# Correttezza del Software: Sogno o realtà

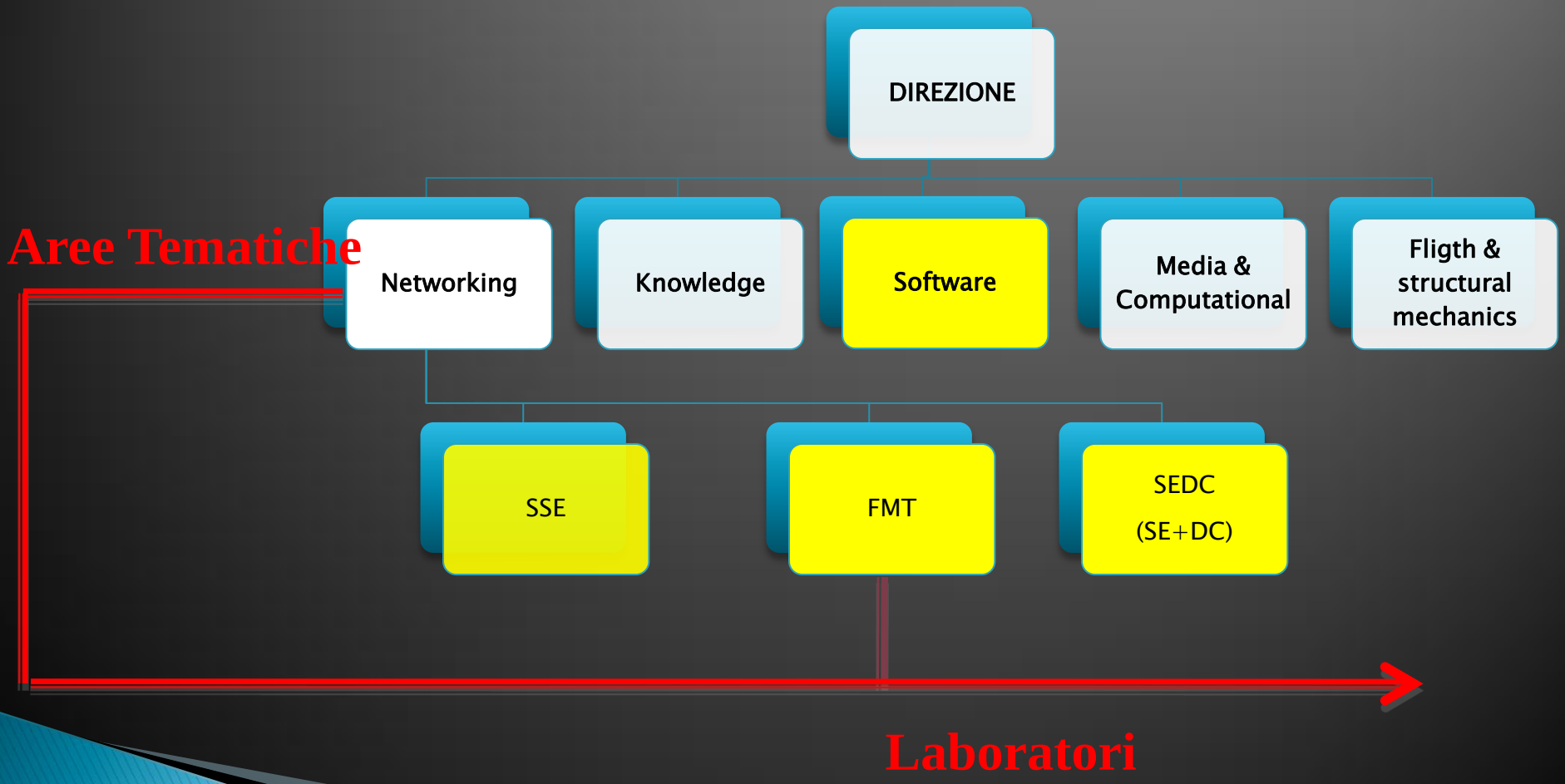
Stefania Gnesi

ISTI-CNR

Seminario “ Logistica e ICT”

Livorno 02/03/2012

# L'organizzazione ISTI



# Logistica, ICT ed errori Software

Lo sviluppo di piattaforme integrate per lo realizzazione di piattaforme intermodali ha come uno degli ingredienti principali l'uso di **sistemi software** che aiutino gli operatori nella gestione di traffici complessi.

