

Corso "Tipologie di progetti e loro ciclo di vita"

Bibliografia

- G. Rossi "Project Management" ISEDI
- A. Andriano "Il sistema informativo gestionale di produzione" F. Angeli
- G. Battistella "La qualità nel software" F. Angeli
- A. Pizzarello "Ingegneria del software" F. Angeli
- B. Dickinson "Developing quality systems" McGrawHill
- X. Castellani "Metodo generale di analisi di un'applicazione informatica" MASSON
- T. DeMarco "Peopleware"
- Feynman "What do you care what other people think?" Norton
(parte 2ª)
- Rockart-DeLong "Executive Support Systems" Dow Jones Irwin

*TIPOLOGIE DI PROGETTI
E
LORO CICLO DI VITA*

1. Introduzione
2. Classificazione
3. Ciclo di vita
4. Ciclo e classificazione
5. Elementi di analisi
6. Conclusioni



Tipologie di Progetti

e

loro ciclo di vita

1. Introduzione
2. Classificazione
3. Ciclo di vita
4. Ciclo e classificazione
5. Elementi di analisi
6. Conclusioni

Italo ~~Medium~~
↑ 1 righe

TITOLO LARGE ITALICO

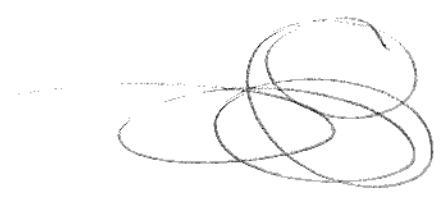
↕ 4 righe Large

18 ~~20~~ large

↓ 1 riga, 2 per titolo

~~3~~
3 sistema medio

25
vnl



Progetto software

idealmente simile
a progetti di tipo
ingegneristico
(strade, edifici)...

ma praticamente
molto differente

Progetto software

- *Elevata* autonomia produttiva
- *Ridotto* controllo tempi/costi
- *Altissima* importanza dei produttori finali e loro relativa insostituibilità

Progetto software

IERI

- Soddifazione esigenze singoli utenti
- Scarsa integrazione

Prodotto software



"opera d' arte"

Progetto software

OGGI

- tendenza ad usare
 - metodi ingegneristici
 - regole ("standard")

Prodotto software



bene industriale

Progetto software

DOMANI

Le persone

- forte riduzione dei programmatori
- maggiore attenzione e coinvolgimento a livello funzionale

I metodi

- controllo tempi e costi di stile industriale
- pianificazione dei risultati

Progetto software

DOMANI

I prodotti

- attenzione ai clienti/
utenti
- alta velocità di modifica
ed automazione della
codifica (CASE, ecc)

I rischi

- riprodurre errori di
gestione già compiuti
in altri ambiti
- effetto cascata degli
errori nei moduli base
(CASE, ecc)

2. Classificazione

Progetti

Classificabili per

- caratteristiche di gestione
- modalità produttive
- tipologia di consegna
- tipo di prodotto finale

Influenti attualmente

- tipologia di consegna
- caratteristiche di gestione

2. Classificazione

Caratteristiche di gestione

- chi si assume i rischi?
- chi gestisce il progetto?

Assunzione dei rischi/costi

- del fornitore
- del committente
- forfettaria del fornitore
- mista

2. Classificazione

Modalità produttive

classificazione di
provenienza e uso in
campo industriale, per
produzioni ripetitive
di prodotti normalmente
non modificabili

Produzione

- a lotti

- a commessa

- a ciclo continuo

Tipologia di consegna

classificazione di
provenienza più
prettamente ingegneristica
e impiantistica.
Normalmente significativa
per prodotti ingegnerizzati
ma non ripetitivi ("custom")

Consegna

- a rilasci successivi
- a consegna finale del
prodotto finito ("as is")
- a consegna finale del
prodotto finito, con
possibilità di varianti
e adattamenti

