

RUP

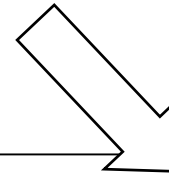
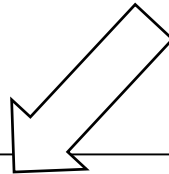
(Rational Unified Process)

Caratteristiche, Punti di forza, Limiti

versione del tutorial: 3.3 (febbraio 2007)

Unified Process

Booch, Rumbaugh, Jacobson



UML

(Unified Modeling Language)

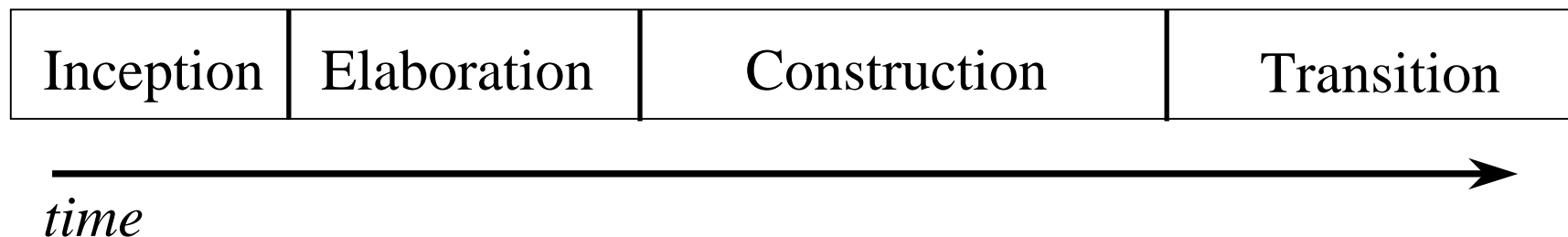
- notazione per rappresentare i sistemi software
- standard OMG (Object Management Group) dal 1997

UP (Unified Process)

- non è uno standard
- è il processo definito dai tre autori (e dalla loro società, Rational - IBM) per l'utilizzo di UML
- versione commerciale: RUP (Rational Unified Process)

Unified Process

- E' un framework (schema generale) di processo, da adattare alle diverse tipologie di progetto
- Caratteristiche primarie: approccio iterativo e incrementale, guidato dai casi d'uso, incentrato sull'architettura



Guidato dai casi d'uso

- La definizione delle modalità d'uso del sistema (casi d'uso) guida tutte le attività del processo
- In particolare, i casi d'uso costituiscono la base per:
 - la definizione e negoziazione dei requisiti, e la loro validazione da parte del committente
 - la progettazione dell'architettura e dei componenti
 - la definizione dei test di accettazione
 - la pianificazione dei rilasci (in un'ottica incrementale) e quindi del progetto

Centrato sull'architettura

- La definizione dell'architettura applicativa e tecnologica costituisce il fondamento tecnico dell'applicazione e del progetto
- Il consolidamento dell'architettura avviene solo quando si è certi (tipicamente, tramite sperimentazioni) della sua fattibilità tecnica
- Fino a quando l'architettura non è consolidata, non esistono elementi sufficienti per determinare (con precisione sufficiente alla definizione di un contratto) tempi, costi e rischi dell'intervento progettuale

Iterativo e incrementale

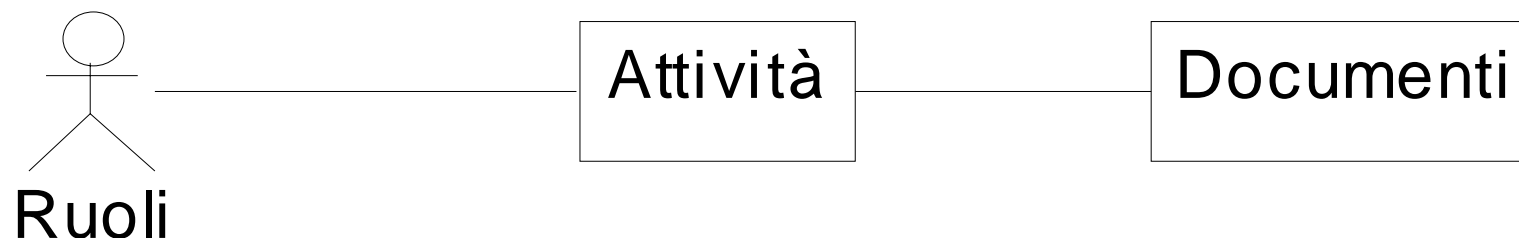
- Iterativo:
 - il progetto si articola in una serie di iterazioni (periodi di lavoro), con lo scopo di ridurre progressivamente i rischi, a partire da quelli principali (es. incomprensioni sui requisiti, incertezze sull'architettura)
 - in ogni iterazione si svolgono, in misura e percentuali diverse, le tipiche attività di sviluppo (es. gestione dei requisiti, design, implementazione, test)
- Incrementale:
 - la realizzazione (ed eventualmente il rilascio) dell'applicazione avviene in modo progressivo
 - la pianificazione è guidata dai casi d'uso e dalle priorità architettoniche (es. precedenza ai componenti infrastrutturali)

