

知の視点から IE～経営工学の諸技術の 適用を拡大する

日本経営工学会 MOT研究会
2011年 9月17日

河野善彌 Creation Project
Koono@vesta.ocn.ne.jp
<http://www.creationproj.org>

概要

はじめに

第1部 ヒトの知

第2部 ソフト開発工程は線形系

第3部 その他の特性

第4部 システム

第5部 総括

付録 人に倣った

ソフトウェア自動設計

はじめに 1

◎工学の使命：世界～産業～企業/組織～人々に**具体的に貢献**

* IE草創期（19世紀末～20世紀初頭）の状況

産業革命後に各種技術が急速に発達し、工場制で製造が大規模化した。

世界的富豪は王侯から実業家に移る。*100年を経て、何となく似ている。*

Carnegie[鋼] Rockefeller[油] Ford[車], Mellon[銀行]

製造や管理は依然として未熟で、IEはこの難局に立向かった。

* IEとその発展

* IEの最も基礎的な原理はプロセスの**経験則「線形性」**である。

例 F. W. Taylor: 時間研究. F. & L. Gilbreth: 動作研究

* 工数系に品質系も加わりTQMも貢献して高品質な大量生産が実現。

* 従来のモノでないソフトウェアの隆盛時代。**定量化が全く無く、
科学性、合理性、貢献意識が全く無い権威的プロセスが闊歩。SWEng**

◎「知」からは「合理的定量的科学的な統一的体系」がある。

これはIEや経営工学の延長*で世界～産業～企業/組織～人々に**具体的に貢献**できる。

* 河野 善彌, 陳 慧, ハッサン アボールハッサン, IE/経営工学の基礎はヒトの知の構造にある

— 人間知能の定量的な側面 —, 予稿B12, p. 110-112, 日本経営工学会, 2008.

はじめに 2

1. 人は言葉を使い、その概念を具体化するトップダウンな生物です。言葉で具体化されて、各種の作業/指示/文書化が行われます。これを使えば、直接/間接作業更には経営等を共通的に扱えます。この「合理的定量的科学的な統一的体系」をIE～経営工学のエキスパートにご紹介し、忌憚無い御意見/異見を頂きます。
2. 簡略化の為、各種実績資料を総括してご説明します。各出典を注記しました。この総覧は下記のHP中の論文をご参照下さい。
3. 報告の構成

