
初心者歓迎！ 2025年のOSSを振り返る

2026/01/31
ワイズプランニング

吉田 行男

■ 【経歴】

- 入社当時は、金融端末のソフトウェア開発に従事。
- 2000年頃から、Linux/OSSのビジネス開発を担当。
- 2012年から、オープンソース専門組織に所属。
- 2019年、定年退職により独立。

■ 【現在の業務】

- OSSを活用したビジネス構築のための支援
 - ✓ 新しい技術/OSSの発掘・評価検証
 - ✓ ビジネス・ソリューションの立ち上げ支援
 - ✓ 人材育成支援
- OSSコンプライアンス管理
 - ✓ ガイドライン作成、社内プロセス構築支援
- 各種講演、執筆及び技術ドキュメント翻訳



(*)2018/11：北東アジアOSS推進フォーラムにて「OSS貢献賞受賞」

◆ 【最近の活動】

➤ 記事執筆

- ✓ 「趣味」と「仕事」の境界線をシームレスにする「新世代エンジニア」の生存戦略とは
(<https://thinkit.co.jp/article/38865>)
- ✓ LF Researchが2025年の金融サービスにおけるOSS活用の現状を調査したレポートを公表
(<https://thinkit.co.jp/article/38905>)



➤ 翻訳

- ✓ 「2025年 金融サービスにおけるオープンソースの現状」
(<https://www.linuxfoundation.jp/blog/2026/01/japanese-version-of-the-2025-state-of-open-source-in-financial-services/>)



➤ その他

- ✓ Youtubeでの「OSS界隈のニュース深読み」
https://www.youtube.com/playlist?list=PLAsEyLm5aGOAOuSxOS_GoiQr3wzD4s23Y



Contents

1.2025年のOSSを振り返る

 1.1. 日本のオープンソースの現状

 1.2. 開発者の動向(GitHub, StackOverflow, TIOBE Index)

2. 2026年に知っておくべきOSSテクノロジー

Contents

1.2025年のOSSを振り返る

 1.1. 日本のオープンソースの現状

 1.2. 開発者の動向(GitHub, StackOverflow, TIOBE Index)

2. 2026年に身につけるべきOSSテクノロジー

◆ 「日本のオープンソースの現状 2025」を LFがリリース(*1)

➤ 調査概要

✓ 調査期間：2025年5月

✓ 回答者：851名(内 日本に本社のある組織からの回答：141名)

- 下記条件を満たすこと

» 回答者は OSS の概念について少なくともある程度理解していること

» 回答者は、OSS の経験と認識を共有する意思のある実在の人物であること

» 回答者は自分の雇用形態を識別でき、組織を代表できること

(*1) <https://www.linuxfoundation.jp/publications/2025/12/world-of-open-source-japan-2025-jp/>

- ◆ 「日本のオープンソースの現状 2025」を LFがリリース
 - 調査結果の要約



ビジネス価値

69%の日本企業が過去1年でOSSによる価値向上を報告。世界平均(54%)を上回る成長。



サポートの期待

89%が12時間以内の応答時間を要求。OSSは「基盤的ビジネスインフラ」へ進化。



ガバナンスの壁

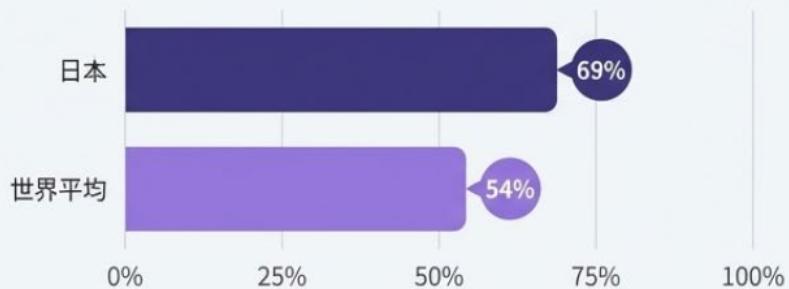
OSPO導入は41%に留まる。知的財産に関する懸念が貢献の最大の障壁に。

- ◆ 「日本のオープンソースの現状 2025」を LFがリリース
 - 日本のオープンソースパラドックス
 - ✓ 価値認識は世界をリードする一方、ガバナンスの成熟度は遅れている

69%

日本企業の69%が、過去1年間でOSSによるビジネス価値の向上を報告。これは世界平均の54%を大幅に上回る。

ビジネス価値の向上を報告した企業



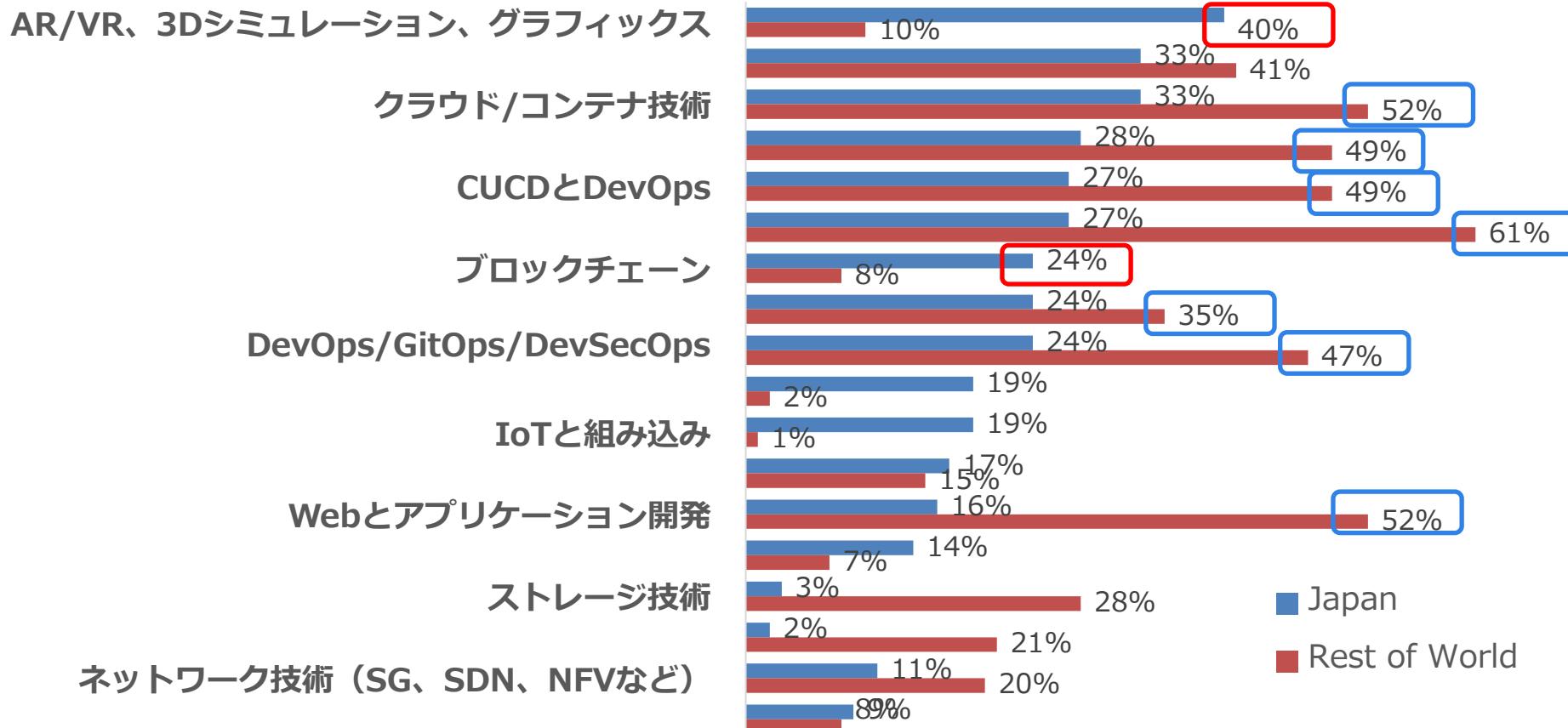
41%

しかし、オープンソースプログラムオフィス (OSPO) を導入している組織はわずか41%、明確な OSS戦略を策定しているのは39%に過ぎない。



◆ 「日本のオープンソースの現状 2025」を LFがリリース

➤ 技術分野におけるOSSの活用



日本は戦略的セグメントではリードしているものの、基盤インフラにおけるOSS導入は遅れている

◆ 「日本のオープンソースの現状 2025」を LFがリリース(*1)

