サービス運用に携わるエンジニアのための

# 「エンジニアリング」再入門

Internet Week 2025

運用設計ラボ合同会社

波田野 裕一

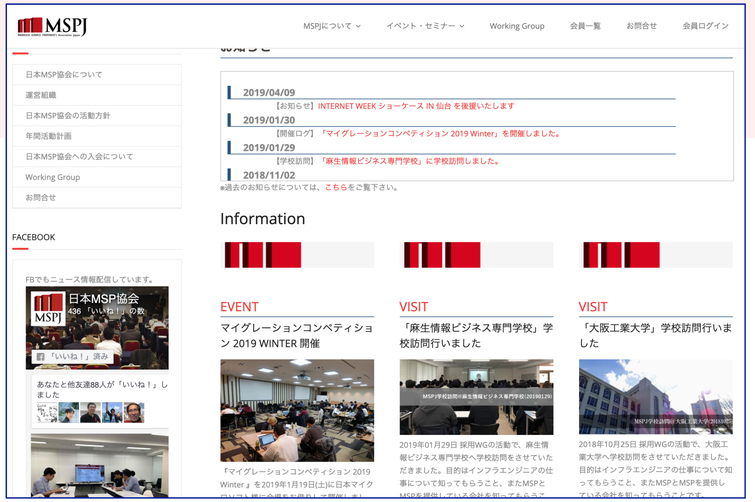
2025-11-26

# 日本MSP協会

IT情報基盤の運用サービスを提供するマネージド・サービス・プロバイダ及びIT情報基盤の運用に携わる技術者等と協力し、**運用の技術向上と品質向上、運用技術に携わる人材の発掘と育成、運用に関連する様々な評価軸を整理して明確化**するために日本MSP協会を設立します。そして、利用者にとって最適なIT情報基盤の選定と、適切なコストで安全かつ効率的に基盤を運用する指標を提供することで、さらなるIT産業界の活性化に貢献していきます。

MSPJ
MANAGED SERVICE PROVIDER'S Association Japan

https://mspj.jp/



# 運用設計ラボ合同会社

Operation 運用設計Lab

シニアアーキテクト

## 波田野 裕一

AWS Samurai 2017 (個人)
AWS Samurai 2020 (CLI専門支部)

AWS Community Hero

日本MSP協会 特別会員

**インプットご支援**

OpsLearn ®

科学的工学的な考え方に基づく講義とワークショップで
30年先も活きる運用設計スキルを身に付ける

OpsCLI ®

世界で最もAWS APIの仕様に忠実なeラーニングで
10年先も活きるAWSスキルを身に付ける

**現場での実践ご支援**

運用設計支援　　よろず相談

**アウトプットご支援**

**ホワイトペーパー/ガイドライン策定支援**
ホワイトペーパやガイドラインの制作や内製をご支援いたします。

# 概要

みなさんは「エンジニアリング」という言葉に、どのようなイメージを持っているでしょうか。日本のIT業界においては、「エンジニアリング」を以下のイメージで使用している方が少なくないのが現状です。

- 流行している最新技術を使いこなすのが、格好良い「エンジニアリング」である。
- 他人が理解できないような超絶的な技術を駆使する事が、最高の「エンジニアリング」である。
- 至高の技術を結集し、孤高の立場で組織をリードする事が、真の「エンジニアリング」である。

このような認識が日本のIT(特にサービス運用)において多くの課題を産み出す原因となっています
本セッションでは、このような「勘違い」を正し、あるべき「エンジニアリング」の形について解説、議論していきます。

# アジェンダ

1. こんな「エンジニアリング」に困ったことはありませんか
2. 「エンジニアリング」とは
3. 「エンジニアリング」の4つの特性
4. 「エンジニアリング」と「運用」

# 1. こんな「エンジニアリング」に困ったことはありませんか

# こんな「エンジニアリング」に困ったことはありませんか

| | |
|---|---|
| **独り仕事** | **チーム仕事を嫌がり、独りで仕事**をしようとする |
| **神秘主義** | 超絶技が良いことで、**他人が理解できることは価値が低い**と考える |
| **斜に構えて説明しない** | 「わかってくれる人がわかれば良い」と**説明責任を軽視**する |
| **技術しか興味ない** | ビジネス(目的)より**技術(手段)を重視**する |
| **安定性は後回し** | 実績がある安定したものよりも、**先進性や独自性を優先**する |
| **熱しやすく冷めやすい** | 動作するまでは熱心だが、**本番環境の継続性にあまり興味が無い** |
| **新しい事にしか興味無い** | ビジネスよりも、**新しい技術の知見を賞賛されることを重視**する |

# こんな「エンジニアリング」に困ったことはありませんか

独り仕事

神秘主義

斜に構えて説明しない

技術しか興味ない

安定性は後回し

熱しやすく冷めやすい

新しい事にしか興味無い

こんな「エンジニアリング」を

**あなた自身がやっていたり**

やっている人に**振り回されていたり**

していませんか?

