life.augmented

# Sicurezza funzionale in Sistemi elettronici: Principi e Applicazioni

Alessandro Bastoni

Esperto in sicurezza funzionale

STMicroelectronics

# Lezione n. 3

## Teoria della capacità sistematica, incluso il modello V e i preliminari su
### Software e strumenti

**Riepilogo:**

- Ciclo di vita della sicurezza
- **Modello V**
- **Note sui requisiti**
- **Sviluppo software**
- Valutazione degli strumenti

life.augmented

# Come affrontare i fallimenti sistematici

Mentre la maggior parte degli standard di sicurezza richiede di mitigare/considerare i guasti sistematici, la questione correlata è più complessa per gli standard basati sul rischio che stabiliscono un livello specifico di integrità della sicurezza anche per i guasti sistematici.

Esistono due modelli principali:

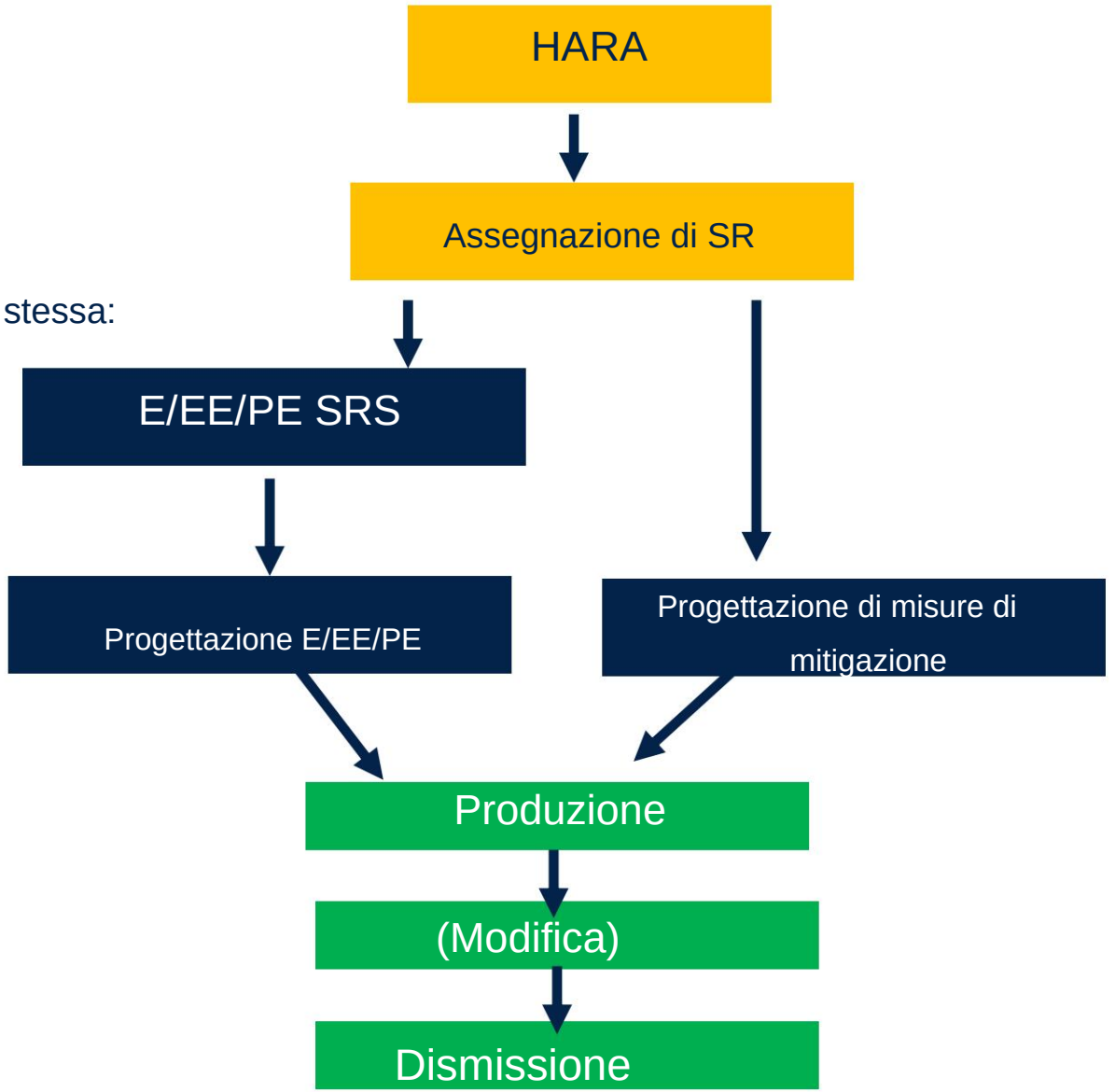
- **Ciclo di vita della sicurezza** formale (basato sulla rigorosa applicazione di regole di sviluppo personalizzate in base al livello di integrità della sicurezza)

- **Argomentazione comprovata in uso,** basata su prove di stabilità/assenza di difetti sistematici nel corso tempo

- I due modelli possono essere applicati anche al software incorporato e agli strumenti software.

# Ciclo di vita della sicurezza

Ogni norma di sicurezza definisce il proprio specifico ciclo di vita della sicurezza; la struttura generale è sostanzialmente la stessa:




Verifica

Documentazione

Valutazione della sicurezza

# Verifica e convalida

**La verifica** è il processo di conferma, attraverso l'esame e l'evidenza oggettiva, che un prodotto, un sistema o un componente soddisfa i requisiti specificati. Risponde alla domanda: "Stiamo costruendo il prodotto correttamente?".

