

Modello di descrizione ed annotazione semantica dei servizi intelligenti

Survey

Nel presente documento si riassumono i contenuti del deliverable "**D2.2_2 Modello di descrizione ed annotazione semantica dei servizi intelligenti**" relativo all'attività di ricerca *RI 2 Definizione di modelli di descrizione ed annotazione semantica di contenuti multimediali e servizi interattivi*, nell'ambito del secondo Obiettivo Realizzativo (OR 2) "Studio di servizi e contenuti intelligenti".

L'attività di ricerca è stata finalizzata alla definizione di un modello di descrizione ed annotazione semantica dei servizi intelligenti integrando opportunamente il modello della logica di business con il modello delle unità astratte di interazione attraverso i dati contenuti nelle entity. Partendo da questa rappresentazione si è fornito il necessario collegamento con il modello del dominio di erogazione/fruizione. L'attività si è concentrata poi sullo studio delle possibilità offerte dall'utilizzo di un algoritmo di predizione basate su modelli probabilistici per l'implementazione di tecniche di "*look-ahead*" in SAPI. Infine, tramite un activity diagram, viene mostrato il processo di esecuzione di un servizio intelligente così definito.

Una delle caratteristiche più rilevanti del framework SAPI è rappresentata dalla scelta di adottare un paradigma di separazione dei vari strati logici di composizione di un servizio web, per garantire una maggiore versatilità di adattamento sulla base di caratteristiche utente, dominio e contesto di fruizione. Questo approccio consente una modellazione separata dei vari livelli che sarà quindi soggetta, entro certi limiti, solo a vincoli di design propri e perciò indipendente dalla struttura e dalla implementazione degli altri livelli. In particolare la modellazione del workflow di un servizio non farà alcuna assunzione né sulle modalità di presentazione dei contenuti né sul design dell'interfaccia utente. Questi livelli non comunicheranno direttamente tra di loro, ma si sincronizzeranno solo sulla base dei dati, contenuti nelle entity, strettamente legati all'interazione con l'utente e necessari all'avanzamento della logica di business (vedi Figura 1).

