

POLITECNICO DI MILANO

Facoltà di Ingegneria

Corso di Laurea Specialistica in Ingegneria Gestionale



OPEN SOURCE E PLM: IL CASO ARAS

Relatore: Prof. Sergio TERZI

Tesina di laurea di:

Germán BRAVO ARIZA

Matricola 764330

Anno Accademico 2011/2012

INDICE

SOMMARIO	3
ABSTRACT	4
1. INTRODUZIONE AL PLM	5
1.1 DEFINIZIONE DI PLM.....	6
1.2 FUNZIONALITÀ DEL PLM.....	6
1.3 IMPLEMENTAZIONE DEL PLM	7
1.4 BENEFICI DELL'IMPLEMENTAZIONE DEL PLM.....	11
2. DATI DI PRODOTTO	14
2.1 FORMATI DI FILE.....	14
2.1.1 Tipi di formato.....	15
2.1.2 Identificazione del formato di un file	16
2.1.3 Struttura di un file.....	18
2.1.4 Lista di formati di file.....	19
2.2 TIPI DI DATO DI UN PRODOTTO	19
2.3.1 Metadati.....	19
2.3.2 Dati di testo.....	21
2.3.3 Dati di audio / video	21
2.3.4 Dati di immagini 3D.....	22
2.3.5 Dati di database	22
2.3.6 Dati di CAD.....	22
2.3.7 Dati geografici	22
2.3.8 Dati WEB	23
2.3.9 Dati non strutturati.....	23
2.3 STANDARD DI DATI.....	23
2.3.1 Standard di dati di prodotto	26
2.3.2 Standard di processi.....	31
3. OPEN SOURCE	35
3.1 COS'È IL SOFTWARE OPEN SOURCE	36
3.2 CONFRONTO OPEN SOURCE / FREE SOFTWARE	38
3.2.1 Paragone storico	40
3.2.2 Paragone ideologico	40

3.2.3	paragone sulle licenze.....	41
3.3	MODELLO DI BUSINESS.....	42
3.4	VANTAGGI / SVANTAGGI DELL'OPEN SOURCE.....	43
3.5	EVIDENZA D'ADOZIONE DELL'OS. ESEMPI.....	44
4.	ARAS INNOVATOR.....	48
4.1	SOFTWARE ARAS INNOVATOR.....	48
4.2	ARCHITETTURA.....	50
4.3	FUNZIONALITÀ.....	55
4.4	ALL'INTERNO DI ARAS INNOVATOR.....	65
4.5	MODELLO DI BUSINESS.....	69
4.6	PIATTAFORMA MICROSOFT CERTIFICATA.....	70
4.7	CREAZIONE DI VALORE PER CLIENTI/PARTNER.....	71
4.8	CASI DI ADOZIONE DI ARAS INNOVATOR.....	75
	Motorola.....	75
	Lear Corporation.....	77
	U.S. Army.....	79
4.9	ARAS INNOVATOR E LE NUOVE TECNOLOGIE.....	80
5.	CONCLUSIONI.....	81
	APPENDICE.....	85
	BIBLIOGRAFIA.....	90
	INDICE DELLE FIGURE.....	93
	INDICE DELLE TABELLE.....	95
	ACRONIMI.....	96
	Ringraziamenti.....	99

SOMMARIO

In questo lavoro di tesi è stato studiato come le tecnologie PLM (Product Lifecycle Management) e l'open source possono aiutare le aziende a raggiungere i loro obiettivi strategici. Questo avviene attraverso la riduzione dei costi, il miglioramento della qualità e la riduzione del time to market, senza compromettere l'innovazione, i servizi e le operazioni di fabbricazione di un prodotto.

Questo studio si è focalizzato sul software Aras Innovator. Prima è stata realizzata un'introduzione al PLM, ai dati di un prodotto e al mondo dell'open source, e poi il software Aras Innovator è stato analizzato più dettagliatamente.

Oggi, le aziende sono esposte a sfide complesse per migliorare lo sviluppo dei loro prodotti come frequenti modifiche di progettazione, numerosi sistemi di dati incompatibili, norme, ecc. Le aziende sono tenute a fare di più con meno risorse, la complessità dei prodotti è crescente e invece il loro ciclo di vita è sempre più compresso. In questa situazione, le imprese devono trovare soluzioni se vogliono rimanere competitive. Ed è qui che entrano in gioco i sistemi PLM in generale e il software Aras Innovator nel nostro caso.

Aras Innovator facilita la gestione del ciclo di vita dei prodotti, progetti e altre attività, consentendo alle imprese di essere più competitive. La combinazione di tecnologie, metodi e buone pratiche di business risolvono i problemi dovuti ai rapidi cambiamenti del mercato. Questi sistemi promuovono il cambiamento all'interno dell'impresa, creando opportunità di miglioramento nei processi e nell'organizzazione. Una volta implementati hanno un impatto positivo in tutti i settori dell'azienda permettendo ad essa di innovare e gestire in modo efficiente i loro prodotti e i servizi in tutto il ciclo di vita aziendale, dalla concezione allo smaltimento (o riciclaggio).

ABSTRACT

The purpose of this thesis job is to study how PLM Technologies and Open Source can help the companies to reach their strategic goals. That can be obtained through cost reduction, quality improvement and reduction of the time to market, without compromising innovation, services and the fabrication of the product.

This study is focused on the software Aras Innovator. First was realized an introduction to PLM, products data and to Open Source. Suddenly, the software was analyzed more specifically.

Nowadays, companies are exposed to complex challenges to improve the development of their products like frequent design changes, numerous incompatible data systems, standards, etc. The companies has to product more with less resources, the complexity of products is increasing, while their life cycles are becoming more and more complicated. In this situation, the companies need to find different solutions to remain competitive. And that's why PLM systems and, more specifically, Aras Innovator Software can be useful.

Aras Innovator makes easier to manage the life cycle of products, projects and other activities. So, the companies become more competitive. The combination of technology, methods and good business practices solve problems caused by rapid changes of market. These systems promote business changes, creating opportunities of improvement in processes and organization. Once they have been developed, they impact positively on each business sector, so the company can innovate and manage efficiently its products and services during all its life cycle, from conception to disposal.

1

INTRODUZIONE AL PLM

La società in cui viviamo è sottoposta a un continuo cambiamento con un enorme dinamismo. Questo significa che i processi che si svolgono in un'azienda devono essere molto più efficaci e veloci, tutte le azioni devono essere ben mirate. La grande evoluzione dell'informatica consente sempre di più mettere a disposizione strumenti per facilitare il lavoro, perfezionarlo e aggiungere valore a questo lavoro. Perciò sorge il PLM, che cerca di gestire più efficacemente tutto ciò che accade al ciclo di vita di un prodotto. L'evoluzione delle tecnologie informatiche ha quindi portato negli anni ad un graduale aumento dell'integrazione e l'interoperabilità dei e tra i sistemi IT, facilitando la gestione dei diversi processi aziendali (progettativi, logistici, produttivi, gestionali...) e automatizzando via via flussi di informazioni fino a quel momento gestiti "manualmente".

Le aziende e i suoi professionisti devono reagire in un modo veloce ed efficace ai cambiamenti che si producono. Si devono utilizzare le stesse risorse in meno tempo sviluppando così nuovi prodotti in un modo più agile. La collaborazione interna deve essere molto più fluida, anche esternamente con clienti e fornitori. Questa è la funzione del PLM, fornire gli strumenti per favorire questa collaborazione. Nella sfida del mercato globale, le aziende devono innovare per sopravvivere, le innovazioni devono

avvenire in tutte le dimensioni (prodotti, processi e organizzazione). Per distinguersi le aziende devono acquisire, gestire e riutilizzare le proprie conoscenze. Per affrontare questa sfida, può adottarsi il PLM correttamente focalizzato sui bisogni di ogni azienda.

1.1 DEFINIZIONE DI PLM

Possiamo definire il PLM come un approccio integrato di business che, supportato dalle tecnologie informatiche, realizza una gestione integrata, cooperativa e collaborativa delle informazioni di un prodotto lungo le diverse fasi del suo ciclo di vita. Il PLM può essere idealmente visto come un sistema di gestione delle informazioni, che integra i “core processes” aziendali e connetta integra controlla i processi di business relativi ai prodotti da realizzare e alle informazioni strettamente legate a questi prodotti. Si può dire che il PLM comprende:

- Un orientamento strategico alla creazione di valore del prodotto
- L'applicazione di un approccio collaborativo per la valorizzazione delle competenze di attori diversi.
- L'uso di un consistente numero di soluzioni IT per la realizzazione della conseguente gestione coordinata, integrata e sicura di tutte le informazioni necessarie alla creazione di valore.

Il PLM non è solo uno strumento software, ma che unisce le dimensioni organizzative (processi), economiche (costi e ricavi), tecniche (attività e persone) e tecnologiche (sistemi IT) [1].

1.2 FUNZIONALITÀ DEL PLM

Il PLM fornisce diverse funzionalità per la gestione di tutte le informazioni del prodotto [2]:

- Strumenti di authoring: Strumenti di progettazione (CAx).

- Strumenti di CAD management e PDM: Supporto ai processi di sviluppo collaborativo di prodotti, tra le quali gestione delle versioni dei disegni e parti, sincronizzazione degli accessi (check-in, check-out), gestione dei permessi degli utenti, eventuale accessibilità ad utenti esterni.
- Gestione delle modifiche: Supporto fornito dai Sistemi Informativi (SI) al processo di gestione delle modifiche (Richiesta di Modifica).
- Gestione documentale: Supporto all'archiviazione e gestione dei documenti realizzati in azienda, del relativo ciclo di vita e di condivisione con i diversi utenti aziendali.
- Gestione dei processi e dei progetti: Supporto all'automatizzazione dei processi di SNP e alla gestione, definizione e monitoraggio dei progetti di SNP.
- Gestione dell'anagrafica: Gestione dell'anagrafica delle parti e dei componenti, realizzata tramite SI, in particolare per agevolare il recupero di informazioni e il conseguente risparmio di tempo in progettazione.
- Gestione delle distinte: Supporto fornito dai SI alla generazione assistita del calcolatore delle distinte base, gestione delle diverse "viste" e delle diverse personalizzazioni/configurazioni.
- Configuratore di prodotto: Supporto dalla configurazione commerciale di prodotto (per preventivazione, marketing, budgeting, allocazione risorse, ecc.).
- Definizione di prodotto e di portafoglio prodotti: Supporto dei SI alla definizione delle specifiche di prodotto e alle scelte del portafoglio dei prodotto a catalogo
- Tracciabilità dei prodotti: Supporto alla tracciabilità e rintracciabilità dei prodotti nelle fasi avanzate del ciclo di vita.

1.3 IMPLEMENTAZIONE DEL PLM

Una volta che l'azienda ha rilevato un certo numero di carenze nel modo di lavorare, viene eseguita l'implementazione di un sistema PLM per gestire al meglio tutte le informazioni disponibili. A questo punto, a seconda delle conoscenze e dei mezzi

dell'azienda, si analizza la situazione di partenza e dove si vuole arrivare o si affida l'analisi a consulenti esterni che fanno le loro proposte.

Alcuni dei fattori che ci spingono a implementare un sistema PLM sono [3]:

- Il ciclo di vita del prodotto deve essere ridotto. Si devono produrre nuovi prodotti in meno tempo.
- La personalizzazione del prodotto ai bisogni specifici del cliente aumenta con rispetto alla fabbricazione di prodotti standard.
- Aumenta la complessità del prodotto, si devono produrre migliori prodotti, di maggiore qualità e mantenere i costi sotto controllo.
- La domanda di documentazione tecnica addizionale come manuali d'uso per gli utenti, norme, specifiche, ecc. richiede più documenti da realizzare e tenere sotto controllo.

Una volta implementato, le soluzioni PLM offrono un impatto molto positivo a tutti i livelli di un'azienda. Tale impatto è superiore a quello prodotto dai sistemi quali ERP (Enterprise Resource Planning), CRM (Customer Relationship Management), o SCM (Supply Chain Management) [3].

L'implementazione di un sistema PLM potrebbe essere giustificata in nove paragrafi [3]:

- 1. Mantenere sotto controllo i dati di prodotto:** I dati di prodotto includono tutte le informazioni che descrivono lo stesso e i processi utilizzati per progettarlo, produrlo, usarlo e mantenerlo. I dati di prodotto sono virtualmente dappertutto in un'azienda. Le grandi aziende hanno milioni di documenti sia in formato elettronico che cartaceo descrivendo i loro prodotti. Le aziende hanno bisogno di implementare una soluzione PLM per garantire il controllo dei dati durante tutto il ciclo di vita di un prodotto. La tecnologia PLM permette all'azienda di archiviare e rendere disponibili tutti i dati riguardanti al prodotto in un ambiente sicuro e di accesso controllato, includendo sia i dati attuali che quelli storici, quando e dove sia necessario.

- 2. Automatizzare i processi relativi ai flussi di lavoro per incrementare la produttività:** Nello sviluppo e la manutenzione di un prodotto, molte attività sono ripetute con sequenze di passi e documenti molto simili. Senza un sistema di automatizzazione di flusso di lavoro è facile per le persone commettere degli errori, come dimenticare un passo o utilizzare un documento sbagliato. PLM consente di identificare i flussi di lavoro e automatizzarlo in modo che possano essere svolti più velocemente. Inoltre, coi flussi di lavoro automatizzati, quando le fasi dei processi evolvono è facile cambiare il flusso di lavoro o segnalare il nuovo approccio per coloro che sono coinvolti.
- 3. Process Reengineering:** Verifica del valore aggiunto. Le aziende che implementano PLM creano processi relativi ai prodotti che sono coerenti e efficaci. PLM offre la possibilità di integrare processi indipendenti. Una buona architettura di processi consente l'esecuzione coerente durante il ciclo di vita, eliminando i punti vuoti tra i processi, rendendoli più coerenti, più veloci e riducono i costi dell'operazione.
- 4. Ridurre il time to market:** In molte aziende ogni dipartimento utilizza le applicazioni che crede saranno più utili per le loro particolari operazioni. Spesso, queste applicazioni non permettono l'accesso alle informazioni generate di altri dipartimenti. Ogni dipartimento diventa così in un'isola di dati di automazione. PLM facilita il flusso di dati attraverso l'organizzazione.
- 5. Sviluppo del prodotto corretto. Implementare una strategia globale di prodotto con la tecnologia PLM. Massimizzare i benefici con prodotti localizzati:** Come risultato della globalizzazione, le aziende devono essere vicine ai clienti in molti luoghi diversi e comprendere le esigenze dei clienti per vendere i loro prodotti in diversi ambienti. Le applicazioni PLM aiutano le aziende a individuare le esigenze del cliente associando dati di prodotti esistenti con nuovi prodotti, specificando caratteristiche del cliente e del mercato. Una buona soluzione PLM permetterà alle aziende di sviluppare a livello globale, e ulteriormente sviluppare a livello locale le versioni di un determinato prodotto.

- 6. Sviluppo collaborativo. Massimizzare le risorse locali, globali, interne ed esterne:** Dalla metà degli anni novanta, molte organizzazioni del settore manifatturiero e di design hanno cominciato a svolgere tutto lo sviluppo del prodotto in un unico luogo. Da allora lo sviluppo del prodotto è effettuato da persone che lavorano insieme, ma in luoghi diversi o diverse aziende. PLM consente alle aziende di gestire e lavorare in modo più efficiente in questo ambiente di networking e collaborazione. Permette alle aziende di lavorare in luoghi diversi grazie ai software di collaborazione.
- 7. Riutilizzo delle informazioni:** Senza un sistema PLM, l'accesso alle informazioni è lento e goffo. C'è anche la possibilità che i dati esistenti si siano persi o, come già detto, esistano varie versioni degli stessi. Attualmente pochissimi prodotti sono sviluppati da zero. La maggior parte delle aziende mischiano variazioni di progetti esistenti e processi con alcune parti nuove, che tutto insieme formano i nuovi prodotti. PLM fornisce gli strumenti per trovare facile e velocemente le informazioni e così riutilizzarle per nuovi progetti e processi.
- 8. Aumentare i ricavi di prodotti in fasi mature:** Le informazioni fornite dal PLM possono essere tanto importanti quanto il feedback fornito dal cliente. Questo aiuta a comprendere i prodotti e migliorare altri già esistenti. Le applicazioni PLM forniscono supporto alle decisioni per la manutenzione predittiva, diagnostici e modelli d'uso dei prodotti. Così i componenti principali possono essere sostituiti prima di un guasto, evitando il costo della non operatività.
- 9. Migliorare la visibilità del prodotto. Indirizzare in modo più efficace con le informazioni fornite dal PLM:** Senza un'applicazione PLM può richiedere molto tempo trovare le informazioni necessarie per la segnalazione sulle specifiche dei prodotti, le caratteristiche e le prestazioni e il progresso del progetto. PLM fornisce rapporti permettendo ai prodotti e processi rivedere, analizzare e simulare gli effetti di diverse decisioni come l'introduzione di nuove tecnologie. Offre una vasta gamma di campioni, tabelle e grafici di

controllo e altri tipi, che aiutano i manager a prendere decisioni migliori riguardanti al prodotto in ogni fase della sua vita.

1.4 BENEFICI DELL'IMPLEMENTAZIONE DEL PLM

Vista la disparità tra i diversi motivi che giustificano l'implementazione di un sistema PLM, la vera sfida è quella di determinare i benefici adeguati a ciascuna azienda. I costi, siano di software, hardware o di implementazione, sono abbastanza precisi, anche se ci sono sempre alcune variazioni durante l'implementazione [3].

La sfida per ogni azienda è sostanzialmente la stessa: ridurre i costi, aumentare i ricavi e, attraverso una combinazione di entrambi, aumentare la redditività. Data la quantità di processi in cui sono coinvolti strumenti di automazione, la maggior parte dei benefici sono concentrati sulla riduzione dei costi. Questa riduzione di solito avviene nelle fasi di sviluppo del prodotto e produzione. Le soluzioni PLM possono anche avere un impatto sui ricavi. Anche se sempre è difficile attribuire l'aumento di ricavi all'implementazione di una nuova soluzione IT, lo è ancora di più, quantificare questi benefici con un certo grado di certezza. Si riconosce tuttavia che le imprese con incrementi di reddito, sono in parte giustificati dai benefici dopo l'implementazione di una soluzione PLM, in particolare quelli derivati dal time-to market (accorciare il lancio del prodotto al mercato) e dell'innovazione e la qualità del prodotto [3].

Le soluzioni PLM hanno un impatto positivo dalle linee più basse dell'organizzazione. Secondo CIMData, una società di consulenza leader mondiale nei servizi di consulenza PLM, la tecnologia offre i seguenti benefici specifici raggiunti per aziende che hanno implementato sistemi PLM [3]:

- 40% di miglioramento nei tempi di cambiamento di prodotto
- 15-30% di riduzione nei prototipi
- 25% di aumento della produttività nella progettazione
- Riduzione dell'83% del tempo di un processo di revisione di ingegneria (da 12 giorni a 2 giorni).

Questi dati corrispondono a implementazioni svolte dalla CIMData, e sono quindi, valori medi ottenuti. Questi miglioramenti misurabili e risparmi sui costi sono il risultato di implementazioni PLM. Le soluzioni PLM offrono una migliore capacità di creare un'organizzazione trasversale mediante lo scambio di informazioni. Seguendo con la stessa fonte (CIMData) possiamo dire che la tecnologia PLM può aiutare a risolvere i seguenti problemi specifici [4]:

- Il 47% degli errori che si trovano in certificazioni ISO sono attribuiti a uno scarso controllo della documentazione tecnica. La gestione dei documenti di qualsiasi soluzione PLM aiuta ad affrontare questo problema.
- Tra il 3% e il 7% del patrimonio tecnico di un'impresa si perde ogni anno causa di una documentazione tecnica non archiviata bene o semplicemente persa.
- Il 20 % del tempo di un ufficio tecnico si perde in cercare la revisione corretta di un piano o documentazione realizzata precedentemente. Con una gestione del cambiamento ben definita e gestita attraverso un software PLM, ci sono notevoli risparmi di tempo nella ricerca delle informazioni.

Nella figura qua sotto possiamo vedere l'impatto dell'implementazione di un sistema PLM in un'azienda. Si noti come c'è un aumento della redditività a causa della riduzione dei costi e l'incremento dei ricavi.

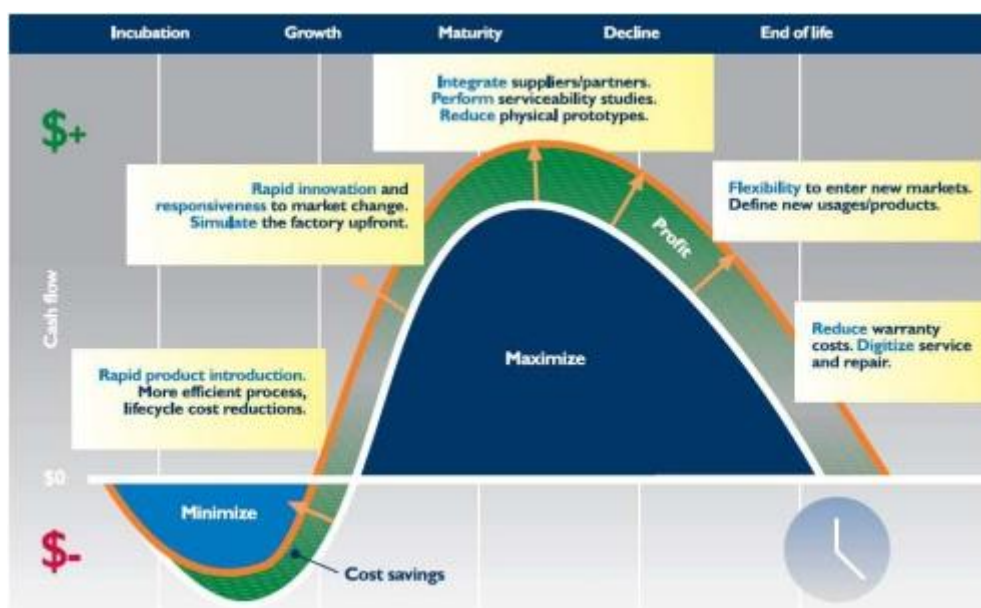


Figura 1: Redditività di un prodotto lungo tutto l'arco del suo ciclo di vita prima e dopo l'implementazione di un sistema PLM

In sintesi, l'implementazione e il successivo utilizzo di un sistema PLM consente di [3]:

- Incrementare la produttività
- Ridurre i costi
- Migliorare la qualità del prodotto e aumentare la capacità di innovazione
- Ridurre il tempo di uscita del prodotto al mercato, accelerando la commercializzazione (Time-to-market).
- Agevolare il rispetto degli standard e le norme
- Migliorare la sicurezza e l'integrità delle informazioni

2

DATI DI PRODOTTO

I dati di un prodotto compongono un insieme di tutte le informazioni del prodotto in sé lungo tutto l'arco del suo ciclo di vita, cioè, dalla progettazione allo smaltimento del prodotto, come possono essere descrizione, caratteristiche, utilizzo, manutenzione, funzionamento, istruzioni, informazioni preventive, disponibilità e così via. La gestione di tutti questi dati è la funzione del PDM (Product Data Management). Questi dati sono salvati in file per un'ulteriore consultazione, modifica, scambio, ecc.

2.1 FORMATI DI FILE

Un formato di file è un modo particolare in cui le informazioni sono codificate per la memorizzazione in un file di computer. Per essere file, le informazioni hanno bisogno di essere rappresentate come bit quando sono conservate in un'unità di disco o un altro dispositivo di memorizzazione digitale. I formati vengono identificati per alcun meccanismo (più avanti approfondiremo su quelli più utilizzati oggi) per aiutare il sistema operativo a determinare a quale programma del computer il file è associato [5].

2.1.1 Tipi di formato

I formati di file possono essere suddivisi in due tipi: proprietary and open format.

Open format

Un formato di file aperto è una specificazione pubblicata per la memorizzazione di dati digitali, solitamente mantenuti da un'organizzazione standard che può quindi essere utilizzato e implementato da chiunque. Un formato aperto può essere usato sia da un software di proprietario che da un software aperto utilizzando le licenze software tipiche utilizzate da ciascuno. Open format sono chiamati anche free file format se non sono gravati da alcun diritti d'autore, brevetti, marchi di fabbrica o di altre restrizioni (ad esempio, un open format di dominio pubblico), in modo che chiunque può utilizzarlo senza alcun costo monetario per qualsiasi scopo desiderato. Esempi di open format sono HTML, JPEG, PHP o XML [6].

Proprietary format

Questi sono formati di file in cui il modo di presentazione dei suoi dati è proprietà intellettuale di un individuo o un'organizzazione che afferma la proprietà sul formato. Il formato proprietario può essere aperto se è pubblicato, o chiuso, se è considerato segreto commerciale. A differenza di un formato libero che non è mai chiuso. Esempi di proprietary format sono MP3, DWG, PSD o RAR [7].

