

Validazione dell'architettura

Dispensa ASW 270
ottobre 2014

*Se pensi che il tuo progetto sia perfetto,
è solo perché
non l'hai mostrato a nessun altro.*

Harry Hillaker



- Fonti

- [SAP] Chapter 21, Architecture Evaluation
- [SSA] Chapter 14, Evaluating the Architecture
- [ESA] Clements, Kazman, Klein, Evaluating Software Architectures, 2002 – Chapter 3, The ATAM – A Method for Architecture Evaluation
- [Kazman, Klein, Clements] ATAM: Method for Architecture Evaluation, Technical report CMU/SEI-2000-TR-004, 2000



* Validazione dell'architettura

- Definire un'architettura non vuol dire solo prendere decisioni di progetto e riportarle in una descrizione architeturale (AD)
 - un'AD (o una sua bozza) descrive solo una possibile architettura candidata
 - ma potrebbe non garantire che gli interessi delle parti interessate siano effettivamente soddisfatti da tale architettura
- In effetti, la definizione di un'architettura richiede lo svolgimento di tre attività fondamentali
 - **analisi architeturale** – identificare/definire il problema che l'architettura deve risolvere
 - **sintesi architeturale** – proporre una soluzione architeturale a fronte del problema che è stato identificato
 - **valutazione (o validazione) dell'architettura** – assicurare la bontà delle decisioni di progetto prese – misurando la soluzione proposta rispetto al problema identificato



Validazione dell'architettura

- La **validazione di un'architettura** ha lo scopo di verificare la “bontà” dell'architettura
 - validazione, non verifica (testing) – l'AD non è eseguibile
 - attività da iniziare il prima possibile, per mitigare importanti rischi tecnici
 - “le scelte sono facili da prendere, ma è difficile cambiarle quando il sistema è implementato”
 - “l'architettura rappresenta le decisioni di progetto importanti che danno forma a un sistema – dove l'importanza è misurata dal costo di cambiamento” [Grady Booch]
 - in realtà, un processo di validazione continua



Validazione dell'architettura

- Quali gli scopi della validazione di un'architettura?
 - non stabilire delle stime precise delle varie qualità
 - ma soprattutto comprendere le conseguenze delle decisioni architettoniche sulla possibilità di controllare le qualità richieste
 - identificare possibili rischi
 - verificare che le problematiche di qualità siano state affrontate
 - comprendere i compromessi tra le qualità del sistema



* Perché validare l'architettura?

- Per le limitazioni dell'AD
 - validazione di astrazioni
 - l'AD è un'astrazione della realtà – le astrazioni effettuate sono ragionevoli e appropriate?
 - verifica della correttezza tecnica
 - l'AD non è eseguibile – “bubbles don't crash”
- Per comunicare
 - per “vendere” l'architettura
 - “dimostrare” alle parti interessate che l'architettura soddisfa i loro interessi
 - per spiegare l'architettura
 - per coinvolgere le varie parti interessate che non vogliono leggere l'AD



Perché validare l'architettura?

- La progettazione coinvolge compromessi
 - le qualità di un sistema dipendono fortemente dalle decisioni architettoniche
 - il sostegno a una qualità spesso avviene a spese di altre qualità
 - importante comprendere le conseguenze di alternative nelle decisioni di progetto alla luce dei requisiti di qualità
 - l'architettura software è il primo elaborato nella vita di un sistema in cui sono presenti decisioni di progetto significative – scelte e compromessi – che vanno comprese/i e validate/i



Perché validare l'architettura?

- Validazione nell'ambito di un processo software
 - validazione di ipotesi e scelte di progetto
 - prima che sia troppo tardi per cambiare idea
 - fornire punti per decisioni manageriali
 - per decisioni go/no-go, prima che siano stati spesi troppi soldi
 - base per un accordo formale
 - per garantire l'integrità tecnica
- Per il valore fornito dalla validazione
 - costo limitato – da uno-due giorni a una-due settimane
 - rispetto al costo di un'architettura povera – può essere disastroso
- In sintesi – miglioramento dell'architettura



