

自主研究

ソフトウェア開発における工程別生産性に関する分析

ソフトウェア開発における工程別生産性に関する分析

奈良先端科学技術大学院大学 情報科学研究科 戸田 航史 松本 健一
財団法人 経済調査会 調査研究部 第三調査研究室 大岩 佐和子 押野 智樹

1. はじめに

ソフトウェア開発の生産性データは、独立行政法人情報処理推進機構ソフトウェアエンジニアリングセンター（IPA/SEC）の「ソフトウェア開発データ白書」^[1]や財団法人経済調査会（以下経済調査会とする）の調査結果^[2]などから開発言語別、開発方法別などの分析結果が公表されている。ただし、生産性を工程別に論じている例は少ない。

工程別の生産性や生産性増減の傾向やパターンが分かれば、ソフトウェア開発のある時点までの工数から将来の（後工程での）工数が予測可能となる。また、工期の長短、開発言語や開発方法が生産性に特に影響を与える工程、逆に、ほとんど影響を与えない工程、などを知ることが、プロジェクト管理（工数管理）を行

うにあたり非常に有益である。

このような観点から、本稿ではソフトウェア開発に関する調査によって収集されたプロジェクトデータ（ソフトウェア開発データリポジトリ）を分析することで、工期の長短、開発言語、開発方法などによる開発工程別の生産性の傾向について分析を試みた。なお、生産性の指標には、100FPあたりの工数（工数密度）を用いることとする。

2. 利用データ

分析に用いるのは、経済調査会が平成19年度から21年度に実施した「ソフトウェア開発に関する調査」で収集したプロジェクトデータ（ソフトウェア開発データリポジトリ）である。同調査は、ソフトウェア開発における生産性、

表-1 分析に用いたプロジェクト特性値

プロジェクト特性値	単位	定義
規模	FP	未調整ファンクションポイントの実績値（実績FP規模）
工数	人月	ソフトウェア開発の実績工数
工期	月	ソフトウェア開発の実績工期
主開発言語	-	使用率が50%以上のプログラミング言語
開発方法	-	開発プロセスモデルや開発技法
（100FPあたりの）工数密度 【生産性指標】	人月 / 100FP	100FPあたりの工数を表す。生産性指標として工程毎に算出（100FPあたりの）工数密度 = （当該工程における）工数 / （規模 / 100）

表-2 プロジェクト特性値の基本統計量

プロジェクト特性値	件数	平均値	標準偏差	最小値	第1四分位数	中央値	第3四分位数	最大値
規模（実績FP規模）	92	1.962	3.737	23	299	650	1.723	26,572
工数（基本5工程）	92	211.0	490.0	1.8	19.8	46.9	163.5	3,558.2
工期	90	11.7	7.9	0.8	6.0	10.0	15.0	47.0
工数密度（基本5工程）	92	12.32	20.02	0.81	4.57	7.68	13.07	148.57

工数、費用に及ぼす要因の特定などを通じて、ソフトウェア開発の実態を明らかにし、その成果を公表することを目的として、平成10年度からほぼ毎年実施している。同調査では、分析用データとして平成13年度から平成21年度までの9年度分、延べ1,693プロジェクトのデータを蓄積しているが、本稿で用いるのは、平成19年度から21年度と同調査で収集されたデータのうち、次の条件を満足する92プロジェクトのデータである。

- 経済調査会が共通フレーム2007と対応付けし、定義した開発工程区分のうち、基本5工程(基本設計、詳細設計、プログラム(以下PGと略す)設計・製造、結合テスト、総合テスト(ベンダ確認))^{*1}がすべて実施され、各工数が記されている。
- ソフトウェアの規模を表すデータ「実績FP規模」が記されている。

分析に用いた主なプロジェクト特性値は、規模(実績FP規模)、工数、工期、主開発言語、開発技法、そして、生産性指標として(100FPあたりの)工数密度、の計6個である。これら特性値の定義を表-1に、基本統計量を表-2にそれぞれ示す。

なお、本稿では、生産性指標として、一般的な「規模／工数」ではなく、その逆数となる工数密度を用いている。ソフトウェア開発における生産性は、開発工程で生成・利用されたプロダクトの規模を開発工数で割った値を用いることが多い。しかし、本稿では工程別の生産性について論じようとしており、収集したプロジェクトデータには各工程で生成されたプロダクト規模データが含まれておらず、また把握も困難である。そのため、最終成果物の規模である「実績FP規模」を用いることとした。当然のことながら、実績FP規模は、工程に関係なくプロジェクト全体で一定である。この実績FP規模を分母にすることで、規模の異なるプ

ロジェクト間で工数を比較できるだけでなく、工程間の工数が比較できることとなる。とすれば、「規模／工数」ではなく、「工数／規模」とする方が自然である。また、本稿では言及しないが、工程間工数比率などの分析結果等との比較も容易になる。

※1 一般にソフトウェア開発では、設計を行い(設計工程)、設計に従ってプログラムを作成し(製造工程)、プログラムの動作をテストする(試験工程)ことが実施される。経済調査会では、ソフトウェア開発を共通フレーム2007と対応付けて、開発プロセス開始の準備、基本設計(A)、基本設計(B)、詳細設計、PG設計・製造、結合テスト、総合テスト(ベンダ確認)、総合テスト(ユーザ確認)の工程に区分した。各工程と共通フレームの対応関係については「月刊積算資料」(2011年3月号以降)を参照されたい。なお、本稿では基本設計(A)、基本設計(B)をまとめて基本設計とし、基本設計から総合テスト(ベンダ確認)までの基本5工程を分析対象とした。

