

Corso di Laurea in Ingegneria Informatica

Corso di Ingegneria del Software

**I modelli di qualità del software
Le norme ISO/IEC 9126 e 25010**

Modello di qualità del software

- ❖ Un modello di qualità del software definisce:
 - Un insieme finito e definito di attributi di un prodotto software, esaustivi, e privi di reciproche sovrapposizioni;
 - Le relazioni di dipendenza logica esistenti fra i diversi attributi
- ❖ Modello tipicamente gerarchico
- ❖ Nella parte più alta della gerarchia gli attributi di qualità del committente, nella parte più bassa quelli del produttore

Il modello di qualità associato ad ISO 9000

- ISO 9000 è una famiglia di standard che ha lo scopo di definire i requisiti di un Quality Management System (QMS) aziendale
- La famiglia ISO 9000 è di riferimento per molti settori (non è riferita solo al software)
- Fornisce un valido riferimento:
 - ai clienti per valutare ed essere garantiti circa le capacità di un produttore di fornire prodotti e servizi di qualità
 - ai produttori per valutare ed essere garantiti circa le capacità di rispondere alle esigenze di mercato ed alle clausole contrattuali inerenti la qualità dei prodotti e servizi

I modelli ISO per la qualità del software

- ❖ I modelli di qualità del software pubblicati in letteratura sono tipicamente a n livelli.
- Il primo livello descrive un insieme di caratteristiche (proprietà), che, nel loro complesso, rappresentano la qualità del prodotto software, eventualmente secondo diversi punti di vista.
- Le proprietà (in genere qualitative, astratte) sono descritte attraverso sottocaratteristiche quantitative, misurabili (in genere da una combinazione di sottocaratteristiche).
- Il grado di possesso che il software ha di questi attributi può essere valutato su una scala di riferimento, facendo ricorso ad opportune metriche ed a meccanismi di rating.

I modelli ISO per la qualità del software

- La famiglia ISO 9000 comprende alcune norme che definiscono modelli di qualità del software, in vari contesti di utilizzo.
- **ISO/IEC 9126-1** - Definisce 6 caratteristiche di qualità principali (astratte) e 27 sottocaratteristiche misurabili attraverso delle metriche, fornite in 3 technical reports (ISO/IEC 9126-2, 3 e 4).
- **ISO/IEC 9241** - Definisce le caratteristiche di qualità di un software “usabile”.
- **ISO 12119** - Definisce le caratteristiche di qualità di un software “Commercial off-the-shelf” (COTS).
- La ISO ha anche pubblicato la norma 14598, che guida alla valutazione della qualità del software, definita secondo i criteri 9126.

ISO/IEC 9126

- ◆ La norma **ISO/IEC 9126 Software engineering – Product quality** contiene il modello di riferimento per le caratteristiche di qualità del software e le metriche per la valutazione della qualità che il software possiede.

PART 1: il modello delle caratteristiche e sottocaratteristiche di qualità del software

PART 2: le metriche per la misura della qualità esterna

PART 3: le metriche per la misura della qualità interna

PART 4: le metriche per la misura della qualità in uso (ISO/IEC TR 9126-4, 2004).

