



NTT DATA

AWS Summit Japan 2026 ~Day2~

開発と運用の壁を越える実現場でのAI活用術

2026/06/26

ソリューション事業本部
クラウド&データセンタ事業部
長谷川 樹

自己紹介



■ **所属：** 株式会社NTTデータグループ
ソリューション事業本部 クラウド&データセンタ事業部

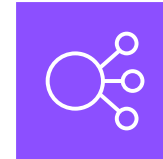
■ **入社：** 4年目

■ **業務内容：** クラウド案件の開発
→ AI活用の施策にも携わる。

■ **好きなAWSサービス**



AWS Lambda



Elastic Load Balancing



目次

1. 昨今のAI・システム開発事情
2. プロジェクトでの取り組み
3. NTTデータの取り組み
4. まとめ



目次

1. 昨今のAI・システム開発事情

2. プロジェクトでの取り組み

3. NTTデータの取り組み

4. まとめ

突然ですが...

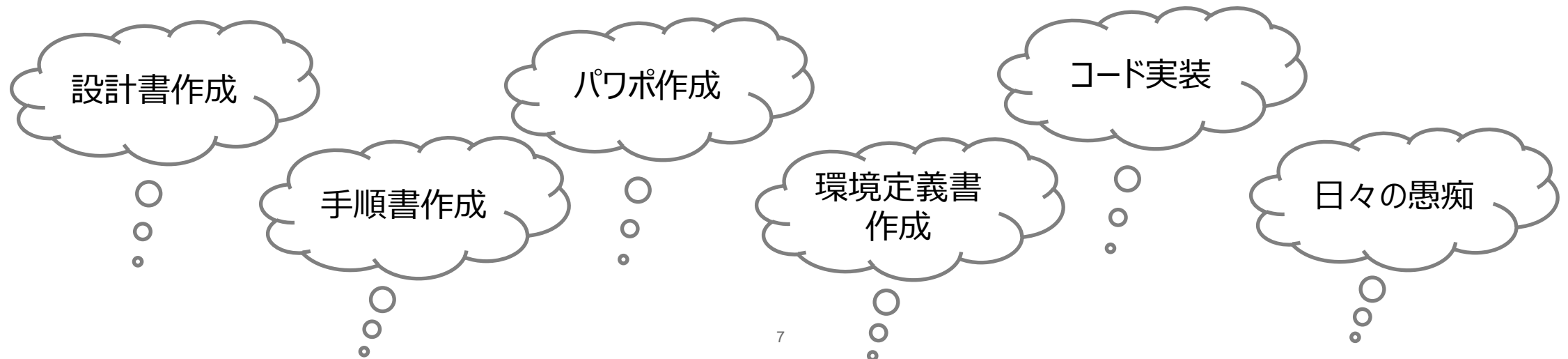
皆さんは業務の中で、AIを利用されていますか？？

使っているそのあなた！

どんな場面でAIを利用されていますか？

