

プロジェクトデータから見える 組込み開発現場の実態

～「組込みソフトウェア開発データ白書2017」及び
最近の分析結果紹介～

独立行政法人 情報処理推進機構(IPA)
社会基盤センター 産業プラットフォーム部
研究員 松田 充弘

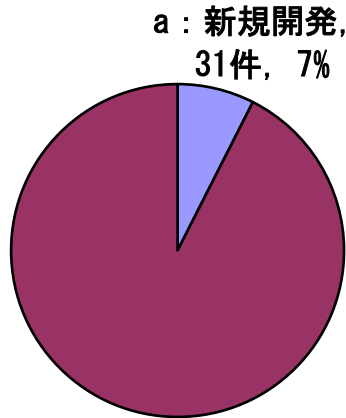


プレゼンテーション内容

1. 収集したプロジェクトデータのプロフィール
2. 組込みソフトウェア開発の生産性
3. 組込みソフトウェア開発の信頼性
4. 組込み製品の特性ごとの生産性・信頼性指標
5. 製品の特性ごとの分析
6. 「組込みソフトウェア開発データ白書2017」の特徴
7. 「組込みソフトウェア開発データ白書」編纂の目的
8. 「組込みソフトウェア開発データ白書」の活用

1. 収集したプロジェクトデータのプロフィール

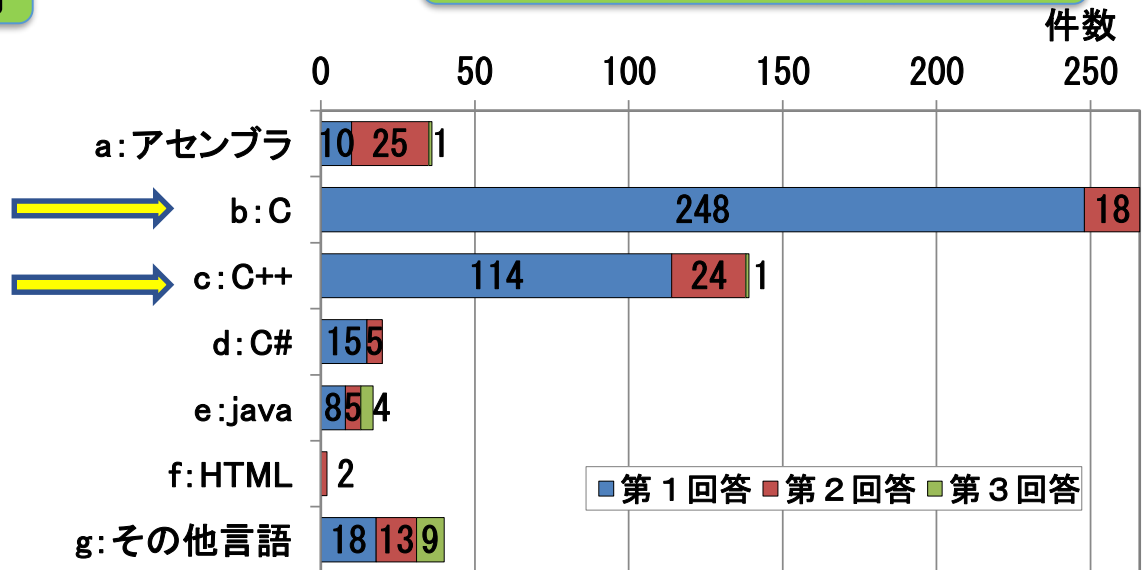
開発プロジェクトの種別



b: 改良(派生)開発,
384件, 93%

N = 415 (未回答: 31件)

開発言語



N = 413 (未回答: 3件)

プロジェクト種別：改良（派生）開発、
言語：C及びC++のプロジェクトデータ
を対象に分析

1. 収集したプロジェクトデータのプロフィール

収集データのSLOC規模

[単位:KSLOC]

