

2023年度

「アジャイルプロジェクトの実態」に関するアンケート

- ◆ データ編
- ◆ 分析と提言編

PMI日本支部 アジャイル研究会

はじめに

本資料は、2023年3月1日～3月31日にPMI日本支部アジャイル研究会が実施した、「2023年度「アジャイルプロジェクトの実態」に関するアンケート」の調査結果、およびその分析を踏まえた提言です。

調査にご協力いただいた皆さま、ありがとうございました。

一般社団法人PMI日本支部
アジャイル研究会

過去の報告

過去の報告は下記にて公開しております。

PMI日本支部 > アジャイル研究会 > アンケート調査結果

<https://www.pmi-japan.org/agilesg/survey/>



「プロジェクトマネジメント研究報告」で
論文形式でもご覧いただけます。

プロジェクトマネジメント研究報告 Vol.3 No.1 2023
2022年度アジャイルプロジェクトマネジメント実態調査報告
－アジャイル成熟に向けたシステムライフサイクルプロセス活用のすすめ－
https://doi.org/10.57323/pmijapan.3.1_14

プロジェクトマネジメント研究報告 Vol.2 No.1 2022
2021年度 アジャイルプロジェクトマネジメント実態調査報告
－作る方法というものを開発していくこと－
https://doi.org/10.57323/pmijapan.2.1_65

プロジェクトマネジメント研究報告 Vol.1 No.1 2021
2020年度 アジャイルプロジェクトマネジメント実態調査報告
－コンテキストに応じた多様性のあるアプローチの提言－
https://doi.org/10.57323/pmijapan.1.1_79

目次

<データ編>

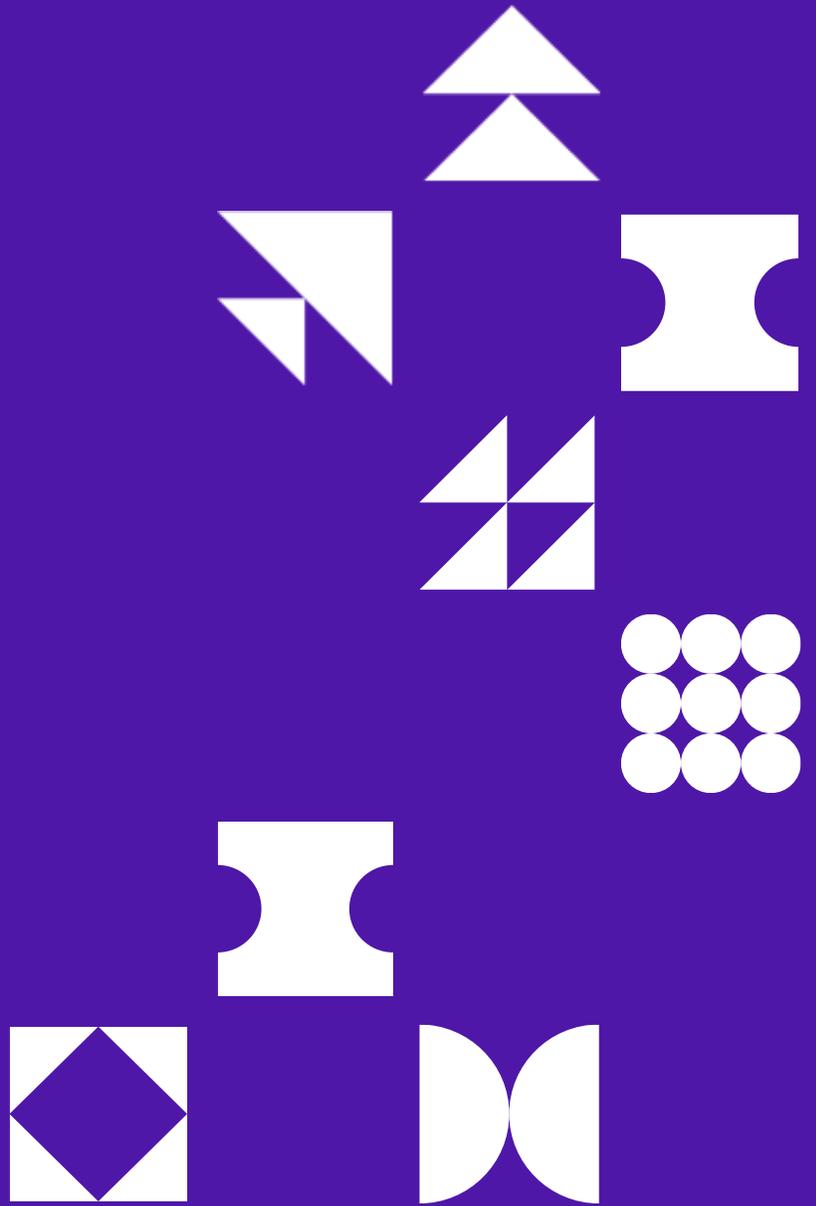
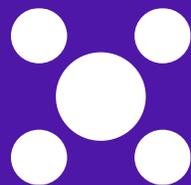
- 調査概要
- 回答者の属性
- 関連資格等の状況
- 組織のプロジェクトへの取組み
- 回答者のアジャイルへの認識
- 回答者の経験したアジャイル適用業務と非アジャイルソフトウェア開発
- 親しい友人や同僚に薦める可能性はどのくらいありますか？(ネット・プロモーター・スコア)

<分析と提言編>

- 今年も推奨されなかった“アジャイル” アンケート結果の振り返り
- 原則の実践
- おわりに

2023年

「アジャイルプロジェクトの実態」
に関するアンケート<データ編>



<データ編>

- ▶ 調査概要
- ▶ 回答者の属性
- ▶ 関連資格等の状況
- ▶ 組織のプロジェクトへの取組み
- ▶ 回答者のアジャイルへの認識
- ▶ 回答者の経験したアジャイル適用業務と非アジャイルソフトウェア開発
- ▶ 親しい友人や同僚に薦める可能性はどのくらいありますか？
(ネット・プロモーター・スコア)

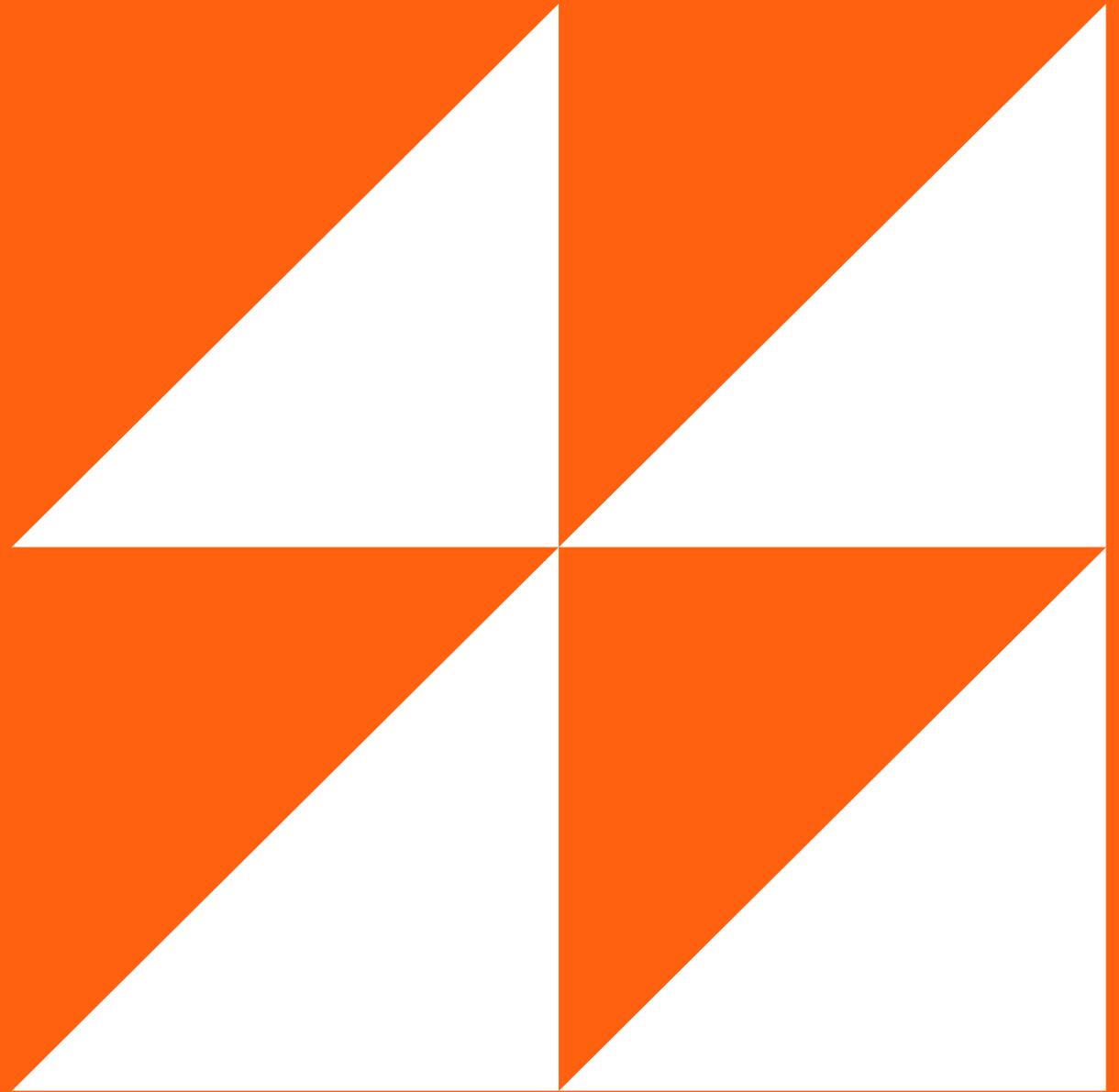
分析と提言編 へ続く

調査概要

- 目的
 - アジャイル プロジェクト マネジメントの導入／展開を推進するための具体策を研究会で検討し、広く情報発信していく。
- 調査方法
 - Webアンケートによる調査
- 調査期間
 - 2023年3月1日 - 2023年3月31日
- 調査対象
 - 有効回答数 271名（2022年度304名,2021年度341名,2020年度226名）

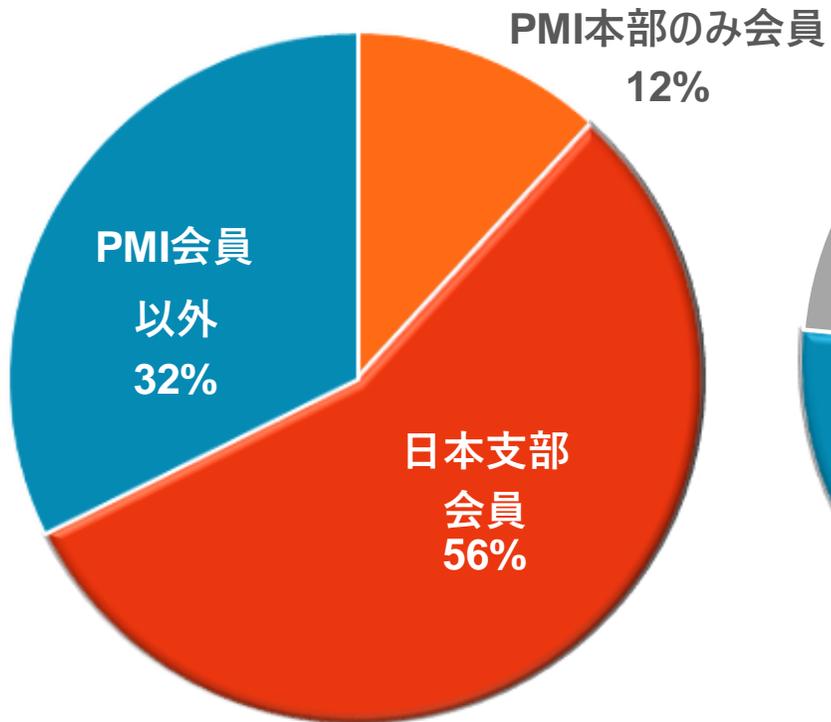


答者の属性

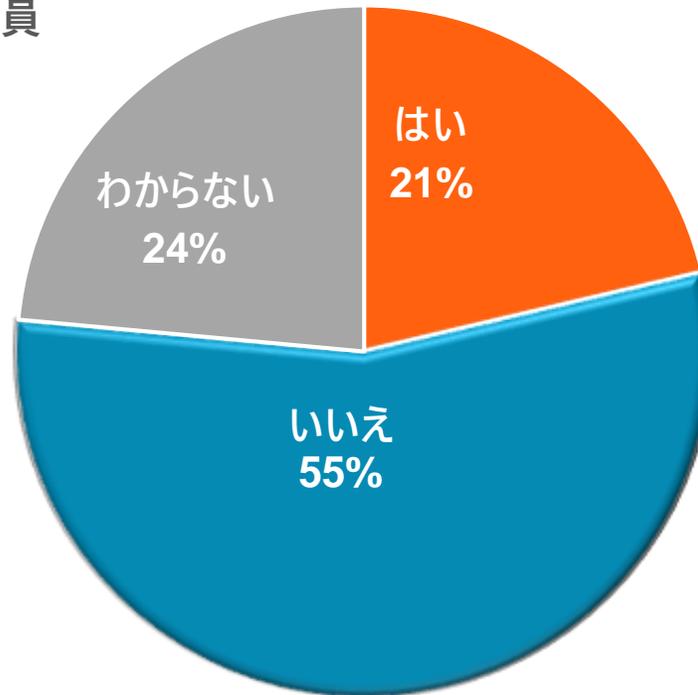


回答者の属性 | PMIとの関係

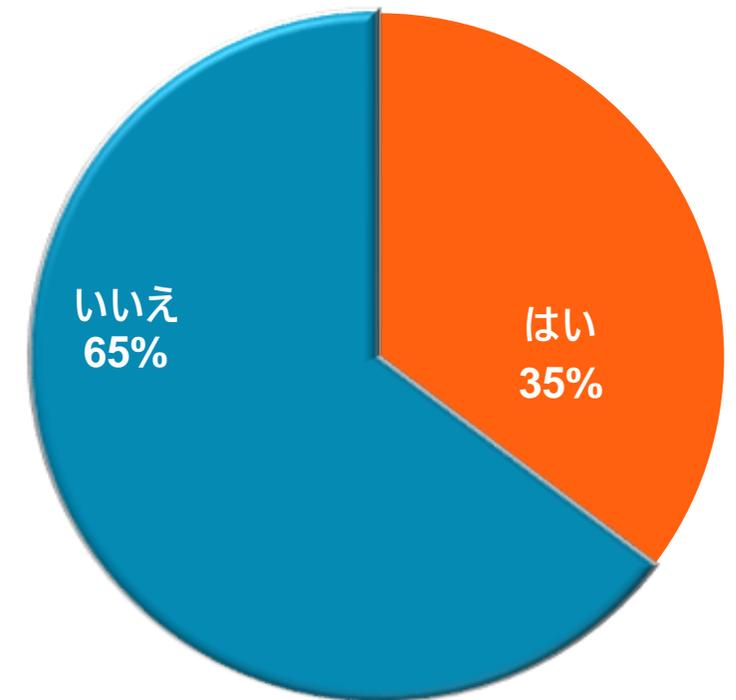
PMI会員の割合



所属組織が法人スポンサー

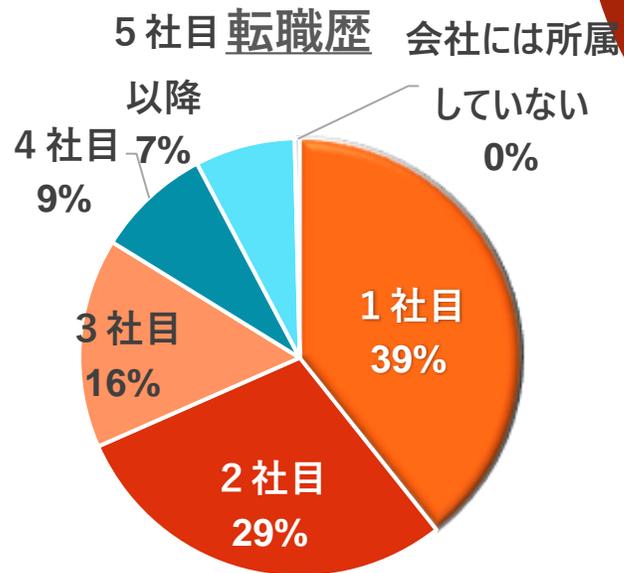
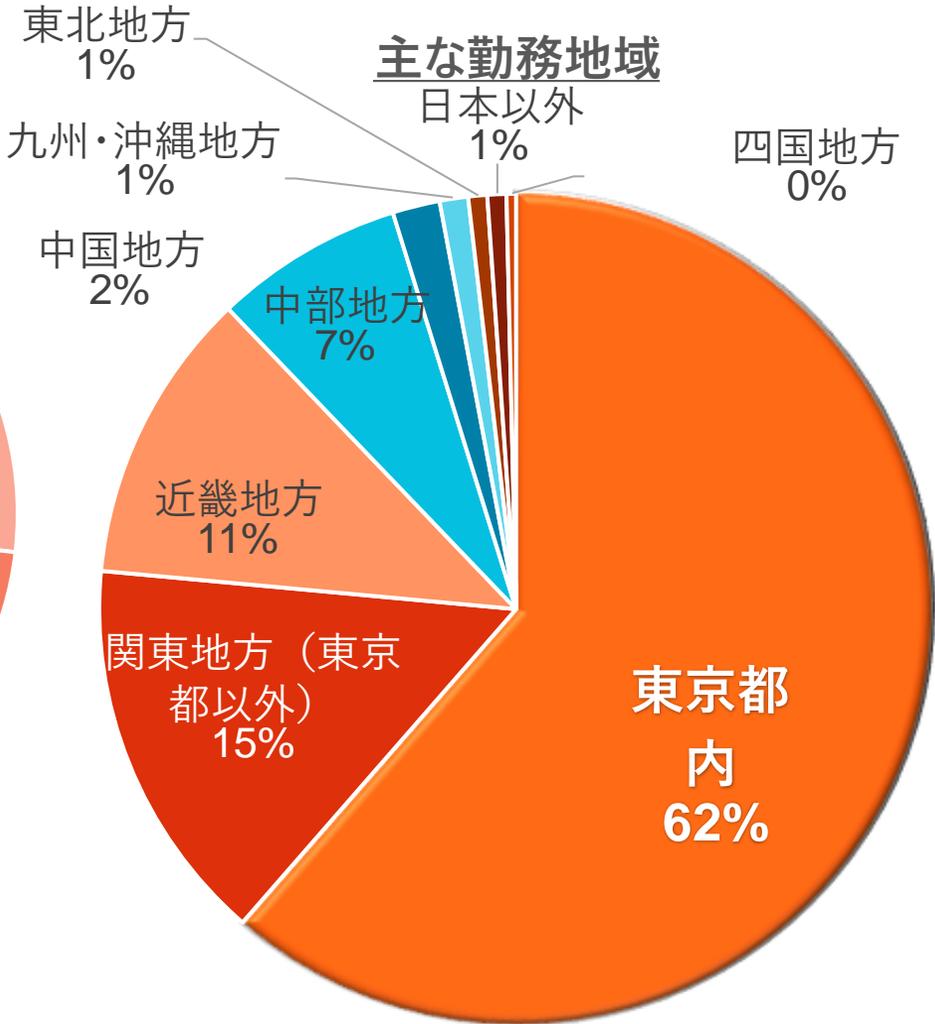
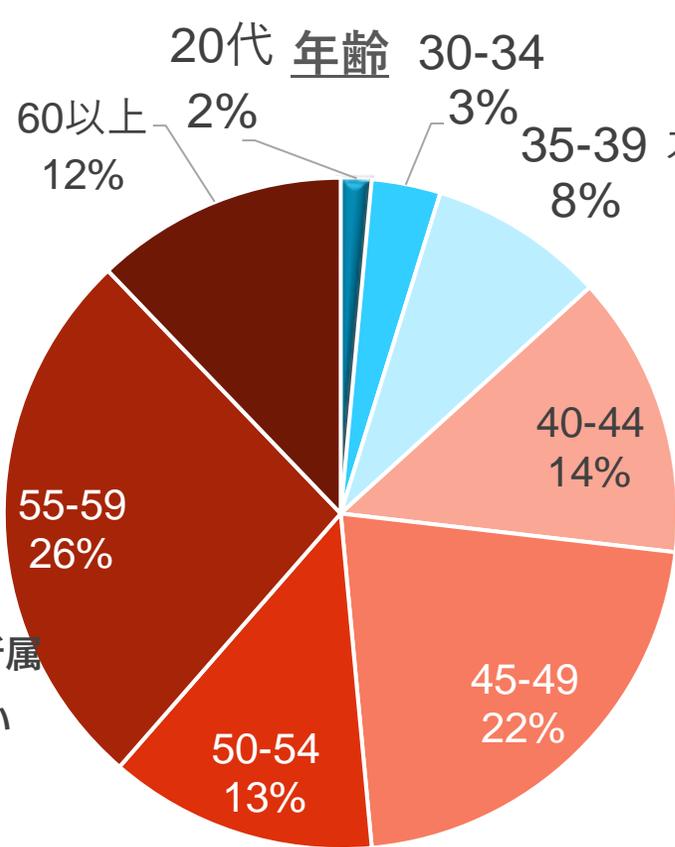
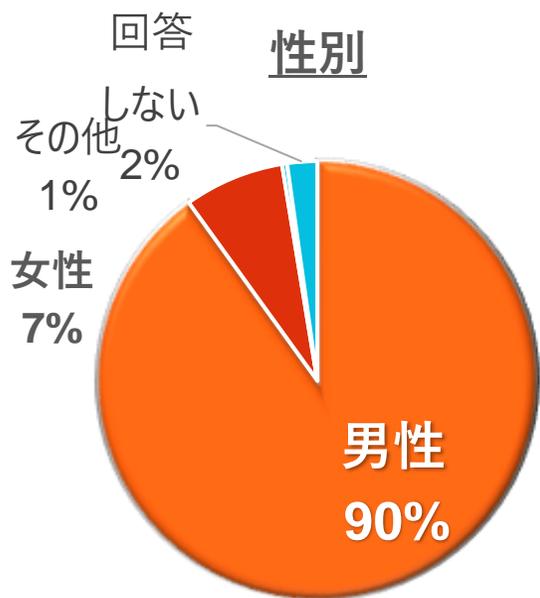


過去にこのアンケートに回答



n=271

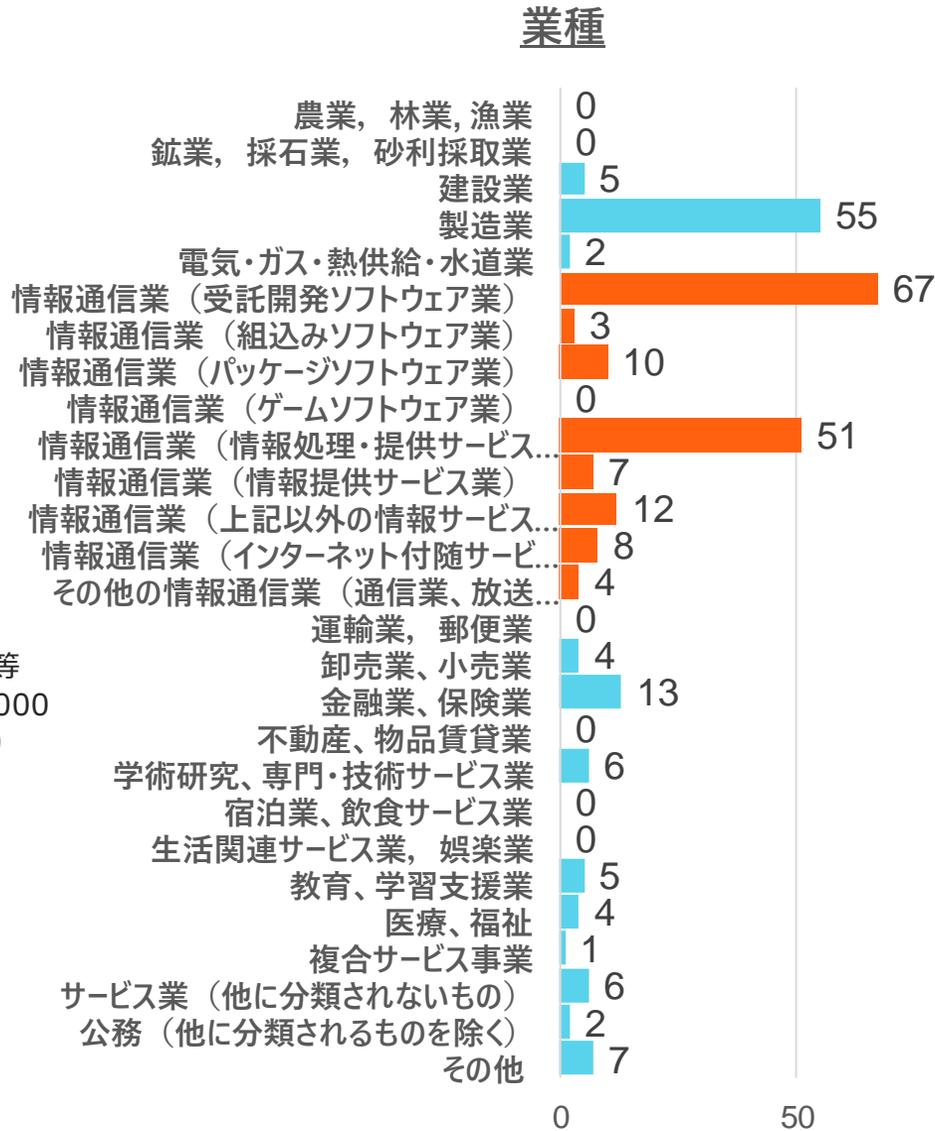
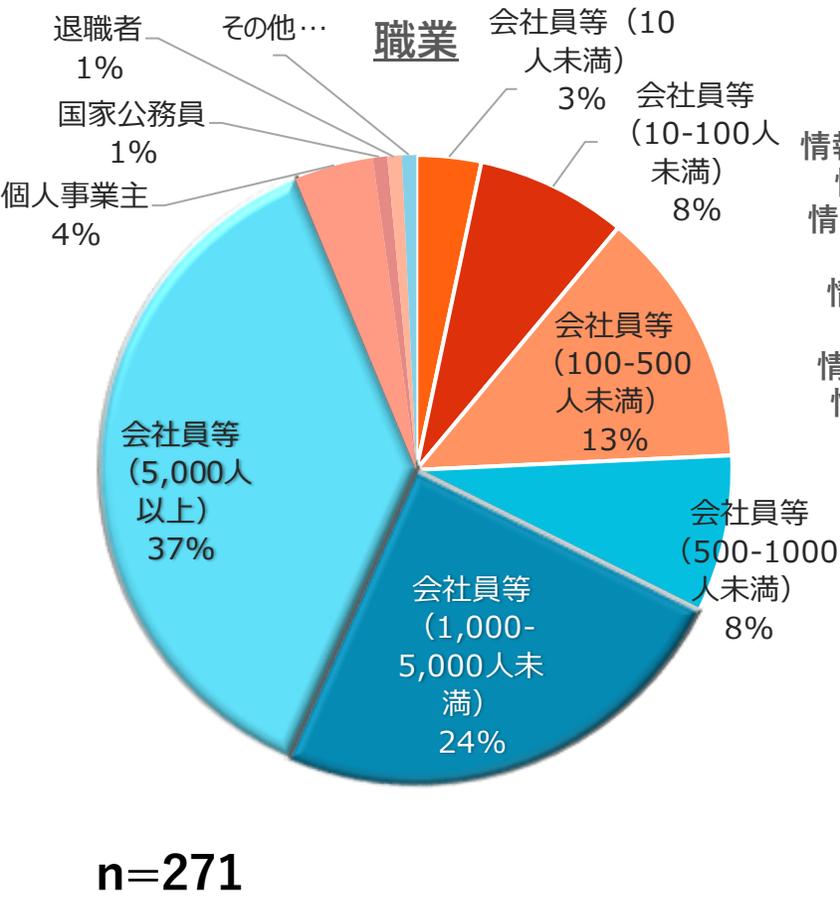
回答者の属性 | 性別・年齢と勤務地・転職歴