# Denver Airport Baggage System Case Study

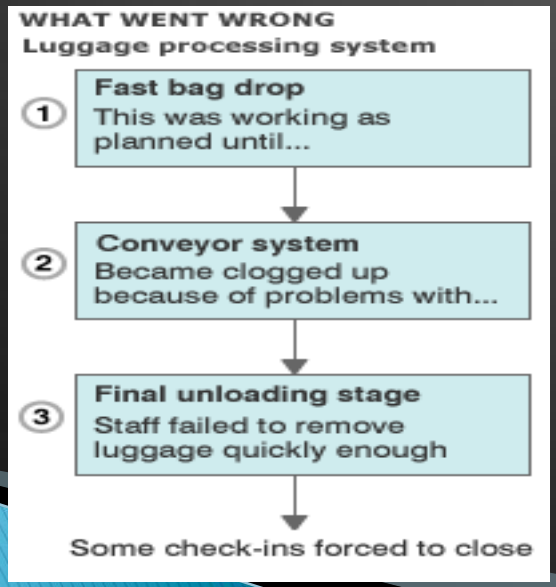
Underestimation of complexity. Complex architecture. Changes in requirements. Underestimation of schedule and budget. Dismissal of advice from experts. Failure to build in backup or recovery process to handle situations in which part of the system failed. The tendency of the system to enjoy eating people's baggage.

**$560M extra cost**

# Heathrow Terminal 5

Opening of Heathrow Terminal 5 labeled a fiasco after 28,000 bags get lost and hundreds of flights are cancelled.