	N	P25	中央値	P75
SLOC規模	415	2.05	6.22	26.05
SLOC規模 (母体含む)	414	28.3	132.2	390.1

収集データの実績工数

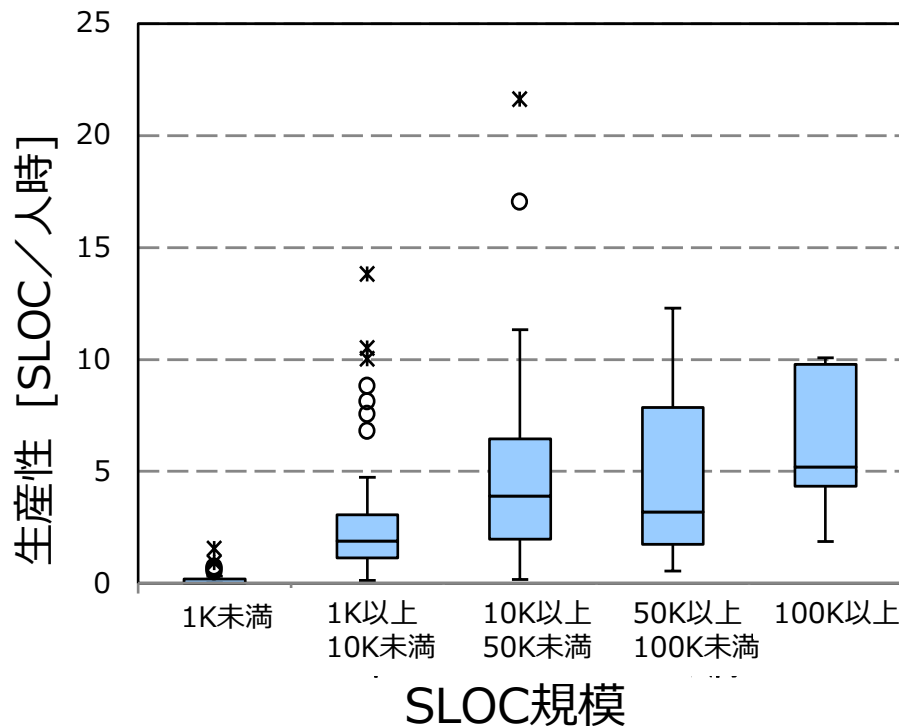
[単位:人時]

	N	P25	中央値	P75
実績工数	392	828	2098	6042

2. 組み込みソフトウェア開発の生産性

SLOC規模別の生産性

開発5工程 言語C/C++ (改良・派生開発)



・生産性は規模に大きくなると高くなる傾向が見えている。
 ・規模の小さいものは管理工数の比率が高くなるため、生産性が上がらないと考えられる。

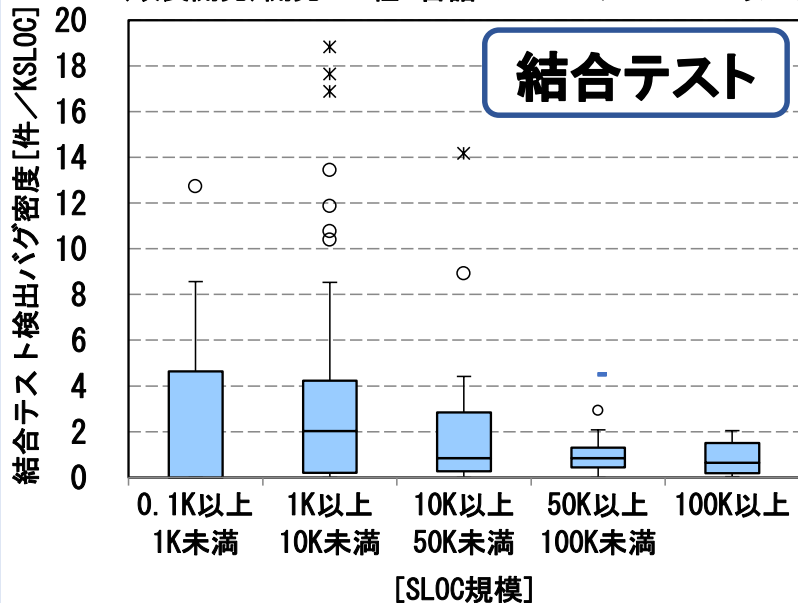
SLOC規模	N	P25	中央	P75
全体	130	0.54	1.86	4.35
1K未満	31	0.01	0.01	0.20
1K以上10K未満	48	1.13	1.90	3.06
10K以上50K未満	34	1.97	3.91	6.46
50K以上100K未満	10	1.75	3.18	7.86
100K以上	7	4.34	5.19	9.79

