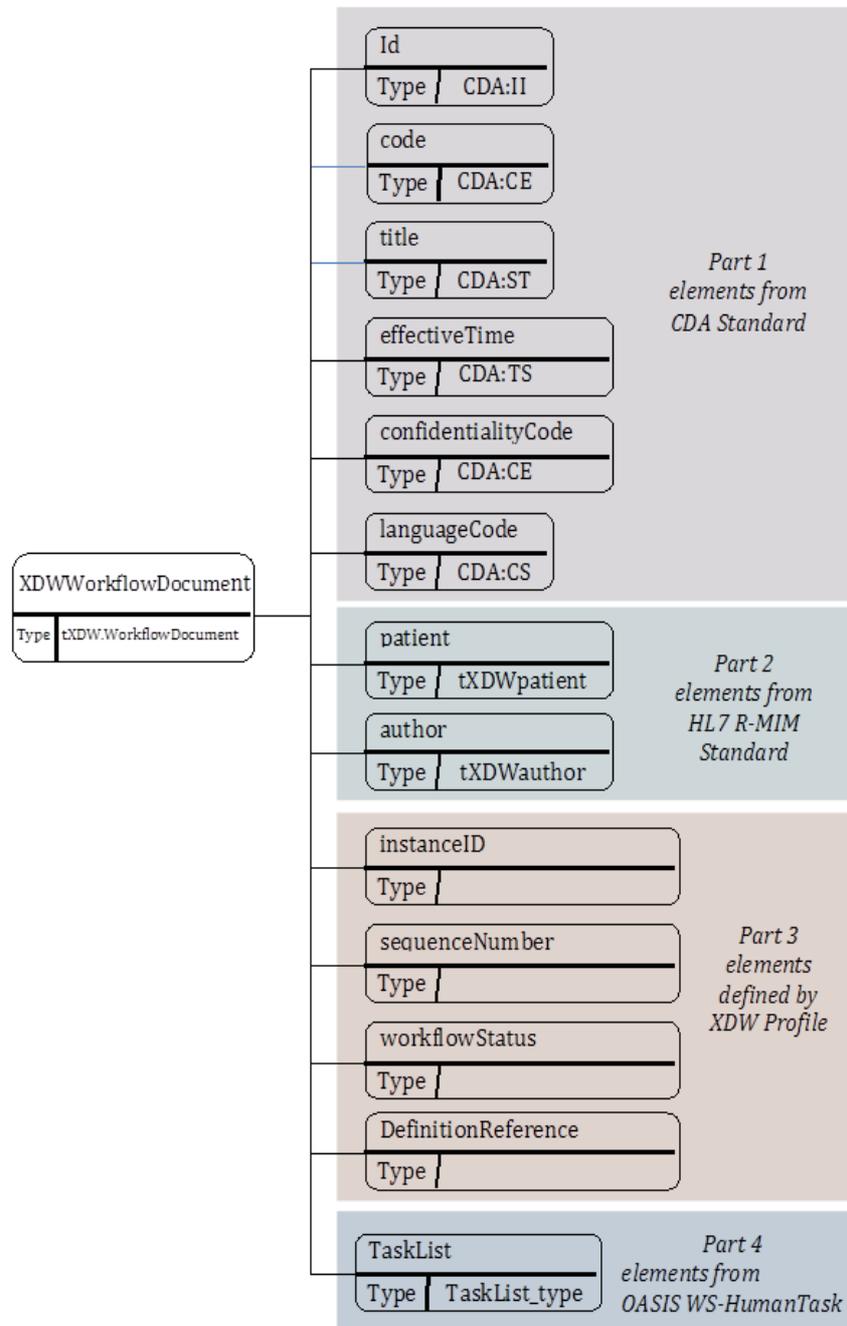
- Parte 1:** elementi derivati dallo standard HL7 CDA (tipo degli elementi: CDA)

- Parte 2:** costituito da due elementi: patient e author, definiti nello XDWSchema con una struttura derivata dallo standard HL7 R-MIM (tipo degli elementi: tXDWpatient e tXDWauthor)

- Parte 3:** elementi definiti dal profilo IHE XDW

- Parte 4:** elemento <TaskList> all'interno del quale sono definiti tutti gli elementi derivati dallo standard OASIS WS-HumanTask. In questa sezione del documento <TaskList> è strutturato come una lista di elementi del tipo <XDWTask> composti da <taskData> (che descrivono tutti i dati che permettono di definire la XDWTask) e <taskEventHistory> che contiene a sua volta una lista di elementi <taskEvent>

Figura 7.11: Struttura dell'XDW Workflow Document



Certi element sono incorporati direttamente dallo standard CDA. Questo significa che gli elementi, la loro definizione e le regole per interpretarli sono definite all'interno dello standard HL7.

Gli elementi Patient e Author sono definiti sulla base dello standard HL7 CDA R-MIM. Lo schema XDW definisce questi elementi utilizzando elementi presi dal linguaggio CDA, e ottenuti eliminando tutti i sottoelementi non necessari all'identificazione del workflow. Gli elementi che richiedono contenuto clinico sono stati rimossi.

7.3.2 Elementi definiti dal profilo IHE XDW

Il Workflow Document contiene anche elementi definiti da IHE:

-<instanceID>: identificatore comune tra tutte le istanze del documento per lo stesso Workflow Document. Rappresenta il Folder.uniqueId del Workflow content Folder all'interno del quale il Workflow Document è contenuto. Deve essere globalmente univoco in quanto condiviso tra diverse organizzazioni. Ogni versione del Workflow Document avrà lo stesso valore di ID.

-<sequenceNumber>: è utilizzato per semplificare la gestione dei cambiamenti avvenuti all'interno del Workflow Document una volta che un lavoro è stato completato. Viene inizializzato a "1" per la prima istanza del Workflow Document e incrementato per ogni aggiornamento del Workflow Document.

-<workflowStatus>: che può assumere due valori:

-OPEN: che significa che sono attesi successivi aggiornamenti per il Workflow Document. Questo può avvenire attraverso la modifica di tasks esistenti o tramite l'aggiunta di nuove tasks o l'aggiornamento di una task esistente mediante un nuovo <taskEvent>. Le Tasks non possono essere cancellate.

-CLOSED: che significa che non possono essere effettuate future modifiche al Workflow Document. Successivi lavori devono essere gestiti attraverso un nuovo Workflow Document.

-<DefinitionReference>: Questo elemento contiene il riferimento ai documenti che definiscono il Workflow. Questi ultimi sono usualmente dei documenti contenenti procedure o politiche per la gestione e l'evoluzione del Workflow specifico. Dato che questi documenti possono essere prodotti da organizzazioni diverse è necessario che ogni riferimento sia descritto tramite un apposito OID, rappresentante la copia del documento stesso presente all'interno del XDS Affinity Domain. Viene assegnato dal creatore iniziale del Workflow Document e deve essere preservato in ogni versione successiva del documento.

7.3.3 Elementi provenienti dallo standard OASIS WS-HumanTask

La descrizione di una task e di un <taskEvent> è presa dallo standard WS-HumanTask. La scelta è ricaduta su questo linguaggio in quanto risulta necessario poter descrivere tasks che sono

state realizzate sia da umani che da organizzazioni.

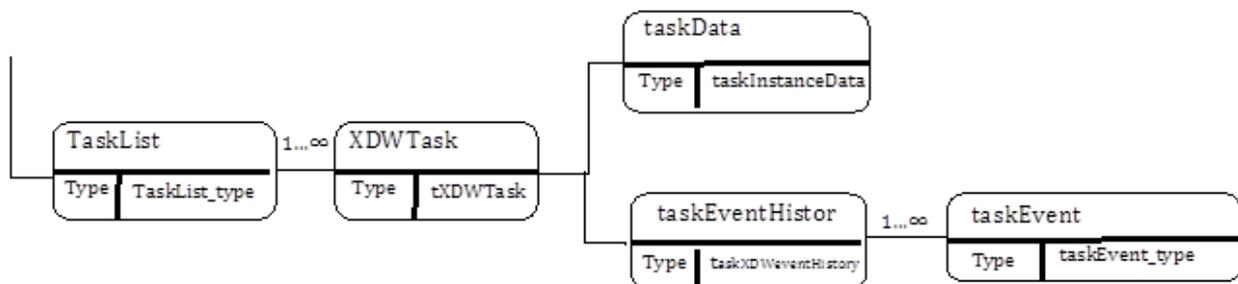
L'elemento `<XDWTask>` raggruppa tutte le informazioni relative ad una task all'interno del Workflow. Questo elemento è strutturato in due sotto-elementi: `<taskData>` e `<taskEventHistory>`.

-`<taskData>` descrive una singola task. Contiene una lunga lista di dettagli relativi alla task, una descrizione della stessa, gli oggetti in input, gli oggetti in output, la descrizione dei possibili fallimenti della task, commenti, altri oggetti e informazioni accessori alla task che non possono essere considerati propriamente degli output. L'elemento `<taskDetails>` include elementi quali `taskId`, `description`, `stato`, ecc.

-`<taskEventHistory>` contiene una lista di elementi `<taskEvent>` che descrivono i cambiamenti avvenuti all'interno della task. Per ogni task si possono avere uno o più `<taskEvent>` che descrive la storia della task. C'è una lunga lista di dettagli relativa ad un `<taskEvent>`: un `<eventType>`, una descrizione, gli oggetti in input al `<taskEvent>`, gli oggetti in output al `<taskEvent>`, descrizione dei possibili fallimenti dell'evento, i commenti ed eventuali oggetti accessori all'evento. I dettagli includono elementi come il task ID, descrizione e stato.

Definizioni e regole, come per esempio lo stato macchina di una task, sono incluse nello standard HumanTask, tuttavia non tutti i tipi di dato ed i web services sono definiti all'interno dello standard.

Figura 7.12: Elementi del XDW Workflow Document derivati da WS-HumanTask



Quando una task viene generata essa contiene il primo `<taskEvent>`. Una task può avere solo un `<taskEvent>` se lo stato della task è non modificabile ed è quindi nata già completata oppure può

avere più stati. In questo secondo caso in ogni istante in cui la task viene modificata un nuovo <taskEvent> viene creato e il Workflow Document è conseguentemente aggiornato.

Quando una nuova task è generata i riferimenti a uno più documenti esterni utili per la task sono posizionati nell'elemento Input della task stessa. Gli inputs di un singolo <taskEvent> possono avere uno o più riferimenti che puntano ai documenti contenuti in questa lista.

Se in un <taskEvent> un documento clinico è prodotto il riferimento a questo documento è posizionato in output all'elemento. L'output di una task sarà costituito dalla lista di tutti i riferimenti ai documenti esterni collezionati da tutti gli elementi <taskEvent>. I documenti clinici in input e output possono essere pubblicati in diverse community, quindi è importante posizionare all'interno del riferimento anche l'XDSdocumentID e l'HomeCommunityID.

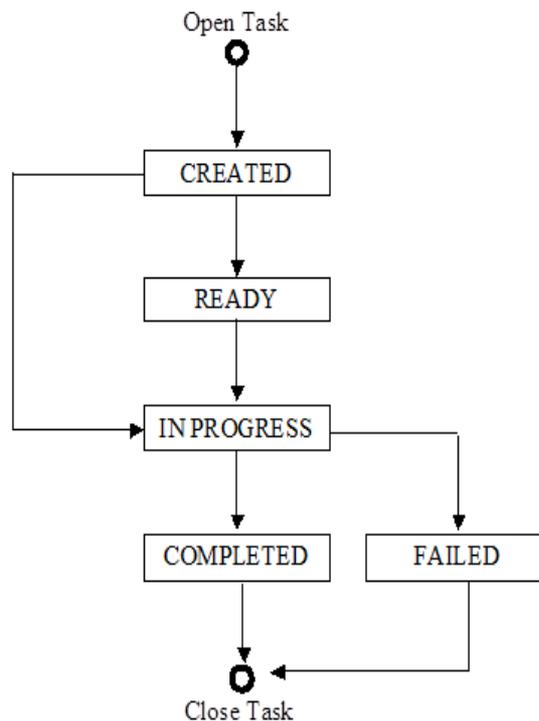
L'XDW Workflow Document definisce una lista di task come una serie di descrizioni delle stesse. Le relazioni tra le task, l'ordine degli elementi in questa lista e il possibile stato di ogni task sono definiti all'interno del Workflow Definition Document.

Il profilo XDW definisce lo stato processato per una task all'interno dell'elemento <taskEvent>. Questi stati sono derivati dallo standard WS-HumanTask. Gli stati principali sono descritti nella tabella seguente:

TASK STATUS	Descrizione	INPUT e OUTPUT
CREATED	Il workflow è aperto. La task è stata creata ma non ha ancora un possessore assegnato	Sono attesi degli INPUT ma nessun OUTPUT
READY	La task è stata creata e il possessore è stato assegnato. La task è pronta per essere eseguita	Stessi INPUT dello stato precedente e possono essere presenti degli OUTPUT
IN PROGRESS	La task è iniziata ed il possessore sta eseguendo le operazioni richieste dalla task stessa	Sono possibili ulteriori INPUT e OUTPUT parziali
FAILED	La task è completata con responso di fallimento. Non è possibile concludere le azioni previste dalla task	
COMPLETED	La task è completata e viene definito un responso della stessa	Tutti gli INPUT utilizzati e tutti gli OUTPUT prodotti

In figura 7.13 è illustrato il diagramma di flusso che presenta i vari step che può seguire una task durante l'esecuzione. Il linguaggio WS-HumanTask offre anche altri possibili stati per descrivere la task, ma sono stati trascurati.

Figura 7.13: Transizione dello stato della task



L'elemento `<XDWTask>` e l'XDW `<taskEvent>` è pienamente estensibile con qualsiasi tipo di elemento definito dallo standard HumanTask. Il profilo XDW definisce gli elementi minimi che devono essere utilizzati, e gli altri profili coinvolti nel processo possono aggiungere elementi per gestire relazioni tra task diverse. Altri profili IHE possono estendere l'elemento `<XDWTask>` o l'elemento `<TaskList>` o il `<taskEvent>` per incorporare informazioni aggiuntive correlate allo specifico tipo di Workflow.

7.4 Gestione del Workflow Document

7.4.1 Gestione del Lifecycle del Workflow Document

Il profilo Cross-Enterprise Document Workflow si avvantaggia della gestione del lifecycle di XDS se utilizzato in un contesto XDS.

