

Sistemi Informativi

Marino Segnan

Sommario

- Motivazione (rationale)
 - che cos'è?
 - perché è importante?
- proprietà del rationale
 - rappresentazione
 - creazione
 - accesso
- stato attuale
- conclusioni

che cos'è il razionale?

- E' il ragionamento che ha portato al sistema
- Include:
 - questioni che sono state esaminate
 - le alternative considerate
 - le decisioni prese
 - i criteri di guida alla presa delle decisioni
 - il dibattito svoltosi durante il processo decisionale

Perche' il razionale?

- Grandi sistemi SW sono simili ad aeroplani passeggeri: sono il prodotto di un gran numero di decisioni prese lungo un esteso arco temporale
 - evolvono i presupposti
 - decisioni in base a vincoli passati
 - criteri conflittuali
- ✂ →
 - alti costi di manutenzione
 - perdita e riscoperta di informazioni

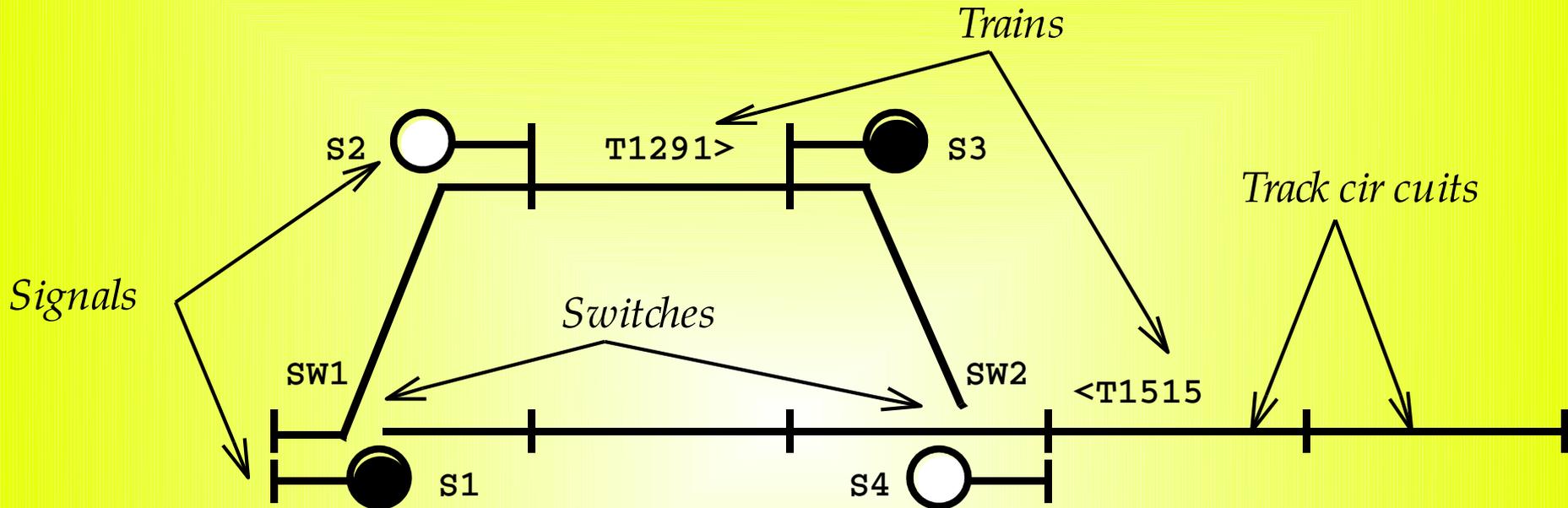
Il razionale aiuta ad affrontare i cambiamenti

- supporta la manutenzione
 - fornisce il contesto del progetto ai manutentori
- migliora l'apprendimento
 - nuovi progettisti apprendono il progetto ripercorrendo le decisioni che lo hanno generato
- migliora analisi e progetto
 - evita ripetute valutazioni di alternative scadenti
 - rende consistenti ed espliciti i compromessi

Livelli di razionale

- niente razionale
 - i motivi sono contenuti in documenti, e-mail, memoria dei partecipanti
- ricostruzione
 - un documento descrive le motivazioni conducenti al progetto finale
- acquisizione
 - i motivi sono registrati durante lo svolgimento del progetto
- integrazione
 - le motivazioni dirigono il progetto

Esempio di Controllo Traffico Centralizzato



- Il sistema consente di controllare remotamente il traffico
- in caso di problemi i percorsi possono essere variati

Questioni (issues)

- sono problemi concreti che non hanno una unica soluzione corretta
- mostrati come domande:

display?:Issue

come mostrare I tratti di binario?

input?:Issue

come deve inserire I comandi il controllore?