RUP

- RUP (Rational Unified Process) è la versione commerciale di Unified Process – è un prodotto di IBM - Rational
- E' utilizzato in contesti di business estremamente competitivi, e in ambiti applicativi eterogenei
- Risponde agli obiettivi primari di time to market, controllo del rischio e visibilità degli stati avanzamento

Cosa c'è in RUP

- un'organizzazione del piano di progetto per fasi sequenziali
- indicazioni sulle attività da svolgere e sulla loro sequenza
- un insieme di ruoli predefiniti
- un insieme di documenti da produrre, con template ed esempi già realizzati



Regole del gioco, adattabili

- RUP è un insieme di **regole del gioco** da seguire nel lavoro quotidiano nei progetti
- alcune (poche), obbligatorie
- la maggioranza, da adattare alle caratteristiche dello specifico progetto

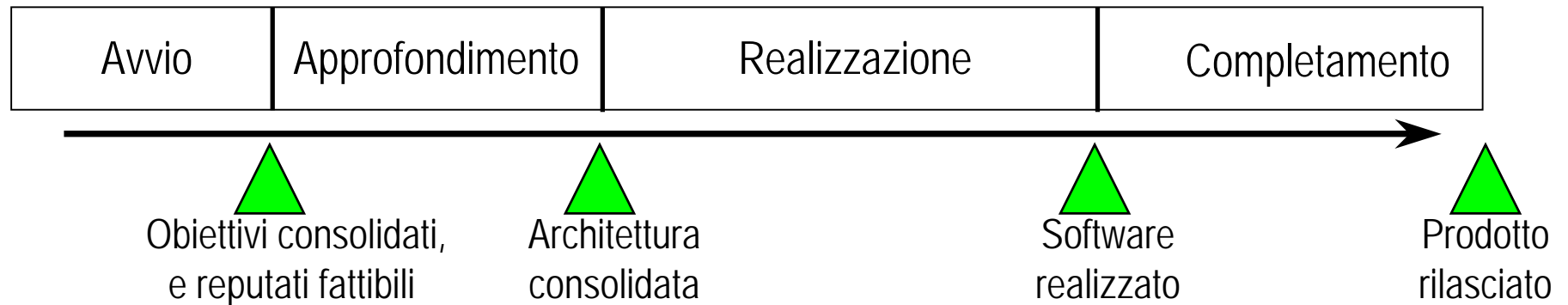
⇒ ogni progetto è diverso dagli altri

⇒ ma è necessario che ogni progetto, per avere successo, segua alcuni principi e pratiche fondamentali

I principi di RUP

1. Affrontare i rischi fin dall'inizio, in modo sistematico.
2. Fornire risultati di valore ai propri clienti e ai propri utenti.
3. Produrre al più presto software funzionante, e farlo evolvere in modo incrementale.
4. Predisporre fin dall'inizio a gestire i cambiamenti.
5. Consolidare il prima possibile un'architettura eseguibile.
6. Costruire il sistema con un approccio che massimizzi la manutenibilità e la riusabilità dei componenti.
7. Lavorare insieme, come un unico gruppo di lavoro.
8. La qualità è nel modo concreto di lavorare, non nella compilazione di documenti a parte.