- ◆ Analizziamo solo la PART 1

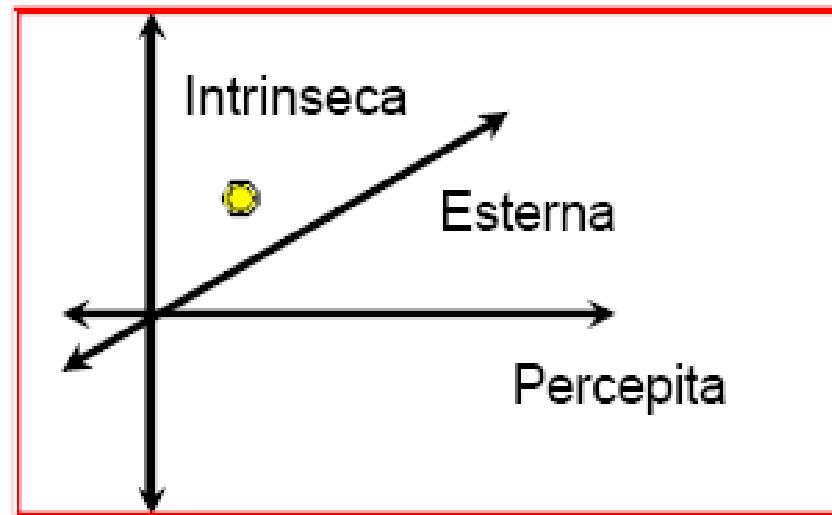
ISO/IEC 9126-1

♦ La ISO 9126 definisce un modello a 4 livelli:

- tre “punti di vista” sulla qualità del software (esterno, interno ed in uso) che qualsiasi progetto di sviluppo deve prendere in considerazione e soddisfare;
- i principali attributi (definiti come requisiti qualitativi) che qualificano un software, secondo i 3 punti di vista;
- per ogni attributo, le sottocaratteristiche (requisiti quantitativi, misurabili) che la rappresentano e che dovranno essere misurate per valutare il livello di qualità effettivamente raggiunto nel software;
- le metriche per effettuare le misure.

I punti di vista sulla qualità

A determinare la qualità complessiva di un software concorrono 3 punti di vista.



PERCEPITA (in uso)

Esprime l'efficacia ed efficienza con cui il software serve le esigenze dell'utente, ed è correlata alla percezione diretta dell'utente.

INTERNA (intrinseca)

Esprime la misura in cui il codice software possiede una serie di attributi statici, indipendentemente dall'ambiente di utilizzo e dall'utente.