3. 組み込みソフトウェア開発の信頼性

SLOC規模別のテスト検出バグ密度

SLOC規模別結合テスト検出バグ密度

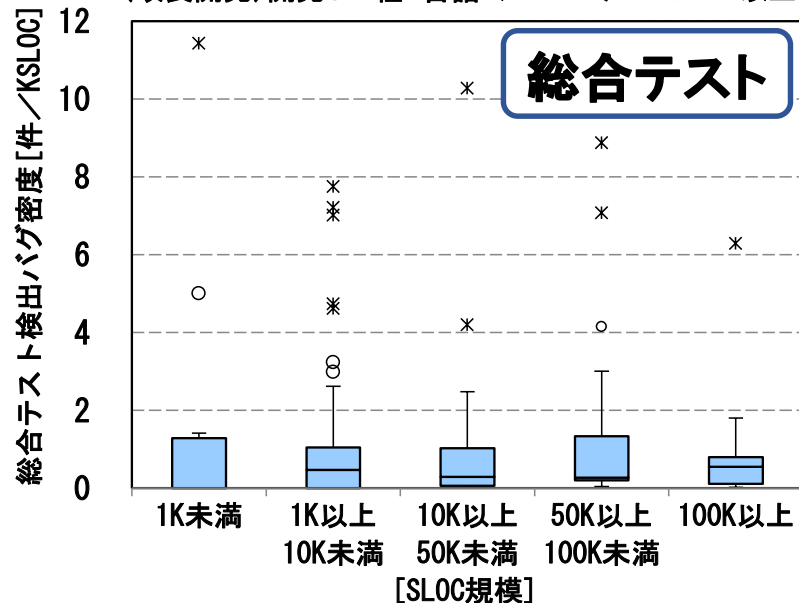
(改良開発)開発5工程 言語C/C++ <0.1KSLOC以上>



SLOC規模	N	P25	中央	P75
全体	168	0.140	1.120	3.100
0.1K以上1K未満	19	0.000	0.000	4.643
1K以上10K未満	73	0.193	2.020	4.237
10K以上50K未満	50	0.267	0.837	2.838
50K以上100K未満	16	0.438	0.841	1.308
100K以上	10	0.182	0.643	1.497

SLOC規模別総合テスト検出バグ密度

(改良開発)開発5工程 言語C/C++ <0.1KSLOC以上>



SLOC規模	N	P25	中央	P75
全体	147	0.027	0.331	1.072
1K未満	15	0.000	0.000	1.279
1K以上10K未満	59	0.000	0.465	1.047
10K以上50K未満	47	0.054	0.290	1.027
50K以上100K未満	17	0.199	0.262	1.331
100K以上	9	0.103	0.542	0.799

結合テストのバグ密度は、規模が小さいとバラつきが大きい傾向が有り指標として扱いづらい。
 総合テストのバグ密度は、規模との関係が見られない。テスト目的がバグ無を確認するための
 標本が多いと考えられる。外れ値の標本は結合テスト同様にデバッグ目的と考えられる。