第1部 ヒトの知

第2部 ソフト開発工程は線形系

第3部 その他の特性

第4部 システム

第5部 総括

付録 人に倣ったソフトウェア自動設計

クリエーション プロジェクト URL <http://www.creationproj.org>

「定量的合理的科学的なソフトウェア開発工程論（第1および2部）」

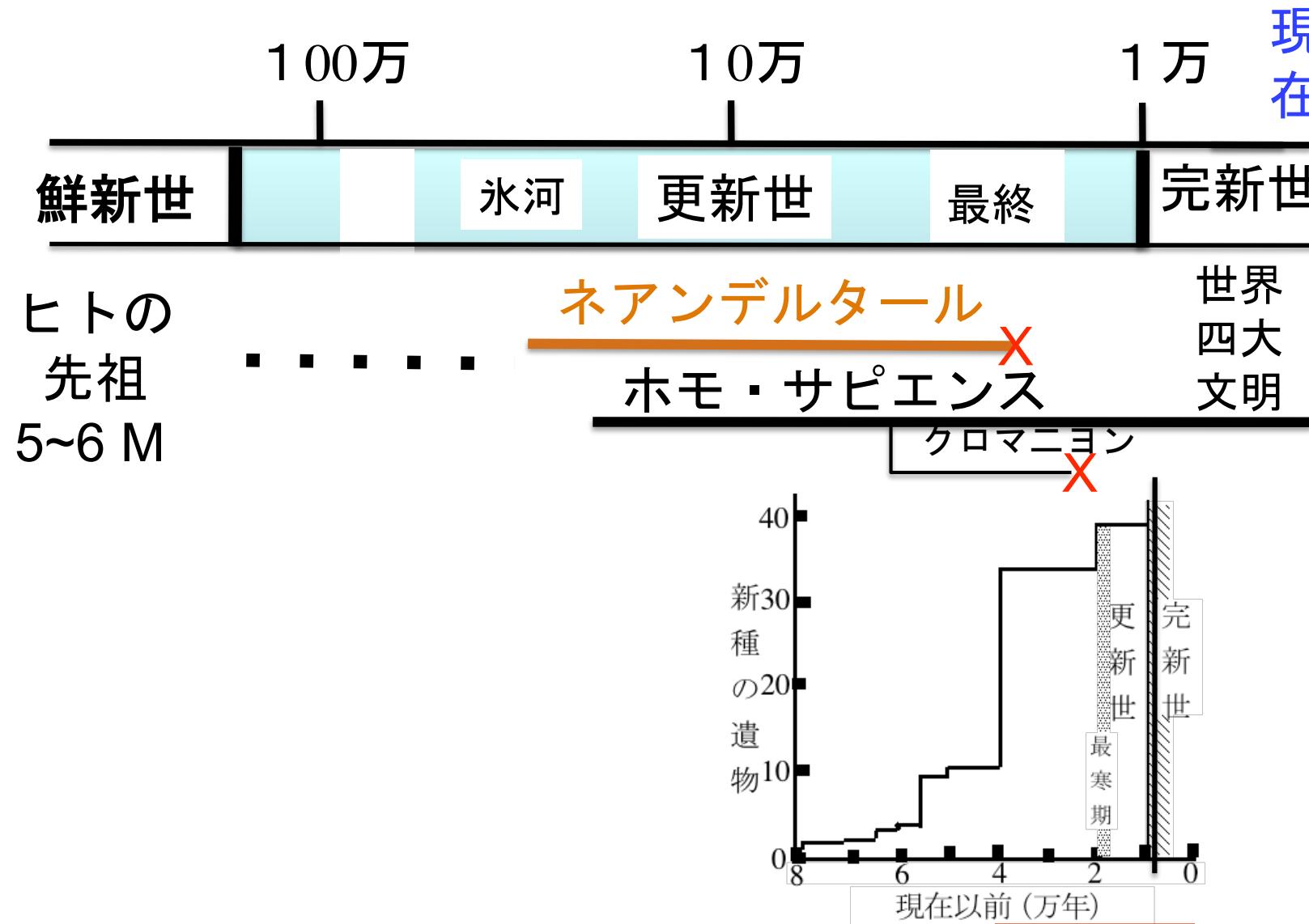
第1部 ヒトの知

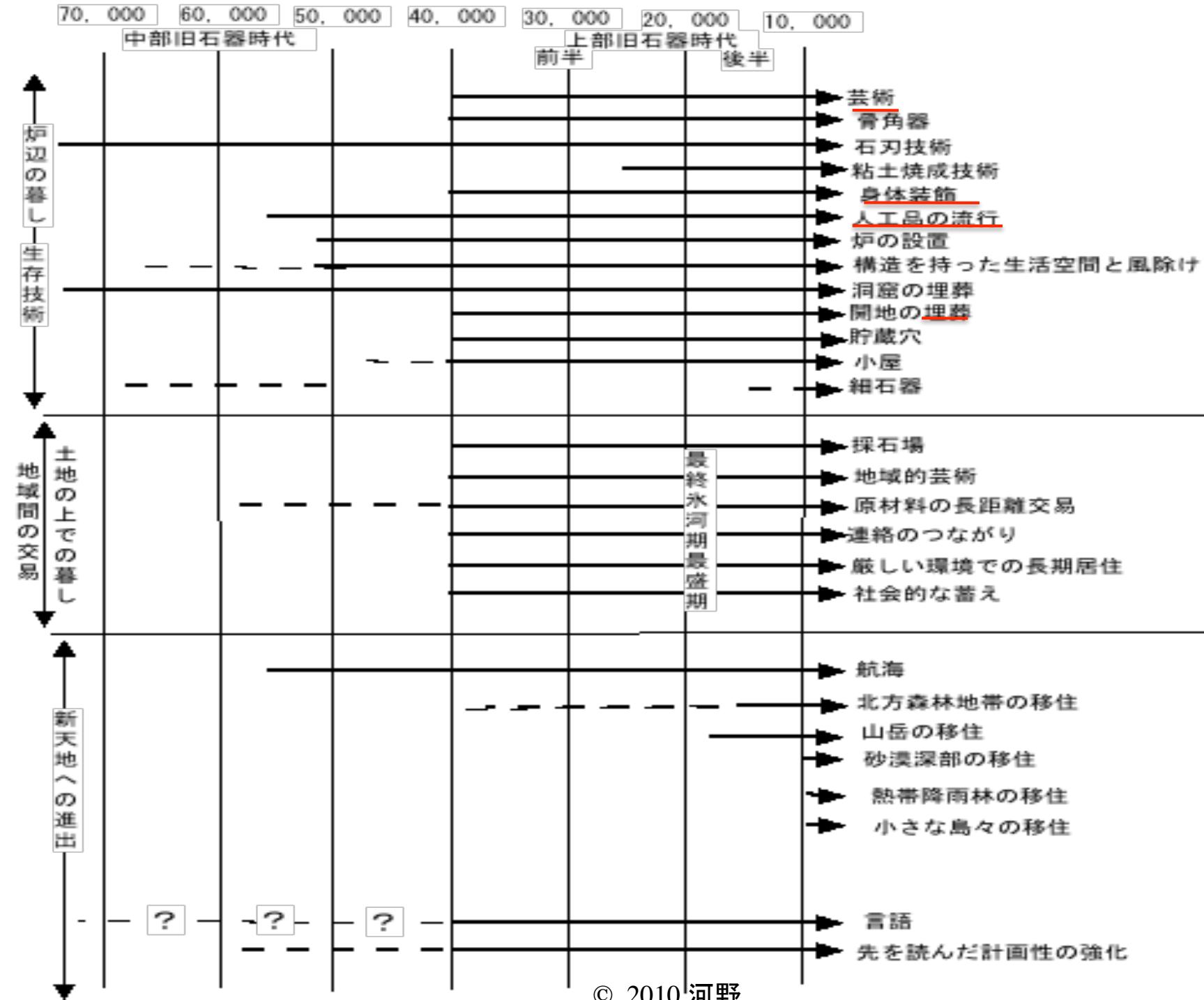
現生人類と他の動物の違い

ヒト	他の動物
個体の生死を超えて、種は進歩を続ける	刷込まれたプログラムで行動する
知の蓄積と再利用 文化/文明特に言語	経験も知も伝わらぬ 突然変異が伝わる

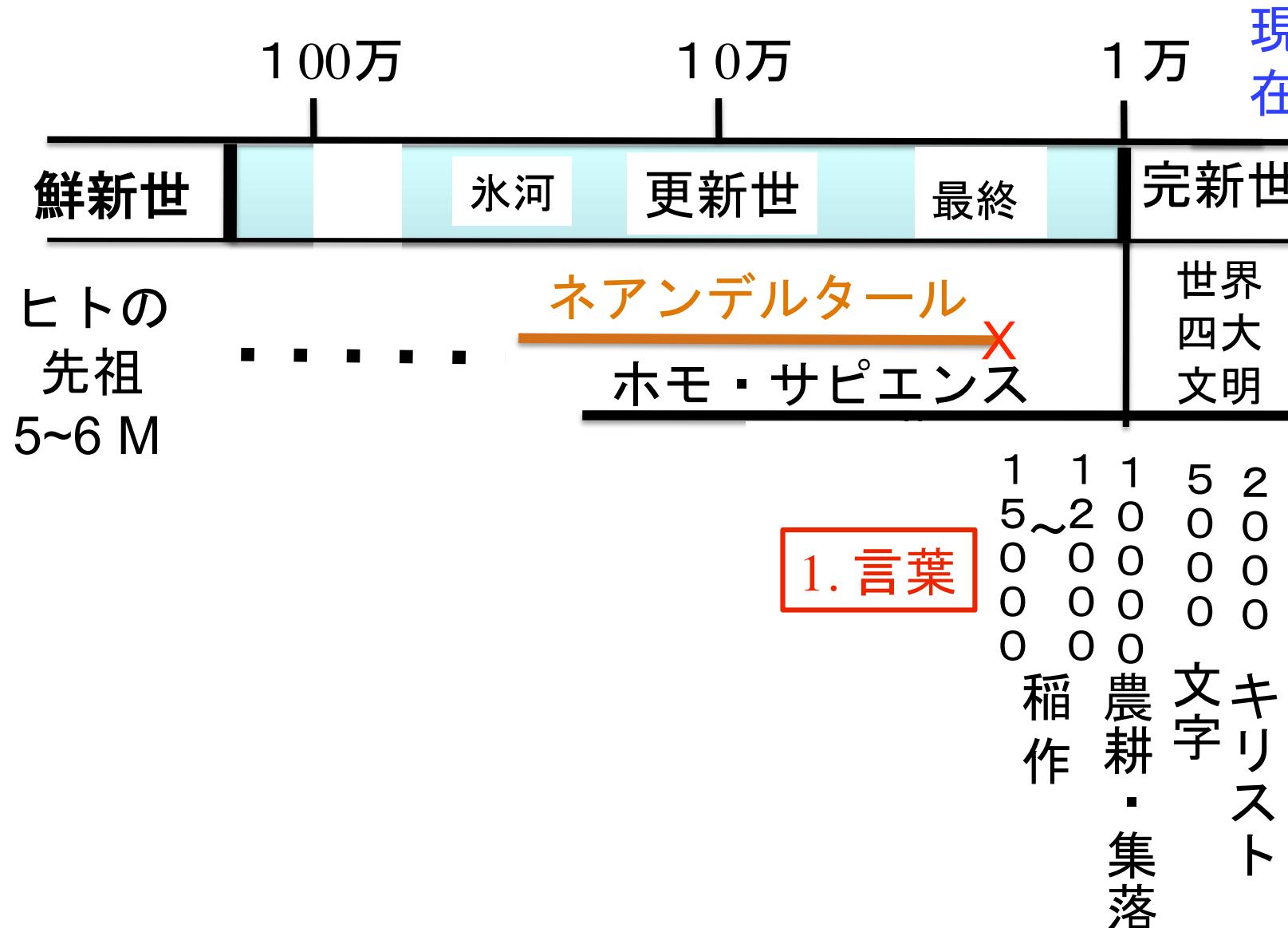
1. 知の誕生
2. Zipfの「労力最小化」原則
3. 語彙能力の発達
4. 意図的行動

1.1 知の誕生





1. 1 知の誕生



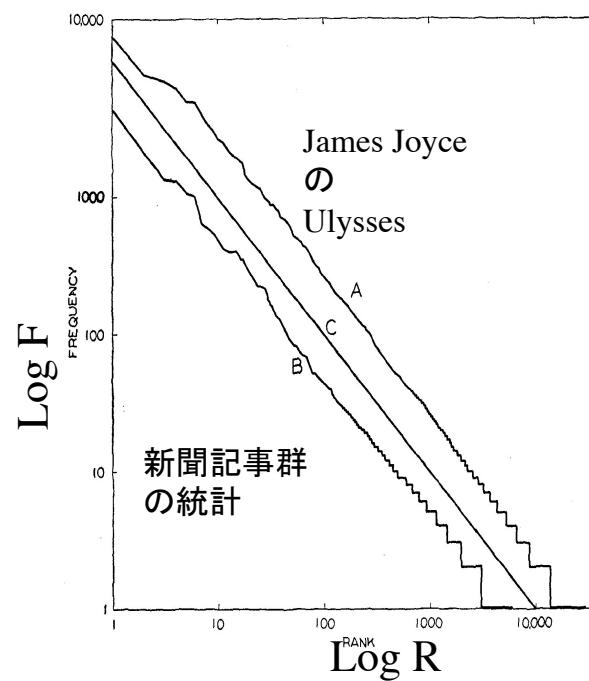
1. 2 Zipfの「労力最小化の原則」

2 「労力を最小化」する。G. K. Zipf : 言語, 人間工学

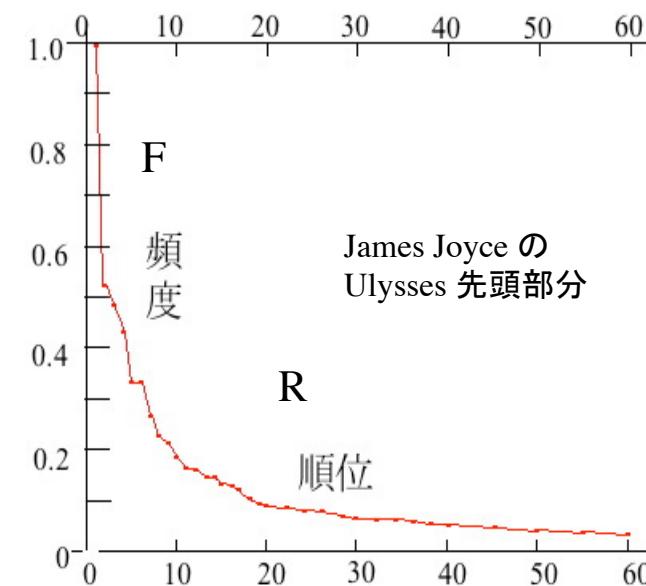
- a. 話者の経済性 1単語に収め文を簡単化する力が働く.....語の使用頻度を増加
- b. 受手の経済性 1単語を1意味に簡単化する力が働く.....語の出現頻度を減少
- c. 人は労力を最小化するように、前記2項のバランスを取って話す。

経験則{頻度順位R × 出現頻度F=一定}を見出し、上記原則を証明した。

Zipf, G. K., Human Behavior and The Principle of Least Effort, Addison-Wesley, 1949.



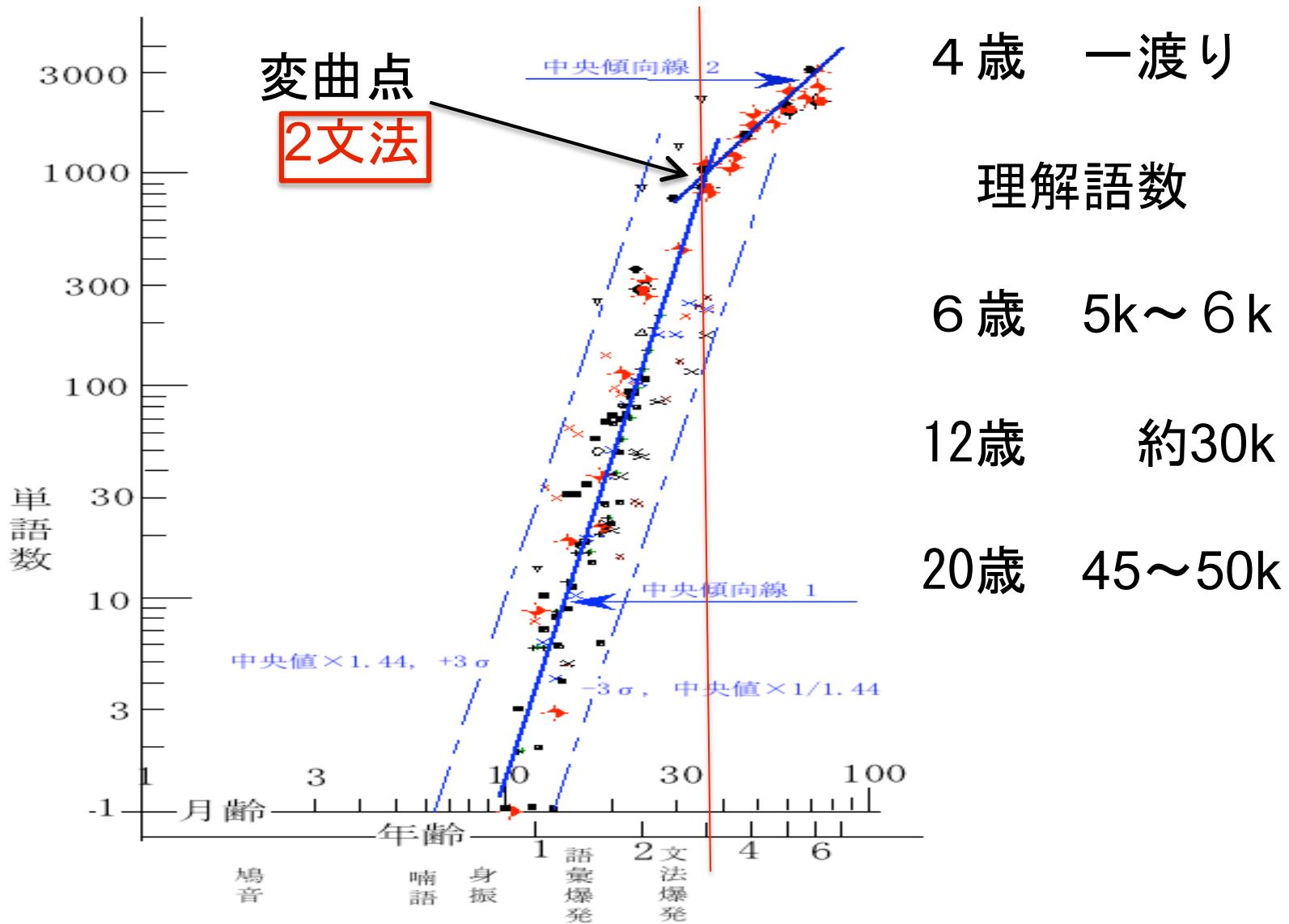
品質分野では上位の少数項目が過半を占めるParetoの法則が基本で、右図のとおり。これは全領域で成立つ。



設計では意識・無意識に多数の選択を行う。意識的に行うには必ず定量評価して決める。定量評価を行わない改善=0。

1. 3 語彙能力の発達

資料一覧	
岩淵 1968,	+
大久保1967, Y児	●
矢田部, 1956	▲
大久保 1984,	×
中西A	×
B	×
C	×
D	×
久保 1914,	◆
駒林1964,	◆
エリコニン 1960,	◆
シュミット	◆
アルキン	◆
澤柳 1919,	◆
ナイス嬢 1917	◆
前田 1983,	◆
B	▼
M	■
O	■
T1	■
若林・細井 2006,	○
N. Lund 2003	○