* Tecniche di validazione

- Alcune qualità possono essere “misurate” in modo “quantitativo” usando delle opportune “metriche”
 - ad esempio, coesione e accoppiamento
 - tali tecniche possono solitamente essere applicate solo al codice – ma non a un progetto o a un’architettura
 - recentemente sono stati definiti (ed apprezzati) metodi per la validazione di architetture di natura “qualitativa” e “basati su scenari”

- Alcuni approcci per la validazione di un’architettura software
 - presentazioni (informali)
 - revisioni formali e walkthrough strutturati
 - valutazione basata su scenari
 - prototipi e sistemi proof-of-concept
 - sistemi scheletro



- Revisioni formali e walkthrough strutturati

- Revisione formale
 - in un gruppo di persone
 - lettura e analisi di un documento pagina per pagina – con commenti e discussioni – giungendo a un accordo sulle eventuali azioni da intraprendere
 - possibile per specifiche e altri documenti scritti, algoritmi, codice, AD, ...

- Walkthrough strutturato
 - “passeggiata” o “prova”, nel senso di prova teatrale
 - simile alla revisione formale
 - presentazione e analisi basata sull’ “esecuzione” di uno o più scenari



Revisioni formali e walkthrough strutturati

□ I partecipanti alla revisione

- sono tipicamente da tre a cinque
- hanno un ruolo – stabilito in anticipo
 - moderatore
 - presentatore – di solito l'autore del documento da validare
 - revisori
- i revisori ricevono il documento da validare in anticipo
- il moderatore svolge il ruolo di segretario
- l'autore deve poi modificare il documento sulla base dei commenti e delle decisioni presi



Revisioni formali e walkthrough strutturati

□ Conseguenze

- ☺ i partecipanti sono molto più coinvolti – possibile un feedback di maggior valore
- ☹ molto più costosa di una presentazione – soprattutto per la preparazione richiesta ai partecipanti
- ☹ i risultati dipendono molto dall'effettiva preparazione dei partecipanti prima dell'incontro



- Valutazione basata su scenari

- Valutazione basata su scenari
 - un approccio strutturato per valutare come l'architettura soddisfa i bisogni delle parti interessate – in termini di qualità esibite dall'architettura
 - ad esempio
 - SAAM – Software Architecture Assessment Method
 - ATAM – Architecture Tradeoff Analysis Method
 - basato su un insieme di scenari architetaturalmente significativi
 - che dovrebbero consentire la validazione dell'architettura
 - possibile sia una valutazione informale che una valutazione più strutturata e formale



Valutazione basata su scenari

- Passi fondamentali
 - comprensione dei requisiti
 - comprensione dell'architettura proposta – AD e presentazione
 - identificazione di scenari (con priorità)
 - alcuni scenari funzionali – altri scenari di qualità
 - analisi dell'architettura
 - viene considerato uno scenario alla volta, analizzando come (quanto) l'architettura soddisfa gli interessi in quella particolare situazione
 - consente di identificare eventuali punti deboli – nonché conflitti tra scenari
 - trarre conclusioni
 - piani per affrontare scenari critici



Valutazione basata su scenari

□ Conseguenze

- ☺ può fornire un'analisi profonda e sofisticata dei punti di forza e di debolezza di un'architettura
- ☺ porta a una comprensione esplicita dei compromessi fatti dall'architettura – aiuta a spiegare le motivazioni per le scelte architettoniche alle parti interessate
- ☺ aiuta le persone stesse che lavorano all'architettura a capire le decisioni prese, le loro motivazioni e le loro implicazioni
- ☹ più complesso e costoso di presentazioni e revisioni
- ☹ richiede una preparazione significativa ai partecipanti
- ☹ benefici scarsi se le parti interessate non possono partecipare in modo profondo, “committed” e costruttivo



- Prototipi e sistemi proof-of-concept

□ Utili per ridurre rischi tecnici

- *prototipo* – un sottoinsieme funzionale del sistema
- *proof-of-concept* – codice che ha lo scopo di dimostrare la fattibilità di un elemento rischioso dell'architettura proposta