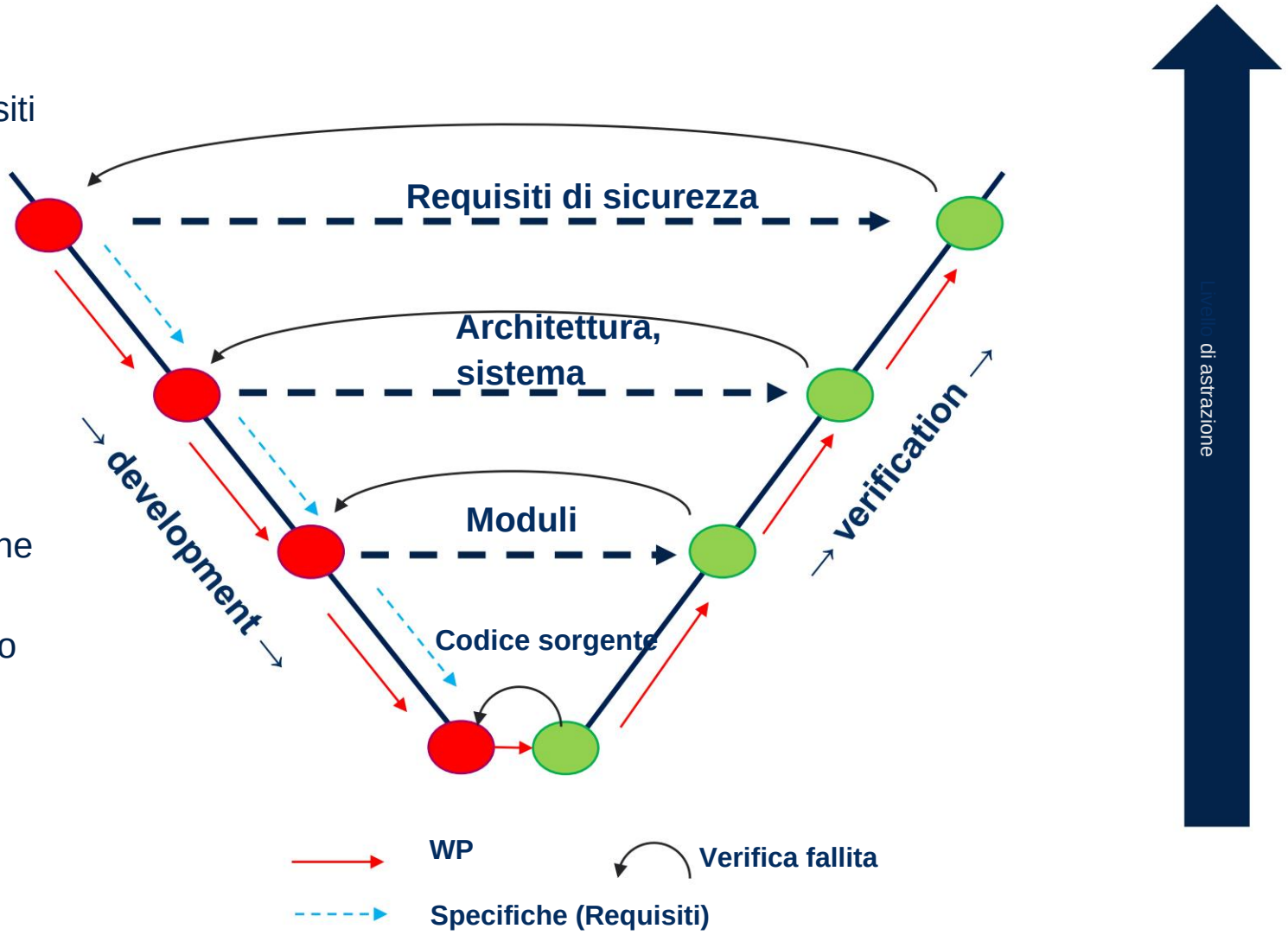
**La convalida** è il processo di conferma, attraverso l'esame e l'evidenza oggettiva, che il prodotto soddisfa i requisiti per l'uso specifico previsto nel mondo reale. Risponde alla domanda: "Stiamo costruendo il prodotto giusto?"

| Aspetto | Verifica | Validazione |
|---|---|---|
| Scopo | Confermare che i requisiti siano implementati correttamente | Confermare che il prodotto soddisfa le esigenze dell'utente e l'uso previsto |
| Messa a fuoco | Conformità alle specifiche | Idoneità allo scopo |
| Attività tipiche | Revisioni, ispezioni, test di unità/componenti | Test a livello di sistema, test di accettazione dell'utente |

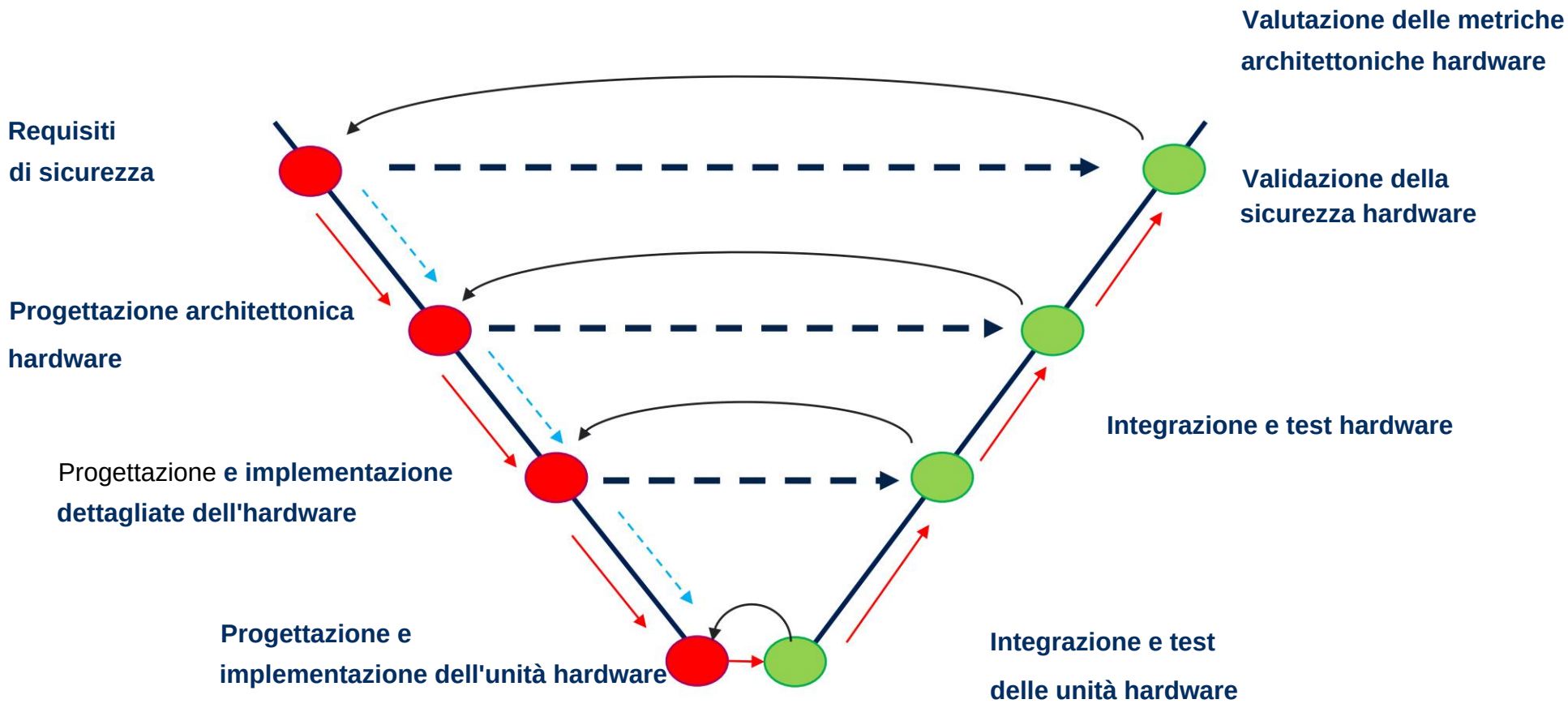
# Caratteristiche del modello V (IEC 61508-3, ISO 26262-6)