Alcuni formati di file sono stati progettati per tipi di dati molto particolari, ad esempio i file PNG memorizzano le immagini statiche con compressione senza perdita di dati. Altri formati invece sono progettati per la conservazione di diversi tipi di dati, ad esempio il formato Ogg può agire come un contenitore per diversi tipi di multimediali, compresa qualsiasi combinazione di audio e video, con o senza testo, e metadati. A volte è possibile che un programma interpreti il contenuto di un file come se fosse codificato secondo un formato diverso da quello con cui è stato creato. Ad esempio si può riprodurre un documento di Microsoft Word come una canzone utilizzando un programma lettore musicale, ma molto probabilmente si otterranno solo rumori e non

musica. Ciò è perché una disposizione ragionevole di bit in un formato non ha quasi mai senso in un altro.

Per molti formati sono state pubblicate delle specifiche che descrivono esattamente come i dati devono essere codificati e che possono essere utilizzate per stabilire se un programma specifico tratta correttamente o meno un determinato formato. Ma non sempre queste specifiche sono disponibili, innanzitutto alcuni formati sono considerati segreti industriali e le loro specifiche non vengono distribuite pubblicamente, come avviene ad esempio per molti dei formati usati dalla Microsoft Office, inoltre in molti casi gli sviluppatori non scrivono un documento di specifica separato, ma definiscono il formato solo implicitamente attraverso il programma che lo gestisce. In questo modo i dati salvati con quel programma non possono essere letti con altri programmi simili (il file può essere sempre letto da qualunque programma mai dati restano incomprensibili se non si conosce il formato con cui sono stati salvati). Risalire ai dati salvati in un formato sconosciuto è sempre possibile attraverso un lavoro di reverse engineering ma è, di solito, un processo molto lungo e costoso. Se il formato in questione è anche criptato, risalire ai dati diventa praticamente impossibile [5].

2.1.2 Identificazione del formato di un file

I file esistenti in un computer non si distinguono tra di loro a basso livello poiché tutti sono sequenze di zero e uno. Tuttavia, in modo che possano essere modificati, visualizzati o processati è necessario conoscere il formati in cui sono stati salvati. Nel corso della storia dell'informatica sono state sviluppate diverse iniziative volte ad identificare rapidamente e in modo sicuro il formato di un file. I due più utilizzate sono l'uso delle estensioni nei nomi dei file e l'utilizzo di "magic numbers".

Estensioni di file

I formati vengono identificati per l'estensione del file, questa estensione è negli tre o quattro ultimi caratteri dopo il periodo che costituisce un intero nome del file. Questo meccanismo è sempre stato utilizzato dai sistemi operativi Macintosh e Windows. L'estensione permette un'identificazione molto veloce del formato, permette anche di

fare delle liste di file classificati per formato in modo agile, è molto trasportabile dato che l'informazione viaggia sempre con i file ma è poco affidabile e poco informativa poiché l'estensione può essere manipolata facilmente e non distingue tra diverse versioni dello stesso formato [8]. Attualmente ci sono diversi servizi gratuiti online che descrivono i formati associati a una determinata estensione:

- <http://www.file-extensions.org>
- <http://filext.com>
- <http://www.fileinfo.net>

I “Magic numbers”

consistono in un codice standard nell'intestazione del file che indica il tipo di formato. Ad esempio i file GIF hanno il codice “GIF89a” (0x474946383961) o “GIF87a” (0x474946383761). Questo meccanismo è stato creato all'interno del sistema operativo Unix. I “numeri magici” sono un meccanismo altamente affidabile e trasportabile dato che l'identificazione è data insieme ai dati stessi. Non è così facile da identificare come l'estensione poiché si deve accedere al file stesso, non basta con guardare il nome. Purtroppo non si applica ai formati di testo, solo a quelli binari. Per i file di testo, in particolare per l'intera famiglia dei sottoformati XML, si utilizza un meccanismo simile ai “numeri magici” ma più esplicito. Le dichiarazioni di tipo di documento descrivono in dettaglio il formato dei dati (ad esempio un grafico SVG porterà sempre la seguente dichiarazione `<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">`). Questi metadati o dichiarazioni di tipo di documento sono il meccanismo più informativo di quelli esistenti e di solito sono abbastanza affidabili. Come succede con i “magic numbers” non è un meccanismo particolarmente veloce e neanche comodo per processare grandi volumi di file poiché richiede accedere al file stesso [8].

In alcuni casi l'identificazione del formato avviene in metadati esterni al file e collegato a questo, come ad esempio nel Resource Description Framework (RDF). In questo caso l'identificazione del file ha problemi di trasportabilità perché si deve mantenere il rapporto tra due file fisicamente separati [8].

2.1.3 Struttura di un file

Ci sono diversi tipi di struttura in un file. I più comuni sono scritti di seguito.

Formati non strutturati (raw memory dump)

Formati di file precedenti utilizzavano raw data format che consistevano in “scaricare” direttamente le immagini di memoria di una o più strutture nel file. Questo presenta vari inconvenienti. A meno che le immagini della memoria hanno anche spazi riservati per le future estensioni, ampliare e migliorare questo tipo di file è molto difficile. Inoltre crea file che potrebbero essere specifici di una piattaforma o un linguaggio di programmazione (ad esempio una struttura che contiene una stringa Pascal non viene riconosciuta come tale in linguaggio C). D'altra parte lo sviluppo di strumenti per leggere e scrivere in questo tipo di file è molto semplice. Le limitazioni dei formati non strutturati portano allo sviluppo di altri tipi di formati di file che possono essere facilmente estesi ed essere compatibili contemporaneamente [5].

Chunk-based formats

In questo tipo di struttura di file, ogni porzione di dati è incorporata in un contenitore che identifica in qualche modo i dati. L'ambito del contenitore può essere identificato per l'inizio e la fine di marcatori di qualche tipo, da un campo di lunghezza esplicita in qualche parte, o da esigenze fisse di definizione del formato del file. Le informazioni che identificano un particolare “pezzo” si possono chiamare di diversi modi come “nome di campo” “identificatore” o “tag”. Questi identificatori sono leggibili e classificano parti di dati come “cognome” o “indirizzo”. Con questo tipo di struttura di file, gli strumenti che non conoscono determinati identificatori semplicemente ignorano quelli che non capiscono, questo può o non può essere utile a secondo del significato reale dei dati saltati [5].

Directory-based formats

Questo è un altro formato estensibile, che assomiglia molto a un sistema di file, in cui è composto il file di “directory entries” che contiene la posizione dei dati all'interno del

file stesso e le sue firme (e in alcuni casi il tipo). Buoni esempi di questi tipi di struttura di file sono immagini di disco, documenti OLE e immagini TIFF [5].

2.1.4 Lista di formati di file

Prima abbiamo visto alcuni meccanismi per identificare il formato di un file. Noi ci concentreremo su quello delle estensioni poiché è il più veloce da identificare e quello che usano i sistemi operativi più importanti.

C'è una quantità grandissima di formati diversi ed è impossibile elencarli tutti, abbiamo già detto prima alcuni siti in cui possiamo trovare le estensioni associate ai suoi formati corrispondenti. Un'altra buona classificazione la possiamo trovare sul sito della Wikipedia, dove sono classificati i formati per la tipologia dei dati (database, documenti, matematici, musicali, di video, XML, ecc) [9] e anche possiamo trovare una classificazione in ordine alfabetico vedendo che programma utilizza ogni estensione e una piccola descrizione [10].

2.2 TIPI DI DATO DI UN PRODOTTO

Le informazioni che compongono i dati di un prodotto possono essere di diversi tipi in base al suo contenuto e all'obiettivo per cui sono stati creati. Questi dati li possiamo classificare in diversi gruppi. Come già detto esiste una quantità grande di file diversi, questi file si possono aggruppare secondo la tipologia di dati, ma anche così esiste una gran quantità di gruppi, per questo motivo di seguito non nomineremo tutti i gruppi.

2.3.1 Metadati

Metadati sono definiti come dati che forniscono informazioni su uno o più aspetti dei dati. Questi aspetti possono essere modalità di creazione dei dati, scopo dei dati, ora e data di creazione, creatore o autore dei dati, posizione su una rete di computer, standard utilizzati, ecc. Ad esempio, una immagine digitale può includere metadati che

descrivono quanto sia grande la foto, l'intensità del colore, la risoluzione della immagine, quando è stata creata e così via [11].

I metadati sono tradizionalmente presenti nei cataloghi a scheda delle biblioteche. Siccome l'informazione è diventata sempre più digitale, i metadati vengono usati anche per descrivere i dati digitali standard specifici di metadati per una particolare disciplina. Descrivendo i contenuti e il contesto del file di dati, la qualità dei dati originali è notevolmente aumentata. Ad esempio una pagina web può includere metadati specificando in quale lingua è scritta, quali strumenti sono stati utilizzati per crearlo e dove andare per ottenere più informazioni, consentendo al browser per migliorare automaticamente l'esperienza degli utenti [11].

I metadati sono dati. Come tali, i metadati possono essere memorizzati e gestiti in un database. Tuttavia, senza un contesto o punto di riferimento può essere impossibile identificare i metadati solo guardandoli. Ad esempio, immaginiamo un database che contiene diversi numeri, tutti con 13 cifre, potrebbero essere i risultati di calcoli o un elenco di numeri da inserire in un'equazione, senza un contesto gli stessi numeri possono essere percepiti come i dati. Invece se diciamo che questo database è un registro di una collezione di libri, quei numeri di 13 cifre possono essere ora il codice ISBN (informazioni che fanno riferimento al libro ma non sono esse stesse le informazioni all'interno del libro [11].

Ci sono diversi modi di classificare i metadati in base a la loro tipologia. Esistono diversi modi perché l'utilizzo dei metadati copre una vasta gamma di campi di applicazione [11].

- Metadati strutturali / metadati guida: gli strutturali sono usati per descrivere la struttura dei sistemi informatici, quali tabelle colonne e indici. I metadati guida invece sono usati per aiutare gli esseri umani a individuare elementi specifici e di solito è espressa come un insieme di parole chiave in un linguaggio naturale.
- Metadati tecnici / metadati di business: quelli tecnici si riferiscono ai metadati interni e quelli di business ai metadati esterni.
- Metadati descrittivi / metadati strutturali / metadati amministrativi: metadati descrittivi sono le informazioni utilizzate per cercare e identificare un oggetto,

gli strutturali danno una descrizione di come i componenti dell'oggetto sono organizzati e gli amministrativi riguardano alle informazioni tecniche tra cui tipo di file.

I metadati, o più correttamente, i vocabolari utilizzati per assemblare le dichiarazioni dei metadati, sono in genere strutturati secondo una concezione standardizzata con un ben definito schema di metadati, includendo standard di metadati e modelli di metadati. Strumenti come vocabolari controllati, tassonomie, tesauri, dizionari di dati e registri dei metadati possono essere utilizzati per applicare un'ulteriore standardizzazione dei metadati. Un'uniformità strutturale dei metadati è anche di fondamentale importanza per lo sviluppo del modello dei dati e nella progettazione di database [11].

Gli standard internazionali si applicano ai metadati. Molto lavoro è stato compiuto nelle comunità nazionali ed internazionali, in particolare gli standard ANSI (American National Standards Institute) e ISO (International Organization for Standardization) per raggiungere un consenso su metadati per la standardizzazione e registri [11].

2.3.2 Dati di testo

I file di testo contengono dati testuali e possono essere salvati in formato testo normale o in formato testo ricco. Mentre la maggior parte dei file di testo sono i documenti creati e salvati dagli utenti, possono anche essere utilizzati dagli sviluppatori di software per memorizzare i dati di un programma. Esempi di file di testo sono documenti di testo, file di log e messaggi e-mail salvati. Comuni estensioni di file di testo comprendono .TXT, .RTF, .LOG, e .DOCX [12].

2.3.3 Dati di audio / video

Questi dati che possono essere riprodotti con un software di audio/video sono contenuti in file di audio/video e utilizzano diversi per codificare e comprimere i corrispondenti dati di audio/video. Comuni estensioni di file audio includono .WAV, .AIF, .MP3 e .MID. Invece per il video sono .MPG, .MOV, .WMV, e .RM [12].

2.3.4 Dati di immagini 3D

I file di immagine 3D comprendono modelli tridimensionali, animazioni 3D, e file di progetto in 3D. I modelli 3D sono tipicamente costituiti da poligoni e vertici che creano la forma del modello. Essi possono includere anche strutture, nonché informazioni di illuminazione e ombreggiatura. Comuni estensioni di file di immagine 3D includono .OBJ, .STP, e i file .MA [12].

2.3.5 Dati di database

I file di database archiviano i dati in un formato strutturato, organizzato in tabelle e campi. Le singole voci all'interno di un database sono chiamati record. I database sono comunemente utilizzati per la memorizzazione di dati a cui fanno riferimento siti web dinamici. Comuni estensioni di file di database includono .DB, .ACCDB, .NSF, e .FP7 [12].

2.3.6 Dati di CAD

I file di CAD (Computer Aided Design) possono contenere disegni 2D o 3D. Essi sono generati dai programmi software CAD, che possono essere utilizzati per creare modelli o piani di architetture. I file CAD 2D sono spesso indicati come disegni, mentre i file 3D sono spesso chiamati modelli, parti o assiemi. Comuni estensioni di file CAD includono .DWG, .DXF, .DGN, e .STL [12].

2.3.7 Dati geografici

La categoria GIS (Geographic Information Systems) include i dati riguardanti ai dispositivi GPS e software di mappatura. Questi file possono contenere mappe, percorsi salvati, e punti di riferimento. Molti file GIS che vengono salvati in formati aperti, che possono essere trasferiti tra dispositivi GPS e sistemi informatici. Comuni estensioni di file GIS includono .GPX, .LOC, e .OV2 [12].

2.3.8 Dati WEB

Questa categoria contiene i dati relativi a siti web e server web. Questi includono pagine web statiche e dinamiche, applicazioni Web e file a cui fanno riferimento le pagine web. I file generati dal software di sviluppo Web sono inclusi in questa categoria. Comuni estensioni di file Web possono essere .HTML, .ASP, .PHP e .CSS [12].

2.3.9 Dati non strutturati

I dati non strutturati (o informazioni non strutturate) si riferiscono alle informazioni che non dispongono di un modello predefinito di dati o non si adatta bene in tabelle relazionali. Informazioni non strutturate sono tipicamente testi pesanti, ma può contenere dati come date oppure numeri. Ciò si traduce in irregolarità e ambiguità che rendono difficile capire i dati utilizzando programmi informatici tradizionali, rispetto ai dati memorizzati in form di campi in banche dati o commentate (con tag semantici) nei documenti [13].

2.3 STANDARD DI DATI

Uno standard è una norma tecnica o un requisito sui sistemi tecnici. Di solito è un documento formale che stabilisce un'uniformità nei criteri tecnici, metodi, processi e pratiche di ingegneria [14]. Lo scopo principale degli standard è evitare il problema dell'incompatibilità di fronte ad uno scambio di dati o informazioni. L'esistenza di uno standard pubblicato non implica che sia sempre utile o corretto. Se un elemento è conforme ad un certo standard, non garantisce necessariamente che sia idoneo ad utilizzarlo in ogni caso particolare. Le persone che utilizzano il prodotto o servizio hanno la responsabilità di considerare gli standard disponibili, specificare quello corretto, assicurare la conformità dello standard e usarlo correttamente.

La maggior parte degli standard sono volontari nel senso che essi sono offerti per l'adozione per parte dell'industria volontariamente, senza essere obbligatorio per legge, solo cercano di facilitare il lavoro. Alcune norme diventano obbligatorie quando loro

sono adottate dai regolatori come un requisito legale in settori particolari. Possiamo classificare gli standard come [15]:

Formal standard

Si riferisce specificamente a uno standard che è stato approvato da un organismo di regolazione standard. Esempi di formal standard sono le norme ISO.

De facto standard

È un'abitudine, convenzione, prodotto o sistema che ha raggiunto una posizione dominante da parte dell'opinione pubblica o dalle forze di mercato (per esempio, entrata presta a un mercato) diventando così uno standard, spesso senza essere stato approvato da qualsiasi organizzazione o ricevendo tale approvazione solo dopo che ha già raggiunto una diffusione. Esempi di standard di facto sono la tastiera QWERTY, la suite di protocolli TCP/IP oppure il formato file Adobe PDF.

De jure standard

Sono norme obbligatorie, o per esprimere lo standard volontario dominante quando c'è più di uno standard disponibile per lo stesso uso.

Gli standard internazionali vengono prodotti da diverse organizzazioni, alcuni solo per uso interno, altre per gruppi di persone, gruppi di aziende oppure una sottosezione di un settore industriale. Il problema sorge quando diversi gruppi si incontrano, ognuno con un'ampia base di utenti che hanno fatto qualcosa ben definita ma che tra di loro è incompatibile. Stabilire standard internazionali è un modo per prevenire o superare questo problema. Le principali attività di un'organizzazione di standard sono lo sviluppo, il coordinamento, la promulgazione, la revisione, la modifica, la riemissione e l'interpretazione delle norme progettate per soddisfare le esigenze di un settore particolare. Ci sono molte organizzazioni di standardizzazione nazionali e regionali, in Italia c'è l'Ente Nazionale Italiano di Unificazione (UNI), ma le tre organizzazioni internazionali che hanno il riconoscimento più internazionale sono la International Organization for Standardization (ISO), la International Electrotechnical Commission

(IEC) e la International Telecommunication Union (ITU). Tutte e tre esistono da molto tempo fa (fondate nel 1947, 1906 e 1865, rispettivamente) e tutte e tre hanno la loro sede a Genève, Svizzera. Queste organizzazioni hanno stabilito decine di migliaia di standard che coprono quasi tutti gli argomenti immaginabili. Molti di essi sono stati adottati in tutto il mondo sostituendo a standard “propri” incompatibili. Alcuni di questi standard sono evoluti da quelli progettati internamente in un settore o in un paese particolare, mentre altri sono stati creati da zero da gruppi di esperti che si incontrano in varie commissioni tecniche [16].

Ci sono alcune motivazioni che spingono all'utilizzo degli standard sia dal punto di vista tecnico che del business. Gli standard internazionali sono un modo per superare le barriere tecniche al commercio internazionale causate dalle diverse norme e regolamenti sviluppate separatamente per ogni nazione, organizzazioni nazionali di standard o società. L'utilizzazione di un modello standard sviluppato può permettere a ciascuna industria di integrare più facilmente diversi prodotti software in modo da comporre il sistema di gestione dei dati di prodotto più adatto alle sue esigenze in modo da ottenere la massima flessibilità ed il minimo tempo per passare dalla concezione di un prodotto alla sua produzione e commercializzazione e per produrre rapidamente gli specifici prodotti ordinati dai clienti. I maggiori vantaggi si avranno se le diverse aziende di uno specifico settore industriale concordano sull'utilizzazione di uno specifico standard in modo da ottimizzare il lavoro di integrazione fra i diversi sistemi informatici utilizzati per costituire una squadra maggiormente competitiva sul mercato. Uno standard internazionale può essere utilizzato applicato direttamente o mediante l'adozione dello stesso modificando lo standard internazionale per adattarsi alle condizioni locali. L'adozione consiste nella creazione di standard nazionali equivalenti che sono sostanzialmente gli stessi in contenuto tecnico ma che può avere differenze editoriali in apparenza (come uso di simboli, unità di misura, sostituzioni di caratteri, ecc.) e differenze editoriali derivate da conflitti con le normative governative e/o regolamenti per esigenze specifiche del settore [17].

Passiamo a vedere di seguito gli standard più conosciuti. Abbiamo fatto una divisione tra standard di dati di prodotto e standard di processi.

2.3.1 Standard di dati di prodotto

STEP (ISO 10303)

ISO 10303 è uno standard che fornisce una rappresentazione delle informazioni di un prodotto insieme con i meccanismi necessari e le definizioni per consentire ai dati del prodotto essere scambiati. Il suo titolo ufficiale è: “Sistemi di automatizzazione e integrazione – Rappresentazione e scambio di dati di prodotto”. È conosciuto informalmente come “STEP” (STandard for the Exchange of Product model data). STEP è sviluppato e mantenuto da ISO TC 184 / SC 4, che vuol dire comitato tecnico TC 184 (Sistemi di automazione e integrazione), sottocomitato SC 4 (Dati industriali). È copyright di ISO e non è disponibile gratuitamente [18].

L'obiettivo generale di STEP è di fornire un meccanismo che sia in grado di descrivere dati di prodotto durante il ciclo di vita di un prodotto, indipendente da qualsiasi particolare sistema. La natura di questa descrizione la rende adatta non solo per lo scambio di file neutrali, ma anche come base per l'implementazione e la condivisione di database e l'archiviazione di prodotti. Si cerca di avere un sistema integrato di database di informazioni sul prodotto che sia accessibile e utile a tutte le risorse necessarie a sostenere un prodotto sul suo ciclo di vita. Lo scambio di informazioni avviene tra diversi sistemi informatici e ambienti associati al ciclo di vita di un prodotto completo compreso la progettazione, la fabbricazione, l'utilizzo, la manutenzione e lo smaltimento. Le informazioni generate su un prodotto durante questi processi vengono utilizzate per molti scopi. Questo uso può coinvolgere molti sistemi informatici, tra cui alcuni possono trovarsi in diverse organizzazioni. Al fine di sostenere tali usi le organizzazioni devono essere in grado di rappresentare le loro informazioni sul prodotto in un modo in cui sia interpretabile per tutti e che le informazioni rimangano complete e coerenti quando sono scambiate tra sistemi informatici diversi [19].

STEP è organizzato in una serie di parti, ciascuna pubblicata separatamente. Queste parti rientrano in una delle seguenti serie: metodi di descrizione, risorse integrate, protocolli applicativi, suite di test astratti, modalità di attuazione e test di conformità. STEP usa un linguaggio di specificazione formale, EXPRESS, per specificare le informazioni sul prodotto da essere rappresentato. L'uso di un linguaggio formale

consente precisione e coerenza della rappresentazione e facilita lo sviluppo di implementazioni. STEP utilizza protocolli applicativi (AP) per specificare la rappresentazione di prodotto per una o più applicazioni. Un complemento allo standard STEP che migliora la sua implementabilità e accettazione è che la suite di test astratti e test di conformità devono essere incorporati nello standard [19].

La possibilità di avere un unico archivio dati di prodotto, che serva per tutte le funzioni aziendali, è essenziale per integrare automaticamente tutti i sistemi informatici utilizzati in azienda e quindi sfruttare al meglio i nuovi modelli di organizzazione, che con vari nomi, quali "produzione snella", "logistica integrata", "qualità totale", "concurrent engineering", prevedono tutti l'integrazione delle diverse funzioni, per rispondere con la massima tempestività e flessibilità alla evoluzione del mercato mondiale. La norma internazionale ISO 10303 è stata concepita per soddisfare queste esigenze e permettere quindi di avere tutti i dati di un prodotto in un unico archivio in modo tale da poterli scambiare senza perdita di integrità e completezza fra diversi utenti (progettisti, produttori, utilizzatori) durante tutto il ciclo di vita del prodotto. Lo standard STEP non corrisponde al concetto tradizionale in cui per "norma" si intende un "insieme di regole per uniformare le caratteristiche di un prodotto o di un materiale". Questa norma è un mezzo studiato per ottenere che sistemi informatici diversi si possano comunicare e scambiarsi informazioni fra loro [20].

STEP NC (ISO 14649)

STEP NC è uno standard che è espressamente concentrato nelle fasi di progettazione e produzione del ciclo di vita del prodotto. In particolare, come le sigle NC (Numerical Control) indicano, si è focalizzata sull'integrazione tra le attività svolte durante la fase di progettazione, cioè la costruzione di CAD/CAM modelli, e quelle direttamente eseguite a livello di reparti, dalle macchine utensile a controllo numerico.

Questo standard è il risultato di uno sforzo internazionale da dieci anni per creare un linguaggio moderno associativo che collega i dati di progettazione CAD utilizzate per determinare esigenze di lavorazione per un'operazione con i dati di processo CAM che risolva questi requisiti [21].

STEP NC consente alle aziende manifatturiere di condividere le informazioni di meccanizzato e misure tra le macchine e su internet. L'utilizzo di questo standard sta apportando un miglioramento nei processi, includendo una riduzione del 15% dei tempi di lavorazione. La misurazione automatica e la compensazione attivata per STEP NC si impegna a produrre parti che soddisfanno i requisiti di precisione per costi inferiori. La simulazione integrata e la verifica attivata invece forniscono la garanzia che ogni parte sarà effettuata in modo corretto e che la produzione si fermerà quando si arriva al punto di non soddisfare i requisiti di progettazione [21].

