



UNIVERSITA' DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

# **Automazione del ciclo di vita di una richiesta di servizio per un'azienda "Service Provider"**

*Laureando:*  
Nicola CHAHINIAN

*Relatore:*  
Prof. Sergio CONGIU

Anno accademico 2011/2012

Ai miei genitori

# Indice

<b>1</b>	<b>Introduzione</b>	5
1.1	Scopo del documento	5
1.2	Descrizione dello stage	5
1.3	Obiettivi dello stage	5
1.4	Tempistiche di svolgimento	6
<b>2</b>	<b>L'azienda: Tempestive s.r.l.</b>	7
2.1	Descrizione dell'azienda	7
2.2	Aree di business	7
2.3	Tipologie di offerta	9
2.3.1	Assistenza/Consulenza generica	9
2.3.2	Progetto a corpo	9
2.3.3	Offerta di vendita software e licenze	9
2.4	Produzioni aziendali	9
<b>3</b>	<b>Software e applicativi utilizzati durante lo stage</b>	11
3.1	IBM <i>Maximo Asset Management</i> : una soluzione per la gestione degli asset aziendali	11
3.1.1	Descrizione	11
3.1.2	Relazioni con il database	13
3.1.3	<i>WebSphere Application Server</i>	13
3.2	IBM <i>Maximo for Service Provider</i>	14
3.2.1	Descrizione	14
3.2.2	Processo di fatturazione in Maximo	16
3.2.3	Il <i>Customer Agreement</i>	18
3.2.4	I <i>work order</i> sul Service Provider	19
3.2.5	Piano di risposta	19
3.2.6	<i>Bill Batch</i> : la fatturazione	20
3.2.7	Altre applicazioni	22
3.3	La creazione dei report: <i>BIRT Report Designer</i>	23
3.4	<i>Tempestive Activity</i> : l'organizzazione delle attività di un'azienda	24
<b>4</b>	<b>Censimento delle offerte</b>	27
4.1	Esempio 1: interventi routinari o di emergenza	27
4.1.1	Scenario	27
4.1.2	Implementazione	27
4.1.3	Consuntivazione e fatturazione	34
4.2	Esempio 2: il subappalto dei servizi	35
4.2.1	Scenario	35
4.2.2	Implementazione	35
4.2.3	Consuntivazione e fatturazione	38
4.3	Offerte aziendali	40
4.3.1	Offerta di consulenza generica	40
4.3.1.1	Dettagli dell'offerta	40
4.3.1.2	Implementazione	41
4.3.2	Progetto a corpo	44
4.3.2.1	Dettagli dell'offerta	44
4.3.2.2	Implementazione	45

4.3.3	Osservazioni personali .....	47
4.4	Creazione automatica di fatture e report per l'azienda .....	47
4.4.1	Design .....	47
4.4.2	Maximo e i report .....	51
<b>5</b>	<b>Progetto finale</b> .....	<b>52</b>
5.1	Premessa al progetto .....	52
5.2	Ciclo delle offerte su Activity .....	52
5.3	Analisi dei requisiti .....	53
5.3.1	Requisiti di sistema .....	53
5.3.2	Requisiti funzionali .....	54
5.3.3	Postazione di lavoro .....	54
5.4	Descrizione del progetto .....	55
5.4.1	Modalità di implementazione .....	55
5.4.2	Tipologie di offerta implementate .....	55
5.4.3	Personale addetto .....	55
5.5	Casi di utilizzo .....	56
5.5.1	Clienti e progetti .....	56
5.5.2	Offerte per consulenze di vario genere .....	57
5.5.3	Progetti a corpo .....	59
5.5.4	Fatturazione .....	60
5.6	Schemi di funzionamento .....	61
5.6.1	Offerta per consulenze .....	62
5.6.2	Progetto a corpo .....	63
5.7	Importazione di dati su Maximo: il MIF .....	64
5.7.1	Architettura .....	64
5.7.2	Trasmissione dei dati tramite <i>interface table</i> .....	66
5.8	Implementazione .....	67
5.8.1	Descrizione generale .....	67
5.8.2	Clienti e progetti .....	68
5.8.3	<i>WorkOrder</i> : gestione degli ordini di lavoro per offerte diverse .....	69
5.8.4	Report e consuntivazione .....	74
5.9	Test finali .....	76
5.9.1	Esecuzione .....	76
<b>6</b>	<b>Conclusioni</b> .....	<b>82</b>
6.1	Conclusioni sul progetto .....	82
6.2	Conclusioni sullo stage .....	83
6.3	Conclusioni sull'azienda .....	84
<b>7</b>	<b>Bibliografia</b> .....	<b>85</b>

## Capitolo 1

# Introduzione

### 1.1 Scopo del documento

Questa tesi rappresenta una relazione sullo stage svolto dallo studente presso l'azienda Tempestive s.r.l.(ex Santin & Associati s.r.l.) della durata di 500 ore nel periodo che va dall'1/9/2011 al 29/02/2012.

Inizialmente verrà descritta l'azienda ospitante lo stage, per poi proseguire nella descrizione del lavoro di censimento delle offerte su IBM Maximo e terminare con la descrizione di un progetto finale. Non mancheranno le considerazioni finali dello studente sugli obiettivi raggiunti o meno.

### 1.2 Descrizione dello stage

Lo stage presso Tempestive s.r.l. prevedeva la crescita dello studente nella conoscenza delle dinamiche aziendali legate all'utilizzo del software *IBM Maximo*. In particolare lo studio doveva essere mirato alla conoscenza del pacchetto di applicazioni aggiuntivo *IBM Maximo for Service Provider*, ancora in gran parte sconosciuto ai membri dell'azienda. Successivamente questo studio si sarebbe reso utile nell'implementazione di un progetto di integrazione fra il suddetto software e l'applicazione *Tempestive Activity*, prodotta dall'azienda stessa.

### 1.3 Obiettivi dello stage

Lo stage presso Tempestive s.r.l. prevedeva sostanzialmente 3 obiettivi generali da raggiungere, ognuno dei quali richiedeva il completamento del precedente per poter essere portato a termine:

- 1) conoscenza del software IBM Maximo Asset Management e delle relative logiche di funzionamento, nonché conoscenza e capacità di utilizzo delle applicazioni aggiuntive legate al pacchetto IBM Maximo for Service Provider, insieme alle relative dinamiche tipiche di un'azienda fornitrice di servizi : la prova di ciò doveva essere data dal censimento all'interno di Maximo di alcune offerte, fra le quali alcune tipiche dell'azienda ospitante lo stage agente in qualità di "service provider";
- 2) capacità di utilizzo del software BIRT Report Designer per la creazione automatica di layout fissi per i report: la prova di ciò doveva essere data dalla creazione di due layout esattamente identici a quelli utilizzati dall'azienda per emettere fatture e stampare i report delle attività svolte per i clienti;
- 3) capacità di far dialogare Maximo con altri software o applicazioni esterni attraverso il modulo di integrazione in esso contenuto: un progetto finale di integrazione con l'applicazione Tempestive Activity doveva dar prova di tale capacità.

## **1.4 Tempistiche di svolgimento**

Lo stage prevedeva che i tre obiettivi esposti nel paragrafo precedente fossero completati all'interno dei 6 mesi senza delle particolari scadenze per ognuno, se non la clausola di raggiungimento dell'obiettivo precedente per passare al successivo.

Le tempistiche sono state le seguenti:

- Obiettivo 1: tre mesi per il completamento;
- Obiettivo 2: un mese per il completamento;
- Obiettivo 3: due mesi per il completamento;

## Capitolo 2

# L'azienda: Tempestive s.r.l.

### 2.1 Descrizione dell'azienda

**Tempestive s.r.l.** (ex Santin e Associati s.r.l.) nasce come società di consulenza IT. Ha una consolidata esperienza nello sviluppo di software e di integrazione di sistema. Progetta, realizza, gestisce e distribuisce soluzioni basate su software e sistemi distribuiti, SOA (Services Oriented Architecture) e BPM (Business Process Management), sistemi di collaborazione per l'information worker, soluzioni di e-Business, commercio elettronico, sistemi di gestione asset, integrazione di sistemi, sistemi embedded, Internet/Intranet, Java, Windows, .NET e soluzioni cloud (servizi IAAS,PAAS,SAAS)

In grado di gestire e integrare diversi tipi di tecnologie, Tempestive fornisce i suoi servizi e le sue consulenze a imprese di diverse aree di business, senza tralasciare la produzione e lo sviluppo di prodotti, alcuni dei quali dedicati a professionisti IT. Questi prodotti sono molto utili per accelerare lo sviluppo di applicazioni sofisticate per le persone non programmatori o che hanno consegne a breve termine.

Tempestive vanta esperienza in molti settori quali il settore bancario, finanziario, assicurativo, logistico, manifatturiero e della pubblica amministrazione. I clienti sono di varie dimensioni.

EAM (Enterprise Asset Management) è un'altra area in cui l'azienda ha acquisito importante esperienza; il suo personale si dedica ad EAM specializzate per servizi di pubblica utilità, petrolio e gas, facility management, norme di sicurezza.

### 2.2 Aree di business

Descriviamo in breve alcuni settori per i quali l'azienda fornisce le proprie soluzioni e le proprie consulenze.

#### *Settore bancario*

Inizialmente venivano operate soluzioni collegate al sistema centrale (porte, transazioni), consentendo all'azienda di utilizzare e integrare sistemi come IBM zOS, CICS, COBOL, IMS, MQ o sistemi più moderni come LINUX/390 e Windows. Tempestive fu tra le prime ad introdurre tecnologie quali J2EE e IBM WebSphere.

Con l'aumento delle imprese nel settore bancario è stato sempre più difficile trasmettere il codice scritto direttamente sui sistemi centrali. Le soluzioni si sono



quindi concentrate su sezioni estremamente importanti, ma trascurate fino ad oggi, come la gestione dei processi documentali, intranet, applicazioni verticali per la gestione del bilancio, per la gestione della sicurezza nei luoghi di lavoro, per comunicare con i clienti e molto altro ancora.

### Alcune soluzioni

- Multi-channel Internet banking.
- Sistema di comunicazione massiva ai clienti via SMS.
- Gestione formato elettronico
- Sistemi di storage
- Sistemi per la gestione dei soci e dei meeting

### Settore Manifatturiero

Piccole e medie imprese spesso sono innovative e dinamiche. Le loro armi sono flessibilità ed efficienza e le loro piccole dimensioni sono sia la loro forza che la loro debolezza. Durante gli anni Tempeptive ha sviluppato diverse soluzioni per le aziende del settore manifatturiero. Il punto di partenza è sempre l'automazione di un processo di business, da cui si arriva poi a potenziare svariate tipologie di soluzioni. Questo tipo di approccio legato al business process non concepisce singole soluzioni, ma soluzioni che vanno integrate. Le soluzioni emerse sono estremamente flessibili e adatte ad aree come quelle del MES (Manufacturing Execution System), SCM (Supply Chain Management), la logistica, CRM (CustomerRelationship Management), e-commerce.

### Alcune Soluzioni per la produzione

- Sistemi di tracciabilità in vari settori (compresi cibo e farmaceutico).
- Portali per la forza vendita e clienti.
- Portali per la produzione aziendale (controllo, efficienza, tracciabilità, ecc.).
- Gestione e acquisizione dati e di sistemi di monitoraggio.

### Smarter Planet

Siamo immersi nell'era digitale perchè ci sono transistor ovunque. Quest'anno più di 2 miliardi di persone nel mondo useranno internet, ma non finisce qui perchè internet è utilizzato da trilioni di apparecchiature che sono collegate e comunicano tra di loro. Ogni giorno, sono generate 15 petabytes di nuove informazioni, circa 8 volte superiore alle informazioni contenute in tutte le biblioteche degli Stati Uniti. Telefoni cellulari, strade, condutture, edifici, piantagioni e animali... tutto crea informazioni utili.

Collegando i sistemi di back-end si è in grado di convertire tutta questa intelligenza, acquisire maggiore conoscenza, agire in modo migliore e più velocemente. Ancor prima che si verifichino i problemi.

Un'azienda come questa, con esperienza nei sistemi integrati e in integrazione di ambienti eterogenei, può permettere ad ogni organizzazione di diventare più smart, migliorando i propri processi, producendo di più e consumando meno, per un mondo migliore.

### Cloud Computing

Potrebbe sembrare che il Cloud Computing sia solo per il personale IT, contrariamente esso è molto utile anche per l'utente finale. Nel passato si realizzavano sistemi bilanciati, resistenti e sicuri





che erano però limitati all'hardware in cui erano installati. Utilizzando il Cloud questo problema e questo limite vengono superati offrendo al personale IT e al business nuove possibilità.

Nell'area dello Smarter Planet, il Cloud Computing offre l'energia necessaria per maneggiare tutti i dati che provengono dal mondo reale e quest'azienda produce e gestisce sistemi che utilizzano questa tecnologia.

## **2.3 Tipologie di offerta**

Descriviamo qua di seguito le tipologie di offerta più frequenti fatte da Tempestive s.r.l. ai propri clienti.

### **2.3.1 Assistenza/Consulenza generica**

Un offerta di questo tipo prevede interventi di vario genere e misura per aziende che non dispongono di personale preparato ad operazioni di tipo sistemistico o di programmazione. L'intervento viene fatto generalmente in presenza di modifiche ad un lavoro eseguito in precedenza, di assistenza sull'uso di un prodotto venduto, di manutenzione di un portale o di un software implementato precedentemente per il cliente. Il valore dell'operazione può essere pesato in base alla figura professionale impiegata nel lavoro, oppure al numero di ore lavorate o di "giornate-uomo"(giornate lavorative complete) impiegate.

### **2.3.2 Progetto a corpo**

Nella maggior parte dei casi si tratta dell'implementazione di una soluzione per soddisfare le esigenze dell'azienda cliente: Portali, Applicazioni Web, Integrazioni fra software e molto altro.

Un progetto di questo tipo viene sviluppato e fatturato in fasi, ognuna delle quali risponde a ben determinate specifiche progettuali.

In termini economici, il valore di un Progetto a Corpo viene valutato in base al numero di "giornate-uomo" impiegate per l'intero lavoro, alla difficoltà nella sua realizzazione e alla quantità di personale impiegata nei lavori.

### **2.3.3 Offerta di vendita software e licenze**

Il software, da semplice strumento di supporto, sta diventando sempre più una vera e propria risorsa strategica da gestire in azienda all'interno di un processo strutturato, noto come Software Asset Management(SAM). A causa delle numerose opzioni di "licensing" disponibili e di database storicamente disallineati e disordinati, la gestione delle licenze software può trasformarsi in un'attività molto impegnativa. Quest'azienda è partner ufficiale di aziende quali Microsoft, IBM e Symantec e si offre come intermediario per la vendita dei relativi software e la gestione delle licenze annesse.

## 2.4 Produzioni aziendali

Elenchiamo alcune applicazioni create dall'azienda stessa, applicazioni vendute e adattate alle esigenze dei propri clienti.

### *Tempestive Activity*

Activity è un'applicazione pensata per le organizzazioni che hanno l'esigenza di verificare il tempo che dedicano allo sviluppo delle attività o di progetti. Ogni azienda infatti, dalla più piccola alla più grande, ha l'esigenza di registrare e gestire le attività svolte dal personale e di monitorare l'efficienza della propria struttura.

Quest'applicazione sarà oggetto di studio e di utilizzo all'interno del progetto finale nel capitolo 5.

### *Tempestive DbPortlet*

DbPortlet è stata studiata per permettere l'accesso ai database in via sicura e guidata. Ne rende disponibili i dati per ogni utente e permette di lavorare su di essi con l'utilizzo di funzionalità pre-configurate. Queste sono totalmente standardizzate allo scopo di ridurre i costi di apprendimento, indipendentemente dai dati che vengono gestiti dagli utenti.

DbPortlet permette inoltre agli utenti evoluti di creare dei portlet per l'accesso ai dati aziendali senza l'utilizzo di linguaggi di programmazione: con alcuni semplici passaggi, è possibile configurare ogni aspetto riguardante i dati e la loro rappresentazione sul portale.

### *Tempestive DbWebPart*

DbWebPart è una "WebPart" creata per Microsoft Sharepoint, ma che funziona anche con applicazioni ASP.NET. Fornisce accesso ai database relazionali e ad altre sorgenti in maniera semplice e veloce, supportando Microsoft SQL Server e tutti i database con l'utilizzo di driver OLE DB o ODBC.

Esso permette l'accesso e la modifica delle informazioni aziendali senza scrivere una linea di codice. Inoltre garantisce la gestione della sicurezza, l'import e l'export di dati, la generazione di documenti in formato PDF o Word, e molto altro. In più si integra con Microsoft Word 2007 e 2010.

Questa applicazione verrà utilizzata per la personalizzazione di Tempestive Activity nel progetto descritto al capitolo 5.

### *Tempestive Ghost Writer*

Ghost Writer è un potente e flessibile generatore di documenti. Con l'utilizzo di modelli di documenti (template), variabili, un semplice motore di regole e connessioni a servizi di dati o alle LOB, si possono generare complessi documenti OpenXML (.DOCX) come contratti, report, offerte, ecc..

### *Tempestive Report Portlet*

ReportPortlet è un portlet per IBM WebSphere Portal che permette di generare dei report utilizzando i dati archiviati nelle tabelle dei database o recuperati da una classe java. I report possono essere generati in diversi formati: pdf, html, xls, csv. Il portlet rileva automaticamente se il report richiede dei parametri da inserire e visualizza una pagina dove l'utente deve inserire i valori adeguati.

## Capitolo 3

# Software e applicativi utilizzati durante lo stage

Nel seguente capitolo descriveremo i software e le applicazioni utilizzate durante lo stage. Più dettagliata rispetto alle altre sarà la descrizione delle funzionalità di IBM Maximo for Service Provider, software centrale nell'evolversi del percorso di stage.

### 3.1 IBM Maximo Asset Management: una soluzione per la gestione degli asset aziendali

#### 3.1.1 Descrizione

Dal punto di vista funzionale, Maximo è un software modulare per gestire asset, servizi, contratti, risorse umane, materiali e approvvigionamenti. Al fine di permettere alle aziende che lo usano di massimizzare il ritorno di investimento, Maximo Asset Management consente di sviluppare programmi completi di manutenzione preventiva, predittiva, routinaria e non pianificata.

La sua caratteristica principale è quella di aiutare l'azienda a classificare i propri asset aziendali (edifici, impianti, macchinari, materiali di supporto alla produzione o all'amministrazione, ecc...) ed a seguirli nel loro ciclo di vita, in particolare nella loro manutenzione attraverso la gestione degli ordini di lavoro e delle risorse necessarie per eseguirli (risorse umane, materiali, equipaggiamenti, rischi associati, SLA previsti ecc.). Particolare attenzione è data alla gestione della sicurezza sul lavoro e del rispetto

dell'ambientale secondo leggi e normative attualmente in vigore. Per supportare tutto questo Maximo contiene, oltre a quanto già accennato, una serie di moduli per la gestione del ciclo passivo (Richieste di Acquisto, Ordini di

Acquisto, Fatture, asset, materiali per la manutenzione, equipaggiamenti ecc.) e per la gestione delle richieste di servizio (utile per esempio nel post vendita per segnalazioni, guasti ad un call center o ad un ufficio preposto alla manutenzione presso i clienti o gli uffici interni).

Il software, pur adattandosi a qualsiasi azienda, è progettato per grandi realtà e per questo contiene un sofisticato sistema di gestione utenti e ruoli autorizzativi, un sistema di workflow delle attività e comprende un tool di integrazione con il quale lo si può integrare con sistemi esterni, primi tra tutti i gestionali più diffusi (per esempio SAP).

Maximo Asset Management consente di gestire operazioni di asset end-to-end e processi aziendali per offrire servizi efficienti ed efficaci in linea con i propri obiettivi aziendali.

E' dotato di un'interfaccia utente "user-friendly" avviabile da browser, con una home page dove è possibile tenere sotto controllo i record o i valori ritenuti più importanti.



E' possibile stabilire dei KPI (Key Performance Indicator) per monitorare le condizioni degli asset e attivare un'azione automatizzata basata su modifiche. È possibile creare, progettare, monitorare, notificare e creare report su componenti di processi chiave tra cui ordini di lavoro, ticket del service desk e ordini di acquisto, incluso lo stato, dall'inizio alla fine. È inoltre possibile includere allegati, tra cui mappe, immagini e URL a ciascun record o attività per migliorare ulteriormente la comunicazione e la produttività.

Maximo consente di identificare il potenziale non utilizzato, ottenere le informazioni necessarie a far convergere gli obiettivi aziendali e gli orientamenti strategici, per ciascun settore del business. Si possono così ridurre i costi di esercizio, minimizzare i rischi ed aumentare la reattività ed i ricavi.

Riassumendo, la sua applicazione nel mondo del lavoro è legata ad aziende con diversi tipi di esigenze. Può essere utilizzato da coloro che:

- hanno esigenza di gestire un insieme di asset in tutte le loro caratteristiche;
- richiedono un controllo accurato degli ordini di lavoro;
- hanno bisogno di una catalogazione efficiente dei prodotti del magazzino;
- vogliono automatizzare il processo delle richieste di servizio attraverso workflow ben determinati;
- vogliono regolare i processi di produzione in base a degli indici statistici di efficienza (KPI);
- hanno esigenza di organizzare dei cicli di manutenzione di diverso tipo per i propri strumenti o asset;
- richiedono sempre una soluzione a tutti i possibili problemi che si possono verificare, in modo da incrementare la sicurezza sul lavoro e sia prevenire che curare gli eventi inaspettati;
- hanno bisogno di un processo di reportistica efficiente;
- molto altro.

Dal punto di vista tecnico, Maximo è un'applicazione J2EE a tre livelli, con un'interfaccia web compatibile con Internet Explorer, una logica applicativa che è eseguita su un Application Server (tipicamente IBM WebSphere Application Server ma in origine era BEA Web Logic) e un sistema di persistenza dati basato su database relazionale (tipicamente IBM DB2, ma può anche essere Oracle o Sql Server) che ora andremo ad analizzare.



IBM WebSphere Application Server consente di aumentare la reattività di business, fornendo a sviluppatori e architetti IT una soluzione per sviluppare, riutilizzare, gestire e integrare applicazioni e servizi SOA (Service Oriented Architecture).

I dati e le applicazioni sono protetti da eventuali attacchi: questo attraverso configurazioni di sicurezza e registro utenti integrati, conformità con gli standard di settore e funzioni di sicurezza dei web service avanzate. L'implementazione completa di Kerberos migliora l'interoperabilità della sicurezza con altre applicazioni e ambienti, aumentando la produttività degli sviluppatori.

WebSphere riduce costi e tempi di inattività per consolidare attività amministrative, workload e infrastrutture con loadbalancing e failover dei server Web potenziati.

Inoltre massimizza la produttività degli sviluppatori con funzioni di compatibilità esclusive che eliminano la necessità di riprogrammare le applicazioni attraverso l'utilizzo di WebSphere Application Server Tools Editions (US).



## 3.2 IBM Maximo for Service Provider

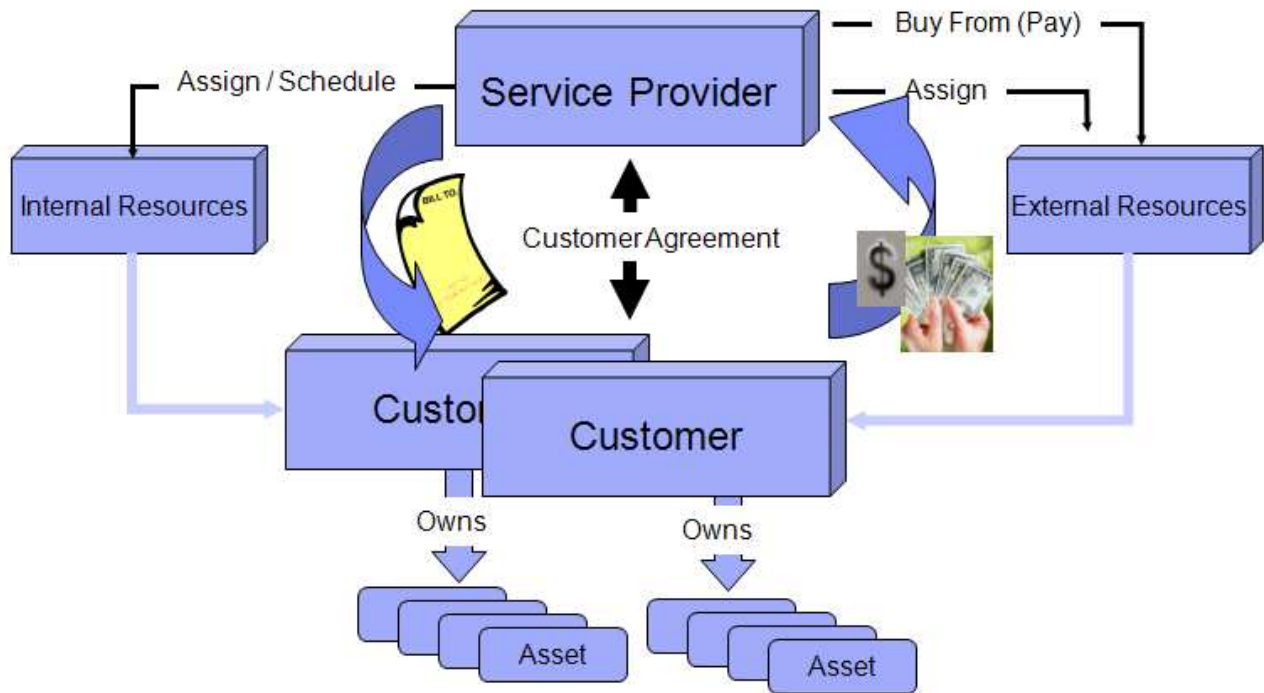
La descrizione di questo insieme di applicativi sarà svolta in maniera più particolareggiata rispetto alle altre applicazioni o software. Questo perché rappresentano il fulcro centrale del ciclo di vita di richieste di servizio da parte dei clienti e del ciclo di fatturazione finale riferito ad essi (argomento principale della presente relazione).

### 3.2.1 Descrizione

Maximo For Service Provider è caratterizzato da un insieme di applicazioni, che vanno ad aggiungersi all'installazione di Maximo Asset Management di base, volte alla gestione di asset e servizi offerti ai propri clienti da parte di un'azienda "fornitrice di servizi" (*service provider*).

Un'azienda "service provider" focalizza la propria attenzione su due tipologie di clientela: i reparti interni dell'azienda stessa e i clienti esterni. Per quanto riguarda i primi, l'obiettivo è che questi dipartimenti possano giustificare il proprio valore complessivo dimezzando i costi e migliorando i servizi forniti (l'azienda in questo caso agisce da "internal service provider"). Per i secondi, invece, si parla di aziende esterne che usufruiscono dei servizi offerti loro dal service provider: l'obiettivo è fornire servizi efficienti a costi ridotti, limitando l'erogazione di servizi gratuiti, grazie a una completa gestione degli ordini di lavoro e di vendita (in tal caso l'azienda è un "external service provider").

Proprio per quest'ultima categoria di clienti, si fornisce il seguente schema concettuale.



[Figura 3.2: modello di business di un Service Provider]

Lo schema mostra come avviene il ciclo di fornitura-pagamento dei servizi. Il Service Provider può offrire ai propri clienti proprie risorse (manodopera, prodotti di cui è proprietaria, beni di consumo creati da se medesima, ecc.) oppure risorse che acquista da fornitori di terze parti. Ciò che regola i relativi termini di utilizzo è l'offerta (il "Customer Agreement") che l'azienda fa al proprio cliente che, in quanto utilizzatore dei servizi, paga al Service Provider quanto è stato pattuito nel contratto.

Per fare tutto ciò in maniera efficiente, Maximo for Service Provider racchiude in se 4 capacità fondamentali:

- permette di ridurre il TCO<sup>1</sup> e i costi di proprietà, garantendo integrità e sicurezza dei dati;
- permette di gestire più clienti con più locazioni fisiche, redigendo un unico contratto per cliente e fornendo regole ben precise per definire i diritti sui servizi;
- emette fatture dettagliate con un ciclo di revisione e approvazione che riduca il DSO<sup>2</sup> e permetta quindi di ricevere il pagamento tempestivo dei servizi
- aumenta l'efficienza del rilascio di servizi automatizzando le notifiche, gli incarichi di responsabilità e i piani di lavoro.

L'obiettivo finale è quindi quello di aumentare il reddito dell'azienda riducendo i costi e fornendo servizi più efficienti a prezzi più bassi attraverso un'accurata gestione dei prezzi e degli ordini di lavoro.

Maximo for Service Provider permette di semplificare e automatizzare due campi di business di difficile gestione per un'azienda fornitrice di servizi:

- 1) dare una risposta adeguata ad ogni richiesta dei clienti;
- 2) fornire una fattura tempestiva e accurata a fine mese o al termine dei lavori.

<sup>1</sup> Il TCO (Total Cost of Ownership) rappresenta tutti i costi del ciclo di vita di un'apparecchiatura informatica IT, per l'acquisto, l'installazione, la gestione, la manutenzione e il suo smantellamento.

<sup>2</sup> Il DSO (Days Sales Outstanding) indica il numero medio di giorni necessario per ricevere il pagamento dopo aver effettuato una vendita. Si ottiene dalla seguente formula: (Crediti/Fatturato)\*numero di giorni.

In particolare per il secondo punto, Maximo fornisce delle accurate regole di tariffazione, tra le quali le seguenti:

- le attività di manutenzione sono tariffate in base al lavoratore, ai materiali, strumenti e servizi impiegati in esse;
- la gestione degli asset viene fatturata in base al numero di asset gestiti moltiplicato per un prezzo unitario per ciascuna classe di asset;
- un asset viene tassato anche in base al livello di performance che fornisce, monitorato utilizzando i KPI<sup>3</sup>;
- in caso di spostamento dell’asset o di cambiamento della sua configurazione, la tariffazione viene modificata;
- è possibile stabilire anche due quote di prezzo particolari: una “quota fissa”, utilizzata come prezzo, e una “quota da non superare”, utilizzata nel caso la somma delle attività svolte tenda a superare un valore prestabilito (appunto la quota che non deve essere superata).

### 3.2.2 Processo di fatturazione in Maximo

L’evolversi di un procedimento di fatturazione varia in base a quelli che sono i termini del contratto: si può avere una fatturazione mensile come una basata su ben determinate fasi di lavoro, si possono fatturare i servizi forniti come gli strumenti o i beni di consumo, si può considerare una tariffazione oraria come una basata sulle modalità di utilizzo di un asset, e vari altri fattori.