Un Workflow Document deve essere creato ed inserito all'interno di una cartella. Il documento iniziale deve includere almeno una task all'interno della taskList ed avere un WorkflowStatus

OPEN. Il Workflow Document è aggiornato quando:

- Le informazioni associate alla task vengono modificate, come per esempio un aggiornamento dei informazioni in output.
- Quando una nuova task viene aggiunta alla <TaskList>
- Quando lo status del Workflow è cambiato in CLOSED

Ogni aggiornamento può essere effettuato utilizzando l'XDS document replace, se ci ritrova nel contesto XDS. La serie di step da seguire sono:

- Aggiornare il Workflow Document con le nuove informazioni richieste. Questo richiede spesso la modifica dell'elemento <taskData>. Ma si può verificare anche un aggiunta di una nuova task all'interno della <taskList> o di un nuovo <taskEvent> di una task.
- Utilizzare l'operazione XDS replace per sostituire il documento modificato alla versione precedente
- Questa operazione può generare un errore dall'XDS Registry se un altro attore ha effettuato un replace prima che l'operazione sia stata completata. Se succede questo, un nuovo contenuto viene realizzato e aggiornato. Questa tipologia di errore è molto rare in quanto la velocità di realizzazione delle tasks è generalmente molto più bassa rispetto all'operazione di replace.

Nel caso in cui si operi in un contesto XDM o XDR l'attore che riceve il documento deve realizzare operazioni di aggiornamento simili.

Quando si arriva nell'istante in cui è necessario interrompere i lavori associati ad uno specifico Workflow, è necessario un ultimo processo di aggiornamento che setta lo status del Workflow a CLOSED.

Le regole specifiche per determinare quando un lavoro è stato completato non sono definite dal profilo XDW. Esse devono essere definite dagli altri profili IHE, dall'amministratore locale del Workflow, o da altre sorgenti. Il profilo XDW, inoltre, non definisce le regole che determinano la gestione del lifecycle del documento, ma gli altri profili coinvolti nelle specifiche istanze di Workflow possono richiedere una certa sequenza di task, che il Workflow inizi con un determinato processo, e che il processo di aggiornamento avvenga in risposta a cambiamenti effettuati su specifiche tasks.

7.4.2 Associazione tra le varie versioni del Workflow Document

Un documento clinico può essere utilizzato come riferimento in più Workflow Document diversi, ed in differenti steps dello stesso Workflow Document. Quando si ha conoscenza di un Workflow Document i documenti clinici ad esso correlati sono sempre rintracciabili attraverso il riferimento (attraverso `XDSDocumentEntry.uniqueId` e `homeCommunityId`) tracciato all'interno del Workflow Document stesso nelle sezioni di input e output delle differenti task.

L'utilizzo di una cartella per raggruppare le diverse versioni del Workflow Document è necessario per avere un id fissato che identifichi l'intero workflow. Dato che il Workflow Document può essere aggiornato più volte (più precisamente ad ogni step evolutivo del workflow), il suo uid/id non è utile per mantenere un riferimento fisso. Posizionando tutte le versioni dello stesso Workflow Document all'interno della stessa cartella permette l'utilizzo del `FolderId` come un link fisso verso lo specifico Workflow. Un Workflow Document può essere contenuto all'interno di una sola Workflow Content Folder, nella quale è presente un'unica versione del documento in uno stato "approved" e tutte le versioni precedenti nello stato "deprecated". Se un workflow genera un sotto-workflow saranno presenti due distinte folders, una per ogni workflow. La relazione tra i due workflow è garantita all'interno del Workflow Document utilizzando l'`XDSFolder.uniqueId` come output della task del parent Workflow Document e come input della prima task del child Workflow Document.

7.4.3 Creazione di un nuovo Workflow

Quando viene completato il primo step di un nuovo workflow, l'**XDW Content Creator** deve:

- creare la prima versione del Workflow Document

In seguito l'**XDS Document Source** raggruppato con l'attore precedente deve:

- creare un nuovo Workflow Context Folder
- inviare il Workflow Document all'**XDES Document Repository**, all'interno della nuova cartella creata utilizzando **ITI-41 Provide and Register Document Set-b**

7.4.4 Aggiornamento il Workflow Document

Per ogni step seguente nel Workflow l'XDW Content Updater deve:

- ottenere la versione più recente del Workflow Document, che rappresenta l'unica versione nello stato approvato contenuta all'interno della cartella (utilizzando l'XDS Document Consumer raggruppato).
- aggiornare il contenuto del Workflow Document (aggiungendo una nuova task o aggiungendo ad una task pre-esistente un elemento <taskEvent>)
- ri-registrare la nuova versione del Workflow Document attraverso un replace della versione precedente (utilizzando l'XDS Document Source raggruppato con l'XDW Content Updater)

Questa nuova versione è automaticamente aggiunta alla cartella corretta attraverso le normali regole XDS per il replacement in una folder.

In un infrastruttura per la condivisione (per esempio XDS) due differenti Content Updaters potrebbero essere nella situazione di apportare contemporaneamente un aggiornamento allo stesso Workflow Document. In questo caso entrambi gli updatator (definiti come Content Updatator A, Content Updatator B) ritirano il Workflow Document (per esempio il WD a cui è associat il document id 1) e lo modificano.

- Content Creator A trasforma in deprecated la versione precedente del documento, e pubblica successivamente la versione aggiornata con il nuovo documentId (per esempio documentId=2)
- Quando il Content Creator B prova a sostituire lo stesso Workflow Document (con documentId=1) con la sua versione aggiornata la transazione viene impedita, perchè il documentId=1 è diventato deprecated in quanto già sostituito con il documentId=2.
- Il Content Creator B ritira dunque la versione corrente del Workflow Document (documentId=2) e la aggiorna con la nuova versione del documento con documentId=3

7.4.5 Ottenere tutti i documenti associati al Workflow

La versione più recente del Workflow Document può essere consultato durante ogni punto del Workflow. La versione del Workflow Document con lo stato approvato, contiene le informazioni

più recenti relative al Workflow ed alle tasks correlate. Quindi un XDW Content Consumer deve solamente analizzare il documento nello stato approvato per ricavare tutte le informazioni necessarie. Ogni Workflow Document contiene dettagli di ogni task che è stata eseguita. Una task o un `<taskEvent>` include il riferimento (`XDSDocumentEntry.uniqueId` e `homeCommunity.Id`) da zero a più input o output documenti clinici. Questi documenti possono essere ottenuti attraverso l'utilizzo delle potenzialità offerte dal profilo XDS, oppure XDR o XDM se sono utilizzati.

7.4.6 Utilizzo del eventCodeList per gestire lo stato del Workflow Document

E' necessario avere una modalità per definire lo stato del Workflow ad ogni step. Questo stato può essere OPEN o CLOSED. Questo stato deve essere definito per ogni step del Workflow. Settando questo metadato a OPEN, l'autore dello step indica che, per la definizione del Workflow e dello step eseguito, necessita ulteriori step per essere completato.

Settando questo metadato a CLOSED, l'autore dello step indica che per la definizione del Workflow e dello step eseguito non sono necessari e non possono essere eseguiti ulteriori steps.

Questo codice deve essere presente per tutti gli XDW Workflow Document e deve essere definito anche per ogni task all'interno dell'eventCodeList, un elemento che permette di tracciare i cambiamenti di stato di una task. L'utilizzo dei codici permette l'utilizzo di query specifiche per individuare Workflow attivi con specifiche proprietà. I codici per eventCodeList possono essere OPEN o CLOSE.

7.5 Metadati: XDS e XDS Folder

I seguenti metadati devono essere utilizzati per descrivere il Workflow Document all'interno di un Affinity Domain XDS. XDW non definisce nuovi metadati e tutti quelli utilizzati sono quelli comuni definiti in ITI-3:4.1.5.

Attributo XDSDocumentEntry	Definizione
author	Rappresenta l'uomo o la macchina che ha prodotto il documento Rappresenta il soggetto che ha aggiornato più recentemente il Workflow Document
authorInstitution (sub-attribute of author)	Nessuna richiesta particolare per il Workflow Document
authorPerson (sub-attribute of author)	Nessuna richiesta particolare per il Workflow Document
authorRole (sub-attribute of author)	Nessuna richiesta particolare per il Workflow Document
authorSpecialty (sub-attribute of author)	Nessuna richiesta particolare per il Workflow Document
availabilityStatus	Nessuna richiesta particolare per il Workflow Document
classCode	<i>Questo codice deve essere definito da LOINC</i>
classCode DisplayName	Dovrà assumere il valore XDW_Workflow_Document
comments	Nessuna richiesta particolare per il Workflow Document
confidentialityCode	Nessuna richiesta particolare per il Workflow Document
creationTime	Nessuna richiesta particolare per il Workflow Document
entryUUID	Nessuna richiesta particolare per il Workflow Document
eventCodeList	Rappresentano gli atti clinici principali. Per il Workflow Document questo campo può assumere due valori OPEN, CLOSE.
eventCodeListDisplay Name	Nessuna richiesta particolare per il Workflow Document
formatCode	urn:IHE:iti:xdw:2011:workflowDoc codesystem: 1.3.6.1.4.1.19376.1.2.3
hash	Nessuna richiesta particolare per il Workflow Document
healthcareFacility TypeCode	Nessuna richiesta particolare per il Workflow Document
healthcareFacility TypeCodeDisplay Name	Nessuna richiesta particolare per il Workflow Document
homeCommunityId	Nessuna richiesta particolare per il Workflow Document
languageCode	Nessuna richiesta particolare per il Workflow Document
legalAuthenticator	Nessuna richiesta particolare per il Workflow Document
contentType	Nessuna richiesta particolare per il Workflow Document

Attributo XDSDocumentEntry	Definizione
patientId	Nessuna richiesta particolare per il Workflow Document
practiceSettingCode	Nessuna richiesta particolare per il Workflow Document
practiceSettingCode DisplayName	Nessuna richiesta particolare per il Workflow Document
repositoryUniqueId	Nessuna richiesta particolare per il Workflow Document
serviceStartTime	Descrive l'istante in cui il servizio descritto è stato iniziato.
serviceStopTime	Nessuna richiesta particolare per il Workflow Document
size	Nessuna richiesta particolare per il Workflow Document
sourcePatientId	Nessuna richiesta particolare per il Workflow Document
sourcePatientInfo	Nessuna richiesta particolare per il Workflow Document
title	Nessuna richiesta particolare per il Workflow Document
typeCode	Descrive il tipo di documento. I vari domini IHE o Affinity Domain XDS devono definire un valore per i differenti tipi di Workflow.
typeCodeDisplay Name	Nessuna richiesta particolare per il Workflow Document
uniqueId	Nessuna richiesta particolare per il Workflow Document
URI	Nessuna richiesta particolare per il Workflow Document

I metadati devono essere utilizzati anche per descrivere l'XDS Folder che contiene tutte le versioni dello stesso Workflow Document e possibilmente i documenti clinici correlati ad esso. XDW in particolare non definisce nuovi metadati e tutti quelli utilizzati sono i comuni metadati usati nello scenario XDS Folder e3 specificati in ITI TF-3:4.1.9. A livello di XDS Affinity Domain è necessario definire certi codici specifici per identificare il tipo di folder utilizzato.

XDSFolder Attribute	Definizione
availabilityStatus	Nessuna richiesta particolare per il Workflow Document
codeList	Contiene il set di codici che specificano il tipo di attività clinica risultato dell'inserimento del documento nella XDS Folder. Un codice deve essere assegnato da IHE-ITI
codeListDisplayName	Il valore deve essere "XDW Workflow Context Folder".
comments	Nessuna richiesta particolare per il Workflow Document
entryUUID	Nessuna richiesta particolare per il Workflow Document
homeCommunityId	Nessuna richiesta particolare per il Workflow Document
lastUpdateTime	Nessuna richiesta particolare per il Workflow Document
patientId	Nessuna richiesta particolare per il Workflow Document
title	Nessuna richiesta particolare per il Workflow Document
uniqueId	Contiene l'identificatore univoco per l'intero Workflow

8. Conclusioni

8.1 Obbiettivi raggiunti

L'approccio offerto dall'applicazione dei linguaggi descrittivi di Workflow era, ad oggi, inutilizzato all'interno del campo della sanità digitale. In particolare non erano definite linee guida condivise per la definizione e la gestione dei cosiddetti Workflow Clinici. A questo si sommavano i forti limiti applicativi legati alla rigidità delle norme che disciplinano la gestione dei documenti clinici.

In questa sezione verranno dunque analizzati i quesiti risolti grazie all'analisi svolta, con particolare riguardo alle nuove conoscenze apprese e soprattutto, al nuovo approccio che è possibile utilizzare per risolvere i problemi connessi alla digitalizzazione dei processi sanitari.

Nelle prime fasi del lavoro svolto infatti, è stato fondamentale apprendere le tematiche connesse all'applicazione dei servizi ICT alla sanità, focalizzando l'attenzione sull'approccio offerto dal processo IHE. Attraverso lo studio dell'applicazione delle linee guida IHE a progetti in fasi di sviluppo avanzate (come HO e in particolare DOGE), è stato possibile comprendere l'approccio alle problematiche offerte da IHE, individuando le principali cause dei problemi emersi all'interno di queste applicazioni costituendo il background specifico risultato poi fondamentale durante il processo di definizione della struttura del Workflow Document XDW. L'analisi di queste osservazioni è diventata lo stimolo specifico per procedere allo sviluppo del profilo XDW ad opera del consorzio Arsenal.IT nelle figure di Ing. Luca Zalunardo e Arianna Cocchiglia.

Successivamente è stato possibile analizzare il processo di definizione di un Workflow (che nella applicazione specifica verrà definito Workflow Clinico), con tutte le potenzialità offerte da questo tipo di approccio in termini di controllabilità, sicurezza, affidabilità ed efficienza. Questo è avvenuto attraverso l'analisi dei linguaggi descrittivi di Workflow presenti in letteratura, con l'obiettivo ultimo di individuare una soluzione adatta agli scopi prefissati. Questo processo di analisi non si è focalizzato esclusivamente sui linguaggi espressamente utilizzati in campo sanitario, ma si è strutturato in un esame a più ampio spettro dal quale è stato possibile osservare forti limiti nell'applicabilità degli standard descrittivi di Workflow individuati al nostro caso specifico. Solamente lo standard WS-HumanTask è risultato avere delle caratteristiche utili alla risoluzione dei problemi individuati, rendendolo lo standard di riferimento per la definizione della seconda versione del Workflow Document, la quale prevede la commistione di diversi linguaggi, ricavando

in questo modo un documento in grado di superare i limiti che comunque presentano i singoli linguaggi utilizzati.

Attraverso un approccio più conservativo si è provato, in prima istanza, a definire una possibile struttura per il workflow Document mediante l'utilizzo dello standard CDA HL7, che permette la definizione della struttura dei documenti clinici mediante un linguaggio XML. Questo ha permesso di esplorare le potenzialità dello standard CDA, arrivando alla definizione della prima versione del Workflow Document ed esplicitando una possibile struttura per il template del documento. Parallelamente è stato utile mostrare anche l'applicabilità in diversi use-cases di interesse, individuando in questo modo i punti di forza e di debolezza delle due proposte. Questo ha portato allo svolgimento di una breve indagine comparativa attraverso la quale è stato possibile individuare i limiti della versione basata sullo standard CDA, riconducibili soprattutto al livello di integrazione dei sistemi ICT ad oggi presenti, e portando alla scelta della versione basata sullo standard WS-HumanTask come documento di riferimento per il profilo XDW.