STEP NC cambia il modo in cui viene fatta a fabbricazione definendo i dati come “fasi di lavorazione”: una libreria di operazioni specifiche che possono essere eseguite su una macchina a controllo numerico. In altre parole, si rompe ogni lavorazione nei passaggi necessari per eseguire l'operazione. Con il concetto di “fasi di lavorazione” il processo produttivo diventa semplificato. Ora, una macchina utensile può ricevere un file con dati STEP NC, sapere cosa significa, e procedere con un'operazione senza istruzioni in più. Non avrà più bisogno di programmare una macchina utensile per ogni singolo pezzo. Il vantaggio della norma va oltre, con queste “fasi di lavorazione” standard tutti i produttori saranno in grado di condividere informazioni in modo affidabile e istantaneamente a qualsiasi parte del mondo [21].

PLCS (ISO 10303-239:2005)

Product Life Cycle Support (PLCS) è un protocollo di applicazione della norma ISO 10303 (STEP). È nato come un'iniziativa supportata dalla industria e dei governi nazionali con l'obiettivo di accelerare lo sviluppo di nuovi standard per l'informazione di supporto di un prodotto. Lo sviluppo dei PLCS è gestito da OASIS (Organization for the Advancement of Structured Information Standards), che è un'organizzazione internazionale non-profit che guida lo sviluppo, la convergenza e l'adozione di standard di e-business. PLCS è destinato ad essere in grado di descrivere i prodotti nel suo intero ciclo di vita (progettazione, produzione, utilizzo e end-of-life), col particolare enfasi sul supporto e lavori richiesti per sostenere e mantener i prodotto nelle condizioni operative.

ISO 10303 AP 239 (PLCS) è stato sviluppato per soddisfare i bisogni dei governi, fabbricanti di apparecchiature e/o prodotti originali, operatori e fornitori di servizi esterni. Ciò che rende interessante PLCS a questa comunità è il supporto per:

- Prodotti complessi ad alto valore
- Molti elementi unici e configurazioni di prodotto
- Lunga vita utile del prodotto
- Esigere in-service i requisiti di supporto
- Costi di supporto in-service che comprendono una parte significativa del costo totale della proprietà

Poiché PLCS è stato sviluppato intorno a STEP, è facile integrare i dati PLCS e le applicazioni all'interno dei sistemi software complessi ed eterogenei, raggiungendo un elevato grado di interoperabilità. PLCS condivide la stessa interfaccia comune di altri software basati su STEP.

PLCS è progettato per essere flessibile e può essere adattato in un contesto di business diverso. Questi adattamenti sono chiamati Business DEX (Data Exchange). Un DEX è un modulo indipendente funzionale che si basa su diverse funzionalità. Queste funzionalità utilizzano diversi modelli come le funzioni che hanno parametri input/output fatti di entità STEP o dati basilari [22].

MANDATE (ISO 15531)

MANDATE (MANufacturing DATa Exchange) è una parte della serie di standard sviluppati dal TC 184 / SC 4. È stato in sviluppo per un periodo abbastanza lungo, ma ci sono solo alcuni risultati facilmente accessibili. L'intenzione principale degli sviluppatori di questo standard è ancora una volta fornire funzionalità per la descrizione e la gestione dei dati di prodotto in tutta la parte di vita di un prodotto riguardante alla sua produzione.

MANDATE cerca di definire una rappresentazione comune per tutte le informazioni in fabbricazione. Lo scopo di questo standard è la rappresentazione di informazioni di produzione e informazioni di risorse tra cui la capacità, il monitoraggio, la

manutenzione e il controllo, così come lo scambio e la condivisione di queste informazioni che includono la memorizzazione, il trasferimento, l'accesso e l'archiviazione. STEP prende una visione di fabbricazione product-oriented mentre MANDATE riguarda ai processi dell'organizzazione che sono svolti per produrre prodotti [23].

PLM XML

Questo è uno standard per facilitare un'alta condivisione dei dati del ciclo di vita di un prodotto tra diverse applicazioni. PLM XML è un open standard ed è basato sugli schemi W3C XML. Rappresentando una vasta gamma di dati di prodotto, sia in modo esplicito che tramite riferimenti, PLM XML fornisce un meccanismo leggero, estensibile e flessibile per il trasporto di altri contenuti di dati su internet. Questa norma si concentra soprattutto sulla fase di progettazione BOL del prodotto.

Lo scopo di questo standard è fornire una soluzione ai problemi di interoperabilità tra le organizzazioni che utilizzano applicazioni software per la concezione, lo sviluppo, la fabbricazione e la manutenzione di prodotti, cercando di ridurre i costi che tutto questo suppone ogni anno [24].

ANSI/ISA-95 (ISO 62264)

ANSI/ISA-95 è uno standard internazionale per descrivere le interfacce tra i sistemi di business delle aziende e dei suoi sistemi di controllo della produzione, si concentra quindi principalmente sulla zona corrispondente alla fase di produzione di un prodotto. ISA-95 è costituita da modelli e la terminologia. Questi possono essere usati per determinare quali informazioni devono essere scambiate tra i sistemi per le vendite, le finanze, la logistica e anche quali devono essere scambiate tra i sistemi per la produzione, la manutenzione e la qualità. Queste informazioni sono strutturate in modelli UML, che sono la base per lo sviluppo di interfacce standard tra i sistemi ERP e MES. Lo standard ISA-95 può essere utilizzato per diversi scopi, ad esempio come guida per la definizione dei requisiti dell'utente, per la selezione di fornitori e come base

per lo sviluppo di sistemi MES e database. ANSI/ISA-95, è stata accettata dalla comunità ISO e pubblicato come ISO 62264 [25].

2.3.2 Standard di processi

PSL (ISO 18629)

Il Process Specification Language (PSL) è un linguaggio neutro per le specifiche di processi per servire come lingua di scambio per integrare varie applicazioni relativi ai processi lungo tutto l'arco del ciclo di vita del processo di fabbricazione (dal processo iniziale di concezione al ritiro del processo).

Il componente principale del PSL è un'ontologia progettata per rappresentare i concetti primitivi che, secondo PSL, sono sufficienti per descrivere processi di produzione di base, ingegneria e processi di business. Si noti che il focus di un'ontologia non è solo sui termini, ma anche sul loro significato.

Il PSL Core è un insieme di assiomi scritti nel linguaggio di base di PSL. Gli assiomi fondamentali PSL forniscono una rappresentazione sintattica dei modelli PSL. Lo scopo di PSL Core è quello di assiomatizzare una serie di semantici intuitivi che sono adeguati per descrivere i processi di base.

Le estensioni sono le terze componenti di PSL. Un'estensione di PSL da le risorse necessarie per esprimere informazioni che coinvolgono concetti che non fanno parte di PSL Core. PSL Core è una teoria relativamente semplice che è adeguata per descrivere una vasta gamma di processi di base. Tuttavia, i processi più complessi richiedono risorse espressive che eccedono quelli di PSL Core [26].

ISO 14224

Questa norma fornisce una base completa per la affidabilità e la manutenzione di dati in un formato standard per le apparecchiature in tutti gli impianti e le operazioni all'interno delle industrie petrolifere del gas naturale e petrolchimiche durante il ciclo di

vita operativa delle apparecchiature. Si descrivono i principi di raccolta dei dati e anche termini e definizioni relativi che costituiscono un “linguaggio di affidabilità” che può essere utile per comunicare esperienza operativa. ISO 14224 descrive anche la qualità dei dati di controllo e le pratiche di garanzia per fornire una guida per l'utente [27].

La standardizzazione di queste pratiche di raccolta di dati facilita lo scambio di informazioni tra le parti (impianti, proprietari, costruttori, ecc). ISO 14224 stabilisce i requisiti che ogni sistema di dati RM (reliability and maintenance), interno o commercialmente disponibile, necessita per soddisfare le esigenze del sistema quando si progetta qualcosa che sarà scambiata attraverso questo sistema RM [27].

ISO 14224 consiglia una quantità minima di dati che vengono richiesti per essere raccolti e si concentra su due questioni principali: requisiti relativi ai tipi di dati da essere raccolti per l'utilizzo in diverse metodologie di analisi e formati di dati standardizzati per facilitare lo scambio di dati di affidabilità e dati di manutenzione tra gli impianti, proprietari, costruttori e appaltatori. I dati che sono raccolti sono di attrezzature, di guasti, di manutenzione, ecc [27].

Con l'uso di questo standard le organizzazioni possono creare una piattaforma comune per la raccolta di dati RM, e fornire dati precisi per aiutare le aziende a quantificare una riduzione del rischio. Con un approccio standardizzato e completo per la raccolta dei dati si può essere in grado di garantire un'organizzazione che è maggiormente in grado di prevenire e controllare i potenziali rischi per tutto il ciclo di vita di qualsiasi apparecchiatura di processo [28].

ISO 15926

La norma ISO 15926 è uno standard per l'integrazione, la condivisione, lo scambio e la consegna dei dati tra i sistemi informatici. Il suo titolo è “Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities” [29]. ISO 15926 è uno standard che facilita la modellazione dei dati e l'interoperabilità con il Semantic Web (RDF + OWL + FOL). Contiene anche una ontologia dei dati di riferimento. Si è sviluppato in origine per la industria del petrolio e

del gas, ma è stato impostato genericamente in modo che possa essere utilizzato per qualsiasi tipo di scambio di dati [30].

ISO 15926 è composto di diverse parti, ciascuna di esse riguardante ad un tema diverso: gestione di impianti, modello di dati, geometria e topologia, reference data, metodi di implementazione dei sistemi di integrazione, ecc [29].

ISO/IEC TR 24774

ISO / IEC TR 24774 presenta le linee guida per gli elementi utilizzati più frequentemente per descrivere un processo (titolo, dichiarazione degli obiettivi, risultati, attività e compiti). Sebbene lo scopo primario della norma è quello di favorire la coerenza nei modelli standard di riferimento dei processi, le linee guida che fornisce possono essere applicate a qualsiasi modello di processo sviluppato per qualsiasi scopo [31].

Un numero crescente di standard industriali, nazionali e internazionali descrivono i modelli di processo. Questi modelli sono stati sviluppati per una serie di scopi, includendo implementazione e valutazione di processi. I termini e descrizione da utilizzare in tali modelli variano per forma, contenuto e livello di prescrizione, Questo standard è destinato a superare questa incoerenza [31].

MIMOSA

MIMOSA è un'alleanza di Operations & Maintenance (O&M), fornitore di soluzioni che si concentrano sullo sviluppo di standard di dati aperti basati sul consenso per abilitare Open Standards-based O&M Interoperability [32].

MIMOSA è un'organizzazione senza fini di lucro dedicata allo sviluppo e l'incoraggiamento dell'adozione di standard aperti di informazione in ambienti di produzione. Questi standard di informazione consentono la gestione collaborativa del ciclo di vita delle risorse. I membri di MIMOSA provengono di società di processi di

produzione, aziende di gestione di impianti, organizzazioni militari, capital equipment OEMs e fornitori di sistemi software di gestione delle risorse [32].

MIMOSA fornisce una serie di standard informativi interconnessi. Il Common Conceptual Object Model (CCOM) fornisce una base per tutti gli standard MIMOSA mentre il Common Relational Information Schema (CRIS) fornisce un mezzo per memorizzare le informazioni aziendali. MIMOSA fornisce anche librerie di riferimento di metadati e una serie di norme per lo scambio di informazioni usando XML e SQL [32].

Inoltre, MIMOSA mantiene forti legami industriali con altri gruppi formali standard. Ad esempio, le norme MIMOSA sono compatibili e costituiscono il riferimento informativo per lo standard ISO 13374-1 [32].

3

OPEN SOURCE

L'idea di open source è incentrata sul presupposto che la condivisione del codice fornisce un programma risultante che tende ad essere di una qualità superiore a quelli dei software di proprietario [33]. In un mondo ideale potremmo supporre che le aziende che non spendono soldi su un software di proprietario stanno attuando con saggezza. Un rapporto da parte degli stati Standish Group dice che l'adozione di modelli open source ha portato a un risparmio di circa di 60 miliardi di dollari all'anno per i consumatori [34].

I programmatori che supportano la filosofia del movimento open source contribuiscono con la comunità open source presentando volontariamente la scrittura e lo scambio di codice di programmazione per lo sviluppo software. Il termine "open source" richiede che nessuno può discriminare un gruppo o non condividere il codice modificato o impedire ad altri di modificare il loro lavoro già modificato. Quest'approccio allo sviluppo del software permette a chiunque di ottenere e modificare il codice open source. Queste modifiche vengono distribuite agli sviluppatori all'interno della comunità open source di persone che stanno lavorando con il software. In questo modo le identità di tutti gli individui partecipanti nella modifica del codice sono comunicate e la trasformazione del codice viene documentata nel tempo. Questo metodo rende difficile stabilire la proprietà di un bit di codice, ma è in linea con la filosofia del

movimento open source. Questi obiettivi promuovono la produzione di “programmi di alta qualità” così come lavorare in modo cooperativo con altre persone che pensano allo stesso modo per migliorare le tecnologie open source [35].

3.1 COS'È IL SOFTWARE OPEN SOURCE

Il Software Open Source (OSS) è un software che è disponibile in formato di codice sorgente. Il codice sorgente e alcuni altri diritti che normalmente sono riservati ai possessori del copyright sono forniti sotto una licenza open source che permette agli utenti di studiare, modificare, migliorare e, a volte, anche distribuire il software. Il software open source è molto spesso sviluppato in maniera pubblica e collaborativa. Il software open source è l'esempio più prominente di sviluppo di open source e spesso è paragonato ai contenuti generati per gli utenti [36].

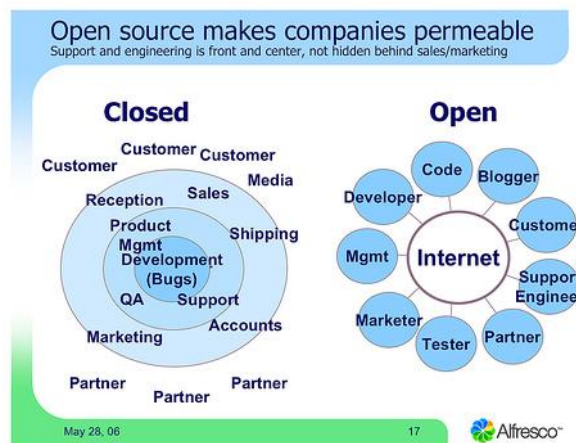


Figura 2: Confronto Software Open Source – Software Chiusi

L'Open Source Initiative (OSI) è una società pubblica benefica californiana, fondata nel 1998. L'OSI è l'assistente dell'Open Source Definition (OSD) ed organismo riconosciuto dalla comunità per la revisione e l'approvazione delle licenze rispetto alla conformità OSD. L'OSI è attivamente coinvolta nella costruzione e la formazione della comunità open source. I membri del consiglio direttivo della OSI viaggiano per il mondo per partecipare alle conferenze ed agli eventi open source, per incontrare

sviluppatori ed utenti dell'open source, e per discutere con i dirigenti dei settori pubblici e privati su come le tecnologie open source, le licenze ed i modelli di sviluppo possano portare vantaggi economici e strategici [37].

Vediamo di seguito i criteri che devono rispettare i software open source in termini di distribuzione. Open source non significa soltanto l'accesso al codice sorgente [37]:

- 1. Ridistribuzione gratuita:** La licenza non può porre restrizioni a nessuno a vendere o donare il software come una componente di una distribuzione maggiore che contiene programmi da diversi sorgenti. La licenza non può richiedere diritti o altri pagamenti per tale vendita.
- 2. Codice sorgente:** Il programma deve includere il codice sorgente e deve permettere la distribuzione in codice sorgente, oltre che in modo compilato. Nei casi in cui qualche prodotto non viene distribuito con il codice sorgente, deve avere un modo ben pubblicizzato di ottenere questo codice per non più di un ragionevole costo di riproduzione di preferenza, scaricato via internet gratuitamente. Il codice sorgente deve essere la forma preferenziale nella quale un programmatore modificherà il programma. Un codice sorgente deliberatamente nascosto non è ammesso. Forme intermedie come l'output di un preprocessore o un traduttore non sono ammesse.
- 3. Lavori derivati:** La licenza deve permettere modifiche e lavori derivati, e deve anche permettere loro di essere distribuiti sotto gli stessi termini della licenza software originale.
- 4. Integrità del codice sorgente dell'autore:** La licenza può impedire che il codice sorgente venga distribuito solo se la licenza consente la distribuzione di "file di correzione" con il codice sorgente con lo scopo di modificare il programma in fase di compilazione. La licenza deve permettere esplicitamente la distribuzione di software costruito da codice sorgente modificato. La licenza può richiedere che i lavori derivati portino un nome e numero di versione diversi dal software originale.

5. **Nessuna discriminazione di persone o gruppi:** La licenza non può discriminare nessuna persona o gruppo di persone.
6. **Nessuna discriminazione di campi di utilizzo:** La licenza non deve impedire nessuno di fare uso del programma in uno specifico campo di impiego. Per esempio non può impedire che un programma venga utilizzato in business o in ricerca genetica.
7. **Distribuzione della licenza:** I diritti collegati al programma devono applicarsi a tutti coloro a cui viene ridistribuito il programma senza necessità di una licenza supplementare per l'esecuzione del programma.
8. **La licenza non deve essere specifica per un prodotto:** I diritti relativi a un programma non devono dipendere dal fatto di essere parte del programma di una particolare distribuzione software. Se il programma è estratto di quella distribuzione e usato o distribuito all'interno dei termini della licenza del programma, tutte le parti a cui il programma viene ridistribuito devono avere gli stessi diritti di quelli che vengono garantiti in unione alla distribuzione del software originale.
9. **La licenza non deve limitare altri software:** La licenza non deve porre restrizioni su altro software che venga distribuito insieme con il software in licenza. Per esempio, la licenza non deve insistere sul fatto che tutti gli altri programmi distribuiti sullo stesso supporto devono essere software open source.
10. **La licenza deve essere tecnologia neutrale:** Nessuna disposizione della licenza può essere basata su una particolare tecnologia o tipo di interfaccia.

3.2 CONFRONTO OPEN SOURCE / FREE SOFTWARE

I termini Free Software e Open Source vengono usati per indicare quella che è sostanzialmente la stessa cosa, ma da punti di vista nettamente diversi.

L'open source è nato alla fine degli anni novanta. Il free software invece, nato negli anni ottanta, riguarda ai software la cui licenza soddisfa una definizione basata sui concetti etici, quali la possibilità di studiare, di aiutare il prossimo o di favorire la comunità. Nello stesso modo che abbiamo visto prima per l'open source, anche il free software ha dei principi base [38]:

1. Libertà di usare il programma senza impedimenti.
2. Libertà di aiutare se stesso studiando il codice disponibile e modificandolo in base alle proprie esigenze.
3. Libertà di aiutare il tuo vicino, cioè la possibilità di distribuire copie del software rielaborato, rendendolo così accessibile a tutti.
4. Libertà di pubblicare una versione modificata del software.

I sostenitori del free software affermano che sia da preferire in quanto pone il dovuto accento sulle questioni morali che stanno alla base di questo tipo di licenze, opponendosi all'uso del software open source avrebbe minore efficacia in tale senso favorendo tra l'altro fraintendimenti del concetto anche guidati da aziende in cattiva fede. I sostenitori del software open source, al contrario, dichiarano che l'espressione "Free Software", dal doppio significato di "Software Libero" e di "Software Gratuito" sia controproducente dal punto di vista della sua diffusione fuori del ambito di essere un hobby o al massimo del ambito universitario. I sostenitori dell'open source, in pratica, pongono l'accento sui vantaggi pratici della diffusione del codice sorgente e dello sviluppo cooperativo su internet del software, interessandosi in maniera minore, se non nulla, dell'aspetto etico della questione.

Nonostante le differenze di forma, è molto difficile che una licenza possa soddisfare i requisiti per una definizione, ma non per l'altra. In particolare l'open source non è necessariamente copyleft, ma può avere licenze permissive esattamente come il free software; non basta avere la possibilità di vedere e modificare per proprio uso i sorgenti di un programma perché questo sia considerabile free software. La confusione riguardo a questi due termini è molto diffusa. Senza dubbio entrambi parlano del software libero

ma le differenze dal punto di vista etico sono importanti. Di seguito vediamo tre diversi confronti: uno storico, uno ideologico e uno riguardo alle licenze [39].

3.2.1 Paragone storico

Per Richard Stallman la società viene vista come una comunità che si nutre della libertà di interazione e collaborazione tra le persone che la compongono. Proprio il concetto di libertà è alla base dell'azione di Stallman e porta alla nascita nel 1985 della Free Software Foundation (organizzazione no profit). L'ambiguità del termine inglese free crea non pochi problemi alla collocazione del movimento: la libertà che esso vuole sottolineare viene spesso confusa con la gratuità, cioè l'assenza di profitto. Stallman è stato il primo a considerare le diverse possibilità di guadagno che il software libero offre: società di consulenza potevano garantire un servizio efficiente e a modici prezzi. Con la nascita di Linux risulta evidente la possibilità di sfruttare tale creazione a livello economico. Proprio da ciò nasce la distinzione tra free software e open source: il movimento free software mette l'accento sulla libertà come condizione essenziale; questa etica così forte ha avuto lontano gli investimenti delle aziende che hanno visto il movimento come eccessivamente ideologizzato.

Il movimento open source risulta invece più disponibile a compromessi di tipo teorico, in favore di uno sviluppo pratico. La commercializzazione viene vista come chiave vincente per un continuo miglioramento del software di partenza. La libertà è quindi professata anche nell'open source, ma viene spesso sacrificata in nome di un utilizzo pratico dei prodotti. Il nuovo termine viene dunque coniato per evitare i problemi precedentemente spiegati riguardo alla parola free ed attirare così gli investimenti di diverse aziende. Con questa nuova immagine il software liberò è finalmente pronto a sfidare il software proprietario sul mercato [39].

3.2.2 Paragone ideologico

Per questo paragona facciamo riferimento ai principi base che abbiamo visto prima sia per l'open source che per il free software.

La differenza fondamentale tra i due movimenti sta nei loro valori, nel loro modo di guardare il mondo. Per il movimento Open Source, il fatto che il software debba essere Open Source o meno è un problema pratico, non un problema etico. Come si è espresso qualcuno, "l'Open Source è una metodologia di sviluppo; il Software Libero è un movimento di carattere sociale". Per il movimento Open Source, il software non libero è una soluzione non ottimale. Per il movimento del Software Libero, il software non libero è un problema sociale e il software libero è la soluzione [40].

3.2.3 Paragone sulle licenze

Per quanto riguarda il free software una delle licenze utilizzata è la GNU Public License (GPL). Questa licenza, ideata da Stallman, mira a rispettare i principi precedentemente esposti, introducendo il copyleft per garantirli. Con il termine copyleft si intende quel meccanismo per cui dopo un'elaborazione di un software libero è vietato ridistribuirlo come software proprietario ed è inoltre necessario segnalare le modifiche attuate all'autore del software. La motivazione è ovvia: alcuni potrebbero approfittarne, spezzando così la catena creativa che ha dato vita al progetto. Questa licenza permette agli autori di mettere la propria proprietà intellettuale al servizio della comunità. In seguito si osservò che non tutti gli sviluppatori erano soddisfatti da questa licenza, e si è scritta un'altra licenza, la LGPL. La LGPL permette di inserire codice in programmi liberi o proprietari, senza che questo influisca sulla licenza finale del software. Queste licenze limitano il pubblico dominio (possibilità di vendere modifiche apportate ad un software libero). Le licenze più comuni riconosciute come in grado di soddisfare i principi del Software Libero sono Mozilla Public License (MPL), Berkeley Software Distribution (BSD).

Sia la Open Source Initiative (OSI) che la Free Software Foundation (FSF) hanno valutato con propri criteri le licenze utilizzate per la distribuzione di software. Non tutte le licenze approvate esplicitamente dalla OSI sono considerate libere (free) dalla FSF, la quale considera libere e persino compatibili con la GPL alcune licenze non approvate dalla OSI. Quest'ultima differenza è dovuta verosimilmente al fatto che le licenze vengono approvate dalla OSI dietro richiesta di chi le ha scritte, mentre il giudizio sulle licenze non scritte dalla FSF è stato dato da questa su propria iniziativa, per fare

chiarimento rispetto alla compatibilità con la propria licenza GPL e anche per ribadire il proprio concetto di "libero" [39].

Ad oggi il termino Open Source appare usato molto più di quello Software Libero. Ad esempio, una grande azienda come la Sun Microsystems, preferisce usare il termine open source. Una motivazione importante può essere che in ambito anglofono il concetto di Libertà è esprimibile solo con la parola Free, parola che significa anche gratis. Questa possibile confusione sembra contro-produttiva per alcune imprese [39].