1. 4 意図的行動

知のレイヤー

上層 心、モラール、意欲等の抽象度の高い理性の上部構造

中層 大脳の新皮質による理性的行動、（遊びを除く）

下層 喜怒哀楽等の情動、脳に刷り込まれた古い部分、本能的。

ヒトの意識のモデル

ヒトは目的概念から、その実現手段概念への展開を繰返す。

1. 空腹 → 2. ラーメンを食べる → 2. 横町の角の店

意図的行動 3

上流 軍事科学 戦争計画法：目的の階層性

中流 構造化設計 拡張構造化設計

下流 ハード製造 「工程」の階層性

第2部 ソフト開発工程は線形系

1. 実績資料 {工数 \propto (作業規模)^{1.00..}}

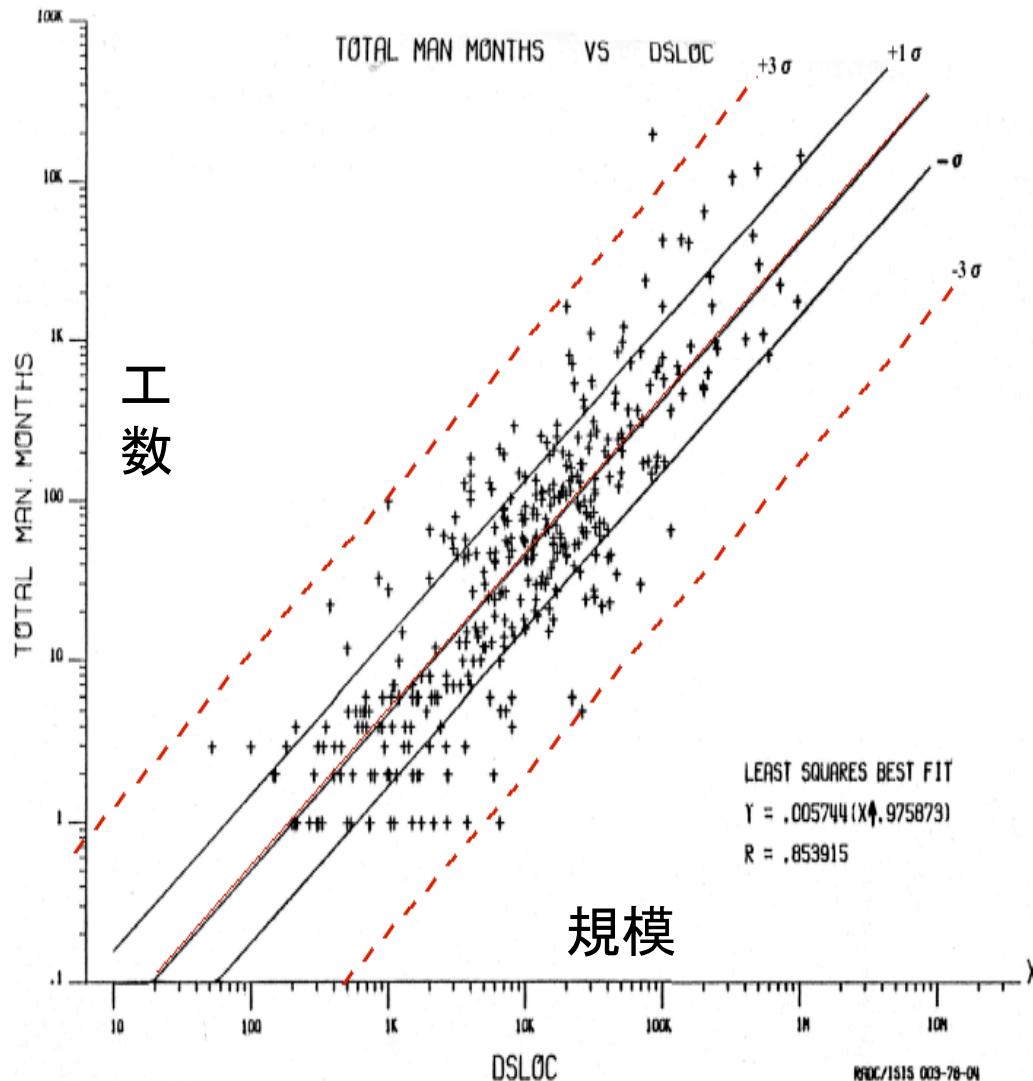
- 1. NelsonのRADC資料
- 2. COCOMO資料, 富士通資料
- 3. 参考 : 工数等の定量的取扱

2. 何故線形性か? 意図的行動の特性

- 1. 最上流 : 経営
- 2. 最下流 : 身体的行動
- 3. 中流 : 設計
- 4. 人の設計手順 (\rightarrow 自動設計)
- 5. 拡張構造化設計の特質
- 6. 定量的特性

2. 1. 1 NelsonのRADC資料 *

米空軍の407開発例の統計処理結果は線形性を証明($Y \propto X^1$).



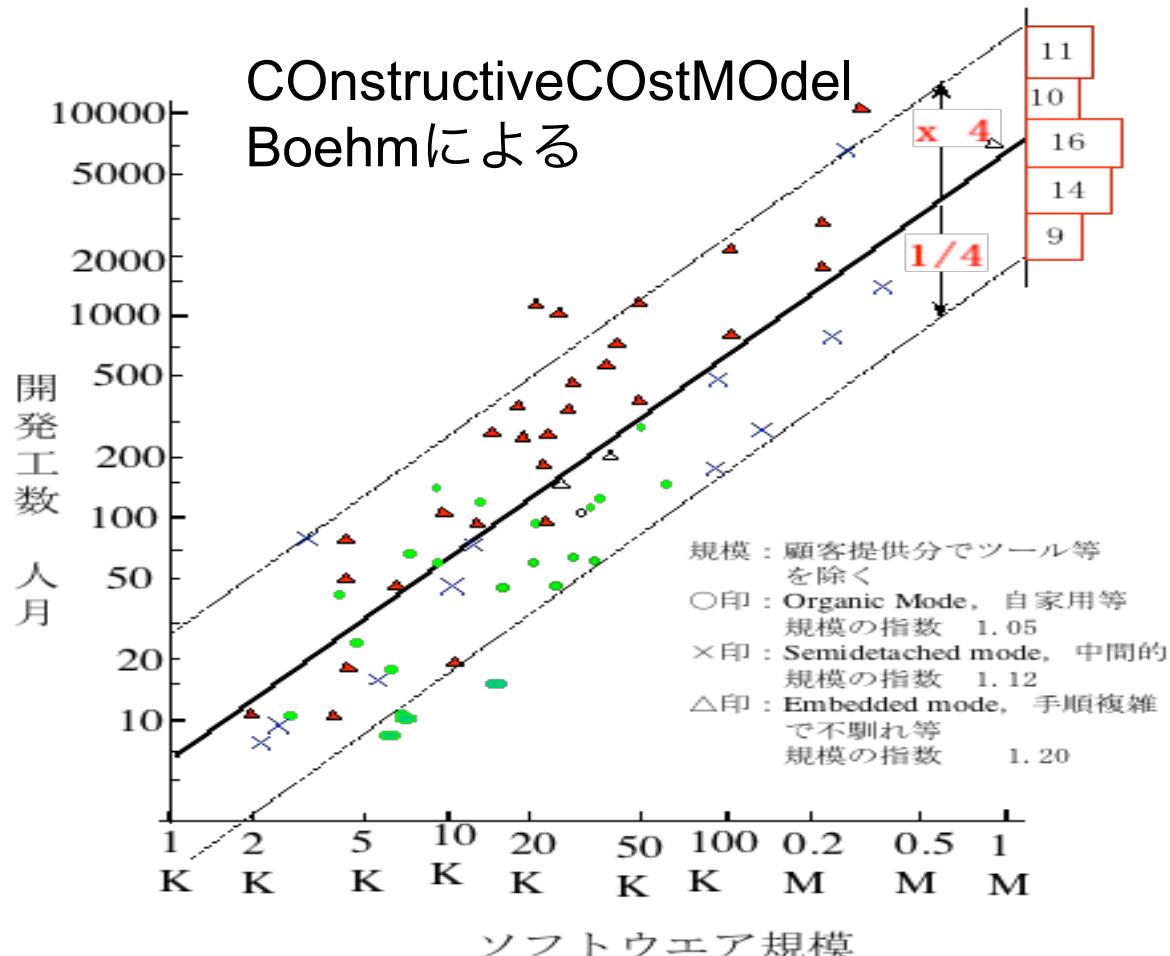
- DSLOC : Delivered Source Code 作業規模である行数.
- 中央傾向線の指数項 $0.975873 \approx 1 \therefore$ 線形性.
- $\pm 3\sigma$ の赤破線は河野追加
- 両対数用紙上で帯状に分布する.
- 図の $3\sigma/\text{平均値} = N$ は 20.

* Richard Nelson, Digest of Software Data Collection And Analysis, 1978, Rome Air Development Center, Air Force Systems Command, Griffis Air Base, NY 13441. 現在はこの(Draft Partial Report)としてコピー残欠しかない. 米 Western Michigan University の図書館にあるのみ. コピーを IPA SEC にお預けしてある. 「コピー<1/2」の原則の為, 各資料の属性などは含まれていない.

2. 1. 2 COCOMO資料[#]と富士通資料^{*}

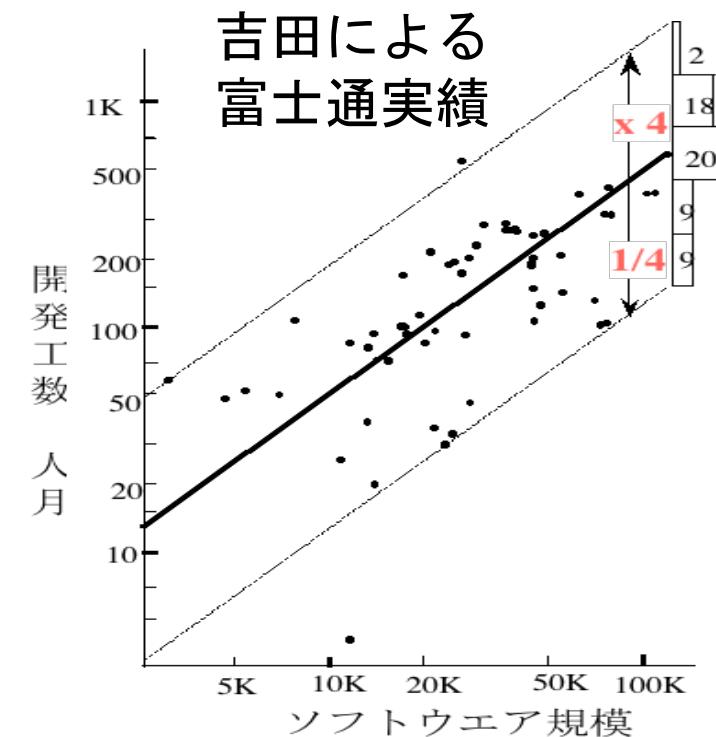
中央傾向線は帯状で指数≈1. 打点は両対数表示で帯状に分布.
帯状領域を5区分し点数を計測表示すると釣鐘状になり対数状.

*Z. Koono, H. Chen and B.H. Far, Expert's Knowledge Structure Explains Software Engineering, Joint Conference on Knowledge-Based Software Engineering 1996, pp. 193 – 197, 1996.



Boehm, B. W, Software Engineering Economics, Prentice Hall, 1981. 中の図の
XY軸を変換し各種傾向線を削除.

© 2011 河野



* 吉田征, ソフトウェアの計量化 : 事例と
実感に違いはないか (統計的な検証) 情報処理26. 1, p.48-51, 1985. (藤野他, パネル討論会: ソフトウェアメトリックスの現状と課題
[昭和58年前期第26回全国大会報告]) 中の図
2, 直線尺度を対数尺度化.