Il profilo XDW così definito è ad oggi in fase di public comment all'interno della comunità IHE, in versione quasi definitiva.

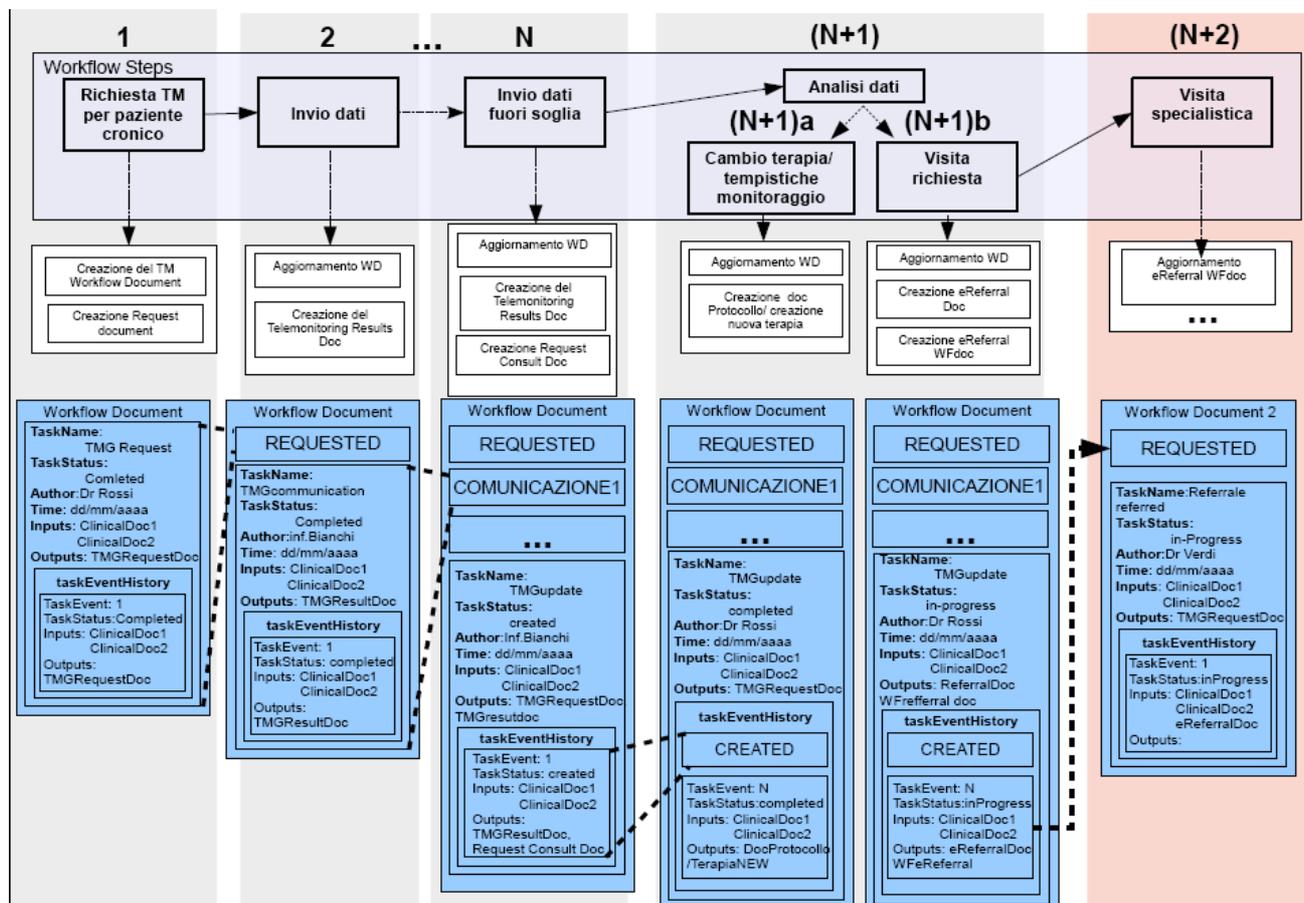
Diversi sono i vantaggi ottenibili dall'applicazione del profilo XDW. Innanzitutto permette di ottenere tutti i benefici connessi alla definizione di un Workflow Clinico, con in più la possibilità di gestire e tracciare il cambiamento di stato dei documenti clinici connessi ad un Workflow paziente-specifico. E' possibile infatti superare il limite imposto dall'utilizzo del metadata availabilityStatus, potendo gestire in questo modo più stati differenti associati ad un documento. Inoltre la sovrastruttura creata dallo standard WS-HumanTask permette di definire un vocabolario dominio-specifico per gli stati possibili per il documento stesso, controllando allo stesso tempo che il Workflow si svolga secondo le regole specifiche definite dal dominio IHE che utilizza il profilo XDW.

Per quanto riguarda il progetto DOGE, il profilo XDW permetterà, durante le prime fasi di applicazione, di tracciare i cambiamenti di stato del documento di prescrizione elettronica, memorizzando all'interno del Workflow Document l'avanzamento del Workflow Clinico associato. All'interno di questo sistema dunque, sarà possibile individuare nuovi punti di controllo per verificare la correttezza degli step eseguiti attraverso l'interazione diretta tra Workflow Document, fascicolo sanitario elettronico del paziente e gli applicativi software in dotazione agli utenti del servizio sanitario.

8.2 Open issues e sviluppi futuri

A seguito di piccoli ritardi accumulati durante la fase di approvazione del profilo stesso infatti, è stato necessario prolungare il periodo di Public Comment fino alla fine di settembre 2011. Il lavoro di progettazione di nuovi applicativi che integrano le nuove linee guida del profilo è già in esecuzione. In seguito all'approvazione definitiva del profilo, le linee guida fornite verranno utilizzate ufficialmente da produttori di software di tutto il mondo per definire soluzioni in grado di superare gli appositi test organizzati da IHE all'interno dell'evento Connetc-a-Thon 2012 di Berna. Per quanto riguarda l'applicazione del profilo XDW al progetto DOGE si prevede che nei prossimi mesi verrà definito il nuovo applicativo di aggiornamento all'infrastruttura già presente, in modo da poter gestire in modo efficace il ciclo prescrittivo digitale di circa 4400 prescrizioni elettroniche prodotte a carico dell'ULSS di Adria. Nei prossimi anni le linee guida su cui si basa il progetto DOGE verranno estese a tutta la regione Veneto e, attraverso le funzionalità offerte dal profilo XDW, sarà possibile gestire circa 60 milioni di prescrizioni elettroniche all'anno.

Fig.8.1: Possibile applicazione futura del profilo XDW al caso del Telemonitoraggio Infermieristico



Dopo questa prima fase di diretta applicazione, si suppone di poter gestire anche casi d'uso più complessi rispetto alla tracciatura dei cambiamenti di stato dei documenti clinici. Sarà possibile infatti estendere l'utilizzo del profilo ad altri progetti relativi a Telesoccorso, Telemonitoraggio e Teleconsulto definiti e già sviluppati sul territorio dal consorzio Arsenal.IT e in fase di progettazione. Nell'immagine 8.1 è mostrato un esempio di applicazione al caso del Telemonitoraggio infermieristico, all'interno del quale si può instaurare un sotto-Workflow connesso ad una richiesta di Teleconsulto. Diventando linea guida per la gestione dei Workflow Clinici di carattere internazionale tuttavia, la portata dei risultati ottenibili è difficilmente immaginabile e di sicuro impatto sull'organizzazione e la razionalizzazione dei processi clinici.

APPENDICE A:

CDA Technical Specifications

Il CDA Hierarchical Description (CDA-HD) è la sorgente definitiva per stabilire le regole di conformità di un documento CDA e da esso deriva la definizione del CDA Schema. Oltre ad aderire alle regole definite dal CDA Schema, un'istanza di documento CDA deve essere conforme alle regole definite nella CDA-HD. La CDA Hierarchical Description è la rappresentazione tabulata della sequenza degli elementi (classi, attributi ed associazioni) che permettono di definire la struttura dell'istanza del documento, senza riferimento alle tecnologie di implementazione (come per esempio XML).

Il CDA-HD serve come sorgente definitiva per definire le regole di conformità che dovrà rispettare il CDA Schema. E' possibile referenziare direttamente il CDA-HD all'interno del documento corrispondente, tramite la stringa "POCD_HD000040".

A.1 CDA header

Lo scopo dell'header è quello di permettere lo scambio dei documenti clinici all'interno e tra le varie istituzioni coinvolte nel processo clinico, facilitando la gestione documentale e la compilazione dei documenti clinici all'interno del ciclo di vita elettronico dell'archivio del paziente. La classe Clinical Document è il punto di inizio del documento CDA, e corrisponde all'elemento XML <ClinicalDocument> che è l'elemento root del CDA stesso.

La classe ClinicalDocument definisce vari attributi che corrispondono all'infrasstructureRoot del RIM (per esempio typeId e templateId). In particolare se ClinicalDocument.templateId viene valorizzato all'interno di un istanza, questo segnala l'imposizione di un set di vincoli per la template del documento, caratterizzandola con un identificatore univoco.

L'attributo ClinicalDocument.typeId è un riferimento esplicito e tecnologicamente neutro al documento CDA, e deve essere valorizzato in questo modo:

-ClinicalDocument.typeId.Root="2.16.840.1.113883.1.3" (rappresenta l'OID dei modelli registrati HL7)

-ClinicalDocument.typeId.Extension="POCD_HD000040" (identificatore univoco per il CDA R.2 Hierarchical Definition)

Gli altri attributi della root ClinicalDocument sono:

- ClinicalDocument.id**: identificatore univoco per l'istanza del documento clinico
- ClinicalDocument.code**: specifica il particolare tipo di documento, e preferibilmente il valore deve essere derivato da LOINC.
- ClinicalDocument.title**: rappresenta il titolo del documento. Deve essere usato nel caso in cui al documento è associato uno specifico e unico nome oppure se deve essere visualizzato dal clinico. Nel caso in cui non venga specificato viene referenziato il valore del ClinicalDocument.code.
- ClinicalDocument.effectiveTime**: rappresenta la data e l'ora in cui il documento è stato creato.
- ClinicalDocument.confidentialityCode**: è un informazione contestuale dell'intero documento CDA, a meno che non venga sovrascritto da un valore annidato nel documento (in particolare nella sezione Context). Indica il livello di riservatezza del documento e può assumere tre valori:
 - N=normal (default)**: regole di riservatezza convenzionali associate alla pratica medica, e solo gli individui autorizzati per scopi medici o di lavoro possono accedere al documento (codeSystem 2.16.840.1.113883.5.25)
 - R=restricted**: possono accedere al documento solo i providers che sono attualmente concorrenti nel processo di cura(codeSystem 2.16.840.1.113883.5.25)
 - V=very restricted**: (codeSystem 2.16.840.1.113883.5.25) accesso molto limitato come dichiarato dalle politiche di privacy definite dallo specifico provider del servizio medico.
- ClinicalDocument.languageCode**: specifica la lingua umana in cui è redatto il documento. Il valore dell'attributo è un identificatore della lingua, definito da IETF (Internet Engineering Task Force) RFC 3096 for Identification of languages, ed. H. Alvestrand (1995). Anche la lingua è una caratteristica contestuale dell'intero documento, a meno che il campo non venga sovrascritto nella sezione annidata Context.
- ClinicalDocument.setId**: rappresenta un identificatore che è comune tra tutte le diverse versioni e revisioni del documento.
- ClinicalDocument.versionNumber**: valore intero che identifica la versione successiva del documento.

La sezione Participants dell'header descrive le classi correlate al evento clinico mediante la partecipazione allo stesso. I vari ruoli possono essere occupati dalla stessa persona fisica o da persone diverse.

- Authenticator**: classe che rappresenta un partecipante il quale attesta l'accuratezza del

documento, ma che non ha la prerogativa di poter attestare questa conformità autenticando legalmente il documento. Ogni documento può avere da uno a molti autenticatori e ad ogni autenticatore è associato un authentication.time.

-Author: classe che rappresenta l'attore, umano o macchina, autore del documento. A volte il ruolo o la funzione dell'autore è indicata nel campo ClinicalDocument.code. In altri casi la specifica del campo ClinicalDocument.code è ripresa o specializzata all'interno del campo author.functionCode o nell'attributo assignedEntity.code, per questo motivo non deve esistere conflitto tra i valori attribuiti a questi campi.

-Custodian: individua l'organizzazione che ha il compito di mantenere lo stato di integrità del documento, preservando le informazioni in esso contenute

-dataEnterer (transcriptionist): rappresenta il partecipante che ha trasformato una nota dettata in un testo.

-informant (source of information): rappresenta le entità che apportano rilevanti informazioni inerenti al caso clinico

-informationRecipient: rappresenta il recipiente che andrà a ricevere una copia del documento. Questo partecipante è l'entità a cui è destinato il documento nel momento in cui viene redatto il documento stesso, e non rappresenta l'insieme delle persone che avranno poi la possibilità di accedere allo stesso. Questa lista non può essere contenuta all'interno del documento CDA ed esula dagli scopi di CDA.

-legalAuthenticator: rappresenta il partecipante che ha legalmente autenticato il documento. Un documento CDA può esistere in uno stato non autenticato, autenticato o legalmente autenticato. Entrambe le autenticazioni richiedono che il documento venga firmato manualmente o digitalmente.

-participant: è utilizzato per rappresentare altri partecipanti non esplicitamente menzionati nelle altre classi, e che sono comunque coinvolti negli atti documentati.

-recordTarget: rappresenta la cartella clinica a cui appartiene il documento. In genere un documento appartiene ad un'unica cartella, in caso contrario è necessario definire più recordTarget participants.

-guardian: è una persona o un'organizzazione che svolge il ruolo di tutore nei confronti del paziente. Questo partecipante prende le decisioni cliniche che riguardano il paziente nel caso in cui quest'ultimo non ne sia in grado. Se non è esplicitamente definito il paziente prende da solo le proprie decisioni mediche.

-responsibleParty: rappresenta un soggetto che ha una responsabilità legale primaria per gli atti documentati. Non è necessariamente presente nell'azione svolta ma interviene nella

stessa attraverso il potere di delega e l'obbligo di rivedere le decisioni insieme ai partecipanti attivi. Si tratta di un soggetto diverso dal legalAuthenticator, in quanto quest'ultimo può o meno essere anche il responsibleParty, ma l'autenticatore svolge un ruolo sulla cartella clinica, trasformando il documento CDA nel suo stato completo.

La sezione relationship dell'header descrive le classi correlate con la root ClinicalDocument attraverso il legame ActRelationship.

-ParentDocument: rappresenta il documento sorgente da cui si ottiene il CDA attuale, attraverso un revisione, aggiunta o trasformazione. ParentDocument.text è modellato come un ED data type e non può essere utilizzato incorporare direttamente il relativo documento.