Analizziamo ora due dei possibili casi di ciclo di fatturazione per comprendere al meglio il suo funzionamento.

Parliamo inizialmente della classica offerta di servizi o materiali messi a disposizione interamente dall’azienda service provider (e non acquistati da aziende di terze parti) per il cliente. Già qui si presenta la differenziazione fra *costi* e *prezzi*. I costi saranno quelli che dovrà sostenere il service provider, nel qual caso si parla semplicemente di costi relativi ai materiali, ai lavoratori e utilizzo di eventuale strumentazione impiegata. I prezzi sono invece quelli imposti dal service provider al cliente (e definiti all’interno del contratto) sui servizi forniti.

In casi di offerte di questo tipo, il prezzo da pagare per il cliente sarà basato sul prezzo di listino definito per l’utilizzo di quei servizi, più un’eventuale scontistica.

Prezzo di listino      -      Sconto      =      Prezzo



[Figura 3.3: prezzo calcolato sul listino]

<sup>3</sup> I KPI (Key Performance Indicators) sono indici che monitorano l’andamento di un processo aziendale. Possono essere indicatori generali, di qualità, di costo, di servizio o di tempo



Il secondo tipo di offerta è basato su un service provider che non è il diretto fornitore di servizi, ma fa da tramite fra il cliente e il reale erogatore del servizio, ovvero un'azienda di terze parti. In casi come questo, vanno considerati due tipi di contratto:

- 1) il contratto fra service provider e l'azienda di terze parti per l'acquisto dei servizi;
- 2) l'offerta fatta dal service provider al cliente per la fornitura del servizio;

In altre parole, il service provider fa da intermediario fra il cliente e i fornitori, evitando così al primo di accollarsi spese e tempo di gestione, oppure per ridurre i costi: se il service provider ha infatti una convenzione con i fornitori per cui il costo dei servizi è inferiore a quello che i fornitori farebbero direttamente al cliente, si darebbe comunque la possibilità a quest'ultimo di risparmiare fornendogli il servizio al prezzo da convenzione (più un piccolo sovrappiù) invece del reale prezzo di listino.

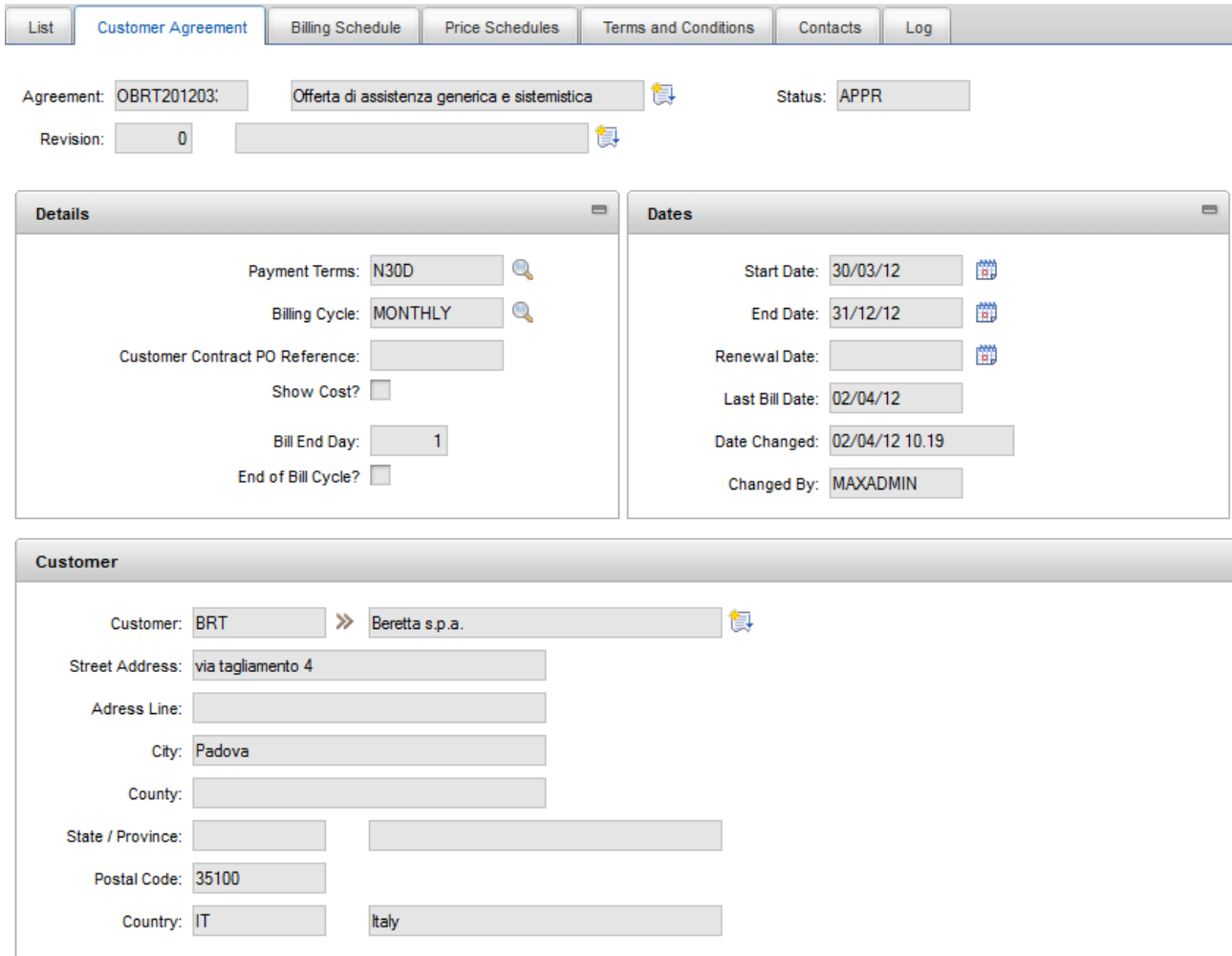


[Figura 3.4: prezzo calcolato sui costi sostenuti]

Al giungere del giorno di emissione della fattura, in essa viene imposto il *prezzo* calcolato sui servizi forniti secondo quanto stabilito dal contratto, considerando tutti gli ordini di lavoro fino a quel momento portati a termine.

### 3.2.3 Il Customer Agreement

L'applicazione di Maximo for Service Provider che si occupa di memorizzare al suo interno i termini dell'offerta per il cliente con i servizi e i relativi prezzi è l'applicazione *Customer Agreement*.



[Figura 3.5: applicazione “Customer Agreement”]

Come si vede dalla figura 3.5, sono presenti campi diversi, alcuni collegati fra loro da ben determinate relazioni, altri indipendenti. Sono presenti informazioni riguardanti i termini di pagamento, le date importanti e informazioni sull'ammontare fatturato o ancora da fatturare. Ognuno di questi oggetti è caratterizzato da un ben determinato cliente (il *customer*), manualmente creato all'interno di un'altra applicazione, con le sue caratteristiche essenziali.

E' interessante soffermarsi sull'area relativa ai piani di prezzo (*price schedule*). In essa vengono gestite tutte le regole di prezzatura dei vari casi di offerta. Si può avere un prezzo orario relativo al lavoro di una ben determinata persona, all'utilizzo di un asset, all'impiego di alcuni materiali o strumenti. In particolare per questi ultimi 3 casi, viene impostato un *listino prezzi* (in Maximo, il “*Price Book*”), che definisce appunto record di asset, materiali, item, ecc. con i relativi prezzi.

Si possono definire più “price schedule” all'interno dello stesso “agreement”, in modo che sia possibile applicare prezzi diversi per casi diversi appartenenti alla stessa offerta.

Ma a cosa va applicato un “price schedule”? Può essere applicato ad ordini di lavoro, attività singole, ordini di vendita, richieste di servizio, ecc.: per ognuno di essi può esserne impostato uno solo per volta, e ciò viene fatto in base alle “condizioni” stabilite sull’“agreement”. Infatti per ogni

“price schedule” si possono impostare un insieme di parametri (condizioni di applicabilità) che li differenziano l’uno dall’altro, cosicché sia possibile distinguerli ed eventualmente applicarli in maniera automatica. Applicare un “price schedule” permette che il relativo ordine di lavoro, richiesta di servizio o altro sia preso in considerazione al momento della fatturazione riguardante quel ben determinato “agreement”.

Una capacità essenziale di questa applicazione è la sua possibilità di creare fatture automaticamente. E’ possibile eseguire infatti un *billing schedule*, ovvero una schedulazione delle fatture che viene programmata al momento della compilazione dell’“agreement”: viene creata mensilmente in automatico in una data prefissata la fattura contenente già tutte le quote fatturate fino a quel momento. Questa funzionalità è di importanza vitale per aziende con molti clienti e molti contratti attivi, poiché permette di trovarsi ogni giorno la fattura già pronta da inviare, riducendo il TCO e aumentando la tempestività delle richieste di pagamento.

#### 3.2.4 I work order sul Service Provider

Un “work order”, è, come dice il significato letterale, un “ordine di lavoro” che parte in seguito ad una richiesta di attività da eseguire. L’applicazione “Work Order”, già presente su Maximo di base, è stata rivista nelle applicazioni del pacchetto Service Provider per renderla compatibile con il ciclo di fatturazione.

Con gli ordini di lavoro, è possibile tenere traccia del lavoro che è stato eseguito in passato e del futuro lavoro pianificato. Le informazioni contenute in un ordine di lavoro includono le attività eseguite, le ore di utilizzo delle risorse, i servizi, i materiali e le attrezzature richiesti per svolgere le attività, gli asset su cui si ha lavorato e le collocazioni in cui il lavoro è stato eseguito. La gestione di un ordine di lavoro spesso implica ordini di lavoro aggiuntivi, ad esempio, un “work order” derivato o la gestione di un progetto che coinvolge più ordini di lavoro. Inoltre, implica la registrazione dei costi, delle comunicazioni e di altri dati relativi al lavoro. Le singole attività eseguite in un work order sono rappresentate dai “task”, singoli record che raccolgono tutte le informazioni riguardanti un’attività svolta: tempo impiegato, ubicazione del luogo di svolgimento, responsabile dell’attività, ecc.. Ad ogni “task” vengono abbinate le relative risorse che verranno poi eventualmente consumate.

Proprio per quest’ultimo motivo, un “work order” può essere abbinato ad un cliente e ad uno dei “customer agreement” attivi per esso: in questo modo se le risorse elencate compaiono all’interno dell’offerta, ereditano automaticamente la prezzatura stabilita in essa.

Un “work order” può essere creato e compilato manualmente, oppure creato automaticamente in seguito ad un ticket (ovvero una richiesta di servizio o assistenza) o ad un “workflow<sup>4</sup>” predefinito.

#### 3.2.5 Piano di risposta

Se si vuole automatizzare il processo di creazione o di compilazione dei record, quest’applicazione rappresenta la soluzione perfetta. Un *Response Plan* è proprio un “piano di risposta” a ordini di lavoro, ticket di assistenza, ordini di vendita, ecc., in quanto riempie alcuni dei relativi campi in maniera automatica, esegue su di essi azioni predeterminate e crea e spedisce notifiche su ciò che avviene.

Un piano di risposta deve essere prima di tutto creato e compilato.

---

<sup>4</sup>Col termine “workflow” si intende una serie di percorsi predeterminati che può seguire un record all’interno del processo aziendale. Definisce inoltre le diverse azioni e notifiche da eseguire nei vari punti del processo. L’automazione dei processi di business ripetitivi e dei processi di gestione record fornisce maggiore efficienza e responsabilità all’interno dell’azienda.

• Response Plan: R1024    Response Plan 1    Attachments

Status: DRAFT    Ranking: 10

Status Date: 23/05/12 12:33    • Applies To: SR

Response Plan Administrator: MROSSI    Apply to Date: Report Date

• You can specify the date range for which this response plan is valid and a date by which a review is required for renewal. You can also select a shift and calendar. The plan is applied only to records that are reported for this calendar and shift.

**Dates**    **Applies To Calendar**

Start Date: 23/05/12    Calendar Organization:    Calendar:    Shift:    End Date:    Review Date:

• When this response plan is applied, any items you specify here are applied to the ticket, work order, or sales order. The job plan or ticket template is applied before any other items. If the job plan or ticket template can not be applied then specify whether to stop applying the response plan or allow it to continue to be applied without the job plan or ticket template.

**Response**

Apply Job Plan:    Stop Response Plan if Job Plan or Ticket Template can not be applied?

Job Plan Site/Org:    Assign Vendor: TEMPESTIVE    Tempestive - Santin & Associati

Apply Ticket Template: INSTCOND    Installazione condizionatore - giorno feriale    Assign Supervisor: DVISENTIN    Diego Visentin

Assign Owner: BBIANCHI    Bruno Bianchi    Assign Crew:    Apply Solution:    Assign Owner Group:    Assign Work Group:    Assign Lead:

[Figura 3.6: applicazione “Response Plan”]

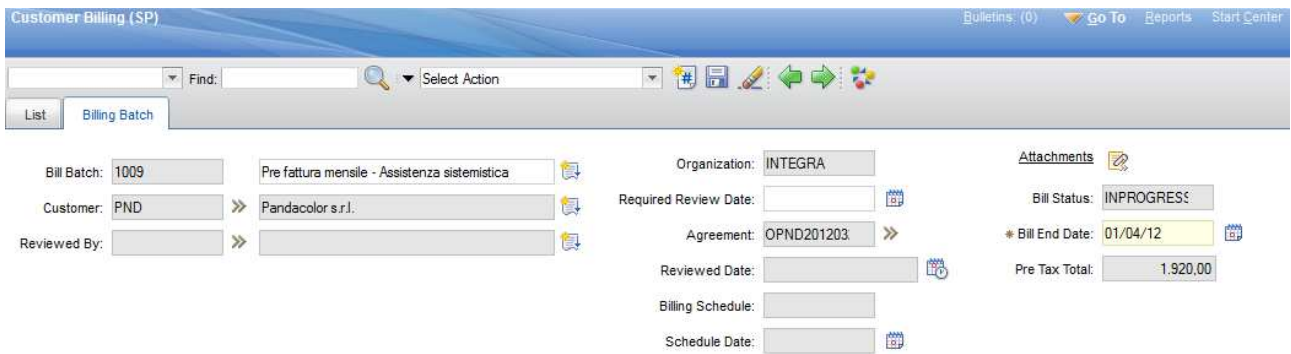
Dalla figura 3.6, notiamo che a seconda dell’oggetto al quale viene applicato il “response plan”, alcuni campi possono essere compilati e altri no. Per esempio, se, come accade in figura, viene applicato ad una “SR” (“service request”, ovvero Richiesta di Servizio), è possibile assegnargli un template predefinito (“ticket template”), un responsabile, un supervisore oppure la soluzione ad un ben determinato problema. Se invece lo applichiamo ad un “work order” allora sarà possibile scegliere il piano di lavoro relativo (“job plan”) oppure il gruppo di lavoro (“work group”).

In ogni caso, è anche possibile stabilire delle azioni da intraprendere, come anche stabilire delle condizioni di applicabilità del piano. Queste ultime vengono controllate nel momento in cui si decide di applicare un “response plan” ad un record: Maximo controlla tutti i campi del record e sceglie il piano di risposta che coincide con alcuni di essi.

Con l’utilizzo dei piani di risposta si ottengono quindi dei piccoli “workflow” capaci di semplificare il lavoro e l’organizzazione agli utenti.

### 3.2.6 *Bill Batch*: la fatturazione

Come e dove vengono raccolti tutti gli ordini di lavoro e di vendita, le richieste di servizio e tutto ciò che può essere fatturabile? Di questo si occupa l’applicazione *Customer Billing*, dalla quale nasce la fattura vera e propria. Essa raccoglie tutto ciò menzionato sopra in un “contenitore” chiamato appunto “bill batch”, nel quale tutti i relativi *costi* e *prezzi* vengono messi in evidenza, raggruppati per tipologia (costi e prezzi sostenuti per la manodopera, quelli per gli strumenti, quelli per i materiali, ecc.) e sommati in valori diversi.



[Figura 3.7: applicazione “Bill Batch”]

Come si nota dalle figure 3.7 e 3.8, il “bill batch” rappresenta una sorta di pre-fattura, nella quale *Pre Tax Total* è il totale che il service provider deve fatturare per il cliente e valori come *Billed Price* e *Agreed Price* invece rappresentano rispettivamente la parte del totale che è stata effettivamente fatturata e quella che ha riscontrato il consenso del cliente.

Maximo prevede che questa pre-fattura sia visibile anche al cliente (in una modalità leggermente differente) e che questi possa definire di essere d’accordo con alcune cifre e non con altre. In quest’ultimo caso il “bill batch” viene posto in uno stato di attesa, fin quando cliente e service provider non si saranno accordati su eventuali modifiche da apportare ai prezzi.

Line	Order	Class	Description	Total Cost	Quote Type	Included Price	Minimum Price	Agreed Price	Billed Price	Status
1	1039	WORKORDER	Richiesta configurazione server	0,00		<input type="checkbox"/>	<input type="checkbox"/>	1.600,00	1.600,00	WAPPR
2	1042	WORKORDER	Rete bloccata	0,00		<input type="checkbox"/>	<input type="checkbox"/>	320,00	320,00	WAPPR

Task	Labor	Name	Approved	Regular Hours	Premium Pay Hours	Line Cost	Line Price	Bill Price
	NCHA	Nicola Chahinian	<input checked="" type="checkbox"/>	8:00		0,00	640,00	640,00
	NCHA	Nicola Chahinian	<input checked="" type="checkbox"/>	4:00		0,00	320,00	320,00
	NCHA	Nicola Chahinian	<input checked="" type="checkbox"/>	8:00		0,00	640,00	640,00

[Figura 3.8: “bill line” (evidenziate in rosso) e consuntivazioni risorse]

I record evidenziati in figura 3.8 rappresentano le “bill line”: ogni record riporta l’ordine di lavoro, il ticket di assistenza o l’ordine di vendita all’interno del quale sono state fatte delle consuntivazioni riguardanti il contratto e il cliente selezionati all’interno del “bill batch”.

Per ogni contratto può esistere un solo “bill batch” attivo.

Il ciclo di una pre-fattura in Maximo è rappresentato da fasi diverse, caratterizzate ognuna da un ben determinato *stato*:

- 1) **IN PROGRESS**: il bill batch è aperto e ogni qualvolta che si consuntiva qualcosa, i relativi prezzo e costo vengono inseriti al suo interno automaticamente;
- 2) **PREBILL**: il bill batch è stato reso disponibile alla consultazione da parte del cliente, che può decidere se approvarlo oppure no;
- 3) **REVIEWED**: il cliente ha preso visione della pre-fattura ed è d’accordo sui prezzi applicati: impostando questo stato si definisce pronto per l’emissione della fattura vera e propria;
- 4) **BILLED**: un bill batch viene posto in questo stato quando la fattura viene emessa e pagata. A questo punto il record relativo diventa esclusivamente un record di storico;

5) **CANCELLED**: un bill batch che è stato creato per errore o nel momento sbagliato;

La stampa della fattura finale è affidata alla reportistica di Maximo: una fattura è un report che prende i dati che gli servono dal database in base al contratto per il quale essa viene redatta. Su come viene creato un report ne parleremo nel paragrafo successivo.

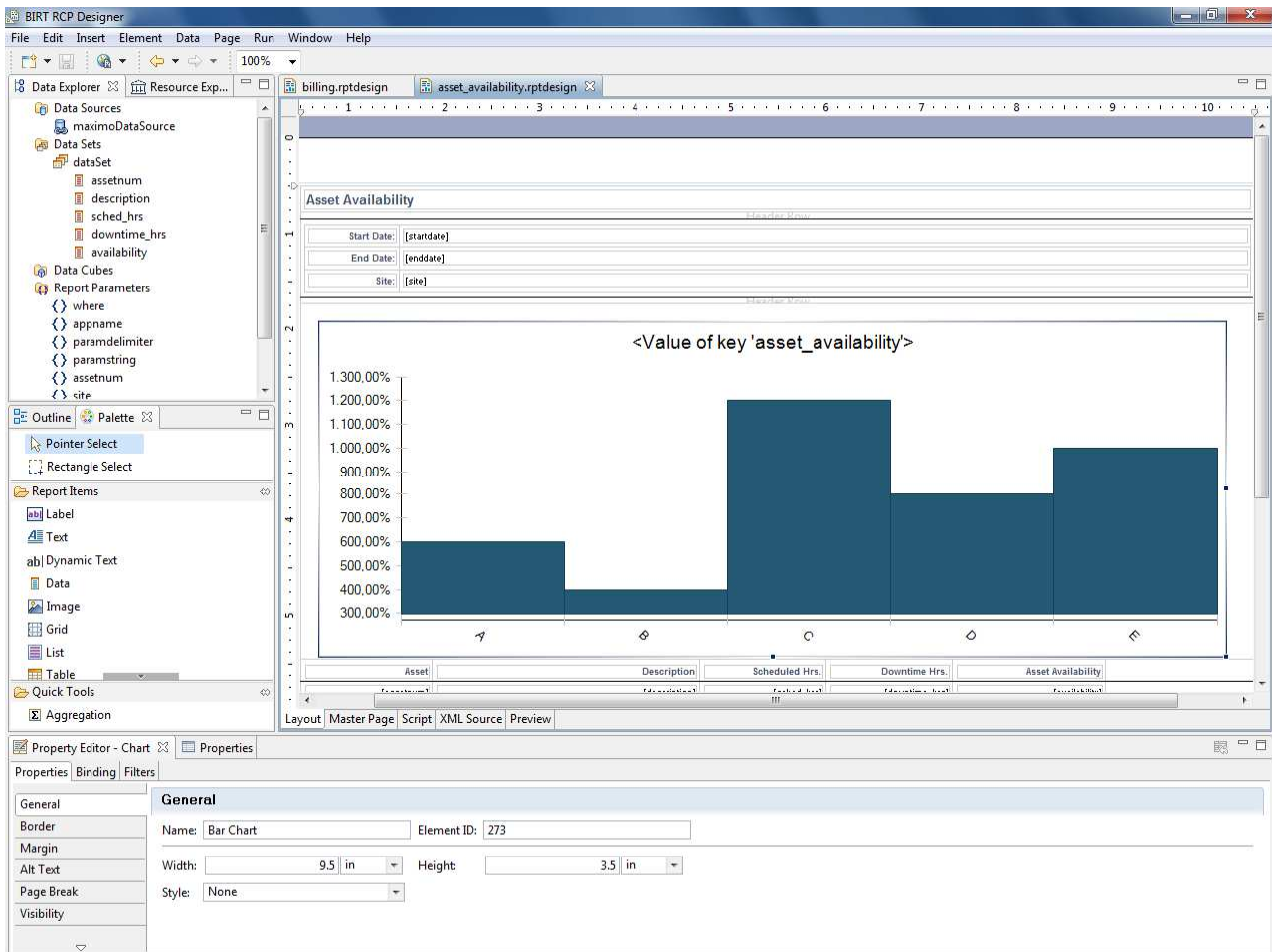
### 3.2.7 Altre applicazioni

Oltre a quelle già menzionate, vi sono molte altre applicazioni che servono per arricchire di funzionalità Maximo for Service Provider, di cui alcune sono elencate qua di seguito:

- *Service Request*: sono i ticket di assistenza che si aprono per un cliente che ha bisogno di un intervento. Queste richieste di servizio contengono tutti i dati relativi al problema del cliente o al lavoro per il quale ha chiesto assistenza e possono avviare in automatico un work order, un workflow, un'altra richiesta di servizio, ecc.;
- *Price Book*: è il listino prezzi, che contiene il prezzo relativo al servizio, materiale o strumento che si intende fornire o utilizzare per il cliente. E' possibile inserire più elementi prezzati all'interno dello stesso "price book", e poi abbinarlo ad uno o più "agreement".
- *Customer*: è l'applicazione che crea e memorizza il cliente e i relativi dati. Ogni cliente può essere inserito in una o più "location";
- *Location*: rappresenta l'ubicazione di dove si svolge fisicamente il lavoro: molte volte corrisponde con le sedi dei clienti, ma può anche rappresentare un'ubicazione di altro genere nel quale svolgere il lavoro per un cliente

### 3.3 La creazione dei report: *BIRT Report Designer*

Il *BIRT* (Business Intelligence and Reporting Tools) *Report Designer* è un software “open-source” fornito per la reportistica e per le funzionalità di “business intelligence” di applicazioni Web, specialmente per quelle basate su Java e JavaEE<sup>5</sup>. Esso ha 2 principali componenti: un generatore di report dotato di interfaccia grafica e basato sull’IDE<sup>6</sup> Eclipse e una componente “runtime” che può essere messa in funzione all’interno di un qualunque ambiente Java.



[Figura 3.9: BIRT Report Designer]

L’obiettivo di utilizzo di questo software è affrontare una vasta gamma di esigenze di reportistica all’interno di una tipica applicazione, da quella operativa o di impresa all’elaborazione analitica multi-dimensionale. I file contenenti i report sono sostanzialmente dei file XML che incorporano il “design” del report e l’URL della sorgente designata per ricavare le informazioni, e hanno estensione “.rptdesign”.

Il “business intelligence” e lo spazio della reportistica si focalizzano su strumenti e capacità che estraggono i dati da un “data source”, li elaborano e presentano le informazioni processate all’utente finale.

Il processo di creazione di un report prevede 4 step fondamentali:

<sup>5</sup>Java EE (dall’inglese “Java Enterprise Edition”) è la versione *enterprise* della piattaforma Java. È costituita da un insieme di specifiche che estendono le funzionalità di base della piattaforma Java

<sup>6</sup>Un “integrated development environment” (IDE), in italiano *ambiente di sviluppo integrato*, è un software che aiuta i programmatori nello sviluppo del codice. Normalmente consiste in un editor di codice sorgente, un compilatore e/o un interprete, un tool di building automatico, e (solitamente) un debugger.

- 1) la definizione del *Data Source*, ovvero della sorgente di dati dalla quale attingere le informazioni da elaborare. Si tratta di definire i driver, l'URL e le credenziali di accesso, solitamente, ad un database;
- 2) la definizione del *Data Set*, ovvero dell'insieme di tabelle e di campi che verranno utilizzati, attraverso un insieme di query;
- 3) la definizione del layout e dei campi del report;
- 4) l'aggiunta di parametri aggiuntivi provenienti da un secondo data set o da definire manualmente al momento dell'apertura del report.

Sostanzialmente, quindi, i report, dopo aver estratto i dati dalle relative sorgenti, eseguono elaborazioni e calcoli su di essi, per rispondere alle esigenze di mercato e presentare i risultati in una forma strutturata e comoda da utilizzare per gli utenti aziendali.

Tra le varie caratteristiche di un report, vi sono le seguenti:

- capacità di ordinare, raggruppare e aggregare i dati con e senza subtotali;
- rilascio delle informazioni sotto forma di pagina web, PDF, documenti stampati, file excel, ecc., oppure con una loro combinazione;
- layout precisi e strutturati, come, per esempio, estratti di conti bancari, bollette, fatture, ecc.;
- tabelle di contenuti;

La creazione di report viene eseguita principalmente da 3 classi di sviluppatori:

- 1) *sviluppatori applicativi*: questi sono sviluppatori che creano applicazioni le quali includono la necessità di recuperare i dati e presentarli sotto forma di rapporti;
- 2) *sviluppatori di report*: non sono programmatori (e quindi non si occupano della scrittura del codice in linguaggio di programmazione), ma utilizzano qualche "tool" grafico per creare i report che poi verranno eventualmente integrati in qualche applicazione;
- 3) *utenti aziendali*: utilizzano i template dei report già creati, modificandone eventualmente il layout a seconda delle proprie esigenze.

#### **3.4 Tempestive Activity: l'organizzazione delle attività di un'azienda**

Tempestive Activity è un'applicazione pensata per le organizzazioni che hanno l'esigenza di verificare il tempo che dedicano allo sviluppo delle attività o di progetti. In essa è possibile definire l'organizzazione delle proprie attività, gestire i progetti, i task, i clienti e raccogliere le attività quotidiane svolte dal proprio personale.

Il servizio permette anche di inserire le varie fasi/sotto attività del progetto, nonché di descrivere dettagliatamente il lavoro che si sta svolgendo.



The screenshot shows the 'Activity' application interface. At the top, there is a navigation bar with 'LE MIE ATTIVITÀ', 'CATEGORIE', 'STATISTICHE', 'STORICO', 'AUTO', 'AMMINISTRATORE', and 'AIUTO'. The main content area is titled 'LE MIE ATTIVITÀ' and contains a table of activities. To the right, there is a 'RIMBORSI' section with a small table.

DATA	CLIENTI	PROGETTI	TASK	PERSONA	DESCRIZIONE	ORE
13/04/2012	Installo s.p.a.	Progetto a corpo - Installazione Maximo for Service Provider	Fase 1 - Installazione Maximo Asset management + Service Provider	Chahinian Nicola	Installazione Middleware	4,00
13/04/2012	Installo s.p.a.	Progetto a corpo - Installazione Maximo for Service Provider	Fase 1 - Installazione Maximo Asset management + Service Provider	Chahinian Nicola	Installazione Service Provider + controllo funzionamento	4,00
13/04/2012	Installo s.p.a.	Progetto a corpo - Installazione Maximo for Service Provider	Fase 1 - Installazione Maximo Asset management + Service Provider	Chahinian Nicola	Installazione Maximon Asset Management	8,00
13/04/2012	Installo s.p.a.	Progetto a corpo - Installazione Maximo for Service Provider	Fase 1 - Installazione Maximo Asset management + Service Provider	Chahinian Nicola	Riunione iniziale + controllo PC aziendali	8,00
06/04/2012	Sistemo s.p.a.	Offerta di Assistenza	Assistenza generica	Chahinian Nicola	Controllo sicurezza PC aziendali	8,00
06/04/2012	Sistemo s.p.a.	Offerta di Assistenza	Assistenza sistemistica	Chahinian Nicola	Ripristino funzionamento Application Server	8,00
06/04/2012	Sistemo s.p.a.	Offerta di Assistenza	Assistenza sistemistica	Chahinian Nicola	Ripristino funzionamento del Server	8,00
02/04/2012	Beretta s.p.a.	Progetto vendita e installazione Maximo Asset Management	Fase 1 - Installazione Maximo	Chahinian Nicola	Installazione Middleware e Maximo Asset Management	8,00
02/04/2012	Beretta s.p.a.	Offerta di assistenza generica e sistemistica	Assistenza sistemistica	Chahinian Nicola	Risoluzione problemi application server	4,00
02/04/2012	Milka s.p.a.	Offerta di vendita e installazione Maximo Asset Management	Fase 1 - Installazione Maximo	Chahinian Nicola	Installazione totale Maximo Asset Management	8,00
02/04/2012	Parmalat s.p.a.	Offerta di Assistenza	Assistenza sistemistica	Chahinian Nicola	Urlare più forte	8,00
30/03/2012	Boriello s.r.l.	Offerta per Applicativi Magazzino	Fase 2 - Istruzione del personale	Chahinian Nicola	Giornata di istruzione del personale	8,00
30/03/2012	Digrim s.p.a.	Offerta di consulenza generica in loco o da remoto	Assistenza generica	Chahinian Nicola	Backup database e consecutivo refresh	8,00

IDPERSONA	ANNO	MESE	
Chahinian Nicola	2012	4	+
Bruno	2012	3	+
Carlos	2012	3	+
Chahinian Nicola	2012	3	+

©2011,2012 Tempestive S.r.l. - Powered by Windows Azure

[Figura 3.10: Tempestive Activity nello storico delle attività svolte]

Come si nota dalla figura 3.10, Activity memorizza i dati inseriti sotto forma di record, appoggiandosi a *Microsoft SQL Server Database*. E' un'applicazione WEB costruita sulla base di Microsoft.NET (e utilizza tecniche AJAX per essere veloce e dinamica) nella quale ogni campo dell'interfaccia grafica è riferito ad un ben determinato campo di una ben determinata tabella(o vista) nel database. Attraverso query di inserimento, ricerca e cancellazione pre-impostate, permette ad un utente base di poter gestire le proprie attività in maniera user-friendly, senza l'utilizzo di codice Sql.