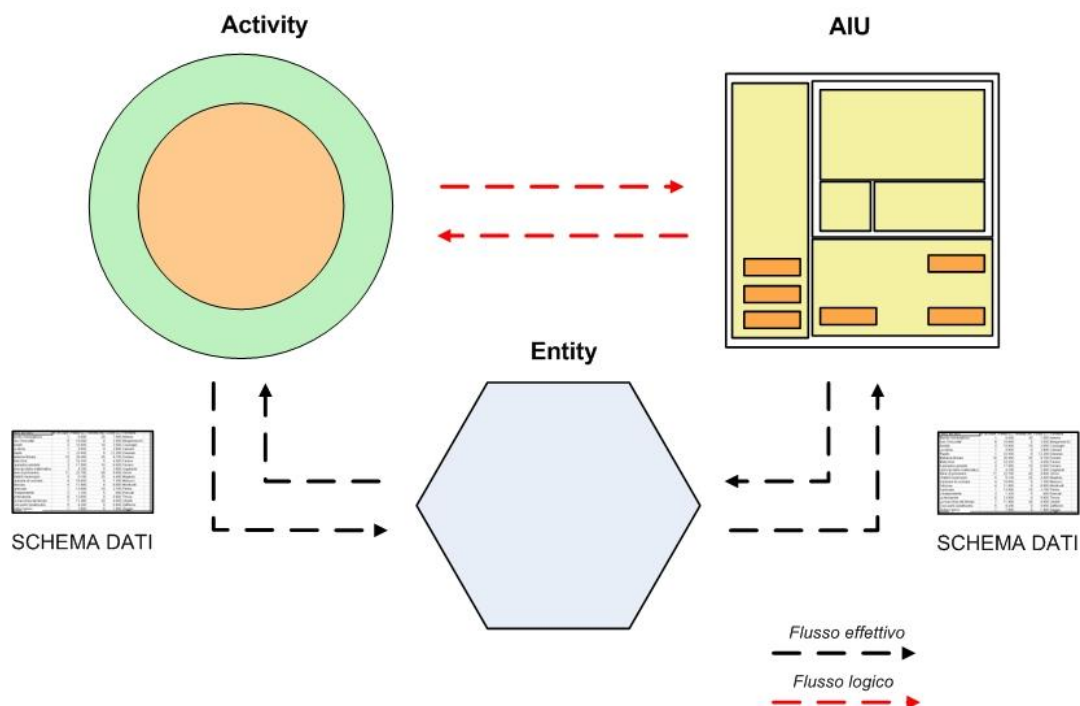


Figura 1: Collegamento di una Activity con una AIU.

Nel processo di composizione di un e-service, per poter individuare i contenuti da presentare all'utente, è necessario innanzitutto conoscerne le informazioni contenute nella logica di business. Questa, infatti, definisce tutti quei dati da scambiare con l'utente che sono strettamente necessari per consentire l'evoluzione del servizio e che indicano, quindi, l'insieme minimo di contenuti da presentare. Ovviamente i contenuti sottoposti all'utente non coincideranno semplicemente con gli input e/o gli output definiti nelle activity, ma saranno ampliati con tutti quegli elementi aggiuntivi che rendono più presentabili e facilmente comprensibili i medesimi input ed output.

Punto essenziale per la definizione di un servizio SAPI è, quindi, la definizione dei parametri di scambio tra business e presentazione. A tal fine ogni activity condividerà con la relativa entity uno schema di modellazione dei dati che si farà carico di trasportare le informazioni relative al numero ed al tipo dei parametri in gioco. A tal fine si definisce una apposita tabella, detta *Schema dei Dati* (SD), che contiene tutte le informazioni scambiate tra una activity, atomica o composta, ed una o più AIU attraverso l'utilizzo di una particolare Entity Composta. Per la definizione di questo schema si recuperano, fondendoli, i dati presenti nel Modello I/O di una AA e quelli contenuti nella Business Entity Table. A questi dati si possono eventualmente aggiungere informazioni che specificano il tipo dei parametri in gioco ed i relativi vincoli quali, ad esempio, la lunghezza massima del dato. Queste informazioni sono da ritenersi, in questa fase della modellazione, come dei semplici suggerimenti assolutamente non vincolanti e comunque slegati dalla effettiva modalità di presentazione dei dati cui si riferiscono.

Schema dei Dati					
Activity (tipo activity)		Entity Composta		Campi Opzionali	
Variabile	Descrizione	EA	SAEL pointer	Tipo	Vincoli

Tabella 1: Schema dei Dati.

I dati contenuti in uno Schema dei Dati, vedi Tabella 1, sono raggruppati in tre macrogruppi, corrispondenti alle tre colonne principali, che racchiudono rispettivamente le informazioni relative alle: activity, entity e ai campi opzionali. Più nel dettaglio un SD avrà i seguenti campi:

- **Activity:** identificativo univoco dell'activity cui è associato lo schema dei dati.
 - **(tipo activity):** specifica del tipo di activity, atomica o composta. Notiamo esplicitamente che quando il tipo di activity non è esplicitamente indicato si assume, per default, che essa è atomica.
 - **Variabile:** nome generico associato all'activity atomica corrispondente al dato di scambio. Ad una variabile possono corrispondere una o più EA
 - **Descrizione:** descrizione di carattere generale del dato di I/O dell'activity. Questa descrizione può eventualmente contenere indicazioni circa le operazioni consentite e la modalità di presentazione più idonea.
- **Entity Composta:** identificativo univoco dell'entity composta che racchiude i contenuti richiesti dall'activity.
 - **EA:** elenco di EA necessarie per eseguire le funzionalità previste dal business corrispondenti ai dati di scambio con il livello di presentazione.
 - **SAEL pointer:** puntatore al SAEL [G] che permette di riferire l'EA nella sua totalità e di identificarla univocamente.
- **Campi Opzionali:** in questa colonna sono racchiuse le informazioni aggiuntive relative al tipo dei parametri ed ad eventuali vincoli.
 - **Tipo:** tipo di dato da attribuire alla variabile. Questi tipi possono essere scelti, ad esempio, tra quelli predefiniti di OWL.
 - **Vincoli:** constraints che stabiliscono delle limitazioni da applicare alla particolare variabile. Queste proprietà possono riguardare ad esempio il massimo numero di caratteri o range di valori ammissibili.