**Vantaggi**

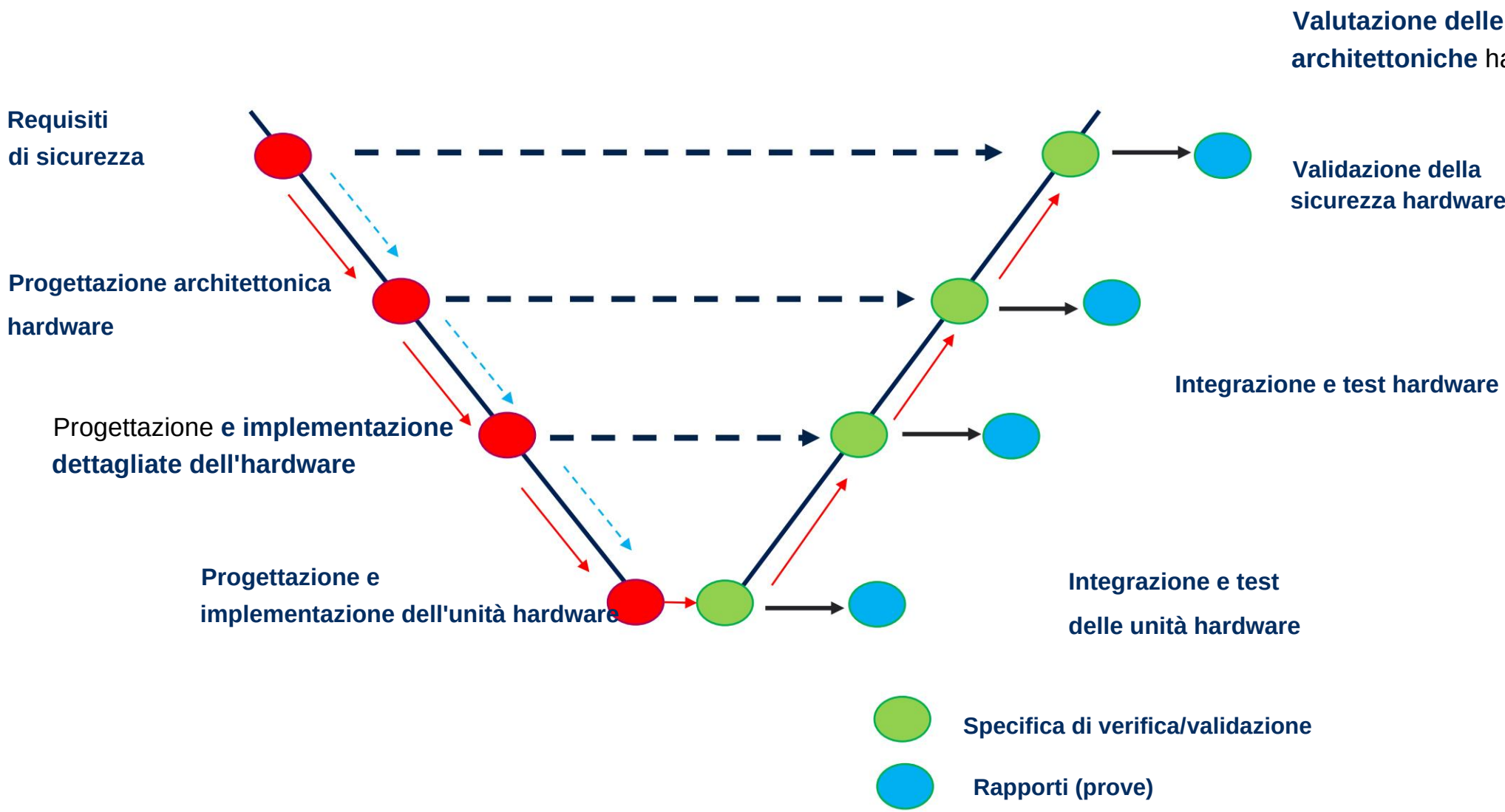
- L'approccio top-down è forzato
- Modello client/server basato sui requisiti tra le fasi
- Ingressi/uscite tra le fasi (WP) ben definito
- Test specificati allo stesso livello di astrazione dello sviluppo
- La non conformità dopo la verifica viene gestita gerarchicamente (potrebbe avere un impatto sulle fasi di sviluppo correlate)
- Tracciabilità garantita all'interno



# Modello V per lo sviluppo hardware (ISO 26262-5) - fasi



# Modello V per lo sviluppo hardware (ISO 26262-5) - documenti



# Informazioni sulla tracciabilità

Il modello V della IEC61508 richiede la tracciabilità in avanti e all'indietro tra diversi set di requisiti di specifica e verifica. HR per SIL3/4, solo R per SIL1/2. Tra le altre tecniche, la tracciabilità è un'ottima risorsa per la sicurezza e la qualità nello sviluppo del software:

Tracciabilità futura: verifica che un requisito venga adeguatamente affrontato nelle fasi successive del ciclo di vita.