2. 1. 3 参考：工数等の定量的取扱

本研究：全ソフトウェア開発は線形系で展開/統合できる。

例) 過去実績を基準として生産性を求め規模から工数を求める。

SwEng: * COCOMO資料の3種。

種毎傾向線で特性を近似する。

規模から総工数を求める。

受託開発業では▲印。

* 先行例の工程毎工数比率で各工程に工数を配分する。

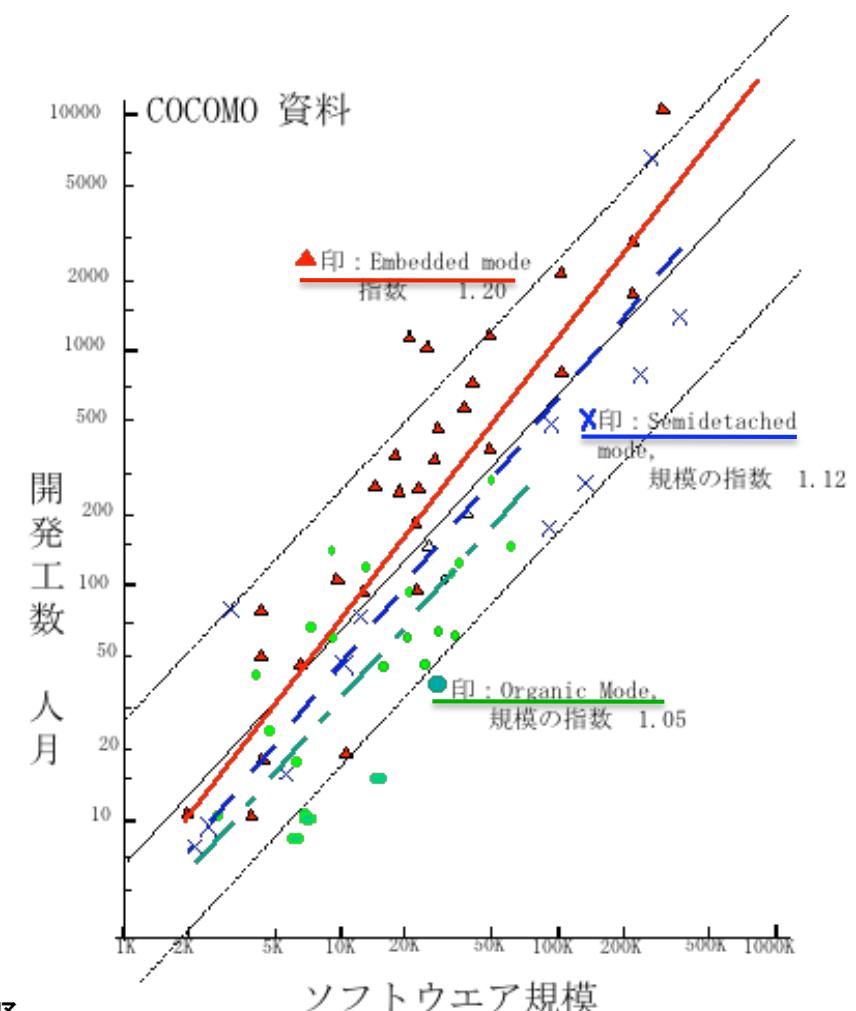
(巨視的～工学的な近似策)

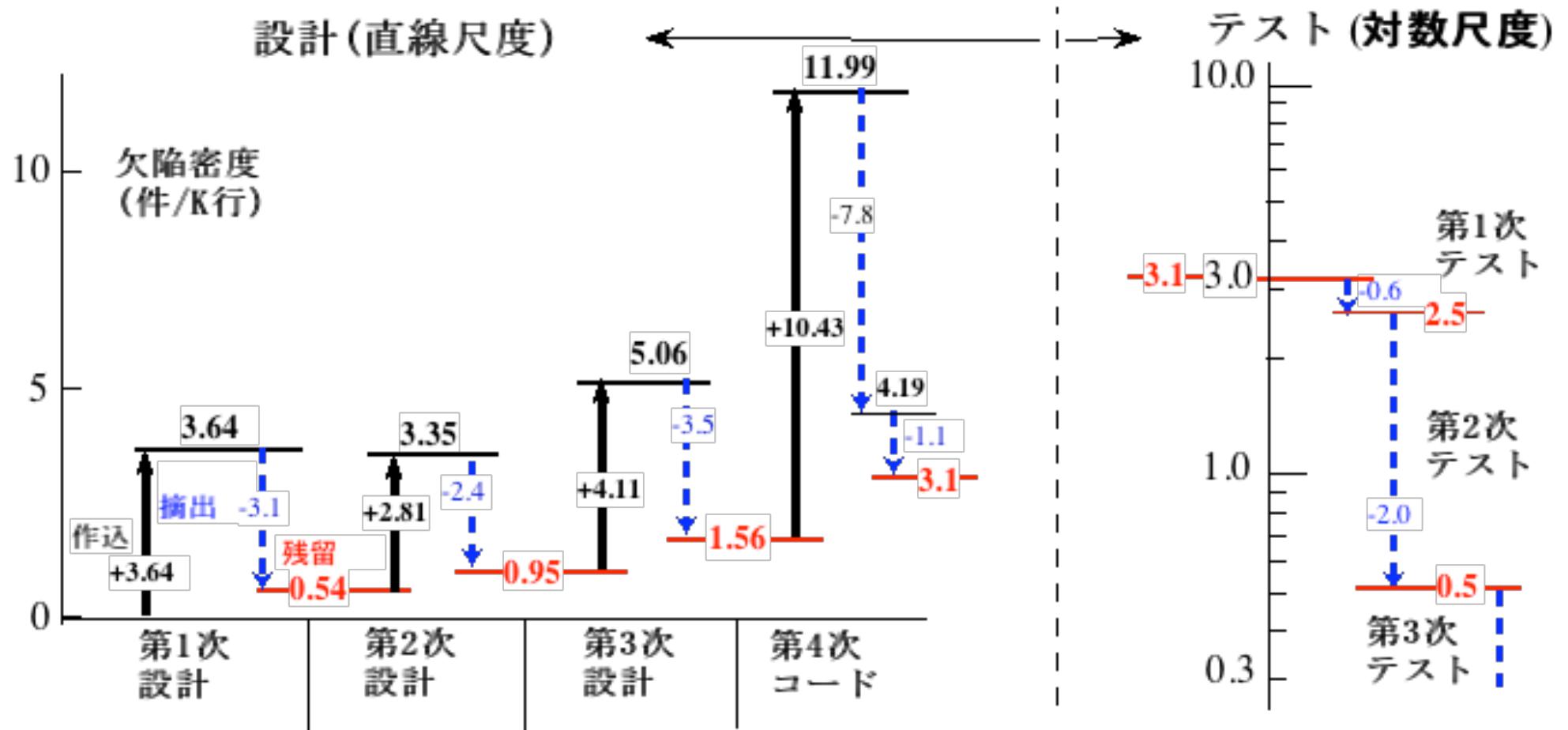
* 自己実績に基づかない。

* プロジェクトは割当工数を忠実に消費する。

* 定量化無しで工学か？
対外説得不可/効果/原因の評価や改善等が有り得ない。

●受託開発で条件が変る。
技術進歩は売上減少に繋がる





設計は誤り作込↑と
机上チェック↓

テストは
減衰機構

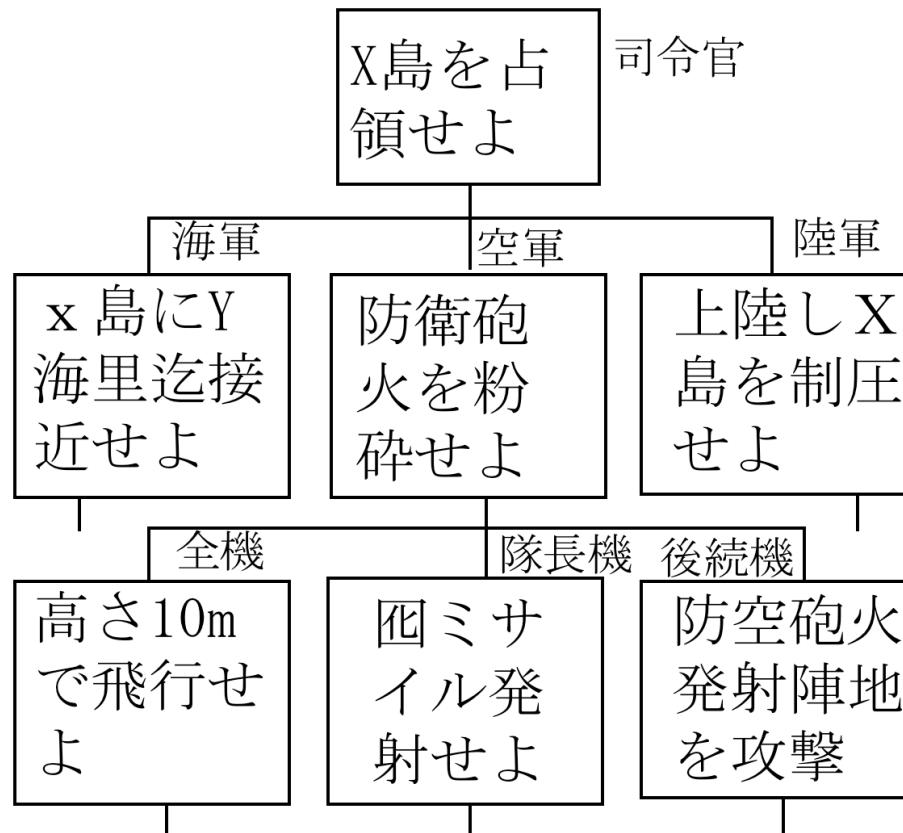
+++

第2部 ソフト開発工程は線形系

1. 実績資料 {工数 \propto (作業規模)¹⁾
 1. NelsonのRADC資料
 2. COCOMO資料, 富士通資料
 3. 参考 : 工数等の定量的取扱
2. 何故線形性か? 意図的行動の特性
 1. 最上流 : 経営
 2. 最下流 : 身体的行動
 3. 中流 : 設計
 4. 人の設計手順 (\rightarrow 自動設計)
 5. 拡張構造化設計の特質
 6. 定量的特性

2. 2. 1 最上流：経営

ヒトは目的を思うと、その実現手段群を想起する。
階層的展開を続け、脳内言語で詳細化する。
詳細化した脳内言語＝脳内言語から各実現手段に移す。



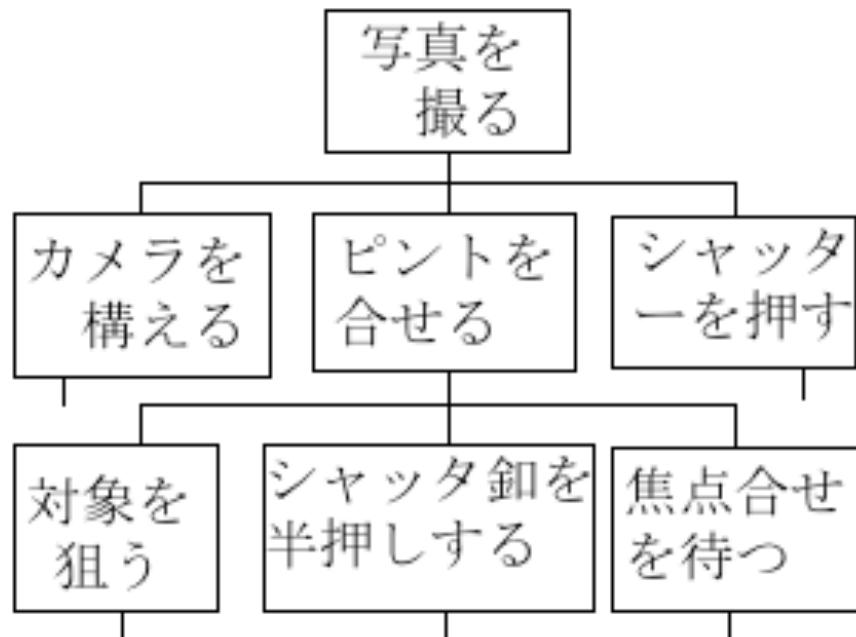
- プロシア軍事哲学Clausewitz(1832)*は過去の戦績の分析から得た経験則を提唱した。軍事科学ではこれを戦争計画の原理とし、「目的の階層性」と名付けた。
- 階層展開連鎖の最末端は兵の1拳動
- 最上位は「戦略」として高級将校に教育し、必勝の戦略の樹立を求めた。その最下位は兵卒の単位的行動になるから、これを具体化した行動マニュアルを作り徹底的に訓練する。

* Carl von Clausewitz, Vom Kriege, 1832. 訳、淡徳三郎、戦争論、徳間書房1999.