-RelatedDocument.typeCode: descrive che tipo di aggiornamento è il documento corrente rispetto al parentDocument. Questo campo può assumere tre valori:

-APND(append): il documento corrente è un supplemento di ParentDocument. Questo supplemento è un nuovo documento nel quale è presente il riferimento al parentDocument, e può estendere o modificare le osservazioni fatte nel documento precedente. Entrambi i documenti rimangono a far parte della cartella clinica del paziente ed entrambi rimangono leggibili.

-RPLC(replace): il documento corrente sostituisce il Parentdocument. Il documento corrente è una nuova versione del documento precedente, il quale viene considerato sostituito ma rimane comunque disponibile per risalire allo storico dell'evento clinico interessato e per eventuali errori commessi nella stesura del nuovo documento.

-XFRM(transform): il documento corrente è una trasformazione del ParentDocument. Il documento corrente è ottenuto da una trasformazione del ParentDocument in formato CDA a partire da un formato diverso. La trasformazione deve garantire che il significato delle informazioni contenute nel documento originale non venga alterato.

-Event: è la classe che rappresenta l'atto clinico principale che viene documentato. In certi casi può contenere il valore che è racchiuso anche nel ClinicalDocument.code oppure una sua specializzazione. In nessun caso comunque è possibile avere un conflitto tra i valori contenuti nei due campi.

-Order: rappresenta la classe degli ordini per indagini successive o specialistiche, contenute all'interno del documento, che verranno poi trasformati in event nelle versioni successive del documento stesso.

-Consents: classe che indica i consensi associati al documento (consenso alle azioni cliniche

e consenso al trattamento dei dati da parte di terzi). I consensi associati al documento CDA sono fondamentali per definire la completezza del documento stesso, il quale necessita del `Consent.statusCode="Completed"` per essere approvato.

-CurrentEncounter: è una classe opzionale, che rappresenta le informazioni correlate all'incontro medico durante il quale avviene l'atto clinico documentato. I documenti clinici non sono generati necessariamente nel momento in cui avviene l'incontro, ma nel momento in cui il clinico ha la necessità di aggiornare la valutazione sull'evento clinico.

A.2 CDA Body

Il body del documento CDA può essere strutturato secondo un linguaggio di markup oppure può essere non-strutturato.

A.2.1 Body non strutturato

Nel caso di un Body non strutturato la classe **NonXMLBody** rappresenta il corpo del documento che si presenta in formati diversi rispetto al linguaggio XML. Questa classe permette di distinguere il body dall'header del documento CDA. Il body può ridefinire attraverso i campi `NonXMLBody.confidentialityCode` il livello di riservatezza che era definito per l'intero documento. Il campo **NonXMLBody.text** è utilizzato per referenziare dati memorizzati esternamente al documento CDA, invece di codificarli direttamente in-linea. Per rendere il body visualizzabile è necessario un tool software in grado di riconoscere la strutturazione del blob e il particolare MIME media type utilizzato.

A.2.2 Body strutturato

La classe **StructuredBody** rappresenta il body di un documento CDA che è composto da una o più sezioni (document section). Anche in questo caso è possibile ridefinire il livello di riservatezza della porzione di documento racchiusa nel body del documento.

Le sezioni del documento possono essere annidate, contengono CDA entries e possono sovrascrivere proprietà contestuali che si propagano dall'header. La classe **Section** permette di definire le varie porzioni in cui è suddiviso il corpo del documento, utilizzandole per facilitare la navigazione umana all'interno delle informazioni stesse, garantendo consistenza e organizzazione al contenuto. I principali campi associati ad una section sono:

- Section.id:** identificatore univoco per l'istanza della section.
- Section.code:** il codice che specifica la tipologia particolare di section (i valori sono preferibilmente presi dall'elenco LOINC).
- Section.title:** rappresenta l'etichetta che viene associata alla section. Se viene valorizzato all'interno del documento, il contenuto deve essere visualizzato come parte del contenuto narrativo del body
- Section.text:** viene utilizzato per memorizzare la parte narrativa che deve essere visualizzato (rappresenta il “narrative block” della section).
- Section.confidentialityCode:** permette di sovrascrivere le impostazioni di riservatezza a livello della specifica section.
- Section.languageCode:** permette di sovrascrivere la scelta della lingua principale con cui viene redatto il documento a livello della specifica section.

All'interno della section possono essere definiti i participants coinvolti nello specifico atto descritto nella particolare sezione. Tra questi si possono individuare le classi **author** e **informant** che permettono di sovrascrivere i valori che si sono propagati dall'header. Un'altra classe di partecipanti all'atto clinico è definita da **subject**, che rappresenta il target principale delle entries memorizzate nel documento. Spesso il subject corrisponde con il valore associato al RecordTarget, ma può anche assumere valori diversi (per esempio un campione istologico o un feto). Il subject può essere definito sia all'interno delle section che direttamente all'interno delle entries, e si propaga all'interno delle successive annidazioni a meno di sovrascritture.

Il costrutto **component** viene utilizzato per annidare una section all'interno di un'altra section, in modo che una section annidata appaia come una component della section più esterna.

La relazione tra la componente narrativa della section (Section.text) e le entries del documento sono codificate attraverso la classe **entry**. Il blocco narrativo di ogni section, unito con il contenuto multimediale referenziato nello stesso blocco, comprende completamente il contenuto autenticato della section.

In termini di relazioni tra blocco narrativo ed entries CDA si può definire un caso specifico ed un caso generico:

- Specific entry/narrative relationship:** la componente narrativa deriva completamente dalle CDA entries. Quando un report che consiste interamente in un contenuto di dati strutturati viene trasformato in un documento CDA l'applicazione che codifica i dati deve garantire che il contenuto autenticato del documento stesso sia una rappresentazione corretta e completa dei dati originari contenuti nel documento sorgente. Questo assicura che il contenuto del narrative block rappresenti l'intero contenuto autenticato della section. In

questo caso il narrative block non contiene nessun informazione che non sia presente anche nelle entries. In questo caso ActRelationship.TypeCode="DRIV" (is derived from) e indica all'applicazione ricevente il CDA che:

- la sorgente per il narrative block sono le entries
- il contenuto delle entries e del narrative block è equivalente

-General entry/narrative relationship: in questo caso la componente narrativa è il contenuto originale del documento. Le CDA entries sono derivate dal contenuto del narrative block, attraverso una codifica effettuata attraverso sistemi di analisi del testo e attraverso l'utilizzo di personale umano, producendo come output sia entries che un report di testo. Il metodo di estrazione delle informazioni deve essere indicato dagli attori coinvolti nel processo di codifica. Non esistono in questo caso asserzioni possibili sull'equivalenza del contenuto di entries e narrative block. In questo caso il campo ActRelationship.TypeCode="COMP" (component).

Può essere stabilito un collegamento tra le CDA entries e il corrispondente contenuto del narrative block e vice versa. L'elemento <content> contiene un identificatore locale opzionale (rappresentato come un attributo XML ID), che costituisce il target di un riferimento. La componente OriginalText presente nelle CDA entries può esplicitamente riferirsi all'identificatore del narrative block, indicando così il testo originale a cui è associato tale attributo.

```
<section>
  <code code="10153-2" codeSystem="2.16.840.1.113883.6.1"
    codeSystemName="LOINC"/>
  <title>Past Medical History</title>
  <text>
There is a history of <content ID="a1">Asthma</content>
  </text>
  <entry>
    <Observation>
      <code code="84100007"
        codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        displayName="history taking (procedure)"/>
      <value xsi:type="CD" code="195967001"
        codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        displayName="Asthma">
        <originalText>
          <reference value="#a1"/>
        </originalText>
      </value>
    </Observation>
  </entry>
</section>
```

Questo collegamento tra entries e narrative block non è necessario, ma diventa fondamentale dove

si vuole rappresentare il testo originale in CDA entries codificate. L'elemento `<renderMultimedia>` contiene l'attributo `referencedObject` (rappresentato come un attributo XML IDREFS) il cui valore deve essere uguale al XML ID di un `ObservationMedia` o al valore di uno o più `RegionOfInterest` delle CDA entries dello stesso documento. Un attributo `MMID` (multimedia identifier), del tipo XML ID, viene aggiunto a `ObservationMedia` e `RegionOfInterest` all'interno del CDA Schema.

```
<section>
  <code code="8709-8" codeSystem="2.16.840.1.113883.6.1"
        codeSystemName="LOINC"/>
  <title>Skin exam</title>
  <text>
    Erythematous rash, palmar surface, left index finger.
    <renderMultiMedia referencedObject="MM1"/>
  </text>
  <entry>
    <ObservationMedia MMID="MM1">
      <id root="10.23.4567.345"/>
      <value xsi:type="ED"
            mediaType="image/jpeg">
        <reference value="left_hand_image.jpeg"/>
      </value>
    </ObservationMedia>
  </entry>
</section>
```

Il campo `Section.text` utilizzato per la componente narrativa delle informazioni che devono essere visualizzate in seguito. Il modello del contenuto di questa sezione contiene:

-**<content>**: utilizzato per racchiudere una stringa di testo, suggerendo anche le caratteristiche della visualizzazione della stessa. Questo elemento può essere utilizzato in modo ricorsivo, permettendo di incapsulare una stringa all'interno di un'altra. Questo elemento contiene l'attributo opzionale **emphasis** che può essere valorizzato con “bold”, “underline”, “italics” o “yes” e può essere utilizzato per indicare il formato del testo presente nel documento sorgente. In ogni caso il ricevente del documento non è obbligato a visualizzare il testo con lo stile proposto. L'attributo `revised`, che può essere valorizzato con “insert” o “delete”, permette di memorizzare quali interventi di modifica del narrative block sono stati approntati nell' ultima versione del documento rispetto la precedente. Se questo campo viene valorizzato è necessario utilizzarlo congiuntamente al sistema di tracciatura delle revisioni proposto da CDA. I cambiamenti apportati alle informazioni correlate alla cura del paziente devono comunque essere visibili, e il documento deve poter trasmettere gli attributi revisionati. Il ricevente del documento è tenuto a visualizzare le modifiche

apportate al documento attraverso uno stile di rappresentazione diverso, in grado di distinguere agevolmente e in modo immediato alla lettura umana le componenti eliminate o corrette.

-<**link**>: è un meccanismo generico per inserire riferimenti esterni, e contiene il singolo elemento <linkHtml>, che permette di realizzare funzionalità simili all'HTML anchor tag. I contenuti multimediali che sono parte integrante del documento, e quindi parte del contenuto verificabile dello stesso, richiedono di essere referenziati tramite l'utilizzo di ObservationMedia CDA entries, le quali sono referenziate nel narrative block tramite l'elemento <renderMultimedia>; per i riferimenti che non sono parte integrante del documento è possibile utilizzare link.

-<**sub**> e <**sup**>: sono gli elementi che permettono di definire pedici e apici. Il soggetto che riceve il documento è tenuto a distinguere nella visualizzazione queste tipologie di carattere.

-<**br**>: è l'elemento utilizzato per indicare interruzione di linea, differisce da <paragraph> perchè non ha contenuto e viene visualizzato senza utilizzare una linea vuota.

-<**renderMultiMedia**>: permette di referenziare elementi esterni che hanno contenuto integrante al documento stesso. Questo elemento ha un opzionale elemento <caption> che contiene l'attributo referencedObject (attributo XML IDREFS), il cui valore deve essere uguale al valore del XML ID dell'ObservationMedia o al valore di RegionOfInterest delle CDA entries presenti nel medesimo documento.

-<**paragraph**>: è simile all'elemento paragraph in HTML, il quale permette di dividere il blocco narrativo in strutture logiche consistenti. Questo elemento possiede un opzionale elemento <caption>, che se presente deve essere posizionato prima di ogni altro elemento.

-<**list**>: è simile all'elemento list in HTML. Ogni elemento <list> può contenere uno o più elementi <item>. <list> possiede un opzionale elemento <caption>, che se presente deve essere posizionato prima di ogni altro elemento. L'attributo **listType** è necessario e stabilisce se la lista è ordinata o meno (disordinata è il valore di default). Liste ordinate vengono visualizzate tramite elenchi numerati, mentre se sono disordinate mediante elenchi puntati, anche se non si tratta di un requisito che il ricevente deve rispettare.

-<**table**>: è simile alla table di HTML, ed ha solo scopi rappresentativi.

-<**caption**>: è un'etichetta che può essere riferita a paragrafi, liste, elementi di liste, tabelle e celle delle tabelle. Può essere utilizzato anche all'interno dell'elemento <renderMultimedia> per indicare un'etichetta alle entries ObservationMedia e

RegionOfInterest referenziate. Una <caption> contiene link e può essere codificata con l'elemento opzionale <localCaptionCode>. Se viene utilizzato <localCaptionCode> deve essere posizionato subito dopo l'elemento <caption> e non può essere ripetuto. I codici definiti per gli elementi caption costituiscono solo una convenienza per l'implementazione locale e non forniscono un'informazione manipolabile e interpretabile dal ricevente il documento CDA.

Le CDA entries rappresentano la componente strutturata e machine-processable all'interno di una section del documento. Un documento clinico contiene una grandissima varietà di contenuti, che richiedono di conseguenza di essere completamente codificati. L'attuale set di possibili CDA entry è stato sviluppato in risposta dei principali requisiti imposti dal campo di applicazione di CDA. Invece di creare una specifica entry per ogni scenario possibile, requisiti simili sono fusi per creare entries più ampie, che poi possono essere adattate alla specifica implementazione.

-Act: Rappresenta una classe derivata dal RIM. Viene utilizzata quando le classi più specifiche sono inadeguate. L'attributo booleano Act.negationInd quando è settato a true indica che l'insieme degli attributi descrittivi della classe act viene negato (senza negare Act.id, Act.moodCode e participants). Un atto “negato” indica che l'autore dell'atto stesso nega che si sia verificato, prendendosi la responsabilità della dichiarazione.

-Encounter: classe derivata dal RIM PatientEncounter e utilizzata per rappresentare incontri correlati a quello descritto (per esempio visite di follow-up). La classe CurrentAct nell'header descrive l'incontro che porta alla stesura del documento, mentre la classe Encounter nel body descrive altri incontri correlati.

-Observation: classe che permette di rappresentare osservazioni codificate. Un'osservazione può avere da zero a molte relazioni referenceRange che permettono di correlare una particolare istanza della classe Observation con un elemento della classe ObservationRange, dove il range di valori per una particolare osservazione può essere definito esplicitamente.

-ObservationMedia: classe che rappresenta oggetti multimediali che sono parte logica del documento corrente. Questa classe permette di inviare oggetti multimediali mediante riferimenti. Non essendo legittima la trasmissione inline di oggetti multimediali, l'utilizzo ObservationMedia.value.BIN non è permesso. La distinzione tra ObservationMedia e ExternalObservation è che le entries ObservationMedia sono parte del contenuto attestato del documento mentre ExternalObservation no.