3.3 MODELLO DI BUSINESS

Ci sono diversi tipi di modelli di business per fare profitti utilizzando un software open source.

Il software open source può essere venduto e utilizzato commercialmente. Si tratta di una parte dell'industria del software. La redditività finanziaria di un software open source può anche provenire dalla vendita di servizi, quali formazione e supporto, piuttosto che il software stesso. L'uso di una doppia licenza prevede l'offerta del software sotto una licenza open source, ma anche nell'ambito di altri termini di licenza di proprietà. I clienti possono essere attratti da un'edizione di origine senza costo e aperta, e poi far parte di un up-sell a un'edizione dell'impresa commerciale.

Inoltre i clienti impareranno del software open source di una società e saranno in grado di generare business in altri prodotti proprietari e soluzioni commerciali. Un'altra possibilità è offrire il software open source solo in forma di codice sorgente, fornendo al contempo i file binari soltanto ai clienti che pagano.

Alcune aziende forniscono l'ultima versione disponibile solo ai clienti paganti. LE aziende forniscono estensioni di proprietario, moduli, plug-in o complementi per un pacchetto open source. Gli sviluppatori indipendenti spesso accettano donazioni. Ad esempio ForgeForge (un deposito di codice sorgente basato nel web), consente agli utenti di donare soldi per progetti realizzati che hanno scelto di accettare donazioni. Gli

utenti di un software possono venire insieme e fare una donazione per l'implementazione di una funzione o una funzionalità desiderata.

Altre situazioni finanziarie includono collaborazioni con altre aziende. A volte una versione commerciale può essere venduta per finanziare lo sviluppo continuo della versione gratuita.

Un'altra possibilità può essere vendere abbonamenti per l'account online e accedere al server. Questa opzione combina software più servizi.

I governi, le aziende o altre organizzazioni non governative possono sviluppare internamente o assumere un imprenditore per la personalizzazione interna di un software, e poi rilasciare il codice sotto una licenza open source.

Esiste una modalità ibrida tra il open e il close software, consiste in aggiungere un closed source a un progetto software senza permesso d'autore e vendere il software risultante. Dopo di un periodo fisso di tempo i patch vengono rilasciati sotto la licenza stessa come il resto del codice di base [41].

3.4 VANTAGGI / SVANTAGGI DELL'OPEN SOURCE

Esperti e ricercatori di software open source hanno individuato diversi vantaggi e svantaggi.

Il vantaggio principale per le imprese è che l'open source è un buon modo per le imprese per raggiungere una maggiore penetrazione nel mercato. Le aziende che offrono software open source sono in grado di stabilire uno standard del settore e, quindi, ottenere un vantaggio competitivo. Contribuisce anche ad avere fedeltà degli sviluppatori, questi si sentono responsabilizzati e hanno un senso di proprietà sul prodotto finale. Inoltre sono necessari meno costi di marketing e servizi logistici per software open source. Aiuta anche alle imprese ad essere aggiornate su tutti gli avanzamenti tecnologici. Si tratta di un ottimo strumento per promuovere l'immagine di una società, compresi i suoi prodotti commerciali. L'approccio dello sviluppo OSS ha contribuito a creare in modo veloce ed economico un software di alta qualità e affidabile. L'open source offre il potenziale per una tecnologia più flessibile e un'innovazione più rapida.

Si dice che è più affidabile poiché normalmente ha migliaia di programmatori indipendenti tentando di risolvere eventuali errori del software. E' Flessibile perché i sistemi modulari consentono ai programmatori di creare interfacce personalizzate o aggiungere nuove competenze ed è innovativa in quanto i programmi open source sono il prodotto della collaborazione tra un gran numero di programmatori diversi. Il software open source può essere sviluppato d'accordo con le esigenze puramente tecniche. Non richiede pensare alle pressioni commerciali che spesso degradano la qualità del software. Queste pressioni rendono gli sviluppatori di software tradizionali prestare più attenzione alle esigenze del cliente che ai requisiti di sicurezza.

Si può pensare che a volte il processo di sviluppo di un open source non può essere ben definito. Le fasi del progetto, test del sistema e la documentazione possono essere ignorate. Tuttavia questo è solo vero per i piccoli progetti, i progetti più grandi e di successo fanno definire ed applicare almeno alcune regole di cui hanno bisogno per rendere possibile il lavoro di squadra. Nei progetti più complessi queste regole possono essere rigorose come ad esempio la revisione.

Non tutte le iniziative OSS hanno avuto successo. Ci sono esperti e ricercatori di software che non sono convinti della capacità open source per la produzioni di sistema di qualità. I problemi più importanti sono la scoperta di un errore in ritardo e la mancanza di prove empiriche (dati raccolti relativi alla produttività e alla qualità). E' anche difficile progettare un modello di business attorno al paradigma open source, di conseguenza, solo i requisiti tecnici possono essere soddisfatti e non quelli del mercato. In termini di sicurezza un software open source potrebbe consentire agli hacker di conoscere le debolezze del software più facilmente dei software chiusi. Ci vogliono meccanismi di controllo al fine di creare una performance efficace di agenti autonomi che partecipano nelle organizzazioni virtuali [36].

3.5 EVIDENZA D'ADOZIONE DELL'OS. ESEMPI

L'open source è stato adottato in diversi ambiti come passiamo a vedere si seguito [42]:

Biblioteche: Stanno utilizzando software open source per sviluppare informazioni e servizi di biblioteca. Lo scopo del software open source è di fornire un software che è

più economico, affidabile e ha una maggiore qualità. Le biblioteche in particolare si approfittano di questo movimento a causa delle risorse che offrono. Promuovono anche le stesse idee di apprendimento e la comprensione di nuove informazioni attraverso le risorse di altre persone. Si tratta di un invito per chiunque di fornire informazioni su vari argomenti. Gli strumenti open source consentono alle biblioteche di creare cataloghi basati sul web.

Agenzie governative e software d'infrastruttura: Queste agenzie stanno utilizzando software open source per infrastrutture come il sistema operativo Linux e il server web Apache per gestire le informazioni. In 2005 un nuovo gruppo di governo è stato avviato sotto il nome di National Center of Open Source Policy Research (NCOSPR), un'organizzazione non-profit promuovendo l'uso di software open source all'interno del governo.

Biologia Sintetica: Questa nuova tecnologia è importante poiché permette di attivare a basso costo nuovi farmaci salvavita, così come contribuisce a produrre biocarburanti che possono aiutare a risolvere il nostro problema energetico. Anche se la biologia sintetica non è ancora venuta fuori della sua fase di "laboratorio", questa ha un gran potenziale per diventare industrializzata nel futuro. Al fine di industrializzare la scienza open source, ci sono alcuni scientifici che stanno cercando di costruire il proprio marchio di esso.

Ambiente militare: Open source ha il potenziale per aiutare i militari. Il software open source permette a chiunque di apportare modifiche che lo migliorano, questa è una forma di invito per le persone a mettere le loro menti insieme per far crescere un software in modo economicamente efficiente. La ragione per cui l'esercito è così interessato è perché è possibile che questo software possa aumentare la velocità e la flessibilità. Sebbene ci sono problemi di sicurezza in questa idea dovuto al fatto che chiunque abbia accesso per modificare il software.

Open source in istruzione: Università e organizzazioni stanno utilizzando prevalentemente software online per educare i propri studenti. La tecnologia open source è stata adottata da molte istituzioni, perché può salvare queste istituzioni dal pagamento alle aziende che forniscono loro questi sistemi software amministrativi. Per

gli educatori, il movimento open source consente l'accesso al software che potrebbe essere utilizzato per insegnare agli studenti come applicare le teorie che stavano imparando. Con le reti aperte, gli insegnanti sono in grado di condividere lezioni, conferenze, e materiale didattico di altri all'interno di una comunità. Un esempio può essere l'Open Course Ware Consortium, un'organizzazione composta da varie università che supportano l'open source e condividono una parte del loro materiale online. Quest'organizzazione, guidata dal Massachusetts Institute of Technology, è stata istituita per facilitare lo scambio di materiali didattici open source.

Open source in sanità: Nel giugno 2009 è stato creato il software open source OpenMRS senza scopo di lucro, questo open source è usato per documentare l'assistenza sanitaria in Nigeria. OpenMRS gestisce funzioni quali allertare gli operatori sanitari quando i pazienti mostrano segni di allarme per le condizioni e registra le nascite e le morti ogni giorno, tra altre caratteristiche. Il successo di questo software è causato dalla sua facilità d'uso rispetto a più complessi software di assistenza sanitaria esclusivamente disponibili nei paesi del primo mondo. Questo software è sviluppato dalla comunità e può essere utilizzato liberamente da chiunque, caratteristica di applicazioni open source. Finora, OpenMRS viene utilizzato in Ruanda, Mozambico, Haiti, India, Cina e le Filippine. Grazie al software open source, il settore sanitario ha a disposizione una soluzione gratuita per implementare gli standard dell'assistenza sanitaria. Non solo apporta beneficio economico, ma la dipendenza minore sul software proprietario consente una più facile integrazione di sistemi diversi, indipendentemente dallo sviluppatore.

Di seguito sono riportati esempi particolari di adozione di open source. Questi eventi sono stati sviluppati attraverso la comunità open source e riecheggiano le ideologie del movimento open source [42]:

Wikipedia: Enciclopedia generata dagli utenti in linea. Ci sono progetti associati in aree accademiche, come Wikiversità, una comunità dedicata alla creazione e lo scambio di materiali didattici.

Project Gutenberg: Prima dell'esistenza di Google Scholar Beta, questo era il primo fornitore di libri elettronici e il primo progetto biblioteca libera.

Google: Il motore di ricerca ha aperto la strada della trasformazione delle applicazioni basate sul Web, come ad esempio libri, riviste accademiche, che sono basate principalmente su software open source. Google continua a fare applicazioni basate su software open source.

Microsoft: Prima dell'estate del 2008, Microsoft è stata generalmente conosciuta come un nemico della comunità open source. Microsoft ha inoltre minacciato di Linux che avrebbero fatto pagare diritti d'autore per aver violato 235 dei loro brevetti. Nel 2008 Sam Ramji, l'allora capo della strategia software open source in Microsoft, ha iniziato a lavorare a stretto contatto con Bill Gates per sviluppare un atteggiamento pro-open di origine nel settore del software così come Microsoft stessa. Ramji, prima di lasciare l'azienda nel 2009, costruì familiarità di Microsoft e il coinvolgimento con l'open source, che è evidente nei contributi di Microsoft di codice open source a Windows Azure, un nuovo servizio web per la creazione e l'hosting di applicazioni in rete, tra altri progetti. Questi contributi sarebbero stati inimmaginabili da parte di Microsoft. Microsoft ha recentemente contribuito al progetto Samba, che è essenzialmente una ricreazione di Block Server di Microsoft Message (SMB), utilizzando il codice open source. Questo servizio permette al desktop di Windows essere integrato con i file di Linux. Può essere eseguito su molte piattaforme e mira a rimuovere gli ostacoli all'interoperabilità. Per Microsoft, questo significava che permette alle piattaforme Linux di competere con il proprio software proprietario di Windows per il bene di promuovere il movimento open source. Il cambiamento di atteggiamento di Microsoft su open source e gli sforzi per costruire una comunità più forte open source è la prova della crescente adozione e l'adattamento del software open source.

Linux: Sistema operativo utilizzato prevalentemente nei server Linux, è stato creato da uno studente nel 1991 insieme ad altri sviluppatori di tutto il mondo.

Apache: Un software server leader e linguaggio di scripting sul web.

MySQL: Un sistema di gestione database.

OpenOffice: Pacchetto software con funzionalità di editore di testo, foglio elettronico e di presentazione.

4

ARAS INNOVATOR

Aras Innovator è un prodotto dell'Aras Corporation. Quest'azienda americana sviluppa e edita software per il Product Lifecycle Management (PLM), l'Advanced Product Quality Planning (APQP), il Lean Product Development e il New Product Introduction (NPI) [43].

4.1 SOFTWARE ARAS INNOVATOR

Aras Innovator è un pacchetto di software aziendale per la gestione di complessi processi di business PLM. Il software di Aras che viene eseguito sul contesto è un software open source che è regolato da licenze OSI approvate. Una comunità di codice sorgente è disponibile per gli abbonati a pagamento [43].

Aras offre una vasta gamma di soluzioni software PLM che consentono alle aziende migliorare l'innovazione, la collaborazione e il coordinamento a scala globale. Offre comprensive applicazioni che forniscono una serie di iniziative strategiche di business attraverso l'intero ciclo di vita del prodotto, comprendendo la gestione della programmazione, l'ingegneria di prodotto e la pianificazione della qualità [44].

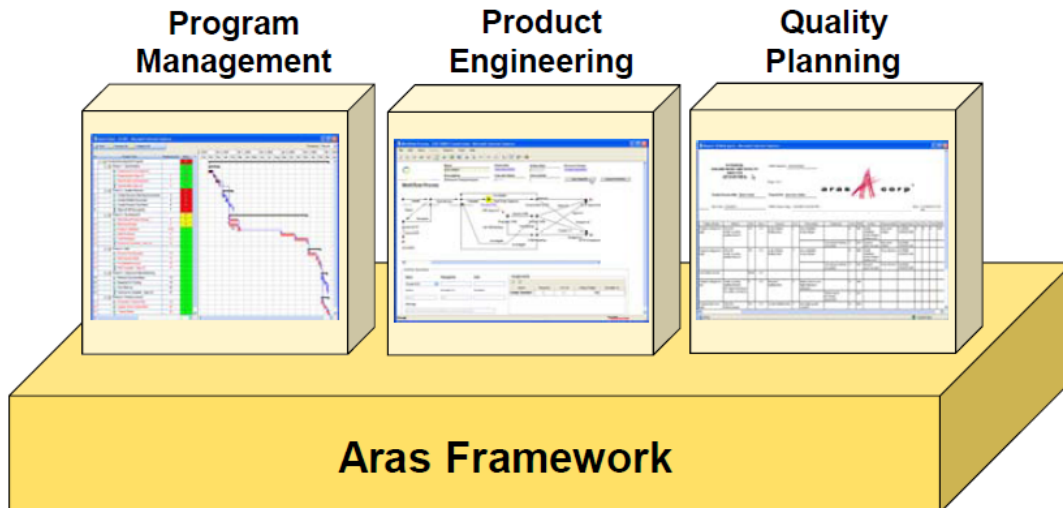


Figura 3: Framework Aras Innovator

L'idea su cui si basa l'azienda è quella secondo cui il tradizionale rapporto cliente/venditore è squilibrato a favore del venditore, che offre un sistema PLM che necessita di mesi per essere implementato, a fronte di un grande investimento iniziale del cliente: per questo Aras ha scelto un approccio diverso, in grado di esaudire le richieste del cliente nel minor tempo possibile.

Si offrono tre tipi di soluzioni software PLM [44]:

1. **Aras Managed Solutions:** Totalmente conforme a OSI e disponibile gratuitamente per essere scaricato sotto un formato aziendale open source. Quindi non ci sono mai le licenze PLM da acquistare.
2. **Aras Community Solutions:** Sviluppato e fornito dai membri della comunità Aras. Disponibile gratuitamente nel sito Project Community dell'Aras.
3. **Aras Commercial Solutions:** Sviluppato da Aras e dei suoi partner. I diritti di licenza possono essere applicati.

Aras è l'unico software PLM con un pacchetto di soluzioni che è alimentato da un avanzato modello di servizio basato su architettura orientata (model-based SOA). E'

anche l'unico fornitore che offre un pacchetto di abbonamento che include tutti gli aggiornamenti e servizi di aggiornamento. Per ultimo, ma certamente non meno importante, nessun altro offre un modello di business aziendale open source, questo è importante perché elimina i costi di licenza PLM.

4.2 ARCHITETTURA

Le soluzioni di Aras si basano sul Framework Aras Enterprise Application, un moderno modello basato su tecnologia SOA (Service Oriented Architecture) progettato per lo sviluppo di applicazioni altamente scalabili, flessibili e sostenibili, che implementano rapidamente e si adattano facilmente alle esigenze aziendali in continua evoluzione, offrendo allo stesso tempo i risultati di business della gestione nel settore al costo più basso [44].

L'Aras Enterprise Application Framework è un avanzato, model-based Service Oriented Architecture (SOA), costruito interamente su standard web aperti e tecnologia standard comprovata. La caratteristica più singolare è l'uso di una tecnologia run-time model-based per definire e gestire gli oggetti di business. Questa è la vera innovazione che ha sviluppato Aras, l'approccio model-based, che è un'architettura di metadati che definisce processi di business all'interno del software aziendale utilizzando modelli grafici a un livello superiore, eseguendo il sistema direttamente dal modello in tempo reale [45].

Nei sistemi "model-driven" è stato creato graficamente con la sintassi UML. Da questo modello tradizionale è generato codice sorgente, che viene poi compilato, legato, testato, messo a punto e, in fine, distribuito. Aras ha preso il concetto di definire processi di business utilizzando modelli grafici ad un livello superiore, eseguendo il sistema direttamente da tali modelli in tempo reale e l'utilizzo di tali modelli per definire e gestire gli oggetti di business. Il risultato è una definizione di sistema molto fluida che si adatta alle esigenze aziendali in continua evoluzione, semplicemente, continuamente e senza mal di testa di tipici paradigmi di personalizzazione delle applicazioni. Il vantaggio dell'approccio "model-based" di Aras sull'approccio "model-driven" è notevole. Le modifiche alle applicazioni aziendali in Aras Innovator sono realizzate in

tempo reale modificando il modello direttamente al contrario delle settimane o mesi che mettono utilizzando i sistemi convenzionali basati su codice compilato, compresi i sistemi “model-driven”. L’approccio “model-based” che usa Aras consente di risparmiare tempo ed elimina l’errore principale che fanno la maggior parte delle aziende con i sistemi “model-driven”, che sarebbe apportare modifiche a livello di codice sorgente e non l’aggiornamento dei modelli originali [46].

Un punto importante che richiede chiarimenti è che Aras Innovator non è l’esecuzione di codice interpretato. L’interpretazione del codice non accade mai perché i modelli stessi sono memorizzati in Aras Innovator in uno schema predefinito che viene utilizzato per una serie di servizi Web in SOA. Questi servizi Web fanno la valutazione dinamica del modello di oggetto di business. Questa differenza è fondamentale per la capacità di Aras Innovator per offrire prestazioni ottimali consentendo nello stesso tempo a una grande flessibilità [45].

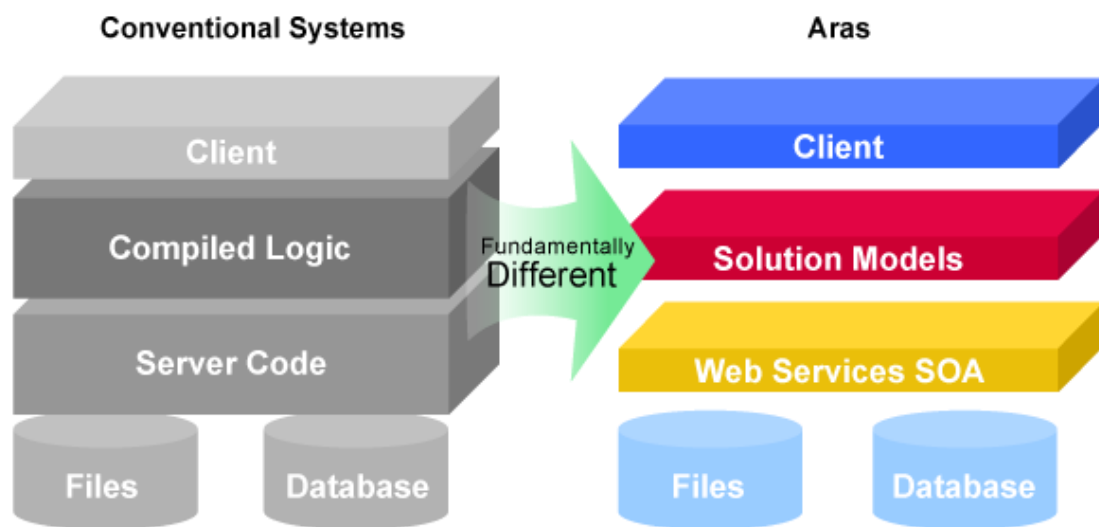


Figura 4: Differenza fondamentale di Aras con altri sistemi convenzionali

I vantaggi di un approccio model-based SOA sono [46]:

- 1. Rapidamente implementabile:** Consente la rapida attuazione di soluzioni partendo da zero oppure partendo da modelli di settore definiti, per un’implementazione aziendale significativamente più veloce.

2. **Facilmente adattabile:** Questo modello rende le soluzioni intuitive a livello commerciale. Diagrammi di flusso e diagrammi di processo definiscono le applicazioni in modelli XML.
3. **Uso e mantenimento senza sforzo:** Il modello è facile da gestire e amministrare. Schermi basati sul browser offrono un look Microsoft e rende le soluzioni Aras Innovator facili da imparare e semplici da usare.
4. **Semplice aggiornamento:** Quando si verificano gli aggiornamenti ai programmi di servizi web, questi programmi vengono sostituiti senza impattare i modelli XML. L'approccio innovativo del software Aras rende più facile che mai per rimanere aggiornati con le più recenti innovazioni tecnologiche.
5. **Totale integrazione:** Il modello rende lo scambio di dati con altri sistemi semplici. Consente l'integrazione d'informazioni provenienti da altri sistemi aziendali senza intervenire direttamente la gestione dei dati.
6. **Alta sicurezza:** Aras è stato progettato da zero per esporre servizi web sicuri per tutti scambi dei dati e l'integrazione dei processi. Gestiona in modo sicuro sia lo stato attuale che la storia della versione. L'accesso sicuro ai dati è garantito tramite autenticazione.

Aras usa modelli di metadati per descrivere tutto: funzionalità applicative, logica di business, schema di dati, schermi, flussi di lavoro, ecc, ecc, ecc... letteralmente tutto. I modelli di metadati definiscono oggetti nel sistema e questi oggetti poi “utilizzano” i servizi di cui loro hanno bisogno.

Per gli architetti alla ricerca di una descrizione generica, Aras Innovator è una architettura orientata ai servizi basata sul web e n-tier, composta da client web, application server, database e file server ed è interamente basata su protocolli Internet standard, tra cui HTTP / HTTPS, XML e SOAP (Simple Object Access Protocol) [45].

Vediamo ora alcune caratteristiche più dettagliatamente:

FLESSIBILITÀ

L'approccio model-based di Aras Innovator permette creare, modificare ed estendere rapidamente le soluzioni aziendali attraverso la combinazione di form, flussi di lavoro, fusioni di file, strutture di relazioni, accesso di sicurezza, e altre numerose applicazioni a livello di business. Form, diagrammi di flusso e diagrammi di processo vengono utilizzati per definire le applicazioni. Un'applicazione viene modificata cambiando un diagramma in un editore di soluzioni grafiche invece di programmando un pezzo di codice. Quando si apporta una modifica nel sistema in esecuzione, viene aggiornata istantaneamente.

Poiché Aras Innovator viene consegnato da un quadro comune, le applicazioni intrinsecamente interagiscono. Ogni applicazione sviluppata con Aras Innovator condivide un insieme coerente di funzionalità e di strutture con il familiare look & feel di Microsoft.

Un altro aspetto importante del model-based di Aras Innovator è che i modelli sono definiti interamente in XML e memorizzati come modelli XML. I modelli XML "sottoscrivono" ai servizi Web SOA in Aras Innovator, una volta che esso è in esecuzione. L'uso di modelli XML per descrivere i modelli che definiscono gli oggetti di business, flussi di lavoro e gli schermi è importante per diverse ragioni. Lavorare con XML è ampiamente conosciuto e i file XML sono facili da manipolare a livello di testo e con gli editor grafici come quelli di Aras Innovator, e più in particolare, i file XML sono portatili. Il model-based semplifica le modifiche [45].

PORTABILITÀ

La portabilità è di particolare importanza perché la condivisione delle applicazioni aziendali è reso facile e veloce. Tutto ciò che serve per spostare un'applicazione tra due diversi sistemi di Aras Innovator è semplicemente esportare il file XML da un sistema e importarlo all'altro sistema. Questa funzionalità è utile per una serie di situazioni quali:

- Migrazione di una logica di business personalizzata per un aggiornamento a una nuova versione della tecnologia SOA di Aras Innovator

- Spostamento di nuove funzionalità da un ambiente di sviluppo a un ambiente di produzione
- Creazione di una nuova applicazione e lo spostamento a un sistema esistente
- Condivisione di un processo di business tra due differenti divisioni aziendali

La portabilità delle applicazioni Aras Innovator è significativa, anche perché le applicazioni ampiamente personalizzate possono essere spostati da un ambiente all'altro in un modo rapido e con il minimo sforzo. In contrasto con le applicazioni aziendali tradizionali che richiedono mesi e talvolta anni di lavoro complesso e costoso per spostarsi creando così notevoli problemi di aggiornamento [45].

