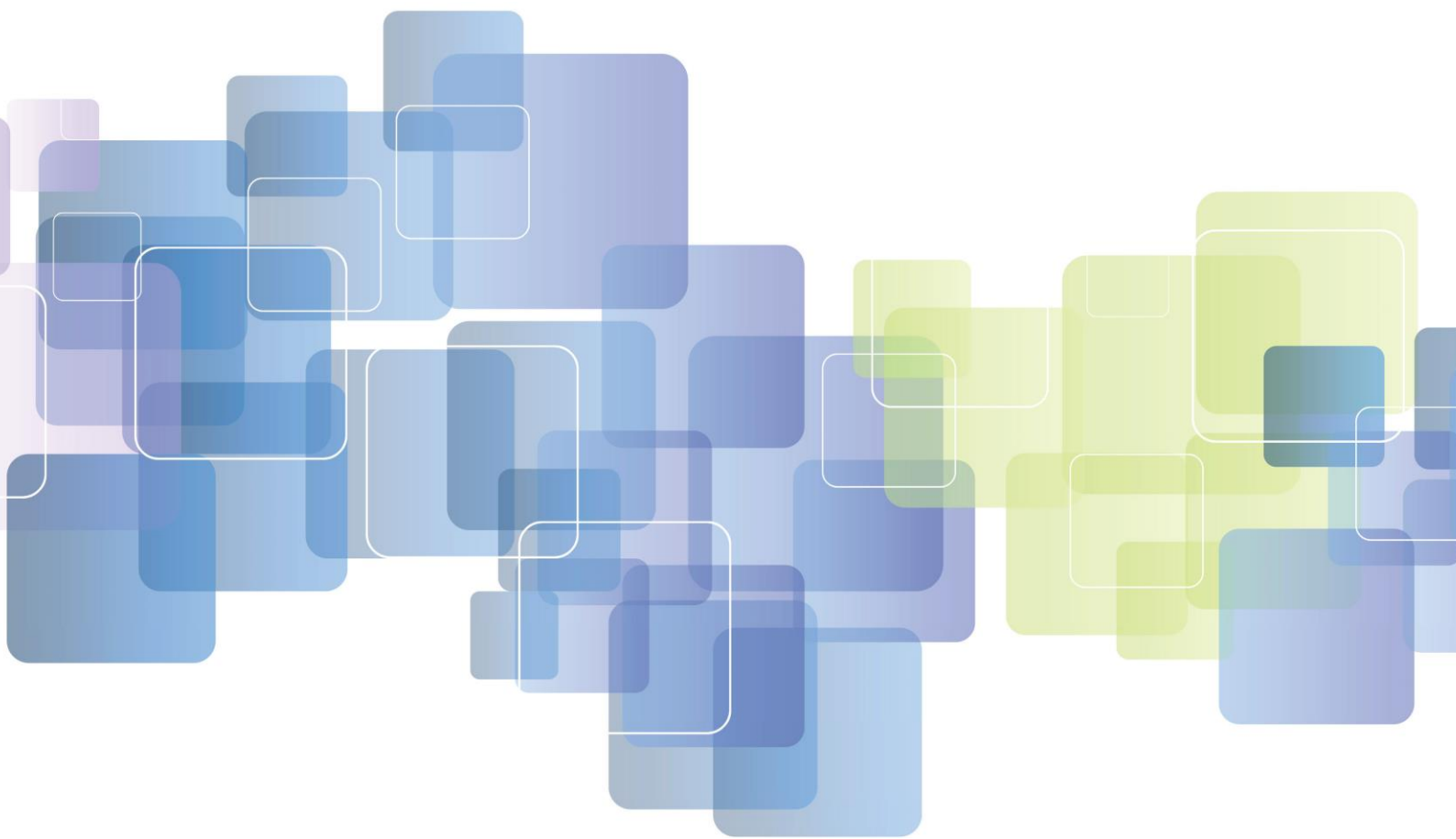


Il Riutilizzo del software

Giorgio Marras



Come sviluppare una metodologia per produrre componenti applicativi riutilizzabili da gestire nel modo più efficace.