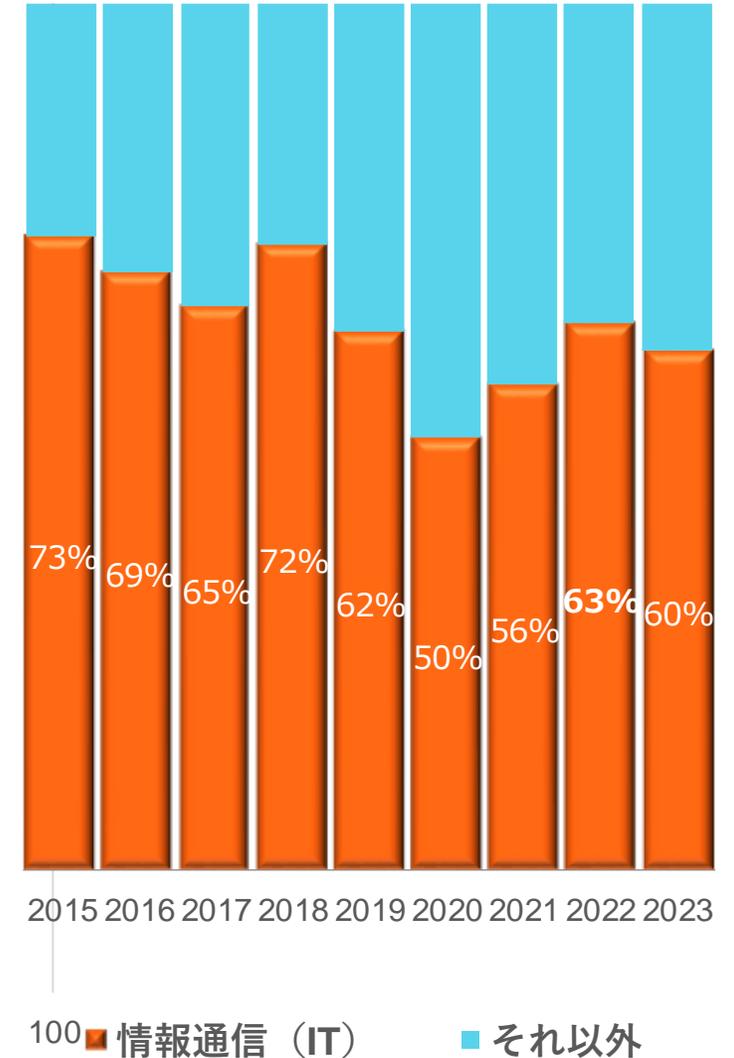
n=271



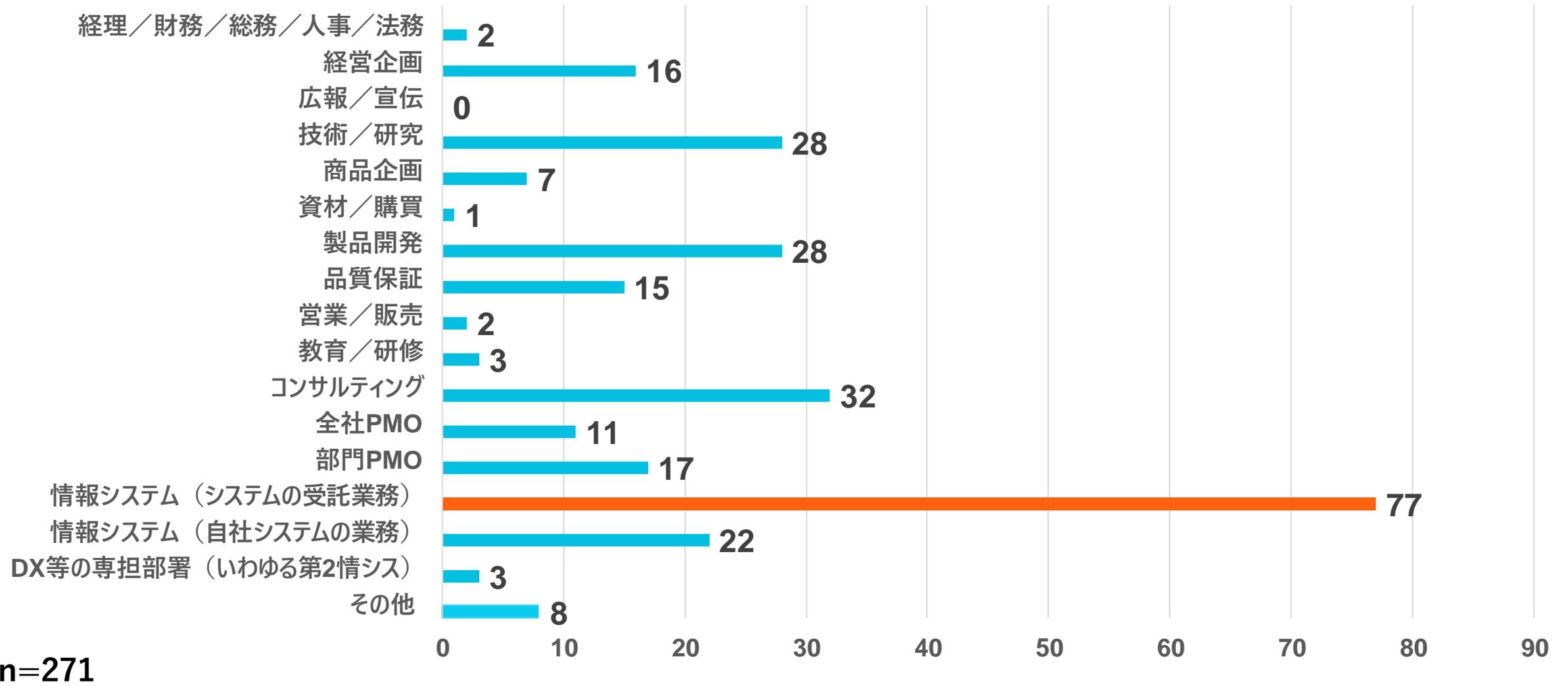
回答者の属性 | 職業と業種



情報通信業の回答者の割合の推移

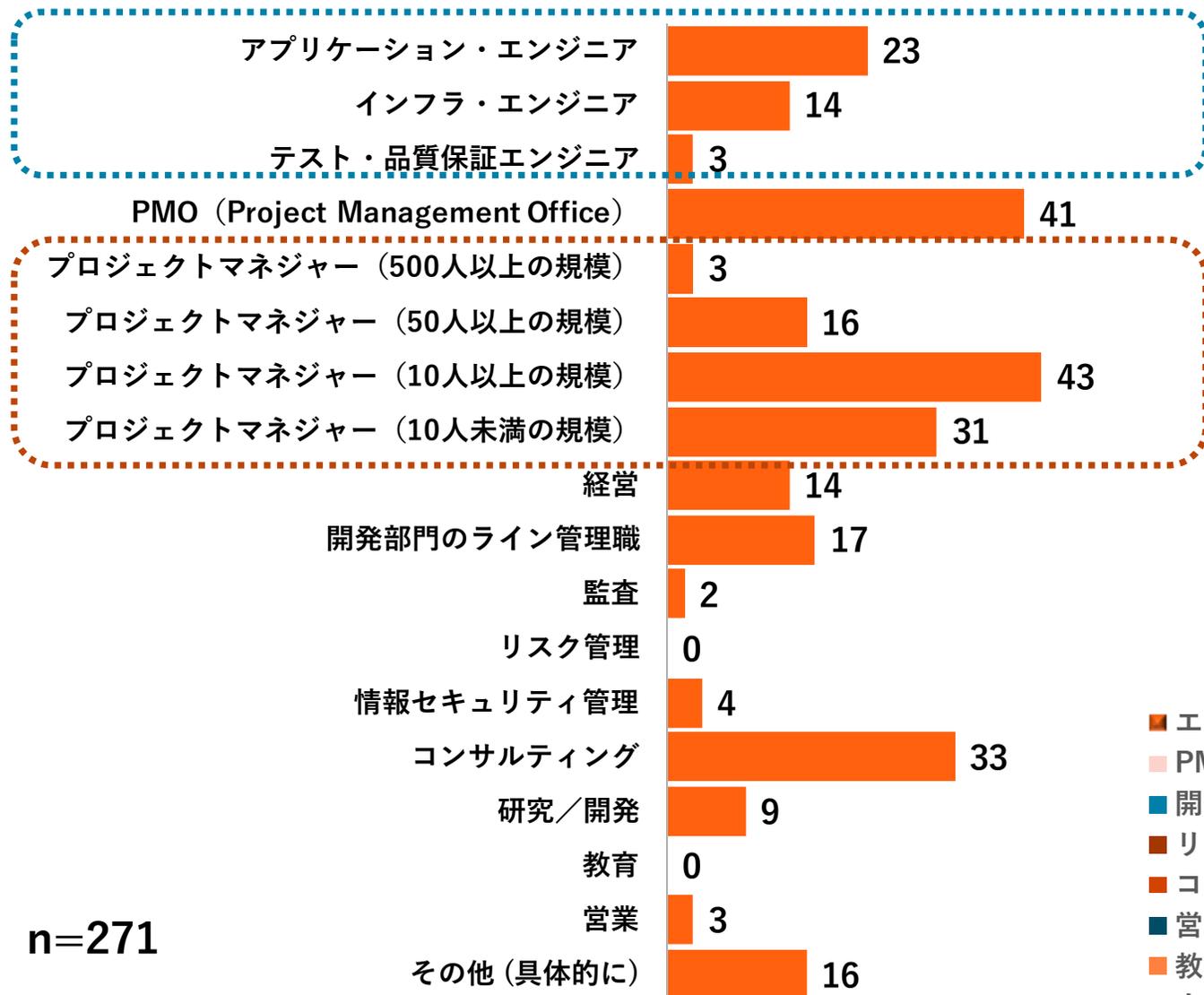


回答者の属性 | 所属部門



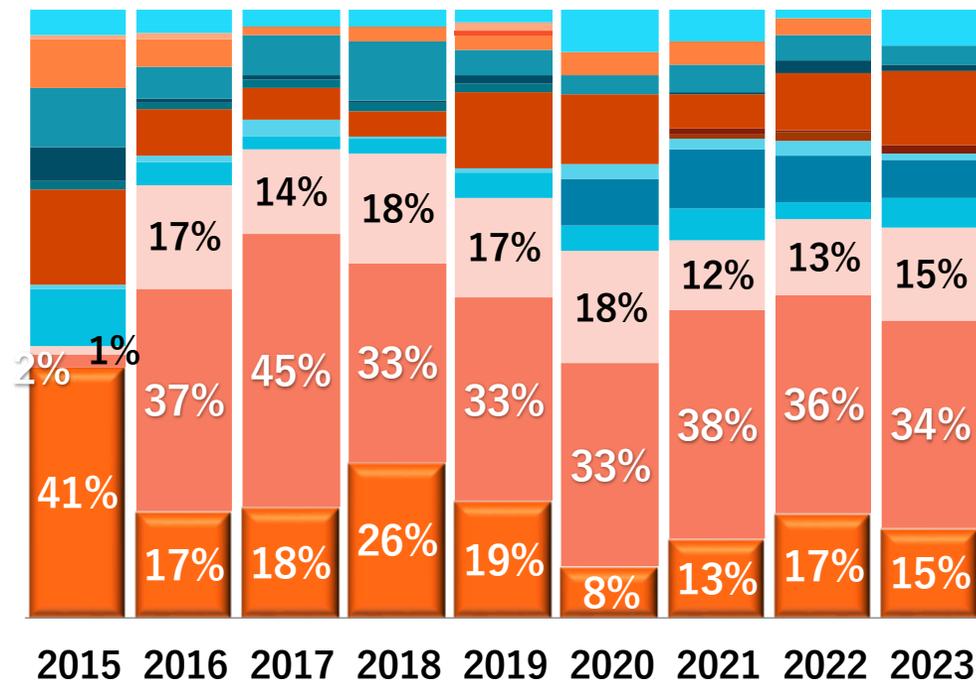
回答者の属性 | 職種

約半分がPMOまたはプロジェクトマネージャー



n=271

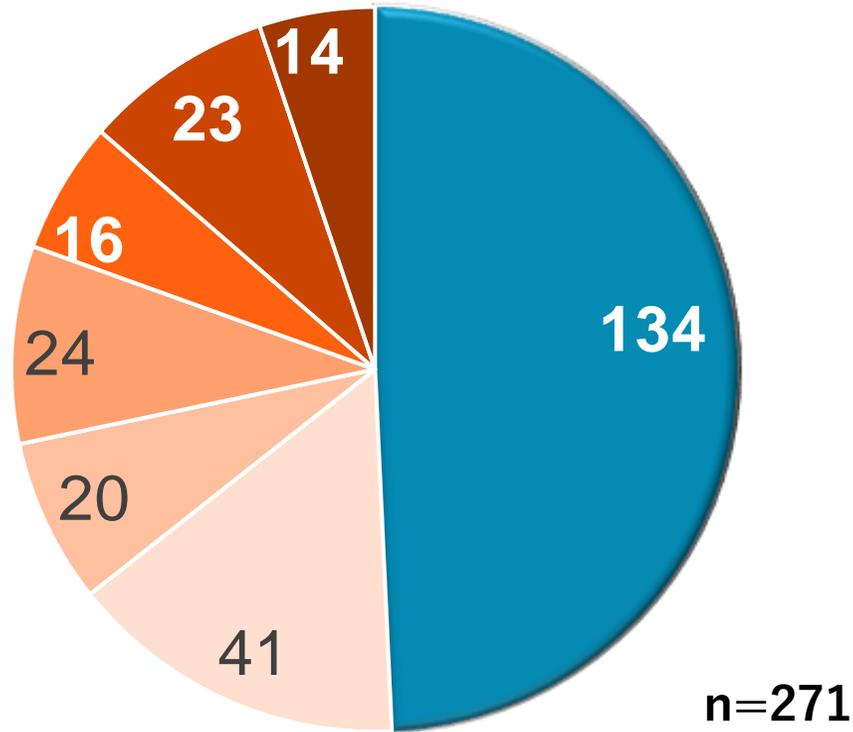
(参考) 職種の推移



- エンジニア
- PMO
- プロジェクト・マネージャー
- 経営
- 監査
- 開発部門のライン管理職
- リスク管理
- 情報セキュリティ管理
- マーケティング
- 研究/開発
- 財務/経理
- 人事/総務
- その他

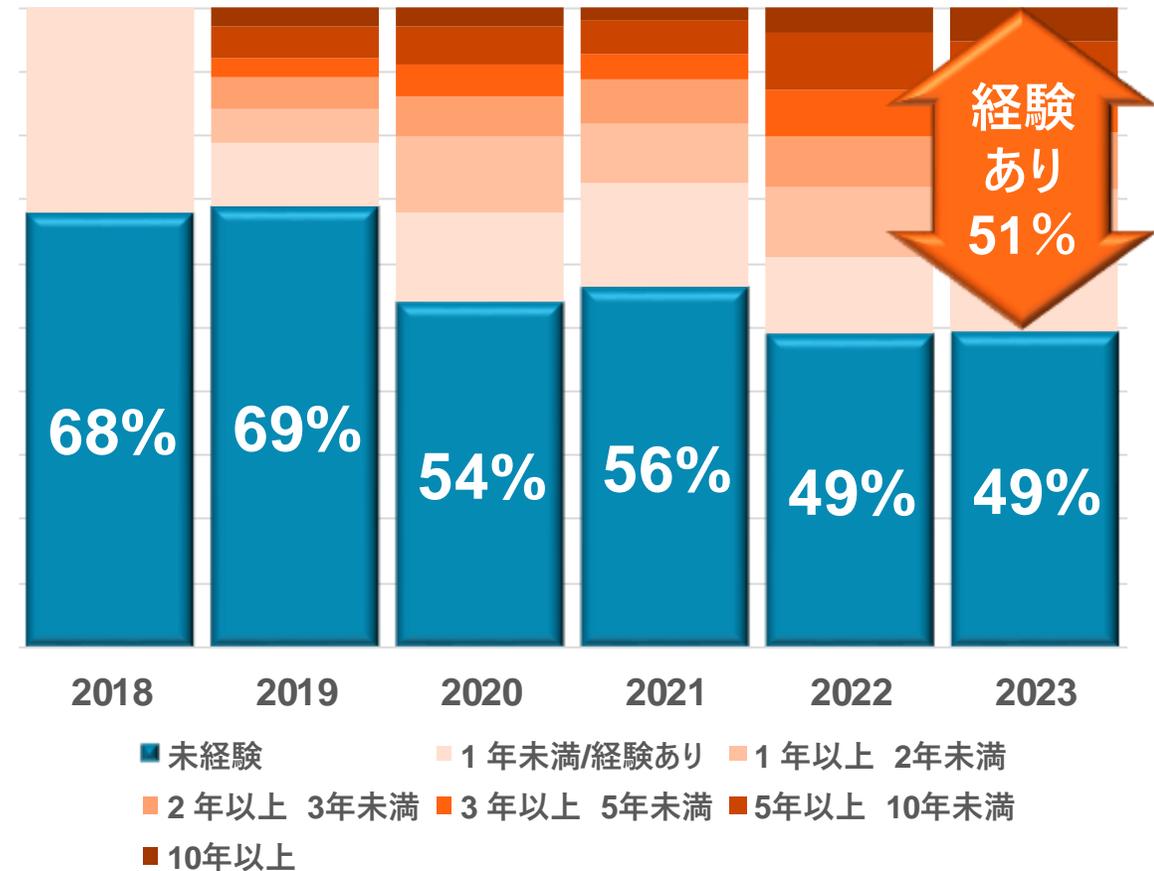
回答者の属性 | アジャイル経験年数

アジャイル経験年数



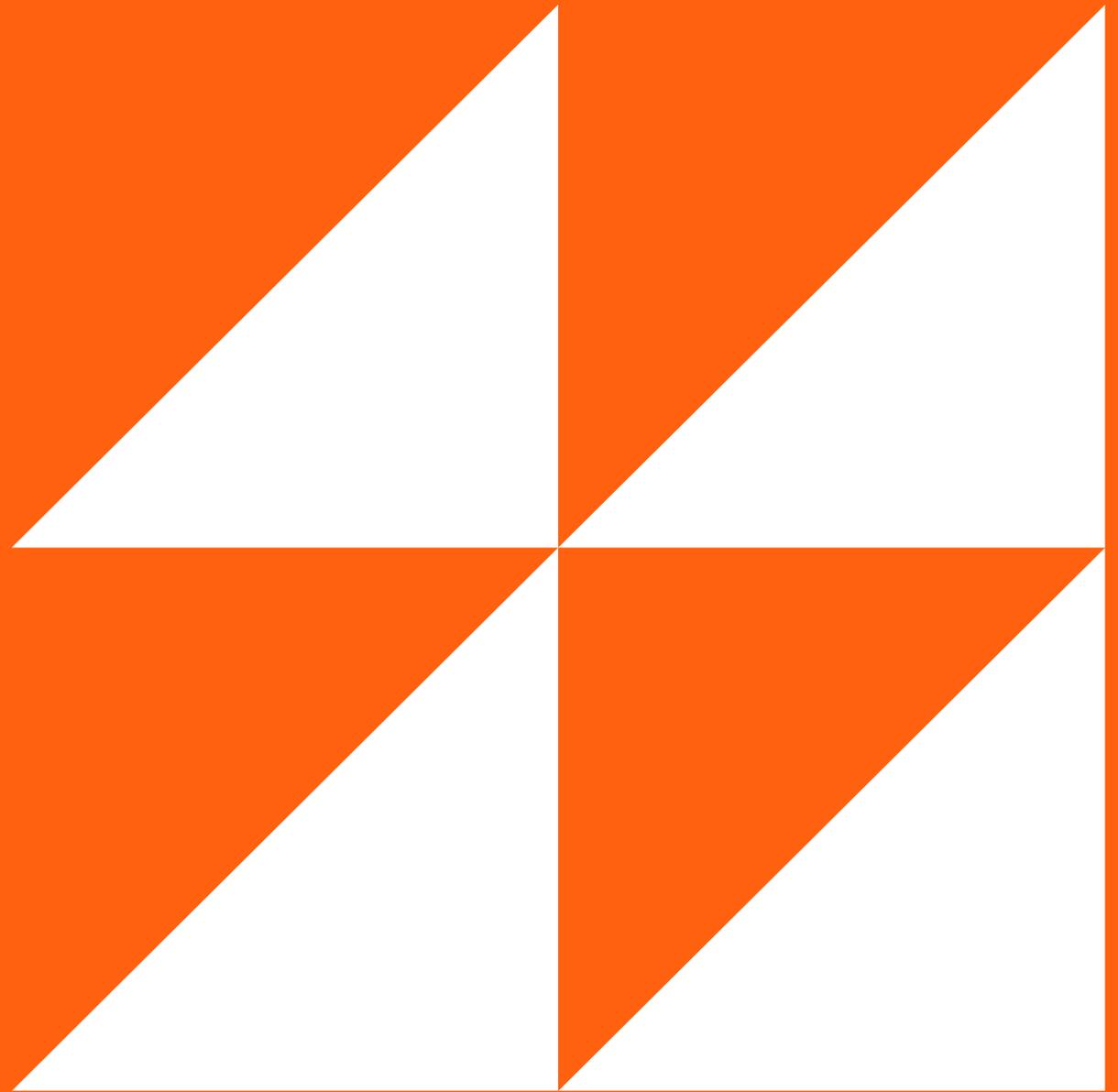
- 未経験
- 1年未満
- 1年以上 2年未満
- 2年以上 3年未満
- 3年以上 4年未満
- 5年以上 10年未満
- 10年以上

(参考) アジャイル経験年数の推移



- 未経験
- 1年未満/経験あり
- 1年以上 2年未満
- 2年以上 3年未満
- 3年以上 5年未満
- 5年以上 10年未満
- 10年以上

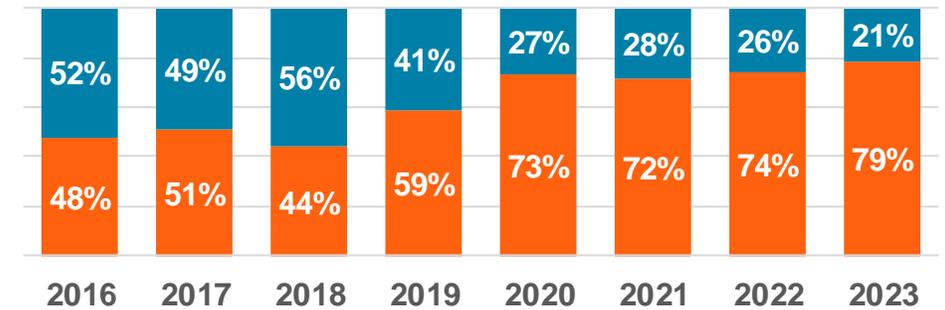
関連資格等の状況



関連資格の状況（PMPおよびPMIアジャイル関連資格の認知度）

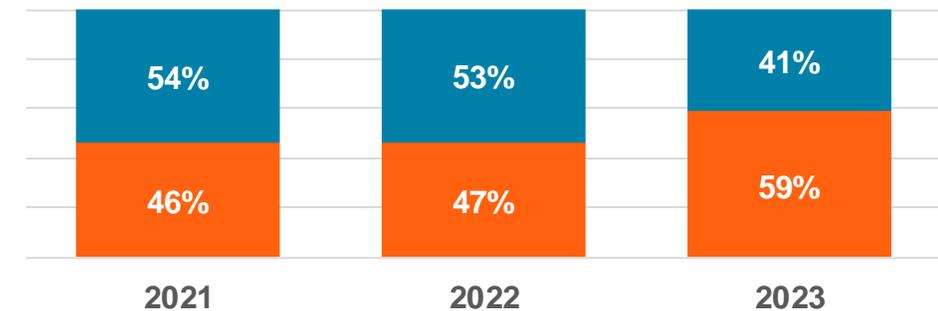
ACP; Agile Certified Practitioner,
 PMP; Project Management Professional,
 DASM; Disciplined Agile Scrum Master
 DASSM; Disciplined Agile Senior Scrum Master
 DAC; Disciplined Agile Coach
 DAVSC; Disciplined Agile Value Stream Consultant

PMP-ACP®認知度の推移

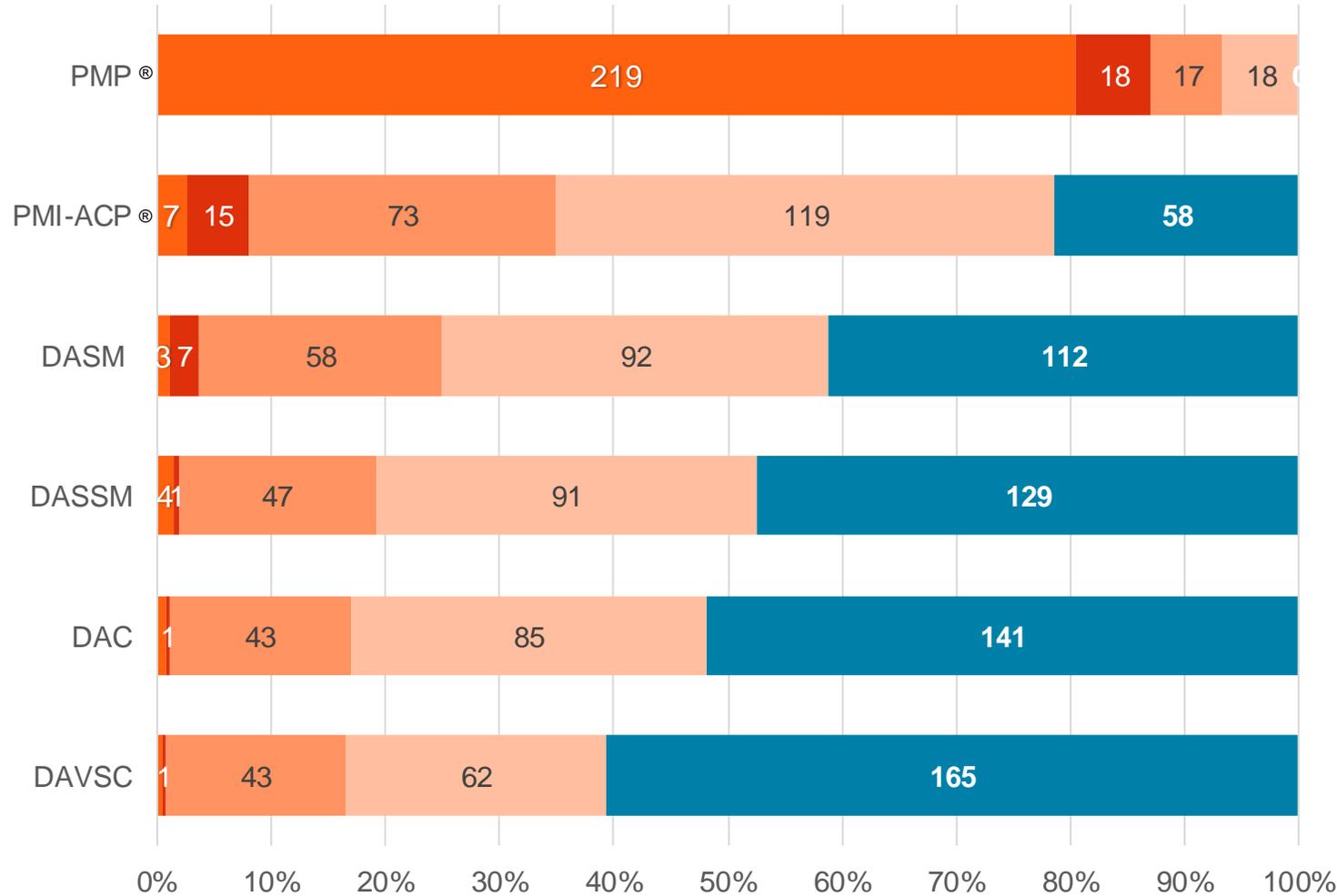


■ 知っている ■ 知らない

DASM 認知度の推移



■ 知っている ■ 知らない

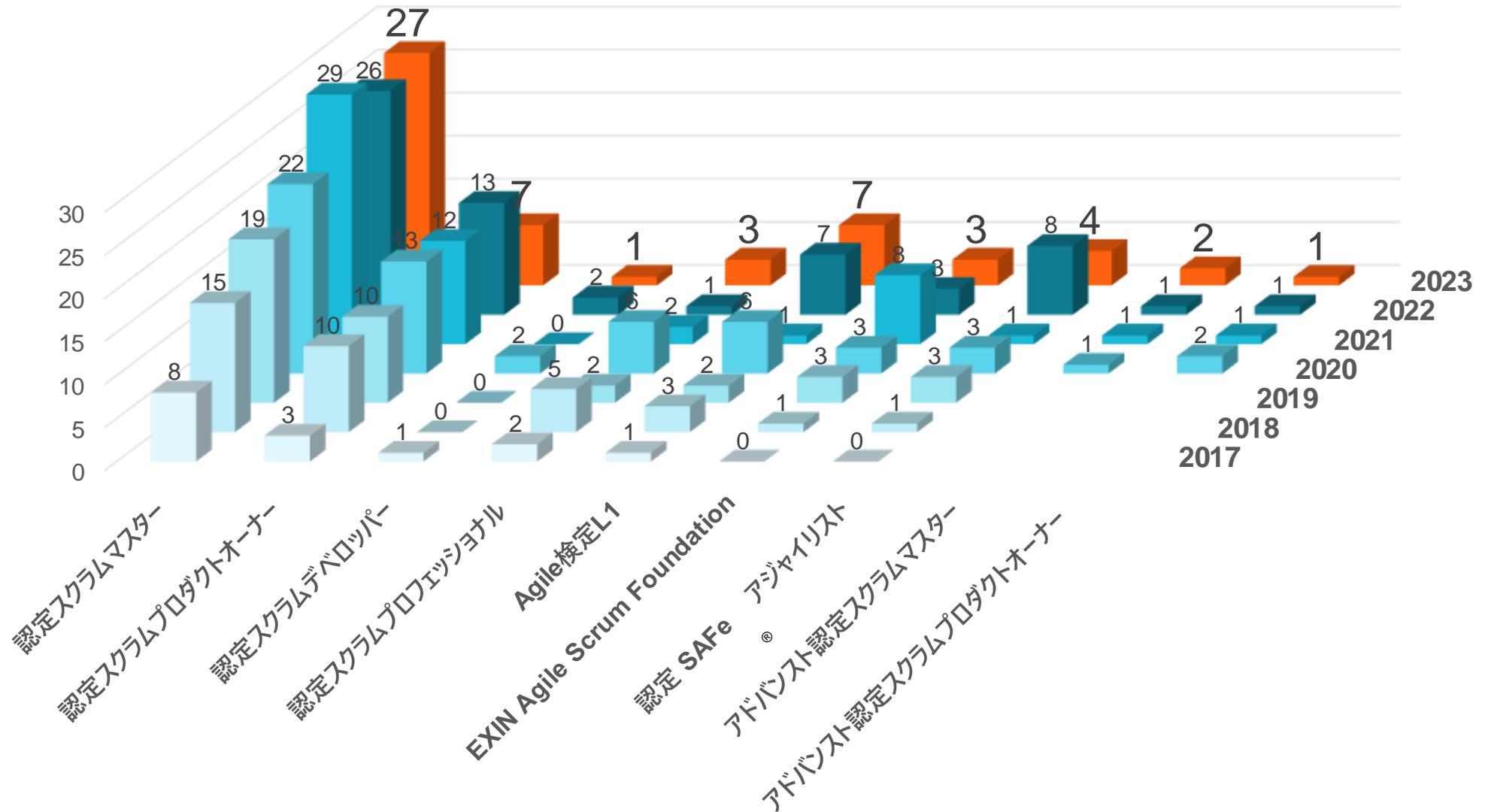


n=271

■ 取得している ■ 取得予定 ■ 興味がある ■ 知っている ■ 知らない



関連資格の状況 | その他のAgile関連資格の取得状況

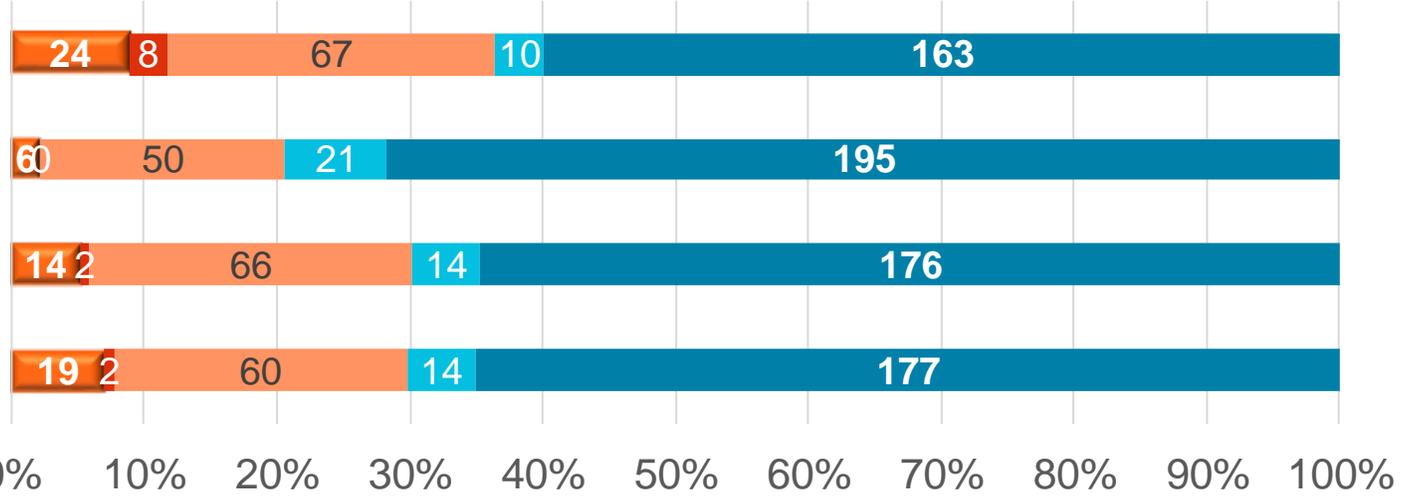


n=271

ディシプリンド・アジャイルについて | ディシプリンド・アジャイル関連書籍

n=271

Choose your WoW

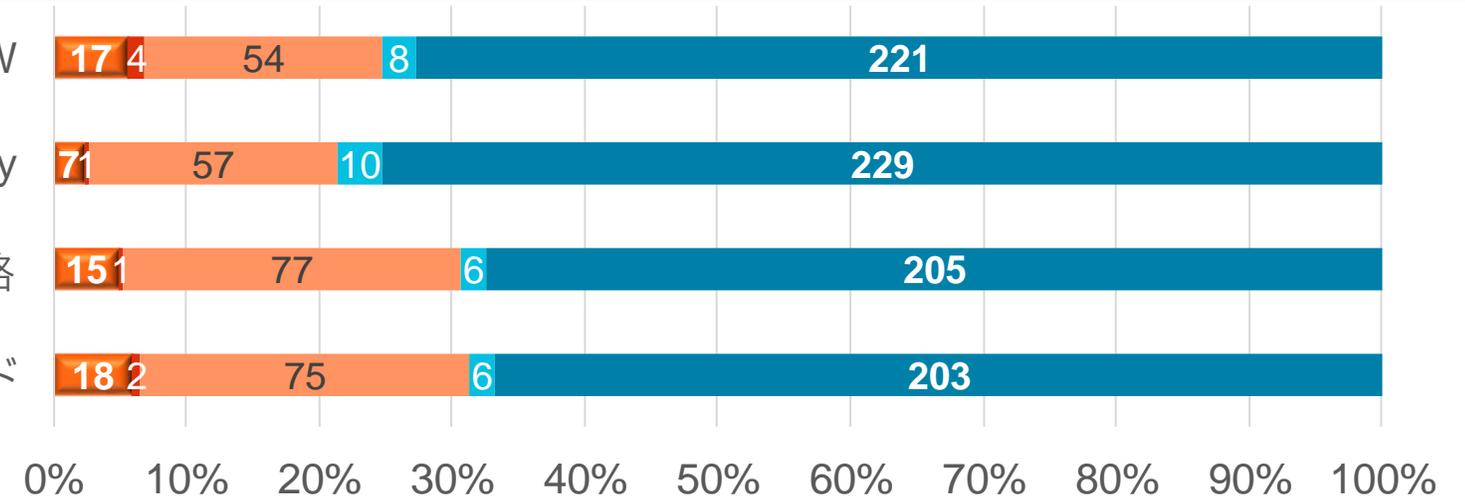


■ 読んだことがある
 ■ 読んでいる途中
 ■ 興味がある
 ■ 興味はない
 ■ 知らない

(参考) 2022年

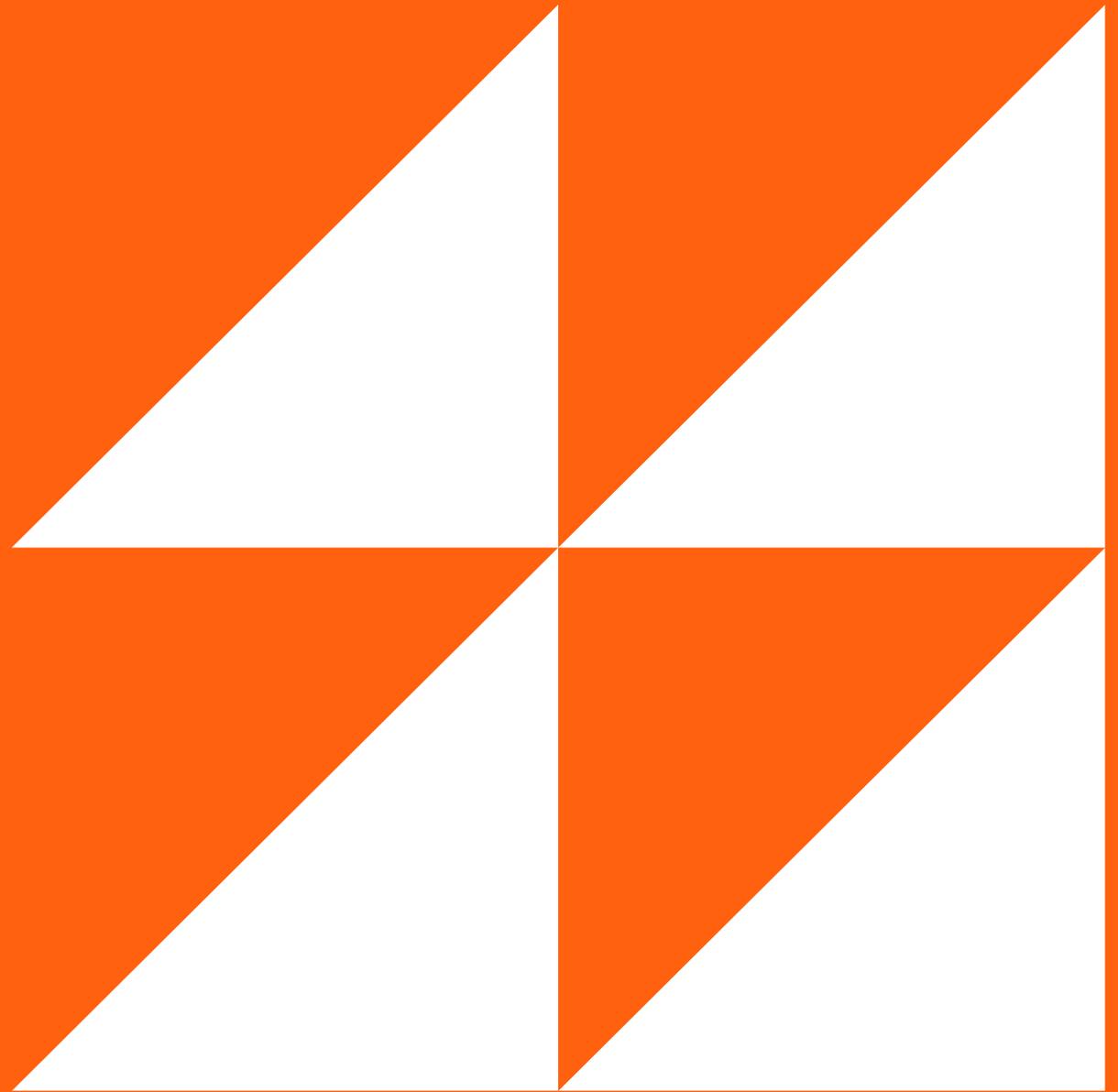
n=304

Choose your WoW

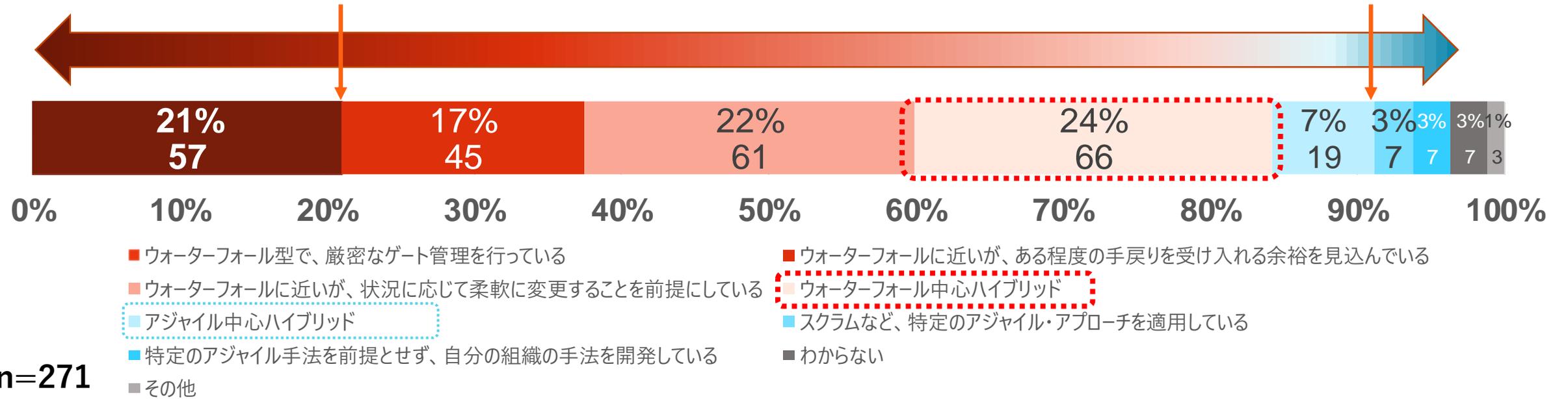


■ 読んだことがある
 ■ 読んでいる途中
 ■ 興味がある
 ■ 興味はない
 ■ 知らない

組織の プロジェクトへの取組み

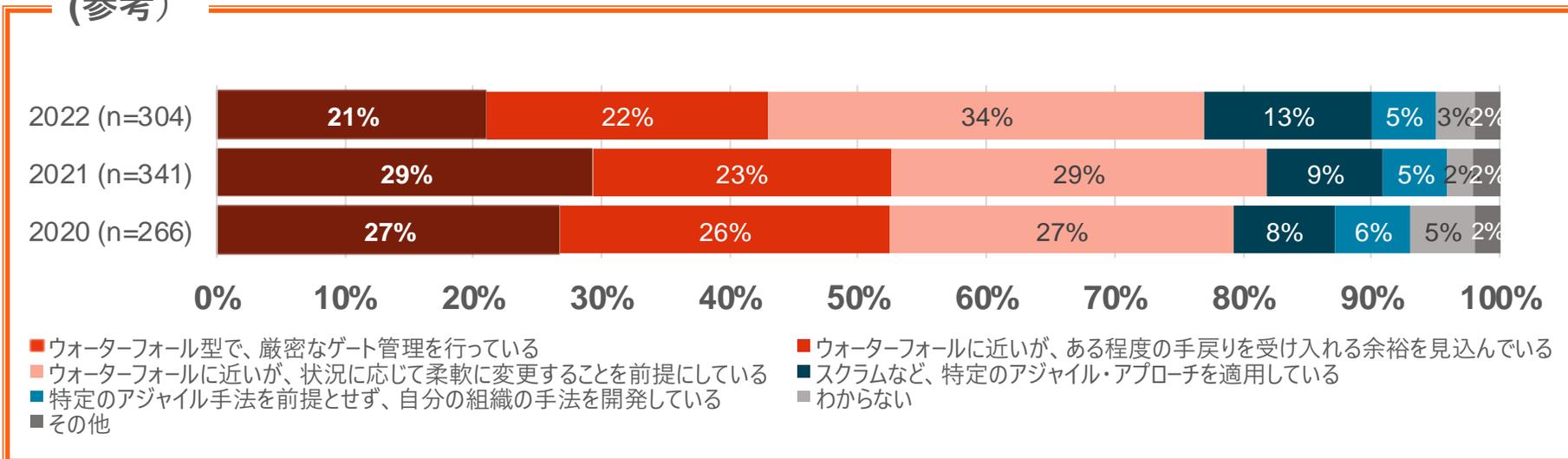


組織のプロジェクトへの取組み | 主要なプロジェクトマネジメント・アプローチ

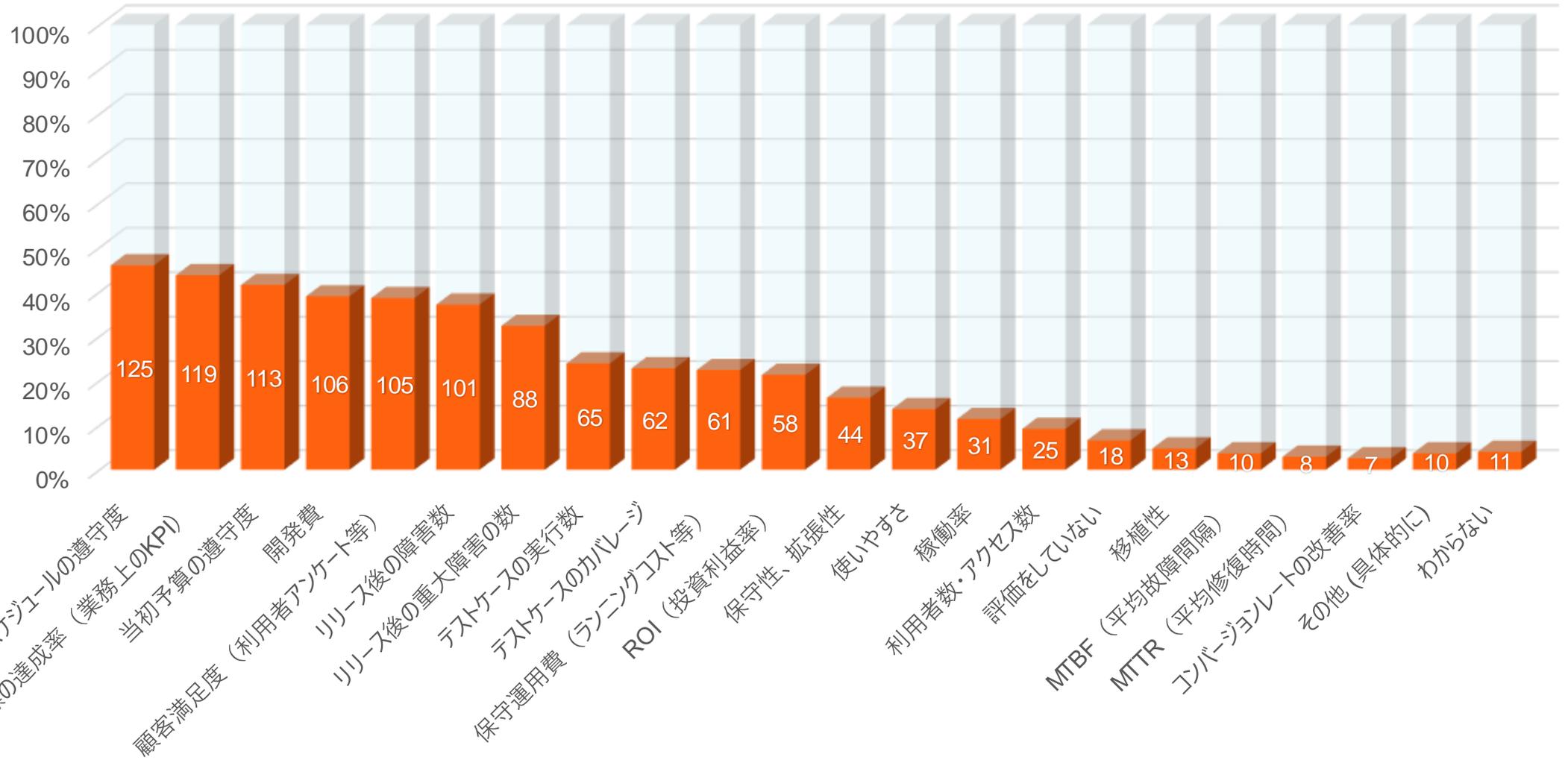


(参考)