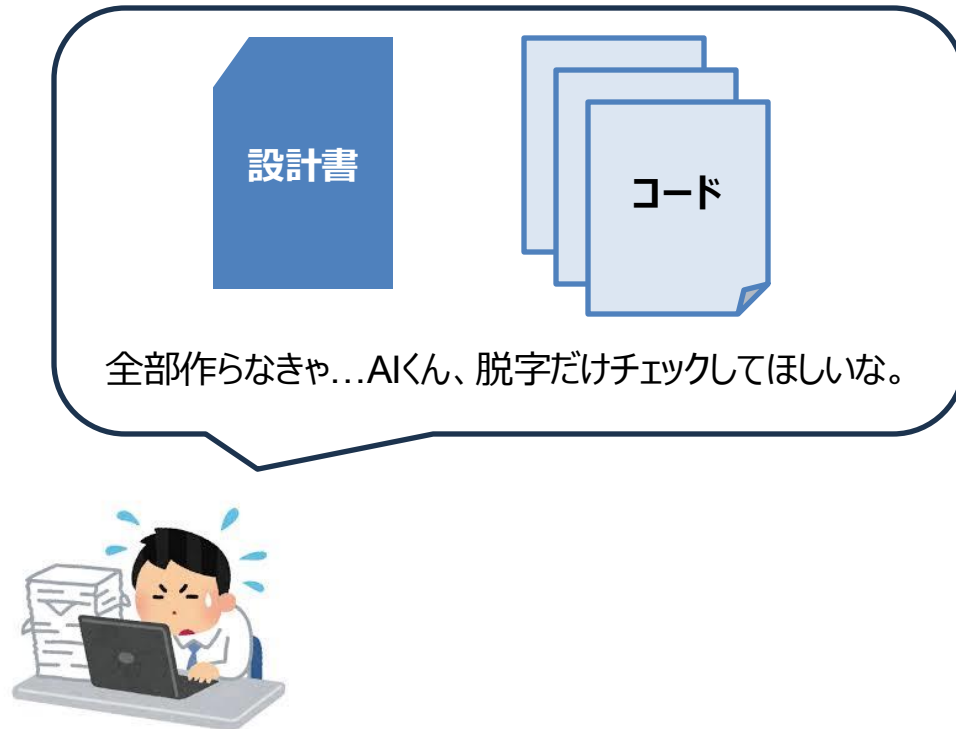
使っているそのあなた！

どんな場面でAIを利用されていますか？

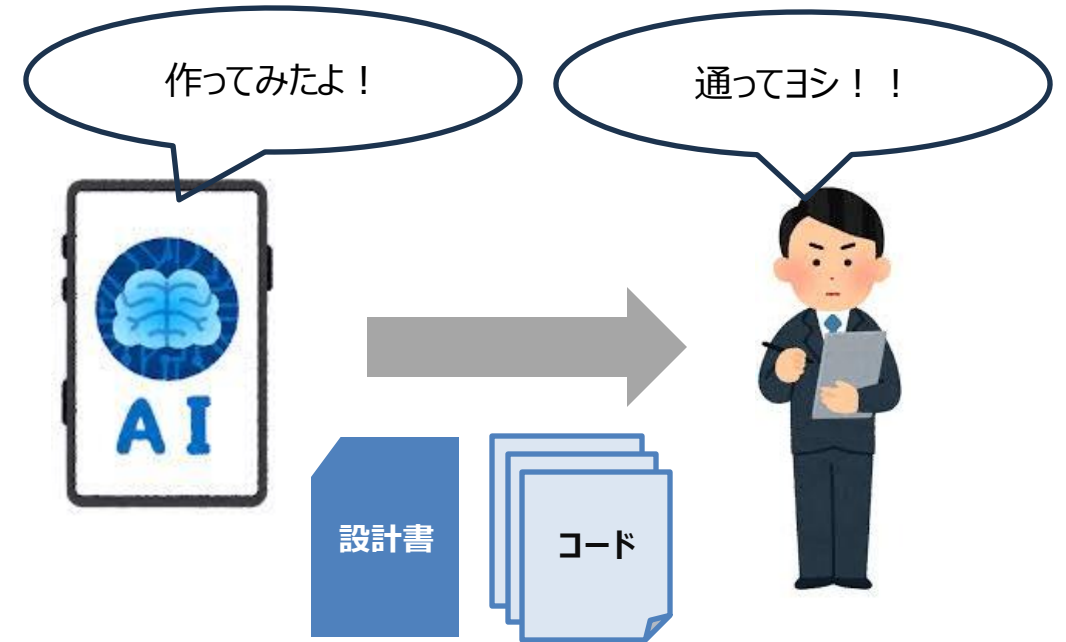


「AIが人の実装のお手伝いをする時代」から「AIがタスクをこなし、人がフォローする」時代へ。

従来の開発



AI-Nativeな開発



人の役割は「実装者」から「レビュアー・責任者」へ

昨今のシステム開発の問題

継続的な短納期での開発、最小構成で構築したシステムの機能拡張による負債が課題。



アジャイル開発によりビジネス価値を継続的に産出していくことが主流。

問題①：短納期開発の継続性

ビジネス拡大により機能追加のペースが速くなり、開発メンバーが疲弊してしまう…

→ 品質の低下・有事の対応遅延につながる。

問題②：システムの機能拡張による負債

機能を細かく構築したことにより、システムが大きくなった際に最適なシステムではなくなってしまう。

→ 運用負荷の増加・コスト増加につながる。

AIを用いて積極的に課題解決を行っている。

問題①：継続的な短納期での開発

問題②：システムの機能拡張による負債

4つの武器

AI活用力

AI基盤

