開発標準の作成における DevOps
～開発標準のインナーソース開発事例～

DevOps for develop standardized methodology.
Case of inner source development for standardized.

株式会社 NTT データ 熊川 一平
Ippei.Kumagawa@nttdata.com
NTT データ 技術革新統括本部 システム技術本部 生産技術部 ソフトウェア技術センタ

発表要旨：

ソフトウェア開発を成功に導くために，技術者の能力の均質化や底上げ，開発プロジェクト内での共通認識の醸成を目的として，開発標準などのドキュメントを作成している企業は非常に多い．しかし，ソフトウェア開発プロジェクトは複雑であるため，開発標準に記すべきノウハウは多岐にわたる．そのため開発標準が重厚長大なドキュメント群となってしまうことが多い．多くの技術者は多忙な開発の中で重厚長大な開発標準を読むこととなり，大変な苦労を感じている．結果として開発標準は技術者から「役に立たない」・「読みたくない」といった評価を受けることがある．また，ソフトウェア開発に関する技術やテクニックは，常に進化しており，開発標準も進化に追従していかなければない．しかし，重厚長大なドキュメントを修正するには，執筆やレビューの工数が負担となり，迅速な改善が難しい．その結果，開発標準に記されているノウハウが，時代遅れとなってしまうことも珍しくなく，技術者からの評価がより悪化してしまう．開発標準が，ユーザである技術者から見て魅力のないドキュメントとなってしまうと，目的であった技術者の能力向上や，共通認識の醸成がうまく進まず，ソフトウェア開発プロジェクトが失敗に終わってしまうことがある．そこで我々は，オープンソースソフトウェアの開発技術・文化を，組織などの閉じたコミュニティで実践するインナーソース開発のノウハウを，開発標準の作成に適用し，このような問題を解決した．本発表では，これらの経験に基づくノウハウや，考え方を紹介する．

キーワード：

開発標準, Standard Methodology,**標準化**,**インナーソース**,DevOps

想定している聴衆

社内など閉じたコミュニティにおいて、開発標準の普及・作成を行う者。
複数チーム間でのノウハウ共有や、スキル格差の解消を検討している者。

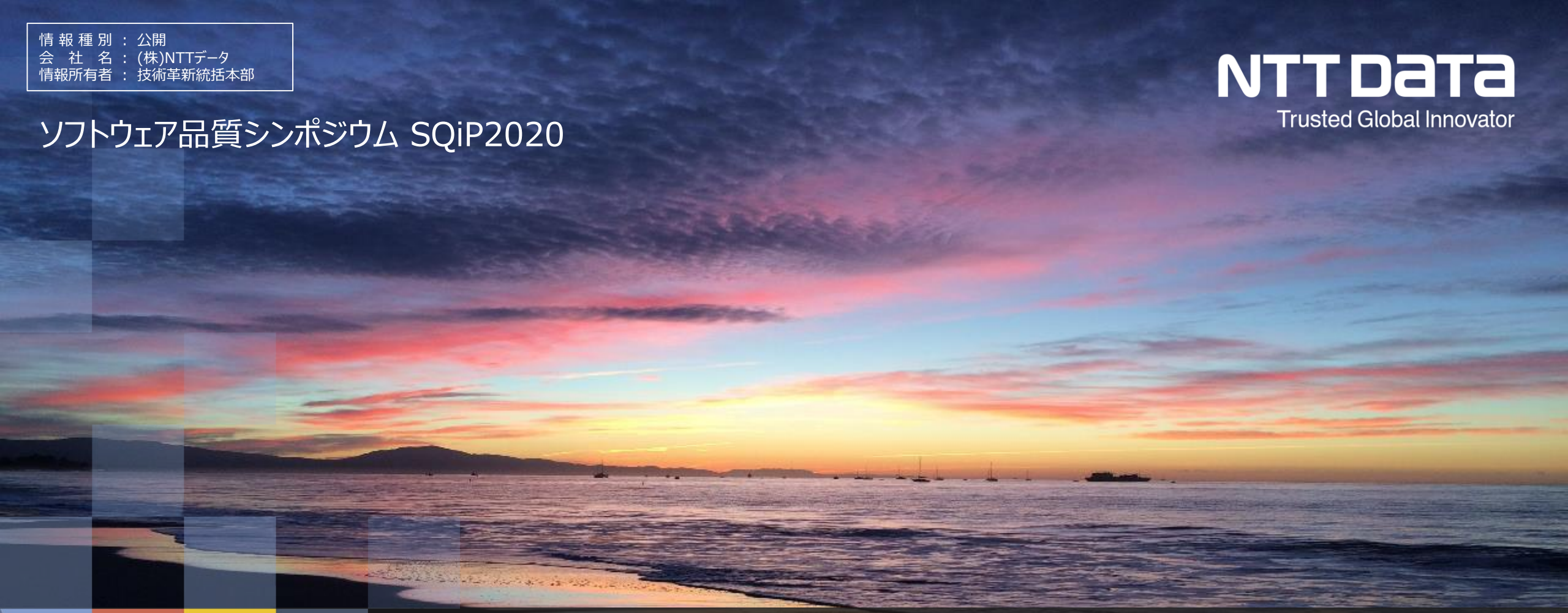
発表者の紹介（全角１００文字）：

大規模金融機関向けシステム開発で経験を積んだ後、ソフトウェア開発プロセスやテスト、品質管理に関する研究開発、技術支援に従事。社外講演、記事執筆なども行う。ソフトウェア品質・テストに関する受賞歴多数。