※ 昨年までの選択肢に加え、「ウォーターフォール中心ハイブリッド」、「アジャイル中心ハイブリッド」の2項目を追加した



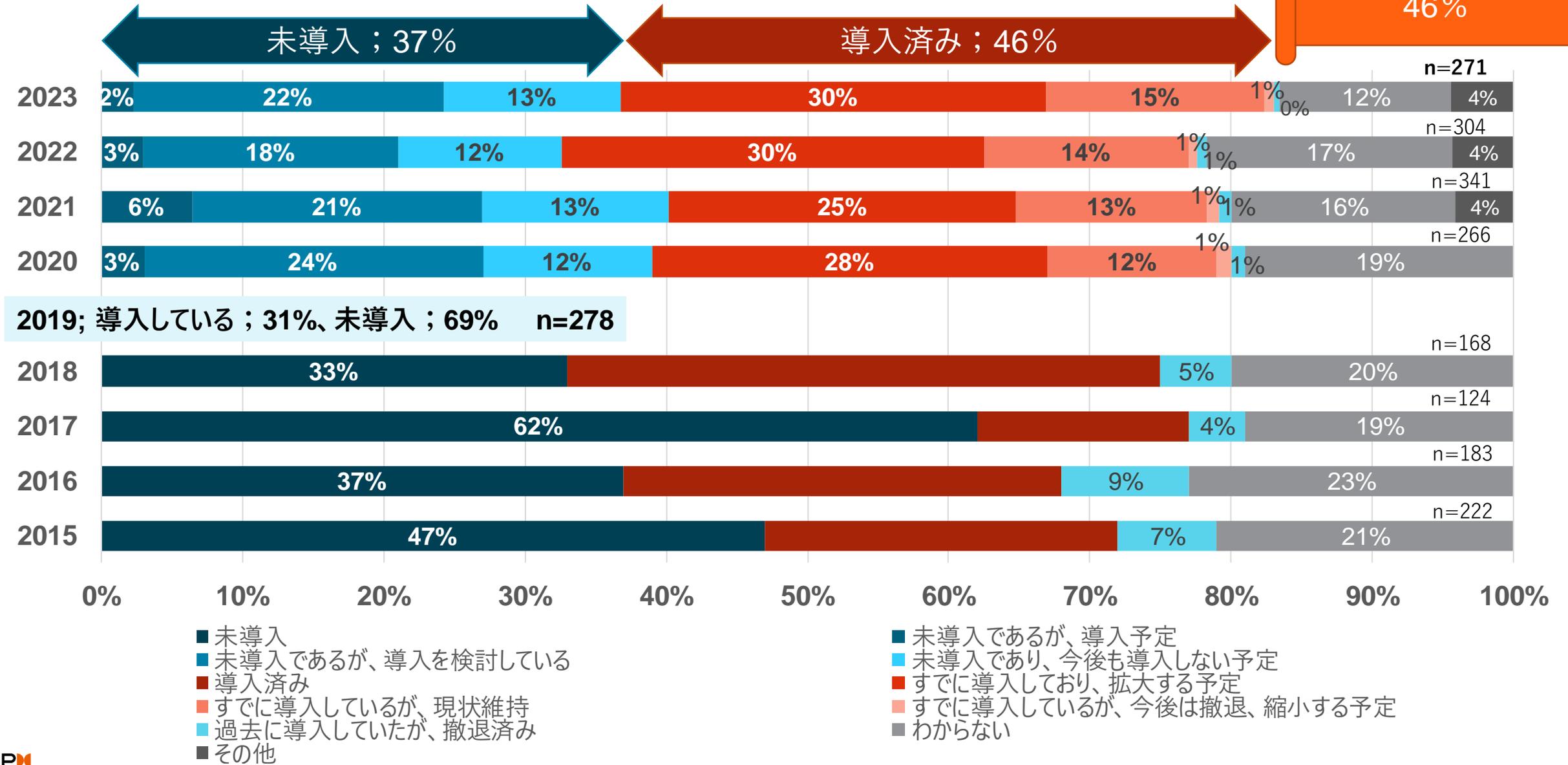
組織のプロジェクトへの取組み | プロジェクトの評価指標



n=271

組織のプロジェクトへの取組み | アジャイル導入方針

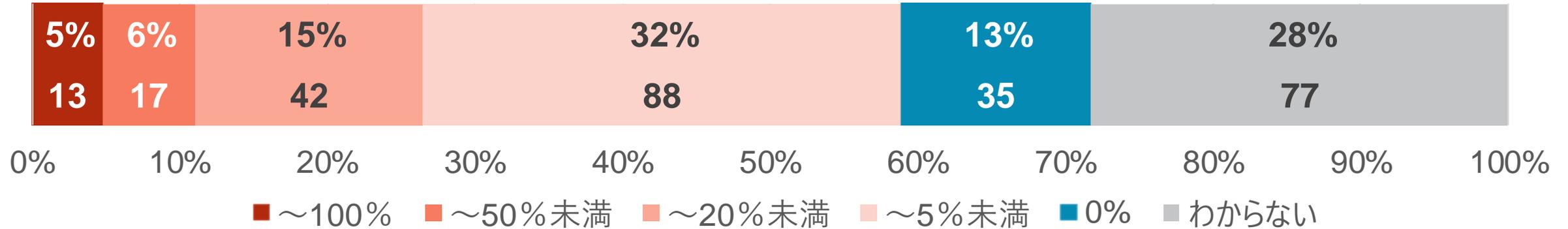
導入済みとの回答は
46%



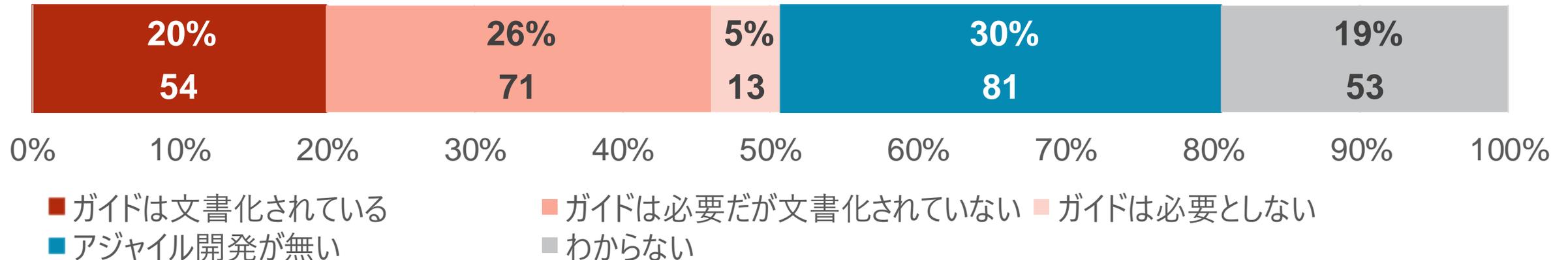
組織のプロジェクトへの取組み | アジャイル関連要員/アジャイル開発ガイド

アジャイル要員の比率は低い n=271

【アジャイル要員のおよその比率】

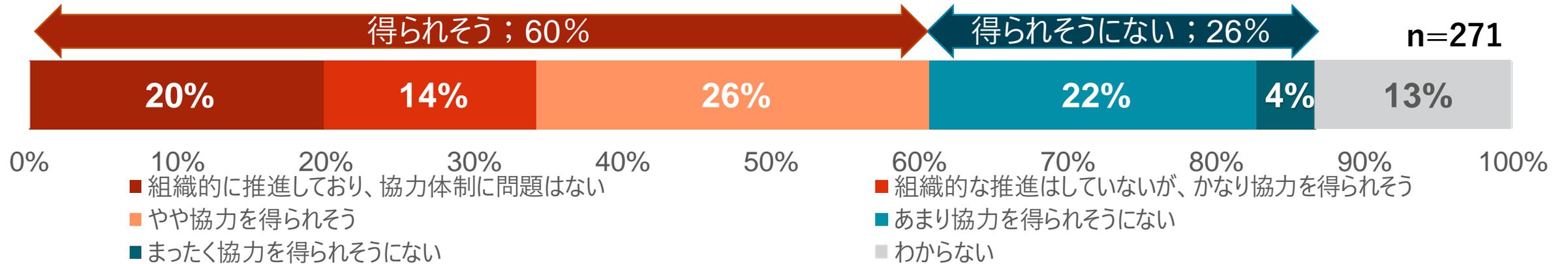


【アジャイル開発向けガイド】

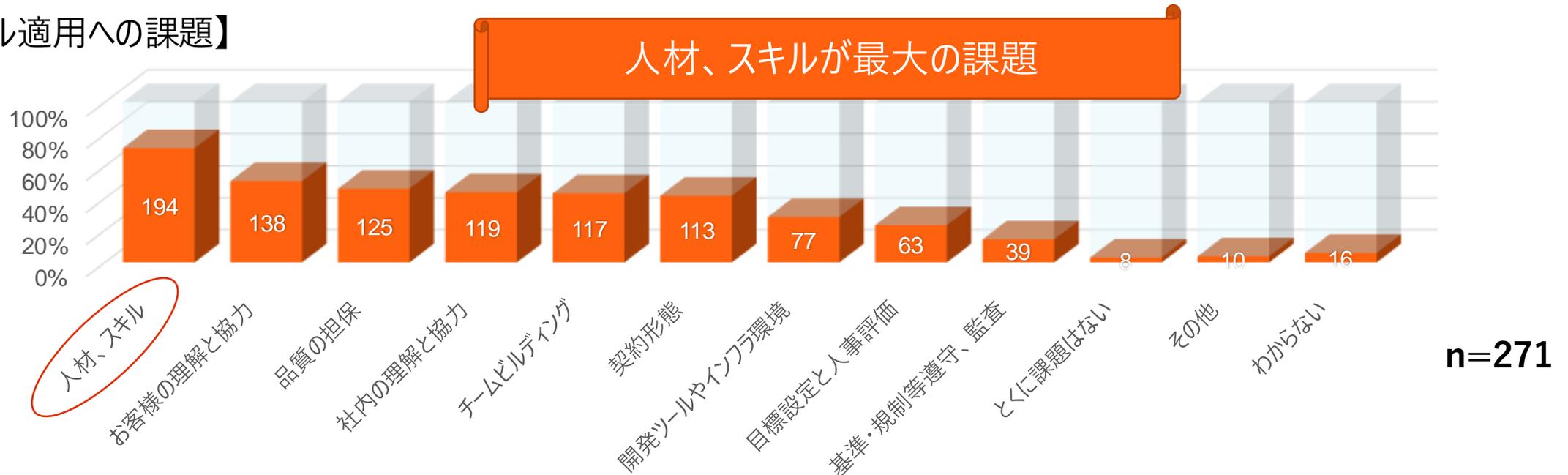


組織のプロジェクトへの取組み | アジャイル適用への周囲の協力/適用への課題

【アジャイル適用への周囲の協力】



【アジャイル適用への課題】



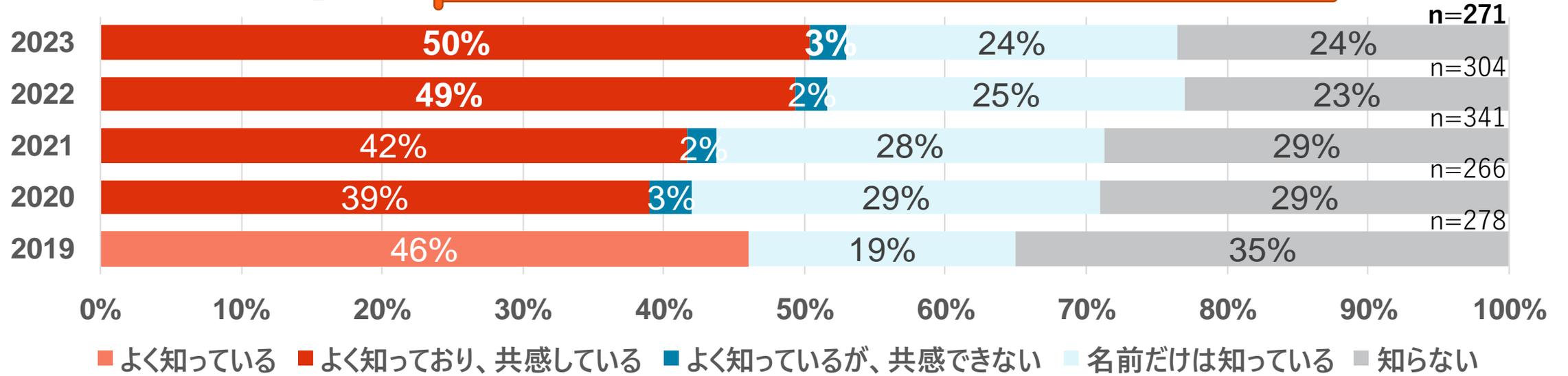
回答者のアジャイルへの

認識

回答者のアジャイルへの認識 | アジャイルマニフェスト

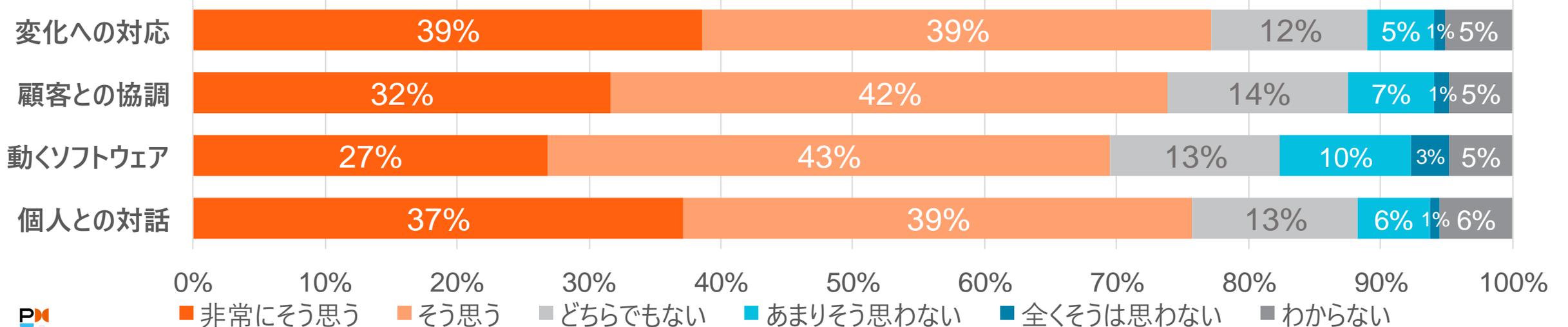
【アジャイルマニフェストの認識】

マニフェストをよく知っており、共感している人が半数

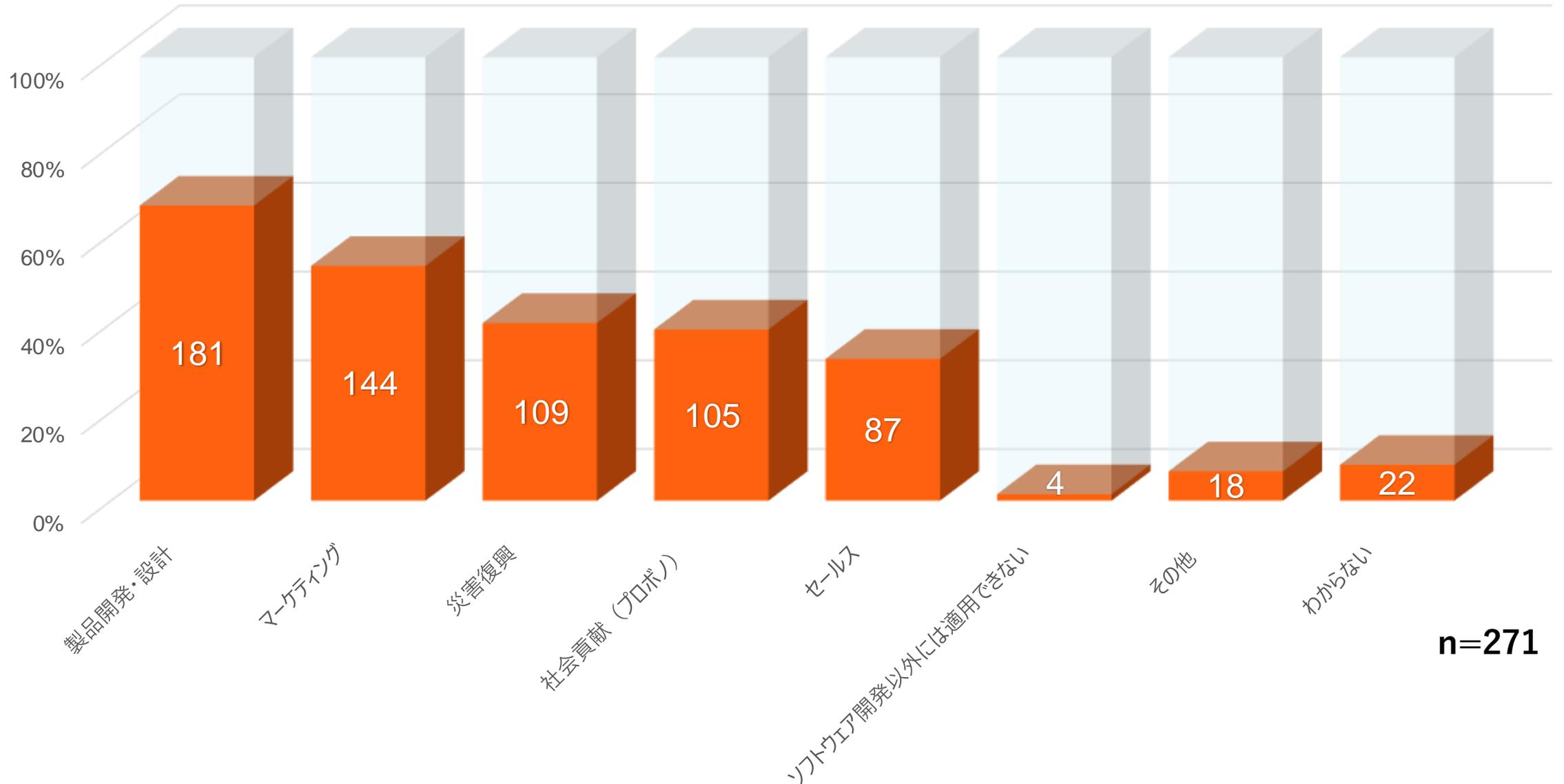


【アジャイルマニフェスト各項目への共感】

n=271



回答者のアジャイルへの認識 | ソフトウェア開発以外の適用可能領域

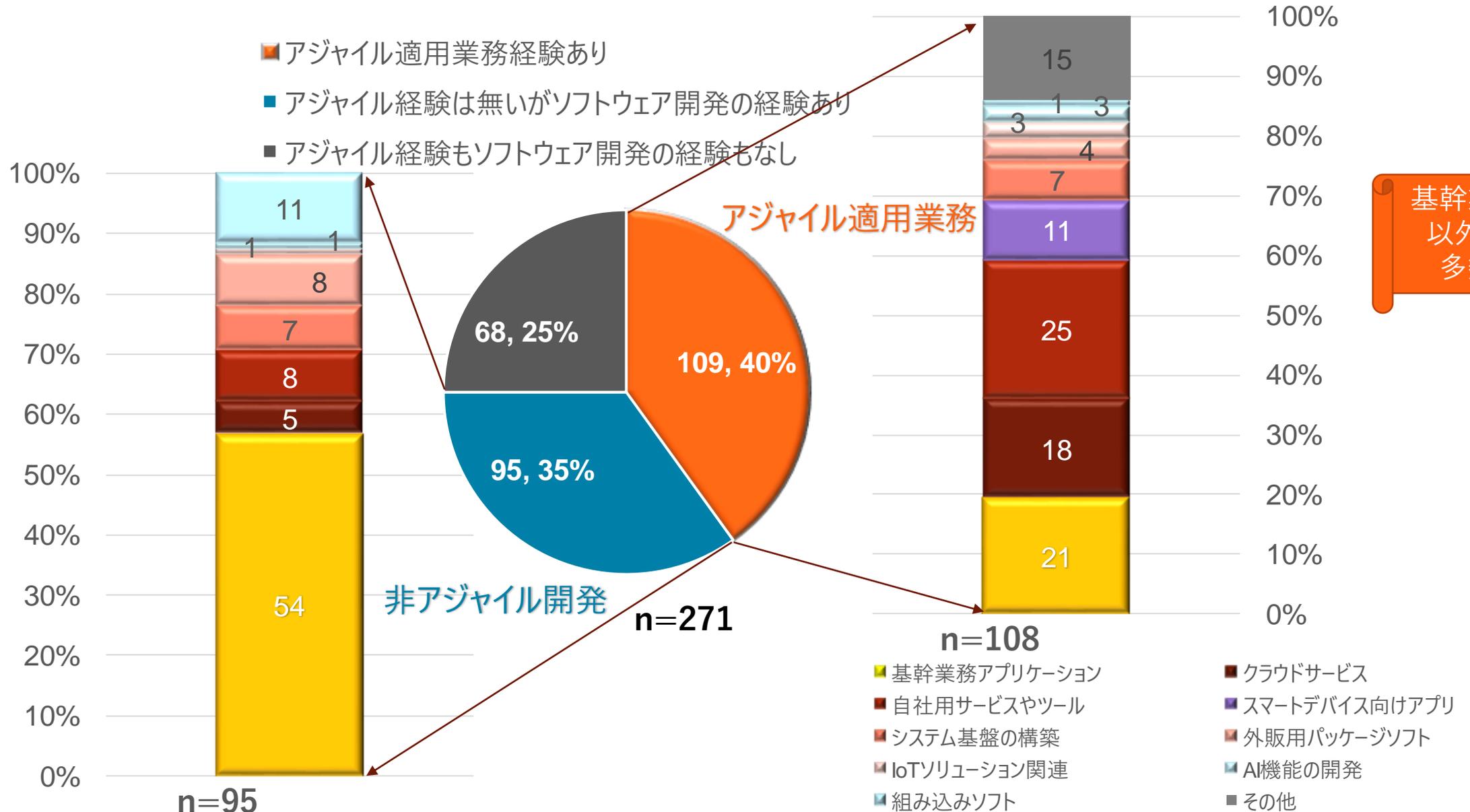


n=271

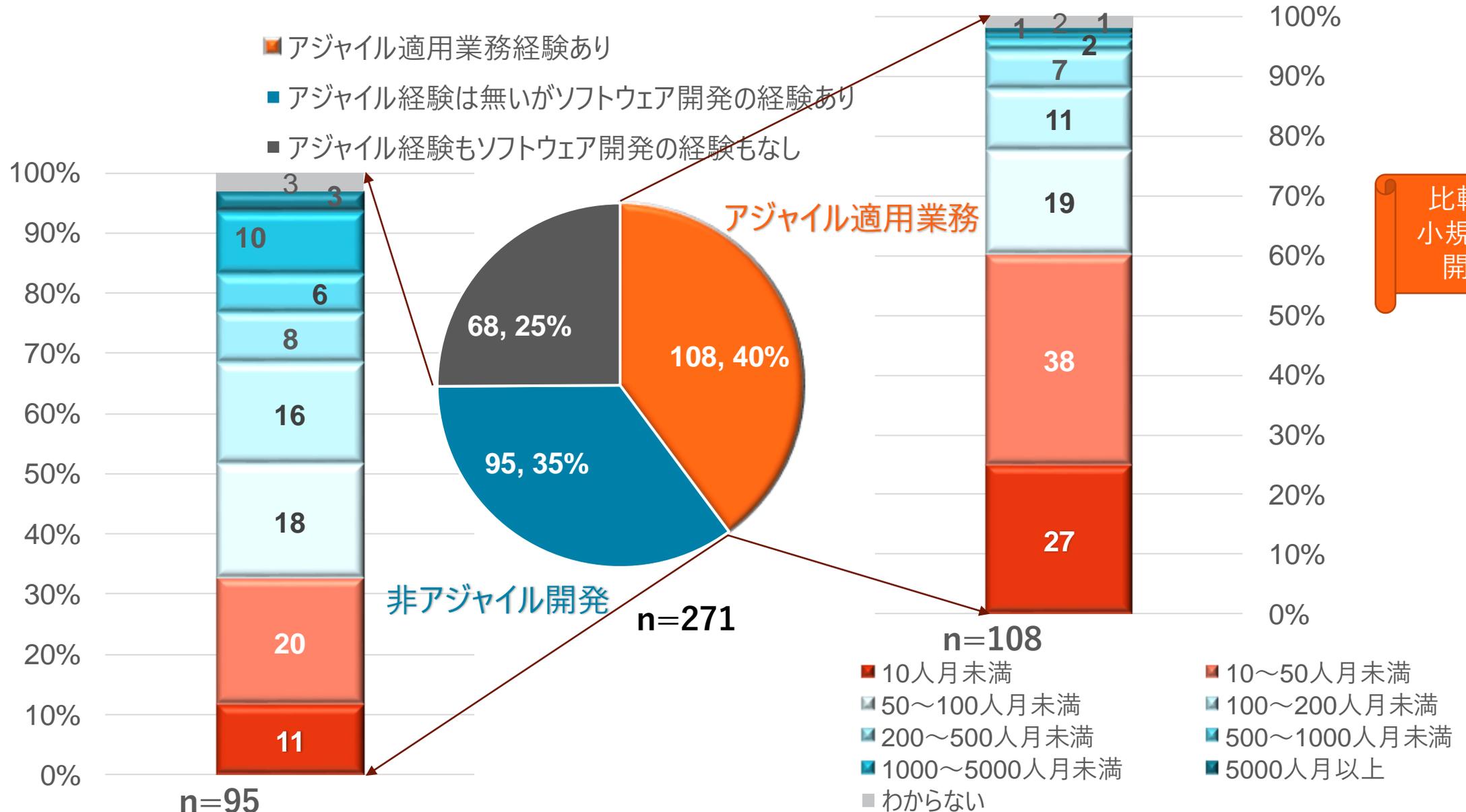
回答者の経験した

アジャイル適用業務と
非アジャイルソフトウェア開発

回答者の経験 | アジャイル適用業務 / 非アジャイル開発 (開発対象)

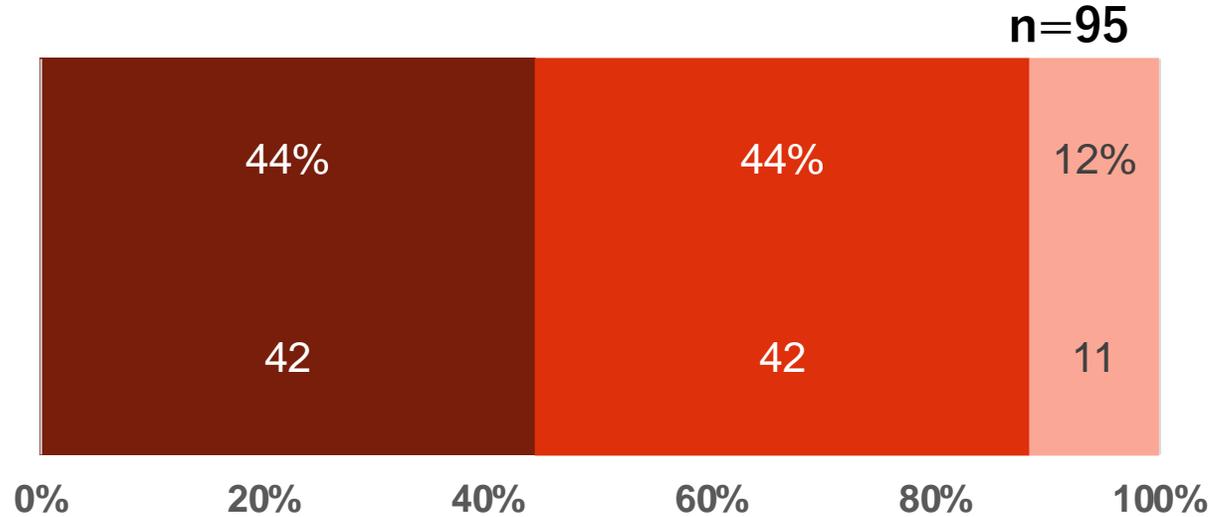


回答者の経験 | アジャイル適用業務 / 非アジャイル開発 (開発規模)



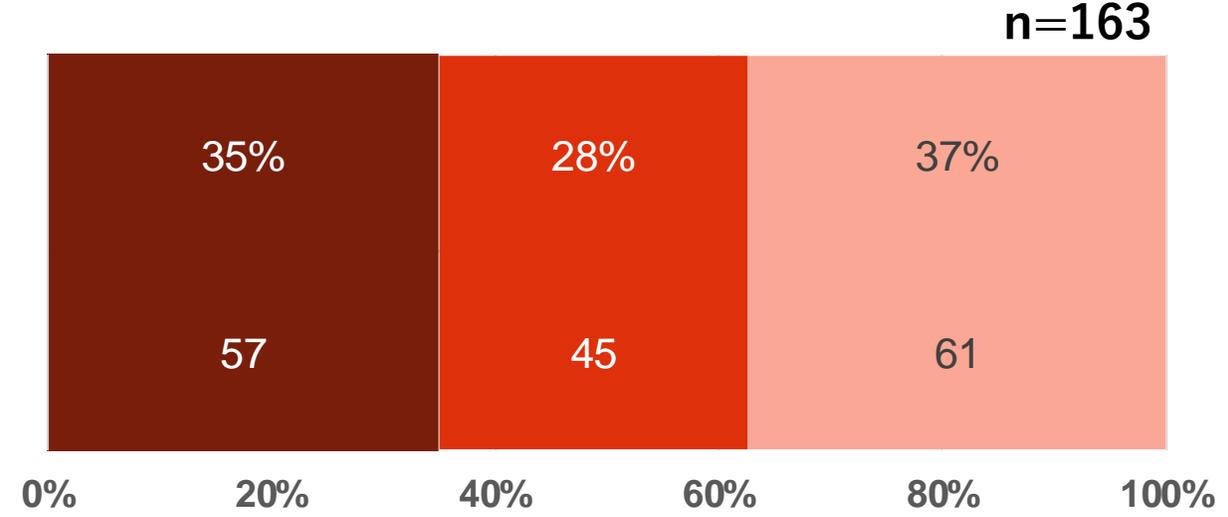
回答者の経験した非アジャイル開発のプロジェクトマネジメント・アプローチ

< 非アジャイル開発経験事例の内訳 >



- 当初見積りとスケジュールを厳守する管理
- 当初見積りとスケジュールにある程度の余裕を加えて、そのバッファを管理
- 当初見積りとスケジュールを状況に応じて柔軟に変更する管理

< 回答者全体の組織のアプローチのうち非アジャイルの内訳 >



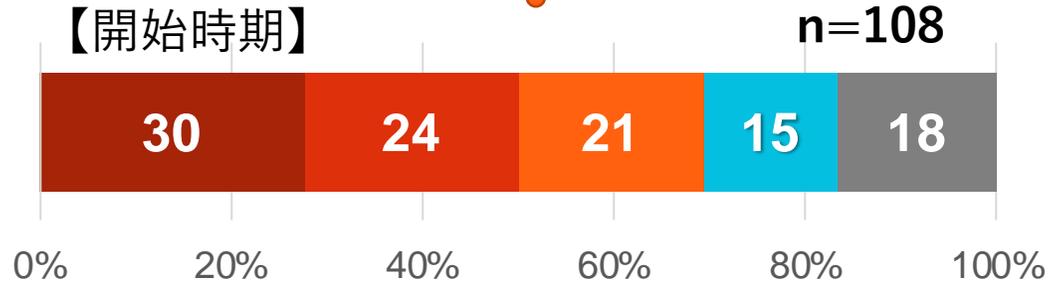
- ウォーターフォール型で、厳密なゲート管理を行っている
- ウォーターフォールに近いが、ある程度の手戻りを受け入れる余裕を見込んでいる
- ウォーターフォールに近いが、状況に応じて柔軟に変更することを前提にしている