クラウドの
開発・運用力

問題解決力

AI×仕様駆動開発による開発速度UP

IaCコードのリファクタリングによる運用負荷軽減



目次

1. 昨今のAI・システム開発事情
- 2. プロジェクトでの取り組み**
3. NTTデータの取り組み
4. まとめ

AIを用いて積極的に課題解決を行っている。

問題①：継続的な短納期での開発

問題②：システムの機能拡張による負債

4つの武器

AI活用力

AI基盤

クラウドの
開発・運用力

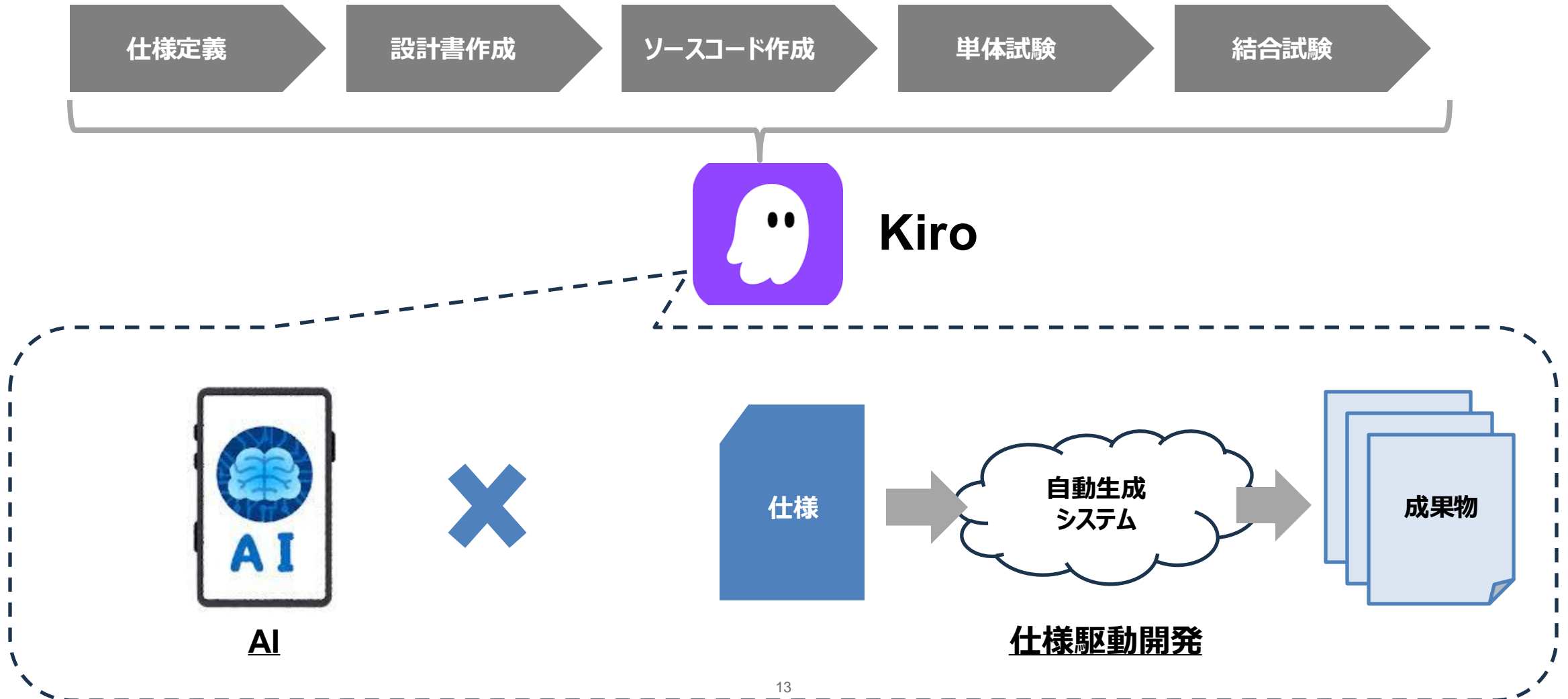
問題解決力

AI×仕様駆動開発による開発速度UP

IaCコードのリファクタリングによる運用負荷軽減

「継続的な短期開発」へのアプローチ

Kiroを用いた仕様駆動開発により、品質を下げずに高速でAPI開発を行う。



「継続的な短期開発」へのアプローチ

仕様駆動開発にあたり、成果物の品質保証が難しい。

仕様駆動開発の概要



仕様書をAIにインプットすることで開発全体をAI駆動で推進

AIを使いつつ、品質を上げて、手戻りをなくすことで、高速開発が実現できる！

Kiroの概要

AWSが提供するLLM付きの新しい統合開発環境
(2025年7月～ 提供開始)