Scendendo di più nel particolare, si distinguono principalmente due tipi di privilegi d'accesso: un accesso amministrativo e un accesso per ogni dipendente dell'azienda che utilizza il software.

L'utente amministratore ha i privilegi di censimento di clienti, relativi progetti, task (attività singole o fasi) di ogni progetto, può eseguire i report riguardanti le attività da esso svolte durante la giornata lavorativa e avere accesso ai report degli altri dipendenti dell'azienda.

L'utente normale, quindi il dipendente non amministratore, può invece avere solamente accesso allo storico dei propri report e alla relativa schermata di creazione.

Ma da cosa è caratterizzato un report?

Essendo il resoconto di un'attività svolta o di come sono state impiegate le ore lavorative di una giornata, un report è caratterizzato da tutte quelle informazioni con le quali si può definire con buona precisione il lavoro prestato: il numero di ore dedicate all'attività, la descrizione dell'attività stessa, il cliente e i relativi progetto e task di riferimento. Nonché è possibile censire anche informazioni riguardanti un'eventuale trasferta e il costo di quest'ultima (costo dei pedaggi, chilometri percorsi, spese di vitto e pernottamento, ecc.).

Una volta inserite e salvate, queste informazioni vengono memorizzate in uno o più record all'interno del database e consultate dall'utente amministratore quando ne abbia bisogno (per

valutare la fattura per il cliente, per definire delle statistiche sul tempo impiegato per le attività, per controllare l'efficienza dei dipendenti, ecc.).

Interessante è sicuramente l'organizzazione delle tabelle relative a quest'applicazione, di cui parleremo prossimamente, poiché è stato oggetto di studio e di alcune modifiche da parte mia per renderlo compatibile con il mio progetto finale di integrazione.

## Capitolo 4

# Censimento delle offerte

La prima parte dello stage consisteva nel venire a contatto con le applicazioni di Maximo Asset Management e Maximo for Service Provider, comprenderne la logica ed il funzionamento, e simulare delle offerte attive per degli ipotetici clienti.

Maximo è stato costruito per rispondere alla maggior parte delle possibili esigenze di un'azienda: in base a ciò di cui si ha bisogno, si possono utilizzare alcune applicazioni e non altre, si possono personalizzare le applicazioni stesse, si possono usare alcuni campi del database e altri no (il “data set” di Maximo prevede per ogni tabella alcuni campi inutilizzati che possono essere usati per arricchire l'insieme delle informazioni necessarie). In più, ogni applicazione contiene dei campi da compilare obbligatoriamente (l'obbligatorietà dipende dai vincoli stabiliti nelle tabelle del database oppure da alcune regole del software) e dei campi facoltativi (alcuni di questi campi facoltativi possono diventare obbligatori nel momento in cui si scelgono alcune opzioni oppure particolari percorsi logici da seguire).

Vediamo ora alcuni dei casi di offerte prese in considerazione, per comprendere meglio nella pratica il loro ciclo di vita all'interno di IBM Maximo. Nel corso della descrizione, vedremo più da vicino il funzionamento e la logica di alcune delle applicazioni del software.

### 4.1 Esempio 1: interventi routinari o di emergenza

#### 4.1.1 Scenario

Lo scenario seguente rappresenta un contratto fra due aziende, la *RedBooksInc* e la *RedBookCus-A*, nel quale la prima fornisce servizi di gestione degli edifici alla seconda. Il contratto prevede interventi di manutenzione (o riparazione) di due tipologie: di emergenza e routinaria (non di emergenza). Nel primo caso, i materiali utilizzati per l'intervento avranno un prezzo di listino prefissato, mentre nel secondo caso verrà applicata una scontistica del 15% rispetto al prezzo di listino. Il contratto prevede anche degli interventi di manutenzione periodica degli ascensori, che verranno fatturati al 40% in più rispetto al prezzo base.

#### 4.1.2 Implementazione

Dopo aver censito il proprio cliente (*RedBookCus-A*), la *RedBooksInc* provvede in Maximo alla creazione del *Customer Agreement*, ovvero del contratto con il cliente.

[Figura 4.1: compilazione delle prime informazioni sull'offerta]

Inizialmente si seleziona il cliente e vengono inseriti quelli che sono i termini di pagamento (“N30D”, ovvero pagamento a 30 giorni dall’emissione della fattura), il ciclo di fatturazione (mensile) e le date di inizio, termine e rinnovo del contratto.

Dopodiché si procede alla creazione di due *price schedule*, quindi di due pianificazioni dei prezzi relative a 2 tipologie di interventi, quella routinaria e quella d’emergenza.

Partendo dal “Non Emergency Price Schedule”, definiamo che questo venga “applicato ai WORKORDER con un ranking di 20”: questo vuol dire che il presente contratto con queste tariffazioni potrà essere affiancato agli ordini di lavoro, prezzando i relativi materiali e servizi nella maniera stabilita.

[Figura 4.2: Definizione di un piano di prezzo (“Price Schedule”)]

Il significato del campo “ranking” lo vedremo in seguito.

Si passa ora alla definizione dei prezzi. Ogni agreement che definisce materiali, servizi o strumenti che rientrano nell’offerta, deve essere abbinato ad uno o più *price book*, ovvero il reale listino prezzi.

[Figura 4.3: listino prezzi per il termostato ambiente (“Price Book”)]

Viene quindi creato un price book per i materiali e al suo interno ci inseriamo un termostato per ambienti del costo, al pubblico, di 70 euro. Una volta abbinato questo listino al nostro precedente “price schedule”, possiamo definire la scontistica del 15%, facendo un check sull’opzione “discount from list price”: in questo modo si definisce che l’utilizzo di quel materiale in un intervento di manutenzione routinaria costerà al cliente il 15% in meno rispetto al normale prezzo di listino.

[Figura 4.4: imposizione scontistica del 15% sul listino prezzi]

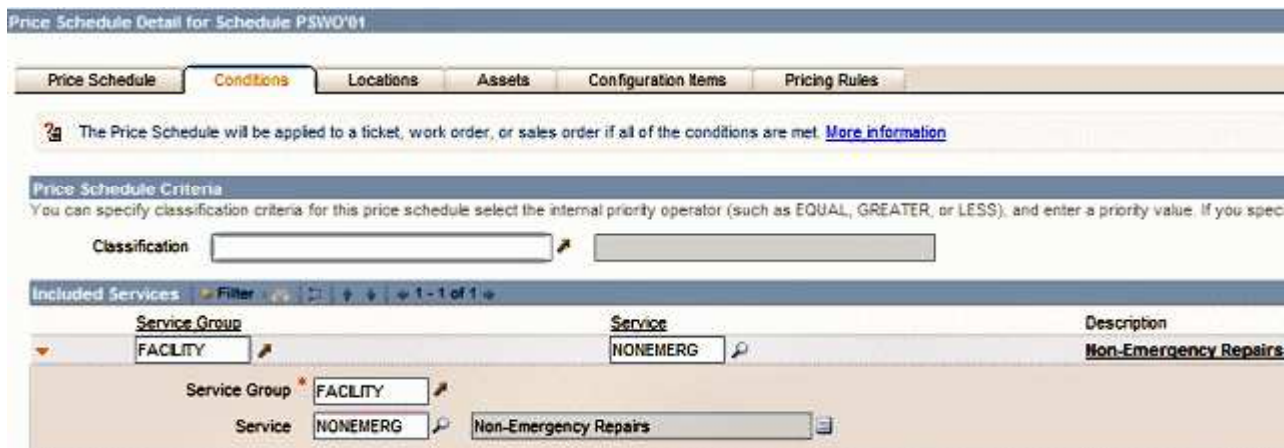
Per il servizio di manutenzione degli ascensori, invece, si prevede per il cliente un sovrappiù del 40% su quanto costa il servizio all’azienda Service Provider. Considerando che la RedBooksInc. paghi 1.000 euro il tecnico addetto a tale intervento, farà pagare a RedBookCus-A una somma chiaramente maggiorata, ovvero 1.400 euro. Impostando quindi sull’agreement l’opzione “markup from item cost”, insieme con il valore “40.00” si otterrà negli ordini di lavoro il risultato suddetto, senza l’applicazione di un listino prezzi (poiché il prezzo viene calcolato sul costo d’acquisto, e non su un prezzo di listino fissato).

L’ultima parte della compilazione di un “price schedule” riguarda le condizioni di applicazione. Quando viene richiesto a Maximo di applicare un piano di prezzo a un ordine di lavoro, ticket, ordine di vendita, ecc., esso esegue diversi confronti basati su una ben determinata logica. In particolare, esegue in sequenza 3 controlli di compatibilità:

1. compatibilità di cliente;
2. compatibilità di condizioni;
3. maggiore priorità.

Per prima cosa l’oggetto (work order, service request, ecc.) a cui deve essere applicato il “price schedule” deve essere associato allo stesso cliente per il quale è stato redatto l’agreement. Se ve ne è più di uno, Maximo controlla che siano applicate le condizioni di applicabilità stabilite nel “price schedule”. Infatti nella sezione *Conditions* del nostro Customer Agreement, possiamo definire particolari classificazioni oppure dei gruppi di appartenenza: se questi gruppi sono gli stessi che vengono impostati nell’oggetto, allora il piano può essere applicato. Nel nostro caso inseriamo il

gruppo di servizi per impianti (“Facility”) e come sotto gruppo il servizio di riparazioni non di emergenza.



[Figura 4.5: condizione di applicabilità basata sul tipo di servizio erogato]

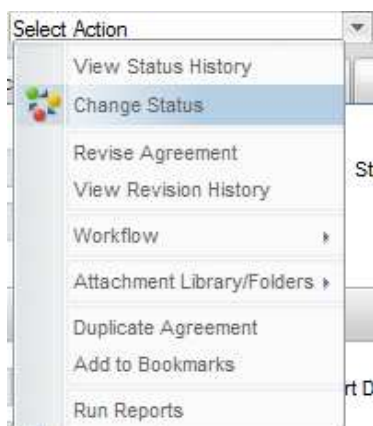
Se ancora vi sono più piani di prezzo con designate le stesse condizioni di applicabilità, allora Maximo, come ultimo tentativo, va a controllare il *ranking* del “price schedule”. Questo rappresenta una sorta di indicatore di “priorità inversa”, nel senso che più basso è e più prioritario diventa: infatti al piano routinario diamo ranking 20, mentre in quello di emergenza inseriamo un 10. Nel momento dell’applicazione, Maximo sceglie quindi il “price schedule” con ranking più basso.

Nel caso che vi siano ancora più “price schedule” con lo stesso “ranking”, Maximo getta la spugna e ne applica uno a caso (solitamente il primo che trova in lista).

Arrivati a questo punto non ci resta che creare un secondo piano per gli interventi di emergenza, simile al precedente ma con le seguenti caratteristiche:

- Descrizione: Price Schedule – Emergency;
- Applicato a WORKORDER;
- Ranking = 10;
- Sconto materiali = 0;
- Selezione dell’opzione “Discount from List Price”
- Condizioni di applicabilità:
  - gruppo: “Facility”
  - sottogruppo: “Emergency Repairs”

Non ci resta che approvare l’offerta appena creata cambiando di stato l’agreement: dal menù “Select Action” si ricerca l’opzione “Change Status” e si imposta lo stato in APPR (Approved).



[Figura 4.6: menù “Select Action” – Cambio di stato]

Terminata la creazione dell'offerta si passa alla creazione di un tipico ordine di lavoro per gli interventi routinari e uno di emergenza.

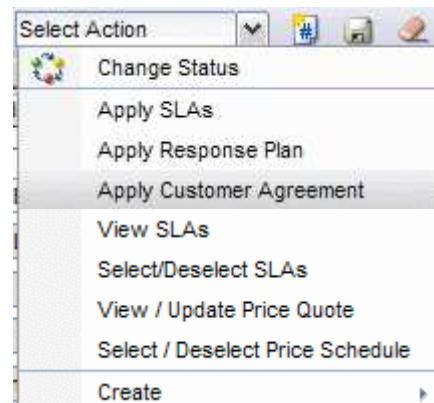
Concentrandoci sul primo, ci servono le seguenti informazioni:

- Descrizione: "Work Order for non-emergency repairs"
- Locazione: RedbookCus-A Location (è la locazione fisica di dove andrà eseguito il lavoro; deve essere creata precedentemente nell'apposita applicazione *Location*);
- Gruppo di applicabilità:
  - o gruppo: "Facility"
  - o sottogruppo: "Non-Emergency Repairs"
- Cliente: RedbookCus-A

In questo modo abbiamo tutte le carte in regola per l'applicazione del Customer Agreement: dal menù "Select Action" scegliamo infatti l'opzione "ApplyCustomer Agreement" e automaticamente verrà affiancata al "work order" l'offerta creata precedentemente con il "price schedule" giusto, ovvero quello di *non-emergenza*.

Ora vediamo di compilare questo ordine, pianificando i materiali e i servizi di cui farà uso. Un ordine di lavoro può infatti rappresentare una singola attività oppure un insieme di attività che fanno uso dello stesso insieme di risorse.

Nella sezione delle pianificazioni, inseriamo quindi un termostato per ambienti.



Work Order: RB'001  
 Work Order for Non-Emergency repairs  
 Parent WO: [ ]

Children of Work Order RB'001

Tasks for Work Order RB'001

Materials

Task	Item	Description
[ ]	29331	Building Thermostat

Task: [ ]  
 Item: 29331 Building Thermostat  
 Line Type: Item  
 Quantity: 1.00  
 Order Unit: [ ]  
 Unit Cost: 50.00  
 Line Cost: 50.00  
 List Sales Price: 70.00  
 Line Price: 59.50

[Figura 4.7: pianificazione materiali di consumo su di un ordine di lavoro]

Notiamo subito, dalla figura 4.7, la solita distinzione fra *costi* e *prezzi*. Il campo “Line Cost” è un campo del Maximo di base e rappresenta il costo totale del materiale per l’azienda service provider, ed è dato dal prodotto fra il costo unitario e il numero di articoli:

$$\text{Line Cost} = \text{Quantity} * \text{Unit Cost}$$

Il campo “Line Price”, invece, è tipico di Maximo for Service Provider e agisce parallelamente al contratto applicato: in base alle regole di prezzo stabilite nel “Customer Agreement”, definisce quale prezzo applicare al cliente, se vi è un qualche tipo di scontistica o “markup”, ecc.. Il tutto viene inserito automaticamente da Maximo: nel nostro caso viene impostato 70 euro come prezzo unitario di listino (“List Sales Price”) e poi, essendo che era stato definito uno sconto del 15% proprio sul prezzo di listino, Maximo definirà un prezzo finale (“Line Price”) di 59,50 euro, utilizzando la seguente formula:

$$(1) \text{ Line Price} = \text{Quantity} * (\text{List Sales Price} + [\text{discount}(\%) \text{ or } \text{Markup}(\%)])$$

ovvero

$$\text{Line Price} = 1 * (70 - (70 * 15\%)) = 59,50$$

La stessa cosa viene effettuata per la pianificazione del servizio di manutenzione degli ascensori. In questo caso, però, viene applicato un sovrappiù sul costo del servizio (se ricordiamo bene,

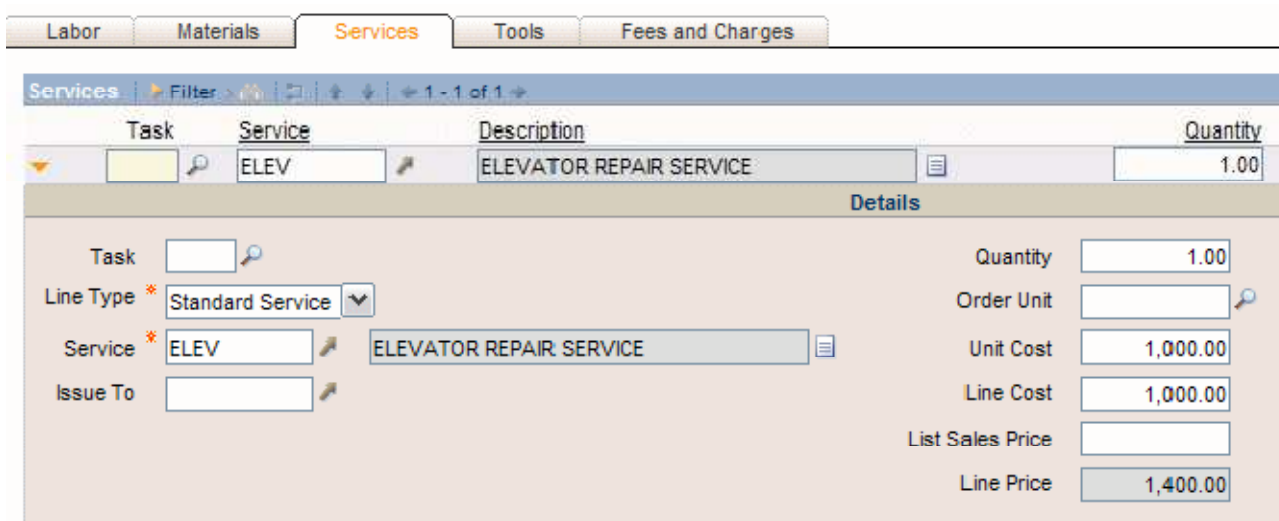


nell’offerta era stata impostata l’opzione “markup from item cost”, definendo di aggiungere il 40% in più al costo del servizio). La formula (1) si trasforma quindi nella seguente formula (2):

$$(2) \text{ Line Price} = \text{Quantity} * (\text{Unit Cost} + [\text{discount}(\%) \text{ or } \text{Markup}(\%)])$$

ovvero

$$\text{Line Price} = 1 * (1.000 + (1000*40\%)) = 1.400$$



Task	Service	Description	Quantity
	ELEV	ELEVATOR REPAIR SERVICE	1.00

Details	
Task	Quantity 1.00
Line Type * Standard Service	Order Unit
Service * ELEV ELEVATOR REPAIR SERVICE	Unit Cost 1,000.00
Issue To	Line Cost 1,000.00
	List Sales Price
	Line Price 1,400.00

[Figura 4.8: Pianificazione servizi da erogare su di un ordine di lavoro]

A questo punto lo stato del Work Order deve essere approvato con il solito cambio di stato e, nel momento in cui i lavori cominceranno, verrà posto nello stato “IN PROGRESS”, definendo così che l’ordine è in via di svolgimento.

Il secondo ordine di lavoro sarà invece riferito agli interventi di emergenza. In esso inseriremo i seguenti valori

- Descrizione: “Work Order for emergency repairs”
- Locazione: RedbookCus-A Location (la stessa del precedente work order);
- Gruppo di applicabilità:
  - gruppo: “Facility”
  - sottogruppo: “Emergency Repairs”
- Cliente: RedbookCus-A

Applicando il “Customer Agreement”, il piano di prezzo che verrà applicato sarà quello delle emergenze (poiché Maximo trova la corrispondenza dei gruppi di applicabilità).

La pianificazione dei lavori conterrà semplicemente il termostato d’ambiente come materiale: in questo caso però, essendo previsto sconto nullo sul prezzo di listino, quello che verrà applicato sarà proprio il prezzo di listino:

$$(3) \text{ Line Price} = \text{Quantity} * \text{ListSalesPrice}$$

ovvero

$$\text{Line Price} = 1 * 70 = 70$$

Il costo sarà invece lo stesso del precedente, quindi 50 euro a pezzo.

Infine approviamo e poniamo “IN PROGRESS” il nostro “work order”;

### 4.1.3 Consuntivazione e fatturazione

Ogni volta che viene eseguita un’attività, questa viene inserita all’interno dell’ordine di lavoro e vengono consuntivate tutte le risorse utilizzate e i servizi forniti. Consuntivare, in un “work order” di Maximo, significa aggiungere le risorse nella sezione *Actuals*, caricandole da quelle pianificate o aggiungendone anche altre di nuove. L’ammontare da fatturare sarà chiaramente basato sul contratto applicato e sugli eventuali listini prezzi ad esso annessi. Nel nostro caso, consideriamo che per ogni attività del “work order” di non-emergenza consuntiviamo l’utilizzo del termostato ambiente e il servizio di manutenzione degli ascensori, per quello di emergenza solamente il termostato.

The screenshot displays the Maximo software interface for a work order. At the top, a navigation bar includes tabs for 'List', 'Work Order', 'Plans', 'Related Records', 'Actuals' (highlighted with a red circle), 'Safety Plan', 'Log', and 'Failure Reporting'. Below this, the 'Work Order' field is set to 'RB'001' and the description is 'Work Order for Non-Emergency repairs'. The 'Site' is 'BEDFORD'. The 'Actuals' section shows a table with columns for 'Sequence', 'Task', and 'Summary', but it is currently empty, displaying the message '...No rows to display...'. Below the table, there are tabs for 'Labor', 'Materials', 'Services' (selected), 'Tools', and 'Fees and Charges'. The 'Services' section shows a table with columns for 'Task', 'Service', and 'Description'. The selected service is 'ELEV' with the description 'ELEVATOR REPAIR SERVICE'. A 'Details' section below this table shows the following information:

Task		Quantity	1.00
Line Type	STDSERVICE	Unit Cost	1,000.00
Service	ELEV ELEVATOR REPAIR SERVICE	Order Unit	
		Line Cost	1,000.00
		List Sales Price	0.00
		Line Price	1,400.00
		Transaction Date	6/20/10 4:31 PM

[Figura 4.9: consuntivazione del servizio erogato su di un ordine di lavoro]

A questo punto non ci resta che fatturare il tutto. Utilizzeremo quindi lo strumento che ci fornisce Maximo for Service Provider per raccogliere tutte le attività e le risorse utilizzate per un ben determinato contratto, ovvero il *Bill Batch*. Esso ci permette di raggruppare tutte le risorse impiegate per le varie attività svolte per un cliente e suddividerle per tipo. Mantiene però la distinzione fra i vari ordini di lavoro, in modo da evidenziare le ore impiegate per l’esecuzione di ognuno. Vedremo meglio l’utilizzo del Bill Batch nel prossimo esempio.

## 4.2 Esempio 2: il subappalto dei servizi

### 4.2.1 Scenario

Nel seguente esempio, RedbookInc. è di nuovo la nostra azienda Service Provider. In questo caso ha un contratto con l'azienda CompanyA per la gestione dei loro edifici, che comprende anche la manutenzione degli ascensori. Questa manutenzione non è eseguita direttamente da RedbookInc., ma viene subappaltata all'azienda CompanyE, che esegue fisicamente l'intervento. Il contratto fra il Service Provider e il cliente CompanyA prevede un servizio di manutenzione per ascensori a un prezzo maggiorato rispetto al costo che RedBookInc. deve sostenere per l'acquisto del servizio da CompanyE: il 30% in più per interventi routinari e il 100% in più per interventi di emergenza.

In questo esempio vedremo anche come viene aperta una richiesta di servizio (*service request*) e come il ciclo degli ordini di lavoro venga automatizzato con l'utilizzo di un *Response Plan*.

### 4.2.2 Implementazione

Una volta creato il cliente CompanyA e l'azienda subappaltatrice CompanyE (che non sarà censita nell'applicazione dei clienti (*Customer*), ma nell'applicazione *Company* che registra tutte le aziende che riforniscono l'azienda Service Provider), si procede alla creazione del "Customer Agreement". Senza scendere nel dettaglio sulla compilazione dei campi di questa applicazione (visionata nell'esempio precedente), vediamo quali sono i valori che ci interessano:

- Termini di pagamento: N30D (entro 30 giorni)
- Ciclo di fatturazione: mensile
- Giorno di fatturazione: il 1° di ogni mese (ciò vuol dire che la fattura mensile conterrà le attività che vanno dal secondo giorno del mese fino al primo giorno del mese successivo compreso).

Anche questa volta creiamo due "price schedule", però la loro applicazione dipenderà dal livello di priorità dell'ordine di lavoro. Il "ranking" di entrambi i piani di prezzo viene impostato a 4: la differenza è data dalla condizione di applicabilità su un intervento di emergenza.

Between Groups	Field Name	Condition	Value
	Reported Priority	EQUALS	2

[Figura 4.10: condizione di applicabilità basata sull'uguaglianza di un campo]

Viene infatti impostata la condizione seguente: *se la “ReportedPriority” del “work order” è uguale a 2, allora applica questo “price schedule”*. In termini di campi da compilare nell’applicazione, questa frase è sintetizzata nella figura 4.10.

La prezzatura viene fatta per entrambi i piani nella sezione dei servizi, impostando per entrambi un “markup” rispetto al costo del servizio:

- 30% per il piano routinario
- 100% per il piano di emergenza.

Fatto questo, possiamo approvare l’agreement e passare alla creazione dei “Piani di Risposta” (*Response Plan*). Concentriamoci sul piano di risposta a richieste di assistenza per emergenze.

Response Plan	CUSARP1	Customer A high priority response plan	Applies To	SR
Status	ACTIVE		Ranking	3
Status Date	28/06/10 13:52			











[Figura 4.11: creazione di un piano di risposta (“response plan”)]

Chiaramente avremo un piano di risposta ad alta priorità, priorità che, come per i “price schedule”, viene definita dalle *Conditions* e dal “ranking”. In più abbiniamo anche il cliente CompanyA e come fornitore reale del servizio (*Vendor*) selezioniamo la CompanyE. Impostiamo quindi le stesse condizioni applicate al piano dei prezzi per gli interventi di emergenza e applichiamo un “ranking” di 3.

Il campo *Applies To* si riferisce al tipo di oggetto al quale si vuole applicare il piano: noi impostiamo le *SR*, ovvero le “Service Request” (richieste di servizio, cioè i ticket di assistenza). Proprio a questo scopo viene creato anche un “ticket template”, ovvero un modello fisso per i ticket di assistenza, contenente dei valori preimpostati e un “job plan” (ovvero un piano di lavoro, una sorta di template dove è possibile pianificare delle attività con tutte le relative risorse). Il “ticket template” viene inserito all’interno del “response plan” e quando quest’ultimo verrà applicato ad una richiesta di servizio eseguirà le seguenti azioni:

- imposta alcuni campi della “service request” come quelli del “response plan”;
- imposta alcuni campi della “service request” come quelli del “ticket template”;
- applicando il template dei ticket alla richiesta di servizio viene automaticamente creato un ordine di lavoro che eredita le attività e le risorse pianificate nel piano di lavoro precedentemente creato (quindi applicando il “ticket template” alla “service request” viene creato automaticamente un “work order” che eredita “task” e risorse dal “job plan” precedentemente creato).



A questo punto ipotizziamo che la RedbookInc riceva una telefonata da CompanyA che gli riporta un problema ad un ascensore da risolvere il prima possibile. Come risultato di questa telefonata viene aperto un ticket di assistenza (quindi una “Service request”) che viene compilata come nella seguente figura 4.12:

Information	
Reported By	CUSA 
Name	Customer A Sao Paulo Building Manager 
Phone	11111111
E-mail	customeraaa@us.ibm.com
Service Request Details	
Summary	Broken Elevator 
Details	The elevator of the Customer A Sao Paulo Building has stopped. The client reported that the elevator was working properly yesterday, but today, when the employees of Sao Paulo building start arriving to work, they noticed that the elevator doesn't move. The elevator is stopped at the 1st floor with the door open.
Asset	CASPBELEV 
Location	CASPB 
Configuration Item	
Target Description	
GL Account	
Asset Site	
Customer	REDBOOKCUS- RedBook Customer A 

[Figura 4.12: compilazione di un ticket di assistenza]

Viene inserita la persona CUSA (referente del cliente) che ha riportato il problema e richiesto assistenza: con questo inserimento vengono automaticamente popolati i campi relativi ai dati di tale persona (Nome, numero di telefono e e-mail), viene impostato il relativo cliente e la locazione del lavoro da eseguire.

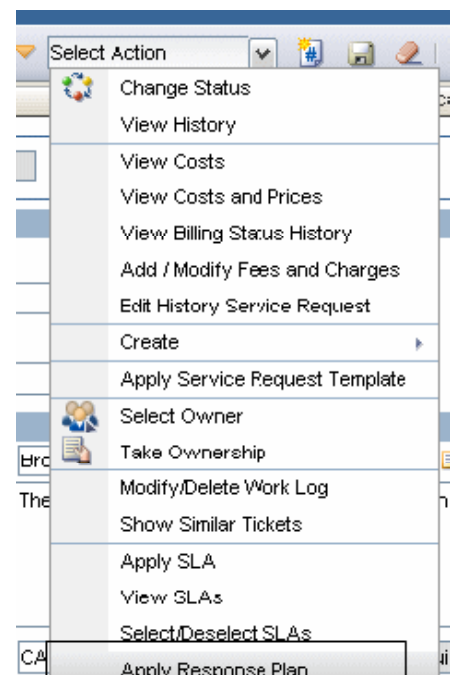
Quindi inseriamo la *Reported Priority* (priorità del problema riportata dal cliente) a 2 e la *Internal Priority* (la priorità assegnata all'intervento secondo il Service Provider) sempre a 2.