AGGIORNAMENTI

A prescindere della quantità di personalizzazione in cui un'applicazione aziendale è stata sottoposta, gli aggiornamenti sono semplici e facili. La combinazione dell'approccio model-based con il SOA risulta in una separazione completa dei sistemi di descrizione e la tecnologia di base. La definizione intera del sistema è descritta nel modello XML. L'archivio permanente dei modelli XML è normalizzato in tabelle e colonne all'interno del database e memorizzati insieme ai dati dell'utente finale. Il codice compilato dei servizi web SOA, il quale definisce i comportamenti e le azioni disponibili per il modello è codice gestito .NET.

Il significato della separazione è importante. Gli aggiornamenti a nuove versioni della tecnologia SOA di Aras Innovator sono semplici comportando solo l'installazione dei file [45].

INTEGRAZIONE

Le interfacce aperte di Aras consentono un'agevole interazione con altri sistemi aziendali. L'integrazione è facilitata da aperti Application Programming Interfaces (API), servizi federativi che legano in profondità l'architettura.

Le API inviano e ricevono messaggi XML / SOAP e pubblicano un dizionario di dati completo fornendo un'interfaccia standard che rende lo scambio di dati con altri sistemi o prodotti d'integrazione semplice.

Le funzionalità federative consentono l'integrazione dei dati provenienti da altri sistemi aziendali senza intervenire direttamente nella gestione dei dati all'interno di Aras Innovator [45].

Aras consente alle aziende di connettersi apertamente alle applicazioni in tutta l'azienda per sfruttare le infrastrutture IT esistenti come ad esempio l'integrazione con altre soluzioni aziendali o sistemi precedenti, come i sistemi ERP (come SAP e Oracle), sistemi MCAD (come CATIA e AutoCad), sistemi EDA (come OrCAD o DXDesigner) e i sistemi PDM / PLM (come Siemens Teamcenter, Dassault ENOVIA, e PTC Windchill). Connettori di sistemi sono disponibili anche per tutti i principali pacchetti, così come i prodotti di Microsoft Office e altri.

MCAD	PDM	EDA	ERP	Others
CATIA	ENOVIA	Allegro	SAP	Msoffice
NX	Agile	OrCAD	Oracle	
Pro/E	Teamcenter	DXDesigner	Infor	
SolidWorks	Windchill	PADS	Dynamics	
SolidEdge		Expendition		
Inventor		Altim		
AutoCAD		CR-5000		

Tabella 1: Integrazione di Aras con altri sistemi

4.3 FUNZIONALITÀ

Il prodotto “Aras Innovator” offre l’intera gamma delle funzionalità di un software PLM. Vediamo di seguito quali sono [47] [48]:

Gestione documentale

La gestione dei documenti in Aras fornisce un deposito sicuro dei file online che garantisce a tutti gli utenti l'accesso a una versione unica della verità in relazione a prodotti, processi e altri file.

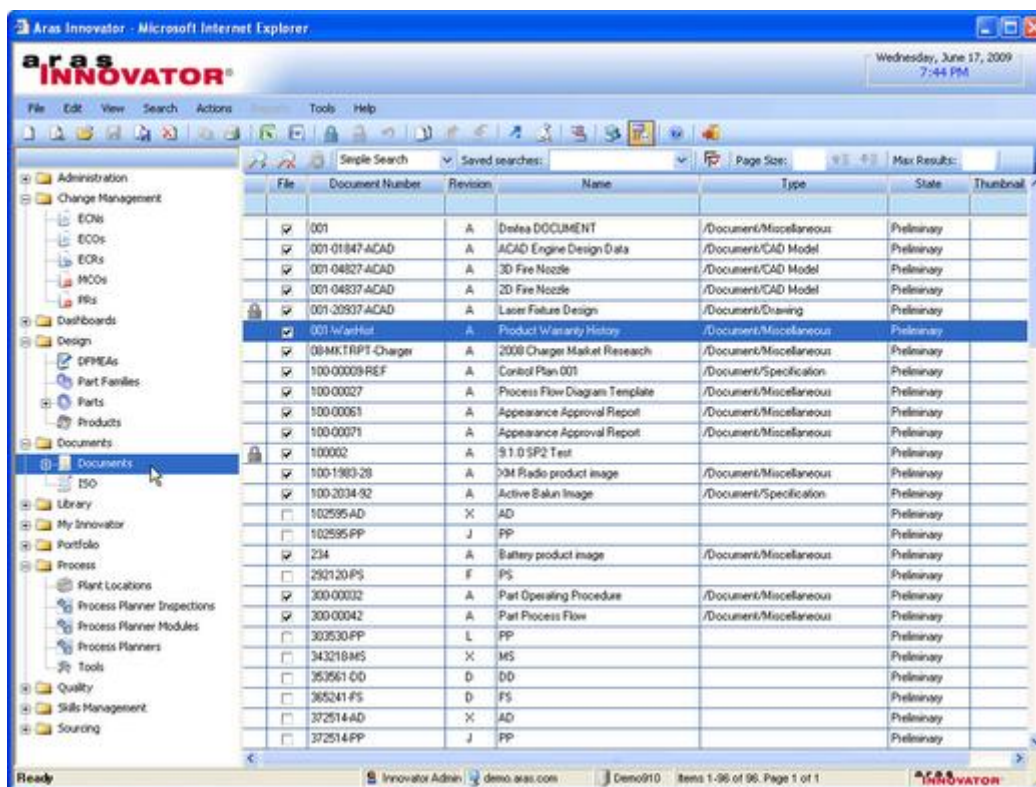
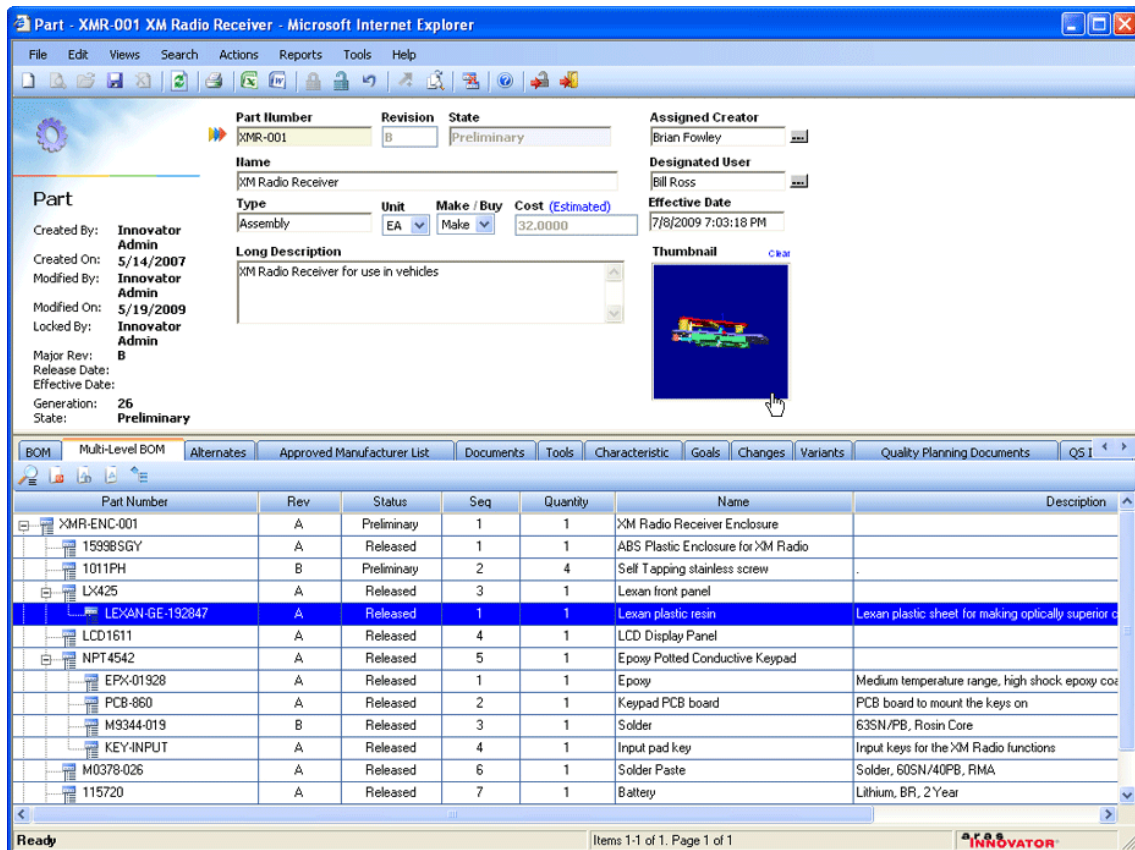


Figura 5: Screenshot Aras Innovator (1/9)

- Gestione documentale per tutti i tipi di documenti e formati di file, tra cui PDF, Word, Excel, disegni, file CAD, TIFF e molti altri.
- Documenti e file sono direttamente associati a ogni progetto.
- Modalità di ricerca semplice e avanzata accompagnata da una terza parte con componenti aggiuntivi per il testo completo, simile a quella di Google.
- Collaborazione online sicura con clienti, fornitori e produttori in outsourcing.
- Revisione completa e controllo delle versioni, controllo delle modifiche.

Gestione BOM (Bill of Materials)

La gestione BOM (distinta dei materiali) fornisce una definizione unica di un prodotto aggregando e collegando tutte le informazioni e gli attributi utilizzati per progettare, produrre e supportare il prodotto in modo strutturato.



The screenshot displays the Aras Innovator software interface for managing a Bill of Materials (BOM). The main window is titled "Part - XMR-001 XM Radio Receiver - Microsoft Internet Explorer". The interface is divided into several sections:

- Part Information:** Fields for Part Number (XMR-001), Revision (B), State (Preliminary), Assigned Creator (Brian Fowley), Designated User (Bill Ross), and Effective Date (7/8/2009 7:03:18 PM).
- Part Details:** Name (XM Radio Receiver), Type (Assembly), Unit (EA), Make/Buy (Make), and Cost (Estimated) (32.0000).
- Long Description:** XM Radio Receiver for use in vehicles.
- Thumbnail:** A small image of the product.
- BOM Table:** A table listing the components of the assembly. The table has columns for Part Number, Rev, Status, Seq, Quantity, Name, and Description.

Part Number	Rev	Status	Seq	Quantity	Name	Description
XMR-ENC-001	A	Preliminary	1	1	XM Radio Receiver Enclosure	
1599BSGY	A	Released	1	1	ABS Plastic Enclosure for XM Radio	
1011PH	B	Preliminary	2	4	Self Tapping stainless screw	
LX425	A	Released	3	1	Lexan front panel	
LEXAN-GE-192847	A	Released	1	1	Lexan plastic resin	Lexan plastic sheet for making optically superior c
LCD1611	A	Released	4	1	LCD Display Panel	
NPT4542	A	Released	5	1	Epoxy Potted Conductive Keypad	
EPX-01928	A	Released	1	1	Epoxy	Medium temperature range, high shock epoxy coe
PCB-860	A	Released	2	1	Keypad PCB board	PCB board to mount the keys on
M9344-019	B	Released	3	1	Solder	63SN/PB, Rosin Core
KEY-INPUT	A	Released	4	1	Input pad key	Input keys for the XM Radio functions
M0378-026	A	Released	6	1	Solder Paste	Solder, 60SN/40PB, RMA
115720	A	Released	7	1	Battery	Lithium, BR, 2 Year

Figura 6: Screenshot Aras Innovator (2/9)

- Garantisce a tutti gli utenti (dipendenti, fornitori e produttori in outsourcing) avere le informazioni giuste in ogni momento.
- Le distinte dei materiali includono visualizzazioni multilivello, elenco delle parti, assiemi, componenti di origine e altro ancora.
- Migliora la qualità del prodotto, la produttività, la collaborazione e il riutilizzo.
- Riduce i costi dei prodotti, gli errori di produzione, scarti e rilavorazioni.

Diagramma di Gantt

Il diagramma di Gantt mostra la pianificazione dei compiti e delle attività, le attività da consegnare, le dipendenze, i percorsi critici, e gli obiettivi. In tempo reale il diagramma di Gantt offre una visione per i project manager e gli utenti finali.

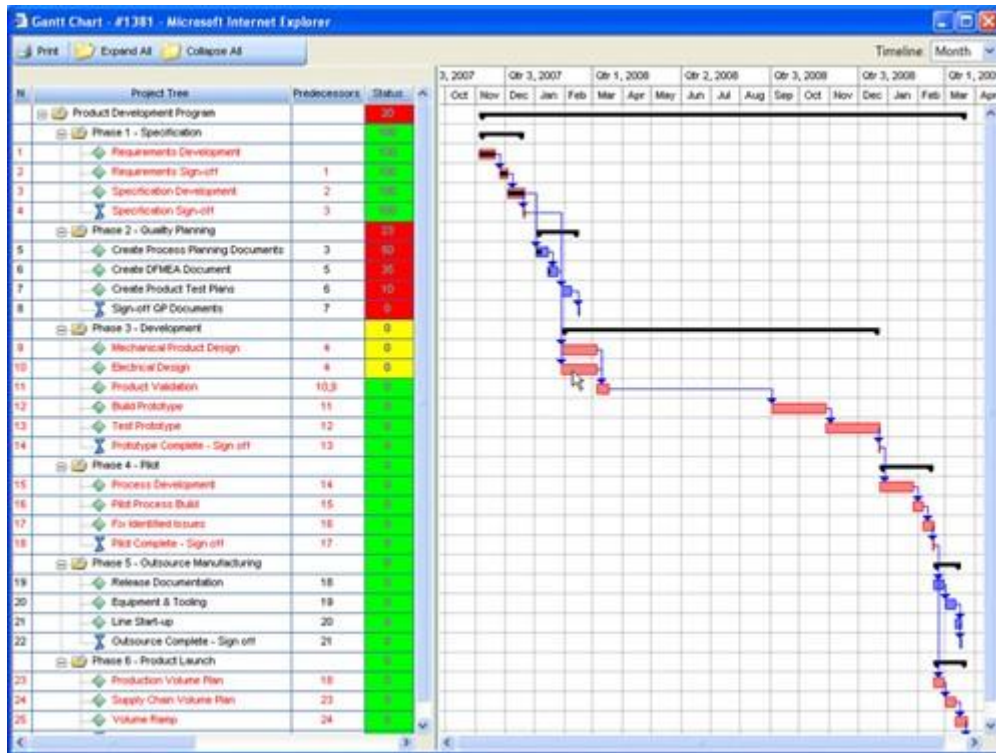


Figura 7: Screenshot Aras Innovator (3/9)

Pianificazione del processo di produzione

La pianificazione del processo di fabbricazione garantisce la precisione delle informazioni e la qualità per i processi di produzione, fornendo una soluzione affidabile, un'unica versione della soluzione in un luogo online sicuro. Gli utenti mettono affidamento su Aras come fonte ufficiale, rilasciando informazioni sui processi produttivi e quelle relative ai dati tecnici.

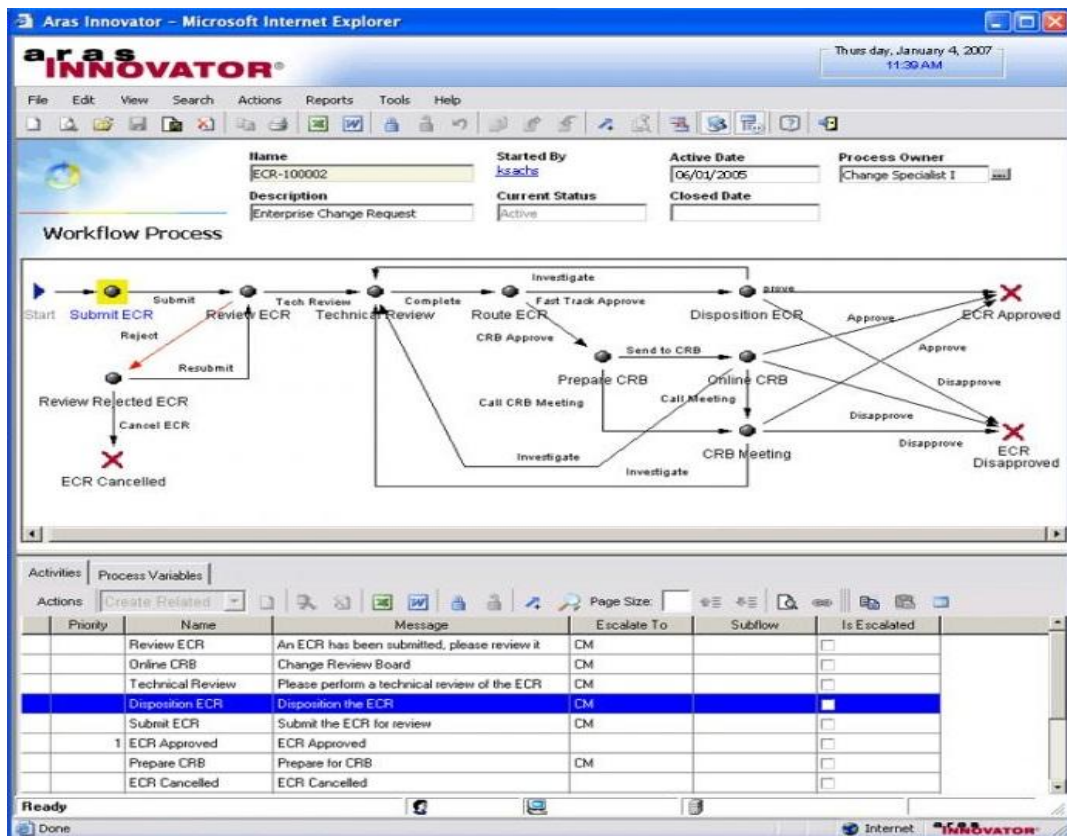


Figura 8: Screenshot Aras Innovator (4/9)

- Automatizza e gestisce tutte le informazioni e i processi necessari per la produzione di parti, l'assemblaggio dei prodotti finali e lo svolgimento dei controlli di qualità del prodotto.
- Fornisce la revisione e il controllo della versione dei piani di processi di produzione, la guida della produzione, le descrizioni dei prodotti, le sequenze delle operazioni, l'utilizzo di strumenti e le istruzioni di impostazione e fasi di esecuzione.
- Assicura l'integrazione con i sistemi ERP delle imprese.

Tracciabilità

Aras permette la tracciabilità dei codici dei fornitori dei componenti, codici seriali e codici di lotto.

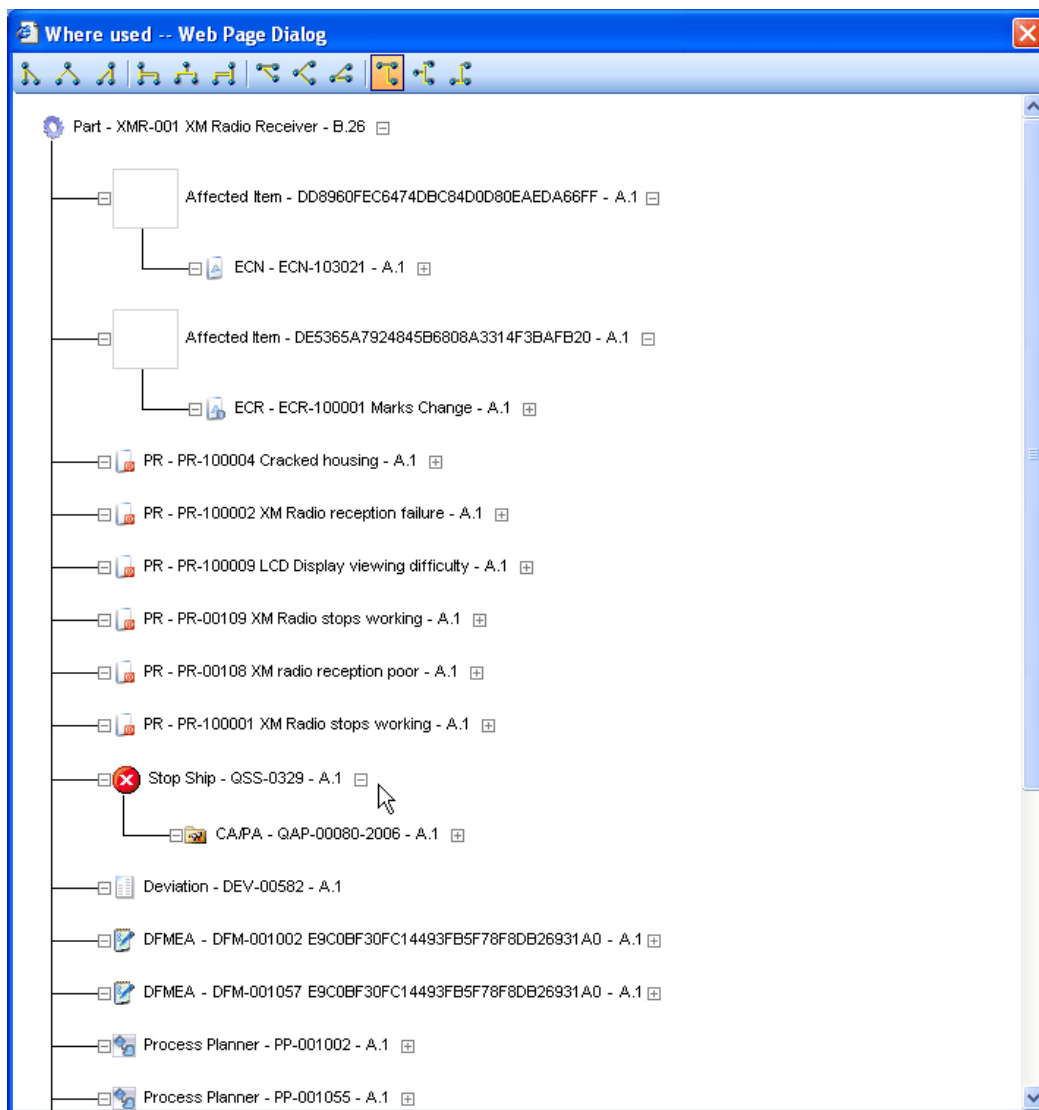


Figura 9: Screenshot Aras Innovator (5/9)

- Identifica le aree problematiche in un modo rapido e semplice, risparmiando un importante tempo e risorse.
- Monitora e gestisce lo storico, l'ispezione, la manutenzione e altre informazioni critiche.
- Migliora gli standard e la conformità con la normativa.

Confronto della struttura delle parti

Una grafica confronta due versioni della stessa parte oppure confronta assemblaggi diversi. La linea di confronto fornisce in tempo reale una rappresentazione visuale, si può fare un semplice doppio clic per analizzare le modifiche nella struttura del prodotto.

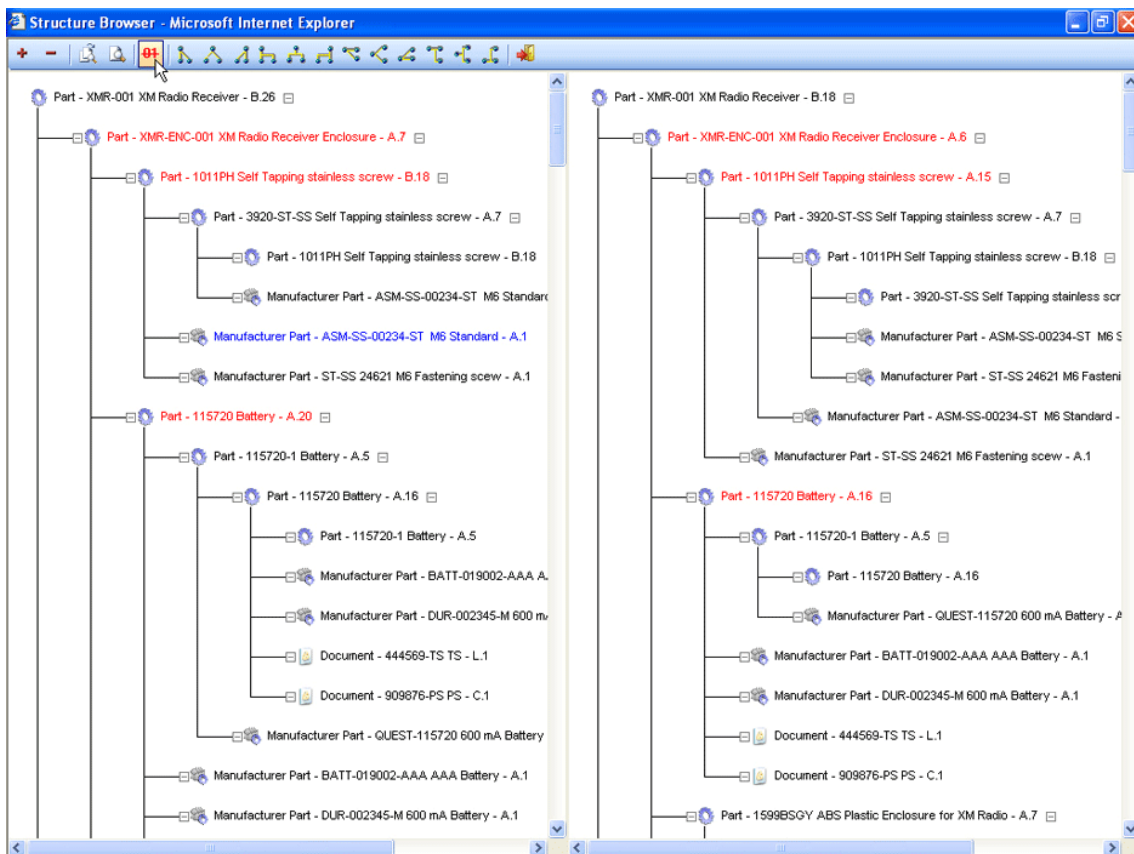


Figura 10: Screenshot Aras Innovator (6/9)

PDM

Il PDM (Product Data Management) in Aras permette alle aziende di monitorare e gestire la creazione, la modifica e l'archiviazione di tutte le informazioni relative ad un dato prodotto, includendo anche i modelli CAD, i disegni, le specifiche e altri documenti associati in un unico e centralizzato deposito di dati.