Proposte

- le risposte alle questioni
- condivisibili fra piu' questioni

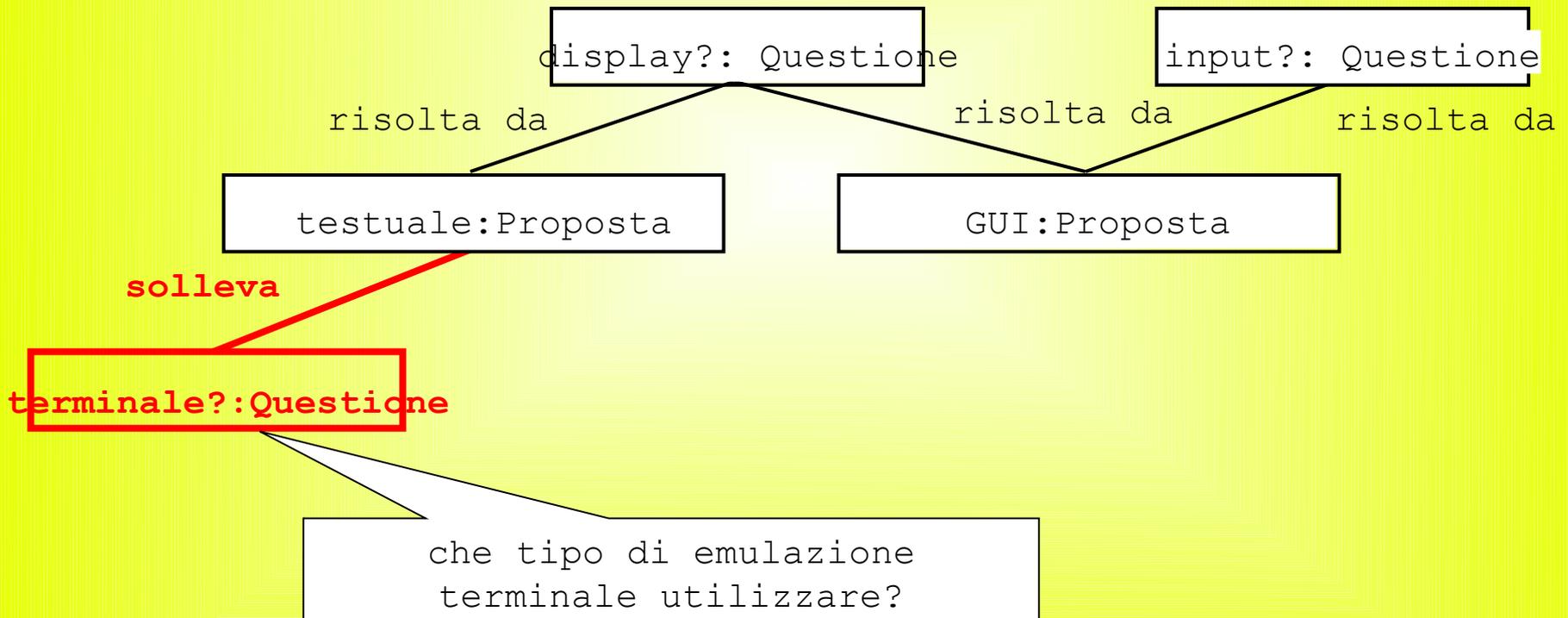


Lo schermo del controllore puo' essere testuale con caratteri grafici per disegnare I binari