情報種別 ： 公開
会　社　名 ： (株)NTTデータ
情報所有者 ： 技術革新統括本部

NTT DATA
Trusted Global Innovator

ソフトウェア品質シンポジウム SQiP2020



「開発標準の作成におけるDevOps ～開発標準のインナーソース開発事例～」

株式会社NTTデータ　技術革新統括本部 システム技術本部
生産技術部 ソフトウェア技術センタ
熊川 一平

熊川 一平（くまがわ いっぺい）

株式会社NTTデータ 技術革新統括本部 システム技術本部 生産技術部 ソフトウェア技術センタ
テクニカルグレード イノベータ(ソフトウェアプロセス)

- ソフトウェアテスト/ソフトウェア品質保証を中心としたR&D活動と、現場適用支援に従事

【主な講演歴】
- JaSST'14 Tokyo ベストスピーカー賞
- JaSST'19 Tokyo ベストスピーカー賞
- ソフトウェア品質シンポジウム2017 SQiP Best Report Effective Award
- ソフトウェア品質シンポジウム2019 SQiP Best Paper Effective Award

みなさんの組織は“開発の標準化”に取り組んでいますか？

みなさんの組織に〝開発標準〟は、ありますか？

“開発標準”は、あったほうがいいと思いますか？

“開発標準”は、好きですか？

開発標準の現状

分厚すぎて
読む気がしない

教科書的で
実用に耐えない

技術的に古い

NTT DATA

Why ?

- 開発標準に「書いておいたほうがいいこと」は多く、ドキュメント量が膨大になりがち。
- 膨大なドキュメントは当然読まれない。
- また、保守性も低い。
- 保守がしにくいので、新しい技術をとりこんで見直すことができない。
- 記載内容が古いと、信頼が失われ余計に読まれなくなる。

保守性が低い

新しい技術がとりこまれない

Why ?

- 開発標準に「書いておいたほうがいいこと」は多く、ドキュメント量が膨大になりがち。
- 膨大なドキュメントは当然読まれない。
- また、保守性も低い。
- 保守がしにくいので、新しい技術をとりこんで見直すことができない。
- 記載内容が古いと、信頼が失われ余計に読まれなくなる。

保守性が低い

新しい技術がとりこまれない

そもそも嫌われていて期待されていない

NTT DATA

ユーザー＝技術者に愛される“開発標準”を作りたい

インナーソース

オープンソースの開発手法や文化を、閉じた組織やコミュニティの中で実践する開発手法

- GithubやGitLabなどのサービスを活用し，ユーザの手による修正がプルリクエストを通じて提案され，コミッタと呼ばれる認定された技術者がレビューすることで，ユーザの修正が開発対象に取り込まれる.
- チケット管理システムなどを通じて，ユーザが公開情報に対して，エラーやバグの報告，改善の提案をすることができる. またチケット上で改善案や修正方法などについて議論される.
- 上記を実現するため，ソースコードや開発の検討状況，レビューの状態などが公開されており，ユーザに対する情報の透明化が図られている.
- CIツールを用いることで，修正内容を即座に開発環境や本番環境にデプロイされる.

IKEA効果

IKEAの家具は自分で組み立てないといけない

でも、自分で組み立てた家具は、既製品よりも愛着が湧く

- 人は、自分で作ったり、自分が関与したモノを過大評価する
- 人は、手間をかけることで思いや愛着を強める

IKEA効果

とあるケーキミックスの話

- 「このケーキミックスは、水と混ぜてオーブンに入れるだけで、おいしいケーキが作れる」

  → このケーキミックスはなぜか売れなかった。
  メーカーはあることに気が付き、このケーキミックスを少しだけ変えた。
  すると、このケーキミックスは売れ始めた。

何を変えたのか？

IKEA効果

とあるケーキミックスの話

- 「このケーキミックスは、水と混ぜてオーブンに入れるだけで、おいしいケーキが作れる」

  → このケーキミックスはなぜか売れなかった。
  メーカーはあることに気が付き、このケーキミックスを少しだけ変えた。
  すると、このケーキミックスは売れ始めた。