回答者全体よりも保守的

回答者の経験したアジャイル適用業務 / 非アジャイル開発 | 開始時期と期間

<アジャイル適用業務>

比較的新しいケースが多い

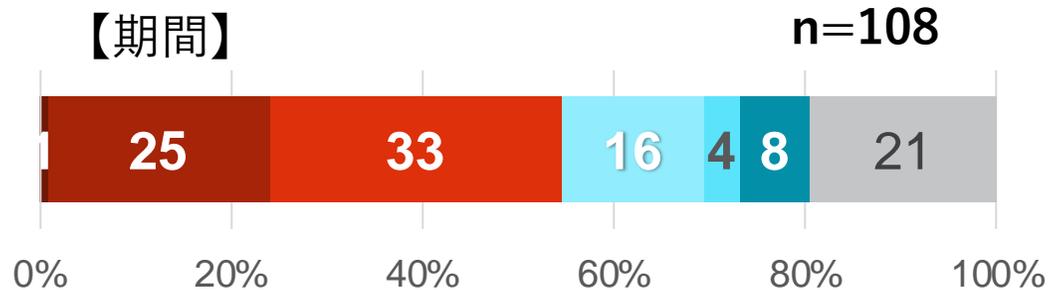


- 1年未満前
- 1年以上から2年未満前
- 2年以上から3年未満前
- 3年以上から5年未満前
- 5年以上前

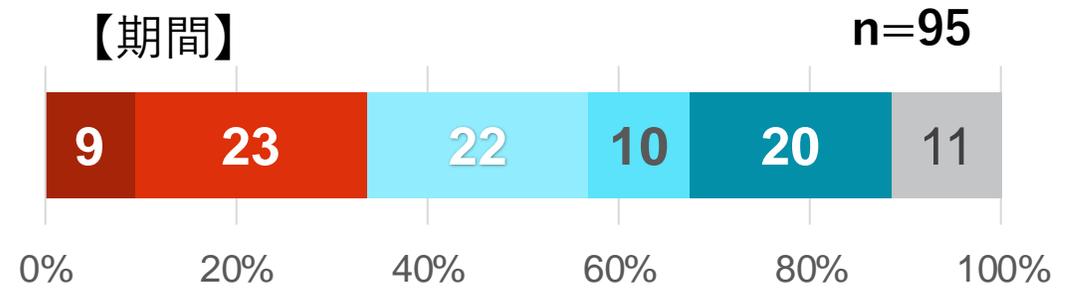
<非アジャイル開発>



- 1年未満前
- 1年以上から2年未満前
- 2年以上から3年未満前
- 3年以上から5年未満前
- 5年以上前



- 2か月未満
- 2か月以上から6か月未満
- 6か月以上から12か月未満
- 12か月以上から18か月未満
- 18か月以上から24か月未満
- 24か月以上
- 今も継続中



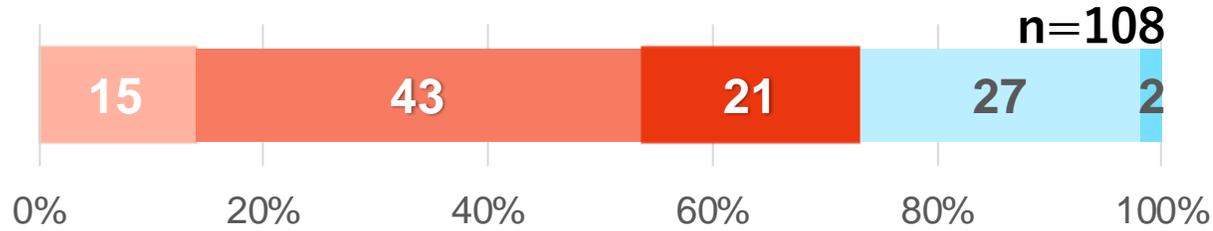
- 2か月未満
- 2か月以上から6か月未満
- 6か月以上から12か月未満
- 12か月以上から18か月未満
- 18か月以上から24か月未満
- 24か月以上
- 今も継続中

比較的短期間

回答者の経験したアジャイル適用業務 / 非アジャイル開発 人数、チーム数

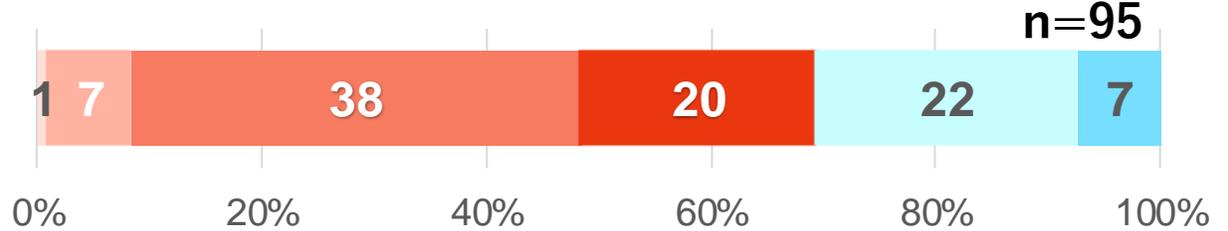
<アジャイル適用業務>

【チームの人数】



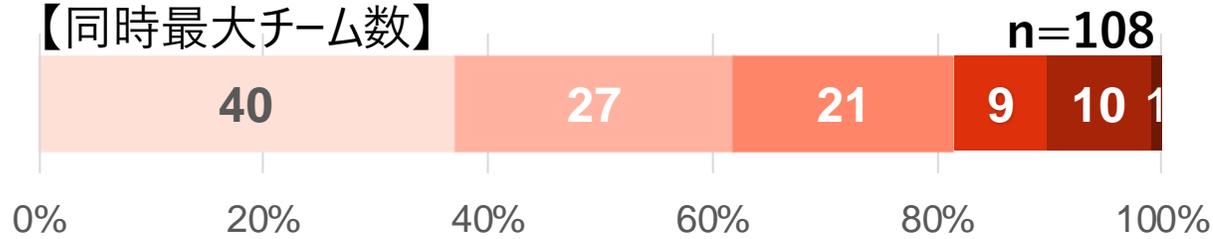
<非アジャイル開発>

【チームの人数】

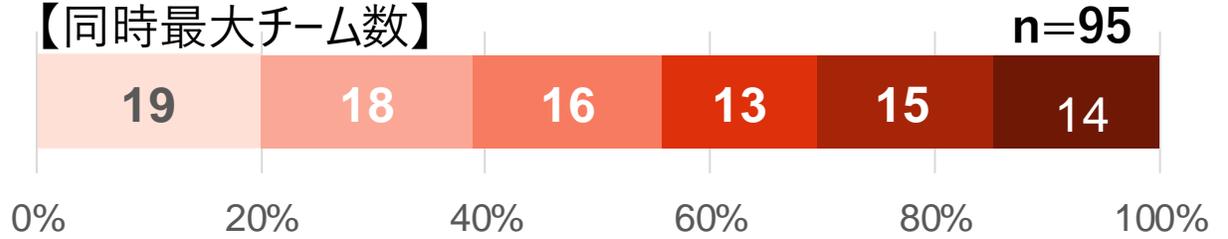


■ 1名 ■ 2から3名 ■ 4から6名 ■ 7から9名 ■ 10から20名 ■ 21名以上 ■ 1名 ■ 2から3名 ■ 4から6名 ■ 7から9名 ■ 10から20名 ■ 21名以上

【同時最大チーム数】



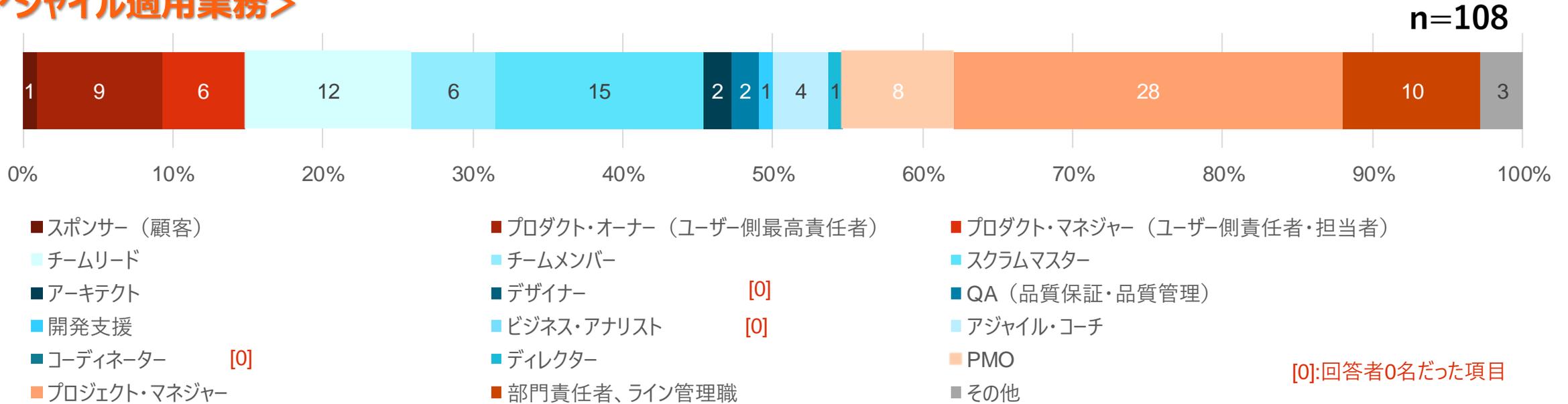
【同時最大チーム数】



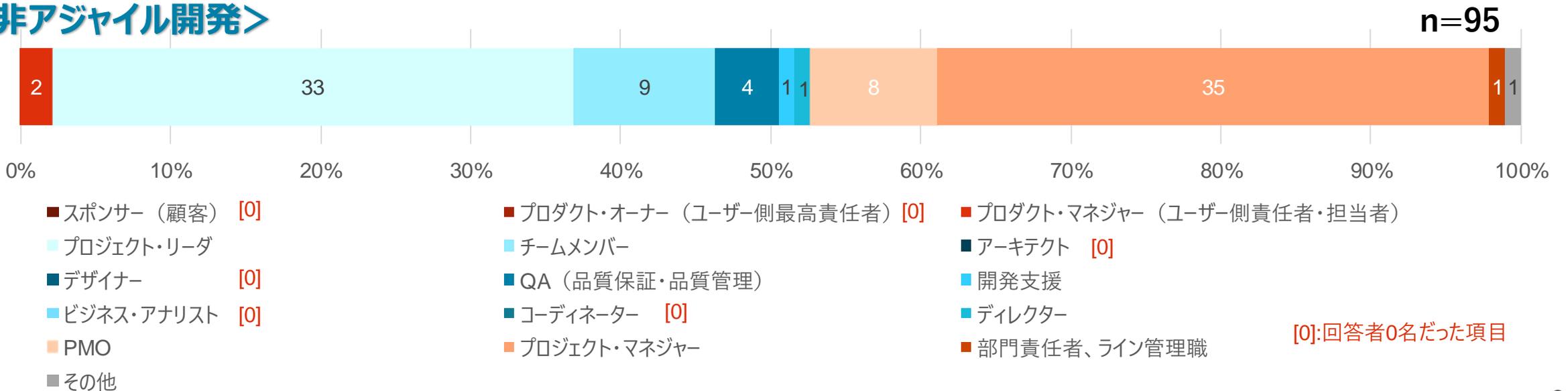
■ 1チーム ■ 2チーム ■ 3チーム ■ 4チーム ■ 5チーム ■ 6チーム以上 ■ 1チーム ■ 2チーム ■ 3チーム ■ 4チーム ■ 5チーム ■ 6チーム以上

回答者の経験したアジャイル適用業務 / 非アジャイル開発 | 回答者のロール

<アジャイル適用業務>

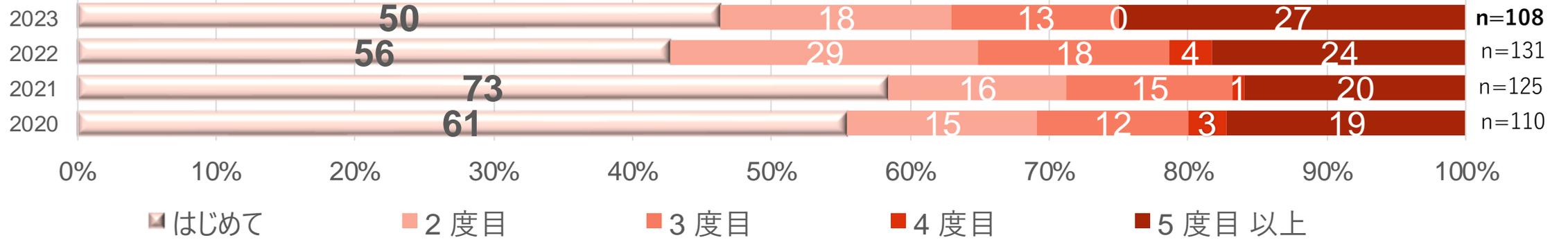


<非アジャイル開発>

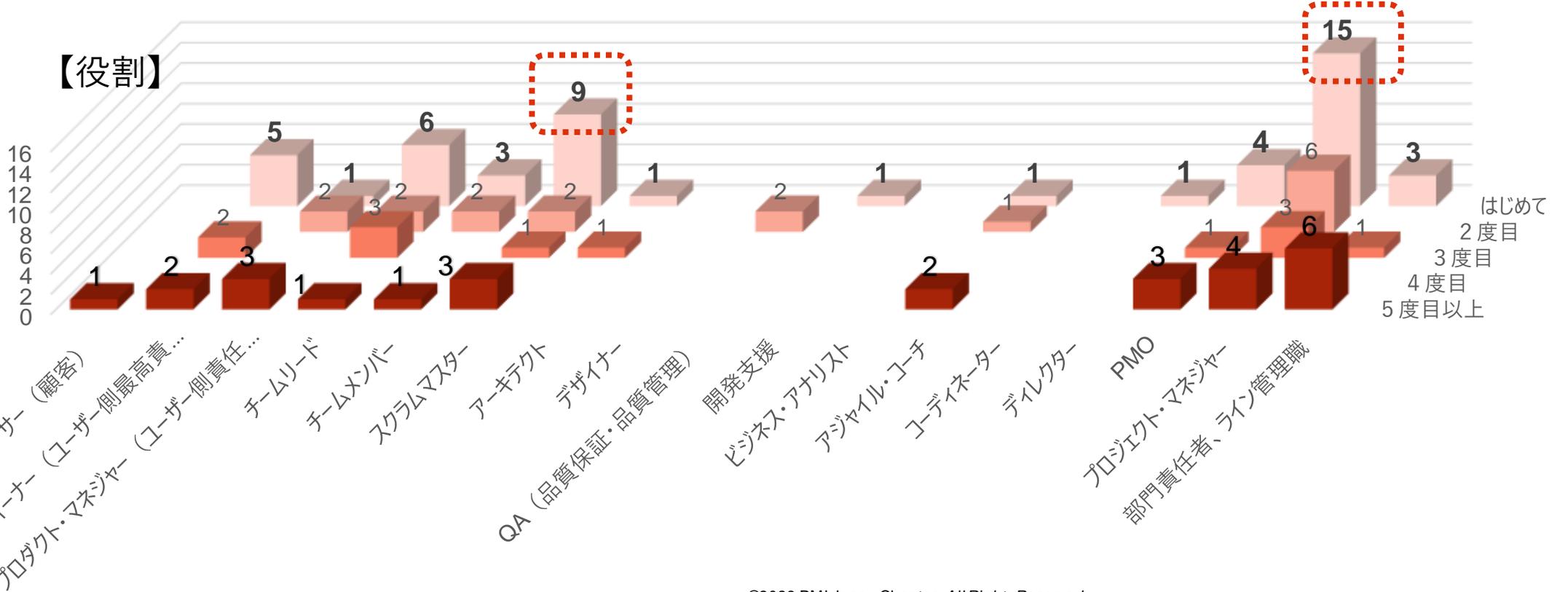


回答者の経験したアジャイル適用業務 | 経験回数と役割

【経験回数】



【役割】



n=108

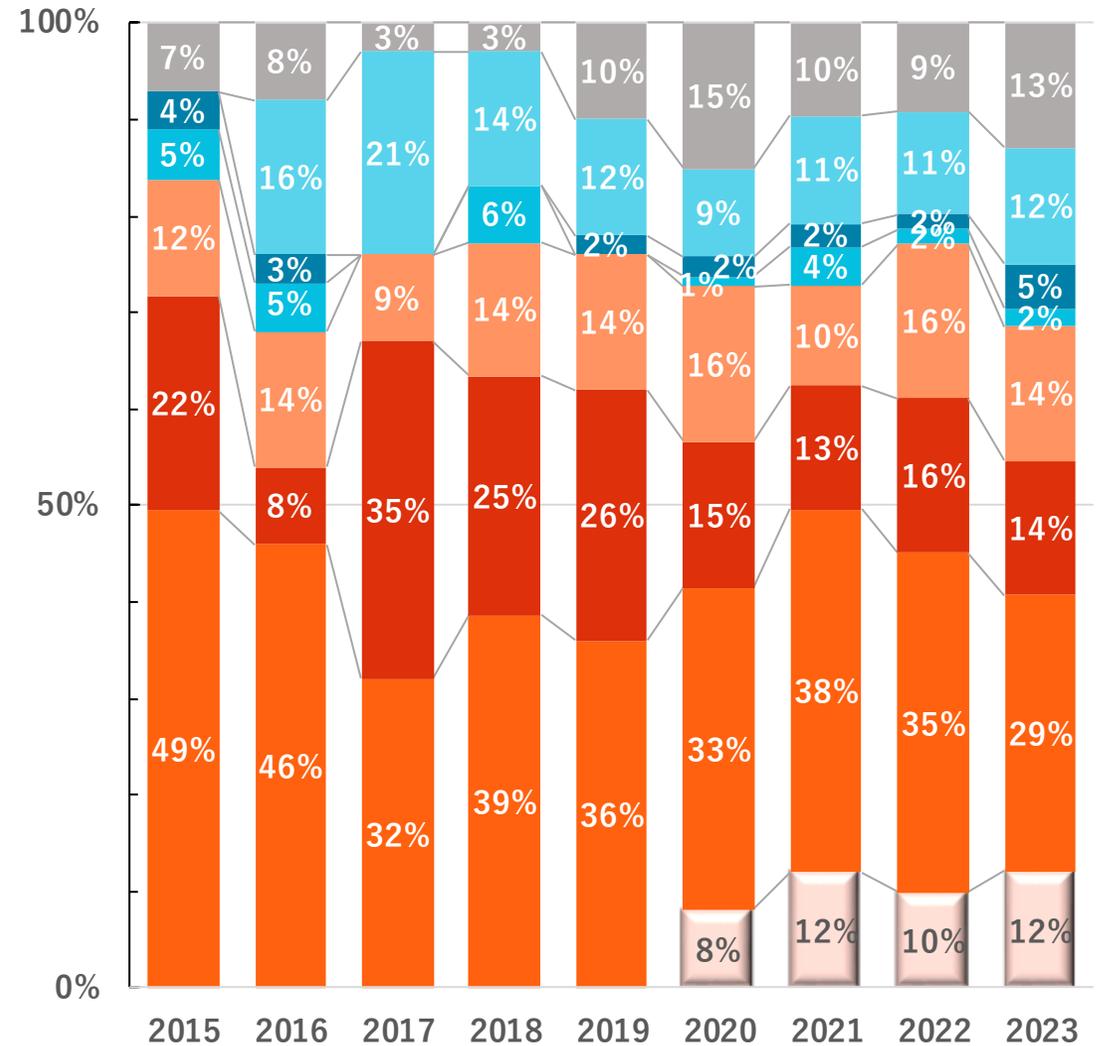


回答者の経験したアジャイル適用業務 | アジャイル採用理由

n=108

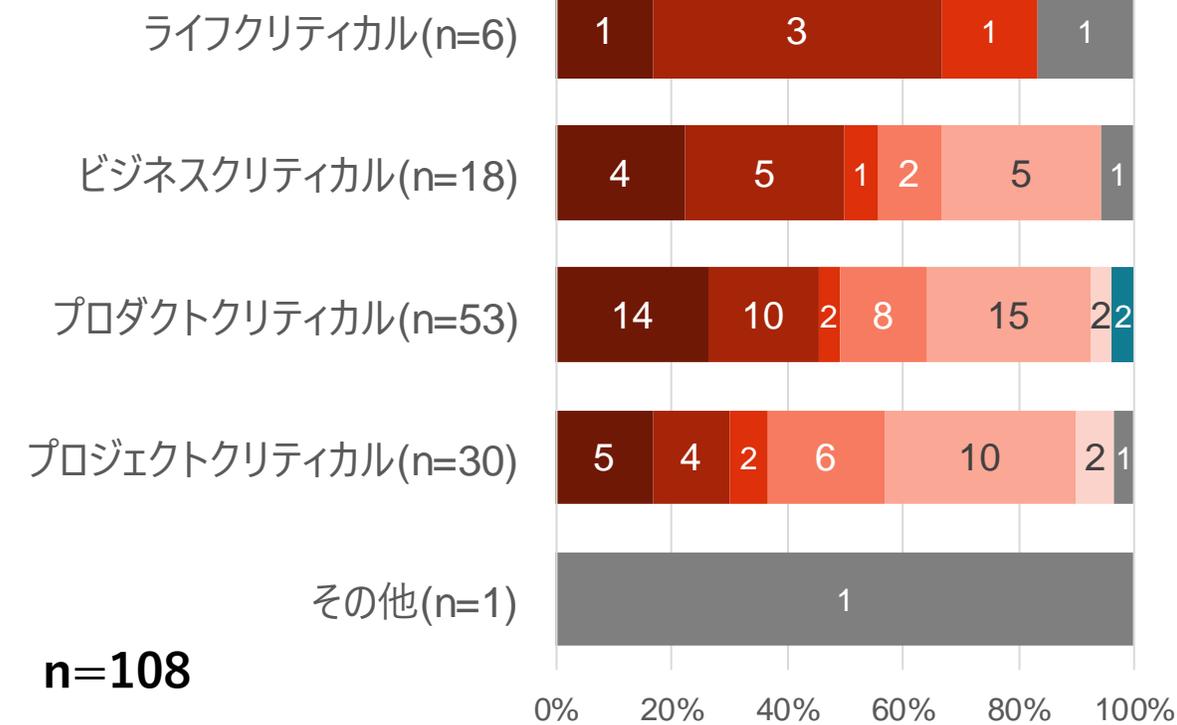


- 組織として適用することが規定されている
- チームが自主的に判断し導入
- 経営層や上司からのトップダウンでの指示
- スポンサーや発注先からの指示
- セミナーなどで興味をもったから
- 競合他社との兼ね合い
- 自分がアジャイルを推進する立場で周囲を説得した
- その他

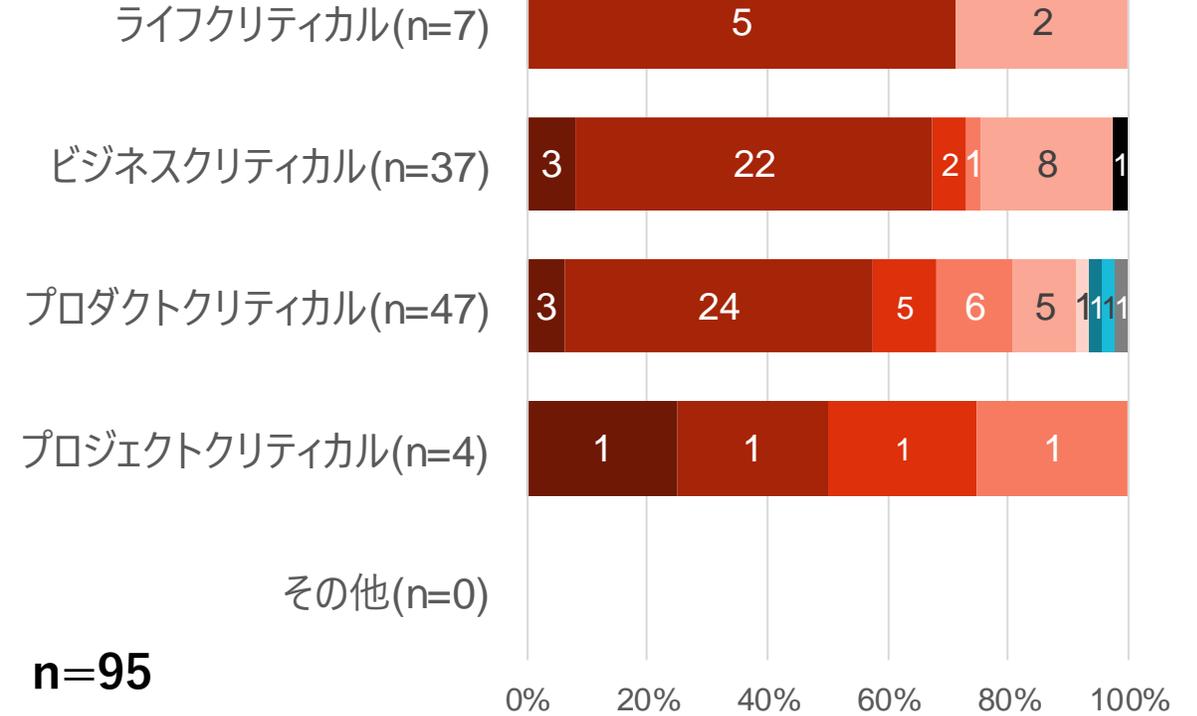


回答者の経験したアジャイル適用業務／非アジャイル開発 プロダクトの性質と課題

<アジャイル適用業務>



<非アジャイル開発>



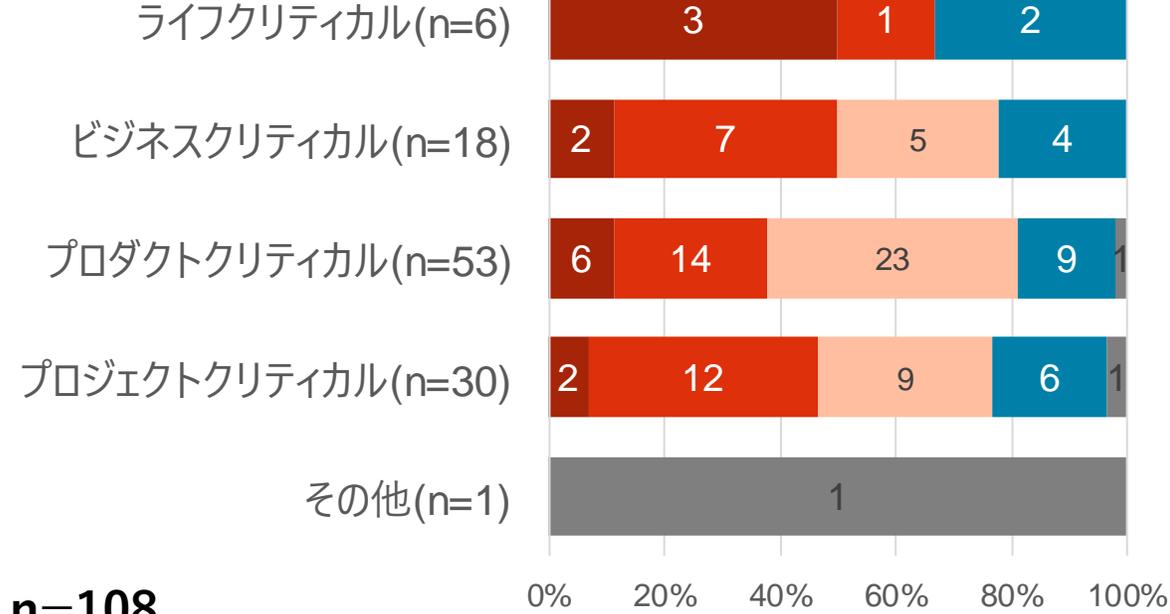
- 満足度
- 品質
- コスト
- スケジュール
- 円滑な業務の実現
- プロジェクトの透明性
- セキュリティ
- セーフティ
- その他 (具体的に)
- わからない

- 満足度
- 品質
- コスト
- スケジュール
- 円滑な業務の実現
- プロジェクトの透明性
- セキュリティ
- セーフティ
- その他 (具体的に)
- わからない

ライフクリティカル (非常に高いリスク：人々の生命や健康に悪影響を及ぼす可能性がある)
 ビジネスクリティカル (高いリスク：組織の財政状態やイメージに悪影響を与える可能性がある)
 プロダクトクリティカル (中程度のリスク：プロダクトまたはサービス全体に悪影響を与える可能性がある)
 プロジェクトクリティカル (低いリスク：チーム自体への投資の損失に限定される)

回答者の経験したアジャイル適用業務／非アジャイル開発 プロダクトの性質と品質管理体制

<アジャイル適用業務>

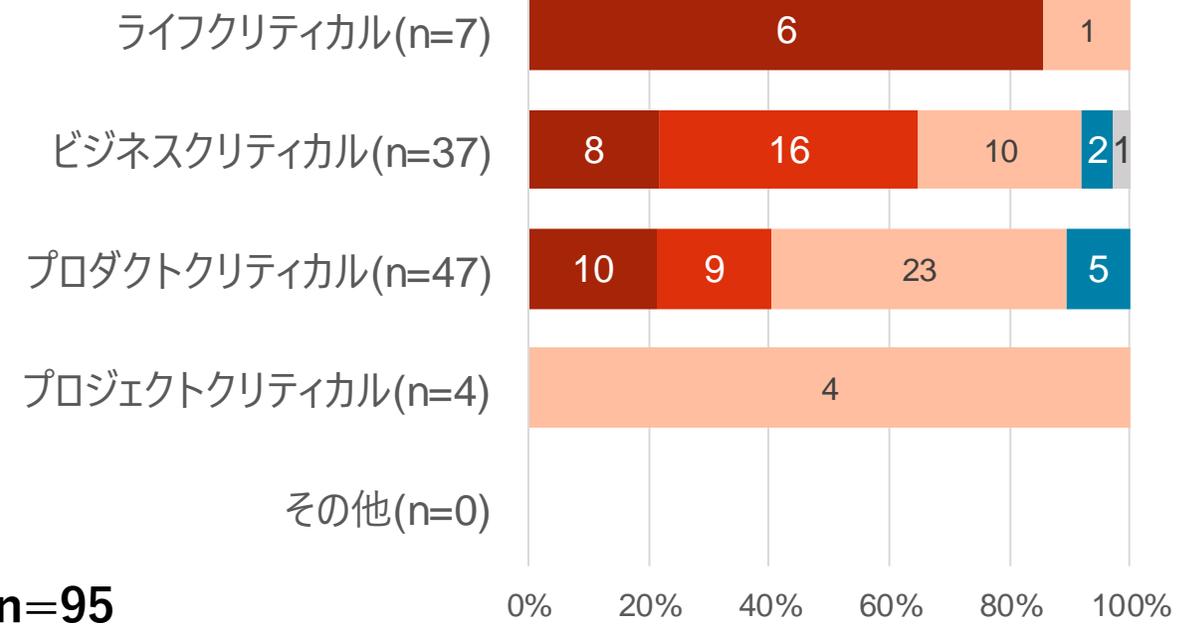


n=108

- 独立した品質保証/テストチームが横断で管理している
- 品質管理/テスト担当をチーム内に設置している
- 専担者はいないが必要に応じてに検査/監査している
- 組織的な品質保証/テスト活動はない
- その他
- わからない

ライフクリティカル（非常に高いリスク：人々の生命や健康に悪影響を及ぼす可能性がある）
 ビジネスクリティカル（高いリスク：組織の財政状態やイメージに悪影響を与える可能性がある）
 プロダクトクリティカル（中程度のリスク：プロダクトまたはサービス全体に悪影響を与える可能性がある）
 プロジェクトクリティカル（低いリスク：チーム自体への投資の損失に限定される）

<非アジャイル開発>



n=95

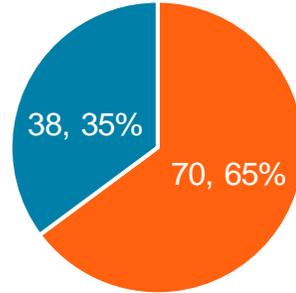
- 独立した品質保証/テストチームが横断で管理している
- 品質管理/テスト担当をチーム内に設置している
- 専担者はいないが必要に応じてに検査/監査している
- 組織的な品質保証/テスト活動はない
- その他
- わからない



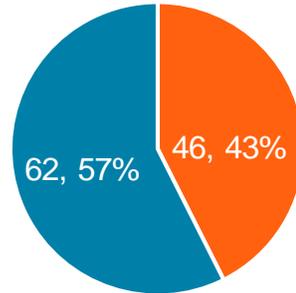
回答者の経験したアジャイル適用業務／非アジャイル開発 完了の定義（Doneの定義）