3. 工程別工数密度

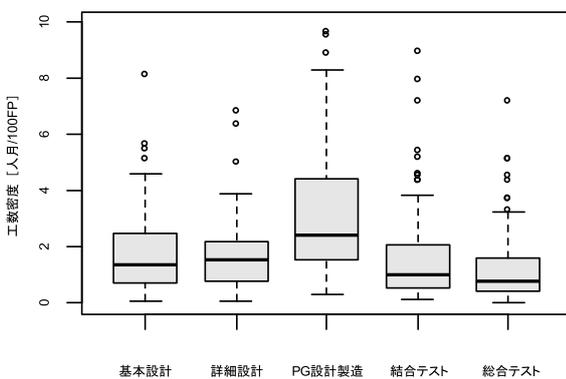
3-1 工程別工数密度の傾向

ここでは、工程間での工数密度の差異をみていく。まず、分析対象として全92プロジェクトにおける基本5工程それぞれの工数密度を図表-1に示す。図表-1より、PG設計・製造工程の工数密度が最も大きく、総合テスト工程が最も小さいことが分かる。その差は平均値でおよそ2.5倍、中央値で3倍である。また、基本設計工程と結合テスト工程の工数密度は、平均も標準偏差(ばらつき)も似通っており、平均値、中央値ともにPG設計・製造工程の約2分の1である。結合テスト工程と総合テスト工程も箱ひげ図の形状は似通っているが、結合テスト工程の方が幾分大きく平均値でおよそ1.2倍、中央値でおよそ1.3倍になっている。

PG設計・製造工程は、設計工程、テスト工

程と比較すると工数密度の平均値が大きだけでなく、標準偏差も大きい。プロジェクト管理（工数管理）の観点としては、影響が大きいので着目すべき工程とみることができる。なお、相対的なばらつきを表す変動係数（＝標準偏差／平均値）でみてみると、工程間の差異はそれほど大きくなく、総合テスト工程はPG設計・製造工程とほぼ同じであることも分かる。

図表-1 基本5工程における工数密度：92プロジェクト



※工数密度が10を超えるものを除いて表示

[単位：人月/100FP]

統計量	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値	2.20	1.87	4.34	2.16	1.75
標準偏差	3.33	2.48	8.73	3.78	3.56
変動係数	1.52	1.33	2.01	1.75	2.04
最大値	27.72	22.24	79.43	27.72	26.86
第3四分位数	2.40	2.19	4.41	2.06	1.58
中央値	1.36	1.51	2.40	1.03	0.80
第1四分位数	0.75	0.77	1.53	0.54	0.44
最小値	0.07	0.06	0.31	0.12	0.03

※変動係数には単位がない。

3-2 工期の長短が工程別工数密度に与える影響

次に、開発工期の長短が工程別工数密度に与える影響についてみていく。開発工期の長短は、経済調査研究レビュー Vol.7「ソフトウェア開発における適正工期に関する分析」^[3]の工数との関係に基づく適正工期限界（適正工期限界＝ $2.65 \times C^{0.30}$ C＝工数）を境界とし、適正工期限界以上の工期のプロジェクトと適正工期限界を下回る短工期のプロジェクトの2つ

のグループを比較した。適正工期限界以上の工期の51プロジェクト、短工期の37プロジェクトにおける基本5工程それぞれの工数密度を図表2に示す。図表2をみると、短工期プロジェクトの方が基本5工程全てにおいて工数密度が高い（生産性が低い）ことが分かる。特に、テスト工程（結合テスト、総合テスト）の工数密度がより高い傾向にあり、その差は平均値でおよそ2倍、中央値で1.5～1.8倍となっている。

短工期の開発のため、上流工程のしわ寄せがテスト工程に及んだことも考えられる。短工期の場合の工数管理では、特に注視すべき点と考えられる。

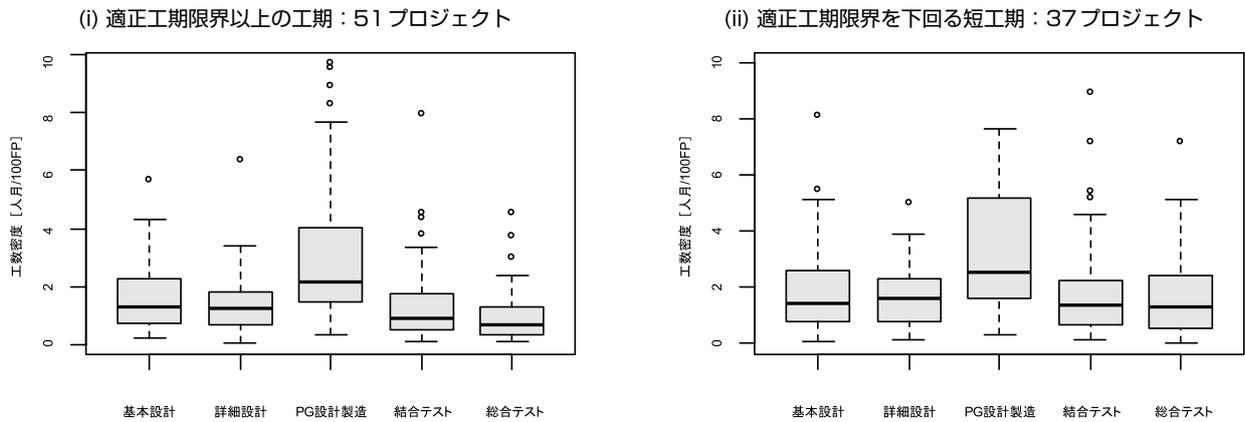
3-3 主開発言語が工数密度に与える影響

次に、主開発言語の違いが、工程別の工数密度に与える影響についてみていく。開発言語がPG設計・製造工程の工数密度に大きな影響を与えることが予想される。また、テストやデバッグの容易性にも関わることから、結合テスト工程、総合テスト工程の工数密度にも影響を与えることが予想される。一方、PG設計・製造工程より上流の基本設計工程や詳細設計工程には直接影響を与えないと考えられる。ただし、例えば、旧来からの金融系アプリケーションでは開発言語にCOBOLが用いられることが多い、など、開発するソフトウェアの適用分野・業種と開発言語にある程度関連性があることから、このような制約がある場合には設計工程の工数密度にも間接的に影響を与えるかもしれない。

ここでは、該当プロジェクトが比較的多い、次の4つの言語群を主開発言語とするプロジェクトを分析対象とする。

- (a) Java：30プロジェクト
- (b) COBOL：8プロジェクト
- (d) VBもしくはVB.NET：9プロジェクト

図表-2 基本5工程における工数密度(工期長短別)