何を変えたのか？

→「卵と牛乳を加える手順を追加した。」

簡単すぎる工程では「自分が作った！」と思えない。既製品と同じように思えてしまう。

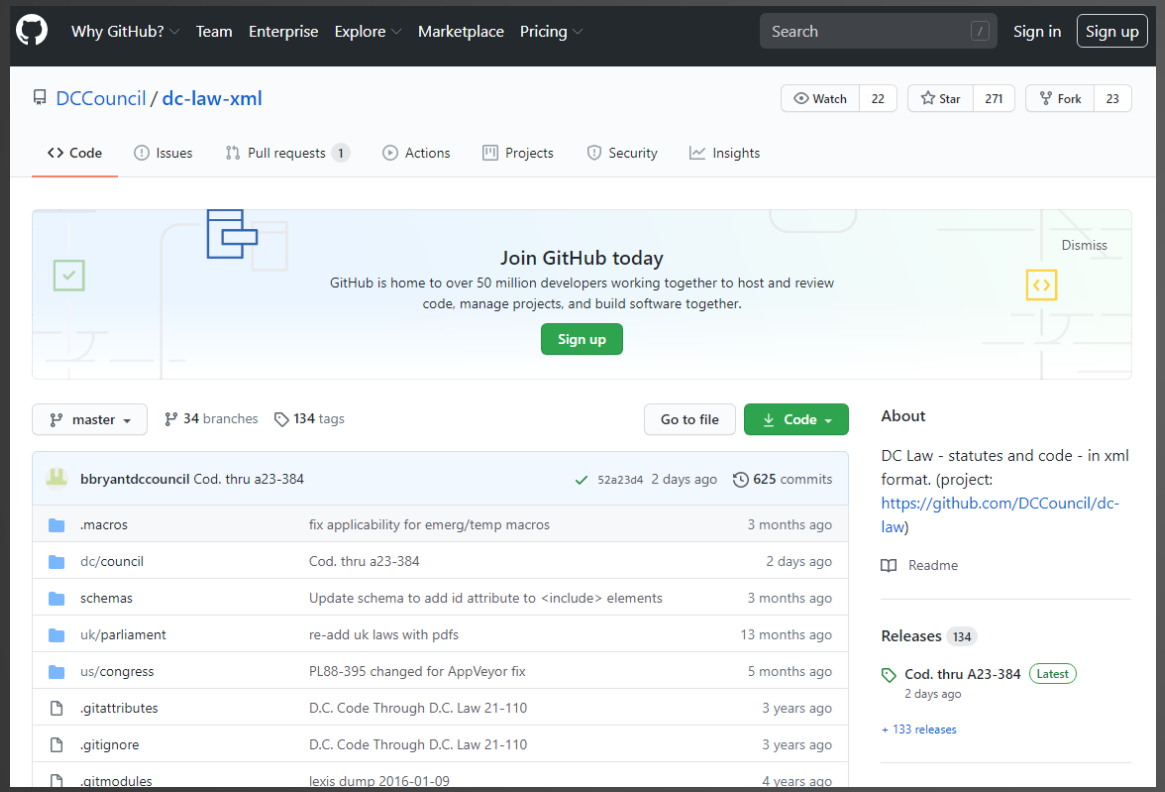
卵と牛乳を加えるだけで、「このケーキは自分がつくったものだ！」と言える人が増える。

出典：TED「What makes us feel good about our work?」,Dan Ariely
https://www.ted.com/talks/dan_ariely_what_makes_us_feel_good_about_our_work

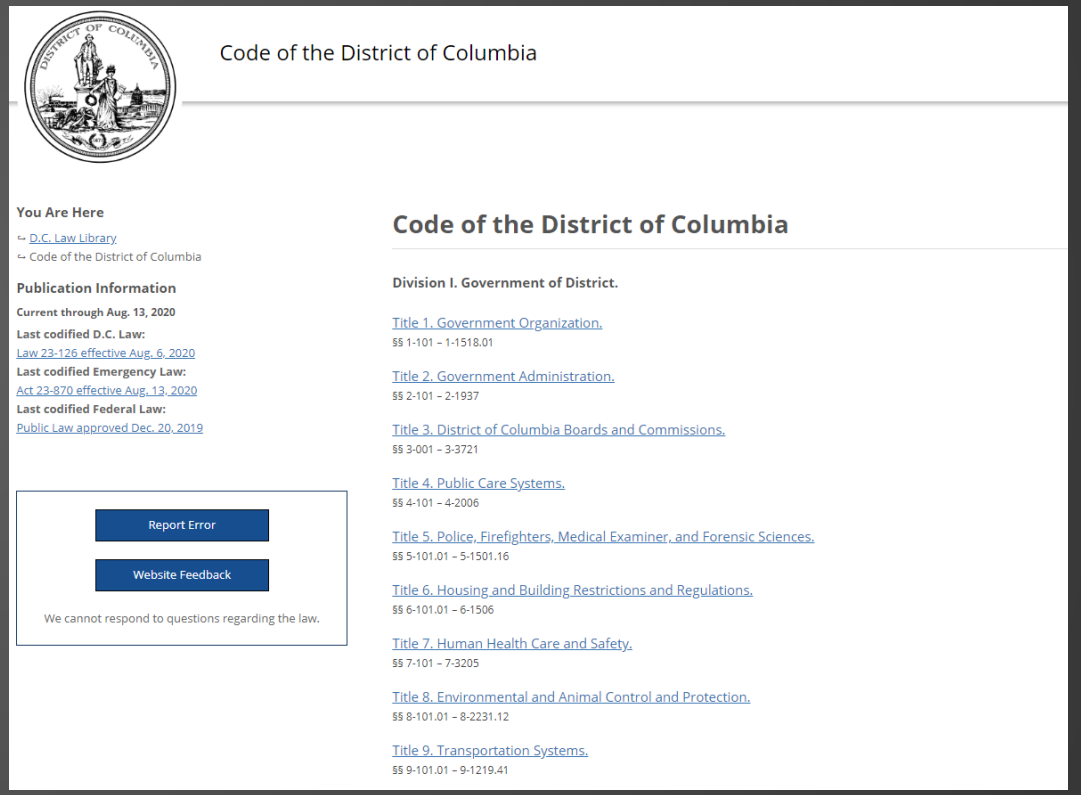
インナーソースを適用し IKEA効果を誘うことで 開発者に愛される “開発標準”を目指す