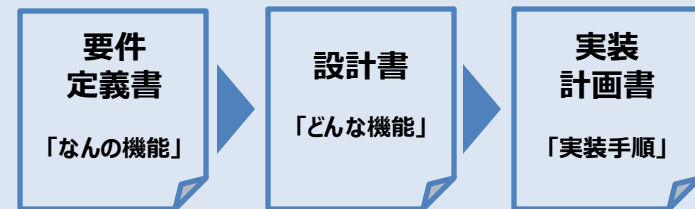
Vibeモード

AIとの対話によりアイデアを
即時作成するモード

→ 局所的な開発・変更向け。

Specモード

以下のドキュメントを作成したうえで、
コードを実装するモード



仕様駆動開発の難しいポイント： AIの成果物をどのように品質保証するかが難しい。

「継続的な短期開発」へのアプローチ

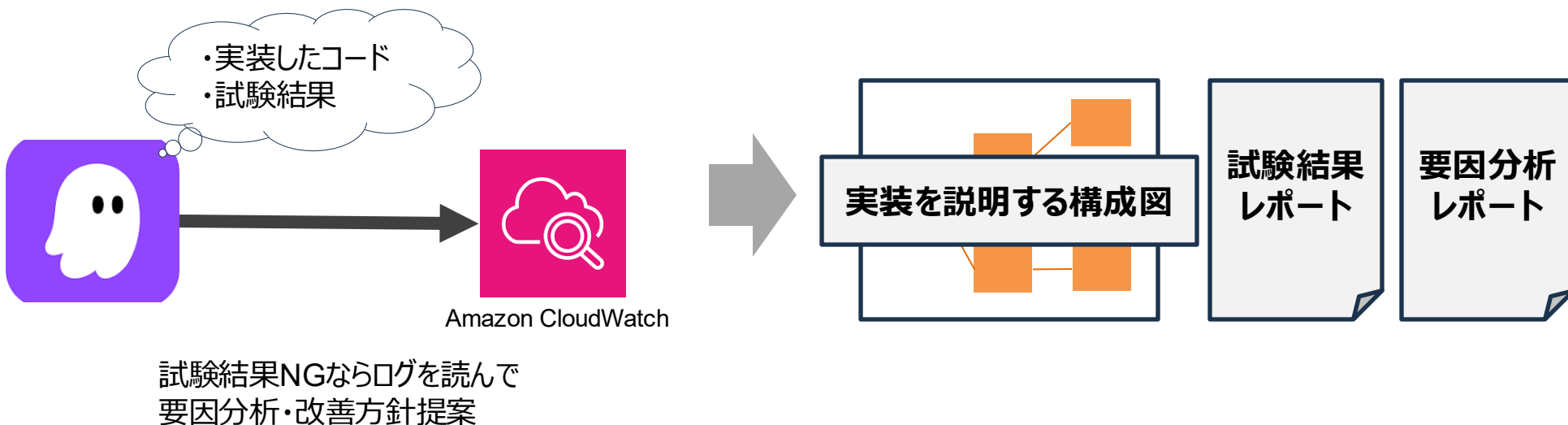
ポイント①：各開発工程のレポート化により人によるレビュー負担を軽減。

課題①

構築や変更がKiroによって効率化されると、「有識者によるレビュー」がボトルネックになる。

解決策

試験実行の都度、結果・要因分析・対応方法案をレポートとして出力させて、
要因分析の結果やネクストアクションの検討を効率化。



「継続的な短期開発」へのアプローチ

ポイント②：設定ファイルをKiro自身に継続的に修正させて品質を担保。

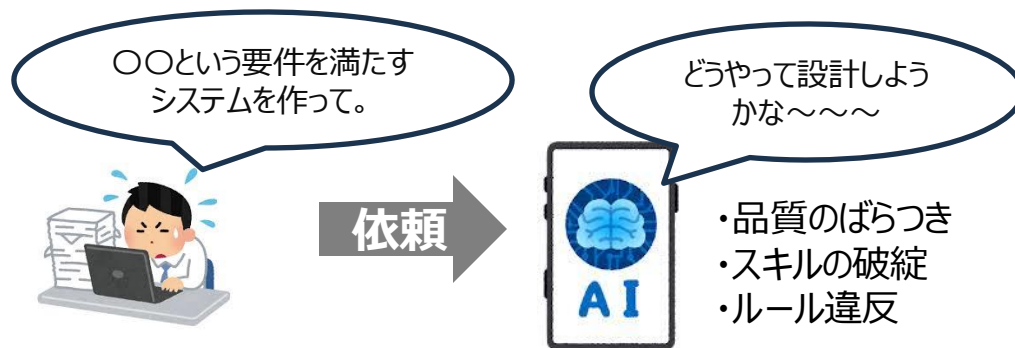
課題②

成果物の品質担保のために制約やルールを詰め込みすぎると、コンテキストがあふれて、逆に遵守されなくなるため、ルールの選別やインプットの与え方の工夫が必要。

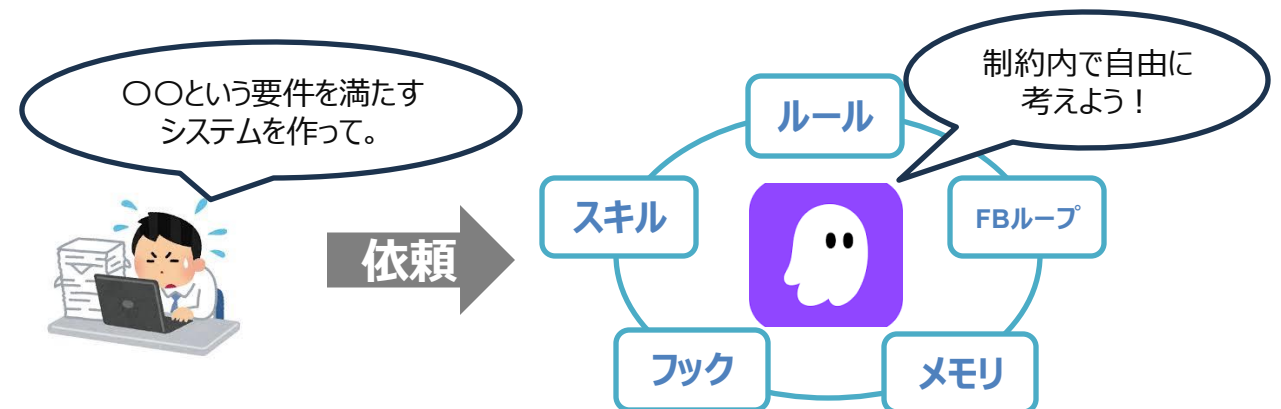
解決策

SkillsファイルやSteeringファイルなどの設定ファイルを、継続的にKiroにアップデートさせ続けることで、再現性高く、成果物の品質を担保。

従来のコンテキストエンジニアリング



ハーネスエンジニアリング



「継続的な短期開発」へのアプローチ

ポイント③：試験項目の品質担保のためにプロパティベーステスト&自己フィードバック。

課題③

- ・試験項目の網羅性の担保が難しい。
- ・試験項目の準備にも時間がかかってしまう。

仕様を満たしているかを検証するために
Kiro自身が自動で生成。

解決策

- ・**プロパティベーステスト**により、網羅性向上 & 試験のサンプル準備を容易に。
- ・作成後の項目の抜け漏れを自身で分析させ、FB結果を踏まえて再度確認を行うことで網羅性向上。