Reported Priority	<input type="text" value="2"/> 
Internal Priority	<input type="text" value="2"/> 

Infine applichiamo il piano di risposta precedentemente creato: dal menù “Select Action” si seleziona l'opzione “Apply Response Plan”. Maximo, fra tutti i “response plan” presenti in memoria, controlla quelli relativi al cliente designato nella “service request”, e se ce n'è più di uno controlla quello con le condizioni di applicabilità giuste: nel nostro caso la *Reported Priority* è quella relativa alle emergenze, quindi applicherà il piano CUSARP1. In questo modo, il campo relativo al *Vendor* verrà impostato a COMPANYE, l'asset designato per la gestione a CASPBELEV (l'ascensore da riparare) e verrà creato un ordine di lavoro per l'intervento.

Quest'ordine di lavoro, la cui pianificazione è stata automaticamente impostata dal “job plan” precedentemente creato, avrà selezionato il servizio di manutenzione straordinaria degli ascensori sull'asset suddetto, e il prezzo verrà regolato in automatico dall'agreement con il 100% di markup rispetto al costo.

A questo punto, dobbiamo aprire una breve parentesi sull'*acquisto* di servizi da un'azienda. Maximo for Asset Management (quindi l'insieme di applicazioni di base di Maximo) si occupa proprio della gestione di tutto ciò che viene



Select Action
Change Status
View History
View Costs
View Costs and Prices
View Billing Status History
Add / Modify Fees and Charges
Edit History Service Request
Create
Apply Service Request Template
Select Owner
Take Ownership
Modify/Delete Work Log
Show Similar Tickets
Apply SLA
View SLAs
Select/Deselect SLAs
Apply Response Plan

fornito da un fornitore all'azienda che utilizza Maximo. Nel nostro caso, RedbookInc acquista un servizio da CompanyE, e gestisce l'acquisto e la fattura relativa a tale servizio attraverso Maximo. Brevemente, riassumiamo i principali passi che un servizio deve seguire prima di essere utilizzabile dal Service Provider:

- 1) viene creato prima di tutto il servizio, definito “service item” nella logica di Maximo;
- 2) viene creato il contratto (in Maximo, “contract”, che non è un “customer agreement”) con il fornitore e al suo interno si inserisce il “service item” precedentemente creato;
- 3) viene creato un ordine di acquisto (“purchase order”) per tale servizio;
- 4) una volta che il servizio viene fisicamente reso disponibile dal fornitore, viene reso tale anche in Maximo tramite l'applicazione “receiving” e può quindi essere utilizzato all'interno di ordini di lavoro, di vendita, ecc..

Quindi RedbookInc stipula un contratto con CompanyE: quest'ultima fornirà un servizio di gestione degli ascensori per i clienti di RedbookInc al costo (per RedbookInc) di 300 euro a intervento.

### 4.2.3 Consuntivazione e fatturazione

A fine mese, RedBookInc può redigere ed emettere la fattura per il cliente CompanyA: crea quindi un *Bill Batch* dove vengono raccolti tutti i servizi forniti nell'ultimo mese.

The screenshot displays the Maximo Bill Batch configuration page. At the top, there are input fields for Bill Batch (CABB5), Customer (REDBOOKCUS-), Organization (EA/LENA), and Agreement (REDBKCA-AG1). The Bill Status is INPROGRESS, and the Bill End Date is 01/07/10. The Pre Tax Total is 600,00.

The main section is titled "Bill Batch Lines" and shows a table with the following data:

Line	Order	Class	Description	Total Cost	Quote Type	Agreed Price	Billed Price	Status
1	1159	WORKORDER		300,00		600,00	600,00	WAPPR

Below the table, there is a "Details" section for Line 1, showing fields for Line Number (1), Class (WORKORDER), Order (1159), Date Completed (28/06/10 21:17), Cust Charge Account, Cust Cost Center, Total Cost (300,00), Agreed Price (600,00), Billed Price (600,00), Agreement Rev Num (3), and Price Schedule (HIPRIO). A button "Copy WO's, Tickets and SO's" is visible at the bottom right of the details section.

At the bottom, there is a "Services" section with a table showing the following data:

Task	Service	Description	Unit Cost	Line Cost	Line Price	Bill Price
		Elevator Maintenance	300,00	300,00	600,00	600,00

[Figura 4.13: “bill batch” di fine mese]

L'applicazione *Bill Batch* raccoglie automaticamente tutti gli ordini di lavoro riguardanti un ben determinato cliente ed uno dei relativi agreement. Una volta selezionati tali due parametri, basta cliccare sul bottone *Copy WO's, Tickets and SO's* e Maximo automaticamente visualizza tutti i record interessati (tale bottone permette in realtà di raccogliere tutti gli ordini di lavoro, ordini di vendita, ticket e singole attività riguardanti uno specifico contratto, ma nel nostro caso verranno raccolti solo gli ordini di lavoro, poiché i “price schedule” del contratto si applicano esclusivamente ai “work order”).

Come notiamo dalla figura 4.13, è stato considerato per il mese di Giugno solo un intervento di emergenza, che quindi è stato fatturato per il cliente al 100% in più rispetto al costo sostenuto dal Service Provider per acquistarlo (notiamo infatti la differenza fra i campi *Total Cost* e *Billed Price*).

Nella parte inferiore dell'applicazione sono raccolti tutti i materiali, servizi, strumenti, ecc. che sono stati utilizzati nel mese in questione. E' anche possibile manualmente modificare l'importo *Bill Price* in caso si voglia fare qualche ulteriore scontistica o applicare qualche tassa in più.

Interessante è il campo *Bill End Date*: questo infatti viene impostato automaticamente da Maximo quando viene selezionato l'agreement desiderato. E' calcolato in base alla data di partenza dell'offerta, alla relativa durata e all'eventuale giorno di fatturazione, tutti parametri inseriti nel "customer agreement".

Un "bill batch" può essere creato anche prima del giorno di fatturazione: ogni volta che verranno aggiunte consuntivazioni ai "work order" già presi in considerazione dal "bill batch", queste saranno automaticamente comprese nella fattura fino al reale giorno di fatturazione. Se vengono creati nuovi "work order", basterà invece dire a Maximo di prenderli utilizzando il solito bottone *Copy WO's, Tickets and SO's*.

E' possibile anche fare in modo che il "bill batch" relativo a un contratto venga creato in automatico da Maximo il giorno di fatturazione: nell'applicazione "Customer Agreement", infatti, esiste proprio una sezione che si occupa di ciò (e si chiama "Billing Schedule").

Ogni ordine di lavoro, di vendita, singola attività, ecc. è dettagliata nelle "bill batch lines": troviamo il totale dei relativi costi sostenuti, il prezzo della merce e servizi da fatturare al cliente (*Billed Price*) e l'ammontare che il cliente è d'accordo a pagare (*Agreed Price*).

Ogni "bill batch line" è un record con un proprio stato, che viene modificato dal Service Provider oppure dal cliente in base a un ciclo ben determinato. Infatti tutti gli ordini di lavoro potrebbero dover essere visionati dal cliente prima di essere fatturati (a seconda chiaramente dei termini di contratto), e, a questo scopo, viene utilizzata l'applicazione "Bill Review". Supponendo un utilizzo di Maximo anche da parte del cliente, quando il "bill batch" viene posto nello stato PREBILL FOR CUSTOMER, esso è visibile a CompanyA utilizzando l'applicazione suddetta.

The screenshot shows the 'Bill Review' application interface. At the top, there are tabs for 'List', 'Bill Review', and 'Terms and Conditions'. Below the tabs, there are several input fields for bill batch details:

- Bill Batch: CABB5
- Customer: REDBOOKCUS-
- Reviewed By: [empty]
- Customer A Bill 5
- RedBook Customer A
- Organization: EAGLENA
- Required Review Date: [empty]
- Agreement: REDBKCA-AG1
- Reviewed Date: [empty]
- Attachments: [empty]
- Bill Status: PREBILL
- Bill End Date: 01/07/10
- Pre Tax Total: 600,00

Below the input fields, there is a message: "PREBILL batches not REVIEWED by the Required Date will be advanced to REVIEWED status - and all WAPPR Lines lines in the batch will be advanced to APPROVED status."

The main section is titled "Bill Batch Lines" and contains a table with the following data:

Line Number	Order	Class	Description	Date Completed	Quote Type	Billed Price	Status
1	1159	WORKORDER		28/06/10 21:17		600,00	WAPPR

Below the table, there is a section titled "Price Details" with a table showing the breakdown of costs:

Transaction	Bill Price
Labor	0,00
Material	0,00
Service	600,00
Tool	0,00
Fees and Charges	0,00

[Figura 4.14: revisione fatturazione da parte del cliente]

In essa CompanyA può controllare tutti i dettagli delle "bill batch lines": per ogni "line" può esaminare le relative risorse impiegate e il prezzo di ogni risorsa. Dopodiché può decidere se una linea è approvata, oppure se c'è bisogno di una qualche revisione. Per quest'ultimo caso l'ordine di lavoro viene posto nello stato di CUSTOMER DISPUTES: il Service Provider vedrà questa modifica e si metterà in contatto con il cliente per apportare eventuali modifiche ai prezzi imposti.

Una volta che tutte le "bill batch lines" sono state approvate (ponendo lo stato in APPROVED), RedbookInc potrà porre lo stato del "bill batch" in REVIEWED BY CUSTOMER ("esaminato dal

cliente”), ed emetterà la fattura. Una volta che la fattura sarà stata pagata, il “bill batch” potrà raggiungere lo stato finale di BILLED, diventando così un record nello storico delle fatture pagate.

Tutto questo ciclo di visione e revisione, da parte del cliente, delle fatture può anche essere tagliato nel caso non sia previsto dal contratto: il Service Provider approverà quindi lui stesso gli ordini di lavoro ed eseguirà gli opportuni cambi di stato senza far uso dell’applicazione “Bill Review”.

Un “bill batch” o una “bill batch line” possono avere molti altri stati definenti la situazione aziendale, ma chiaramente farebbero parte di un’analisi molto più approfondita che non è oggetto della seguente relazione.

### 4.3 Offerte aziendali

Con l’esperienza acquisita nel rappresentare all’interno di Maximo gli esempi precedenti e altri non menzionati, il passo successivo del mio stage è stato censire alcune delle offerte di Tempestive s.r.l., adattando così la logica di Maximo alle dinamiche dell’azienda. Qui di seguito verranno descritte le due tipologie di offerta più comuni per i clienti di Tempestive s.r.l.: un *offerta di consulenza generica* e un *progetto a corpo*. Saranno chiaramente omessi alcuni dettagli riguardanti l’utilizzo delle applicazioni di Maximo poiché menzionati negli esempi precedenti.

A causa delle clausole di segretezza definite dall’azienda per la tutela dei dati sensibili riguardanti l’azienda stessa e i relativi clienti, **ragioni sociali dei clienti, nomi del personale, luoghi e cifre economiche sono stati sostituiti con dati puramente esemplificativi od addirittura immaginari.**

#### 4.3.1 Offerta di consulenza generica

##### 4.3.1.1 Dettagli dell’offerta

- DESCRIZIONE: offerta di attività di consulenza informatica di vario genere, che potrà essere effettuata da 4 figure professionali:
  - Programmatore Junior
  - Programmatore Senior
  - Analista Software
  - Architetto Software
  
- CLIENTE: General Motors s.p.a.
- REFERENTE DEL CLIENTE: Matteo Pantani
- RESPONSABILE AZIENDALE: Bruno Bisello
- SEDI DI LAVORO: uffici di Tempestive s.r.l. nelle sedi di Pordenone e Padova e sede di General Motors s.p.a di Borgoricco (PD)
- PREZZO ATTIVITA’ DI CONSULENZA:
  - Attività Junior: euro 200 a G/U<sup>7</sup>
  - Attività Senior: euro 300 a G/U
  - Attività Analista: euro 450 a G/U
  - Attività Architetto: euro 550 a G/U
- RIMBORSO SPESE DI VIAGGIO DIPENDENTI:
  - Ufficio Pordenone: incluse nel prezzo

---


<sup>7</sup> Giornata Uomo: giornata di 8 ore lavorative, se non diversamente specificato.





- Ufficio Padova: euro 40
- Sede Borgoricco (PD): euro 60

### 4.3.1.2 Implementazione

Una volta che l'offerta viene accettata, la prima cosa da fare è censire il cliente, il relativo referente e le sedi di lavoro aggiuntive.

Customer:   

 Contacts can be people who work for the service provider, the customer, or a vendor of the customer. Contacts must be active in the People (SP) application. You can use the Security Groups (SP) application to set up access privileges for customer contacts.

Contact	Name	Type	Title	Primary Phone	Type	Customer/Vendor	Name
 MPANTANI	Matteo Pantani	CONTRACT		3487293712		GNR	General Motors s.p.a.

[Figura 4.15: referente del cliente]

Le “location” rappresentano il luogo fisico nel quale viene eseguito un lavoro: un'azienda che utilizza Maximo avrà in memoria già le proprie sedi e poi aggiungerà di volta in volta le locazioni dei vari clienti. Queste “location” sono molto importanti, poiché, come vedremo successivamente, possono essere legate ad uno o più clienti e impostate come luoghi di svolgimento delle attività all'interno di un ordine di lavoro.

Chi effettua gli interventi di consulenza sono i dipendenti dell'azienda censiti come “person” su Maximo. Il mestiere che una persona esercita, in Maximo è definito come “craft”, ed essendo che una persona può esercitare anche più mestieri diversi in base alle proprie qualificazioni, la si può identificare con un “labor”. Quest'ultimo rappresenta l'insieme delle associazioni fra una ben determinata persona e i suoi mestieri. Per ogni mestiere, un “labor” può avere prezzi diversi e anche premi extra impostati di default. In più può essere associato ad un calendario di eventi e di turni di lavoro prefissati e ricorrenti. Rappresenta insomma la vera e propria figura professionale che esegue il lavoro e consuntiva le proprie ore lavorate.

Poniamo di avere già in memoria tutti i dipendenti di Tempestive s.r.l. e per ognuno di essi sarà presente un ben determinato “labor” con i relativi “craft”, ma senza alcun prezzo impostato.

Il passo successivo è sicuramente la creazione e la compilazione del “customer agreement”. La fatturazione è di tipo mensile con termini di pagamento a 30 giorni, quindi avremo:

**Payment Terms: N30D**  
**Billing Cycle: MONTHLY**  
**Bill End Day: 1**

Successivamente, programmiamo la fatturazione automatica. La sezione “Billing Schedule” permette di programmare la creazione automatica dei “bill batch” relativi al contratto in caso di fatturazione mensile.

[Figura 4.16: pianificazione della fatturazione automatica mensile]

Come vediamo dalla figura 4.16, impostiamo una fatturazione mensile che si ripeta per 12 mesi. Il campo *First Bill Date* mi definisce il primo giorno nel quale avverrà la creazione automatica del “bill batch” relativo al contratto. Attraverso questa data e il valore del campo *Term*, Maximo definisce la *Final Bill Date*, ovvero l’ultimo giorno di programmazione della fatturazione automatica, e la *Next Bill Date*, cioè il prossimo giorno di fatturazione.

Notiamo, nella parte inferiore della figura, che si possono fissare anche dei prezzi fissi per ogni mensilità (tasse, utilizzo di materiali, gestione di asset, ecc.) che ad ogni scadenza verranno automaticamente aggiunti al totale da fatturare del “bill batch” creato. In questo modo si permette di automatizzare alcune consuntivazioni senza che queste vengano fatte manualmente attraverso ordini di lavoro, di vendita, o altro.

Passiamo ora alla redazione dei prezzi. A questo proposito, si è prima di tutto ricercato il metodo migliore per rappresentare una **giornata-uomo**. Banalmente, in un software dove il lavoro di una persona viene prezzato un TOT all’ora, si è pensato di dividere il prezzo totale di una giornata-uomo di consulenza per le ore della giornata lavorativa (quindi 8 ore), ottenendo quindi un prezzo orario. Nel momento della consuntivazione, Maximo moltiplicherà le otto ore lavorative che verranno impostate per il prezzo orario, ricostruendo quindi il costo di una giornata uomo per il cliente.

Considerando i nostri 4 mestieri sopra menzionati, otteniamo i seguenti prezzi orari:

Programmatore Junior: 25 euro/h  
 Programmatore Senior: 37,5 euro/h  
 Analista Software: 56,25 euro/h  
 Architetto Software: 68,75 euro/h

Vado quindi a creare un “price schedule” e inserisco nelle “pricing rules” i 4 mestieri, con i relativi prezzi, nella sezione “Labor” (nonostante la denominazione della sezione, nell’offerta inserisco i mestieri prezzati (“craft”), e non i “labor”: questi ultimi saranno utilizzati negli ordini di lavoro al momento della consuntivazione per identificare il reale lavoratore che ha eseguito l’attività).

Craft	Skill Level	Internal	External	Vendor	Labor Trans Type	Calendar	Shift	Price
SENIOR		<input checked="" type="checkbox"/>	<input type="checkbox"/>					37,50
JUNIOR		<input checked="" type="checkbox"/>	<input type="checkbox"/>					25,00
ARCHITEC		<input checked="" type="checkbox"/>	<input type="checkbox"/>					68,75
ANALYST		<input checked="" type="checkbox"/>	<input type="checkbox"/>					56,25

[Figura 4.17: impostazione dei prezzi per ogni tipo di figura professionale coinvolta]

Vediamo ora cosa succede quando viene effettuata un'attività di consulenza.

Ipotizziamo che venga eseguita una consulenza di due giorni presso la sede del cliente e che in essa siano impegnati, il primo giorno un analista, il secondo un programmatore senior.

Creiamo chiaramente un "work order" nuovo che avrà i seguenti valori:

- CLIENTE: General Motors s.p.a.
- LOCATION: sede di General Motors di Borgoriccio
- SUPERVISOR: Bruno Bisello

La presenza del cliente impostato, comporterà l'applicazione dell'offerta creata precedentemente e si potrà così procedere all'inserimento delle singole attività.

Definiamo quindi i seguenti due task:

1. Riunione per discussione su utilizzo di Tempestive ACTIVITY da parte del cliente
2. Installazione del software e primi cenni sull'utilizzo.

In entrambi questi task, verrà in automatico impostata la location dell'ordine di lavoro e in più inseriamo le 8 ore lavorate per ognuno e il responsabile dell'intervento.

Tasks for Work Order 1111

Sequence	Task	Summary	Estimated Duration	Status
1	10	Riunione per discussione su utilizzo di Tempestive	8:00	WAPPR
2	20	Installazione del software e primi cenni sull'utilizzo	8:00	WAPPR

**Task Information**

Task: 20 Installazione del software e primi cenni sull'utilizzo

Sequence: 2

Status: WAPPR

Classification: [ ]

Classification Description: [ ]

Under Flow Control?

Flow Action: [ ]

Flow Action Assist?

Launch Entry Name: [ ]

Attachments: [ ]

Inherit Status Changes?

Accepts Charges?

Owner: NCHAHINIAN

Owner Group: [ ]

**Work Reference and Measurement Information**

Reference WO: 1113

Location: GNR General Motors s.p.a. - sede di Borgoriccio (PD)

Asset: [ ]

Service Group: [ ]

Service: [ ]

Observation: [ ]

Inspector: [ ]

Measurement Point: [ ]

Measurement Value: [ ]

Measurement Date: [ ]

[Figura 4.18: task del "work order" creato]

Passando direttamente alla consuntivazione, aggiungiamo i nostri due lavoratori, che in automatico verranno prezzati come definito dal "customer agreement"

Task	Labor	Name	Approved	Start Date	Start Time	End Time	Regular Hours	Rate
10	BBIS	Bruno Bisello	<input checked="" type="checkbox"/>	23/05/12	9:00	18:00	8:00	0,00
20	NCHA	Nicola Chahinian	<input checked="" type="checkbox"/>	23/05/12	9:00	18:00	8:00	0,00

Labor	
Task: 20	Labor: NCHA >> Nicola Chahinian
Approved? <input checked="" type="checkbox"/>	
Details	
Craft: SENIOR	Start Date: 23/05/12
End Date: 23/05/12	Type: WORK
Skill Level:	Start Time: 9:00
End Time: 18:00	Timer Status:
Regular Hours: 8:00	Line Cost: 0,00
Rate: 0,00	Line Price: 300,00 (37,5 * 8)

[Figura 4.19: imposizione automatica del prezzo al momento della consuntivazione]

Come vediamo, Maximo, utilizzando i prezzi orari imposti sul “customer agreement”, calcola il prezzo finale moltiplicando il numero di ore lavorative (8) per il costo orario di un programmatore senior (37,5 euro), ottenendo 300 euro per una giornata – uomo.

Essendo che i 2 interventi sono stati eseguiti nella sede del cliente, dobbiamo aggiungere anche i costi di trasferta: nella sezione “Tax and Fee” aggiungiamo 2 voci da 60 euro per le due trasferte a Borgorico, definendole come spese di viaggio (“trip charge”)

Task	Type	Description	Line Price
10	TRIP	Trasferta Borgorico	60,00
20	TRIP	Trasferta Borgorico	60,00

[Figura 4.20: costi di trasferta]

A questo punto a fine mese (o il giorno stabilito per la fatturazione) Maximo crea automaticamente il Bill Batch, raccogliendo al suo interno l’ordine di lavoro che abbiamo creato e consuntivato precedentemente (l’utilizzo specifico del Bill Batch lo vediamo nella successiva offerta).

## 4.3.2 Progetto a corpo

### 4.3.2.1 Dettagli dell’offerta:

- DESCRIZIONE:
  - vendita di 2 licenze del software Tempestive dbPortlet per 20 dbPortlet (10 ciascuna) complessive;

- realizzazione di 6 portlet “custom” per la visualizzazione dei KPI.
- CLIENTE: Uliveto s.r.l.
- REFERENTE DEL CLIENTE: Giorgio Molinari
- RESPONSABILE AZIENDALE: Roberto Vitali
- SEDI DI LAVORO: le attività verranno interamente svolte da remoto dalle sedi di Tempestive Srl di Pordenone e Padova. L’attività di installazione e configurazione dei portlet sarà svolta on site presso la sede di Uliveto s.p.a. (Parma)
- PREZZO PROGETTO
  - Licenza Tempestive dbPortlet, prezzo di listino cadauno: euro 1.500;
  - Prezzo del progetto complessivo, comprensivo delle due licenze d’uso dbPortlet: euro 5000 (spese di trasferta incluse);
- MODALITA’ DI FATTURAZIONE:
  - 20% dell’ammontare complessivo alla firma dell’offerta;
  - Saldo a termine attività.

### 4.3.2.2 Implementazione

Dopo aver creato, come nell’esempio precedente, il cliente con il relativo rappresentante e la locazione della propria sede, andiamo a redigere il contratto.

Questa volta il “customer agreement” sarà molto più breve da compilare. Questo perché non vi sono risorse da fatturare con prezzi orari, fatturazioni automatiche da programmare o condizioni da specificare. Vengono compilati infatti solamente il termine di pagamento (N30D), la date di inizio e di fine contratto e viene creato un “price schedule” da applicare ad un “work order”: quest’ordine di lavoro conterrà le fasi del nostro progetto. Con questo abbiamo terminato la compilazione dell’“agreement”: infatti, i corrispettivi che il cliente dovrà pagare a Tempestive saranno contenuti nell’ordine di lavoro, legati ad una ben determinata fase del progetto.

Passiamo quindi alla creazione dell’ordine di lavoro. Questo, nel caso di un progetto a corpo, contiene le seguenti informazioni

- Informazioni generiche sul lavoro;
- Fasi di tutto il progetto;
- Corrispettivi per ogni fase;
- Consuntivazioni delle fasi;
- (eventuale) automatismo del passaggio da una fase alla successiva.

In particolare quest’ultima è un’opzione particolare dell’applicazione “Work Order”: si può fare in modo che l’ordine di esecuzione dei “task” segua un ben preciso schema di svolgimento, eseguendo prima alcuni “task” di altri. Questo, nella pratica, si risolve con dei consecutivi cambi di stato: una volta che queste attività sono state numerate in sequenza, quando il “work order” viene posto nello stato IN PROGRESS, la prima attività, nell’ordine, assume il medesimo stato, mentre le altre rimangono nello stato WAITING FOR APPROVAL (letteralmente, “in attesa di approvazione”). Quando l’attività viene terminata, il relativo “task” viene (manualmente) posto nello stato di completamento e la seconda attività nell’ordine viene (automaticamente) posta nello stato IN PROGRESS. In questo modo si segue un flusso di lavoro prestabilito, diminuendo il numero di click che un utente deve fare ogni volta che termina una fase e ne inizia un’altra.

Pianifichiamo, quindi, il nostro ordine, definendo 4 “task”:

1. Vendita 2 licenze Tempestive dbPortlet
2. FASE 1: Implementazione di 6 portlet “custom”
3. FASE 2: Test di funzionamento
4. FASE 3: Istruzione dipendenti

In più sulla sezione “Tax e Fee” inseriamo i 2 corrispettivi del pagamento:

1. 20 % dell’ammontare totale alla vendita delle licenze: 1000 euro
2. Il saldo al termine delle attività: 4000 euro

**Tasks for Work Order 1114**

Sequence	Task	Summary	Estimated Duration	Status	Owner	Owner Group
1	10	Vendita 2 licenze Tempestive dbPortlet	0:00	INPRG	GMOLINARI	
2	20	FASE 1: implementazione di 6 portlet custom	64:00	WAPPR	NCHAHINIAN	
3	30	Fase 2: test di funzionamento	32:00	WAPPR	NCHAHINIAN	
4	40	FASE 3: istruzione dipendenti	48:00	WAPPR	NCHAHINIAN	

**Fees and Charges**

Task	Type	Description	Line Price
10	MANAGEMENT	Corrispettivo iniziale	1.000,00
40	MANAGEMENT	Corrispettivo termine lavori (saldo finale)	4.000,00

[Figura 4.21: fasi e relativi corrispettivi nell’ordine di lavoro di un progetto a corpo]

Come notiamo dalla figura 4.21, ogni corrispettivo è legato alla propria attività; questo per permettere una completa comprensione dell’operazione e del pagamento.

A questo punto il nostro ordine di lavoro viene posto IN PROGRESS e, di conseguenza, anche il primo task.

Alla firma dell’offerta da parte del cliente, viene fatturato, come da contratto, il primo corrispettivo. In Maximo, quindi, questo ammontare viene consuntivato, il primo task viene posto nello stato COMPLETE e viene creato (manualmente) il relativo “bill batch”, che includerà il corrente ordine di lavoro con la sua consuntivazione.

Al termine di tutte le fasi verrà consuntivato il saldo del pagamento: verrà creato così un secondo “bill batch” che raccoglierà dentro di sé **lo stesso ordine di lavoro**, ma con solamente la seconda consuntivazione.

**Bill Batch Lines**

Line	Order	Class	Description	Total Cost	Quote Type	Agreed Price	Billed Price	Status
1	1114	WORKORDER	Progetto a corpo - creazione Portlet per visualizzazione KPI	0,00		1.000,00	1.000,00	APPROVED

**Fees and Charges Details**

Type	Description	Calculation	Line Price	Bill Price
MANAGEMENT	Corrispettivo iniziale	ENTERED	1.000,00	1.000,00

[Figura 4.22: Fatturazione FASE 1]

Line	Order	Class	Description	Total Cost	Quote Type	Agreed Price	Billed Price	Status
1	1114	WORKORDER	Progetto a corpo - creazione Portlet per visualizzazione KPI	0,00		4.000,00	4.000,00	WAPPR

Type	Description	Calculation	Line Price	Bill Price
MANAGEMENT	Corrispettivo termine lavori (saldo finale)	ENTERED	4.000,00	4.000,00

[Figura 4.23: saldo a termine lavori]

Maximo quindi tiene traccia delle risorse e dei servizi che vengono consuntivati: quando un elemento di un ordine di lavoro (o di vendita) viene fatturato in un “bill batch”, l’ordine potrà essere fatturato in quello successivo solo se al suo interno avrà una o più nuove consuntivazioni di elementi non fatturati in quello precedente. In questo modo, viene data la possibilità, per esempio, di eseguire un ordine di lavoro a cavallo fra due mesi: nel primo mese verranno fatturate alcune risorse, nel secondo altre, tutte però appartenenti allo stesso lavoro.

### 4.3.3 Osservazioni personali

Queste rappresentazioni delle offerte dell’azienda Tempestive s.r.l. possono comunque essere introdotte anche in altri modi all’interno di Maximo. Ad esempio, nel caso della consulenza, si può creare un “work order” per ogni attività, oppure definire il prezzo a giornata-uomo come il prezzo totale applicato ad un’ora di lavoro, ecc.; nel caso del progetto a corpo, si potevano ad esempio creare due “price schedule”, uno per un ordine di vendita (per vendere le licenze) e uno per l’ordine di lavoro (per inserire le fasi dei lavori in corso).

Il mio personale obiettivo è stato quello di sintetizzare tutte le caratteristiche delle offerte, dell’intervento di assistenza e del progetto a corpo in modo da far fare ad un ipotetico utilizzatore del software il minor numero di click possibile: questo in vista del progetto finale, nel quale questa prerogativa sarà uno degli obiettivi più importanti.

## 4.4 Creazione automatica di fatture e report per l’azienda

### 4.4.1 Design

Finora il nostro punto di arrivo sulla creazione di un’offerta all’interno di Maximo è stato sempre il “bill batch”. Ogni azienda ha però anche la necessità di un metodo di creazione automatica di fatture o di rapporti delle attività svolte, sia per uso interno all’azienda stessa, sia da presentare ai clienti. Un software che permette di creare dei report è **BIRT Report Designer**. Come spiegato nel capitolo 2, questo software permette di interagire con i database e creare, con i dati messi a disposizione, diverse tipologie di report.

Un file “.rptdesign” (il file del report creato con questo software) si integra benissimo con IBM Maximo, poiché al suo interno contiene un modulo per la gestione della reportistica che utilizza proprio file con questa estensione.

Un modo per rendere più automatizzata un’azienda Service Provider con molti clienti sarebbe proprio possedere un metodo per la creazione automatica delle fatture. Ciò che serve per realizzarlo è appunto:

- un database di dati aziendali riferiti ai clienti (come quello di Maximo);
- BIRT Report designer.

Prendendo spunto dal “layout” delle fatture di Tempestive, con l’utilizzo del suddetto software ho creato un template<sup>8</sup> per le fatture che si integrasse con il database di Maximo.

