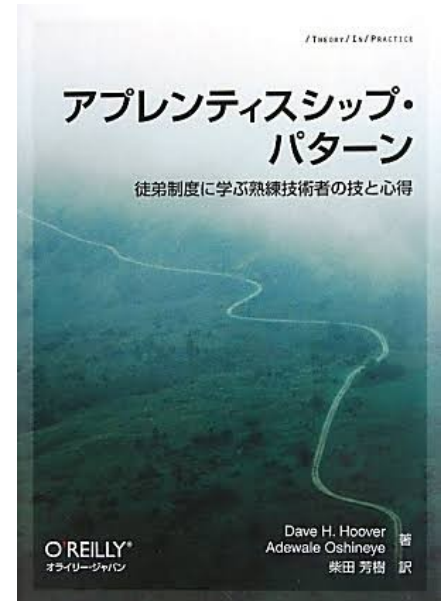


JaSST Review'20

ソフトウェア設計における 意思決定とそのレビューの秘訣

川島 義隆
@kawasima

Software development is
an ignorance-reduction activity



Architectureとは

- Martin Fowler 「重要かつ変更が困難な意思決定である」
- Grady Booch 「システムを形作る重要な設計上の意思決定であり、その重要性は変更のコストによって測られる」
- Michael Keeling 「ソフトウェアが望ましい品質特性や他の性質をどうやって達成していくかについての重要な設計上の意思決定である」 Design It!

Architecturally Significant Requirements (ASR)

Architectural Significant Requirements

構造

アーキテクチャのパターンやスタイルに影響を与える決定

例) マイクロサービス間のデータの共有の仕方

機能特性

技術の選択がパフォーマンスに影響を与え、パフォーマンスがアプリケーションの重要な側面である場合

例) APIのプロトコル RESTful? GraphQL? gRPC?

依存関係

システム内のコンポーネントやサービス間の統合点。システム全体のスケーラビリティ、モジュラリティ、アジリティ、テスト容易性、信頼性に影響を与える。

例) サービスをどう分割するか?

インタフェース

サービスやコンポーネントがどのようにアクセスされ、どのようにオーケストレーションされるか。

例) APIをコールするか? EventをPub/Subするか?

「Fundamentals of Software Architecture」より

アーキテクチャ



設計上の重要な意思決定

アーキテクチャ設計の記録



意思決定の記録

アーキテクチャ設計の記録のレビュー



意思決定の記録のレビュー

Architecture Decision Record(ADR)

アーキテクチャ上の意思決定を記録するフォーマット

Title: タイトル

Status: 意思決定の状態

Context: 決定を迫られている状況

Decision: 代替案と選択した決定事項

Consequence: 予想される結果/実際の結果

ADRの目的

こういう事態を防ぎたい

- プロジェクト新規参画者が、決定の背景を知らずに「古臭い」などを理由に、既存の仕組みを置き換えてしまう。
- 過去の意思決定した状況とは大きく変わってしまっているのに、見直しがされない

ADRの例

システムAの更新データをシステムBに同期させる方式

ステータス: Accepted

コンテキスト

- システムAでの更新は、Bでも参照するので、ほぼリアルタイムで送信したい。
- B→Aの同期に関しては、今のところ存在しないので考えなくて良い。

決定

- 案1: Kafkaでメッセージを同期させる
 - [Cons] 設定を適切にできる要員がない。
- 案2: AWS SQSを使い、メッセージを同期させる
 - [Pros] マネージドなサービスなので、構築が簡単
 - [Cons] メッセージのサイズ上限がある。
- 案3: システムBにAPIの口を作って呼ぶ
 - [Cons] システムB側に対応してもらう必要がある。
 - [Cons] システムBがダウンしていると同期ができない。