-Organizer: questa classe può essere utilizzata per creare raggruppamenti arbitrari di CDA entries che condividono un contesto comune. Questa classe può essere utilizzata in modo

ricorsivo.

-Procedure: classe che descrive le procedure mediche eseguite (anche in questo caso è disponibile Procedure.negationInd)

-RegionOfInterest: rappresenta la regione di interesse di un'immagine su cui è focalizzata l'azione clinica.

-SubstanceAdministration: classe utilizzata per eventi correlati alla terapia e alla anamnesi dei medicinali assunti. L'attributo SubstanceAdministrator.priorityCode permette di definire la priorità di somministrazione delle sostanze. Altri campi che descrivono come gestire la terapia sono doseQuantity, rateQuantity, maxDoseQuantity e effectiveTime.

```
<SubstanceAdministration moodCode="RQO">
  <effectiveTime xsi:type="IVL_TS">
    <low value="20020101"/>
    <high value="20020201"/>
  </effectiveTime>
  <effectiveTime xsi:type="PIVL_TS"
  operator="A">
    <period value="12" unit="h"/>
  </effectiveTime>
  <routeCode code="PO"
  codeSystem="2.16.840.1.113883.5.112"
  codeSystemName="RouteOfAdministration"/>
  <consumable>
    <manufacturedProduct>
      <manufacturedLabeledDrug>
        <code code="318821008"
        codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        displayName="
        Captopril 25mg tablet"/>
      </manufacturedLabeledDrug>
    </manufacturedProduct>
  </consumable>
</SubstanceAdministration>
```

-Supply: classe che è utilizzata per rappresentare lo scambio di materiale da un'entità ad un'altra. Un prodotto dispensato è associato con l'atto supply attraverso un product participant, che connette lo stesso ruolo ManufacturedProduct utilizzato per SubstanceAdministrator.

I participant presentati nell'header possono essere sovrascritti nel body (classi author, consumable, informant, participant). In particolare un participant è un'entità che occupa uno dei possibili ruoli (ParticipantRole class). L'entità che svolge un ruolo è un device (Device class) o un'altra entità (playingEntity class).

-performer: E' una persona, un dipartimento o un'organizzazione che ha il compito di realizzare di una particolare azione (Act).

-specimen: è una parte di un'entità che è il target di una specifica indagine clinica, ed è utilizzato quando l'indagine è eseguita su un oggetto derivato dal paziente.

-subject: può essere descritto in una sezione del documento CDA e propagarsi all'interno delle sezioni annidate.

Possono essere definite delle classi che descrivono le relazioni tra varie entries e tra entries e oggetti esterni al documento CDA:

-component: ha una sorgente tra gli Organizer e un target che è un'altra entry.

-precondition: classe utilizzata per esprimere una condizione che vede essere valutata true prima che un'altra attività si possa svolgere

-entryRelationship: CDA modella e identifica diversi modelli e legami per descrivere lo scenario dei riferimenti. Questo scenario permette di legare tra loro diverse entry appartenenti allo stesso documento, oppure legare entry con oggetti esterni.

-reference: CDA entris possono contenere riferimenti ad oggetti esterni. Questi documenti esterni non sono parte del contenuto autenticato del documento. Ogni oggetto è caratterizzato da un identificatore ed un codice, e contiene l'attributo RIM Act.text, che viene utilizzato per memorizzare l'URL e il MIME type dell'oggetto. Gli oggetti esterni hanno usualmente moodCode="EVN". La classe reference contiene l'attributo reference.separatableInd, che indica se il documento sorgente può essere interpretato indipendentemente dal target del CDA. Se l'attributo è "false" il pacchetto scambiato dovrebbe includere il target del riferimento in modo che il recipient sia in grado di visualizzarlo. Le classi target per la relazione reference sono: ExternalAct, ExternalDocument, ExternalObservation e ExternalProcedure.

APPENDICE B:

Tecnical Specifications BPEL4People

B.1 Design del linguaggio

L'estensione BPEL4People è definita in modo che sia composta dal un livello di specificazione più alto del linguaggio BPEL, in modo che il linguaggio possa utilizzare, se necessario, tutte le caratteristiche offerte da BPEL. Allo stesso modo Tutti gli elementi e gli attributi definiti nell'estensione sono resi disponibili per qualsiasi versione eseguibile e astratta dei processi BPEL. In particolare il linguaggio definisce un set di attributi e elementi in grado di coprire i differenti e complessi pattern di interazioni umane che si possono realizzare durante l'esecuzione di un processo.

Un processo BPEL4People deve necessariamente utilizzare elementi BPEL4People extension e elementi definiti dallo spazio di nomi WS-HumanTask. L'elemento `<b4p:humanInteractions>` è opzionale e contiene la dichiarazione degli elementi che provengono dallo spazio di nomi WS-HumanTask che sono: `<htd:logicalPeopleGroups>`, `<htd:tasks>` and `<htd:notifications>`.

L'elemento `<htd:logicalPeopleGroup>` specifica il gruppo logico di persone utilizzato nella definizione in-linea della human task o della people activity. L'attributo `name` specifica il nome del gruppo, e deve essere un identificatore univoco all'interno di tutti i nomi definiti all'interno dell'elemento `<b4p:humanInteractions>`.

L'elemento `<htd:task>` è utilizzato per fornire una definizione in-linea della human task. La sintassi e la semantica dell'elemento è specificatamente definita all'interno dello standard WS-HumanTask. L'attributo `name`, specifica il nome della task, e deve essere un identificatore univoco della task all'interno dell'elemento `<htd:tasks>`.

L'elemento `<htd:notification>` è utilizzato per fornire la definizione di una notifica in-linea. La sintassi e la semantica dell'elemento sono definite dalla specifica WS-HumanTask. L'attributo `name`, specifica il nome della notifica e deve essere unico tra i nomi di tutte le notifiche definite all'interno dell'elemento `<htd:notifications>`.

L'elemento `<b4p:peopleAssignments>` è utilizzato per assegnare delle persone ad un ruolo

generico correlato al processo.

Il tipo di attività `<b4p:peopleActivity>` è introdotto nella specificazione BPEL4people ed è utilizzato per modellare interazioni umane all'interno dei processi BPEL. Questa tipologia di attività è inclusa nelle attività BPEL `<bpel:extensionActivity>`, la quale è utilizzata come incapsulatore dell'elemento.

```
<bpel:process ...
  ...
  xmlns:b4p="http://www.example.org/BPEL4People"
  xmlns:htd="http://www.example.org/WS-HT"
  ...
  <bpel:extensions>
    <bpel:extension
      namespace="http://www.example.org/BPEL4People"
      mustUnderstand="yes"/>
    <bpel:extension
      namespace="http://www.example.org/WS-HT"
      mustUnderstand="yes"/>
  </bpel:extensions>
  <bpel:import
    importType="http://www.example.org/WS-HT" .../>
  ...
  <b4p:humanInteractions?>
    <htd:logicalPeopleGroups/?>
      <htd:logicalPeopleGroup name="NCName">+
        ...
      </htd:logicalPeopleGroup>
    </htd:logicalPeopleGroups>
    <htd:tasks?>
      <htd:task name="NCName">+
        ...
      </htd:task>
    </htd:tasks>
  <htd:notifications?>
    <htd:notification name="NCName">+
      ...
    </htd:notification>
  </htd:notifications>
</b4p:humanInteractions>
<b4p:peopleAssignments>
  ...
</b4p:peopleAssignments>
  ...
  <bpel:extensionActivity>
    <b4p:peopleActivity name="NCName" ...>
      ...
    </b4p:peopleActivity>
  </bpel:extensionActivity>
  ...
</bpel:process>
```

Il tipo di attività `<b4p:peopleActivity>` è utilizzato per modellare interazioni umane all'interno di processi BPEL. La nuova attività è inclusa all'interno dell'elemento `<bpel:extensionActivity>` che avvolge la stessa.

```
<bpel:scope ...>
  ...
  <b4p:humanInteractions>?
    ...
  </b4p:humanInteractions>
  ...
  <bpel:extensionActivity>
    <b4p:peopleActivity name="NCName" ...>
      ...
    </b4p:peopleActivity>
  </bpel:extensionActivity>
  ...
</bpel:scope>
```

B.2 Concetti base del linguaggio

Molti concetti che stanno alla base del linguaggio BPEL4People sono ereditati dallo standard WS-HumanTask.

I ruoli umani generici e connessi al processo, definiscono cosa un individuo o un gruppo di persone a cui è stato assegnato un ruolo possono fare con un'istanza del processo stesso. Questi ruoli completano l'insieme di generic human roles definiti dallo standard WS-HumanTask. Esistono tre differenti ruoli di questo tipo:

-Process Initiator: è la persona associata al triggering dell'istanza del processo nell'istante della sua creazione. E' di solito definito automaticamente dall'infrastruttura del processo, ma questa scelta può essere sovrascritta attraverso un'esplicita assegnazione. E' necessario garantire che runtime esista almeno una persona associata a questo ruolo.

-Process stakeholders: sono le persone che possono influenzare la progressione dell'istanza del processo, per esempio aggiungendo nuove task al processo o semplicemente osservando il progresso dello stesso. Gli scopi di questo soggetto sono molto più ampi rispetto alle linee guida

proposte dallo standard BPEI4people. Il process stakeholder è associato ad un istanza del processo. Se non è definito, il process initiator diventa automaticamente il process stakeholder dell'istanza creata. Deve essere garantito nel runtime che almeno una persona sia associata a questo ruolo.

-Business administrators: sono le persone a cui è permesso amministrare azioni del processo, come per esempio risolvere eventuali deadlines del processo stesso. A differenza di un process stakeholder il business administrator ha un interesse in tutte le istanze dello stesso processo. Se non è definito un business administrator il process stakeholder assume questo ruolo. Durante il runtime è necessario che almeno un soggetto sia associato a questo ruolo.

La sintassi per la definizione dei vari ruoli è presentata di seguito.

```
<b4p:peopleAssignments>
  <htd:genericHumanRole>+
    <htd:from>...</htd:from>
  </htd:genericHumanRole>
</b4p:peopleAssignments>
```

L'elemento astratto genericHumanRole introdotto dallo standard WS-HumanTask è esteso con i seguenti ruoli connessi al processo:

```
<b4p:peopleAssignments>
  <b4p:processInitiator>?
    <htd:from ...>...</htd:from>
  </b4p:processInitiator>
  <b4p:processStakeholders>?
    <htd:from ...>...</htd:from>
  </b4p:processStakeholders>
  <b4p:businessAdministrators>?
    <htd:from ...>...</htd:from>
  </b4p:businessAdministrators>
</b4p:peopleAssignments>
```

All'interno dell'elemento <b4p:peopleAssignments> possono essere utilizzati solo ruoli umani connessi al processo. L'associazione degli individui agli specifici ruoli avviene durante l'inizializzazione del processo. In quanto tale, dunque, l'assegnazione dei ruoli fa parte degli scopi

dell'inizializzazione del processo (che avviene in corrispondenza di un elemento `<bpel:process>` o `<bpel:scope>`) ed influisce sulla riuscita dell'inizializzazione. Di conseguenza se fallisce l'assegnazione fallisce l'intera istanza del processo.

Il responsabile di una task, di una notifica o di un processo è necessario che una persona venga assegnata allo specifico ruolo. Questa assegnazione può avvenire in tre diversi modi:

-attraverso gruppi logici di persone: I gruppi logici di persone definiscono che persone o insiemi di persone possono interagire con una task umana o una notifica. I dettagli di come vengono utilizzati questi gruppi sono definiti nello standard WS-HumanTask. Questi gruppi sono specificati come parte del processo di definizione del business. Possono essere definiti sia a livello del processo sia al momento della definizione degli scopi dello stesso. Questi gruppi possono essere utilizzati come riferimento per l'assegnazione da molteplici attività. Ogni gruppo logico è legato ad una specifica query di persone durante l'esecuzione. Se viene effettuata la stessa richiesta di assegnazione ciò non comporta necessariamente lo stesso risultato, ma solamente il fatto che venga riutilizzata la stessa query.

L'attività `<assign>` del linguaggio BPEL permette di manipolare i valori associati ai gruppi logici. Per assegnare una persona ad un gruppo logico o da un gruppo logico viene utilizzata l'attività `<copy>`.

L'espressione utilizzata da BPEL4People per fare questa assegnazione da un gruppo logico ad un altro è la seguente:

```
<bpel:from b4p:logicalPeopleGroup="NCName">
  <b4p:argument name="NCName" expressionLanguage="anyURI"?>*>
    value
  </b4p:argument>
</bpel:from>

<to b4p:logicalPeopleGroup="NCName"/>
```

Delle varianti a questa struttura possono includere ulteriori elementi `<b4p:argument>`, che permettono di specificare gli argomenti necessari per svolgere la query di persone all'interno del gruppo logico specificato. L'attributo `expressionLanguage` è facoltativo e specifica il linguaggio utilizzato nell'espressione; se non presente viene fissato di default il valore associato allo stesso

attributo nella sezione racchiusa più vicina.

-attraverso processo automatico associando valori costanti: sono applicabili tutte le versioni di assegnazione automatica supportate dallo standard BPEL. In aggiunta è possibile anche questa variante:

```
<htd:genericHumanRole>
  <bpel:from variable="NCName" part="NCName"?>
    ...
  </bpel:from>
</htd:genericHumanRole>
```

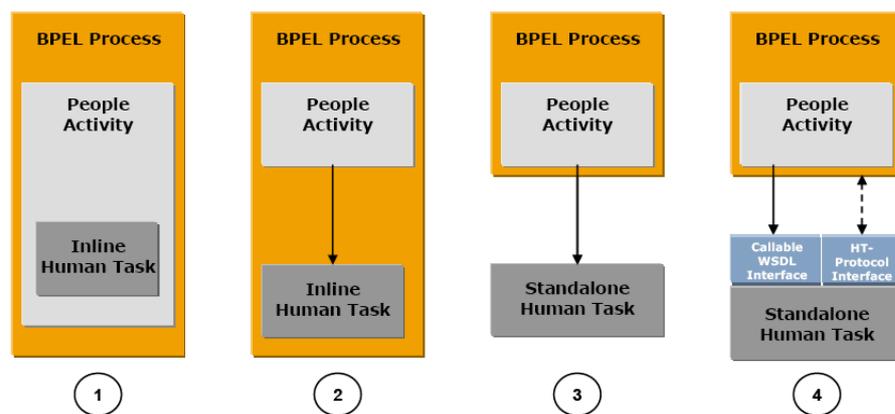
L'attributo variable è utilizzato per associare persone descritte mediante un valore costante associato al business process.

-attraverso espressioni ad-hoc: E' possibile modificare le espressioni di associazione utilizzando espressioni create ad-hoc

Quando viene specificata un'assegnazione è utilizzato il metadato htd:tOrganizationalEntity definito nello standard WS-HumanTask. Questo permette di assegnare ai ruoli specificati o una lista specifica di utilizzatori o un gruppo non specificato di persone.

