

ソフトウェア・メトリクス調査2025【ガイドブック】

2025年4月10日

1. はじめに

「ソフトウェア・メトリクス」は、情報システムに関するQCD「品質(Quality)・費用(Cost)・工期(Delivery/Time)」とユーザー満足度(User Satisfaction)を評価する評価項目と基準値について、情報システムの開発・保守に取り組んできた企業のアンケート回答に基づいてまとめたものである。

その内容は過去の多くのプロジェクトを参考に自社プロジェクトを進めるに際しての指標となるものであり、開発生産性の向上を図り、プロジェクトの失敗を避けるための指針として活用していただければ幸いである。

なお、「ソフトウェア・メトリクス調査2025」は、従来の調査事項に対して、ユーザー企業を中心に活用状況についてのアンケート調査を行い、主に活用されているメトリクスに絞り込んでアンケートを実施し、分析・考察・活用の仕方をまとめたガイドブックとしている。

2. 調査対象プロジェクト数と開発区分

- 調査対象プロジェクト数(件)

	2025 単年	2018-2020累積	累積
開発プロジェクト(件)	145	221	366

- 開発フェーズ区分(2018から継続)

2018～ 3区分	要件定義	設計～結合テスト	総合テスト
-----------	------	----------	-------

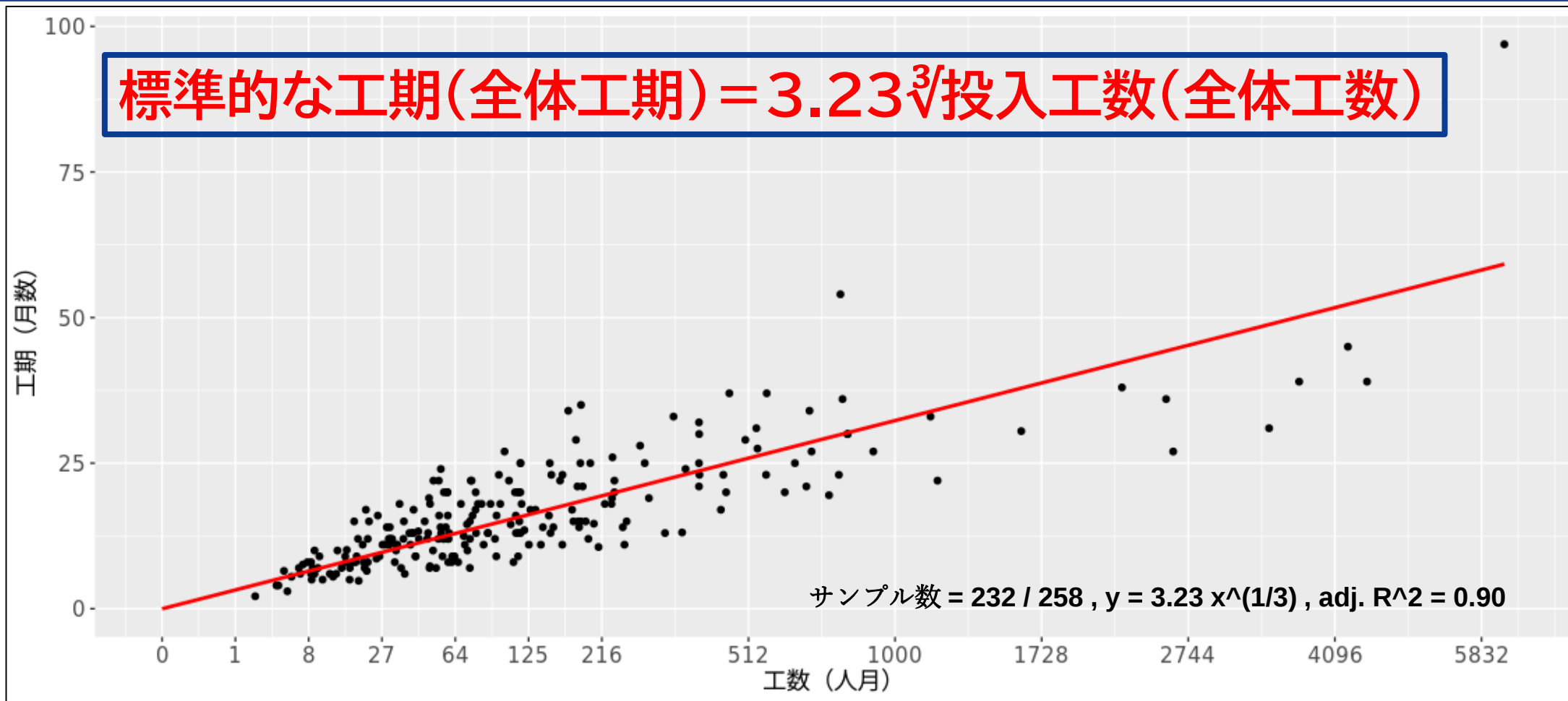
- 以下、図表番号は、「ソフトウェア・メトリクス調査2025【ガイドブック】」に対応。

3. ソフトウェア・メトリクス

- (1) システム開発の全体工数と全体工期の関係
- (2) 開発フェーズ別の工数と工期
- (3) プロジェクトの実績工期と標準工期
- (4) 工期遅延度への影響
- (5) 予定(計画)工期
- (6) 品質の評価
- (7) 換算欠陥数、換算欠陥率
- (8) 開発総費用
- (9) 工数区分別 平均単価

