

Dimensions of Software Configuration

On the Configuration Context in Modern Software Development

Norbert Siegmund, Nicolai Ruckel, and Janet Siegmund

In Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2020).

Association for Computing Machinery, New York, NY, USA, 338-349.

DOI:<https://doi.org/10.1145/3368089.3409675>

背景

- 様々な分野で「configuration」という言葉がある
 - テスト, 仮想化, ソフトウェアの設定, 機械学習
- Configurationに関する活動に明確な関連性がなかった
 - ↓ CI/CD, 仮想化とコンテナ, IaC, DevOps
- Configurationに関連性が出てきた
 - お互いの分野が高度に**相互作用**していること
 - configurationとは何かを包括的な視点で捉える必要がでてきた

問題点

包括的な視点がない場合,

configurationに関するソフトウェア工学の研究の
一般性や適用可能性等が損なわれる

従来

configurationに関して
孤立した見方



必要な視点

configurationに関して
包括的な見方

- 一般性にかける結果になってしまう

- 現実的なconfiguration問題に対応することができる

目的

どのような要素がconfigurationを構成し
どのように相互作用するのかを明らかにすること

- Configuration に関する研究の位置づけをするための
フレームワークを提供
 - 研究者が自分たちの作業や研究をコンテキストの中に位置づけ、
どのような要因を考慮する必要があるのか

研究の流れ

- フレームワーク作成
 - 開発者インタビュー + 2つの論文 [5][34]
- フレームワークをもとに評価したい論文ごとにモデルを構築する
- モデルの適応可能性を検証
 - 構成の側面に関連する異なる領域から16の追加研究論文（計18編）

[5] Thorsten Berger, Daniela Lettner, Julia Rubin, Paul Grünbacher, Adeline Silva, Martin Becker, Marsha Chechik, and Krzysztof Czarnecki. 2015. What is a Feature?: A Qualitative Study of Features in Industrial Software Product Lines. In Proc., Int'l Conference on Software Product Lines (SPLC). ACM, 16–25.
<https://doi.org/10.1145/2791060.2791108>

[34] Mohammed Sayagh, Noureddine Kerzazi, Bram Adams, and Fabio Petrillo. 2018. Software Configuration Engineering in Practice: Interviews, Survey, and Systematic Literature Review. IEEE Transactions on Software Engineering 42, 6 (2018), 646–673.
<https://doi.org/10.1109/TSE.2018.2867847>

フレームワーク | 作成

- インタビュー (11名)
 - configurationに関する一般的な質問
 - 日常的なconfigurationに関連する作業
 - ベストプラクティスと改善のための領域
- 論文
 - 「feature」という言葉の意味, 使い方, 解釈についての研究 [5]
 - 異なるソフトウェア成果物にまたがる構成オプションの作成・維持・および使用に関する課題とベストプラクティスを明らかにした研究 [34]

フレームワーク | 8次元47要素

意図

Intent

Functional

Non-functional

Unknown Environment

Distributed Development

Reduced Testing Effort

Knowledge Preservation

Code Reuse

A/B Testing

...

