

# 初中級プロマネのための 現場で活かせ！統計情報2

土屋俊樹  
株式会社ハイマックス

**概要**：公開されている統計情報から、システム開発規模を簡単に把握する方法を紹介します。

**キーワード**：見積り作業、システム開発、ソフトウェアデータ白書、統計情報

## 1. プロジェクトマネジャーの憂い

初中級プロジェクトマネジャーにとってシステム開発作業における最大の難関のひとつに見積り作業があります。よく言われているように見積り作業はシステム開発におけるプランニングそのものであり、経験豊富なプロジェクトマネジャーであっても難易度の高い作業であります。ところがシステム開発の現場では、開発規模全体を見積るために要件定義をしているにもかかわらず、その段階で上司や顧客から「超概算で良いのでシステム全体の開発規模感を教えてほしい」とのムチャ振り要求がごく普通にあります。<sup>\*1</sup>

今回は、そんな初中級プロジェクトマネジャーの悩みに多少なりとも手助けとなる方法を紹介します。「とりあえず機能一覧から製造規模くらいなら何とか積上げで算出できたけど…」という段階まで見積りできたらしめたものです。そこから先は、公開されている統計情報を使って全体規模感を把握しましょう。

## 2. 統計情報の活用

2018年10月にIPAよりソフトウェア開発データ白書の最新版（2018-2019）（以降データ白書）<sup>\*2</sup>が発表されました。前回版と比べ内容がより一層充実しています。関係各位・各社の尽力に敬意を表します。筆者前回レポート<sup>\*3</sup>では、主に生産性、工程別比率、品質指標値の統計情報をご紹介しましたが、今回は統計値を最新版にリフレッシュするとともに、筆者なりの実際の現場での使い方を示したいと思います。

尚、当手法は筆者の勝手な解釈に基づいた使用法であり、当然ながら結果を保証したりするものではありません。その目的を十分にご理解の上ご活用ください。

## 3. 開発規模の算出

本書ではシステム開発規模＝投入工数（人月<sup>\*4</sup>）をとらえます。一般的にシステム開発規模は、次の式で表せます。

式 1) プログラム製造規模(SLOC) ÷ 生産性(SLOC/月) = システム開発規模(人月)

当たり前のことですが、プログラム製造規模と生産性が分かればシステム開発規模（人月＝投入工数）が算出できるわけです。プログラム製造規模とは、所謂個々のプログラム Step 数（SLOC：Source Lines Of Code）を積上げた合計値です。プログラム製造規模の見積り方法は、SE クラスの方であれば経験があると思われるので割愛します。

次に生産性ですが、データ白書にはいくつかの生産性数値があります。新規開発における中央値\*5を使って開発工数を把握してみましょう。

### 3-1. 言語別生産性から算出

まずは開発言語別の生産性です（付録 1 参照）。これはシステム開発 5 工程全体（基本設計、詳細設計、製作、結合テスト、総合テスト）の生産性になります。制作工程（コーディング＋単体テスト）の生産性ではないことに注意してください。全体製造規模を採用言語における生産性で割ることにより、開発 5 工程の工数を算出します。

$$\text{式 2) 全体製造規模(SLOC)} \div \text{言語別生産性(月当り)} = \text{開発 5 工程工数(人月)}$$

■パターン1		開発言語別生産性より、単純に全体工数を算出							
↓全体SLOC	100,000	÷	基本設計	詳細設計	製造・単体T	結合テスト	総合テスト	Step/月	←Java生産性
			797						N=415
		=	125.48					人月	←全体工数

図. パターン 1 では、Java 生産性（797 SLOC／月：中央値）の場合を示しています。例では Java 製造規模が 100,000 SLOC（100 KSLOC）だった場合、システム開発 5 工程では 125.48 人月必要なことがわかります。（より正確には、中央値が 125.48 人月、上下 50 パーセントの範囲で 72.35 人月～201.61 人月である、と捉えるべきかもしれません）

### 3-2. 業種別生産性から算出

次に業種別の生産性です（付録 2）。計算方法は、パターン 1 と同様です。

■パターン2		業種別生産性より、単純に全体工数を算出							
↓全体SLOC	100,000	÷	基本設計	詳細設計	製造・単体T	結合テスト	総合テスト	Step/月	←金融・保険業生産性
			667						N=308
		=	149.93					人月	←全体工数
↓全体SLOC	100,000	÷	基本設計	詳細設計	製造・単体T	結合テスト	総合テスト	Step/月	←卸売・小売業生産性
			963						N=88
		=	103.85					人月	←全体工数