→ Pro: risorsa chiave per valutare correttamente l'impatto di un cambiamento in base all'aggiornamento/modifica dei requisiti di alto livello

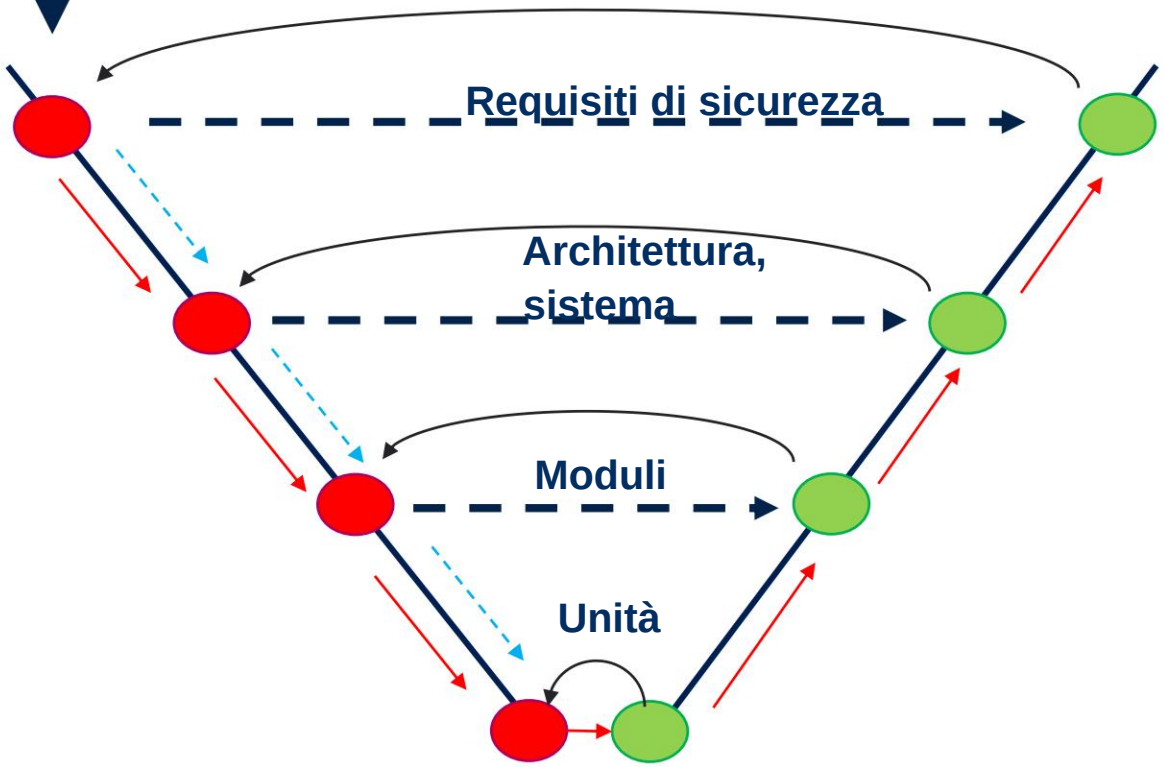
Rintracciabilità a ritroso: verifica che ogni decisione di implementazione sia chiaramente giustificata da qualche requisito.

→ Pro: diminuisce la probabilità di funzioni/parti hw inutilizzate e ingiustificate

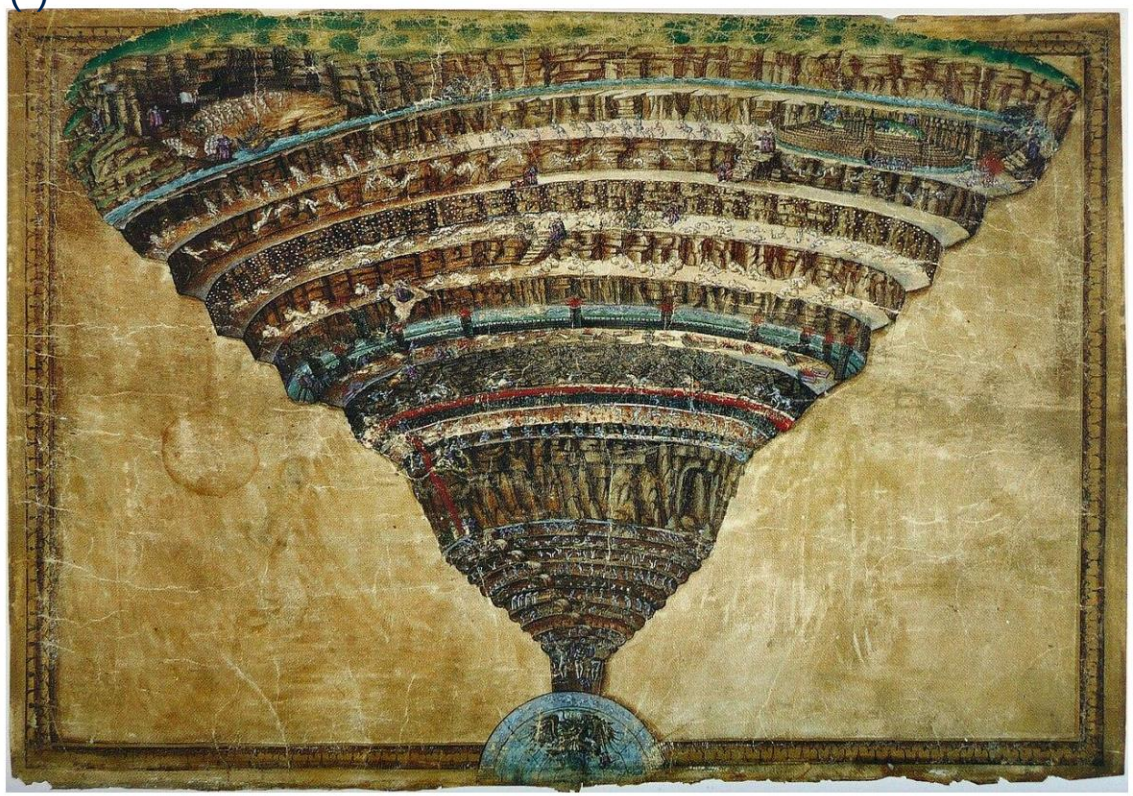
ATTENZIONE: aggiungere la tracciabilità in un progetto in un secondo momento si trasformerà in un incubo!