講師(波田野)は、

**やらかしてましたし、振り回されてもいました**

# 困った「エンジニアリング」の分類

| | |
|---|---|
| **独り仕事** | **チーム仕事を嫌がり、独りで仕事**をしようとする |
| **神秘主義** | 超絶技が良いことで、**他人が理解できることは価値が低い**と考える |
| **斜に構えて説明しない** | 「わかってくれる人がわかれば良い」と**説明責任を軽視**する |
| **技術しか興味ない** | ビジネス(目的)より**技術(手段)を重視**する |
| **安定性は後回し** | 実績がある安定したものよりも、**先進性や独自性を優先**する |
| **熱しやすく冷めやすい** | 動作するまでは熱心だが、**本番環境の継続性にあまり興味が無い** |
| **新しい事にしか興味無い** | ビジネスよりも、**新しい技術の知見を賞賛されることを重視**する |

これらはエンジニアリング
ではなく
悪い意味での
**「職人技」**

これらはエンジニアリング
ではなく
悪い意味での
**「研究」**

# こんな「エンジニアリング」の弊害

## 悪い意味での「職人技」の弊害

職人技とは、身につけた熟練した技術によって、手作業で物を作り出す技のこと

**独り仕事**

**チーム仕事を嫌がり、独りで仕事**をしようとする

**業務がブラックボックス化、既得権益化し、運用現場が硬直化**する

**神秘主義**

超絶技が良いことで、**他人が理解できることは価値が低い**と考える

保守や引き継ぎができず、**長期的に負債化、ボトルネック化**する

**斜に構えて説明しない**

「わかってくれる人がわかれば良い」と**説明責任を軽視**する

周囲からは理解されず、**関与する上司、後輩などの評価も低下**させる

**技術しか興味ない**

ビジネス(目的)より**技術(手段)を重視**する

**ビジネスから乖離した振る舞いで評価が低下**し、厄介者扱いされる

# こんな「エンジニアリング」の弊害

## 悪い意味での「研究」の弊害

研究とは、新しい知識や技術を発見し、その新たな活用方法を見つけるための活動のこと

### 安定性は後回し

実績がある安定したものよりも、**先進性や独自性を優先**する

**運用業務が不安定**になり、**保守や引き継ぎも困難**になる

### 熱しやすく冷めやすい

動作するまでは熱心だが、**本番環境の継続性にあまり興味が無い**

**保守・廃止できない環境の乱立や不安定化**により**現場が混乱**に陥る

### 新しい事にしか興味無い

ビジネスよりも、**新しい技術の知見を賞賛されることを重視**する

**現実から乖離した振る舞いで評価が低下**し、厄介者扱いされる

# 2. 「エンジニアリング」とは

# 「エンジニアリング」という言葉の誕生

## 「エンジニアリング」という言葉の誕生

**産業革命開始の約60年前**が初出　　イギリスで工業が急速に多様化した時代 (蒸気機関の黎明期)


(出典: Merriam-Webster辞典)

# 「エンジニアリング」がなぜ誕生したか調べてみました

## 「エンジニアリング」という言葉の誕生

**産業革命開始の約60年前**が初出　　イギリスで工業が急速に多様化した時代 (蒸気機関の黎明期)

## 産業革命の特色は「大量生産の実現」

産業革命により、**大量の製品**を**一定の効率**で**安定的・持続的に生産**できるようになった

## 産業革命の礎となった「エンジニアリング」

エンジニアリングは　　**適切なQCD**(品質、コスト、納期)で
**反復再現的な製造方法**で
**持続可能な形**で　　**工業製品の大量生産を実現**した

大切なことは、**エンジニアリングによって「製品」というビジネス上の価値が生まれる**こと

# ITの「エンジニアリング」は何を目指すのか考えてみました

## ITにおける「エンジニアリング」は、工業「エンジニアリング」の応用

**その本質**(大量の生産・処理)**に違いは無い**

## IT(情報技術)**の特色は「大量の情報処理の実現」**

ITにより、**大量の情報**を**一定の効率**で**安定的・持続的に処理**できるようになった

## ITにおける「エンジニアリング」

ITのエンジニアリングは

**適切なQCD**(品質、コスト、納期)で
**反復再現的な処理方法**で
**持続可能な形**で

**情報の大量処理を実現**する

大切なことは、**エンジニアリングによって「情報」というビジネス上の価値が生まれる**こと

# ITエンジニアリングが産み出す「情報」

人が作り出す情報 (例)