2. 2. 2 最下流：身体的行動

工程（プロセス、身体的行動）の階層性：

工程は、下位の工程群に展開できる。逆に該下位工程群を統合すれば上位の原工程に復旧する。展開する度に微小化するから、最後は単位的な筋肉動作に帰する。ハード製造作業の基本原則。

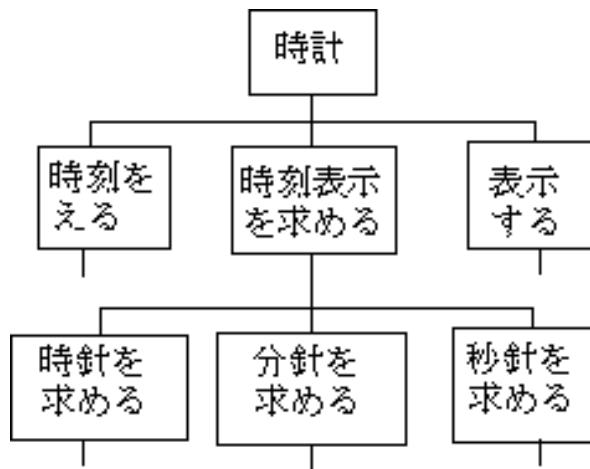


- 工程は(その両端面が定義されているとして), その両端面で定義する。言葉で行動を定義すると, 言葉で言葉を定義するようになり, 混乱する。通常規格や図面で定義。
- あるレベル以下は, 通常は言葉では意識できなくなる。この場合には反射的に作動である。小脳は脳内言語で階層展開し続けていると推定。
- 階層性がヒトの全行動を制御する。目的実現を目指しTop downに行動。

2. 2. 3 中流：設計

設計は最上流の経営と最下流の身体的行動の中間の行動.

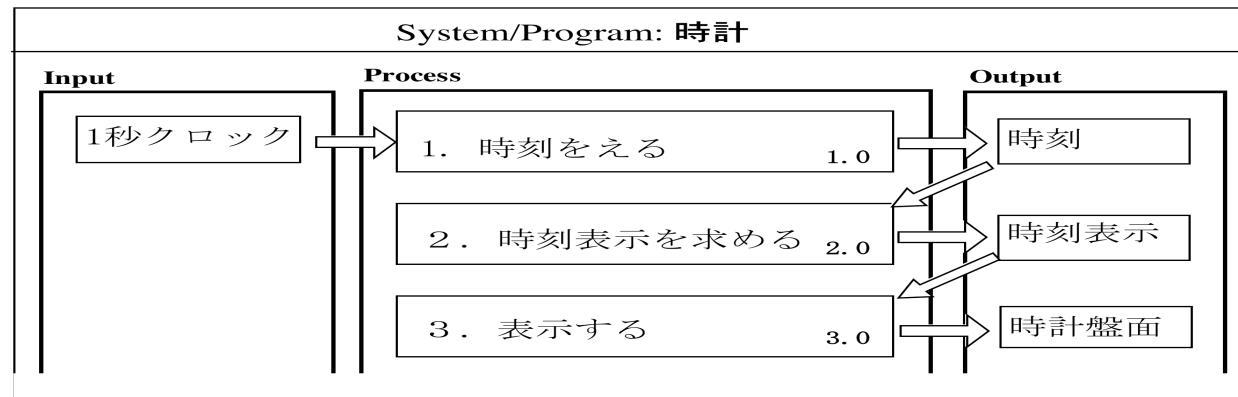
構造(化)設計 ('70-'80)
機能を階層展開せよ



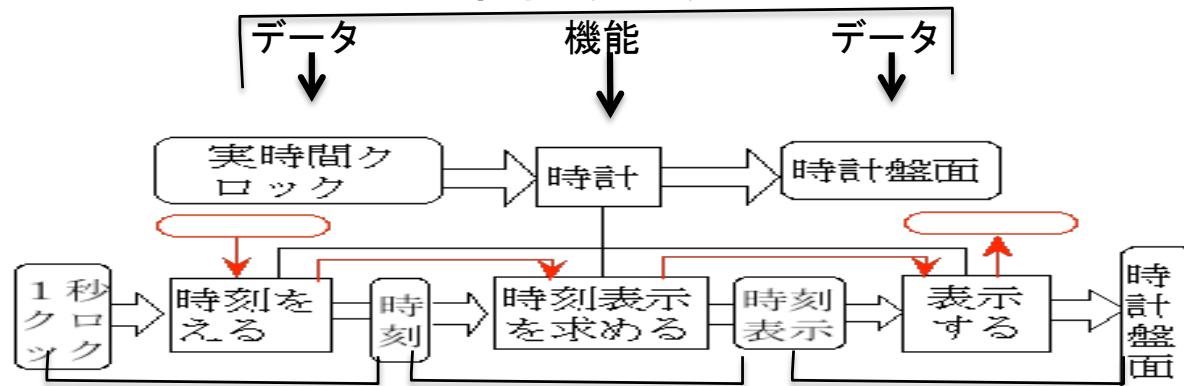
- ・常用されるが、厳守はされない。
- ・正しい分割が困難。
- ・**プロセスラミング** ~~自然言語~~
- ・悪い分割をすると、ソース規模が増大。

Hierarchy plus Input-Process-Output*
分割が容易化した。この展開図でプログラマ化。

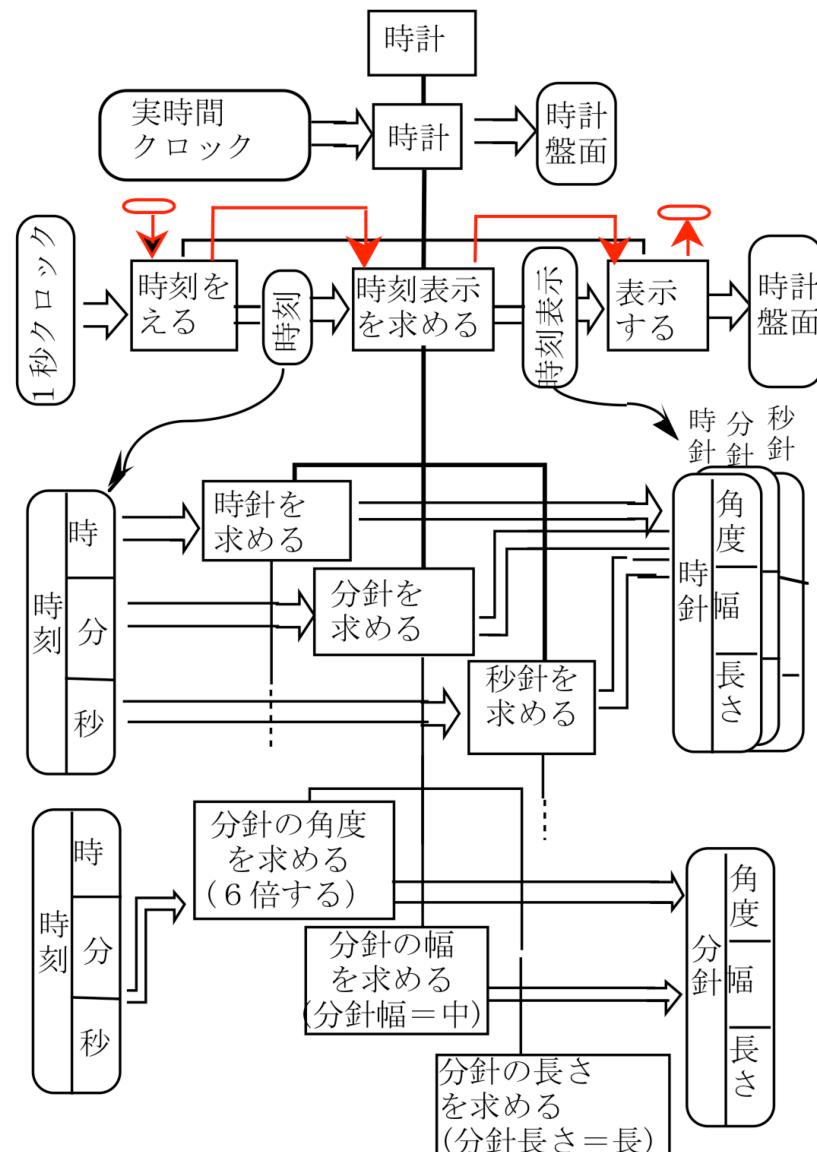
*IBM, HIPO-Design Aid and Documentation Technique, GC20-1851-1, IBM, 1975.



拡張構造(化)設計* 単位概念に限る
単位データフロー

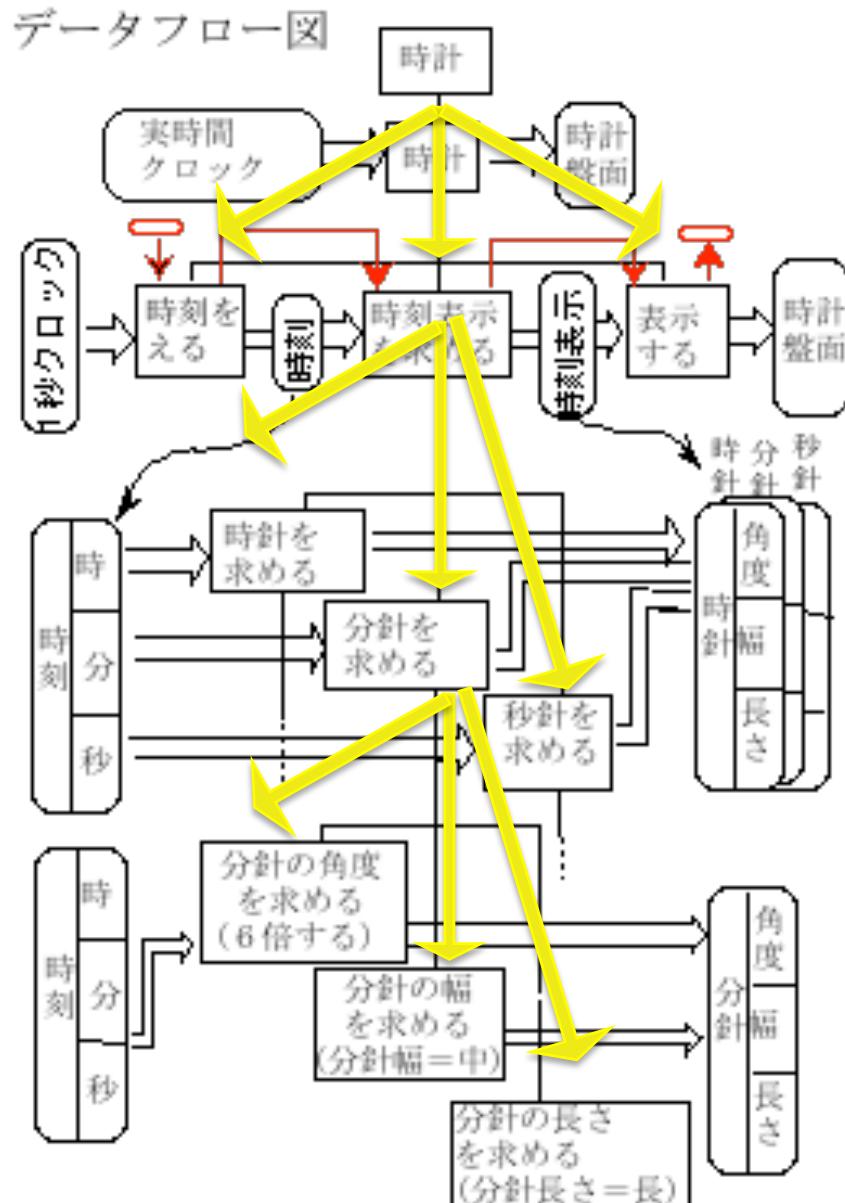


2. 2. 4 人の設計手順 (\Rightarrow 自動設計)