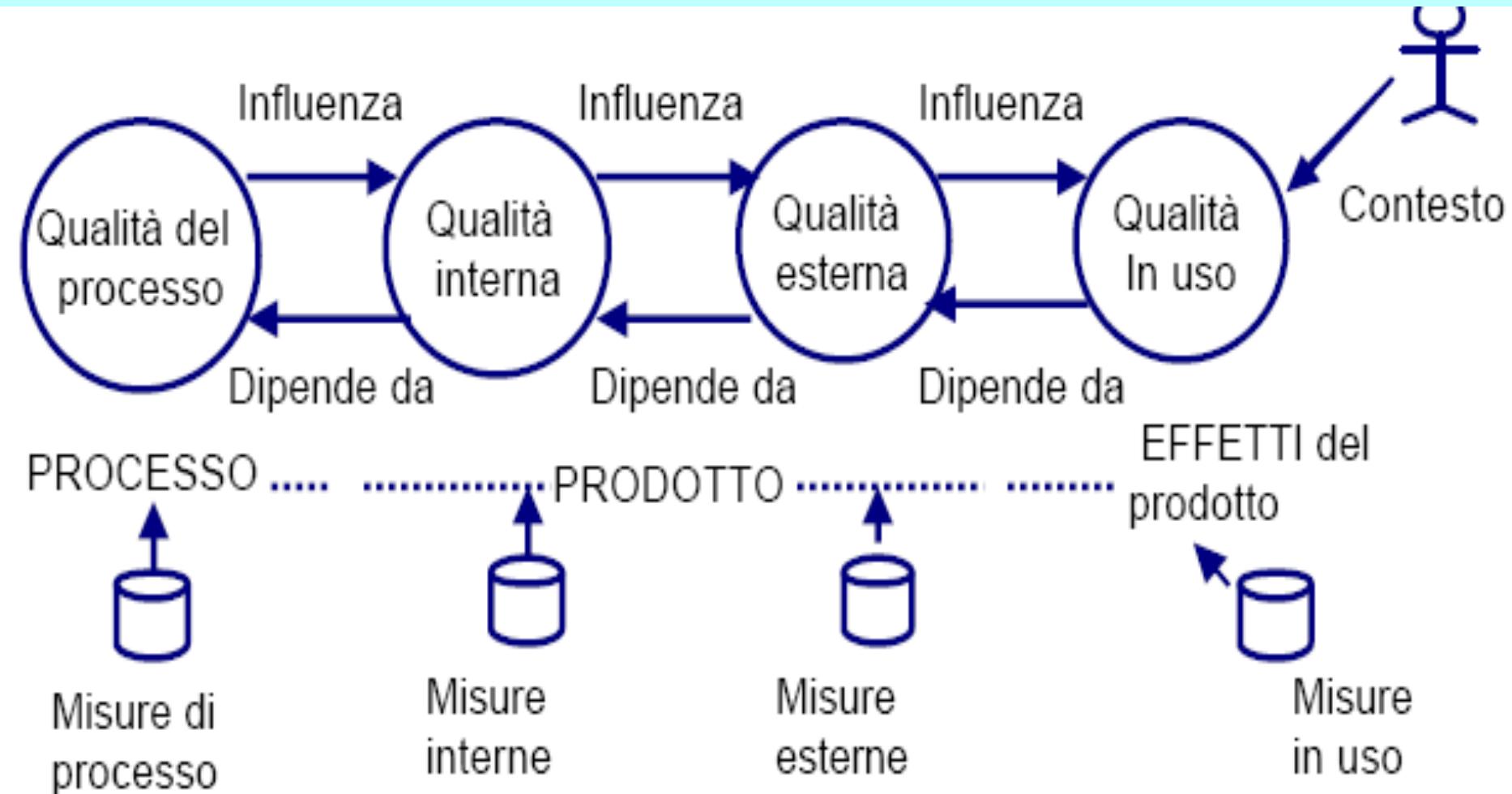
ESTERNA

Esprime il comportamento dinamico del software, nell'ambiente d'uso.

I punti di vista sulla qualità

- ◆ La **qualità in uso** riguarda il livello con cui il prodotto si dimostra utile all'utente nel suo effettivo contesto d'utilizzo, in particolare la capacità del prodotto di dare efficacia ed efficienza al lavoro dell'utente, a fronte di una sicurezza di utilizzo e di una soddisfazione nel far uso del prodotto.
- ◆ La **qualità esterna** è quella rappresentata dalle prestazioni del prodotto e dalle funzionalità che offre (il prodotto è visto come una black box da testare).
- ◆ La **qualità interna** rappresenta le proprietà intrinseche del prodotto (quelle misurabili direttamente sul codice sorgente, sul suo flusso di controllo).