開発工程

Stage

Prod

Pre-Prod

Test

Dev

Technical

Infrastructure

Development

種類

Type

Domain

Build

Deployment

Load Time

Run Time

束縛時間

Binding
Time

Create

Maintain

Bind

Own

Deprecate

Remove

Life Cycle

複雑さ

Complexity

High dependency

Low dependency

No dependency

Distributed Scope

Local Scope

Devs

Ops

DevOps

Technical User
End User

Stakeholder

利害関係者

ライフサイクル

Artifact

成果物

Source Code

Config Code

Config File

Environment Variable

Database

Preprocessor Code

Directory Service

Feature Model

Spreadsheet

Product Map

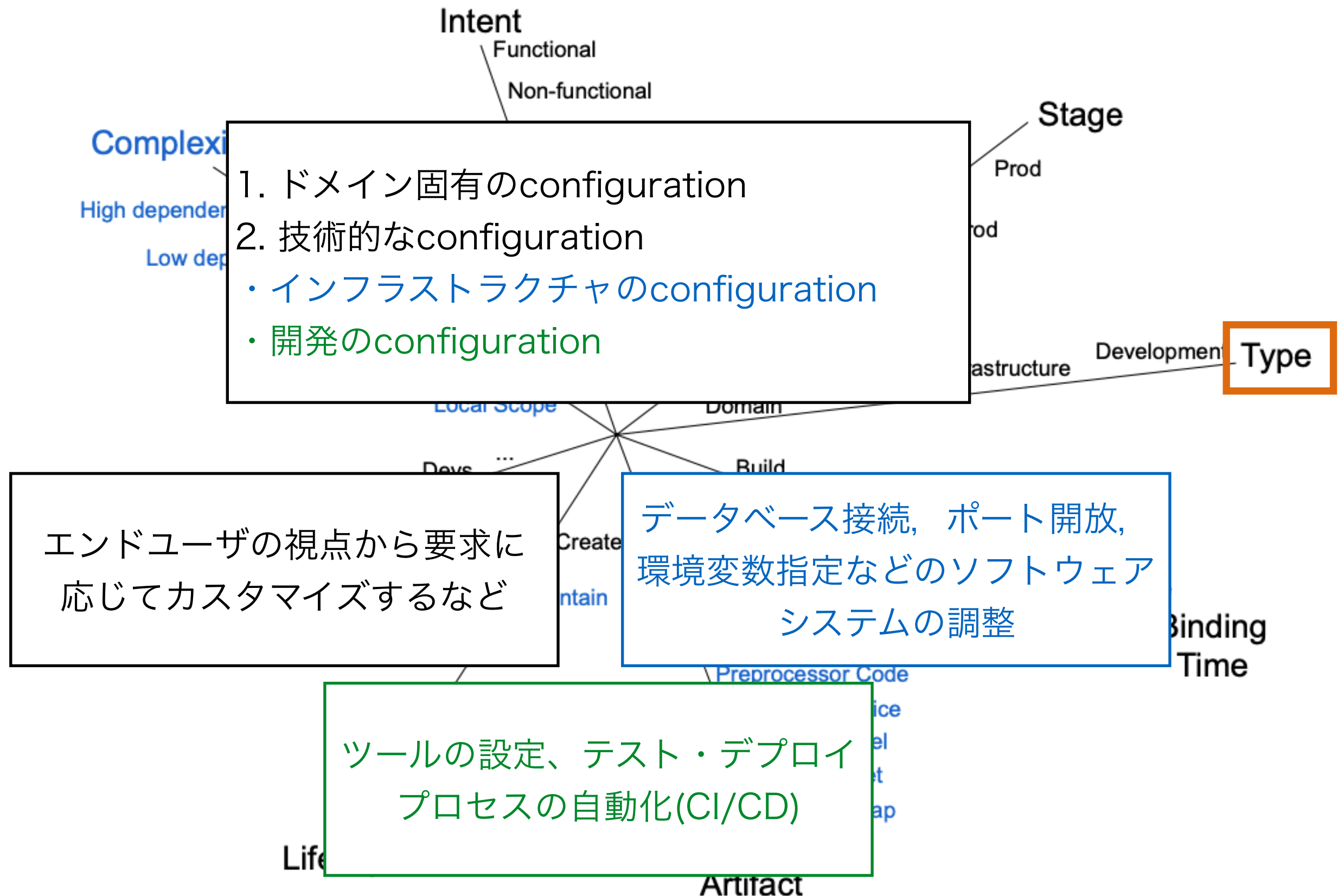
GUI

黒：インタビュー

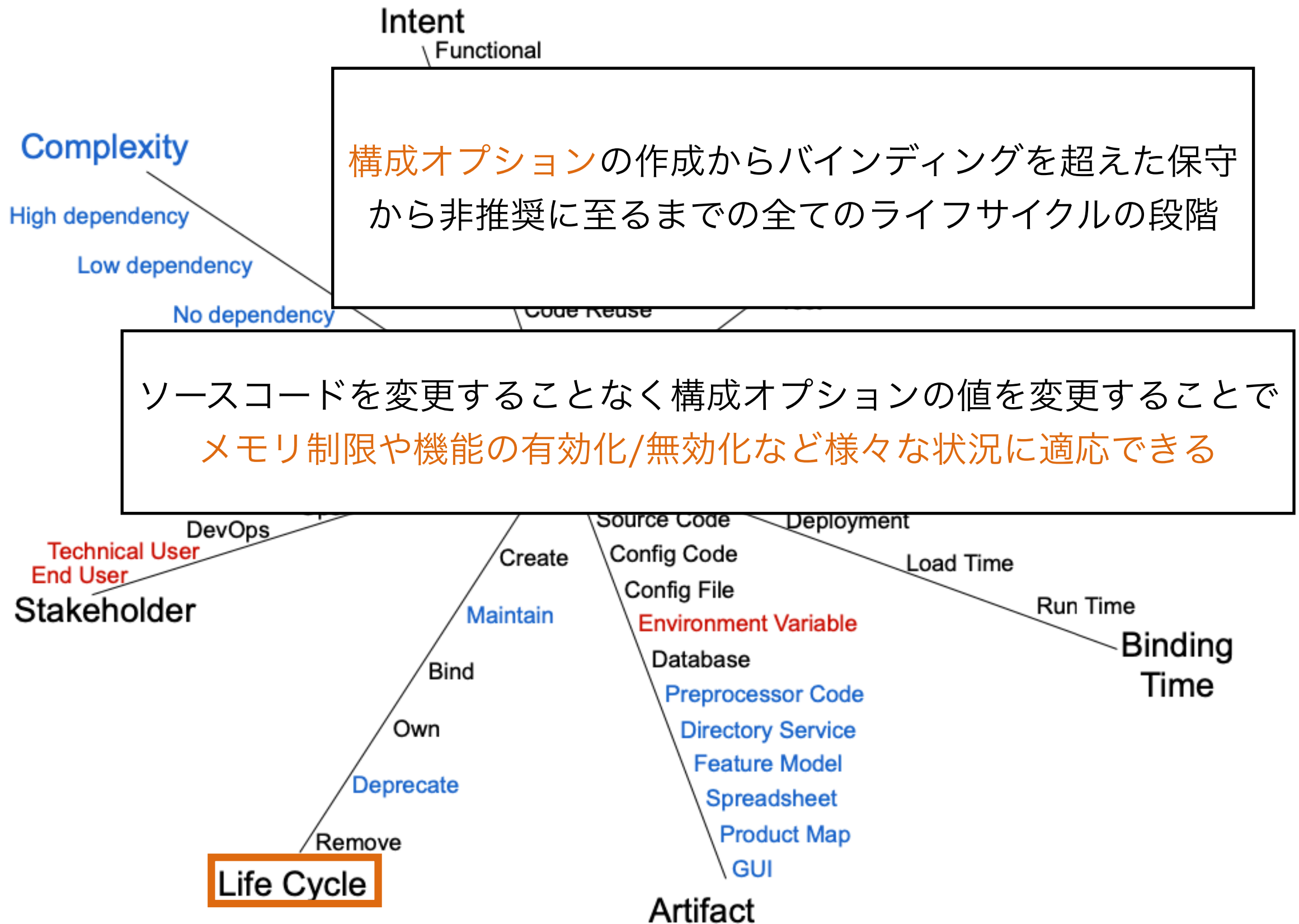
青：[5]

赤：[34]