- 要求定義
- 概要設計書
- アーキテクチャ
- 詳細設計書
- 手順書
- ソースコード

システムが作り出す情報 (例)

- 実行ファイル
- インタフェース
- マスターデータ
- テンポラリデータ
- パケット
- ログ

大切なことは、**エンジニアリングによって「情報」というビジネス上の価値が生まれる**こと

# ITにおける「エンジニアリング」とは何か?

**適切なQCD**で　ビジネス視点でバランスを取る

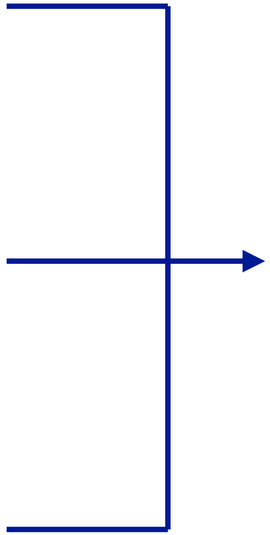
口よりデータに語らせる

**反復再現的な処理方法**で　チームで活動する

業務を言語化・構造化する

**持続可能な形**で　引き継ぎを重視する

陳腐化の回避を重視する



**情報の大量処理**
を実現

大切なことは、**エンジニアリングによって「情報」というビジネス上の価値が生まれ続ける**こと

# 「職人技」との違い

| | エンジニアリング | 職人技 |
|---|---|---|
| 適切なQCDで | ビジネス視点でバランス | Q(品質)に重点 |
| | 口よりデータ | 経験を重視 |
| 反復再現的な処理方法で | チームで活動 | (主に)個人で活動 |
| | 業務を言語化・構造化 | 主観を研ぎ澄ます |
| 持続可能な形で | 引き継ぎを重視する | 引き継ぎは無頓着 |
| | 陳腐化の回避を重視する | (陳腐化の回避を重視する) |

職人技とは、身につけた熟練した技術によって、手作業で物を作り出す技のこと

# 「研究」との違い

| | エンジニアリング | 研究 |
|---|---|---|
| 適切なQCDで | **ビジネス視点でバランス** | **Q(品質)**に重点 |
| | **口よりデータ** | (口よりデータ) |
| 反復再現的な処理方法で | **チーム**で活動 | (主にチームで活動) |
| | **業務を言語化・構造化** | **事実や事象**を言語化・客観化 |
| 持続可能な形で | **引き継ぎを重視**する | **論文**を重視 |
| | **陳腐化の回避を重視**する | **新規性**が不可欠 |

