Ingegneria del Software mod. A
Progettazione di software reattivo

Docente: Tullio Vardanega
tullio.vardanega@math.unipd.it

Perché distinguere

- Significative differenze, sia funzionali che non-funzionali, intercorrono tra sistemi *software* definiti come *reattivi* e quelli definiti come *interattivi*
  - Le differenze funzionali delineano i diversi domini di applicazione e le diverse modalità di servizio
  - Le differenze non-funzionali caratterizzano le diverse architetture di sistema e le diverse modalità di sviluppo, di verifica e di servizio

Definizione: sistema reattivo

- I sistemi reattivi non possono essere adeguatamente descritti in termini puramente funzionali o relazionali
  - La visione funzionale o relazionale descrive un programma come una trasformazione da uno stato iniziale ad uno stato finale
  - I sistemi reattivi agiscono permanentemente su (od interagiscono con) un ambiente circostante
    - Tale interazione non ha, normalmente, uno stato terminale

Caratterizzazione - 1

- I sistemi reattivi devono avere conoscenza dettagliata delle caratteristiche loro ambiente di esecuzione
- Tali caratteristiche possono avere grande influenza sull'architettura e sul comportamento del programma
  - Tipo di processore
  - Struttura e mappa di memoria
  - Tipo ed interfaccia dei dispositivi collegati
  - Sorgenti, forme e modalità di interruzione
  - Gestione dell'orologio

Caratterizzazione - 2

- Il *software* di un sistema reattivo è parte integrante ed essenziale dell'ambiente di sistema
  - L'uno ha bisogno dell'altro
    - Il *software* senza l'ambiente di sistema perde la sorgente e la destinazione dei propri stimoli e delle proprie attività
    - Il sistema (l'ambiente) senza il *software* perde il suo elemento di integrazione e di controllo

Caratterizzazione - 3

- Il *software* reattivo è normalmente soggetto a vincoli di tempo reale
  - Non esprimibile come velocità di esecuzione
    - Né troppo presto, né troppo tardi, con una tolleranza prefissata
  - Il flusso di controllo all'interno del programma è regolato dall'arrivo di eventi che richiedono un trattamento (di controllo, di risposta) da completarsi entro un preciso limite temporale (*deadline*)
  - Questo rende necessaria la conoscenza dei meccanismi e delle politiche di ordinamento (*scheduling*) delle attività

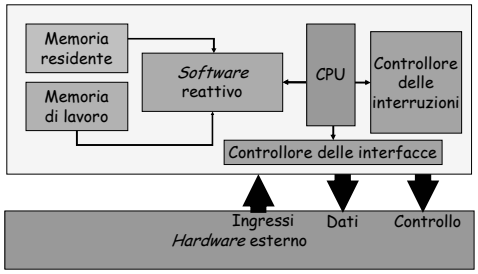
Caratterizzazione - 4

- Il *software* reattivo può interagire anche con dispositivi convenzionali quali schermo, *mouse* e tastiera, ma con azioni a bassa priorità
  - Basse frequenze di arrivo a fronte di ampi intervalli di risposta
- La principale attività di interazione si svolge però con dispositivi non guidati da operatore umano (p.es.: sensori, attuatori)

Definizione: sistema interattivo

- Il *software* interattivo è generalmente soggetto a vincoli temporali laschi od inesistenti
  - Limitate conseguenze da tempi di risposta variabili o non predicibili
- Non gestisce la piattaforma *hardware* del sistema
  - La piattaforma è prevalentemente di tipo *general-purpose*
  - La sua gestione è demandata ad un sistema operativo indipendente dal programma
- Opera in stretto contatto con (in risposta ai comandi di un) operatore

Sistema reattivo: visione a blocchi



Sistema reattivo: visione a livelli

Definizione: sistemi a tempo reale

- Dall'*Oxford Dictionary of Computing*
  - Si definisce "sistema a tempo reale" qualunque sistema nel quale l'istante temporale nel quale i valori di uscita sono prodotti è significativo
- Inoltre
  - Il tempo di reazione agli stimoli provenienti dall'ambiente (incluso il trascorrere del tempo) è determinato dalle caratteristiche ed i requisiti dell'ambiente

Conseguenze della definizione

- La correttezza di un sistema a tempo reale (da dimostrare mediante validazione) dipende non soltanto dal valore prodotto da una calcolo, ma anche dall'istante nel quale esso sia stato prodotto
  - Correttezza (di comportamento) funzionale
    - Rispetto al valore della risposta
  - Correttezza (di comportamento) temporale
    - Rispetto al tempo di risposta
- Una risposta ritardata equivale ad una risposta errata