3-1.システム開発の全体工数と全体工期の関係

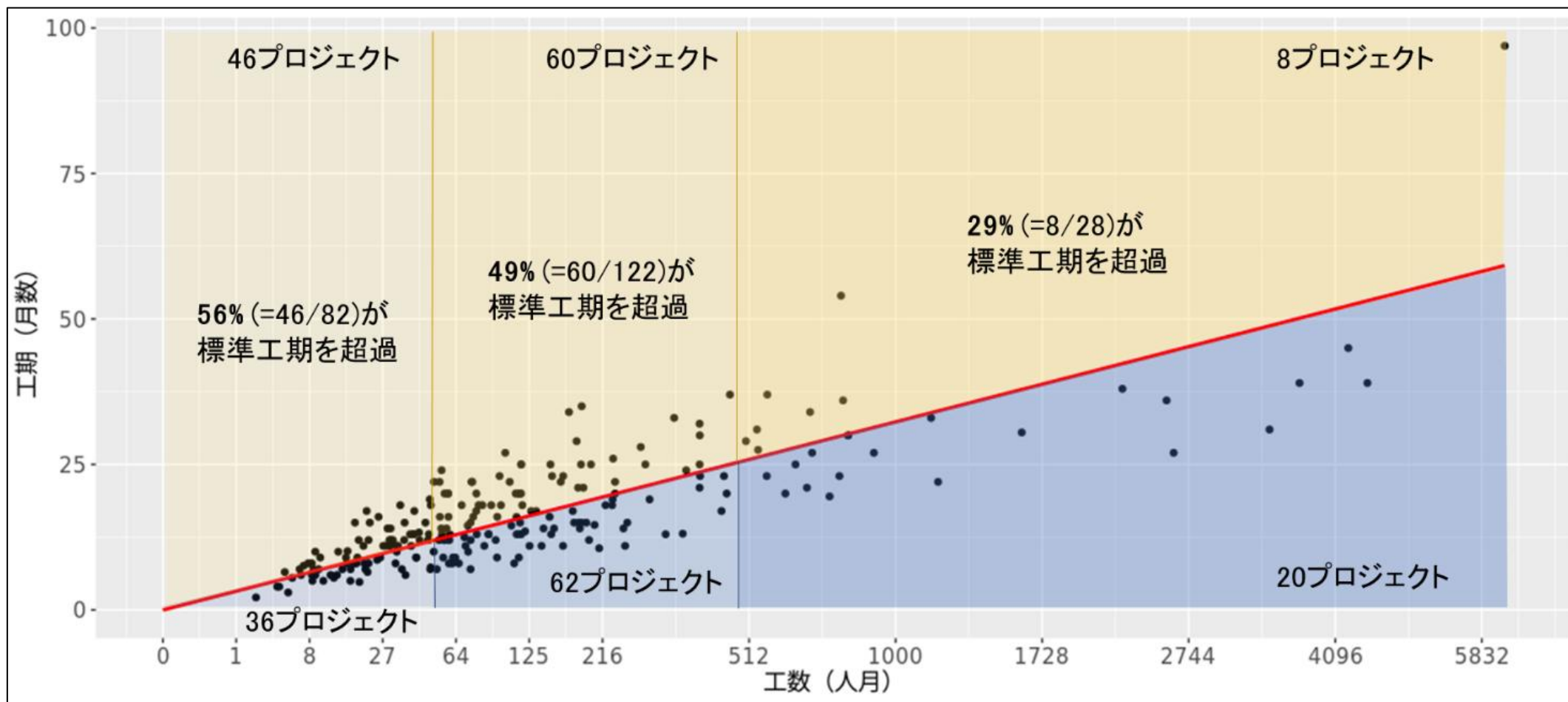
図表-2-1-1



- ◆ 自由度調整済み決定係数が0.9であることから、本調査の90%が上記メトリクスで説明できる。
- ◆ 例:125人月のプロジェクトは、工数の立方根が5となるので、係数3.23を積算し、16.15か月が標準的な工期(全体工期)となる。
- ◆ ある工期で実施できるプロジェクトの規模(工数)の目安を得ることができる。

3-1-1.開発規模別の実績工期が標準工期を超過する割合

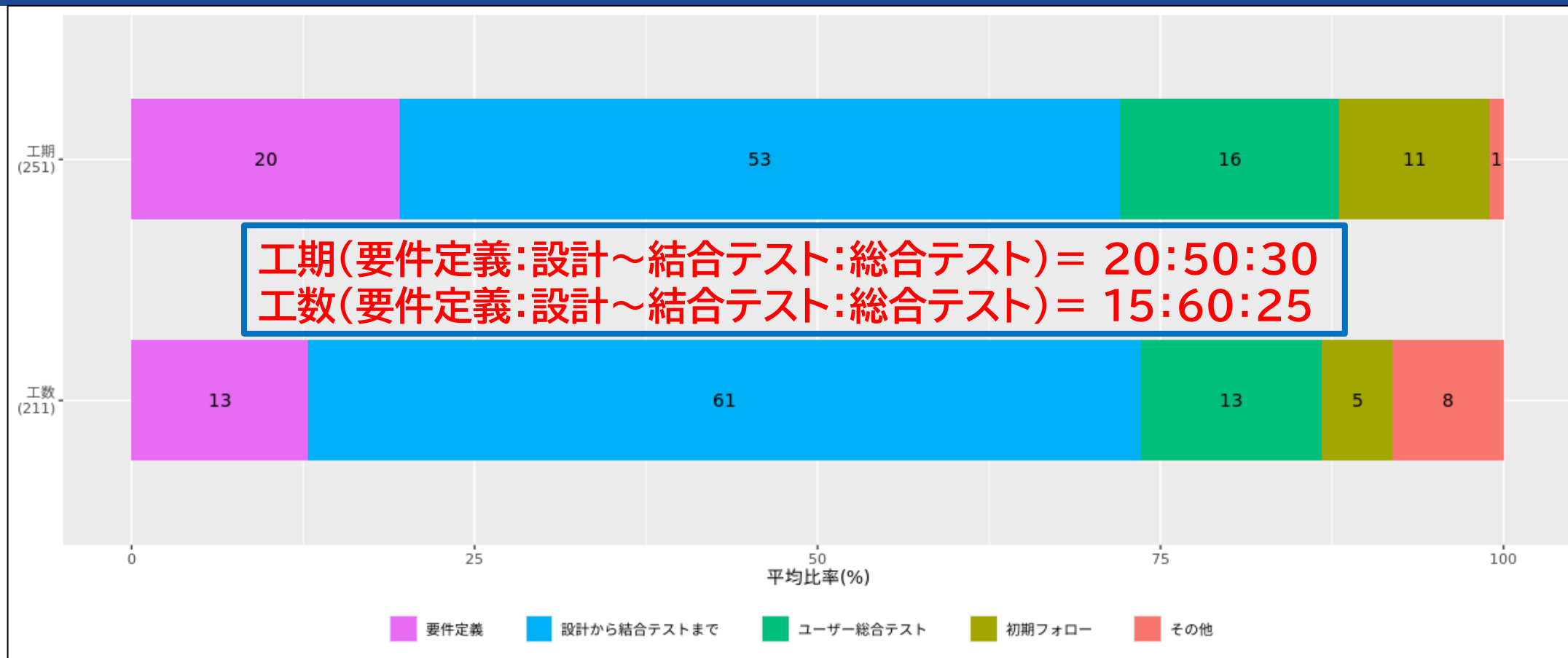
図表-2-1-3



- ◆ 「実績工期:500人月以上のプロジェクト＝標準工期を下回る」傾向(71%)にあるのは、分担できる作業が多いことから、投入人員を増やすことによって短縮できる作業の割合が工期に影響すると考えられる。
- ◆ 「実績工期:50人月未満のプロジェクト＝標準工期を上回る」(56%)にあるため、標準工期に対して余裕を持って工期を計画することが望ましい。

3-2.開発フェーズ別の工数と工期

図表-2-2-1



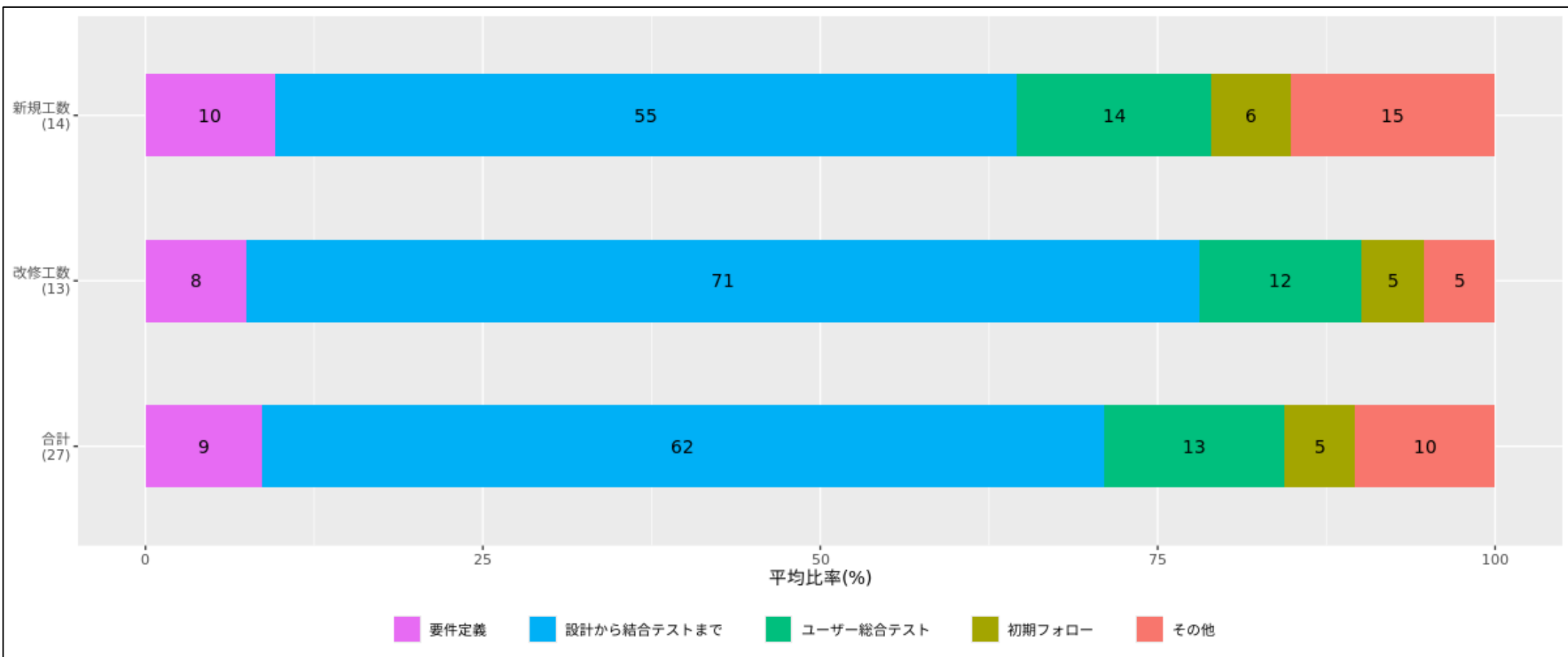
- ◆ 工期、工数の比率に用いた開発工程とアンケート回答の開発工程との対比は、以下の通り。
要件定義:設計～結合テスト:総合テスト(ユーザー総合テスト、初期フォロー、その他)