Una volta individuato il servizio per cui si vogliono definire le EA, l'individuazione dei contenuti si articola negli step di seguito illustrati.

1. Selezione della activity che hanno informazioni da scambiare con le AIU.
2. Individuazione delle variabili che necessitano di una EA, a partire dal Modello Input/Output della activity atomica selezionata (a tal fine bisogna riferirsi al campo **ELR** del modello I/O dell'activity).

3. Verifica, nel SAEL, dell'eventuale esistenza di una EA che si adatta al caso.
4. Creazione della EA aggiungendo un riferimento nel SAEL (Solo se lo step 3 dà esito negativo).
5. Recupero del SAEL pointer relativo alla EA individuata.

Individuazione, se possibile, del tipo e di eventuali vincoli da associare al parametro.

Oltre alla modalità di integrazione tra logica di business e logica di presentazione è fondamentale anche il collegamento del modello definito per la logica di business con quello del dominio di SAPI descritto riportato in Figura 2. In merito a tale integrazione è ragionevole definire la classe operazione del tutto equivalente alla classe activity. Un'operazione, infatti, può essere vista, come l'insieme di più sotto-processi, che in questo caso corrisponderebbero ad una activity composta, oppure, come un unico processo, che corrisponderebbe ad una activity atomica. Nel caso di contenuto intelligente l'operazione associata coinciderà propriamente con un'activity atomica. La necessità di mantenere due classi equivalenti è stata dettata dalla volontà di permettere il riutilizzo immediato dei servizi di business forniti da Poste Italiane anche a livello di modellazione. Come infatti riportato in nel documento D2.1_1, un "servizio", così come definito nel modello di Poste Italiane, può essere associato ad una "operazione" del modello di SAPI e quindi divenire un componente di una logica di business di un servizio intelligente, ovvero un'activity. L'associazione di un "servizio" ad una "activity atomica" od una "activity composta", sarà definita dal progettista in fase di definizione del servizio intelligente.

