

RELAZIONE DEL PROGETTO

LOGICHE DI PRODUZIONE CON SISTEMI DISTRIBUITI

CORSO DI SISTEMI DISTRIBUITI

ANNO ACCADEMICO 2012/2013

CORSO DI LAUREA MAGISTRALE IN INGEGNERIA E SCIENZE
INFORMATICHE

CRISTIANO BRAGAGNI

CRISTIANO.BRAGAGNI@STUDIO.UNIBO.IT

MATRICOLA 609819

INTRODUZIONE

Il progetto proposto per il corso di sistemi distribuiti nasce come idea in seguito ad alcune osservazioni che, nell'ambito della progettazione e gestione dei sistemi business, ho potuto constatare da vicino, a fronte della mia breve esperienza in questo tipo di settore.

Il problema principale nei prodotti software, con cui solitamente mi confronto durante le mie esperienze lavorative, è la necessaria fase di configurazione delle risorse che esso utilizza. Questa fase è spesso molto delicata e risulta indispensabile per poter garantire al sistema una sorta di base di conoscenza, sulla quale poter guidare l'utente attraverso la gestione delle risorse.

Proprio queste fasi di configurazione, in cui è necessario talvolta esplicitare ogni singola relazione, risulta spesso troppo delicata e richiede uno sforzo iniziale pesante. I sistemi sono spesso colmi di conoscenza per poter applicare complesse logiche che modificano lo stato dello stesso.

Una delle prime domande che mi sono posto, una volta completato il corso di sistemi distribuiti, era mirata a dare un modo alternativo di creare un sistema software, il quale non doveva necessariamente essere mirato a definire relazioni tramite una fase di configurazione; secondo il progetto che avevo in mente si è cercato di fornire una certa dinamicità, dove lo scambio d'informazioni può avvenire tra tutte le entità che concorrono nella vita del sistema.

Al momento della creazione del progetto si è pensato essenzialmente ad un obiettivo: *flessibilità*. L'idea di non dover conoscere tutto l'ambiente circostante per poter attuare le logiche di business è l'idea che dà spunto al progetto.

La costruzione del progetto nasce con l'impegno di voler ridurre la conoscenza che ogni parte del sistema possiede su tutto il resto che lo circonda. La ridotta conoscenza garantisce la dinamicità del sistema, anche di fronte a cambiamenti che possono incorrere all'interno dei suoi confini.

L'utilizzo di sistemi distribuiti permette di sfumare l'idea di confine del sistema; la caratteristica di questo approccio permette di abbattere i limiti dalla comunicazione, dando la possibilità di aprire la strada ad un nuovo tipo di tecnologie.

L'ambito del progetto è focalizzato sui sistemi di produzione / lavorazione di risorse. La scelta dell'ambito è abbastanza legata alle esperienze fatte, ma i principi di quanto utilizzato sono validi anche per ambiti diversi.

PROBLEMA

Come si può capire dall'introduzione, il problema trattato da questo progetto è incentrato sui sistemi che permettono la trasformazione di risorse; nello specifico viene focalizzata la parte che si dedica alla trasformazione di materie prime, le quali, selezionate e trattate, permettono di ottenere semilavorati, che allo stesso modo possono rientrare in ulteriori flussi di lavorazione.

L'ambito del problema riguarda i sistemi di produzione: all'interno dei sistemi di produzione si ha la necessità di impostare flussi di lavoro che permettono, tramite l'utilizzo di specifiche risorse, la trasformazione di una serie di beni in altri.

In ogni sistema di produzione, lo scopo principale è quello di trasformare un input in un output, secondo determinate logiche. L'idea che è stata pensata a questo tipo di scenario è proprio quella di una catena di lavorazione, in cui i beni vengono sottoposti a diverse lavorazioni, ognuna con lo scopo di alterare lo stato delle risorse in input per dare vita ad un altro tipo di risorsa in output.

Cercando di dare forma alle idee, è possibile riconoscere nelle descrizioni precedenti i concetti tipici di un sistema di produzione: in un ipotetico scenario l'azione che fa muovere il sistema è la richiesta da parte di un qualsiasi soggetto di un bene. Ogni bene è definito da un insieme di step, i quali portano alla creazione del bene richiesto. Ogni step include una serie di altre risorse, le quali andranno a comporre il bene finale, e una risorsa che effettua un trattamento sulla lista dei beni definiti richiesti precedentemente.