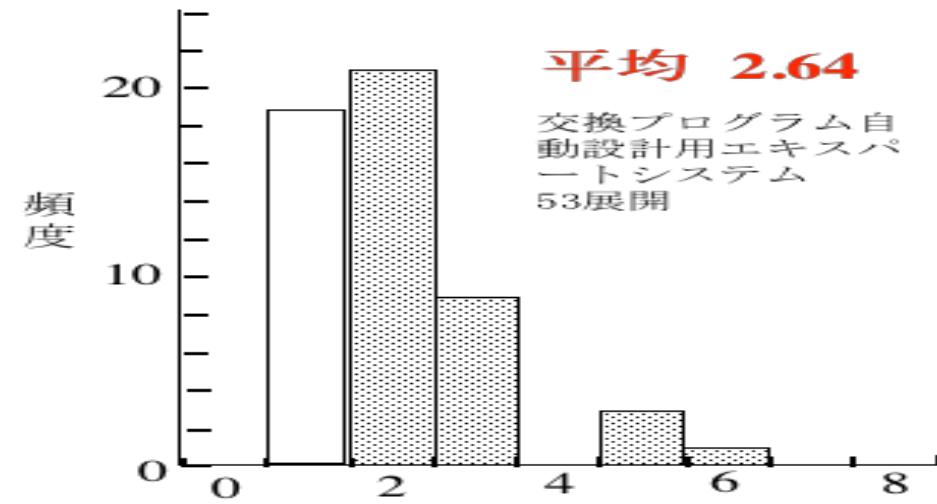
- 機能「時計」に入力/出力データを付加し親概念「単位データフロー」にする。
- 該データフローを展開し「詳細化データフロー」+Fc (Flowchart)にする。「詳細化データフロー」は(約3の)子単位データフローになる。
- 各子単位データフロー毎に展開し前記(詳細化データフロー+Fc)を付加することを繰返す。
- 経営問題もハード設計/製造もできる。
- 最末端の「分針の角度を求める」は、[(分針の角度) × 6倍]に置換し、実現手段に移行する。プログラムは本質ではありえない。
- 各展開を自動化すれば、自動設計できる。(ハード設計自動化同一)

2. 2. 5 拡張構造化設計の特質

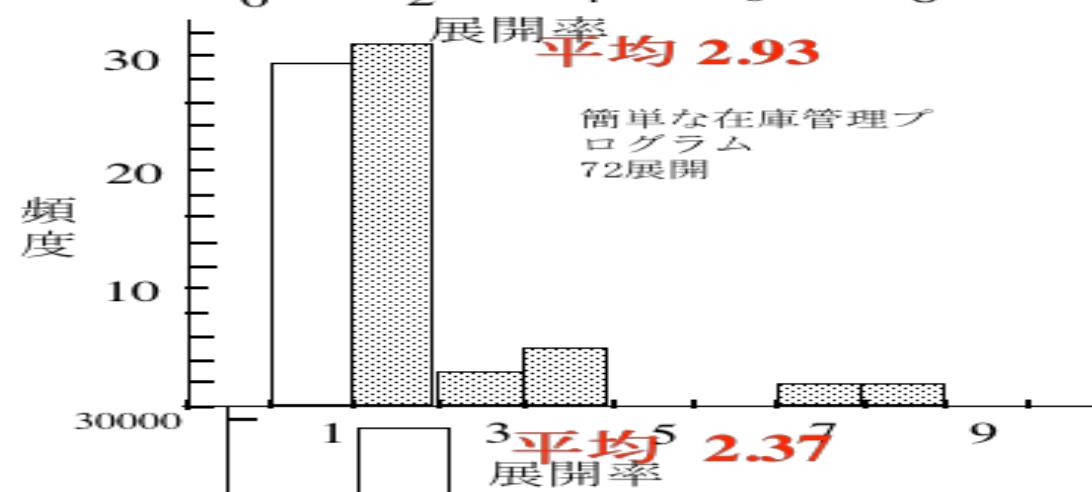


- ・全ての意図的行動（設計）に適用
- ・良い設計になる
单一概念/正確～明解な言葉
(科学技術用語を含む)
- ・脳内言語は自然言語で可
科学技術用語を含む
- ・展開毎に明確化/具体化/詳細化
展開率 3弱 推定 $e = 2.71828$.
- ・等価な実現手段に移行する
例：Codeや論理ゲートで可
- ・親概念一子概念群は単位知識
∴系は全て線形系

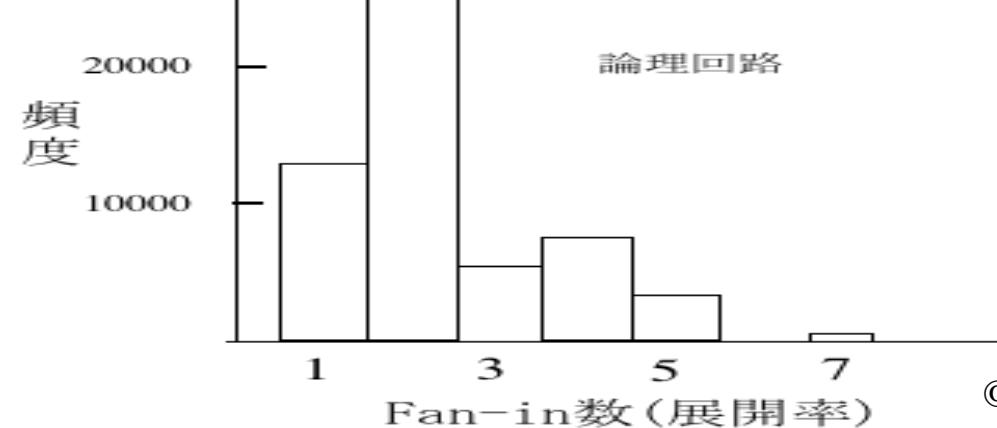
Soft 1



Soft 2

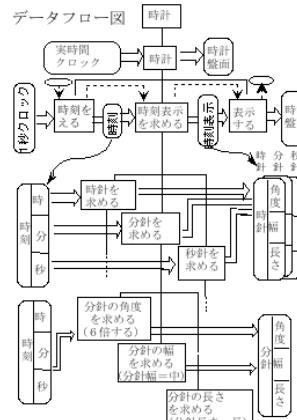


Hard



2. 2. 6 定量的特性

意図的行動

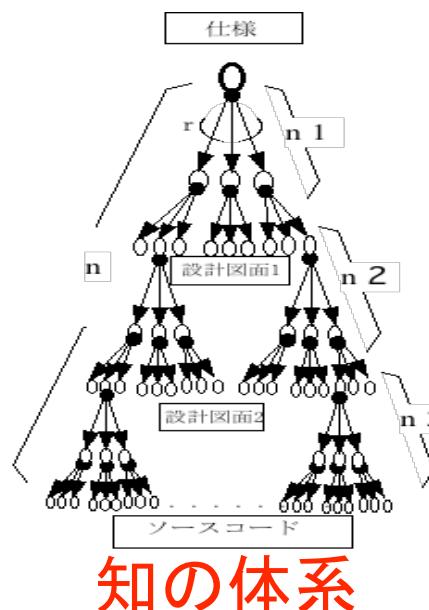


ヒトの意図的作業は線形系である
IEの諸経験則を理論的に裏付けた

定率展開する階層展開網モデルを考える

○印：情報 ●印：処理 定率で展開する
展開率一定なら等比級数公式で定量評価可能

階層展開網モデル



$$\text{生産性} \propto \frac{\text{知的処理総数}}{\text{最終出力数}} \times \tau$$

τ : 処理当りの微小時間

$$\text{誤り率} \propto \frac{\text{知的処理総数}}{\text{最終出力数}} \times \varepsilon$$

ε : 処理当りの微小誤り率

*Z. Koono, H. Chen and B.H. Far, Expert's Knowledge Structure Explains Software Engineering, Joint Conference on Knowledge-Based Software Engineering 1996, pp. 193–197, 1996.

第3部 その他の特性

1. V型網

- 1. ソフトとハード開発と製造
- 2. 管理のV型網

2. 誤り（欠陥）とテスト

- 1. 誤りの作込み(Thayers資料)
- 2. 誤りの謎 1, 2
- 3. テストは定率の減衰機構
- 4. 減衰特性の実測例
- 5. 誤りとテストの構造
- 6. 作込み工程と再発防止

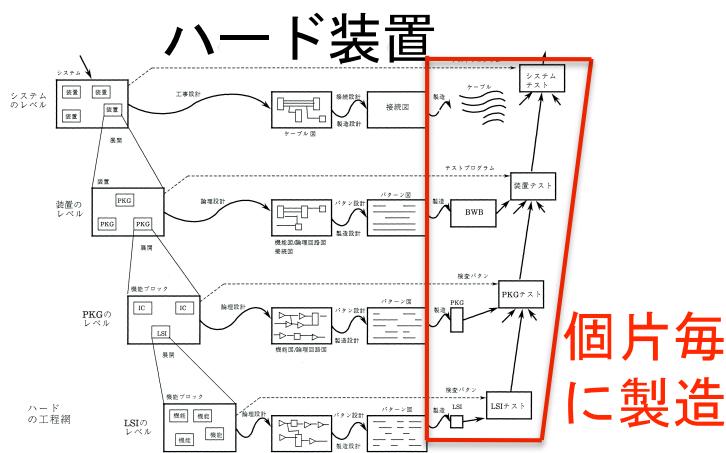
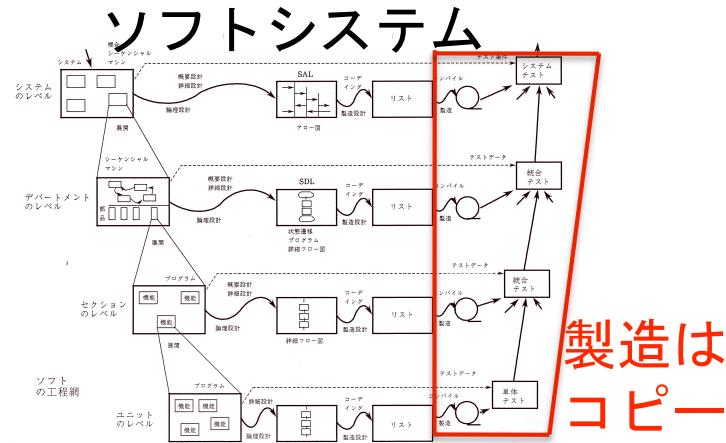
3. 対数正規分布

