

Processi di Sviluppo Software

Introduzione

Giuseppe Calavaro

Processi di sviluppo software - Agenda

- Differenza tra Programmazione e Progettazione SW
- I Processi di Sviluppo Software
 - Waterfall
 - Spirale
 - RUP e UP
 - XP e Agile
- Metodologie, Metodi, Modellazione e Processi
 - OMT + OOSE + Booch Method => UML e RUP
- RUP, Agile, SPEM ed evoluzioni in corso
 - UP – Unified Process
 - SPEM - Software & System Process Engineering Metamodel
 - EPF – Eclipse Process Framework
 - IBM Method Composer
 - DAD – Disciplined Agile Delivery
- Scrum e DAD in dettaglio
- RUP in dettaglio

L'informatica ed i primi computer

- I primi computer erano programmati in linguaggio macchina,
- Svolgevano compiti semplici, tipicamente calcoli, che richiedevano una grande potenza elaborativa
 - Calcoli balistici
 - Decifrazione di codici
 - Conteggio di dati a scopi statistici
- I compiti svolti dai computer erano descritti in un programma, che aveva tutte le istruzioni per leggere gli input, svolgere le elaborazioni e produrre un output
- Uno dei problemi principali era:
 - Come semplificare le attività di scrittura di queste istruzioni da dare al computer?
- L'informatica si concentrò quindi su
 - Linguaggi di programmazione
 - Algoritmisempre più evoluti per semplificare l'attività di programmazione

I sistemi informatici sono divenuti molto più complessi

- I programmi software sono componenti di applicazioni software
- Le applicazioni software sono componenti di sistemi software, eventualmente distribuiti e paralleli
- I sistemi software sono componenti di sistemi più complessi a vari livelli di aggregazioni
- Le varie applicazioni software sfruttano componenti già esistenti che forniscono servizi specifici
 - Essere stati sviluppati precedentemente da noi stessi
 - Questi possono essere acquistati (COTS) o disponibili gratuitamente
 - Database, Application Server, Web Server,
- Anche parte del codice sorgente potrebbe essere già sviluppato
 - Da noi stessi (noi o colleghi della stessa organizzazione)
 - Da parte di altri (Open Source)

Dalla Programmazione alla Progettazione

- Progettazione: Sviluppare le varie parti necessarie ad implementare un qualcosa complesso
 - Capire cosa si vuole ottenere e gestire il cambiamento di queste esigenze
 - Descrivere i requisiti iniziali e la loro evoluzione
 - Organizzare le attività
 - Sviluppare i vari sottoprodotti necessari per l'implementazione completa del sistema finale
 - Documentare tali sottoprodotti per una loro manutenzione ed evoluzione nel tempo
 - Documentare le modalità d'uso del prodotto

L'Ingegneria del Software

- Nel 1968, alla *NATO Science Committee Conference* fu coniato il nome “*Software Engineering*”
 - Fu riconosciuto che lo sviluppo di sistemi software diveniva sempre più complesso
 - Che al crescere della complessità il numero di progetti che fallivano cresceva esponenzialmente
 - Che era necessario stabilire delle pratiche solide scientificamente su come sviluppare sistemi software
- ***Ingegneria del Software***: L'applicazione di conoscenze scientifiche allo sviluppo e manutenzione dei sistemi software
- ***Software***: Tutti gli elaborati associati allo sviluppo, validazione, esercizio e manutenzione di sistemi software
 - Per esempio: il codice, la documentazione, i requisiti, il progetto, i manuali utente, i casi di test, l'elenco dei difetti, gli script d'installazione, etc.
- ***Processo di sviluppo software***: è la descrizione di chi fa cosa, come e quando per sviluppare il sistema software

Progettazione in Edilizia

La progettazione è articolata su tre livelli

- Progetto Preliminare
- Progetto Definitivo
- Progetto Esecutivo

La struttura tripartita rappresenta un principio inderogabile:

Il potere riconosciuto dalla legge al responsabile di progetto di integrare o modificare, previa congrua motivazione, le prescrizioni stabilite dei progetti preliminari, definitivi ed esecutivi, **NON può condurre all'azzeramento di una delle suddette fasi**, pena l'illegittimità dell'intera procedura

Il Regolamento dei Lavori Pubblici (Edilizia)

- Regolamento emanato con DPR n.554/1999 operativo dal 28 luglio 2000
- Sostituisce Regio Decreto n.350 del 1895
 - Ha rappresentato un punto di riferimento costante per gli operatori del settore
- “Dal confronto con il precedente ordinamento emerge come accanto a profonde innovazioni permangano forti elementi di continuità con il passato”
 - Motivi?
 - Forse l’edilizia è più semplice dell’informatica?
 - Più probabilmente è una scienza estremamente più matura
 - Sicuramente è determinante la concretezza dell’ingegneria edile rispetto all’ingegneria del software

I Processi di Sviluppo Software

- Nel tempo si è cercato di definire dei processi di sviluppo software che mettessero insieme le best practices comprese con lo sviluppo di progetti di successo
- Il modello a Cascata
 - Il modello a V
- Il modello a Spirale
- Il modello Iterativo
 - Il RUP / UP
 - XP – eXtreme Programming
 - Scrum
 - DAD – Disciplined Agile Delivery

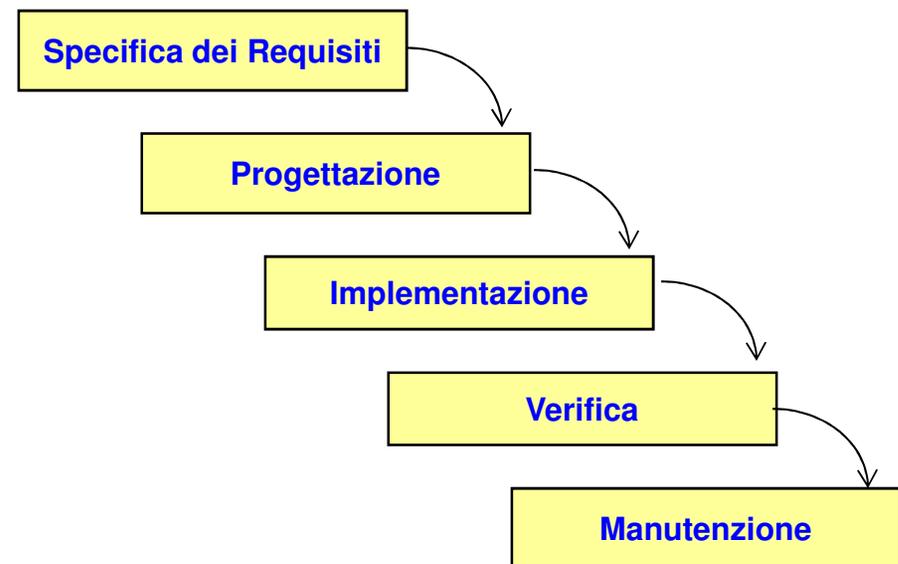
Il Modello a Cascata

- La prima descrizione formale del modello a cascata è tipicamente attribuita ad un articolo del 1970 di Winston Royce:
"Managing the Development of Large Software Systems"
 - In realtà nell'articolo non viene dato il nome di modello a Cascata (Waterfall)
- Il processo è diviso in passi che sono chiamate Fasi
- Ogni passo deve essere completamente terminato prima di passare al successivo