---

<sup>8</sup> N.B.: la fattura di seguito è riferita al periodo in cui l’azienda Tempestive s.r.l. non aveva ancora cambiato nome e quindi assunto la nuova ragione sociale. Si riferisce quindi alla stessa azienda ma con il vecchio nome di “Santin e Associati s.r.l.”. Inoltre **tutti i dati ivi riportati sono dati casuali e non corrispondenti nella realtà per ragioni di privacy dell’azienda suddetta.**





**SANTIN E ASSOCIATI**  
software e sistemi distribuiti

Spettabile

**GENERAL MOTORS S.P.A.**

via Adria, 4

35010 BORGORICCO (PD)

FATTURA

DEL 2011-11-30

RIFERIMENTO: NOSTRA OFFERTA OHYP20110415

DESCRIZIONE	QUANTITA'	COSTO/UNITARIO	IMPORTO
SALDO OFFERTA PER ATTIVITÀ DI CONSULENZA GENERICA			€ 870.00
IMPONIBILE			€ 870.00
IVA 20%			€ 174.00
<b>TOTALE</b>			<b>€ 1,044.00</b>

(S.E.&O.) CONTRIBUTO CONAI ASSOLTO OVE DOVUTO - IL CLIENTE CONTROLLI I PROPRI DATI COMUNICANDO EVENTUALI INESATTEZZE

**PAGAMENTO**

Veneto Banca - Padova

IBAN AAABBBCCDDDEEEFFF111222333

IBM Premier Business Partner  
MICROSOFT Gold Certified Partner  
SYMANTEC Silver Partner

[www.santineassociati.com](http://www.santineassociati.com)  
[sales@santineassociati.com](mailto:sales@santineassociati.com)

**Sede legale**

Santin e Associati S.r.l.  
I-33170 Pordenone (PN)  
Via P.Amalteo 32/A

Cap. Soc. 54.254 i.v.  
P.IVA e Reg.Imp. IT 11223344556  
REA CCIAA PN 55667

**Sedi operative**

I-33170 Pordenone (PN)  
Via P. Amalteo 32/A  
Tel. +39 0434 570305

35142 Padova (PD)  
Via Palermo 42/4  
Tel. +39 049 8774060

[Figura 4.24: Fattura]

Guardando la figura 4.24, possiamo distinguere, dall'alto verso il basso, tre tipi di informazione:

1. *Prima parte in alto a destra*: dati riguardanti il cliente e l'indirizzo della relativa sede;
2. *Data, offerta, descrizione dell'offerta e imponibile*: dati riguardanti l'offerta e la fatturazione;
3. *IVA e totale*: calcoli eseguiti sull'imponibile.
4. Dati riguardanti l'azienda Santin e Associati s.r.l.

Ognuna di queste informazioni, è frutto di diverse interrogazioni effettuate sul database di Maximo. Vediamo allora come sono stati eseguiti i 4 step definiti nel capitolo 3:

- 1) Definizione del *Data Source*:  
driver: com.ibm.db2.jcc.DB2Driver (v3.62)  
URL: jdbc:db2://10.1.2.103:50000/maxdb75<sup>9</sup>
- 2) Definizione del *Data Set*: l'insieme delle tabelle e dei relativi campi da considerare si ottiene con una semplice interrogazione eseguita sul database di Maximo. Vengono prese in considerazione le tabelle dei "bill batch", dei "customer agreement", dei "customer" e altre tabelle.
- 3) Definizione dei parametri di input: due valori in particolare devono essere inseriti manualmente a richiesta prima di eseguire un report, ovvero l'*identificativo* dell'offerta e il *tasso di IVA* da applicare all'imponibile. Vengono creati quindi due parametri di inserimento, in particolare due stringhe che verranno opportunamente formattate prima di essere inserite nel report. L'identificativo dell'offerta non è altro che il campo "Agreement" della tabella PLUSPAGREEMENT del database di Maximo, campo di partenza per ottenere tutti i valori desiderati.
- 4) Definizione del layout e dei campi del report; procedendo dall'alto verso il basso del report abbiamo
  - **Informazioni cliente**: valori ottenuti con "query" eseguite sulla tabella del cliente e della relativa "location";
  - **Data**: la data di partenza dell'offerta, ottenuta dalla tabella dei "customer agreement";
  - **Riferimento offerta**: l'identificativo del record del contratto;
  - **Offerta**: la stringa di testo di descrizione dell'offerta, ottenuta dalla tabella relativa al contratto per il cliente;
  - **Imponibile**: rappresenta il campo *Pre Tax Total* presente sul "bill batch" relativo alla fatturazione corrente;
  - **IVA**: l'iva viene calcolata sull'imponibile, utilizzando la seguente formula all'interno del programma:

$$\text{'€} + (\text{dataSetRow}[\text{"PRETAXTOTAL"}]/100) * \text{params}[\text{"IVA"}].\text{value}$$

Come notiamo, rappresenta una concatenazione di una stringa con un valore numerico calcolato sul prodotto fra campo *Pre Tax Total* (diviso per 100) e il valore del parametro sull'IVA inserito manualmente ;

- **Totale**: anche qui abbiamo una stringa concatenata ad un valore calcolato. Il totale si ottiene infatti sommando l'IVA all'imponibile, con la seguente formula:

$$\text{'€} + (\text{dataSetRow}[\text{"PRETAXTOTAL"}] + ((\text{dataSetRow}[\text{"PRETAXTOTAL"}])/100) * \text{params}[\text{"IVA"}].\text{value})$$

<sup>9</sup> 10.1.2.103 è l'indirizzo IP della locazione fisica del database, 50.000 è la porta con il quale comunica IBM DB2, "maxdb75" è il nome del database di Maximo

## 4.4.2 Maximo e i report

Come abbiamo già spiegato, è possibile utilizzare i file “.rptdesign” per creare un report direttamente da Maximo. L’applicazione “Report Administration” raccoglie proprio tutti i file di quel tipo e li sincronizza con il software.

The screenshot shows the Maximo Report Administration interface. At the top, there are tabs for 'List', 'Report', 'Security', and 'Performance'. Below the tabs, the 'Report' tab is active, displaying the following configuration:

- Report File Name: billing.rptdesign
- Report Type: BIRT
- Imported by: MAXADMIN
- Fatturazione automatica
- Application: PLUSPBILL (circled in red)
- Report Folder: PLUSPBILLBATCH
- Last Import Date: 30/03/12 18.09

Below the configuration fields is a 'Settings' section with various checkboxes and input fields for report behavior, such as 'Limit Records?', 'No Request Page?', 'Browser View?', 'Max Record Limit', 'Use Where Clause?', 'Direct Print?', 'Schedule Only?', 'Display Order', 'Priority', 'Toolbar Sequence', 'Direct Print Location', 'Direct Print with Attachments?', and 'Direct Print with Attachments Location'. There are 'Generate Request Page' and 'Preview' buttons at the bottom right of the settings section.

At the bottom, there is a 'Parameters' section with a table of input parameters:

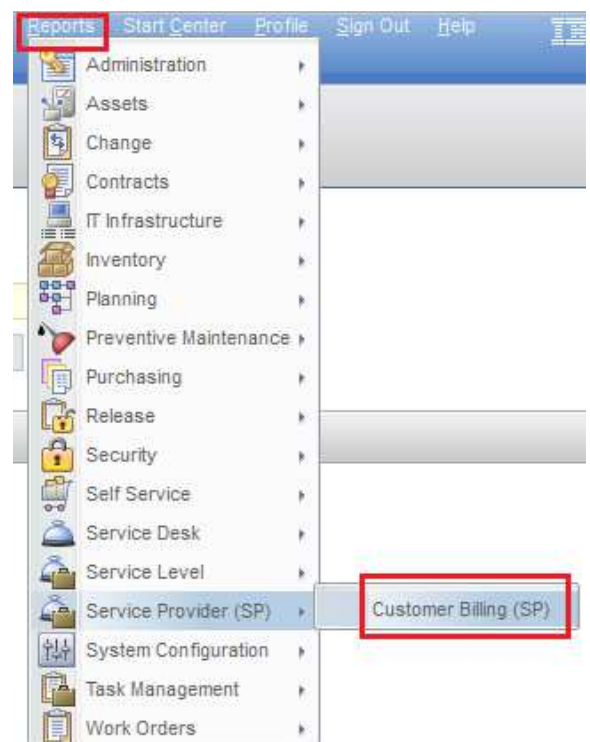
Parameter Name	Attribute Name	Sequence	Display Name	Required
Contratto (circled in blue)	agreement	1	Seleziona il contratto desiderato	<input checked="" type="checkbox"/>
IVA (circled in blue)		2	Inserisci IVA da applicare	<input checked="" type="checkbox"/>

[Figura 4.25: la configurazione di un file .rptdesign all’interno di Maximo. Cerchiata in rosso è l’applicazione che lancia il report corrente e cerchiati in blu sono i parametri di ingresso]

Quando viene inserito un nuovo file di reportistica, questo viene associato prima di tutto ad un’applicazione: questo perché sia possibile lanciare il report direttamente da quell’applicazione oppure dal menù *Report* presente nella parte alta dell’interfaccia grafica.

E’ necessario anche allineare i parametri di input, in modo che Maximo, quando viene lanciato un report, li richieda utilizzando la sua interfaccia. E’ possibile definire anche una *tabella di lookup*, ovvero una tabella dalla quale andare a selezionare direttamente i valori ricercati, senza doverli scrivere manualmente.

Una volta salvato, il mio report è pronto per l’uso all’interno di Maximo. Nel caso della fattura precedentemente creata, ho scelto di associarla all’applicazione “Bill Batch” in modo che, una volta approvata, la fattura possa essere stampata direttamente dall’applicazione che contiene proprio i dati della fatturazione.



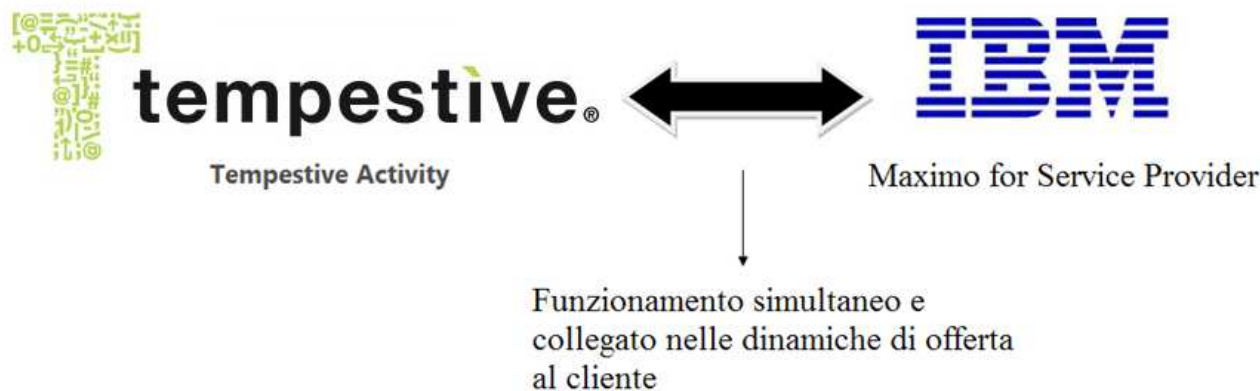
## Capitolo 5

# Progetto finale

### 5.1 Premessa al progetto

Lo stage presso Tempestive s.r.l. prevedeva come obiettivo finale lo sviluppo di un progetto che permettesse allo stagista di utilizzare le conoscenze apprese sull'utilizzo di IBM Maximo e sul ciclo di vita delle richieste di servizi fatte ad un'azienda service provider. Il progetto consiste in un'opera di integrazione fra il software IBM Maximo e l'applicazione Tempestive ACTIVITY, applicazione di produzione dell'azienda per la gestione delle attività aziendali e descritta nel capitolo 3.

Quest'opera di integrazione richiede una comunicazione fra il software e l'applicazione per lo scambio di informazioni riguardanti i clienti, le relative offerte e le attività ad esse connesse. Lo scopo di questo lavoro è unire le capacità di organizzazione delle attività dei dipendenti (svolte per i clienti dell'azienda) di Activity con le potenzialità del ciclo di fatturazione di Maximo for Service Provider. Utilizzando quindi i dettagli contenuti nei report eseguiti dai dipendenti, si vuole automatizzare la creazione delle fatture e dei rapporti sulle attività eseguite.



[Figura 5.1: premessa al progetto]

### 5.2 Ciclo delle offerte su Activity

Attualmente Tempestive s.r.l. utilizza l'applicazione Activity per gestire e tracciare le attività eseguite per i propri clienti.

Quando viene emessa un'offerta da parte dell'azienda, la responsabile commerciale provvede a censirla sull'applicazione insieme col relativo cliente. I passi successivi, che vengono eseguiti durante i lavori, sono i seguenti:

- la responsabile commerciale inserisce i task<sup>10</sup> relativi all'offerta;

<sup>10</sup> Il significato dei "Task" varia a seconda dell'offerta di riferimento. Possono rappresentare le fasi di un progetto, un insieme di attività eseguite su uno stesso asset, oppure diverse tipologie di attività di consulenza o di assistenza. Nel corso della relazione verrà maggiormente chiarito il loro senso.

- il dipendente che ha eseguito un attività per un cliente crea su Activity il report relativo al suo lavoro, selezionando il task adeguato e inserendo tutte le informazioni riguardanti l'attività svolta e le eventuali spese aggiuntive per le trasferte;
- al momento della fatturazione, la responsabile commerciale crea, se richiesto dal cliente, un report delle attività svolte fino a quel momento utilizzando i dati contenuti nello storico dei report dei dipendenti su Activity. Il report viene creato manualmente con un uso combinato di Microsoft Excel e Microsoft InfoPath. Con le stesse modalità, crea infine la fattura per il cliente.

Il progetto seguente permetterà di eliminare alcuni passaggi descritti, come l'inserimento dei task e la redazione manuale dei report delle attività e delle fatture.

### 5.3 Analisi dei requisiti

Di seguito analizzeremo l'insieme dei requisiti minimi (hardware e software) per l'utilizzo dei software in questione e le reali caratteristiche delle macchine utilizzate

#### 5.3.1 Requisiti di sistema

Di seguito i requisiti di sistema relativi all'installazione del software e delle varie applicazioni

Prodotto	Sistema Operativo	CPU	RAM	Altro
IBM Maximo – Client Workstation	Microsoft Windows XP Professional, Windows Vista o Windows 7, 32 bit & 64 bit	2-4 GHz o superiore	2-4 GB o superiore	Scheda grafica SVGA 1024x768 o superiore
IBM Maximo – Administrative Workstation	Microsoft Windows XP Pro, Windows Vista o Windows 7, 32 & 64 bit	2-4 GHz o superiore	4 GB o superiore	
IBM WebSphere Application Server	Microsoft Windows Server 2003, 2008, 2008 R2 (32-bit & 64-bit), Redhat Enterprise Linux versioni 4,5 e 6 (sistemi x-86 a 32 & 64 bit), SUSE Linux Enterprise Edition versioni 9, 10 e 11 (sistemi x-86 32 & 64 bit)	Dual CPU o superiore	2GB per processore o superiore	11 GB liberi di spazio su disco (comprensivi per installazione componenti software Maximo)
IBM DB2 Database	Windows XP Pro, Windows Vista o Windows 7, Windows Server 2003,2008 e 2008 R2 (tutti sia a 32 che a 64 bit)	Tutti i processori Intel e AMD	512 MB o superiore	20 GB di spazio su disco (comprensivi di installazione tabelle e dati di Maximo)

[Figura 5.2: Requisiti di sistema per i componenti dell'installazione di IBM Maximo]

.NET Framework 4	Windows Xp Home o Professional, Windows Vista o Windows 7, Windows Server 2003, 2008 e 2008 R2 (tutti sia a 32 che a 64 bit)	1 GHz o superiore	512 MB o superiore	Spazioso disco – 850 MB (32 bit) – 2 GB (64 bit)
Microsoft Sql Server 2012 Express	Windows Vista o Windows 7, Windows Server 2008 e 2008 R2 (32 & 64 bit)	– 1GHz o superiore (32 bit) – 1,4 GHz o superiore (64 bit)	1GB o superiore	6 GB di spazio su disco

[Figura 5.3: Requisiti di sistema per i componenti dell'installazione di Tempestive Activity]

### 5.3.2 Requisiti funzionali

Il progetto deve permettere la realizzazione di una comunicazione parziale fra il software e l'applicazione in questione, tale da permettere ai dipendenti dell'azienda di svolgere delle operazioni ripetitive e nel minor tempo possibile (diminuzione del TCO aziendale) per la gestione del ciclo di vita di una richiesta di servizi da parte di un cliente.

In particolare si vuole:

- inserire il software IBM Maximo (già utilizzato dall'azienda soltanto per i relativi clienti) all'interno delle dinamiche di Tempestive s.r.l., adattandolo ad esse;
- collegare il software IBM Maximo all'applicazione Tempestive Activity, già utilizzata dall'azienda per l'organizzazione delle proprie attività, progetti, task, clienti, ecc.;
- rendere Tempestive Activity adatto per la gestione della fatturazione rivolta ai clienti dell'azienda nel suo ruolo di Service Provider;
- ridurre il TCO aziendale eliminando alcune operazioni già eseguite routinariamente all'interno dell'azienda;

### 5.3.3 Postazione di lavoro

Mi sono stati messi a disposizione per il progetto 2 computer che in questa sede chiameremo *Computer 1* e *Computer 2*, il primo su cui installare IBM Maximo (con relativo database), il secondo su cui installare Tempestive Activity (con relativo database). Le caratteristiche dei due computer sono le seguenti:

<i>Computer 1</i>	<i>Computer 2</i>
<b>Sistema operativo:</b> Windows 7 Enterprise	<b>Sistema operativo:</b> Windows 7 Enterprise
<b>Versione SO:</b> 64 bit	<b>Versione SO:</b> 64 bit
<b>CPU:</b> Intel Core 2 CPU	<b>CPU:</b> AMD Athlon 64 Processor 3200 + 2.02 GHz
<b>RAM:</b> 3GB	<b>RAM:</b> 3 GB

[Figura 5.4: caratteristiche delle postazioni coinvolte nel progetto]

Le due macchine sono collegate fra loro attraverso la LAN dell'azienda.

## 5.4 Descrizione del progetto

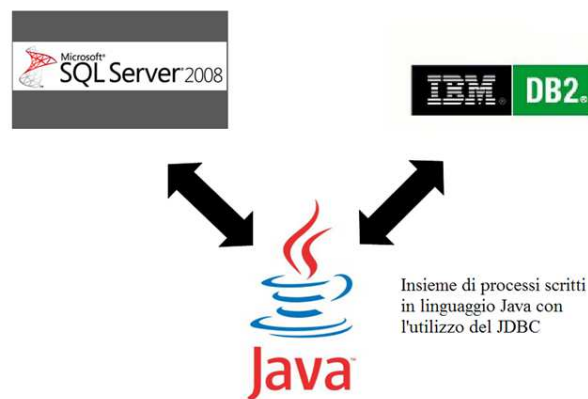
### 5.4.1 Modalità di implementazione

L'azienda in questione mi ha richiesto un collegamento simultaneo fra i suddetti programmi, lasciandomi campo libero sul come eseguire questo tipo di integrazione.

Essendo che entrambi si appoggiano ad un database per la memorizzazione dei dati, l'idea è stata quella di eseguire lo scambio di informazioni direttamente fra di essi: attraverso interrogazioni, inserimenti, cancellazioni, ecc. eseguiti su "IBM DB2 Database" e su "Microsoft Sql Server 2008 Database" si vogliono allineare i dati di clienti, offerte, ordini di lavoro, ecc. e poi affidare a Maximo la stampa di fatture e report attività.

Gli esecutori di queste "query" saranno un insieme di processi sempre attivi e programmati per un controllo continuo dei due database e, in base al verificarsi di alcune condizioni, capaci di eseguire ben determinate operazioni.

Questi processi sono stati scritti in linguaggio Java, una scelta fatta soprattutto per allinearli al mio percorso di studi universitario, poiché è stato il linguaggio con cui sono maggiormente venuto a contatto. Con l'utilizzo del JDBC, questi processi si collegano ai database ed eseguono azioni sia routinarie che straordinarie (ovvero legate a particolari record o a particolari valori di essi).



### 5.4.2 Tipologie di offerta da implementate

Tempestive s.r.l. fornisce ai propri clienti diverse tipologie di offerta: consulenze di diverso tipo, interventi di assistenza, progetti a corpo, vendita di licenze software, ecc.. Implementarle tutte all'interno del progetto in un periodo di 2 mesi sarebbe stato troppo lungo e oneroso. Si è scelto così, insieme al mio tutor aziendale, di considerare solamente i due tipi di offerta più frequenti:

- 1) *offerte di consulenza di vario genere;*
- 2) *progetti a corpo.*

L'ambito di questi due contratti è stato già esposto nel capitolo relativo all'azienda (Capitolo 2), di conseguenza da ora in poi descriveremo solamente le relative dinamiche applicate al progetto di integrazione da me svolto.

### 5.4.3 Personale addetto

Evidenziamo 3 figure professionali all'interno dell'azienda che utilizzeranno appieno o in parte il risultato del progetto corrente:

1. **Responsabile commerciale:** colui che si occupa della parte commerciale dell'azienda, ovvero redige e gestisce i contratti, le fatture, i rimborsi, ecc.;
2. **Responsabile del progetto:** è il dipendente che è stato incaricato di seguire e coordinare le operazioni relative ad un progetto o ad un insieme di consulenze. E' lui stesso che è entrato

in contatto con il referente del cliente e con il quale ha pattuito i termini e le condizioni da applicare al contratto;

3. **Dipendente lavoratore:** è il normale dipendente che svolge o partecipa alle attività di un ordine di lavoro.

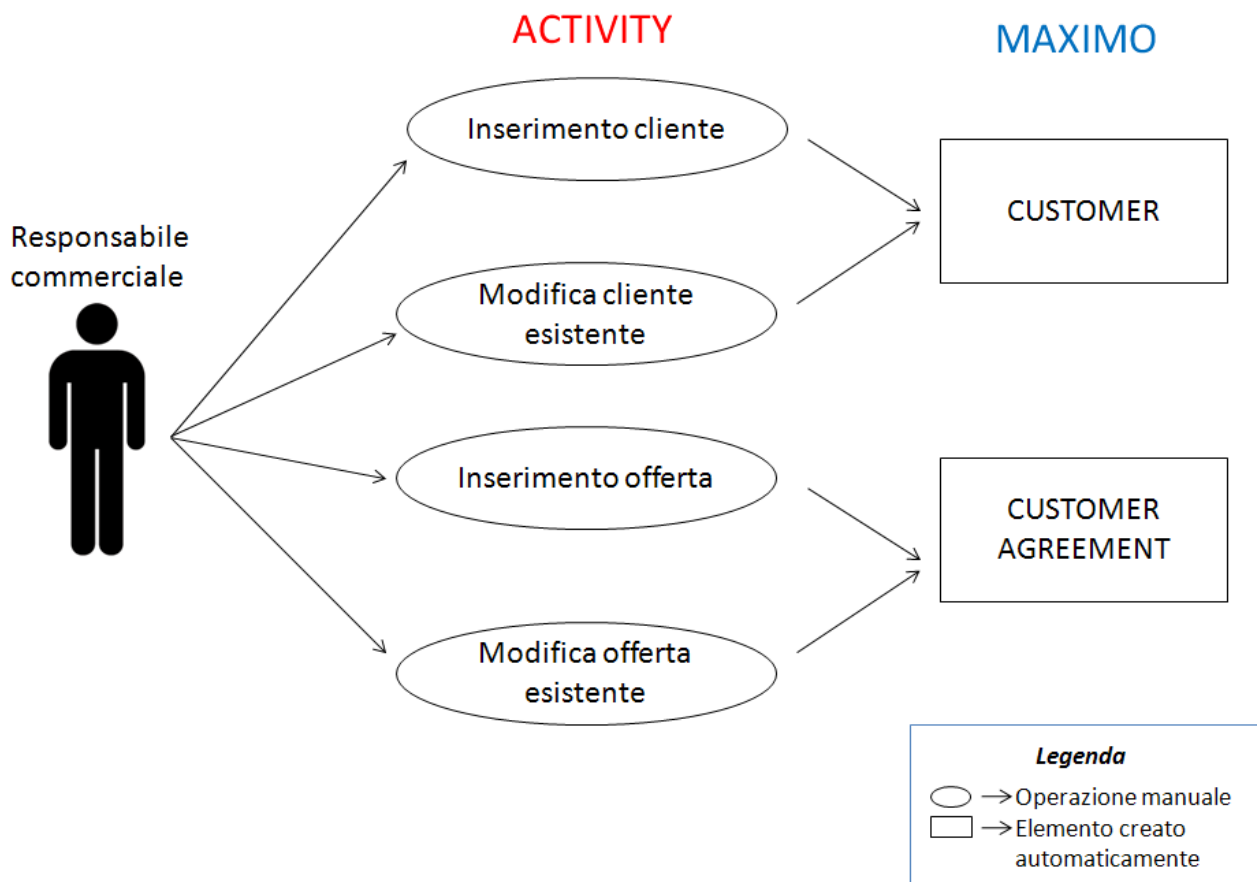
Ognuna di queste figure ha delle incombenze specifiche che elencheremo passo a passo nei seguenti *casi di utilizzo*.

## 5.5 Casi di utilizzo

Di seguito vedremo quali sono le informazioni da gestire per il corretto andamento delle offerte all'interno dell'integrazione. Considereremo sia flussi di informazioni che saranno organizzati nello stesso modo per entrambe le tipologie di offerta prese in considerazione, sia flussi peculiari di ognuna. Definiremo diversi casi di utilizzo e gestione, che poi assembleremo in 2 unici schemi finali.

### 5.5.1 Clienti e progetti

Per qualunque tipologia di offerta, le prime cose da registrare sono le informazioni riguardanti il cliente e il relativo contratto.



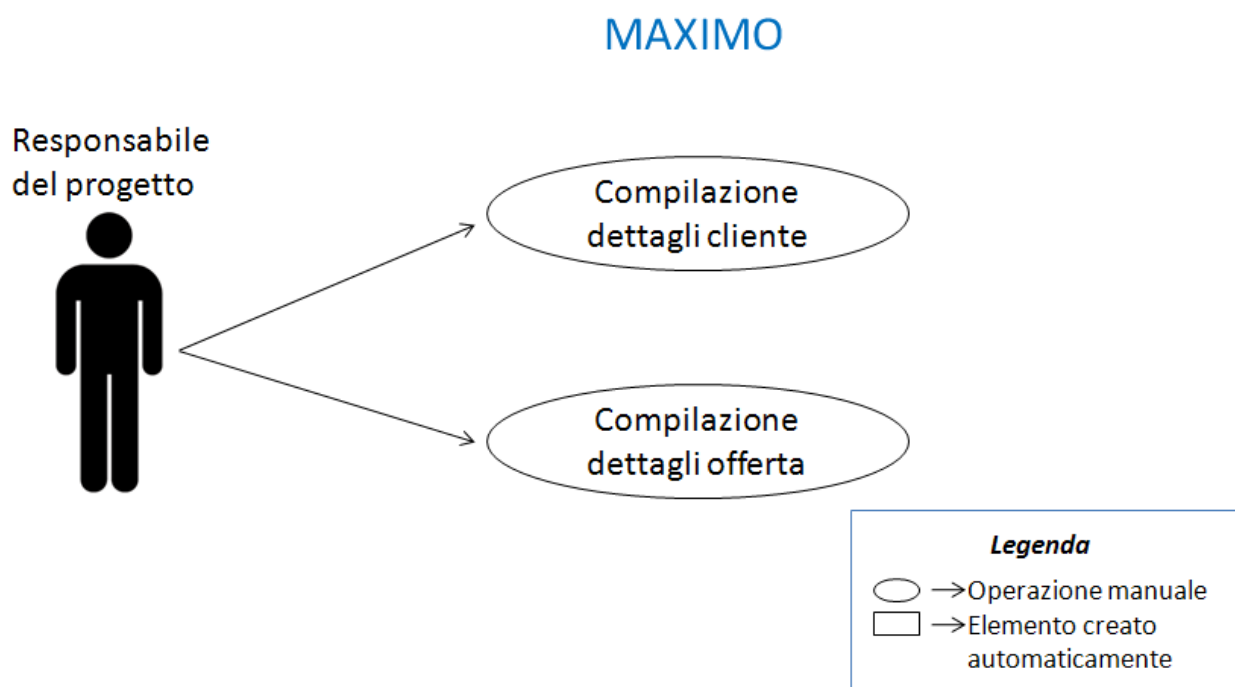
[Figura 5.5: creazione cliente e progetto da parte del responsabile commerciale]



In Tempestive il responsabile commerciale, non appena viene accettata un'offerta, provvede a censirla su Activity insieme con il relativo cliente. Si può provvedere anche alla modifica di un record esistente in caso di errori di inserimento, di cambio dei termini di contratto o di altri valori. Il tutto viene riportato automaticamente su Maximo secondo le seguenti dinamiche:

- l'inserimento del cliente comporta la creazione di un nuovo "customer" in Maximo;
- l'inserimento dell'offerta comporta la creazione di un nuovo "customer agreement" in Maximo;
- la modifica di eventuali campi di un record comporta la modifica su Maximo dei record legati al cliente e/o all'offerta.

Su Activity vengono inserite solamente alcune informazioni relative al cliente e all'offerta, mentre altri dettagli vengono successivamente inseriti manualmente all'interno delle applicazioni di Maximo da parte del responsabile del progetto.



[Figura 5.6: compilazione dettagli progetto e cliente da parte del responsabile del progetto]

Importante, come vedremo, sarà la compilazione del "customer agreement", ovvero la compilazione dell'offerta dalla quale dipenderà l'applicazione dei prezzi sulle attività di lavoro che verranno eseguite dai dipendenti.