□ Conseguenze

- ☺ forniscono prove concrete della validità di decisioni tecniche – ad un punto del progetto in cui cambiamenti sono ancora possibili
- ☺ opportunità per imparare e comprendere una tecnologia
- ☺ dimostrazioni utili alle parti interessate
- ☹ costosi da creare, richiedono tempo – vanno usati solo per decisioni importanti da giustificare i costi



- Sistemi scheletro

- La forma definitiva della validazione dell'architettura – la costruzione dell'architettura
 - inizia con una prima versione *scheletro* del sistema – che implementa la struttura del sistema ma contiene solo un nucleo minimale delle funzionalità – tale da provare la bontà dell'architettura
 - se accettato, viene mantenuto – non scartato
- Conseguenze
 - ☺ la migliore forma di validazione possibile
 - ☺ il sistema scheletro viene usato anche dopo la validazione
 - ☹ la forma più costosa di validazione
 - ☹ richiesta esperienza e capacità di sviluppo notevoli



* Il metodo di valutazione ATAM

- La validazione delle architetture può essere basata sull'applicazione di metodi di valutazione basati su scenari
 - metodi maturi, applicati in numerosi casi e convalidati dall'esperienza
- **SAAM** è il capostipite di questi metodi
 - *Software Architecture Analysis Method*
 - analizza un'architettura rispetto alla modificabilità – nelle sue varie forme, ad es., portabilità, estensibilità, integrabilità, ... – e alla copertura funzionale
- **ATAM**
 - *Architecture Tradeoff Analysis Method*
 - estensione di SAAM per valutare anche altre importanti qualità – prestazioni, sicurezza, disponibilità, ...



ATAM

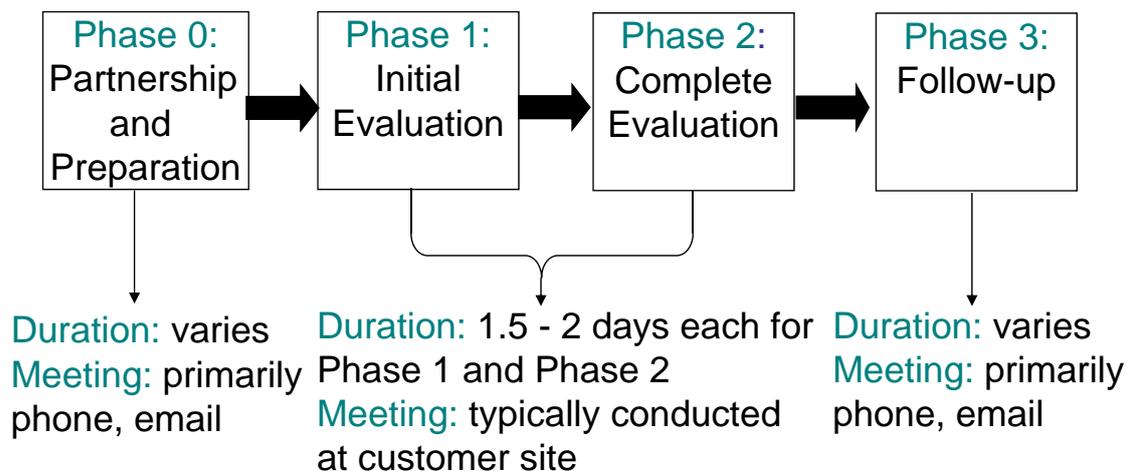
□ ATAM

- *Architecture Tradeoff Analysis Method*
- lo scopo di ATAM è determinare le conseguenze di decisioni architettoniche alternative, alla luce degli attributi di qualità
- più precisamente, l'obiettivo di ATAM è comprendere la capacità di un'architettura software rispetto al controllo di un insieme di più attributi di qualità, solitamente contrastanti
- ATAM riconosce la necessità di un compromesso (trade-off) tra i diversi attributi di qualità
 - questo compromesso va identificato durante la definizione dell'architettura del sistema – e non durante la sua costruzione



Fasi di ATAM

□ Valutazione condotta in quattro fasi



- particolarmente significative le fasi 1 e 2 – ausiliarie le fasi 0 e 3
- nota: una variante “leggera” (ma meno efficace) di ATAM – Lightweight Architecture Evaluation – può essere svolta in circa 4-6 ore