Tipo di prodotto finale

classificazione che
riporta quasi alle
origini "artigianali"
del software

Prodotto finale

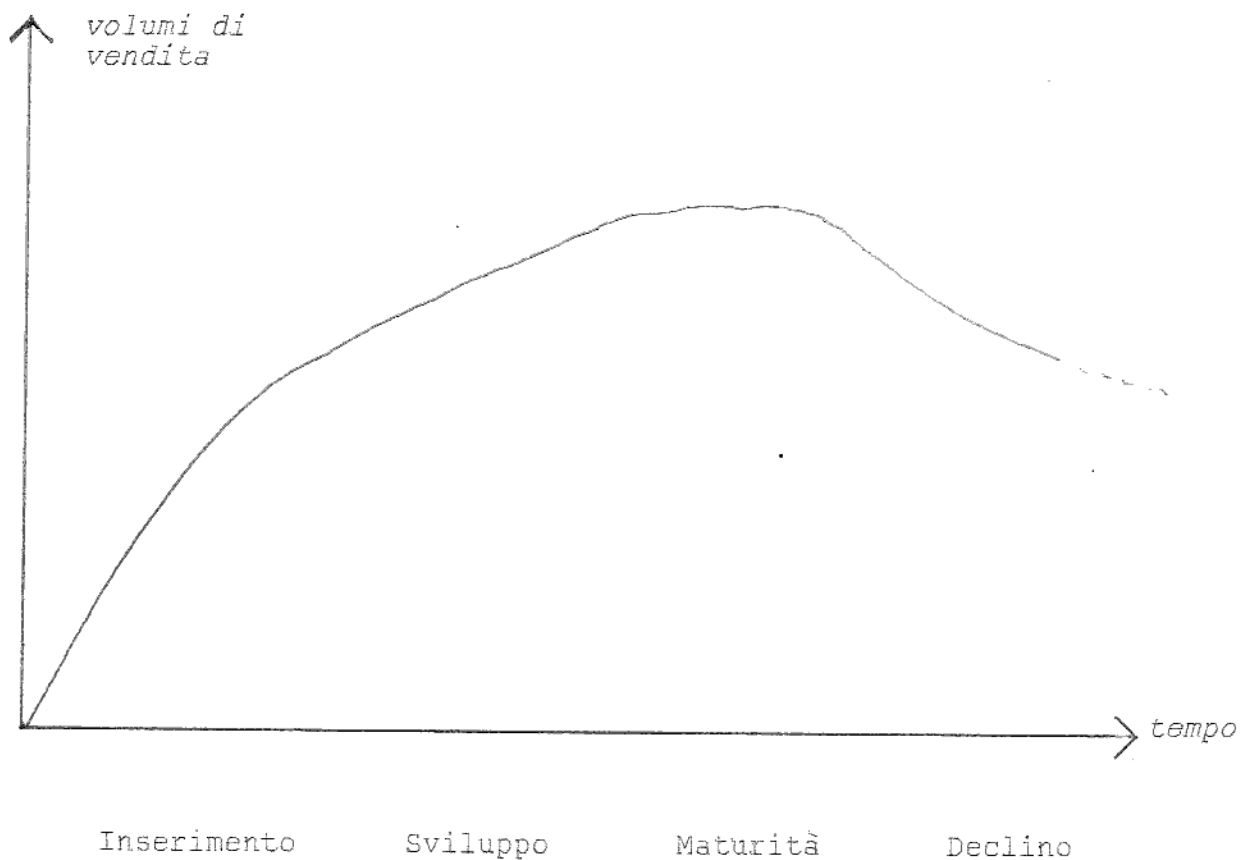
- prodotto base ("standard")
- prodotto base con
modifiche ("customized")
- prodotto su specifiche
dell'utente ("custom")
- manutenzione di prodotto
preesistente (di vario tipo)

2. Classificazione

Influenti attualmente

- Pacchetti
- Custom
- Chiavi in mano
- Manutenzione
- Gestione procedure e centro
- Progetti pilota

Ogni prodotto è caratterizzato dall'attraversare alcune fasi:



ogni fase presenta differenti problemi di gestione. Dal punto di vista produttivo, analizzeremo le prime due.