Fasi di RUP



- **Avvio** (Inception) – riduzione rischi su obiettivi - definizione dei limiti, della fattibilità e della giustificazione economica dell'intervento
- **Approfondimento** (Elaboration) – riduzione rischi architettureali - approfondimento dei requisiti e produzione di un primo incremento del sistema
- **Realizzazione** (Construction) – riduzione rischi di non finire in tempo utile - produzione incrementale del sistema (completamente testato e integrato)
- **Completamento** (Transition) – riduzione rischi di rilasciare un prodotto non accettabile - rimozione ultimi errori, e produzione della versione rilasciabile del sistema

Fase 1: Avvio (Inception)

Può essere considerata chiusa quando il Committente e tutte le parti interessate concordano che:

- gli obiettivi ed i confini del progetto (cosa è da fare, cosa non lo è) sono definiti, ed esistono indicazioni chiare sulle priorità
- le stime di tempi e di costo sono sensate (anche se imprecise)
- i requisiti principali per il progetto sono stati identificati
- i rischi principali per il progetto sono stati identificati, e per ciascun rischio esiste una strategia per evitarlo o contenerlo

➤ Se il progetto non riesce a raggiungere questa pietra miliare è opportuno cancellarlo, o ridimensionarlo notevolmente, o rimandare l'inizio della fase di Approfondimento (Elaboration).

Fase 2: Approfondimento (Elaboration)

Può essere considerata chiusa quando il Committente e tutte le parti interessate concordano che:

- i requisiti non funzionali (caratteristiche di qualità del prodotto) sono definiti, e considerati stabili
- il piano di progetto e le stime dei costi sono aggiornati, sufficientemente dettagliati e credibili per la realizzazione delle funzionalità richieste
- l'architettura (organizzazione interna del sistema, funzionamento delle tecnologie scelte) è stabile
- il test e la valutazione di prototipi hanno dimostrato che i principali elementi di rischio sono stati affrontati e risolti

➤ Se il progetto non riesce a raggiungere questa pietra miliare è opportuno cancellarlo, o ridimensionarlo notevolmente, o rimandare l'inizio della fase di Realizzazione (Construction).

Fase 3: Realizzazione (Construction)

Può essere considerata chiusa quando il Committente e tutte le parti interessate concordano che:

- il Prodotto software è già completamente realizzato, integrato e testato
- Se il progetto non riesce a raggiungere questa pietra miliare è opportuno rimandare l'inizio della fase di Completamento (Transition), e pianificare invece estensioni alla fase di Realizzazione per la messa a punto del prodotto.

Fase 4: Completamento (Transition)

Può essere considerata chiusa quando:

- il Committente è soddisfatto
 - tutte le parti interessate sono d'accordo nel considerare completato il progetto
- A questo punto, il progetto è concluso. Modifiche ed evoluzioni successive innescano un nuovo progetto di manutenzione evolutiva.

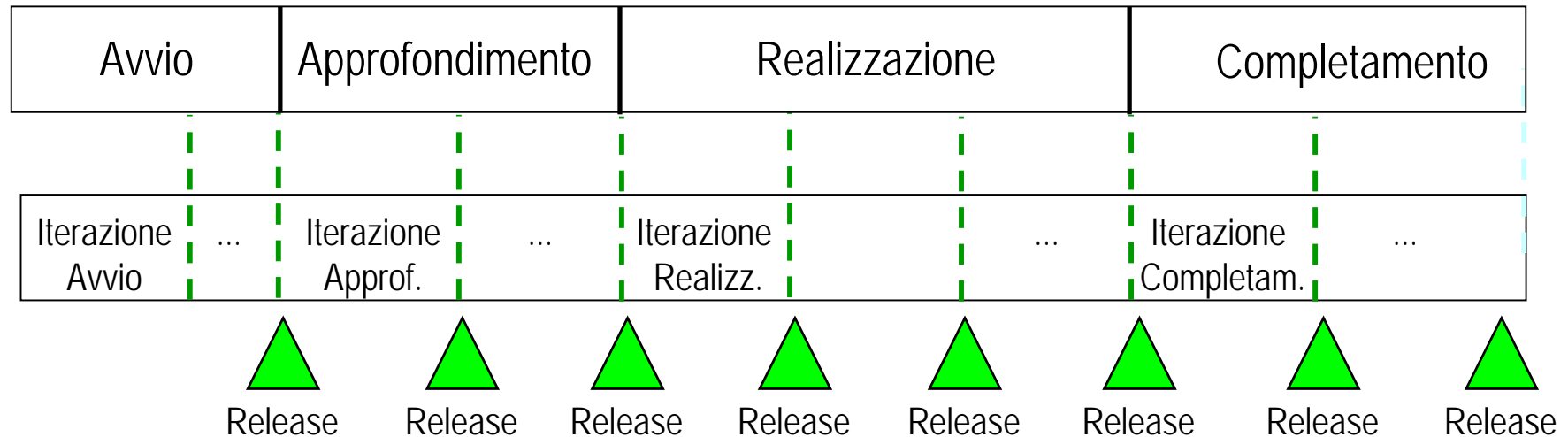
Durata ed effort delle fasi in un progetto “medio”