Il processo ATAM

- Le fasi 1 e 2 di ATAM sono relative a due aspetti importanti ma distinti nella valutazione dell'architettura
 - *valutazione centrata sull'architettura* – fase 1
 - eseguita dalle parti interessate fondamentali – chi crea l'architettura, l'acquirente, gli utenti principali
 - focalizzata sulla comprensione dell'architettura e delle decisioni prese
 - *valutazione centrata sulle parti interessate* – fase 2
 - coinvolge rappresentanti dalla comunità più ampia di tutte le parti interessate
 - verifica più ampia dell'architettura



Il processo ATAM

1. Present the ATAM
2. Present business drivers
3. Present architecture
4. Identify architectural approaches
5. Generate quality attribute utility tree
6. Analyze architectural approaches

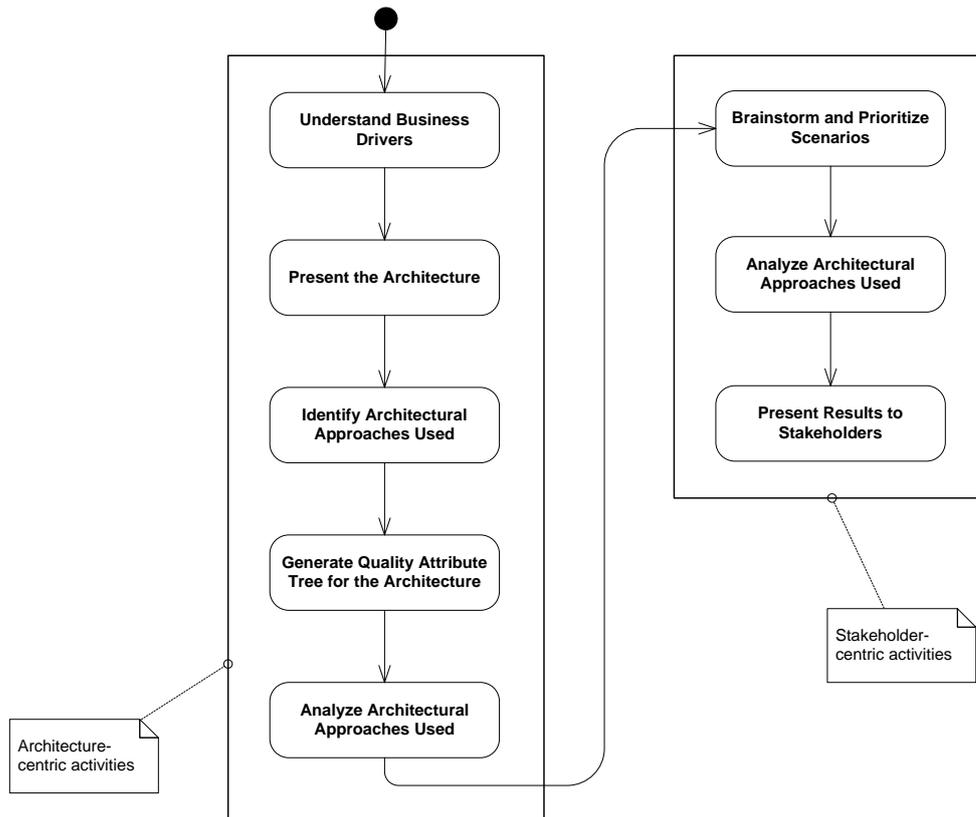
Phase 1

7. Brainstorm and prioritize scenarios
8. Analyze architectural approaches
9. Present results

Phase 2



Il processo ATAM [SSA]



25

Validazione dell'architettura

Luca Cabibbo - ASw



Passi di ATAM (1-2)

- 1) Presentazione di ATAM
 - passi, obiettivi e tecniche di ATAM
- 2) Presentazione dei business driver – vedi [SSA, Ch. 8]
 - un **business goal** è uno scopo/aspirazione specifico dell'organizzazione
 - ad es., il 15% delle proprie vendite realizzate online
 - un **business driver** è una qualche forza che agisce sull'organizzazione che richiede che essa si comporti in un certo modo per proteggere/far crescere il suo business
 - ad es., bisogno competitivo di entrare nel mercato online, anticipando il comportamento futuro degli acquirenti
 - forniscono il contesto per il sistema in discussione
 - utile per confermare (all'architetto) la comprensione dei business driver – possibili nuove intuizioni