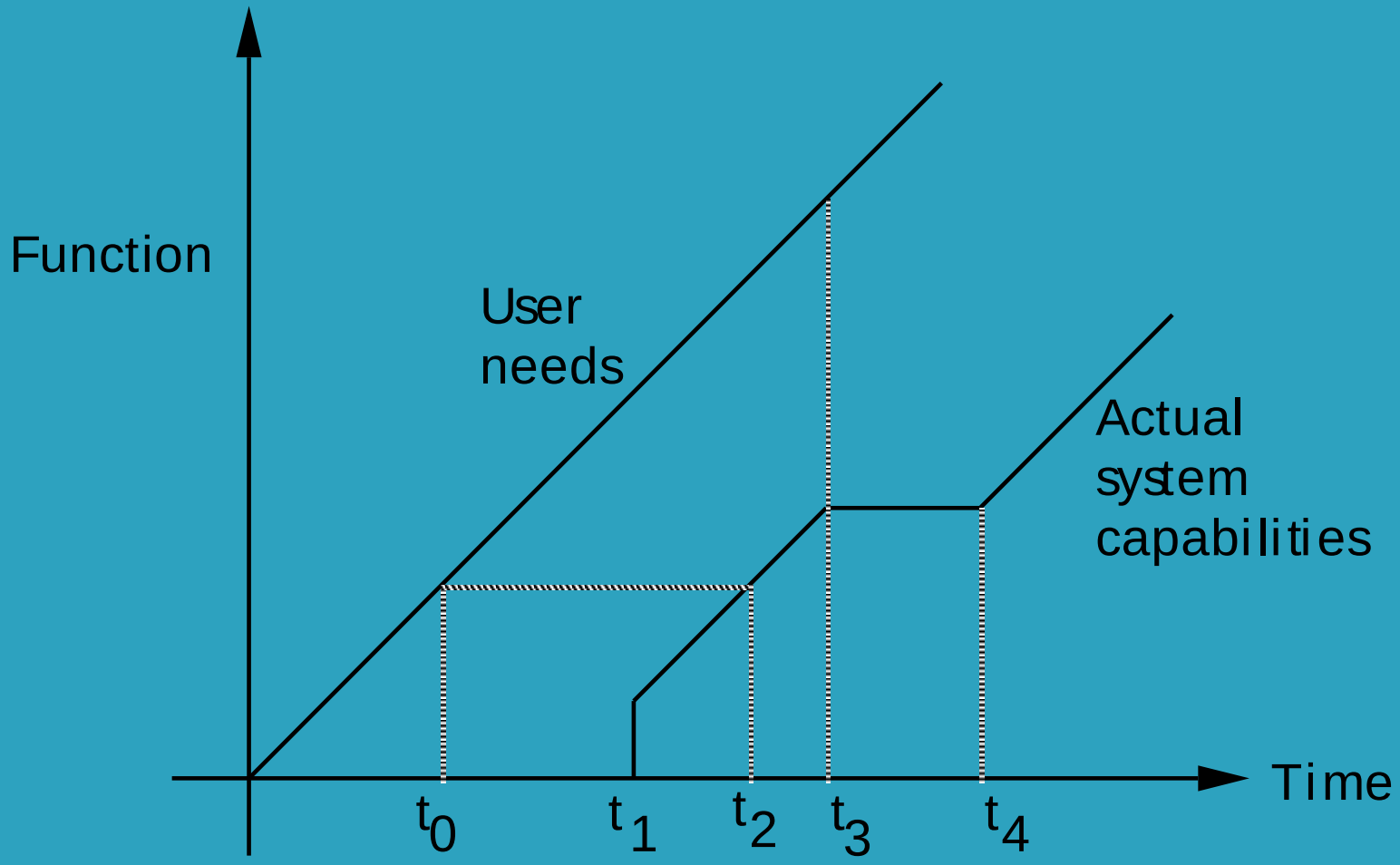
Programming errors in the baggage system.
Failure to test the system with realistic loads.




# Prodotti Software

- Differenti dai tipi tradizionali  di prodotti
  - intangibili
    - difficili da descrivere e valutare
  - Malleabili
  - Altamente interagenti con utenti umani

# Timeliness: Una descrizione visuale

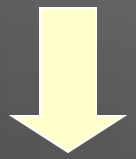


# Malfunzionamento, Anomalia, Errore

- **Malfunzionamento (Failure).** Comportamento errato del sistema. (In medicina: il sintomo)
- **Anomalia (Fault).** Causa del malfunzionamento nel senso di porzione del programma (procedura, componente, modulo) che dà origine al malfunzionamento. (In medicina: l'organo all'origine del sintomo)
- **Errore (Bug).** Il vero e proprio errore che causa l'anomalia. (In medicina: la malattia)

# Motivazione

- Separazione tra programmatore e utente (anni ’60)
- Sw di grandi dimensioni
- Sw sviluppato in team
- Applicazioni critiche



**Ingegneria del Software:**

disciplina che regola lo sviluppo di Sw di buona qualità; gestisce e organizza la realizzazione, manutenzione, ed evoluzione del Sw.

# 1. Qualità del buon prodotto software

# Qualità del buon prodotto Sw

1. **Affidabile:** in grado di procedere senza malfunzionamenti.
2. **Corretto:** rispetta le specifiche ($\Rightarrow$ affidabile).
3. **Robusto:** in grado di gestire gli eventi eccezionali ($\Rightarrow$ corretto).
4. **Safe:** non procura pericolo all'essere umano ($\Rightarrow$ robusto).
5. **Secure:** protegge l'accesso alle informazioni.

segue . . .