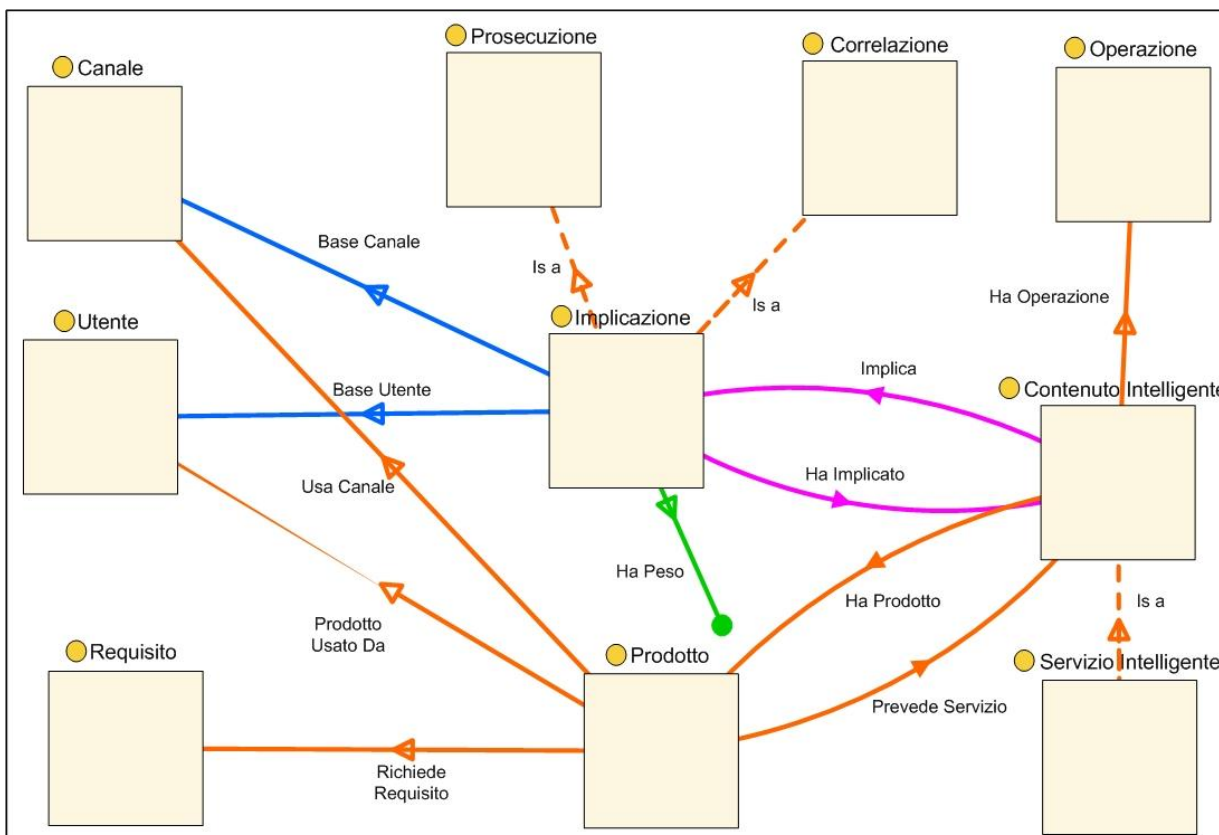


Figura 2. Modello del dominio di erogazione/fruizione.

Partendo dalla specifica del modello di composizione di un servizio intelligente, nell'ambito del progetto SAPI si vogliono valutare le possibilità offerte dalla messa a punto di un sistema predittivo che riesca ad inferire l'esigenze di un utente che usufruisce di un servizio web, anticipandone opportunamente le mosse per venire incontro ai suoi bisogni. In particolare, lo scopo delle tecniche di plan recognition applicate per la predizione della prossima azione da eseguire avrà il duplice effetto di permettere, tramite degli opportuni accorgimenti tecnici, una fruizione più veloce ed efficiente del servizio e di consentire agli utilizzatori di sperimentare nuove funzionalità, anche inattese, messe che possono risultare per loro più interessanti rispetto all'iter di interazione normalmente eseguito. Tutto questo si tradurrà in un miglioramento della *user experience* percepita dall'utente finale.