26

Validazione dell'architettura

Luca Cabibbo - ASw



Passi di ATAM (3-4)

- 3) Presentazione dell'architettura
 - l'architetto presenta l'architettura – a un livello di dettaglio “appropriato” – a secondo dell'uditorio, del livello di avanzamento, del tempo a disposizione, ...
 - enfasi su comunicazione e comprensione – non analisi e valutazione
 - presentazione basata sulle viste dell'AD

- 4) Identificazione degli approcci architettureali usati
 - un *approccio architettureale* è una decisione architettureale di alto livello – di solito catturata da uno o più stili architettureali principali che caratterizzano l'architettura – per garantire le qualità più importanti del sistema
 - enfasi su comunicazione e comprensione

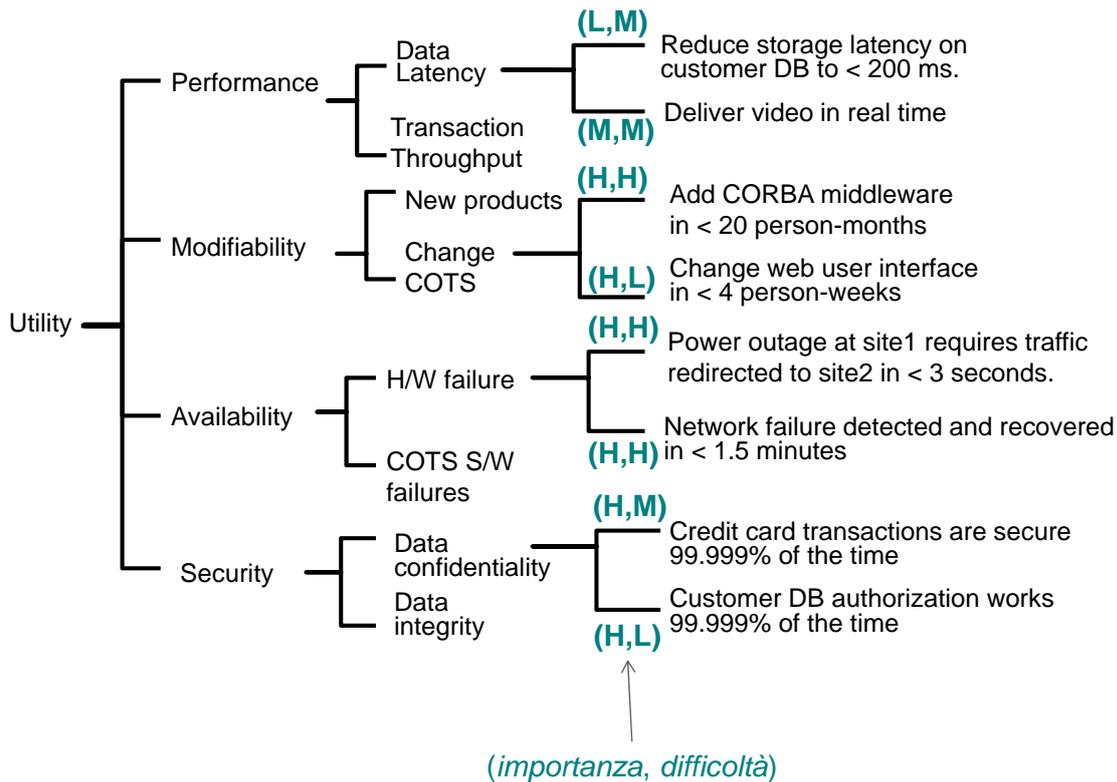


Passi di ATAM (5)