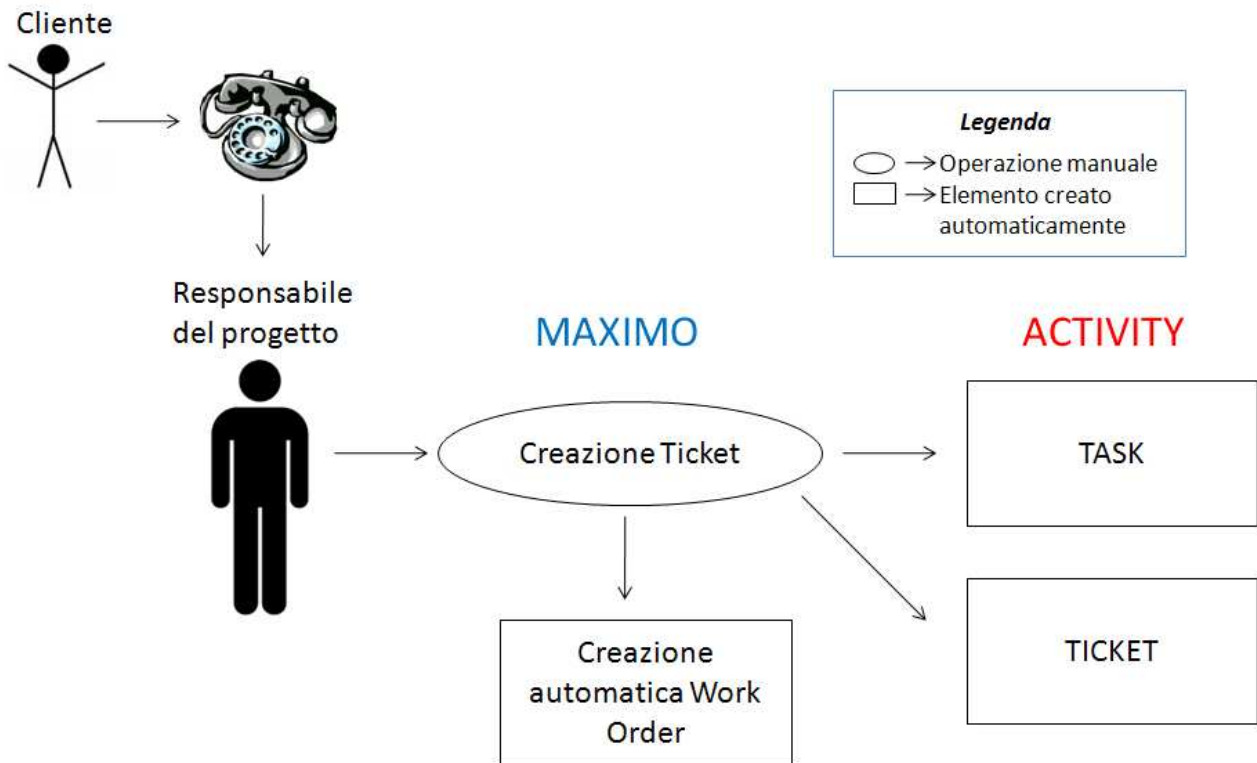
Dopo aver terminato la redazione del contratto e averlo approvato, le due tipologie di offerta prendono strade diverse all'interno del processo di integrazione. Vediamo qua di seguito tutti i passaggi eseguiti dal personale per ognuna.

### 5.5.2 Offerte per consulenze di vario genere

Un'attività di consulenza (considerate le decisioni prese a priori con Tempestive per il progetto corrente) parte sempre da un ticket di assistenza. Il ticket può essere redatto:

- dal cliente tramite Maximo (se si ipotizza un utilizzo di questo software anche da parte del cliente);

- dal responsabile del progetto: questo riceve una telefonata o una mail dal referente del cliente e, in base alle richieste, crea una nuova “service request” su Maximo.



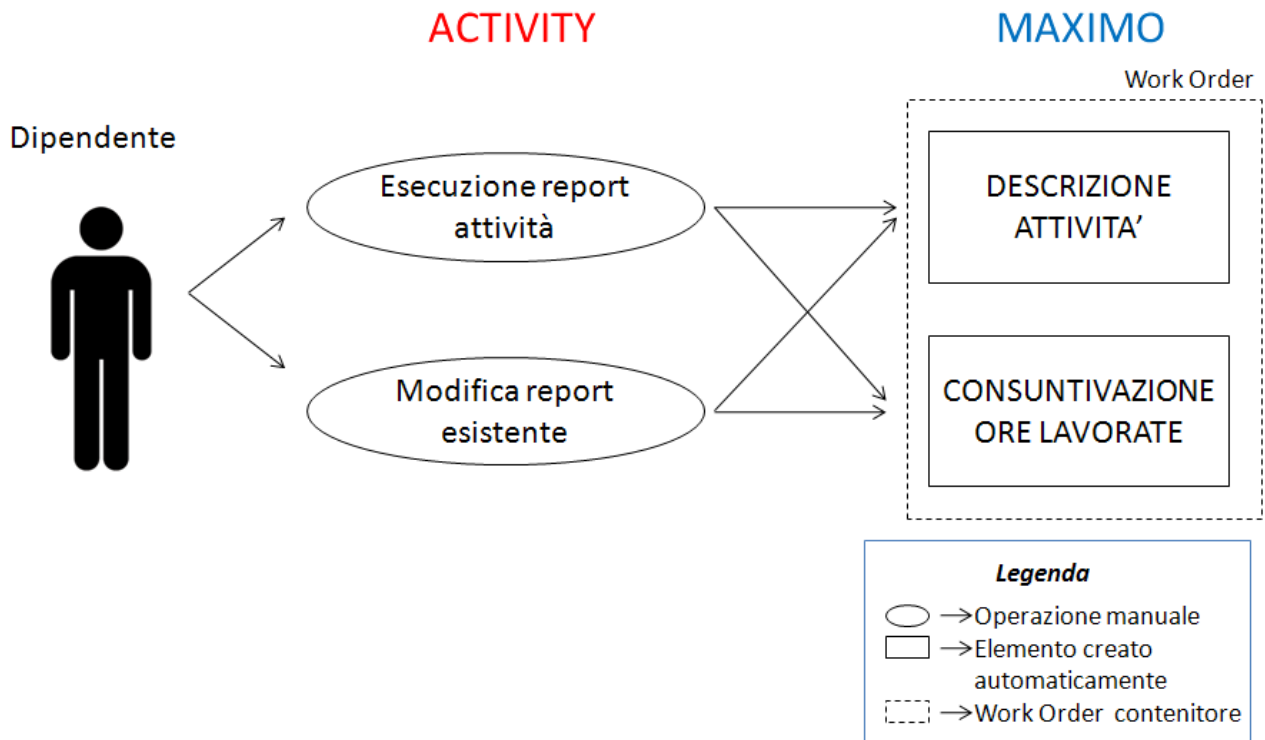
[Figura 5.7: creazione di un ticket di assistenza da parte del responsabile del progetto]

Una delle **novità** rispetto al precedente utilizzo di Activity da parte dell'azienda è che i task, ovvero le tipologie di attività di un'offerta, non dovranno essere più inserite manualmente, ma verranno importate automaticamente da Maximo ad Activity nel momento della creazione di un ticket.

Ciò non toglie il fatto che sia ancora possibile inserire dei task direttamente su Activity, ma chiaramente tutti i report eseguiti dai dipendenti su quei task non verranno fatturati da Maximo. Un esempio di un task di questo tipo è il classico task “Non Fatturabile” che spesso viene inserito dalla responsabile commerciale dell'azienda per permettere ai dipendenti di inserire rapporti di attività che non comportano alcun tipo di spesa per il cliente.

Come vediamo dalla figura 5.7, la creazione del ticket su Maximo comporta anche la creazione automatica di un ordine di lavoro legato ad esso e l'importazione di informazioni relative al ticket su Activity.

L'ordine di lavoro che si è venuto a creare sarà il destinatario delle consuntivazioni di tutti i lavori eseguiti dai dipendenti per il relativo ticket.



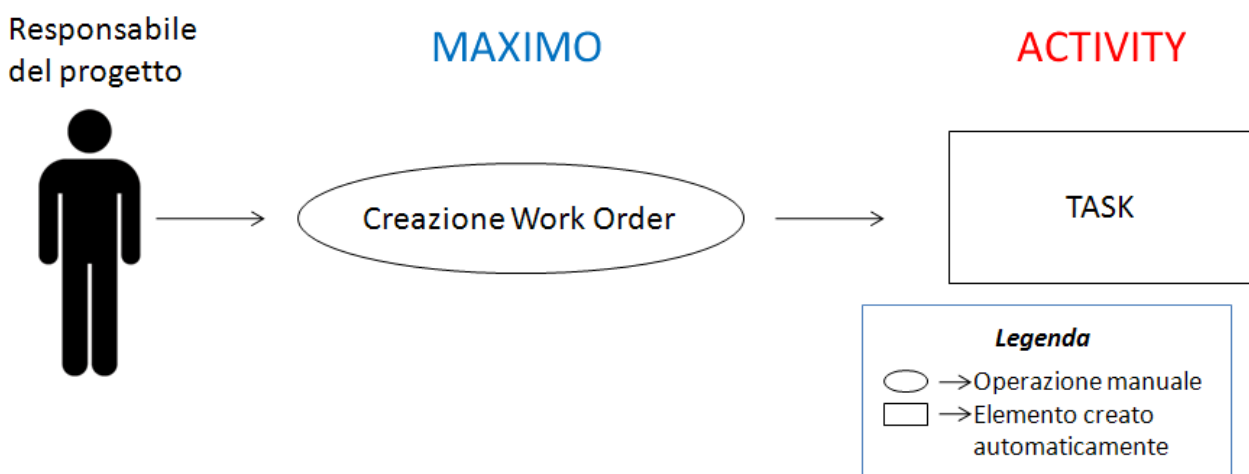
[Figura 5.8: compilazione di un report da parte di un dipendente – caso di offerta di consulenza]

Il dipendente esegue un report sull'attività svolta. Questo rapporto, una volta salvato, viene trasmesso a Maximo sottoforma di descrizione e consuntivazione di risorse, le quali poi verranno fatturate a fine mese.

Il dipendente può anche modificare un report esistente: le modifiche verranno automaticamente trasmesse a Maximo.

### 5.5.3 Progetti a corpo

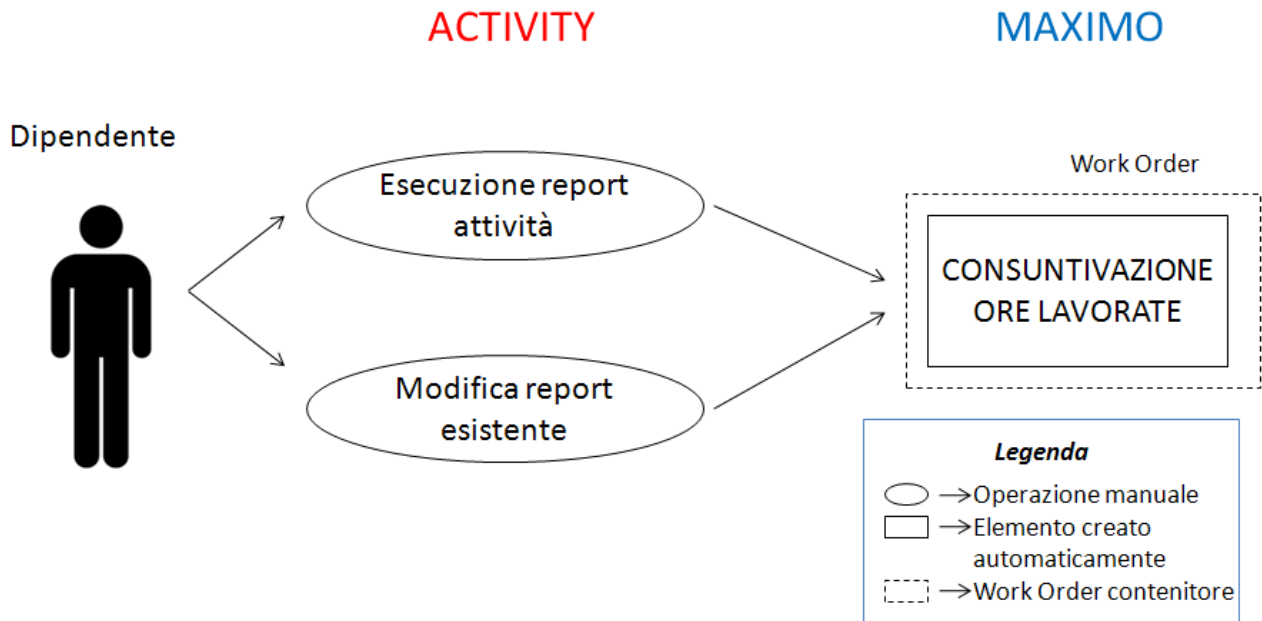
In un progetto a corpo, dopo la compilazione dell'offerta, la parte più importante è rappresentata dalla creazione, da parte del relativo responsabile, dell'ordine di lavoro che conterrà le fasi progettuali.



[Figura 5.9: creazione dell'ordine di lavoro contenente le fasi del progetto a corpo da parte del relativo responsabile]

Queste fasi sono rappresentate dai task del “work order” e vengono automaticamente importate su Activity come task dell’offerta di riferimento<sup>11</sup>. Come notiamo, anche qui è stato implementato un automatismo nella creazione dei task su Activity, senza che questi vengano inseriti manualmente.

Un dipendente è quindi in grado di eseguire i propri report sul task desiderato



[Figura 5.10: compilazione di un report da parte di un dipendente – caso di progetto a corpo]

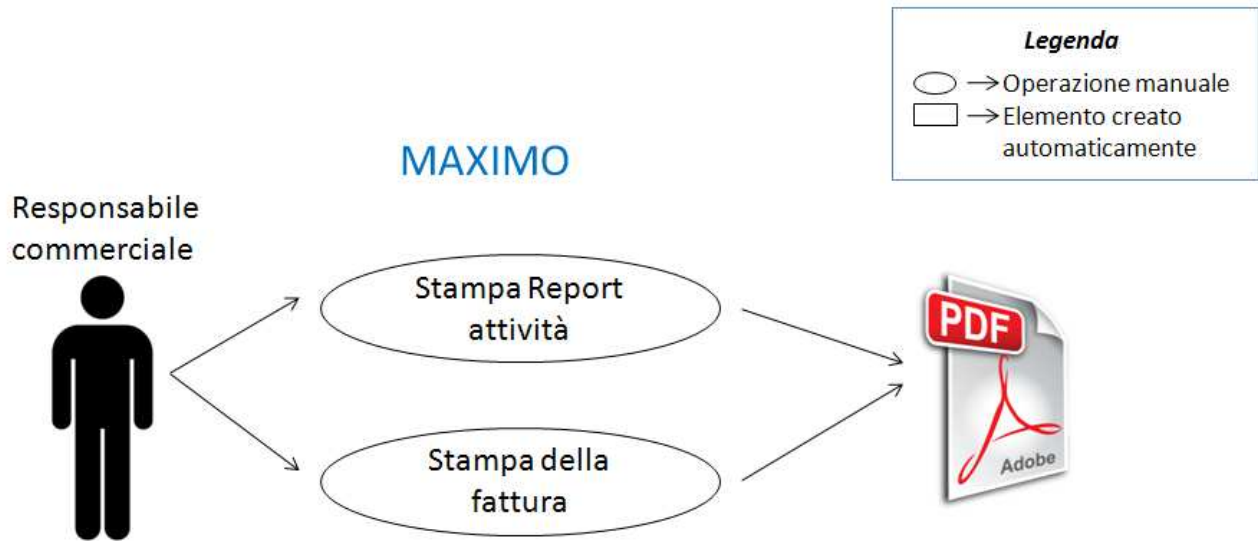
In questo caso viene riportata su Maximo solo la consuntivazione delle ore lavorate, riferite ad un ben determinato task del “work order” creato in precedenza (la descrizione dell’attività svolta rimane su Activity, ed in seguito vedremo il perché di ciò).

#### 5.5.4 Fatturazione

Altra **novità** rispetto a prima è la possibilità di stampa della reportistica senza l’utilizzo combinato di Microsoft InfoPath e Excel.

In entrambe le tipologie di offerta, infatti, viene creato in automatico (o manualmente a seconda dei termini di pagamento) il “bill batch” relativo alle attività svolte. Tramite questo, la responsabile commerciale è in grado di stampare il report delle attività e/o la fattura per il cliente con qualche semplice click.

<sup>11</sup> I task di un’offerta su Activity non sempre rappresentano i task del relativo “Work order” su Maximo: come vedremo in seguito, questa equivalenza varrà nel caso dei progetti a corpo, ma non in caso di un’offerta di consulenza.



[Figura 5.11: stampa di un report da parte del responsabile commerciale]

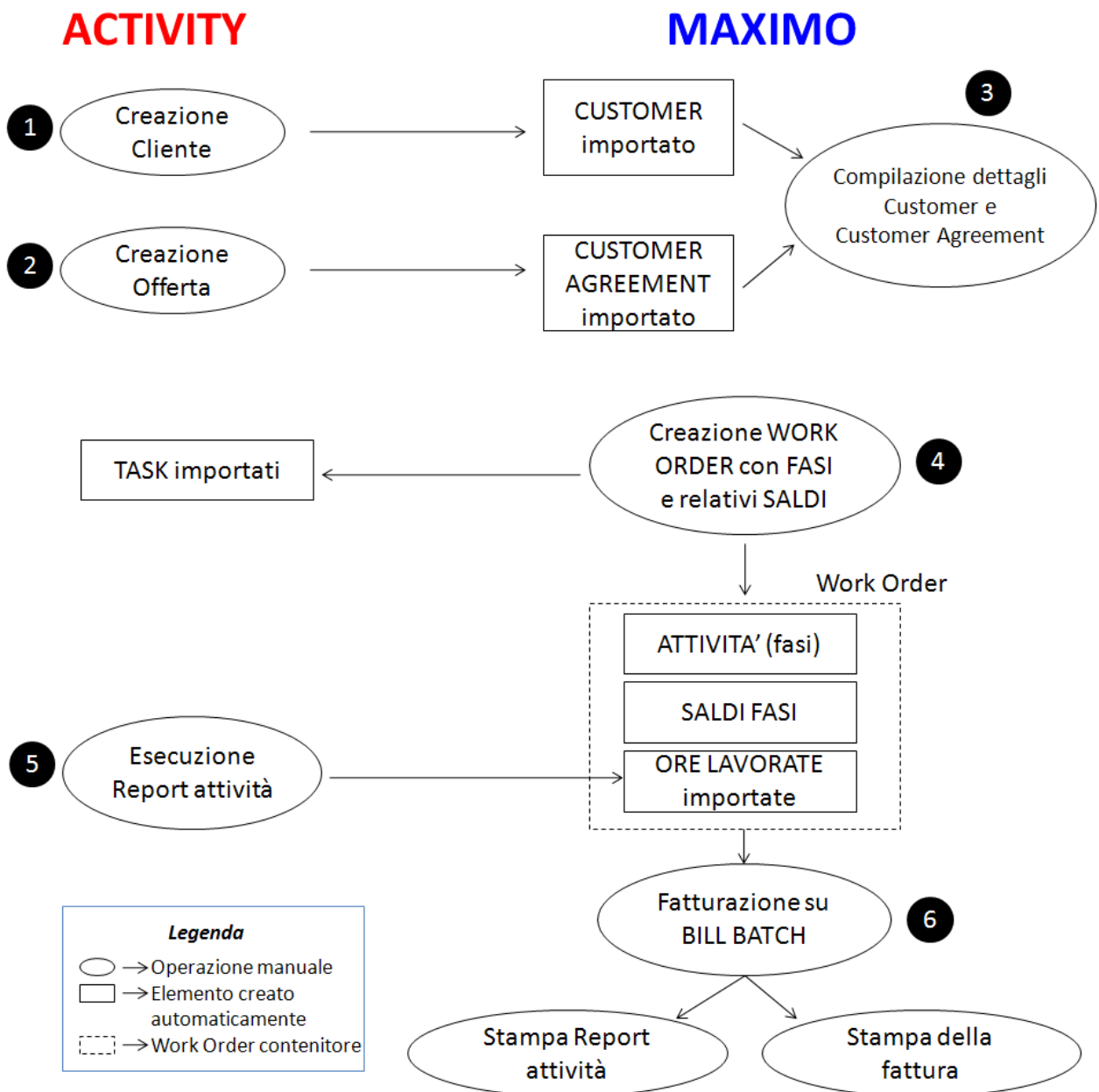
## 5.6 Schemi di funzionamento

A questo punto riassumiamo tutti i passaggi precedenti in 2 schemi, uno per ogni tipologia di offerta. In essi è evidenziata anche la sequenza cronologica con il quale avvengono i passaggi (tramite i numeri su sfondo nero). Per la comprensione dei singoli elementi, si faccia riferimento alla legenda in basso a sinistra nelle figure.



- l'evento provoca l'importazione su Activity delle informazioni relative e la creazione automatica di un task e di un ticket;
  - l'evento provoca la creazione automatica di un "work order" su Maximo con applicato il "price schedule" corretto;
- 5) Con l'avvenire degli interventi, su Activity vengono redatti i report relativi alle attività svolte
- l'evento provoca la consuntivazione su Maximo: vengono importate la descrizione dell'attività svolta e il numero di ore dedicate ad essa;
  - l'evento provoca a fine mese la raccolta di tutte le attività nel "bill batch" mensile (creatosi *automaticamente*) per la fatturazione;
- 6) Vengono stampati il rapporto sulle attività svolte e la fattura per il cliente.

### 5.6.2 Progetto a corpo



[Figura 5.13: schema progettuale per progetti a corpo]

Descriviamo la sequenza degli avvenimenti rappresentata in figura 5.13:

- 1) Come figura 5.12;
- 2) Come figura 5.12;
- 3) Come figura 5.12;
- 4) Creazione manuale del “work order” relativo al progetto: questo prevede manualmente l’impostazione del “price schedule” corretto, la creazione delle fasi delle attività e degli eventuali corrispettivi (saldi) che il cliente dovrà pagare per ogni fase portata a termine
  - l’evento provoca l’importazione su Activity delle informazioni relative alle fasi del progetto per la creazione dei task (viene creato un task per ogni fase);
- 5) Con l’avvenire degli interventi, su Activity vengono redatti i report relativi alle attività svolte
  - l’evento provoca la consuntivazione su Maximo: vengono importate le ore impiegate per lo svolgimento delle attività;
- 6) Al termine di ogni fase, viene creato su Maximo *manualmente* il “bill batch” dove fatturare la fase appena terminata;
- 7) Vengono stampati il rapporto sulle attività svolte e la fattura per il cliente

## 5.7 Importazione di dati su Maximo: il MIF

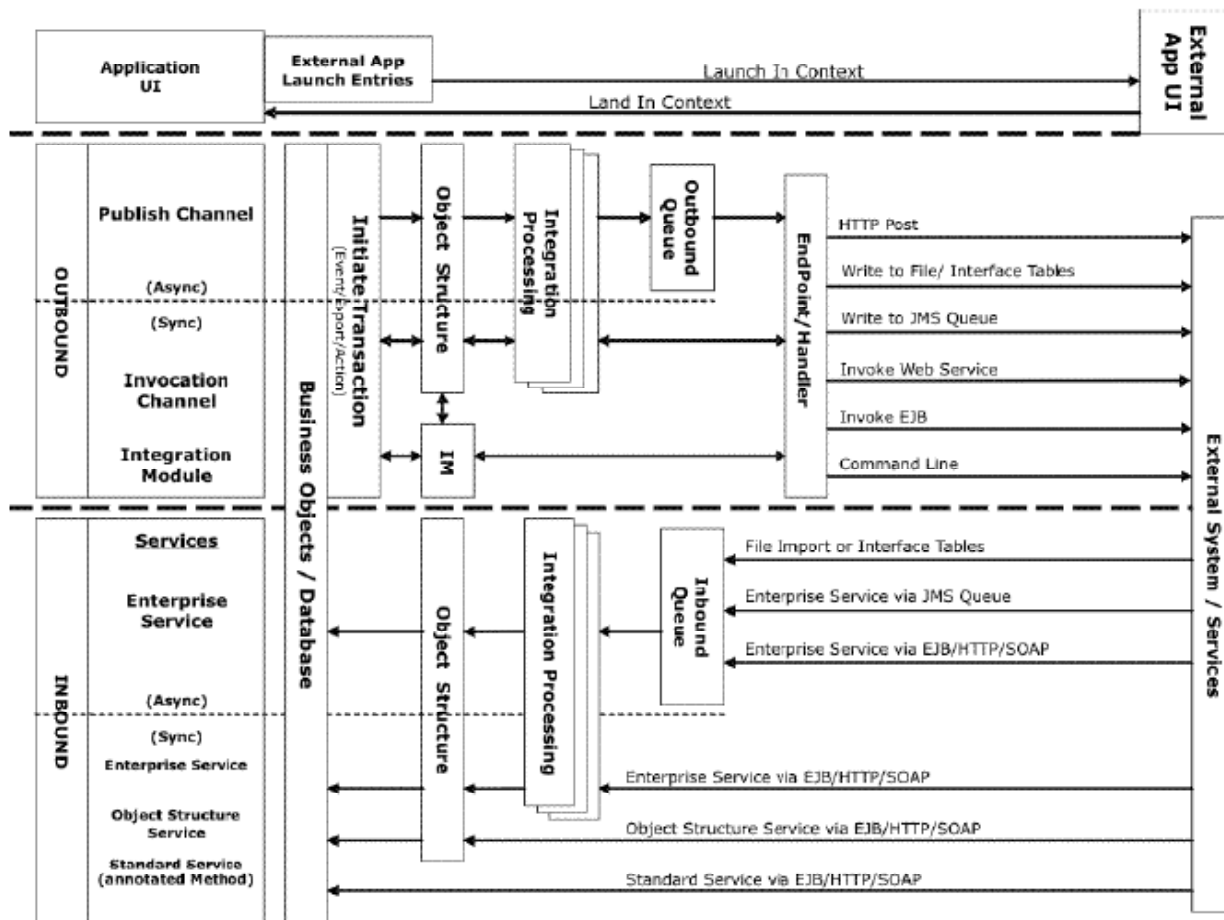
### 5.7.1 Architettura

Prima di iniziare a descrivere l’implementazione del progetto corrente, è bene soffermarsi brevemente sulle modalità con cui i dati vengono importati da un sistema esterno su IBM Maximo. Esso infatti è dotato di un modulo di integrazione che permette di connettersi e sincronizzare dati con ogni tipo di soluzione di terze parti. Con uno strumento di questo tipo, Maximo può godere dei seguenti vantaggi:

- non è necessario scrivere codice aggiuntivo per implementare la comunicazione fra i sistemi;
- molte integrazioni possono essere realizzate con solo una piccola quantità di codice JavaScript aggiuntivo;
- comprende più di due dozzine di connettori di comunicazione predefiniti per gestire l’I/O;
- la comunicazione fra i connettori può essere settata con una semplice finestra di comunicazione;
- i connettori supportano diversi tipi di protocolli di comunicazione e sorgenti di dati, fra cui: Java Database Connectivity (JDBC), Lightweight Directory Access Protocol (LDAP), Remedy, HiperText Transfer Protocol (http), Really Simple Syndication (RSS), International Development Markup Language (IDML), e molti altri.

Dal punto di vista dell’interfaccia utente, il set di applicazioni che permette di integrare il proprio sistema con l’esterno è definito come MIF (Maximo Integration Framework). Attraverso di esso, l’utente può definire dei veri e propri “workflow” all’interno del proprio framework.





[Figura 5.14: l'architettura del MIF]

L'architettura del MIF, mostrata in figura 5.14, mostra come le informazioni passano da Maximo ad un sistema esterno (e viceversa) attraversando diversi componenti applicativi e utilizzando diversi protocolli; la comunicazione bidirezionale può avvenire sia in tempo reale che in modalità batch<sup>12</sup>.

Sarebbe interessante uno studio approfondito di tutta la struttura del MIF, ma ai fini della presente relazione analizziamo solamente gli elementi più importanti e di cui faremo uso all'interno del progetto di integrazione.

- **Object structure:** è il livello di dati comune usato per tutti i processi di ingresso e uscita. Rappresenta uno o più oggetti di business e permette di
  - fornire o domandare dati;
  - inserire, aggiornare, cancellare dati;
  - eseguire l'import e l'export di dati.
- **Publish channel:** è il canale attraverso il quale i dati vengono trasmessi asincronamente dal sistema alle applicazioni esterne;
- **Enterprise service:** è il canale attraverso il quale i dati vengono importati dalle applicazioni esterne al sistema interno (utilizza sia chiamate sincrone che asincrone);
- **End points:** gli end point e i loro gestori facilitano il passaggio del flusso di dati in uscita verso la relativa destinazione. I dati vengono prima inseriti in una cosa JMS e poi inviati al sistema esterno. Contengono i protocolli e i dati necessari a identificare le specifiche di comunicazione con le varie destinazioni. Alcuni end point predefiniti sono:

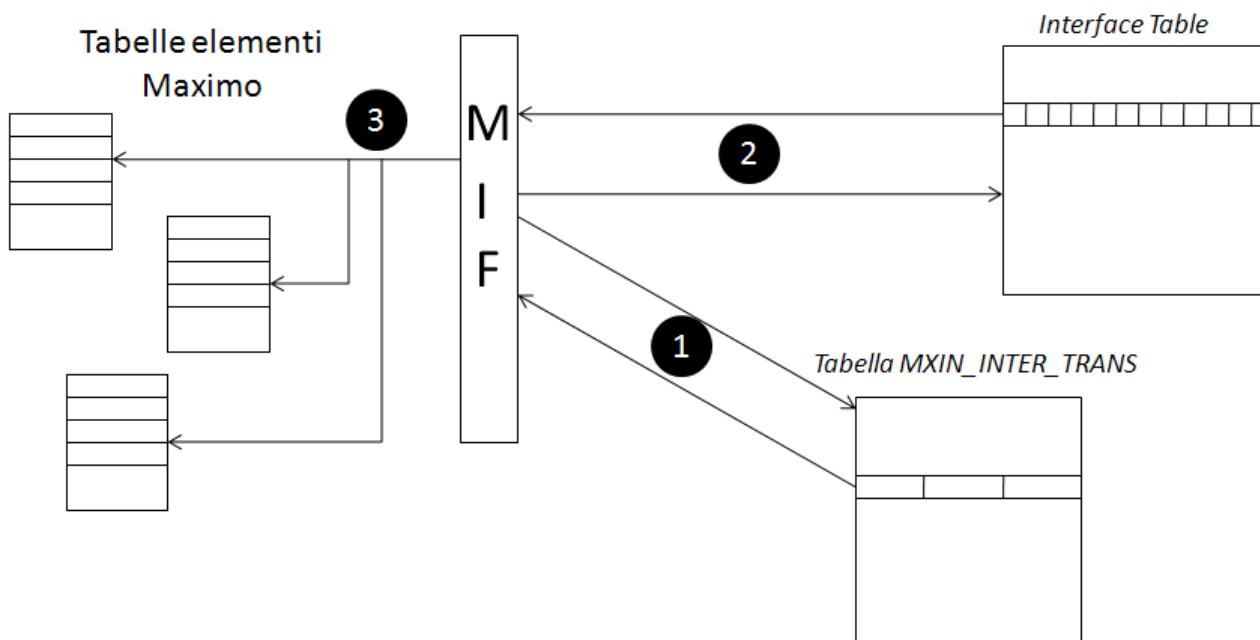
<sup>12</sup> La "batch mode", o modalità a lotti, prevede l'accumulo in memoria delle informazioni elaborate per poi trasmetterle tutte insieme. Al contrario, nella modalità "real time" l'elaborazione e la trasmissione procedono parallelamente.