3-2.開発フェーズ別の工数と工期

- ◆ 要件定義: 工期比率(20%)>工数比率(13%)。要件定義は他の工程と比較して、作業を分担して進めることが適さないために工期比率が大きくなっているものと考えられる。
- ◆ 設計から結合テスト: 工期比率(53%)<工数比率(61%)。設計から結合テストまでの工程は他の工程と比較して、作業の分割が行いやすく作業分担によって工期比率が小さくなっているものと考えられる。
- ◆ 総合テスト: 工期比率(28%) 工数比率(26%)と近い値。総合テストの内訳を比較するとそれぞれに特徴がある。
 - ユーザー総合テスト: 工期比率(16%) 工数比率(13%)と近い値。ユーザー総合テストは、多くのメンバーに参加してもらいシステム内容を理解してもらう目的で行い、稼働後早期にプロジェクトの目標を達成することが肝要である。
 - 初期フォロー: 工期比率(11%)>工数比率(5%) 約2倍となっている。初期フォロー時は、対応事項が予め特定されていないことが多いため、対応方針の検討・策定の期間が必要となるため、工期比率が大きくなっているものと考えられる。
 - その他フェーズ: 工期比率(1%)<工数比率(8%)。その他フェーズに含まれる明確に分類できない開発業務は、アドホックに認識された課題の解決に費やされ、随時対応するために工数を要するが、工期に与える影響は小さいと考えられる。

3-2-1. 500人月以上の新規／改修プロジェクト別の工数比率

図表 2-2-2

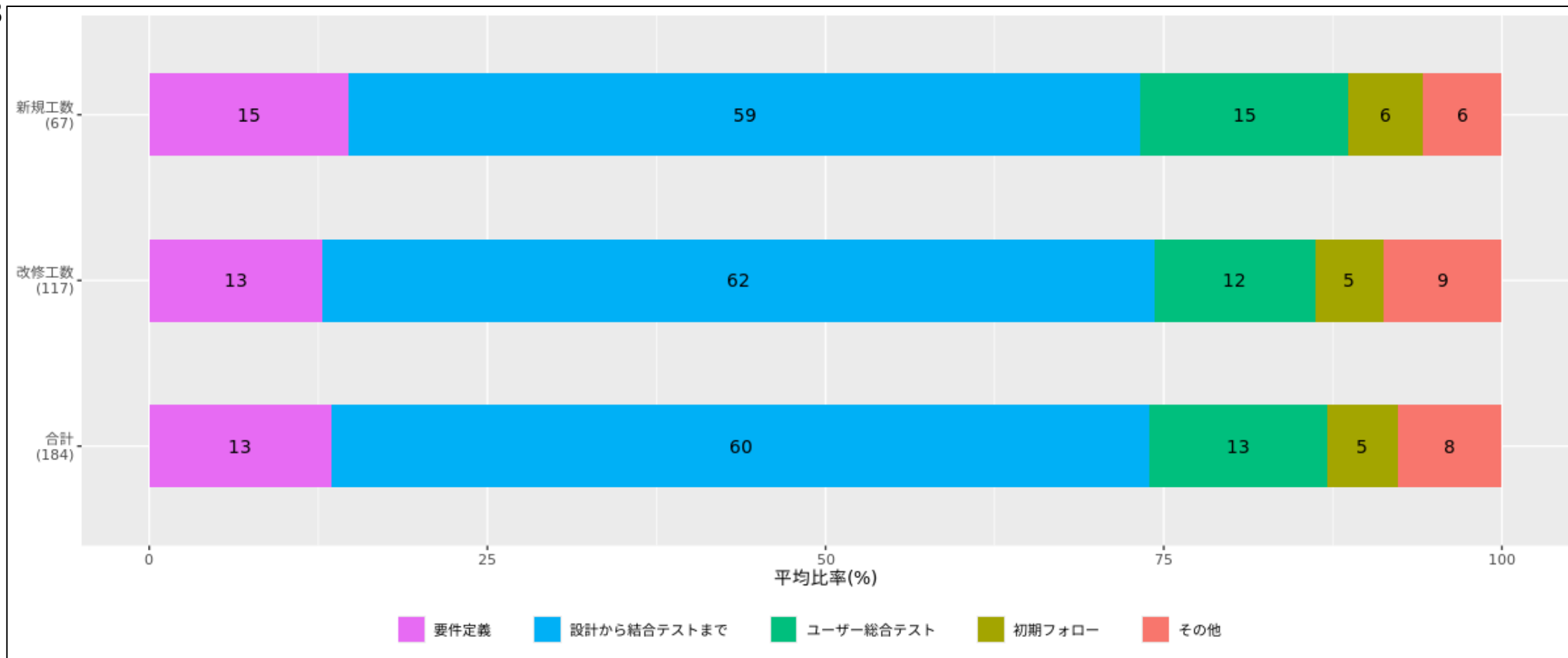


3-2-1. 500人月以上の新規／改修プロジェクト別の工数比率

- ◆ 設計から結合テスト: 新規工数比率(55%) < 改修工数比率(71%)。
 - 500人月以上の改修プロジェクトは、要件定義・ユーザー総合テスト・初期フロー・その他の工程における業務が定めやすく、設計から総合テストまでに工数を集中させやすいと考えられる。
 - 500人月以上の新規プロジェクトは、開発環境やドキュメント整備にかかる工数が改修プロジェクトに対して大きくなることから、設計から結合テストまでの新規工数比率が改修工数比率よりも小さくなると考えられる。
- ◆ その他: 新規工数比率(15%) > 改修工数比率(5%)。新規プロジェクトは、事前に定めきれない業務が多く発生していると考えられる。