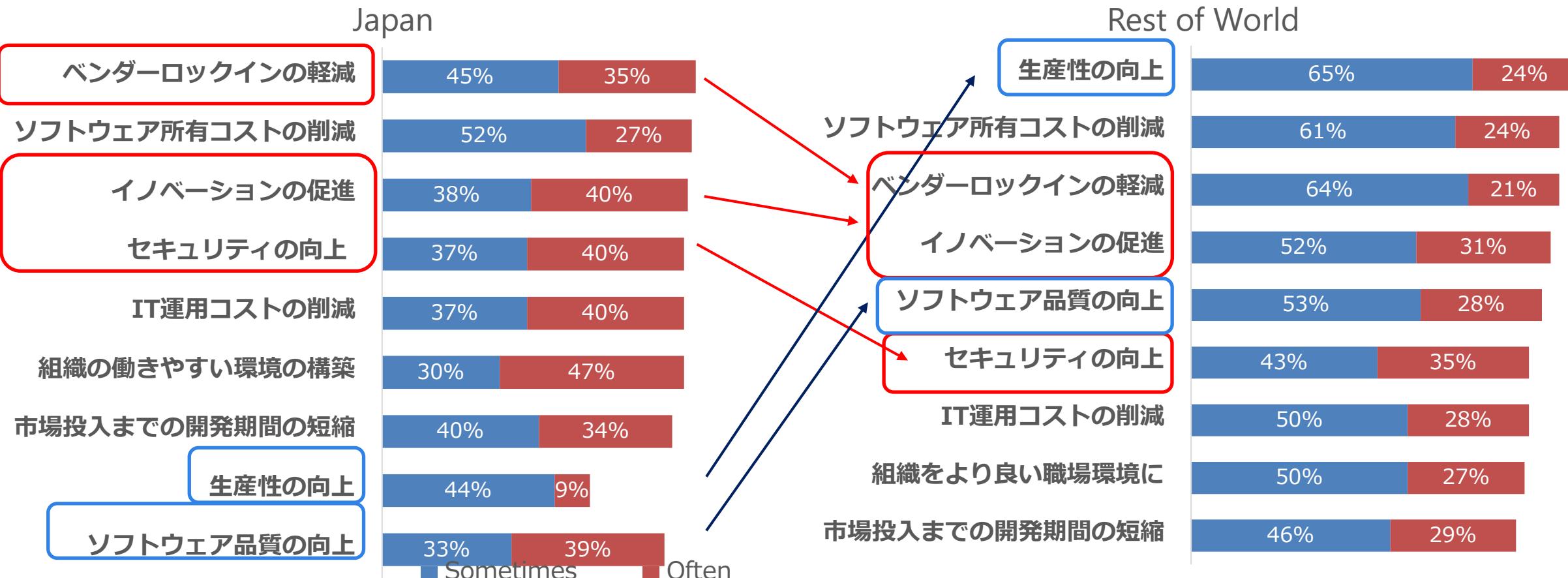
- セキュリティ評価は表層的：
コミュニティの健全性といった重要な指標が見過ごされている
- ✓ セキュリティ評価手法の比較

評価手法	日本	世界平均
自動セキュリティテストツールの使用	40%	29%
ソースコードの手動レビュー/検査	33%	27%
直接的な依存関係の評価	35%	36%
コミュニティの活動レベルチェック	26%	47%
Common Criteria 認証の信頼	52%	13%

形式主義的なガバナンスが
リスクの増大につながる恐れが！！

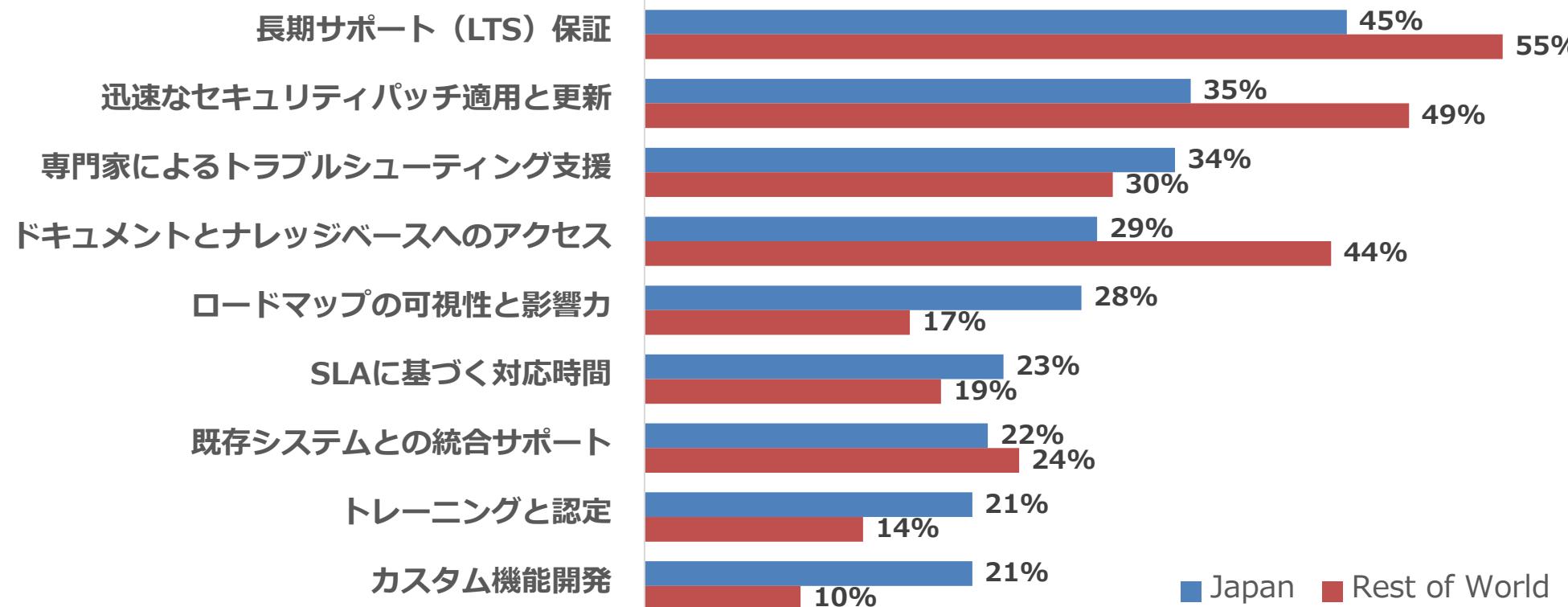
◆ 「日本のオープンソースの現状 2025」を LFがリリース

➤ オープンソース化によって最も恩恵を受けることは？



◆ 「日本のオープンソースの現状 2025」を LFがリリース

➤ オープンソース技術を本番環境で使用する際、サポートで最も期待する点は？



本番環境向けOSSにはエンタープライズレベルのサポートが必要
(日本の89%が12時間以内での応答を期待)

OSSは低コストの代替手段ではなく、強力な信頼性を必要とするコアインフラに？

◆ 「日本のオープンソースの現状 2025」を LFがリリース

➤ OSSの導入を制限する障壁は、知的財産への懸念と明確な社内ポリシーの欠如

◆ 貢献を阻む障壁

52% 知的財産の漏洩懸念

40% 技術的な制約と課題

34% 投資対効果の不確実性

◆ 導入を阻む障壁

51% OSS利用に関する明確なポリシーや
ガイダンスの欠如

44% 知的財産のライセンスに関する懸念

専門的な知見とリスクを管理するための
ガバナンス体制の構築が不可欠

◆ 「日本のオープンソースの現状 2025」を LFがリリース

- 日本のポテンシャルを解放する5つの戦略的提言

#	提 言	具 体 策
1	基盤インフラの導入加速	専門分野の強みを維持しつつ、OS、DevOps、DB等の導入ギャップを解消する。
2	包括的なガバナンス体制の確立	OSPO導入に加え、明確な戦略と公式見解を策定し、特にIP懸念に対処する。
3	セキュリティ評価の実践強化	Common Criteria認証を補完する多角的な評価（依存関係の評価、コードレビュー、コミュニティ健全性監視）を導入する。
4	エンタープライズ級サポート体制の確保	ミッションクリティカルなシステムには、日本の高い期待水準（12時間以内の応答など）を満たす商用サポート契約を締結する。
5	積極的な関与による戦略的参加の促進	開発者トレーニングやアップストリームへの貢献、重要な依存関係への投資を優先する。

果たしてこれで十分なのか？

Contents

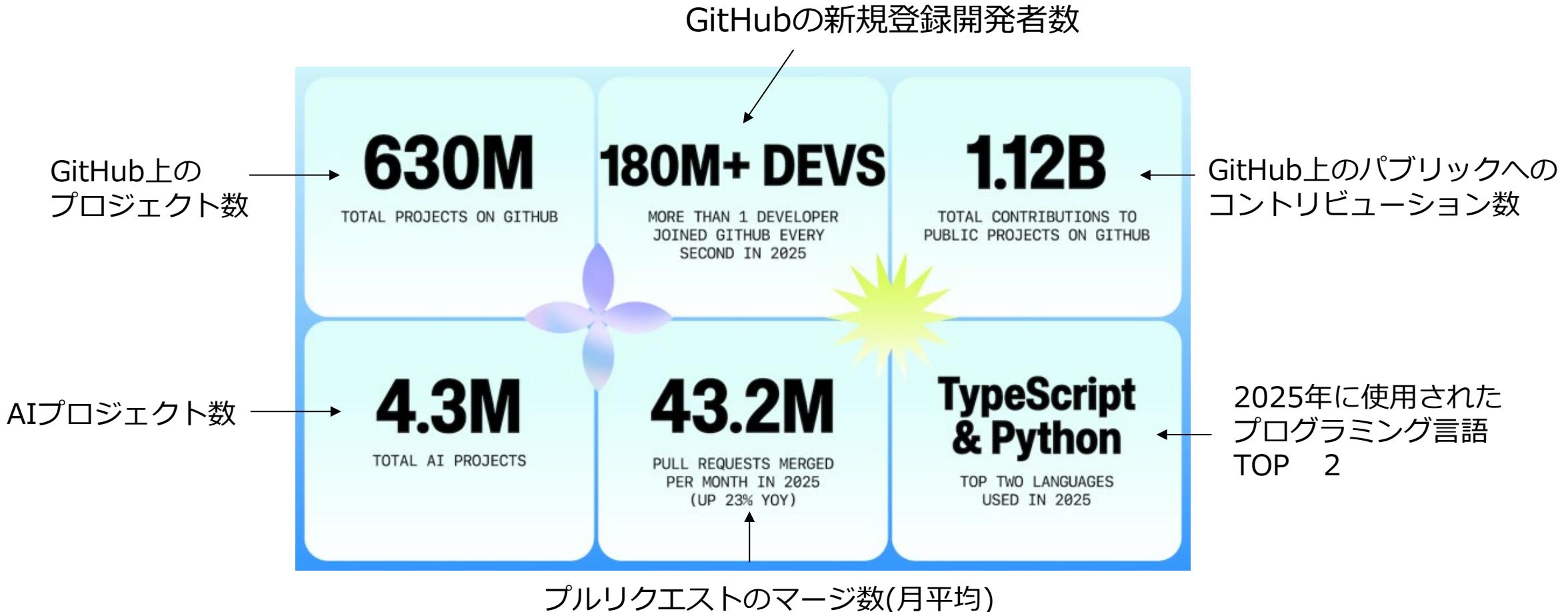
1. 2025年のOSSを振り返る

1.1. 日本のオープンソースの現状

1.2. 開発者の動向(GitHub, StackOverflow, TIOBE Index)