※工数密度が10を超えるものを除いて表示

[単位：人月/100FP]

統計量	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値	1.69	1.49	3.08	1.44	0.99
標準偏差	1.25	1.09	2.39	1.43	0.92
変動係数	0.74	0.73	0.78	0.99	0.93
最大値	5.67	6.36	9.69	7.95	4.56
第3四分位数	2.30	1.85	4.02	1.77	1.32
中央値	1.29	1.28	2.14	0.90	0.71
第1四分位数	0.74	0.71	1.49	0.52	0.37
最小値	0.25	0.06	0.34	0.12	0.10

※変動係数には単位がない。

[単位：人月/100FP]

統計量	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値	2.61	2.24	4.13	2.61	2.21
標準偏差	4.60	3.57	5.04	4.70	3.49
変動係数	1.76	1.59	1.22	1.80	1.58
最大値	27.72	22.24	29.55	27.72	20.41
第3四分位数	2.61	2.28	5.19	2.26	2.42
中央値	1.40	1.62	2.57	1.39	1.31
第1四分位数	0.76	0.78	1.58	0.67	0.51
最小値	0.07	0.13	0.31	0.15	0.03

※変動係数には単位がない。

(d) ASPもしくはASP.NET：9プロジェクト

なお、主開発言語とは、当該プロジェクトにおける使用率が50%以上の言語のことである。2つの言語が50%ずつ使用された場合、それら2つの言語が共に主開発言語となる。ただし、今回の分析対象では同一プロジェクトで2つの言語が50%ずつ使用されたケースはなかった。

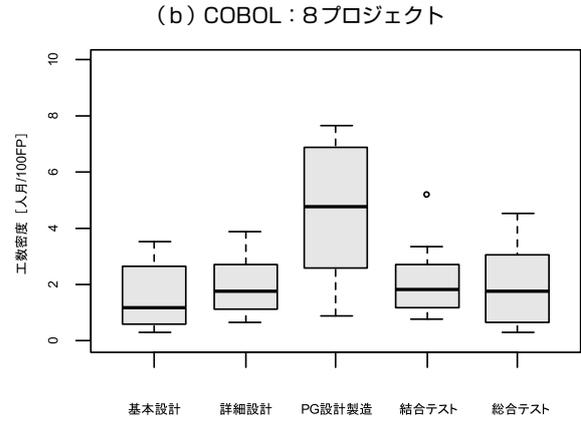
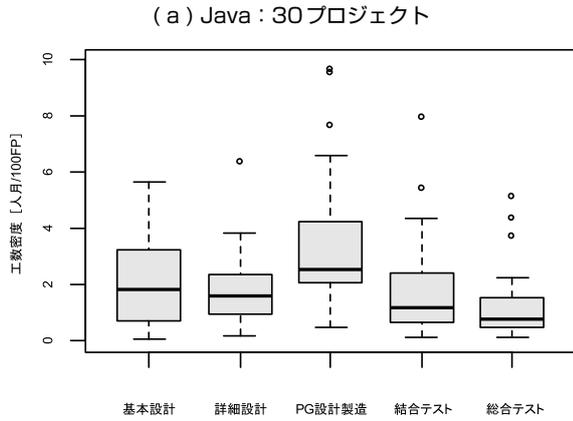
主開発言語ごとの工程別工数密度を図表-3に示す。まず、主開発言語を「Java」とするプロジェクト(図表-3(a))は、プロジェクト数が30件と比較的多いためか、箱ひげ図の形状は図表-1と大差なく、平均値や中央値も大きな違いはない。ただし、標準偏差は5工程全てにおいて小さくなっている。基本設計工程、詳細設計工程、結合テスト工程で約2分の1、総合テスト工程で約3分の1、PG設計・製造工程では約4分の1にもなっている。今回の分析結果からみると、「Java」プロジェクトはばらつき

が小さく、工数超過や進捗遅延といった開発管理リスクが比較的小さいと言える。

主開発言語を「COBOL」とするプロジェクト(図表-3(b))は、プロジェクト数が8件と少ないが、箱ひげ図の形状は図表-1と大差ない。比較的大きな違いは、PG設計・製造工程と総合テスト工程の第3四分位数と中央値が大きくなっている(箱が縦に長くなっている)。数値でみると、中央値がそれぞれ約2倍となっている。ただし、基本5工程全体で平均値はほとんど変わらず標準偏差が小さくなっている点は、「Java」プロジェクトと同様である。標準偏差の値は、詳細設計工程と総合テストで約2分の1、基本設計工程と結合テスト工程で約3分の1、PG設計・製造工程では約4分の1にもなっている。「COBOL」プロジェクトも、今回の分析結果からみると、工数超過や進捗遅延といった開発管理リスクが比較的小さいと言える。

主開発言語を「VBもしくはVB.NET」とするプロジェクト(図表-3(c))の工数密度は、箱

図表-3 基本5工程における工数密度（主開発言語別）

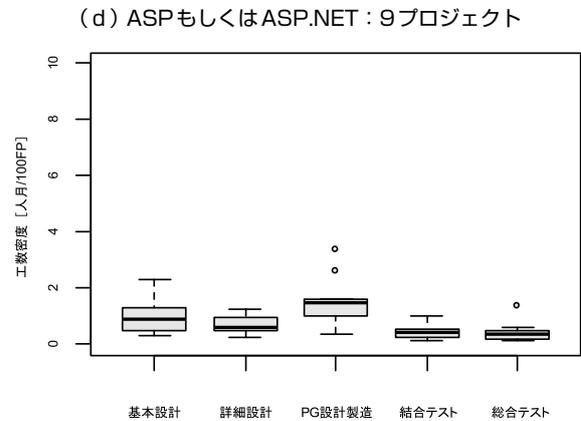
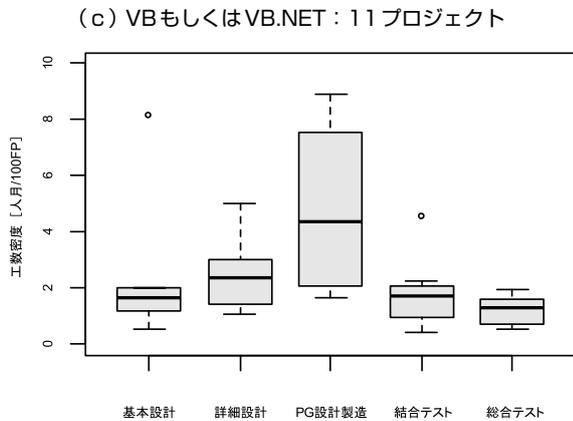