法令などをGithubで管理している事例

ワシントンの法律はGithubで管理されている。
実際、誤字をプルリクエストで修正されていたりする。



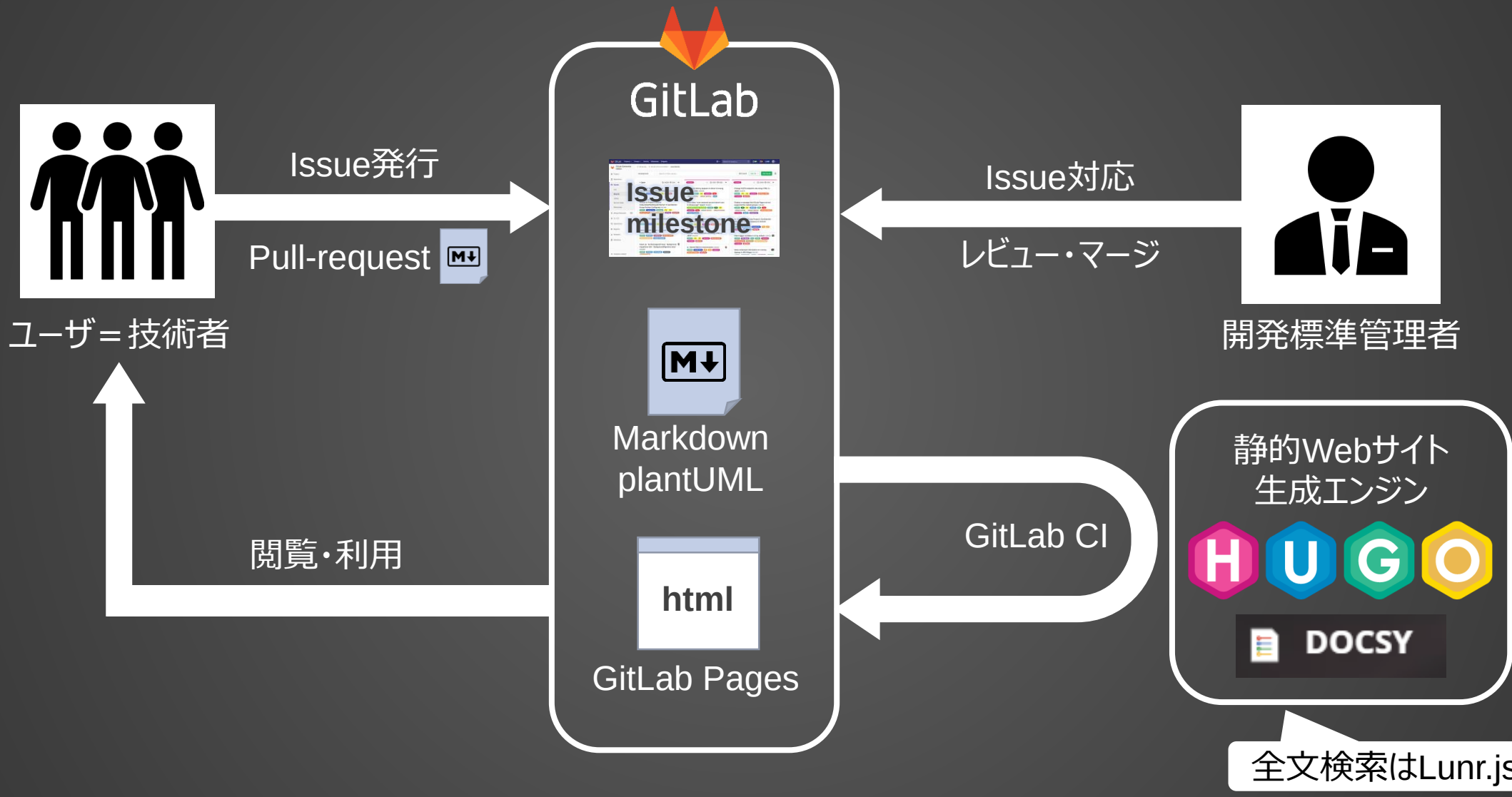
Github上ではxmlで管理



Html変換して開示

NTT DATA

ドキュメント開発におけるインナーソース開発



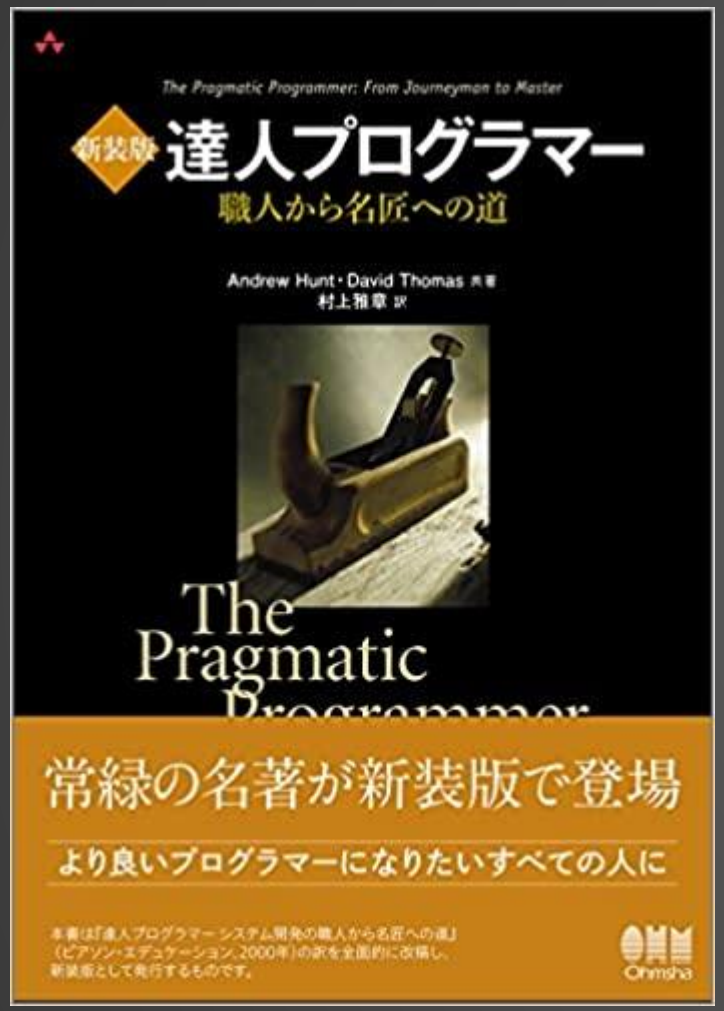
NTT DATA

バイナリ→プレーンテキスト

多数のユーザからの修正を取り込みつつ、安定した品質を保つために。レビューが非常に重要。
しかし、ゆっくりレビューしている暇はない。
「更新が遅くなる → 時代遅れになる → ユーザに嫌われる」
この流れはアプリケーションも、開発標準も同じ。

よって、プルリクエスト(マージリクエスト)での差分レビューは必須要件。
Excel方眼紙・神Excelはバイナリだから廃止。

