

> PROCESSI

> intro

I processi sono insiemi di attività che hanno lo scopo di creare o sviluppare un certo prodotto software. Gli insiemi di attività che costituiscono un processo sono strutturati in cicli che seguono un certo modello. Legati ai processi abbiamo anche metodi e tecniche, il metodo è il carattere che acquisisce un processo in base all'attività che sta dettagliando ed è ciò che definisce come deve essere applicata ciascuna tecnica implicata, mentre le tecniche sono procedure formalizzate per l'esecuzione di specifiche attività. Tecniche e metodi costituiscono le pratiche, come progettazione, ispezione, testing o anche linee di prodotto. Le pratiche quindi rappresentano il carattere di un processo dipendente da determinate tecniche applicate alle attività. Parliamo anche di buone pratiche quando hanno una certa valenza empirica e derivano da lezioni generali su applicazioni di metodi e tecniche fatte in progetti passati.

Degli strumenti per gli sviluppatori sono i tools, ossia semplicemente applicazioni esterne per l'esecuzione di task ripetitivi che svolgono una funzione di supporto agli sviluppatori nell'esecuzione di una pratica. I tools devono essere adeguati alle caratteristiche della pratica, allo scopo ed alla dimensione dell'applicazione, agli obiettivi di qualità da raggiungere, alla competenza degli sviluppatori ed a risorse e tempo disponibili per il progetto.

> cicli

I cicli seguono un certo modello di sviluppo, consideriamo tre famiglie di modelli: i modelli a cascata i modelli evolutivi ed i modelli estremi.

- Nei modelli a cascata, il ciclo segue un'evoluzione lineare di attività. Il principio base è ogni attività inizia quando si è completata la precedente, il classico paradigma waterfall, inoltre le attività a valle utilizzano manufatti intermedi prodotti da attività a monte. Per poter applicare questo modello sono necessari alcuni prerequisiti, è importante che il dominio dell'applicazione sia stabile e ben compreso da chi dovrà lavorare sullo sviluppo, la stabilità del dominio restituirà una certa correttezza e stabilità nei requisiti del processo che potranno essere comunque modificati in casi eccezionali. In più durante il ciclo di produzione il business del committente non dovrà variare, e questa situazione non dovrà gravare con conseguenze insostenibili.
- Nei modelli evolutivi il principio base è sviluppare prima le parti più critiche dell'applicazione in modo da poter avere alcuni feedback da parte del committente e proseguire evolvendo l'applicazione con i feedback positivi. Abbiamo due tipologie di modelli evolutivi, incrementale ed a spirale.
 - Nella tipologia di modello evolutivo incrementale, il prodotto viene suddiviso in sotto-prodotti, i quali vengono sviluppati da altrettanti sotto-processi in modalità a cascata. Ogni sotto-processo sarà provvisto di fasi come requisiti, progettazione ed implementazione. L'idea base di questo approccio è la suddivisione delle features in classi di importanza, si darà priorità alle features a più alta criticità. Questo modello si adatta ai casi in cui c'è incertezza tra i requisiti, quindi al

termine di ogni incremento il committente dovrà verificare i servizi realizzati. In caso di insuccesso comunque l'investimento sprecato sarà minimo, altrimenti si potrà procedere con il successivo incremento.