2. 2026年に身につけるべきOSSテクノロジー

- ◆ GitHubが、開発者の動向を調査した年次レポート「Octoverse 2025」を公開
 - コミュニティの爆発的拡大



(*) <https://github.blog/news-insights/octoverse/octoverse-a-new-developer-joins-github-every-second-as-ai-leads-typescript-to-1/>

◆ GitHubが、開発者の動向を調査した年次レポート「Octoverse 2025」を公開

➤ インド：成長を牽引するエンジン

✓ 国別の開発者数



- 1位が米国、昨年2位の中国を抜いて今年の2位はインド、3位が中国
- インドは着実に開発者人口が増加

◆ GitHubが、開発者の動向を調査した年次レポート「Octoverse 2025」を公開

- AIが標準となった開発環境
- ✓ AI活用の新たなパラダイム



LLM SDKの普及

110万件以上のパブリックリポジトリがLLM SDKを導入。AI機能の組み込みが「特別なこと」ではなくなった。



即時採用

新規開発者の80%が、登録から1週間以内にGitHub Copilotを利用。AIアシスタントは初心者の必須ツールに。



Vibe Coding

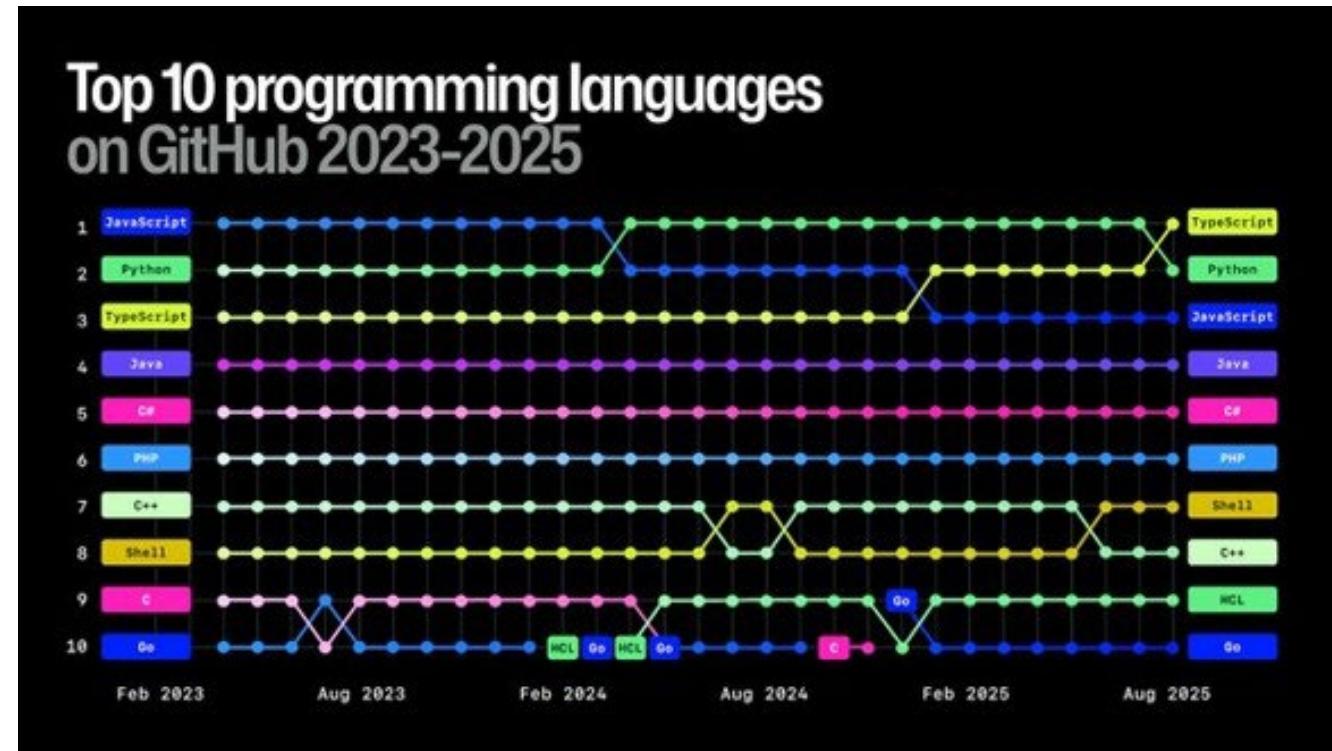
アイデアから実行可能なプロトタイプへ。AIを通して直感的に構築する「バイブ・コーディング」が台頭。

◆ GitHubが、開発者の動向を調査した年次レポート「Octoverse 2025」を公開

➤ プログラミング言語の地殻変動

✓ TypeScriptは現在、GitHubで最も使用されている言語

– TypeScriptはコントリビューター数において、2025年1月に初めてJavaScriptを抜いて2位となり、2025年8月に初めてPythonを抜いて首位に



◆ GitHubが、開発者の動向を調査した年次レポート「Octoverse 2025」を公開

➤ プログラミング言語の地殻変動

✓なぜ今、TypeScriptなのか？

AIエージェントとの親和性

- AIによるコード生成において、「型」の存在は信頼性を劇的に向上させる。TypeScriptの厳格な型定義は、AIエージェントが出力するコードの検証を容易にし、プロダクション環境への導入を早める。
- 一方、Pythonは依然としてAI研究やデータサイエンスの現場で絶対的な王者として君臨している。

◆ GitHubが、開発者の動向を調査した年次レポート「Octoverse 2025」を公開

➤ 未来への展望

✓ AIネイティブなセキュリティ:

- 開発初期段階からの自動化された脆弱性検知が、ソフトウェア供給網の安全を保証。

✓ コラボレーションの変革:

- AIエージェントが開発者間、あるいはリポジトリ間の調整役を担い、開発効率を最大化。

✓ グローバル・イノベーションの民主化:

- AIの支援により、世界中のあらゆる場所から、誰でもアイデアをコードに落としめる時代の到来。

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

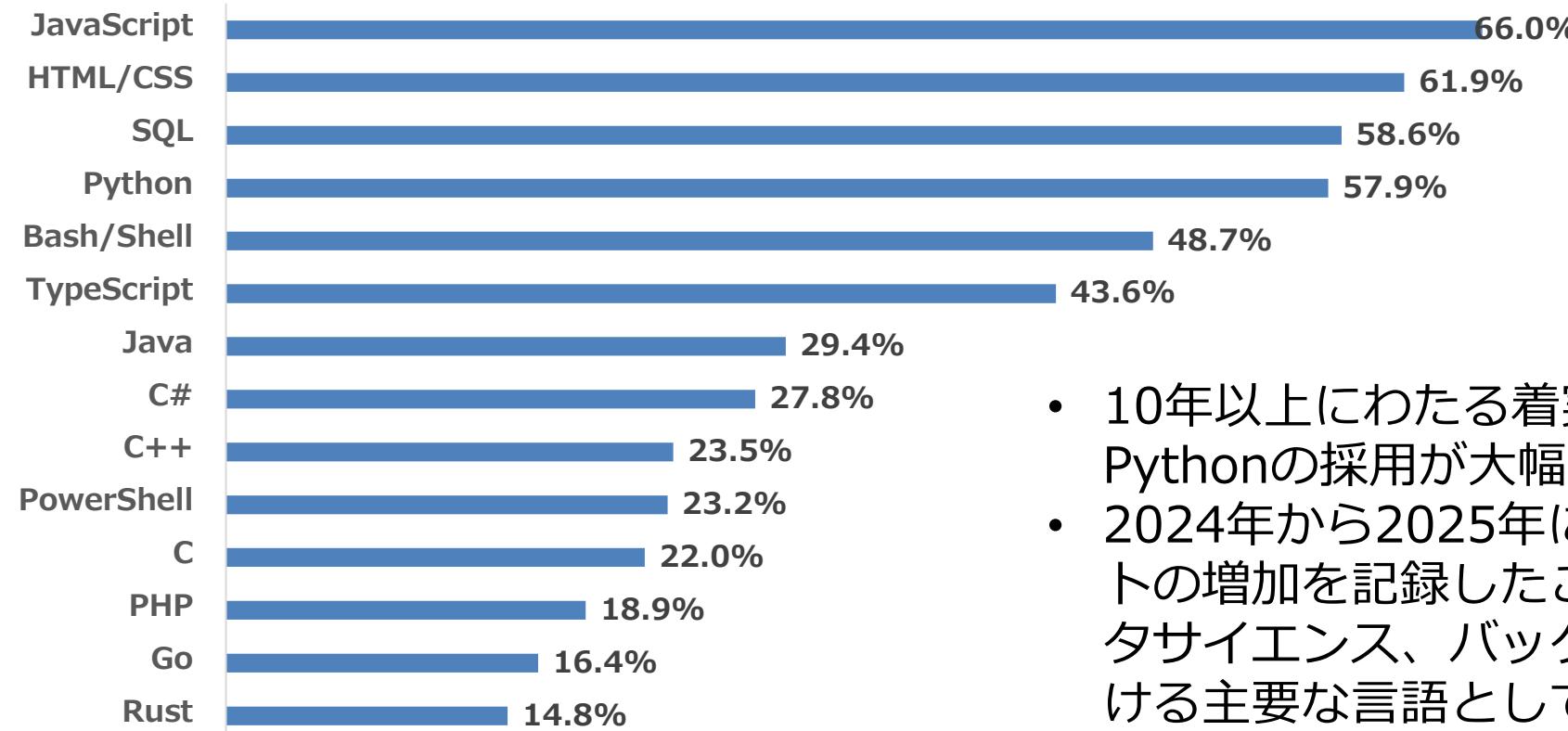
➤ 主な調査結果：AIツールの利用と信頼

- ✓ **AIツールの利用が拡大**: 開発者の80%がワークフローでAIツールを利用していると回答し、その採用が引き続き増加している。
- ✓ **信頼度は低下**: 一方で、AIが生成するコードの正確性に対する信頼度は低下。AIの回答を信頼していると答えた開発者は前年の40%から29%に減少。
- ✓ **「ほぼ正しい」コードへの不満**: 開発者の45%が、「ほぼ正しいが、完全ではない」AIソリューションへの対応に最も不満を感じていると回答しており、これがデバッグ時間を増加させる一因となっていることを示唆。
- ✓ **依然として人への信頼が厚い**: AIの回答を信頼できない場合、75%の開発者が他の人に助けを求める的回答しており、複雑な問題解決においては依然として人間同士の協力が重視されていることが浮き彫りに。

(*) <https://survey.stackoverflow.co/2025>

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

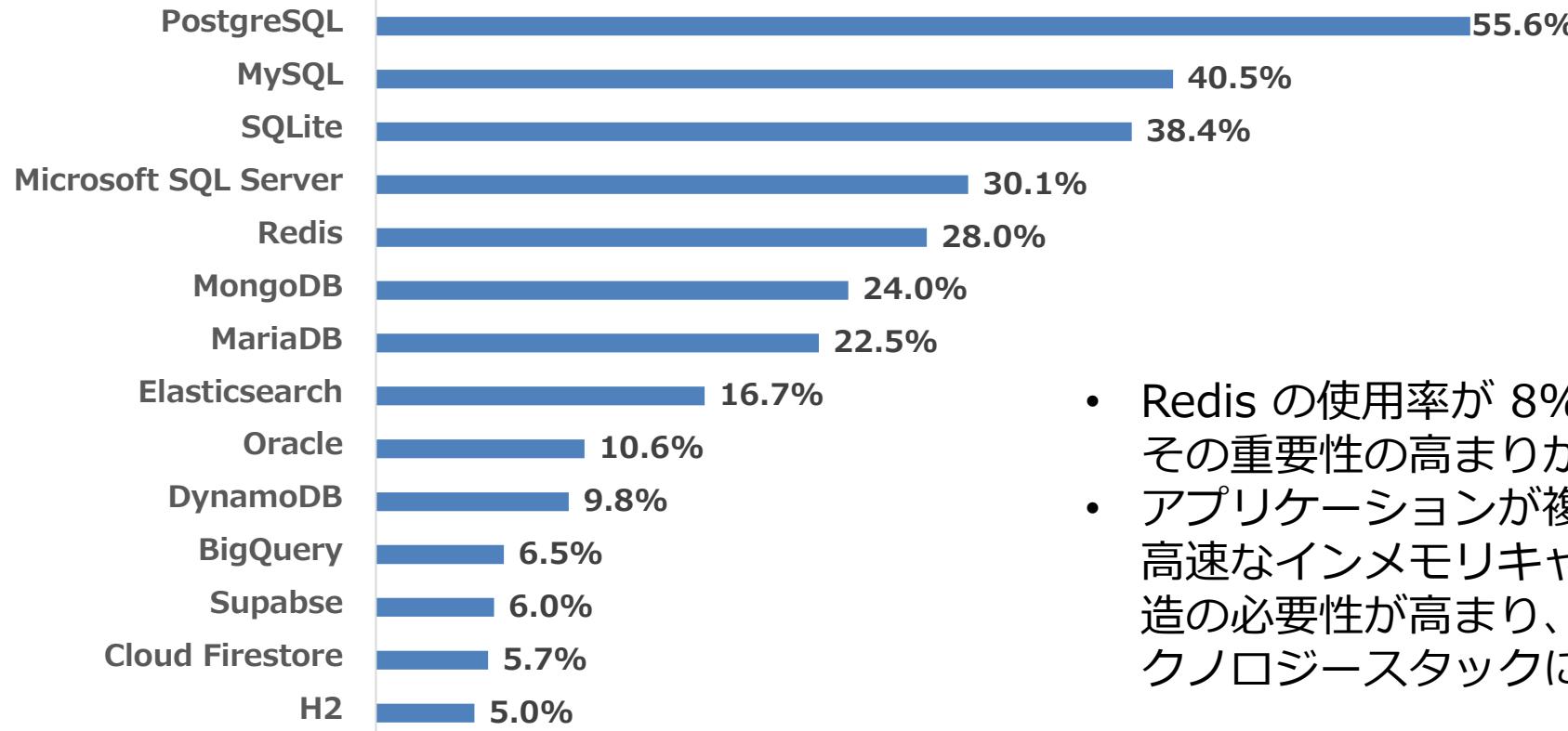
- 調査結果：最も人気のあるテクノロジー(プログラミング言語)



- 10年以上にわたる着実な成長を経て、Pythonの採用が大幅に加速
- 2024年から2025年にかけて7ポイントの増加を記録したことは、AI、データサイエンス、バックエンド開発における主要な言語としての地位を確立していることを示している。

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

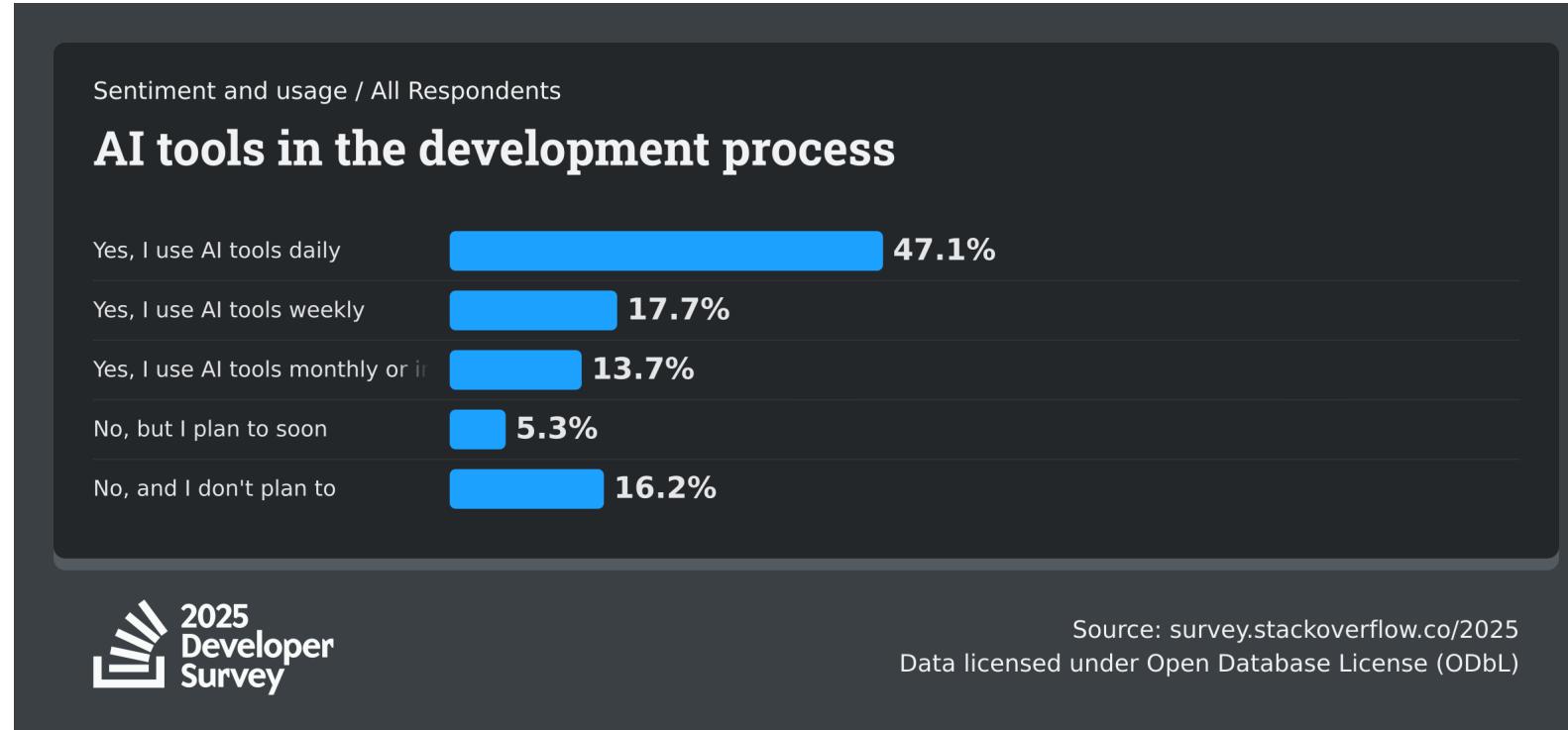
➤ 調査結果：最も人気のあるテクノロジー(データベース)