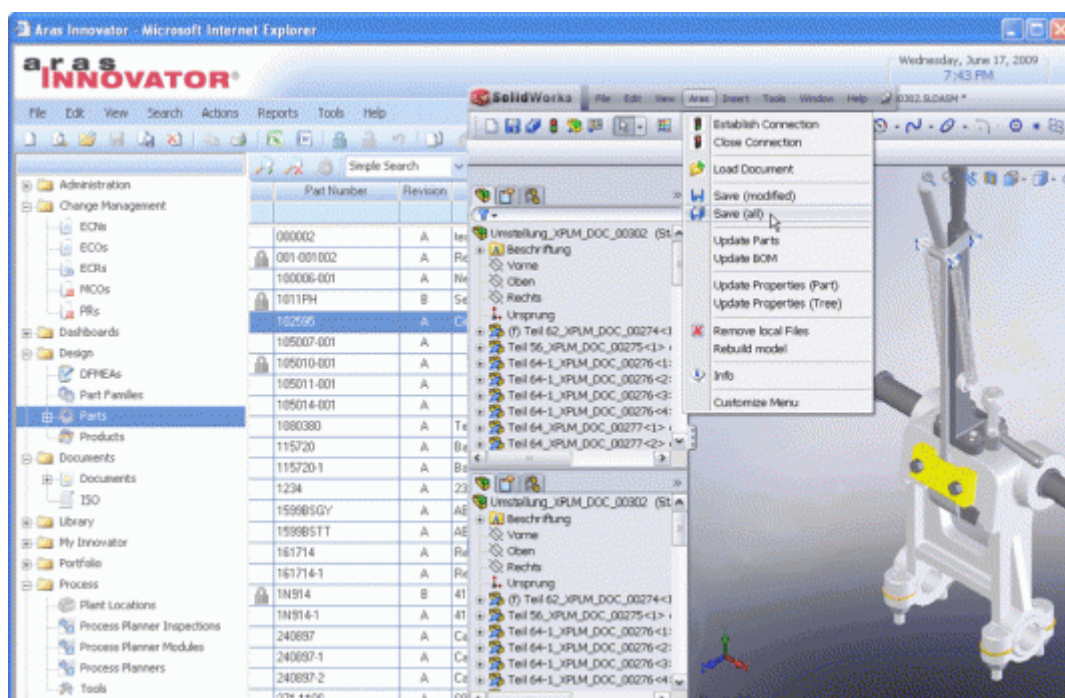


Figura 11: Screenshot Aras Innovator (7/9)

- Promuove la sicurezza, la collaborazione online, lo scambio di dati e il riutilizzo in tutta l'azienda, comprendendo il progetto e la gestione del programma, ingegneria e design, qualità, acquisti e molto altro.
- Permette di migliorare la progettazione di riutilizzo, sviluppare prodotti migliori più velocemente e migliorare la gestione delle modifiche.
- Le caratteristiche includono: check-in e check-out, gestione delle modifiche in ingegneria, controllo e rilascio delle versioni, gestione della distinta dei materiali, flussi di lavoro grafici automatizzati, visualizzazione di file e gestione della collaborazione di progettazione.

Collaborazione

Spazi di lavoro del progetto di collaborazione includono il piano di progetto, la programmazione e le assegnazioni con l'accesso sicuro contemporaneo ai membri del team con elementi di business e file allegati, così come diagrammi di Gantt e lo stato del progetto in tempo reale.

The screenshot displays the Aras Innovator web application interface. At the top, the browser window title is 'Project - 1381 XM Satellite Radio - Microsoft Internet Explorer'. The main content area shows project details for '1381 XM Satellite Radio' with a manager 'Innovator Admin'. The project status is 'Active'. Below this, there are sections for 'Project Plan', 'Team', 'Documents', 'Parts', and 'Meetings'. A table below the navigation tabs shows the project structure and task details.

N	Project Tree	Predecessors	Status	Leader [...]	Lead Role	Plan Start	Plan Finish	Duration	Hours	Attach
	Product Development Program		30			11/5/2007	3/23/2009	361		
	Phase 1 - Specification		100			11/5/2007	12/21/2007	35		
1	Requirements Development		100	Don Fowler		11/5/2007	11/20/2007	12	48	ABU-1003-03
2	Requirements Sign-off	1	100	Don Fowler		11/26/2007	12/3/2007	6	16	ABU-1003-03
3	Specification Development	2	100	Don Fowler		12/4/2007	12/21/2007	14	48	PS-1003-03 Ne
4	Specification Sign-off	3	100	Fred Thompso		12/21/2007	12/21/2007	0	16	PS-1003-03 Ne
	Phase 2 - Quality Planning		23			1/3/2008	2/15/2008	32		
5	Create Process Planning Documents	3	50	Tony Norton		1/3/2008	1/14/2008	8	16	PP-001055
6	Create DFMEA Document	5	35	Tony Norton		1/15/2008	1/24/2008	8	16	DFM-001057 E
7	Create Product Test Plans	6	10	Tony Norton		1/29/2008	2/7/2008	8	16	
8	Sign-off QP Documents	7	0	Fred Thompso		2/15/2008	2/15/2008	0	16	Multiple
	Phase 3 - Development		0			1/29/2008	12/23/2008	236		
9	Mechanical Product Design	4	0	Don Fowler		1/29/2008	3/5/2008	27	64	CAD-004004 X
10	Electrical Design	4	0	Don Fowler		1/29/2008	3/5/2008	27	64	CAD-004003 X
11	Product Validation	10,9	0	John Boyle		3/6/2008	3/17/2008	8	16	

Figura 12: Screenshot Aras Innovator (8/9)

Gestione dei progetti

Aras fornisce funzionalità complete per ridurre i costi del progetto, migliorare i margini di profitto e ridurre i cicli di sviluppo dei prodotti.

Project Num	Name	Ph 1	Ph 2	Ph 3	Ph 4	Ph 5	Ph 6	Location	Sched Due
1320	ad1845	100	100	100	90	40	10	China Plant 1	09/24/2005
1321	adsl2101	100	100	90	70	0	0	China Plant 1	04/14/2006
1322	as1846 System	100	100	100	50	0	0	Poland Plant 5	06/15/2006
1323	ad1848	100	80	30	0	0	0	Windsor Plant	06/21/2006
1324	ad1855	100	100	90	20	10	0	Dayton, OH	12/15/2005
1325	adsl2111	100	100	100	100	80	70	Malaysia Plant	10/13/2006
1326	as1805 System	100	100	100	100	50	30	Mexicali Plant	06/22/2006
1327	ad1622	100	100	80	90	40	0	Windsor Plant	05/16/2005
1328	ad1411	100	100	100	80	60	0	Windsor Plant	08/11/2006
1329	as1822 System	90	10	0	0	0	0	China Plant 5	05/08/2006
1330	ad1422	90	20	10	0	0	0	Malaysia Plant	02/17/2006
1331	ad1455	100	100	80	40	70	0	China Plant 1	10/21/2005
1332	adsl2140	100	100	100	80	90	10	China Plant 7	07/26/2007
1333	ad1650	100	100	80	70	20	0	Windsor Plant	05/19/2006
1334	adsl2135	90	100	100	0	0	0	China Plant 4	11/15/2007
1335	ad1844	70	90	80	30	0	0	Mexicali Plant	12/01/2005
1336	as1440 System	100	90	80	70	10	0	China Plant 4	07/15/2006
1337	as1401 System	80	40	0	0	0	0	Malaysia Plant	08/21/2008
1338	ad1601	40	90	30	0	0	0	Dayton, OH	03/15/2007
1339	ad1522	100	100	40	50	10	0	Poland Plant 5	11/5/2005
1340	adsl2166	100	100	100	100	90	50	China Plant 3	08/06/2005
1341	ad1212	5	20	0	0	0	0	Dayton, OH	10/08/2008
1342	ad1245	80	40	30	0	0	0	Malaysia Plant	11/04/2007
1343	as2202 System	100	100	100	80	10	30	Windsor Plant	10/30/2005
1344	ad1876	100	100	80	60	0	0	Poland Plant 3	04/21/2006
1345	ad2429	100	100	80	30	10	0	China Plant 2	03/29/2006

Figura 13: Screenshot Aras Innovator (9/9)

- Migliore visibilità in programmi e progetti, avvisi e-mail, e collaborazione di lavoro del progetto in spazi online.
- Gli obiettivi e i rischi del progetto sono gestiti online e forniscono aggiornamenti di stato con la possibilità di accedere ai dettagli e prendere decisioni.
- I principi del PMI (Project Management Institute) per la gestione dei progetti sono incorporati nei modelli per fornire standard di best practice.

4.4 ALL'INTERNO DI ARAS INNOVATOR

Aras Innovator è un framework per applicazioni internet basate su Microsoft .NET i cui componenti di base dal punto di vista dell'architettura sono quelli che possiamo vedere nella figura [49].

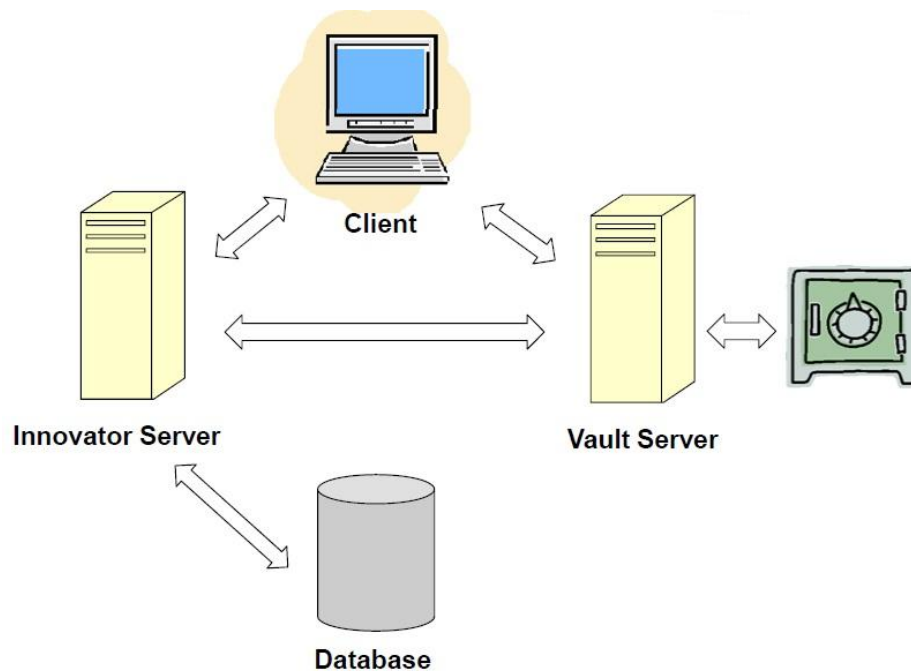


Figura 14: Componenti di base Aras Innovator

Client: Aras Innovator utilizza una interfaccia web-based browser che richiede poche risorse da parte dall'utente. Ci vuole soltanto Microsoft .NET 2.0 o superiore e Internet Explorer 6 o superiore.

Innovator Server: L'esecuzione come applicazione Microsoft .NET richiede la piattaforma Windows Server 2003 or 2008.

Database: Tutte le regole di configurazione e il codice, così come gli oggetti di soluzioni di business sono memorizzati nel database di Microsoft SQL Server 2005, 2008.

Vault Server: Un'applicazione server separata gestisce le informazioni sui file che sono collegati agli oggetti nel database di SQL Server.

Il quadro Aras è costituito da un insieme di modelli di base che consentono di creare soluzioni per risolvere i problemi di business. Utilizzando pagine web basate su form, è possibile configurare rapidamente una nuova soluzione. Il quadro è completamente configurabile ed estensibile consentendo di apportare le modifiche necessarie per supportare un particolare problema di business [49].

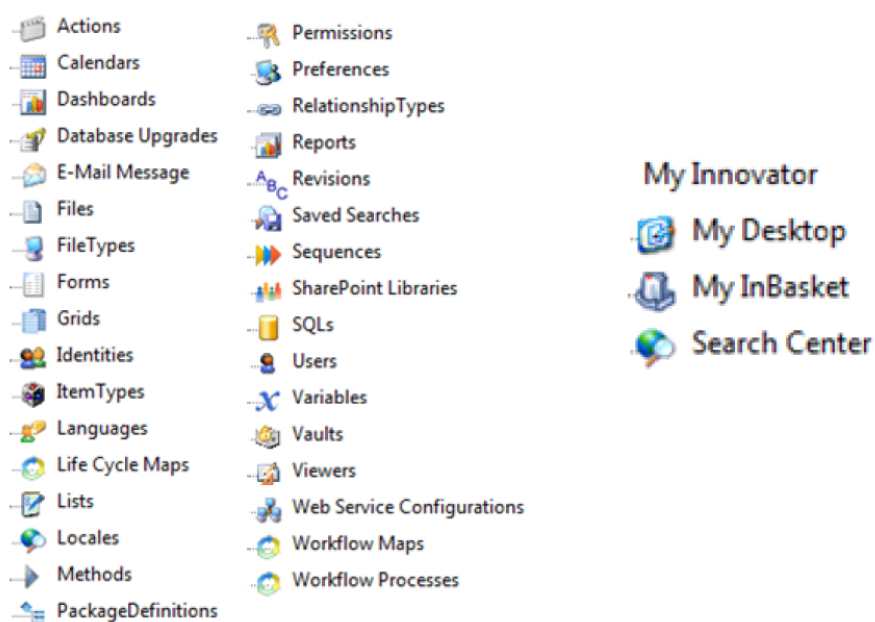


Figura 15: Modelli di base di Aras

My Innovator è un'interfaccia standard che è incorporata nel prodotto consentendo agli utenti accedere a [49]:

My Desktop: Posizione comoda per l'accesso ai prodotti di uso comune.

My InBasket: Destinazione per le assegnazioni creati da un flusso di lavoro.

Search Center: Posizione centralizzata per la ricerca di oggetti creati e memorizzati nel database.

Parliamo ora dei componenti di base delle soluzioni [49].

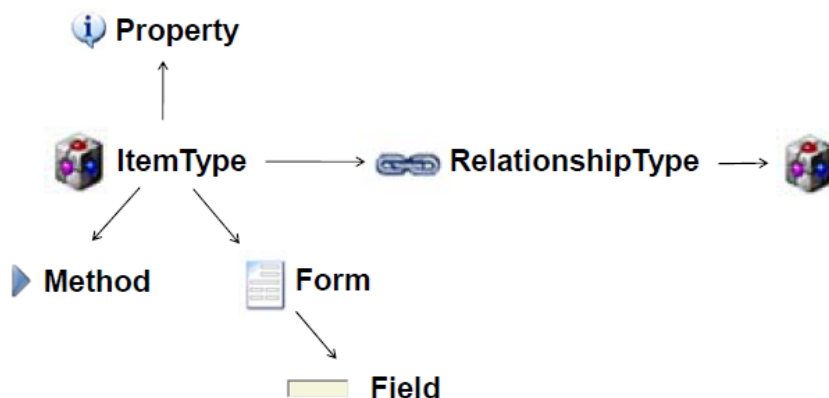


Figura 16: Componenti di base delle soluzioni di Aras Innovator

ItemType: È un modello che definisce la struttura di un oggetto di business nel database. È possibile creare ItemTypes in Aras Innovator che definiscono come i dati vengono memorizzati nel database e come gli oggetti creati da questo tipo si comporteranno. Per esempio, una Parte è un ItemType standard nel database che definisce diversi tipi di gruppi che possono essere modellati e creati.

Property: Una proprietà è un pezzo di informazioni tracciate su un elemento. Ogni proprietà ha un nome, un'etichetta, un tipo di dati e una serie di altre impostazioni per definire il suo comportamento.

Relationship Type: I tipi di relazioni definiscono quali ItemTypes sono collegati l'uno all'altro, ed i comportamenti del rapporto.

Form: Un form consente a un utente di interfacciarsi con un elemento ed è spesso usato per visualizzare e modificare le istanze di un ItemType particolare.

Field: Un campo è un'interfaccia utente utilizzato per visualizzare le informazioni in un form. I campi possono essere, ma non sono tenuti a essere associato a una proprietà.

Method: Un metodo è un innovatore ItemType che contiene un eseguibile .NET in linguaggio C #, VB, Javascript. Azioni ed eventi attivano l'esecuzione dei metodi. I metodi determinano il comportamento di un oggetto.

Vediamo ora il collegamento tra Oggetto – Tipo di oggetto e Relazione – Tipo di relazione [49].

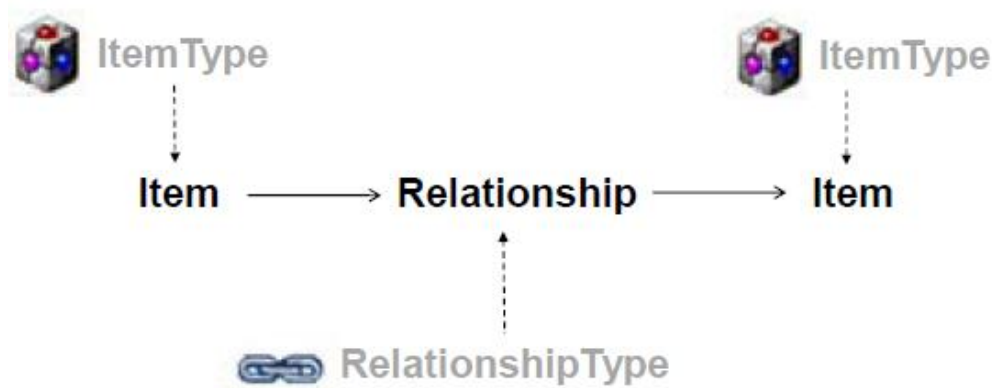


Figura 17: Oggetti e Relazioni

Item: Un'istanza di un ItemType è un oggetto. Ogni oggetto ha valori di proprietà diversi e può contenere un ciclo di vita ed essere controllato da un flusso di lavoro.

Relationship: ItemType si collegano per formare relazioni. Il modello per una relazione è il RelationshipType. Quando un utente finale collega due oggetti, un oggetto di collegamento viene creato anche nel database per la connessione.

Aras Innovator contiene un ricco modello di sicurezza altamente configurabile che consente di controllare chi ha accesso al sistema e ciò che essi sono autorizzati a fare.

Identità: Costituiscono la base per il modello di protezione di Aras Innovator. Le identità sono gerarchiche, e può rappresentare un utente, un gruppo, un gruppo di gruppi, e così via.

Utente: Per accedere a Aras Innovator, un utente deve essere definito nel sistema con le credenziali (nome utente e password). Utenti appartengono a una o più identità sopra definiti.

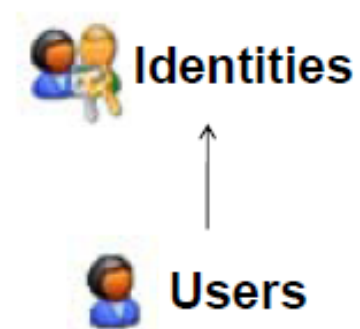


Figura 18: Utenti e identità

4.5 MODELLO DI BUSINESS

Come già detto la versione completa del prodotto di Aras Innovator può essere scaricata gratuitamente del sito. Solo la versione Aras Commercial Solutions può avere costi aggiuntivi di diritti di licenza (possiamo fare il paragone a un conto premium).

Il modello di business di Aras è stato progettato per supportare una soluzione aziendale open source che elimina diritti di licenza. Aras è in grado di mantenere questo modello attraverso il pagamento di servizi supplementari e sottoscrizioni [43].

Come fa Aras a guadagnare? Loro dicono che le aziende pagano per il suo supporto (formazione, servizi e consulenza) e non per il suo prodotto. Invece di vendere direttamente il prodotto Aras ha scelto di educare le persone e permettere loro di prendere la decisione da soli in quanto a quale fornitore credono che si adatti meglio alle sue iniziative aziendali. Inoltre eliminando una vendita costosa Aras contribuisce alla capacità di eliminare i costi di licenza del software PLM. Quello che vendono invece sono pacchetti di abbonamento annuali, consulenza e formazione. Aras è libero di essere usato se gli utenti non hanno bisogno di assistenza, ma le aziende che hanno bisogno di aiuto pagano per i servizi e il supporto che fornisce Aras [43].

I sostenitori delle aziende open source dicono che la perdita di fatturato legato alle licenze, in particolare per i software aziendali costosi, deve essere compensato con costi operativi inferiori. In generale, ciò significa che tagliando le vendite e il marketing. Aras ha eliminato le sue posizioni di vendita e li sta sostituendo con ingegneri di applicazioni che forniscono supporto ai clienti. Peter Schroer (CEO di Aras) ha detto: "Queste sono le persone con cui i clienti vogliono parlare comunque. Non vogliono parlare con un addetto alle vendite" [50].

I fornitori di PLM vendono il loro supporto e l'accesso alla loro proprietà intellettuale come un'unità. I venditori di PLM open source separano il supporto e la proprietà intellettuale. Aras regala la proprietà intellettuale e vende il supporto separatamente. Se non ti piace la loro versione di supporto, puoi allocare risorse interne e sviluppare il software libero che Aras ti ha dato. Il software open source cambia il punto di monetizzazione, rimuovendo il supporto della proprietà intellettuale dall'equazione

economica. Il denaro non viene speso per acquisire PLM, ma di svilupparlo e migliorarlo. Aras afferma che il 5% delle imprese che installano Aras Innovator alla fine spendono soldi con Aras relativo al supporto. E, in quel 5% dei prodotti installati, il suo fatturato è in crescita al 60% all'anno. [43].

Secondo Aras, eliminando i costi di licenza del software PLM, si riducono drasticamente i costi iniziali e totali, col conseguente minor costo totale nel settore. Sempre secondo loro Aras costa circa il 68% meno rispetto ai tradizionali fornitori di PLM. [43].

Il cliente vince e vince Aras. Aras ha preso la decisione di scrivere la loro applicazione su piattaforme come Microsoft .Net e SQL Server perché tutti i loro clienti avevano già le infrastrutture Microsoft. Poi Aras ha passato da un modello di licenze tradizionali a un modello di open source che può essere scaricato gratuitamente. Così Aras ha eliminato tutti i loro addetti alle vendite e invece hanno ingegneri di applicazioni la cui funzione è aiutare i loro clienti con l'installazione e la personalizzazione del prodotto [50].

4.6 PIATTAFORMA MICROSOFT CERTIFICATA

Aras utilizza le tecnologie Microsoft per la sua infrastruttura (il quale considerato un po' strano nel mondo open source ma ora vedremo il senso). Per eseguire Aras Innovator richiede solo Windows Server, SQL Server e .NET Framework. Aras ha preso la decisione di abbandonare il supporto multiplatforma e ha lavorato per integrare il proprio software con le più recenti tecnologie basate su Windows, tra cui il SharePoint e Office.

La ragione per cui utilizza le tecnologie Microsoft è che queste tecnologie sono tipicamente "qualificate" in azienda, la società ha già molte persone che conoscono le competenze di Microsoft, e la maggior parte delle aziende hanno un accordo esistente in luogo di acquisto per i prodotti Microsoft. Tutto questo rende più facile per il cliente usare Aras Innovator.