Il "software reuse" è probabilmente una delle poche tecniche capaci di influire positivamente tanto sulla qualità del software quanto sui relativi costi e tempi di produzione. Le ragioni sono intuibili ma, per completezza, può essere opportuno riepilogarle brevemente.

Innanzitutto le più ovvie: il software viene riutilizzato solo quando si è sicuri delle sue buone qualità e della sua efficienza. Ciò mette al riparo da errori che possono essere commessi durante l'attività di produzione e, nel contempo, influisce positivamente sia sui costi di sviluppo, sia sulla diffusione di nuovi prodotti in tempi brevi (time to market).

Le ragioni meno manifeste riguardano invece un insieme di attività supplementari, ad esempio la preparazione delle componenti o il supporto alle "ordinarie" tecniche di progettazione, indispensabili per consentire un effettivo riutilizzo del software e inserire tale pratica come parte integrante del processo di sviluppo del software.

Design for reuse e with reuse

Paolo Tiribelli, Amministratore di Sodalia, spiega che " il riutilizzo del software, [com'è stato autorevolmente affermato], non è un fatto spontaneo, non nasce da sé, ma va programmato, preparato e supportato da un'ideale strategia, dal convinto sostegno degli organi decisionali dell'impresa e da adeguati investimenti.

A proposito di strategia, la figura 1 mostra una modalità di ingegnerizzazione dell'applicazione informatica (il Prodotto) strutturata in livelli logico-funzionali, a loro volta articolati mediante moduli caratterizzati da un differente potenziale di riutilizzazione.

Tale strategia suggerisce di progettare il prodotto "A" con una tecnica (design for reuse) che massimizzi il riutilizzo dei suoi componenti in vista della realizzazione del prodotto "B" e, di contro, di progettare "B" in base ai componenti riutilizzabili già disponibili (design with reuse).

A parte ciò, sussiste un'altra modalità di riutilizzazione del software che si impone nell'iterative development: quando cioè i componenti di un'applicazione vengono riutilizzati per realizzare una versione successiva dell'applicazione stessa (riutilizzazione verticale).

La figura 2 aiuta a spiegare graficamente questo concetto.

Essa, intende chiarire l'idea della cosiddetta riutilizzazione orizzontale la quale presuppone e richiede interventi di reingegnerizzazione (generalizzazione) dei componenti per migliorarne la portabilità, l'affidabilità, la qualità, etc.

Un altro importante presupposto di un efficace programma di software-reuse è la "libreria" centralizzata dei componenti riutilizzabili la cui principale funzione è quella di "stivare" e rendere accessibili i componenti stessi.

Altre funzioni, non meno importanti, della libreria sono quelle che presiedono alla "classificazione" dei prodotti prima dell'inserimento nel repository ed al loro successivo reperimento (retrieval), a richiesta dei potenziali riutilizzatori. Dalla loro efficienza, come mostra la tabella 1, dipende gran parte dei risultati del programma di riutilizzazione.

Di conseguenza, se molti sono i vantaggi attesi dal riutilizzo del software, molte sono anche le precondizioni che bisogna garantire alla "fabbrica" perché i benefici si realizzino. Ne diamo conto nel paragrafo successivo".

Un programma aziendale per il riutilizzo del software

"Sodalia", prosegue Tiribelli, "ha istituito un Programma aziendale per il riutilizzo del software, concepito per fare del riuso una pratica sistematica del processo iterativo di sviluppo. La strategia adottata si articola secondo tre direttrici:

- il processo di sviluppo per il riutilizzo del software (design for reuse)
- il processo di sviluppo con il riutilizzo del software (design with reuse)
- la realizzazione di una "libreria" per la gestione dei prodotti (e dei sottoprodotti) riutilizzabili.