Qualità del software nel ciclo di vita

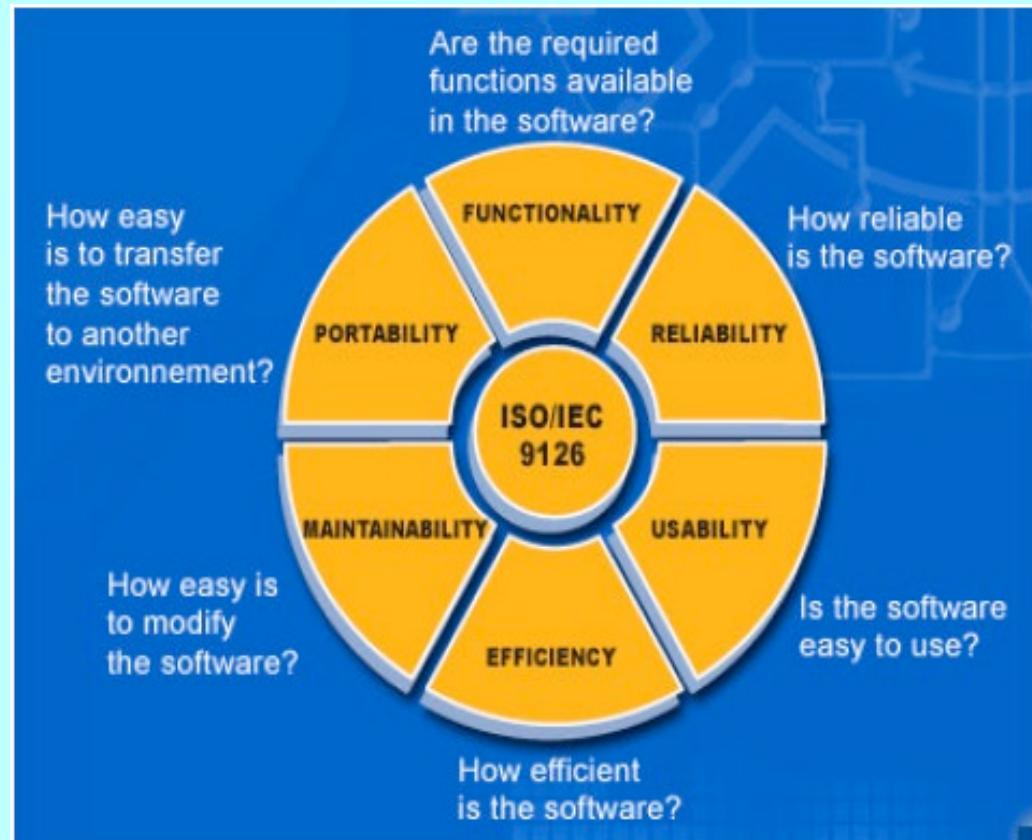


Il modello di qualità

- ♦ Il modello recepisce la distinzione fra attributi che caratterizzano la qualità vista dall'utente ed attributi della qualità vista in produzione
 - **CARATTERISTICHE ESTERNE**
 - **CARATTERISTICHE INTERNE**