Aras porta una nuova proposizione di valore al mercato delle soluzioni software aziendali fornendo vantaggi diretti ai clienti per le aziende che utilizzano le tecnologie Microsoft. La posizione di Microsoft è favorevole a che Aras offra soluzioni open source poiché vedono una crescita enorme dei partner e come loro stanno trovando modi innovativi per sfruttare il valore della piattaforma Microsoft nell'ambito di una vasta gamma di modelli di licenza.

Aras è l'unica soluzione PLM aziendale che sia mai stata certificata per l'intera piattaforma Microsoft, tra cui Windows Server, SQL Server e .NET Framework. Infatti, Aras è l'unica soluzione PLM aziendale in ottenere la certificazione su qualsiasi prodotto della piattaforma di Microsoft.

Il test "Certified for" è il test più completo della Microsoft che effettua circa 100 casi di test per ogni prodotto Microsoft testato per confermare indipendentemente la conformità della soluzione. Le certificazioni ottenute da Aras forniscono il massimo livello di fiducia delle imprese disponibile per soluzioni aziendali [45].



Figura 19: Compatibilità Aras Innovator e Microsoft

4.7 CREAZIONE DI VALORE PER CLIENTI/PARTNER

La natura flessibile delle applicazioni aziendali di Aras Innovator consente un potenziale di creazione di valore per i clienti e partner di Aras. Permette un'innovazione di processo aziendale precedentemente irraggiungibile e accelera notevolmente il miglioramento continuo. L'attivazione di pratiche commerciali progressiste mette alle

società in grado di attaccare i concorrenti in nuovi modi per conquistare quote di mercato.

Un altro importante vantaggio che offre Aras Innovator è la possibilità di consolidare e razionalizzare il patrimonio ereditato e le applicazioni in un unico punto (soluzione moderna). La possibilità di creare rapidamente potenti soluzioni aziendali con una perfetta interoperabilità e facilmente integrabili con applicazioni strategiche offre un modo conveniente per ritirare i sistemi preesistenti.

Per le società di consulenza che offrono soluzioni software aziendali per i processi di business complessi, Aras Innovator fornisce un veicolo per offrire maggiore valore sia al cliente che per aumentare i ricavi. Una grande parte dell'incarico di consulenza può essere speso per attività di consulenza strategica poiché le risorse di programmazione che si richiedono per creare, integrare e modificare le applicazioni sono minori. Con Aras Innovator i consulenti sono in grado di spendere più tempo a lavorare con il cliente sui problemi di business critici invece di spenderlo in altri compiti.

Inoltre, i consulenti e integratori di sistemi sono in grado di creare e confezionare soluzioni industriali specifiche e uniche applicazioni ad alto valore per fornire nuove opportunità di ricavo dalla competenza [45]..

Esiste una rete di partner di Aras che è una comunità mondiale composta dai principali integratori di sistemi, sviluppatori di soluzioni, aziende di hosting, formatori e fornitori di tecnologia tra cui [43]:

- ✓ Softech
- ✓ Focus PLM
- ✓ Minerva
- ✓ T System
- ✓ Datatel Solutions
- ✓ Centecnia

Le soluzioni di Aras sono basate sulle prestazioni per aziende di tutte le dimensioni poiché le sue peculiarità sono quelle di fornire una soluzione che si adatti alle necessità

dell'utente (e non viceversa). Aras è presente in molti settori diversi, essendo adatto soprattutto alle aziende di grandi dimensioni. Di seguito vediamo alcuni dei clienti più importanti divisi per settori [43]:

- ✓ **Aerospaziale e Difesa:** Advatech Pacific, Esterline, Insitu, L-3 Communications, Lockheed Martin, Syracuse Research Corporation, The U.S. Army Watervliet Arsenal
- ✓ **Automobilistico:** Rolls-Royce, Freudenberg-NOK, Akebono Brake Corporation, Delphi, GETRAG FORD, Transmissions, Lear Corporation, Nemak, Nexteer Automotive, Ogihara, America Corporation, Spartan Chassis, Spartan Motors
- ✓ **Materiali da costruzione:** Uponor
- ✓ **Comunicazione:** MiTek Corporation, Ubidyne GmbH
- ✓ **Conglomerato:** Freudenberg, GE Global
- ✓ **Immagine digitale:** Carestream Health, HiTi Digital, Xerox
- ✓ **Elettronica:** Allen-Vanguard, Amtech Electrocircuits, EverFocus Electronics Corp., Fox Electronics, Frequency Electronics
- ✓ **Ingegneria:** Hitachi, Ingersoll Rand, ITT, Varian Semiconductor Equipment Associates
- ✓ **Prodotti Industriali:** GSI, HYDROMETER
- ✓ **Infrastrutture:** HAKAMA
- ✓ **Macchinario:** NSK Global, Structural Concepts Corporation
- ✓ **Fabbricazione:** ACCO Brands, Aristocrat Leisure Limited
- ✓ **Microelettronica:** MPI Corporation
- ✓ **Networking:** Tejas Networks, Tellabs, ZyXEL Communications
- ✓ **Farmaceutico:** Basic Research, Deutsche Blisterunion, Teva Pharmaceutical Industries
- ✓ **Generazione di energia:** Babcock & Wilcox
- ✓ **Telecomunicazioni:** Motorola
- ✓ **Giocattoli:** MEGA Brands



Figura 20: Clienti più notevoli di Aras Innovator

Come possiamo vedere c'è una grande quantità di aziende che già utilizzano Aras Innovator, si tratta quindi di un software già provato nel mercato.

Vengono riportate di seguito alcune frasi dette per alcuni dei principali manager delle aziende che hanno implementato Aras Innovator [51].

“Aras è diverso rispetto ad altri prodotti PLM. Aras ci offre la flessibilità necessaria per tenere il business e mettere tutti sullo stesso terreno di gioco senza cercare di racchiuderli dentro”. (Manager of Engineering Support, Xerox)

“Altri sistemi PLM non sembrano raggiungere tutti nel flusso di lavoro, perché prima o poi i costi di licenza diventano proibitivi. Sapevamo che non ci sarebbe questo problema con Aras”. (Director of Engineering Tools and Processes, Lear Corporation)

4.8 CASI DI ADOZIONE DI ARAS INNOVATOR

Motorola

Motorola è una società con 36,6 miliardi di dollari di fatturato annuo che è conosciuta in tutto il mondo per l'innovazione nel settore delle comunicazioni. L'azienda sviluppa tecnologie, prodotti e servizi che rendono possibili esperienze mobili. Quest'azienda è in continua crescita di business (15-20% annuale) e questo fa più difficile coordinare progetti e risorse, aumentando così l'inefficienza. Quindi avevano il bisogno di standardizzare i processi e volevano anche avere tutte le informazioni aggiornate. Hanno creato un "Mozzo Centrale" per business usando Aras Innovator che consente la gestione di valore aggiunto, la gestione delle risorse e la pianificazione e il monitoraggio dei costi delle prestazioni per un miglior coordinamento e collaborazione. Lo vediamo nelle seguenti figure [51]:



Standardize Our Processes Across the Organization

- + Seamless integration with other information systems
- + Ability to see all relevant information, deliverables, staff, costs, materials
- + Easily roll up to levels higher than single project
- + Reduce the time spent collecting data and generating metrics
- + Focus on the analysis of data and metrics for timely business decisions



OUR PLAN

Create a 'Central Hub' for Business Operations Using Aras Innovator

- + Create a framework for standardization, basis for quality improvements
- + Manage core business processes, projects, costs / budgets, products
- + Focus on our business processes, not the software implementation
- + Phased implementation over time – don't try to do too much all at once
- + Bring data to the user's desktop and integrated into their daily activities

43

WHAT WE'RE DOING

The screenshot shows the Aras Innovator Project - Medium 134 interface. The interface includes a menu bar, a toolbar, and a main content area. The main content area is divided into several sections: Project Information, Project Plan, and a detailed project table. The Project Information section shows the project name 'Medium 134', project number '134', project coordinator 'Don Hodge', and project lead 'Moto Project Imported'. The Project Plan section shows a tree view of the project structure, including 'Investigation Phase', 'Support', 'Documentation', 'Reviews', and 'Implementation Phase'. The detailed project table shows the following data:

N	Project Tree	Predecessors	Status	MBS	Code	Code	Charge	Charge	Suffix
	134.mpp								
	Elect Production (134)								
	Investigation Phase								
	Support								
	Documentation								
1	Write Project Management Plan	6	0	AA	1	1.0.01	134010		A
2	Prepare PIR Documentation		0	AA	1	1.0.01	134010		B
3	Prepare CIR Documentation	1	0	AA	1	1.0.01	134010		C
4	Finalize Project Plan	6	0	AA	1	1.0.01	134010		D
	Reviews								
5	Conduct Project Initiation Review (PIR)	2	0	AA	1	1.0.02	134010		E
6	Conduct Contract Initiation Review (CIR)		0	AA	1	1.0.02	134010		F
	Implementation Phase								
7	Manufacturing Test								
	Perform System	12	0	AA	3	3F.03	134011		A

Annotations in the image include:

- Development Project Prioritization**: Points to the 'Priority' dropdown menu.
- Import Microsoft Project**: Points to the 'Import Microsoft Project' button.
- Detailed Tracking, Budget, etc.**: Points to the 'Project Plan' section.
- Work Breakdown Structure**: Points to the 'Project Tree' section.
- Visual Status Aids**: Points to the 'Status' column in the project table.

The Motorola logo is visible in the bottom right corner of the screenshot.

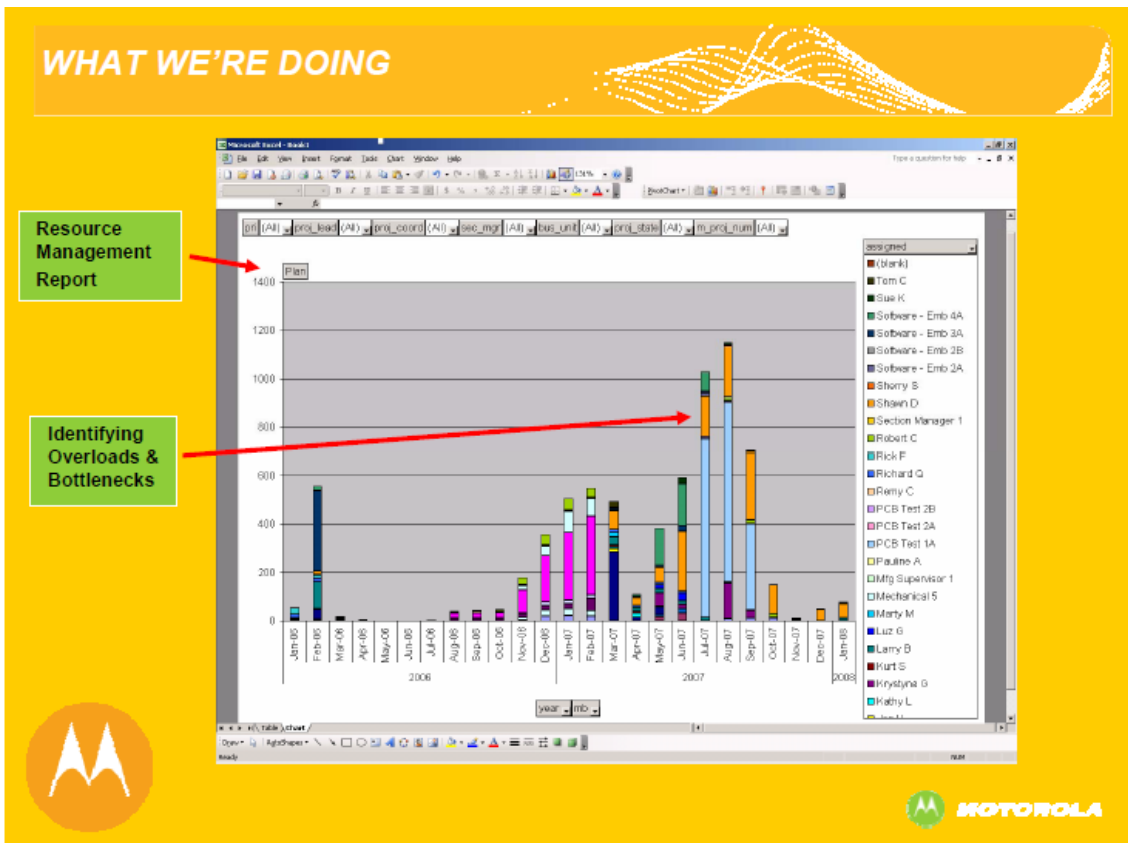


Figura 21: Slide dell'implementazione di Aras Innovator da parte della Motorola

Lear Corporation

Lear Corporation è un fornitore leader mondiale di sistemi di sedili per automobili e sistemi di gestione di potenza elettrica. I prodotti della sono progettati e fabbricati da un team integrato costituito da più di 75.000 addetti in 202 stabilimenti di 35 paesi.

Le ragioni della scelta di Aras sono stati la configurabilità, l'interfaccia utente e che non ha costi di licenza. Gli obiettivi che cercava la Lear con l'implementazione di Aras erano fare i dati del prodotto disponibili per tutte le funzioni per il proprio utilizzo, usare i processi di best-in-class per la gestione dei dati di prodotto, creare i necessari collegamenti per consentire l'automazione [51].

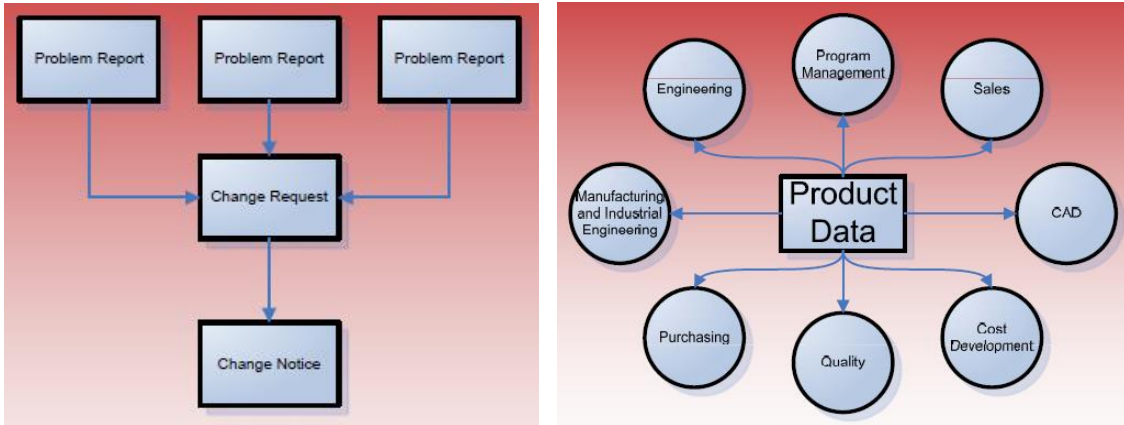


Figura 22: Obiettivi della Lear prima dell'implementazione

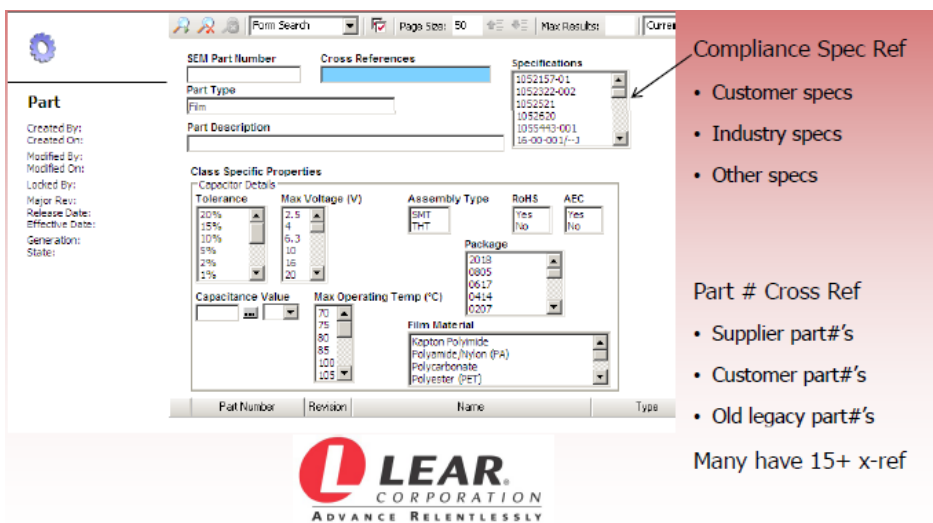
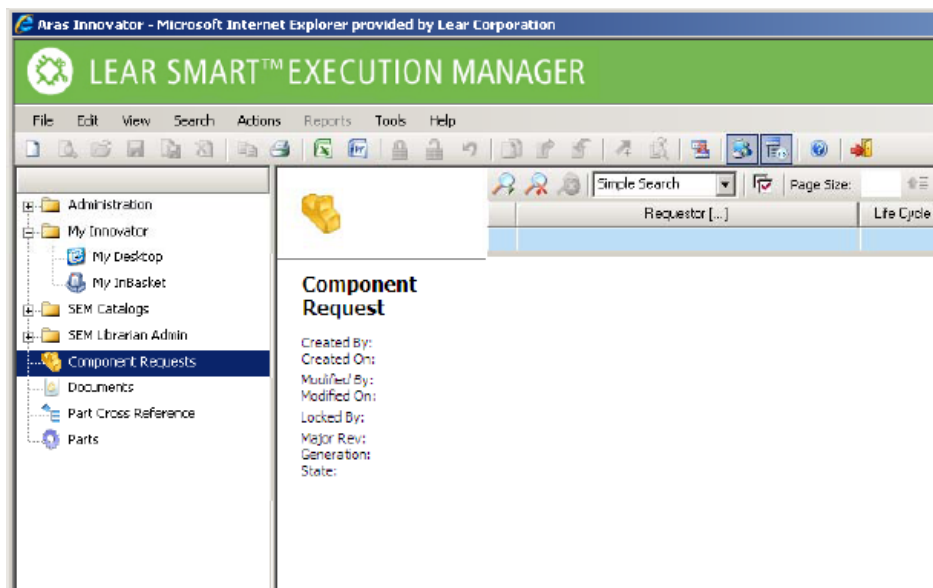


Figura 23: Implementazione di Aras Innovator da parte della Lear

U.S. Army

L'U.S. Army Watervliet Arsenal, è parte integrante dell'U.S. Army's Tank-automotive and Armaments Command (TACOM), un comando importante dell'Army Materiel Command. L'U.S. Army è produttore principale di tubi di pistola per l'esercito così come componenti di precisione per i cannoni di artiglieria, unità marine e motori a razzo.

L'U.S. Army stava cercando una modernizzazione nel sistema di business. Per affrontare questo problema hanno implementato la suite di soluzioni di Aras Innovator. Oggi Aras è diventato il sistema principale all'U.S. Army. I prodotti sono realizzati con processi di fabbricazione efficiente. Utilizzando Aras come il sistema per la pianificazione del processo produttivo, le attrezzature e la gestione utensili, e la manutenzione preventiva e calibrazione dell'U.S. Army è stato in grado di portare a un miglioramento continuo. Aras ha fornito una base standard per le operazioni offrendo allo stesso tempo la flessibilità necessaria per adattare i processi in modo rapido, efficientemente e riducendo i costi.

I benefici che hanno ottenuto sono:

- ✓ Sistema moderno di business per MBOM (Manufacturing Bill of Materials).
- ✓ Capacità di citare in modo più efficiente per garantire margini.
- ✓ Migliore visibilità nei processi di produzione.
- ✓ Tempi di consegna più brevi e tempi di ciclo ridotti
- ✓ Processi MBOM integrati al sistema ERP

L'approccio aperto di Microsoft che utilizza Aras consente le priorità di business dell'U.S. Army determinare come le funzionalità delle soluzioni addizionali sono lanciate al mercato poiché non c'è nessun costo per il software e nessun utente, modulo o licenza da acquistare. Andando avanti, l'US Army WVA capisce chiaramente i benefici che il flusso di lavoro flessibile online on-line fornisce per la gestione di complessi processi aziendali, combinato con la capacità di gestire file in un formato sicuro e compatibile. Più di recente l'U.S. Army ha esteso la soluzione Aras Innovator per automatizzare ulteriori processi in tutta l'azienda [51].

4.9 ARAS INNOVATOR E LE NUOVE TECNOLOGIE

È possibile usare Aras Innovator in dispositivi smartphone, sono uscite applicazioni per il sistema operativo Android, per iPhone/iPad e per Blackberry [52]. In questa versione l'utente può avere accesso allo stato di un progetto, alla valutazione del rapporto dello stesso, alle versioni precedenti e all'avanzamento della programmazione. Inoltre può aggiungere facilmente requisiti specifici del progetto, includendo clienti e requisiti di conformità, ecc [53].

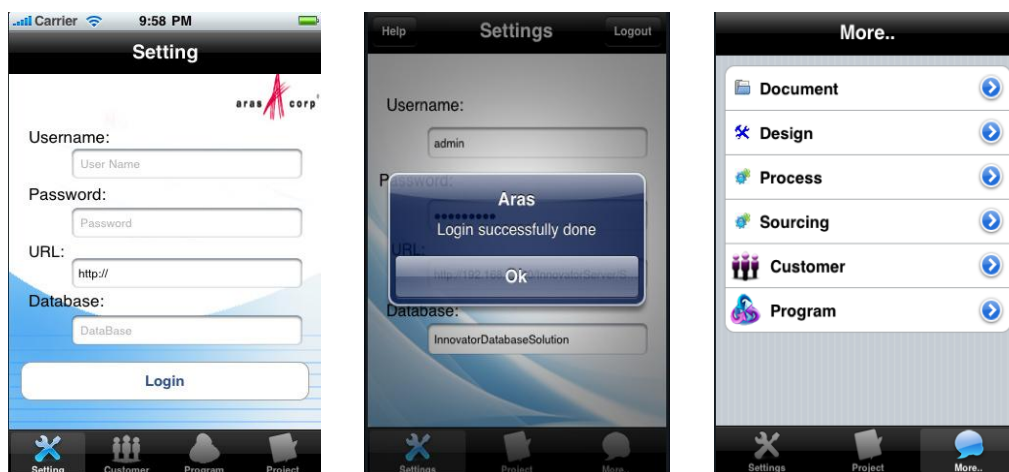


Figura 24: Aras Innovator sugli smartphone

5

CONCLUSIONI

Uno dei fattori chiave per le imprese è la compatibilità quando avviene uno scambio di dati con informazioni. Nella progettazione e la produzione, molti sistemi sono utilizzati per la gestione dei dati tecnici di un prodotto. Ogni sistema ha i suoi propri formati di dati in modo che la stessa informazione deve essere inserita più volte in più sistemi portando così alla ridondanza e agli errori. Il problema non è unico per la progettazione e la produzione, ma più grave per i dati 3D poiché i dati sono complessi e portano a maggiori errori e incomprensioni tra gli operatori [54]. Le informazioni scambiate possono essere le più svariate, pertanto i file in cui esse sono contenute potranno avere ognuno un formato adeguato a contenere l'informazione specifica. L'unico limite allo scambio di informazioni è rappresentato dal formato dei dati utilizzato per lo scambio delle stesse. Infatti, poiché su Internet sono collegate macchine sulle quali possono essere installati sistemi operativi diversi tra loro, è possibile che non tutti siano in grado di interpretare determinati formati di file, o che il formato con il quale viene trasportata l'informazione in questione è un formato proprietario e chi ne detiene i diritti non ha rilasciato al pubblico la struttura dello stesso. Il National Institute of Standards ha stimato che l'incompatibilità di dati è un problema di 90 miliardi di dollari per l'industria manifatturiera [55]. A questo problema rispondono i sistemi PLM in

generale e Aras Innovator in particolare con l'integrazione dei dati, rendendo possibile la lettura/scrittura/modifica di informazioni senza commettere errori.

L'adozione di software open source da parte delle aziende è cresciuta significativamente negli ultimi anni. Prodotti open source come Linux, JBoss e Apache offrono vantaggi di funzionalità, controllo e costo. Aras ha aggiunto questi vantaggi ai sistemi PLM. Possiamo dire che Aras Innovator è la nuova alternativa ai sistemi proprietari che sono disponibili finora sul mercato.

Nonostante questo software può essere scaricato gratuitamente, Aras è un'azienda for profit, in cui l'obiettivo è ottenere reddito. Non sono un gruppo di persone fanatici dell'open source che vogliono "cambiare il mondo", ma un gruppo di persone esperte in software PLM che hanno intenzione di portare un prodotto con completa funzionalità sul mercato e gestire un business profittevole allo stesso tempo. La differenza è che hanno scelto un modello di business diverso [50].

Vediamo l'open source come un gruppo puramente collaborativo di persone provenienti dalle diverse società lavorando nel loro tempo libero, nel caso di Aras Innovator c'è qualcosa di verità ma in generale lo sviluppo di base lo fanno gli sviluppatori di Aras. Quando le aziende scelgono Aras Innovator è logico pensare che pagheranno il canone di abbonamento per la manutenzione e l'assistenza, sebbene c'è l'opzione di non farlo e avere i loro propri sviluppatori. Quindi possiamo dire che Aras ha clienti e che questi clienti li pagano [50].