l'interfaccia per il controllore potrebbe essere grafica

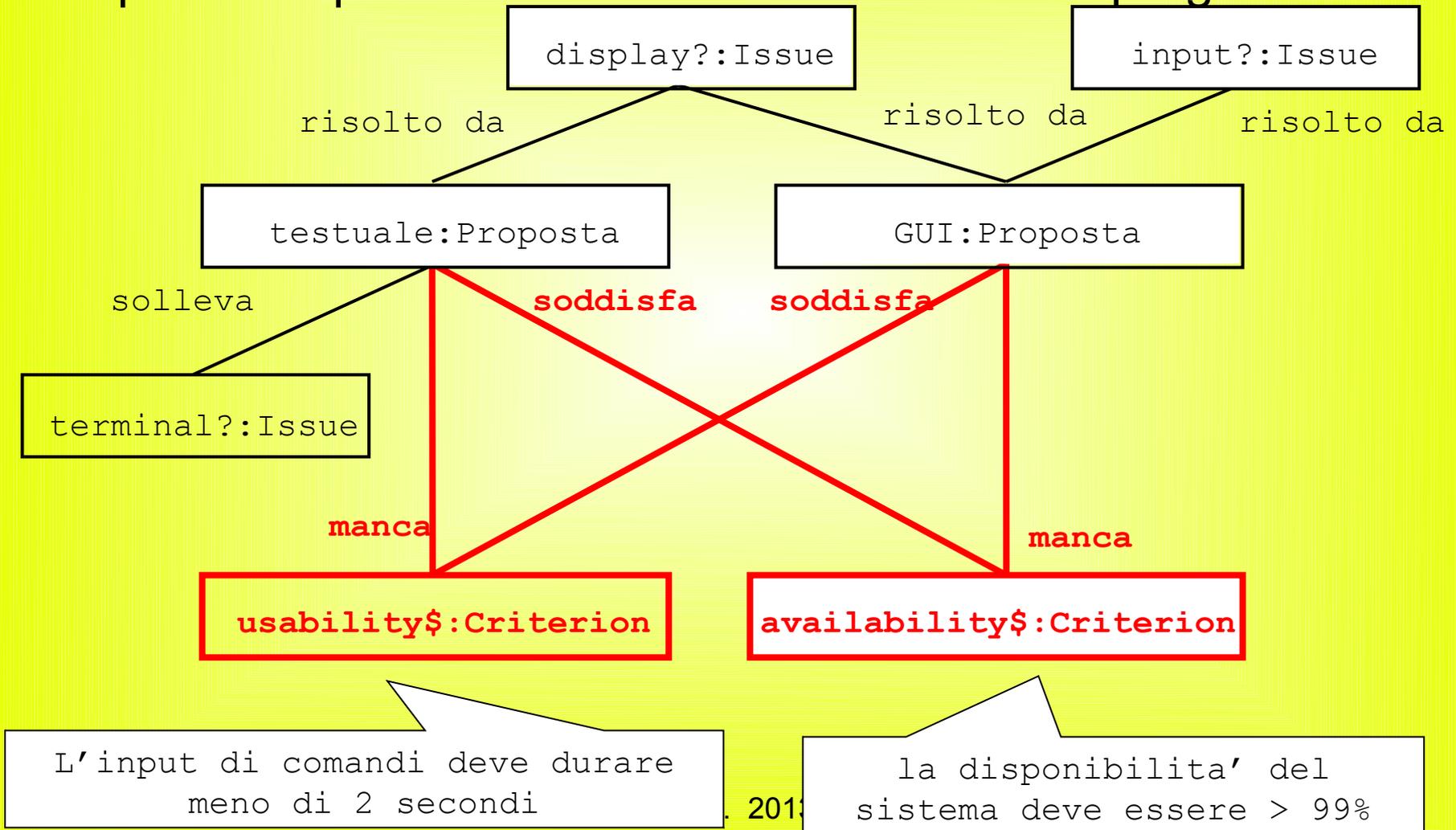
Questioni conseguenti

- introdotte dalle proposte



Criteri

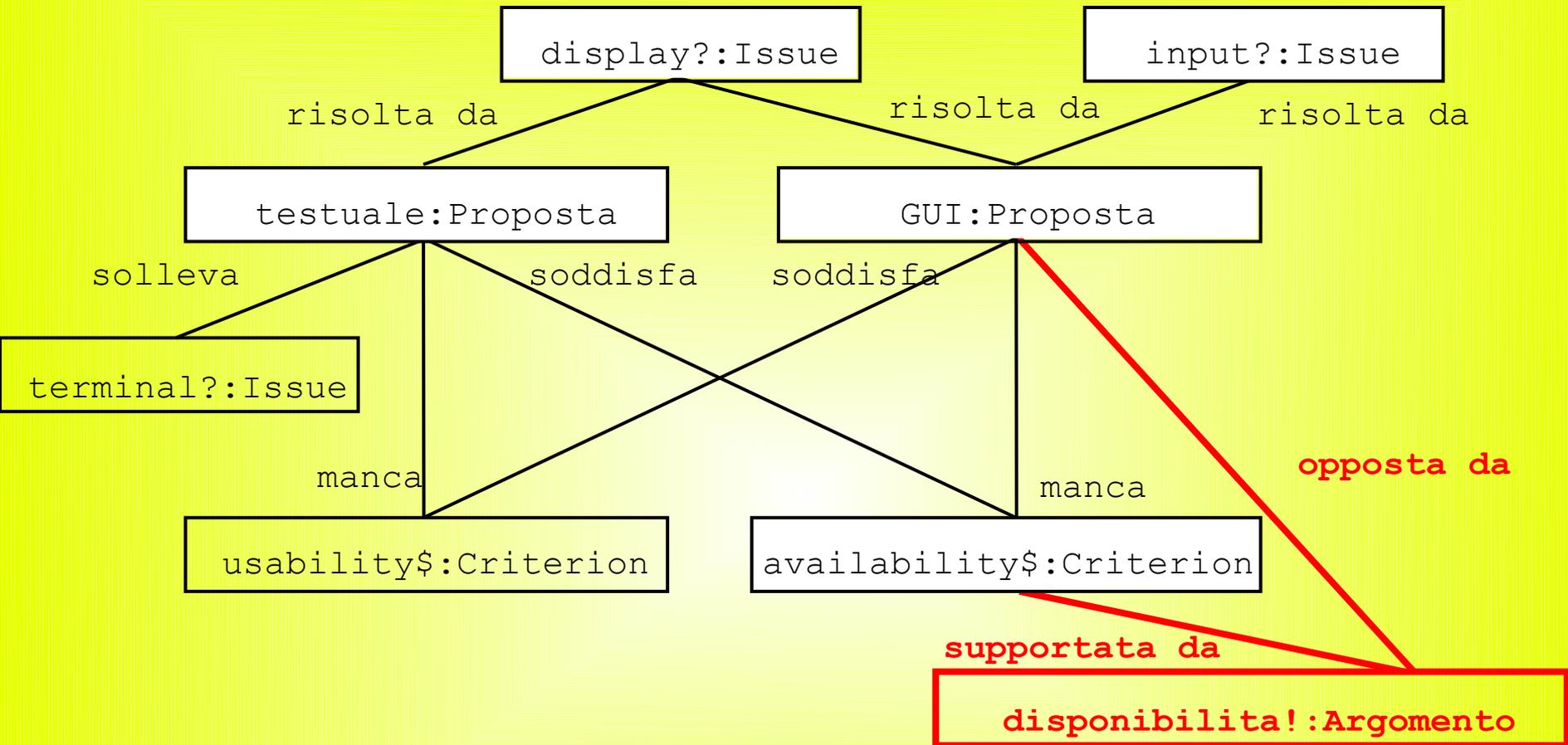
- per quantificare la bonta' di una proposta
- spesso requisiti non funzionali o obiettivi del progetto



Argomentazioni

- la maggior parte del razionale
- la discussione per arrivare ad una decisione
- puo' contrastare o appoggiare qualunque altra parte del documento
- importanti le **motivazioni** che scartano le proposte **non adottate**