Passi per l'implementazione di un piccolo programma a scopi personali

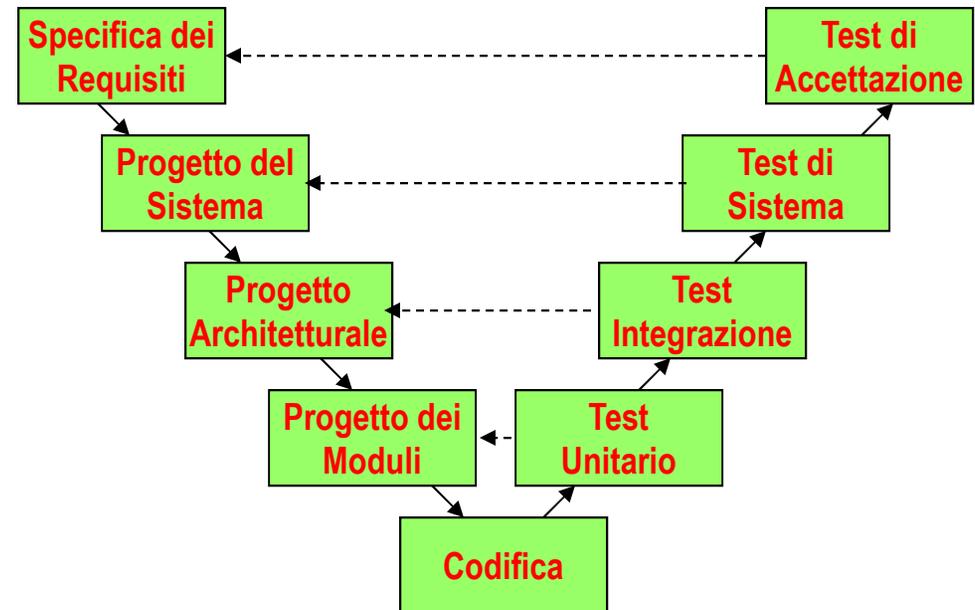


Passi per l'implementazione di un sistema complesso di grandi dimensioni da consegnare ad un cliente



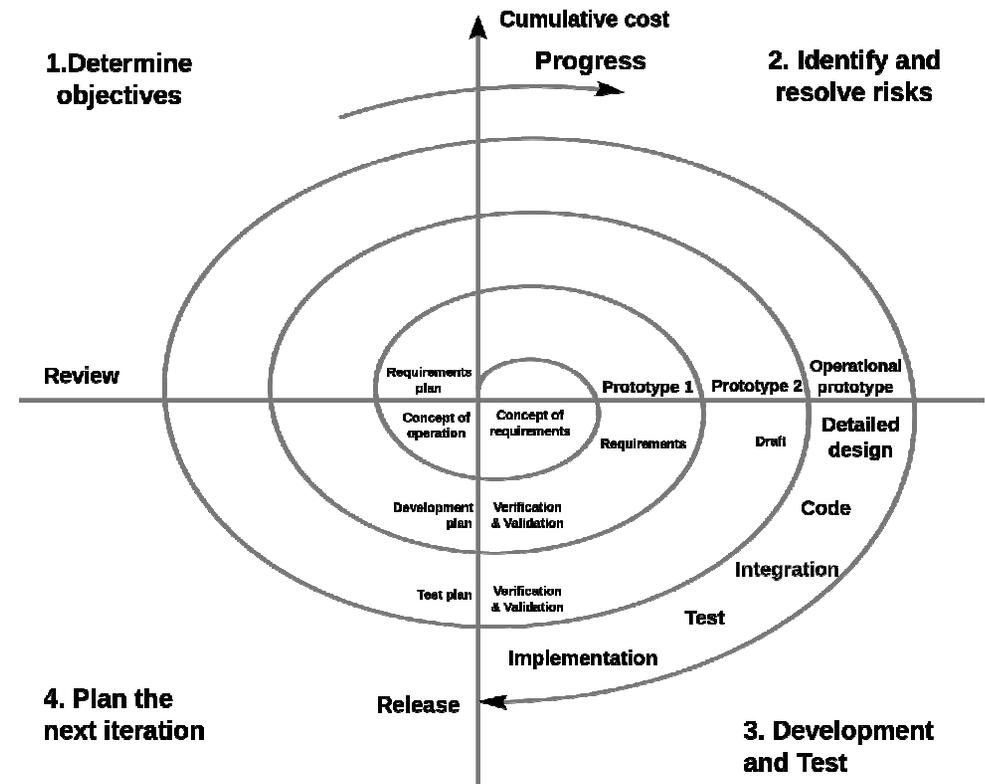
Il modello a V

- Il modello a V rappresenta un modello di sviluppo (non solo del software ma tipicamente di sistemi complessi che includono anche l'hardware) che può essere considerato una evoluzione del modello a cascata
- Il DOD-STD-2167A (evoluto in MIL-STD-498) è il più famoso esempio di standardizzazione di tale processo