[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		2.03	1.83	3.46	1.81	1.19
標準偏差		1.55	1.26	2.39	1.78	1.23
最大値		5.67	6.36	9.69	7.95	5.13
第3四分位数		3.20	2.34	4.24	2.42	1.53
中央値		1.82	1.61	2.56	1.18	0.80
第1四分位数		0.71	0.98	2.08	0.66	0.51
最小値		0.07	0.17	0.49	0.15	0.12

[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		1.60	1.97	4.64	2.17	1.98
標準偏差		1.22	1.19	2.50	1.46	1.54
最大値		3.53	3.88	7.67	5.20	4.56
第3四分位数		2.46	2.28	6.52	2.39	2.74
中央値		1.19	1.75	4.78	1.82	1.76
第1四分位数		0.63	1.39	2.82	1.31	0.87
最小値		0.32	0.67	0.91	0.79	0.29



※工数密度が10を超えるものを除いて表示

[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		4.36	4.11	6.68	4.02	2.87
標準偏差		8.03	6.12	8.01	7.95	5.84
最大値		27.72	22.24	29.55	27.72	20.41
第3四分位数		2.02	3.00	7.54	2.10	1.58
中央値		1.66	2.35	4.35	1.69	1.29
第1四分位数		1.18	1.44	2.07	0.95	0.70
最小値		0.56	1.09	1.68	0.39	0.51

[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		0.99	0.70	1.58	0.45	0.43
標準偏差		0.65	0.34	0.93	0.27	0.38
最大値		2.33	1.22	3.39	1.00	1.33
第3四分位数		1.28	0.94	1.59	0.56	0.50
中央値		0.86	0.63	1.49	0.43	0.36
第1四分位数		0.46	0.48	1.02	0.25	0.21
最小値		0.29	0.24	0.34	0.12	0.10

ひげ図の形状は、PG設計・製造工程を除き**図表-1**と大差がないようであるが、数値で比べてみると、平均値、中央値、標準偏差いずれも大きくなっていることが分かる。特に、基本設計工程と詳細設計工程は、平均値が約2倍、標準偏差が約2.5倍となっている。なお、PG設計・製造工程については、平均値が約1.5倍、中央値が約1.8倍となっているものの、標準偏差はほとんど差がない。今回の分析結果からみると、「VBもしくはVB.NET」プロジェクトは、工数超過や進捗遅延といった開発管理リスクが、設計工程において比較的大きいと言える。

主開発言語を「ASPもしくはASP.NET」とするプロジェクト(**図表-3(d)**)の工数密度は、他の開発言語によるプロジェクトよりも小さいことが箱ひげ図からも分かる。数値をみるとその差がよりはっきりする。**図表-1**と比べると、平均値は、基本設計工程で約2分の1、詳細設計工程とPG設計・製造工程で約3分の1、結合テスト工程では約5分の1、総合テスト工程では約4分の1に過ぎない。また、中央値ではいずれの工程も約2分の1となっている。標準偏差はいずれの工程も5分の1以下となっている。今回の分析結果からみると、「ASPもしくはASP.NET」プロジェクトは4つの開発言語の中で、最も生産性の高い言語ということが出来る。更に、ばらつきが小さく、工数超過や進捗遅延といった開発管理リスクが比較的小さいと言える。

この節全体を通してみると、箱ひげ図の形状からは、主開発言語の違いが、当初予想していたとおり、PG設計・製造工程の工数密度に特に影響を与えているように見える。ただし、基本統計量の平均値、中央値、標準偏差などを比較してみると、主開発言語の違いが工数密度に与える影響は、基本5工程全体に及んでいることが分かる。工程別にみても、「COBOL」プロジェクトと「VBもしくはVB.NET」プロジェクトは設計工程(基本設計+詳細設計)とテスト工程(結合テスト+総合テスト)の工

数密度に大きな差がないのに対し、「Java」プロジェクトと「ASPもしくはASP.NET」プロジェクトはテスト工程に比べ設計工程の工数密度が高い(工数密度中央値比較では約2倍)ことが分かる。主開発言語による工数密度の工程別傾向を予め知っておくことは、開発管理上、非常に重要である。

3-4 開発方法が工数密度に与える影響

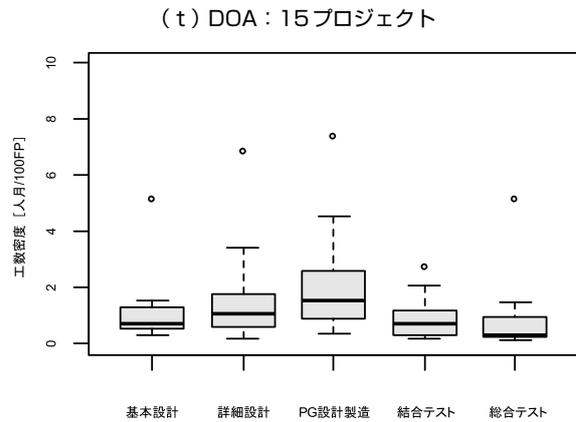
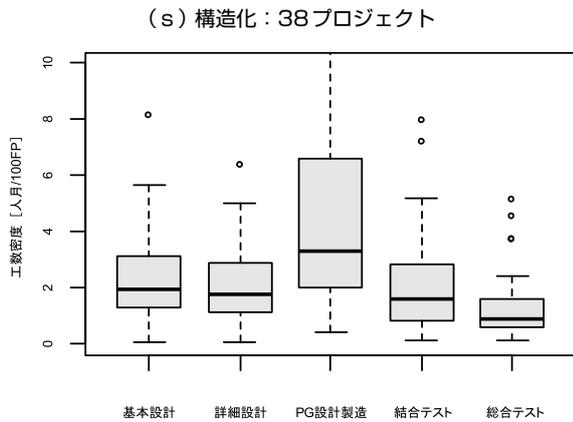
次に、開発方法の違いが、工程別の工数密度に与える影響についてみていく。開発方法とは、開発プロセスモデルや開発技法であり、基本設計工程および詳細設計工程の工数密度に直接大きな影響を与えることが予想される。また、PG設計・製造やテストの容易性にも関わることから、PG設計・製造工程以降の工数密度にも影響を与えることが予想される。