People Activity è l'attività di base utilizzata per integrare le interazioni umane all'interno del processo BPEL. Queste interazioni possono essere integrate in vari modi (figura 7.1).

Figura 7.1: Inegrazione in BPEL delle People Activity.



Nel primo e nel secondo modello proposti le task umane sono definite come parte interna del processo BPEL. Nel primo caso una task inline è definita come parte di una People Activity. Alternativamente nel secondo modello una task può essere definita come il costrutto base del processo BPEL. In questo caso la stessa task può essere utilizzata per definire più People Activity attraverso composizione ricorsiva di task diverse.

Nel terzo modello viene mostrato l'utilizzo di task a se stanti definite nello stesso ambiente di definizione del processo, senza quindi la specificazione delle interfacce dei WS coinvolti nella task. Di conseguenza l'invocazione di una task avviene attraverso un'implementazione specifica. Questo caso è simile al secondo, tuttavia la definizione della task è totalmente indipendente da qualsiasi processo. Come conseguenza principale la task non ha diretto accesso al contesto del processo.

Nel quarto modello le task sono a se stanti e definite in un ambiente differente da quello utilizzato per il processo. La principale differenza con il caso precedente è legata al fatto che la task è associata ad un WS dotato di interfaccia, interpellata mediante gli specifici protocolli del WS.

In aggiunta il protocollo di coordinazione dello standard WS-HumanTask è utilizzato per realizzare la comunicazione tra task e processo. In questo modo i cambiamenti di stato si possono propagare tra le task, permettendo così la gestione del life-cycle del processo. Con questo approccio, processi definiti all'interno di sistemi BPEL sono trasferibili all'interno di sistemi BPEL4People. I due sistemi sono resi dunque interoperabili.

APPENDICE C:

Template dell'XDW WORKFLOW DOCUMENT (WS-HumanTask-based)

C.1 Definizione del Template del XDW Workflow Document

Nelle tabelle seguenti sono presentati tutti gli elementi obbligatori nella struttura del Workflow Document. Per ogni elemento è definita l'opzionalità e lo standard da cui ne deriva la struttura:

-R: elemento Required (obbligatorio) per il profilo XDW

-O: elemento Optional (opzionale) per il profilo XDW

-<XDW.WorkflowDocument>: elemento del tipo tXDW.WorkflowDocument

Elemento XDW.WorkflowDocument	Standard	Opzionalità	Descrizione
Id	HL7 CDA	R	Document ID
Title	HL7 CDA	O	Titolo rappresentabile
effectiveTime	HL7 CDA	R	Data dell'aggiornamento più recente
ConfidentialityCode	HL7 CDA	R	
languageCode	HL7 CDA	O	
versionNumber	HL7 CDA	O	
Patient	HL7 CDA	R	Informazioni relative al paziente, derivato dal RIM, e limitando il contenuto alle informazioni strettamente necessarie
Author	HL7 CDA	R	Informazioni relative all'autore, elemento derivato dal RIM restringendo il contenuto alle informazioni strettamente necessarie
instanceId	IHE	R	

Elemento XDW.WorkflowDocument	Standard	Opzionalità	Descrizione
sequenceNumber	IHE	R	
workflowStatus	IHE	R	OPEN se sono permesse modifiche al Workflow Document CLOSED se modifiche al documento sono proibite
DefinitionReference	IHE	R	OID che permette di fornire il riferimento ai documenti documenti che definiscono la tipologia di Workflow
TaskList	OASIS_WS - HumanTask	R	Lista di tutte le task e della loro storia.

-<Patient>: elemento del tipo tXDW.Patient

Elemento Patient	Standard	Opzionalità	Descrizione
	HL7 CDA	R	

-<Author>: elemento del tipo tXDW.Author. Deve essere presente almeno uno dei due elementi descritti nella seguente tabella

Elemento Author	Standard	Opzionalità	Descrizione
assignedPerson	HL7 CDA	RC	La persona che aggiorna il Workflow Document
assignedAuthoringDevice	HL7 CDA	RC	La macchina che aggiorna il Workflow Document

-<TaskList> : tipo dell'elemento TaskList

Elemento TaskList	Standard	Opzionalità	Descrizione
XDWTask	OASIS_WS- HumanTask	R	List of tasks

-<XDWTask>: tipo dell'elemento tXDW.Task

Elemento XDWTask	Standard	Opzionalità	Descrizione
taskData	OASIS_WS-HumanTask	R	Description of the current task (status, inputs, outputs, etc.)
taskEventHistory	OASIS_WS-HumanTask	R	History of the changes to the current task (dates, changes, etc.)

-<taskData>: elemento del tipo tTaskInstanceData. XDW aggiunge queste restrizioni alla definizione fatta da OASIS per i taskDetails:

-taskData/input deve contenere l'elemento taskData/input/part per ogni documento che devono essere utilizzati in input alla task. Questo deve essere un elemento del tipo tXDSregisteredDocument

-taskData/output deve contenere un elemento taskData/output/part per ogni documento prodotto durante lo svolgimento della task e che deve essere condiviso. Questo deve essere un elemento del tipo tXDSregisteredDocument

-ogni documento clinico registrato nell'XDS Registry deve essere identificato all'interno degli elementi taskData/output/part, taskData/input/part e taskData/attachment Infos/info attraverso il corrispondente documentID.

Il tipo di elemento tXDSregisteredDocument è un tAttachmentInfo che ha XDSDocumentEntry.uniqueId come attachmentInfo/identifier e un attachmentInfo/accesstype valorizzato con “urn:IHE:iti:2011:xdw:XDSregistered”.

Elemento taskData	Standard	Opzionalità	Descrizione
taskDetails	OASIS_WS-HumanTask	R	
description	OASIS_WS-HumanTask	R	Descrizione testuale
Input	OASIS_WS-HumanTask	R	Riferimento URI a tutti i documenti clinici in input
Output	OASIS_WS-HumanTask	R	Riferimento URI a tutti i documenti in output
Fault	OASIS_WS-HumanTask	O	Descrizione del fallimento della task

Elemento taskData	Standard	Opzionalità	Descrizione
renderings	OASIS_WS-HumanTask	O	Elemento proibito da XDW
comments	OASIS_WS-HumanTask	O	Commento semplice testuale relativo alla task
attachmentInfo	OASIS_WS-HumanTask	O	Riferimenti e descrizioni dei documenti connessi all'esecuzione della task, ma che non possono essere trattati come inputs o outputs

-<taskDetails>: tipo di elemento tTaskDetails

Elemento taskDetails	Standard	Opzionalità	Descrizione
Id	OASIS_WS-HumanTask	R	ID interno per la task
taskTypes	OASIS_WS-HumanTask	R	
Name	OASIS_WS-HumanTask	R	
Status	OASIS_WS-HumanTask	R	Valore che descrive lo stato della task come descritto nelle sezioni precedenti
Priority	OASIS_WS-HumanTask	O	
taskInitiator/taskStakeholders	OASIS_WS-HumanTask	O	
potentialOwners	OASIS_WS-HumanTask	O	Owners nella terminologia HumanTask sono persone o organizzazioni che realizzano una task
businessAdministrators	OASIS_WS-HumanTask	O	
actualOwner	OASIS_WS-HumanTask	O	Il realizzatore attuale della task
notificationRecipients	OASIS_WS-HumanTask	O	
createdTime	OASIS_WS-HumanTask	O	
createdBy	OASIS_WS-HumanTask	R	

Elemento taskDetails	Standard	Opzionalità	Descrizione
lastModifiedTime	OASIS_WS- HumanTask	R	
lastModifyBy	OASIS_WS- HumanTask	O	
activationTime	OASIS_WS- HumanTask	O	
expirationTime	OASIS_WS- HumanTask	O	
isSkipable	OASIS_WS- HumanTask	O	
hasPotentialOwners	OASIS_WS- HumanTask	O	
startedByTimeExists	OASIS_WS- HumanTask	O	
completedByTimeExists	OASIS_WS- HumanTask	O	
presentationName	OASIS_WS- HumanTask	O	
presentationSubject	OASIS_WS- HumanTask	O	
renderingMethodExists	OASIS_WS- HumanTask	O	
hasOutput	OASIS_WS- HumanTask	O	
hasFault	OASIS_WS- HumanTask	O	
hasAttachments	OASIS_WS- HumanTask	O	
hasComments	OASIS_WS- HumanTask	O	
Escalated	OASIS_WS- HumanTask	O	
searchBy	OASIS_WS- HumanTask	O	
Outcome	OASIS_WS- HumanTask	O	
parentTaskId	OASIS_WS- HumanTask		XDW proibisce l'utilizzo di sub-task

Elemento taskDetails	Standard	Opzionalità	Descrizione
hasSubTasks	OASIS_WS- HumanTask		XDW proibisce l'utilizzo di sub-task

-<taskEventHistory>: tipo di elemento tXDWeventHistory

Elemento taskEventHistory	Standard	Opzionalità	Descrizione
taskEvent	OASIS_WS- HumanTask	R	

-<taskEvent>: elemento del tipo taskEvent_type

Elemento taskEvent	Standard	Opzionalità	Descrizione
Id	OASIS_WS- HumanTask	R	
eventTime	OASIS_WS- HumanTask	R	
Identifier	OASIS_WS- HumanTask	R	
Principal	OASIS_WS- HumanTask	O	
eventType	OASIS_WS- HumanTask	R	
startOwner	OASIS_WS- HumanTask	O	
endOwner	OASIS_WS- HumanTask	O	
Status	OASIS_WS- HumanTask	R	
hasData	OASIS_WS- HumanTask	O	

Elemento taskEvent	Standard	Opzionalità	Descrizione
eventData	OASIS_WS- HumanTask	O	Questa struttura contiene gli elementi che son stati cambiati dall'evento.
faultName	OASIS_WS- HumanTask	O	
	OASIS_WS- HumanTask		

APPENDICE D:

Template dell'XDW WORKFLOW DOCUMENT (CDA-based)

D.1 Introduzione

I template definiscono modelli a livello di documento, section, clinical statement ed entry, che vincolano ulteriormente il CDA. I template sono identificati da un templateId con OID valido, che indica che il documento, section, clinical statement o entry identificato non solo è conforme ai requisiti del CDA, ma è anche conforme ai modelli di vincoli identificati dal template.

La raccolta di singoli vincoli e l'assegnazione di un identificativo unico al template fornisce un meccanismo semplice all'originator per asserire la conformità con i suddetti vincoli.

L'identificativo del template non ha semantica. Nella seguente sezione viene presentato in forma tabellare il template per il Workflow Document. Le varie tabelle utilizzate adottano le seguenti convenzioni:

-R: "Required" per un elemento che deve sempre essere presente all'interno di un Workflow Document. Se ci sono informazioni valide per valorizzare il campo il dato deve essere obbligatoriamente associato all'elemento. Se non vi sono informazioni accessibili o trasmissibili il campo deve contenere un elemento in grado di identificare l'assenza del valore associato, e le motivazioni per tale omissione.

-R2: "Required if data present". L'elemento deve essere presente nel Workflow Document il valore per il suddetto campo esiste. Se l'informazione non può essere trasmessa l'elemento deve contenere un valore che indica la ragione dell'omissione. Se invece, l'informazione non è accessibile al creatore del documento, o l'informazione non è accessibile in una maniera ben definita, o il creatore ritiene espressamente che l'informazione sia assente, l'elemento non deve essere totalmente presente.

-O: "optional". Identifica un elemento opzionale, che può essere presente indipendentemente che l'informazione sia accessibile o meno.

-C: "conditional". Identifica un elemento la cui presenza è R, R2, O in dipendenza di un'altra condizione. Deve essere presente una nota che spiega quando l'elemento è required ecc.

D.1.1 Responsabilità originator

Un “originator” deve includere un `templateId` in maniera tale da dichiarare la conformità con un particolare `template` (per esempio un originator deve includere il `template` del “Workflow Document” per affermare che il documento in oggetto è una istanza di “Workflow Document”).

Un originator non necessariamente deve includere un `templateId` per ogni `template` applicato per gli “oggetti” utilizzati all’interno del documento (e.g. un originator che include nel proprio documento una sezione Medications non deve obbligatoriamente includere il `medications templateId` del CCD, anche se la rappresentazione di questa sezione è conforme a questo `template`).

D.1.2 Responsabilità recipient

Un recipient può rifiutare un’istanza che non contiene un particolare `templateId` (e.g., un recipient che si attende di ricevere solo “Workflow Document” può rifiutare una istanza priva del `templateId` del “Workflow Document”).

Un recipient può utilizzare gli oggetti presenti in una istanza di documento che non contengono `templateId` (e.g., un recipient può processare un atto di tipo `substanceAdministration` all’interno della sezione Medications section, anche se l’atto non include `templateId`).

Un recipient non deve riportare un errore di conformità qualora fallisca la conformità verso un particolare `template` su classi per cui non è dichiarata la conformità verso quel `template` e per cui non è richiesta esplicitamente la conformità da altri `template`. (e.g. gli atti di `substanceAdministration` all’interno della sezione CCD Medications che non hanno un `templateId` non devono necessariamente essere conformi al CCD `template` per i medication clinical statements. Comunque esistono dei casi come ad esempio la sezione CCD payers, in cui è richiesto che gli atti che rappresentano la copertura finanziaria del paziente siano conformi al “policy act template”. In questo caso un recipient può indicare che quegli atti presenti nel “coverage act” che non rispettano i vincoli del “policy act template” non sono permessi.