図. パターン 2 では、金融・保険業種と卸売・小売業種の生産性を示しています。データ白書では、代表的な 5 業種の生産性があります。開発するシステム特性によって使い分けができます。

### 3-3. 工程別生産性より算出

データ白書には開発 5 工程全体の生産性以外に開発工程別の生産性（付録 3）も記載されています。特定の開発言語ではありませんが各工程の規模を把握したいときに何かと重宝します。

$$\text{式 3) 全体製造規模(SLOC)} \div \text{工程別生産性(月当り)} = \text{該当工程の工数(人月)}$$

■パターン3 工程別の生産性より工程別工数、全体工数を算出

↓全体SLOC		基本設計	詳細設計	製造・単体T	結合テスト	総合テスト	
100,000	÷	4,992	4,912	2,320	4,176	6,560	Step/月 ←工程別生産性
	=	20.04	20.36	43.11	23.95	15.25	人月 ←工程別工数
		5工程小計 122.71					人月 ←全体工数

図. パターン 3 では、全体製造規模を各工程の生産性で割ることにより、各工程での開発規模を算出しています。100KSLOC 規模であれば、基本設計に約 20 人月必要なことがわかります。

### 3-4. 工程別工数比率より算出

工程別の統計値には生産性だけでなく工程別工数比率（付録 4）もあります。例えば制作工程の工数がわかれば、各工程、およびその積上げで工程全体の工数が算出できます。

$$\text{式 4) 製造工程工数} \div \text{製造工程比率} \times \text{該当工程比率} = \text{該当工程の工数(人月)}$$

■パターン4 パターン3の製造工数を元に、工程別の工数比率で、工程別工数を算出

↓パターン3製造工程工数		基本設計	詳細設計	製造・単体T	結合テスト	総合テスト	
43.11	×	16.9%	17.5%	32.6%	20.0%	13.0%	←工程別工数比率
	=	22.35	23.14	43.11	26.45	17.19	N=304 ←工程別工数
		5工程小計 132.24					人月

図. パターン 4 では、パターン 3 で算出した製造工程の工数をもとに開発 5 工程の工数を算出しています。

### 3-5. 要件定義工数を算出

最後に要件定義工程の工数を算出してみましょう。要件定義工程の工数について妥当な見積りをするには中々難しいものです。データ白書では要件定義工程の工数比率（付録 5）から平均的な要件定義工程の工数が算出できます。

$$\text{式 5) 5 工程工数} \div \text{5 工程工数比率} \times \text{要件定義工程工数比率} = \text{要件定義工程の工数}$$

■パターン5 パターン4の全体工数を元に、要件定義工数比率で要件定義工数を算出

↓パターン4全体工数		要件定義	基本設計	詳細設計	製造・単体T	結合テスト	総合テスト	
132.24	×	8.9%	91.1%					←工程別工数比率
	=	12.92	132.24					N=177 ←工程別工数
		全体小計 145.16					人月	←要定義含む全体工数

図. パターン 5 ではパターン 4 の 5 工程工数を元に要件定義工程の工数を算出しています。

新規開発においては開発5工程に対して、約9%（約13人月）の工数をかけて要件定義をする必要があることがわかります。

#### 4. 超概算工数をどう使うか

今回はソフトウェアデータ白書の各種生産性や工程比率等に着目し、プログラム製造規模からスピーディに新規開発時の規模感を把握する方法を紹介しました。当然ですがデータ白書にも書かれている通り、システム開発は各ステークホルダーとの関係やプロジェクト特性、自社や協力会社のスキル等、様々な要素が絡んでおり、それらを考慮せず一概に統計値だけで見積り作業を行うことは非常に危険な行為です。また、今回紹介した生産性自体も採用する生産性によって開発規模も少しずつ異なり、これが正解というものがあるわけではありません。そのような超概算工数をどのような場面で使えばよいのでしょうか。