Event Based Test

特定の例をサンプルとして用意してテスト。

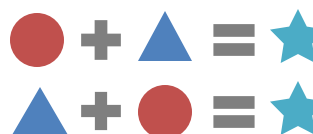
例：
 $1 + 2 = 3$
 $100 + 1000 = 1100$
 $1 * 0 = 0$

→ 事前に想定できたものしかテストできない。

Property Based Test

仕様による普遍性を使って、任意の入力でテスト。

人が想定していなかった入力やバグを洗い出すのが得意。

例：


答えは変わらないはず

→ 多くの入力で検証可能。

「継続的な短期開発」へのアプローチ

工数削減に成功し、開発メンバーの負荷軽減に貢献。



Kiro

✓ AIを用いた開発高速化

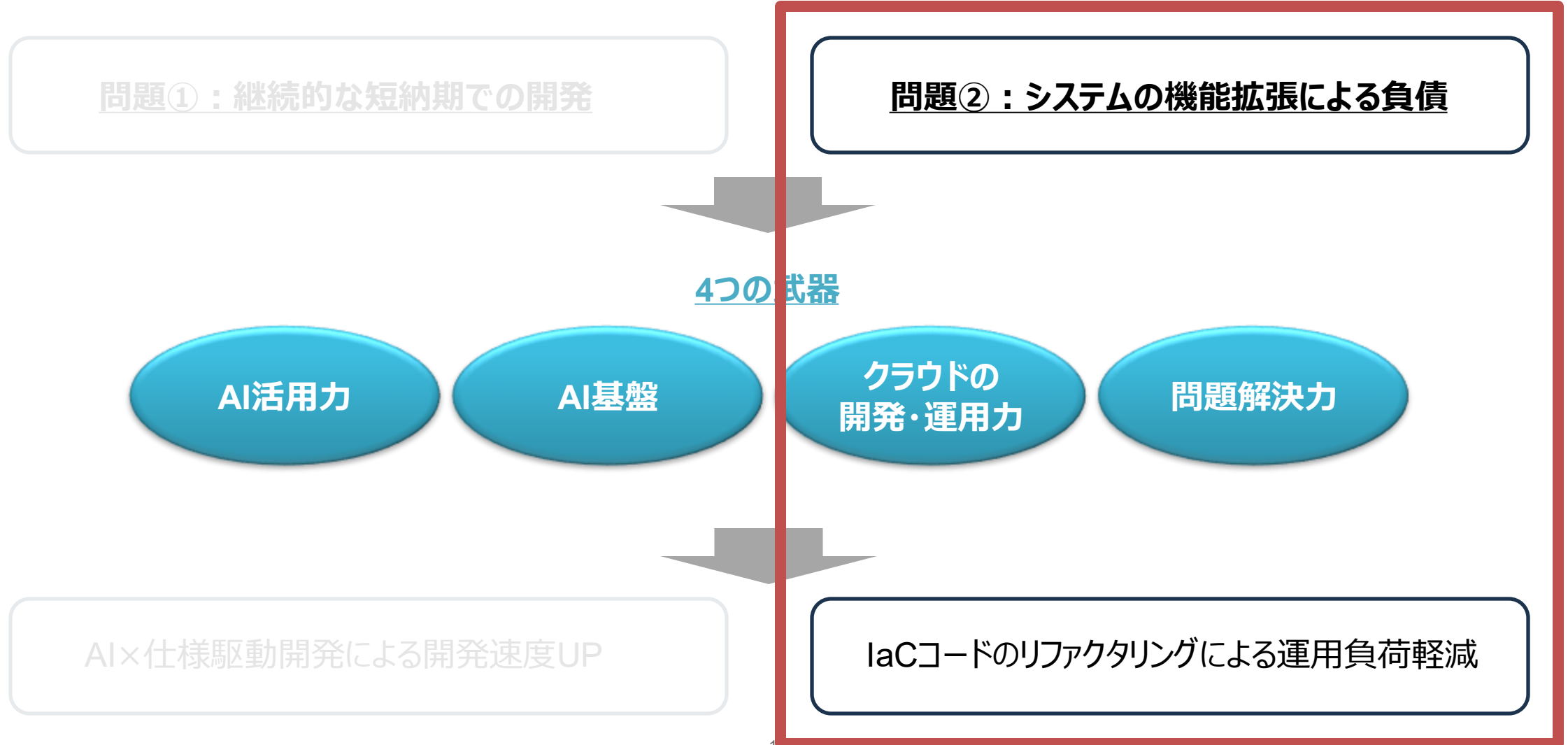
✓ 仕様駆動開発を用いた品質担保による手戻り工数の削減



少なくとも実装・試験以降のフェーズにおいて、