特にコンテキストをしっかりと書いていると
第三者レビューがやりやすい

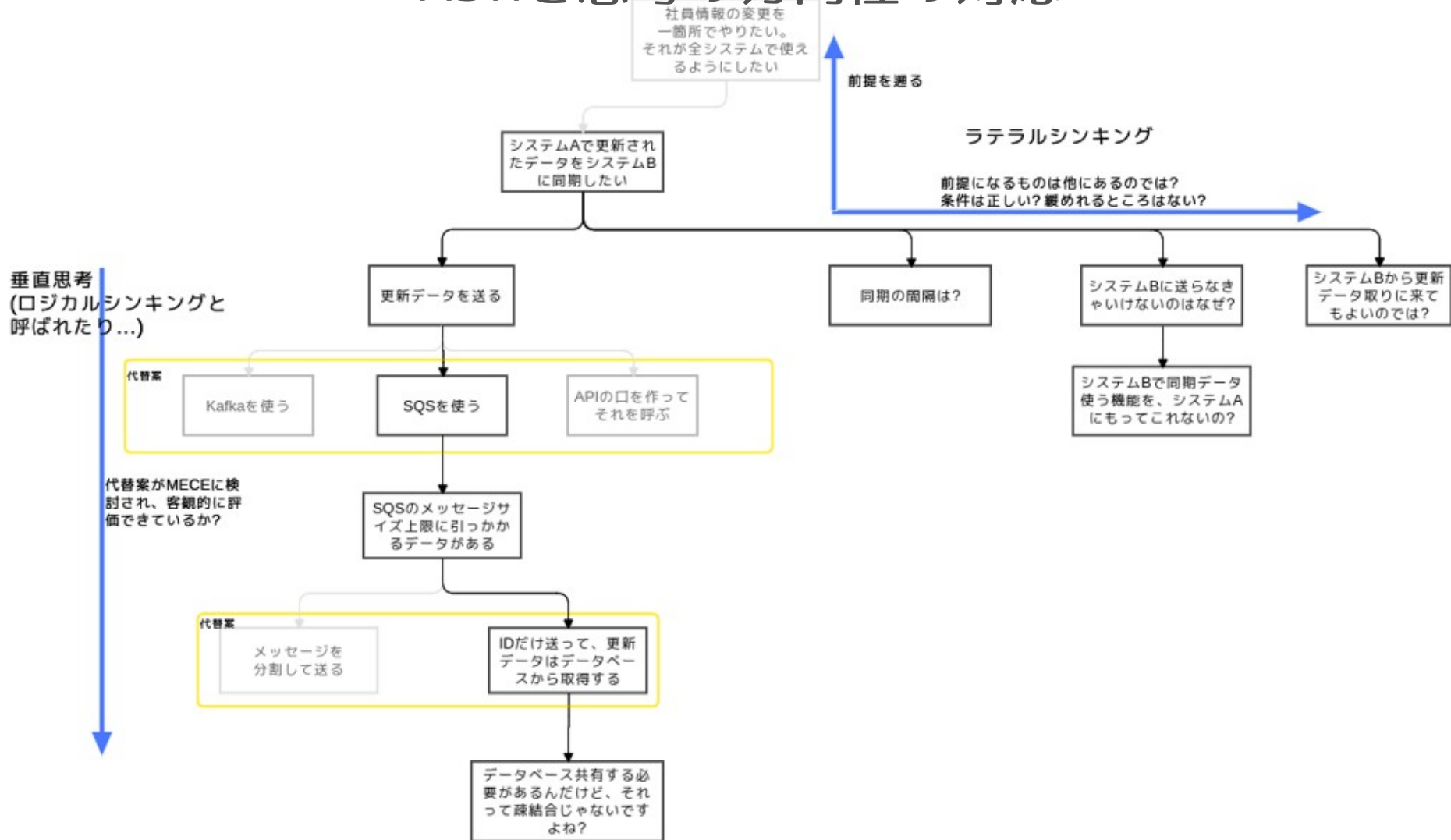
Architecture Decisionの主要レビューポイント

- **コンテキスト**に書かれた前提や制約が確かなものであるか？
- 代替案が論理的に検討され、公平な評価をもって**デシジョン**が導き出されているか？

垂直思考と水平思考

- 垂直思考
 - いわゆるロジカルシンキング
 - 代替案をMECEに検討し、思考を掘り下げる
- 水平思考
 - 思考の幅をひろげ、本質を考える
 - 「抽象化」と「常識を疑うこと」が鍵

ADRと思考の方向性の対応



デシジョンにおける代替案の検討 ⇒ 垂直思考

コンテキストの抽象化と多角的検討 ⇒ 水平思考

垂直思考: デシジョン

- 評価軸を定め、そのバリエーションにより取りうる代替案を洗い出す
- 代替案それぞれについて、各評価軸について Pros/Cons を公正な目で記述する。
- さらに細かい決定が必要ならば、ドリルダウンした ADR を作る

水平思考:コンテキスト

- 問題を理解する
 - 明らかなファクトは何か?
 - 条件は何か? 各条件が直交しているか?
 - 制約は何か? 時間/お金/人など観点が足りている?
- これらを書き下し「解決すべき状況」として記述する

水平思考の肝：抽象化

- コンテキストは「解決すべき問題」が記述されているか？ 手段が書かれていないか？
- 類似の問題をかき集め、共通点を見いだす
- 近い将来起こることが予想される、追加される機能や環境の変化を含めて考えると…

水平思考の肝：疑いの目

- 前提条件/制約を疑う、緩和する
 - 「〇〇さんが言ってるから」みたいな根拠は覆りやすい
 - この条件/制約を緩和できれば、代替案検討に大きなインパクトがあるのに…
 - 時間経過による制約の変化はあるか？

視点・視野・視座と意思決定構造の関連性

視点

どこに着目するか

影響する箇所

代替案の評価・選択

左右する要素

センス
(という名の結局は論理性)

視野

見えている範囲

代替案をどれだけ出せるか

経験

視座

見ているものの抽象度

コンテキストを適切に捉えているか?