# La trappola del modello V

Il diavolo del modello V è qui – vedi riferimento [R3]




(*)



Il diavolo di Dante è qui

*(*) La voragine infernale, Sandro Botticelli (Fonte: Wikipedia)*

# SOTIF (Sicurezza della funzionalità prevista)

ISO/PAS 21448: fornisce linee guida per garantire la sicurezza nei sistemi avanzati di assistenza alla guida (ADAS) e nei veicoli autonomi. SOTIF (Safety Of The Intended Functionality)

SOTIF affronta i rischi per la sicurezza derivanti dalle funzionalità previste di un sistema, soprattutto quando non sono presenti guasti o anomalie.

SOTIF si concentra sui pericoli causati da limitazioni prestazionali, condizioni ambientali o uso improprio, andando oltre la tradizionale sicurezza basata sui guasti. Sposta l'attenzione sulle specifiche di sistema incomplete, che per loro natura non vengono intercettate dal modello V.

Lo scopo della norma SOTIF è identificare e mitigare i rischi correlati al corretto comportamento del sistema che possono comunque portare a situazioni non sicure. Integra la norma ISO 26262, coprendo scenari in cui il sistema si comporta come progettato ma presenta comunque rischi per la sicurezza.

.

# Adattamento dei metodi nel modello V

Il modello formale V prescrive per ogni fase elenchi di metodi consigliati.

Le raccomandazioni sono inserite in tabelle e classificate in questo modo:

HR/++ la tecnica o la misura è altamente raccomandata per il livello di integrità della sicurezza correlato; se non utilizzata, la motivazione alla base del suo mancato utilizzo deve essere dettagliata e concordata con il valutatore.

R/+: la tecnica o la misura è consigliata per questo livello di integrità della sicurezza come inferiore

raccomandazione a una raccomandazione HR/++ o come misura aggiuntiva del margine di sicurezza.

NR: la tecnica o la misura non è assolutamente raccomandata per questo livello di integrità della sicurezza.

# Adattamento dei metodi nel modello V

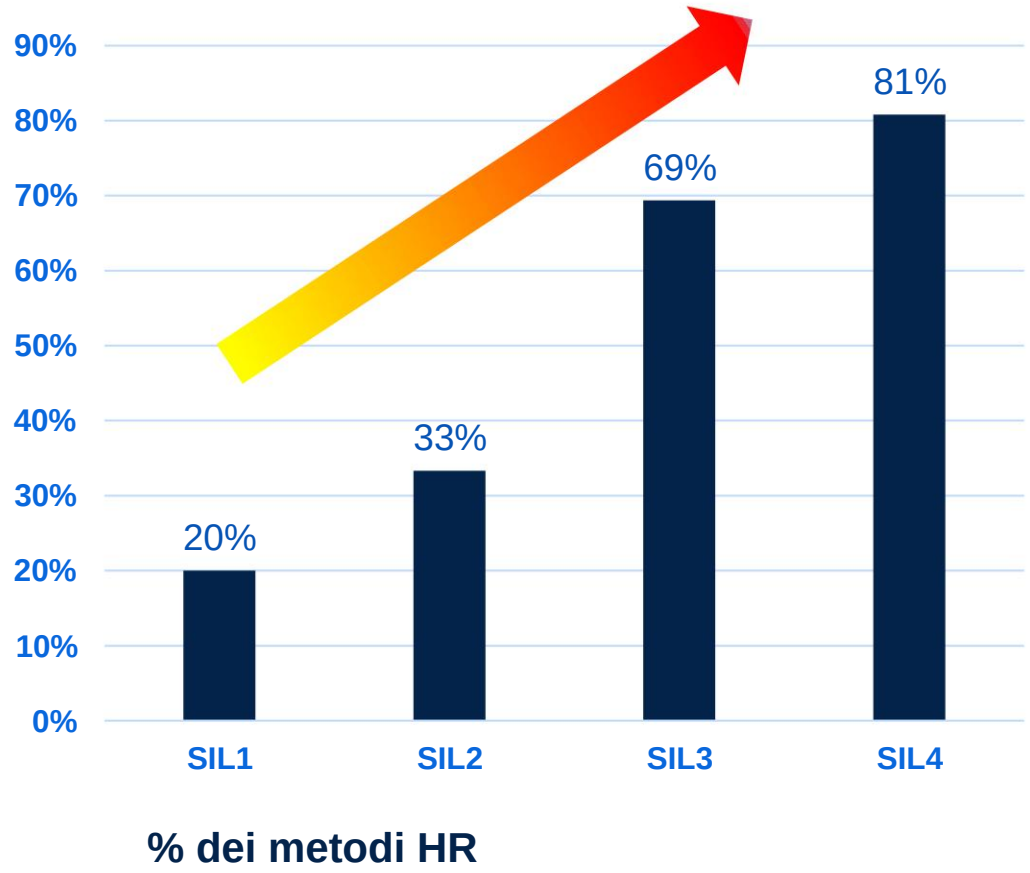
Esempio nozionale di tabella.

| | | SIL 1 | SIL 2 | SIL 3 | SIL 4 |
|---|---|---|---|---|---|
| 1 | Misure contro la rottura della tensione, variazioni di tensione, sovratensione, bassa voltaggio | R | Risorse umane | Risorse umane | Risorse umane |
| 2 | Aumento dell'immunità alle interferenze | R | Risorse umane | Risorse umane | Risorse umane |
| 3 | Separazione spaziale di più linee | Risorse umane | Risorse umane | Risorse umane | Risorse umane |
| 4 | … | … | … | … | … |
| 5 | … | … | … | … | … |

# Contenuto delle tabelle IEC 61508-3 vs SIL/SC

La richiesta di applicazione del metodo dipende fortemente dal livello SIL:

Le “raccomandazioni generiche” si trasformano in “prescrizioni dettagliate” sul lato inferiore della V (ad esempio coperture delle metriche software, approccio numerico)



# Requisiti

Un **requisito** è formalmente definito come:

una dichiarazione documentata

descrivere una condizione o una capacità

che un sistema, un prodotto o un servizio deve soddisfare

per soddisfare le esigenze o i vincoli delle parti interessate.

Questa definizione è coerente con i principali standard di ingegneria dei sistemi e del software.

I requisiti costituiscono il pilastro fondamentale del modello formale a V.