 - **CARATTERISTICHE IN USO**
- ♦ Le caratteristiche esterne sono organizzate in due livelli gerarchici (caratteristiche, sottocaratteristiche e relazioni di dipendenza tra esse)
- ♦ Le sottocaratteristiche sono poi relazionate alle caratteristiche interne

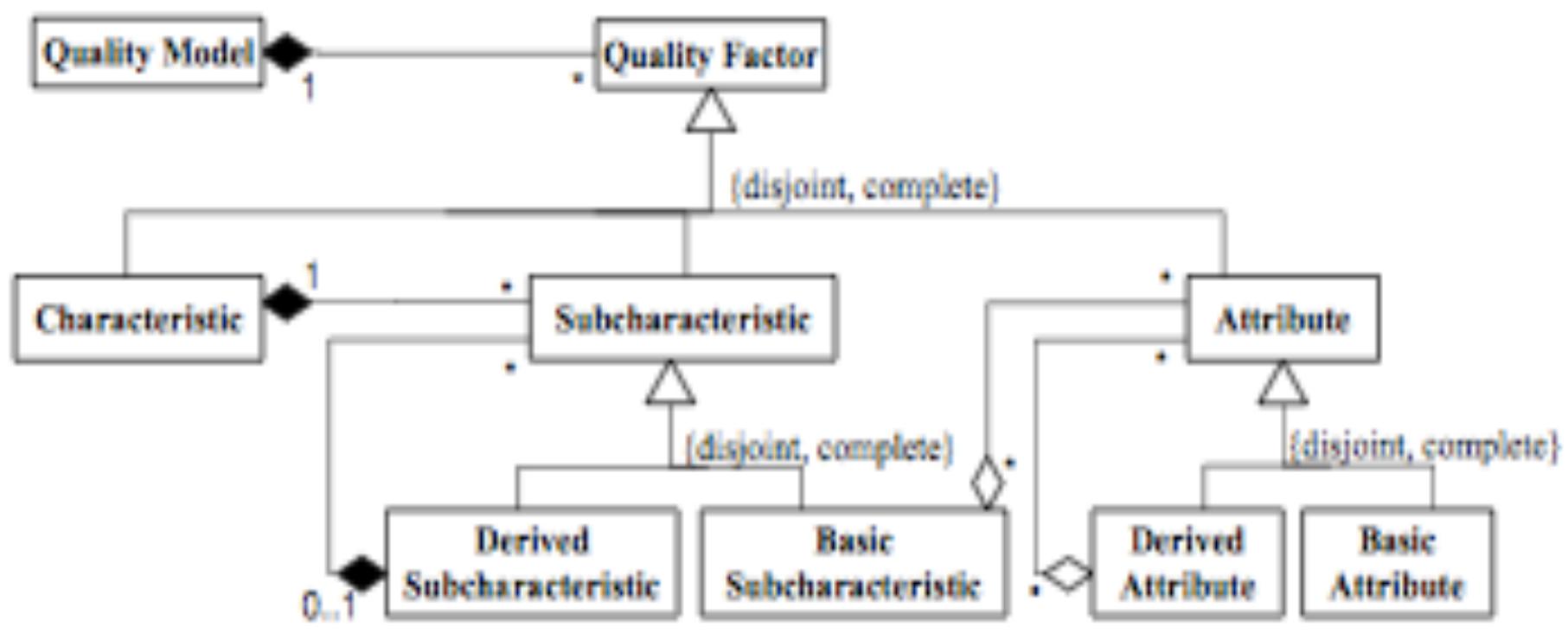
Le caratteristiche esterne di primo livello