<アジャイル適用業務>

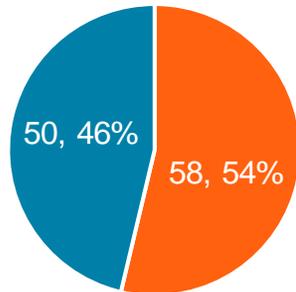
要求仕様/バックログ
アイテム/ユーザー
ストーリーの「完了」



フェーズ/スプリント/
イテレーションの「完了」



受け入れ/リリースの
「完了」



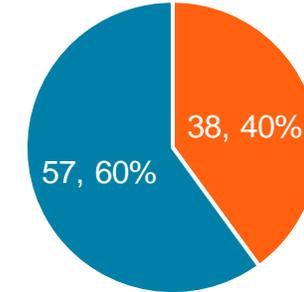
n=108



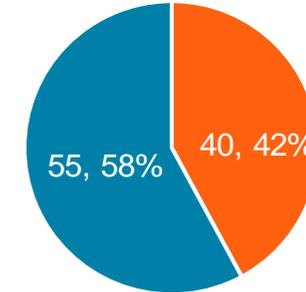
■ 定義している ■ 定義していない

<非アジャイル開発>

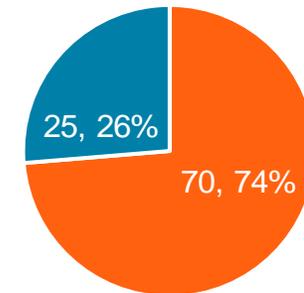
要求仕様/バックログ
アイテム/ユーザー
ストーリーの「完了」



フェーズ/スプリント/
イテレーションの「完了」



受け入れ/リリースの
「完了」



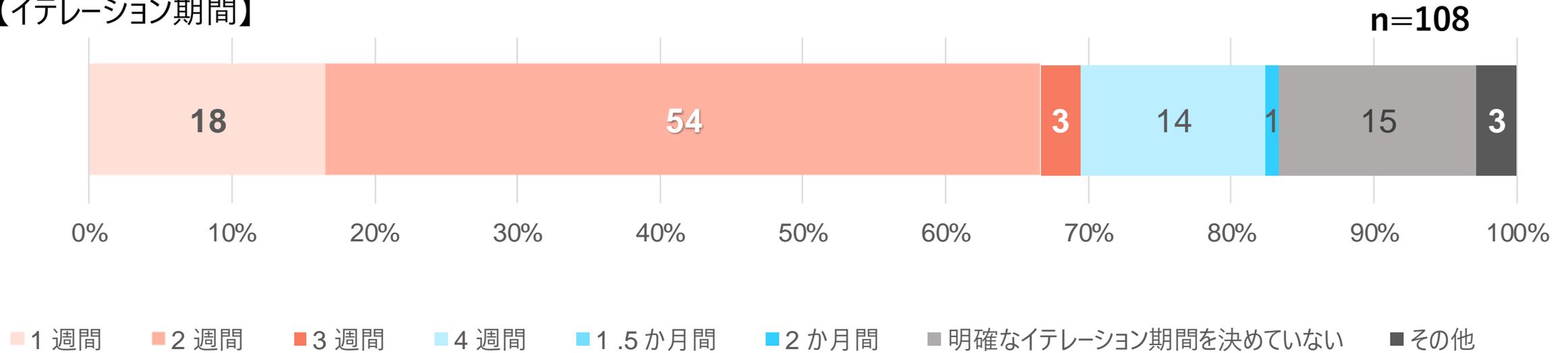
n=95

■ 定義している ■ 定義していない

回答者の経験したアジャイル適用業務 イテレーション期間

<アジャイル適用業務>

【イテレーション期間】

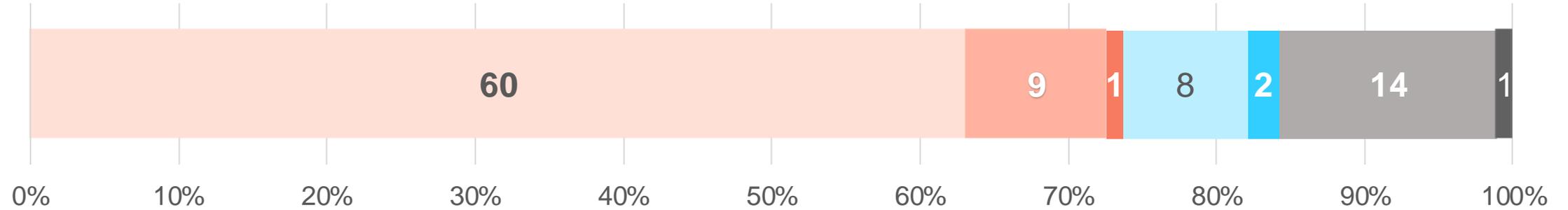


回答者の経験した 非アジャイル開発 品質確認サイクル

<非アジャイル開発>

【品質確認サイクル】

n=95

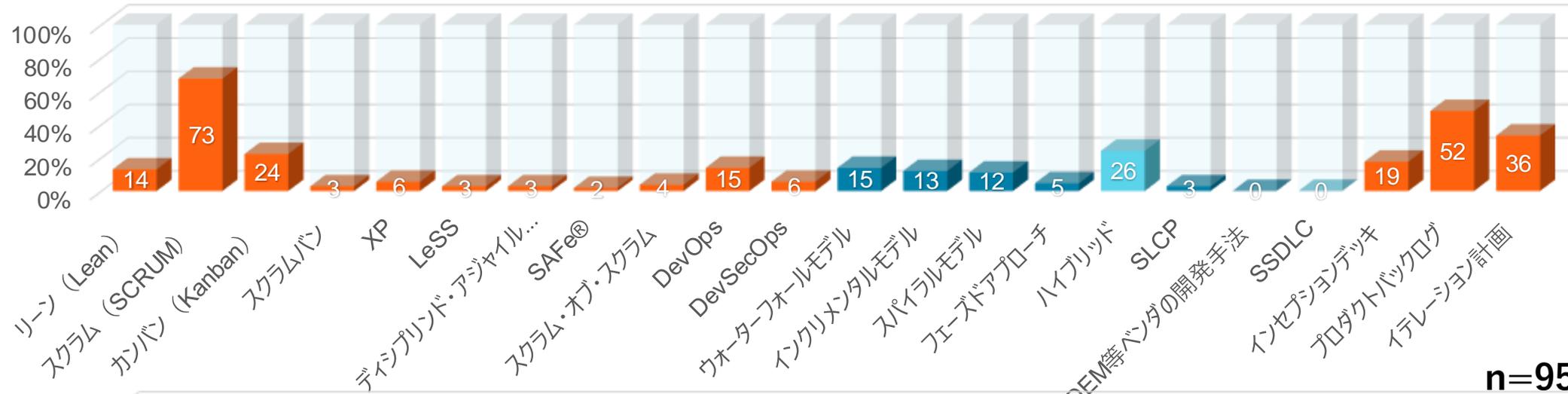


■ 1 週間 ■ 2 週間 ■ 3 週間 ■ 4 週間 ■ 1.5 か月間 ■ 2 か月間 ■ 明確なチェックサイクルを決めていない ■ その他

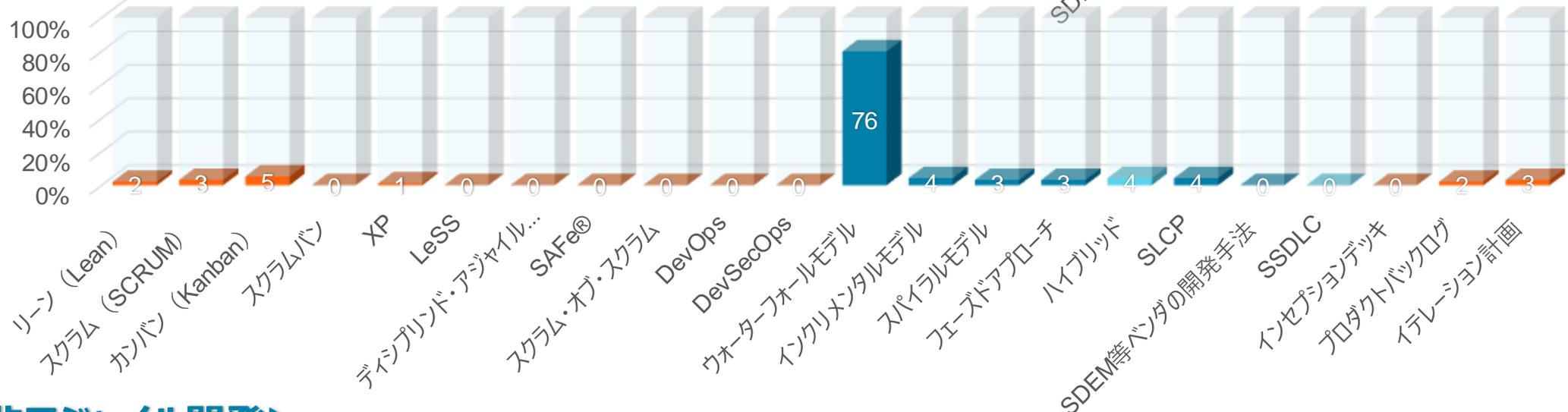
回答者の経験したアジャイル適用業務 / 非アジャイル開発 導入されたプロセス・手法、プラクティス 1/2

<アジャイル適用業務>

n=108



n=95



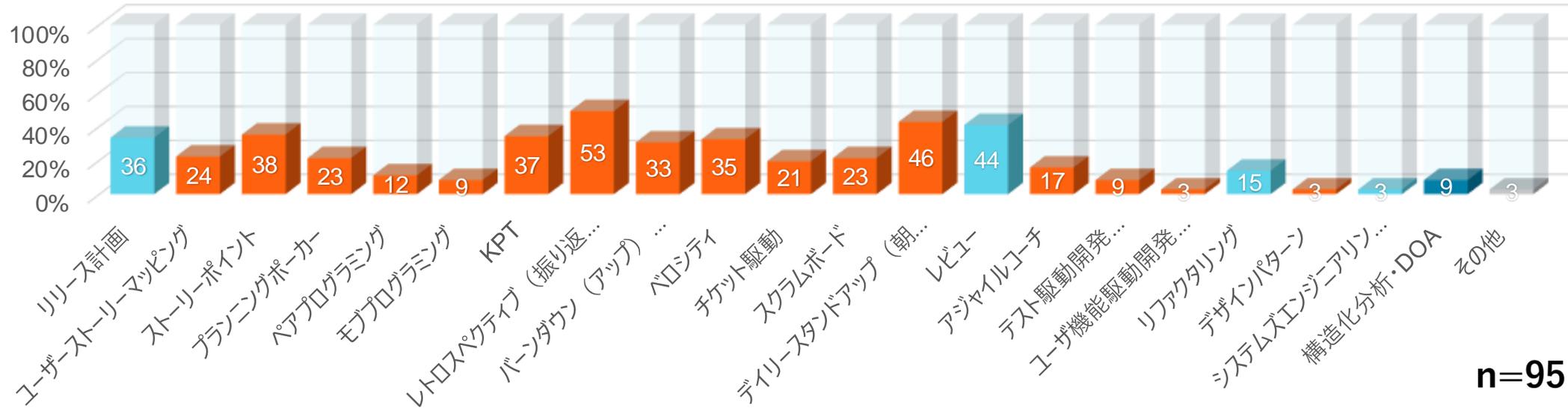
<非アジャイル開発>



回答者の経験したアジャイル適用業務 / 非アジャイル開発 導入されたプロセス・手法、プラクティス 2/2

<アジャイル適用業務>

n=108



n=95

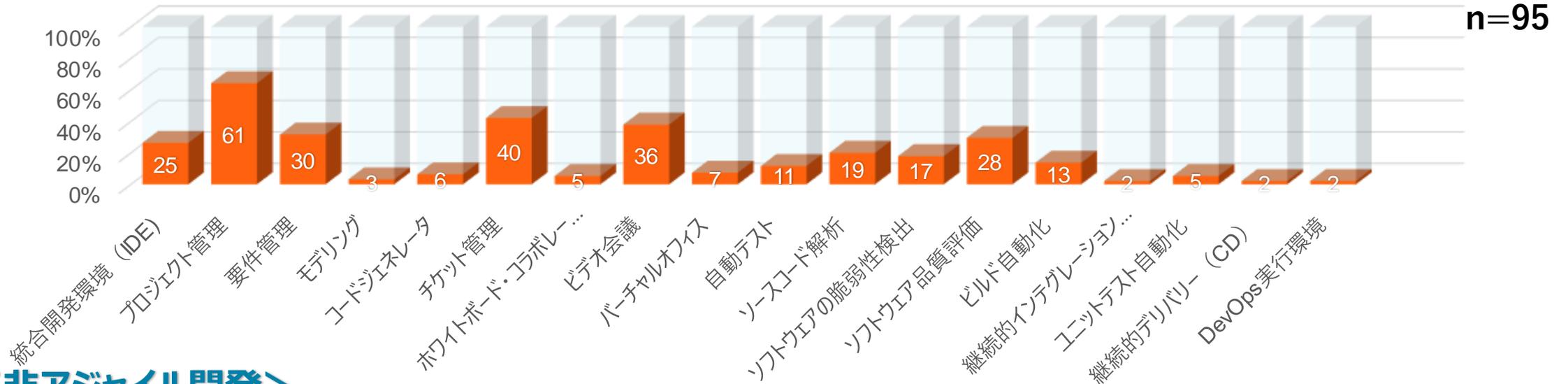
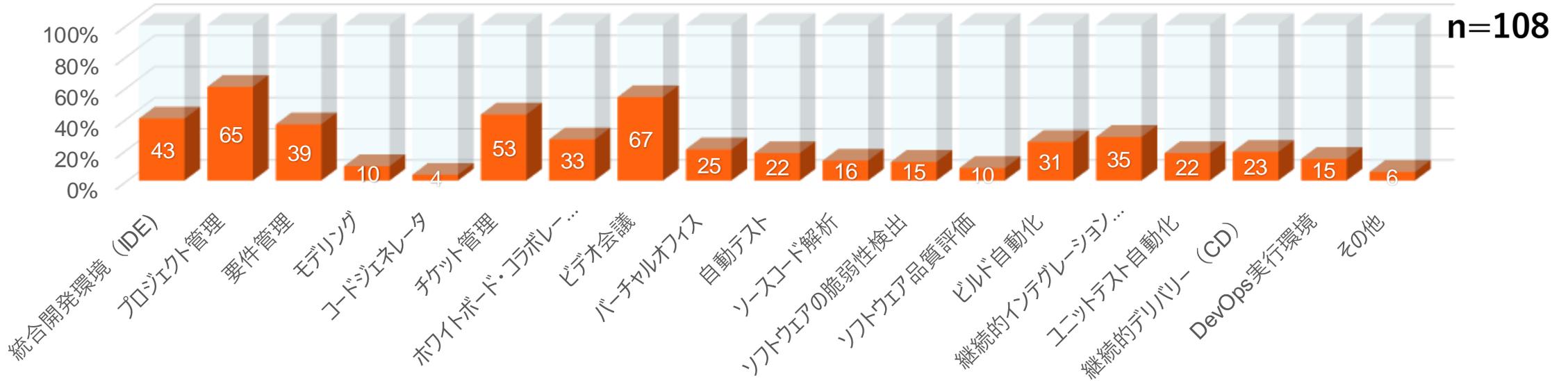


<非アジャイル開発>



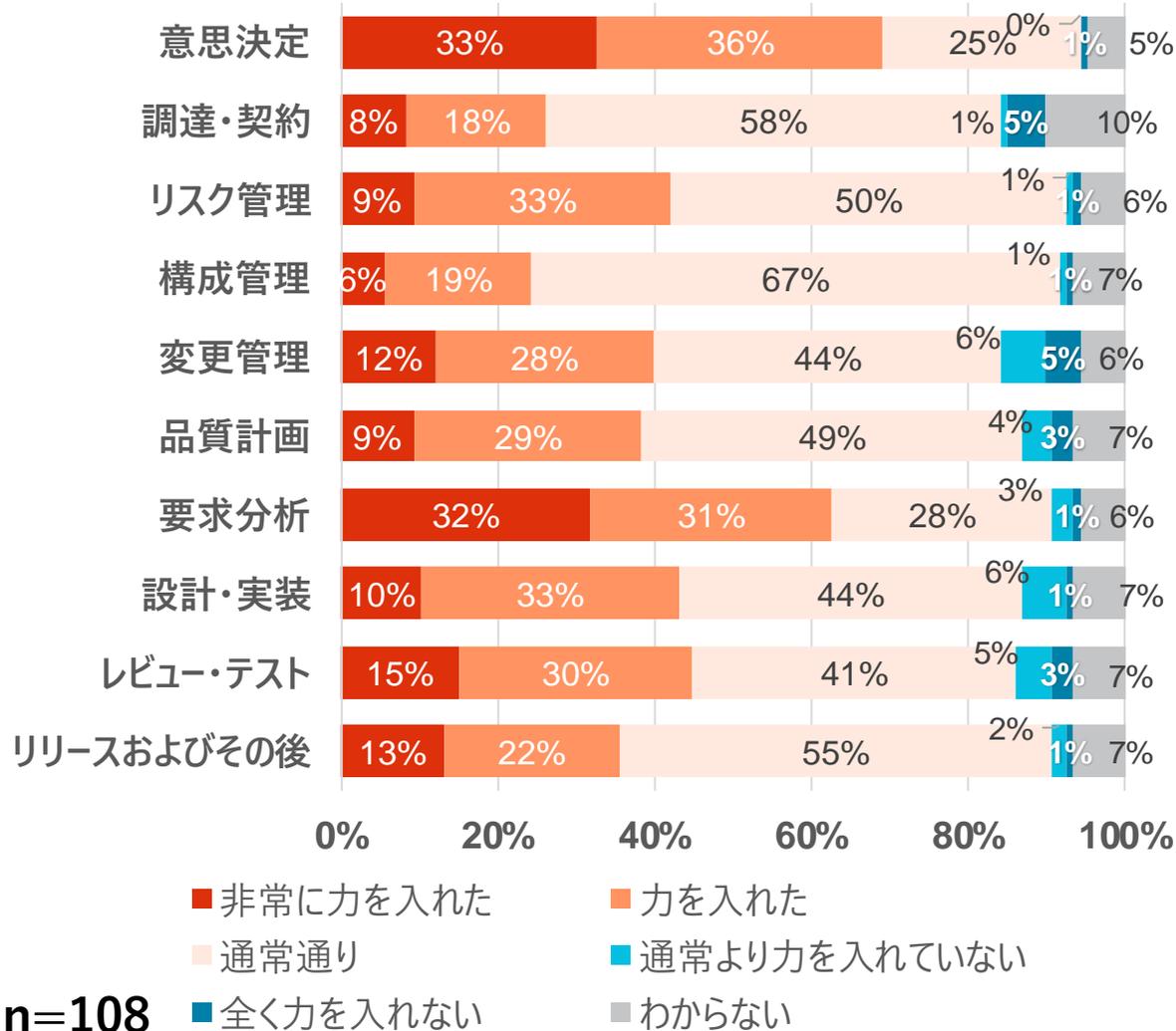
回答者の経験したアジャイル適用業務／非アジャイル開発ツールの利用

<アジャイル適用業務>



回答者の経験したアジャイル適用業務 / 非アジャイル開発 マネジメントの重点

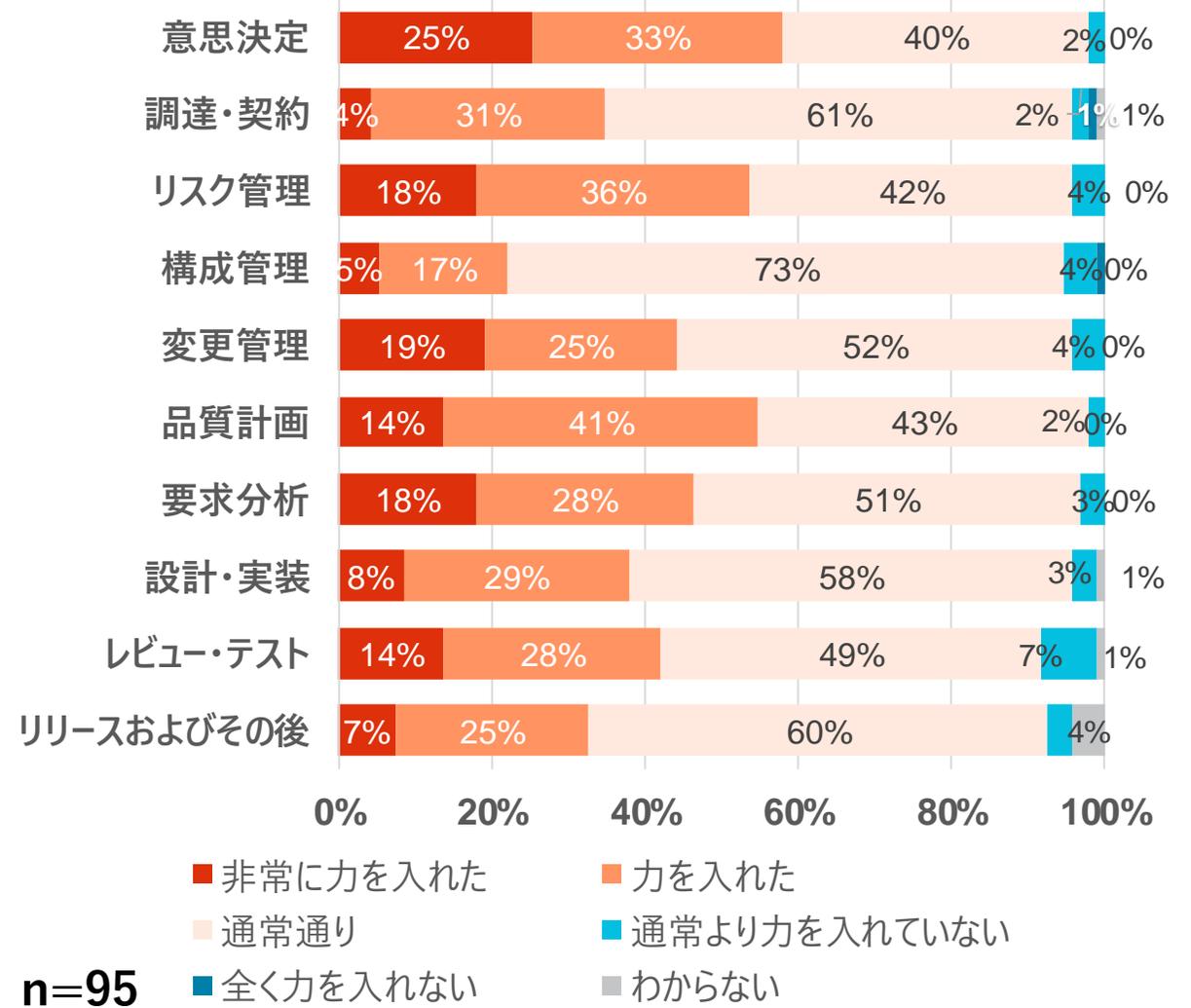
<アジャイル適用業務>



n=108



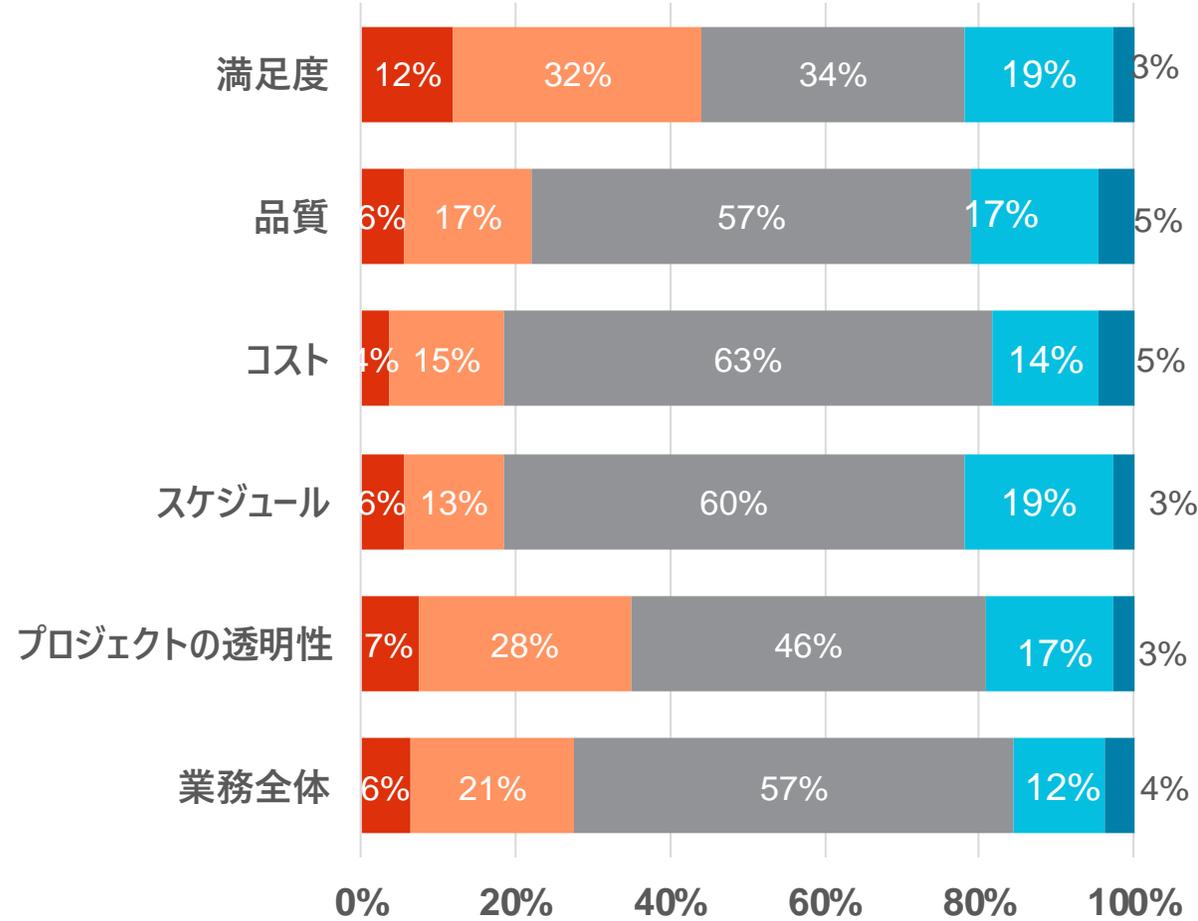
<非アジャイル開発>



n=95

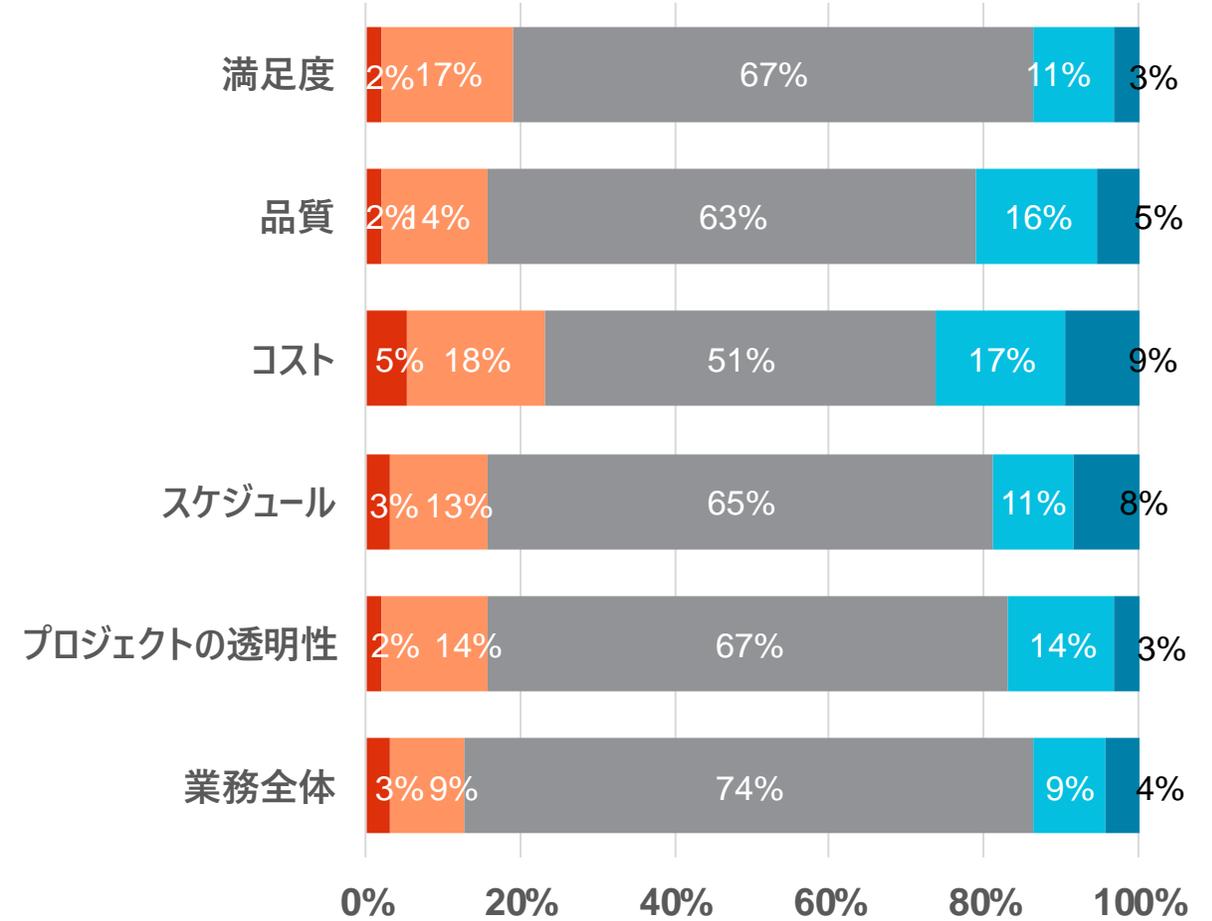
回答者の経験したアジャイル適用業務／非アジャイル開発 成果達成度（当初予定比）

<アジャイル適用業務>



■ 大幅に上回る ■ やや上回る ■ 予定通り ■ やや下回る ■ 大幅に下回る
n=108

<非アジャイル開発>



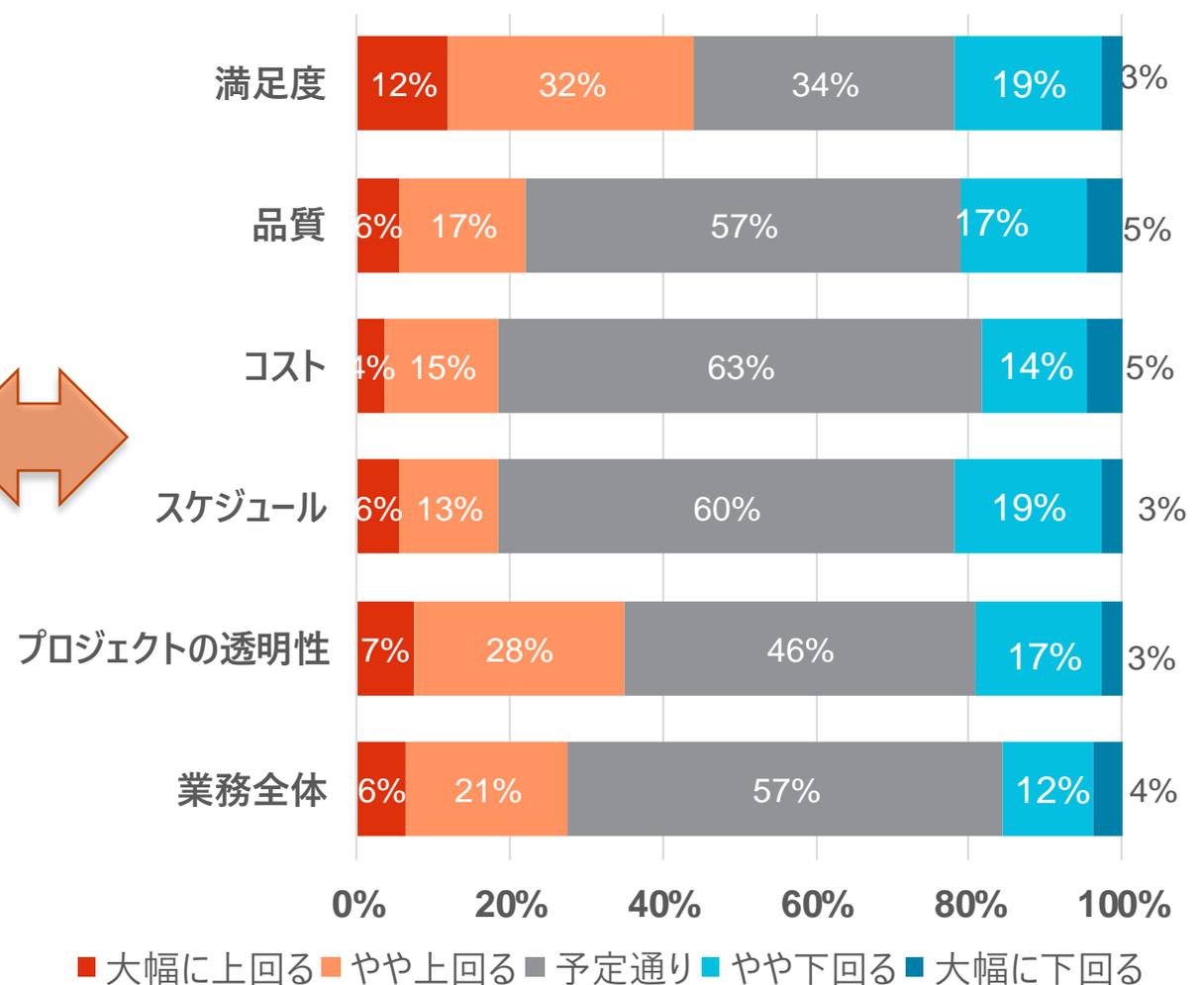
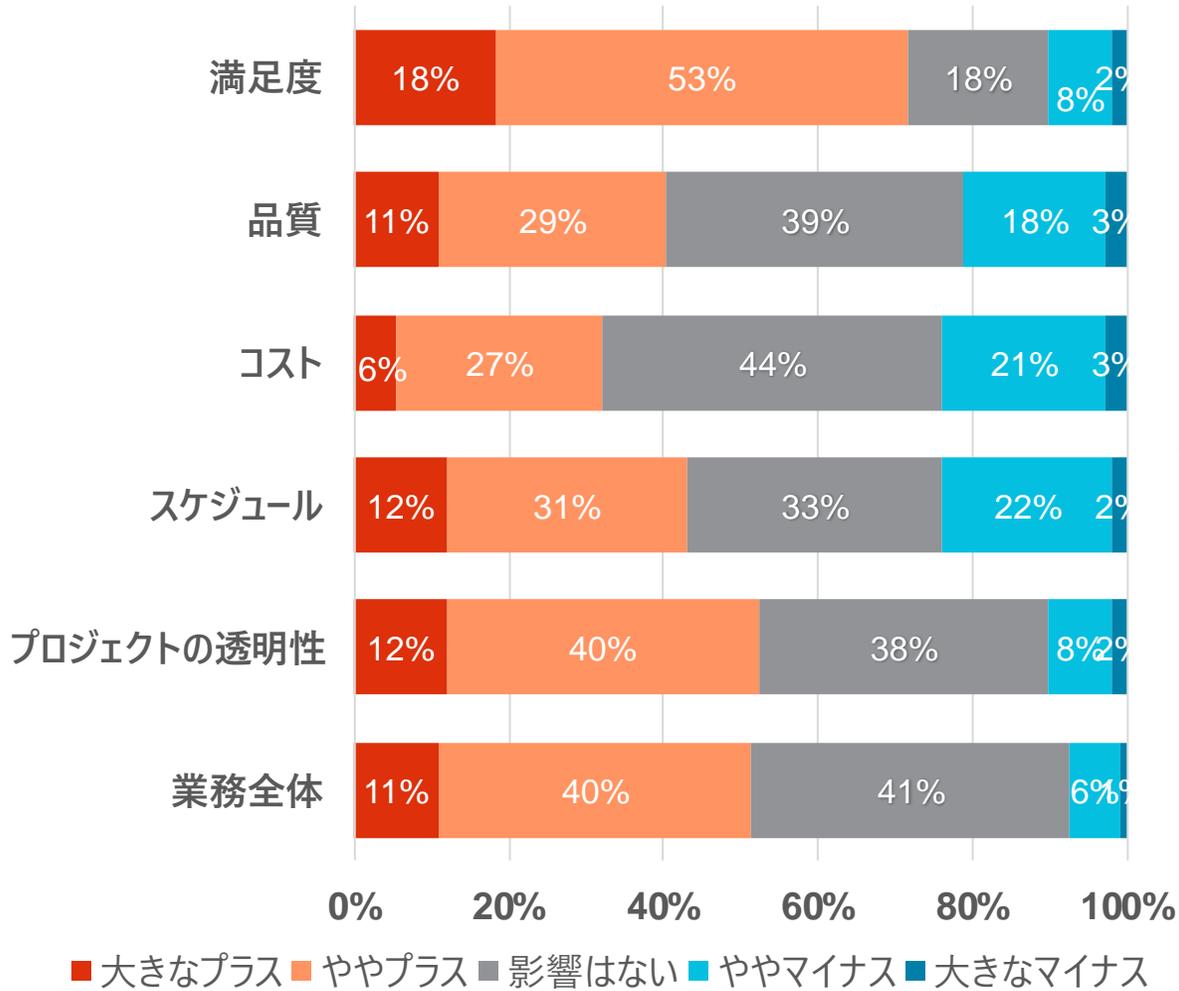
■ 大幅に上回る ■ やや上回る ■ 予定通り ■ やや下回る ■ 大幅に下回る
n=95



回答者の経験したアジャイル適用業務 | アジャイルの成果への貢献

(業務の成果にアジャイルはどの程度影響したと感じていますか?)