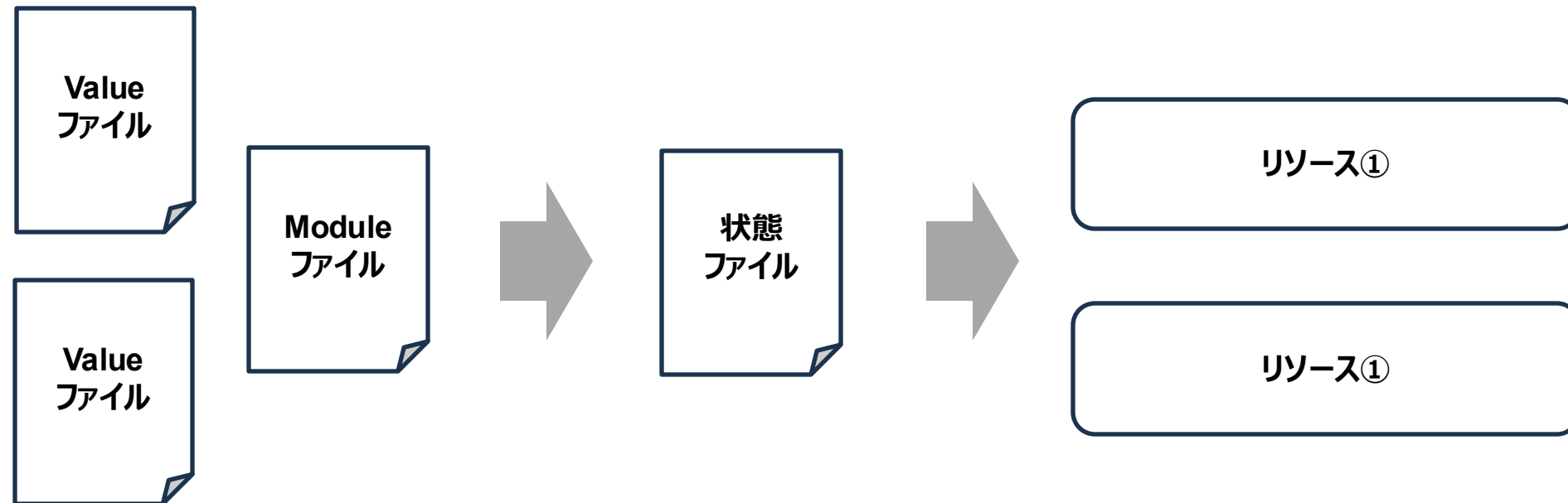
品質を落とさずに、工数の大幅削減を実現！

AIを用いて積極的に課題解決を行っている。



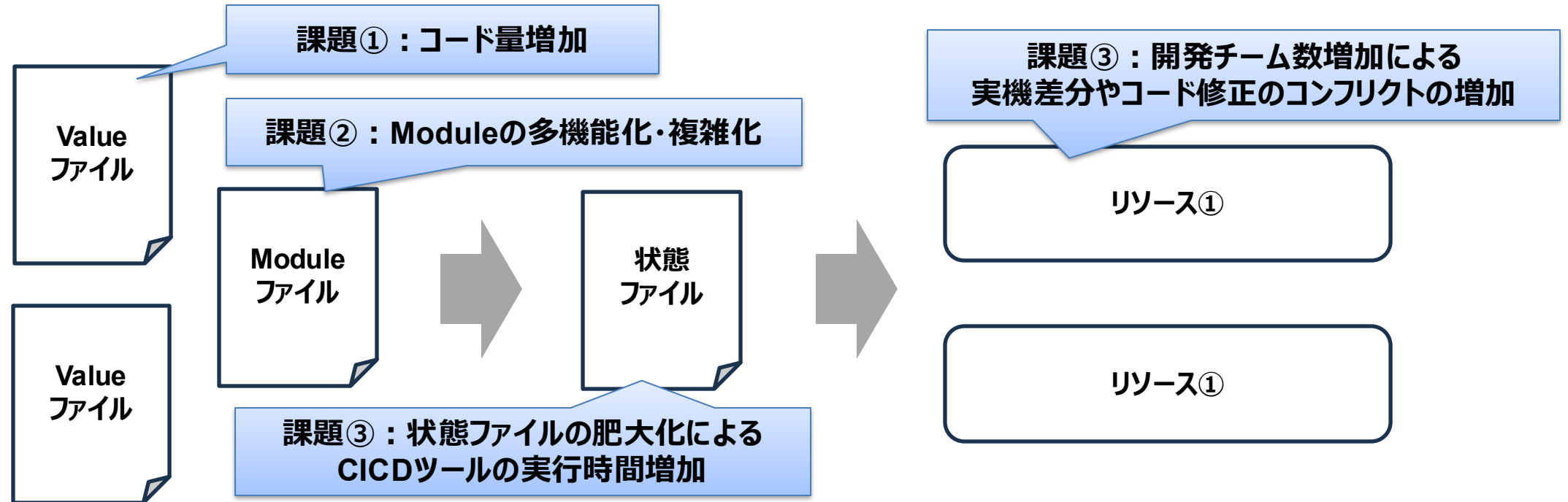
「機能拡張による負債」へのアプローチ

機能追加が多くなるとIaCコード量や状態ファイルの大きさが増加する。



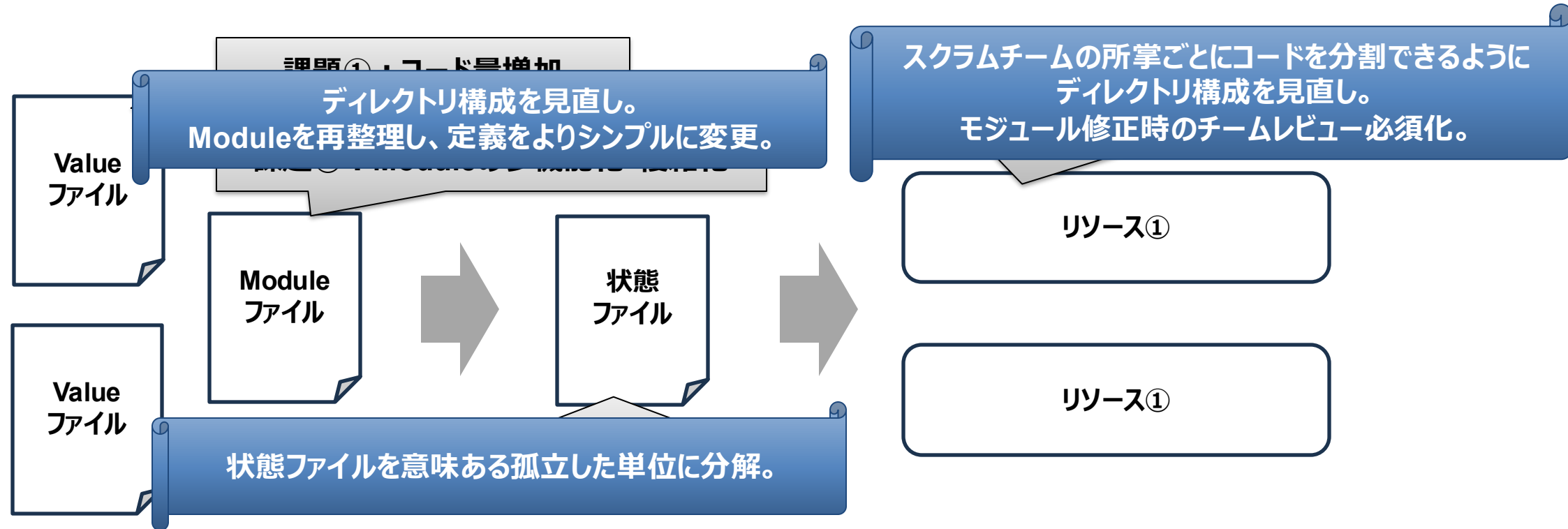
「機能拡張による負債」へのアプローチ

laCコードの可読性低下や運用負荷の増加を招く。



「機能拡張による負債」へのアプローチ

肥大化したIaCコードをAIを活用してリファクタリングして、負債を解消。



工夫点

- ✓ import/move機能により **既存リソースに影響を与えずに** リファクタリング。
- ✓ リファクタリングを実行できるスクリプトを作成させることで **トークン消費削減&オフラインでもリファクタ作業可能** な状態へ。