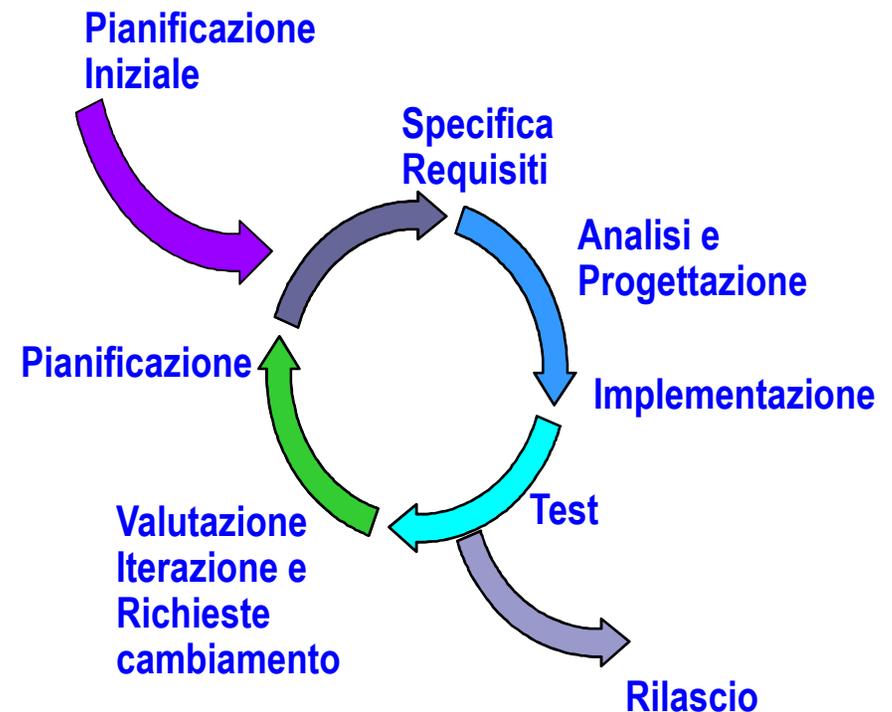
Il Modello a Spirale

- Il modello a spirale è un modello di processo guidato dalla gestione del rischio
- Il modello fu introdotto da Barry Boehm in un articolo del 1986 “A Spiral Model of Software development and Enhancement”
- Successivamente Boehm ha presentato questo modello come un “Generatore di modelli di processo”
 - Guida all’adozione di uno o più modelli di processo, quali il processo a cascata, incrementale, o evolutivo