Fasi

per fasi di un progetto
intenderemo i vari passi
da svolgere per arrivare
al prodotto finale

Fasi iniziali

ignoreremo le fasi
precedenti l' inizio
del progetto:

- contatto
- contratto
- definizione di massima
degli obiettivi

Analizzeremo ora le fasi di un generico progetto EDP, rinviando la discussione delle differenze per classificazione.

Progetto EDP

Nasce da

esigenze del committente,
indicate nel contratto o
nella bozza di contratto

richiede

definizione del contesto
delle risorse
dei metodi

3. Ciclo di vita

Allo scopo della nostra trattazione individuiamo le seguenti fasi o "età" del progetto:

- Studio di fattibilità
- Analisi funzionale
- Analisi di dettaglio
e generale
- Programmazione
- Verifica
- Accettazione e sue sottofasi
- Manutenzione

3. Ciclo di vita

- Ogni fase produce documenti e raccomandazioni per la successiva, sulla base di:
 - ciò che ha fornito la fase precedente
 - la discussione, analisi ed approfondimento

- Ogni fase ha una durata pianificata ed una reale, che possono influire sulla successiva serie di attività

- Ogni fase può comportare talvolta la ri-esecuzione di una o più fasi

3. Ciclo di vita

- Un progetto di intervento sull' esistente è comunque un progetto di sviluppo ex-novo
- Un errore svolto nelle prime fasi se corretto nelle ultime può avere un costo anche 200 volte superiore al costo iniziale
- Gli utenti dovrebbero sempre essere coinvolti nelle prime fasi
- Le fasi sono a loro volta scomponibili in alcune sottofasi

Fattibilità

- analisi dell' esistente e sua critica
- definizione obiettivi
- decisione: scelta della soluzione

Analisi funzionale

- definizione standard
- analisi informazioni da fornire ("output")
- definizione logica archivi
- analisi informazioni da ricevere ("input")
- definizione transazioni
- definizione sicurezza
- definizione controlli
- schema funzionale con elenco dei singoli moduli funzionali
- sintesi dei moduli (programmi o loro gruppi)
- rettifica dello studio
- indicazioni per la fase successiva

Analisi

- comporta la definizione fisica degli archivi e degli standard tecnici,
- in fase dettagliata, produce le informazioni necessarie ai programmatori
- di norma non dovrebbe modificare l' an. funz.

Programmazione

- realizzazione dei programmi
- verifica dei singoli ("unit testing")

Verifica

- controllo aderenza output a quanto richiesto
- è scomposta nei livelli:
 - per gruppi omogenei ("string", "stream")
 - globale ("system test")

3. *Ciclo di vita*

Accettazione

- verifica di usabilità da parte degli utenti, previa produzione della documentazione (manuali) e addestramento
- preparazione ambiente di produzione
- fase di parallelo con le procedure esistenti
- avvio
- esercizio

Delle fasi sopra indicate,
- fattibilità
- analisi funzionale
sono le più importanti, pur avendo un costo diretto di meno del 10% nei grossi progetti

Manutenzione

può essere di tre tipi:

- correttiva, teoricamente evitabile se è svolto correttamente lo sviluppo
- perfettiva, tende a migliorare o ritoccare leggermente
- adattiva, adegua la procedura o l'applicazione a mutate esigenze e costituisce un vero e proprio progetto

La scomposizione dei costi di un'applicazione è spesso:

20%	sviluppo
80%	manutenzione

prima che venga sostituita.