ここでは、次の3つの開発技法のいずれかを用いたプロジェクトを分析対象とする。

- (s) 構造化：38プロジェクト
- (t) DOA (Data Oriented Approach)：15プロジェクト
- (u) オブジェクト指向：35プロジェクト

開発技法ごとの工程別工数密度を**図表-4**に示す。まず、開発技法を「構造化」とするプロジェクト(**図表-4(s)**)は、箱ひげ図の形状でみると**図表-1**と大差がない。強いて言えば、PG設計・製造工程の第3四分位数と中央値が大きくなっている(箱が縦に長くなっている)。平均値は、基本5工程いずれにおいても若干大きくなっているが、標準偏差は大きくなっている工程と小さくなっている工程がある。具体的には、基本設計工程と詳細設計工程では大きくなっているが、PG設計・製造工程では逆に小さくなっている。結合テスト工程でも若干大きくなっているが、総合テスト工程ではほとんど変わっていない。

図表-4 基本5工程における工数密度 (開発技法別)



※工数密度が10を超えるものを除いて表示

※工数密度が10を超えるものを除いて表示

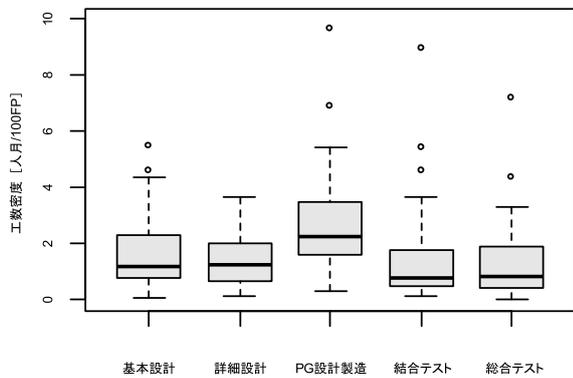
[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		2.95	2.56	4.86	2.76	1.83
標準偏差		4.42	3.52	5.03	4.52	3.31
最大値		27.72	22.24	29.55	27.72	20.41
第3四分位数		3.11	2.84	6.53	2.71	1.61
中央値		1.96	1.75	3.31	1.62	0.88
第1四分位数		1.30	1.13	2.01	0.85	0.60
最小値		0.08	0.06	0.41	0.15	0.12

[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		1.96	1.53	7.21	2.21	2.55
標準偏差		3.61	1.70	20.06	5.29	6.84
最大値		14.29	6.86	79.43	21.14	26.86
第3四分位数		1.31	1.75	2.59	1.16	0.98
中央値		0.71	1.07	1.55	0.74	0.29
第1四分位数		0.55	0.61	0.87	0.29	0.22
最小値		0.32	0.17	0.34	0.21	0.14

(u) オブジェクト指向：35プロジェクト



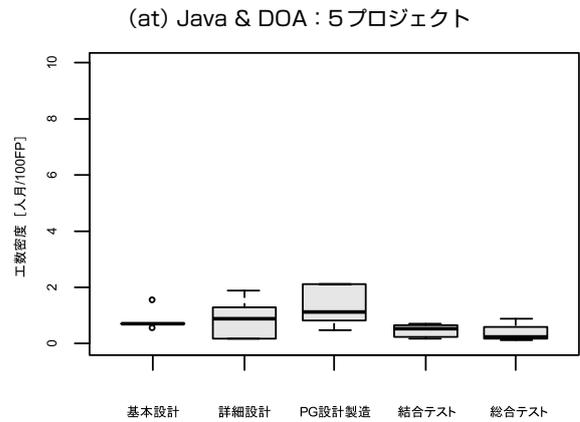
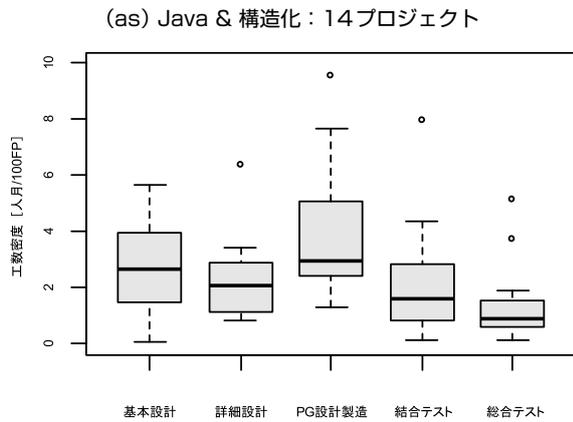
[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		1.59	1.31	2.73	1.51	1.44
標準偏差		1.35	0.83	1.84	1.82	1.47
最大値		5.50	3.67	9.69	8.95	7.19
第3四分位数		2.28	2.00	3.48	1.75	1.89
中央値		1.20	1.22	2.23	0.76	0.82
第1四分位数		0.76	0.63	1.58	0.45	0.43
最小値		0.07	0.13	0.31	0.12	0.03

開発技法を「DOA」とするプロジェクト(図表-4(t))は、箱ひげ図の形状からすると、図表-1よりも工数密度のばらつきが小さくなっているように見える。数値で比べてみると、基本設計工程と詳細設計工程、そして、結合テスト工程では、平均値も標準偏差も大きく変わっていないが、PG設計・製造工程では、平均値が約1.7倍、標準偏差が約2.3倍と共に大きくなっている。また、総合テスト工程でも、平均値が約1.5倍、標準偏差が約1.9倍となっている。しかし、上記の平均値、標準偏差の数値が高いのは、突出した値が含まれているためで、箱ひげ図で比較をすると3つの開発技法の中では、第1四分位数と第3四分位数の幅が最も少なく、また、中央値が最も小さくなっている。

