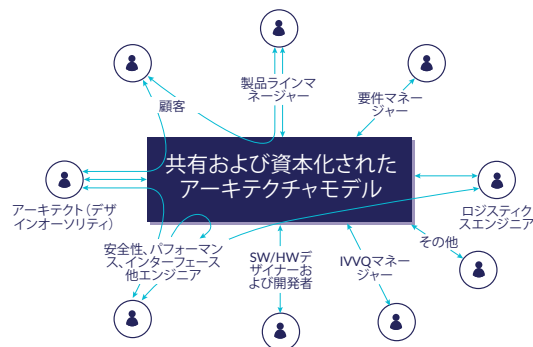
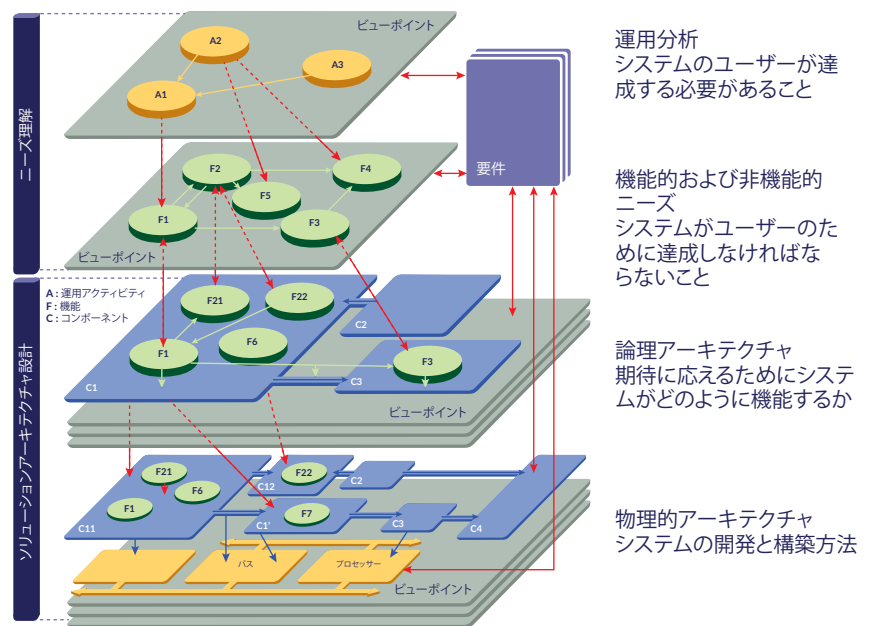