3-2-2. 500人月未満の新規／改修プロジェクト別の工数比率

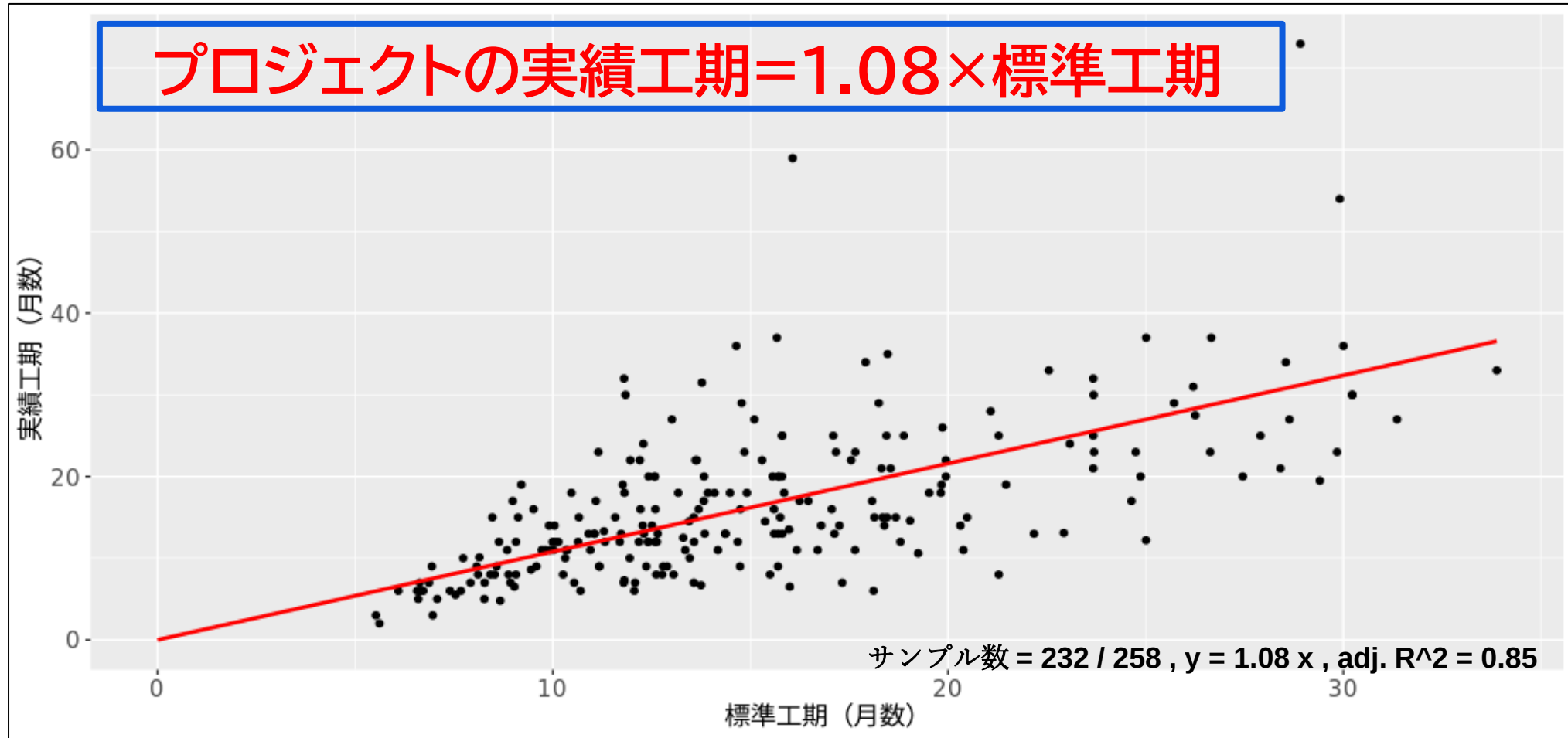
図表 2-2-3



- ◆ 500人月未満のプロジェクトでは、新規と改修の工程比率の差異が小さい。500人月未満のプロジェクトでは、新規・改修に関わらず同様な業務と業務量が求められているためと考えられる。

3-3.プロジェクトの実績工期と標準工期

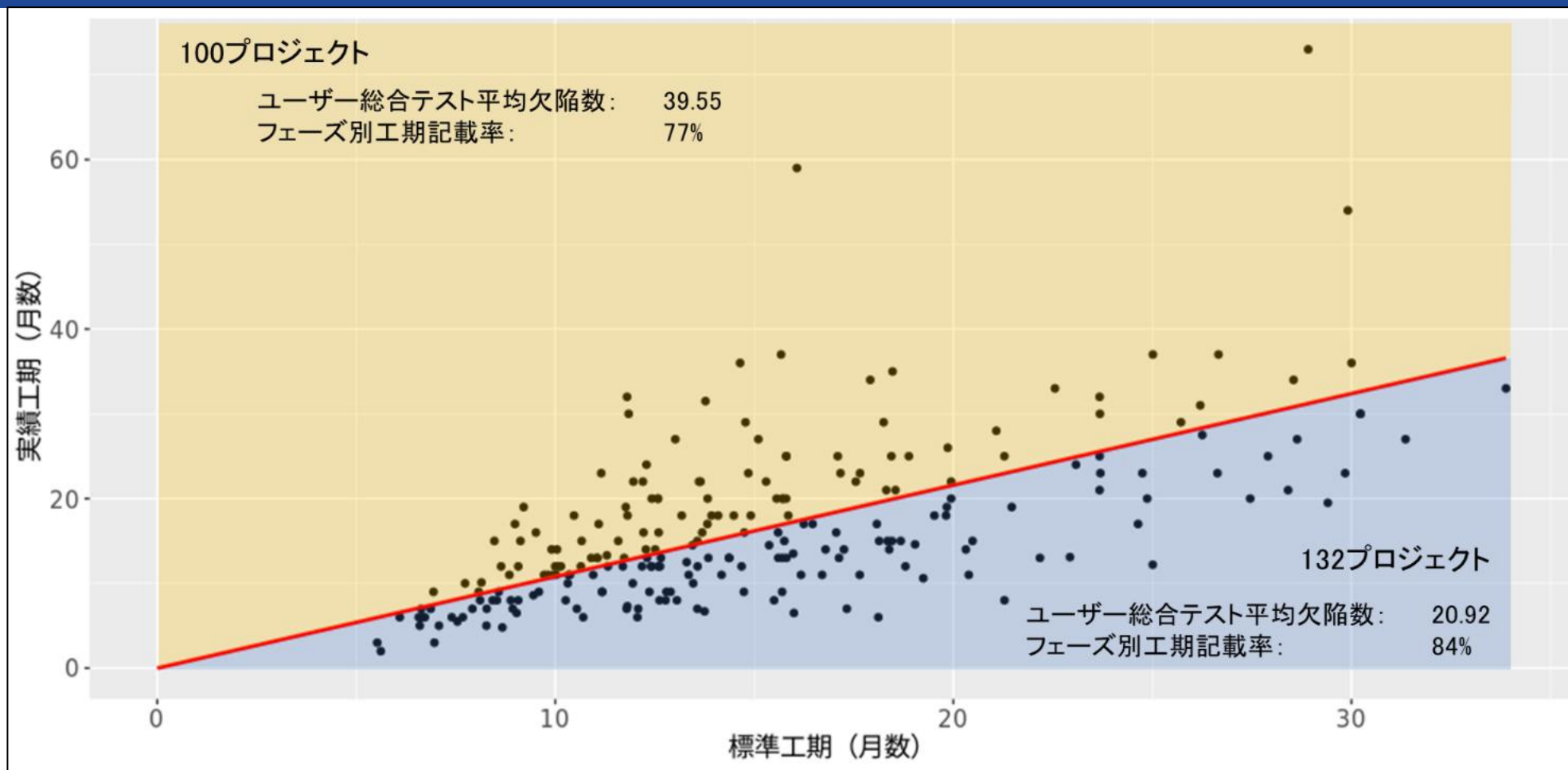
図表 2-3-1



- ◆ 自由度調整済み決定係数が0.85であることから、本調査の85%が上記メトリクスで説明できる。
- ◆ プロジェクトの実績工数が標準工期の1.08倍となっていることから、標準工期から1割程度余裕を持って計画を立案することが安全であると考えられる。