Costi

In alcune analisi è stata individuata una scomposizione dei costi di sviluppo che sembra abbastanza sorprendente:

- 20% documentazione
- 10% codifica
- 9% analisi
- 46% verifica
- 15% modifiche

come pure lo sono i dati sul costo di verifica nel progetto Apollo:

80%

Nei piccoli sistemi è stata rilevata una media del 30% (ottimistica)

4. Ciclo e classificazione

Analizzeremo ora le modifiche al ciclo di vita base dovute ai diversi tipi di progetti

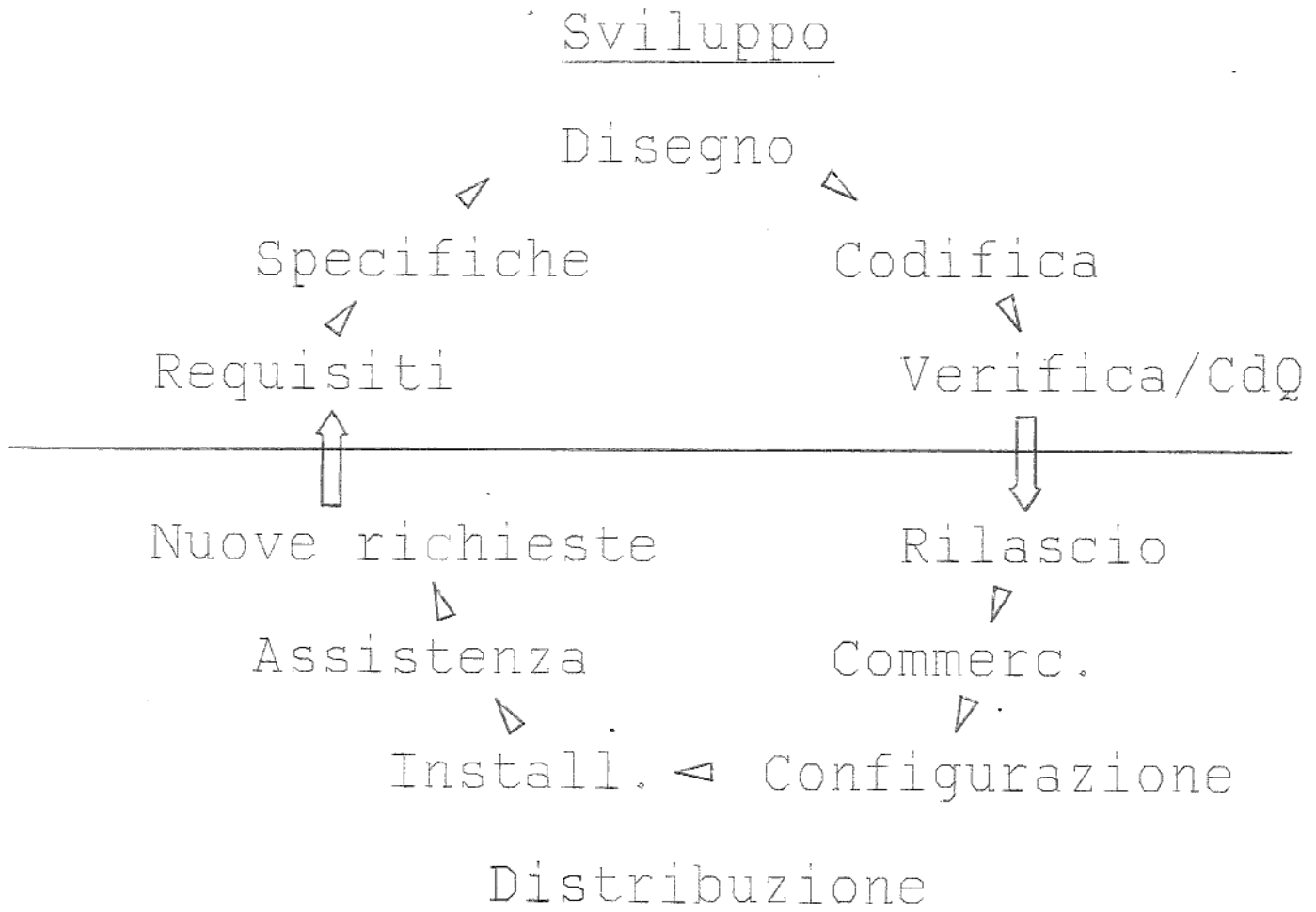
Ogni progetto-tipo ha sue proprie caratteristiche relative a:

- peso della fase
- suddivisione dei costi tra le fasi
- modalità di sviluppo
- metodo di selezione contatti con gli utenti
- rischi assunti dal fornitore
- criticità dei tempi

Pacchetti

- richiedono una preliminare analisi del mercato scelto
- sono veri e propri prodotti
- il fornitore deve assumere in proprio costi notevoli: un pacchetto per PC non complesso ha un costo di sviluppo intorno ai 120 mil.
- esiste il rischio di precoce obsolescenza
- un ritardo nei tempi può costare quote di mercato (vd. Lotus, IBM, ..)
- rendono necessarie strutture interne di verifica e commercializzazione, non legate allo sviluppo

Pacchetti



Il successo di un pacchetto dipende in misura rilevante dalla sua presenza nei canali di vendita

Custom

solitamente i progetti custom sono:

- progetti ex-novo
- modifica di pacchetti
- adattamento applicazioni del cliente

- La gestione di progetti di questo tipo comporta quasi sempre l'assunzione dei costi hardware, software e logistici di sviluppo da parte del cliente
- La pianificazione è di solito poco affidabile, soprattutto per le continue modifiche
- Spesso i tempi reali sono nettamente superiori al previsto

Custom

Ciclo pratico

- mescolamento delle fasi e tutte con riciclo
- modifiche funzionali durante la codifica o successive fasi
- parallelo non sempre eseguito
- fasi di verifica e accettazione spesso solo formali



- i tempi quasi mai sono rispettati
- i costi decollano
- il cliente è cronicamente insoddisfatto

Chiavi in mano

- Moda recente tra i clienti
- Rovesciamento dei costi di sviluppo non previsti sul fornitore
- Necessità di capacità imprenditoriali
- Rischio di non valutare o sottovalutare i costi logistici
- Spesso comportano riduzione degli utili
- Necessità di controllare l'uso delle risorse del cliente (progetti.mf)



- Spesso i primi chiavi in mano di aziende custom sono in perdita
- Richiedono notevoli capacità di analisi costi e pianificazione temp risorse

Chiavi in mano

spesso ultimamente i clienti separano le fasi tra più società:



Chiavi in mano

Esiste il rischio che il cliente forzi la mano chiedendo modifiche e varianti non previste



ricordare che un chiavi in mano ("turn key") dovrebbe essere consegnato al cliente senza che questi intervenga dopo le fasi iniziali

Manutenzione

Analisi e documentazione

Riorganizzazione

Definizione obiettivi

Fattibilità

Analisi funzionale

Sviluppo

Verifica

Accettazione

- Spesso viene sottovalutata la complessità del compito
- Le modifiche spesso devono essere trasparenti
- Più complesso modificare il lavoro altrui che comprenderlo e inserirvisi: rischio di critica non costruttiva e superficiale

*Gestione procedure
e centro*

- Molte aziende medio-piccole potrebbero cedere la gestione del proprio centro a terzi per ridurre i costi
- Iniziano ora ad entrare sul mercato anche società non legate al cliente
- Comprese sono anche la manutenzione correttiva delle procedure e talvolta il ruolo dell'utente/dell'EDP verso altre società
- Spesso è necessaria una preventiva riorganizzazione
- Motivazioni: turn-over, riduzione costi, efficienza
- Simile per complessità più alla gestione temporanea di un'azienda/divisione
- Indispensabile l'aiuto di persone in grado di analizzare l'azienda