Argomentazioni

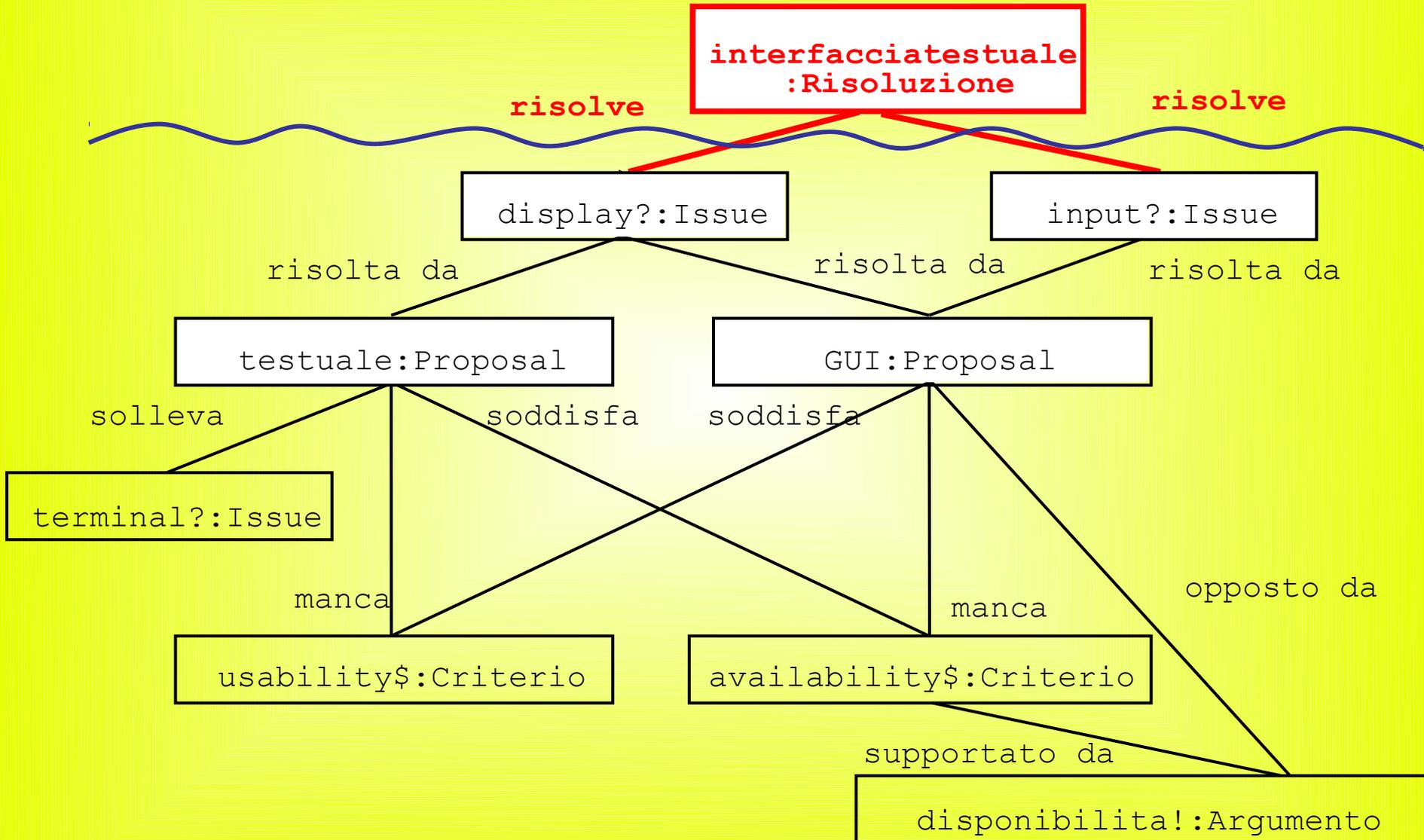


GUI sono piu' difficili da implementare e testare .Potrebbero percio' introdurre errori fatali nel sistema che controbilancerebbero i benefici di usabilita' introdotti