3-3-1.実績工期が標準工期を超えるプロジェクトと標準工期以下のプロジェクトの特徴

図表 2-3-2

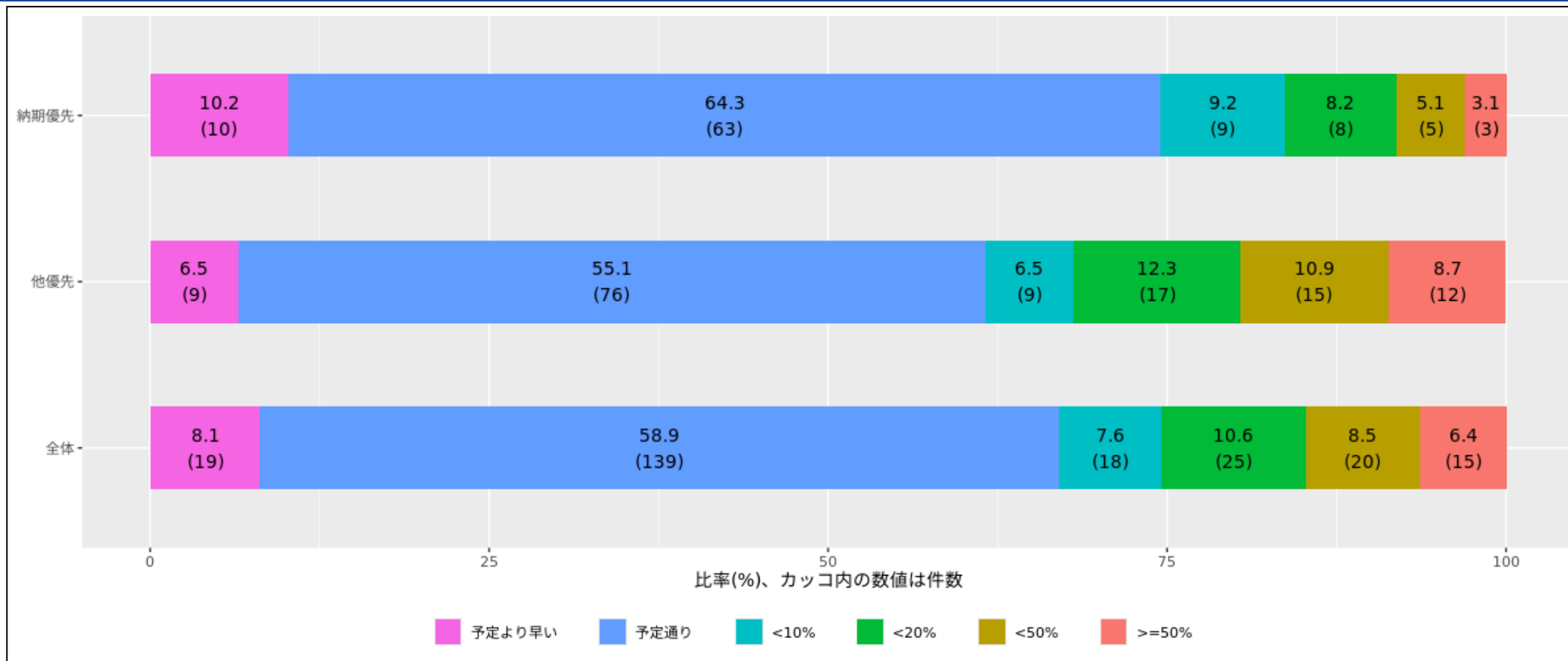


- ユーザー総合テスト(平均欠陥数):実績工期が 標準工期以下(20.92) < 標準工期を超過(39.55)。ユーザー総合テストの欠陥数を抑制することで、標準工期以下に実績工期を収められると考えられる。
- フェーズ別工期を管理しているプロジェクトの方が、標準工期以下に実績工期を抑制できる傾向があると考えられる。

3-4.工期遅延度への影響

納期優先したプロジェクトは、
予定工期で完了する傾向にある。

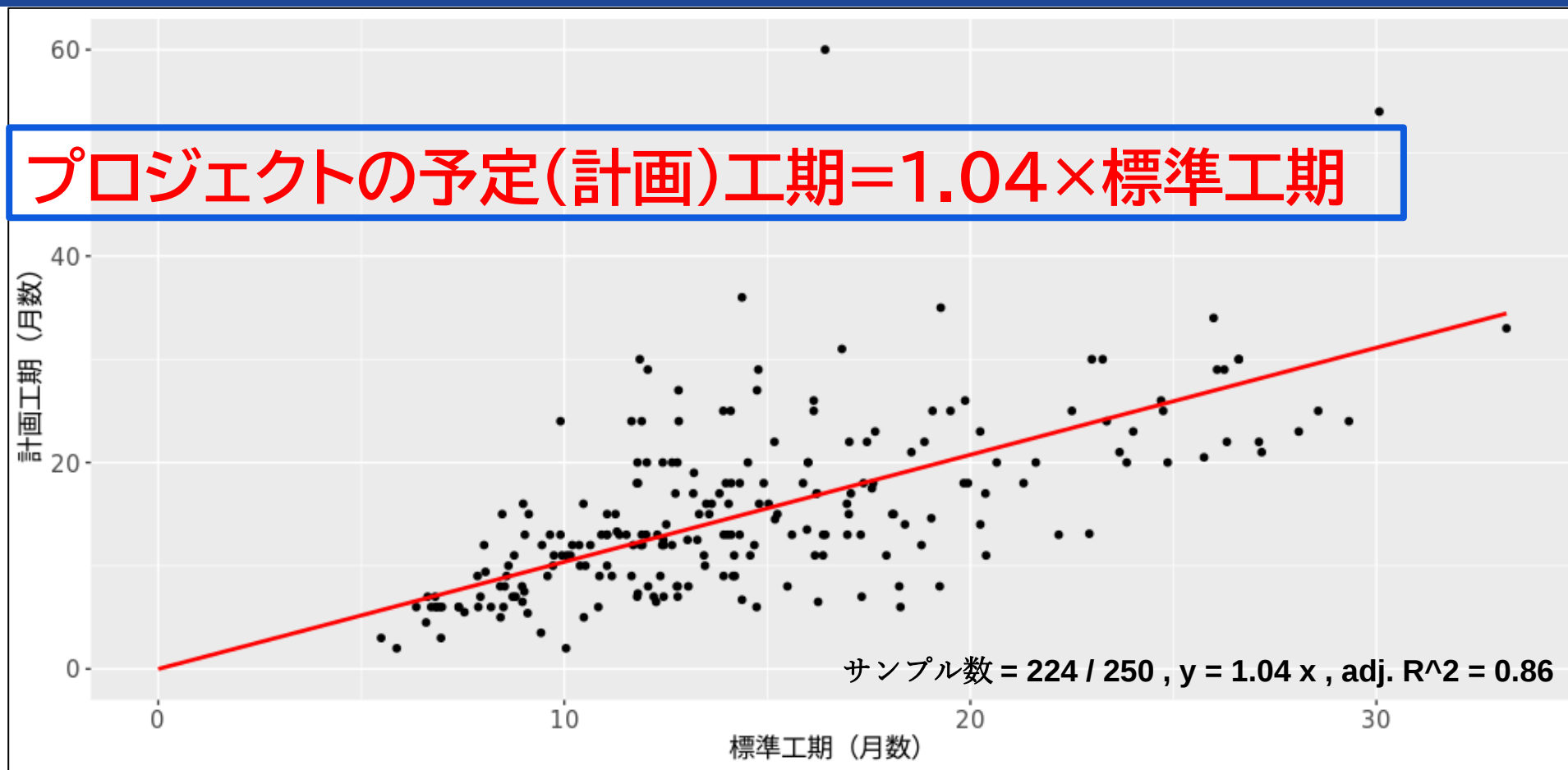
図表 2-4-1



- ◆ プロジェクトにおいて納期を優先することと、工期遅延との関係性を定量的に分析した。ここで、工期の遅延度は、(実績工期／計画工期)－1 と定義。
- ◆ 納期を優先したプロジェクトは、予定工期までに完了(74.5%)することが多い。
- ◆ 納期を優先したプロジェクトが、20%以上の遅延度になることは少ない(8.2%)。他の事項を優先したプロジェクトが、20%以上の遅延度は19.6%となった。

