

運用設計導入コース (補講 第2回)

# Aslsフレームワーク

---

運用設計ラーニング

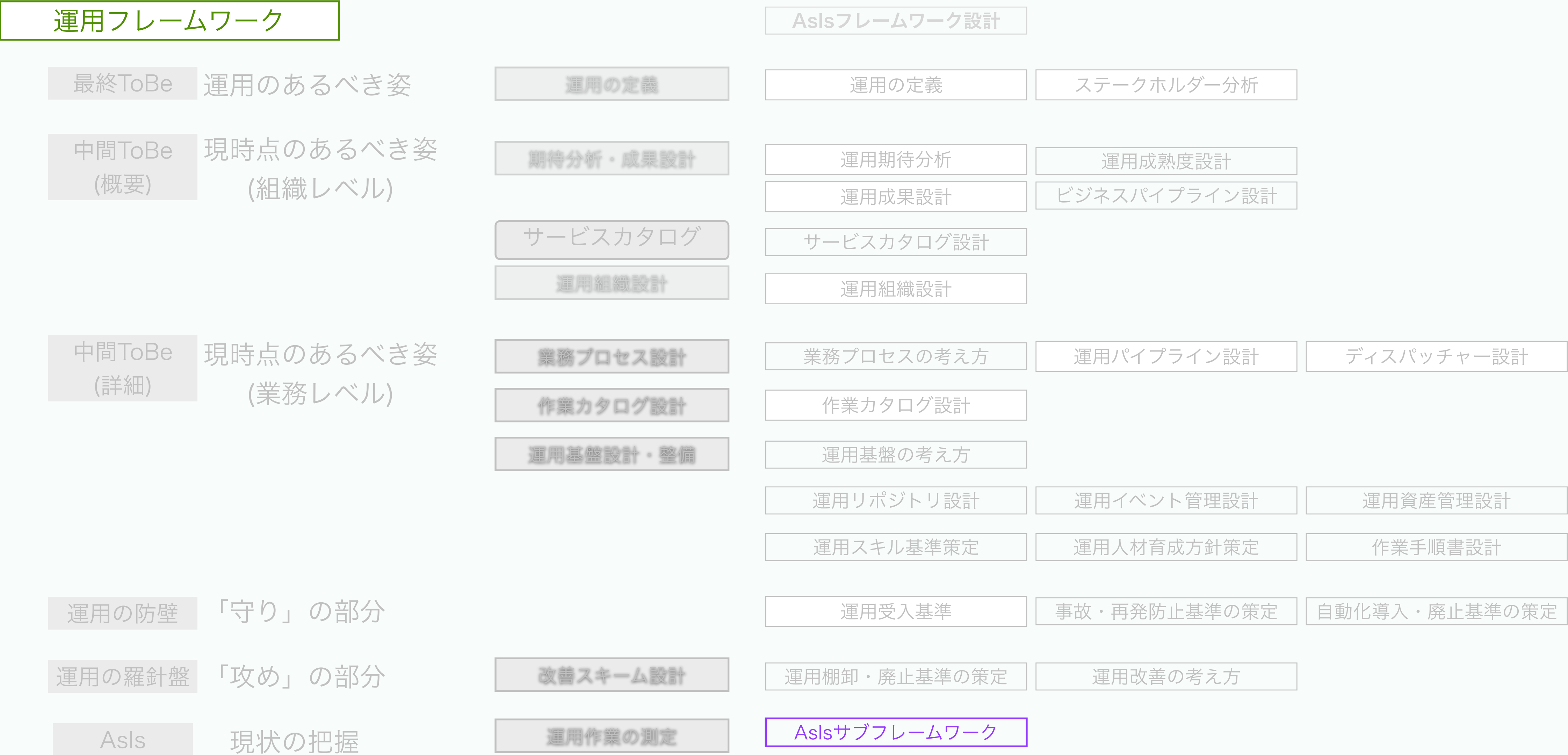
2023-03-09

<https://www.opslearn.jp/>

本編 Aslsフレームワーク

# Aslsフレームワーク インプットの収集と蓄積

# OpsLearnの運用設計モデルにおける「Aslsフレームワーク」



本編 Aslsフレームワーク

# Aslsフレームワーク

Aslsフレームワークの復習

# 運用課題の分析に必要なこと

運用の現実(Asls)を把握することが何より重要

「運用課題」というネガティブなAslsだけではなく、運用のポジティブなAslsも含めて把握すべき

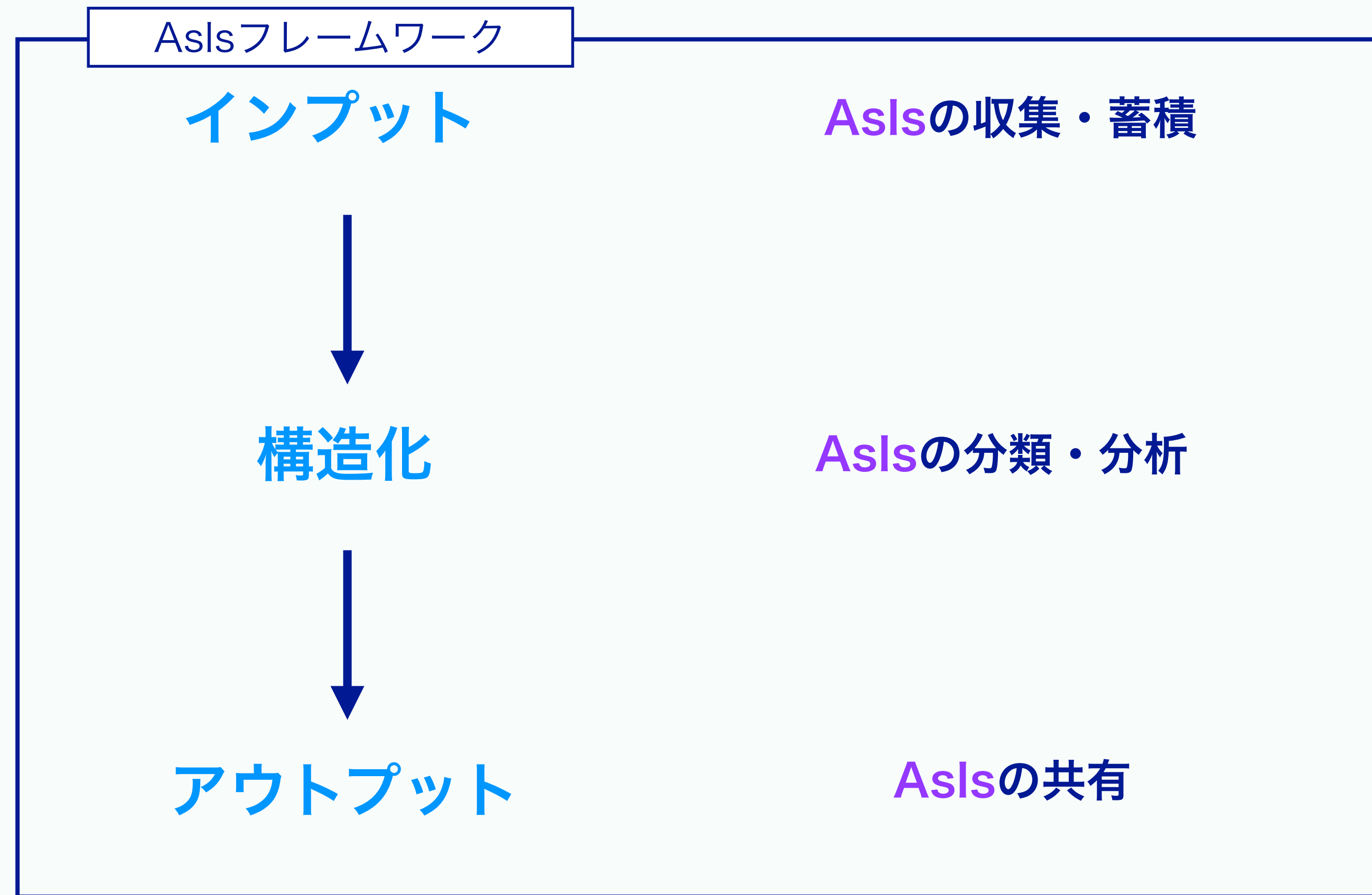
運用現場のAslsの収集と蓄積

運用現場のAslsを運用現場と設計側で共有

運用現場のAslsの継続的な分析

既に運用が動いている場合は、一番最初にやるべきこと。