- Nella tipologia di modello evolutivo a spirale la caratteristica principale è la ripetizione ciclica di analisi dei rischi, sviluppo e validazione. Molto simile al modello evolutivo incrementale, si pone enfasi su risorse e budget considerandole di massima criticità. Ogni incremento guiderà l'avanzamento considerando l'analisi dei rischi, in ogni fase si farà uso delle esperienze acquisite da fasi omologhe svolte nello stesso processo.
- I modelli estremi sono fortemente centrati sulla rapidità di produzione, si mira a restituire subito informazioni, applicazioni o modelli di decisione. Alla base dello sviluppo con modelli estremi c'è lo sviluppo agile, cioè capacità di reagire rapidamente ed appropriatamente ai cambiamenti riorientando il lavoro e le risorse ed applicando prontamente iniziative di sviluppo. L'idea base è il raggiungimento del giusto trade-off tra costo e valore, è particolarmente indicato per lo sviluppo con tempi e risorse di produzione limitati.
 - Un esempio di processo agile è l'extreme programming xp, in cui abbiamo la definizione di scenari di uso per la definizione dei requisiti per il processo, abbiamo un piano di sviluppo suddiviso in piccole interazioni e graduali rilasci di servizi, abbiamo lo sviluppo con il metodo del pair programming pp, abbiamo successivamente lo unit testing sul codice ed infine un'attività di miglioramento del prodotto.
 - Il pp è un tecnica di sviluppo del codice in cui progetto, codice e test vengono prodotti da team di due sviluppatori che ricoprono due figure diverse e che vengono frequentemente invertite: driver ed observer. Il driver ha fisicamente il controllo di tastiera e mouse e scrive il codice, l'observer offre supporto rilevando errori o proponendo semplicemente alternative per ottimizzare e accelerare la produzione. Con il pp si mira ad unificare quindi le figure di analista, progettista e sviluppatore rendendo la comunicazione meno rumorosa.

> nota trade-off

Nel campo dell'economia, il trade-off è una relazione tra costo e valore in cui la perdita di valore di una delle due entità caratterizza l'aumento di valore dell'altra.

Nel modello estremo, un trade-off inadeguato vuol dire una relazione tra costo e valore errata, una soluzione è ottimizzare il costo evitando che superi il valore dell'applicazione.

> nota unit testing

Nella programmazione, lo unit testing è una procedura usata per la verifica di singole porzioni di codice, appunto unità. Per unità si intende una minima parte di codice sorgente. Nella programmazione procedurale l'unità può essere il singolo programma, la funzione o la procedura, mentre nella programmazione ad oggetti può essere il metodo.

> sviluppo con qualità

Nello sviluppo con qualità, è importante osservare alcune pratiche nell'esecuzione del processo.

1. La prima pratica è considerare di migliorare la qualità del prodotto migliorando la qualità del processo, sicuramente senza un processo maturo è difficile ottenere un prodotto di buona qualità e la probabilità di scartare parti del prodotto è alta. Tuttavia un processo maturo non garantisce un buon prodotto, conta anche un terzo fattore l'esperienza degli sviluppatori. In alcuni casi non è strettamente collegata una migliore qualità del processo per una migliore qualità del prodotto, ma è da valutare anche assumere sviluppatori con più esperienza.
2. La seconda pratica è apportare modifiche in modo sicuro, un processo può essere direttamente il sistema dal quale sono generati tutti gli altri sistemi dell'impresa. La modifica del processo è critica, bisogna conoscere il sistema per modificarlo senza rischi. Un buon metodo è raccogliere dati storici, e poi applicare modelli di previsione che mettano in relazione le azioni previste nel processo e le dirette conseguenze sul prodotto.
3. La terza pratica è garantire un certo ritorno economico oltre alla soddisfazione del committente, quindi un giusto trade-off tra qualità tecnica, costo e tempo di esecuzione. Un trade-off inefficace non necessariamente può voler dire processo non sufficientemente maturo, ma può anche essere necessità di migliori sviluppatori. E' necessario quindi disporre di modelli di qualità che includano qualità tecnica e parametri di controllo.
4. La quarta pratica è ottenere un certo ritorno economico anche sulle iniziative di miglioramento, quindi un prodotto che resti di qualità e che conservi il giusto bilancio tra investimenti e ritorno economico. E' importante avere dati e modelli per prevedere i ritorni economici sulle varie iniziative di miglioramento proposte.
5. La quinta pratica è ottenere conformità nell'esecuzione, l'iniziativa è istituzionalizzare il prodotto attraverso la diffusione dello stesso per uniformare il suo rigore nell'esecuzione, ma per il miglioramento della qualità è necessaria la crescita professionale degli sviluppatori, il continuo adeguamento del contesto operativo e buone competenze per l'uso appropriato delle tecnologie a sua disposizione.
6. La sesta pratica è comunicare la logica mediante la documentazione, il problema è comunicare la motivazione che giustifica ogni passo di processo per evitare che l'ingegnere non conoscendo le motivazioni dietro un certo passaggio, possa decidere di eliminarlo generando problemi che possano gravare sulla qualità del prodotto. È importante quindi documentare ogni decisione dietro gli inserimenti, modifiche o eliminazioni di un passo nella descrizione di un processo.