製品ドメイン別の分類

⇒ 多種多様のため分析ドメインを網羅できない

組込み製品や制御システムの特性によって、生産性や信頼性の指標は違うはず



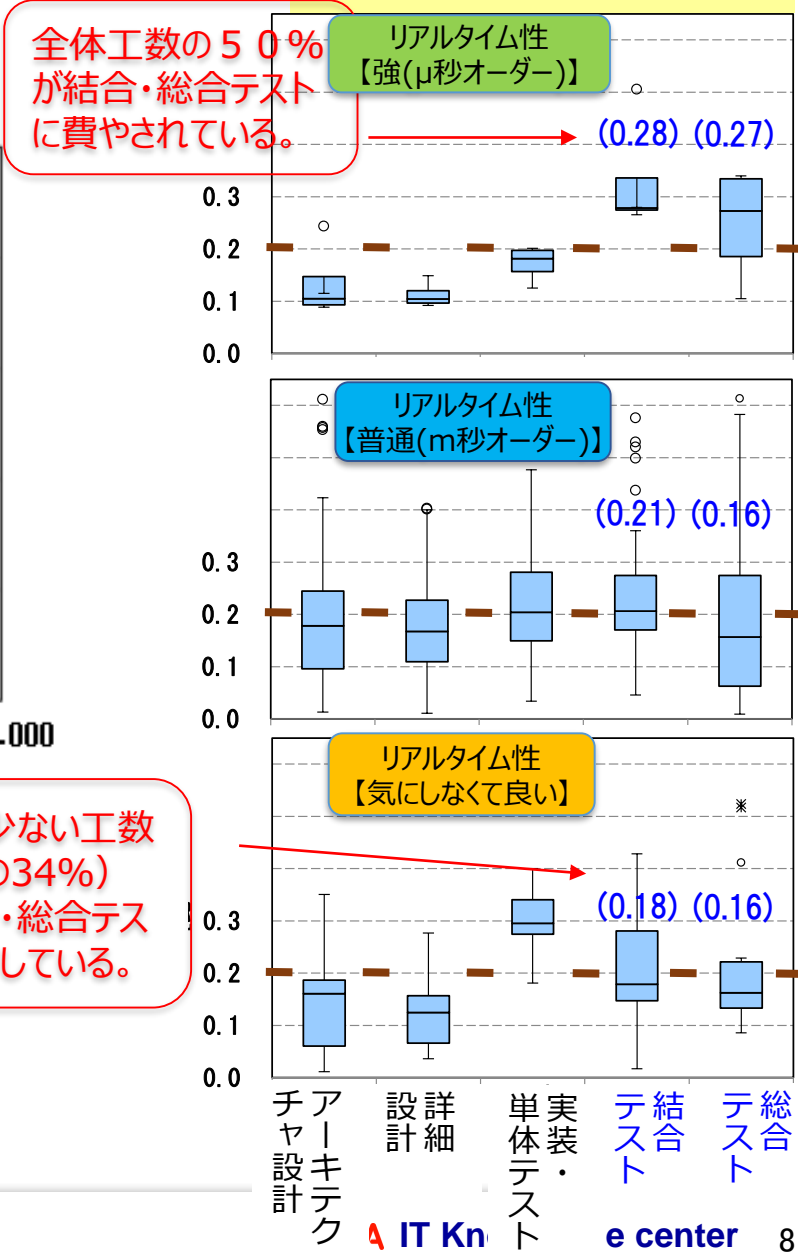
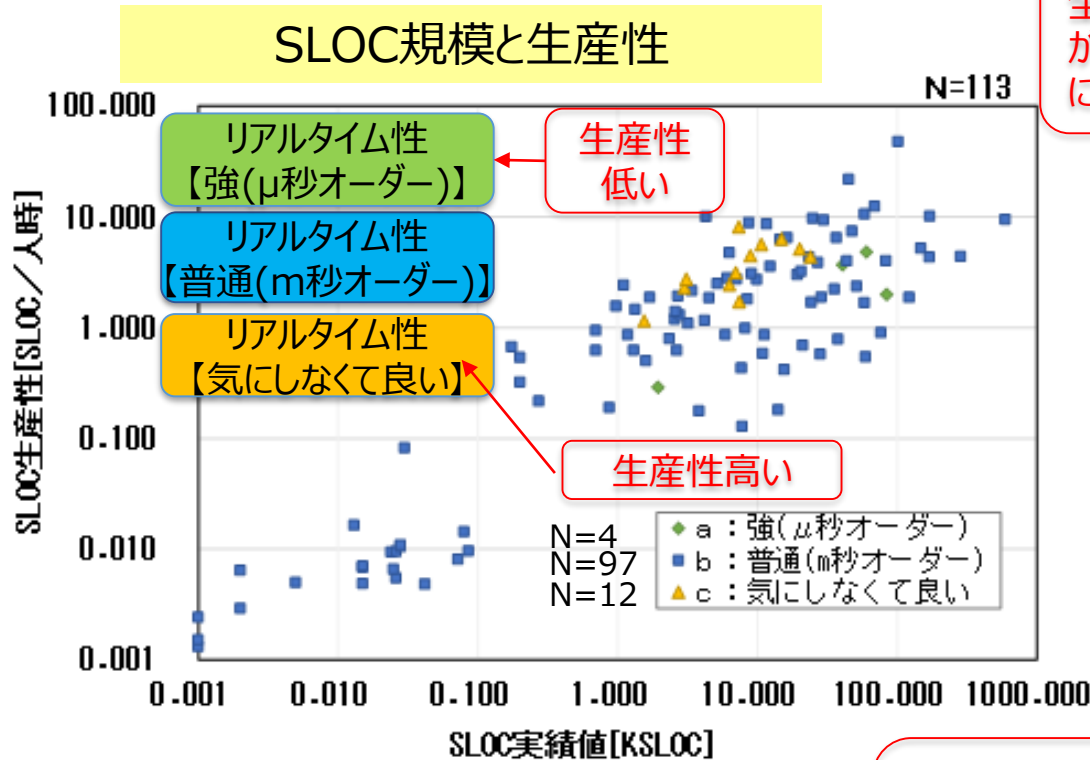
対象製品の特性を分析すれば指標の傾向が分かるようにしたい