# Aslsを知るためのサブフレームワーク

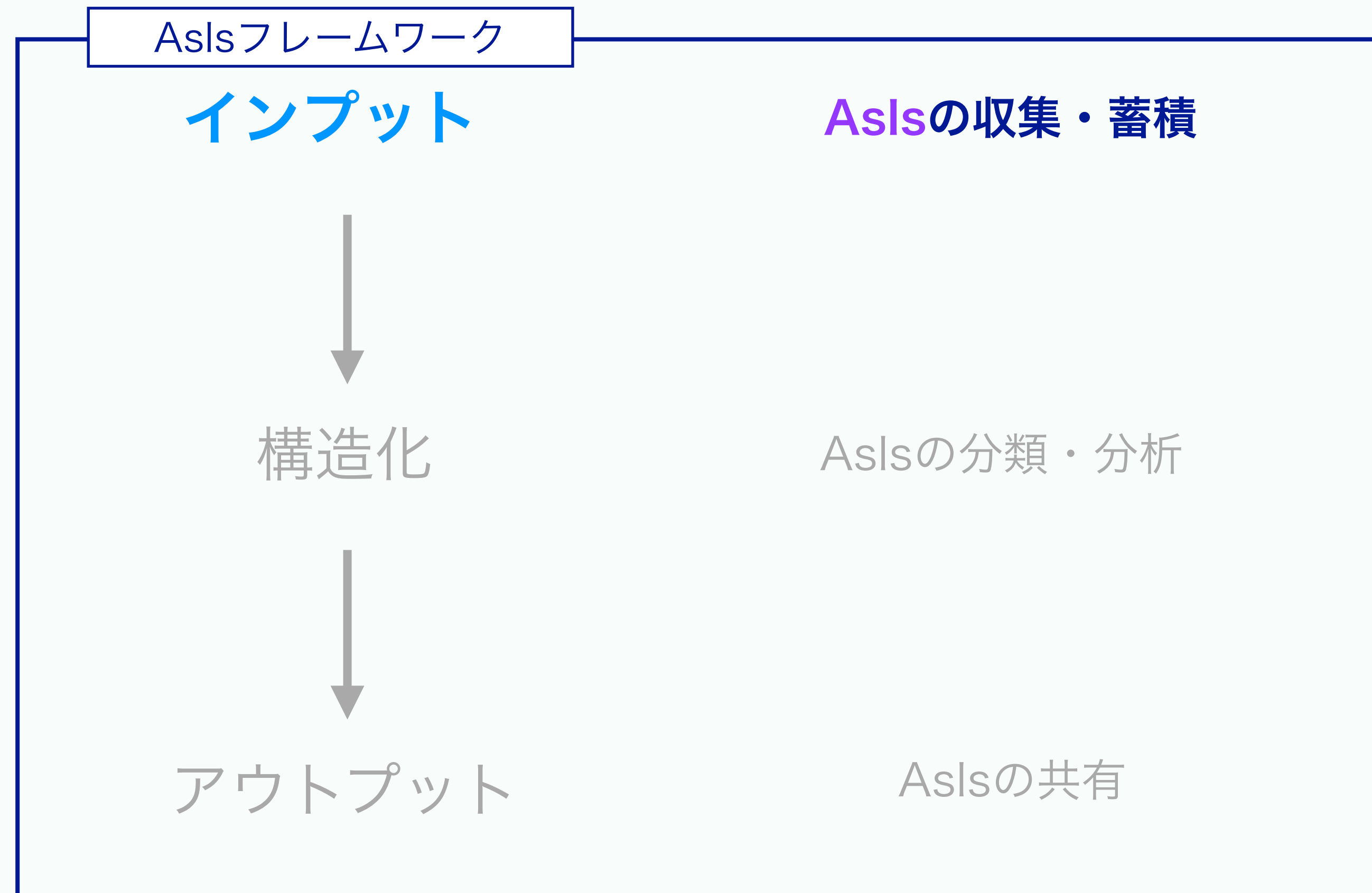


重要! 課題を見つけても、いきなり対応しようとするしない。

本編 Aslsフレームワーク

Aslsフレームワークの基点 = インプット

# Aslsフレームワークの基点 = インプット



# Aslsフレームワークの基点 = インプット

インプット

Aslsの収集・蓄積

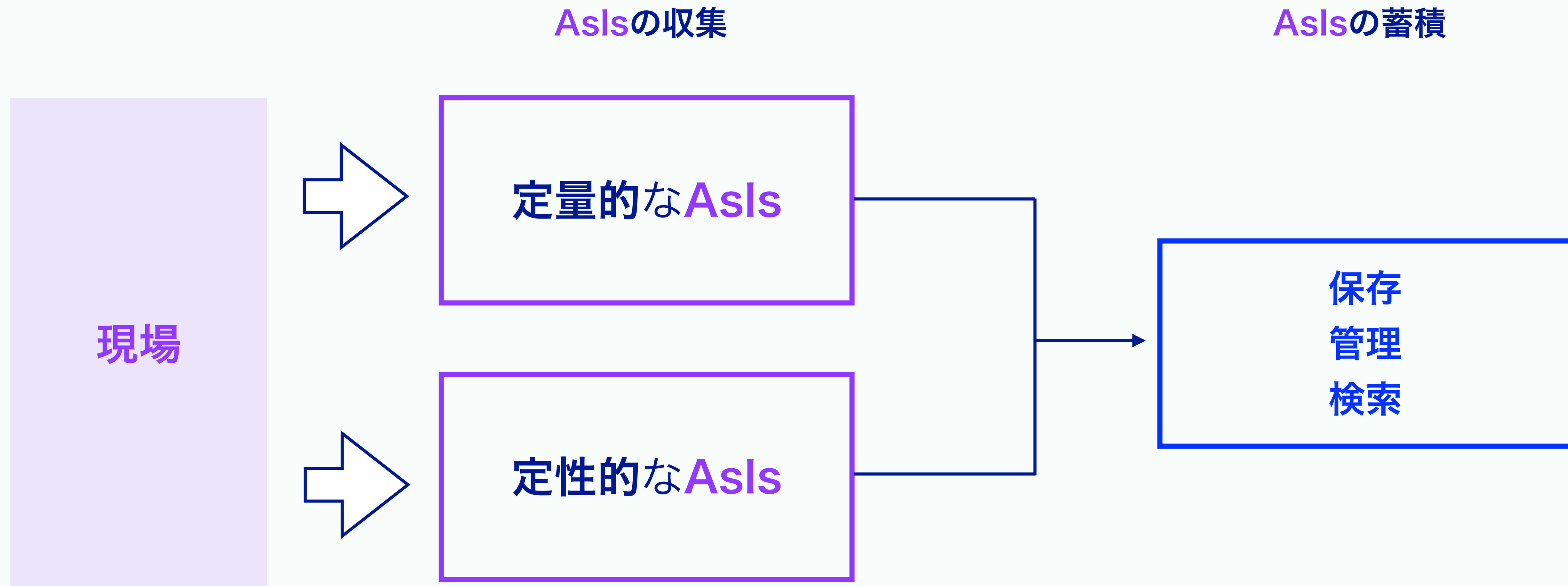
運用現場の現状を「ありのままに把握」するために

個人の「努力」や「気付き」だけに依存しないための  
**Asls収集・蓄積の「仕組み作り」が不可欠**

「自然とAsls情報が集まる」ための環境を実現する

# Aslsフレームワークの基点 = インプット (全体像)

## インプット



本編 Aslsフレームワーク

# Aslsインプットの基本構造

# Aslsインプットの基本構造

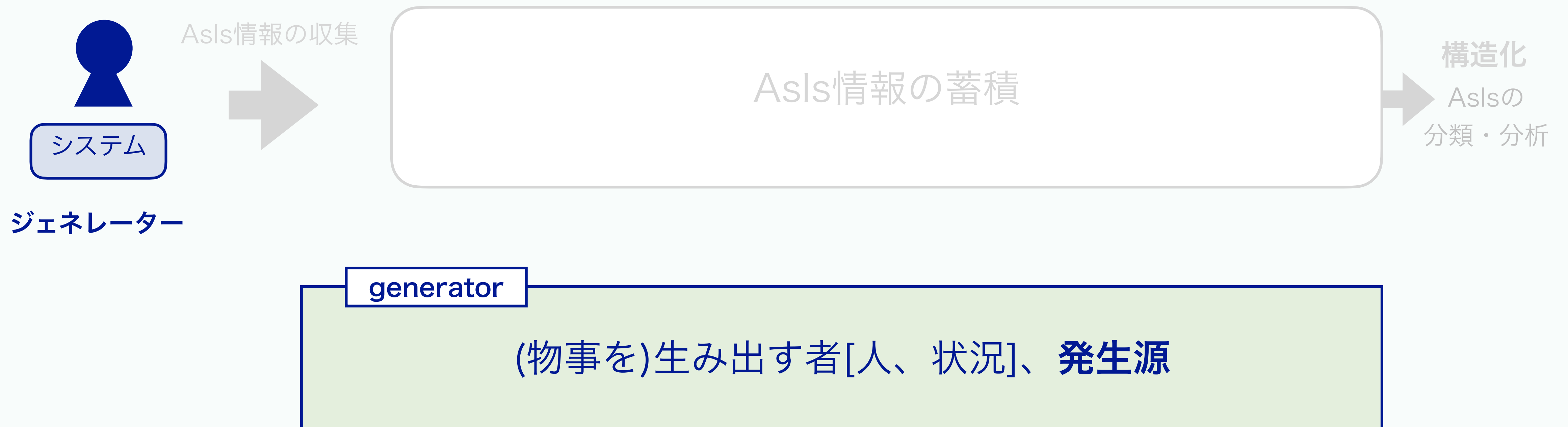


本編 Aslsフレームワーク