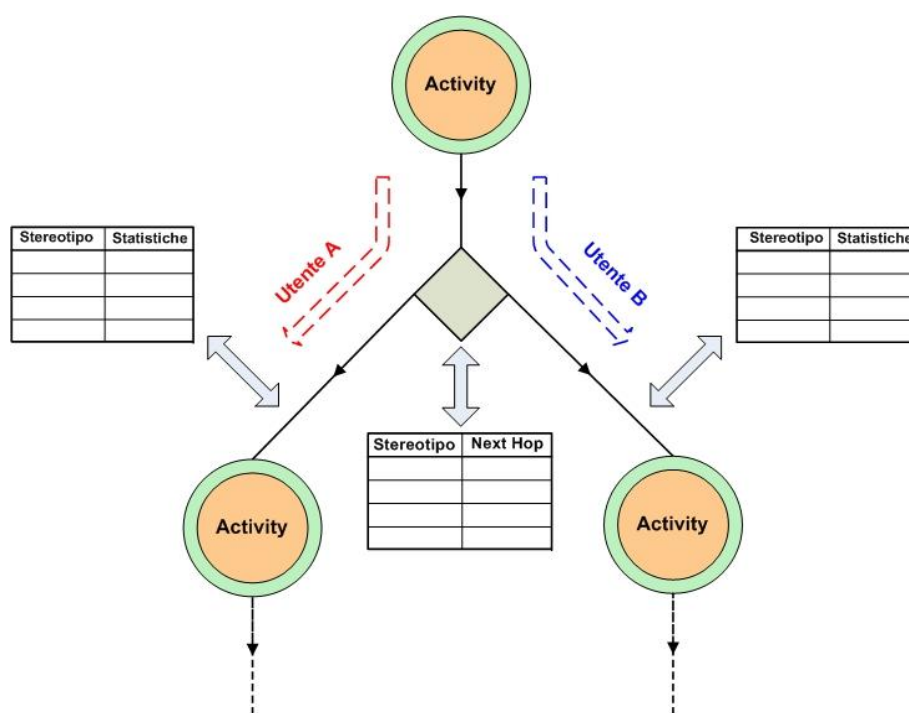


Figura 3: Esempio di predizione comportamento utente.

In SAPI si affronta il problema dell'implementazione di un sistema di predizione utilizzando un approccio innovativo rispetto a quello classico analizzato in precedenza. Infatti, i sistemi di "web request prediction" tipicamente tentano di predire la prossima pagina web che l'utente vorrà visitare partendo da quelle che ha già visitato. Tale approccio comporta il notevole svantaggio di avere una complessità che, fissati tutti gli altri parametri del modello di predizione scelto, è sicuramente proporzionale all'insieme di tutte le pagine navigabili. A ciò si aggiunge anche il fatto che, nel caso di interazione web finalizzata all'esecuzione di un particolare goal connesso al business, i percorsi di navigazione possibili sono fortemente legati al workflow di composizione della logica di business.

Partendo da queste considerazioni, in SAPI si è scelto di effettuare una predizione sulle unità di business e non sulle pagine web. In particolare, si procederà inizialmente alla risoluzione dei punti di branch previsti dalla logica di business, realizzando, quindi, un primo tipo di predizione a livello di activity e non di contenuti o pagine HTML. Partendo da questo tipo di look-ahead sarà poi possibile eseguire in cascata una predizione del percorso di navigazione di pagine web associato indirettamente alla particolare activity fino a giungere ad una pre-

compilazione automatica di quei dati che l'utente dovrebbe digitare per riempire le form previste dal modello di business. In quest'ottica i modelli e le tecniche normalmente utilizzati dai tradizionali sistemi di predizione saranno opportunamente interpretati in SAPI alla luce della nuova impostazione del problema.

A valle dello studio della letteratura sull'argomento si è deciso di impiegare un modello di predizione probabilistico ricavato a partire da una opportuna rappresentazione dello storico delle informazioni di sessione. In particolare, per la predizione effettuata a livello di business, la struttura che rappresenta le informazioni utente sarà dedotta a partire dalle istanze del modello di composizione associato all'activity composta del servizio. Per quanto riguarda poi la predizione delle pagine da sottoporre all'utente si applicherà a cascata un ulteriore modello probabilistico che farà riferimento ai pattern di navigazione della entity composta collegata alla specifica activity predetta. Si passerà a questo punto alla predizione di alcuni dati che l'utente dovrebbe digitare per consentire l'avanzamento del flusso di esecuzione di un servizio.

Un ulteriore argomento di ricerca è costituito, infine, dalla volontà di affiancare al tipico meccanismo di apprendimento di tipo history-based una tecnica di addestramento basata sulla conoscenza ontologica fornita dalle informazioni contenute nei modelli di utente ricavati dal Profiler. Questo approccio dovrebbe permettere di evitare il problema della "partenza a freddo" tipico dei sistemi di predizione che per l'addestramento si basano esclusivamente sullo storico dell'interazione utente.

L'attività di ricerca si conclude con la descrizione macroscopica delle "attività" svolte dal blocco funzionale "Deployer" in seguito alla scelta, da parte dell'utente, di un servizio intelligente di SAPI.