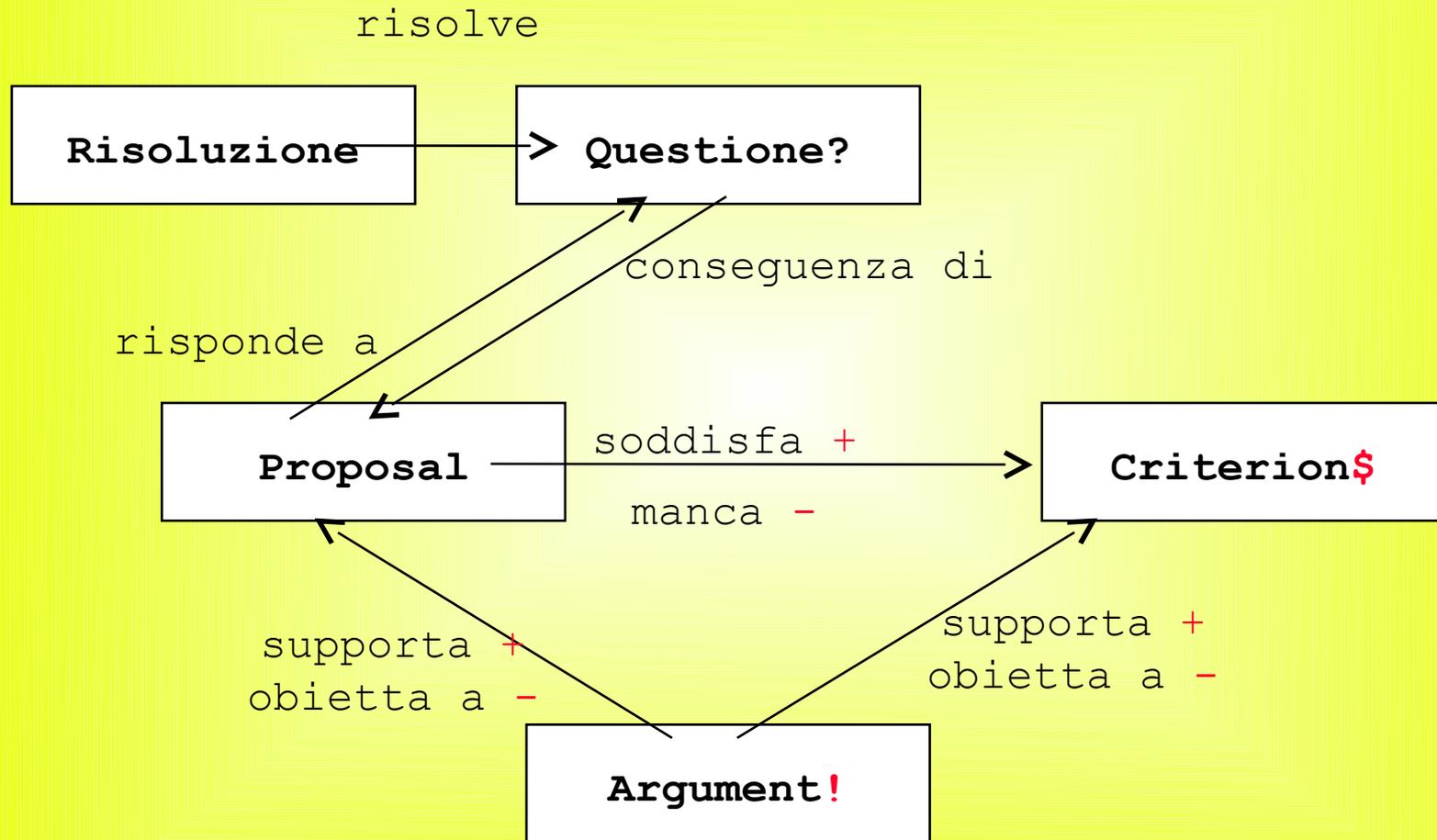
Risoluzioni

- rappresentano le decisioni
- riassumono l'alternativa prescelta e le sue motivazioni
- Una questione risolta si dice **chiusa**
- Può essere riaperta con automatica demozione della risoluzione

Risoluzioni



Rappresentazione del razionale: modello di una questione



Rappresentazione del razionale

- molti modelli proposti:
 - IBIS, QOC, DRL, WinWin
 - simili, con diversi livelli di dettaglio
 - devono essere integrati ovviamente con I tool di project management e la metodologia di sviluppo

Creazione del razionale

- molta informazione da raccogliere
- distrae gli sviluppatori (un po' come la contabilità')
- e' costoso formalizzare la conoscenza

Esempio di discussione

- il responsabile pubblica una agenda i cui gli oggetti sono delle **questioni**
- i partecipanti rispondono con **proposte** o altre **questioni**
- incorporate le risposte, il responsabile organizza la riunione in cui si discute un singolo albero di questioni
- il verbalizzatore registra tutto in termini di **questioni, proposte, criteri, argomenti**
- il verbalizzatore registra le decisioni in termini di **risoluzioni e compiti** (action items)

(Bruegge) Record and replay example: database discussion agenda

3. Discussion

I[1] Which policy for retrieving tracks from the database?

I[2] Which encoding for representing tracks in transactions?

I[3] Which query language for specifying tracks in the database request?

(Bruegge) Record and replay example: database discussion

I[1] Which policy for retrieving tracks from the database?

Jim: How about we just retrieve the track specified by the query? It is straightforward to implement and we can always revisit it if it is too slow.

Ann: Prefetching neighboring tracks would not be much difficult and way faster.

Sam: During route planning, we usually need the neighbor tracks anyway. Queries for route planning are the most common queries.

Jim: Ok, let's go for the prefetch solution. We can revert to the simpler solution if it gets too complicated.

(Bruegge) Record and replay example: database discussion minutes

3. Discussion

I[1] Which policy for retrieving tracks from the database?

P[1.1] Single tracks!

A- Lower throughput.

A+ Simpler.