- 1. 対数用紙上の帯状分布
- 2. 規模も大きなバラツキ
- 3. 対数正規分布
- 4. プロセス（工程）の純化
- 5. プロセスの進化

2. 習熟効果

- 1. 対数習熟効果
- 2. 習熟（学習）効果は知の集積
- 3. 効果を習熟工学で定量評価
- 4. 他の習熟効果例

3. 1. 1 ソフトとハード開発と製造



V型網でのソフトとハードの差異
両者は階層的に展開し統合される。

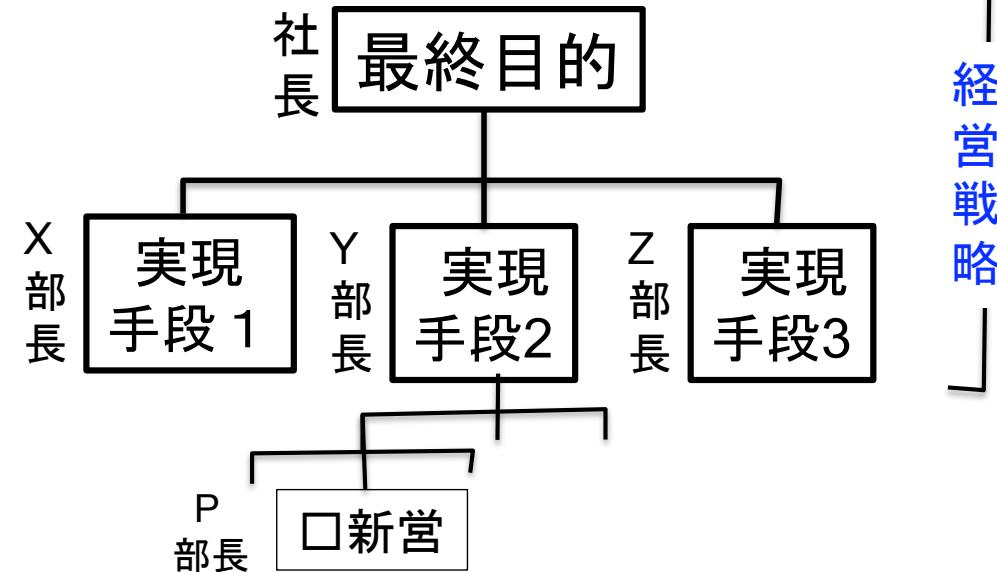
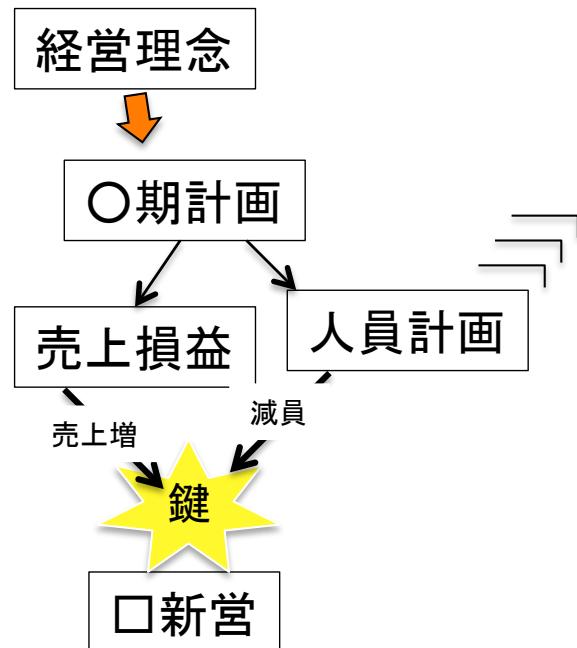
△ 赤枠はN出力を得る製造である。

- ・ソフトでは自動的にコピー可。
- ・ハードでは対象物毎の加工が必要。単作業化し伎倆の資格者が行なう。欠陥0。人手作業部分は工数を要し品質も下がる。自動挿入、基板半田付なら略ソフト同様。

△ V型の左設計で、ソフトでは人作業。ハード設計同様に極力自動化し、区分毎にシミュレーションすれば生産性も品質も大幅に向上する。ハードと同レベルに頑張ろう。

* Koono, Z. and Soga M., Structural way of thinking as applied to quality assurance management, IEEE Journal on Selected Areas in Communications, Vol. 8, No. 2, pp. 291-300, 1990.

3. 1. 2 管理のV型網



組織：階層制が原則、巨視的な最適化。

社長：全責任を負う。執行の為に最終権力を持つ。

基本：「目的の階層性」（意図的行動）に準拠。

経営の「最終目的」を立て、「実現手段」を各部門長に課す。

期毎方針→期毎計画を立て遂行させる

ハード/ソフト設備投資（新宮）もその一環

設備投資の推進 ハード/ソフト同一である。

第3部 その他の特性

1. V型網

- 1. ソフトとハード開発と製造
- 2. 管理のV型網

2. 誤り（欠陥）とテスト

- 1. 誤りの作込み(Thayers資料)
- 2. 誤りの謎 1, 2
- 3. テストは定率の減衰機構
- 4. 減衰特性の実測例
- 5. 誤りとテストの構造
- 6. 作込み工程と再発防止

3. 対数正規分布

- 1. 対数用紙上の帯状分布
- 2. 規模も大きなバラツキ
- 3. 対数正規分布
- 4. プロセス（工程）の純化
- 5. プロセスの進化

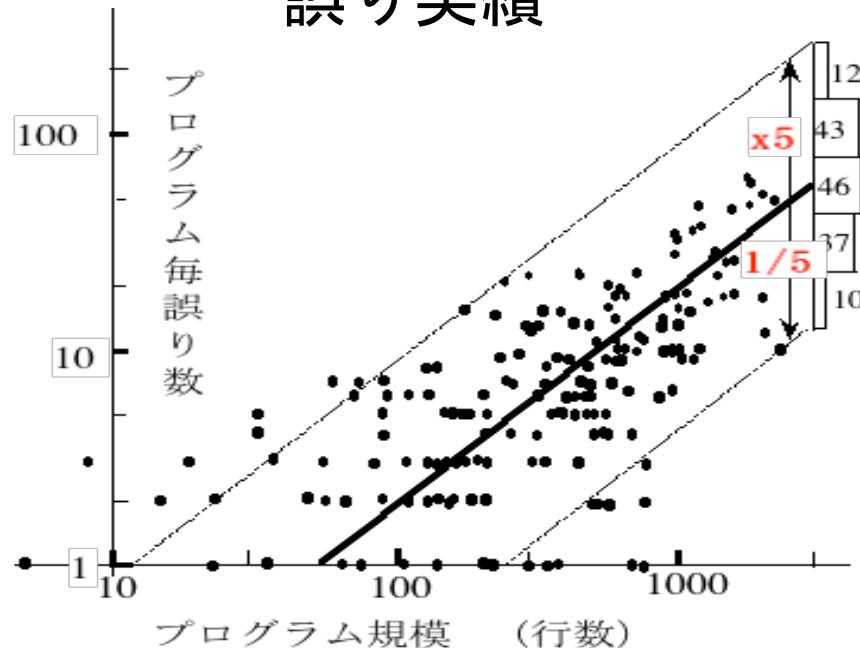
4. 習熟効果

- 1. 対数習熟効果
- 2. 習熟（学習）効果は知の集積
- 3. 効果を習熟効果で定量評価
- 4. 他の習熟効果例

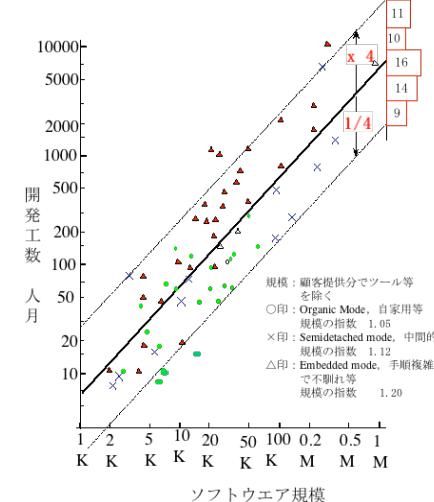
3. 2. 1 誤の作り込み(Thayers資料)

誤り率も同様な線形性がある。それは工数の線形性と同じ原理で、処理当たり微小確率で処理を誤る。

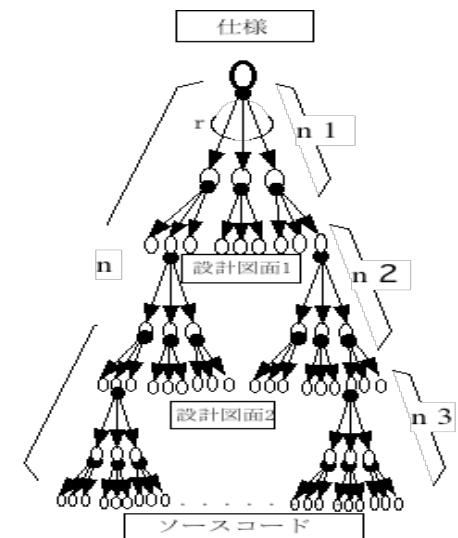
Thayersの第3プロジェクト
誤り実績



Thayers, T. A., Software Reliability Study, RADC-TR-76-238, Final Technical Report Aug 1976, Rome Air Development Center, Air Force Systems Command, Griffiss Air Force Base, New York 13441. 中の第3プロジェクト中のFig. 4-11の直線尺度を対数尺度に変換。本資料は、くたびれたコピーがBritish Libraryにある。

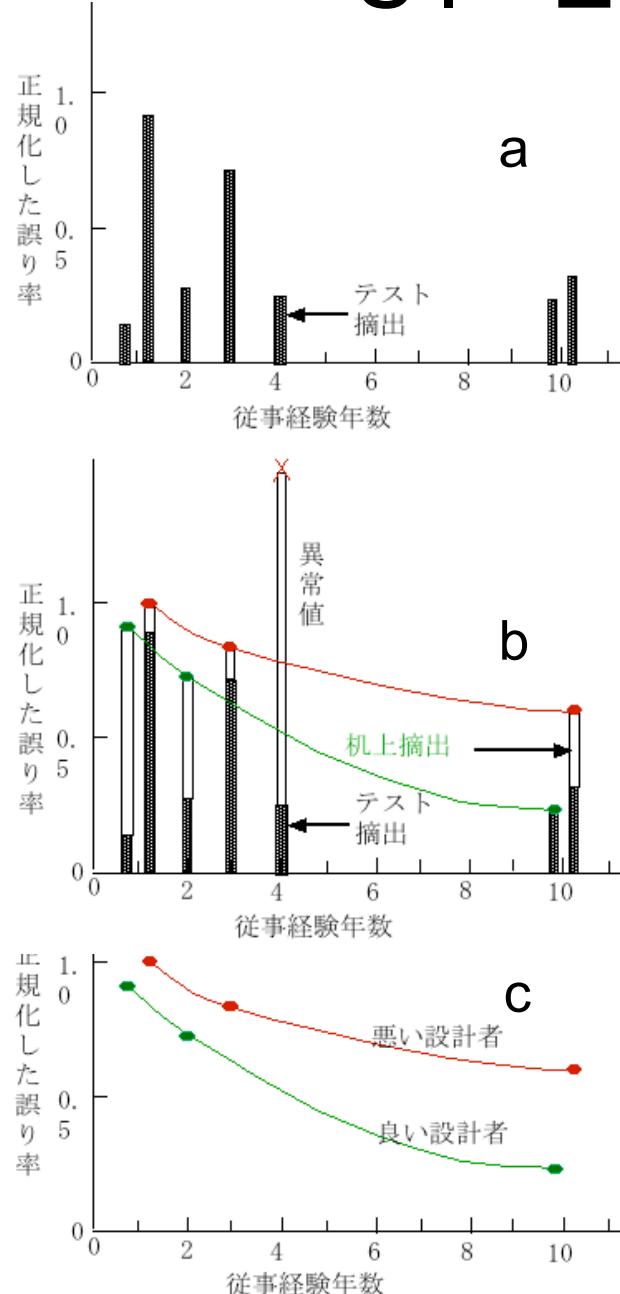


階層展開網モデル



*Z. Koono, H. Chen and B.H. Far, Expert's Knowledge Structure Explains Software Engineering, Joint Conference on Knowledge-Based Software Engineering 1996, pp. 193–197, 1996.