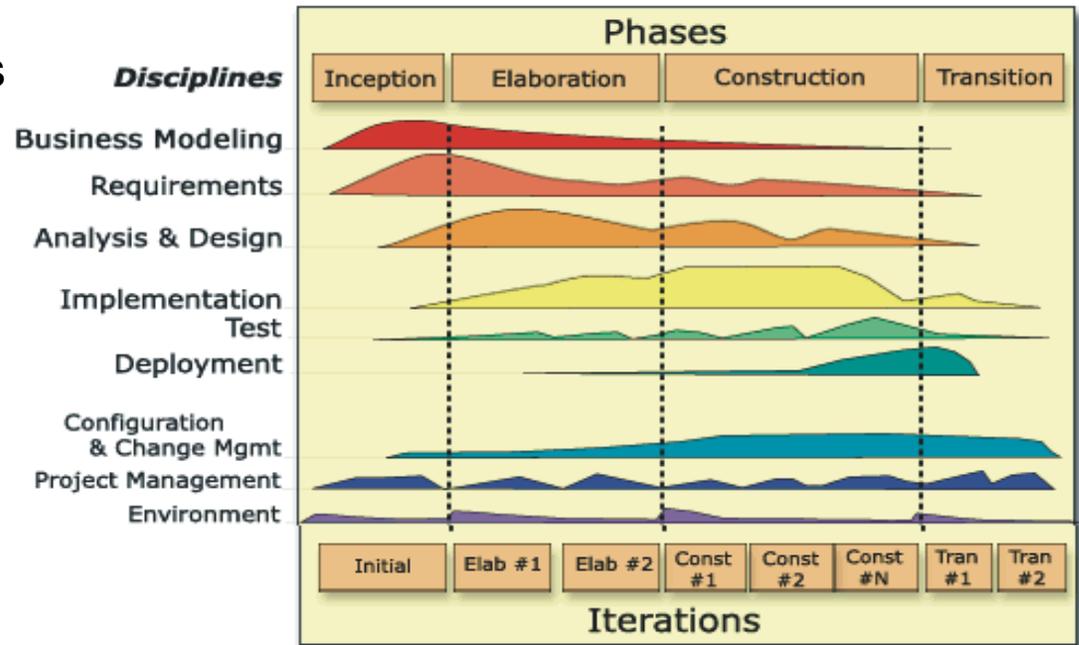
Il Modello Iterativo

- Il software viene sviluppato in maniera incrementale
- Guidato dallo sviluppo di versioni eseguibili
- I cambiamenti sono gestiti ad ogni iterazione
- La verifica di qualità avviene fin dall'inizio ed in maniera continuativa
- Ne esistono diverse varianti



Il RUP e UP

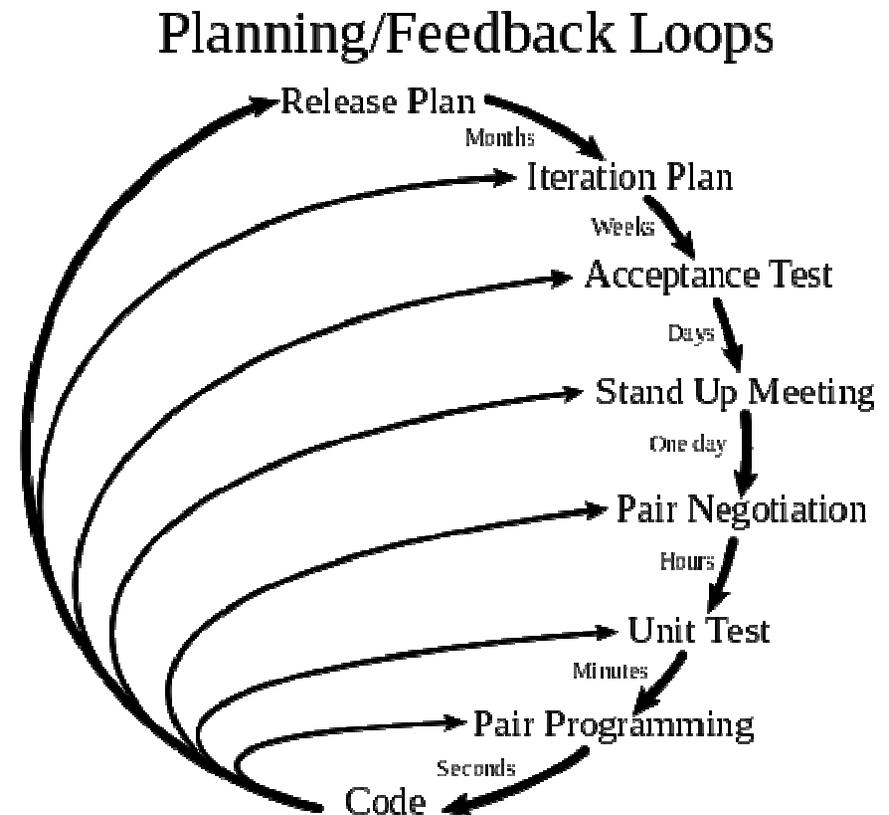
- RUP – Rational Unified Process
 - Sviluppato dalla Rational (ora brand di IBM)
 - Venduto come prodotto per istanziare un processo
- UP – Unified Process
 - Nome generico per fare riferimento al processo e non al prodotto IBM
- I due nomi sono spesso usati in maniera intercambiabile
 - Visto che entrambi rappresentano un metaprocesso che deve essere istanziato, è difficile dire se un processo è istanza dell'uno o dell'altro.