# Qualità del buon prodotto Sw (segue)

6. **Manutenibile:** facile da modificare (facilmente sottoponibile a manutenzione correttiva, perfettiva, evolutiva o adattativa).
7. **Riparabile:** facilmente sottoponibile a manutenzione correttiva.
8. **Evolvilbile:** facilmente sottoponibile a manutenzione correttiva o perfettiva.
9. **Portabile:** facilmente sottoponibile a manutenzione adattativa.

segue . . .

# Qualità del buon prodotto Sw (segue)

10. **Comprensibile:** fa capire all'utente cosa sta succedendo.
11. **Usabile:** con interfaccia utente facile da usare.
12. **Efficiente:** fa un uso limitato delle risorse.
13. **Interoperabile:** facilmente integrabile con altri prodotti Sw.

# 2. Processo di sviluppo

# Fasi principali del processo di sviluppo

- **Specifica**
  - Analisi dei requisiti
  - Specifica dei requisiti
- **Sviluppo**
  - Progettazione
  - Implementazione
- **Verifica**
  - Verifica (vera e propria)
  - Validazione
- **Evoluzione**
  - Manutenzione
  - Rilascio

# Metodologie del processo di sviluppo

**Il processo di sviluppo deve essere codificato:**

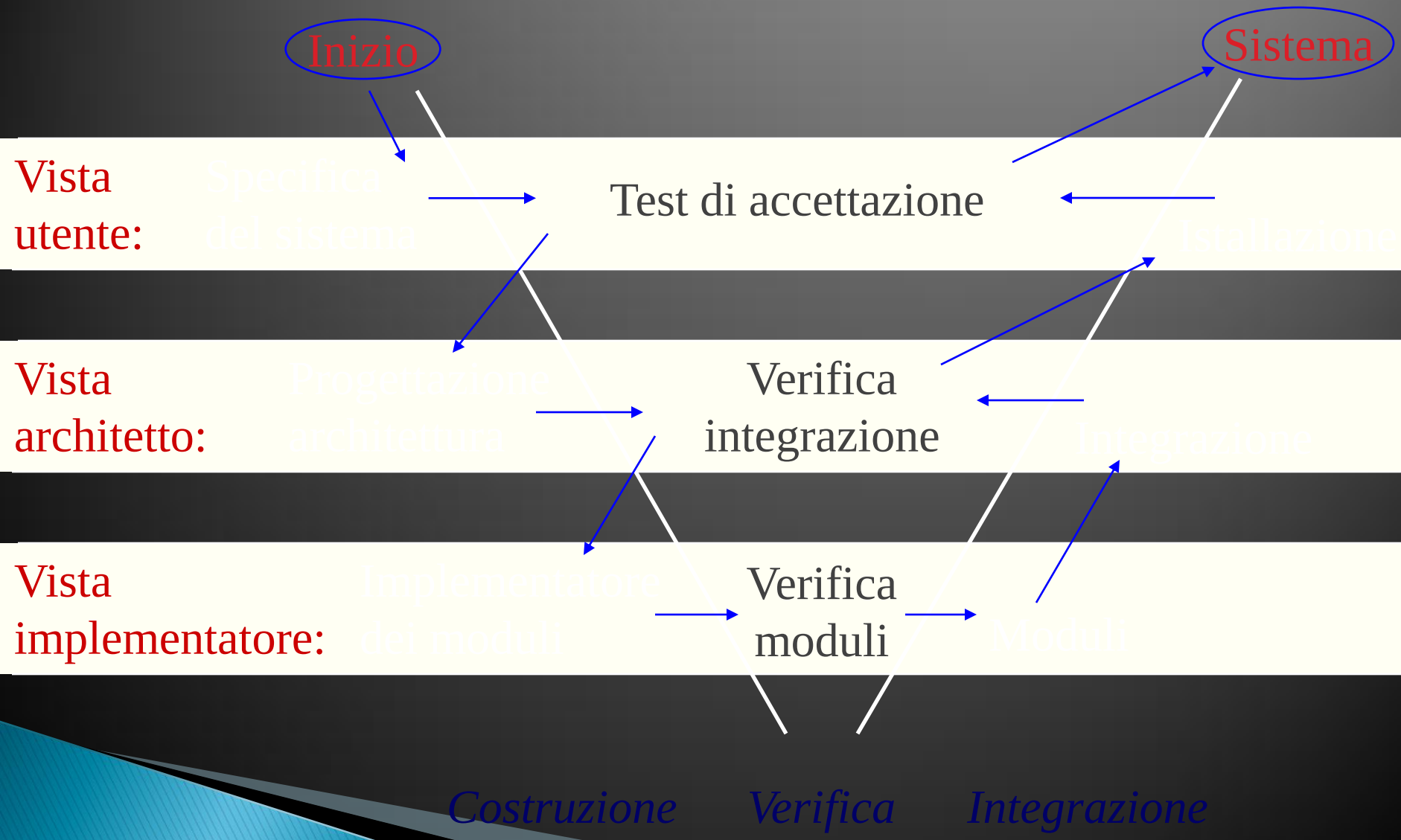
**Un buon processo aumenta la probabilità di avere un buon prodotto**