NTT DATA



達人プログラマー
職人から名匠への道

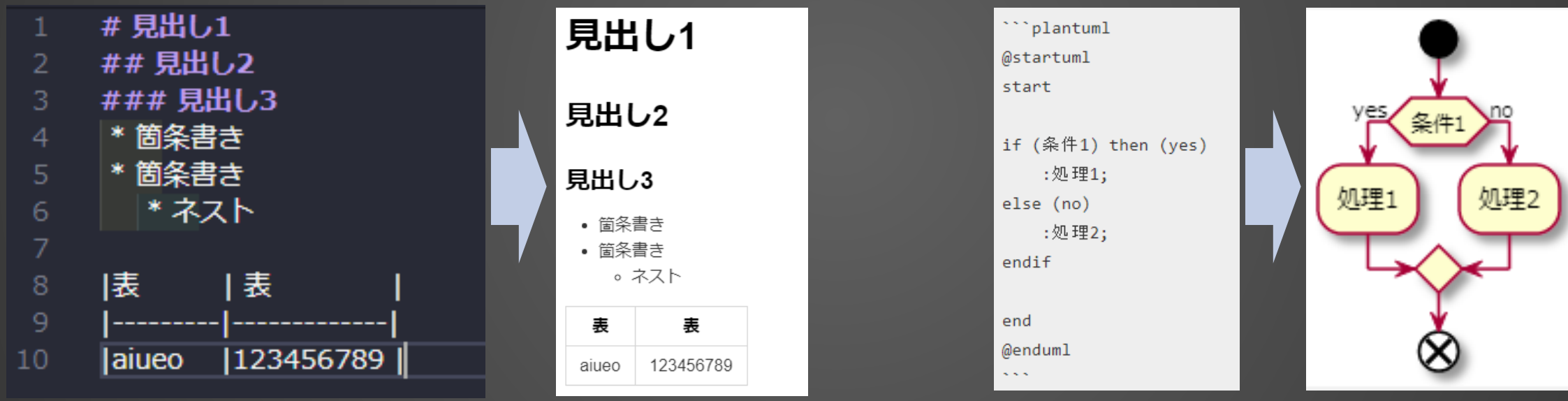
Andrew Hunt (著), David Thomas (著), 村上雅章 (翻訳)

第3章 プレイン・テキストの威力

- 達人プログラマーが取り扱う情報は、知識です。その知識を永続的に格納するためのフォーマットで最も適しているものが、プレインテキストです。
- プレイン・テキストのメリットは、透明性が保証される、様々な活用ができる、テストが用意になる。
- ソースコード管理システムは巨大な*UNDO*キー。すべてをソースコード管理システムで管理すること。それがソースコードでなくても。
- 実効可能ドキュメント、ドキュメントからコードを生成する。ドキュメントとコードの２つをメンテナンスすることは*DRY*原則に反する。プレインテキストならば、スクリプト言語によって加工が可能。