職位

ADRレビュー虎の巻

「枝葉末節」に思える問題設定

- 症状
 - 解決したい課題がとてもSpecificで、代替案の良し悪しを評価しづらい
- 原因
 - 「視点」のズレ
- 対策
 - 対話しながら、ロジックツリーを逆向きに再構成する。
 - もともとの目的を導き出す。

例) <https://jflute.hatenadiary.jp/entry/20170611/askingway1>

似通った代替案

- 症状
 - 代替案にあまり大きな差異がなく、どれ選んでも好みの問題に思える
- 原因
 - 視野が狭い
 - コンテキストの不足
- 対策
 - 対話により代替案の評価軸を加えてやる

不公平な代替案

- 兆候
 - 特定の代替案だけ、Pros/Consがたくさん書いてある。
- 原因
 - 暗に結論ありきで代替案の評価を書いている。
 - 勘の鋭い人が陥りがち
- 対策
 - 評価軸を明確にし、各案に対してPros/Consを公平に列挙する。

「先日のあれ、松・竹・梅の3案考えてきました」

「竹案だけやたら深堀られてない？（極端回避性を使っているな…!）」

Specific過ぎるコンテキスト

- 症状
 - 達成したい姿とADRの内容に乖離がある感じがする
- 原因
 - 抽象化の不足（手段の目的化）
- 対策
 - コンテキストの抽象レベルを一段階あげる
 - 場合によっては上位組織にエスカレーション

とある旅行系スタートアップ

「リテンション強化のために会員機能を作りたい」

- 会員機能を作るとこれらの機能が必要になる
 - 会員登録
 - パスワード変更
 - パスワード再発行
 - ID(メールアドレス変更)
 - メールアドレスが上手くできなかったときの問合せ窓口
- 非会員でも同じ機能が使える場合、ログインしたらそれを引き継ぎたい。

本当に達成したいのは、リテンション強化であり会員機能ではない。

会員機能そのものがリテンションに効くわけではない。

その時点で考えられていた機能は、「物件のお気に入り登録」のみ

- 非会員ではクッキーに保存したIDで、会員はメールアドレスをキーに、お気に入り进行管理する
- 「会員」という言葉は使わない。
- メールアドレスに結びつけたユーザは、メールアドレスを入力すると、ログインリンクがメールで送られ、それをクリックすると、お気に入りが見える。
- 「会員」ではないので、メールアドレス変更などには特に対応しない。

八方塞がり

- 症状
 - どの代替案も決め手を欠く
- 原因
 - 前提条件・制約が厳しすぎる/現状に寄り添い過ぎている
- 対策
 - コンテキストのうち、緩める/諦めることができるものがないかを議論する
 - 場合によっては上位組織にエスカレーション

広告掲載サイト

クライアント毎に購入した枠数分(月次)だけ、記事掲載できます。

- 毎月、翌月分の枠数だけ載せる記事をクライアントに入稿してもらいます。
- 記事は当月分のものと同じものを流用して入稿することも可能です。
- 記事は掲載前に内容の審査を行うため、3営業日前までにクライアントに入稿してもらう必要があります。
- 入稿された記事は、カスタマ向けサイトに内容をコピーすることで表示・検索されるようになります。