研究とは、新しい知識や技術を発見し、その新たな活用方法を見つけるための活動のこと

# 「エンジニアリングであるもの」vs.「エンジニアリングでないもの」

## エンジニアリングであるもの (例)

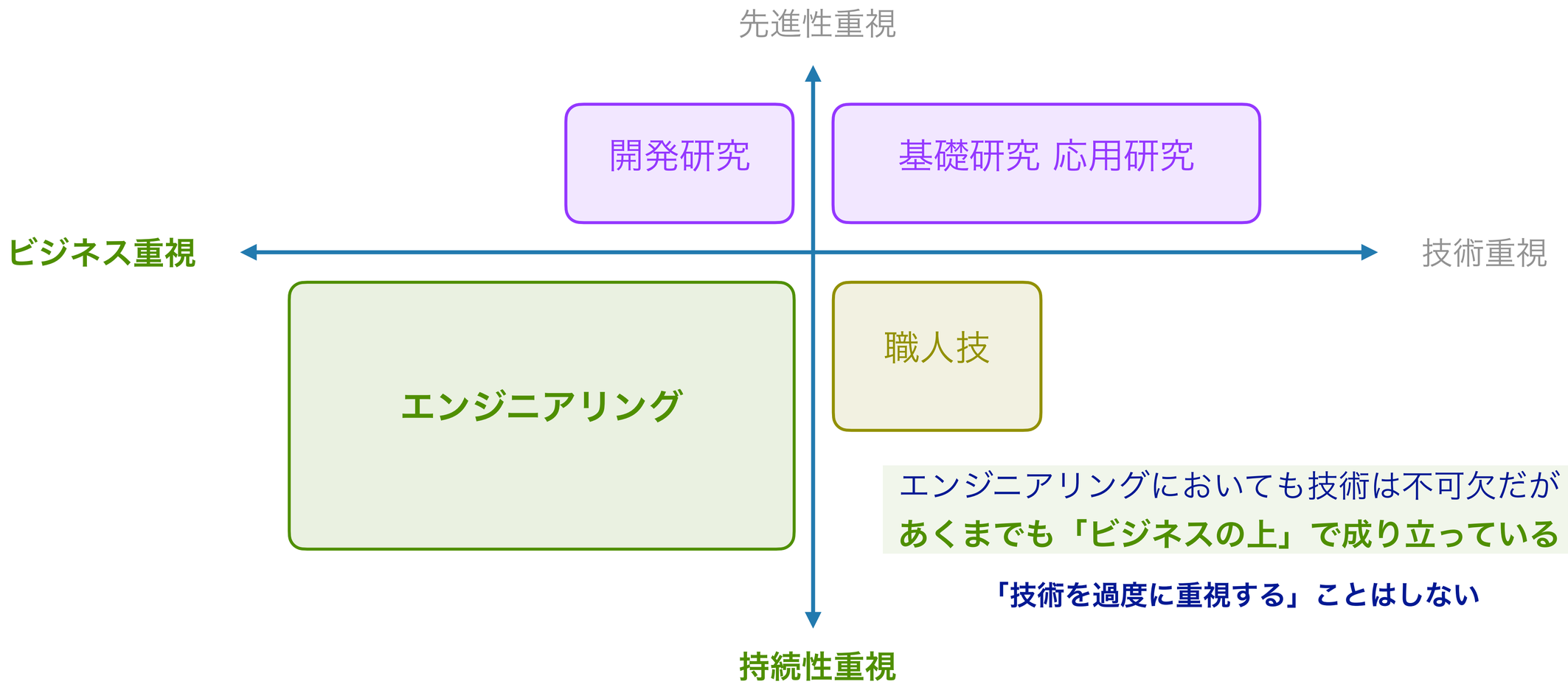
- **ビジネスが主**、技術が従
- **チームで協調**した実践活動
- **言語化されて客観的**
- **持続性重視**(陳腐化回避)
- **持続性重視**(引き継ぎ重視)

## エンジニアリングでないもの (例)

- **ビジネスより技術優先**
- **組織から乖離**した個人活動
- **言語化されず主観的**
- **持続性軽視**(陳腐化受容)
- **持続性軽視**(引き継ぎしない)