# Aslsインプットの基本構造

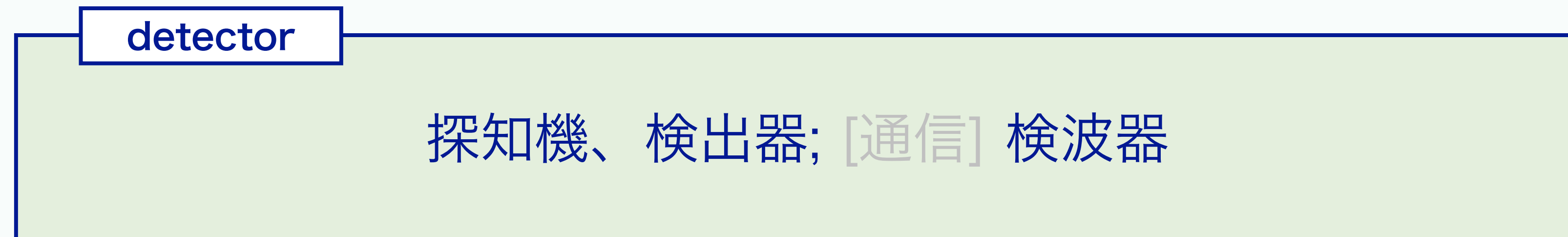
Aslsインプットの基本要素

# Aslsインプットの基本要素 (収集: ジェネレータ)



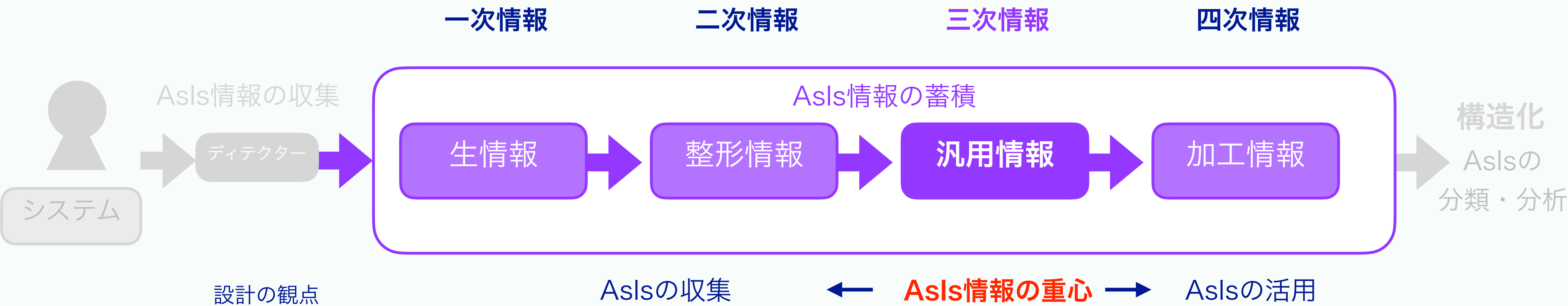
Asls情報の「発生源」となる人やシステム

# Aslsインプットの基本要素 (収集: 検知器)



Asls情報の入口となる仕組み

# Aslsインプットの基本要素 (蓄積)



# Aslsインプットの基本要素 (蓄積: 一次情報)



ディテクターから得られた**未加工のデータ・情報**

Asls情報が最初に記録される段階。

一次情報は、最も生の情報であることから、原則として永久保存。

(更新型は随時更新)

# Aslsインプットの基本要素 (蓄積: 二次情報)



未加工のデータ・情報を、**扱いやすいフォーマットに変換する**

一次情報に比べて、情報を欠落させることが多い。

例えば、メールのreceivedヘッダーは用途が無いために欠落させる、など。

二次情報は、一次情報を破棄する場合のみ永久保存。

(更新型は随時更新)