- 5) Generazione di un albero di utilità degli attributi di qualità
 - definizione delle qualità critiche per il sistema e identificazione degli scenari che le caratterizzano
 - si parte dalle qualità di alto livello richieste
 - ciascuna qualità viene suddivisa in aree importanti
 - ciascuna area, considera gli scenari specifici che la illustrano
 - assegna un “voto” a ciascuno scenario, per importanza (per il successo del sistema) e difficoltà (per il raggiungimento dello scenario)
 - il risultato è un albero degli attributi di qualità che fornisce un insieme di scenari per valutare l'architettura
 - albero di *utilità* – la radice indica la bontà complessiva del sistema
 - obbliga le parti interessate a considerare, esplicitare e dare priorità alle qualità richieste del sistema



Esempio - albero di utilità



Passi di ATAM (6)

- 6) Analisi degli approcci architetturali usati
 - analisi volta a comprendere in modo approfondito come il sistema sostiene (oppure se non sostiene) ciascuno scenario di qualità
 - analisi sistematica di ciascuno scenario – per capire *come* l'architettura sostiene lo scenario e *quali* decisioni architetturali sono importanti per raggiungerlo
 - possibili diverse tecniche di validazione
 - importante identificare le decisioni/alternative critiche e i rischi



Punti di rischio

- Importante identificare le decisioni/alternative critiche e i rischi

- *risk*

- decisione che potrebbe creare problemi futuri in qualche attributo di qualità
 - comprendono decisioni importanti che non sono state prese, oppure decisioni prese ma le cui conseguenze non sono state completamente comprese
 - ad es., non è chiaro come scrivere moduli di logica applicativa sull'application server (nota: decisione non presa) – ci potrebbero essere ridondanze e replicazioni di codice, riducendo la modificabilità del sistema



Punti di non rischio

- Importante identificare le decisioni/alternative critiche e i rischi

- *non risk*

- decisione che promuove qualità che aiutano a raggiungere gli obiettivi di business – senza sollevare rischi per l'architettura
 - ad es., nell'ipotesi corrente in cui si attende mediamente un messaggio al secondo, l'elaborazione di un messaggio richiede circa 30ms ed è svolta in un processo ad alta priorità, una deadline soft di 1sec sembra ragionevole



Punti di sensibilità

- Per identificare i rischi, è necessario comprendere quali sono le decisioni che hanno effetto sul controllo degli attributi di qualità
 - **sensitivity point**
 - proprietà di uno o più componenti (o della relazione tra di essi) o decisione di progetto che è critica per raggiungere una certa qualità
 - qualche piccolo cambiamento (negli scenari o nei loro parametri) potrebbero causare differenze significative nella possibilità di controllare quell'attributo di qualità
 - pertanto indica un possibile rischio
 - ad es., il tempo di elaborazione di un messaggio importante potrebbe essere sensibile alla priorità del processo a priorità minima tra quelli coinvolti nell'elaborazione del messaggio
 - ad es., il livello di confidenzialità delle informazioni scambiate in rete potrebbe essere sensibile al numero di bit della chiave usata per la cifratura



Punti di compromesso

- Per identificare i rischi, è necessario comprendere quali sono le decisioni che hanno effetto sul controllo degli attributi di qualità
 - **tradeoff point**
 - proprietà o decisione di progetto che ha effetto su più attributi di qualità, ed è un punto di sensibilità per almeno un attributo di qualità
 - ovvero, una decisione di progetto che richiede un compromesso tra più qualità
 - si tratta delle decisioni di progetto più critiche – bisogna identificarle e prendere tali decisioni in modo opportuno
 - ad es., il numero di bit della chiave usata per la cifratura ha impatto sia sul livello di confidenzialità che sulle prestazioni
 - ad es., per raggiungere i livelli richiesti di prestazioni sono state usate delle librerie specifiche per la piattaforma hardware di riferimento – ma il componente risultante non è più portabile