- Il RUP è uno degli output dell'unificazione dei "3 amigos"
 - OMT – Object Modeling Technique di Jim Rumbaugh
 - OOSE – Object Oriented Software Engineering di Ivar Jacobson
 - Booch Method di Grady Booch

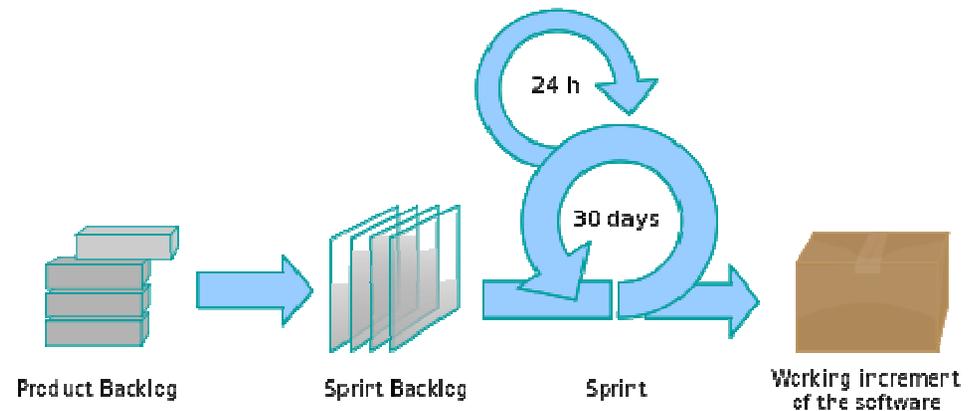
XP – eXtreme Programming

- E' una metodologia di sviluppo software creata da Kent Beck
 - IEEE COmputer nell'ottobre 1999 "Embracing Change with Extreme Programming"
- Ha l'obiettivo di migliorare la qualità del software e rispondere ai cambiamenti dei requisiti
- Si focalizza su cicli di sviluppo molto brevi
- Potremmo considerarla una possibile istanza dell'approccio Agile allo sviluppo del software



Scrum

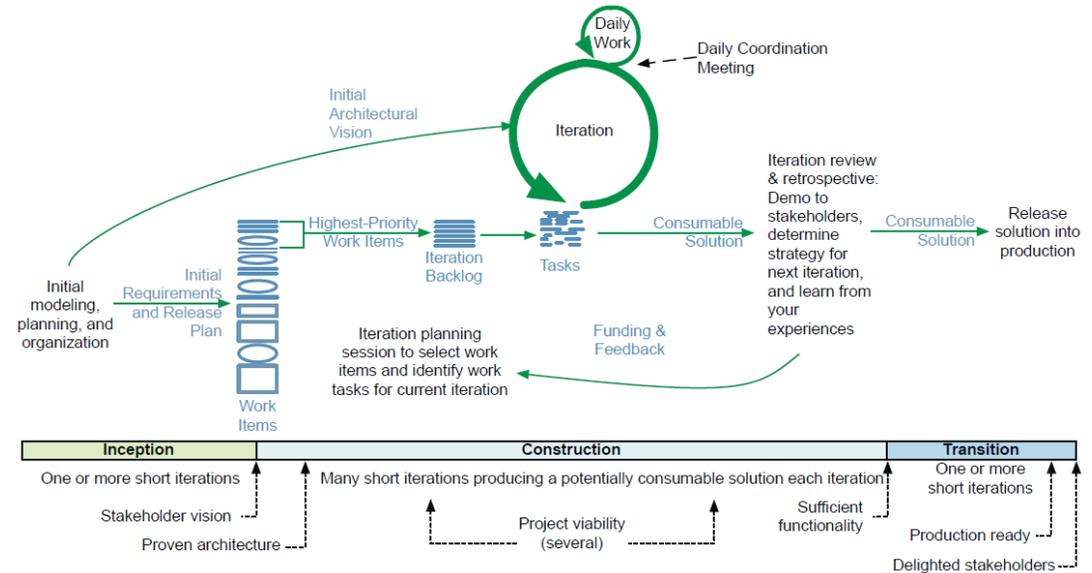
- E' uno dei processi Agile più diffusi
- Uno dei principi chiave è che il cliente cambia idea su cosa vuole durante lo sviluppo
- Adotta quindi un approccio empirico, accettando che il problema non può essere compreso nella sua interezza e si focalizza nello sviluppare velocemente versioni eseguibili dei requisiti a più alta priorità
- Le iterazioni sono chiamate Sprint e sono timeboxed
- I ruoli coinvolti nello sviluppo sono:
 - Product Owner
 - Team di sviluppo
 - Scrum Master