Importanza del tempo di risposta

- Sistemi a criticità massima (*hard*)
  - Quelli nei quali è assolutamente necessario che la risposta venga fornita entro la scadenza (*deadline*) assegnata
- Sistemi a criticità elevata (*firm*)
  - Quelli nei quali la degradazione di servizio causata da risposte *occasionalmente* ritardate è tollerabile entro limiti fissati
- Sistemi a criticità contenuta (*soft*)
  - Quelli nei quali le risposte possono essere ritardate entro limiti fissati senza causare danno al sistema

Aspetti architetturali

- Un sistema reattivo (a tempo reale) è tipicamente composto da uno o più elaboratori collegati a dispositivi esterni indipendenti, con ciascuno dei quali il programma interagisce concorrentemente
  - Questo riflette la natura del mondo reale, dove ciascun elemento costitutivo esiste ed opera concorrentemente agli altri
  - Può essere conveniente rappresentare direttamente tale concorrenza nell'architettura del programma

Esecuzione ciclica o concorrente

- La rappresentazione della concorrenza inerente ad un sistema reattivo può essere
  - Diretta (esplicita): tanti flussi di controllo concorrenti quante le entità concorrenti del sistema
  - Indiretta (sequenziale): un solo flusso di controllo che determina esplicitamente la sequenza di attività da eseguire

Raffronto tra strategie

- La rappresentazione indiretta presenta svantaggi non trascurabili
  - Comporta la creazione e la gestione di sovrastrutture non appartenenti al problema (ma solo alla sua rappresentazione)
  - Complica la fase di costruzione laddove il comportamento delle attività da serializzare sia inizialmente incerto, rendendolo difficilmente modificabile
  - Comporta una decomposizione artificiale del problema e del sistema
  - Causa bassa scalabilità ed elevato accoppiamento, specie in relazione al trattamento dei guasti

Strumenti linguistici

- La rappresentazione diretta di concorrenza richiede l'ausilio del linguaggio di programmazione e del suo supporto a tempo di esecuzione per
  - Specificare l'istante temporale di attivazione di date attività
  - Specificare l'istante temporale di completamento (i.e. non oltre) di tali attività
  - Rilevare e gestire la violazione dei vincoli temporali
  - Consentire, dove richiesto, la modifica dinamica dei vincoli temporali

Criteri guida - 1

- Modello di rappresentazione della concorrenza (*computational model*)
  - Raccoglie, in modo organico, i criteri ed i vincoli per la progettazione e la codifica del programma
  - Consente di assicurare che il programma avrà, per costruzione, un comportamento predicibile, misurabile, analizzabile e gestibile in modo coerente e sistematico
  - Consente il disaccoppiamento dei componenti del sistema e, conseguentemente, il loro sviluppo parallelo

Criteri guida - 2

- Un modello di rappresentazione adeguato definirà
  - Il tipo dei componenti a partire dai quali il sistema potrà essere costruito
    - Il comportamento temporale che li caratterizza ed i corrispondenti requisiti loro assegnati dal sistema
    - Le loro regole d'uso ai fini di una costruzione consistente del sistema
  - Le modalità ed i meccanismi di comunicazione e sincronizzazione tra componenti e con l'ambiente esterno
  - Le modalità di esecuzione ed il corrispondente algoritmo di ordinamento
    - Le caratteristiche (p.es.: accuratezza, politica) dell'implementazione dell'algoritmo di ordinamento
    - Le tecniche di analisi statica dell'ordinamento conseguente

Esempio - 1

- Un modello di rappresentazione reale
  - Tipo dei componenti
    - Ciclico : ad attivazione su base temporale periodica
    - Sporadico : ad attivazione su evento interno (software) od esterno (interruzione)
    - Singolo evento di attivazione per componente, seguito da attività indivisa
  - Modalità di comunicazione e sincronizzazione
    - Asincrona a strutture condivise protette mediante regioni critiche (monitor) con priorità superiore

Esempio - 2

- (segue)
  - Modalità di esecuzione ed ordinamento
    - Esecuzione concorrente su singola CPU
    - Ordinamento di flussi di controllo concorrente con selezione secondo priorità con prerilascio
    - Orologio accurato, con espressione di tempo di attesa assoluto
  - Tecniche di analisi statica associate
    - Analisi del tempo di risposta
      - Dato il tempo di esecuzione indipendente di ciascun componente, determina le interferenze che esso causa e subisce dagli altri componenti di sistema, sia indipendenti che cooperanti (i.e.: tempo di attesa su risorsa bloccata), ed il tempo di completamento (di risposta) che ne deriva

Java: carenze di modello

- Permette concorrenza mediante `thread` con priorità
  - Inadeguati livelli di priorità
- Permette comunicazione e sincronizzazione mediante metodi `synchronized` con innalzamento di priorità
  - Inefficace la strategia di innalzamento di priorità
- Permette attese temporali, mediante `sleep`, o su evento
  - Consente periodi di attesa scarsamente accurati (non meno di; esattezza non garantita) su tempi relativi
- Prevede, ma non garantisce, ordinamento a priorità (statica) con prerilascio