(成果達成度 (当初予定比) 【再掲】)



n=108

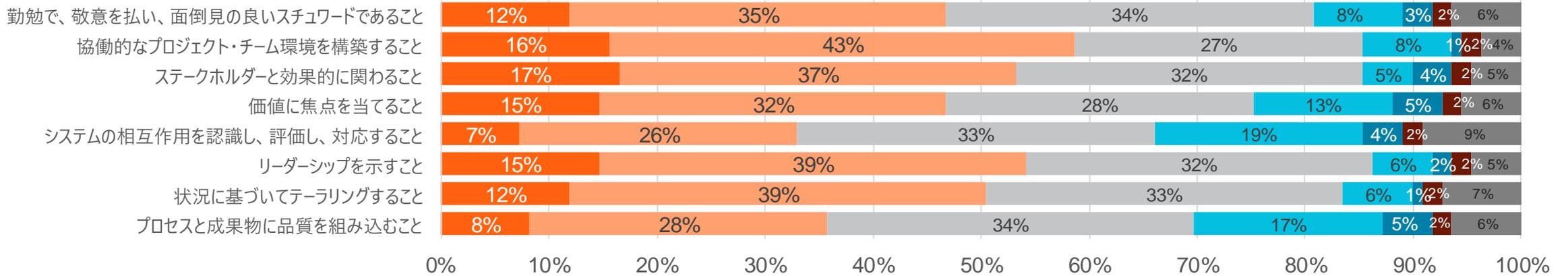
n=108

貢献は高く評価される傾向にあるがマイナスも少なくない

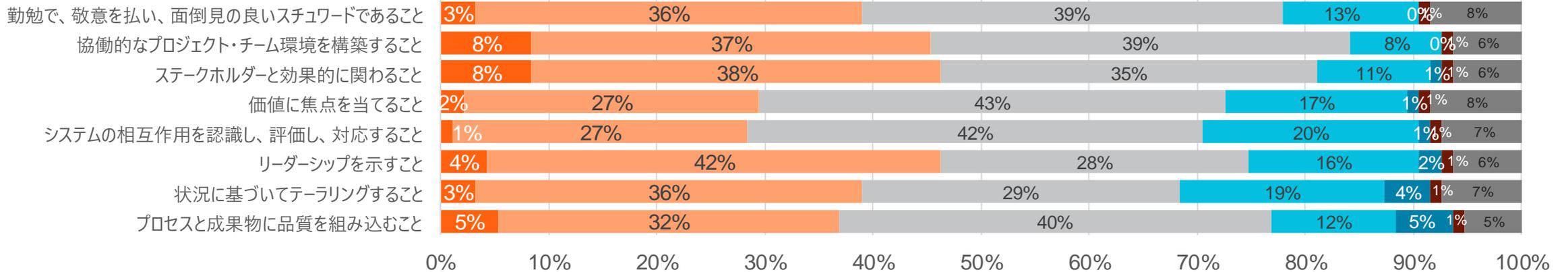


回答者の経験したアジャイル適用業務 PMBOK®ガイドの原則とマニフェストの実践 1/2

<アジャイル適用業務>



<非アジャイル開発>



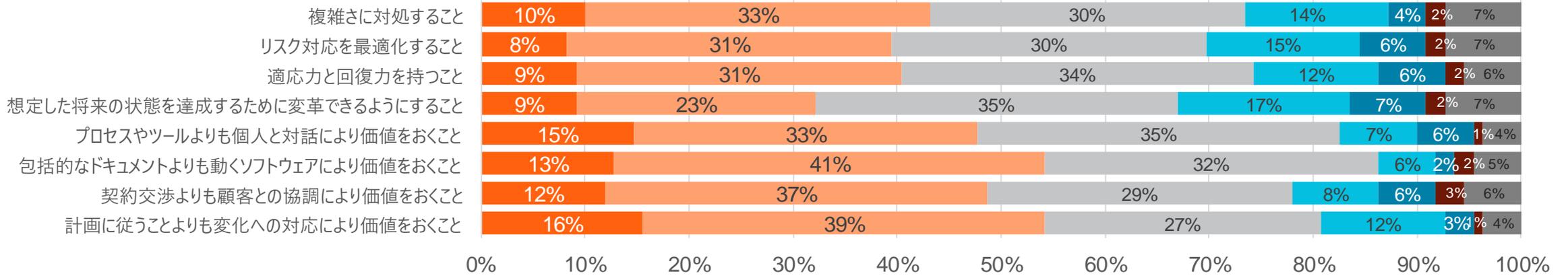
- 非常によく実践できていたと思う
- 実践できていたと思う
- 実践できていた時もあればできていないときもあった
- あまりよく実践できていなかった
- 全く実践できていなかった
- この考え方は同意できないので適用していない
- わからない



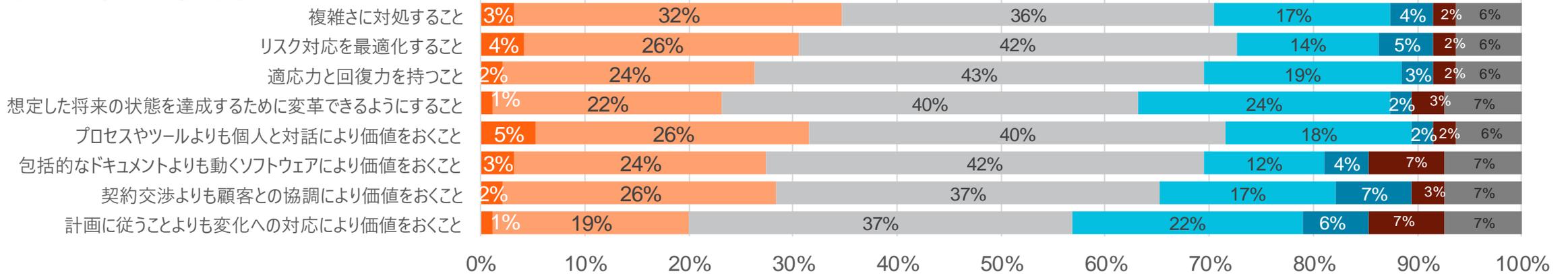
n=108

回答者の経験したアジャイル適用業務 PMBOK®ガイドの原則とマニフェストの実践 2/2

<アジャイル適用業務>



<非アジャイル開発>



- 非常にうまく実践できていたと思う
- 実践できていたと思う
- 実践できていた時もあればできていないときもあった
- あまりうまく実践できていなかった
- 全く実践できていなかった
- この考え方は同意できないので適用していない
- わからない

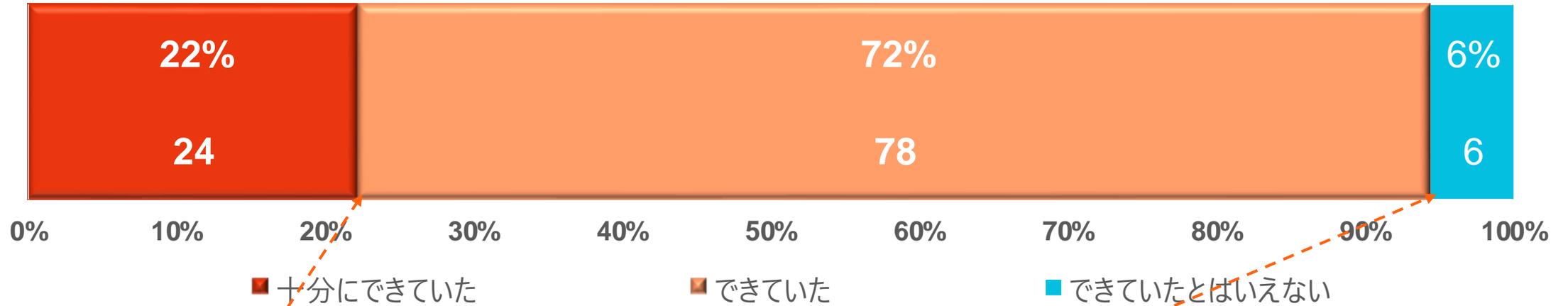


n=108

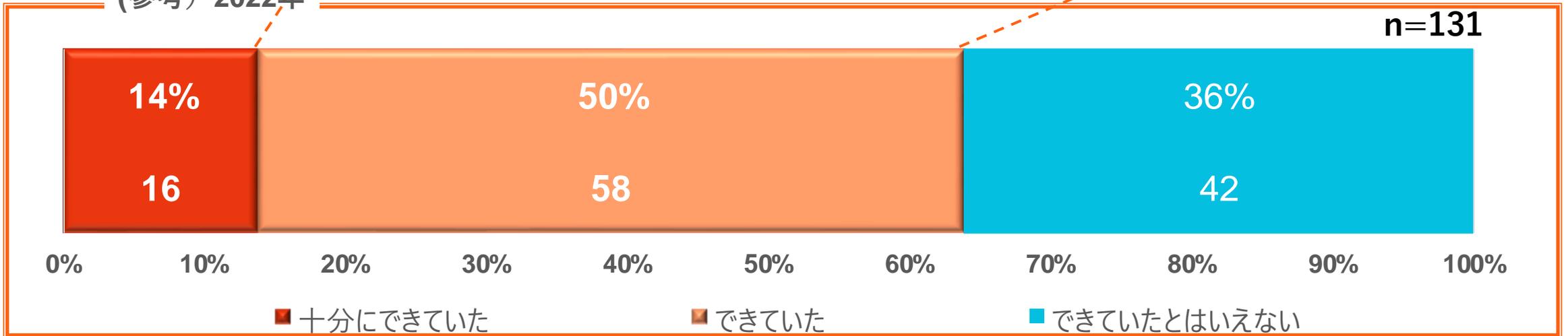
回答者の経験したアジャイル適用業務 | チームの評価

【あなたのチームは、「誰が」「どのように」「何の」作業をするかを選択できましたか】

n=108



(参考) 2022年

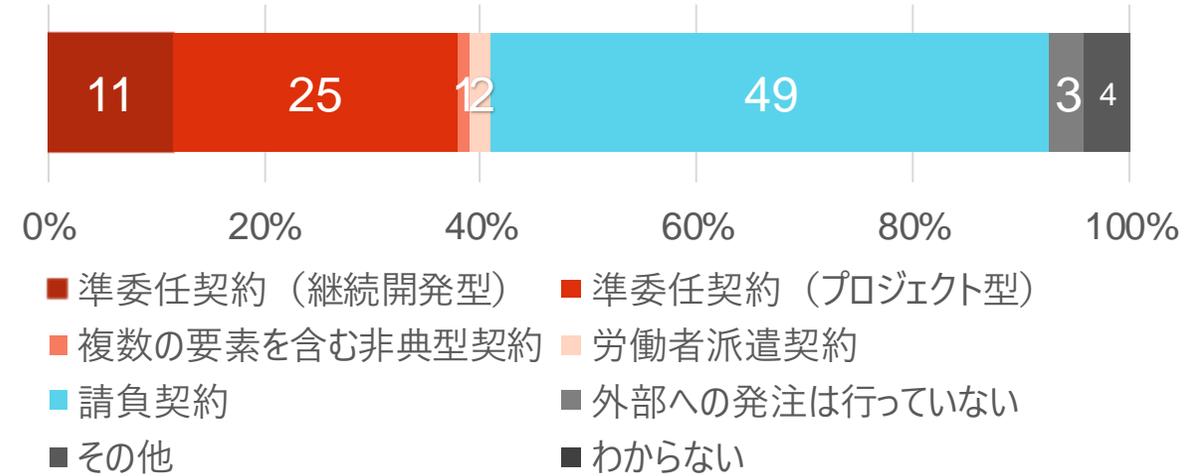
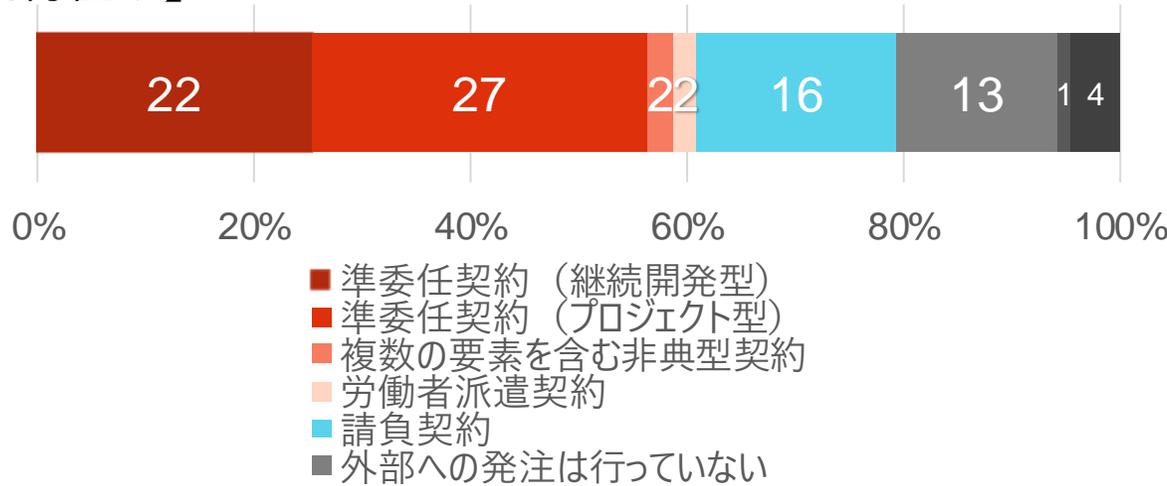


回答者の経験したアジャイル適用業務／非アジャイル開発 | ソフトウェア開発契約

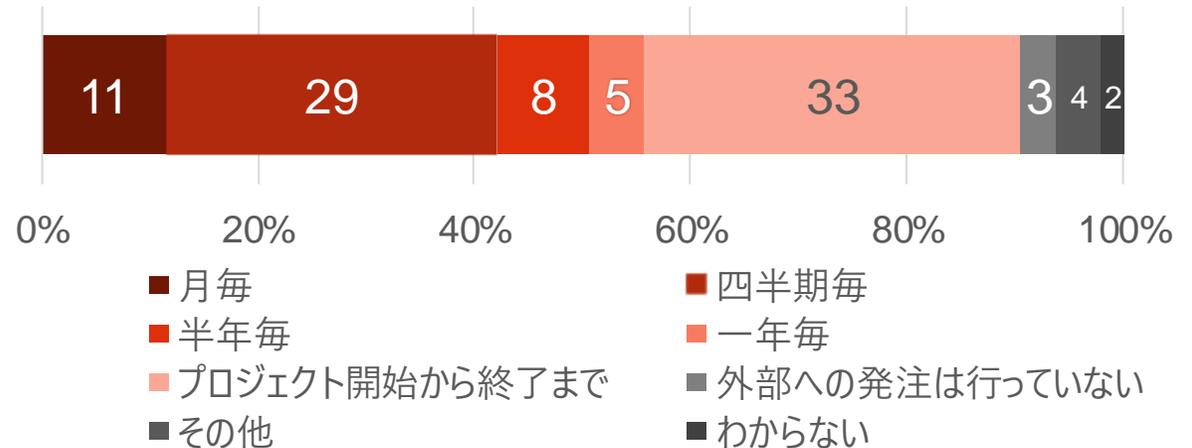
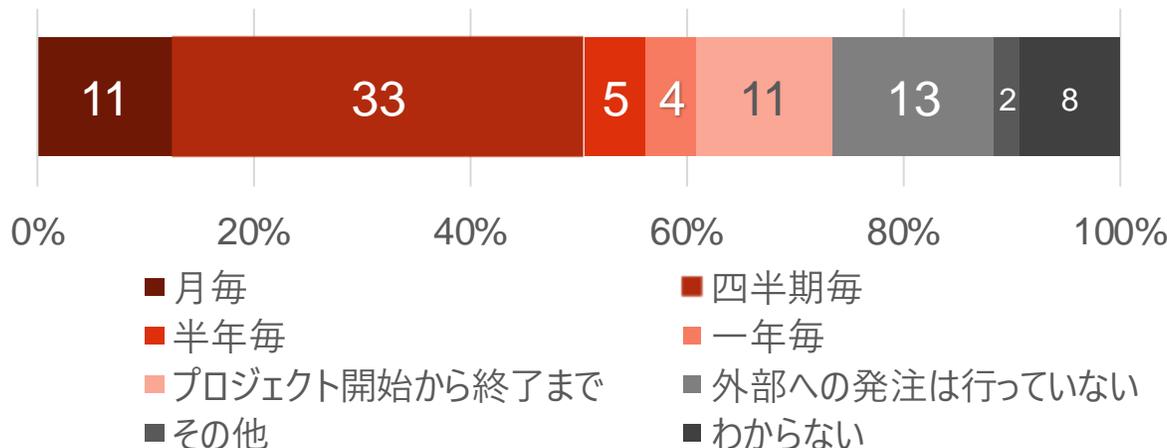
<アジャイル適用業務> n=87

<非アジャイル開発> n=95

【契約種類】



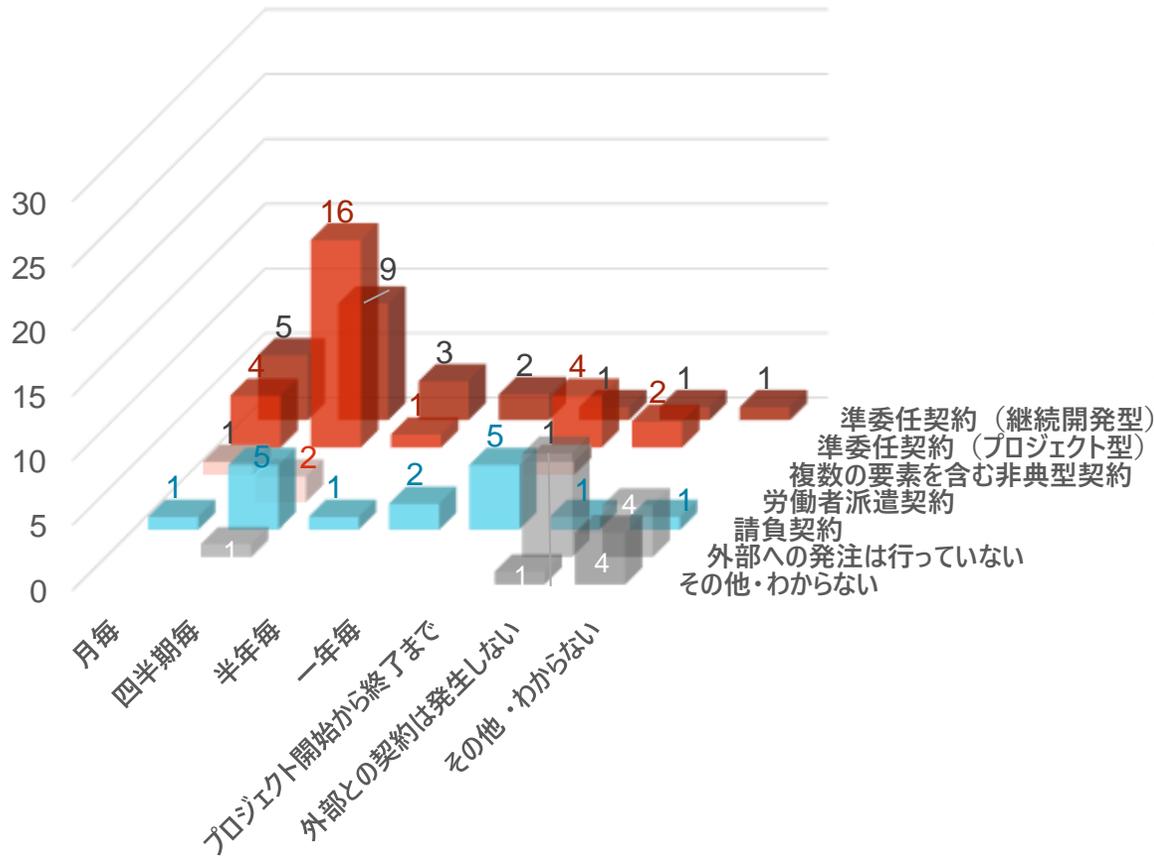
【契約サイクル】



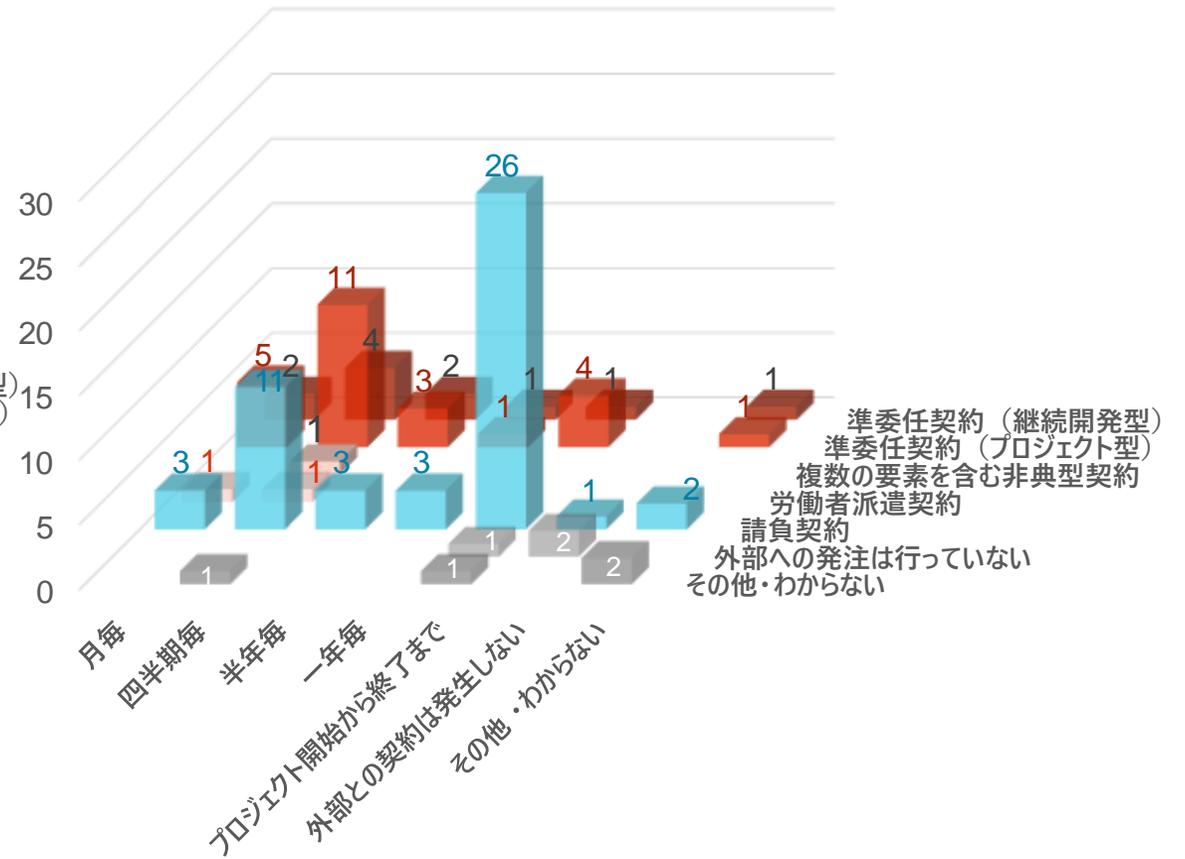
回答者の経験したアジャイル適用業務／非アジャイル開発 | ソフトウェア開発契約

【契約種類と契約サイクル】

＜アジャイル適用業務＞ n=87



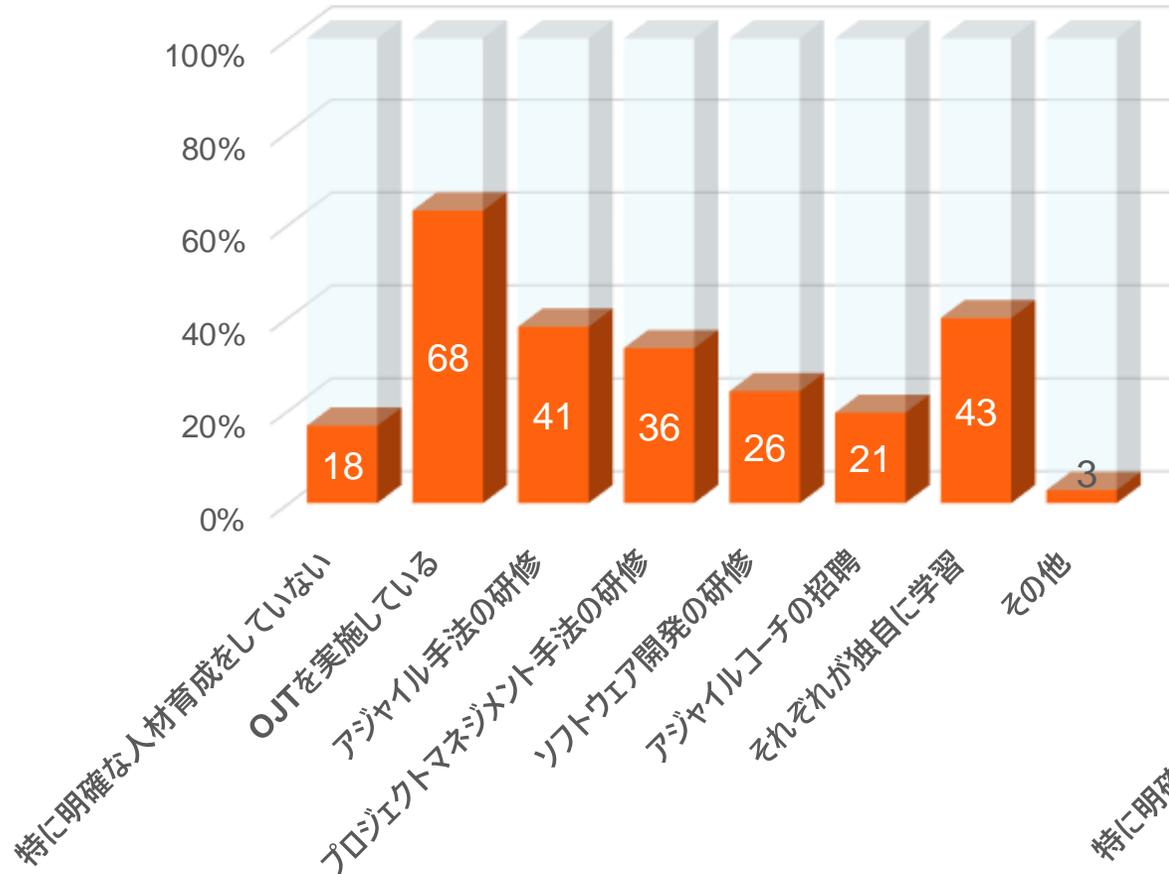
＜非アジャイル開発＞ n=95



回答者の経験したアジャイル適用業務／非アジャイル開発 | 人材育成

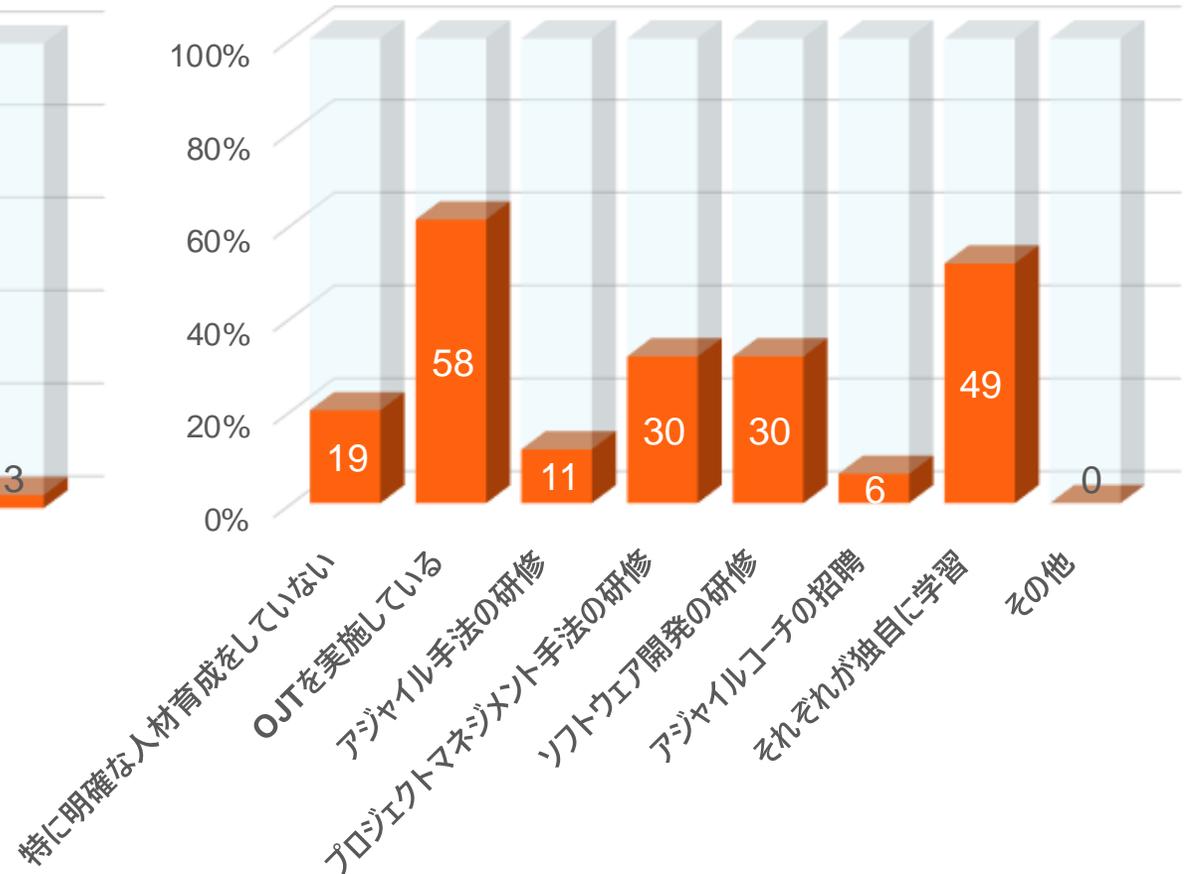
【あなたのチームの人材育成法】

<アジャイル適用業務>



n=108

<非アジャイル開発>



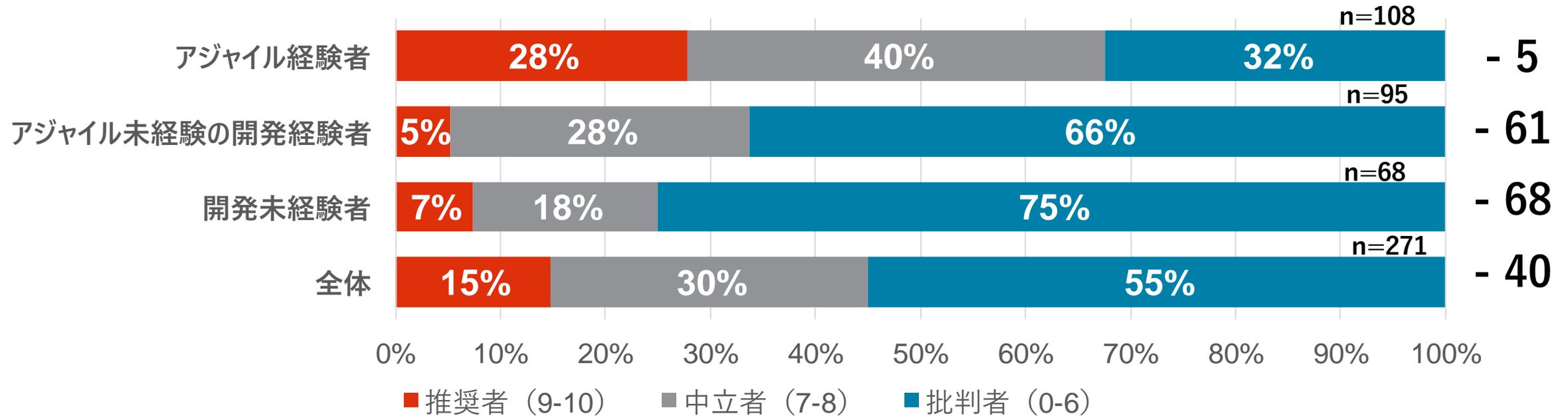
n=95

アジャイルを

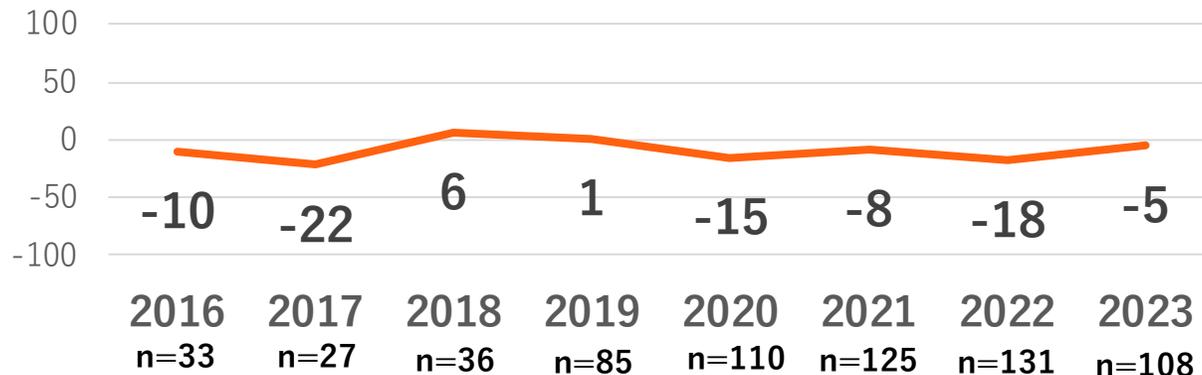
親しい友人や同僚に薦める
可能性はどのくらいありますか？
(ネット・プロモーター・スコア)

親しい友人や同僚に薦める可能性はどのくらいありますか？

【NPS（推奨者の割合 - 批判者の割合）】

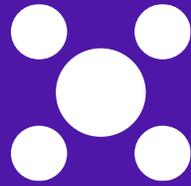


【アジャイル経験者のNPSの推移】

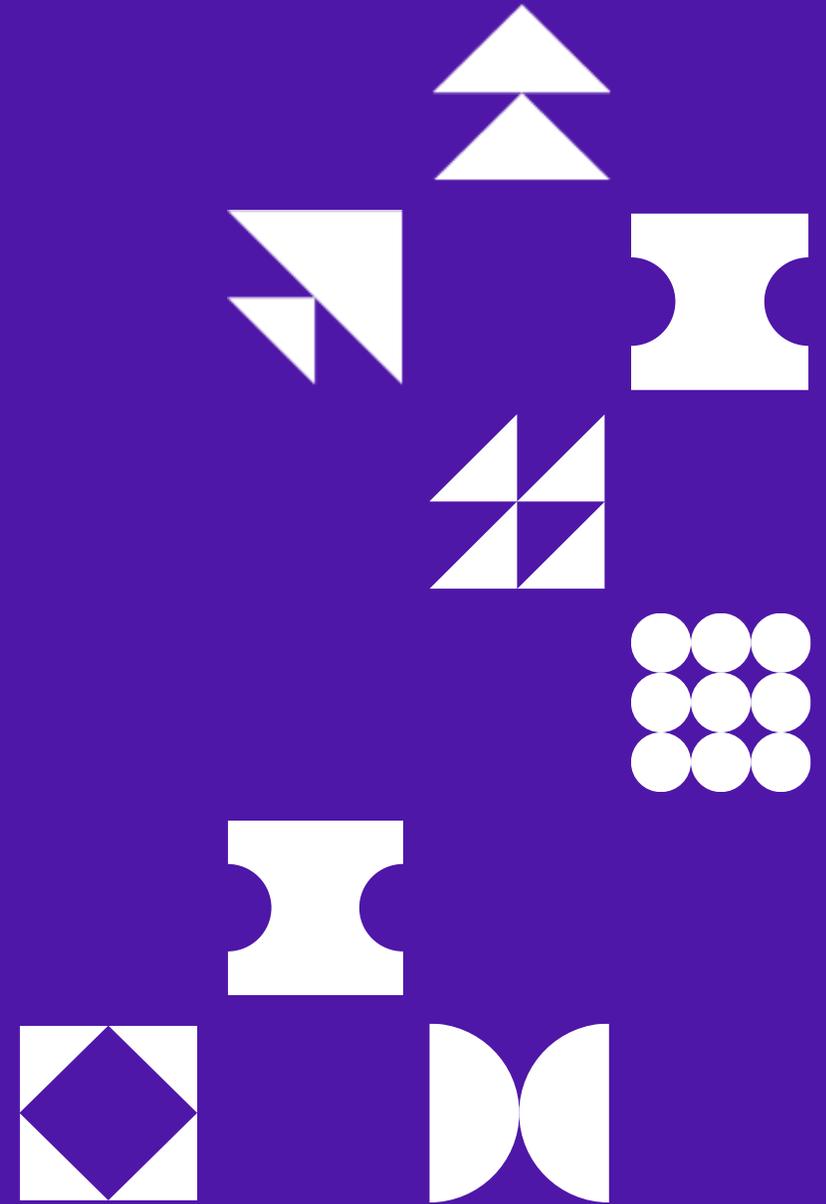


分類	評価
推奨者 (Promoter)	ロイヤルティが高い熱心な顧客。自らが継続購入客であるだけでなく、他者へサービスを勧める『推奨』の役割も担う。
中立者 (Passive)	満足はしているが、それ程熱狂的ではなく、競合他社になびきやすい。
批判者 (Detractor)	劣悪な関係を強いられた不満客。放置しておくと思評を広める恐れがある。