Riassumendo, i tipi di risorse di cui si parla sono:

- I beni che rappresentano l'input del sistema, cioè beni che il sistema accetta e che serviranno per alimentare i flussi di trasformazione degli stessi;
- I macchinari, i quali permettono di apportare le dovute trasformazioni ai beni in input, effettuando su di essi specifici trattamenti, i quali porteranno ad un nuovo tipo di risorsa in output.
- I prodotti finali, che rappresentano l'output del sistema, i quali richiedono, per la loro trasformazione, l'utilizzo sia di beni che di macchine per completare la trasformazione.

Una volta immaginata la catena di montaggio ho pensato a quei concetti che è normale proporre in un problema di questo tipo: un concetto fondamentale è un magazzino che permette lo stoccaggio di risorse in attesa di un eventuale richiesta.

Tra gli altri concetti si possono aggiungere quello di un eventuale unità che permette di introdurre nuovi beni all'interno del sistema, come ad esempio può essere un unità di accettazione merci all'interno di un sistema di produzione.

Un obiettivo del sistema, insieme a quello di creare un magazzino e un sistema di accettazione beni, sarà quello di fornire un sistema di tracciamento dello stato degli ordini, il quale permetta di poter ricostruire tutti i passaggi relativi ad un determinato ordine.

Intorno alla gestione di questi aspetti ruota l'implementazione del progetto, che mira a rendere trasparente l'associazione che avviene tra i beni e i macchinari, i quali sono a conoscenza solamente delle logiche di gestione delle proprie funzionalità.

L'insieme dei concetti e delle funzionalità sopra descritte, fornisce la base per i prerequisiti che dovrà coprire questo progetto; il tutto, inserito in un contesto di sistema distribuito, dovrà caratterizzare la creazione del progetto.

APPROCCIO SCELTO - LAVORO SVOLTO

Per riuscire a collocare ogni concetto in modo significativo all'interno del progetto, si è cercato di suddividere ogni aspetto dell'implementazione in una determinata area logica, in maniera tale da non confondere le diverse esigenze espresse in fase di definizione del problema.

Prima di poter definire un piano di lavoro, è quindi opportuno fare chiarezza sulla parte realizzativa del progetto, sia dal punto di vista di modellazione dei concetti, sia per quanto riguarda l'implementazione delle diverse funzionalità di cui verrà trattata l'implementazione.

La decisione dell'approccio si è basata sostanzialmente in quattro concetti che mi hanno permesso di dividere il problema:

- Modellazione dei concetti.

Nella modellazione dei concetti cerca di definire oggetti e relazioni di tutte le entità che all'interno del sistema verranno definite. Questa modellazione offre il supporto necessario per poter gestire la parte del sistema che comporta l'applicazione delle logiche di business necessarie.

- Definizione degli agenti.

Questo capitolo del progetto richiede di progettare gli agenti che potrebbe essere opportuno sviluppare ai fini realizzativi.

- Organizzazione del sistema e progettazione delle interfacce necessarie.

Nella parte di organizzazione viene definita la struttura grafica che permette all'utente la gestione delle risorse, la pianificazione delle lavorazioni, il controllo degli ordini e l'accettazione di nuovi elementi.

- Creazione di prototipi di accesso ai dati e di gestione delle logiche, i quali permettono sostanzialmente di conoscere gli step necessari per la trasformazione di un determinato bene in un altro.

Questa fase è richiesta per poter permettere al sistema avere uno strato dedicato di accesso ai dati. Inoltre, in questa fase vengono anche definite le configurazioni necessarie per permettere al sistema di sapere in quale modo devono essere gestiti i flussi di processo per la trasformazione delle risorse.

Anche se questi argomenti hanno poco a che fare tra di loro ho deciso di intraprendere questa strada per non confondere concetti tra di loro.