→ **メンテナンスはAIにさせる** ことで運用負荷も軽減。



目次

1. 昨今のAI・システム開発事情
2. プロジェクトでの取り組み
- 3. NTTデータの取り組み**
4. まとめ

顧客特性を踏まえたうえで、AIを駆使してクラウド開発を支援しております。

4つの武器



開発

Kiroを用いた仕様駆動開発

仕様駆動開発により、高速な追加機能開発に対する人的負荷を軽減。

→ ボトルネックになりがちな成果物の品質保証や品質向上のためのチューニングを、Kiroの機能を用いて改善。

運用

laCコードのAIによるリファクタリング

肥大化したlaCコードをリファクタリングことで運用負荷を軽減。

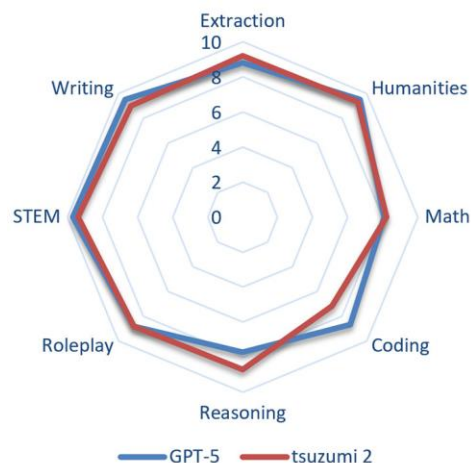
→ 人の手で実行するには大変なリファクタ作業をAIで実施させることで、現実的な作業工数に抑えることに成功。