Markdown + plantUML

MarkdownもplantUMLも非常に簡単に執筆可能なプレーンテキストのドキュメント形式
プレーンテキストなので、プルリクエストで差分レビューが可能になり、レビュー効率が非常に高い。
また、2次利用もしやすく、応用も容易い。（全文検索など）



markdown

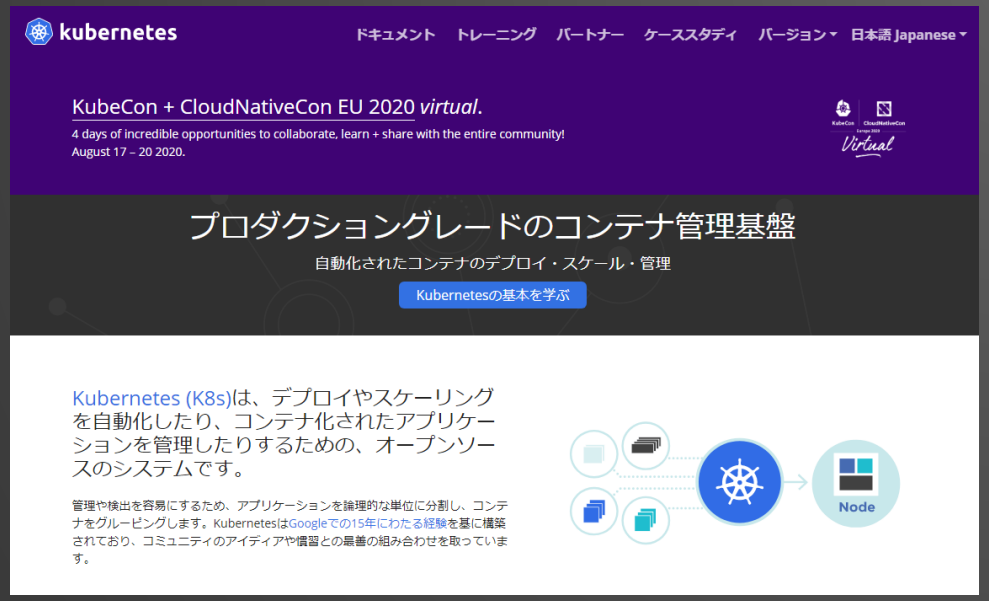
plantUML

NTT DATA

Hugo + Docsy

Hugo：
OSSの高速に動作する静的サイトジェネレータ。Markdownドキュメントをinputにhtmlを生成。
カスタマイズ性に富み、生成するhtmlのテンプレートも用途にあわせて沢山。

Docsy：
Google製のOSSで、Hugoで使う技術文書用テンプレート。
k8sのサイトもDocsy製。



NTT DATA

効果

保守性が低い

新しい技術がとりこまれない

プレーンテキストの高保守性

多くの修正・多くのリリース

コミット回数	2.9 回/日
プルリクエストのマージ回数	1.5 回/日
消化チケット数	1.47 枚/日
リリース回数	0.5 回/日
リリース1回あたりの作業時間	10分 /回