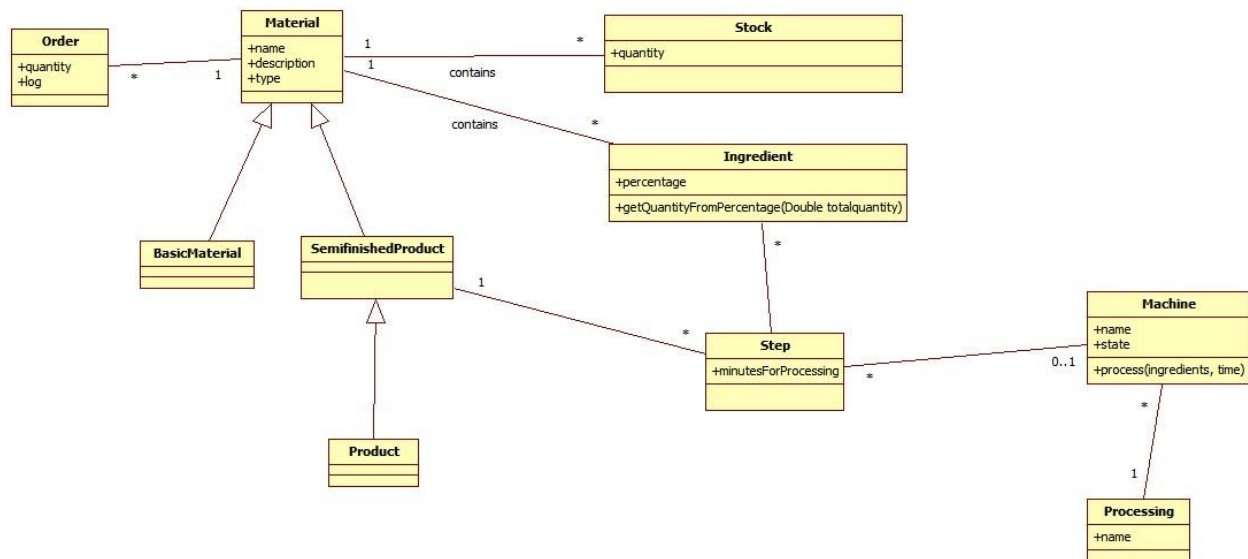
Sostanzialmente, i concetti mi aiutano a modellare le relazioni che ci sono tra le entità del problema, mentre gli agenti mi permettono di modellare entità autonome che scambiano dati tra di loro. Le logiche sono necessarie per il sistema per avere una base di conoscenza sui dati.

E' stato pensato di gestire l'interfaccia grafica perché si pensa che sia necessario dare uno strumento all'utente, magari che gli permetta di monitorare lo stato del magazzino e dei flussi.

MODELLAZIONE DEI CONCETTI

La definizione dei concetti necessari per gestire la logica del programma è il primo passo per capire in che modo affrontare il progetto che si ha di fronte. Per definire i termini con cui verrà progettato il sistema bisogna utilizzare un linguaggio di modellazione che mi permetta di avvicinare i concetti finora descritti ad una visione concreta, con lo scopo di definire una base per l'implementazione successiva.

Quindi, adottando il formalismo UML, in particolare il diagramma delle classi, espongo una rapida rappresentazione dei concetti base che andranno a comporre la base del progetto.



Definizione di ogni singola entità:

- Order: rappresenta un ordine, che non è altro che la richiesta di un determinato prodotto. Nelle realtà; la modellazione del prodotto è strutturalmente diversa, visto che esso è legato ad un insieme di altri elementi (data, cliente, lista di oggetti ordinati) che si discosta molto da questo tipo di implementazione.
- Material, definisce in maniera generale gli attributi che compongono in maniera basilare un prodotto generico. Esso viene poi specializzato da diverse classi;
- BasicMaterial, rappresenta i prodotti che hanno la possibilità di essere accettati, i quali vengono importati nel sistema direttamente; essendo prodotti di base, non hanno un flusso che permette di poterli creare;
- SemifinishedProduct: rappresentano i prodotti derivanti da trasformazioni; non è possibile importare direttamente nel sistema questi tipi di prodotti ma bisogna crearli seguendo un determinato flusso;
- Product: rappresenta il prodotto finale, quello che ha realmente valore per il cliente.
- Stock: questo tipo di rappresentazione è utile per la gestione del magazzino; permette di definire una quantità legata ad un materiale;
- Ingredient: definisce la percentuale legata ad un materiale, il senso della percentuale è legata al tipo di step, il quale indica in quale misura deve essere prelevato il prodotto sulla base della percentuale definita;
- Step: ogni semiprodotto, per essere lavorato, necessita di un insieme di lavorazioni successive; ogni lavorazione richiede un determinato numero di ingredienti. Ogni step può includere o meno una lavorazione.
- Processing: definisce un tipo di lavorazione a cui può essere sottoposto un insieme di ingredienti;
- Machine: rappresenta una macchina, la quale fornisce un tipo di lavorazione.

Questa modellazione permette di capire la base sulla quale poggia il progetto realizzato. Seppur sia molto semplice questa implementazione, un futuro progetto potrebbe essere quello di aggiungere concetti in maniera tale da migliorare il lavoro svolto.

DEFINIZIONE DEGLI AGENTI

La modellazione degli agenti è sicuramente una delle attività che non era mai stata svolta prima; ciononostante si è cercato di fornire una propria implementazione sensata della struttura di questi nuovi oggetti attivi del sistema.