# システム、ソフトウェア、ハードウェアアーキテクチャを定義、分析、設計、検証するためのツール化された方法

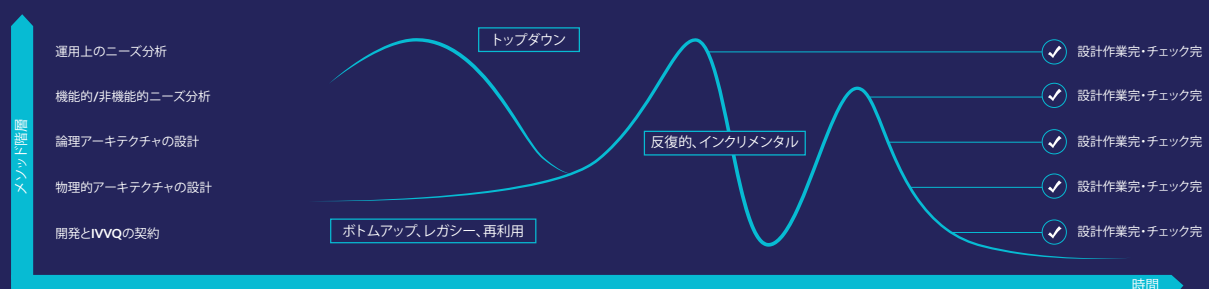
## エンジニアリングにおける効率的なコラボレーションのサポート



## 運用上のニーズに対するソリューションの検証/正当化により、影響分析が容易になります



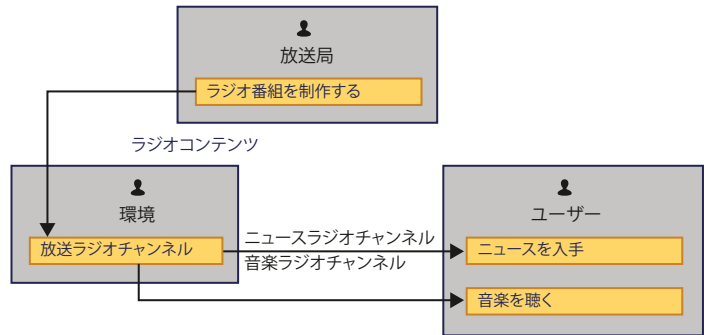
## ほとんどのプロセスと互換性があります トップダウン・ボトムアップ型、反復型、レガシーベース型、混合型等



## 顧客の運用ニーズ分析

システムのユーザーが達成する必要があること

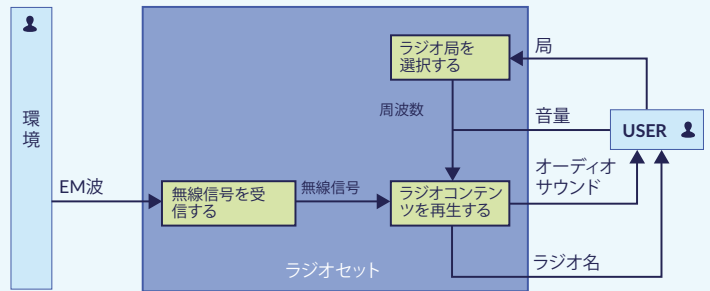
- ✓ 運用機能を定義する
- ✓ 運用上のニーズ分析を実行する



## システム/SW/HWのニーズ分析

システムがユーザーのために達成しなければならないこと

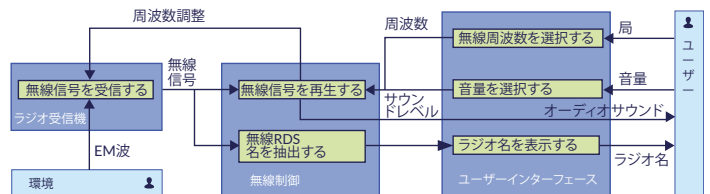
- ✓ 機能のトレードオフ分析を実行する
- ✓ 機能分析と非機能分析を実行する
- ✓ 要件を形式化して統合する



## 論理アーキテクチャ設計

期待に応えるためにシステムがどのように機能するか

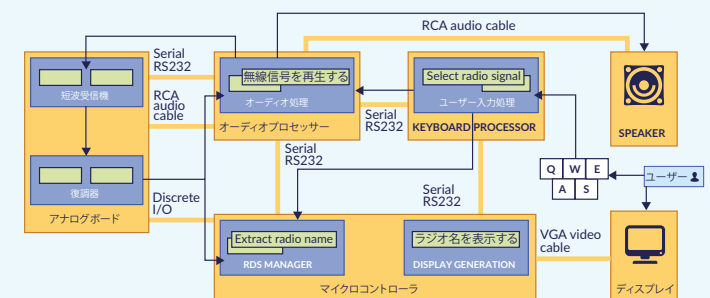
- ✓ アーキテクチャの推進要因と視点を定義する
- ✓ 候補となるアーキテクチャの構成をコンポーネントを用い作成する
- ✓ 妥協できる最適なアーキテクチャを選択する



## 物理アーキテクチャ設計

システムの開発と構築方法

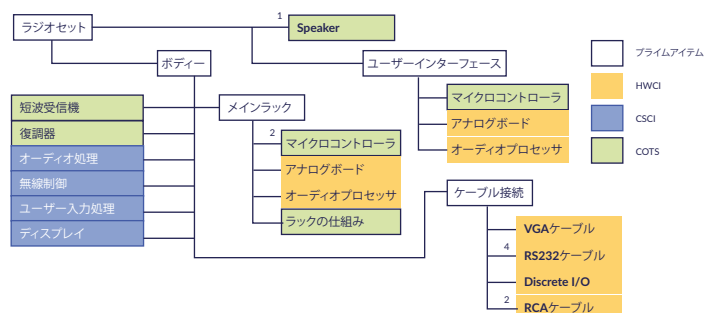
- ✓ アーキテクチャパターンを定義する
- ✓ 既存の資産の再利用を検討する
- ✓ 物理的な参照アーキテクチャを設計する
- ✓ 検証し確認する