# Aslsインプットの基本要素 (蓄積: 三次情報)



整形済の情報を、**扱いやすい組み合わせに変換する**

大まかな用途に合わせて情報の組合せを決める。

例えば、監視対象を構成(セグメント)別、用途(サービス)別、経年別、など個別に作成する、など。

三次情報は、最も使われる情報であることから、随時更新される。

# Aslsインプットの基本要素 (蓄積: 四次情報)



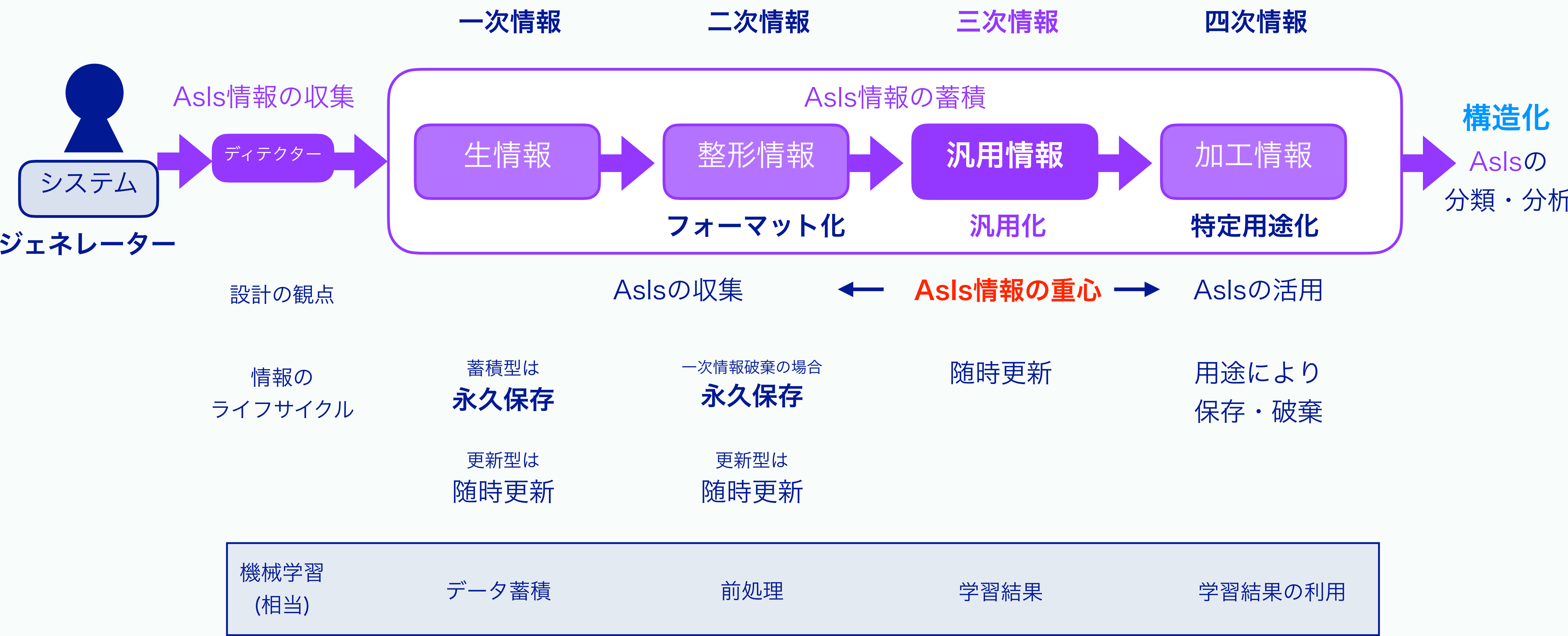
汎用情報を、**特定用途のために変換する**

用途の実現に最も最適な形を追求する。

例えば、特定相手のレポート、一時的な障害リスト、など。

四次情報は、用途に合わせて保存・破棄される。

# まとめ: Aslsインプットの基本要素



本編 Aslsフレームワーク

# Aslsインプットの基本構造

Aslsインプットの実装モデル

# Aslsインプットの実装モデル



# Aslsインプットの実装モデル (収集: 検知器)



## 受動型 検知器がジェネレータからの入力を待つ

例えば、メールクライアントでの受信や、監視システムのログ収集機能の「受信部分」、など。

## 能動型 検知器がジェネレータに対して収集しに行く。

例えば、監視システムの死活監視機能やエージェントの「検知部分」、など。

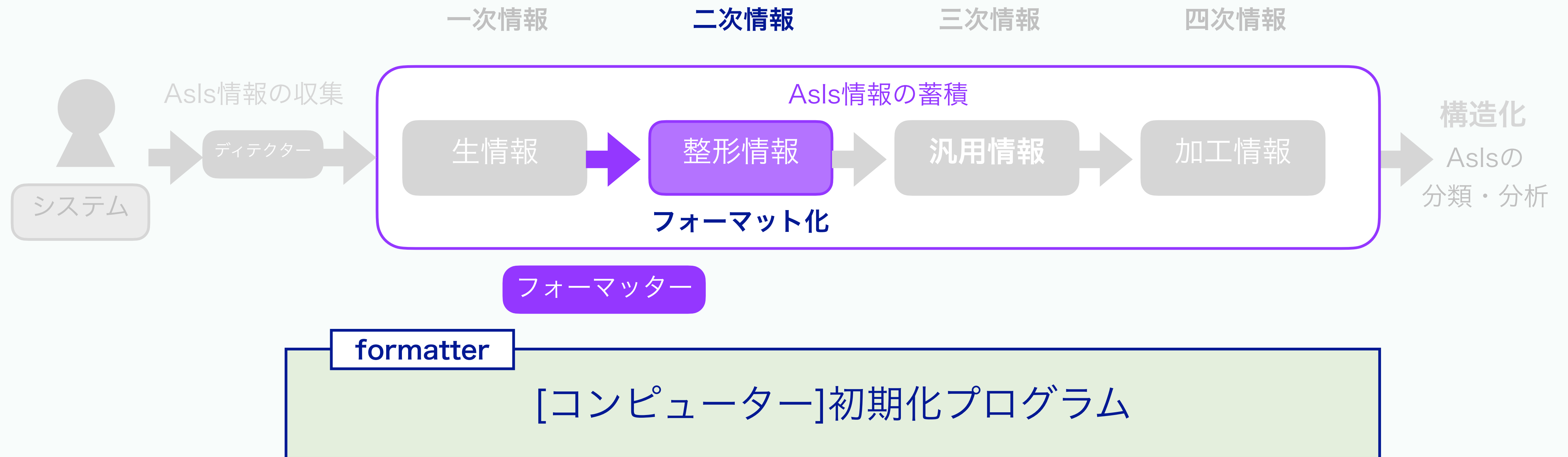
# Aslsインプットの実装モデル (蓄積: 一次情報の保存)



## 検知器の出力を、加工せずに保存する。

例えば、受信した顧客メールを、共有ストレージに保存する、  
syslogデーモンの出力したログを、共有ストレージに保存する、など。

# Aslsインプットの実装モデル (蓄積: 二次情報の加工)



**生情報を、人力やスクリプトで構造化形式に加工する。**

例えば、顧客メールのFrom、Date、本文をTSV(CSV)形式で保存する、  
syslogデーモンの出力したログをパースして、JSON形式で保存する、など。

# Aslsインプットの実装モデル (蓄積: 三次情報の加工)



整形情報を複数組み合わせ、新しい関係を示す情報を提供する。

例えば、顧客メールのDateと、その過去24時間にsyslogデーモンの出力したログを  
関係付けた情報を生成して、メールとsyslogの関係を示す情報を提供する。

# Aslsインプットの実装モデル (蓄積: 四次情報の加工)



汎用情報から、用途に合わせて加工する。

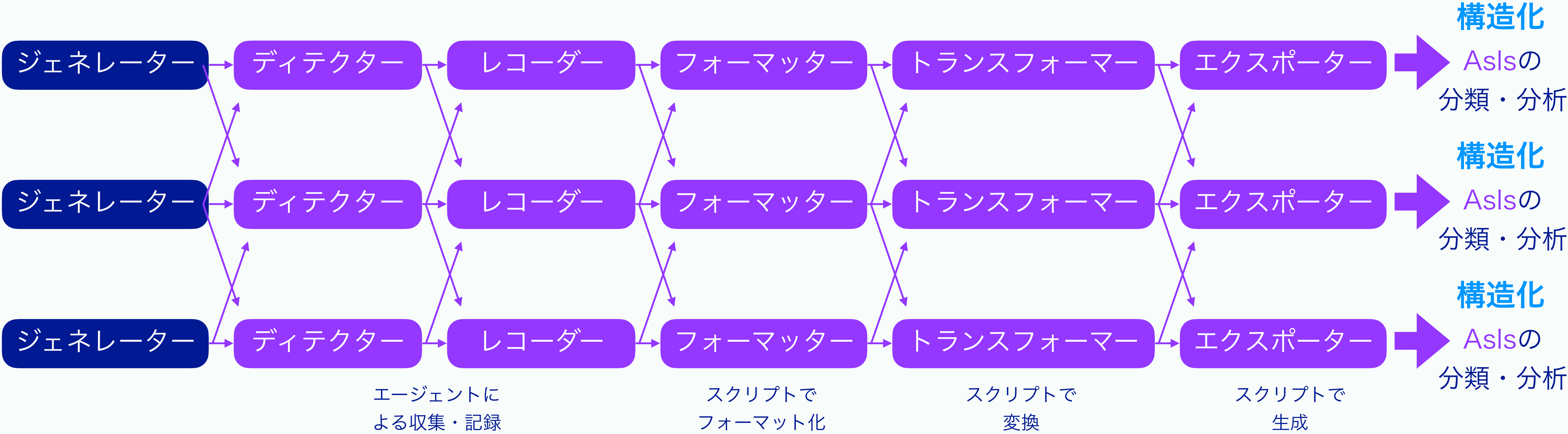
例えば、特定の顧客メールのDateと、その過去24時間にsyslogデーモンの出力したログの種別と各件数を表示するレポートを自動生成して提供する。