- ❖ 6 caratteristiche esterne di primo livello
 - Funzionalità
 - Affidabilità
 - Efficienza
 - Usabilità
 - Manutenibilità
 - Portabilità



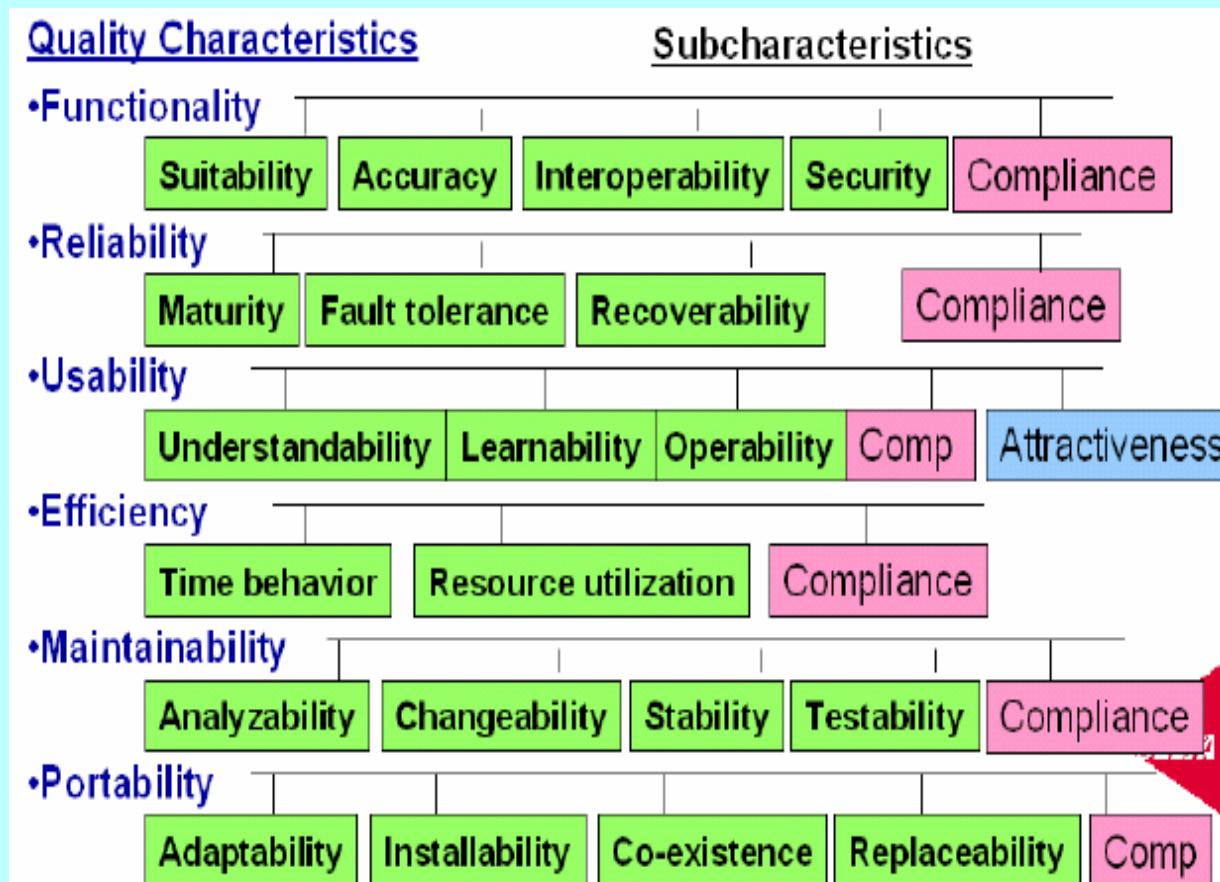
Modello concettuale

- ◆ Modello concettuale UML per lo standard ISO/IEC 9126



Le sottocaratteristiche

- Le 6 caratteristiche principali sono ulteriormente specificate da 27 sotto-caratteristiche.



Funzionalità

- ◆ **Funzionalità**: capacità di fornire servizi tali da soddisfare, in determinate condizioni, requisiti funzionali esplicativi o impliciti (il software fa ciò per cui è stato acquistato).
- ◆ Le sottocaratteristiche correlate sono:
 - Adeguatezza: presenza di funzioni appropriate per compiti specifici che supportano obiettivi dell'utente.
 - Accuratezza: capacità di fornire risultati corretti in accordo con i requisiti dati dall'utente.
 - Interoperabilità: capacità di interagire con altri sistemi.
 - Sicurezza: capacità di proteggere programmi e dati da accessi non autorizzati e di consentire quelli autorizzati.

Affidabilità

- ◆ **Affidabilità**: capacità di mantenere le prestazioni stabilite nelle condizioni e nei tempi fissati (il software reagisce bene a variazioni esterne);
- ◆ Le sottocaratteristiche correlate sono:
 - Maturità (robustezza): capacità di evitare malfunzionamenti della applicazione (*failure*) a seguito di condizioni di guasto (error) dovute ad errori (*fault*).
 - Tolleranza: capacità di mantenere determinati livelli di prestazione in caso di malfunzionamenti (non degrado).
 - Recuperabilità: capacità e velocità, in caso di malfunzionamenti, di ripristinare dei livelli di prestazione predeterminati e di recuperare i dati.

Usabilità

- ◆ **Usabilità**: capacità di essere compreso, appreso, usato con soddisfazione dall'utente in determinate condizioni d'uso (il software gestisce bene l'interazione con gli utenti);
- ◆ Le sottocaratteristiche correlate sono:
 - Comprensibilità: capacità di ridurre l'impegno richiesto agli utenti per cadirne il funzionamento e le modalità di utilizzo.
 - Apprendibilità: capacità di ridurre l'impegno richiesto agli utenti per impararlo ad usare.
 - Operabilità: capacità di mettere in condizione gli utenti di farne uso per i propri scopi e controllarne l'uso.
 - Attrattività/Piacevolezza: capacità di essere piacevole per l'utente che ne fa uso.