## 開発契約

各デザイナー/開発委託先に期待すること

- ✓ コンポーネントのIWVQ戦略を定義する
- ✓ PBS(製品詳細構造)とコンポーネントの統合契約を定義・実施する



- 運用能力
- アクター、運用エンティティ
- アクターの活動
- 活動とアクターの間の相互作用
- 活動と相互作用で使用する情報
- 運用プロセスの連鎖活動
- 動的動作のシナリオ

- アクターとシステムとケイパビリティ/機能
- システムとアクターの機能
- 機能間のデータフロー交換
- データフローを渡す機能チェーン
- 機能と交換、データモデルで使用する情報
- 動的動作のシナリオ
- モードと状態

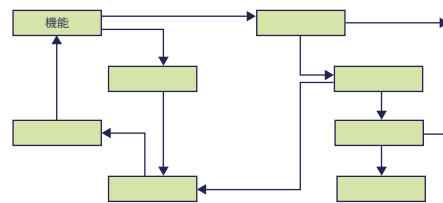
同じ概念に加えて:

- コンポーネント
- コンポーネントのポートとインターフェース
- コンポーネント間の交換
- コンポーネントへの機能割り当て
- 機能交換の割り当てによるコンポーネントインターフェースの正当化

同じ概念に加えて:

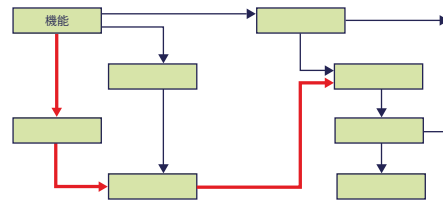
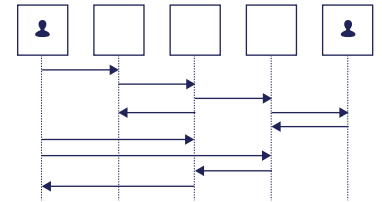
- 論理コンポーネントを詳細化し機能動作を実装する動作コンポーネント
- 動作コンポーネントのリソースを提供する実装コンポーネント
- 実装コンポーネント間の物理リンク

- 構成アイテムツリー
- 部品番号、数量
- 開発契約(予想される動作、インターフェース、シナリオ、リソース消費、非機能属性、など)

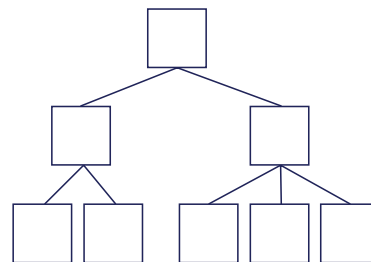


**データフロー:**機能、運用、活動の相互作用と交換

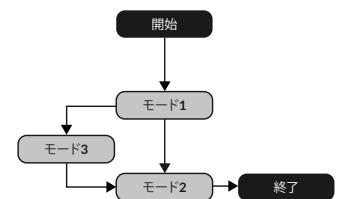
**シナリオ:**アクター、システム、コンポーネントの相互作用および交換



機能および運用活動による機能チェーンと運用プロセス

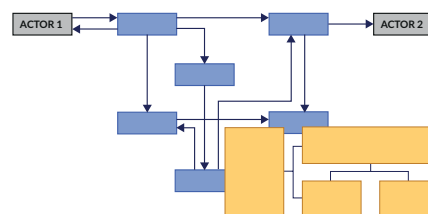
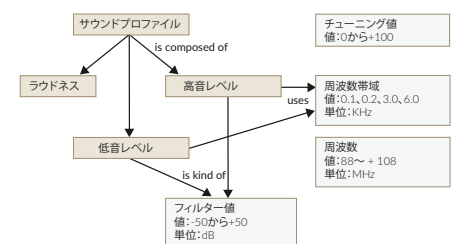