D.2 Template dell'Header:

D.2.1 Attributi

Elemento del CDA Schema	Definizione	Cardinalità	Opzionalità	Vocabolario	Tipo di dato	Commenti
ClinicalDocument	La classe ClinicalDocument è il punto di ingresso all'interno del R-MIM CDA, e corrisponde all'elemento XML <ClinicalDocument> che rappresenta la radice per ogni documento CDA	1..1	R		ClinicalDocument	
ClinicalDocument/RealmCode/@code	Definizione del contesto di validità	1..1	R	Tags for identification of country	ST	
ClinicalDocument/typeId	Un riferimento esplicito	1..1	R		II	
ClinicalDocument/typeId/@root	tecnologicamente neutro alla specifica	1..1	R		OID	Valore Fisso: "2.16.840.1.113883.1.3"
ClinicalDocument/typeId/@extension	CDA R2	1..1	R		ST	Valore Fisso: "POCD_HD00040"
ClinicalDocument/templateId	Definizione del	1..1	R			
ClinicalDocument/templateId/@root	template per l'XDW Workflow Document	1..1	R		OID	Valore Fisso: "2.16.840.1.113883.2.9.10.X"
ClinicalDocument/templateId/@extension		1..1	R		ST	Valore Fisso:

tension						“IT_WF_DOCUMENT_001”
ClinicalDocument/id/@root	Rappresenta	1..1	R		OID	OID dell'ente sanitario locale
ClinicalDocument/id/@extension	l'identificatore univoco per un'istanza del documento	1..1	R		ST	UID del documento (docId)
ClinicalDocument/id/@displayable		0..1	O		BL	
ClinicalDocument/id/@AssigningAuthorityName		0..1	R2		ST	
ClinicalDocument/code		Un codice che specifica	1..1	R		CE
ClinicalDocument/code/@code	il particolare tipo di documento	1..1	R	LOINC	CS	Codice LOINC definito per l' XDW Workflow Document
ClinicalDocument/code/@displayName		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile
ClinicalDocument/code/@codeSystem		1..1	R		OID	Fixed value : (LOINC OID) “2.16.840.1.113883.6.1”
ClinicalDocument/code/@codeSystemName		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile. Valore fisso: “LOINC”
ClinicalDocument/code/@codeSystemVersion		0..1	R		ST	Valore fissato: (rappresenta la versione più recente del vocabolario LOINC) “2.19”
ClinicalDocument/<title>		Titolo del documento	0..1	C		ST
ClinicalDocument/effectiveTime	Elemento che definisce	1..1	R		TS	
ClinicalDocument/effectiveTime/@value	la data di creazione del documento	1..1	R		GMT	GMT = (Greenwich Mean Time)
ClinicalDocument/languageCode	Descrive la lingua in	1..1	R			
ClinicalDocument/languageCode/@code	cui è stato scritto il documento	1..1	R	[RFC3066]	CS	Valore Fissato: “en-GB”

ClinicalDocument/confidentiality Code	Codice che identifica la sensibilità delle	1..1	R		CE	
ClinicalDocument/confidentiality Code/@code	informazioni contenute e definisce quali	1..1	R	N,R,V	CS	Valore Fissato: "N"= Normal
ClinicalDocument/confidentiality Code/@codeSystem	informazioni sono accessibili o nascoste	1..1	R		OID	Valore fissato: "2.16.840.1.113883.5.25"
ClinicalDocument/confidentiality Code/@codeSystemName		1..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile Valore Fissato: "HL7 Confidentiality Code"
ClinicalDocument/confidentiality Code/@displayName		1..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile Valore fissato: "Normal"

D.2.2 Partecipanti

D.2.2.1 Paziente

Elemento del CDA Schema	Definizione	Cardinalità	Opzionalità	Vocabolario	Tipo di dato	Commenti
ClinicalDocument/recordTarget	Elemento che permette di descrivere il soggetto principale del Workflow	1..1	R		SET<RecordTarget >	
ClinicalDocument/recordTarget/patientRole	Elemento che definisce che il soggetto del Workflow	1..1	R		PatientRole	

	è un paziente					
ClinicalDocument/recordTarget/patientRole/id	Definisce l'ID principale del paziente	1..3	R		SET<II>	
ClinicalDocument/recordTarget/patientRole/id/@root	(lo stesso contenuto nell'XDS patientId). Possono essere utilizzati altri ID per gestire varie tipologie	1..1	R	Per valorizzare questo elemento si guardi la descrizione presente nelle sezioni precedenti	OID	
ClinicalDocument/recordTarget/patientRole/id/@extension		1..1	R		ST	
ClinicalDocument/recordTarget/patientRole/id/@assigningAuthorityName	di paziente (straniero, temporaneamente presente o altri casi)	0..1	R2		ST	
ClinicalDocument/recordTarget/patientRole/addr	Definizione dell'indirizzo del paziente. E' possibile memorizzare più di un indirizzo.	0..*	O		AD	
ClinicalDocument/recordTarget/patientRole/addr/@use		0..1	O	HL7 Address Use Code	SET<AD>	Possibili valori per "use": -WP: work place -PST: postal address -H: home address -TMP: temporary address -omitted : non definito
ClinicalDocument/recordTarget/patientRole/addr/<streetName>		0..1	O		ST	
ClinicalDocument/recordTarget/patientRole/addr/<houseNumber>		0..1	O		ST	

ClinicalDocument/recordTarget/patientRole/addr/<postalCode>		0..1	O		ST	
ClinicalDocument/recordTarget/patientRole/addr/<city>		0..1	O		ST	
ClinicalDocument/recordTarget/patientRole/addr/<country>		0..1	O	Code ISO 3166-1	ST	
ClinicalDocument/recordTarget/patientRole/telecom	Definizione del numero di telefono del	0..*	O		TEL	
ClinicalDocument/recordTarget/patientRole/telecom/@use	paziente	0..1	O	HL7 v3 telecommunicationAddressUse	SET<TEL>	Valori possibili per "use": -HP: home phone -WP: work phone -MC: mobile cellular
ClinicalDocument/recordTarget/patientRole/telecom/@value		0..1	O		URL	
ClinicalDocument/recordTarget/patientRole/patient	Elemento che definisce informazioni anagrafiche del patientRole	1..1	R			
ClinicalDocument/recordTarget/patientRole/patient/name	Elemento che definisce in modo strutturato il nome del	1..1	R		SET<PN>	
ClinicalDocument/recordTarget/patientRole/patient/name/<given>	paziente	1..1	R2		ST	
ClinicalDocument/recordTarget/patientRole/patient/name/<family>		1..1	R2		ST	

ClinicalDocument/recordTarget/patientRole/patient/name/<suffix>		1..1	R2		ST	
ClinicalDocument/recordTarget/patientRole/patient/name/@nullFlavor		0..1	C		ST	Necessario se il nome non è valorizzato o non è disponibile. In questo caso assume il valore: nullFlavor="MSK"
ClinicalDocument/recordTarget/patientRole/patient/AdministativeGenderCode	Elemento che definisce il sesso del paziente	1..1	R		CE	
ClinicalDocument/recordTarget/patientRole/patient/AdministativeGenderCode/@code		1..1	R		CS	
ClinicalDocument/recordTarget/patientRole/patient/AdministativeGenderCode/@displayName		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile
ClinicalDocument/recordTarget/patientRole/patient/AdministativeGenderCode/@codeSystem		1..1	R		OID	Valore fissato: "2.16.840.1.113883.5.1"
ClinicalDocument/recordTarget/patientRole/patient/AdministativeGenderCode/@codeSystemName		0..1	C		ST	Valore Fissato: "HL7 Administrative Gender". Obbligatorio se deve essere mostrato in formato leggibile
ClinicalDocument/recordTarget/patientRole/patient/birthTime	Elemento che definisce la data di nascita del paziente	1..1	R		TS	
ClinicalDocument/recordTarget/patientRole/patient/birthTime		1..1	R			Il formato di questo dato è descritto nelle

atientRole/patient/birthTime/@value						sezioni precedenti
ClinicalDocument/recordTarget/patientRole/patient/birthPlace	Elemento che definisce il luogo di nascita del paziente	1..1	R		Birthplace	
ClinicalDocument/recordTarget/patientRole/patient/birthPlace/@nullFlavor		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile. In questo caso è valorizzato: nullFlavor="MSK"
ClinicalDocument/recordTarget/patientRole/patient/birthPlace/place		1..1	R2		Place	
ClinicalDocument/recordTarget/patientRole/patient/birthPlace/place/addr		1..1	R2		AD	
ClinicalDocument/recordTarget/patientRole/patient/birthPlace/place/addr/<country>		1..1	R2	Code ISO 3166-1	ST	
ClinicalDocument/recordTarget/patientRole/patient/birthPlace/place/addr/<city>		1..1	R2		ST	

D.2.2.2 Autore:

Elemento del CDA Schema	Definizione	Cardinalità	Opzionalità	Vocabolario	Tipo di dato	Commenti
ClinicalDocument/author	Elemento che definisce l'autore del documento	1..1	R		SET<Author>	
ClinicalDocument/author/time	Data e ora della creazione del documento	1..1	R		TS	
ClinicalDocument/author/time/@value		1..1	R		GMT	
ClinicalDocument/author/assignedAuthor	Elemento che definisce che un autore è un entità nel ruolo di assignedAuthor	1..1	R		AssignedAuthor	
ClinicalDocument/author/assignedAuthor/id	Definisce l'ID principale dell'autore.	1..*	R	Per valorizzare	SET<II>	
ClinicalDocument/author/assignedAuthor/id/@root	Altri ID possono essere usati per identificare	1..1	R	questo elemento si	OID	
ClinicalDocument/author/assignedAuthor/id/@extension	localmente l'autore del documento	1..1	R	guardi la descrizione	ST	
ClinicalDocument/author/assignedAuthor/id/@assigningauthorityName		1..1	R2	presente nelle sezioni	ST	
ClinicalDocument/author/assignedAuthor/id/@displayable		0..1	O	precedenti	BL	

ClinicalDocument/author/assignedAuthor/assignedAuthoringDevice	Elemento che definisce il device utilizzato per scrivere il Workflow Document	1..1	R		AuthoringDevice	
ClinicalDocument/author/assignedAuthor/assignedAuthoringDevice/<softwareName>	Elemento che definisce il nome del device Software utilizzato per scrivere il documento	1..1	R		ST	
ClinicalDocument/author/assignedAuthor/representedOrganization	Elemento che definisce l'organizzazione a cui appartiene l'autore	1..1	R2		Organization	
ClinicalDocument/author/assignedAuthor/representedOrganization/id	Definisce l'ID principale dell'organizzazione	1..1	R2		SET<II>	
ClinicalDocument/author/assignedAuthor/representedOrganization/id/@root		1..1	R2		OID	
ClinicalDocument/author/assignedAuthor/addr	Elemento che definisce l'indirizzo dell'autore del documento	0..*	O		AD	
ClinicalDocument/author/assignedAuthor/addr/@use		1..1	R2	HL7 Address Use Code	SET<AD>	

ClinicalDocument/author/assignedAuthor/addr/<streetName>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<houseNumber>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<postalCode>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<city>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<country>		1..1	R2	Code ISO 3166-1	ST	
ClinicalDocument/author/assignedAuthor/telecom	Elemento che definisce il numero di telefono dell'autore	0..*	O		TEL	
ClinicalDocument/author/assignedAuthor/telecom/@value		1..1	R2		URL	
ClinicalDocument/author/assignedAuthor/telecom/@use		1..1	R2	HP,WP,MC	SET<TEL>	Possibili valori per "value": -HP: home phone -WP: work phone -MC: mobile cellular

D.2.2.3 Custode

Elemento del CDA Schema	Definizione	Cardinalità	Opzionalità	Vocabolario	Tipo di dato	Commenti
ClinicalDocument/custodian	Definisce il custode del documento. Rappresenta l'entità responsabile della memorizzazione del documento	1..1	R		custodian	
ClinicalDocument/custodian/assignedCustodian	Elemento che descrive il ruolo del custode	1..1	R		AssignedCustodian	
ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization	Elemento che descrive l'organizzazione che agisce nel ruolo di custode del documento	1..1	R		CustodianOrganization	
ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization/id	Definisce l'ID univoco dell'organizzazione custode	1..*	R		SET<II>	
ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization/id/@root		1..1	R		OID	
ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization/id/@extension		1..1	R		ST	

ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization/id/@assigningAuthorityName		1..1	R		ST	
ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization/id/<name>	Elemento che definisce il nome del custode	0..1	O		ON	
ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization/id/<telecom>	Elemento che definisce il numero di telefono del custode	0..1	O		TEL	
ClinicalDocument/custodian/assignedCustodian/representedCustodianOrganization/id/addr	Elemento che definisce l'indirizzo del custode	0..1			AD	

D.2.3 Documenti Correlati

Elemento del CDA Schema	Definizione	Cardinalità	Opzionalità	Vocabolario	Tipo di dato	Commenti
ClinicalDocument/RelatedDocument	Elemento che definisce il documento da cui deriva la versione attuale del Workflow Document	1..1	R		SET<RelatedDocument>	
ClinicalDocument/RelatedDocument/@typeCode	Definizione del tipo di relazione esistente tra la versione attuale del Workflow Document e la versione da cui deriva.	1..1	R		CS	Valore fissato: "RPLC"
ClinicalDocument/RelatedDocument/parentDocument	Elemento che definisce il parent document	1..1	R		ParentDocument	
ClinicalDocument/RelatedDocument/parentDocument/id	Elemento che definisce	1..1	R		SET<II>	
ClinicalDocument/RelatedDocument/parentDocument/id/@root	l'identificatore univoco per il parent document	1..1	R		OID	UID del documento da cui deriva la versione attuale (docId)
ClinicalDocument/RelatedDocument/parentDocument/id/@extension		1..1	R		ST	
ClinicalDocument/RelatedDocument		0..1	R2		ST	

ent/parentDocument/id/@assigningAuthorityName						
Clinicaldocument/setId	Rappresenta un	1..1	R			
Clinicaldocument/setId/@root	identificativo univoco	1..1	R		OID	
Clinicaldocument/setId/@extension	per tutte le differenti versioni dello stesso Workflow Document	1..1	R		UID	docId
Clinicaldocument/setId/@assigningAuthorityName		0..1	R2		ST	
ClinicalDocument/versionNumber	Valore intero che	1..1	R			
ClinicalDocument/versionNumber/@number	distingue le varie versioni del Workflow Document.	1..1	R		INT	Valore numerico intero progressivo. Per la prima versione il valore è ="1"

D.3 Template del Body Strutturato

Il Body del Workflow Document deve essere del tipo structuredBody, all'interno del quale possano essere riconosciute in modo semplice tutte le differenti task realizzate. Quindi si è deciso di definire uno specifico template CDA in grado di descrivere la tipica struttura di una task (composta da una sezione CONTROL INFO e due sezioni di INPUT e OUTPUT). Ogni task è racchiusa in due element component/section. In questa section si definiscono le informazioni correlate alla task (ovvero la CONTROL INFO section) quali: lo “stepcode” che definisce la task (il cui vocabolario è definito dallo specifico dominio che utilizza il profilo XDW), l'autore della task, e due sezioni foglia definite dallo stesso templateId che identificano la struttura delle sezioni di INPUT e di OUTPUT.

D.3.1 Primo livello: sezione CONTROL INFO

templateId	“[OID_definito_per_il_template_della_section_task]”
Descrizione Generale	Il primo elemento che questa sezione DEVE contenere è il templateId che descrive la struttura di una sezione task. La sezione DEVE anche contenere l' elemento <code>, che permette di descrivere lo step performato, utilizzando un vocabolario fornito dallo specifico dominio IHE che utilizza il profilo XDW. Questa sezione DEVE contenere anche un elemento <author> che permette di identificare l'ente responsabile per la specifica task descritta. Questo elemento DEVE contenere a sua volta un elemento <time> che permette di memorizzare il tempo di creazione della task e l'elemento <assignedAuthor> che descrive il ruolo dell'autore e le informazioni correlate ad esso. La sezione task PUO' contenere una sezione

			di INPUT e/o una sezione di OUTPUT.			
Elemento del CDA Schema	Definizione	Cardinalità	Opzionalità	Vocabolario	Tipo di dato	Commenti
ClinicalDocument/component/structuredBody/component/section		1..1	R			
ClinicalDocument/component/structuredBody/component/section/templateId		1..1	R			
ClinicalDocument/component/structuredBody/component/section/templateId/@root		1..1	R		OID	Valore fissato: “[OID da definire]”
ClinicalDocument/component/structuredBody/component/section/code	Codice che descrive la task eseguita,	1..1	R			
ClinicalDocument/component/structuredBody/component/section/code/@code	utilizzando un vocabolario specifico per il particolare dominio	1..1	R	Vocabolario dominio specifico	CS	
ClinicalDocument/component/structuredBody/component/section/code/@displayName	IHE che utilizza il profilo XDW	0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile.
ClinicalDocument/component/structuredBody/component/section/code/@codeSystem		1..1	R		OID	
ClinicalDocument/component/structuredBody/component/section/code/		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile.