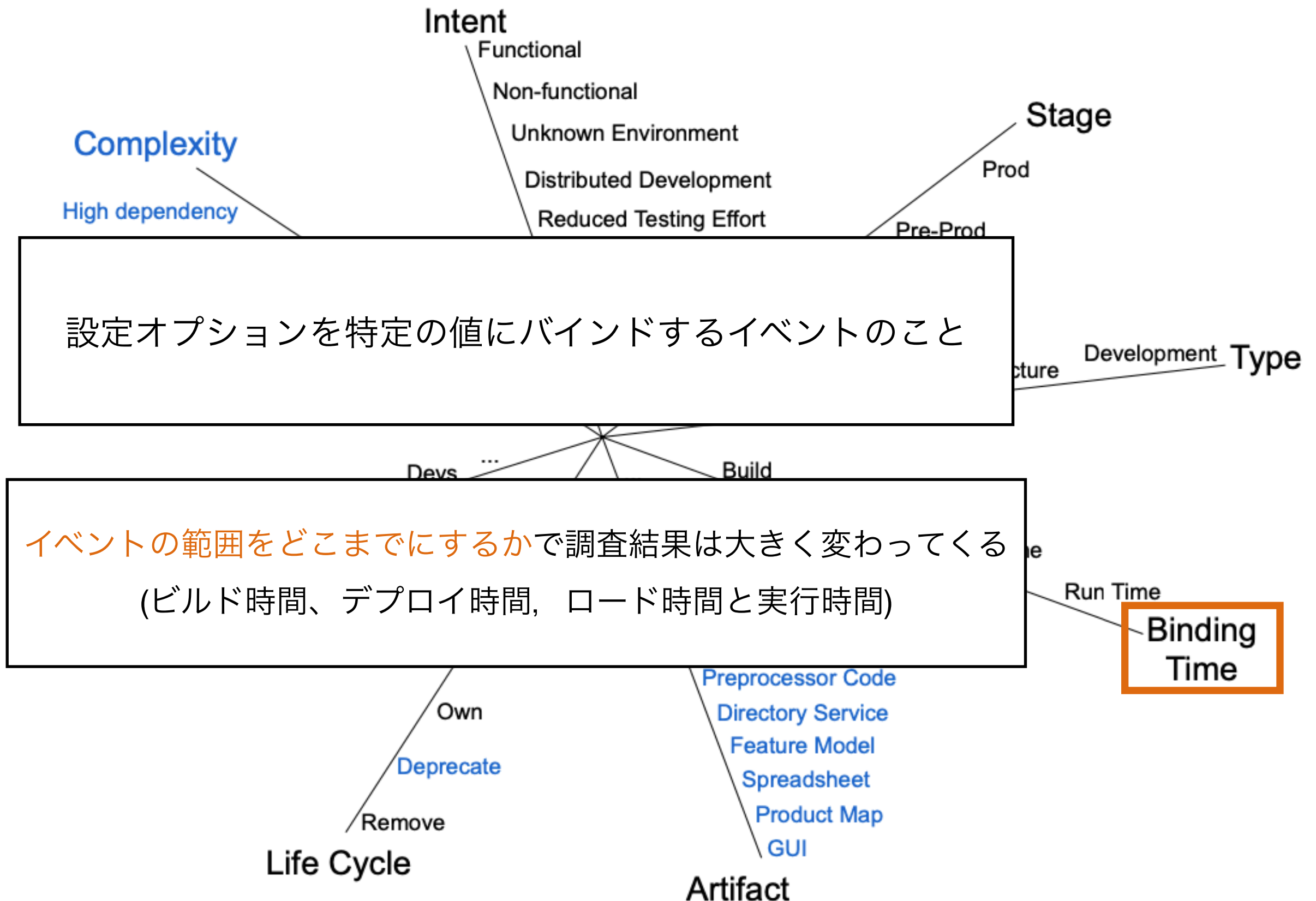
フレームワーク | Type



フレームワーク | Life Cycle



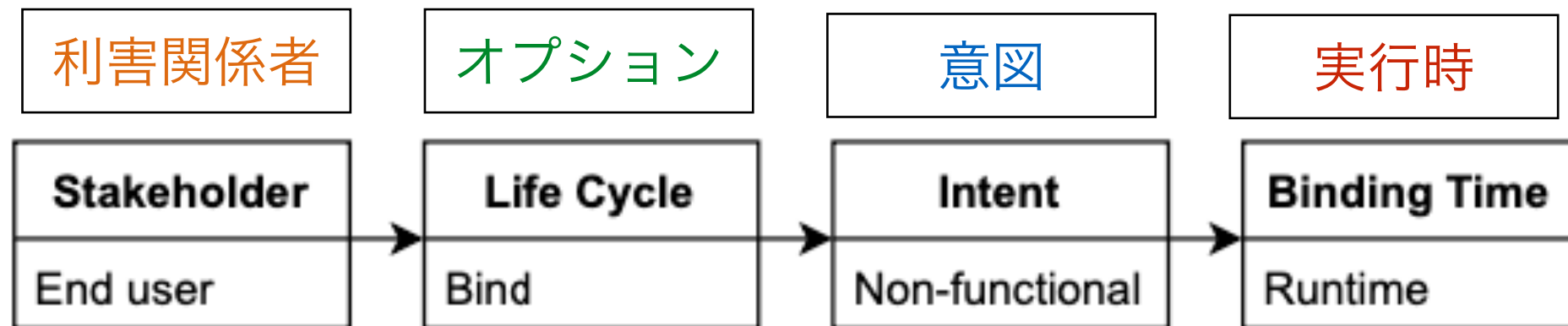
フレームワーク | Binding Time



フレームワーク | モデル適応例

論文内容

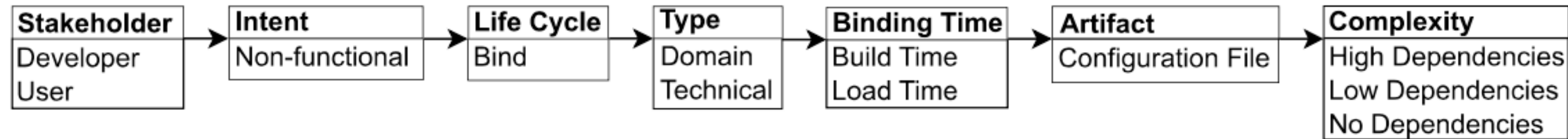
性能最適化を目的としエンドユーザがどのようにオプションを設定しているかを実行時のオプションのみで分析した研究



- 他の4次元について言及されていないことがわかる

configurationを含む研究における
曖昧さや欠陥した次元を特定することに役立つ

モデル検証 | パフォーマンス最適化の研究 [22][26]



- Stage次元がない
 - どの段階のパフォーマンスが最適化されるのかが不明
- infrastructureタイプがパフォーマンスに影響を与えるにも関わらず考慮されていない
 - それに伴うconfigurationアーティファクトも考慮する必要がでてくる