機能とコンポーネントの詳細化



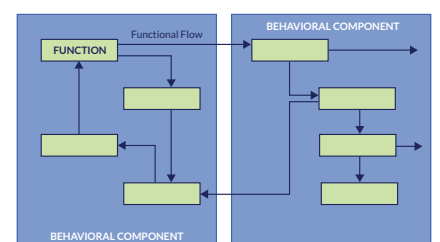
アクター、システム、コンポーネントのモードと状態

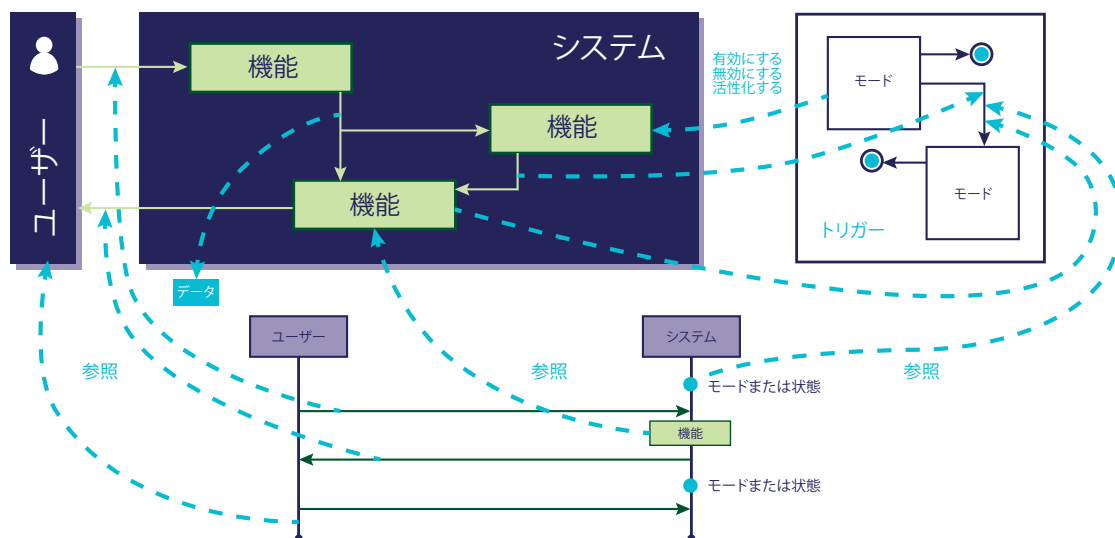
**データモデル:**データフローとシナリオの内容、インターフェースの定義と正当化



**コンポーネントの配線:**すべての種類のコンポーネント

**割り当て:**アクターへの運用活動、コンポーネントへの関数、実装コンポーネントへの動作コンポーネント、インターフェースへのデータフロー、構成アイテムへの関連要素





## 非機能的および産業的利害関係に対するソリューションの検証とチェック

| 方法階層         | パフォーマンスに特化したデータサンプル               | 安全性に特有のデータサンプル        |
|--------------|-----------------------------------|-----------------------|
| 運用上のニーズ分析    | 脅威に対する最長反応時間                      | 恐れられる出来事              |
| 機能/非機能的ニーズ分析 | 脅威に反応する機能チェーン(FC)<br>FCでの最大許容待ち時間 | イベントに関連する重要な機能チェーン    |
| 論理アーキテクチャ設計  | 複雑さの処理と交換<br>機能チェーンの割り当て          | 機能チェーンを保護する冗長パス       |
| 物理アーキテクチャ設計  | FCでのリソース消費<br>結果として生じる計算待ち時間      | 一般的な障害モード<br>FCでの障害伝播 |
| 開発とIWQ契約     | 待ち時間を満たすために割り当てられたリソース            | 必要な信頼性レベル             |

- ✓ コストとスケジュール
- ✓ インターフェース
- ✓ パフォーマンス

- ✓ メンテナンス性
- ✓ 安全/セキュリティ
- ✓ ...

- ✓ IWQ
- ✓ 製品ポリシー