# まとめ: Aslsインプットの実装モデル



参考: Aslsインプットの基本要素を分ける意義

一つの**Aslsインプット要素**を**多用途**に使える



新しいAslsインプットを短期に実装  
Aslsインプット要素の保守工数を低減

本編 Aslsフレームワーク

# Aslsインプットの実装イメージ

本編 Aslsフレームワーク

# Aslsインプットの実装イメージ

定量的Aslsの収集

# 復習: Aslsのインプット (収集: 定量的Asls)

## インプット Aslsの収集

現場の定量的な情報を収集する仕組みが必要。

Step1

収集対象の選定

収集する定量的情報を選定する。

Step2

一次情報の決定

収集する定量的情報に必要な一次情報を決定する。

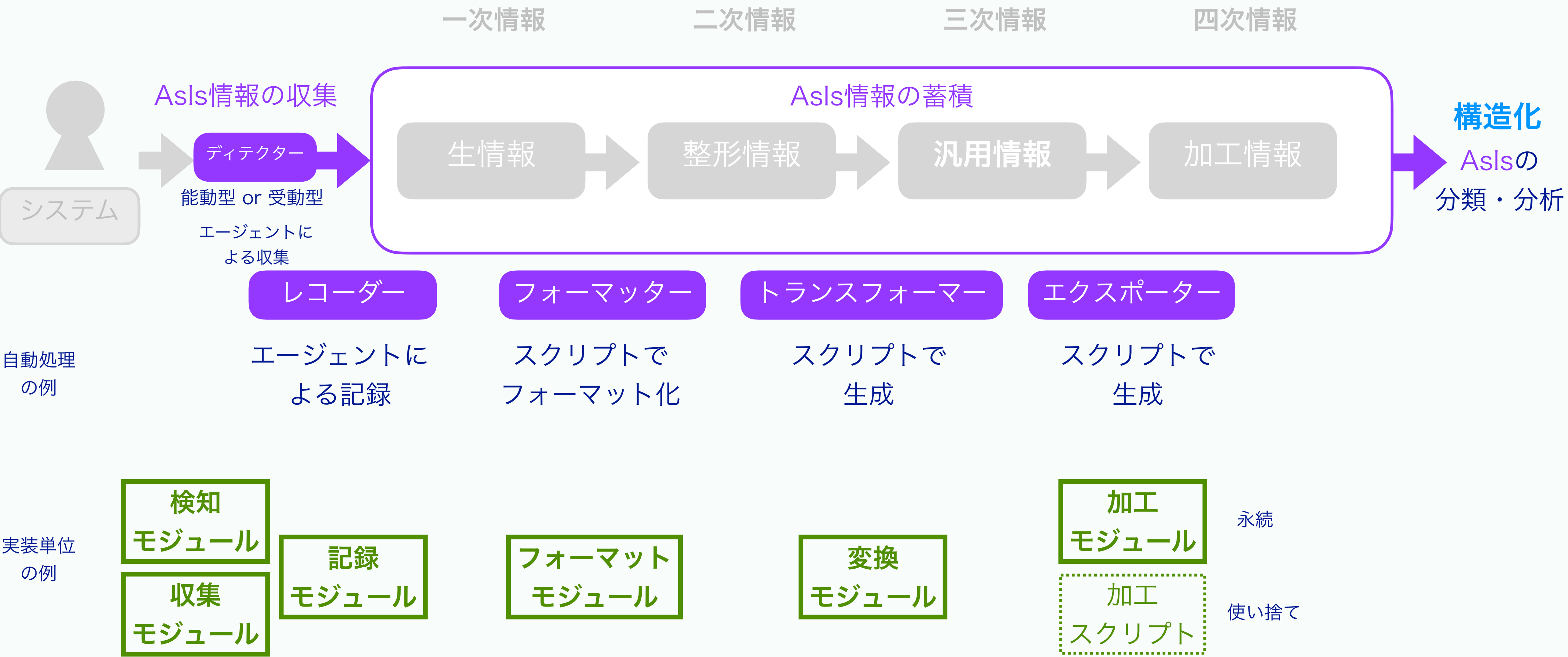
Step3

データフローの整備

一次情報を定量的情報に変換するためのデータフローを整備する。

データフローの整備により、定量的情報の自動集計への下地を整える。

# Aslsインプットの実装 (定量的なAsls)



# 定量Aslsインプット実装のメリット



- 短期的な効果が顕著      実装すればするほど、収集できる定量的Aslsの質と量が向上する。
- 副作用が少ない      読み取りが中心の自動化になるため、「自動化による副作用」が比較的少ない。
- 積み上げ型の活動      環境変化があっても、過去の実装が全く無駄になることはない。(経験は活かせる)