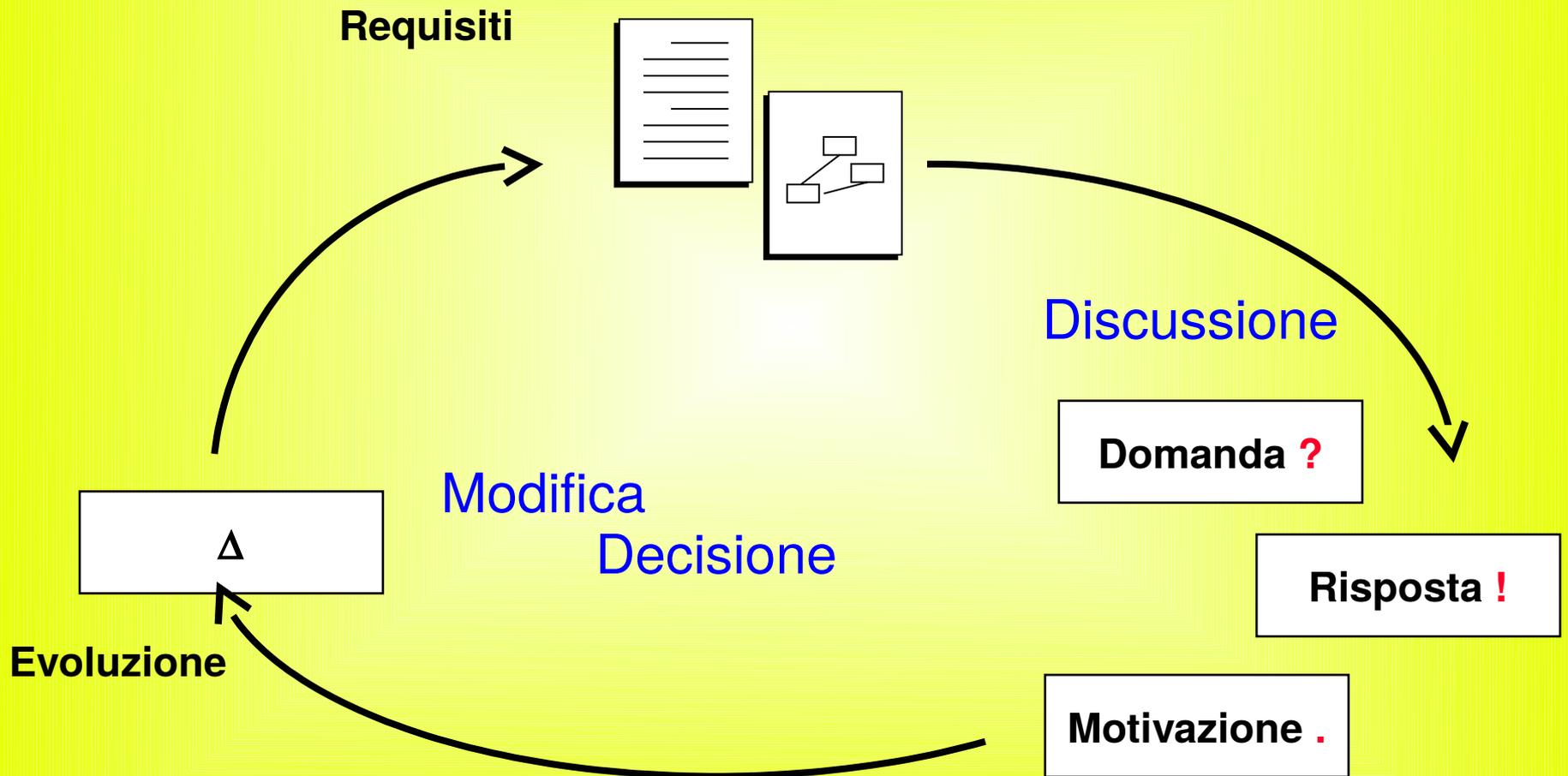
P[1.2] Tracks + neighbors!

A+ Overall better performance: during route planning, we need the neighbors anyway.

{ref: 1/31 routing meeting}

R[1] Implement P[1.2]. However, the prefetch should be implemented in the database layer, allowing use to encapsulate this decision. If all else fails, we will fall back on P[1.1].

Applicazione della metodologia all' analisi dei requisiti



Accesso al razionale

– Browsing e ricerca

- testuale permette la ricerca dei nodi interessanti
- il modello delle questioni collega rapidamente i nodi interessanti

– Critica del progetto

- il razionale consente di valutare rapidamente la qualità' di un progetto

Problemi

- Formalizzare la conoscenza e' costoso
 - mantenere un modello consistente e' costoso
 - acquisire e mantenere il razionale ancor di piu'
- Gli sviluppatori del momento non percepiscono benefici
 - la prioritá' dell'attivitá' sara' sempre bassa
- Gli approcci proposti non gestiscono formalmente problemi di grosse dimensioni