筆者は例えば開発立上げ当初の規模感把握に使ったりしています。全体開発規模が不明な企画段階や要件定義段階では、まずは大まかな開発規模感が得られれば良く、正確な数値が求められるわけではありません。<sup>\*6</sup> またシステム開発中や開発完了後には、その工程の振り返りとして統計値と比較した分析を行ったりしています。

そして、データ白書の数値は、基本的に成功したプロジェクトの数値<sup>\*7</sup>なのです。つまりこれらの統計値からあまりブレない開発をする限り、成功する確率が高まるのではないかと筆者は考えます。

まずは数値に慣れる意味でも現場で統計値を実際に使ってみる、というのはいかがでしょうか。

ただし、念押しいたしますが統計値を元に算出した値をそのまま見積り工数に使う行為は、データ白書にもある通り非常に「危険な使い方」なので、あくまでも規模感把握のための目安、評価の為に比較する数値、と割り切って使いましょう。

#### 5. おわりに

最新版のデータ白書では、今回紹介した以外にも非常に役に立つと思われる統計値が多く掲載されています。辞書のように分厚い本ではありますが、興味のある方はぜひ手に取り実務で試してみてください。定性的なKKD（経験、勘、度胸）に統計情報の定量的な裏付けをすることで、よりスマートなプロジェクトマネジメント(?)を实践したいものです。

## 【余談】

### ウォーターフォール型開発が97.4%！\*8

データ白書で集計した 1,338 件のプロジェクトのうち、反復型（≡アジャイル型開発）はわずか 1.7%しかありません。これは統計データを提供している企業が、いわゆる大型開発を主流としている大手ベンダーだからでしょうか。一方で統計データのうち、決して大規模とはいえない 100KSLOC 以下の中小開発規模のプロジェクトは 8割弱\*9となっています。つまり統計データを提供している企業では、開発規模の大小を問わずまだまだウォーターフォール型開発が主流と見て取れます。もしくは各企業において統計データを取りやすい開発がウォーターフォール型開発だから、ということかもしれません。筆者もウォーターフォール開発を主に従事としておりアジャイル開発経験は多くありません。

一方で今年度（2018年春）、アジャイル開発とは最も縁遠いと思われたシステム監査試験（筆者の個人的な見解です）でも、午後Ⅱ小論文でアジャイル型開発におけるシステム監査を問う問題が出題されていました。筆者は受験生だったのですが、まさかこんな問題が出るなんて、、、と問題文を眺めながら 1 分間ほど呆然としてしまいました。

世の中、ウォーターフォール開発がまだまだ主流だよな、などと油断してはいけませんね。

【著者略歴】 土屋 俊樹 (つちや としき)



株式会社ハイマックス 第2事業本部第4部 シニアコンサルタント  
主に大手流通小売業向けシステム開発にコンサルタント、およびプロジェクトマネジャーとして従事。近年は人材育成関連研修の企画を担当する。  
情報処理技術者 (AU、ST、PM、AE、DB、SM、SC)  
CITP 認定番号 15000003 シビックテック SIG 副部長

参考文献、および脚注

\*1 「企画・要件定義段階で全体開発規模感を要求される」

経営側からみると既に企画・要件定義段階で投資は始まっています。そしてシステム投資全体予算の確定は要件定義後になるとしても、その投資予算を経営判断として承認するためには、それ相応の準備（根回しと言ったりします）と時間が必要になります。よって、プロジェクト責任者はできるだけ早い段階で全体開発規模を掴み、社内調整する時間を確保する必要があるのですね。

\*2 「ソフトウェア開発データ白書 2018-2019」

発行元 独立行政法人情報処理推進機構 (IPA) 社会基盤センター

<https://www.ipa.go.jp/sec/publish/tn12-002.html>

軽いアンケートに回答すれば、全頁の PDF がダウンロードできます。太っ腹です。

\*3 「CITP アニュアルレポート 2016 現場で活かせ！統計情報」

[https://citp-forum.ipsj.or.jp/2019/01/07/citp\\_report2016\\_tsuchiya/](https://citp-forum.ipsj.or.jp/2019/01/07/citp_report2016_tsuchiya/)

上記レポート掲載後、現場を担う複数の PM の方から「有用だった」との感想をいただきました。感想をいただけたことに感謝するとともに、データ白書という先人の努力の結晶をより実戦で使えるようにすることの大切さを再認識いたしました。

\*4 1人月=160時間 (1人日=8時間 1人月=20日として換算します)