	Avvio	Approfondimento	Realizzazione	Completamento
Durata (tempo trascorso)	10%	30%	50%	10%
Effort (gg/persona)	5%	20%	65%	10%

stima
costi e tempi

 T_0  T_1  T_2

RUP: fasi e iterazioni

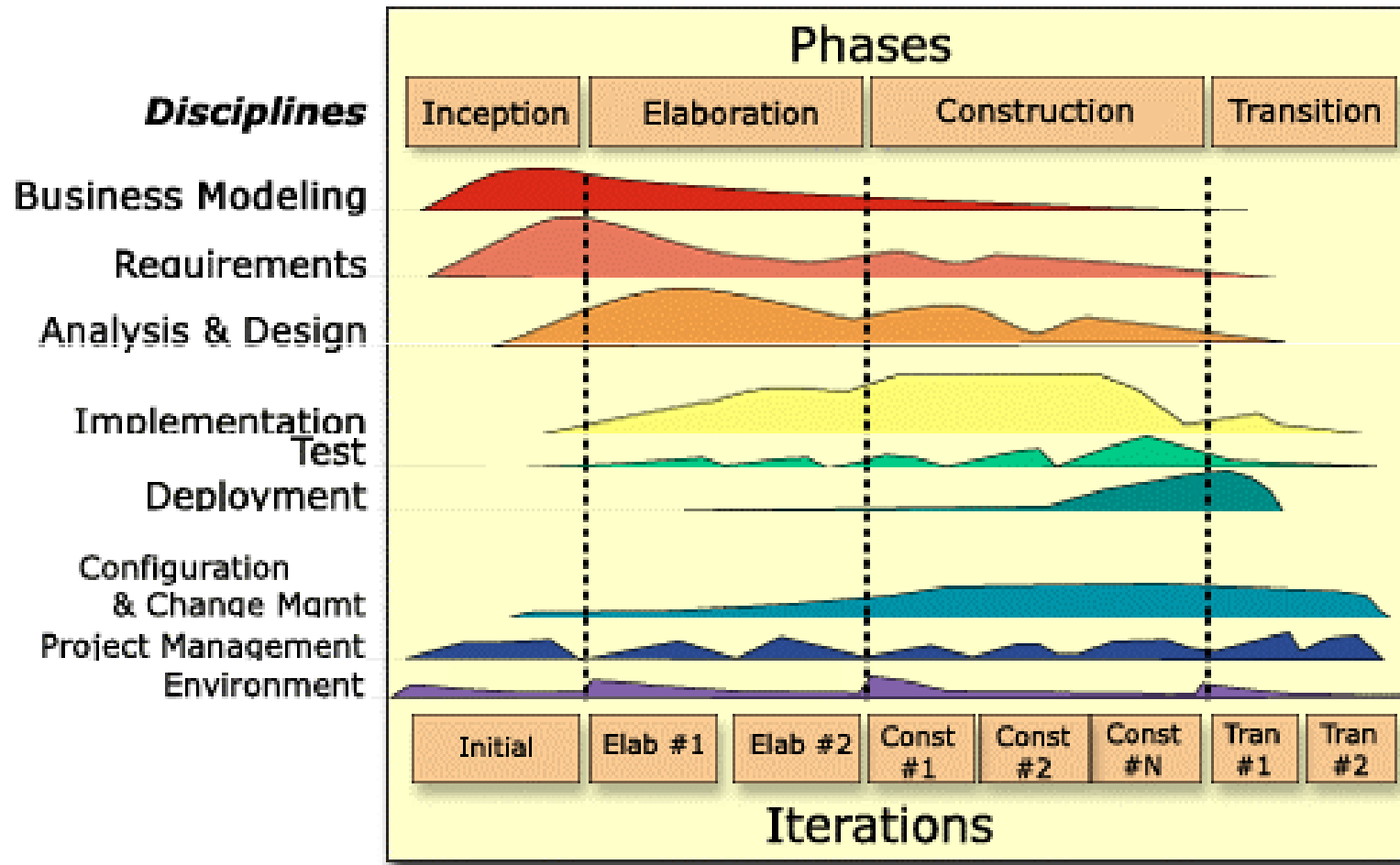


- le iterazioni sono unità di pianificazione a “grana fine”
- ogni iterazione ha un piano, obiettivi e criteri di valutazione
- in ogni iterazione si svolgono attività di gestione requisiti, analisi, progettazione, codifica e test
- ogni iterazione produce un risultato concreto (e gestito in configurazione)

RUP: tipologie di attività (discipline)

- **Business Modeling** - quando è opportuno chiarire ruoli e responsabilità
- **Requirements**
- **Analysis and Design** - se il sistema non viene acquisito
- **Implementation** - se il sistema non viene acquisito
- **Test**
- **Deployment** (attività propedeutiche al rilascio)
- **Configuration & Change Management**
- **Project Management**
- **Environment** (adattamento processo e ambiente di lavoro)

Fasi e tipologie di attività



Fasi e tipologie di attività

Attenzione!

- Le fasi sono **sequenziali**, e corrispondono a milestone significativi per Committenti, Utenti, Management
- Le discipline (tipologie di attività) **non sono sequenziali**, e vengono svolte dal progetto in ogni iterazione
- Il numero delle iterazioni dipende dalle scelte del Project Manager e dai rischi del progetto

Ripartizione indicativa effort sulle discipline nelle diverse fasi RUP

Disciplina	Inception	Elaboration	Construction	Transition	TOTALI
Project Management	14%	12%	10%	14%	11%
Requirements	38%	18%	8%	4%	11%