# Qualità di un buon processo

- **Comprensibile:** si riesce a capire e ad usare?
- **Accettabile:** è accettato dagli ingegneri del Sw?
- **Affidabile:** evita gli errori di processo o e in grado di correggerli?
- **Supportato/Supportabile:** è supportato/supportabile con strumenti CASE? (computer aided sw engeneering)
- **Robusto:** affronta i problemi inattesi?
- **Manutenibile:** può essere esteso o migliorato se necessario?
- **Rapido:** porta al prodotto finito velocemente?

# Modello a "V"

Tiene in conto le varie viste del sistema



# Progettazione

| Cosa<br>DSR | → | Come<br>DSP |
|---|---|---|

**Principi base:**

- **Modularità.** Il progetto deve scomporre il sistema in **sottosistemi** di dimensioni e **complessità ridotte**. Questo perché la somma delle complessità e minore della complessità della somma ed anche per permettere lo **sviluppo in parallelo** dei sottosistemi.
- **Astrazione.** Il progetto deve essere simultaneamente abbastanza **astratto** da essere vicino alle specifiche ed abbastanza **concreto** e dettagliato da essere vicino all'implementazione.

# Verifica

- **Verifica.** La verifica è l'attività svolta da chi ha realizzato il prodotto per vedere se ci sono errori rispetto alle specifiche.

- **Validazione.** La validazione è l'attività svolta dal committente per vedere se il programma soddisfa le richieste per quanto riguarda le funzionalità e l'usabilità.

**Note.**
- La **correttezza** di un programma rispetto a una specifica è generalmente un problema **indecidibile**.
- La specifica può essere una fonte di informazione per la verifica.

# Conclusioni

Approcci di programmazione che includano:

- Utilizzo di tecniche semiforma o formali di specifica;
- modellazione e analisi delle infrastrutture critiche;
- verifica,validazione , monitoraggio e cerificazione.

Sono gli ingredienti principali per lo sviluppo di applicazioni software affidabili.

# DISTRETTO PER LE TECNOLOGIE FERROVIARIE, L'ALTA VELOCITÁ E LA SICUREZZA DELLE RETI

PIANO STRATEGICO DI SVILUPPO 2012-2015

1) migliorare l'intero sistema dei trasporti e delle infrastrutture;
2) creare un moderno sistema di mobilità che deve risultare sostenibile sia dal punto di vista economico e sociale che da quello ambientale

Tra gli obiettivi del PSS che richiedono soluzioni in ICT abbiamo:

Sistemi di tracciamento e localizzazione del materiale rotabile e relativo carico
Realizzazione di processi di logistica multi-modale

# Application-specific qualities

- E.g., information systems
  - Data integrity
  - Security
  - Data availability
  - Transaction performance.