# Cosa è realmente necessario?


# Requisiti qualità

La norma ISO/IEC/IEEE 29148:2018 (Requirements Engineering) descrive esplicitamente gli attributi di qualità che i requisiti devono soddisfare, tra cui:

**Atomicità:** il requisito esprime una singola esigenza o capacità (per evitare ambiguità e complessità).

**Univocità:** il requisito ha una sola interpretazione.

**Completezza:** il requisito include tutte le informazioni necessarie.

**Coerenza:** il requisito non è in conflitto con gli altri.

**Verificabilità:** il requisito può essere verificato tramite ispezione, analisi, test o dimostrazione.

**Modificabilità:** il requisito può essere modificato senza introdurre errori.

**Tracciabilità:** il requisito può essere ricondotto alla sua origine e ai relativi artefatti.

**Correttezza:** il requisito riflette accuratamente le esigenze delle parti interessate.

**Assicurato dal modello V**

life.augmented

# Come scrivere (buoni) requisiti

Metodi semi-formali:

- Linguaggio naturale strutturato (SNL) – ("Il sistema deve [azione] [oggetto] [sotto condizioni].")
- Diagrammi di stato
- Tabelle decisionali
- Casi d'uso UML,

Metodi formali:

- Reti di Petri
- Lega
- ...

life.augmented

# Linguaggio naturale strutturato (SNL) - esempio

Basato su un modello coerente, ad esempio: [l'attore] deve [azione] [oggetto] [sotto condizioni].

Linee guida:

- Utilizzare un linguaggio chiaro e preciso
- Usa la forma attiva
- Utilizzare una terminologia coerente (glossario)
- Evitare affermazioni negative
- Utilizzare nomi singolari e numerazione coerente
- Limitare l'uso dei pronomi

# Argomento dimostrato in uso (*)

Sulla base della dimostrazione, supportata dall'esperienza operativa in un periodo di tempo specifico e prolungato, che la probabilità di guasti sistematici sconosciuti è sufficientemente bassa per il livello di integrità della sicurezza target.

Le principali problematiche legate a questo approccio sono:

ÿRichiede la presenza di una procedura di monitoraggio credibile per i guasti sul campo

ÿIn alcuni casi, difficilmente collegabile a specifici livelli di integrità sistematica

ÿDi solito, adatto solo per componenti molto semplici, perché le modifiche alla configurazione possono invalidare l'argomento

ÿDipendenza da molteplici fattori, incluso il processo di produzione dei componenti

*(*) si possono trovare nomi diversi sull'ecosistema degli standard di sicurezza*

# Valutazione degli strumenti

Le norme IEC 61508 e ISO 26262 definiscono un approccio strutturato per valutare l'affidabilità degli strumenti software utilizzati nello sviluppo e nella verifica dei sistemi di sicurezza. Ciò garantisce che gli strumenti non introducano o non rilevino errori che potrebbero compromettere la sicurezza funzionale.

La struttura è simile: gli strumenti vengono valutati in base alla loro capacità di influenzare (negativamente) l'implementazione o la verifica della funzione di sicurezza. Successivamente, si valuta la possibilità di identificare in un secondo momento i problemi introdotti.

Il risultato della valutazione è la prescrizione di requisiti specifici sugli utensili, che possono prevedere:

- Adozione di strumenti “certificati” sviluppati esplicitamente secondo un ciclo di vita di sicurezza
- Adozione di misure di mitigazione aggiuntive (ad esempio confronto degli output degli strumenti, comprovato nell'uso) discussione, ecc.)

# Valutazione degli strumenti – IEC61508

Uno strumento di supporto software offline è un'applicazione software che supporta una o più fasi del ciclo di vita dello sviluppo del software, ma non ha alcuna influenza diretta sul sistema di sicurezza durante il suo runtime. Questi strumenti sono classificati in tre classi in base alla loro interazione con il sistema di sicurezza:

**Strumenti T1:** questi strumenti non producono output che influiscano direttamente o indirettamente sul codice eseguibile (inclusi i dati) del sistema di sicurezza.

**Strumenti T2: questi** strumenti supportano il test o la verifica del progetto o del codice eseguibile. Sebbene errori in questi strumenti possano far sì che i difetti passino inosservati, non possono introdurre errori nel software eseguibile stesso.

**Strumenti T3:** questi strumenti generano output che contribuiscono direttamente o indirettamente al codice eseguibile del sistema correlato alla sicurezza.

# Valutazione degli strumenti – IEC61508

La scelta degli strumenti deve essere giustificata. Si applicano quindi i seguenti requisiti:

ÿ. **Documentazione:**

Tutti gli strumenti T2 e T3 devono avere specifiche o documentazione chiare che ne descrivano dettagliatamente il comportamento e i vincoli di utilizzo.

**ÿValutazione:**

Valutare gli strumenti T2 e T3 per comprendere quanto siano affidabili e identificare possibili modalità di errore che potrebbero avere un impatto sul software eseguibile. Applicare misure di mitigazione se necessario.

**ÿProva di conformità (solo T3):**

Fornire la prova che gli strumenti T3 soddisfano le specifiche, sulla base di un utilizzo positivo in passato e/o di una convalida formale.

# Metodi di analisi della sicurezza (FMEDA/FTA/DFA/ETA/Markov)

**Riepilogo:**

- **FMEA**
- **FMEDA**
- **Accordo di libero scambio**
- **DFA**
- **Analisi di Markov**

life.augmented

# Principi di FMEA (Analisi delle modalità e degli effetti dei guasti)

Identificare in modo proattivo le potenziali modalità di guasto nei prodotti, nei processi o nei sistemi per migliorare l'affidabilità e la sicurezza.

Elementi chiave:

- ÿ Modalità di guasto: modo specifico in cui una parte o un processo può guastarsi (ad esempio, crepa, cortocircuito). ÿ Effetto: impatto del guasto sul sistema o sull'utente (ad esempio, perdita di funzionalità, pericolo per la sicurezza). ÿ Causa: causa principale o fattore scatenante della modalità di guasto (ad esempio, difetto del materiale, errore umano).

Utilizzare il Risk Priority Number (RPN) o metriche simili basate su:

- ÿ Gravità (S): quanto è grave l'effetto. ÿ Occorrenza (O): probabilità che si verifichi un guasto. ÿ Rilevamento (D): probabilità di rilevare il guasto prima che raggiunga il cliente

Processo iterativo: aggiornare regolarmente la FMEA durante le modifiche alla progettazione, la produzione e il feedback sul campo per mantenere l'efficacia.