Analysis - Design	19%	36%	16%	4%	19%
Implementation / Unit Test	8%	13%	34%	19%	27%
Test	8%	10%	24%	24%	20%
Deployment	3%	3%	3%	30%	6%
Environment / CM	10%	8%	5%	5%	6%

fonte: Cocomo II (Barry Boehm - dati verificati da Boehm con Rational)

Nota: sono numeri da prendere “con le molle”: ogni progetto ha le sue specificità

RUP: tipologie di progetto

- Nuovo sviluppo di dimensioni rilevanti
- Nuovo sviluppo di dimensioni medie-piccole
- Manutenzione evolutiva
- ...
- Manutenzione correttiva

⇒ Ad ogni tipologia di progetto corrisponde una diversa modalità di percorrenza delle fasi di RUP

⇒ es. le fasi di Avvio (Inception) ed Approfondimento (Elaboration) per una manutenzione correttiva possono durare alcuni minuti...

Punti di forza di RUP

- E' un processo ampiamente utilizzato, da anni, in contesti eterogenei: è sperimentato e consolidato
- Comprende una massa notevole di linee guida e template: fornisce anche indicazioni immediatamente operative
- Definisce in modo approfondito (anche se in termini necessariamente generali):
 - i ruoli coinvolti nel processo di sviluppo
 - le attività da effettuare
 - input e output per ogni attività

Limiti di RUP (1)

È un processo il cui ambito è esclusivamente il singolo progetto.

Non tratta aspetti che escono dall'ottica del progetto singolo:

- l'evoluzione di un sistema, dall'ideazione alla fine dell'utilizzo, attraverso una sequenza di progetti
- le attività da svolgersi a livello "enterprise" (modelli di dominio, librerie di componenti per il riuso, gestione qualità, gestione in produzione, ecc.)