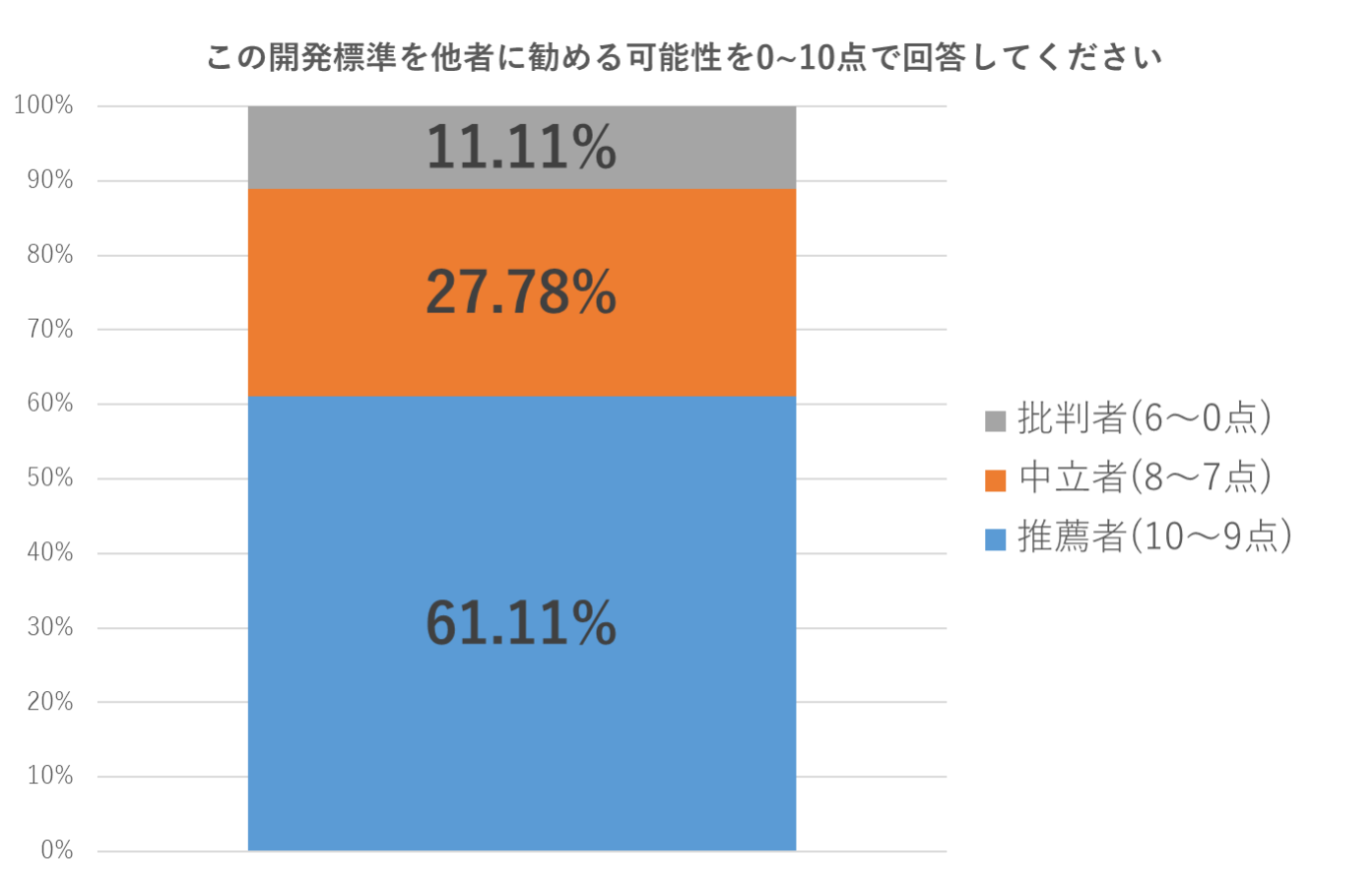
効果

そもそも嫌われていて
期待されていない

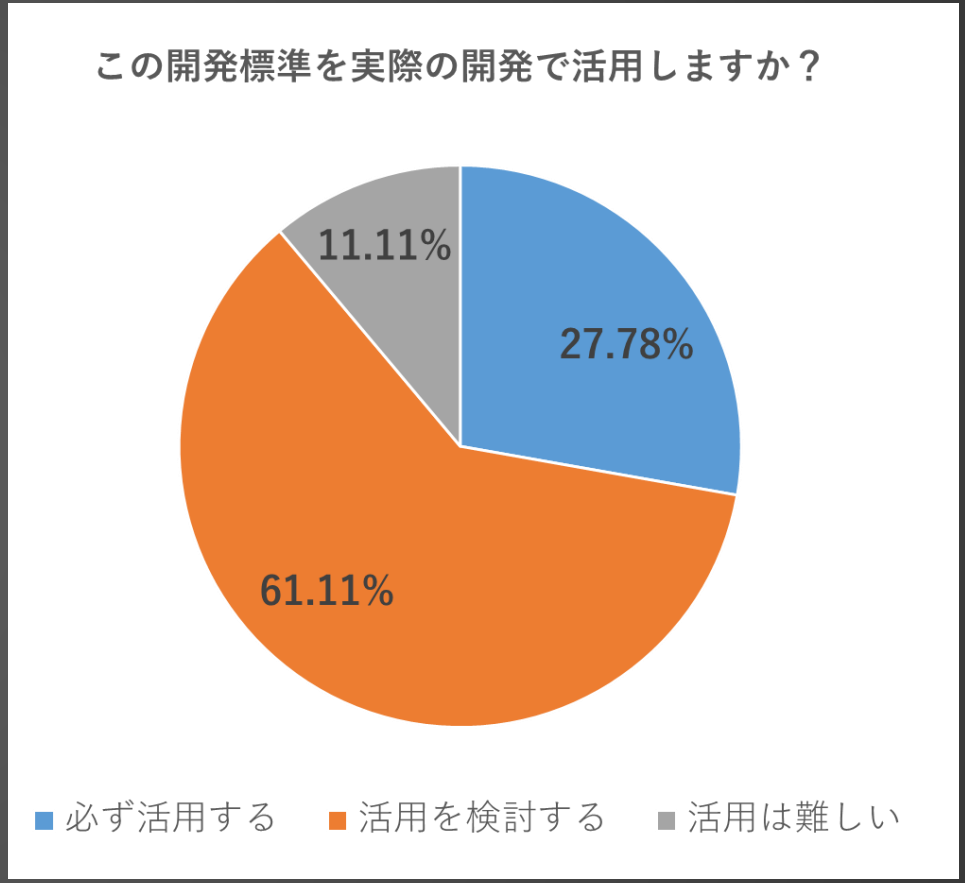
少しずつファンが付き
空気が変った

- ドキュメントを読んだユーザから改善提案のissueが切られた
- 技術解説ドキュメントのPull-Requestが送られてきた
- 全面採用を宣言してくれるPJが増えた
- この開発標準いいね！イケてるね！って言ってくれた
- 他の社内用ドキュメントを執筆している部署が、真似し始めた