開発技法を「オブジェクト指向」とするプロジェクト(図表-4(u))は、箱ひげ図の形状でみると図表-1と大差ないようであるが、数値で比べてみると、工数密度の平均値も標準偏差も

図表-5 基本5工程における工数密度 (Java & 開発技法別)

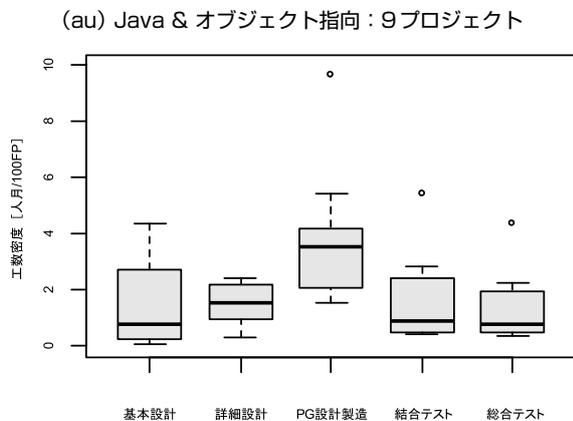


[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		2.69	2.27	3.93	2.14	1.45
標準偏差		1.61	1.43	2.45	2.00	1.39
最大値		5.67	6.36	9.54	7.95	5.13
第3四分位数		3.92	2.79	4.86	2.65	1.56
中央値		2.64	2.05	2.98	1.58	0.87
第1四分位数		1.57	1.23	2.45	0.86	0.65
最小値		0.08	0.86	1.28	0.15	0.12

[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		0.84	0.88	1.33	0.48	0.40
標準偏差		0.40	0.72	0.75	0.24	0.32
最大値		1.55	1.88	2.11	0.74	0.89
第3四分位数		0.72	1.28	2.11	0.66	0.59
中央値		0.71	0.88	1.10	0.55	0.22
第1四分位数		0.69	0.21	0.83	0.24	0.18
最小値		0.54	0.17	0.49	0.21	0.14



[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		1.67	1.50	3.80	1.68	1.34
標準偏差		1.61	0.75	2.56	1.66	1.31
最大値		4.35	2.42	9.69	5.43	4.35
第3四分位数		2.72	2.17	4.18	2.42	1.94
中央値		0.76	1.53	3.57	0.89	0.76
第1四分位数		0.27	0.95	2.07	0.51	0.51
最小値		0.07	0.33	1.53	0.41	0.34

小さくなっていることが分かる。平均値はいずれの工程も約0.7倍である。標準偏差は、基本設計工程、結合テスト工程、総合テストでそれぞれ約2分の1、詳細設計工程で約3分の1、PG設計・製造工程で約5分の1である。なお、中央値はそれほど大きくは変わっていない。

この節全体を通してみると、いずれの開発技法も箱ひげ図の形状は類似しているものの、開発技法によって工数密度のばらつきと値には差があることが分かる。基本5工程において、工数密度のばらつき（第1四分位数と第3四分位数の幅）、工数密度（中央値）ともにDOA < オブジェクト指向 < 構造化の順となっている。

3-5 Java による開発において開発技法が工数密度に与える影響

図表-3と図表-4を比較すると、主開発言語が工数密度に与える影響は、開発技法による影響と同等かそれ以上のようにみえる。主開発言語

を限定せずに開発技法の違いだけに着目しても有意な結果は得られない可能性がある。そこで、データ件数が最も多かった「主開発言語をJavaとするプロジェクト」を対象に、開発技法の違いが工数密度に与える影響をみることにする。対象プロジェクト数は次のとおりである。

- (as) Java & 構造化：14プロジェクト
- (at) Java & DOA：5プロジェクト
- (au) Java & オブジェクト指向：9プロジェクト

「主開発言語をJavaとするプロジェクト」における開発技法ごとの工程別工数密度を**図表-5**に示す。「Java & 構造化」プロジェクト(**図表-5(as)**)は、全体の半数を占めることもあり、箱ひげ図の形状は、**図表-3(a)**で示した「Java」プロジェクトとも、**図表-4(s)**で示した「構造化」プロジェクトとも、似通っている。ただし、数値で比べると、基本5工程全てにおいて、「構造化」プロジェクトよりも標準偏差が約2分の1と小さくなっている。

「Java & DOA」プロジェクト(**図表-5(at)**)は5件しかいないため、結果に偏りがある可能性があるが、箱ひげ図の形状は、**図表-3(a)**で示した「Java」プロジェクトよりも、**図表-4(t)**で示した「DOA」プロジェクトに近い。「DOA」プロジェクトと数値で比べてみると、工数密度の平均値と標準偏差が共に小さくなっていることが分かる。平均値は、特にテスト工程で小さくなっており、結合テスト工程では約5分の1、総合テスト工程では約6分の1である。標準偏差は、結合テスト工程、総合テスト工程共に、約20分の1しかない。

「Java&オブジェクト指向」プロジェクト(**図表-5(au)**)の箱ひげ図の形状は、**図表-3(a)**で示した「Java」プロジェクトとも**図表-4(u)**で示した「オブジェクト指向」プロジェクトとも似通っている。また、数値で比べてみても、どちらも大きな違いはみられない。これは、もともと「Java」プロジェクトと「オブジェクト指向」

プロジェクトの工数密度が、基本5工程全てにおいて似通った値となっていたためである。

この節全体を通してみると、主開発言語をJavaと限定することは、基本5工程全てにおいて工数密度の標準偏差を小さくし、開発技法が工数密度に与える影響をより明確にする効果があるようである。今回はデータ数の関係で主開発言語がJavaのケースのみみてみたが、他の言語でも同様のことが言える可能性がある。ただし、結論を下すためには更に多くのデータで分析を行う必要がある。