Aras è un'azienda che è riuscita a crescere in un periodo di economia in recessione. Se un'impresa sta prendendo in considerazione l'implementazione di un sistema PLM, sembra che in anticipo Aras Innovator potrebbe essere una soluzione adatta poiché abbiamo visto che è già un mercato provato (testato con referenze importanti come abbiamo detto prima: Xerox, Motorola, Rolls Royce, ecc).

Vediamo ora i vantaggi e gli svantaggi dell'implementazione di Aras Innovator:

Vantaggi:

- Non ci sono tasse di licenza: Il prodotto può essere scaricato gratuitamente e poi è l'azienda chi decide se vuole pagare per l'assistenza o meno. Inoltre se le aziende espandono i loro utilizzo delle soluzioni non hanno un costo aggiuntivo del software.
- Tecnologia avanzata: Architettura orientata ai servizi (Approccio “model-based”).
- Soluzioni open source: Flessibile, scalabile e collaborativo.
- Quadro comunitario: Garantito, affidabile e gestito.
- Piattaforma commerciale standard: Provata, affidabile e qualificata.

Svantaggi:

- Pur essendo gratis, il prodotto solo ha senso se fornisce un valore all'azienda. Se il valore è basso, anche a titolo gratuito non lo rende utile.
- Costi di apprendimento: È necessario imparare a utilizzare la nuova tecnologia da parte da tutti gli addetti dell'impresa.
- Costi di “fedeltà”: Possibile interruzione dei rapporti con altri fornitori.
- Investimenti: I sistemi PLM prendono tempo e risorse per l'attuazione, e c'è la possibilità che ci siano costi per l'hardware e per altre infrastrutture di supporto.

Per il futuro, la consolidazione di Aras Innovator sul mercato dipenderà del grado di accettazione di questo software da parte delle aziende. In questo mondo in continuo cambiamento, le tecnologie alla fine diventano obsolete e, di solito, vengono sostituite per nuove tecnologie che forniscono risultati migliori. Noi ci troviamo a questo punto con l'approccio “model-based” che può sostituire l'approccio “model-driven”. Le decisioni delle aziende diranno qual sarà il futuro del software aziendale e più in particolare dell'architettura del software PLM aziendale. Le aziende globali hanno

speso troppo tempo e denaro sui vecchi sistemi e riconoscono che quelli più moderni con l'architettura SOA sono necessari per rimanere competitivi.

Fuori dello spazio del PLM molte aziende stanno sperimentando con l'open source. Nel mondo del PLM per adesso Aras Innovator è l'unico prodotto che pensa all'open source, ma il futuro sembra molto promettente. Sebbene stiamo parlando di tecnologia, dobbiamo prendere in considerazione il modello di business, questo è un fattore chiave per la consolidazione di un prodotto sul mercato.

Un vantaggio dell'open source è che c'è la possibilità di maggiori sviluppi futuri del software poiché essendo una comunità aperta ci sono più persone che potrebbero portare un nuovo sviluppo e miglioramento del software.

Sempre guardando al futuro si può pensare a sviluppare di più il software per gli smartphone e i tablet per non rimanere indietro nel progresso tecnologico. Sebbene esistono già alcune applicazioni, queste non offrono ancora tutte le funzionalità delle versioni per il computer e sapendo che il mercato degli smartphone e dei tablet è in crescita negli ultimi anni sembra una buona opportunità per rendere il prodotto attrattivo.

APPENDICE

In quest'appendice si spiega il lavoro fatto nell'accordo di collaborazione con l'azienda SofTech. SofTech è un'azienda che offre soluzioni per la gestione del ciclo di vita di un prodotto, accelerando il rilascio di un prodotto e aumentando i profitti. Quest'azienda sviluppa, supporta e commercializza soluzioni software PLM e software CAD, tra cui Aras Innovator, che è stato il software utilizzato per me.

Il mio lavoro si concentrava sulla parte dell'impostazione della lingua italiana nel software, poiché non esiste ancora una versione in italiano, traducendo anche i termini dall'inglese all'italiano.

Aras Innovator ha la capacità di memorizzare proprietà come "Proprietà multilingue". Una proprietà multilingua memorizzerà valori multipli per la stessa proprietà. Il valore visualizzato sarà basato sul linguaggio definito in Aras Innovator per la tua impostazione locale. Se non vi è alcun valore per una lingua specifica, il valore per la lingua di default (inglese) sarà mostrato. Vediamo ora più dettagliatamente come procedere per impostare una nuova lingua in Aras Innovator:

1. Configurazione della lingua

Il primo passo è la configurazione del server, impostiamo il codice di lingua per il server web. Editiamo il file \Innovator\Server\web.config e poi modifichiamo l'etichetta globalizzazione mettendo culture = "it-IT".

```
<!-- Use culture attribute to tell Innovator Server which
locale to use when parsing float numbers. Example:
culture="en-US" for english (United States)
culture="en-GB" for english (United Kingdom)
culture="de-DE" for german (Germany)
culture="de-CH" for german (Switzerland) -->
<globalization requestEncoding="utf-8"
responseEncoding="utf-8" culture="en-US" />
```

La seconda cosa che dobbiamo fare è definire l’Oggetto Lingua in Aras Innovator. Accediamo come amministratore e selezioniamo Administration\Languages dal TOC (Table of Contents), poi dalla toolbar principale selezioniamo “Create a New Item”. Uscirà questa finestra:

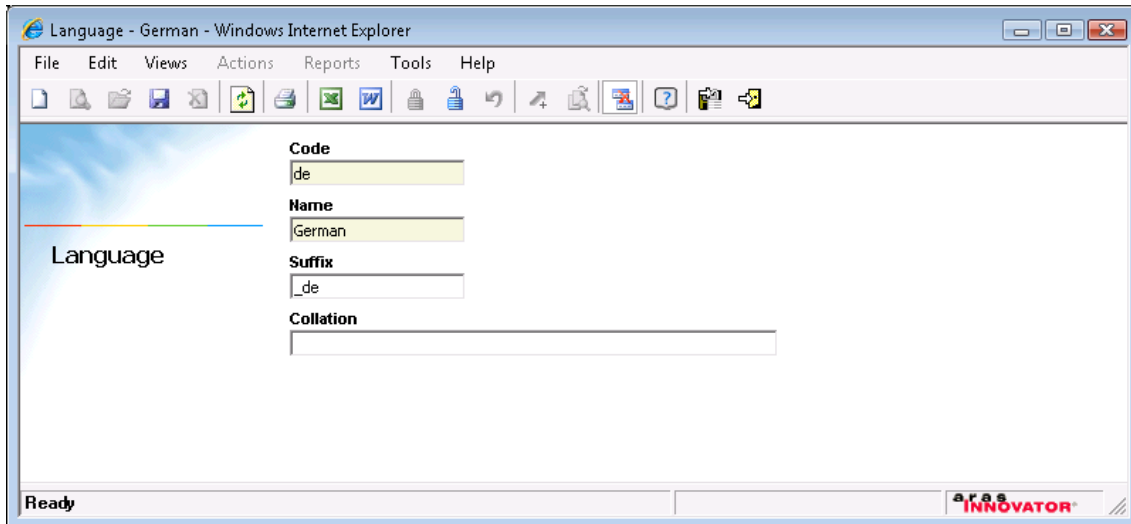


Figura 25: Impostazione della lingua in Aras Innovator

Qua mettiamo la lingua desiderata. Nel nostro caso Code = it, Name = Italian e Suffix = _it. Analogamente si fa per definire la localizzazione selezionando Administration\Locales dal TOC.

Il seguente passo per l'aggiunta di una lingua è quello di aggiornare le risorse XML che definiscono i menu, le toolbar e gli elementi dell'interfaccia utente. I file XML delle lingue specifiche dovranno essere messi nella cartella /Client accanto al predefinito (Inglese). Le cartelle delle diverse lingue dovrebbero essere in formato xml.{Codice Oggetto Lingua}, noi mettiamo xml.it.

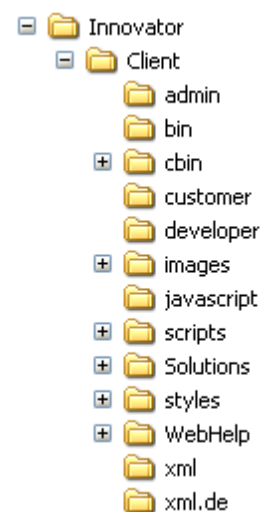


Figura 26: Cartella \Innovator\Client\xml.it

2. L'interfaccia utente

Ogni sessione di utente Aras Innovator definisce una localizzazione. Quando si inserisce un valore in un campo multilingua, il valore verrà salvato nel linguaggio definito dalla sessione. Se la lingua di sessione non esiste nel database, il valore verrà salvato nella lingua di default (inglese).

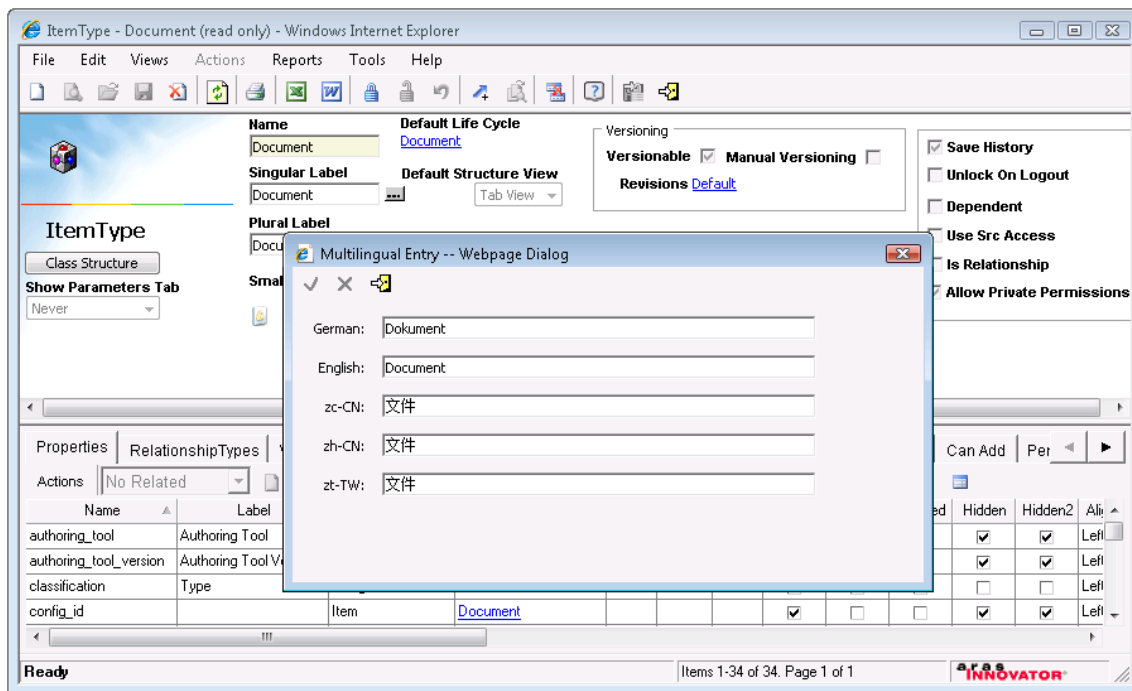


Figura 27: Proprietà multilingua Aras Innovator

Nella foto qui sopra, vediamo il dialogo multilingue per l'etichetta Singular Label. Si noti che questa etichetta ha un pulsante con i puntini accanto al campo. Selezionando questo aprirà i dialoghi multilingua che consentono all'utente di vedere il valore dell'etichetta per le lingue diverse.

3. Compilazione della lingua

Una volta che un linguaggio è stato definito, ci sono due modi principali per compilare la lingua. In primo luogo, è possibile inserire manualmente tutti i dati linguistici. In secondo luogo, è possibile importare una lingua utilizzando il Aras Innovator Language Pack Management Utility. La versione in italiano non c'è ancora, questo lavoro lo sta realizzando la SofTech, quindi per la prima volta si deve fare manualmente. Per farlo

manualmente dobbiamo modificare tutti i file che sono nella cartella \xml.it. Questo è un lavoro lungo e tedioso e ci vuole un po' di tempo a finirlo. Nella figura qua sotto possiamo vedere una parte di una delle tabelle di traduzione su cui stavo lavorando.

3	English Word	Translation
4		
5	Created By	Creato da
6	Created On	Creato il
7	Modified by	Modificato da
8	Modified On	Modificato il
9	Locked By	Bloccato da
10	Major Rev	Rev Maggiore
11	Release Date	Data Rilascio
12	Effective Date	Data Effectivity
13	Generation	Generazione
14	State	Stato
15		
16	Entry	Inserimento
17	Return Selected	Ritorna Selezionato
18	Clear Selected	Pulisci Selezionato
19	Report	Report
20	Process	Processo
21	Workflow	Workflow
22		
23		field_toolbar.xml
24	New Textarea	Nuova Area di Testo
25	New Dropdown	Nuova Lista di scelte
26	New List Box	Nuova Lista di riepilogo
27	New List Box Multi Select	Nuova Lista di riepilogo a Selezione Multipla
28	New Checkbox	Nuova Checkbox
29	New Radio Button List	Nuovo Elenco Radio Button

Figura 28: Tabella di traduzione

4. Esempio

Consideriamo in primo luogo una richiesta con action = "get" da parte di un utente.

```
<Item type="ItemType" id="B88C14B99EF44982..." action="get"
select="label"/>
```

Questa richiesta restituisce l'etichetta nella lingua della sessione dell'utente stabilita al momento della connessione. Se non c'è un'impostazione locale che corrisponde al cliente al momento in cui ha aperto la sessione, il valore predefinito (inglese) verrà restituito. Ovviamente anche se il cliente ha messo la lingua inglese nelle sue impostazioni.

```
<Item type="ItemType" id=" B88C14B99EF44982...">
<label xml:lang="en">Document</label>
</Item>
```

Se ora invece abbiamo un cliente italiano, il valore restituito sarà un altro.

```
<Item type="ItemType" id="B88C14B99EF44982...">
<label xml:lang="it">Documento</label>
</Item>
```

Le richieste possono specificare la lingua desiderata nella risposta, includendo l'attributo "lingua" nel tag elemento. Possono essere richieste più di una lingua.

```
<Item type="ItemType" id="B88C14B99EF44982..." action="get"
select="label" language="en,it"/>
```

Con questa richiesta si ottiene:

```
<Item type="ItemType" id=" B88C14B99EF44982..." >
<label xml:lang="it">Documento</label>
<label xml:lang="en">Document</label>
</Item>
```

BIBLIOGRAFIA

- [1] S. Terzi, M. Garetti, C. Bongiardina, M. Macchi, “*Product Lifecycle Management: Definizione, caratteristiche e trend evolutivi*”, 2010-2011, Politecnico di Milano.
- [2] S. Terzi, “*Corso in Product Lifecycle Management*”, 2009-2010, Politecnico di Milano.
- [3] T. Carlos Grau Abarca, “*Implantación de un sistema PLM en una empresa de fabricación en base a productos*”, 2008, Universidad Pontificia Comillas, Madrid.
- [4] www.cimdata.com
- [5] European Parliament, “*Amended Software Patent Directive*”, 2003.
- [6] http://www.linfo.org/free_file_format.html
- [7] <http://www.digitalpreservation.gov/formats/sustain/sustain.shtml#disclosure>
- [8] M. Térmens, M. Ribera, “*El control de los formatos en la preservación digital*”, 2009, Universitat de Barcelona.
- [9] http://en.wikipedia.org/wiki/List_of_file_formats
- [10] [http://en.wikipedia.org/wiki/List_of_file_formats_\(alphabetical\)](http://en.wikipedia.org/wiki/List_of_file_formats_(alphabetical))
- [11] Bruce E. Bargmeyer, Daniel W. Gillman, “*Metadata standards and metadata registries: an overview*”, Washington.
- [12] <http://www.fileinfo.com>
- [13] http://en.wikipedia.org/wiki/Unstructured_data
- [14] http://en.wikipedia.org/wiki/Technical_standard
- [15] http://en.wikipedia.org/wiki/De_facto_standard

- [16] European Union, “*Directive 98/34/EC of the European Parliament and of the Council of 22 June 1998 laying down a procedure for the provision of information in the field of technical standards and regulations*”, 2009.
- [17] http://es.wikipedia.org/wiki/Est%C3%A1ndar_internacional
- [18] http://en.wikipedia.org/wiki/ISO_10303
- [19] “*STEP application handbook, ISO 10303, version 3*”, June 2006.
- [20] L. Lauro, “*Storia della norma internazionale sullo scambio dei dati di prodotto STEP*”, Politecnico di Torino.
- [21] <http://www.steptools.com/library/stepnc>
- [22] <http://www.plcs-community.com/en/reminderooverview-what-is-plcs>
- [23] A. F. Cutting-Decelle, J. J. Michel, “*ISO 15531 MANDATE: a standardised data model for manufacturing management*”, 2003, Geneva.
- [24] http://www.plm.automation.siemens.com/en_us/products/open/plmxml
- [25] <http://www.isa-95.com>
- [26] http://en.wikipedia.org/wiki/ISO_18629
- [27] “*ISO 14224: Petroleum, petrochemical and natural gas industries -- Collection and exchange of reliability and maintenance data for equipment*”, 2006.
- [28] <http://www.dyadem.com/standards/iso-14224>
- [29] “*ISO/TS 15926: Industrial automation systems and integration—Integration of life-cycle data for process plants including oil and gas production facilities—Part 8: Web Ontology Language (OWL) implementation*”, 2011.
- [30] <http://15926.org>
- [31] “*ISO/IEC TR 24774: Process Description Guide*”, 2007
- [32] <http://www.mimosa.org>
- [33] http://es.wikipedia.org/wiki/C%C3%B3digo_abierto
- [34] http://standishgroup.com/newsroom/open_source.php
- [35] R. E. Wyllys, “*Overview of the Open-Source Movement*”, 2000, University of Texas.
- [36] Verts, William T. “*Open source software*”, 2008.

- [37] <http://www.opensource.org>
- [38] <http://www.softwarelibero.it>
- [39] http://it.wikipedia.org/wiki/Differenza_tra_software_libero_e_open_source
- [40] Christopher R. Gabriel, “*Perché ‘Software Libero’ è meglio di ‘Open Source’*”, 2007.
- [41] Phoronix, “*Towards A Real Business Model For Open-Source Software*”, 2010.
- [42] http://en.wikipedia.org/wiki/Open_source_movement
- [43] <http://www.aras.com>
- [44] <http://www.aras.com/solutions>
- [45] Aras Corp, “*The Aras Innovator Model-based SOA for Enterprise PLM. Combining the Model-based Approach with the Service-Oriented Architecture*”, 2011.
- [46] <http://www.aras.com/technology/model-based-soa.aspx>
- [47] <http://www.technologyevaluation.com/es/research/articles/nota-de-producto-aras-innovator-21809>
- [48] <http://www.steepgraph.com/plm-services/aras-innovator>
- [49] Aras Corp, “*Configuring Solutions, Student Guide, Version 9.3*”, 2011.
- [50] D. Dodge, “*Microsoft and Open Source? Aras leads the way*”, 2007.
- [51] <http://www.aras.com/customers>
- [52] <http://www.porchys.com/arasiphone.html>
- [53] http://es.androidzoom.com/android_applications/productivity/aras-innovator-android_bdxrb.html
- [54] S. B. Brunnermeier and S. A. Martin, “*Interoperability Cost Analysis of the U.S. Automotive Supply Chain*”, 1999.
- [55] D. Marsini, “*Il formato dei dati*”, 2011.

INDICE DELLE FIGURE

Figura 2: Redditività di un prodotto lungo tutto l’arco del suo ciclo di vita prima e dopo l’implementazione di un sistema PLM	12
Figura 2: Confronto Software Open Source – Software Chiusi	36
Figura 3: Framework Aras Innovator	49
Figura 4: Differenza fondamentale di Aras con altri sistemi convenzionali	51
Figura 5: Screenshot Aras Innovator (1/9)	56
Figura 6: Screenshot Aras Innovator (2/9)	57
Figura 7: Screenshot Aras Innovator (3/9)	58
Figura 8: Screenshot Aras Innovator (4/9)	59
Figura 9: Screenshot Aras Innovator (5/9)	60
Figura 10: Screenshot Aras Innovator (6/9)	61
Figura 11: Screenshot Aras Innovator (7/9)	62
Figura 12: Screenshot Aras Innovator (8/9)	63
Figura 13: Screenshot Aras Innovator (9/9)	64
Figura 14: Componenti di base Aras Innovator	65
Figura 15: Modelli di base di Aras	66
Figura 16: Componenti di base delle soluzioni di Aras Innovator	67
Figura 17: Oggetti e Relazioni	68
Figura 18: Utenti e identità	68
Figura 19: Compatibilità Aras Innovator e Microsoft	71
Figura 20: Clienti più notevoli di Aras Innovator	74
Figura 21: Slide dell’implementazione di Aras Innovator da parte della Motorola	75, 76, 77

Figura 22: Obiettivi della Lear prima dell'implementazione	78
Figura 23: Implementazione di Aras Innovator da parte della Lear	79
Figura 24: Aras Innovator sugli smartphone	80
Figura 25: Impostazione della lingua in Aras Innovator	86
Figura 26: Cartella \Innovator\Client\xml.it	86
Figura 27: Proprietà multilingua Aras Innovator	87
Figura 28: Tabella di traduzione	88

INDICE DELLE TABELLE

Tabella 1: Integrazione di Aras con altri sistemi	55
--	----

ACRONIMI

ANSI: American National Standards Institute

AP: Protocollo Applicativo

API: Application Programming Interfaces

APQP: Advanced Product Quality Planning

BOL: Beginning Of Life

BOM: Bill of Materials

CAD: Computer Aided Design

CAM: Computer Aided Manufacturing

CCOM: Common Conceptual Object Model

CRIS: Common Relational Information Schema

CRM: Customer Relationship Management

DEX: Data EXchange

EDA: Electronic Design Automation

ERP: Enterprise Resource Planning

FSF: Free Software Foundation

GIF: Graphics Interchange Format

GIS: Geographic Information Systems

GNU: GNU's Not Unix

GPL: GNU Public License

HTML: HyperText Markup Language

HTTP: HyperText Transfer Protocol

HTTPS: HyperText Transfer Protocol Secure

IEC: International Electrotechnical Commission

IP: Internet Protocol

ISBN: International Standard Book Number

ISO: International Organization for Standardization

IT: Information Technology

ITU: International Telecommunication Union

JPEG: Joint Photographic Experts Group

MANDATE: MANufacturing DATa Exchange

MBOM: Manufacturing Bill of Materials

MES: Manufacturing Execution Systems

NC: Numerical Control

NPI: New Product Introduction

O&M: Operations & Maintenance

OASIS: Organization for the Advancement of Structured Information Standards

OLE: Object Linking and Embedding

OSD: Open Source Definition

OSI: Open Source Initiative

OSS: Software Open Source

PDF: Portable Document Format

PDM: Product Data Management

PHP: Hypertext Preprocessor

PLCS: Product Life Cycle Support

PLM: Product Lifecycle Management

PMI: Project Management Institute

PNG: Portable Network Graphics

PSL: Process Specification Language

RDF: Resource Description Framework

RM: Reliability and Maintenance

SCM: Supply Chain Management

SI: Sistemi informativi

SNP: Supply Network Planning

SOA: Service Oriented Architecture

SOAP: Simple Object Access Protocol

SQL: Structured Query Language

STEP: STandard for the Exchange of Product

TCP: Transmission Control Protocol

TIFF: Tagged Image File Format

TOC: Table of Contents

UML: Unified Modeling Language

UNI: Ente Nazionale Italiano di Unificazione

XML: eXtensible Markup Language

Ringraziamenti

Vorrei ringraziare il professor Terzi per avermi dato la possibilità di svolgere questa tesina e per avermi seguito durante l'esecuzione del lavoro.

Ringrazio anche Dante Cislighi e il resto di colleghi della SofTech per avermi dato l'opportunità di collaborare con loro nell'azienda rendendo così più facile lo svolgimento del lavoro.

Ringrazio tutti i miei amici che sono stati lontani durante la mia permanenza a Milano e anche tutta la gente conosciuta in questo periodo, alcuni di essi amici per sempre, per aver fatto che la mia fase all'università sia stata più facile ogni giorno, è sembrata che è passata troppo veloce.

Per ultimo ringrazio tutta la mia famiglia che mi ha dato la possibilità di avere tutto quello che ho e possa riuscire ad avere in un futuro.

Grazie a tutti.