- Redis の使用率が 8% 増加したことは、その重要性の高まりが明らかに
- アプリケーションが複雑化するにつれて、高速なインメモリキャッシュとデータ構造の必要性が高まり、Redis は現代のテクノロジースタックに欠かせない要素に

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

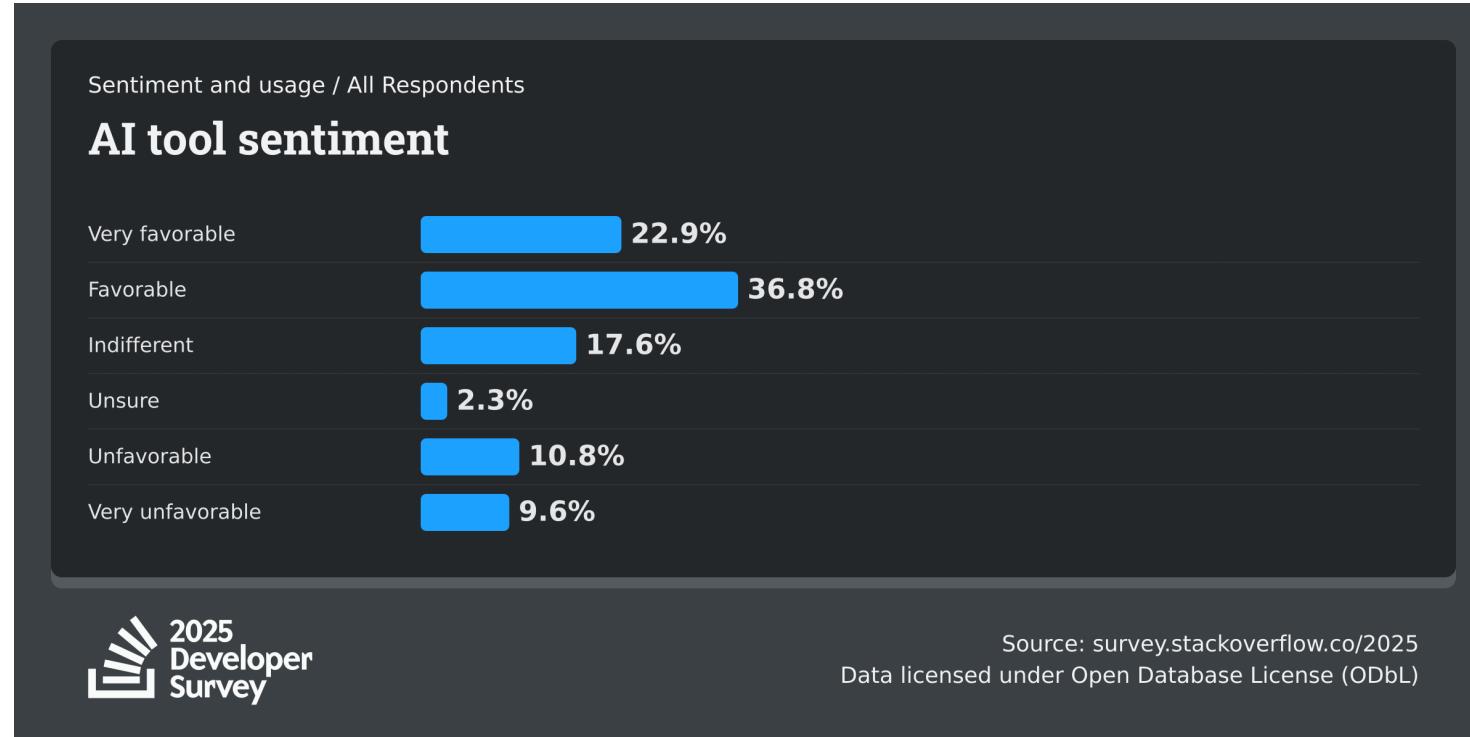
➤ 調査結果：AIツールの利用について



- 84%が開発プロセスにおいてAIツールを使用しているか、または使用予定(昨年の76%から増加)
- プロフェッショナル開発者の51%がAIツールを毎日使用している

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

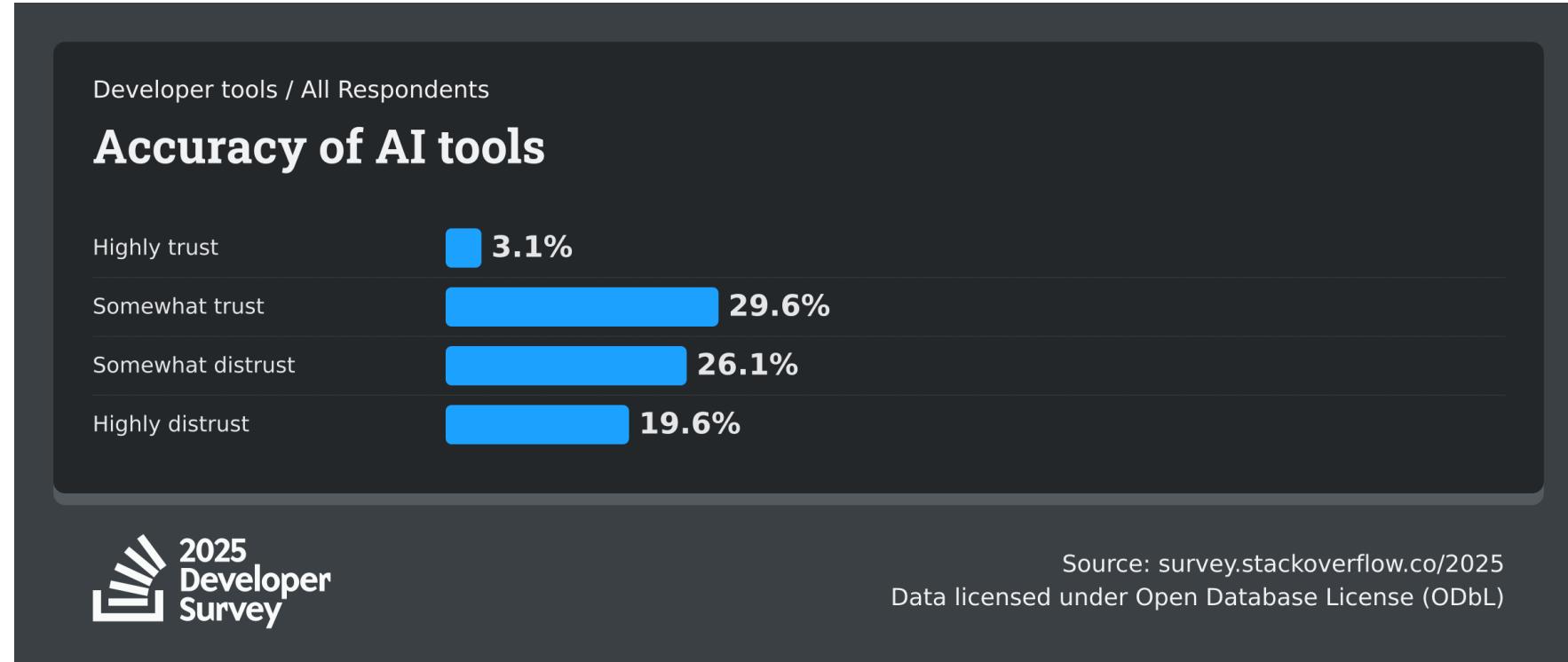
➤ 調査結果：AIツールへの感情



- AIツールに対するポジティブな感情は、使用状況とは逆に2025年に減少(70%以上(2024) ⇒ 60%(2025))
- 専門家は、プログラミングを学習中の人々(53%)よりも全体的に高い好意的な感情(61%)。