3-5. 予定(計画)工期

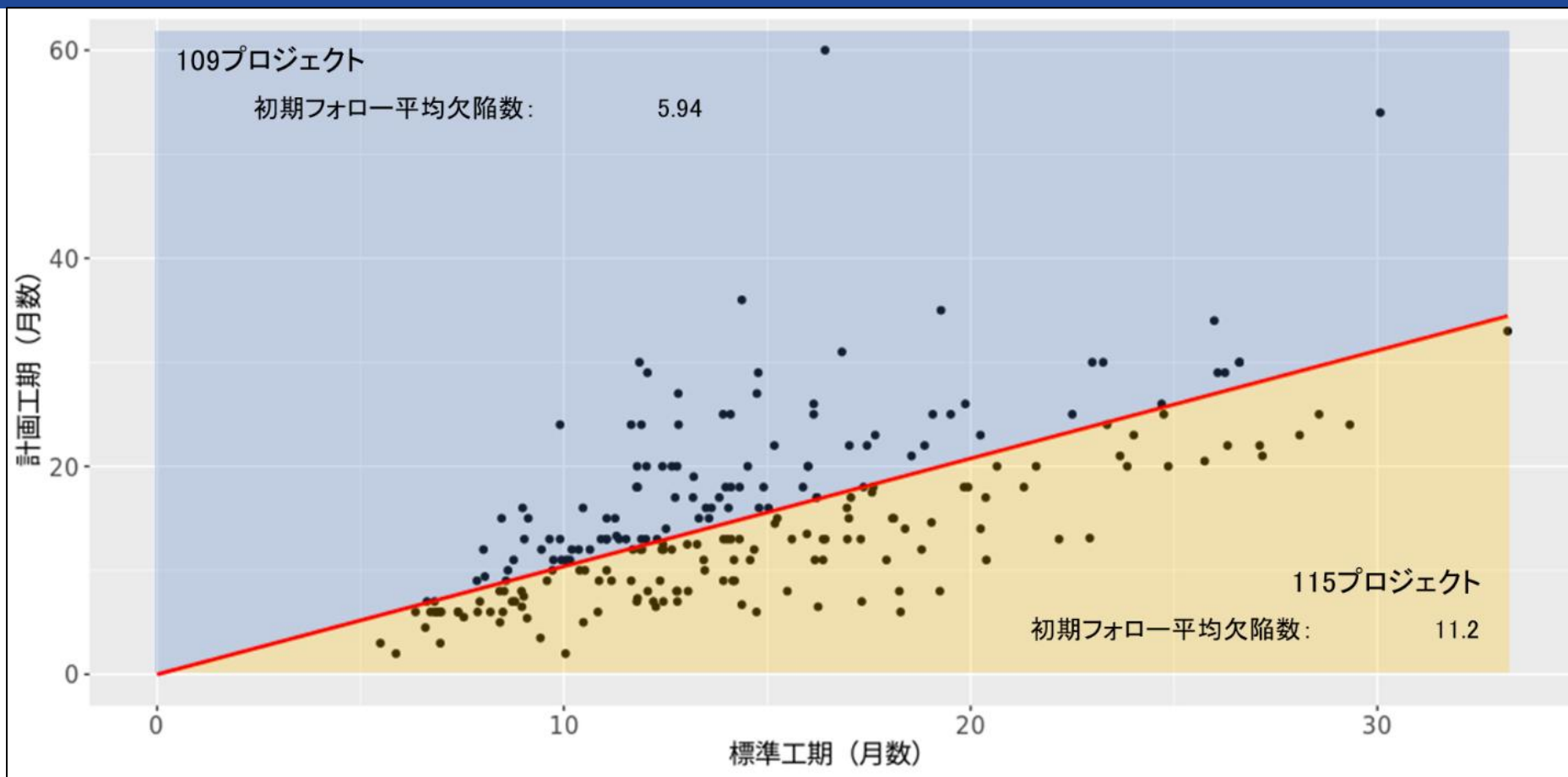
図表 2-5-1



- ◆ プロジェクトの計画工期は、標準工期の1.04倍とほぼ標準工期に近い。
- ◆ プロジェクト予定(計画)工期遵守を優先する場合に注意すべきこととして以下の点が考えられる。
「納期優先の意識付け」、「要求仕様の明確化と仕様変更の最小化などの納期遵守の工夫」、
「標準工期より余裕をもった計画工期を設定する」

3-5-1. 計画工期が標準工期を超えるプロジェクトと標準工期以下のプロジェクトの特徴

図表 2-5-2



- ◆ **平均欠陥数(初期フォロー):** 計画工期が標準工期以上(5.94) < 計画工期が標準工期未満(11.2)
 - 計画工期が相対的に短いことによって、システム使用方法の検討が浅かったり、例外事象への配慮が不十分であったり、各業務項目の工期を過小に見積もったことによって負担が生じたりしていることで、欠陥数が相対高位となっていると考えられる。

3-6.品質の評価

- ◆ ユーザーが、ユーザー総合テストと初期フォローフェーズにおいて発見した不具合を欠陥としてカウントし、その影響度の大きさによって大中小(重度・中度・軽度)に分類した。

欠陥(重度) = システムにとって致命的で緊急対応が必要な障害であり、
5人日以上 の負荷を要する場合

欠陥(中度) = システムにとって致命的ではないが緊急対応が必要な障害
(大でも小でもない障害)で、その復旧に要する時間が中程度である。

欠陥(軽度) = 軽微で緊急対応の必要がない程度の障害、その復旧に要する時間は
半日以内で対応可能。

3-7.換算欠陥数、換算欠陥率

- ◆ 検出された欠陥を、次式にもとづいて「換算欠陥数」「換算欠陥率」として算出し指標化した。

換算欠陥数(重み付け欠陥数) = $2 \times (\text{欠陥数}_\text{大}) + 1 \times (\text{欠陥数}_\text{中}) + 0.5 \times (\text{欠陥数}_\text{小})$