大切なことは、**エンジニアリングによって「情報」というビジネス上の価値が生まれる**こと

# 「職人技」「研究」「エンジニアリング」の位置付け




エンジニアリングにおいても技術は不可欠だが
**あくまでも「ビジネスの上」で成り立っている**

**「技術を過度に重視する」ことはしない**

# まとめ:「エンジニアリング」とは

## ITにおける「エンジニアリング」とは

適切なQCDで

反復再現的な処理方法で

持続可能な形で

→ 情報の大量処理を実現すること

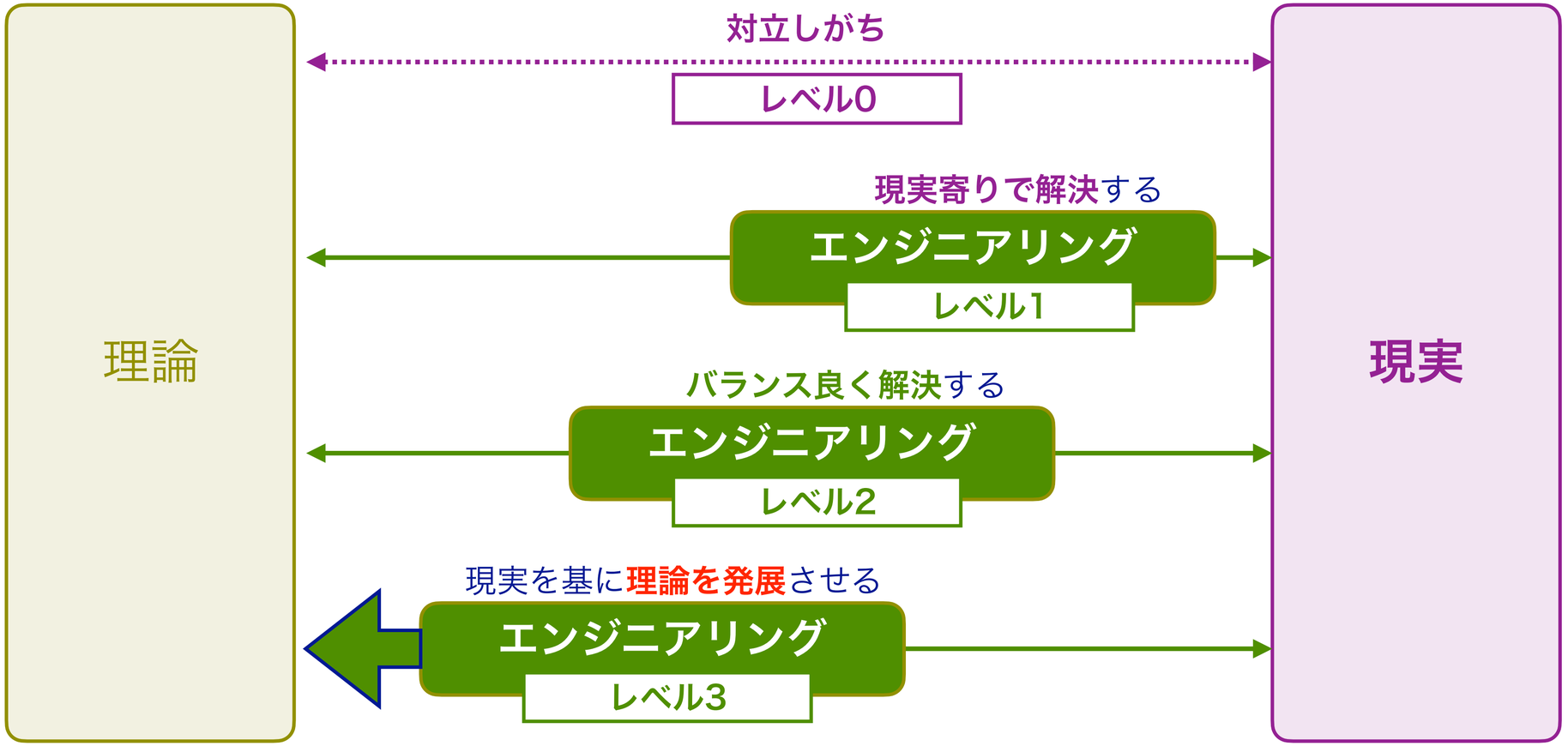
「職人技」とは混同しないこと

「研究」とも混同しないこと

## エンジニアリングは「ビジネス」と「持続性」を重視する

# 参考: 「エンジニアリング」の役割

「理論」と「現実」の橋渡しをするのが「エンジニアリング」の役割



# 3.「エンジニアリング」の4つの特性

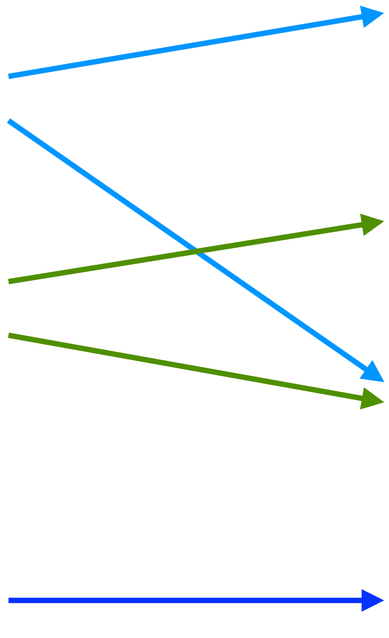
# エンジニアリングが具えるべき4特性

ITにおける「エンジニアリング」

**適切なQCD**
で計測・判断

**反復再現的な処理方法**
を設計・実装

**持続可能な形**
で実現・継続



**客観的**: 物事を第三者の視点で語る

**論理的**: 物事を論理の法則で語る

**合理的**: 物事を理性や道理で語る

**持続的**: 物事が中断しない様に考える

# 客観的

適切なQCDで　**客観的**　**第三者の視点**で物事を見る、考える

「**一人称**」**を主語**にして物事を語らない

「**3つの三人称**」**を意識**して物事を語る

| | |
|---|---|
| 「顧客の視点」 | 顧客から見た**費用対効果 (Q/CxD)** |
| 「経営者の視点」 | 経営者から見た**費用対効果 (Q/CxD)** |
| 「専門家の視点」 | 専門家から見た**生産性 (Q/CxD)** |