# Esempio di FMEA

## Esempio nozionale (processo)

| Processo<br>Fare un passo | Potenziale<br>Modalità di errore | Effetti potenziali Cause p | otenziali (S) (O) (D) Azioni consigliate da | ll'RPN | | | |
|---|---|---|---|---|---|---|---|
| Componenti di saldatura | Giunto di saldatura scadente | Malfunzionamento del dispositivo o guasto | Saldatura insufficiente errore dell'operatore | 9 4 | | 3 108 | Addestrare gli operatori, migliorare i controlli del processo di saldatura |
| Posizionamento dei componenti | Componenti disallineati | Cortocircuito o circuito aperto il circuito | Disallineamento durante posizionamento | 8 | 3 | 4 96 | Utilizzare macchine di posizionamento automatizzate, aggiungere l'ispezione visiva |
| Test di assemblaggio finale | Test funzionali incompleti | Dispositivi difettosi spedita al cliente | Procedura di prova incompleto | 10 2 | | 5 100 | Standardizzare le procedure di test, aggiungere una checklist di test |

- **Gravità (S):** Impatto sul sistema o sull'utente (scala da 1 a 10, 10 = più grave).
- **Occorrenza (O):** Probabilità di fallimento (scala da 1 a 10, 10 = più frequente).
- **Rilevamento (D):** probabilità che il guasto venga rilevato prima del rilascio (scala da 1 a 10, 1 = altamente rilevato).
- **RPN:** *RPN=S×O×DRPN=S×O×D;* valori più alti indicano una priorità più alta

life.augmented

# Principi di FMEDA (Modalità di guasto, effetti e Analisi diagnostica)

Simile alla FMEA con le seguenti specifiche:

ÿInclude l'indicazione della diagnostica (mirata a mitigare/rilevare i guasti)

ÿQuantitativo: eliminare l'RPN a favore dei calcoli dei tassi di errore.

ÿFornisce un risultato complessivo in termini di DC e SFF (SPF)

ÿPer ogni riga della modalità di guasto, include informazioni sul modello di guasto associato (per calcolare correttamente la distribuzione del guasto)

ÿSolo Hw: non può essere applicato a processi e software

L'argomento chiave è il problema della distribuzione dei guasti (come associare la modalità di guasto individuale al tasso di guasto del sistema)

# Principi dell'analisi dell'albero dei guasti (FTA)

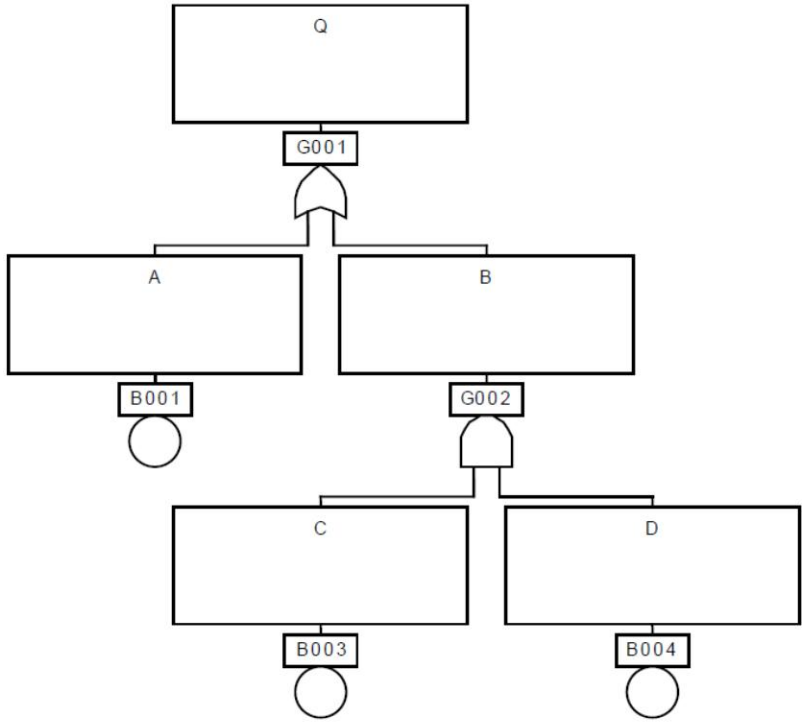
L'FTA è un metodo analitico deduttivo top-down utilizzato per identificare le cause dei guasti a livello di sistema.

Scopo: analizzare sistematicamente come le combinazioni di guasti di base possano portare a un evento critico indesiderato (l'evento principale).

Caratteristiche principali:

- ÿ Inizia con un evento principale definito (guasto o pericolo del sistema). ÿ
- Utilizza porte logiche (AND, OR) per mappare le relazioni tra guasti. ÿ L'analisi top-down
- scorre verso il basso fino a quando non viene trovato un evento di base (causa principale). ÿ Aiuta
- a visualizzare i percorsi di guasto e le loro interdipendenze.

# Principi dell'analisi dell'albero dei guasti (FTA)



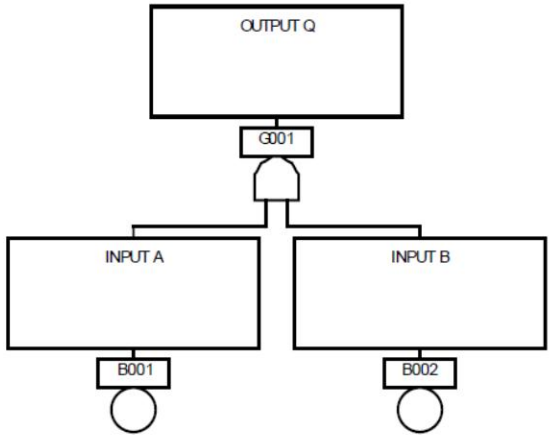




**Figure 4-5. The AND-Gate**

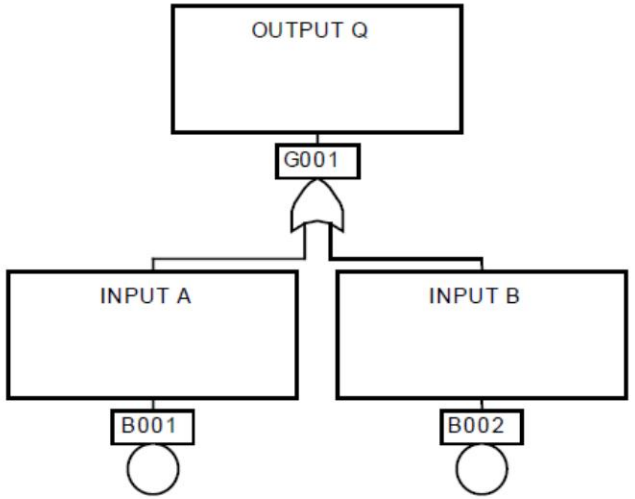



**Figure 4-2. The OR-Gate**

Per spiegazioni complete fare riferimento al documento di riferimento [R4], sezione 4.1 Simbologia: i componenti fondamentali dell'albero dei guasti