換算欠陥率(重み付け欠陥率) = 換算欠陥数 ÷ 全体工数

- ・換算欠陥率によって、Aランク(換算欠陥率=0)～Fランク(換算欠陥率 ≥ 3)に分類している

以降、プロジェクト品質を換算欠陥率によって6段階のランクに分類し、分析した。

A ランク: 換算欠陥率 = 0

B ランク: $0.00 < \text{換算欠陥率} < 0.25$

C ランク: $0.25 \leq \text{換算欠陥率} < 0.50$

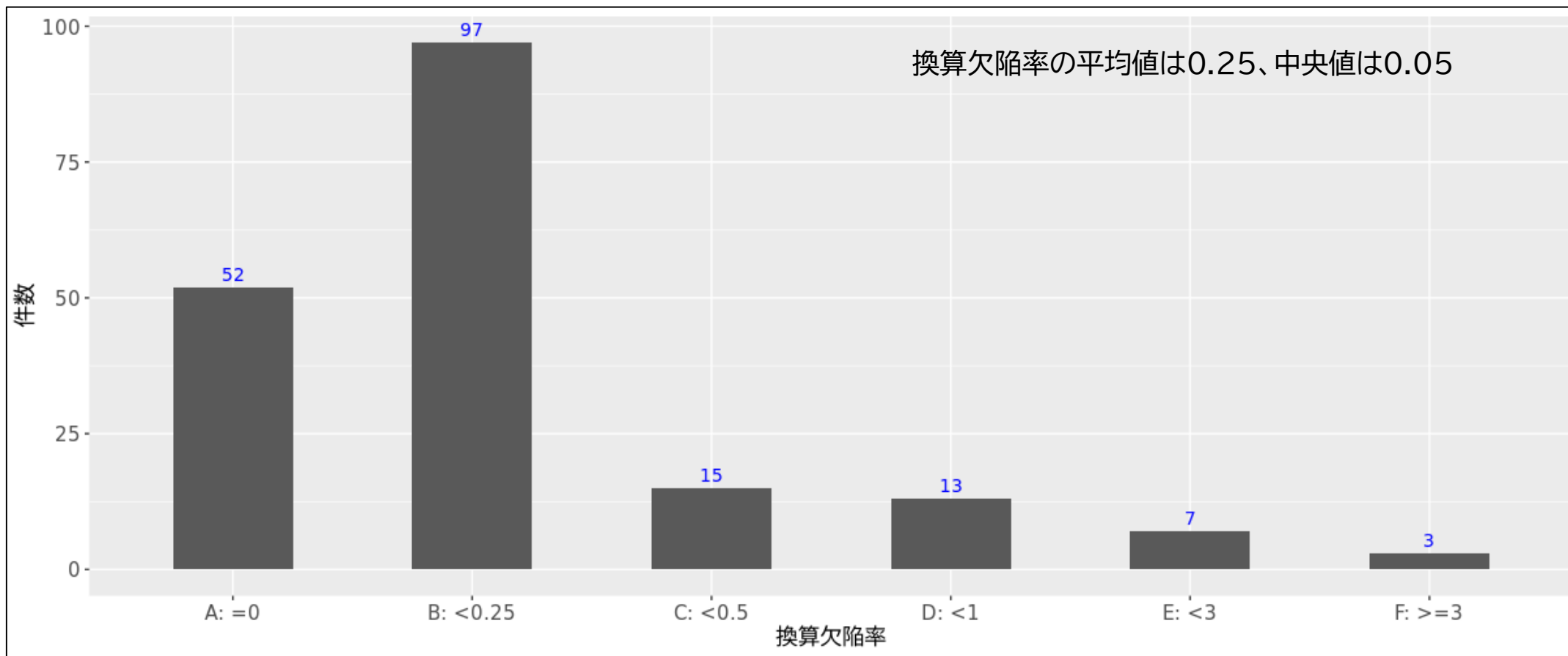
D ランク: $0.50 \leq \text{換算欠陥率} < 1.00$

E ランク: $1.00 \leq \text{換算欠陥率} < 3.00$

F ランク: $3.00 \leq \text{換算欠陥率}$

3-7-1.換算欠陥率のヒストグラム

図表 2-7-1

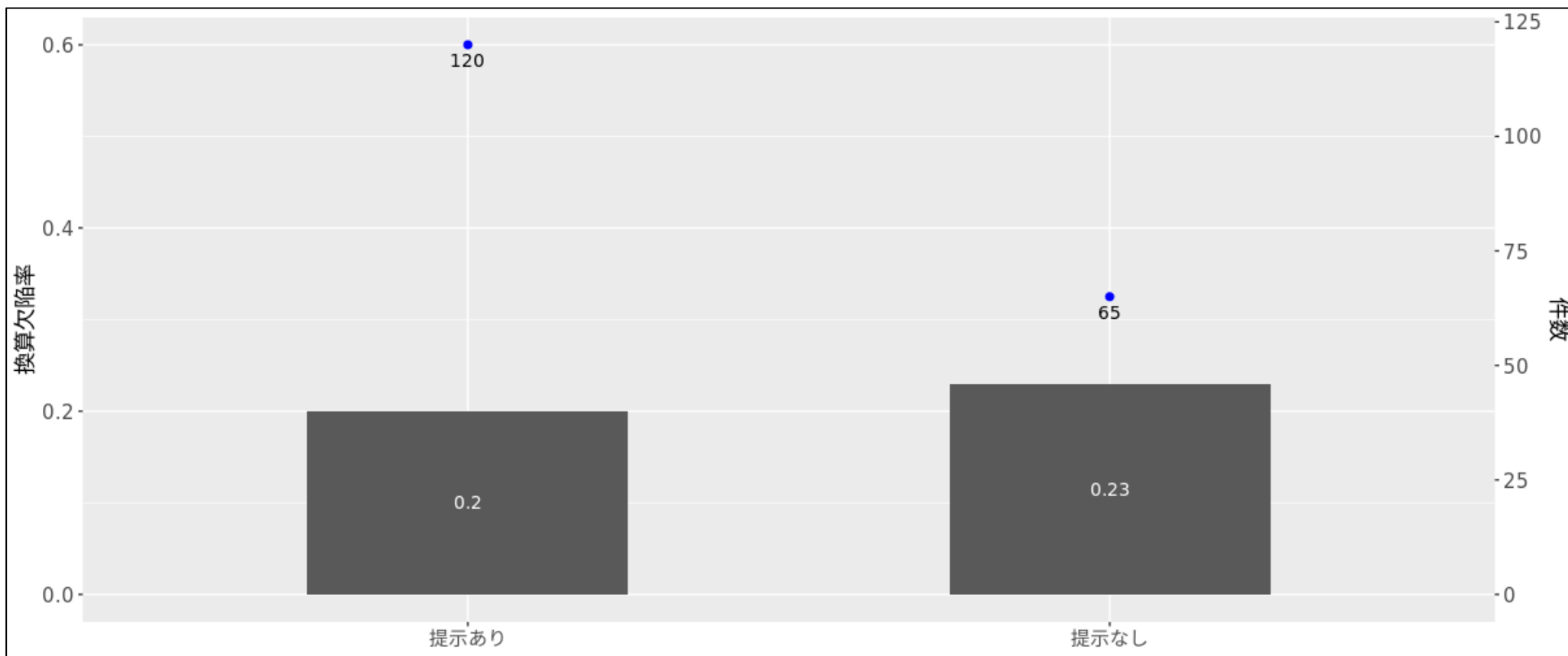


品質目標:換算欠陥率目標=換算欠陥数/開発工数=0.05 以下

- ◆ 換算欠陥率の平均値が0.25であることから、換算欠陥率の目標を0.25とすることも有益である。

3-7-2.品質目標の提示の有無による換算欠陥率の違い

図表 2-7-2



品質目標の提示は、換算欠陥率を減らす傾向がある

- ◆ プロジェクトの2/3は品質目標を提示しており、品質目標を提示しているプロジェクトの換算欠陥率(0.2)は、品質目標を提示していないプロジェクトの換算率(0.23)と比較して、抑制されている。

3-7-3.要求仕様の変更と換算欠陥率

仕様が不明確、大きな仕様変更が発生するほど品質は劣化する

図表 2-7-3

要求仕様変更	件数	平均換算欠陥率	最大換算欠陥率
変更なし	50	0.192	3.125
軽微な変更	97	0.213	3.404
大きな変更	33	0.481	4.576