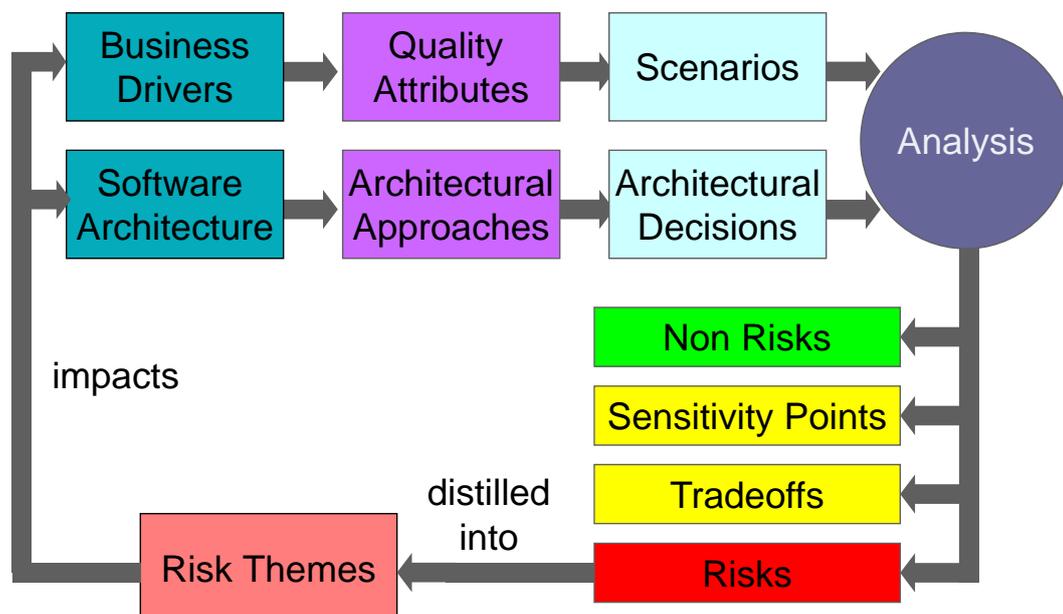


Obiettivo di ATAM

- Scopo di ATAM è determinare le conseguenze di decisioni architeturali alternative, alla luce degli attributi di qualità
 - più precisamente, l'obiettivo di ATAM è comprendere la capacità di un'architettura software rispetto al controllo di un insieme di più attributi di qualità, solitamente contrastanti
 - scopo non è effettuare un'analisi precisa – piuttosto, scoprire i rischi creati da decisioni architeturali
 - correlazioni tra decisioni architeturali e previsioni circa le proprietà/qualità del sistema
 - identificazione e documentazione esplicita dei compromessi
 - a valle, possibile/necessario intraprendere altre attività per la mitigazione dei rischi identificati – altra analisi, progettazione, prototipazione, ...



Flusso concettuale di ATAM





Dalla fase 1 alla fase 2

- La prima metà del processo di valutazione pone le basi per una migliore comprensione dell'architettura, e comprende anche una valutazione con le principali parti interessate
 - la valutazione prosegue verificando che l'architettura soddisfa le priorità di tutte le parti interessate



Passi di ATAM (7)

- 7) Brainstorming e assegnazione di priorità agli scenari
 - insieme alle parti interessate, viene creato un elenco di scenari *importanti per le parti interessate*, nel presente e nel futuro
 - compresi scenari di casi d'uso, di cambiamento (ad es., di funzionalità, carico e prestazioni) ed esploratori (casi estremi di cambiamento)
 - comunemente un'estensione degli scenari identificati nel passo 5
 - le parti interessate assegnano una priorità a tali scenari
 - gli scenari vengono inseriti nell'albero di utilità



Passi di ATAM (8-9)

- 8) Analisi degli approcci architeturali usati
 - analogo al passo 6 – ma anche alla luce dei nuovi scenari
 - auspicabilmente conferma la bontà dell'architettura – ma può portare all'identificazione di problemi e deficienze

- 9) Presentazione dei risultati alle parti interessate
 - per far comprendere alle parti interessate come l'architettura sostiene i loro scenari
 - illustrazione dei punti di sensibilità e dei compromessi
 - per far comprendere che un cambiamento su certi requisiti potrà o non potrà essere soddisfatto



