

# アーキテクチャ検証手法ATAMの適用可能性評価

山下裕介

## 開発における問題点

システムアーキテクチャを開発初期に検証することはトラブル抑止のために非常に重要である。現在私はアーキテクチャ検証の独自手法を開発し検証を続けているが、アーキテクチャが非機能要求に与えるトレードオフの影響について、分析する手法が開発できていなかった。  
(トレードオフ例:暗号化はセキュリティ要求を満たすが、性能要求には否定的に作用する)

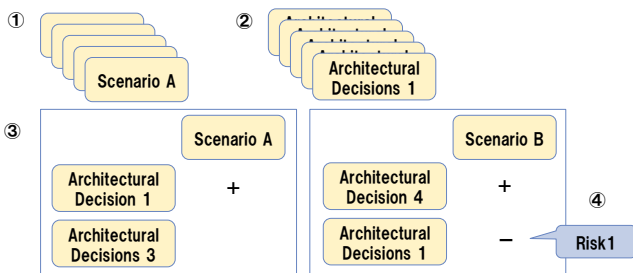
## 手法の適用による解決

アーキテクチャと品質要求とのトレードオフ関係を分析するための手法である“ATAM”を実際のシステムに適用し、ATAMの適用可能性を評価した。  
※ATAM: Architecture Tradeoff Analysis Method

## ATAMの適用

### ■ATAMコンセプト

- ① 非機能要求をユーティリティツリーで洗い出す
- ② アーキテクチャ上の決定を洗い出す
- ③ 両者の関係性を定義する
- ④ リスクの有無を判断する



### ■ATAM適用課題

No	適用課題	ATAM	対応
1	前提参加者が集められない	アーキテクト、PM(顧客)、評価者およびシステム資料で評価	評価者・システム資料で評価
2	情報不足の場合の対応手順が無い	インプット情報規定無し(情報不足時は評価参加者から情報収集)	不足情報は指摘

### ■ATAM適用結果

- ・あるシステムに関するリスクを9件検出  
非機能定義不足:4件、アーキテクチャ検討不足:3件  
トレードオフ部分:1件、その他:1件
- ・課題としていたトレードオフ部分のリスクを検出
- ・検討不足が多いプロジェクトであった。今回の指摘により情報が具体化された後、再度トレードオフ分析が必要

## 評価

### ○利点

ATAMはアーキテクチャが非機能要求に対して否定的に影響を与える部分を検出する手順が提供されている。この手順については、実際のシステムでも十分適用可能であった。

### ○考慮が必要な点

No	考慮ポイント	内容
1	十分なインプット情報が前提	トレードオフ分析であるため、当然対象とする非機能要求とアーキテクチャの情報が必要。
2	アーキテクチャの与える影響判断はノウハウが必要	アーキテクチャが非機能要求に与える影響判断にはアーキテクトの知見が必要。ATAMではその判断に対する具体的な手順やノウハウは提供されていない。
3	一般に非機能要求はシナリオで定義されない	非機能要求は一般に数値で定義されることが多い。ATAMの手順に従うには、非機能要求の定義をすべてシナリオに変換し、さらに優先度・実現性を定義するタスクが必要。

## 今後の課題

### ○アーキテクトノウハウの可視化・蓄積

アーキテクチャと非機能要求の関係性の定義は現状はアーキテクトのノウハウが必要である。

過去のアーキテクト指摘の蓄積を可視化する、一般に知られている非機能要求のトレードオフパターンを抽出するなどの対応で属人性の排除が可能となる。

※トレードオフパターン例: 性能とセキュリティ、可用性と運用性、など。

### ○評価インプット情報の定義

トレードオフを分析するには非機能要求とアーキテクチャの情報が十分揃っている必要がある。そのために、非機能要求は非機能要求グレードなどの標準を活用して定義、アーキテクチャは構成する要素を静的・動的側面などから分解して定義しておくことが重要である。