- Non ha un singolo padre, ma uno dei lavori miliari è stato il libro del 2001 di Ken Schwaber e Mike Beedle: "Agile Software Development With Scrum"

Disciplined Agile Delivery (DAD)

- DAD è un framework di decisione di processi basato su:
 - SCRUM
 - eXtreme Programming (XP)
 - Lean Software Development
 - Kanban
 - Unified Process (UP)
 - Agile Modeling (AM)
 - Agile Data
 - Enterprise Methods
 - Agile Development Practices
- Sviluppato da Scott Ambler in IBM tra il 2006 e il 2012



<http://disciplinedagileconsortium.org/>

Metodologie, Metodi, Modellazione e Processi

- Molti Ingegneri del Software che sono riusciti a portare avanti progetti software con successo hanno poi pubblicato il loro metodo o metodologia
- Quando i tre amigos si riunirono in Rational, inizialmente volevano unificare le loro metodologie, ma sapevano che una metodologia unica non poteva rispondere a tutte le esigenze dei progetti software così disparati
 - Dal software per gestire sistemi missilistici, al gestionale di una banca, di un magazzino o di una centrale telefonica, ad un video gioco, o all'App su un telefonino
- Per questo hanno creato
 - UML – Unified Modeling Language
 - Un linguaggio di modellazione universale per la descrizione degli elaborati di progettazione del software
 - RUP – Rational Unified Process
 - Un meta-processo che potesse essere modificato, a seconda degli ambiti di progetto da sviluppare,
 - basato su best practices sviluppate dall'esperienza di tanti progetti
 - e che avesse un repository di tecniche di progettazione e template di elaborati da scegliere o personalizzare a seconda delle esigenze

Evoluzione del RUP

Il RUP è evoluto in 3 direzioni:

1. Influenzando la standardizzazione di OMG del metaprocesso SPEM
 - Dal quale sono nati processi come OpenUP, Eclipse Open Framework (EOF) etc.
 - <http://www.omg.org/spec/SPEM/>
2. In un prodotto per la definizione di processi di sviluppo
 - IBM Rational Method Composer
 - Strumento venduto da IBM che permette di fare il tailoring e pubblicare il proprio processo adattato alle proprie esigenze ed obiettivi
 - <http://www-01.ibm.com/software/rational/rup/>
3. Facendo da padre putativo a vari altri metodi di sviluppo
 - Evoluzioni ed adattamenti di processi, pratiche e tecniche emergenti, come per esempio quelle tipiche dei movimenti di “Sviluppo Agile”, al RUP
 - Per esempio: Unified Process (UP), Agile Modeling (AM), Disciplined Agile Delivery (DAD) e Enterprise Unified Process (EUP)
 - http://en.wikipedia.org/wiki/Unified_Process
 - <http://www.agilemodeling.com/>
 - <http://www.disciplinedagiledelivery.com/>
 - <http://www.enterpriseunifiedprocess.com/>