@codeSystemName						
ClinicalDocument/component/structuredBody/component/section/code/@codeSystemVersion		1..1	R		ST	
<ClinicalDocument/author	Elemento che definisce l'autore della task.	1..1	R		SET<Author>	
ClinicalDocument/author/time	Data e ora della creazione della task.	1..1	R		TS	
ClinicalDocument/author/time/@value		1..1	R		GMT	
ClinicalDocument/author/assignedAuthor	Elemento che definisce che un autore è un'entità nel ruolo di assignedAuthor	1..1	R		AssignedAuthor	
ClinicalDocument/author/assignedAuthor/id	Definisce l'ID principale dell'autore. Altri ID possono essere usati per identificare localmente l'autore del documento	1..*	R	Per valorizzare questo elemento si guardi la descrizione presente nelle sezioni precedenti	SET<II>	
ClinicalDocument/author/assignedAuthor/id/@root		1..1	R		OID	
ClinicalDocument/author/assignedAuthor		1..1	R		ST	

uthor/id/@extension						
ClinicalDocument/author/assignedAuthor/id/@assigningauthorityName		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/id/@displayable		0..1	O		BL	
ClinicalDocument/author/assignedAuthor/assignedAuthoringDevice	Elemento che definisce il device utilizzato per scrivere il Workflow Document	1..1	R		AuthoringDevice	
ClinicalDocument/author/assignedAuthor/assignedAuthoringDevice/<softwareName>	Elemento che definisce il nome del device Software utilizzato per scrivere il documento	1..1	R		ST	
ClinicalDocument/author/assignedAuthor/representedOrganization	Elemento che definisce l'organizzazione a cui appartiene l'autore	1..1	R2		Organization	
ClinicalDocument/author/assignedAuthor/representedOrganization/id	Definisce l'ID principale dell'organizzazione	1..1	R2		SET<II>	
ClinicalDocument/author/assignedAuthor/representedOrganization/id/@root		1..1	R2		OID	

ClinicalDocument/author/assignedAuthor/addr	Elemento che definisce l'indirizzo dell'autore del documento	0..*	O		AD	
ClinicalDocument/author/assignedAuthor/addr/@use		1..1	R2	HL7 Address Use Code	SET<AD>	
ClinicalDocument/author/assignedAuthor/addr/<streetName>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<houseNumber>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<postalCode>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<city>		1..1	R2		ST	
ClinicalDocument/author/assignedAuthor/addr/<country>		1..1	R2	Code ISO 3166-1	ST	
ClinicalDocument/author/assignedAuthor/telecom		Elemento che definisce il numero di telefono dell'autore	0..*	O		TEL
ClinicalDocument/author/assignedAuthor/telecom/@value	1..1		R2		URL	
ClinicalDocument/author/assignedAuthor/telecom/@use	1..1		R2	HP,WP,MC	SET<TEL>	Valori possibili per "value": -HP: home phone -WP: work phone -MC: mobile cellular
INPUT section	Sezione definita per descrivere e	0..1	O			

	memorizzare il riferimento a documenti e dati utili per completare la task			templateId: "1.2.3.4.5.6.7.8.9.0"
OUTPUT section	Sezione definita per descrivere e memorizzare il riferimento a documenti e dati prodotti dalla task	0..1	O	templateId: "1.2.3.4.5.6.7.8.9.0"

D.3.2 Secondo livello: sezioni INPUT/OUTPUT

templateId	“[OID_definito per_CDAtemplate_INPUT/OUTPUT_section]”
Descrizione Generale	<p>La sezione INPUT/OUTPUT DEVE contenere il templateId che definisce le regole di conformità che deve rispettare la sezione. Ogni sezione INPUT/OUTPUT deve contenere un elemento <code> per distinguere tra una sezione di INPUT o di OUTPUT. Questo codice è definito da un vocabolario specifico definito dallo specifico dominio IHE. La sezione DEVE contenere un Narrative Block, contenuto tra due elementi <text>.</p> <p>Questo blocco è strutturato utilizzando l'elemento <list> all'interno del quale ogni <item> corrisponde ad un documento (prodotto in output alla task o fornito in input alla stessa). La sezione DEVE anche contenere il blocco processabile a livello macchina dove, l'elemento <code> permette di definire il tipo di <act> processato. Gli elementi centrali del blocco sono I riferimenti ai documenti esterni. Per ogni documento è utilizzato un elemento <reference> che contiene un elemento <externalDocument> che permette di memorizzare esplicitamente il riferimento al documento esterno utilizzando tre differenti elementi <id>:</p> <ul style="list-style-type: none">-Un documentId;-Un repositoryId;- Un homecommunityId; <p>Per ogni riferimento esrno deve essere definito un elemento <code>,</p>

			utilizzando il dizionario LOINC, il quale permette di descrivere il tipo di documento. Inoltre, l'elemento obbligatorio <originalText> permette di creare una relazione tra le informazioni contenute nel narrative block e quelle contenute nel blocco processabile.			
Elemento del CDA Schema	Definizione	Cardinalità	Opzionalità	Vocabolario	Tipo di dato	Commenti
Component/section		1..1	R			
Component/section/templateId	Codice che	1..1	R			
Component/section/templateId/@root	identifica il template delle sezioni di INPUT/OUTPUT	1..1	R		OID	
Component/section/code	Codice che permette	1..1	R			
Component/section/code/@code	di fare distinzioni tra	1..1	R		CS	
Component/section/code/@displayName	sezioni di INPUT e OUTPUT	0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile.
Component/section/code/@codeSystem		1..1	R		OID	
Component/section/code/@codeSystemName		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile.
Component/section/code/@codeSystemVersion		1..1	R		ST	
Component/section/<title>	Elemento che definisce il titolo della sezione	0..1	C		ST	Fixed value: "section INPUT" or "section OUTPUT" Obbligatorio se deve

						essere mostrato in formato leggibile.
Component/section/text	Elemento che racchiude il narrativ e block della sezione	1..1	R			
Component/section/text/list	Elemento che permette di strutturare il blocco narrativo come una lista	1..1	R			
Component/section/text/list/<caption>	Elemento che definisce l'etichetta associata alla lista	1..1	R		ST	Valore fissato "Input documents" o "Output documents"
Component/section/text/list/item	Elemento che definisce ogni componente della lista	1..*	R			
Component/section/text/list/item/<content>	Elemento che definisce le informazioni leggibili correlate con il contenuto della porzione processabile a livello macchina	1..1	R		ST	Definisce una descrizione del documento di cui è memorizzato il riferimento
Component/section/text/list/item/content/@ID	Definisce l'etichetta per il documento	1..1	R		ST	Il valore di questo elemento deve essere

	descritto nell'elemento <content>					utilizzato all'interno del blocco processabile a livello macchina per indicizzare il documento descritto
Component/section/entry	Elemento che definisce il blocco processabile a livello macchina	1..1	R		SET<Entry>	
Component/section/entry/act	Elemento che	1..1				
Component/section/entry/act/@classCode	descrive il tipo di entry processabile	1..1				Valore Fissato="ACT"
Component/section/entry/act/@moodCode		1..1				Valore Fissato="EVN"
Component/section/entry/act/code	Codice che descrive	1..1	R		CE	
Component/section/entry/act/code/@code	l'atto clinico. Questo codice è definito dal	1..1	R		CS	
Component/section/entry/act/code/@displayName	vocabolario specifico del dominio IHE che utilizza XDW	0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile.
Component/section/entry/act/code/@codeSystem		1..1	R		OID	
Component/section/entry/act/code/@codeSystemName		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile.

Component/section/entry/act/code/ @codeSystemVersion		1..1	R		ST	
Component/section/entry/act/referen ce	Elemento utilizzato per creare un riferimento per ogni documento descritto nel blocco narrativo.	1..*	R		SET<Reference >	
Component/section/entry/act/referen ce/@typecode		1..1	R		ST	Valore Fissato="REFR"
Component/section/entry/act/referen ce/originaltext	Elemento che permette di correlare il blocco narrativo con quello processabile	1..1	R			
Component/section/entry/act/referen ce/originaltext/reference	Elemento che memorizza l'etichetta utilizzata nel narrativ e block associata al documento posizionato come riferimento esterno.	1..1	R			E' necessario utilizzare per questo elemento la stessa etichetta utilizzata nel narrative block aggiungendo il simbolo "#".
Component/section/entry/act/referen ce/originaltext/reference/@value		1..1	R		ST	
Component/section/entry/act/referen ce/externaldocument	Elemento che permette di memorizzare esplicitamente il riferimento esterno	1..1	R		ExternalDocume nt	
Component/section/entry/act/referen ce/externaldocument/@classCode		1..1	R		ST	Valore Fissato="DOC"
Component/section/entry/act/referen ce/externaldocument/@moodCode		1..1	R		ST	Valore Fissato="EVN"

Component/section/entry/act/reference/externaldocument/id	Identificatori che permettono di identificare ogni documento.	3..3	R		SET<II>	documentId
Component/section/entry/act/reference/externaldocument/id/@root		3..3	R		OID	repositoryId
Component/section/entry/act/reference/externaldocument/id/@extension		3..3	R		CS	homeCommunityId
Component/section/entry/act/reference/externaldocument/code	Elemento che definisce il tipo di documento memorizzato all'interno dell'elemento <externalReference >	1..1	R		CE	
Component/section/entry/act/reference/externaldocument/code/@code		1..1	R		CS	
Component/section/entry/act/reference/externaldocument/code/@codeSystem		1..1	R		OID	Valore Fissato="2.16.840.1.113883.6.1"
Component/section/entry/act/reference/externaldocument/code/@codeSystemName		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile. Valore Fissato="LOINC"
Component/section/entry/act/reference/externaldocument/code/@displayName		0..1	C		ST	Obbligatorio se deve essere mostrato in formato leggibile.
Component/section/entry/act/reference/externaldocument/code/@codeSystemVersion		1..1	R		ST	Valore Fissato="2.19"

BIBLIOGRAFIA

- 2004, Robert H. Dolin, Liora Alschuler, Sandy Boyer, Calvin Beebe: "HL7 Clinical Document Architecture, Release 2.0"
- 2007, Gregorio Mercurio: "Come HL7 contribuisce nel settore della telemedicina"
- 2009, Arianna Cocchiglia: "Gli standard di Cooperazione Applicativa: Il progetto Health Optimum" tesi magistrale
- 2010, HL7 Italia: "Identificazione OID HL7 Italia" ver. 2.2
- 2009, HL7 Italia "Implementation Guide Clinical Document Architecture (CDA) Rel. 2: Prescrizione" ver. 1.01
- 2010, Tavolo Permanente per la sanità elettronica "Specifiche tecniche per la creazione del "Documento di Referto" secondo lo standard HL7-CDA Rel. 2" ver. 1.1
- 2009, Tavolo Permanente per la sanità elettronica "Standard tecnici per la creazione del "Documento di Prescrizione" secondo lo standard HL7-CDA Rel. 2" ver 2.0
- 2003, Mag Murray, "Strategies for the Successful Implementation of Workflow Systems within Healthcare: A Cross Case Comparison", 36th Hawaii International Conference on System Sciences
- 2003, W.M.P. Van der Aalst, Akhil Kumar: "XML Based Schema Definition for Support of Inter-organizational Workflow"
- 2000, W.M.P. Van der Aalst, P. Bartherlmess, C.A. Ellis, J. Wainer: "Workflow Modeling using Procllets"
- 2002, Rupa Krishnan, Lalitha Munaga, Kamalakar Karlapalem: "XDoC-WFMS: A Framework for Document Centric Workflow Management System"
- 2004, Eric D. Browne, Michael Scherefl, James Warren: "Goal-focused Self-Modifying Workflow in the Healthcare Domain"
- 2002, Jaeho kim, WoojongSuh, Heeseok Lee, "Document-based Workflow modeling: a case-based reasoning approach"
- 2002, R.S. Mans, N.C. Russel, P.J.M. Bakker, W.M.P. Van der Aalst, A.J. Moleman: "Procllets in Healthcare"
- 2005, Thomas Fahringer, Qin Jun, Stefan Heinzer: "Specification of Grid Workflow Applications with AGWL: An Abstract Grid Workflow Language"
- 2002, Sriram Krishnam, Patrick Wagstrom, Gregor von Laszewski: "GSFL: A Workflow Framework for Grid Services"
- 2010, IHE International: IHE IT Infrastructure (ITI) Technical Framework, Volume 1, (ITI TF-1) Integration Profiles, Revision 7.0
- 2007, Richard Lenz, Manfred Reichert: "IT support for healthcare processes (premises, challenges, perspectives)"
- 2004, M. Stefanelli: "Knowledge and Process Management in Health Care Organizations"
- 2000, Edward A. Lee, Steve Neuendorffer: "MoML — A Modeling Markup Language in XML — Version 0.4"
- 2008, *The Workflow Management Coalition Specification*: "Process Definition Interface:XML Process Definition Language" ver.2.1a
- 2006, Stefano Ratto: "Studio e sviluppo di un'applicazione per l'accesso distribuito ai dati di Tissue Array mediante agenti mobili" Appendice D: "STRUMENTI E LINGUAGGI PER WORKFLOW"
- 1998, Shi Meilin, Yang Guangxin, Xiang Yong, Wu Shangguguan: "Workflow Management Systems: A Survey"
- 2002, Maurizio Tesconi, Andrea Marchetti, Oreste Signore: "Descrizione e gestione di workflow documentali con una

applicazione basata su XML”

2001, IBM Software Group: “Web Services Flow Language (WSFL 1.0)”

2010, OASIS: “Web Services – Human Task (WS-HumanTask) Specification Version 1.1 ”

2005, Maurizio Tesconi, Andrea Marchetti, Salvatore Minutoli: “XFlow: An XML-Based Document-Centric Workflow”

2008, Robert M. Shapiro: “XPDL 2.1 *Integrating Process Interchange & BPMN*”

2002, W.M.P. Van der Aalst, A.H.M. TerHofstede: “YAWL: Yet Another Workflow Language”

2010, Saccavini Claudio, Cocchiglia Arianna: “*ArsenàIT_Progetto_DOGE_v03_2010-03-09*” ver.4.0

2007, Oracle Inc.: “WS-BPEL Extension for People (BPEL4People), Version 1.0 ”