「定量的AsなAsls情報」に裏打ちされた「定性的Asls情報」は精度が高い

本編 Aslsフレームワーク

# Aslsインプットの実装イメージ

定性的Aslsの収集

# 復習: Aslsのインプット (収集: 定性的Asls)

## インプット      Aslsの収集

現場の定性的な情報が上がってくる、上がってきやすい仕組みが必要。

Step1

三現主義

マネージャやアーキテクトが定期的に現場に話を聞きに行く。

Step2

場の整備

気軽にヒヤリハットの存在を共有する場を整備する。

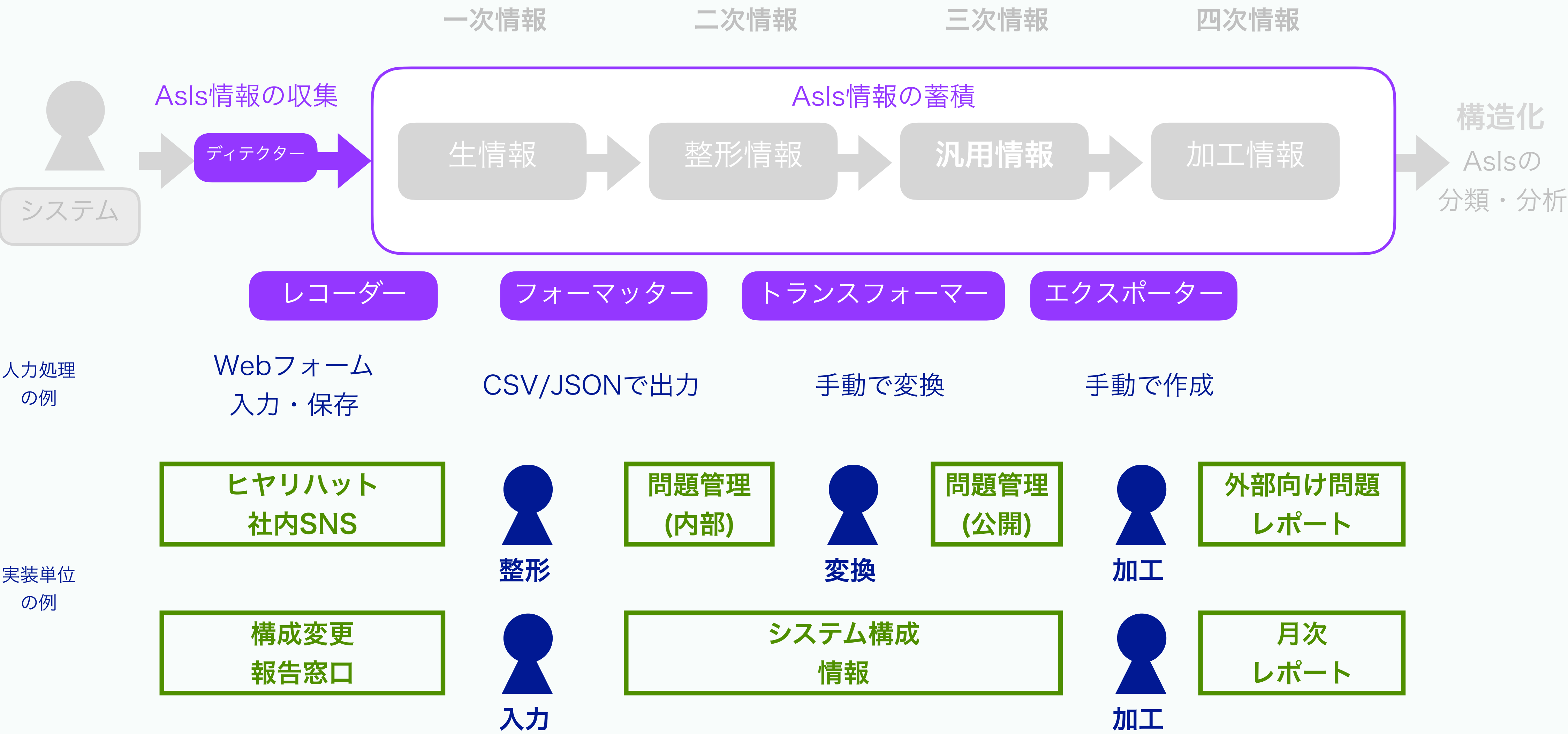
Step3

文化の醸成

最初にミスした人やネガティブ情報を上げた人を評価する。

「現場に聞きに行くこと」から初めて、仕組み化、文化の形成をしていく。