Limiti di RUP (2)

- RUP è un framework di processo generico, non mirato ad alcuna tipologia specifica di applicazioni
- Ha origine in una cultura di sviluppo mirata alla realizzazione di prodotti commerciali, e quindi non approfondisce:
 - la gestione dei rapporti contrattuali tra committenti “business” e sviluppatori
 - la gestione dei rapporti contrattuali con fornitori
 - l’acquisizione (ed eventuale personalizzazione) di package commerciali

Limiti di RUP (3)

- RUP è notevolmente complesso. Per potersi adattare a qualunque esigenza particolare, definisce decine di ruoli, centinaia di task e di documenti, e li descrive in migliaia di pagine.
- L'utilizzo di RUP "as is" (senza adattamenti) è assolutamente sconsigliabile, in quanto si corre il rischio di utilizzarlo in modo burocratico, inefficace e inefficiente.
- RUP è forse troppo flessibile: anche se nasce come processo iterativo, diverse organizzazioni lo interpretano stiracchiandolo in un approccio a cascata.

RUP va personalizzato

- RUP deve essere adattato, senza stravolgimenti, alle esigenze e alle caratteristiche dell'organizzazione
- L'adattamento deve tenere conto di vari fattori:
 - Tipologia di sistemi da realizzare (automazione, gestionale, e-commerce)
 - Cultura dell'organizzazione in cui viene calato
 - Struttura organizzativa, ruoli, responsabilità
 - Modalità di rapporto con i Committenti
 - Differenze tra progetti di nuovo sviluppo ed evoluzioni
- L'adattamento deve avvenire in modo progressivo, e costituisce a sua volta un progetto, iterativo e incrementale

Derivazioni di RUP

- La complessità di RUP, ed il fatto che possa essere interpretato in modo burocratico, ha portato alcuni metodologi a cercare di integrarlo con approcci più agili, in particolare XP e Scrum.
- Le derivazioni di RUP più interessanti sono al momento due:
 - OpenUp/Basic, in ambito Eclipse
 - Essential UP, creato da Ivar Jacobson

RUP - Costi

- Impatto organizzativo
- Impatto culturale
- Spese tecnologiche



- Costi di impianto
- Costi a regime

Costi - Impatto organizzativo

- Il processo può risultare molto diverso dagli “stili” di lavoro in essere nell’organizzazione che lo adotta
- Può portare ad una ridefinizione dei ruoli, rispetto a quelli in essere
- Può portare ad una ridefinizione delle modalità di comunicazione e di rapporto con Committenti e Utilizzatori
- Può portare ad una ridefinizione delle modalità di comunicazione e di rapporto con eventuali Fornitori

Costi - Impatto culturale

RUP può comportare, per il personale dell'organizzazione:

- Un cambiamento significativo del modo di lavorare (procedure, responsabilità, tecniche)
- Una diversa percezione del proprio ruolo in rapporto alla committenza ed agli utilizzatori
- Una ridefinizione di alcuni concetti cardine dello sviluppo (analisi, requisiti, test, manutenzione)
- La necessità di apprendere nuove tecniche

Costi - Tecnologici

- L'utilizzo del nuovo processo può essere in alcuni casi agevolato dall'acquisizione di strumenti specifici per le attività primarie di sviluppo, tra cui:
 - gestione dei requisiti e delle richieste di cambiamento
 - analisi e design
 - generazione di codice
 - testing
- Tali strumenti sono oggi ampiamente disponibili sul mercato
- La disponibilità degli strumenti non è comunque un prerequisito indispensabile per l'efficacia del processo

Costi - Impianto

Lavoro necessario per:

- adattamento del processo
- interazione con Funzioni aziendali coinvolte, e con eventuali soggetti esterni (Clienti, Fornitori)
- divulgazione
- supporto ai progetti nella fase iniziale di diffusione

Costi - a Regime

- Il processo deve essere gestito in un'ottica di continuo miglioramento e adattamento all'evolversi delle esigenze aziendali
- Va, inoltre, adattato ad ogni specifico progetto, per tenere conto delle specifiche caratteristiche e limitazioni

Bibliografia di base su RUP

- Ivar Jacobson, Grady Booch, James Rumbaugh: *Unified Software Development Process* , Addison-Wesley 1999
- Philippe Kruchten : *The Rational Unified Process. An Introduction. 3rd Edition.* , Addison-Wesley 2003 - disponibile anche in traduzione italiana, con lo stesso titolo, in una edizione precedente (2000).
- Per Kroll, Philippe Krutchen: *The Rational Unified Process Made Easy. A Practitioner's Guide to the RUP* , Addison-Wesley 2003
- Walker Royce : *Software Project Management. A Unified Framework* , Addison-Wesley 1998