- ① リアルタイム性(時間制約)
- ② 自然環境からの影響度合い
- ③ ユーザの多様性
- ④ 法規等による規制度合い
- ⑤ M2Mの有無
- ⑥ ネットワーク接続の有無
- ⑦ 稼動(非停止、オンデマンド)
- ⑧ オンライン保守の可否
- ⑨ 障害リスク(TYPE)

5. 製品の特性ごとの分析

「リアルタイム性(時間制約)」の違い

工数実績の工程比率



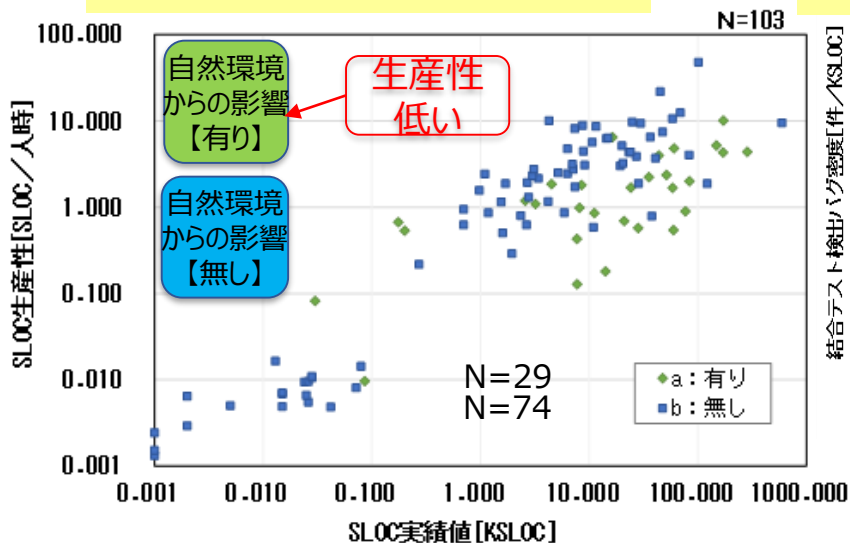
リアルタイム性（時間制約）が強いものと気にしなくて良いものとは、生産性に違いがあることが明らかになった。