Più in particolare il programma comprende:

1. la definizione di un "modello di processo" per "l'analisi di dominio" volta alla individuazione delle opportunità di riuso del software. A tale scopo è stata sviluppata una specifica metodologia denominata "SODA"(Sodalia's Object-Oriented Domain Analysis) che consente di eseguire la ricognizione di esigenze ricorrenti così da configurare la possibilità di risposte applicative simili (se non identiche), configurabili con limitati (o senza) interventi di adattamento (customization).
2. l'istituzione della funzione RSO (Reuse Support Organization), una struttura organizzativa di supporto che si avvale di risorse totalmente dedicate allo sviluppo di componenti for reuse. La ragion d'essere di tale struttura riposa sul fatto che la costruzione di un componente riutilizzabile richiede uno "sforzo" (effort) di almeno cinque volte quello speso per la realizzazione di un componente non riusabile. Di conseguenza la maggior mole di lavoro non può essere scaricata sulle ordinarie strutture di progetto. Queste, infatti, sono finalizzate alla realizzazione di applicazioni sotto condizioni vincolanti sia in termini di tempo, sia in fatto di costi. Non è così sorprendente, quindi, il fatto che i capi progetto sono molto restii ad accollarsi oneri supplementari che

metterebbero a repentaglio il successo del progetto di cui sono responsabili. L'RSO fornisce inoltre il supporto metodologico ai gruppi di progetto per l'attività di design with reuse, e promuove il riutilizzo degli artifact esistenti.

3. Lo sviluppo e la gestione del Sodalia's Asset Library Management System (SALMS), un sistema concepito per dare visibilità, a livello aziendale, di tutti gli asset riutilizzabili (requisiti, object model, test case, codice, etc.) contenuti nella libreria RSAL (Reusable Software Assets Library). Tale libreria è provvista di un meccanismo di "classificazione" che aggiunge ad ogni componente un set di informazioni supplementari (livello di riusabilità, report su precedenti esperienze di riuso, note per l'adattamento e l'integrazione, etc.), al fine di meglio orientare il potenziale riutilizzatore e di migliorare l'efficacia della ricerca. RSAL è stata collaudata oltre che in Sodalia, dove viene impiegata come un "ordinario" strumento di produzione, anche in un più ampio contesto internazionale: il progetto TARSAL del Programma ESPRIT, finanziato dalla Comunità Europea, ha consentito una sperimentazione "a largo raggio" da cui Sodalia ha reperito suggerimenti ed idee che hanno consentito di migliorare le prestazioni della "libreria", oggi più funzionale ed estesa ai maggiori ambienti operativi".

Conclusione

"Cinque anni fa", conclude Tiribelli, "Sodalia decise di trarre profitto dalle tecnologie più promettenti, anche se all'epoca poco accreditate, per la produzione, a costi contenuti, di software di qualità capace di soddisfare i bisogni sempre più complessi ed urgenti degli operatori dei servizi di telecomunicazione. Il processo di produzione fu basato sin dall'inizio su un approccio iterativo/sperimentale object-oriented accoppiato con tecniche di riutilizzazione del software. Tale processo è stato applicato in modo sistematico grazie alla metodologia SIMEP (vedi prima parte dell'articolo), ad un forte convincimento da parte del Management che ha portato all'istituzione di apposite strutture organizzative ed alla realizzazione di specifici tool di supporto. Oggi il riutilizzo del software costituisce in Sodalia una pratica corrente: i gruppi di progetto, infatti, la praticano sia a livello di componenti di piccole dimensioni (fine-grain artifacts), sia a livello di componenti di medie dimensioni (medium-grain artifacts) ed i risultati sono evidenti anche in termini di pay-back".