TsuzumiやGPUaaSといったプライベートな国内AI基盤でセキュアなAI活用を促進。

Tsuzumi 2

1GPU動作可能な軽量モデル

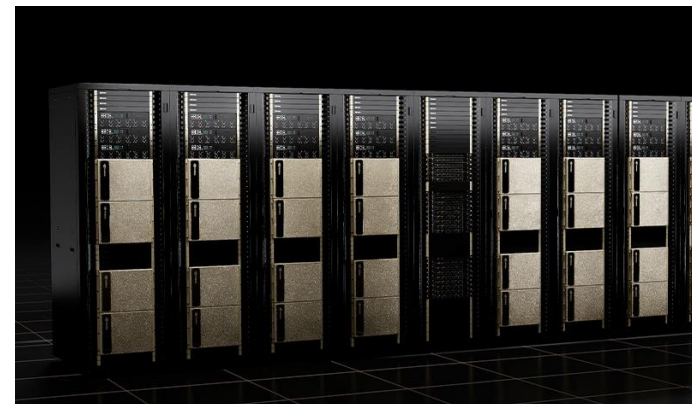
→ 法人のお客様に求められる
タスク処理能力の強化・知識の増強



GPU as a Service

NVIDIAアクセラレーテッドコンピューティングを活用した
大規模機械学習向け基盤提供サービス

→ 大規模言語モデル（LLM）開発のほか
膨大なデータの活用を支援

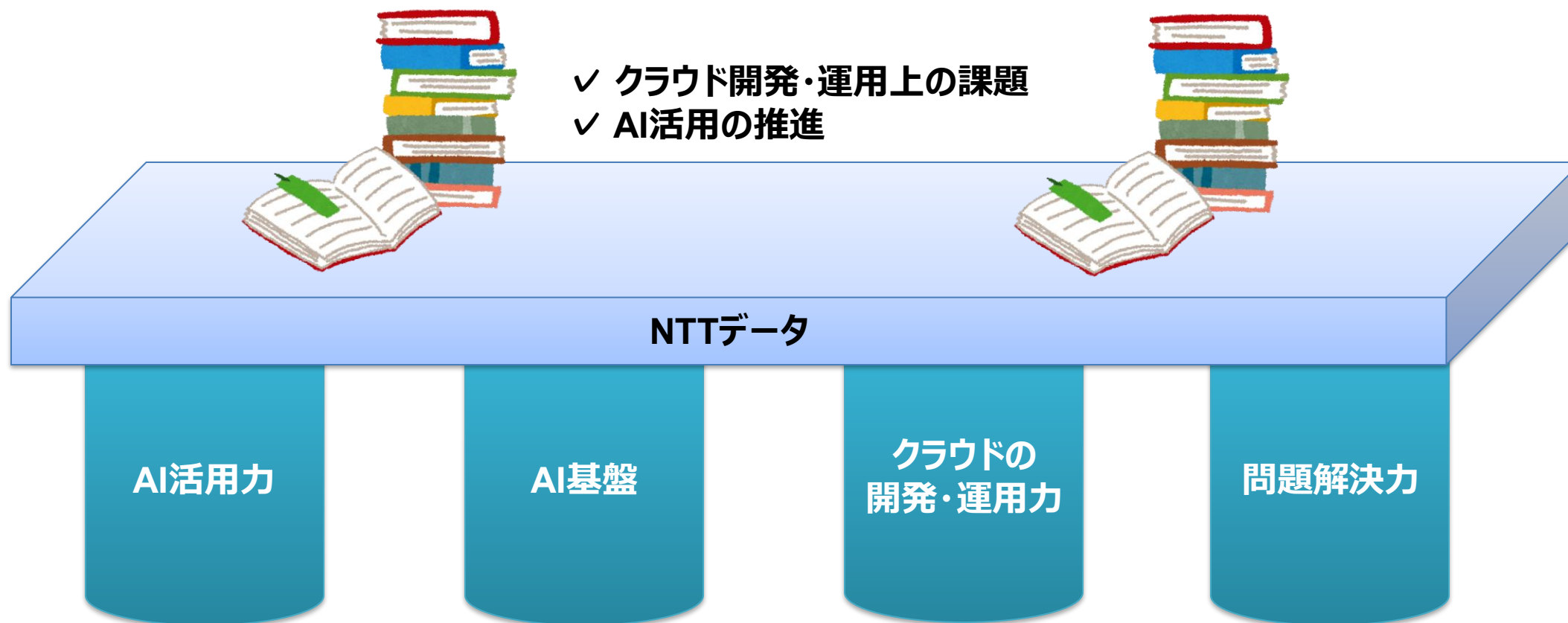




目次

1. 昨今のAI・システム開発事情
2. プロジェクトでの取り組み
3. NTTデータの取り組み
- 4. まとめ**

クラウド開発やAI活用でのお悩み事、ぜひご相談ください。



長きにわたるクラウド・システム開発での知見・問題解決力を生かし、あなたのお困りごとを解決します！！

The image features a low-angle, wide shot of a modern city skyline under a clear blue sky. Two prominent skyscrapers with white and blue facades dominate the center. Other buildings of varying heights are visible in the background and foreground. The scene is captured in a cinematic style with a slight color grade. The text 'NTT Data' is superimposed in the center in a bold, white, sans-serif font.

NTT Data

補足

「継続的な短期開発」へのアプローチ

ポイント：コンテキストの引継ぎをSteeringファイルを用いて実施。

課題

作業者間での作業引継ぎや複数人での作業において、作業背景やトラブル事例が含まれているため、同様のコンテキストで作業したいが、Kiro IDEでは標準でコンテキストとのエクスポート機能がない。
※CLIのみ現状存在する。

解決策

一時的にSteeringファイルとしてコンテキストを書き込んでもらい作業者間で共有することで、コンテキストを引き継ぎ、品質を均一化。