3-6 構造化もしくはオブジェクト指向による開発において主開発言語が工数密度に与える影響

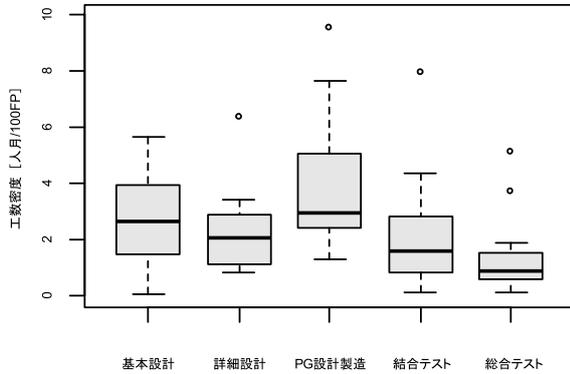
先に述べたとおり、開発技法と工数密度の関係は、主開発言語と工数密度の関係よりも複雑である可能性がある。3.5において、主開発言語をJavaと限定したのと同様に、開発技法を限定することで、主開発言語が工数密度に与える影響をより明確にすることができるかもしれない。そこで、データ件数が最も多かった「開発技法を構造化とするプロジェクト」を対象に、主開発言語の違いが工数密度に与える影響をみることにする。対象プロジェクトは次のとおりである。

- (sa) 構造化 & Java：14プロジェクト
- (sb) 構造化 & COBOL：5プロジェクト
- (sc) 構造化 & (VB & VB.NET)：8プロジェクト
- (sd) 構造化 & (ASP & ASP.NET)：1プロジェクト

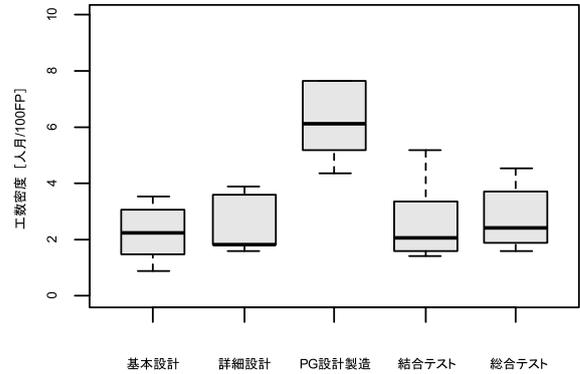
「開発技法を構造化とするプロジェクト」における主開発言語ごとの工程別工数密度を**図表-6**に示す。ただし、「構造化 & ASPもしくはASP.NET」プロジェクトは、1プロジェクトのみしかなく分析対象とはしないが、**図表-3**と**図表-6**の対比ができるよう、参考までに**図表-6**に結果を示している。

図表-6 基本5工程における工数密度 (構造化 & 主開発言語別)

(sa) 構造化 & Java : 14 プロジェクト



(sb) 構造化 & COBOL : 5 プロジェクト



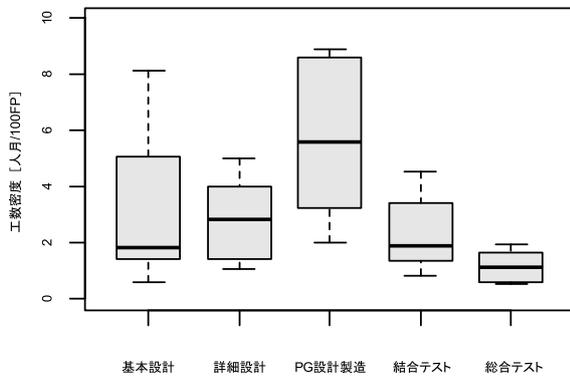
[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		2.69	2.27	3.93	2.14	1.45
標準偏差		1.61	1.43	2.45	2.00	1.39
最大値		5.67	6.36	9.54	7.95	5.13
第3四分位数		3.92	2.79	4.86	2.65	1.56
中央値		2.64	2.05	2.98	1.58	0.87
第1四分位数		1.57	1.23	2.45	0.86	0.65
最小値		0.08	0.86	1.28	0.15	0.12

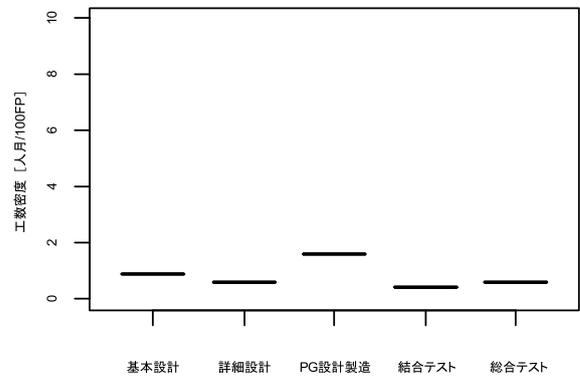
[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		2.25	2.55	6.21	2.73	2.84
標準偏差		1.08	1.10	1.47	1.57	1.25
最大値		3.53	3.88	7.67	5.20	4.56
第3四分位数		3.04	3.61	7.66	3.35	3.71
中央値		2.27	1.83	6.14	2.08	2.42
第1四分位数		1.49	1.81	5.19	1.59	1.90
最小値		0.90	1.62	4.38	1.45	1.62

(sc) 構造化 & VBもしくはVB.NET : 8 プロジェクト



(sd) 構造化 & ASPもしくはASP.NET : 1 プロジェクト



※工数密度が10を超えるものを除いて表示

[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		5.64	5.00	8.31	5.24	3.47
標準偏差		9.23	7.08	8.96	9.15	6.86
最大値		27.72	22.24	29.55	27.72	20.41
第3四分位数		3.56	3.55	8.46	2.84	1.53
中央値		1.85	2.85	5.63	1.90	1.15
第1四分位数		1.55	1.57	3.80	1.54	0.66
最小値		0.60	1.09	2.01	0.80	0.51

[単位：人月/100FP]