その要因は、工数実績の工程比率から分かるように、テスト工程にかかる工数に因るものであることも明らかになった。

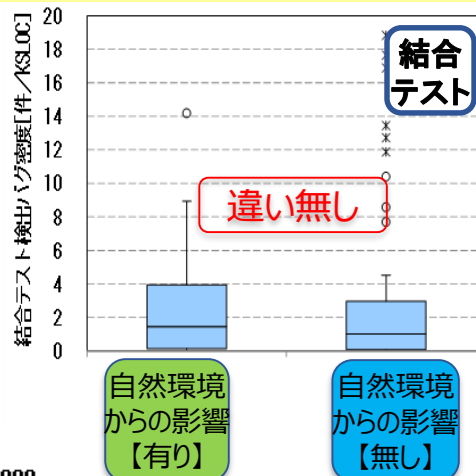
5. 製品の特性ごとの分析

「自然環境からの影響の有無」の違い

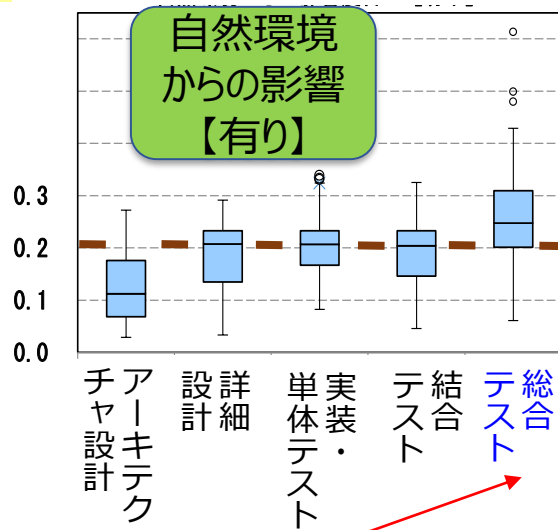
SLOC規模と生産性



テスト検出バグ密度

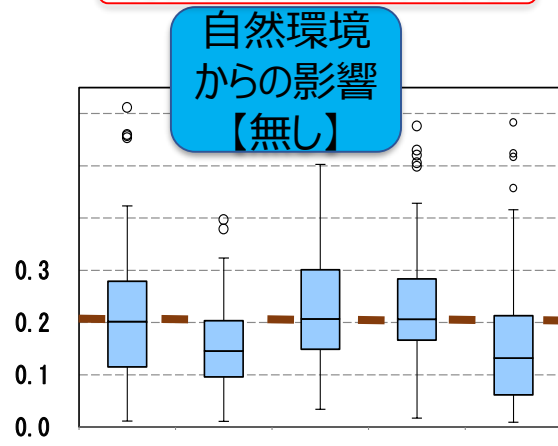
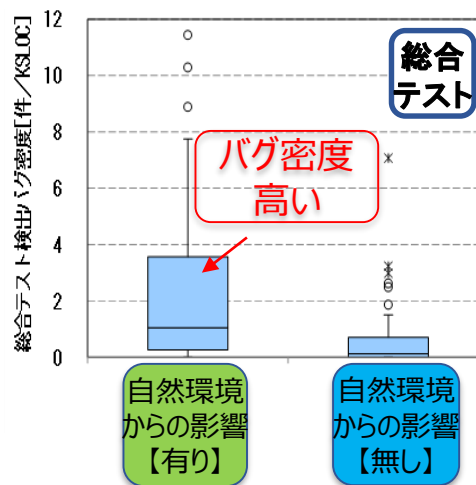