18の論文を用いて検証を行い

曖昧さや欠落する次元の特定ができた

まとめ

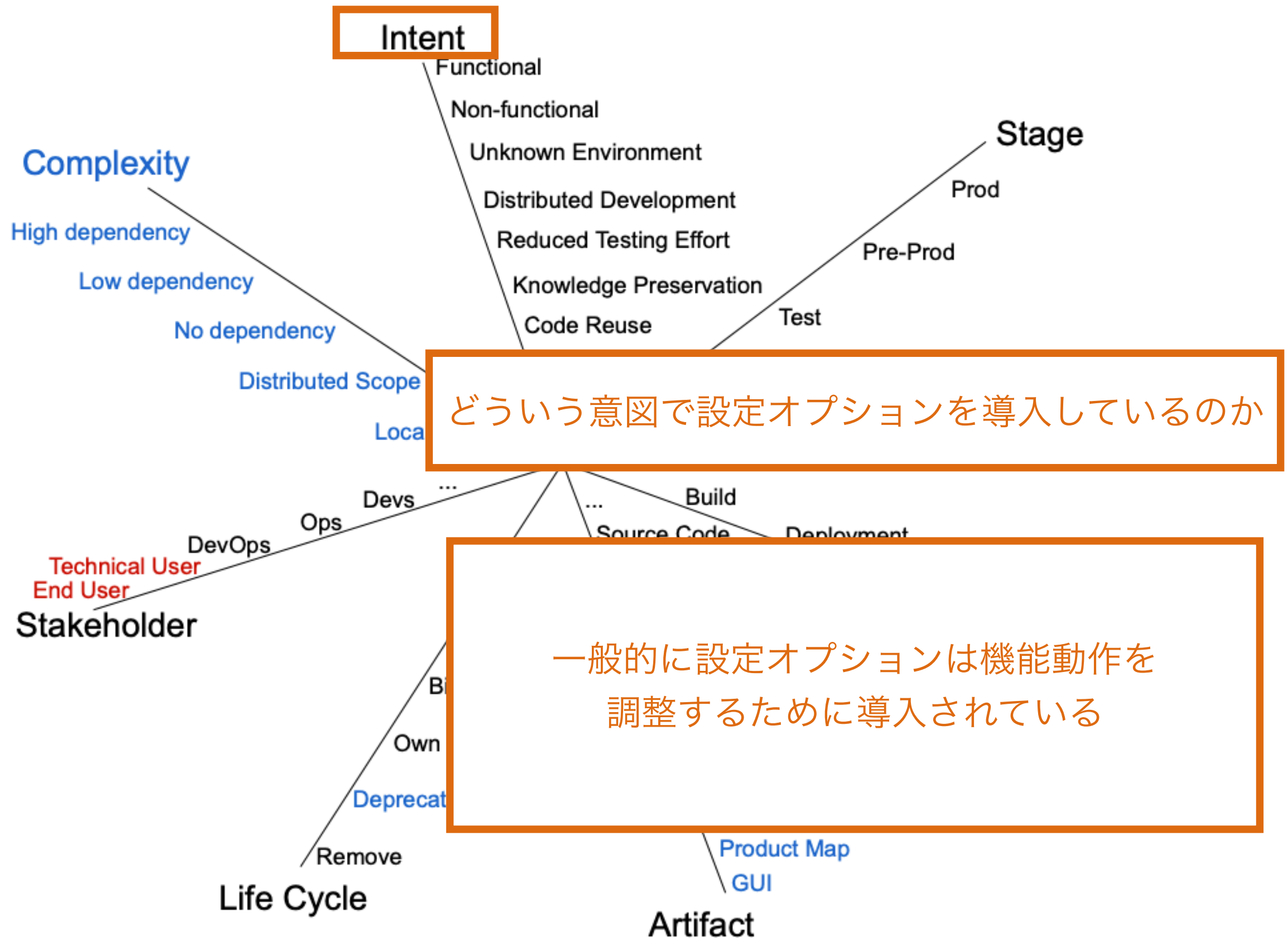
- Configurationの定義等が曖昧で関連する研究の一般性が損なわれている
- Configurationに関する研究の位置づけをするためのフレームワークを提供
 - 8つの次元, 47の要素
- フレームワークからモデルを作成することで, 研究の曖昧性や欠落している次元を特定できる