2023年



アジャイル プロジェクト マネジメント 意識調査報告 < 分析と提言編 > ~ 原則の実践 ~



< 分析と提言編 > 原則の実践

- ▶ 今年も推奨されなかった“アジャイル” アンケート結果の振り返り
- ▶ 原則の実践
- ▶ おわりに

今年も推奨されなかった “アジャイル” アンケート結果の振り返り

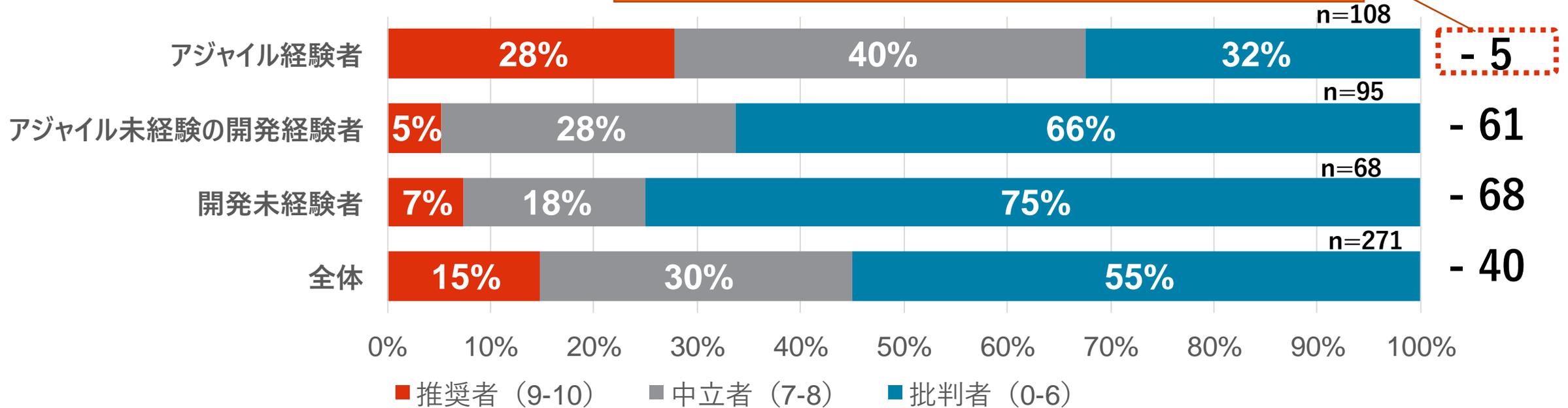
アジャイルは「今年も」アジャイル経験者の「多く」に「推奨されません」でした。



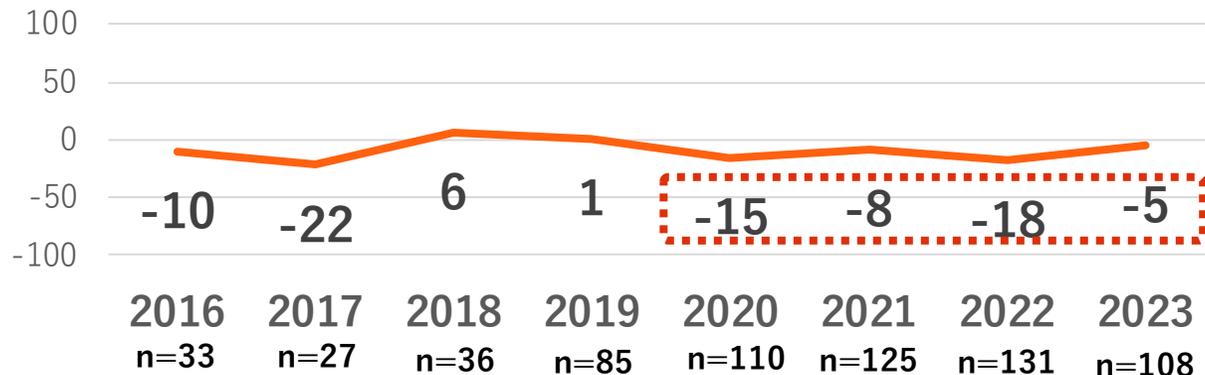
親しい友人や同僚に薦める可能性はどのくらいありますか？ [再掲]

【NPS（推奨者の割合 - 批判者の割合）】

今年もNPSがマイナスとなった要因を探ってみます

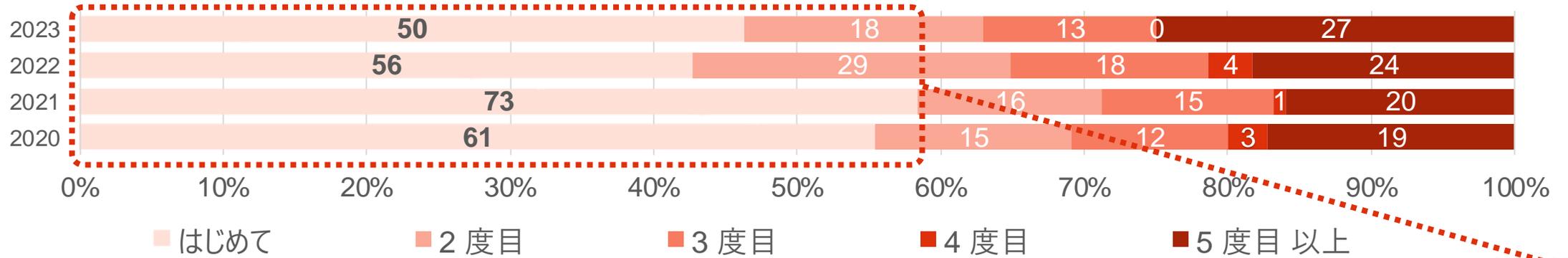


【アジャイル経験者のNPSの推移】

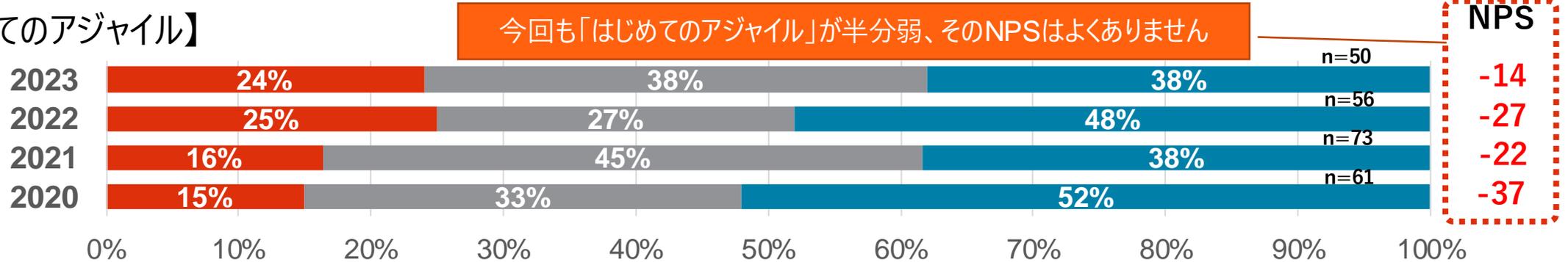


分類	評価
推奨者 (Promoter)	ロイヤルティが高い熱心な顧客。自らが継続購入客であるだけでなく、他者へサービスを勧める『推奨』の役割も担う。
中立者 (Passive)	満足はしているが、それ程熱狂的ではなく、競合他社になびきやすい。
批判者 (Detractor)	劣悪な関係を強いられた不満客。放置しておくと思評を広める恐れがある。

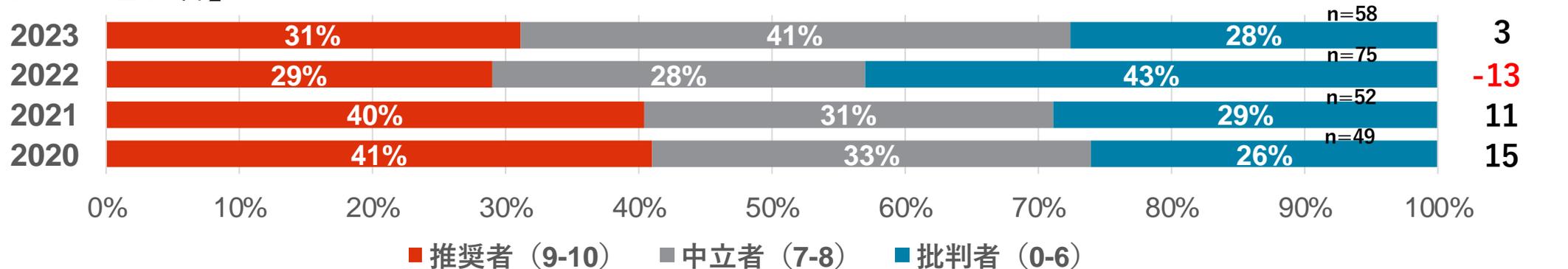
「はじめてのアジャイル」「2回目以上の経験者」のNPSの推移



【はじめてのアジャイル】

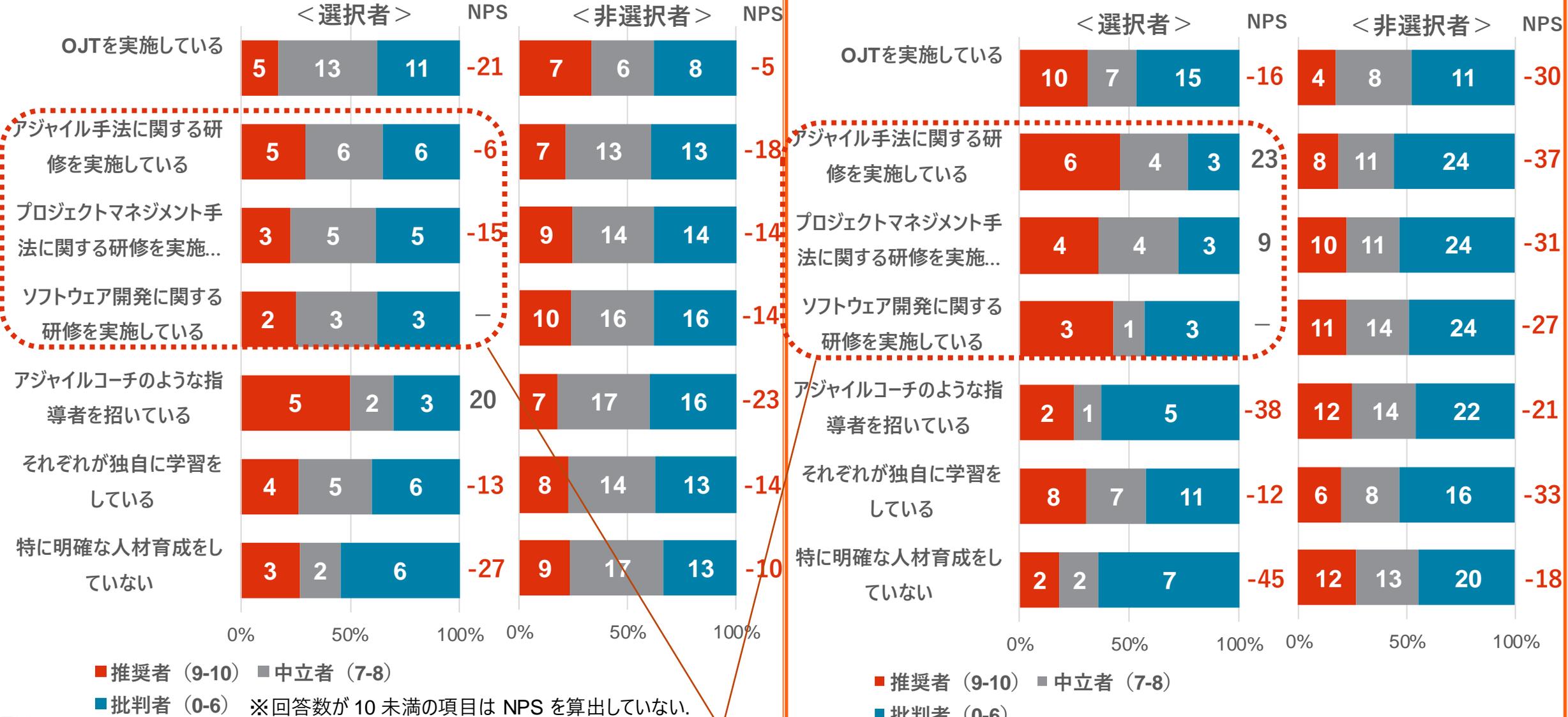


【2回目以上の経験者】



「はじめてのアジャイル」での「チームの人材育成」とNPS

(参考：2022年)



今年の「研修の効果」も、今年は見られません……

「はじめてのアジャイル」のNPS毎の生の声（批判者0-6）

NPS	人数	コメント
1	1	N.A.
2	-	
3	2	<ul style="list-style-type: none"> 顧客の要求を段階的に取り入れる開発手法としてはよいが、要求が固まらずに成果実現が<u>遅延するリスクがある</u>
4	1	N.A.
5	9	<ul style="list-style-type: none"> どんなチームができるかで、その後の完成度や品質にばらつきが出る ベンダーを使う場合準委任契約の場合が多くそうすると責任範囲が曖昧となる <u>経験が浅いため、プロジェクトのコントロールについて学習が必要で、評価できていないため。</u> 実際的な手法である 理屈では有効だと思うが自分自身が<u>上手く使いこなせていない</u>ので、他人へ積極的に進めることが難しい。
6	6	<ul style="list-style-type: none"> <u>ユーザー部門の理解と協力を得られないと実践は難しい</u>為。

批判者によるコメントは少ないが、「遅延リスク」、「品質」、「経験／使いこなし」、「ユーザー部門の理解と協力」などを課題とするものが見られた

「はじめてのアジャイル」のNPS毎の生の声（中立者7-8）

NPS	人数	コメント
7	9	<ul style="list-style-type: none">• <u>お客様がアジャイル開発について知識があり、かつ協力的でない</u>と実施できない。• <u>スピード感としなやかさ</u>• <u>まずはやってみることをお勧めする。その上で改善点を修正して次回につながる、これこそがアジャイルアプローチ</u>だと思う。まずは<u>影響、リスクの少ないところ</u>から。• <u>環境の変化への迅速な対応</u>• <u>組織的に、アジャイルを理解し期待するメンバー</u>の中であれば、必ず成果を出せると考えます。アジャイルに対して疑念を持ち抵抗感のあるメンバーが、上位に居ると対応困難です。• <u>多くの場面で最適な手法だ</u>と思っているから。
8	10	<ul style="list-style-type: none">• <u>楽しい</u> ○• <u>適用範囲を間違えなければ、確実にメリット</u>はある

「はじめてのアジャイル」のNPS毎の生の声（推奨者9－10）

NPS	人数	コメント
9	3	<ul style="list-style-type: none"> アジャイル開発が向くプロジェクトと向かないプロジェクトがあるが、向くプロジェクトには薦める？ 適応できる領域があれば積極的に薦めたいが、アジャイルだけを薦めたいわけではない。
10	9	<ul style="list-style-type: none"> カンバンを活用したおかげで、よくわからず文句を言っていたステークホルダーが黙った。 これからの時代、顧客価値を採求するにあたり、計画通りに進む可能性などは低く、<u>適応しながら迅速に進めることが重要</u>だと考えるから ソフトウェア開発に限らず<u>必要と認識</u>しているため チームが成長できるものであり、また世の中の状況的に<u>必須なスキル</u>と感じているため 価値に注目し、より良いプロダクトを生み出す<u>効果がある</u>と感じるから。○ 外部情勢の変化が激しい情勢でもあり、変化に柔軟に、かつスピーディーに対応するしくみづくりが<u>今後益々重要</u>になってくると思っているため。

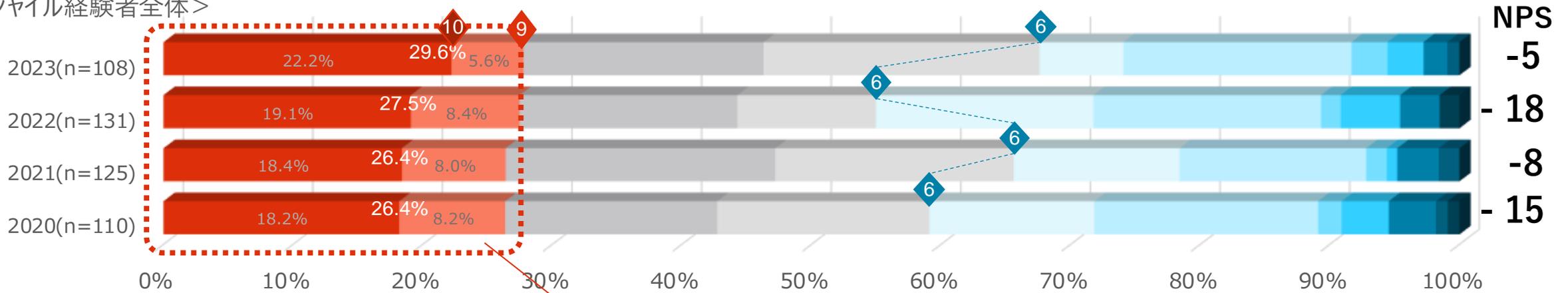
推奨者によるコメントには「必要だから」が多く見られた。

「はじめてのアジャイル」の課題の発掘には、さらに踏み込んだ調査が必要。

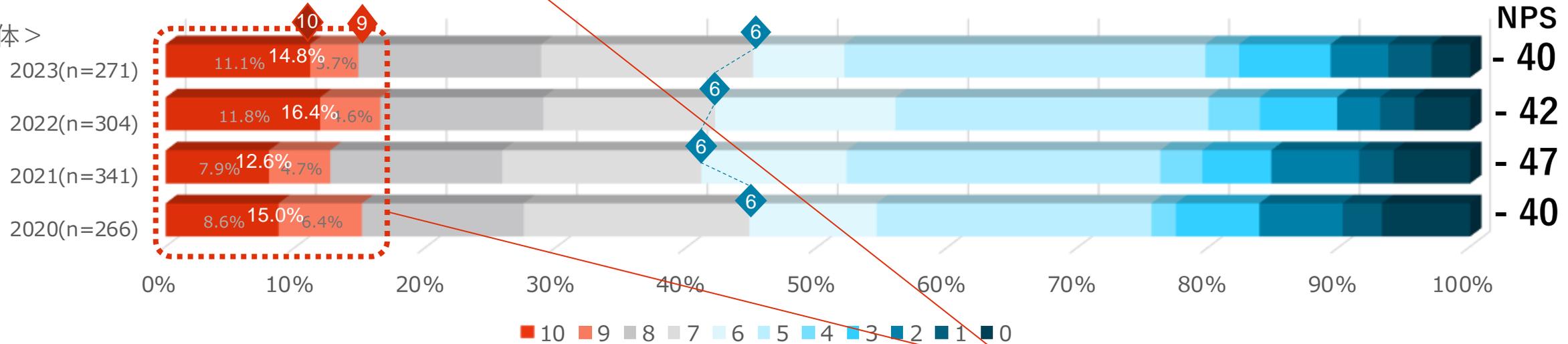
NPSで回答者がつけたスコア 推奨者の比率は殆ど増えていない

【NPS（回答分布の内訳推移）】

<アジャイル経験者全体>



<全体>

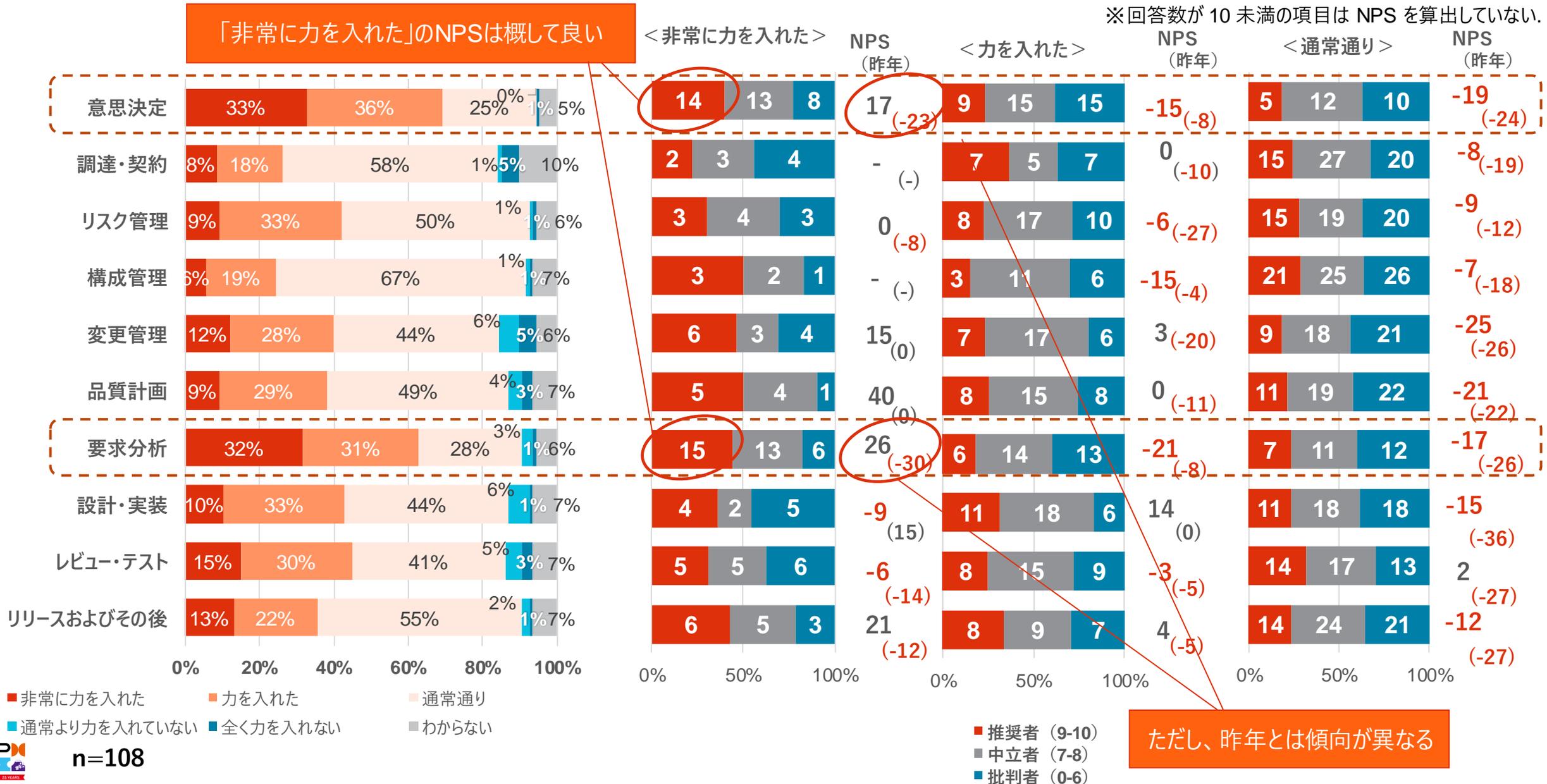


この4年間推奨者は殆ど増えていない → アジャイル推奨者に見られる傾向は？

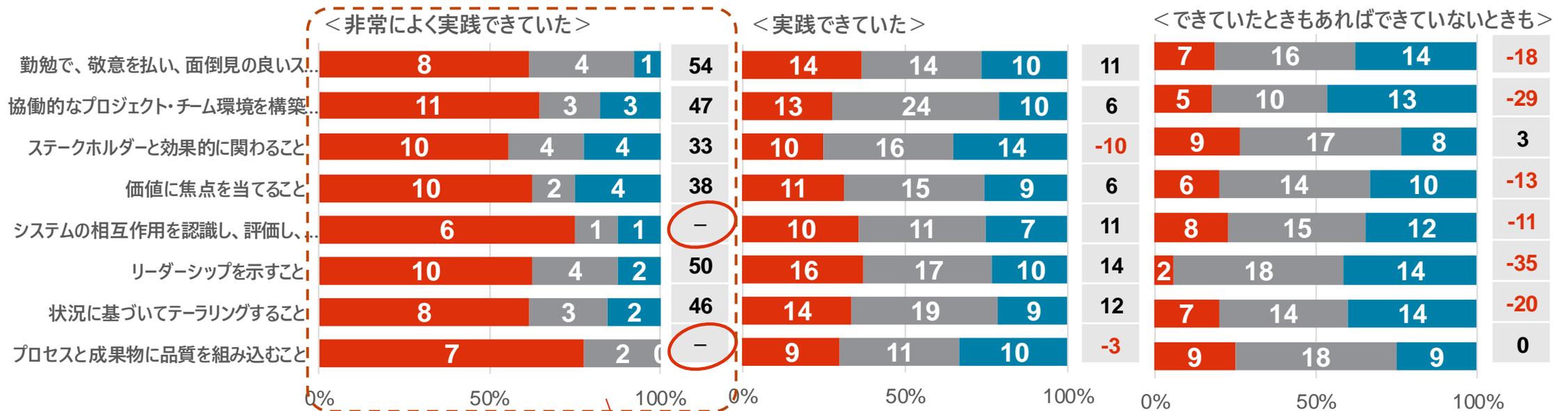
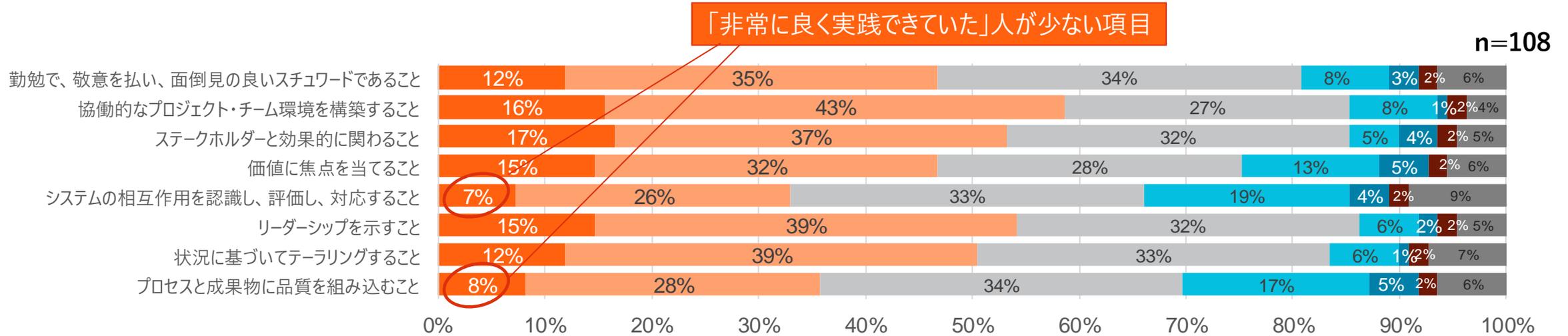


アジャイル推奨者の傾向 マネジメントの重点とNPS

※回答数が10未満の項目はNPSを算出していません。



アジャイル推奨者の傾向 PMBOK®ガイドの原則とマニフェストの実践 1/2



PMBOK®ガイドの原則が「非常に良く実践できていた」人のNPSは抜群

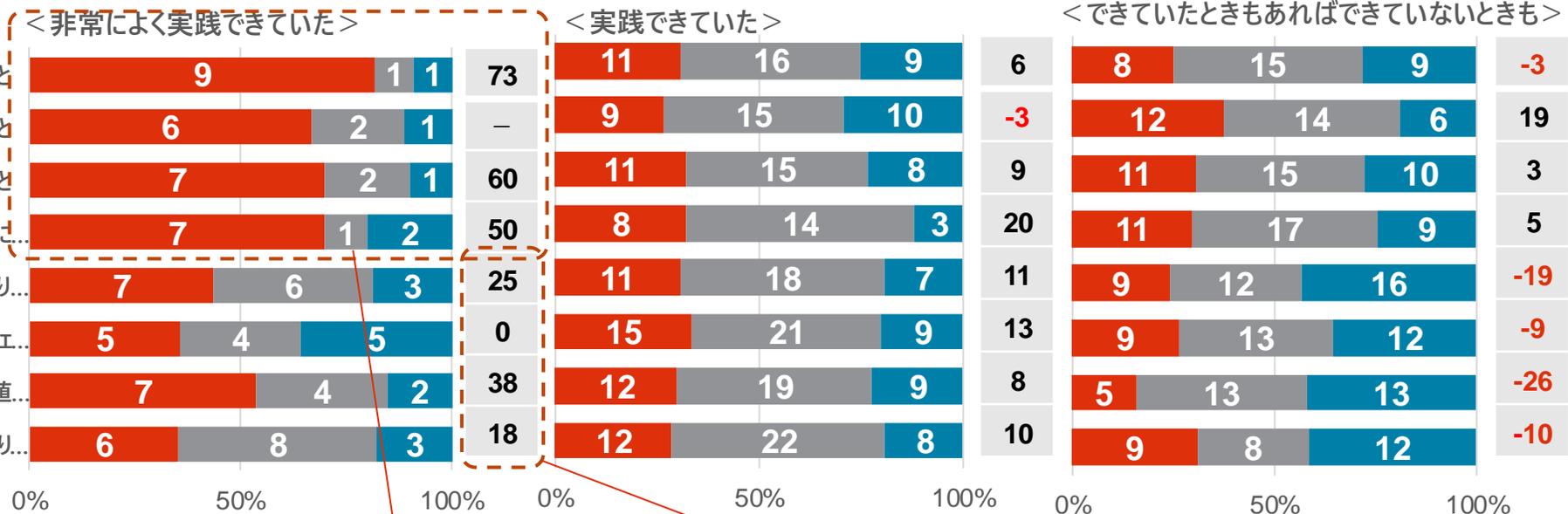
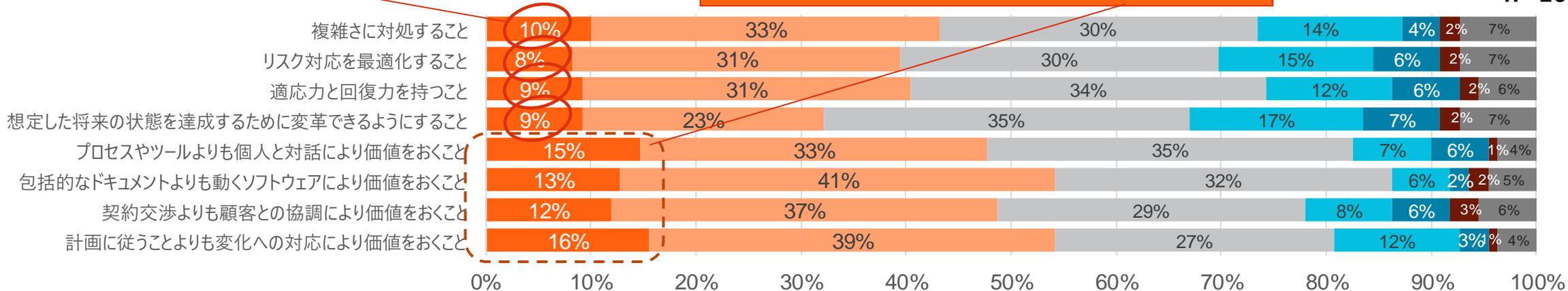
■ 推奨者 (9-10) ■ 中立者 (7-8) ※回答数が10未満の項目はNPSを算出していない。
■ 批判者 (0-6)

アジャイル推奨者の傾向 PMBOK®ガイドの原則とマニフェストの実践 2/2

「非常に良く実践できていた」人が少ない項目

マニフェストは「非常に良く実践できていた」が少ない

n=108



PMBOK®ガイドの原則が「非常に良く実践できていた」人のNPSは抜群

■ 推奨者 (9-10) ■ 中立者 (7-8) ※回答数が10未満の項目はNPSを算出していません。
■ 批判者 (0-6)

マニフェストが「非常に良く実践できていた」人のNPSはそこそこ

原則の実践

NPS向上のヒント

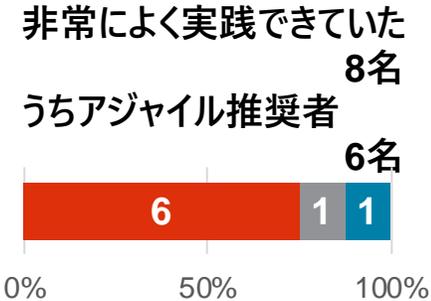
「非常に良く実践できていた」人が少しでも増えれば……

PMBOK®ガイドで「非常に良く実践できていた」人が少なかった原則

n=108

システムの相互作用を認識し、評価し、対応すること

プロジェクト・パフォーマンスに前向きな影響を与えるため、全体的な方法で、プロジェクト内およびプロジェクト周辺の変化し続ける状況を認識し、評価し、それに対応する。



プロセスと成果物に品質を組み込むこと

品質に継続的に焦点を当て、プロジェクトの目標を満たし、ニーズ、使い方、および関連ステークホルダーが設定した受け入れ要求事項に合致した成果物を生み出す。



複雑さに対処すること

プロジェクトの複雑さを継続的に評価し、それに対応することで、プロジェクトチームがプロジェクトライフサイクルに適切に対応できるようなアプローチと計画になる。



リスク対応を最適化すること

プロジェクトとその成果へのプラスの影響を最大限に高め、マイナスの影響を最小限に抑えるため、好機と脅威の両方のリスク・エクスポージャーを継続的に評価する。



適応力と回復力を持つこと

組織及びプロジェクト・チームのアプローチに適応力と回復力を備えることは、プロジェクトが変化に適応し、挫折*から回復し、プロジェクト作業を推進する上で役立つ。

*英文はsetback



想定した将来の状態を達成するために変革できるようにすること

プロジェクトの成果によって生み出される意図した将来の状態へと現状から移行するために必要となる、従来と異なる新たな振る舞いやプロセスを採用し維持するために、影響を受ける人々に準備してもらう。



[Source] Project Management Institute, Inc., プロジェクトマネジメント知識体系ガイド (PMBOK®ガイド) 第7版+プロジェクトマネジメント標準, PMI日本支部, 2021, p.37,47,50,53,55,58

システムの相互作用を認識し、評価し、対応すること

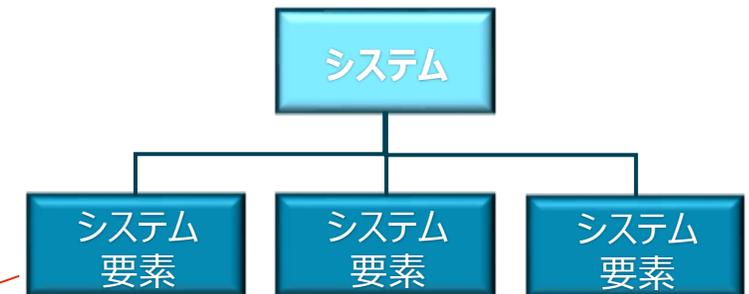
システムとは

システム (system)

- 一つ以上の明記された目的を達成するために組織された相互に作用する要素の組合せ。

システム要素 (system element)

- システムを構成する要素の集合の一部分。
- ✓ 例 システム要素の例は、ハードウェア、ソフトウェア、データ、人間、プロセス（例えば、利用者にサービスを提供するプロセス）、手順（例えば、操作指示）、設備・施設、資材、自然に発生する実体又はそれらの組合せ。



システム要素はシステムのこともある = システム・オブ・システムズ

システムの相互作用を認識し、評価し、対応すること

システム・オブ・システムズ (SoS)