ユーザ調査



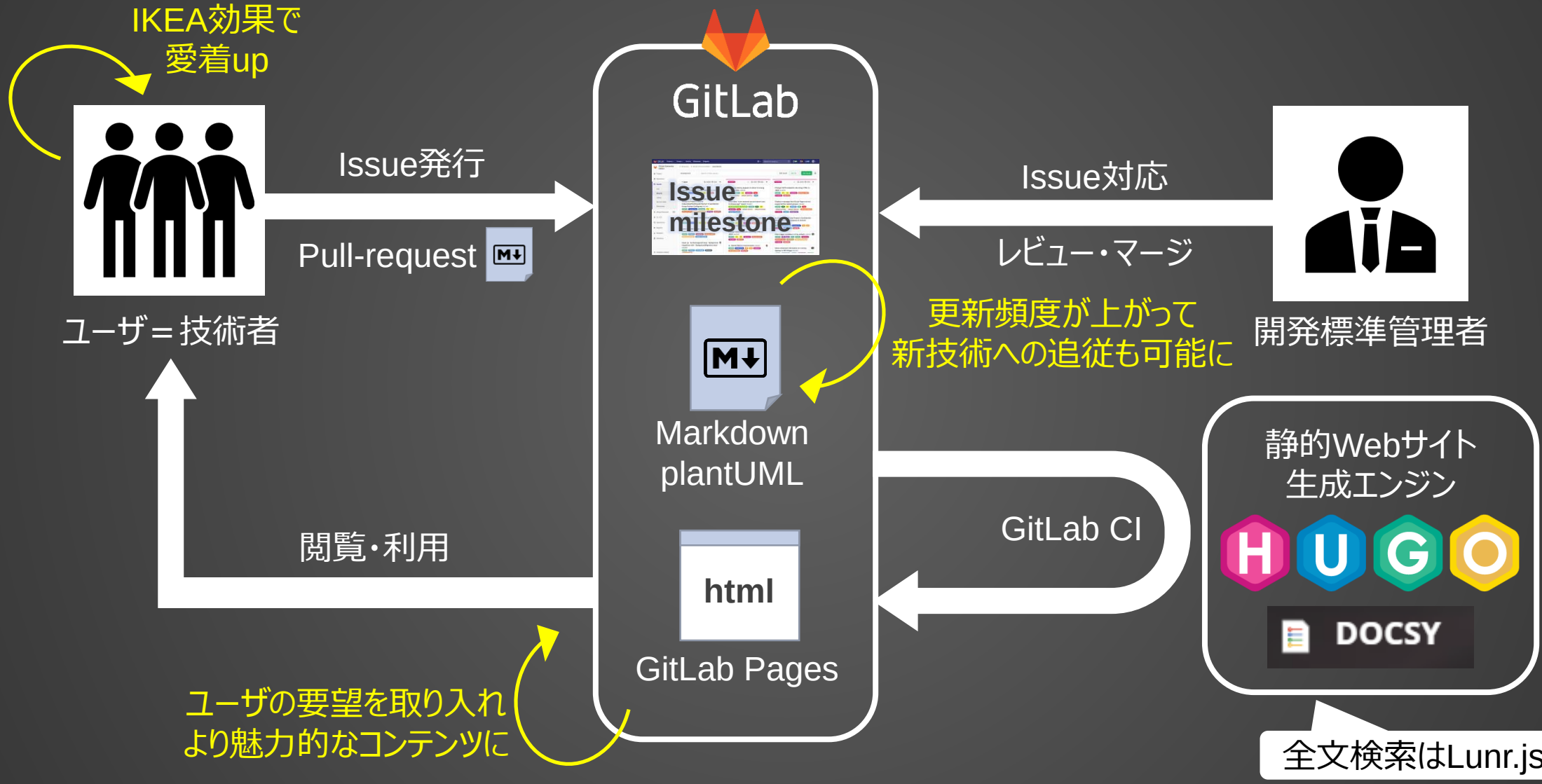
NPSは+50%と高い数値を記録している



90%近い人たちが活用を検討してくれている

NTT DATA

ドキュメント開発におけるインナーソース開発（再掲）



NTT DATA

IKEA効果

IKEAの家具は自分で組み立てないといけない
でも、自分で組み立てた家具は、既製品よりも愛着が湧く

- 人は、自分で作ったり、自分が関与したモノを過大評価する
- 人は、手間をかけることで思いや愛着を強める

ただし、以下のことに気を付けないといけない。

- 人は、手間をかけたものが、他者にとっての無関心であると大きく失望する。
- 人は、手間をかけたものを、無下に扱われると大きく失望する。

人の意見を尊重し
協力してくれる人には
全力で感謝の意を
示しましょう。

ありがとう！

- チームメンバのみんな
- NTTデータ 生産技術部の皆様
- NTTデータ ソフトウェア技術センタの皆様
- トライアルに協力してくれた皆様

NTT DATA

NTT DATA

Trusted Global Innovator