*Gestione procedure
e centro*

Studio azienda



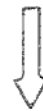
Fattibilità



Pianificazione



Parallelo



Avvio



Gestione

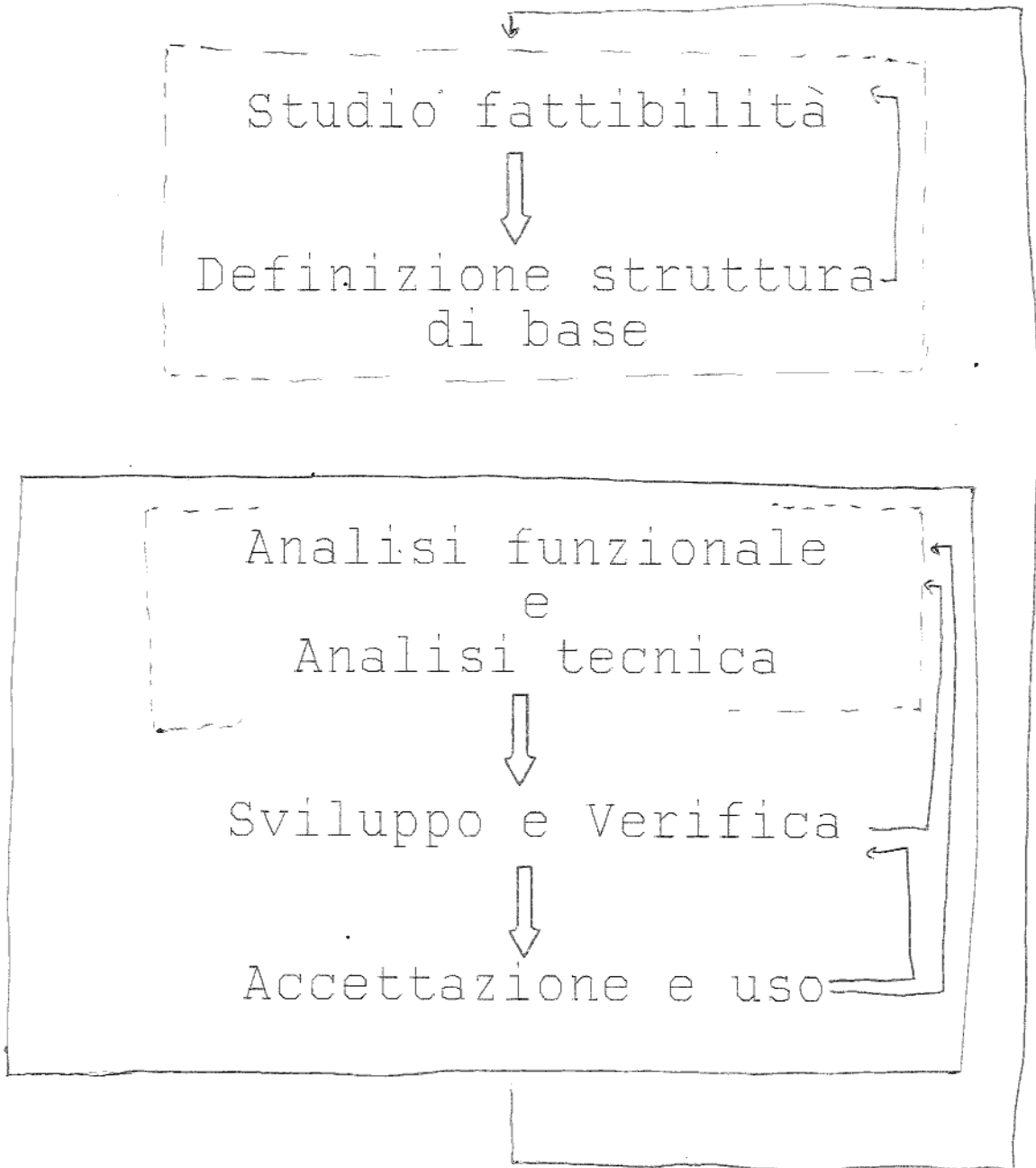
Progetti-pilota

- Servono a convincere le persone del cliente della utilità dell'uso di una tecnologia/prodotto
- Spesso i costi rimborsati dal cliente non coprono tutte le spese
- Talvolta i progetti-pilota sono usati come metodo di soluzione
- Applicazione ad alta velocità di evoluzione del contesto (direzionali, analisi, ecc)
- La pianificazione deve tendere a dare risultati visibili in tempi quasi reali ("on-the-fly")