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

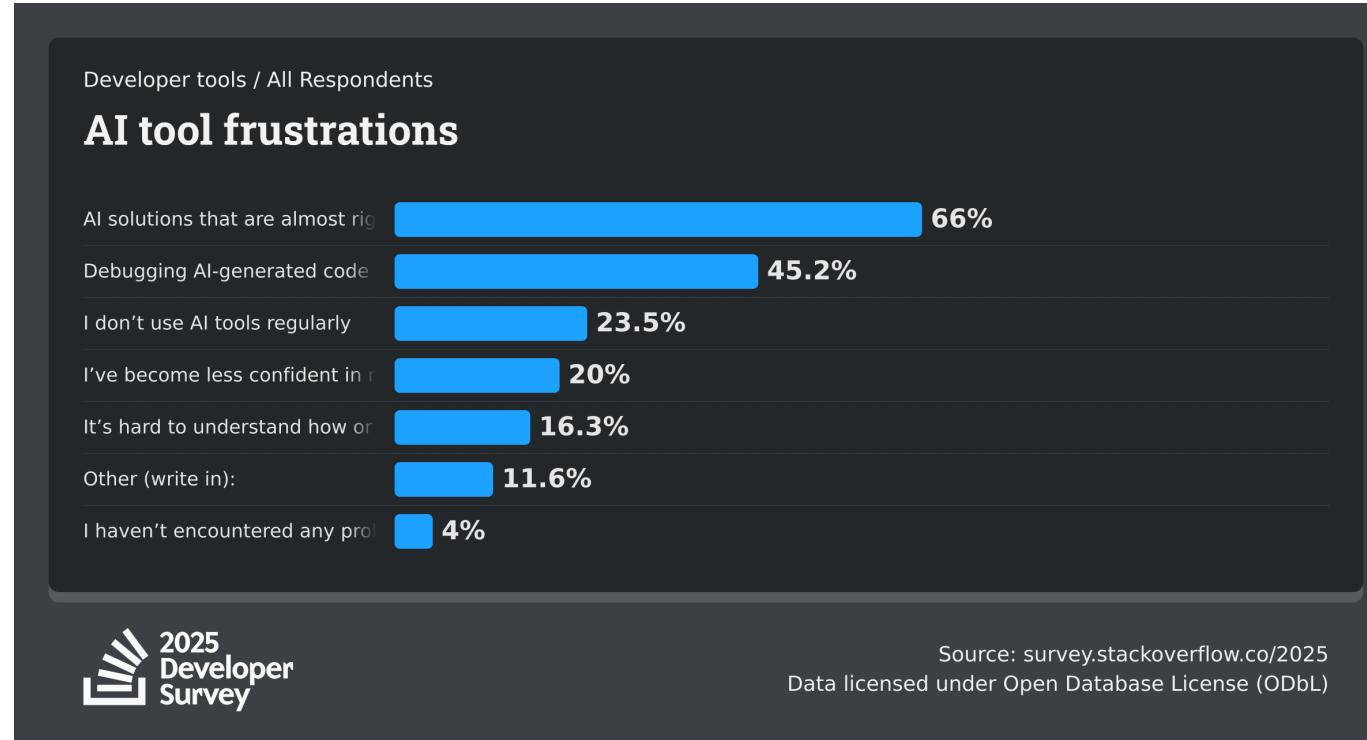
- 調査結果：AIツールの正確性について



- AI ツールの正確性を信頼する開発者(33%)よりも、不信感を抱く開発者(46%)の方が多く、その出力を「非常に信頼している」とレポートした開発者はごくわずか(3%)。
- 経験豊富な開発者は最も慎重で、「非常に信頼している」割合が最も低く(2.6%)、「非常に疑っている」割合が最も高い(20%)結果。責任を負う役割にある者にとって人間による検証の必要性が広く認識されている！

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

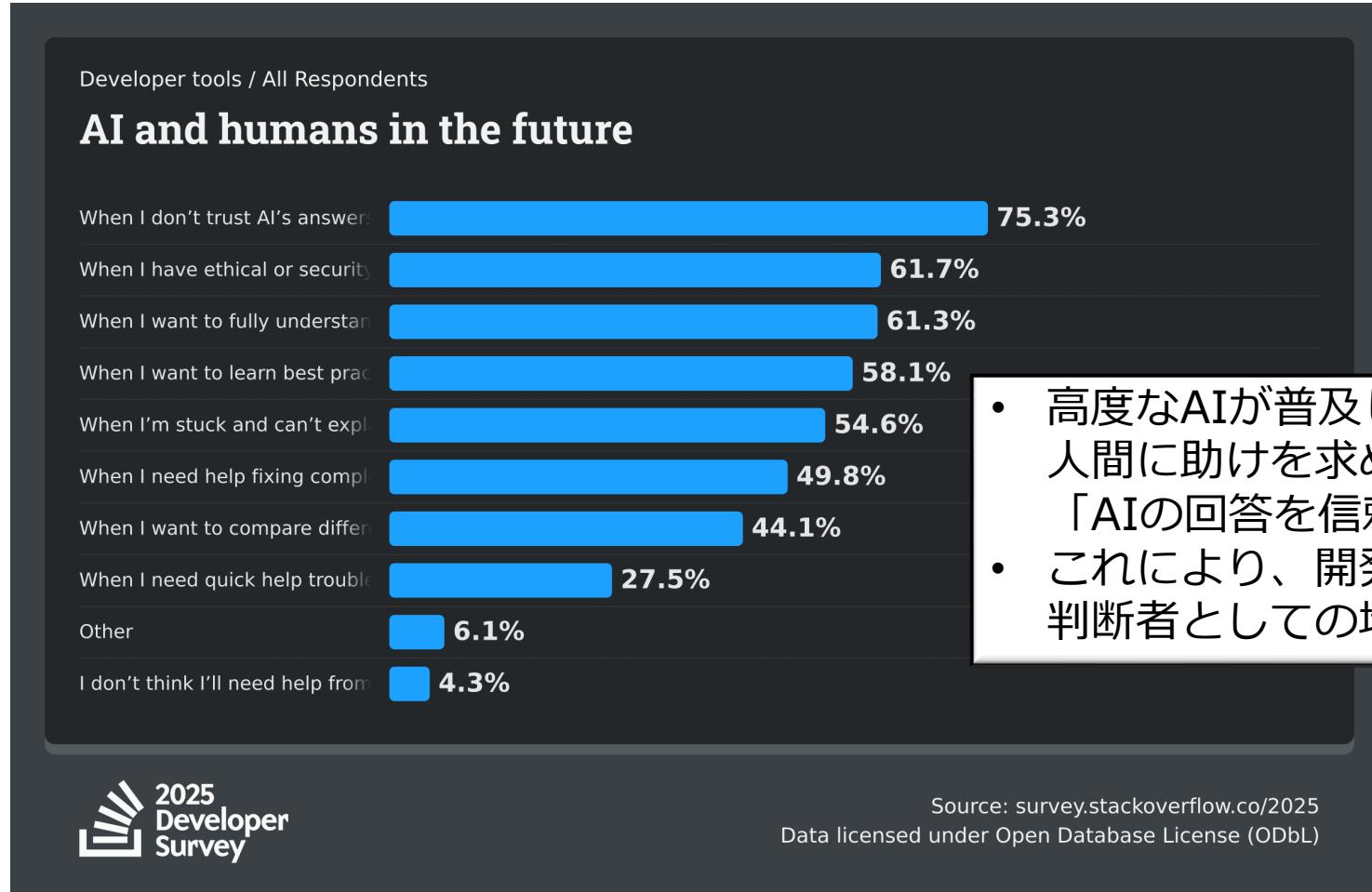
➤ 調査結果：AIツールへの不満



- 開発者の 66% が挙げた最大の不満は、「ほぼ正しいが、完全ではない AI ソリューション」への対応。これは 2 番目に多い不満「AI によって生成されたコードのデバッグに時間がかかる」(45%) につながることが多い。

◆ 「Stack Overflow」が、「The 2025 Developer Survey」を発表

➤ 調査結果：未来におけるAIと人間



- 高度なAIが普及した未来において、開発者が人間に助けを求める最も大きな理由は「AIの回答を信頼できない場合」(75%)。
- これにより、開発者は品質と正確性の最終的な判断者としての地位を確立することになる！？

◆ C#がTIOBE Indexの「プログラミング言語オブ・ザ・イヤー」に選出

➤ TIOBE Indexとは？

✓ 概要：

- 検索エンジンのクエリ数に基づき、世界中でどのプログラミング言語が最も話題になっているかを数値化した指標。

✓ 目的：

- 自分のプログラミングスキルが市場の需要と一致しているかの確認。
- 新規システム開発における戦略的な言語選定の判断材料。

✓ 算出根拠：

- Google, Bing, Yahoo!, Wikipedia, YouTubeなど、主要25サイトの検索結果をベースに算出
 - » エンジニア数：全世界の熟練エンジニアの数に基づいた検索ボリュームを算出
 - » 教育講座：大学やオンラインプラットフォームで提供されるコース数を反映
 - » サードパーティ：ライブラリ、ツール、関連ベンダーの経済圏の広さを評価

(*) <https://www.tiobe.com/tiobe-index/>

- ◆ C#がTIOBE Indexの「プログラミング言語オブ・ザ・イヤー」に選出
➤ 2026年1月 プログラミング言語ランキング (Top 10)

順位	言語名	レーティング (%)	前年比
1位	Python	22.61%	持続的な首位
2位	C	10.99%	微減傾向
3位	Java	8.71%	横ばい
4位	C++	8.67%	Cと激しい競合
5位	C#	7.39%	急成長 (+2.94%)
6位	JavaScript	3.03%	安定
7位	Visual Basic	2.41%	維持
8位	SQL	2.27%	安定
9位	Delphi/Object Pascal	1.98%	根強い人気
10位	R	1.82%	Top 10復帰