Innanzitutto, occorre precisare che la modellazione degli agenti ha seguito molto la modellazione del diagramma delle classi; infatti, dove è stato possibile, si è cercato di associare ad una entità che poteva avere un ruolo definito, indipendente e attivo, con un corrispondente agente che permetteva di aggiungere funzionalità all'oggetto, sfruttando appunto la differenza che c'è in queste metodologie di progettazione.

All'atto della definizione del progetto si è scelto di utilizzare come modello di coordinazione di agenti distribuiti il progetto che porta il nome di TuCSon; questo modello offre un sistema di meta-coordinazione che si è scelto di sfruttare per permettere un sistema reattivo e gestito da operazioni volte a coordinare gli agenti del sistema.

L'analisi è stata svolta concentrandosi sulla definizione di oggetti autonomi che permettevano una modellazione sensata in accordo con lo scopo del progetto. Quindi, si è iniziato analizzando le attività che potevano avere una possibile gestione autonoma.

Tra le entità è stato possibile riconoscere:

- Un agente che potesse modellare la dinamicità di un macchinario, indipendente per quanto riguarda la gestione e l'acquisizione degli ingredienti da lavorare;
- Un agente in grado di gestire un ordine, dalla creazione al suo completamento.

Immaginando la progettazione del sistema, è facile immaginare l'implementazione del macchinario, mentre risulta meno definita quella dell'agente che gestisce l'ordine: semplificare l'agente che esegue l'ordine è necessario perché permette di semplificare la complessità del sistema generale.

Viste le precedenti valutazioni, è stato deciso di definire ogni agente per tipo di prodotto da gestire. Questo soluzione deriva dalla natura diversa nella gestione delle tipologie di prodotto: la presenza di prodotti finiti, semilavorati e prodotti base non permette di poter creare un solo agente per tutti e tre, anche se le implementazioni cercheranno di sfruttare gli aspetti comuni tra i diversi oggetti.

Oltre agli agenti attivi del sistema, si è poi pensato a quegli agenti che permettono di gestire i messaggi provenienti da questi: dunque sembra che ci sia l'esigenza di creare un insieme di agenti che, insieme al sistema di coordinazione, permettano di coordinare le attività e informare l'utente finale sullo stato del sistema.

Elenchiamo quindi la scelta finale sugli agenti che sono stato definiti:

- BasicMaterialAgent: gestisce i messaggi per l'accettazione del prodotto;
- ProductAgent/ SemifinishedProductAgent: gestisce la richiesta delle varie risorse e la comunicazione necessaria per la richiesta di lavorazione da parte di una macchina; la differenza tra i due è che il prodotto aggiunge funzionalità di notifica dell'avvenuta creazione del prodotto;
- MachineAgent: gestisce la macchina, dall'accettazione di una lavorazione, alla lettura degli ingredienti, fino alla trasformazione del prodotto;
- ListenerMachineAgent: gestisce i messaggi che permettono all'utente di capire lo stato delle macchine;
- ListenerOrderAgent: riceve i messaggi sullo stato dell'avanzamento degli ordini;
- WarehouseAgent: in realtà permette tramite due agenti di gestire nuovi arrivi di merce e nuove richieste di materiale; comprende la gestione del magazzino;

Un ultimo appunto per quanto riguarda gli agenti va rivolto alla gestione del meta-modello: è stato utilizzato soprattutto per migliorare la gestione riguardante lo scambio dei messaggi tra i diversi agenti del sistema.

ORGANIZZAZIONE DEL SISTEMA

E' in questa fase del progetto che viene definita la divisione netta tra l'ambiente su cui girano determinati agenti e il ruolo che si è cercato di dare al modello.

In questa fase si è deciso di dividere il sistema nei seguenti modi:

- Un'applicazione principale, che permette di valutare lo stato del sistema, gestire il magazzino, accettare le richieste di nuovi ordini, valutare lo stato degli ordini;
- Tante applicazioni distribuite che rappresentano le macchine di lavorazione prodotti, che tramite la comunicazione di messaggi si registrano al nodo, e attendono richieste di lavorazione.

Per ogni tipo di applicazione sono state definite le funzionalità minime che il sistema deve essere in grado di garantire, e, partendo da queste, sono state elencate le maschere necessarie per dare all'utente la possibilità di implementare i casi d'uso riguardanti il sistema.

Le maschere necessarie sono:

- Applicazione principale:
 - Home: rappresenta sinteticamente lo stato del sistema;
 - Inbound: permette accettare prodotti;
 - Outbound: permette di lanciare ordini;
 - Stato ordini: permette di consultare i log per controllare lo stato degli ordini;
 - Stato macchine: permette di controllare lo stato delle macchine;
 - Warehouse: mostra i depositi di materiale presente in magazzino;
- Applicazione macchina:
 - Registrarsi ad un nodo;
 - Home che mostra il log sullo stato della macchina

Questo rappresenta l'elenco delle maschere che permettono una gestione a 360° del sistema di produzione, almeno per quanto riguarda la porzione del sottosistema che è stata analizzata.