# Principi dell'analisi dell'albero dei guasti (FTA)

Breve riepilogo delle principali regole/specificità dell'FTA

**Regola del completamento del gate:** tutti gli input di un particolare gate devono essere completamente definiti prima di intraprendere un'ulteriore analisi di uno qualsiasi di essi.

**Nessuna regola gate-to-gate:** gli ingressi dei gate devono essere eventi di guasto correttamente definiti e i gate non devono essere collegati direttamente ad altri gate.

**Minimum Cut Set:** è la più piccola combinazione di guasti di base che può causare il guasto di livello superiore (evento principale). Rappresenta un insieme minimo di guasti dei componenti che portano al guasto del sistema e uno dei principali vantaggi dell'esecuzione di un'analisi FTA nel sistema.

# Analisi dei guasti dipendenti (DFA)

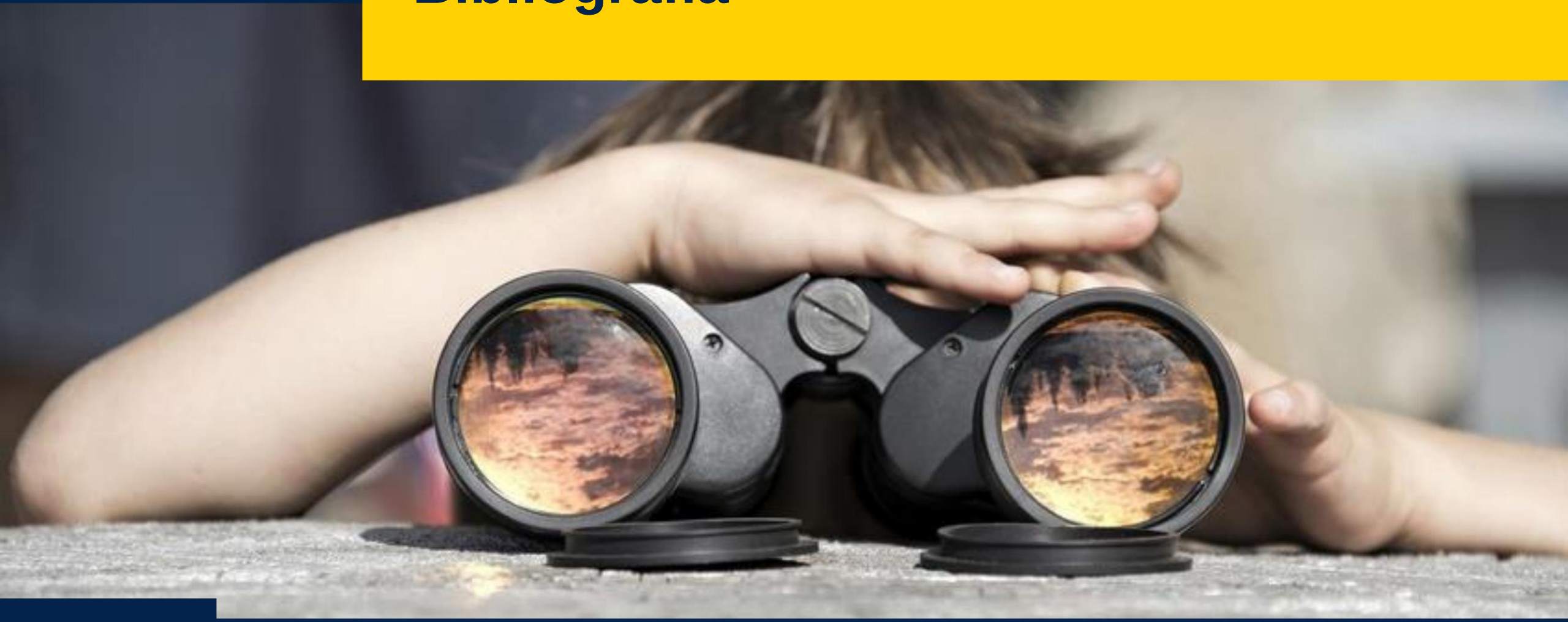
Lo scopo è identificare e analizzare i guasti che non sono indipendenti ma si verificano a causa di una causa comune o di una dipendenza tra i componenti...

Viene esplorato in dettaglio principalmente in ISO26262

Si avvale di altre tecniche di analisi della sicurezza (principalmente FTA e talvolta FMEA), concentrandosi sulla rilevazione di guasti dipendenti.

Gli standard di sicurezza aiutano la ricerca con tabelle guida specifiche (argomenti tipici da analizzare nella ricerca di tali DFA)

# Bibliografia



# Documenti di riferimento

[R1]: Affidabilità della microelettronica: modellazione basata sulla fisica dei guasti e valutazione della durata di vita - Jet Propulsion LaboratoryCalifornia Institute of Technology Pasadena, California

[R2]: : Manuale di affidabilità dei semiconduttori – Renesas Electronics, Rev.2.50 gennaio 2017

[R3]: ExoMars 2016 - Schiaparelli Anomaly Inquiry (ESA) scaricato da https://exploration.esa.int/web/mars/-/59176-exomars-2016-schiaparelli-anomaly-inquiry

[R4]: Manuale dell'albero dei guasti con applicazioni aerospaziali - Ufficio di sicurezza e garanzia della missione della NASA, V 1.1 2002 ,

[R5]: il software FTA aperto può essere trovato sul web, ad esempio https://www.fault-tree-analysis.com/free-fault-tree-analysis-software, oppure verifica il download di OpenFTA

# Grazie



life.augmented