- MXFLATFILE: crea file di dati;
- MXIFACETABLE: crea transazioni sulle locali “interface table” (che saranno oggetto di utilizzo nel progetto corrente);
- MXXMLFILE: crea file XML.
- *External system*: rappresenta la singola applicazione esterna che scambia dati con il sistema. Deve essere sincronizzata con un “end point” e, in base alla direzione del flusso di dati, con un “enterprise service” o con un “publish channel” o con entrambi.

### 5.7.2 Trasmissione dei dati tramite *interface table*

Essendo che la comunicazione fra le due parti dell’integrazione avviene utilizzando i relativi database, su Maximo si è pensato di utilizzare il metodo delle *Interface tables* implementato dal MIF. Una “interface table” è una tabella del database di Maximo creata sulla base di un oggetto struttura (“Object structure”), che ne definisca i campi e il legame con le altre tabelle.

Vediamo come avviene l’inserimento di un record in una tabella di questo tipo e il successivo import di dati all’interno delle tabelle di Maximo.



[Figura 5.15: step eseguiti dal MIF per importare dati attraverso le “interface table”]

Il sistema esterno crea un record di dati in una “interface table” (tramite una query di inserimento) e inoltre inserisce un record supplementare nella tabella “MXIN\_INTER\_TRANS”. E’ attraverso quest’ultimo record che Maximo, che periodicamente controlla questa tabella, va a prendere i dati contenuti nelle “interface tables”.

Un record della tabella MXIN\_INTER\_TRANS è strutturato in questo modo:

TRANSID	EXTSYSNAME	IFACENAME
---------	------------	-----------

L’identificatore TRANSID identifica la transizione: tutti i record aventi questo “id” verranno processati contemporaneamente. Invece IFACENAME e EXTSYSNAME sono rispettivamente il nome della “interface table” e dell’“external system” collegato con l’esterno.

Quando Maximo controlla questa tabella, il MIF esegue i seguenti passi (riferimento figura 5.15):

- 1) prende i valori del record contenuto in essa;
- 2) in base al campo TRANSID, accede alla tabella IFACENAME e prende tutti i record aventi quell'identificatore;
- 3) processa i valori ottenuti da quei record e li inserisce nei campi giusti delle tabelle giuste di Maximo (utilizzando l'“external system” EXTSYSNAME) con la formattazione adeguata (basata su quanto definito nell' “object structure”)

In questo modo si ottengono inserimenti, aggiornamenti e cancellazioni sicuri sulle tabelle di Maximo.

## 5.8 Implementazione

### 5.8.1 Descrizione generale

Giungiamo quindi alla descrizione dei 5 processi Java che rappresentano il motore del progetto di integrazione. Essi sono stati inseriti e fatti girare in background nel *Computer 2* (quello su cui è installato Activity) attraverso il *Task Scheduler* di Windows; vengono avviati ogni minuto a distanza di 10 secondi l'uno dall'altro.

Ogni processo è rappresentato da una classe Java ben precisa che esegue dei controlli sui uno dei due (o entrambi) database e, in base ad alcune regole, esegue delle ben precise operazioni.

La connessione ai database viene instaurata attraverso i driver JDBC: ogni classe Java presenta inizialmente le seguenti righe di codice, per esempio, per DB2:

1. `//Definizione Driver e connessione al database DB2`
2. `Class.forName ("com.ibm.db2.jcc.DB2Driver");`
3. `Connection db2Conn = DriverManager.getConnection ("jdbc:db2j:net://10.1.2.102:50000/`
4. `maxdb75","userDB2","passwordDB2");`

Chiaramente, “10.1.2.102” rappresenta l'indirizzo IP del *Computer 1* all'interno della rete aziendale: attraverso la porta 50.000 tutti i processi potranno connettersi ai dati contenuti nel database “maxdb75” di Maximo con utente “userDB2” e password “passwordDB2”.

Al database di Activity, invece, ci si conatterà semplicemente in locale attraverso la porta 1434.

Vediamo le operazioni che svolgono i 5 processi all'interno dell'integrazione:

1. **Clients:** sincronizza i clienti fra le due parti e controlla eventuali modifiche;
2. **Progetti:** sincronizza le offerte fra le due parti e controlla eventuali modifiche;
3. **Task\_Corpo:** controlla specificatamente le attività dei progetti a corpo su Maximo e le sincronizza con i task di Activity, controllando anche eventuali modifiche apportate;
4. **Report:** controlla se vi sono nuovi report su Activity, nel qual caso esegue la relativa consuntivazione su Maximo. Esegue anche un controllo di eventuali modifiche;
5. **WorkOrder:** controlla lo stato degli ordini di lavoro su Maximo per eseguire azioni diverse.

Nei prossimi paragrafi vedremo come questi processi si inseriscono all'interno delle dinamiche descritte nelle figure 5.12 e 5.13.

Inoltre nel progetto era compresa anche qualche piccola aggiunta da apportare all'applicazione Activity per renderla compatibile con il ciclo di fatturazione di Maximo e con i processi Java. Erano previste però *solamente* aggiunte di campi, tabelle o elementi dell'interfaccia grafica, e **nessuna** modifica al codice o agli elementi già presenti. Anche questi arricchimenti verranno descritti man mano nei paragrafi che seguono.

### 5.8.2 Clienti e progetti

La prima cosa da fare per quest'implementazione era sicuramente pensare alla sincronizzazione dei clienti e delle offerte da entrambe le parti dell'integrazione. Entrambe le offerte partono con l'inserimento in Activity del cliente e dell'offerta relativa.

Tempestive s.r.l. procede al censimento di questi due elementi utilizzando degli identificatori, univoci per ogni cliente e per ogni offerta.

Per ottenere questi due valori, la responsabile commerciale dell'azienda procede come segue:

- **Cliente:** identificativo formato dalle prime 3 lettere consonanti della ragione sociale. In caso di omonimie si procede con la selezione della consonante successiva (prima per l'ultima lettera, poi per la seconda e poi per la prima) fino ad ottenere una tripla di caratteri univoca;
- **Progetto:** identificativo formato dalla lettera "O" più l'identificativo del cliente e più la data di stipula del contratto nella forma "anno-mese-giorno". Casi di omonimie non ce ne possono essere poiché l'identificativo del cliente è unico e non è prevista dall'azienda la firma di più di un'offerta in un giorno per lo stesso cliente.

Un esempio di questi 2 approcci può essere il seguente:

*Rag. Sociale cliente:* Parmalat s.p.a.  
*Data di firma dell'offerta:* 22/5/2012  
*ID cliente:* **PRM**  
*ID offerta:* **OPRM20120522**

In Activity l'identificatore di un record viene creato con "NEWID()", una funzione "Transact-SQL" che permette di creare un valore di tipo "uniqueidentifier"<sup>13</sup>; questo viene rappresentato come una stringa di 24 caratteri (ad esempio, la stringa "6F9619FF-8B86-D011-B42D-00C04FC964FF").

In Maximo, invece, ogni tabella ha un identificatore rappresentato semplicemente da un numero naturale progressivo. Alcune tabelle però hanno anche degli altri campi i cui valori devono essere univoci fra tutti i record memorizzati.

Per l'integrazione si è pensato proprio di utilizzare uno di questi ultimi campi su Maximo e aggiungere un campo in più alla tabella relativa ai clienti e alle offerte su Activity<sup>14</sup>, nel quale aggiungere proprio l'identificativo usato da Tempestive s.r.l., sia per i clienti che per le offerte. In questo modo avremo un campo in comune per ogni record su entrambi i database.

Questi identificatori sono calcolati direttamente dal processo **Clienti** e dal processo **Progetti** prendendo i dati direttamente dal database di Activity. Ad esempio il seguente pezzo di codice comporta la creazione dell'identificatore relativo all'offerta.

1. `String ragSoc = rs1.getString(3);`
2. `String agreement = "";`
3. `Calendar calendar = Calendar.getInstance();`
4. `calendar.setTime(rs1.getDate(4));`
5. `SimpleDateFormat date = new SimpleDateFormat("yyyyMMdd");`
6. `String formattedDate = date.format(calendar.getTime());`
7. `agreement = "O" + ragSoc + formattedDate;`

---

<sup>13</sup> Il tipo di dati "uniqueidentifier" consente di archiviare valori binari a 16 byte che fungono da identificatori univoci globali (GUID). La funzione NEWID di Transact-SQL e le funzioni o i metodi API dell'applicazione generano nuovi valori "uniqueidentifier" a partire dal numero di identificazione della scheda di rete e dal numero univoco del clock della CPU.

<sup>14</sup> Si era pensato inizialmente di utilizzare il campo identificatore dei record su Activity come identificatore anche su Maximo, ma la lunghezza del primo era maggiore del numero massimo di caratteri che poteva supportare il secondo.

8. //Inserimento id di Maximo nei Report
9. `String idMax="UPDATE ACTIVITY.ACTIVITY.Category2 SET Agreement="" +  
agreement + " WHERE IdCat2=" + rs1.getString(1) + """;`
10. `sqlst.executeUpdate(idMax);`

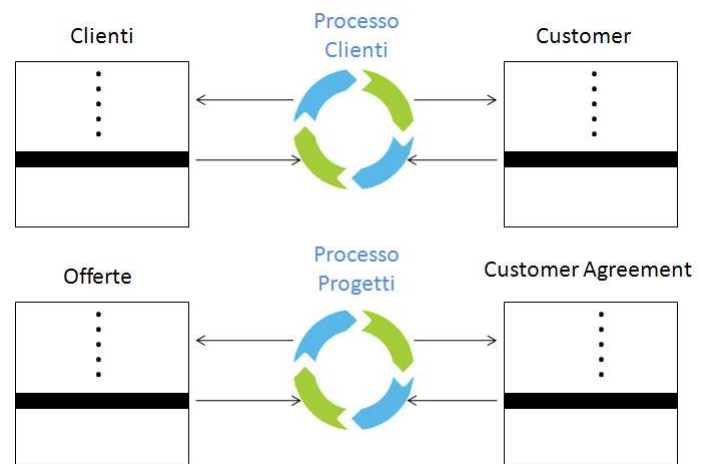
Come notiamo, il valore finale (calcolato alla riga 7) è ottenuto come concatenazione di 3 stringhe. La ragione sociale del cliente e la data di firma dell’offerta sono stati ottenuti eseguendo una query sulle tabelle di Activity e memorizzando i dati nel “result set” *rs1*. In particolare, la data ottenuta è stata formattata nella maniera richiesta (riga 5) per poi essere inserita in una stringa (riga 6). Alla fine l’identificatore ottenuto è stato inserito nel nuovo campo (*Agreement*) della tabella della offerte di Activity (riga 10).

A questo punto definiamo come avviene il processo di controllo dei record e di sincronizzazione in presenza di nuovi record. Il processo **Clients** esegue i seguenti passi:

- 1) accede al database di Activity e copia su un array tutti gli identificatori (quelli condivisi con Maximo) e su un secondo array la relativa descrizione del cliente;
- 2) per ogni cliente ricerca il medesimo su Maximo attraverso una query;
- 3) se il record viene trovato, viene controllata la corrispondenza delle descrizioni e, in caso di modifiche avvenute su Activity, viene aggiornato Maximo;
- 4) se il record non viene trovato, viene creato il corrispondente “customer” su Maximo.

Quando viene creato automaticamente un cliente su Maximo, vengono solo importati l’identificatore e il nome per esteso del cliente. Il resto dei dati viene inserito direttamente in Maximo dalla persona addetta (il responsabile del progetto).

Il processo **Progetti** funziona essenzialmente allo stesso modo del precedente. I passi che esegue sono gli stessi, ma con la differenza che vengono sincronizzati l’identificatore, la descrizione, la data di partenza e di termine dell’offerta. Chiaramente anche per ognuno di questi campi è previsto un controllo con conseguente aggiornamento su Maximo in caso di modifica su Activity.



### 5.8.3 WorkOrder: gestione degli ordini di lavoro per offerte diverse

Come già spiegato precedentemente, dopo l’approvazione dell’offerta si distinguono due strade per le due tipologie di offerta, anche se entrambe portano alla creazione di un ordine di lavoro:

- in un *progetto di consulenza* viene redatto un ticket che successivamente crea in automatico un “work order”
- in un *progetto a corpo* un “work order” viene creato manualmente dal responsabile del progetto per definire le fasi dei lavori.

Quello che ancora non è stato menzionato, è che per le offerte di consulenza anche l’applicazione del “price schedule” è automatica (al contrario dell’altra tipologia in cui viene fatto manualmente), e questo grazie ad un’opportuna “escalation” configurata precedentemente e schedulata opportunamente da Maximo. Un “escalation” è un sistema di monitoraggio automatico dei record che permette di eseguire delle azioni predefinite quando un record incontra le condizioni

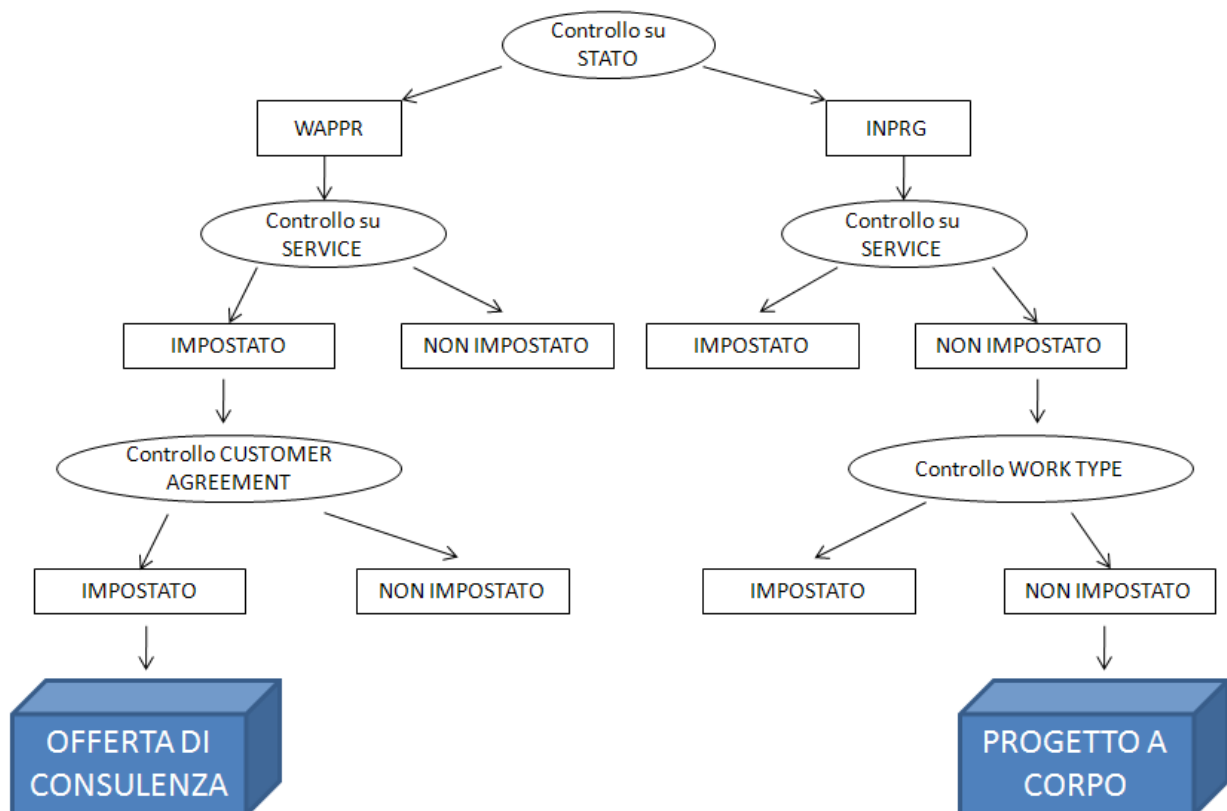
stabilite nella “escalation” stessa. Nel nostro caso, era stato imposto che venisse applicato il contratto giusto ad ogni nuovo “work order”. L’escalation quindi periodicamente controlla tutti i record degli ordini di lavoro e se ne trova uno senza offerta impostata, vi allega l’“agreement” giusto con il “price schedule” corretto secondo le condizioni di applicabilità ivi impostate.

A questo punto nella tabella degli ordini di lavoro saranno presenti 4 tipi di record:

- ordini di lavoro per offerte di consulenza appena creati;
- ordini di lavoro di progetti a corpo appena creati;
- ordini di lavoro attivi e in corso d’opera;
- ordini di lavoro terminati (che devono essere ancora fatturati o che rappresentano semplicemente dei record di storico).

Il processo **WorkOrder** entra in gioco proprio qui: esso esegue alcuni controlli su ogni record, capisce di che record si tratta ed esegue o meno alcune azioni.

Per comprendere di che tipologia sono i record che analizza, il processo controlla il valore di alcuni campi. Lo schema sottostante mostra come esso sia in grado di capire quando si trova di fronte ad un ordine nuovo (non ancora processato) e di che offerta si tratti.



[Figura 5.16: sequenza di controlli eseguiti per definire la tipologia di offerta]

I valori su cui basare i controlli vengono ricavati da un’interrogazione eseguita sulla tabella dei “work order”. Dopodiché si procede controllando lo stato del record: se è WAITING FOR APPROVAL, si procede controllando se è impostato un servizio di manutenzione, nel qual caso sappiamo già che si tratta di un ordine di lavoro riguardante un’offerta di consulenza. Rimane da controllare se l’“escalation” ha già visto il record e applicato il relativo “agreement”: se così, si inizierà a processare i dati da importare su Activity.

Nel caso, invece, lo stato sia IN PROGRESS, potremmo avere sia un ordine di lavoro già in corso d’opera (e quindi già processato), sia un ordine per un progetto a corpo ancora da processare.

Viene controllato quindi che il servizio di manutenzione non sia impostato (escludendo quindi che si tratti di un offerta di consulenza già avviata) sia che il campo riguardante il tipo di lavoro non sia stato ancora editato (per essere sicuri che l'ordine non sia stato ancora processato). Confermato ciò, siamo di fronte a un "work order" contenente le fasi di un progetto a corpo, che verrà quindi importato su Activity.

In pseudocodice, possiamo definire quanto detto in questo modo:

```

1. if status = WAPPR then
2.     if service ≠ NULL then
3.         if agreement ≠ NULL then
4.             ..
5.             //Offerta di consulenza
6.             ..
7. else
8.     if service is NULL then
9.         if workType is NULL then
10.            ..
11.            //Progetto a corpo
12.            ..
13. end

```

Chiaramente, questi sono i passaggi che ho deciso di utilizzare per distinguere le varie casistiche: niente avrebbe vietato di implementare dei controlli diversi per ottenere le stesse informazioni.

Una volta compreso di quale offerta si tratta, vanno modificati alcuni campi del record:

- *consulenza*: impostazione stato del record a IN PROGRESS e impostazione del *Work Type* in "CONS" (valore creato appositamente per il caso corrente);
- *progetto a corpo*: impostazione *Work Type* in "CORPO" (valore creato appositamente per il caso corrente).

Ciò viene fatto con una semplice query di modifica (sempre passando attraverso il MIF).

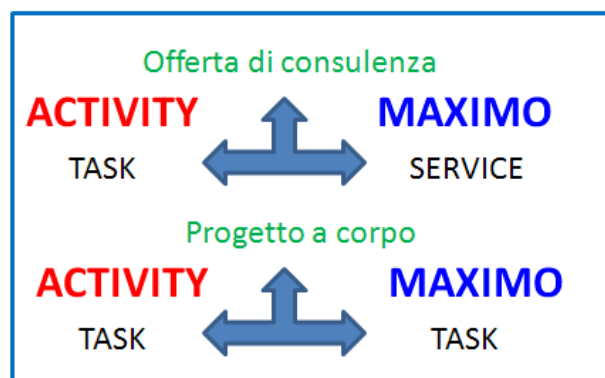
Quindi **WorkOrder** avvia l'importazione dei dati su Activity: questa avviene con essenzialmente le stesse modalità in entrambe le tipologie di offerta, ma le informazioni importate cambiano.

Prima di tutto vengono importate le informazioni per creare i task su Activity. A riguardo, dobbiamo spiegare la differenza fra i suddetti task e i task di un "work order" su Maximo.

In una consulenza, ciò che viene erogato è un servizio di un certo tipo in base alle richieste avanzate dal cliente. A questo proposito quello che su Maximo è il servizio offerto (*Service*) viene inserito come task su Activity e i task del "work order" di riferimento rappresenteranno le attività svolte per quel servizio (quindi la descrizione del report effettuato da un dipendente).

Un progetto a corpo, invece, è rappresentato dalle sue fasi, che sono inserite manualmente dal responsabile del progetto come task del

"work order" di riferimento. In questo caso, i task dell'ordine di lavoro rappresenteranno proprio i task del progetto su Activity.

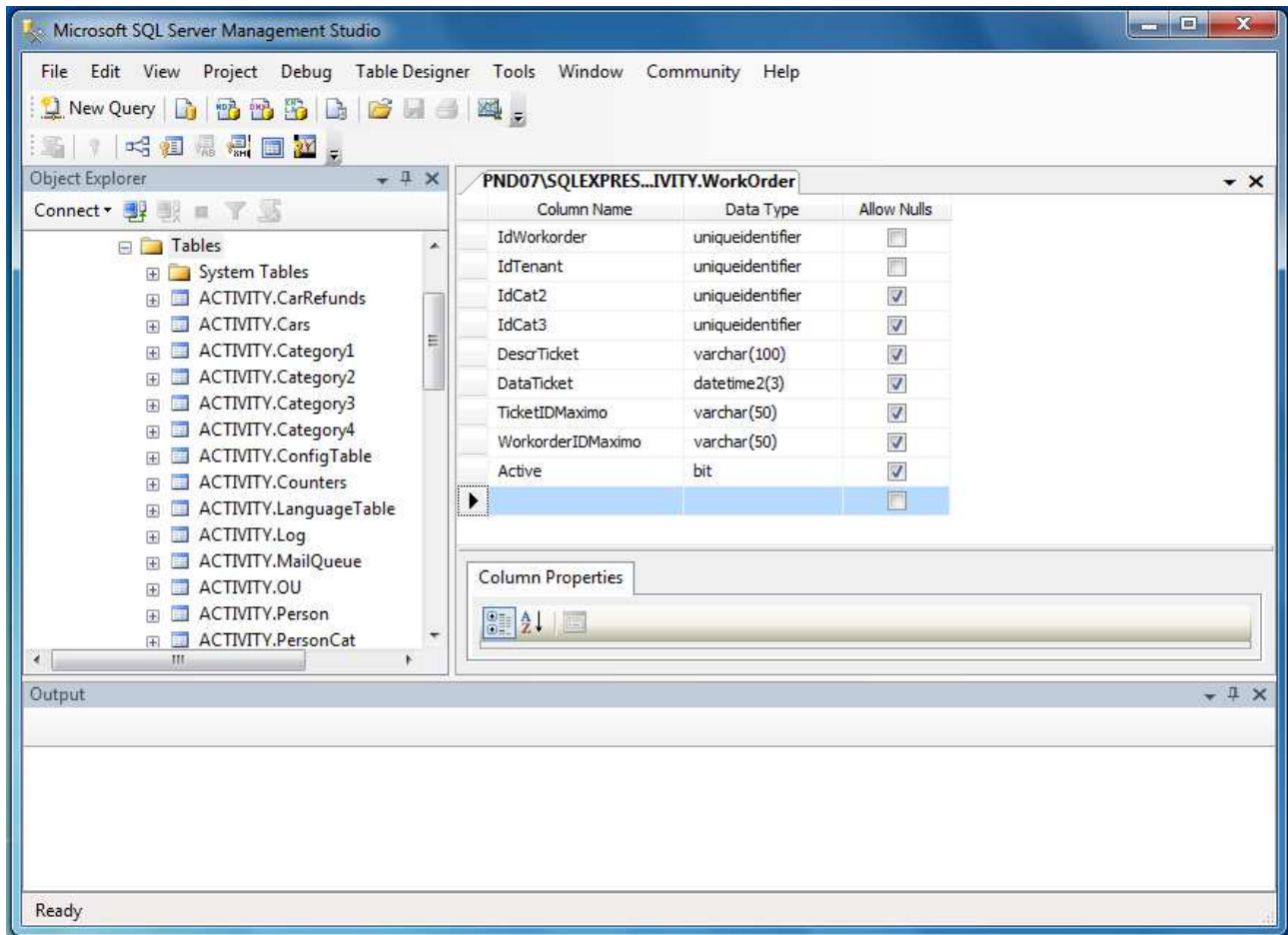


Importati i task si procede a importare le informazioni riguardanti l'ordine di lavoro e il ticket di assistenza, quest'ultimo solo nel caso di una consulenza.

Activity, chiaramente, non è stato creato per mantenere questo tipo di informazioni, ma le esigenze attuali richiedono proprio un mantenimento delle suddette per i seguenti requisiti progettuali:

- il dipendente deve poter essere in grado di selezionare il ticket di assistenza al quale le sue ore lavorative sono dedicate;
- deve essere possibile consuntivare le ore lavorate dei dipendenti sull'ordine di lavoro corretto per poter ottenere una fatturazione precisa e coerente.

Per questo è stata creata su Activity una nuova tabella, chiamata appunto “WorkOrder”.



[Figura 5.17: definizione della tabella “WorkOrder” sul database di Activity]

Questa tabella contiene un proprio id univoco (che ne fa da chiave primaria), gli id del progetto e del task relativo al ticket, informazioni sul ticket e l'id del “work order” su Maximo nel quale si andrà a consuntivare. In questo modo otteniamo una corrispondenza fra i ticket, i task e gli ordini di lavoro fra Activity e Maximo.

Ma come mai questa tabella è stata chiamata “WorkOrder” anziché “Ticket”? La risposta sta nella distinzione fra le due tipologie di offerta: con una consulenza vengono compilati tutti i campi della tabella, mentre con un progetto a corpo, non esistendo un ticket, tutti i campi riguardanti quest'ultimo vengono impostati a NULL e di conseguenza si ottiene solamente l'abbinamento fra il task e il relativo ordine di lavoro su Maximo.



Abbinato a questa tabella, è stato aggiunto un elemento all'interfaccia grafica dell'applicazione: una “drop-down list” dalla quale sia possibile selezionare il ticket per il quale si sta lavorando ed eseguire un report riferito ad esso. La lista contiene elementi del seguente tipo:

(id del ticket in Maximo – Data creazione ticket – Descrizione del problema)

es.: (1066 – 23/05/2012 – Blocco del server)

Questi dati vengo presi direttamente dalla tabella “WorkOrder”<sup>15</sup>. Chiaramente, nel caso il ticket venga chiuso, non sarà più selezionabile.

Distinguiamo quindi le due query di inserimento eseguite da **WorkOrder** sulla suddetta tabella con 2 esempi:

*Offerta di consulenza:*

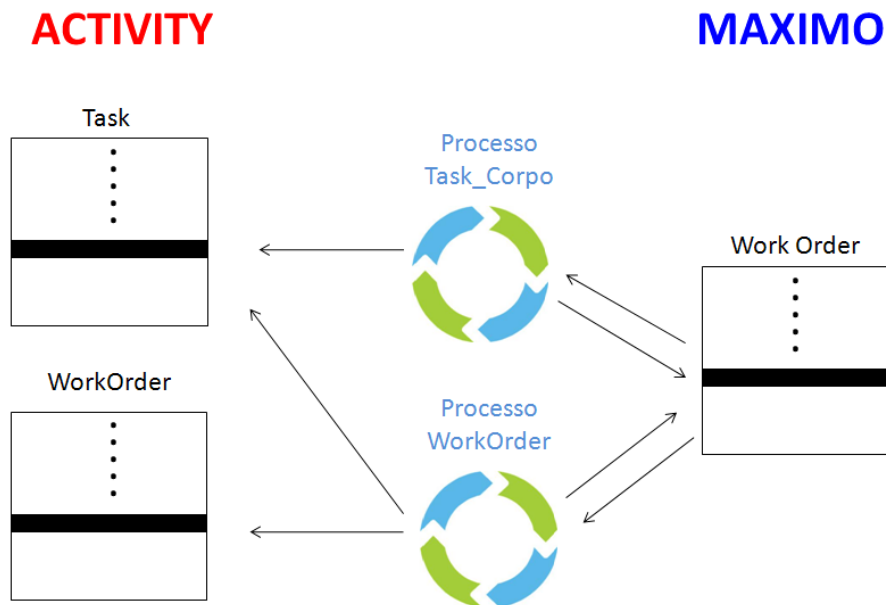
```
INSERT INTO ACTIVITY.WorkOrder values
(NEWID(), //Id univoco della tabella
'11111111-2222-3333-4444-555555555555', //Campo sempre uguale (campo inutilizzato)
'HYD5462T-89G6-8FDA-87FE-HDG67J425J7F', //Id offerta
'4GJ79FQW-34H2-3G32-H7DE-2HEKSU3KAU58', //Id Task
'Blocco del Server', //Descrizione del ticket
'2012-05-23', //Data del Ticket
'1066', //Id del ticket suMaximo
'1032', //Id del work order suMaximo
'1') //Definisce se il ticket è attivo (1) o chiuso (0)
```

*Progetto a corpo:*

```
INSERT INTO ACTIVITY.WorkOrder values
(NEWID() ,
'11111111-2222-3333-4444-555555555555',
'HYD5462T-89G6-8FDA-87FE-HDG67J425J7F',
'4GJ79FQW-34H2-3G32-H7DE-2HEKSU3KAU58',
NULL,
NULL,
NULL,
'1032',
'0')
```

Oltre al processo **WorkOrder**, vi è un altro processo che esegue un continuo controllo degli ordini di lavoro. **Task\_corpo** infatti si occupa esclusivamente degli ordini di lavoro già processati aventi “Work Type = CORPO” (ovvero gli ordini di lavoro dei progetti a corpo); questi hanno già i corrispettivi task memorizzati su Activity, ma nel caso in cui vengano eseguite delle modifiche su Maximo alla descrizione delle attività, il processo se ne accorge e aggiorna immediatamente i record dei task corrispondenti su Activity.

<sup>15</sup> Non ci soffermiamo sulla descrizione delle modalità di creazione di questo elemento grafico. Basti sapere che attraverso la creazione di una “vista” sul database si è definita la formattazione delle informazioni da mostrare e l'elemento grafico è stato creato utilizzando il software Tempestive dbWebPart.



[Figura 5.18: le attività dei processi “Task\_Corpo” e “WorkOrder”]

#### 5.8.4 Report e consuntivazione

Quando un dipendente esegue un report su Activity, vi pone la descrizione dell’attività svolta, la data, il progetto per cui sta lavorando, il numero di ore lavorate e altri valori. Ciò che però è importante ai fini della fatturazione sono proprio le ore che vengono consuntivate. Deve essere quindi presente un processo che si accorga della presenza di nuovi report su Activity e consuntivi le relative ore lavorate sull’ordine di lavoro corretto: questo è quello che fa **Report**.