次月の掲載枠購入

記事作成



記事審査

掲載処理

一見シンプルだが、掲載処理に回した後のキャンセルや、枠数に満たない場合の催促など、異例処理がたくさん潜んでいる。

キャンペーン用原稿を
先に入稿したい



購入枠の買い増し

次月の掲載枠購入

入稿数不足分の督促



記事作成

記事審査

掲載処理確定後の差し替え

掲載処理

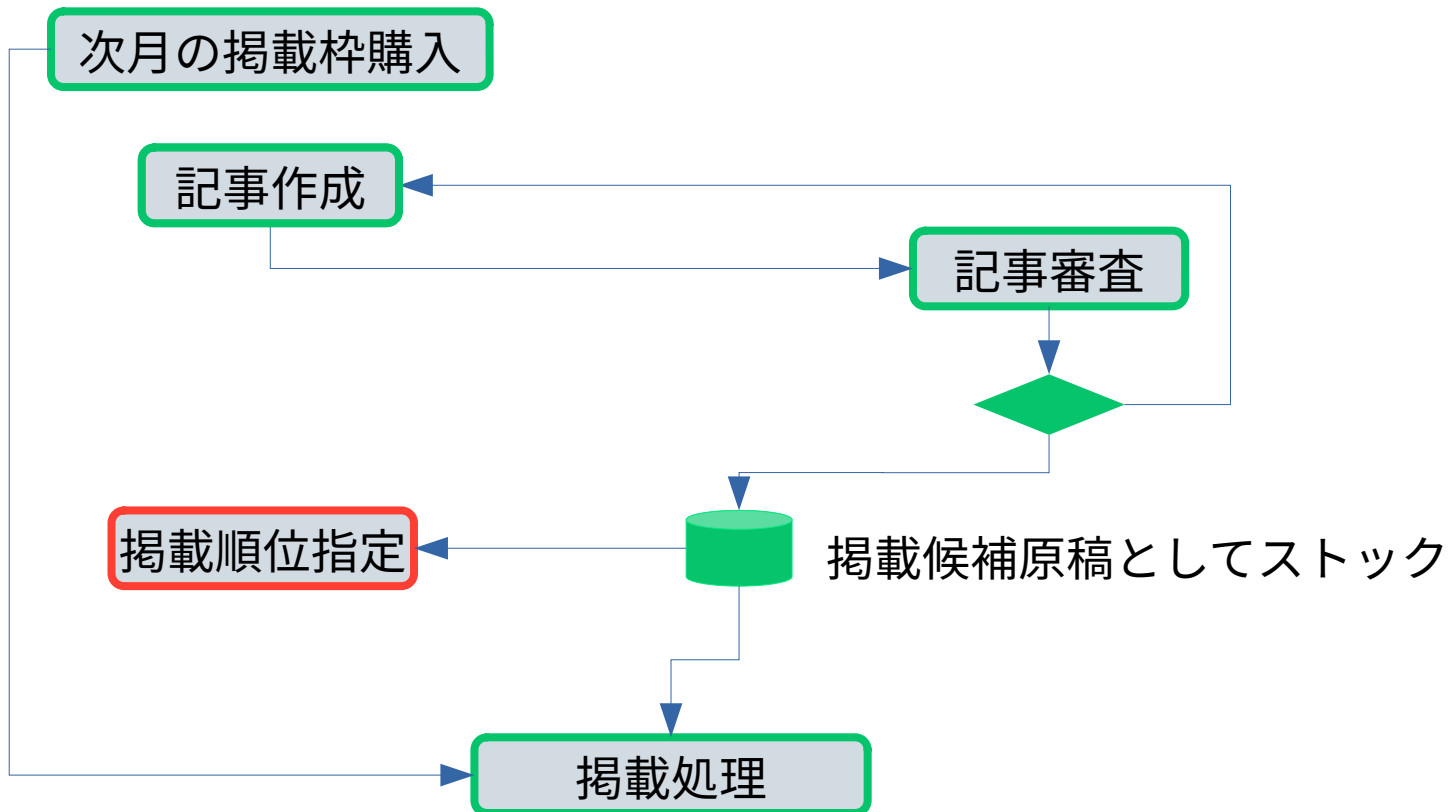
そもそも解決したい課題は何か？



クライアントが掲載レディなものを、購入した枠
数分だけ勝手に掲載してほしい



掲載枠購入と記事作成を業務上、密に連携させる
必要はない。



≡ 秋の料理特集

≡ ヘルシーレシピ

≡ 残り物で作るスペシャルコース

≡ …

↑ 次月掲載予定ライン

抽象レベルの違いによる衝突

- 症状
 - 「そもそも～」の話に言及しすぎて、レビューイの顔色が曇る
- 原因
 - 抽象レベルの違い
 - 抽象度の低い方から、高い方は見えない
- 対策
 - 「垂直思考/水平思考」「視点・視野・視座」の構造を共有する
 - 「そもそも～」の話に悪意がないことを、相手に理解してもらう

緊急度に合わせて抽象レベルのチューニング

- 症状

- レビューイは答えを欲しがっているが、レビューアが前提その他が気になって、話が進まない。

- 原因

- 急ぎ「答え」が求められているときに、水平思考のレビューをしてしまう

- 対策

- まず意思決定の緊急度合いを確認し、決定が技術的負債を生むものであれば、暫定対処と恒久対処にADRを分割する

中身に入る前に場のセッティングを

いきなり「レビュー」を始めるのではなく、
「これこれこういう理由で、コンテキストを水
平思考で確認する」というのを先に宣言し、
参加者全員で合意してスタートしよう。

Wrap up

- 視点・視野・視座の足りないところを補える人をレビューアに選定しよう。
- 意思決定のレビューには、垂直思考だけでなく水平思考も重要である。
- だが、抽象化や疑いの目は、反感を招くので、感情ではなく形式的にそうやっているということを、事前に参加者全員で合意しよう。