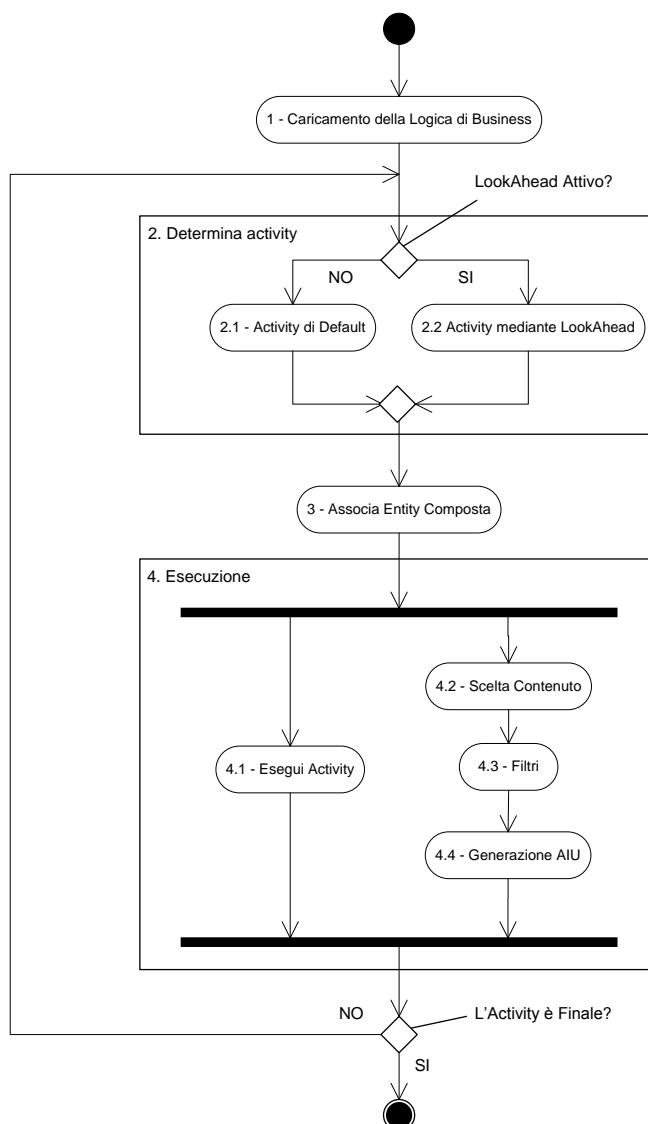


Figura 4: Activity Diagram di composizione di un servizio.

In Figura 4 è riportato l'activity diagram del processo ed una spiegazione dei passi rappresentati. A partire dallo stato iniziale, che rappresenta nel nostro caso la scelta dell'utente, viene caricata la Logica di Business corrispondente (passo 1). Il passo successivo è determinare l'activity da istanziare (passo 2). Come abbiamo già visto, le activity di un medesimo workflow sono opportunamente orchestrate attraverso un modello di composizione che potrebbe subire un processo di personalizzazione mediante l'utilizzo di tecniche di look-ahead che, in un dato istante, potrebbero essere attive o meno. Se il look-ahead non è attivo allora l'esecuzione passa alla 2.1 (si sceglie la prossima attività di default), altrimenti alla 2.2 (si sceglie l'attività suggerita dal look-ahead). Una volta scelta l'activity, il Deployer deve scegliere l'entity composta ad essa associata (passo 3). In seguito si passa all'esecuzione concorrente di activity ed entity composta (passo 4). In particolare, l'esecuzione dell'entity composta si articola in tre passi elementari ovvero la scelta dei contenuti per istanziare l'AIU (passo 4.2), l'applicazione dei filtri all'entity stessa (passo 4.3) e, infine, la generazione dell'AIU (passo 4.4). Al termine del passo di esecuzione, se l'ultima

activity eseguita è finale allora il servizio termina, altrimenti si ritorna a selezionare l'attività successiva e il processo si reitera.