# Aslsインプットの実装イメージ (定性的なAsls)



# Aslsインプットの実装の留意点



定量的Aslsの補完が前提

定性的Asls情報だけでは、根拠が希薄なため活用しにくいことが多い。

ボトムアップアプローチが中心

貴重なAsls情報を失わないために、ディテクターとレコーダーから実装する。

仕組み+文化

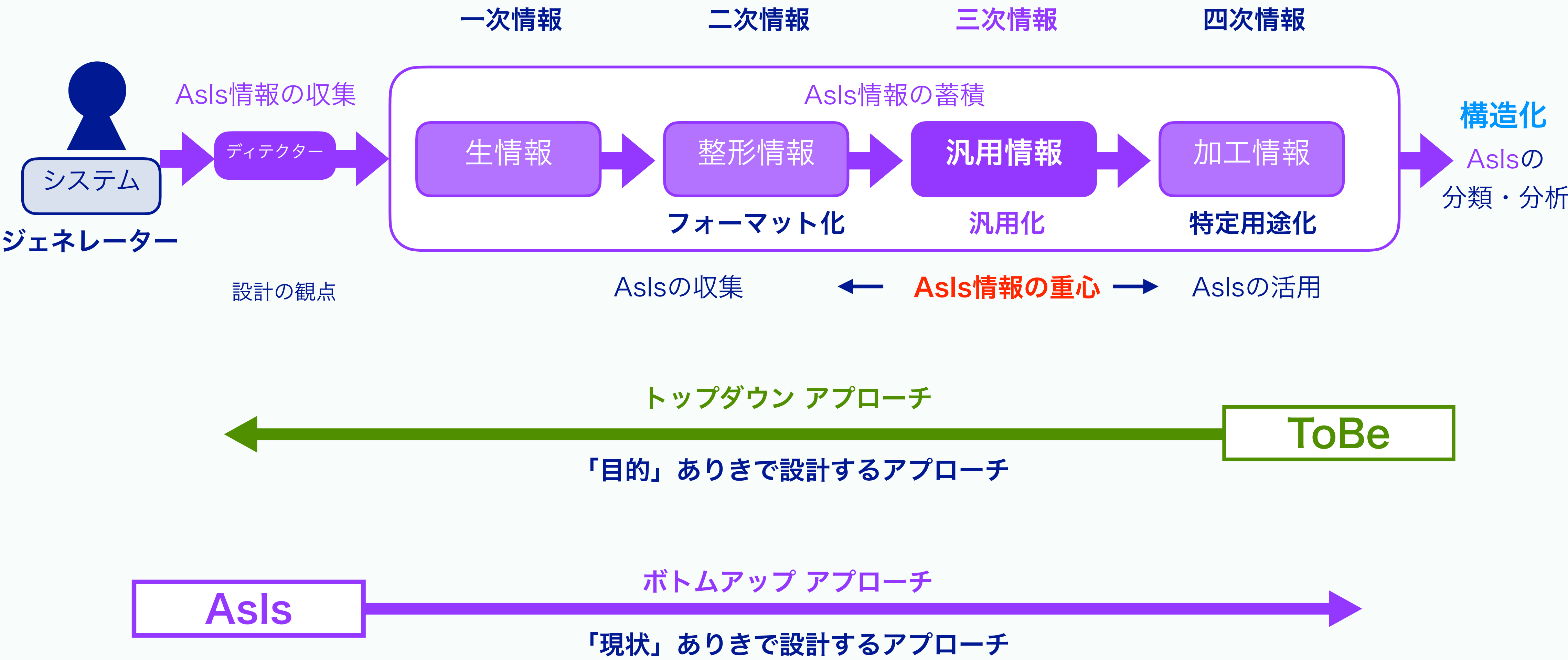
定性的Aslsの収集に「仕組み」が不可欠だが、それを「活用する文化」が無いと機能しない。

「定量的AsなAsls情報」に裏打ちされない「定性的Asls情報」は精度が低い

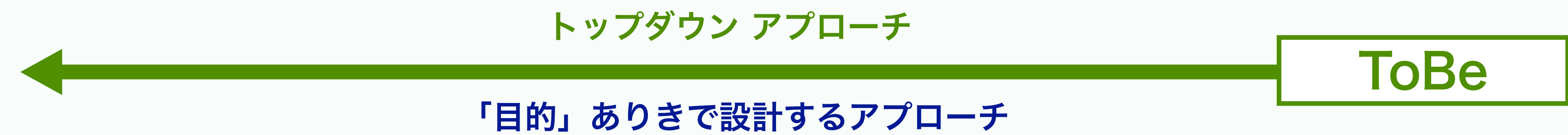
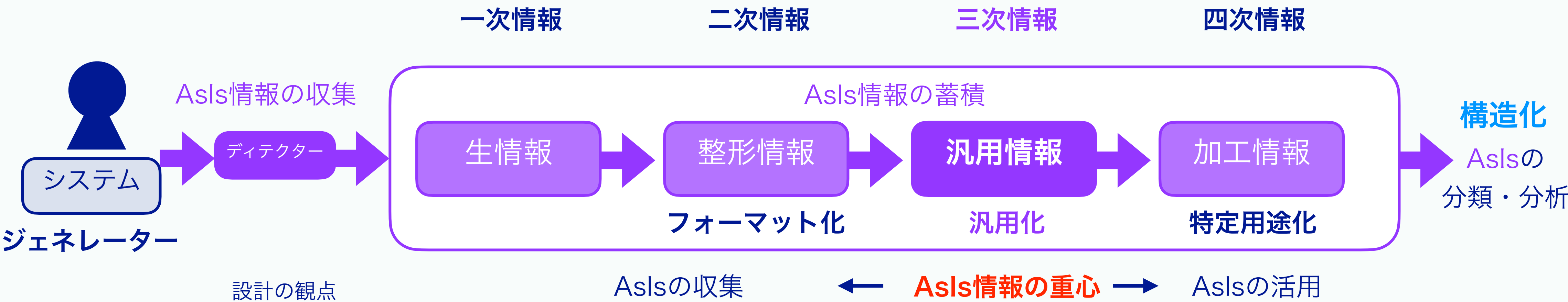
本編 Aslsフレームワーク