統計量	工程	基本設計	詳細設計	PG設計製造	結合テスト	総合テスト
平均値		0.86	0.63	1.59	0.43	0.60
標準偏差		NA	NA	NA	NA	NA
最大値		0.86	0.63	1.59	0.43	0.60
第3四分位数		0.86	0.63	1.59	0.43	0.60
中央値		0.86	0.63	1.59	0.43	0.60
第1四分位数		0.86	0.63	1.59	0.43	0.60
最小値		0.86	0.63	1.59	0.43	0.60

「構造化 & Java」プロジェクト(図表-6(sa))は、図表-5(as)で示した「Java & 構造化」プロジェクトと同じである。従って、その箱ひげ図の形状は、図表-3(a)で示した「Java」プロジェクトとも、図表-4(s)で示した「構造化」プロジェクトとも、似通っている。そのため、これも同様に、数値で比べると、基本5工程全てにおいて、「構造化」プロジェクトよりも標準偏差が約2分の1と小さくなっている。

「構造化 & COBOL」プロジェクト(図表-6(sb))の箱ひげ図の形状は、PG設計・製造工程が少し異なるものの、全体としては、図表-4(s)で示した「構造化」プロジェクトよりも、図表-3(b)で示した「COBOL」プロジェクトに近い。数値で比べてみると、工数密度の平均値は「構造化」プロジェクトと似通った比較的大きな値となっているが、標準偏差は逆に「COBOL」プロジェクトと似通った比較的小きな値となっている。

「構造化 & (VB & VB.NET)」プロジェクト(図表-6(sc))の箱ひげ図の形状は、図表-3(c)で示した「VB & VB.NET」プロジェクトとも、図表-4(s)で示した「構造化」プロジェクトとも少し異なる。特に、基本設計工程の第3四分位数が大きくなっている(箱が縦に長くなっている)。また、PG設計・製造工程では逆に第3四分位数が小さくなっている(箱が縦に短くなっている)。数値で比べてみると、平均値も標準偏差も「VB & VB.NET」プロジェクトに近く、特に標準偏差は、基本5工程全てにおいて、「構造化」プロジェクトの約2倍の値となっている。

この節全体を通してみると、開発技法を構造化と限定した場合、箱ひげ図の形状が一部異なる場合があるものの、全体としては、主開発言語における工数密度の傾向と近いものとなった。つまり、主開発言語が工数密度に与える影響は、開発技法による影響による影響と同等かそれ以上ということが考えられる。ただし、結論を下すためには更に多くのデータで分析を行う必要がある。

4. まとめ

本稿では、財団法人経済調査会が平成19年度から21年度に実施した「ソフトウェア開発に関する調査」で収集されたプロジェクトデータ(ソフトウェア開発データリポジトリ)を分析することで、ソフトウェア開発で用いられた主開発言語や開発技法が、基本5工程(基本設計、詳細設計、PG設計・製造、結合テスト、総合テスト(ベンダ確認))における工数密度に与える影響についてみてきた。得られた主な知見は次のとおり。

- (1) PG設計・製造工程は、設計工程、テスト工程に比べ工数密度の平均値、標準偏差ともに大きい。プロジェクト管理(工数管理)の観点としては、影響が大きいので着目すべき工程とみることができる。
- (2) 短工期のプロジェクトは基本5工程全てにおいて工数密度が高く(生産性が低く)、特にテスト工程の工数密度がより高い(多くの工数を投入している)傾向がある。
- (3) 主開発言語の違いが工数密度に与える影響は、基本5工程全体に及んでおり、PG設計・製造工程の工数密度に特に影響を与えている。「Java」プロジェクトと「ASPもしくはASP.NET」プロジェクトはテスト工程に比べ設計工程の工数密度が高い(多くの工数を投入している)傾向がある。4つの言語の中で最も工数密度が低い(生産性が高い)のは「ASPもしくはASP.NET」プロジェクトである。
- (4) 開発技法の違いが工数密度に与える影響をみると、基本5工程全てにおいて、工数密度のばらつき(第1四分位数と第3四分位数の幅)、工数密度(中央値)ともDOA<オブジェクト指向<構造化の順となっている。
- (5) 主開発言語をJavaと限定した場合、基本5工程全てにおいて工数密度の標準偏差を小

さくし、開発技法が工数密度に与える影響をより明確にする効果があるようである。

- (6) 開発技法を構造化と限定した場合、主開発言語が工数密度に与える影響は、開発技法による影響と同等かそれ以上ということが考えられる。

今回の分析で得られた知見をより一般化するためには、分析対象データ数を増やし、同一条件下における分析が必要と考えられる。また、各工程における工数密度には、開発プロセスモデル（ウォーターフォール、繰り返し型プロセス等）が影響を与えることも予想されるが、本稿で分析対象としたプロジェクトのほとんどが「ウォーターフォール」型の開発であったため、開発プロセスモデルに基づく分析は実施しなかった。更に、「ソフトウェア開発に関する調査」では、開発全体での生産性変動要因として、「信頼性要求度」、「SE・プログラマの経験と能力」、「発注者の参画割合」、「プラットフォームの適正」等が挙げられている。これら要因と生産性の関係を工程別に議論することも、今後の重要な課題である。

参考文献

- [1] 独立行政法人情報処理推進機構ソフトウェアエンジニアリングセンター（IPA/SEC）：“ソフトウェア開発データ白書2010-2011”（最新版）、2010年
- [2] 財団法人経済調査会：“ソフトウェア開発データリポジトリの分析”，2010年
 ※財団法人経済調査会が、毎年実施する「ソフトウェア開発に関する調査」で収集したプロジェクトデータのデータベース。平成13年度から平成21年度までの9年度分で、のべ1,693プロジェクトのデータを蓄積している。平成13年度から平成18年度までの6年度分については、蓄積データの収集・分析結果を冊子「ソフトウェア開発データリポジトリの分析」（234ページ）として、平成22年7月に発行した。同冊子のPDF版が財団法人経済調査会のWebサイトの「研究成果」で確認できる。
<http://www.zai-keicho.or.jp/>
- [3] 戸田航史、松本健一、押野智樹、高橋昭彦：“ソフトウェア開発における適正工期に関する分析”，経済調査レビュー Vol.7，財団法人経済調査会，2010年