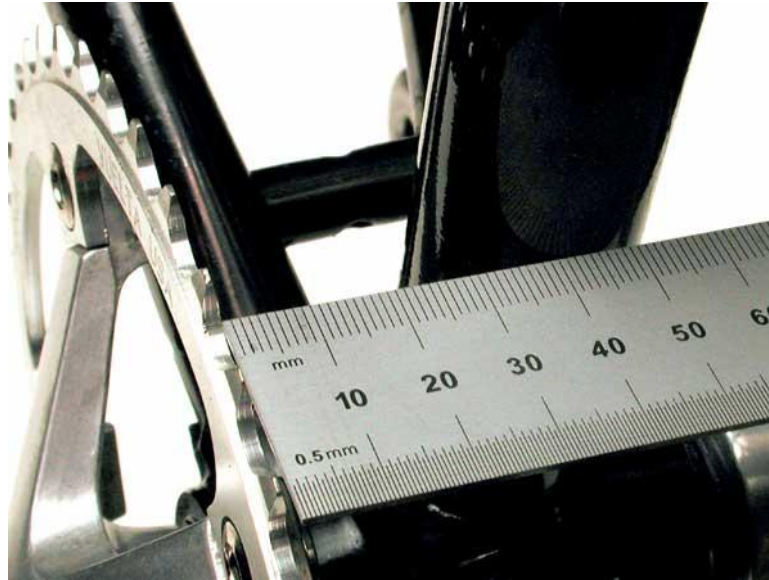


# Architecture Assessment

# Architecture can't be measured



The very purpose of a blueprint is to provide a tangible artifact that can be used to visualize, specify, construct, document - and reason about - a system

# Why Architectural Analysis?

- The earlier you find a problem in a software project, the better off you are.
- An unsuitable architecture will bring disaster on a project.
- Architecture evaluation is a cheap way to avoid disaster.



# Architecture Evaluation



Wouldn't it be nice to know in advance if you've placed your bet on a winner, as opposed to waiting until the system is mostly completed?

# Architecture Evaluation



The cost to fix an error found during requirements or early design phases is orders of magnitudes less to correct than the same error found during testing.

# Architecture Evaluation



An unsuitable architecture will precipitate disaster on a project. Schedules and budgets will be blown out of the water as the team scrambles to back-fit and hack their way through the problems.

# Architecture Evaluation



*An architecture evaluation doesn't tell you "yes" or "no," "good" or "bad," or "6.75 out of 10."*

An architectural evaluation will tell you that the architecture has been found suitable with respect to one set of quality attributes and problematic with respect to another set of quality attributes .

# Architecture Evaluation



You can bet that no architecture will evaluate better than all others in all areas. Instead, one will outperform others in some areas but underperform in other areas.



# Architecture Assessment



The evaluation will first identify what the areas of interest are and then highlight the strengths and weaknesses of each architecture in those areas.

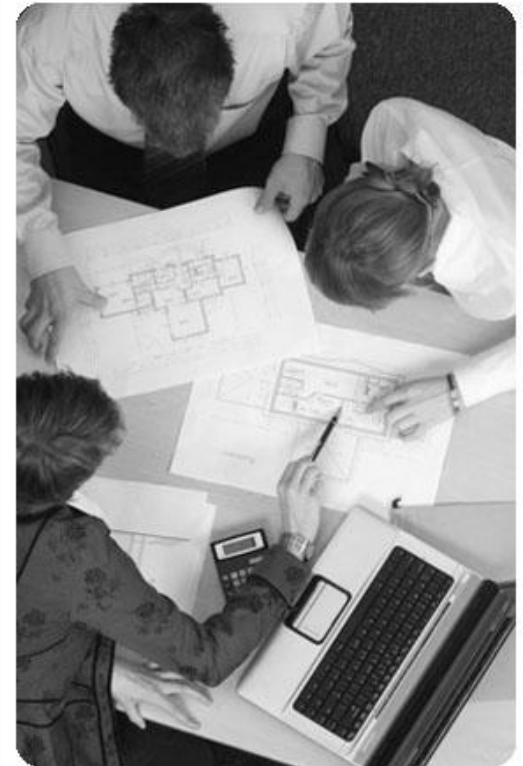
# Architecture Evaluation



Evaluation need not wait until an architecture is fully specified. It can be used at any stage in the architecture creation process.

# Assessment techniques

- ADR
  - Active Design Review
- ARID
  - Active Reviews for Intermediate Designs
- ATAM
  - Architecture Tradeoff Analysis Method
- CBAM
  - Cost Benefit Analysis Method
- SAAM
  - Software Architecture Analysis Method



Method	Role	Input	Output	Description
<b>ADR</b>	Technical reviewer	Design Documents	Review Documents	Evaluate Partial Architecture. (Module)
<b>ARID</b>	Technical reviewer	Design Documents	Review Documents	Evaluate Partial Architecture. (Module)
<b>ATAM</b>	Software architect	Output of QAW	Sensitive Points Tradeoffs Risk/Non Risk	Evaluate Architecture w.r.t . Quality Attributes.
<b>CBAM</b>	Software architect	Output of ATAM	Cost of Arch. Benefit of Arch.	Calculate Cost v/s Benefit of Architectural Strategies.
<b>SAAM</b>	Software architect	Output of QAW	Scenario weight w.r.t. each Arch.	Compare two Architectural Approaches



# Architecture Assessment Types

## Qualitative assessment:

- *A is better suited than B*
- Can be done with very little data

## Quantitative assessment

- *A can handle about 2500 transactions/s, B can handle 500 transactions/s.*
- Requires more data than qualitative assessment

## Assessment of **theoretical min/max**

- *A can handle about 2500 transactions/s and scale to 3000, B can handle 500 transactions/s and scale to 750*
- Requires the more data of all assessment types