Architettura Strategica di Prodotto

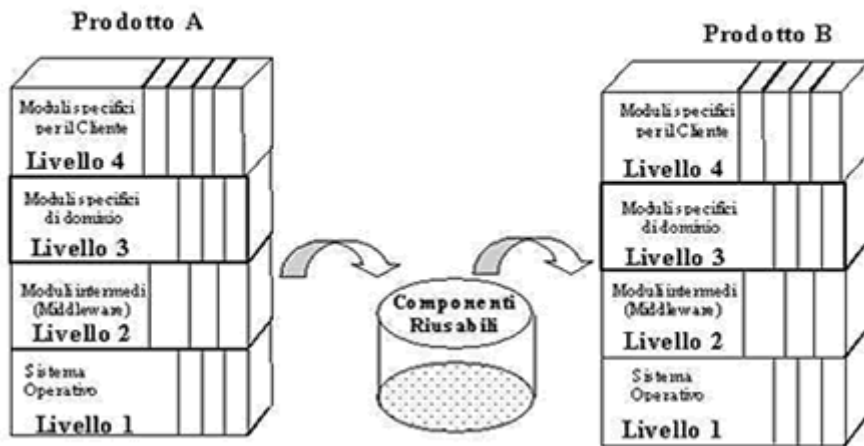


Figura 1

Riutilizzazione verticale ed orizzontale

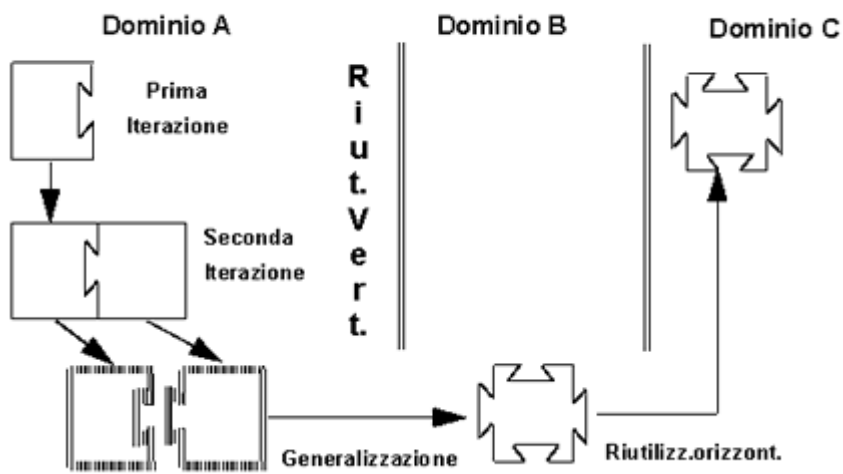


Figura 2

Riusabilità del software

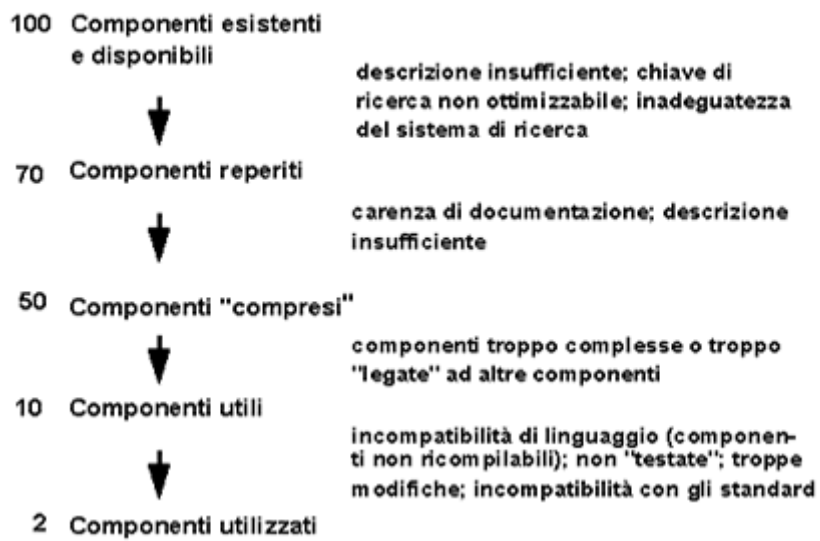


Tabella 1



Tecnet Dati s.r.l.
C.so Svizzera 185 -
10149 - Torino (TO), Italia
Tel.: +39 011 7718090 Fax.: +39 011 7718092
P.I. 05793500017 C.F. 09205650154
www.tecnetdati.com