所感

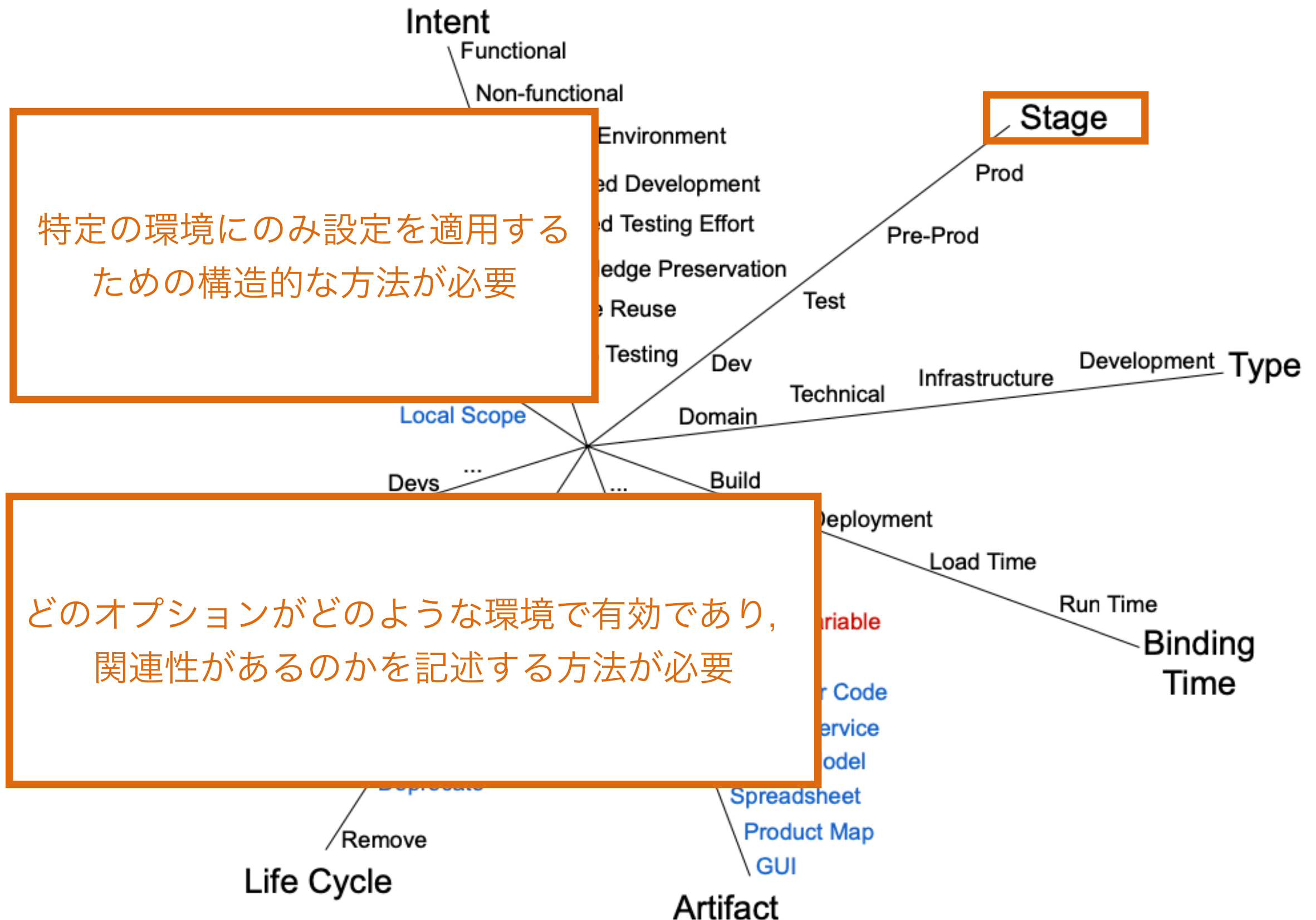
- 構成オプションという言葉は初めて聞いた
- Configurationという言葉は多くの分野で耳にするため
- 他の研究から言葉が多く引用されていたり，追加資料等が多く，理解するには時間が必要であると感じた。