工数実績の工程比率



総合テストに工数がかかる

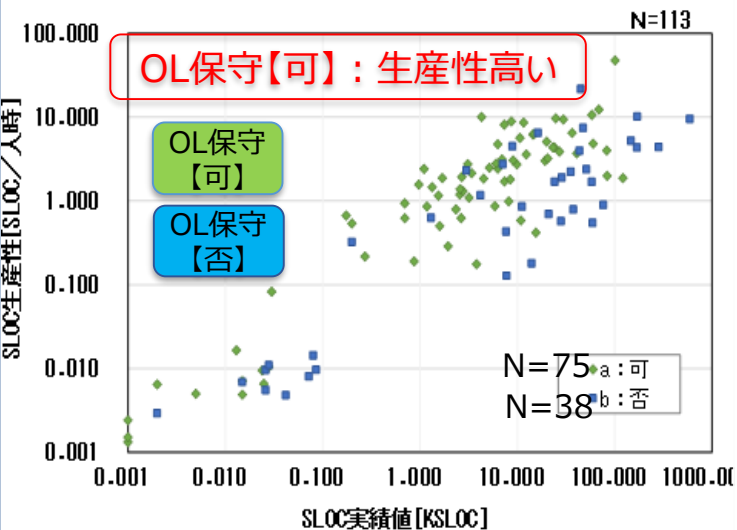
自然環境からの影響を受ける場合は、生産性が低くなる傾向が見られた。その要因は、総合テストでなければ実施できない作業が多くなり、その結果、総合テストに工数がかかるためと考えられる。総合テストで検出するバグが多いことで裏付けられる。



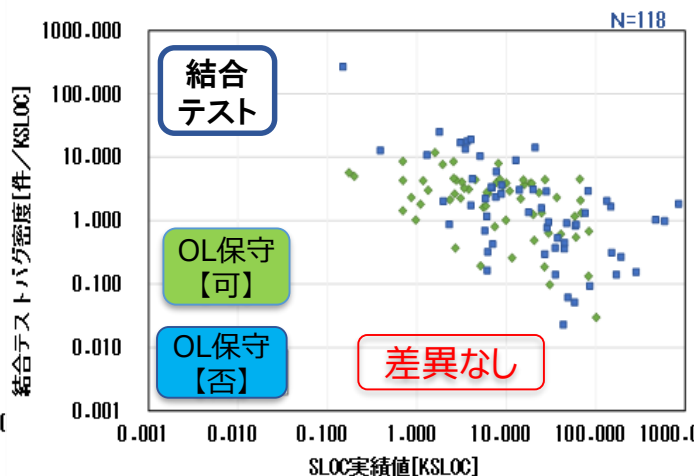
5. 製品の特性ごとの分析

「オンライン保守の可否」による生産性・信頼性(バグ密度)