> processi reali

Per il raggiungimento dell'obiettivo strategico per un'impresa abbiamo il prodotto ottenuto da una serie di processi, normalmente verrebbero considerati solo i processi direttamente implicati nello sviluppo e quindi quelli primari ma per una panoramica completa sul processo di sviluppo è importante considerare tutti i processi che effettivamente lavorano e anche in background. Definiamo tutti questi processi come processi reali, l'aumento del numero di processi da considerare in fase di progettazione comporta un aumento del numero di problemi dovuti all'interazione ed integrazione dei processi, in fase di progettazione dovremo pianificare a priori

la gestione di tutte le componenti usate in uno sviluppo reale.

Tutto ciò si riassume nello standard ISO 12207, in cui vengono definiti tutti i processi coinvolti da una organizzazione per lo sviluppo di prodotti software come processi reali, e vengono classificati come processi primari, processi di supporto e processi organizzativi.

- I processi primari caratterizzano la produzione, determinano creazione realizzazione e manutenzione di un prodotto software e si suddividono in cinque sottocategorie di processi: l'acquisizione, la fornitura, lo sviluppo, la gestione operativa e la manutenzione.
 1. Il processo di acquisizione definisce le attività che l'acquirente deve compiere per entrare in possesso di un prodotto software.
 2. Il processo di fornitura definisce le attività che il fornitore deve compiere per rendere un prodotto software disponibile all'acquisto.
 3. Il processo di sviluppo definisce le attività che un'organizzazione deve compiere per definire e sviluppare un prodotto software.
 4. Il processo di gestione operativa definisce le attività che deve compiere il programmatore all'interno dell'organizzazione stessa.
 5. Il processo di manutenzione definisce le attività che l'organizzazione deve compiere per la manutenzione del prodotto software.

- I processi di supporto contribuiscono al successo di un ciclo di produzione e si occupano della qualità di un progetto, principalmente offrono supporto ed aiutano l'esecuzione dei processi primari. In questa classe di processi abbiamo otto sottocategorie: documentazione, gestione della configurazione, assicurazione di qualità, verifica, validazione, revisione, audit e gestione dei problemi.
 1. Il processo di documentazione è dedicato alla scrittura della documentazione e rappresenta il valore aggiunto per il miglioramento della qualità rispetto alla classica progettazione.
 2. Il processo di gestione della configurazione costituisce un'infrastruttura logica per tutti i manufatti previsti nel processo, in cui vengono catalogati e classificati. Questo processo fornisce una panoramica degli artefatti ottenuti dell'esecuzione di un processo, agevola il recupero all'interno dell'organizzazione per eventuali verifiche di integrità ed aderenza agli standard preposti.
 3. Il processo di assicurazione della qualità definisce quali operazioni compiere per assicurarsi che prodotto e processo siano conformi agli standard qualitativi fissati dall'organizzazione. Si procede definendo e pianificando le attività, e per ogni attività si definiscono standard, metodi, procedure e tools, quindi si procede con l'esecuzione delle attività definite.
 4. Il processo di verifica definisce le attività da compiere per determinare se l'esecuzione del processo è conforme al modello di processo adottato e se il prodotto è conforme al modello atteso in termini di costi, risorse, requisiti e vincoli.
 5. Il processo di validazione definisce le attività per la validazione del modello per stabilire se il modello costruito è quello voluto in base al modello teorico.