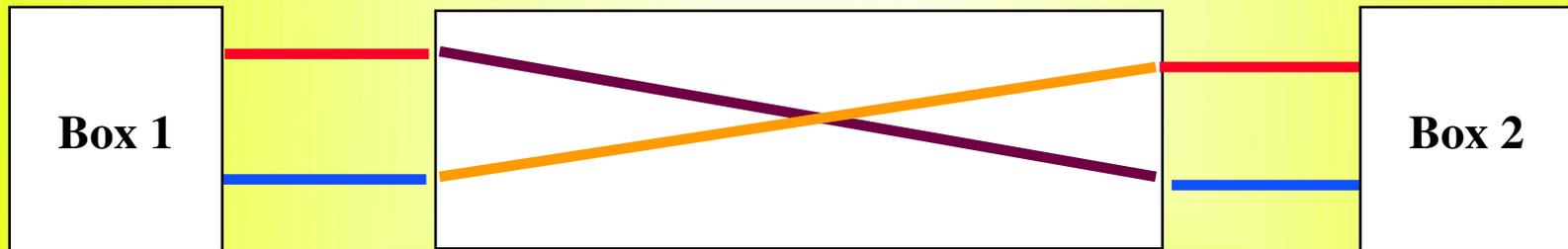
Accenni a problemi di comunicazione (1)

- “due scatole di comunicazione fabbricate da due diversi fornitori erano collegate da una coppia di fili”



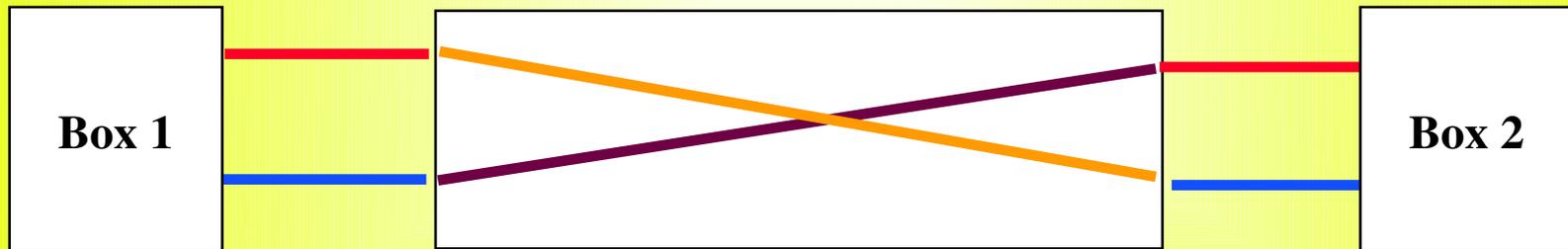
Esempio di comunicazione (2)

- “grazie ad un attento controllo si scopre che i fili sono invertiti”



Esempio di comunicazione (3)

- “entrambi i fornitori pongono rimedio al problema ...”



La comunicazione e' importante

- in grandi progetti piu' tempo per comunicare che per scrivere il progetto
- un ingegnere deve imparare le “soft skills” :
 - gestire una riunione
 - presentazione di una parte del progetto
 - collaborare con i membri ed altri gruppi
 - negoziare i requisiti col cliente
 - scrivere documentazione tecnica del progetto

ruoli in una riunione

- **gestore (primary facilitator)**
 - organizza la riunione e la guida
 - definisce l'ambito e scrive l'agenda
 - distribuisce l'agenda ai participant
- **segretario verbalizzante**
 - registra la riunione
 - identifica le azioni ed i problemi
 - li fornisce ai partecipanti
- **gestore del tempo**
 - controlla la tempistica

esempio di agenda

informazioni generali	data, ora, durata, edificio, sala, gestore, verbalizzatore, responsabile del tempo
risultati attesi	capire che cosa ci ostacola nell'inizio del prototipo
stato attività già previste	durata: 15 minuti situazione del codice del parser comandi
problemi da discutere e risolvere	durata: 35 minuti come trattare input senza formato? come trattare l'output? compatibilità con versioni precedenti
conclusioni	durata : 5 minuti assegnare i compiti considerazioni sullo svolgimento della riunione

Riassumendo

- acquisire la storia decisionale e' importante:
 - Argomenti alternativi
 - criteri progettuali espliciti
 - importante per future modifiche
- il modello delle questioni
 - offre una soluzioni strutturata per la acquisizione
 - facilita la navigazione della documentazione
- la pratica deve essere adottata durante tutto il processo, ma occorrera':
 - ridurre la distrazione
 - fornire incentivi a breve termine