# Aslsインプットの設計アプローチ

# Aslsインプットの設計アプローチ



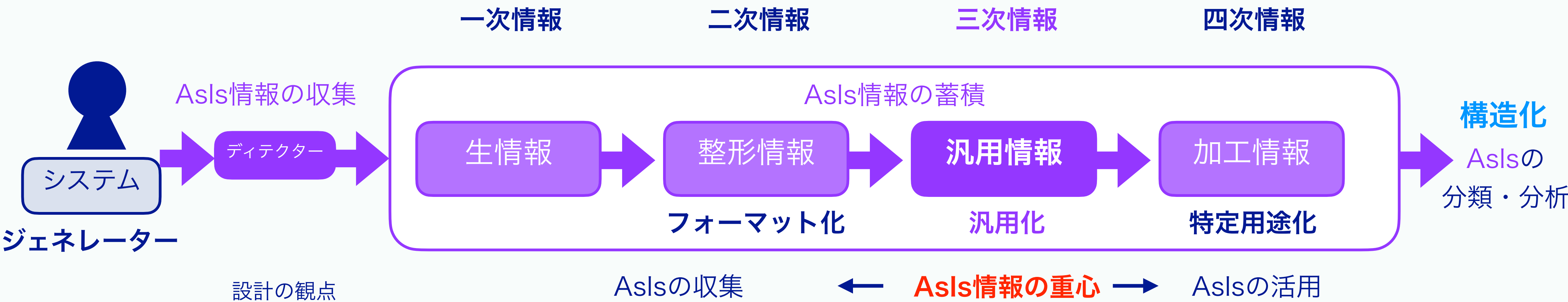
# Aslsインプットのトップダウンアプローチ設計



合致するケース      収集したい汎用情報が明確  
必要な一次情報が明確、収集方法も確定

運用に本当に必要な情報を、最短で収集できる体制になる。

# Aslsインプットのボトムアップアプローチ設計



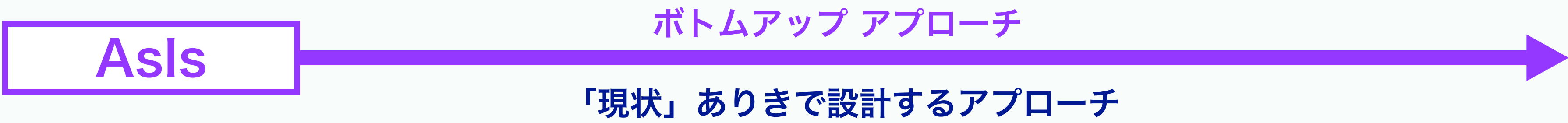
合致するケース

収集したい汎用情報が不明瞭。

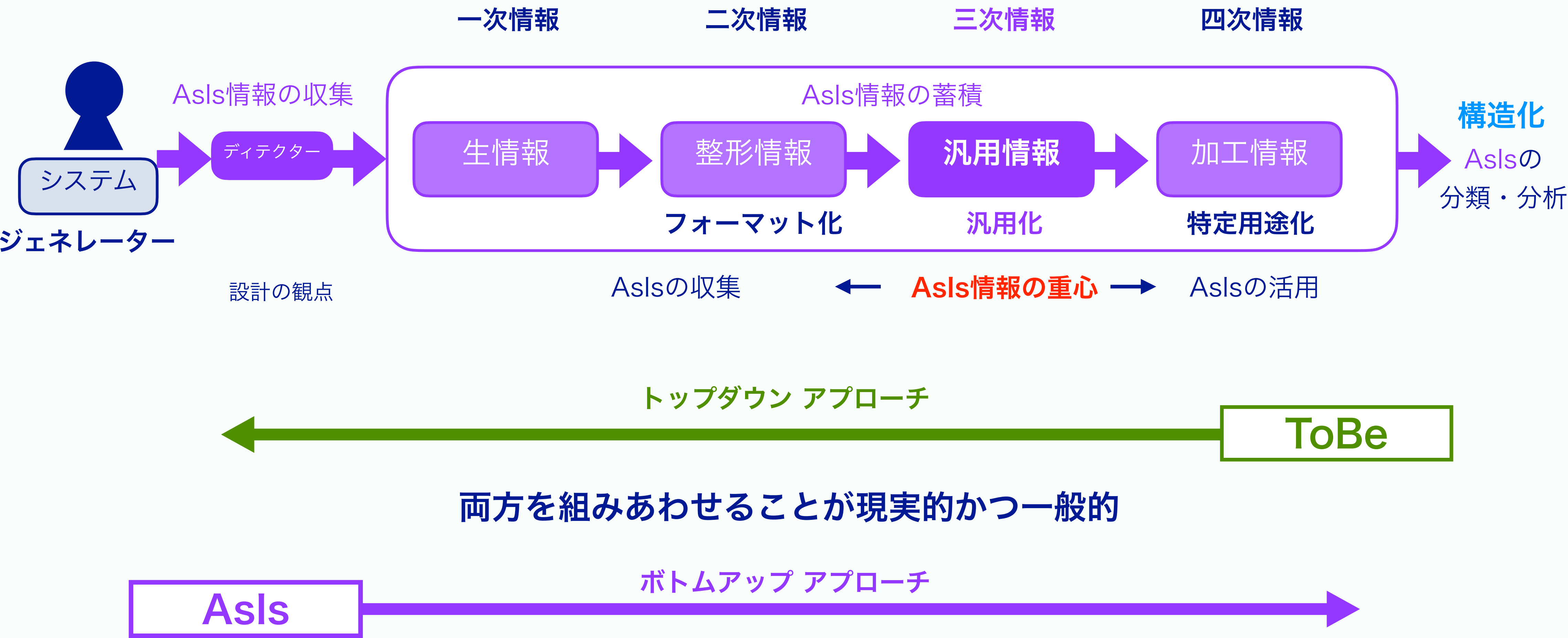
すでに収集できている、できる可能性が高い「定量的Asls情報」がある。

収集対象が「定性的Asls情報」の場合。

収集できる可能性の高い情報を、最短で収集できる体制になる。



# まとめ: Aslsインプットの設計アプローチ



本編 Aslsフレームワーク

## 講義のふりかえり

# Aslsフレームワークの基点 = インプット

インプット

Aslsの収集・蓄積

運用現場の現状を「ありのままに把握」するために

個人の「努力」や「気付き」だけに依存しないための  
**Asls収集・蓄積の「仕組み作り」が不可欠**

「自然とAsls情報が集まる」ための環境を実現する

# Aslsインプットの基本要素



# Aslsインプットの設計アプローチ