6. Il processo di revisione definisce le attività da compiere per valutare lo stato ed i prodotti di un'attività. È principalmente una fase che offre strumenti per i clienti in quanto permette sia di controllare l'evoluzione del progetto, che di assicurarsi che vengano rispettate le loro necessità. Ci sono tre tipi di revisione, revisione del contratto, revisione delle caratteristiche tecniche e revisione di accettabilità del prodotto in generale.
 7. Il processo di audit è una valutazione della conformità di determinati dati o procedure a determinati requisiti di processo e di contratto.
 8. Il processo di gestione dei problemi, definisce le attività necessarie ad individuare e risolvere i problemi.
- I processi organizzativi caratterizzano la struttura logica dell'organizzazione, quindi si organizza la ripartizione di compiti competenze responsabilità tra gli organi. Si suddividono in quattro categorie: gestione del processo, gestione delle infrastrutture, miglioramento e addestramento e formazione.
 1. Il processo di gestione del processo regola l'attività dei processi coinvolti nel ciclo di vita di un progetto.
 2. Il processo di gestione delle infrastrutture definisce come determinare e mantenere le infrastrutture per l'esecuzione dei processi.
 3. Il processo di miglioramento definisce come consolidare, misurare e controllare i processi coinvolti nel ciclo di vita di un prodotto software.
 4. Il processo di addestramento e formazione definisce come formare istruire e mantenere aggiornato il personale di un'organizzazione.

> nota audit

Un processo di audit è una valutazione o controllo di dati o procedure, all'interno di un'organizzazione le audit sono delle attività che hanno lo scopo di determinare la conformità di certi processi, strutture o procedure a determinate caratteristiche richieste.

> collegamento

Finora abbiamo visto i processi da una panoramica esterna, da un punto di vista dell'output prodotto. Abbiamo definito il processo di business come una rete di relazioni tra agenti che devono cooperare per raggiungere un determinato fine strategico per l'organizzazione, abbiamo visto quindi l'interazione tra processi ed i problemi che si creano dalla loro integrazione. Ora passiamo a considerare l'interno di un processo, astruendo sulla pratica legata al metodo implementato ed alle tecniche adottate. Vediamo quindi una descrizione di processo.

> descrizione di processo

Descrivere un processo vuol dire descrivere in maniera comprensibile, corretta, completa e non ambigua un insieme di operazioni che hanno lo scopo di trasformare un input in un output, considerando un certo insieme di decisioni ed una certa rete di relazioni tra applicazioni, tool, personale.

Una descrizione di un processo comprende molte componenti, si può fare una distinzione tra componenti obbligatorie e componenti variabili. La descrizione delle attività da seguire è una

componente di carattere obbligatorio, specifica le fasi da eseguire per raggiungere gli obiettivi, ad esempio analisi, progetto, codifica e test. Altre componenti come definizione dei ruoli, delle competenze, del contesto operativo dell'organizzazione, della struttura dei manufatti, dei tools di supporto e degli scenari procedurali sono invece componenti di carattere variante perché lasciano gradi di libertà al manager. Se da un lato questa caratteristica aggiunge elasticità e adattabilità nelle descrizioni di un processo, dall'altro comporta rischi di non conformità nelle descrizioni stesse.

Qualsiasi tipo siano le componenti, i dettagli presenti in ognuna devono essere tracciabili con quelli di qualsiasi altra componente con cui interagisce. Se ad esempio una certa componente prevede l'impiego di un certo tool, da qualche parte deve essere dichiarato il ruolo ed il livello di abilità della persona che utilizzerà quel tool, altro esempio se nella descrizione di un processo compare un certo documento dovrà anche esistere un o scenario procedurale in cui viene prodotto quel documento.

> la modellazione

Nella modellazione si passa a formalizzare la logica, quindi abbiamo dei linguaggi di modellazione PML che hanno lo scopo principalmente di aiutare a comprendere ciò che si sta progettando, fornire strumenti per simulazioni per poterne valutare l'andamento per eventuali ottimizzazioni, verificare l'impatto che le attività hanno sulle capacità e competenze dell'impresa e manipolare gli aspetti dinamici del processo.

Esempi di linguaggi di modellazione sono UML ed FSP.