Efficienza

- ◆ **Efficienza:** rapporto fra prestazioni e quantità di risorse utilizzate, in condizioni definite di funzionamento (il software usa bene le risorse disponibili);
- ◆ Le sottocaratteristiche correlate sono:
 - Comportamento rispetto al tempo: adeguati tempi di risposta, tempi di elaborazione e throughput rates per eseguire le funzioni richieste, sotto determinate condizioni.
 - Uso di risorse: utilizzo di una quantità e di una tipologia di risorse adeguate per eseguire le funzioni richieste, sotto determinate condizioni.

Manutenibilità

- ◆ **Manutenibilità**: capacità di essere modificato con un impegno contenuto (per evoluzioni, correzioni, etc.)
- ◆ Le sottocaratteristiche correlate sono:
 - Analizzabilità: capacità di limitare l'impegno richiesto per diagnosticare carenze o cause di malfunzionamenti, o per identificare parti da modificare.
 - Modificabilità: capacità di limitare l'impegno richiesto per modificare, rimuovere errori o sostituire componenti.
 - Stabilità: capacità di ridurre il rischio di comportamenti inaspettati a seguito della effettuazione di modifiche.
 - Testabilità: capacità di essere facilmente testato per validare le modifiche apportate.

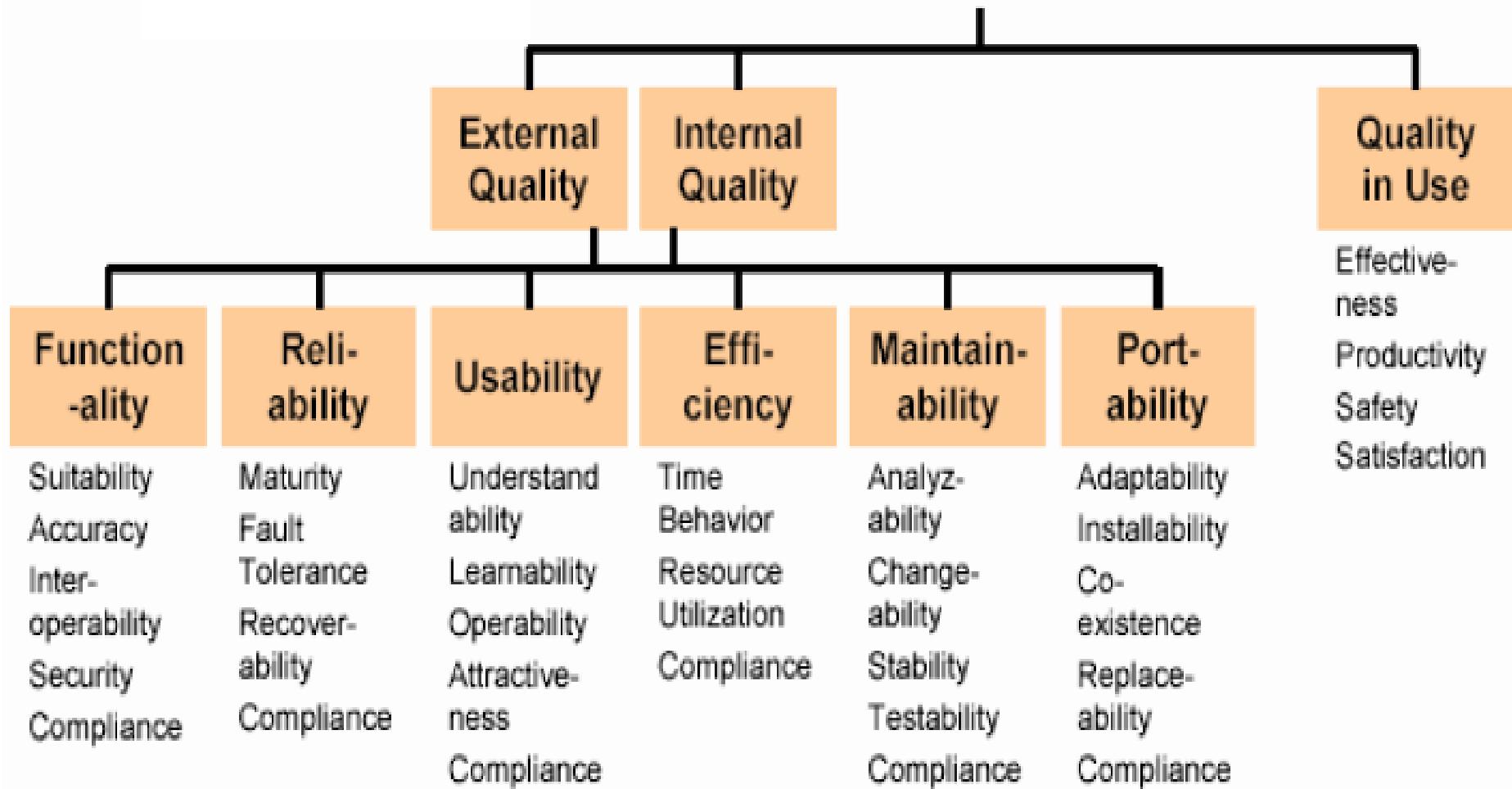
Portabilità

- ◆ **Portabilità:** facilità con cui il software può essere trasferito da un ambiente operativo ad un altro
- ◆ Le caratteristiche correlate sono:
 - Adattabilità: capacità di adattarsi a nuovi ambienti operativi limitando la necessità di apportare modifiche.
 - Installabilità: capacità di ridurre l'impegno richiesto per installarlo in un particolare ambiente operativo.
 - Coesistenza: capacità di coesistere con altri software nel medesimo ambiente, condividendo risorse.
 - Sostituibilità: capacità di essere utilizzato al posto di un altro software per gli stessi compiti nello stesso ambiente.