# 参考: 立場による「QCD」の違い

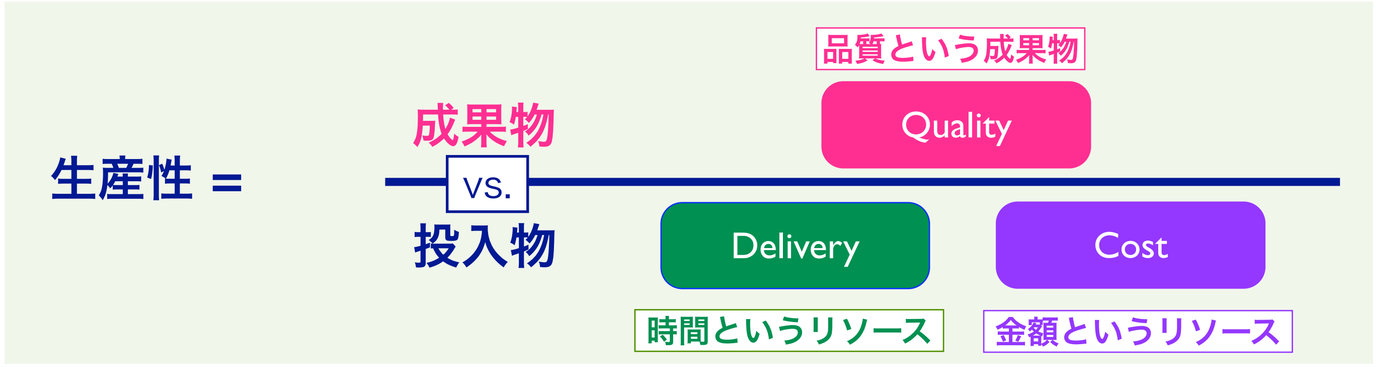
「顧客の視点」　顧客から見た**費用対効果 (Q/CxD)**

「経営者の視点」　経営者から見た**費用対効果 (Q/CxD)**


**品質**・**納期**・**コスト**の
3つの指標で**選択**・**評価**する

**コストに**
**過剰にフォーカスしがち**

「専門家の視点」　専門家から見た**生産性 (Q/CxD)**




**品質に**
**過剰にフォーカスしがち**

主に**品質**・**納期**の
2つの指標で**選択**し、
全ての指標で**評価**する

# 参考: エンジニアリングにおける「QCD」へのスタンス

「専門家の視点」　専門家から見た**生産性 (Q/CxD)**

**生産性 =** **成果物** vs. **投入物**

品質という成果物: Quality

Delivery（時間というリソース）　Cost（金額というリソース）

品質に
過剰にフォーカスしがち

主に**品質**・**納期**の
2つの指標で**選択**し、
全ての指標で**評価**する

**独立変数**: Quality, Delivery

生産性を向上させる場合に、**品質(Q)**と**納期(D)**を**主に制御**する。

**従属変数**: Cost

**コスト(C)**は、**結果として増減**する。

(良い例) 品質と納期を調整して、コストを下げる (制御できている)

**コストを独立関数として操作するのは悪手**

(悪い例) コストを削減して、品質と納期が劣化する (制御できていない)