SPEM - Software & System Process Engineering Metamodel

- E' un metamodello per descrivere processi e framework di processi definito da OMG
 - Scaricabile gratuitamente da: <http://www.omg.org/spec/SPEM/>
- Utilizzato per definire IBM Method Composer
- Sulla specifica formale, oltre ai riferimenti al RUP, si trova:
 - Esempio di come il “Microsoft Solution Framework (MSF) Agile” può essere rappresentato usando SPEM 2.0
 - Descrizione di come il progetto EPF-Eclipse Open Framework (www.eclipse.org/epf), che è un progetto open source, fornisce un framework di processi basati su SPEM 2.0
 - OpenUP/Basic è un metodo di sviluppo software minimale, completo ed estensibile, sviluppato utilizzando EPF

EPF – Eclipse Open Framework

- I due goal di EPF: (<http://projects.eclipse.org/projects/technology.epf>)
 1. Fornire un framework estensibile e strumenti esemplari per la definizione, configurazione e pubblicazione di processi di sviluppo software
 2. Fornire dei processi esemplari ed estensibili che supportano lo sviluppo iterativo, agile e incrementale
- Vi possiamo trovare i seguenti processi (scaricabili dal sito)
http://www.eclipse.org/epf/downloads/configurations/pubconfig_downloads.php
 - **OpenUP**

E' una versione snella del processo UP, che applica un approccio iterativo ed incrementale ad un ciclo di vita strutturato. OpenUP adotta un approccio pragmatico dell'Agile che si focalizza sulla natura collaborativa dello sviluppo software. Questo processo è indipendente da strumenti a suo supporto ed è un processo a basso cerimoniale che può essere esteso per indirizzare una grande varietà di tipi di progetti.
 - **Scrum**

Framework di gestione dei processi Agile empirico usato per sviluppare in maniera iterativa e rilasciare al cliente incrementale di valore sempre maggiore. Scrum si basa su team che si organizzano autonomamente e su un cliente, detto Product Owner, che fornisce al team la lista delle feature desiderate usando un meccanismo di priorità basato sul valore di business.
 - **XP – eXtreme Programming**

Processo di sviluppo che può essere usato da team di sviluppo di piccole o medie dimensioni per lo sviluppo di software di alta qualità con tempistiche e budget predicibili e minimo overhead

IBM Rational Method Composer

- E' l'evoluzione di RUP
- E' il prodotto IBM che permette di creare istanze di processo
- Ha un repository di processi basati sulle best practice adottate da migliaia di progetti nel mondo
- Permette di evitare il reinventare da zero un processo, riusando ciò che è stato implementato con successo e generalizzato dagli specialisti Rational
- Può comunque mettere a disposizione dei processi pronti-all'uso senza personalizzazioni, per iniziare velocemente e poi modificare il processo sulla base delle proprie esperienze e lessons-learned.

Unified Process (UP)

- http://en.wikipedia.org/wiki/Unified_Process
- OpenUP
 - Una istanza agile di UP che contiene l'insieme minimo di pratiche per adottare un processo di sviluppo agile, snello e leggero.
 - Essendo un processo minimale, non copre molti argomenti che un team di progetto di medie-grandi dimensioni potrebbe avere l'esigenza di coprire

SEMAT – Software Engineering Method and Theory

- E' una iniziativa lanciata nel dicembre 2009 da Ivar Jacobson, Bertrand Meyer e Richard Soley per rilanciare l'ingegneria del software come una disciplina rigorosa scientificamente
- Il lavoro si divide in 4 aree fortemente collegate:
 - La Pratica
 - Le cose che abbiamo, produciamo o utilizziamo durante lo sviluppo
 - La Formazione
 - La creazione e divulgazione di percorsi di formazione
 - La Teoria
 - La teoria generale dell'ingegneria del software
 - La Comunità
 - Comunità regionali o nazionali che sviluppano SEMAT nei propri territori
- <http://semat.org/>