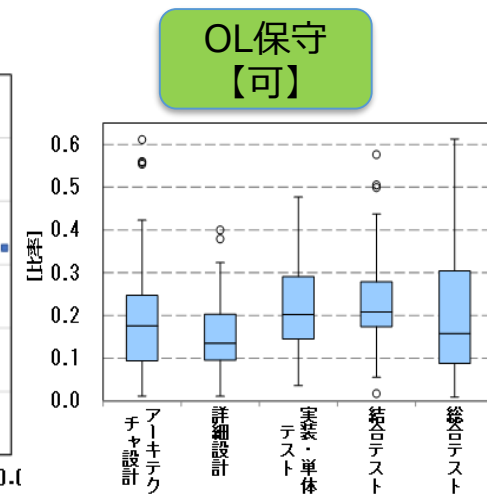
SLOC規模と生産性



テスト検出バグ密度

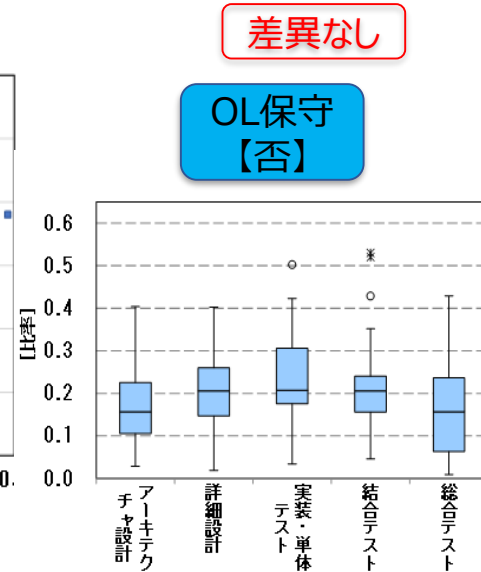
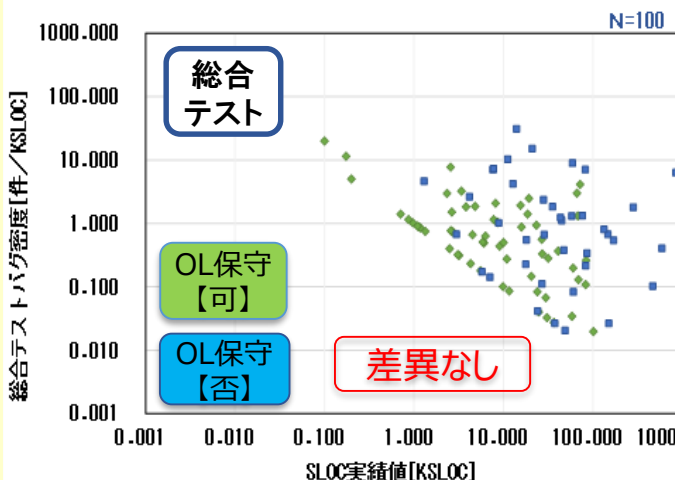


工数実績の工程比率



- ・生産性は、オンライン保守【可】の方が高くなる傾向が見られる。
- ・結合テスト及び総合テストのバグ密度については、顕著な違いはみられない。
- ・また、工数の工程比率についても、大きな違いは見られない。

出荷後にオンライン保守が可能であっても、品質確保作業に違いがないと言える。



組込みソフトウェア開発データ2017」



ご紹介いたします。

6. 「組込みソフトウェア開発データ白書2017」の特徴 **IPA**

2015年11月18日

「組込みソフトウェア開発データ白書2015」発行

データ件数
174件

2017年11月15日

「組込みソフトウェア開発データ白書2017」発行

データ件数
416件

◆2017年版の特徴

組込み製品の特性により生産性・信頼性の違いが見えるようになって来た。

- ①リアルタイム性の違い
- ②自然環境からの影響有無
-等

● 定量データを活用した効果的な プロジェクトマネジメントの推進

- IoT時代の組込みシステム開発の担い手は、今の若い世代である。
- ベテラン世代が作り上げた現在の開発スタイルは、時代の変化に沿って行く必要がある。
- 既に現役世代は、違和感を持っているかもしれない。
- 今の仕事を測って見れば、既に合わなくなったプロセスが見えるのではなかろうか。
- 「組込みデータ白書」に掲載されたプロジェクト管理データを参考に今の仕事を測ってほしい。

8. 「組込みソフトウェア開発データ白書」の活用

● 統計的に分析された**指標値**を参考にする

- 今の仕事を測って見たら、見直すべき仕事が見えるはず。
- 別の必要な作業に気が付くかもしれない。
- その判断の参考となるのが「組込みデータ白書」の分析結果。

（気付き例 1）白書では、リアルタイム性が強く要求されるシステムは全体工数の50%をテストに充てている。自分達は、40%であるが、設計レビューで品質を上げているので問題無い。次期開発プロジェクトは新技術を使うが、このままで良いだろうか？

（気付き例 2）白書では、出荷後に品質問題が頻発するプロジェクトは、コーディングに掛かる工数が大きい（40%）。自分たちの作業もコーディングに40%の工数を割いている。これで良いのだろうか？

ご清聴ありがとうございました

◆データ提供企業募集◆

- 秘密保持契約締結
 - 厳重なデータの管理実施
- 委員会（データ提供企業で構成）にて一緒に分析
 - 参加企業名と委員名は非公開
 - 委員会内部向け「データ白書」毎年発行
 - 委員会は年3回程度