Maximo non contiene nessun elemento simile ai report eseguiti su Activity: c’era bisogno di avere qualcosa che mantenesse traccia dei report processati e di eventuali modifiche apportate.

Si è pensato quindi di creare una tabella di supporto, sempre su Activity, con la quale confrontare i record della tabella dei report

IdWorkReport	Tenant	DescrWorkReport	DateWorkReport	TimeAmount
--------------	--------	-----------------	----------------	------------

I 4 campi creati sono gli stessi della tabella WorkReport (la tabella dei report): **Report** controlla la coincidenza degli id di entrambe le tabelle:

- se trova la coincidenza, controlla che anche gli altri campi siano uguali e, in caso contrario (cioè quando un report viene modificato), aggiorna la tabella di supporto e i valori importati su Maximo (quelli che verranno consuntivati nell’ordine di lavoro);
- se non trova la coincidenza, vuol dire che il report non è ancora stato processato: quindi crea il record sulla tabella di supporto e avvia la consuntivazione su Maximo.

Si è scelto di inserire queste quattro informazioni nella tabella di supporto poiché sono quelle che verranno anche modificate su Maximo (a parte il campo “Tenant” che è stato inserito per allinearsi con la struttura delle altre tabelle, nelle quali è sempre presente), mentre le altre rimangono su Activity e, anche se modificate, non producono alcun cambiamento ai fini della fatturazione.

Quando arriva il momento di consuntivare i dati del report su Maximo, avviene il solito controllo sull’offerta per comprendere di che tipologia si tratti. Anche se il procedimento di consuntivazione

è essenzialmente lo stesso, un report di una consulenza e uno per un progetto a corpo, come già detto in precedenza, si differenziano in quanto:

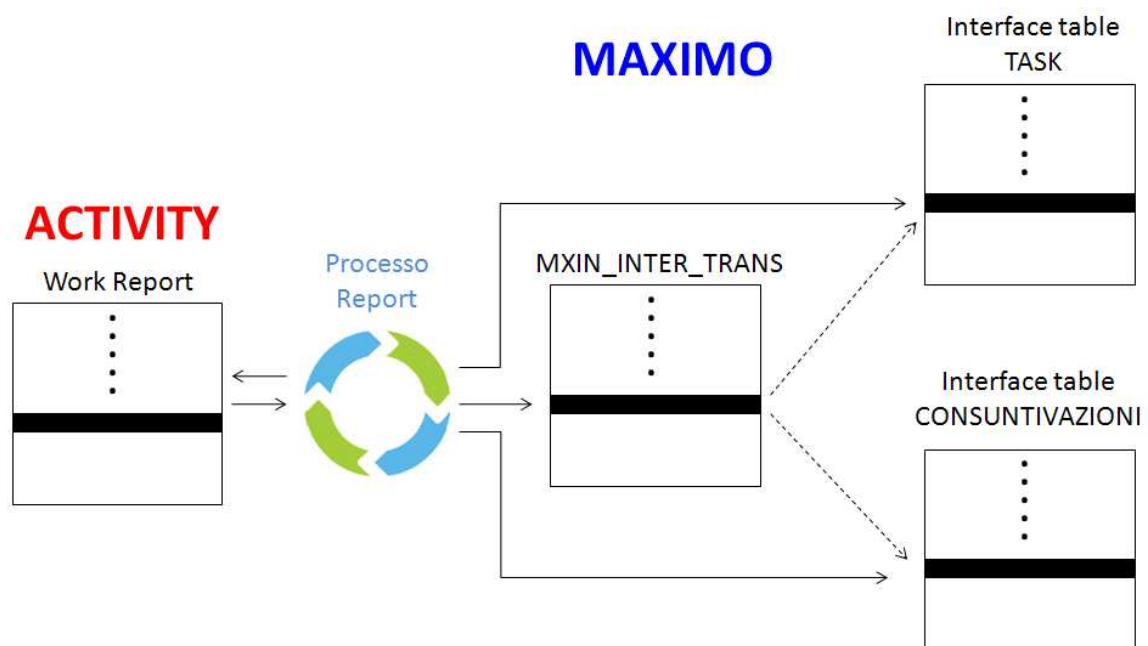
- nel primo caso sul “work order” di Maximo deve essere creato sia il task relativo all’attività svolta (che ne conterrà la descrizione), sia la consuntivazione delle ore del lavoratore;
- nel secondo caso sul “work order” di Maximo vengono solamente consuntivate le ore del lavoratore.

Una volta recuperati i dati che servono su Activity, **Report** esegue alcuni inserimenti sulle “interface table” desiderate: da là Maximo provvederà a sistemare i dati nelle tabelle giuste.

In un’offerta di consulenza, i dati da trasportare in Maximo saranno i seguenti:

- Lavoratore
- Descrizione attività
- Data esecuzione attività
- Ore lavorate

Questi, insieme ad altre informazioni di supporto, vengono inseriti nelle due “interface table” riguardanti il task e la consuntivazione delle ore. Inoltre viene eseguito anche l’inserimento di due record nella tabella MXIN\_INTER\_TRANS, record che punteranno alle due “interface table” suddette: in questo modo Maximo, che controlla periodicamente la tabella MXIN\_INTER\_TRANS, sa dove vi sono dei record nuovi da importare.



[Figura 5.19: le attività del processo “Report”]

Per i progetti a corpo, **Report** importa gli stessi dati che importa nella consulenza (esclusa la “descrizione delle attività”), ma inserisce solamente un record sulla “interface table” delle consuntivazioni e uno su MXIN\_INTER\_TRANS. Questo perché la consuntivazione sarà riferita ad un task già presente nel “work order” del progetto, ovvero una delle fasi (e quindi non servirà creare un task nuovo).

Il resto del lavoro sulla fatturazione è a carico delle dinamiche di Maximo, già descritte nei paragrafi precedenti.

## 5.9 Test finali

I test eseguiti al termine di questo progetto non hanno potuto garantire standard di affidabilità e prestazioni, poiché sono stati eseguiti solamente su alcuni esempi di offerte e non su una così grande mole di dati da fornire certezze di utilizzo in condizioni simili alla realtà.

Mostriamo quindi uno degli esempi utilizzati per testare il corretto funzionamento dell'integrazione.

Ricordiamo, come nel capitolo precedente, che a causa delle clausole di segretezza definite dall'azienda per la tutela dei dati sensibili riguardanti l'azienda stessa e i relativi clienti, **ragioni sociali dei clienti, nomi del personale, luoghi e cifre economiche sono stati sostituiti con dati puramente esemplificativi od addirittura immaginari.**

CLIENTE: Sistema s.p.a.

OFFERTA: offerta di consulenza generica e sistemistica

SERVIZI OFFERTI:

- Assistenza generica: 70euro/h
- Assistenza sistemistica: 90 euro/h

CONTATTO CLIENTE: PPupino (Pino Pupino)

DATA DI FIRMA DELL'OFFERTA: 04/04/2012

### 5.9.1 Esecuzione

Una volta firmato il contratto, il responsabile commerciale di Tempestive s.r.l. censisce il cliente e la relativa offerta.

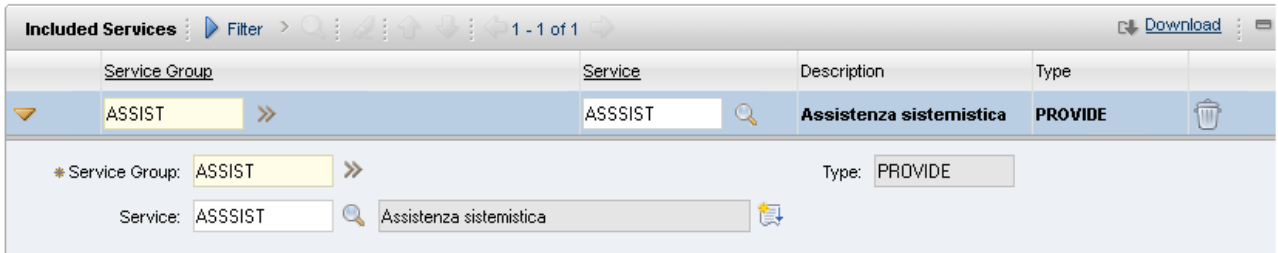
The image shows two screenshots from a software application. The top screenshot, titled 'PROGETTI', displays a form for creating a project. The 'DESCRIZIONE' field contains 'Offerta di Assistenza'. Other fields include 'DA DATA' (04/04/2012), 'A DATA' (31/12/2012), 'PREZZO ABILITATO' (checked), 'TIPO FATTURAZIONE' (Prezzo a Ora), 'PREZZO', 'DURATA', and 'ATTIVO' (checked). The user is identified as 'Chahinian Nicola' and the timestamp is '04/04/2012 17:10:52'. A large blue arrow points from this form to the bottom screenshot.

The bottom screenshot shows the 'Customer Agreements (SP)' form. The 'Agreement' field contains 'OSST20120404' and the 'Description' field contains 'Offerta di Assistenza', which is circled in red. A yellow box with the text 'Informazioni importate automaticamente' is overlaid on the right side. The 'Dates' section shows 'Start Date' as '04/04/12' and 'End Date' as '31/12/12', both circled in blue. Other fields include 'Revision' (0), 'Payment Terms', 'Billing Cycle', 'Customer Contract PO Reference', 'Show Cost?' (unchecked), 'Bill End Day', 'End of Bill Cycle?' (unchecked), 'Renewal Date', 'Last Bill Date', 'Date Changed' (06/04/12 16.03), and 'Changed By' (MAXADMIN).

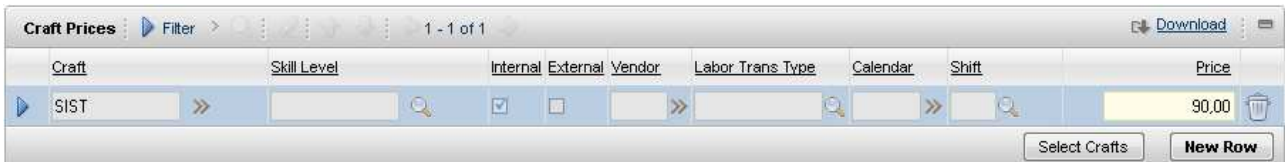
[Figura 5.20: test offerta]

Il processo **Clienti** e il processo **Progetti** si occupano di importare i dati su Maximo: come vediamo dalla figura 5.20, viene creato un “customer agreement” con il suo identificatore (OSST20120404) e vengono importati la descrizione dell’offerta (evidenziata in rosso), e le date di inizio e termine del contratto (evidenziate in blu). Chiaramente questa offerta verrà legata al cliente “SST”.

L’“agreement” viene quindi compilato dal responsabile del progetto, che conosce tutti i termini dell’offerta che ha pattuito con il cliente. Verranno impostati i prezzi orari di un sistemista (per gli interventi di tipo sistemistico) e di un programmatore junior (per le consulenze generiche), impostando chiaramente anche i servizi relativi nelle condizioni di applicabilità.



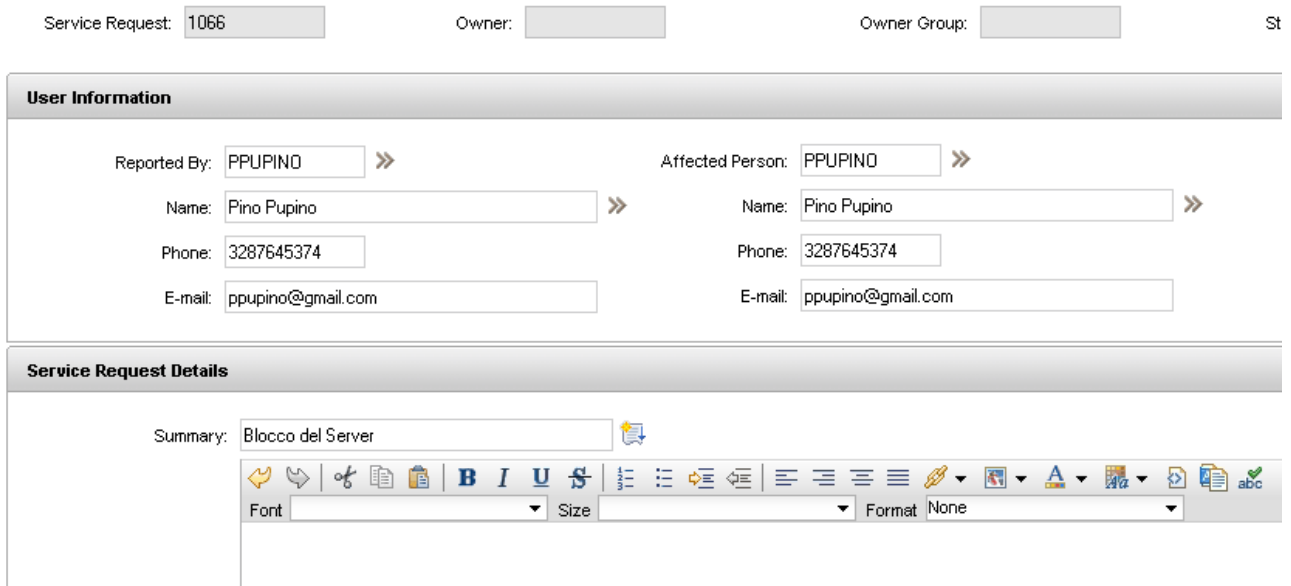
[Figura 5.21: test condizioni di applicabilità]



[Figura 5.22: test prezzi sull'offerta]

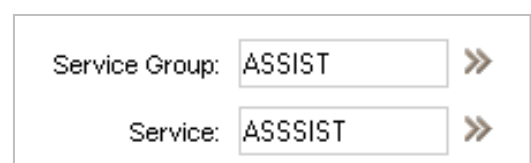
Avremo quindi due piani di prezzo (“price schedule”), ognuno con le sue tariffe e le sue condizioni di applicabilità

Fatto questo, tutto è pronto per un richiesta di intervento da parte del cliente. Il Sig. Pino Pupino, infatti, richiede un’assistenza per un improvviso blocco del server. Il responsabile del progetto crea quindi un ticket di assistenza, e, conoscendo la natura del problema, imposta un servizio di tipo sistemistico.

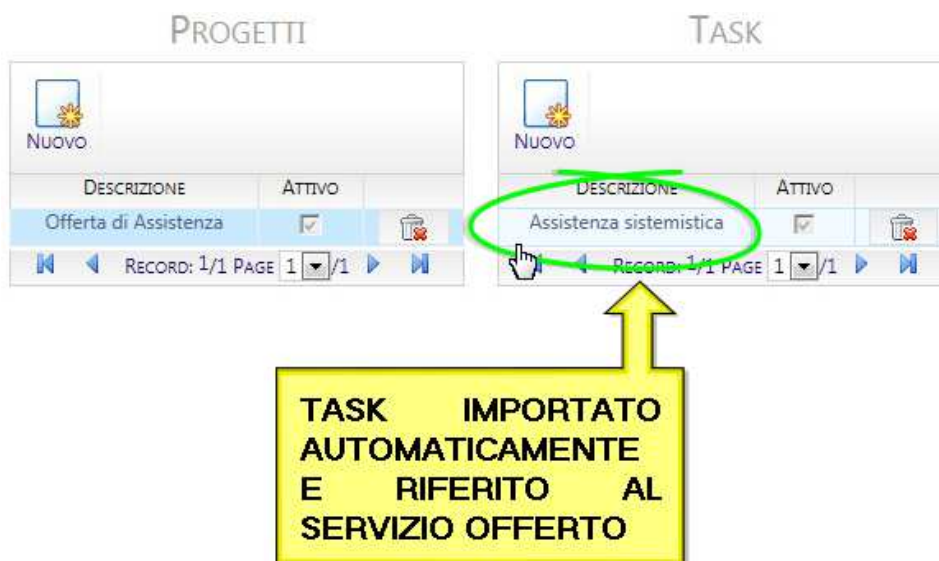


[Figura 5.23: test ticket di assistenza]

Il servizio impostato, fa in modo che l’ordine di lavoro che si creerà automaticamente erediterà lo stesso “Service” e il processo **WorkOrder** importerà proprio questo servizio come task in Activity. In più con l’applicazione automatica dell’“agreement” (eseguita dalla “escalation” che periodicamente controlla le offerte applicate agli ordini di lavoro) Maximo riconoscerà il servizio come condizione di applicabilità del “price schedule” corretto, impostando quindi la tariffazione del sistemista a 90 euro l’ora.



Otteniamo quindi l'importazione delle informazioni da Maximo ad Activity ad opera del processo **WorkOrder**.



[Figura 5.24: test task dell'offerta]

Siamo quindi pronti per eseguire i report relativi a questa richiesta di servizio. Il dipendente, una volta eseguito l'intervento, redigerà il report sull'attività da lui svolta.

### LE MIE ATTIVITÀ

SALVA
 CHIUDI

PERSONA Chahinian Nicola

DATA

ORE

LOCALITÀ

ATTIVO

CLIENTI  ▼

PROGETTI  ▼

TASK  ▼

TICKET  ▼

DESCRIZIONE

NOTE

[Figura 5.25: test report del dipendente]

I dati inseriti nel report saranno salvati nel database di Activity con un nuovo record, che verrà notato dal processo **Report**: quest'ultimo si prenderà le informazioni riguardanti la consuntivazione e la descrizione dell'attività svolta e creerà su Maximo, all'interno del "work order" corretto (quello che era stato creato precedentemente in automatico con il ticket):

- 1) il task relativo all'attività svolta;
- 2) la consuntivazione delle ore lavorate dal dipendente per quell'intervento.

The screenshot displays two sections of the Maximo interface. The top section, titled "Tasks for Work Order 1073", shows a table with columns for Sequence, Task, Summary, Estimated Duration, and Status. A single row is visible with Sequence 10, Task "Ripristino funzionamento del Server", Estimated Duration of 8:00, and Status "COMP". Below this is a navigation bar with tabs for Labor, Materials, Services, Tools, and Fees and Charges. The bottom section, titled "Labor", shows a table with columns for Task, Labor, Name, Approved, Start Date, Start Time, End Time, and Regular Hours. A single row is visible with Task 10, Labor NCHA, Name Nicola Chahinian, Start Date 06/04/12, and Regular Hours 8:00. There are "Select Labor" and "Select Planned" buttons at the bottom right of the labor section.

[Figura 5.26: test consuntivazione attività e risorse]

Come notiamo, abbiamo l'attività con il relativo "labor", quest'ultimo coincidente con il dipendente che ha fatto il report. Il prezzo dell'intervento sarà quindi calcolato da Maximo utilizzando le regole stabilite nell'offerta, e quindi moltiplicando le ore lavorate per il costo orario.

The screenshot shows the "Details" section of the labor record. It contains several fields: Craft (SIST), Start Date (06/04/12), End Date, Skill Level, Start Time, End Time, Regular Hours (8:00), Line Cost (0,00), Rate (0,00), and Line Price (720,00). The values 8:00 and 720,00 are circled in red, indicating the automatic calculation of the line price based on the regular hours and rate.

[Figura 5.27: test impostazione automatica dei prezzi]

A fine mese, viene eseguita la fatturazione per tutti gli interventi eseguiti (in quel mese). Il "bill batch" (creatosi automaticamente) raccoglierà quindi in automatico tutti gli ordini di lavoro contenenti consuntivazioni di quel mese, dal quale potrà essere creata la fattura in formato PDF.



List **Bill Batch**

Bill Batch: 1020    Fatturazione mensile - Aprile    Organization: INTEGRA    Attachments

Customer: SST    Sistema s.p.a.    Required Review Date:     Bill Status: INPROGRESS

Reviewed By:     Agreement: SST20120404    \* Bill End Date: 01/05/12    Pre Tax Total: 2.000,00

Reviewed Date:     Billing Schedule:

Schedule Date:

---

**Bill Batch Lines**    Filter >    1 - 3 of 3    Download

Line	Order	Class	Description	Total Cost	Quote Type	Included Price	Minimum Price	Agreed Price	Billed Price	Status
1	1073	WORKORDER	Blocco del Server	0,00		<input type="checkbox"/>	<input type="checkbox"/>	720,00	720,00	WAPPR
2	1075	WORKORDER	<b>Malfunzionamento Application server</b>	<b>0,00</b>		<input type="checkbox"/>	<input type="checkbox"/>	<b>720,00</b>	<b>720,00</b>	WAPPR
3	1077	WORKORDER	Richiesta di controllo sicurezza del PC	0,00		<input type="checkbox"/>	<input type="checkbox"/>	560,00	560,00	WAPPR

---

**Labor**    Materials    Services    Tools    Fees And Charges

**Labor**    Filter >    1 - 1 of 1    Download

Task	Labor	Name	Approved	Regular Hours	Premium Pay Hours	Line Cost	Line Price	Bill Price
	NCHA	Nicola Chahinian	<input checked="" type="checkbox"/>	8:00		0,00	720,00	720,00

[Figura 5.28: test fatturazione automatica]

## Capitolo 6

# Conclusioni

### 6.1 Conclusioni sul progetto

Prima di tutto, ricordiamo che il progetto finale di integrazione fra Tempestive Activity e IBM Maximo è frutto di un grosso studio e di tutto il lavoro precedentemente descritto in questa relazione.

Il tutto si è basato su un percorso che partiva dalla conoscenza delle funzionalità di Maximo Asset Management, alla conoscenza delle logiche di Maximo for Service Provider (sconosciute quasi completamente all'azienda ospitante lo stage) fino allo sviluppo delle capacità di integrare il software con l'applicazione Activity. Nella pratica, questo percorso è descritto all'interno di questa relazione, portando alla conclusione un periodo di stage svolto con successo.

Chiaramente, le possibilità di arricchimento di quest'integrazione sono molteplici. Con un po' di tempo in più, sarebbe stato interessante incorporarvi alcune logiche molto funzionali, fra cui:

- *un ciclo di gestione delle “location”*: su Activity, la locazione di lavoro è rappresentata semplicemente da una casella di testo in cui scrivere dove è stata effettuata l'attività lavorativa. Su Maximo invece, come descritto nei capitoli precedenti, esiste un ciclo completo di gestione delle “location” legate a ben determinati clienti.  
Un buon obiettivo sarebbe stato trasformare quella casella di testo su Activity in una casella di scelta multipla (“drop down list”) collegata attraverso qualche logica alla tabella delle “location” di Maximo. In questo modo anche la consuntivazione sarebbe stata più precisa;
- *un ciclo di gestione dei rimborsi di trasferta*: su Activity questo tipo di informazioni sono presenti quando si inserisce un nuovo report: per poter sincronizzare questi dati con Maximo si sarebbe dovuto ricorrere sia all'implementazione del ciclo delle “location” descritto al punto precedente, sia all'aggiunta di codice nei processi Java.
- *la configurazione di Maximo per la creazione automatica di ticket di assistenza a partire dalla ricezione di una mail strutturata*: molto interessante, e di enorme utilità per la riduzione del TCO aziendale. Si sarebbe potuto configurare Maximo per accedere ad una casella e-mail (in maniera sicura e protetta) e controllare la posta in arrivo e, in caso di ricezione di una particolare tipologia di e-mail strutturata, avviare un processo di creazione automatica di una “service request”, togliendo al responsabile del progetto l'onere di crearla manualmente previa telefonata del/al cliente.

Queste e tante altre le possibilità per ottenere maggior efficienza dall'integrazione.

Per quanto riguarda gli obiettivi funzionali del progetto che ci si era preposti, i risultati sono stati raggiunti discretamente. In particolare, gli obiettivi erano:

- inserire il software IBM Maximo (già utilizzato dall'azienda per i relativi clienti) all'interno delle dinamiche di Tempestive s.r.l., adattandolo ad esse
  - **raggiunto parzialmente**: il progetto è stato creato considerando solo due tipologie di offerta fatte dall'azienda ai clienti, uno dei motivi per cui esso non è entrato realmente in funzione all'interno di essa.

- collegare il software IBM Maximo all'applicazione Tempestive Activity, già utilizzata dall'azienda per l'organizzazione delle proprie attività, progetti, task, clienti, ecc.;
  - **raggiunto quasi completamente:** i due software comunicano perfettamente nelle dinamiche che ci si era preposto di implementare. Permane qualche piccolo “bug” di funzionamento in alcuni particolari frangenti;
- rendere Tempestive Activity adatto per la gestione della fatturazione rivolta ai clienti dell'azienda nel suo ruolo di Service Provider;
  - **raggiunto completamente:** la fatturazione legata ai report eseguiti dai dipendenti avviene in maniera precisa e completa grazie all'interazione con Maximo e alla creazione di layout fissi per i report con l'utilizzo di BIRT Report Designer;
- ridurre il TCO aziendale eliminando alcune operazioni già eseguite routinariamente all'interno dell'azienda;
  - **raggiunto quasi completamente:** sono state eliminate tutte le operazioni manuali che ci si era preposto di sostituire: la creazione manuale della fattura e dei report delle attività e l'operazione di inserimento dei task delle offerta su Activity. Permangono, come già accennato precedentemente, ancora alcune dinamiche che sarebbero potute essere automatizzate con un po' di tempo in più a disposizione.

## 6.2 Conclusioni sullo stage

Gli obiettivi descritti nel paragrafo 1.3 sono stati raggiunti nelle seguenti modalità:

- 1) conoscenza del software IBM Maximo Asset Management e delle relative logiche di funzionamento, nonché conoscenza e capacità di utilizzo delle applicazioni aggiuntive legate al pacchetto IBM Maximo for Service Provider, insieme alle relative dinamiche tipiche di un'azienda fornitrice di servizi : la prova di ciò doveva essere data dal censimento all'interno di Maximo di alcune offerte, fra le quali alcune tipiche dell'azienda ospitante lo stage agente in qualità di “service provider”;
  - **raggiunto parzialmente:** alcune delle offerte prese in esame hanno trovato spazio all'interno della logica di Maximo; per altre permane ancora il dubbio su come inserirle;
- 2) capacità di utilizzo del software BIRT Report Designer per la creazione automatica di layout fissi per i report: la prova di ciò doveva essere data dalla creazione di due layout esattamente identici a quelli utilizzati dall'azienda per emettere fatture e stampare i report delle attività svolte per i clienti;
  - **raggiunto completamente:** i due layout sono stati creati in tutte le loro parti e rispecchiano appieno i modelli pre-esistenti. In questa relazione non è stato mostrato il report riguardante le attività svolte per i clienti, poiché il file relativo è rimasto dentro ad un server dell'azienda che è morto nel pieno delle mie attività e non so che fine abbia fatto;
- 3) capacità di far dialogare Maximo con altri software o applicazioni esterni attraverso il modulo di integrazione in esso contenuto: un progetto finale di integrazione con l'applicazione Tempestive Activity doveva dar prova di tale capacità;
  - **raggiunto parzialmente:** il modulo di integrazione di Maximo permette di interagire con i più svariati software e applicazioni e con diverse modalità. Io ho utilizzato un'integrazione fra database attraverso il MIF e le “interface table”, ma sicuramente ci sarebbero stati anche altri modi per implementare lo stesso passaggio di informazioni.

### 6.3 Conclusioni sull'azienda

Tempestive s.r.l., a mio parere, è un'azienda molto dinamica e innovativa. Il cliente viene prima di ogni altra cosa, ed ogni progetto ha un proprio responsabile, anche se chi ci lavora è molte volte un team completo di programmatori di ogni tipo, sistemisti ed esperti di marketing.

I titolari dell'azienda sono persone molto disponibili e che non si comportano come "leader", ma come persone con molta esperienza alle spalle, disponibili ad aiutare e consigliare i propri dipendenti ponendosi anche al loro stesso piano: una dote non sempre presente nelle persone di alto livello sociale e culturale. Anche i dipendenti si sono dimostrati molto disponibili su ogni mio dubbio e perplessità: non sono mancate le volte in cui hanno perso diverse ore per risolvere alcuni problemi di tipo sistemistico, burocratico, applicativo, ecc. legati alle mie scarse conoscenze iniziali.

Per quanto riguarda il materiale e la strumentazione messami a disposizione, posso ritenermi più che soddisfatto, poiché mi sono stati forniti ben 3 PC, una macchina virtuale su un server e due lavagne sulle quali poter apporre i miei schemi e discutere di essi con i dipendenti presenti.

Se posso segnalare una piccola nota negativa, è solo nel fatto che all'inizio delle mie attività gli obiettivi del mio stage non erano molto ben definiti; problema poi in realtà risolto in corso d'opera grazie alla disponibilità del mio tutor aziendale Bruno Bisello.

# Bibliografia

- 1 Tempestive s.r.l., [www.tempestive.com](http://www.tempestive.com)
- 2 IBM Maximo Asset Management Certification Guide, <http://www.redbooks.ibm.com/redbooks/pdfs/sg247761.pdf>
- 3 IBM Maximo User Guide, [http://publib.boulder.ibm.com/tividd/td/ITSerDsk/sdug621/en\\_US/PDF/621\\_mx\\_ug.pdf](http://publib.boulder.ibm.com/tividd/td/ITSerDsk/sdug621/en_US/PDF/621_mx_ug.pdf)
- 4 IBM Maximo System Administration Guide, [http://pic.dhe.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_sys\\_admin\\_guide.pdf](http://pic.dhe.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_sys_admin_guide.pdf)
- 5 IBM Maximo Workflow Implementation Guide, [http://publib.boulder.ibm.com/tividd/td/ITMax/621\\_mx\\_wkfl\\_imp/en\\_US/PDF/621\\_mx\\_wkfl\\_imp.pdf](http://publib.boulder.ibm.com/tividd/td/ITMax/621_mx_wkfl_imp/en_US/PDF/621_mx_wkfl_imp.pdf)
- 6 Implementing IBM Maximo for Service Provider, <http://www.redbooks.ibm.com/redbooks/pdfs/sg247891.pdf>
- 7 DSO, <http://www.investopedia.com/terms/d/dso.asp>
- 8 BIRT Report Designer, <http://www.eclipse.org/birt/phoenix/project/description.php>
- 9 Requisiti di sistema su IBM Maximo, <http://www.ibm.com/developerworks/wikis/display/maximo/System+Requirements+for+Version+7.5.0.1+Maximo+Products>
- 10 Requisiti di sistema su Microsoft .NET Framework, <http://msdn.microsoft.com/en-us/library/8z6watww.aspx>
- 11 Requisiti di sistema su Microsoft SQL Server 2012, <http://msdn.microsoft.com/it-it/library/ms143506.aspx>
- 12 Tipo di dato “uniqueidentifier”, <http://msdn.microsoft.com/it-it/library/ms190215%28v=sql.105%29.aspx>