Progetti-pilota

- I progetti-pilota sono caratterizzati da:
 - tempi ridotti di sviluppo
 - definizione di una "base" sulla quale sono aggiunte soluzioni a problemi
 - continua richiesta di modifiche e adattamenti da parte degli utenti
- il progetto-pilota può diventare base per progetti tradizionali

Progetti-pilota



Generalità

ogni progetto ha diversi
tipi di costi

- logistici
 - locali
 - elettricità
 - altri materiali e risorse
- personale
- risorse macchina

Solitamente nella
pianificazione vengono
considerati i costi del
personale (i giorni-uomo)
ignorando gli altri, cosa
"adatta" solo per
progetti custom

Generalità

- la definizione e pianificazione dovrebbe anche i costi non sopportati, per:
 - evitare problemi nei rapporti col cliente
 - essere in grado di mutare progetto se richiesto
 - poter giustificare il progetto dal lato economico
- ultimamente molti clienti chiedono assicurazioni contrattuali sui costi di sviluppo

Generalità

- solitamente i costi dei locali, anche quando addebitati, sono valutati molto bassi
- con la nascita dei primi centri servizi venne inserito nei sistemi operativi il concetto di costo macchina
- i costi macchina di norma analizzati sono:
 - spazi disco
 - tempo di cpu
 - linee stampate

Generalità

- nei costi macchina sono compresi anche:
 - numero e durata delle sessioni
 - impiego dei canali I/O
 - eventuali costi di TLC in caso di uso remoto
- esistono anche costi accessori, quali luce, telefoni e simili

Costo-macchina

Nel valutare le risorse necessarie e' utile anche considerare per ogni tipo di attivitè l' uso medio risorse e la frequenza

Vedremo quindi attivita' con elevato tempo di elaborazione in memoria (CPU-BOUND), che sono le piu' costose, ed altre con numero e peso di I/O su disco e simili elevati (I/O-BOUND)

E' da ricordare che un minuto di CPU puo' costare anche diverse centinaia di migliaia di lire

CPU-BOUND

tra queste attività sono comprese le compilazioni, spesso svolte senza effettiva necessità.

Ricordarsi che:

- una compilazione di un programma di grosse dimensioni può costare anche diversi minuti (alcuni milioni)
- sono da svolgersi dopo aver verificato a tavolino il sorgente
- "intasano" la macchina, poiché richiedono anche la presenza di diversi archivi su disco

I/O-BOUND

le attività legate a ingresso/uscita di informazioni, quali le stampe o la prova di programmi di immissione dati comportano:

- costi bassi come uso di CPU
- costi accessori elevati
- tra i costi accessori vanno considerati:
 - i tempi necessari per ottenere i tabulati
 - eventuali pause per carichi macchina
 - ecc.

Misti

esistono anche attività
e programmi in grado di
usare:

- notevoli quantità di
CPU
- notevoli spazi su disco

i programmi misti sono da
scomporre, ove possibile,
in due componenti:

- elaborazione
- aggiornamenti

Controllo risorse

per ridurre il costo delle risorse i metodi piu' usati sono:

- l' autocontrollo
- la limitazione delle risorse utilizzabili dal singolo

ognuno dei metodi ha controindicazioni, ma il problema maggiore in un controllo risorse macchina e simili e' la scarsa affidabilitè dei singoli in caso di non rispetto dei tempi

Controllo risorse

Autocontrollo

- vantaggi
 - basso costo di gestione del metodo

- svantaggi
 - inaffidabile in fasi critiche

 - inapplicabile in grossi gruppi di lavoro

- in generale, e' applicabile
 - nelle fasi iniziali del coding e test

 - per gruppi di lavoro costanti o quasi

Controllo risorse

Limitazione risorse

- vantaggi
 - permette di valutare e controllare meglio:
 - i costi
 - le modalita' operative
 - le capacita' analitiche del personale

- svantaggi
 - burocratizzazione del gruppo
 - costosa gestione del metodo
 - richiede episodici interventi del gestore di variazione dei limiti

5. Elementi di analisi

Lista di controllo

cosa serve per un
progetto EDP?