Caratteristiche della qualità in uso

- **Efficienza**: capacità di supportare un utente nel raggiungere i suoi obiettivi con accuratezza e completezza in un dato contesto.
- **Produttività**: capacità di supportare un utente nello spendere l'appropriata quantità di risorse in relazione all'efficienza dei risultati da raggiungere.
- **Soddisfazione**: capacità di soddisfare un utente in un dato contesto d'uso.
- **Sicurezza**: capacità di raggiungere accettabili livelli di rischio per le persone, l'ambiente di utilizzo, le attività dell'utilizzatore, in un dato contesto d'uso.

Riepilogo modello 9126



Caratteristiche e punti di vista

Rilevanza delle caratteristiche per i vari punti di vista

Vista Caratteristica	Utente	Sviluppatore	Gestore	Committente
Funzionalità	Alta	Bassa	Bassa	Media
Affidabilità	Media	Media	Alta	Media
Usabilità	Alta	Bassa	Media	Media
Efficienza	Media	Media	Alta	Media
Manutenibilità	Bassa	Alta	Media	Media
Portabilità	Bassa	Alta	Alta	Media

ISO 25010

- ◆ E' uno standard ISO del 2011 che rimpiazza ed estende lo standard ISO 9126
- ◆ E' parte di un più ampio insieme di standard
 - (la famiglia 25000, detta anche SQuaRE - System and Software Quality Requirements and Evaluation)

Qualità in uso



Qualità del prodotto



ISO 25010 vs ISO 9126

◆ Rispetto ad ISO 9126:

- I requisiti di quality in use sono completamente cambiati
- Alcuni cambiamenti ci sono stati nei requisiti di qualità del prodotto:
 - ◆ Sono scomparsi i requisiti di compliance
 - ◆ Sono stati «promossi» i requisiti di security e compatibility
 - ◆ Sono comparsi alcuni requisiti specifici di queste ultime due oltre ad alcuni altri requisiti come user interface aesthetics

◆ E' scomparsa la separazione tra qualità interna e qualità esterna