\*5 データ白書では、最少、P25、中央、P75、最大、平均の統計値があります。本書では、感覚としてもっとも現実に近いと思われる中央値を採用しています。

\*6 筆者の個人的な経験ですが、その昔お客様の CIO に「システム開発規模の概算見積は、桁数さえ間違わなければ良い」と良く言われました。もっともそのお客様は年商数千億規模の大会社ではありましたが。

\*7 「成功したプロジェクトの数値」

データ白書 P77 図 4-14-5 プロジェクト成否の自己評価 参照

\*8 「ウォーターフォール型開発が97.4%！」

データ白書 P53 図 4-5-1 開発ライフサイクルモデル 参照

\*9 「100KSLOC 以下の開発規模のプロジェクトは8割弱となっています。」

データ白書 P61 図表 4-8-10 SLOC 実績値の基本統計量 (SLOC で実績値を集計した場合)

**付録** 出典 IPA「ソフトウェアデータ白書 2018-2019」 ※生産性数値は、人月換算しています。

**付録 1**

図表 8-4-41 ●主開発言語別の SLOC 生産性の基本統計量 (新規開発) [SLOC/人月]

主開発言語	N	最小	P25	中央	P75	最大	平均	標準偏差
b : COBOL	133	85	506	781	1,131	3,478	918	3.79
g : C 言語	82	2	496	822	1,307	52,038	1,826	35.99
h : VB	87	144	674	1,158	1,726	25,595	1,702	17.86
o : C#	83	216	595	973	1,406	8,973	1,379	8.89
p : Visual Basic.NET	70	2	667	1,114	1,675	13,560	1,600	12.71
q : Java	415	51	496	797	1,382	14,539	1,237	9.95

**付録 2**

図表 8-4-37 ●業種別の SLOC 生産性の基本統計量 (新規開発) [SLOC /人月]

業種 (大分類)	N	最小	P25	中央	P75	最大	平均	標準偏差
F : 製造業	162	2	666	1,107	1,661	12,179	1,480	9.28
H : 情報通信業	165	2	459	789	1,270	14,539	1,115	8.66
J : 卸売・小売業	88	115	635	963	1,550	13,213	1,320	9.72
K : 金融・保険業	308	51	390	667	1,075	25,595	1,030	11.57
R : 公務 (他に分類されないもの)	93	66	670	1,131	1,472	52,038	2,531	47.48

**付録 3**

図表 7-7-2 ●工程別 SLOC 生産性の基本統計量 (新規開発) [SLOC/人月]

	N	最小	P25	中央	P75	最大	平均	標準偏差
基本設計	265	16	2,784	4,992	8,320	1,299,600	14,752	527.2
詳細設計	265	0	2,688	4,912	8,672	269,312	9,424	130.2
製作	265	0	1,584	2,320	3,968	803,040	6,544	308.6
結合テスト	265	0	2,384	4,176	7,616	173,952	8,032	98.6
総合テスト (ハンダー確認)	265	0	3,856	6,560	13,248	1,195,008	22,592	608.7
開発 3 工程 (詳細～結合)	265	0	672	1,072	1,776	19,840	1,552	11.9

**付録 4**

図表 7-1-16 ●工程別の実績工数の比率の基本統計量 (新規開発) [比率]

工程	N	最小	P25	中央	P75	最大	平均	標準偏差
基本設計	304	0.001	0.115	0.155	0.207	0.568	0.169	0.088
詳細設計	304	0.003	0.129	0.167	0.211	0.884	0.175	0.090
製作	304	0.003	0.260	0.326	0.379	0.870	0.326	0.112
結合テスト	304	0.004	0.149	0.199	0.241	0.446	0.200	0.079
総合テスト	304	0.001	0.073	0.116	0.167	0.507	0.130	0.080

付録 5

図表 7-1-18 ●要件定義工程も含めた工程別の実績工数の比率の基本統計量（新規開発） [比率]

工程	N	最小	P25	中央	P75	最大	平均	標準偏差
要件定義	177	0.001	0.054	0.089	0.129	0.400	0.097	0.064
開発 5 工程	177	0.600	0.871	0.911	0.946	0.999	0.903	0.064

各図表の見方、前提条件、制約条件、集計の方法等は、データ白書本体をご参照ください。

以上。