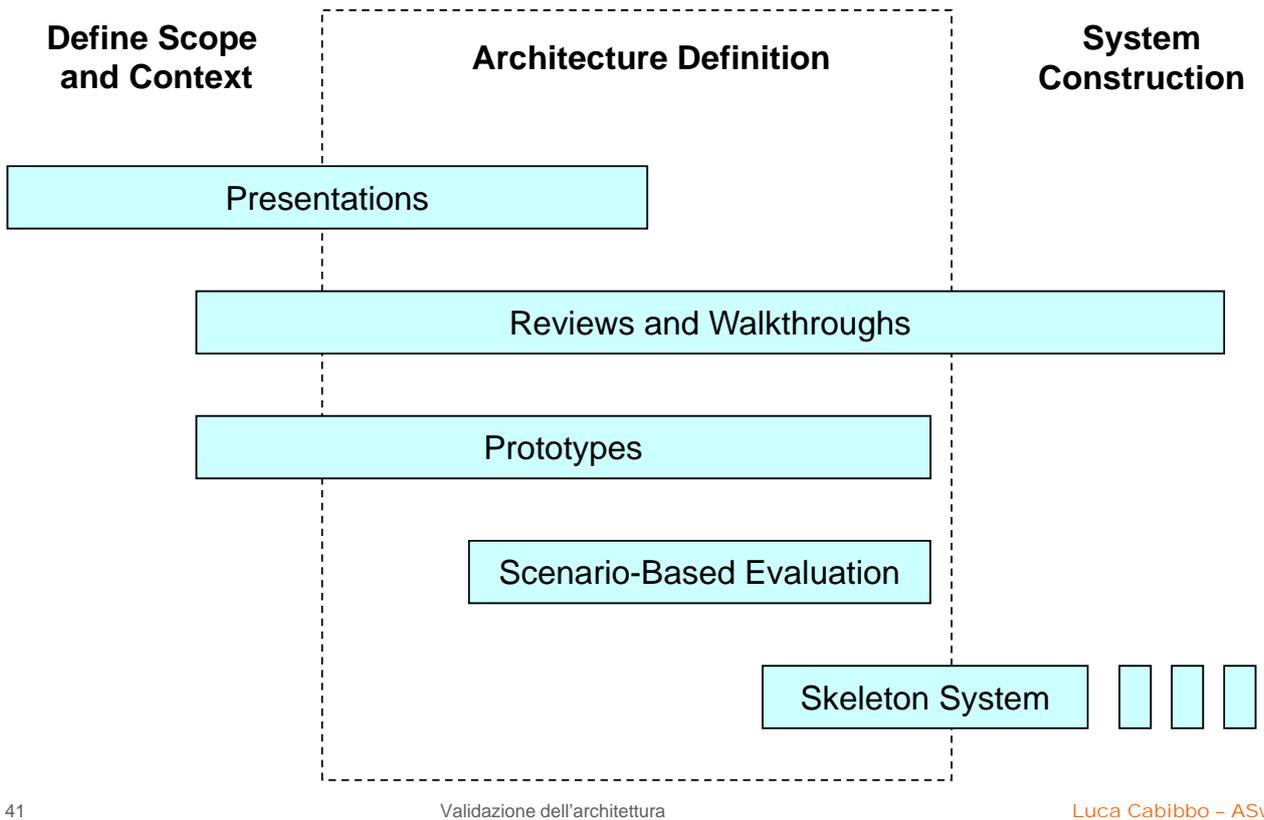
Quando applicare ATAM

- In teoria, il momento migliore per applicare ATAM è
 - subito dopo che l'architettura è stata definita/specificata
 - quando non c'è codice o ce n'è ancora poco

- In pratica, ATAM può essere utilizzato in modo efficace anche nelle seguenti situazioni
 - valutazione tra più architetture candidate alternative
 - valutazione di sistemi esistenti (ricostruzione dell'architettura), prima di una loro evoluzione significativa
 - per decidere se “aggiornare” o “sostituire”



* Validazione e processo di sviluppo



41

Validazione dell'architettura

Luca Cabibbo - ASw



* Risultati di una validazione

- Importante registrare i risultati delle validazioni – per evitare incomprensioni sulle decisioni prese
 - resoconti o verbali delle riunioni
 - discussioni e risultati
 - log delle decisioni
 - revisioni
 - rapporti di valutazione
 - firma formale di documenti
 - ad es., per accettazione dell'architettura

42

Validazione dell'architettura

Luca Cabibbo - ASw