補足

調査結果 | Intent



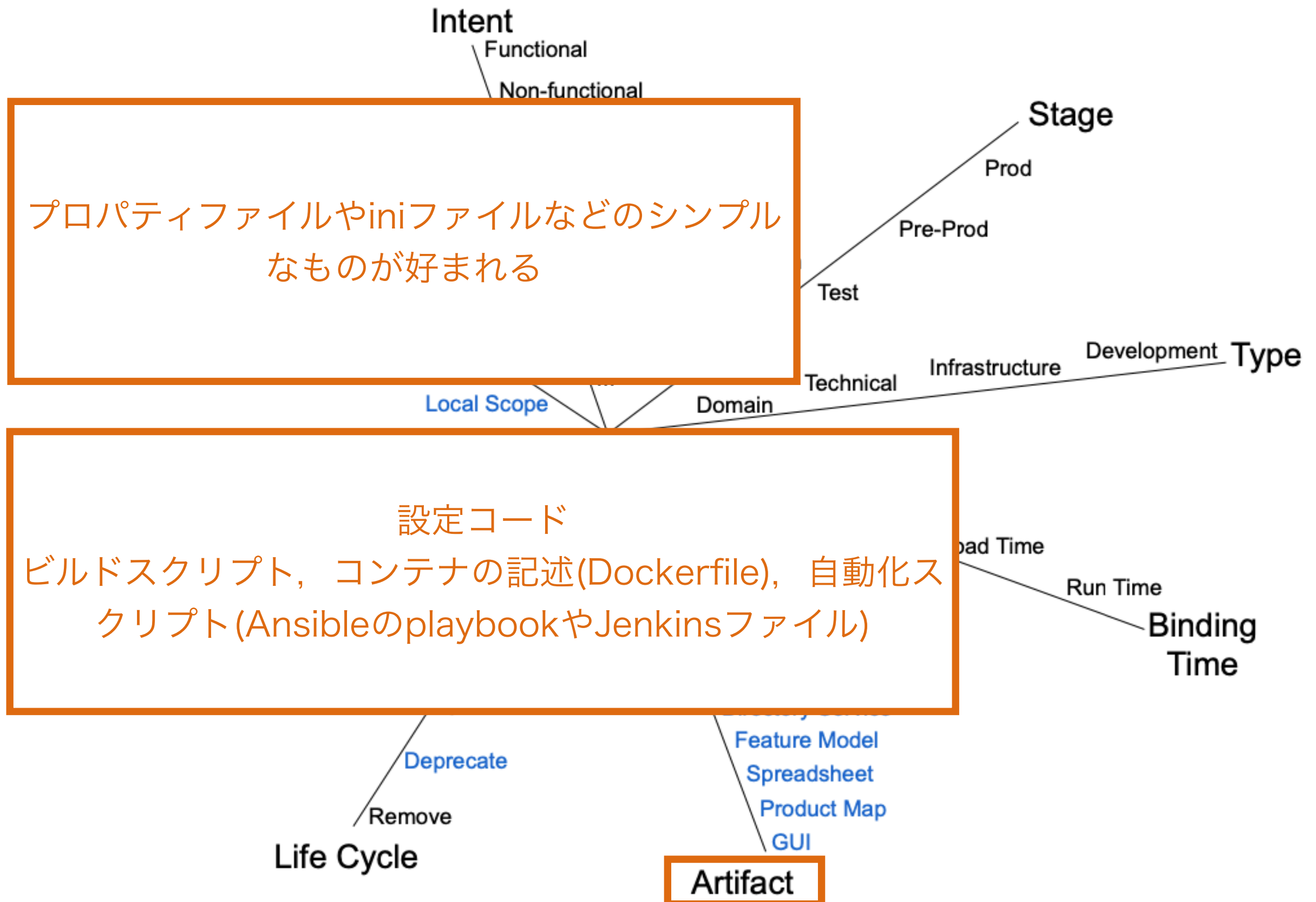
調査結果 | Stage



特定の環境にのみ設定を適用するための構造的な方法が必要

どのオプションがどのような環境で有効であり、関連性があるのかを記述する方法が必要

調査結果 | Artifact



調査結果 | Stackholder