# 参考: 「納期(D)」という盲点

「顧客の視点」　顧客から見た**費用対効果 (Q/CxD)**

「経営者の視点」　経営者から見た**費用対効果 (Q/CxD)**


専門家視点

比較的**容易に平準化**できる

顧客・経営層視点

**満足度が向上**しやすい

「専門家の視点」　専門家から見た**生産性 (Q/CxD)**


**再設計**や**改善**の
**初期対象**としてオススメ

# 論理的、合理的

反復再現的な処理方法で **論理的** **論理の法則**に適って物事を見る、考える

言っていることが、**文面として正しい** 表層的に正しい

**合理的** **理性や道理**に適って物事を見る、考える

言っていることが、**道理として正しい** 本質的に正しい

注意すべき組合せ

**論理的** X **非合理的** 「はい、論破」のような物言い

**非論理的** X **合理的** 説明が苦手なエンジニアの物言い

# 持続的

持続可能な形で　**持続的**　ある**状態が中断しないで続く**ようにする

**一人しかできない状態**にしない

**「陳腐化」を回避**し続ける

**「復元力」を意識**する

# 求められるエンジニアリング的な「美学」

**主観的な言葉で語る**のではなく　　客観的

**ロジック・モデル・データ**に語らせる

**実装に重きを置く**のではなく　　論理的　合理的

**理論やモデル**(アーキテクチャ)**に重き**を置く

**近視眼的な視点で見る**のではなく　　持続的

**長期的・俯瞰的な視点**で物事を捉える

# エンジニアとしての「4つの特性」の伸ばし方

| | | | |
|---|---|---|---|
| レベル1 | 客観的 | ステップ1 | **専門家としての視点**で語る。(主語が一人称ではない) |
| | | ステップ2 | **根拠が明確**である。(公式情報や参考文献を明示している) |
| | | ステップ3 | **経営層やユーザーの視点**で語る。(経営層やユーザーの声を代弁している) |
| レベル2 | 論理的 | ステップ1 | **文法的に正しい + 読み手にやさしい。**(ローコンテキストである) |
| | | ステップ2 | **論旨がシンプル**である。(開始から結語までが最短である) |
| | | ステップ3 | **論理構造が明確**である。 |
| レベル3 | 合理的 | ステップ1 | **AsIs(現状把握)が現実的**である。 |
| | | ステップ2 | **ToBe(ゴール像)が適度に理想的**である。 |
| | | ステップ3 | AsIsとToBeの**ギャップを解消する道筋が最短かつ適切**である。 |
| レベル4 | 持続的 | ステップ1 | 業務要素に対する**変更フローが明確**である。 |
| | | ステップ2 | 業務の全てが**サイクル指向**である。(ライフサイクルを明確に意識している) |
| | | ステップ3 | 業務の全てが**疎結合分散**に設計されている |

# まとめ:「エンジニアリング」の4つの特性

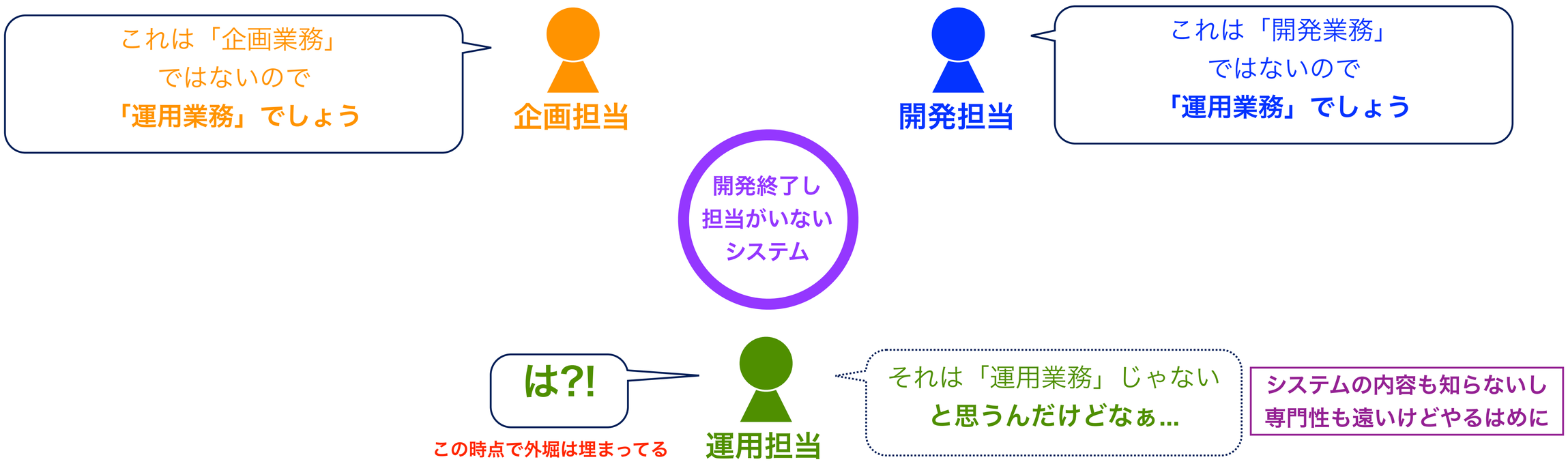
適切なQCDで → **客観的** : **第三者の視点**で物事を見る、考える

反復再現的な処理方法で →
- **論理的** : **論理の法則**に適って物事を見る、考える
- **合理的** : **理性や道理**に適って物事を見る、考える