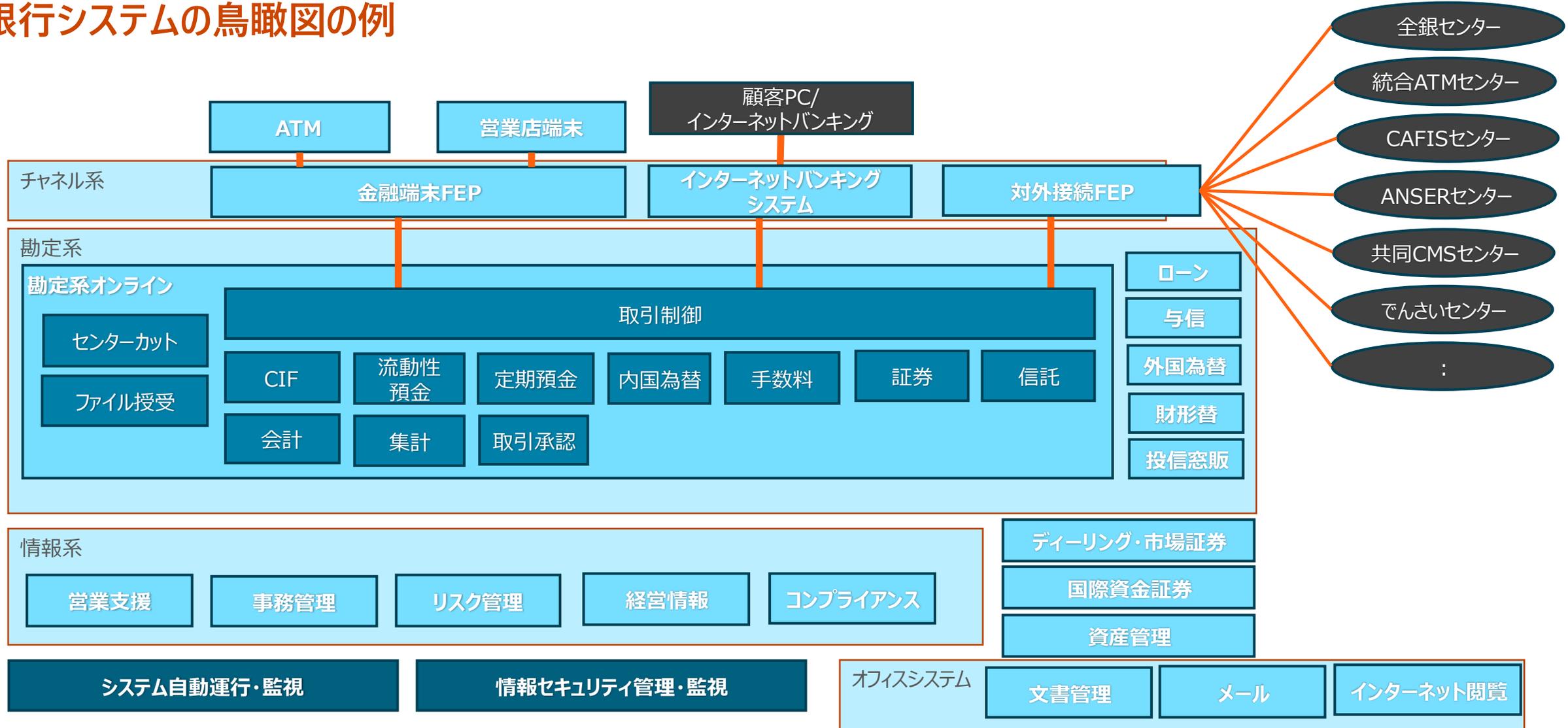
多くのシステムは、必ずしも自らの管理下でないシステムとも協調し、適応する必要がある

システム オブ システムズ (SoS, System of Systems) は、その要素それぞれがシステムである一つの対象システムである。SoS は個々のシステムでは達成できないあるタスクのために一連のシステムをまとめている。SoS を構成する、個々の構成システム (constituent system) は、SoS 内で協調し、かつ、SoS の目標に合致するために適応しながら、自身の管理、ゴール及び資源を保つ。

SoS の種類 (System of System Types)	説明
仮想的な (Virtual)	<ul style="list-style-type: none">- 管理権限の中心がない。- 集約して合意された目的がない。- 他の種類のSoS に比べて見えにくい、SoS を維持するためのメカニズムによって、創発的に振る舞う。
協調的な (Collaborative)	<ul style="list-style-type: none">- 合意された目的を満たすためにコンポーネントシステムが自発的に相互作用する。- 標準を相互運用し、実施し、そして維持する方法を共同で決定する。
認められた (Acknowledged)	<ul style="list-style-type: none">- SoS に対する目標、指定された管理者、及び資源が認識されている。- 構成システムは独立した所有、管理及び資源を保持する。
指揮管理された (Directed)	<ul style="list-style-type: none">- 特定の目的を満たすために構築され管理される統合されたSoS。- 中央から管理し発展させる。- コンポーネントシステムは独立して運用する能力を維持する。- 通常の運用モードが中央の目的に従属している。 <p>注記 この種類のSoS では、SoS を指揮する何らかの中央がある。</p>

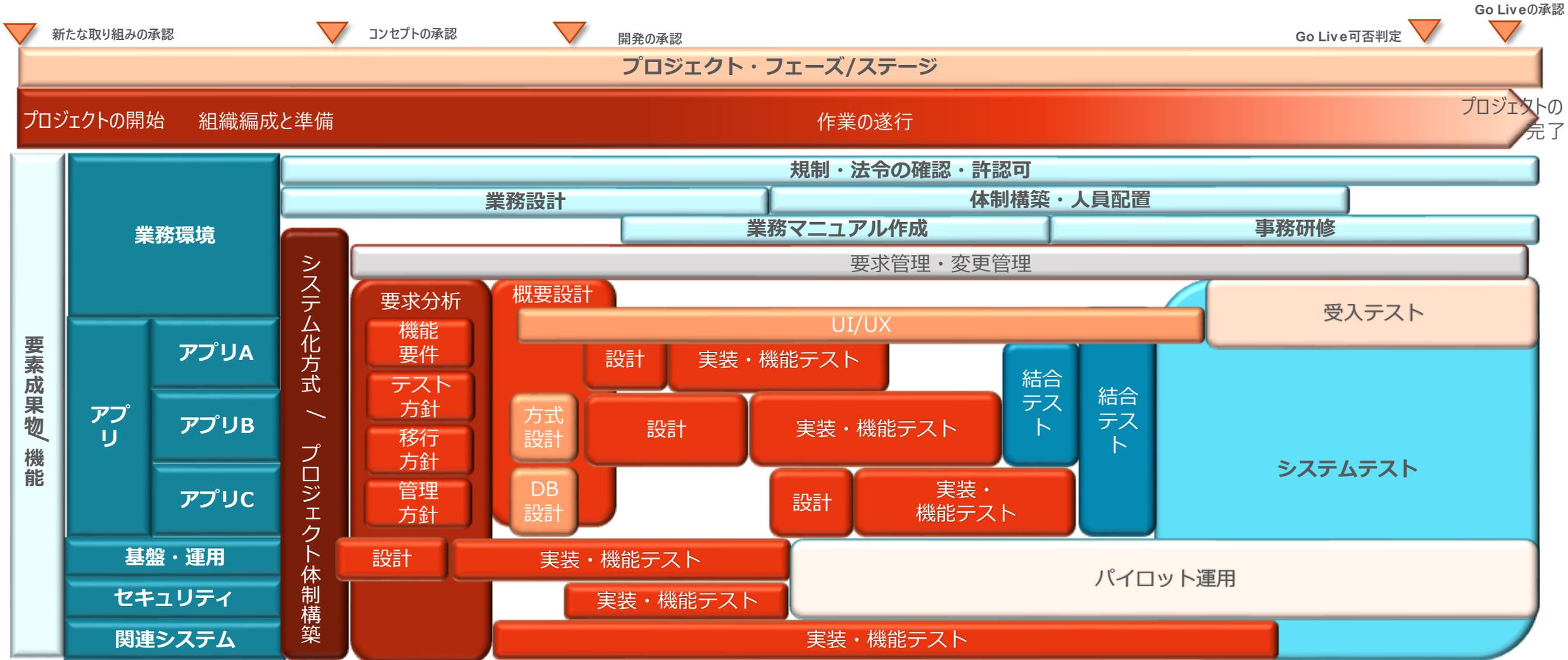
システムの相互作用を認識し、評価し、対応すること

銀行システムの鳥瞰図の例



システムの相互作用を認識し、評価し、対応すること

プロジェクトの要素とその関連の例



システムの相互作用を認識し、評価し、対応すること

システムエンジニアリングとライフサイクル

システムエンジニアリング（systems engineering）

- 利害関係者のニーズ、期待及び制約の集合を、解決するソリューションへ変換するため、及びソリューションが用いられる全期間を通じて、それを支援するために要求される、技術上及び管理上の作業の全体を統括するような、複数の専門分野を横断した取組方法。

ライフサイクル（life cycle）

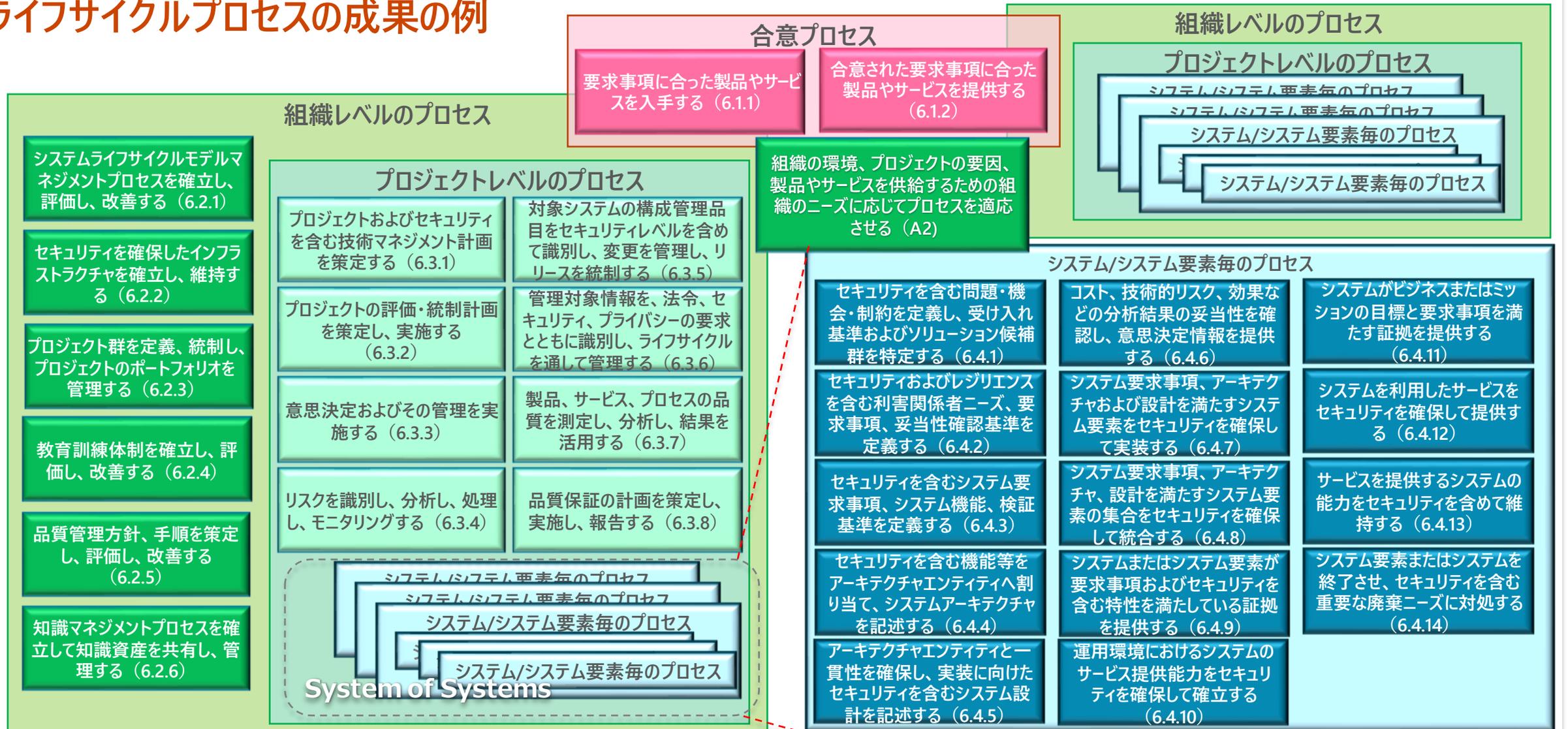
- システム、製品、サービス、プロジェクト又は人が作った他の実体の構想から廃止までの漸進的な発展。

ライフサイクルモデル（life cycle model）

- 段階に編成されることもあるライフサイクルに関係するプロセス及びアクティビティの枠組みで、情報伝達及び理解のための共通に参照できる役割をもつもの。

システムの相互作用を認識し、評価し、対応すること

ライフサイクルプロセスの成果の例



複雑さに対処すること

複雑さを管理する

メンテナンスしやすいシステム

1. 小さなモジュール：モジュールが小さければ小さいほど、変更（あるいは修理）の影響を限定しやすくなる。
2. モジュールの独立性：1つのモジュールの内部要素が他のモジュールの内部要素に依存している割合が小さいほどメンテナンスはやさしい。
3. ブラックボックス：システム、プログラム、あるいはモジュールを、そのプロセスの変換の詳細を無視して入力と出力の観点だけから捉える。
入れ子型構造 – ブラックボックスが入れ子型構造になっているモジュール体系は読みやすく、修正とデバッグもやりやすい。
4. 概念モデル：概念的に理解しやすく問題の正確に合わせたモデル化が必要。
5. 基本と細部の区別：細かい部分の変更は基本部分の変更に比べてはるかに多い。

メンテナンスとテストがしやすいトップダウン設計を作り上げる戦略 ➡ 昔は・・・構造化設計

複雑さに対処すること

複雑さを管理する

少しでも大きな規模で複雑さを管理したいなら、よりテクニカルなソフトウェアの情報システムと同じように、人間の組織という情報システムの複雑さを管理する必要があります。

ソフトウェア産業は、全体として今触れたようなことにほとんど注意を払っていないような感じがします。そのため、巨大な泥だんごのようなシステム、手に負えなくなった技術的負債、絶望的な数のバグ、自分のシステムに変更を加えるのを恐がる組織といった少しでもソフトウェア開発に携わったことがあればおなじみの光景が生まれているのです。

あらゆる情報システムで構造的に複雑さを管理するつもりなら、これら5要素についても考える意味があります。

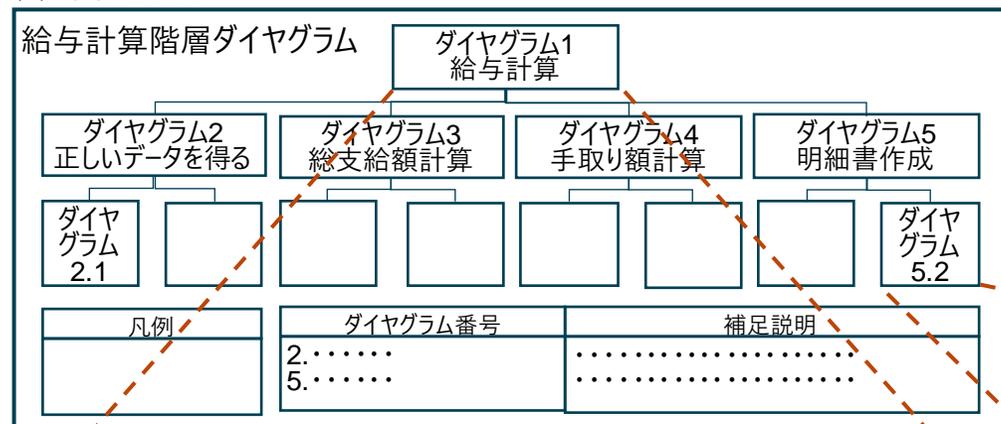
- モジューラ-性
- 凝集度（一体性）
- 関心の分離
- 情報隠蔽/抽象化
- カップリング

⇨ Tom Demarcoの「メンテナンスしやすいシステム」

複雑さに対処すること

構造化設計技法の例 HIPO / IPT

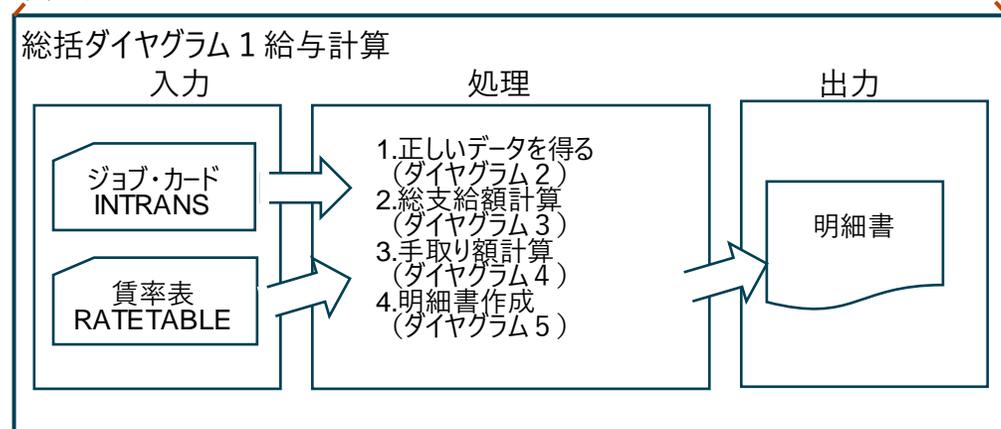
(1) 図式目次



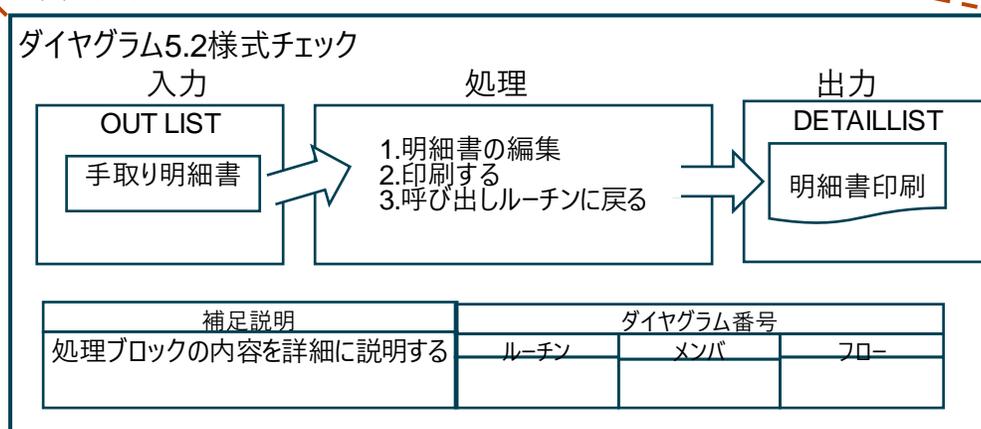
IPT(Improved Programming Technologies)はプログラムの生産性と信頼性向上のために、IBMが長年の経験に基づいて開発した技法ツール、管理方法などの種々の技術の総称である。

IPTの仕様書体系はHIPO(Hierarchy plus Input Process Output)である。HIPOは基本的には3種類のフォームシートからなる。即ち、処理単位間の階層構造を記述する図式目次(H図)、各処理単位の入出力と詳細機能との関係を機能の実行順を含めて記述する総括ダイヤグラム及び詳細ダイヤグラム(IPO図)、さらにH図やIPO図の補足として論理の詳細や処理機能とモジュール名との相互参照などを記述する補足説明の3種類である。

(2) 総括ダイヤグラム



(3) 詳細ダイヤグラム



複雑さに対処すること

プロセス・マイニング — 業務及びシステムの見える化

情報システムとそれが支えている業務プロセスの関係は、ますます強固でわかり難くなっています。

プロセスマイニングの目的はイベントデータを利用してプロセス関連の情報を取り出すこと、たとえばエンタープライズ・システムに記録されたイベントをよく観察することによって自動的にプロセスのモデルを発見することにあります。



プロセスと成果物に品質を組み込むこと

テスト・ファースト

仕様は、コードを書き終えてから書くものではありません。コードを書く前に必要なものです。そこで、コードを書く前に仕様（テスト）を書くこととなります。まだコードがないので、仕事を楽にすることに、より明確に重点が置かれます。この時点での目標は、できる限り仕様を単純明確に表現するために、テストをできる限り単純にすることです。

このアプローチに従えば、定義上、設計を抽象化することになります。テストケースをうまく書けるようにするために、考えが表現しやすくなるようなインターフェイスを定義することになり、コード自体も使いやすくなります。仕様（テスト）を書くことは設計作業の一部です。コード自体の仕組みから切り離れた形で、ほかのプログラマーに自分のコードをどのように使ってもらいたいかを設計するわけです。コードの実装の詳細について考える前に、これをすべて済ませることになります。抽象化を基礎とするこのアプローチは、コードが何をしなければならないかとそれをどのように実現するかを切り離すために役立ちます。この時点では、ふるまいをどのように実装するかについてはほとんど、あるいはまったく言っていません。それはあとで考えることです。

テストできないものは作らない（必要ならスタブやドライバを同時に作る）

➡ 必ずどうやってテストするかを考えてから作り始める・・・テストを考えられないなら既に品質に問題がある

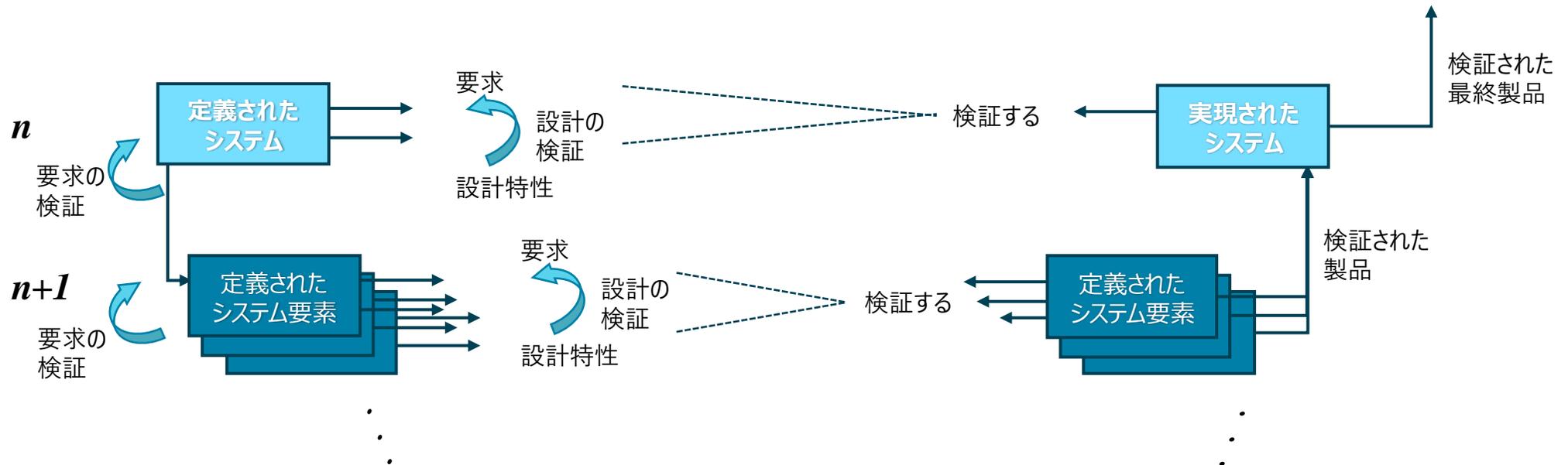
リスク対応を最適化すること

技術的リスクを管理する

技術的リスク：システムライフサイクル内で、システムの技術的要求が達成されない可能性。

一つ一つのシステム要素の品質を管理する ➡ 手戻りを最小限にする最適化されたリスク管理

各システムおよびそれぞれのシステム要素を検証し、上位レベルのシステムへ統合される前にできるだけ修正する。この図で、検証という用語が示されている箇所では常時、該当する検証プロセスが実行される。



適応力と回復力を持つこと

可能性の高い変化は事前に対応を組み込み、起こりうる遅れには必要な余裕を確保しておく

適応力；変化する状況に対応する能力

➡ 事前に予想される変化は必要に応じて事前に対応しておく（ただし複雑になる）

回復力；影響を緩和する能力、遅れから回復する能力

➡ 事前に予想できない事態に対し、合理的な範囲であらかじめ余裕を組み込んでおく

※ この事態は、プロジェクトの能力を超える事態に遭遇したときにしか発生しない

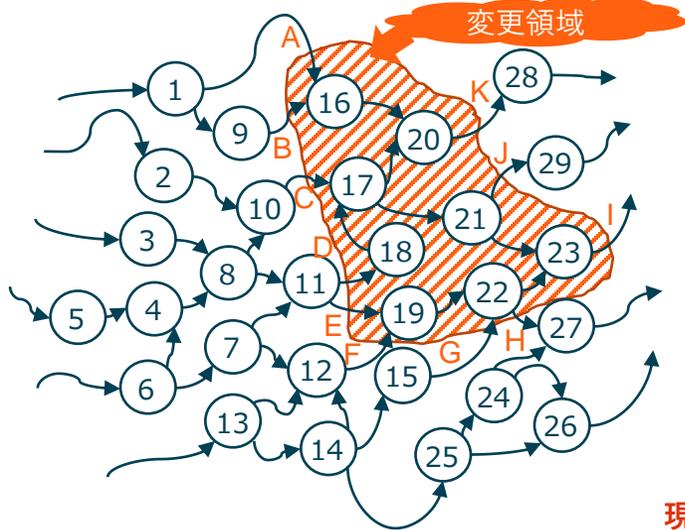
想定した将来の状態を達成するために変革できるようにすること

現在の業務の分析・可視化を十分に行い、意図した将来への移行に責任を持つ

影響を受ける人々が、現在の状態からプロジェクトの成果によって生み出される意図された将来の状態に移行するために必要な、新しく異なる行動やプロセスを採用し、維持できるように準備する。

Prepare those impacted for the adoption and sustainment of new and different behaviors and processes required for the transition from the current state to the intended future state created by the project outcomes.

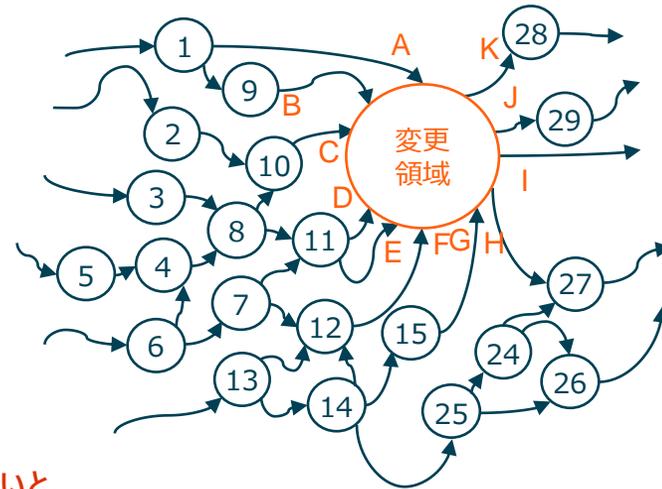
- システムの導入は必ずBPRである — 現在の業務の分析状況を見ればプロジェクトの成否が分かる



現行システムの論理モデル

- ✓ 変更領域のプロセスの再設計
- ✓ 変更領域の現行システムデータの将来システムへの移行
- ✓ 変更領域との接続点A~Kのプロセスの再設計

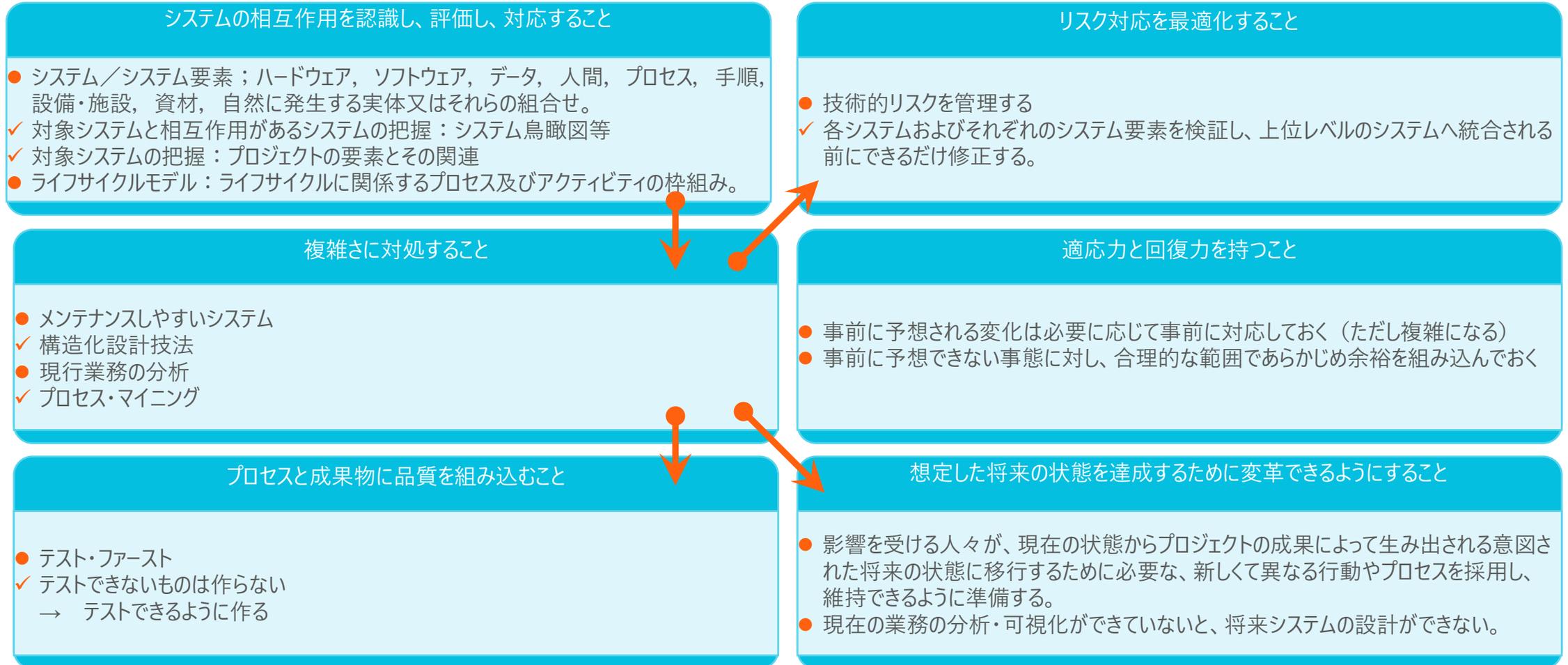
現在の業務の分析・可視化ができていないと、
将来システムの変更領域の設計はできない



将来システムの論理モデル

原則の実践

システムの相互作用を認識し、複雑さに対処することが重要ポイント



おわりに

複雑さの管理を手の内に納める

- ✓ このアンケートで2016年から算出しているNPS、ずっとマイナスです・・・
 - ✓ アジャイルの経験者は、回答者の半分を超え、緩やかな普及は続いています、推奨者は増えていません
 - ✓ 「はじめてのアジャイル」のNPSが低いのは、アジャイルが「成功体験に繋がっていない」可能性があります
 - ✓ 成果達成度は、アジャイル適用業務／非アジャイル開発に若干差異はありますが、大きくはありません
- ✓ PMBOK®ガイドの原則が「実践できていた」方のNPSを測定したところ、多くの方が推奨者でした
 - ✓ 推奨者の比率が特に高かったのは「非常に良く実践できていた」人が少なかった原則です
 - ✓ そこで、「非常に良く実践できていた」人が少なかった原則を選び、その実践を提言のテーマに選定しました
- ✓ ソフトウェア開発に「銀の弾」が存在しないのは、取り扱う領域が複雑系だからです
 - ✓ 「非常に良く実践できていた」人が少なかった原則は、複雑さの管理に関連するものが中心でした
 - ✓ 複雑さを管理する技術は、構造化分析等、昭和の時代から取り組まれていました
 - ✓ テスト駆動開発など、複雑さを管理する高度な技術は、ソフトウェア開発の根源的性質に対応するものです
 - ✓ 実践の提言が、皆さまが複雑さの管理を手の内に納めるヒントになれば幸いです

END



この資料の内容は発表者個人の見解です。
発表者の所属会社・組織等とは関係ありません。

PMI日本支部 アジャイル研究会からの お知らせ

アジャイル研究会の仲間になりませんか？

グローバルでプロジェクトマネジメントをリードするPMIおよびPMI日本支部の立場で、アジャイルの普及、啓発を行っています。そのため、アジャイルの最も基本となるアジャイルマニフェスト（アジャイルの価値と原則）の考えに基づき、PMBOK®ガイドとアジャイルの関連性を重視し、議論や情報収集を行い研鑽しています。また、外部関連団体とも連携を図り、PMI内外に対して積極的に情報発信も行っていきます。

- 定例会での意見交換・議論(毎月第三木曜日19:00~21:00にZoom開催)
 - とにかくアジャイルについて語り合おう
 - アジャイルに関する意識調査（2015年より継続実施中）
 - 明治大学との共同研究PMIフォーラムでの発表
 - PMI日本支部内でのアジャイルに関する連携
 - 外部団体との連携（IPA、JISA等）
 - アジャイルに関するオンライン勉強会を計画中
- 応募資格 PMI日本支部会員で、年間を通じて積極的に活動いただける方。

詳しくは
こちらへ



成田 和弘 : Kazuhiro Narita



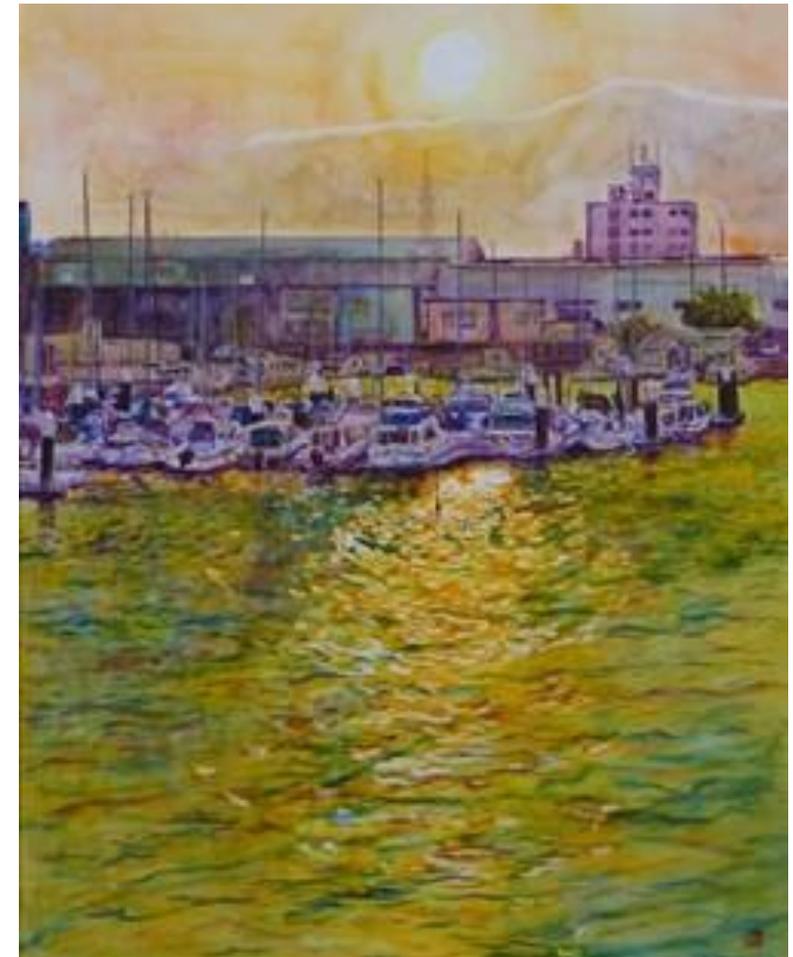
勘定系開発、IT投資戦略、基盤開発、システム統合から監査まで35年間の銀行員生活を経て、現在は有限責任監査法人トーマツに勤務
アジャイル研究会、プロジェクトマネジメント研究会、ビジネスアナリシス研究会

他団体 ; システム監査学会 (理事)、日本クラウドセキュリティアライアンス(運営委員)、ISACA東京支部(運営委員) など

(R)研究論文) トヨタ生産方式のシステム開発および監査への実装 (システム監査 第32巻第1号) (<https://www.sysaudit.gr.jp/gakkaishi/ronbun/201906/narita.pdf>) 他

趣味 ; 絵画 (2021年, 2018年日本水彩展入選、2021年, 2019年千葉県展入選)、 「からすみ」 「梅干し」 作り

資格 ; CIA、CISA、情報処理技術者試験 (システム監査技術者、システムアナリスト、プロジェクトマネージャー、情報セキュリティアドミニストレータ、ITストラテジスト他)



夕陽 2021千葉県展



本資料の著作権について

本資料の著作権は、PMI日本支部に帰属しています。
本資料および動画の一部または全部を著作権者に無許可で複製、転載、公衆送信、口述、上映、出版、頒布、貸与、編集するなどして使用することは著作権法に反することとなります。