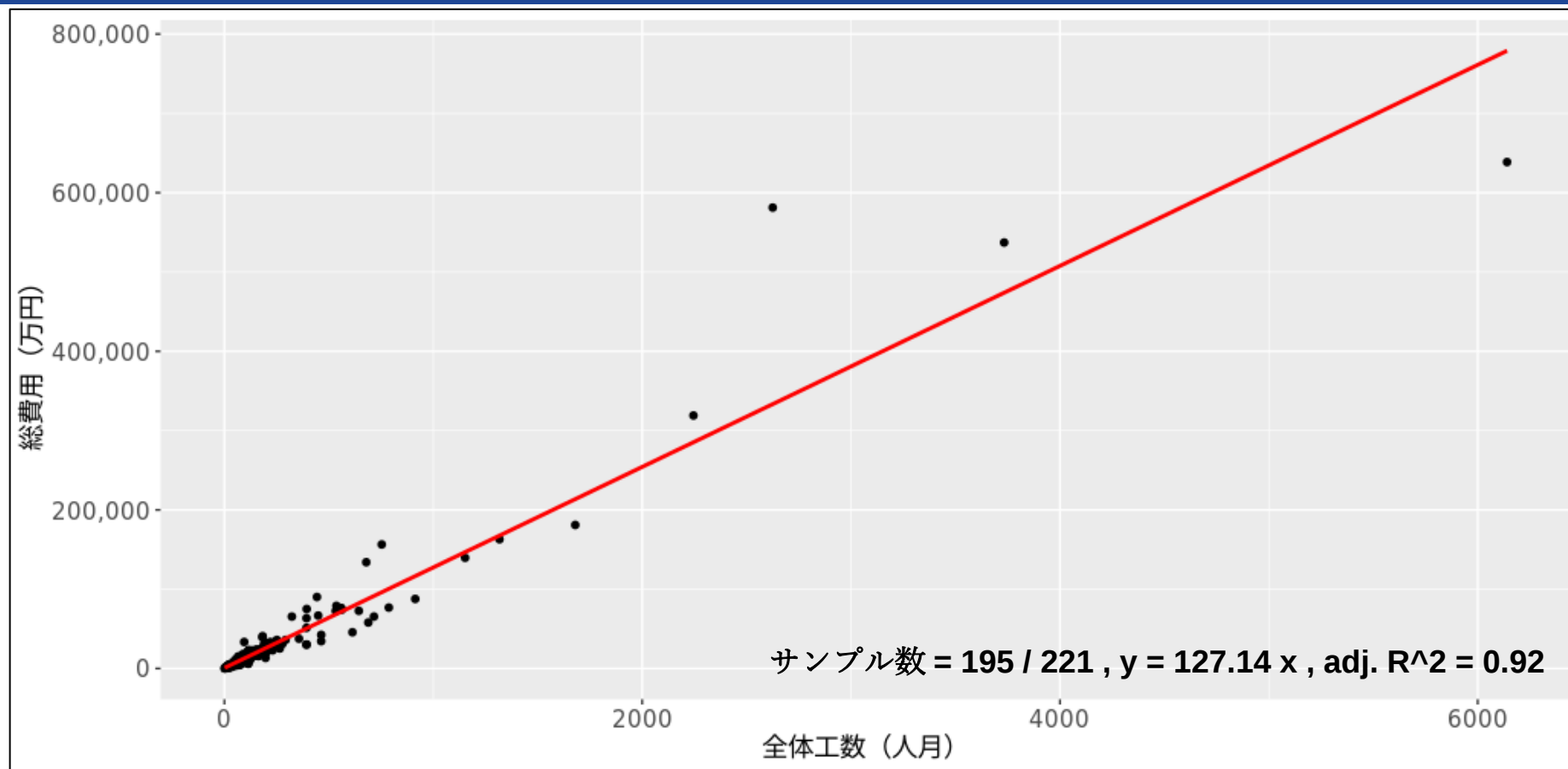
- ◆ 要求仕様変更の程度が大きくなるにつれて、平均および最大換算欠陥率が増加した。

3-8.開発総費用

$$y = 127.14 x$$

x : 実績工数(全体工数)、 y : 開発総費用(万円)

図表 2-8-1



- ◆ 自由度調整済み決定係数が0.92であるため、総費用の92%は上記メトリクスによって説明できる。
- ◆ 回帰係数が127.14であるため、分析対象データに対する人月単価は127万円が妥当である。
- ◆ パッケージの活用によって、外注コストにおける人月単価の違いが見られた。(詳細は後述)

3-9.工数区分別 平均単価

図表 2-9-1

スクラッチ開発プロジェクトの工数区分別の平均単価

工数区分	<10人月	<50人月	<100人月	<500人月	>=500人月	全体
件数	6	23	22	36	12	99
外注コスト平均	850	2951	7570	19381	146901	27273
工数平均	5	31	70	205	1542	284
加重平均単価	185	95	109	95	95	96

- ◆ 全体の加重平均単価は96万円であった。
- ◆ 開発規模が10人月を超えるプロジェクトでは、開発工数規模による単価の差異は小さかった。スクラッチ開発は広く用いられ、実績が蓄積されている開発手法であるため、プロジェクト規模による影響が小さいものと考えられる。

3-9.工数区分別 平均単価

図表 2-9-2

パッケージ利用開発プロジェクトの工数区分別の平均単価

工数区分	<10人月	<50人月	<100人月	<500人月	>=500人月	全体
件数	1	13	8	12	5	39
外注コスト平均	560	2890	10407	28844	166583	33344
工数平均	9	31	70	221	1089	232
加重平均単価	63	94	150	130	153	144

- ◆ パッケージ利用開発の全体加重平均単価(144万円)は、スクラッチ開発の全体加重平均単価(96万円)よりも高い結果となった。パッケージ利用開発は、パッケージ使用のスキル・ノウハウが必要とされるため、スクラッチ開発と比較して高単価となっていると考えられる。また、パッケージ開発技術者が需要に対して十分に供給できていない状況にあるとも考えられる。
- ◆ 開発規模が100人月以上のプロジェクトでは、100人月未満のプロジェクトよりも単価が高い結果となった。開発規模が大きくなることで、パッケージについてのさらに高度なノウハウが必要となるなどの理由によって、高単価人材が必要になったと考えられる。

3-9.工数区分別 平均単価

図表 2-9-3

SaaS利用開発プロジェクトの工数区分別の平均単価

工数区分	<10人月	<50人月	<100人月	<500人月	>=500人月	全体
件数	0	4	6	5	0	15
外注コスト平均		3715	8010	42832		18472
工数平均		40	67	273		128
加重平均単価		93	120	157		144

- ◆ 全体の加重平均単価は144万円で、パッケージ利用開発プロジェクトの全体の加重平均単価(144万円)に等しかった。
- ◆ SaaS利用開発の加重平均単価がスクラッチ開発に対して高位となり、パッケージ利用開発と同程度となっているのは、SaaS利用開発においてもSaaS使用のスキル・ノウハウが必要とされるためと考えられる。
- ◆ 尚、SaaS利用開発プロジェクトの工数と単価については、サンプル数が少ないため、参考情報としていただきたい。

4.活用事例インタビュー

ソフトウェア・メトリクスを活用していただいている企業事例をご紹介します。
詳細は、ガイドブックをご参照ください。

- ◆【事例1】大規模や未経験案件の事前リスク確認、
失敗プロジェクトの事後評価に利用
- ◆【事例2】経営層への説明に業界標準として利用
- ◆【事例3】親会社との共通言語として活用

5. おわりに

2020年ソフトウェア・メトリクス調査報告書の発行以来、4年間調査を中断していましたが、JUAS会員の皆さまにアンケート方式で報告書の利用状況を調査したところ、主なメトリクスについて活用されていることが確認できました。今回のガイドブックでは従来の多岐にわたる調査項目を見直し、活用度の高いメトリクス上位項目に絞り込んで調査を実施したことで、多くの有効回答数を得ることができました。調査方針見直しについてご理解いただくとともに、メトリクス調査にご協力を頂いた全ての皆さまに感謝を申し上げます。

2026年ソフトウェア・メトリクス調査では、今回の調査に加え、これらソフトウェア開発環境変化に伴い、生産性・品質についてどのように考えていけば良いのかについてのプロジェクトを発足させます。皆さまが抱える課題に協会として対応していけるよう、努めていきますので、引き続きご理解とご協力を賜りますよう、よろしくお願いします。最後に、本ガイドブックが皆さまのお役に少しでも立つことを願っております。

JUASソフトウェア・メトリクスのお問い合わせは、
下記にて承っております。

JUASソフトウェア・メトリクス事務局

〒104-0045 東京都中央区築地1丁目13－14 NBF東銀座スクエア 2階
E-mail: swm-juas@juas.or.jp

※ご質問はメールにてお願いします。