- locali
- risorse hardware
- risorse software
- persone
- attrezzature varie

5. Elementi di analisi

Lista di controllo

Locali

- sono consigliabili:
 - ben aerati e luminosi
 - possibilmente non "open space"
 - con stampanti, telefoni e altre fonti di rumore fuori dallo spazio di programmazione/analisi
 - con sufficiente spazio per ogni persona
 - non affollati

5. Elementi di analisi

Lista di controllo

Risorse hardware

- e' necessario pianificare:
 - lo spazio disco di sviluppo
 - lo spazio disco per le varie fasi di test
 - il numero di terminali
 - il numero di prese
 - il numero di telefoni e stampanti
 - la quantita' di CPU (se da pianificare)

5. Elementi di analisi

Lista di controllo

Risorse software

- assicurarsi che tutti i prodotti software:
 - compilatore, linker
 - sistema OLTP, DBMS
 - editor di sistema
 - sistema di esecuzione in prova e correzione (DEBUGGER, tools)
 - eventuali sistemi di
 - gestione versioni dei programmi
 - controllo progetti

siano funzionanti

5. Elementi di analisi

Lista di controllo

Risorse software

- talvolta puo' essere necessario predisporre nastri con programmi-base da installare sulla macchina del cliente

- e' consigliabile un ambiente scomposto in tre parti:
 - sviluppo

 - test utente

 - produzione

salvo esistano problemi di spazio

5. Elementi di analisi

Lista di controllo

Risorse software

- iniziare a costruire archivi di base per i test dopo l' an. gen.

- far svolgere i test su copie o estratti degli archivi di base

Persone

- possono essere presenti:
 - a tempo pieno

 - a tempo parziale

 - a richiesta

- pianificare subito le risorse non a tempo pieno

Lista di controllo

- Attrezzature di supporto
- ogni progetto necessita di persone, strutture, programmi e procedure di supporto
 - occorre pianificare attentamente anche le fasi di salvataggio e controllo
 - i progetti dovrebbero sempre prevedere regolari salvataggi su nastro o disco dell' ambiente di sviluppo, con gestione congiunta EDP-progetto

5. Elementi di analisi

Lista di controllo

Attrezzature di supporto

- definire prima anche:
 - moduli da usare
 - ruoli e assegnazioni
 - gestione comunicazioni:
 - interne
 - esterne
 - gestione dello stato di avanzamento, ritardature
 - segnalazione eventuali criticita' sul progetto

6. Conclusioni

Oggi e domani

- i progetti tendono a diventare sempre piu' simili a realizzazioni di impianti e strutture
- i prodotti saranno sempre meno artigianali
- i clienti cercano di trasferire sui fornitori costi di sviluppo oggi fuori controllo
- progressivamente negli anni passati sono state inserite nei contratti le penali da inadempienza contrattuale

6. Conclusioni

Oggi e domani

- le penali piu' comuni previste sono per:
 - ritardo
 - eventuali danni
 - incompletezza
- e' necessario che le societa' di consulenza interessate a progetti software si trasformino in imprenditori
- i capi progetto devono imparare a svolgere compiti attualmente nella pratica marginali

6. Conclusioni

Oggi e domani

- tra i compiti previsti, e talvolta già oggi svolti:
 - pianificare
 - controllare i costi
 - gestire il personale
- esiste il rischio di ripetere errori di gestione già svolti in altri campi
- il software almeno in parte rimarrà un frutto di lavoro intellettuale e non da "catena"