GESTIONE DELLE LOGICHE

La richiesta di questa parte è importante per definire la conoscenza basilare del sistema per permettere di reagire in maniera sensata alle richieste provenienti dagli utenti esterni; concretamente, consistono in un insieme di classi che si possono dividere due package:

- Dal, strato del software con il quale viene definito l'accesso ai dati che compongono la parte informativa del sistema.
- Biz, struttura che permette di applicare le dovute logiche ai dati provenienti dallo strato logico.

Questi due strati insieme hanno l'ambizione di rendere indipendente l'accesso ai dati dalla logica che viene applicata su di essi. In realtà consistono in una semplice creazione di oggetti che verranno poi gestiti dai livelli superiori.

Questa parte serve per definire i flussi di lavorazione dei prodotti, quindi serve esclusivamente per definire agli agenti il tipo di operazioni che possono effettuare.

LAVORO SVOLTO

Grazie alla fase in cui è stato definito l'approccio scelto, non è bastato altro che sviluppare tutti i concetti che sono stati definiti precedentemente. In realtà, non solo la definizione dei modelli precedenti bastava alla creazione del progetto: infatti, alcune sezioni importanti dovevano ancora essere definite.

Il lavoro si è diviso nella progettazione dell'interfaccia utente pianificatore e nella progettazione di quella supervisore della macchina. Le due implementazioni hanno avuto in comune solo lo sviluppo del modello dei dati.

Una volta progettate le entità e definite le loro funzionalità, si rende necessario capire come queste interagiscono tra di loro, quali sono i messaggi che gestiscono e quale ruolo hanno all'interno del sistema.

Per riuscire a spiegare su quale linea è stato sviluppato il progetto si rende necessario spiegare un tipico esempio di gestione del flusso di un ordine.

1. L'utente richiede la gestione di un nuovo ordine al magazzino: tale ordine consiste nella definizione di un prodotto da creare e una quantità da produrre;
2. Il magazzino accetta l'ordine e fa partire la produzione del suddetto;
3. Una volta che un ordine viene attivato viene lanciato il rispettivo agente. Un prodotto si compone da un insieme di step; ogni step è definito da un insieme di risorse e, opzionalmente, un macchinario che esegue una trasformazione.
 - a. Ogni risorsa necessaria viene richiesta al magazzino; le risorse già presenti vengono prelevate, mentre quelle non presenti vengono gestite dipendentemente dal prodotto: se il prodotto è un prodotto base lo si attende in accettazione; se il prodotto è un semilavorato si lancia il rispettivo agente.
 - b. Una volta che tutte le risorse dello step sono state accettate, si aspetta la disponibilità di un agente macchina che gestisca le risorse per la lavorazione.
 - c. Una volta finita la lavorazione, si procede con lo step successivo; finiti gli step il prodotto è disponibile.

Con il modello di gestione del flusso visto sopra si è iniziata l'implementazione degli agenti; si è cercato di fornire far comunicare gli agenti, e, quando possibile, si è cercato di utilizzare il meta-modello di coordinazione per gestire una migliore sincronia tra lo scambio di messaggi tra gli agenti.

La maggior parte del lavoro è stato speso nella migliore definizione di questa coordinazione tra agenti: la presenza di questi agenti all'interno del progetto permette la gestione dei flussi necessari allo sviluppo delle problematiche espresse nella fase di definizione del progetto.

RISULTATO

Il risultato è sicuramente aderente ai requisiti erano stati fissati all'atto della definizione del progetto; l'utilizzo di agenti risultano concretamente centrato, sia per quanto riguarda la gestione delle macchine, sia per quanto concerne la creazione dei prodotti.

Dal punto di vista operativo, quindi, il progetto presenta aspetti sicuramente nuovi, anche introdotti dall'utilizzo di tecnologie come TuCSoN che presentano funzionalità appropriate allo scopo finale del progetto.

Dal lato dell'usabilità il progetto ha cercato di dare una visione globale sullo stato del sistema, fornendo un'interfaccia intuitiva e semplice, in grado di attivare le logiche che permettono al sistema di poter trasferire e scambiare informazioni con gli agenti che si registrano al nodo centrale.

Dal punto di vista dell'utente, quindi, il progetto definisce una gestione completa per quanto riguarda le funzionalità che sono state implementate.