◆ C#がTIOBE Indexの「プログラミング言語オブ・ザ・イヤー」に選出

➤ 2025年の年間最優秀言語：C#

✓ 選出理由：

- 2025年の1年間で最もレーティングを伸ばした言語 (+2.94%)。

✓ C#の勢い：

- 成長率：

3年間で2度目の受賞。

- パラダイムシフトの成功：

Windows専用からクロスプラットフォームへ、
Microsoft専有からオープンソースへと見事に進化。

- 最新トレンドの導入：

モダンな機能の採用が早く、エンタープライズ市場で
Javaに代わる選択肢として地位を固めている。

Year	Winner
2024	C#
2023	Python
2022	C#
2021	C++
2020	Python
2019	C
2018	Python
2017	C
2016	Go

◆ C#がTIOBE Indexの「プログラミング言語オブ・ザ・イヤー」に選出

➤ 注目すべき「勝者」と「敗者」

2025年から2026年初頭にかけての大きな動き。

✓ 勝者 (Winners):

- Perl : 前年32位から11位へ驚異的なランクアップ。かつてのWeb標準言語が再注目
- R : 統計/データサイエンスの需要により、18位→10位へ上昇しトップ10返り咲き。
- Rust : 過去最高位の13位を記録。安全なシステム言語としての地位を確立中。

✓ 敗者 (Losers):

- Go: 前年の7位→ 16位へと大きく後退。クラウドネイティブ分野での競争が激化。
- Ruby : トップ20圏外へ後退。
- Visual Basic : 緩やかな減少傾向が続く。

◆ C#がTIOBE Indexの「プログラミング言語オブ・ザ・イヤー」に選出

➤ まとめ・今後の展望

✓ Python 1強時代:

- 学習の容易さとAI需要により、当面は首位が揺らぐことはない。

✓ C#の台頭:

- 開発効率と汎用性を武器に、上位3位 (Python, C, Java) の壁を壊す勢いが。

✓ 専門特化と言語の多様化:

- RやRust、Perlの再浮上に見られるように、用途に応じた言語の使い分けが加速。

Contents

1.2025年のOSSを振り返る

 1.1. 日本のオープンソースの現状

 1.2. 開発者の動向(GitHub, StackOverflow, TIOBE Index)

2. 2026年に知っておくべきOSSテクノロジー

◆ 5つの主要な柱

AI Agents & Local LLMs

自律型エージェントとローカル
推論の台頭

Vector Databases

RAGを支えるAIインフラ

Platform Engineering

DevExの標準化とIDPの普及

Modern Data Stack

Open Table Formatsと
Vector DB

Security First

SBOMとサプライチェーン
セキュリティ

1. AI Agents & Local LLMs

➤ **自律型エージェントへのシフト:**

単なるチャットボットから、タスクを
自律的に遂行するエージェントへ。

Key OSS: LangGraph, AutoGen

➤ **ローカル推論の民主化:**

プライバシーとコストの観点から、エッジ
デバイスでのSLM(小規模言語モデル)実行が
加速。

Key OSS: Ollama, vLLM, Llama.cpp

➤ **RAGの進化:**

静的な検索から、動的でコンテキストを
理解した高度な検索拡張生成へ。



1. AI Agents & Local LLMs

➤ AIエージェント

「目的を与えれば、自分で考え、道具を使い、自律的に行動してタスクを完了させるAI」のこと

1. 従来のAI（LLM）との決定的な違い

特徴	従来のAI (LLM / チャットボット)	AIエージェント
役割	知識の検索、文章生成、相談相手	タスクの実行、問題解決
スタンス	受動的（ユーザーの質問を待つ）	能動的 / 自律的（ゴールに向かって動く）
できること	「旅行プランを提案して」と言うと、プランの文章を作る。	「旅行プランを立てて」と言うと、空席を検索し、ホテルを予約し、カレンダーに登録する。
外部との連携	基本的に内部知識のみ（最近は検索機能付きもある）	外部ツール（Webブラウザ、メール、API、計算機など）を自由に使いこなす。

1. AI Agents & Local LLMs

➤ AIエージェント

2. AIエージェントの仕組み：どうやって動いているか？

1. 知覚 (Perception) : ユーザーからの指示や、環境 (Webサイトの情報など) を受け取る。
2. 計画 (Planning / Brain) : LLM (脳) を使って、「このゴールを達成するには、どの順序で何をすべきか？」を考え、タスクを分解する。
3. 行動 (Action) : 計画に基づき、ツールを使う。
 - » Web検索を行う
 - » Pythonコードを実行して計算する
 - » ファイルを保存する
 - » メールを送信する
4. 観察と修正 (Reflection) : 行動の結果を見て、「うまくいったか？」を判断し、エラーが出たら、別の方針を自分で考え直して再トライします。

1. AI Agents & Local LLMs

➤ AIエージェント

✓ AIエージェント用OSSフレームワーク比較

フレームワーク	設計思想 (パラダイム)	特徴・強み	最適なユースケース
LangGraph	グラフベース (状態管理)	厳密なワークフロー制御、複雑なループ、高い信頼性	業務プロセスの自動化、カスタマーサポート
CrewAI	ロールベース (役割分担)	直感的な役割定義、人間のような協調、構築の速さ	コンテンツ制作、マーケティング、リサーチ
AutoGen	会話ベース (対話型)	柔軟なエージェント間対話、コード実行、動的解決	ソフトウェア開発、データ分析、技術検証
PydanticAI	スキーマベース (型安全)	型定義による堅牢性、ツール呼び出しの正確性	金融、医療などミスが許されない高度な統合

◆ 2026年に知っておくべきOSSテクノロジー

2. Vector Databases

✓ Vector Databaseとは？

1. 画像、音声、テキストなどの複雑なデータを、意味的な類似性に基づいて検索・処理するために最適化されたデータベース
2. これらの複雑なデータは、数値の配列である「ベクトル」と呼ばれる形式に変換され、データベースに格納
3. このベクトル化されたデータを効率的に管理し、類似性検索を高速に行うためのインデックス構造やアルゴリズムを装備

✓ 特徴

1. 高次元データ処理: 画像、テキスト、音声など、高次元のデータを効率的に処理可能
2. 類似性検索: 従来のキーワード検索ではなく、データの意味的な類似性に基づいて検索可能
3. 高速な検索性能: 大量のデータに対しても、高速な検索が可能
4. AIとの親和性: 機械学習や深層学習などのAI技術との親和性が高く、AIを活用したアプリケーションの開発に役立つ

◆ 2026年に知っておくべきOSSテクノロジー

2. Vector Databases



Qdrant

Rust製で超高速。フィルタリング機能が強力で、本番環境での採用が急増中。



Milvus

スケーラビリティに優れたクラウドネイティブな設計。大規模データセットに最適。



Chroma

開発者体験（DX）を最優先。Pythonエコシステムとの統合が容易で、プロトタイプから製品化までスムーズ。

◆ 2026年に知っておくべきOSSテクノロジー

3. Platform Engineering (IDP)

- ✓ 定義と注目される背景

エンジニアの認知負荷

現代のクラウドネイティブ環境は複雑化し、開発者が「コードを書く」以外のインフラ構築、CI/CD、セキュリティ等の知識を求められすぎている。

セルフサービス化

必要なインフラやツールを、チケット制の待ち時間なしに、開発者が自分たちでオンデマンドに用意できる仕組みを提供する。

- ✓ Internal Developer Platform (IDP)

- プラットフォームの核

- » IDPは、開発者がアプリケーションをデプロイ、実行するために必要なツールやプロセスの集合体。
- » 「開発者のためのポータル」が窓口となり、その背後にある複雑なインフラ操作を抽象化する。

◆ 2026年に知っておくべきOSSテクノロジー

3. Platform Engineering (IDP)

- ✓ 領域別主要 OSS ツールの比較

カテゴリ	代表的な OSS	プラットフォームにおける役割
Developer Portal	Backstage	サービスカタログ、ドキュメント、開発ポータル
Infrastructure	Kubernetes / Crossplane	コンテナ実行基盤、クラウド資源の抽象化
CI/CD & GitOps	ArgoCD / Tekton	自動デプロイ、継続的デリバリーの自動化
Policy Engine	Kyverno / OPA	ガバナンス、セキュリティポリシーの自動適用