持続可能な形で → **持続的** : ある**状態が中断しないで続く**ようにする

# 4.「エンジニアリング」と「運用」

# よくある「運用でよろしく」




運用技術を磨いてきたはずなのに**「なんでも屋」さんに**なっている…

# 「運用」の特徴を説明できずに苦労することが多い

## 企画と開発は、名前から仕事内容を想像しやすい

**企画**: 需要・要求調査、企画資料作成、稟議、など

**開発**: プロダクトの設計、進捗管理、コーディング、など

## 運用は、名前から仕事内容を想像しにくい

**「運用はいつも忙しそうだが何をやっているかわからない」**と言われてしまう

# 企画や開発と「運用」の違いを考えてみました

企画担当

開発担当

**企画**から**リリース**までが仕事

プロジェクト型業務

**成果に区切り**(リリース)**がある**ので
**特徴が目に付きやすい**

**全く同じ**プロジェクト
**全く同じ**リリース
**は存在しない**

毎回**やり方が違う**

毎回**結果が違う**

**非反復的**な業務

**非再現的**な業務

運用担当

**リリース**から**サービス廃止**までが仕事

非プロジェクト型業務

**成果に区切りが無い**ので
**特徴が目に付きにくい**

毎回**やり方が違う**運用
は**維持できない**

毎回**結果が違う**運用
は**評価されない**

**反復的**な業務

**再現的**な業務

＋

**「24時間365日」の活動**
が期待されている

# 「エンジニアリング」と「運用」

**ITにおける「エンジニアリング」とは**

**適切なQCD**で

**反復再現的な処理方法**で

**持続可能な形**で

→ **情報の大量処理を実現すること**

**「エンジニアリング」と「運用」は相性が良い**

**運用担当**

**リリース**から**サービス廃止**までが仕事

非プロジェクト型業務

**成果に区切りが無い**ので
**特徴が目に付きにくい**

毎回**やり方が違う**運用
は**維持できない**

毎回**結果が違う**運用
は**評価されない**

**反復的**な業務

**再現的**な業務

＋

**「24時間365日」の活動**
が期待されている

# 「エンジニアリングでないもの」vs.「エンジニアリングであるもの」

**エンジニアリングでないもの** (例)

- **ビジネスより技術優先**
- **組織から乖離**した個人活動
- **言語化されず主観的**
- **持続性軽視**(陳腐化受容)
- **持続性軽視**(引き継ぎしない)

**エンジニアリングであるもの** (例)

- **ビジネスが主**、技術が従
- **チームで協調**した実践活動
- **言語化されて客観的**
- **持続性重視**(陳腐化回避)
- **持続性重視**(引き継ぎ重視)

「エンジニアリングではないもの」による運用

1. 人が理解しにくい運用になる
2. システムが扱いにくい運用になる
3. 論理破綻・矛盾による無駄・無意味が多い運用になる

**「エンジニアリング」による運用**

1. **人が理解しやすい運用になる**
2. **システムが扱いやすい運用になる**
3. **論理的に正しいことが検証された運用になる**

# エンジニアリング ではないもの/であるものによる運用

## 人材面(育成や引き継ぎ)、システム面(自動化)で大きな差が発生する

### 「エンジニアリングではないもの」による運用

1. 人が理解しにくい運用になる
2. システムが扱いにくい運用になる
3. 矛盾・論理破綻による無駄・無意味が多い運用になる

- ドキュメント化工数が膨大になる
- 中途・新人の戦力化に時間がかかる
- 環境変化への対応に遅れがちになる
- Whyが失われると硬直化に繋がる
- ツール製品を導入しても効果が出にくい (連携しにくい)

### 「エンジニアリング」による運用

1. **人が理解しやすい運用になる**
2. **システムが扱いやすい運用になる**
3. **論理的に正しいことが検証された運用になる**

- ドキュメント化工数を必要最小限にできる
- 中途・新人の戦力化に時間がかからない
- 環境変化への対応が比較的容易にできる
- Whyが失われにくいので硬直化を避けやすい
- ツール製品を導入したときに効果が出やすい (連携しやすい)

# まとめ:「エンジニアリング」と「運用」

| ITにおける「エンジニアリング」 | | 運用の価値 |
|---|---|---|
| **適切なQCD**で | → | **リリース**から**サービス廃止**までが仕事<br>(QCDの対象の一番近くにいる) |
| **反復再現的な処理方法**で | → | **反復的**な業務　　**再現的**な業務 |
| **持続可能な形**で | → | **「24時間365日」の活動**が期待されている |

情報の大量処理を実現すること

**「エンジニアリング」による運用**

1. **人が理解しやすい運用になる**
2. **システムが扱いやすい運用になる**
3. **論理的に正しいことが検証された運用になる**

**「運用」の価値は「エンジニアリング」から生まれる**

# Operation運用設計Lab

http://www.operation-lab.co.jp/