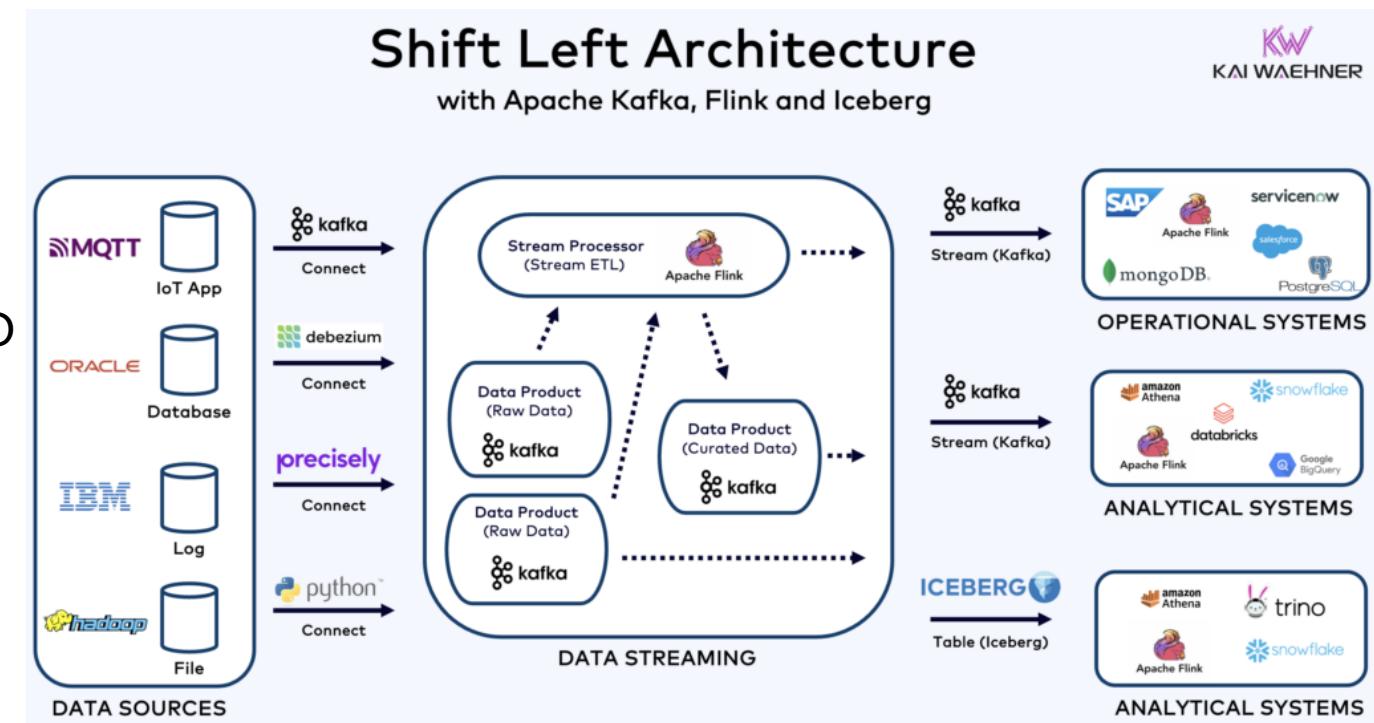
◆ 2026年に知っておくべきOSSテクノロジー

4. Modern Data Stack

データレイクハウスの確立

特定のベンダーにロックインされない
 「オープンテーブルフォーマット」が
 2026年のデータ基盤の主役に。

- **Apache Iceberg:**
 高性能なテーブルフォーマット。ACID
 トランザクションとスキーマ進化を
 サポート。
- **DuckDB:**
 インプロセスOLAPデータベース。
 ローカルでの高速データ分析における
 ゲームチェンジャー。



◆ 2026年に知っておくべきOSSテクノロジー

4. Modern Data Stack

✓ Apache Iceberg

- Netflixによって開発された、巨大な分析データセットのためのオープンなテーブルフォーマットで、現在、Apache Software Foundationのトップレベルプロジェクト
- 従来のデータレイクの課題
 - » 一貫性の欠如: 複数のユーザーやプロセスが同時にデータを書き込もうとすると、データが壊れやすい。
 - » パフォーマンスの低下: データ量が増えるにつれて、必要なデータを探し出すのに時間がかかる。
 - » スキーマ変更の困難さ: テーブル構造の変更が難しく、エラーの原因になりやすい。

特定の技術に縛られず、オープンで拡張性の高いデータ分析基盤を構築可能に

◆ 2026年に知っておくべきOSSテクノロジー

4. Modern Data Stack

✓ Apache Iceberg

- 主な特徴とメリット

» 信頼性の高いトランザクション (ACID特性):

» データレイク上のデータに対しても、データベースのように安全なデータの読み書き（コミット、ロールバックなど）が可能。これにより、データの破損や不整合を防ぐことが可能。

» スキーマ進化のサポート:

» ビジネスの成長に合わせてテーブルの列を追加、削除、名前変更、型変更などを安全に行うことが可能。古いデータ形式と新しいデータ形式が混在していても、問題なくクエリを実行できます。

» Time Travel (タイムトラベル) :

» 過去のある時点のテーブルの状態を再現し、クエリを実行することができます。これにより、誤った更新からの回復や、過去のデータ分析が容易に。

» パフォーマンスの向上:

» Icebergはデータのメタ情報を効率的に管理し、不要なファイルをスキャンしないようにすることで、クエリのパフォーマンスを大幅に向上（Partition Pruning、述語プッシュダウンなど）。

» 多様なエンジンとの連携:

» Apache Spark, Flink, Presto, Trino, Hiveなど、多くの分散処理エンジンから利用することが可能。特定のエンジンに縛られることがない。

◆ 2026年に知っておくべきOSSテクノロジー

4. Modern Data Stack

✓ Apache Iceberg

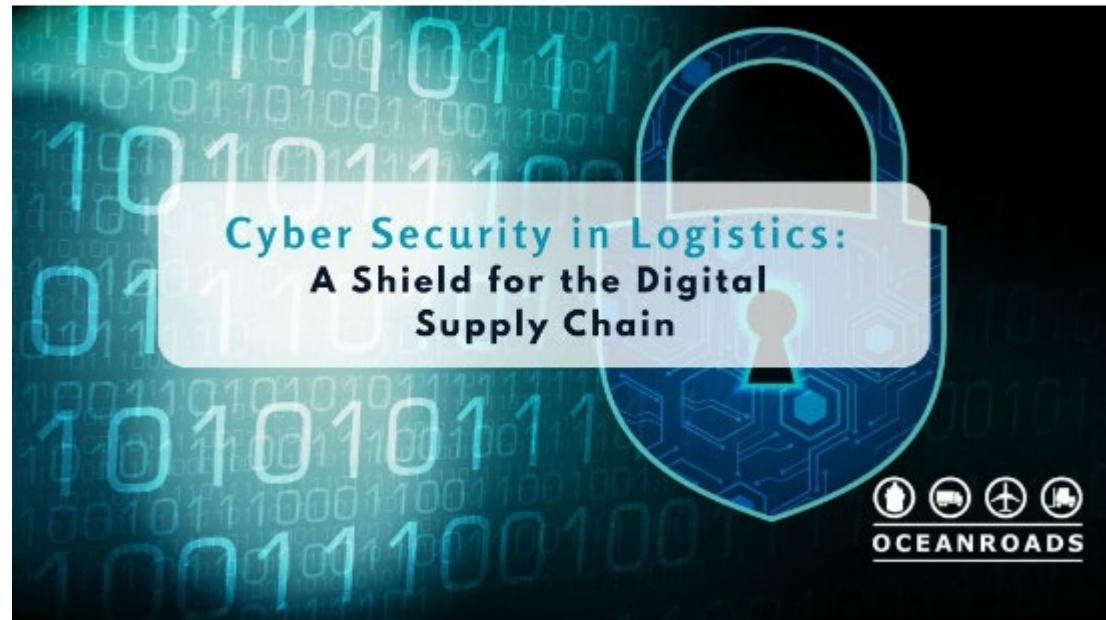
– Apache Icebergをサポートする主要なOSS

ソフトウェア名	サポート状況と特徴
Apache Spark	最も重要なサポートエンジン。Icebergの読み書き、スキーマ変更、パーティション操作など、ほぼ全ての機能をサポートしています。Icebergプロジェクト自体がSpark用のライブラリを公式に提供しています。
Trino (旧 PrestoSQL)	高速な分散SQLクエリエンジン。Icebergコネクタを通じて、Icebergテーブルに対するインタラクティブな分析クエリを高速に実行できます。
PrestoDB	Trinoの元となった分散SQLクエリエンジン。同様にIcebergをサポートしており、大規模データセットの分析に利用されます。
Apache Hive	伝統的なデータウェアハウス基盤。Hive 4.0以降でIcebergのサポートが統合され、既存のHive環境からIcebergテーブルを操作できます。
Apache Flink	ストリーミングとバッチの両方を処理できるフレームワーク。Icebergをシンク（出力先）およびソース（入力元）として利用でき、リアルタイムデータパイプラインの構築に強力です。
Dremio	データレイクハウスプラットフォーム。Icebergをネイティブフォーマットとして強力にサポートしており、クエリの高速化やセルフサービスBIを実現します。
ClickHouse	高速な分析用列指向データベース。Icebergテーブルを直接クエリする機能のサポートが進んでいます。

◆ 2026年に知っておくべきOSSテクノロジー

5. Security First : Supply Chain & SBOM

"Secure by Design" の徹底



ソフトウェアサプライチェーン攻撃への対策が必須化。開発プロセスの透明性が求められる。

- **Sigstore:** ソフトウェアの署名と検証を民主化。コードの出所を確実に保証する。
- **OpenSSF Scorecard:** オープンソースプロジェクトのセキュリティリスクを自動評価・スコアリング。
- **SBOM (Software Bill of Materials):** ソフトウェア部品表の生成と管理が標準プロセスに統合。

最後に

◆ アップトプット共創型 エンジニアコミュニティ

LinuC Open Network(<https://linuc.community/>)のご紹介

➤ 主なプロジェクト

- ✓ 認定試験開発プロジェクト
- ✓ 学習教材作成プロジェクト
- ✓ 学習支援活動プロジェクト
- ✓ 新技術社会実装プロジェクト

LinuC Open Networkとは



アウトプット共創型エンジニアコミュニティ

LinuC Open Network（LiON）は、ITエンジニアの成長と活躍を後押しすることを目的に、NPO法人LPI-Japanのサポートのもと、有志によって運営されているオープンなコミュニティです。どなたでもご参加いただけます。

END

初心者歓迎！
2025年のOSSを振り返る

2026/01/31
ワイズプランニング

吉田 行男