3. 2. 2 誤りの謎 1



テスト実績例（プロジェクトXチームY）

横軸：ソフト従事経験年数

縦軸：縦軸，ある値で正規化

？全く傾向が掴めない？ 謎だ！

悩んだ末、そのチームリーダに相談した。
彼は「原資料からデータを再発掘」。
人毎資料中からテスト前の机上抽出記録
を見出し、白棒グラフに再現した。

アッ！！系統性が見える！！

リーダ「緑線は能力の高い人、赤線は
能力が低い～この領域では新人。」
「これは当たり前と云うことか！」

リーダ「初心者には易しい問題を与え、経験者
者には難しい問題を与え、平均して図の
カーブになるように調整している。リーダ
依存度が高い。」

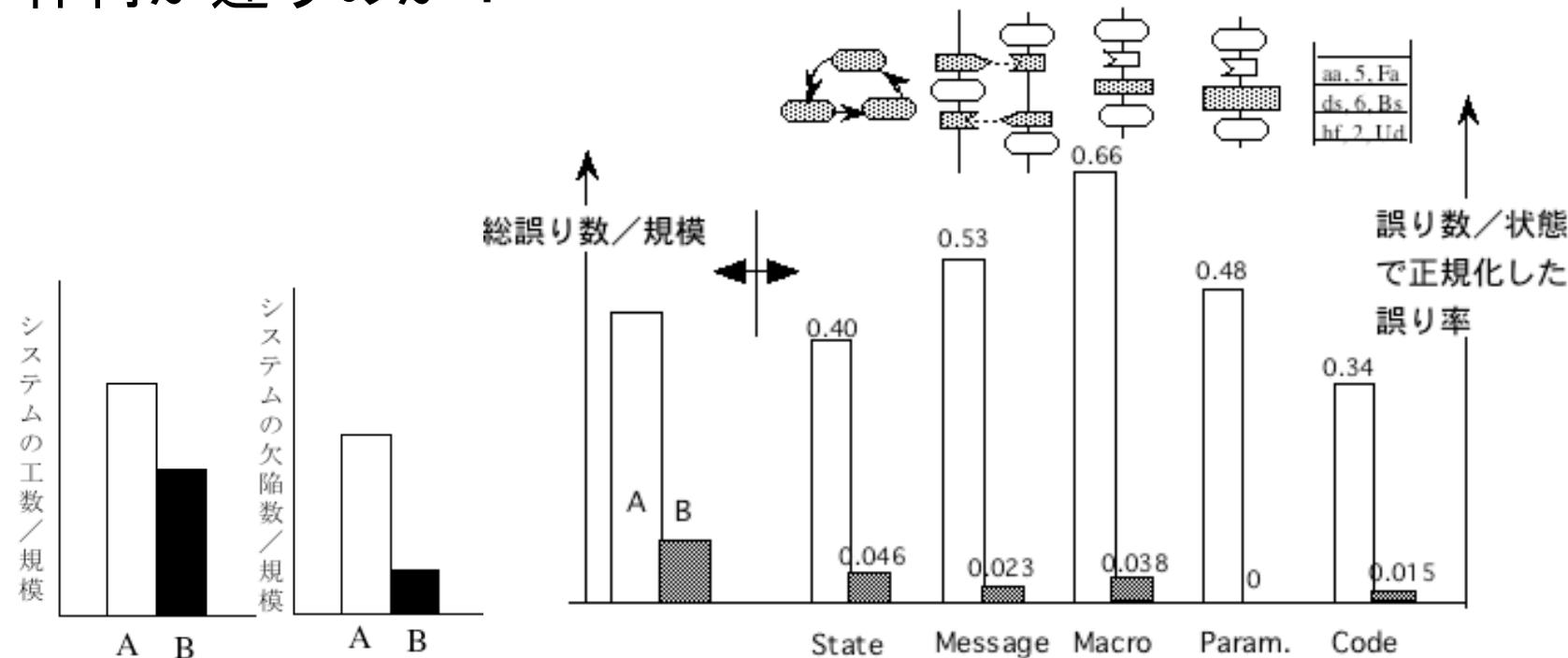
3. 2. 2 誤りの謎 2

同一architecture～process の2系でシステムSを構成した。

Aは終始要員不足、テストでバグ多発、大ピンチ！

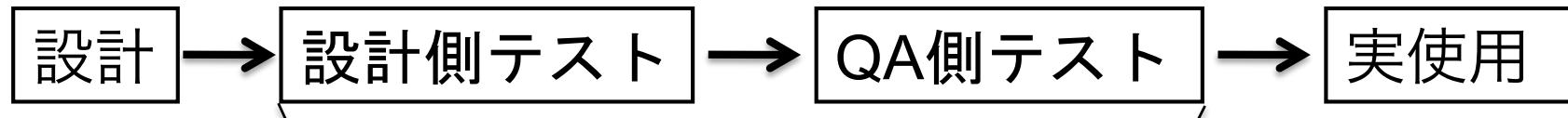
Bは、素々と作業して早々と撤収。

一体何が違うのか？

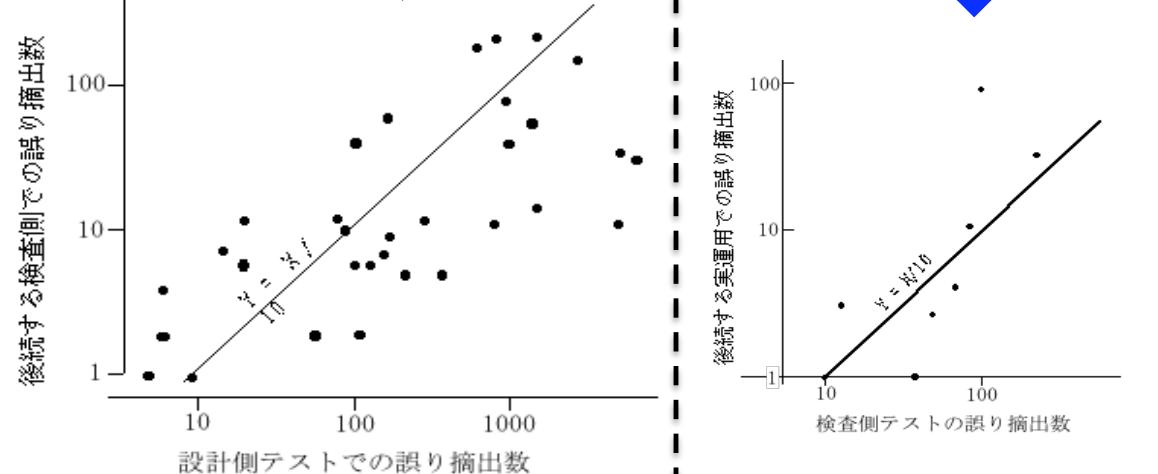


Bリーダは、細かいステップ毎に入念に机上チェックさせた

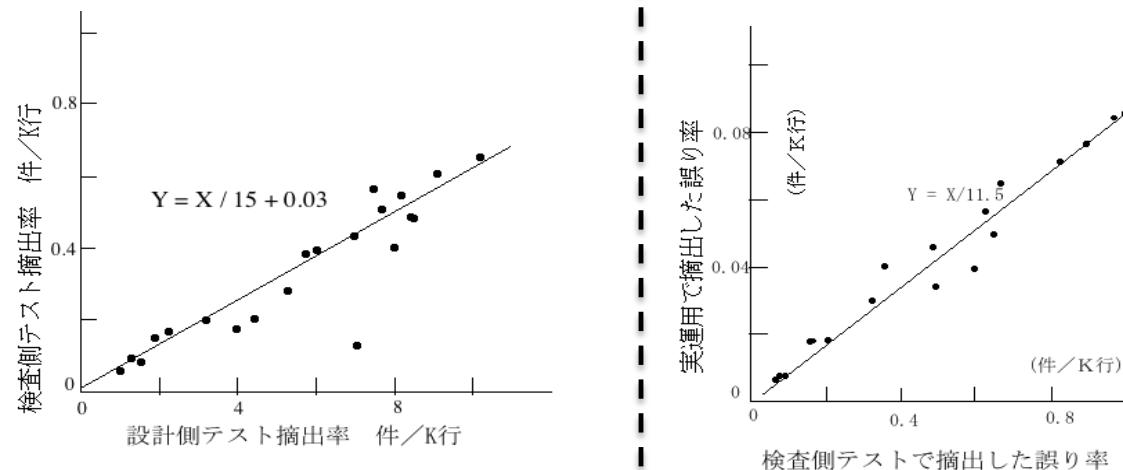
3. 2. 3 テストは定率の減衰機構



初回



後続



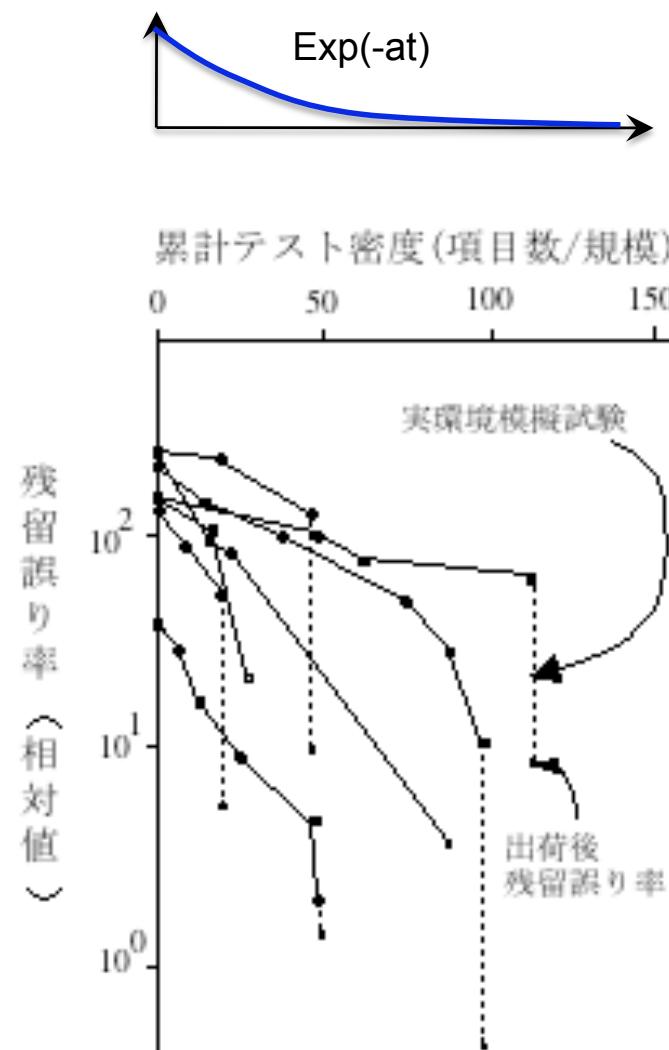
・最上段はある実例
ハードでの経験則
製造品質∞稼働品質
を参照し相關2例を得た。

初回は1:10の相関があり
バラツキが大

続いて1部署に限定し
再調査、バラツキ激減。
比率の詳細も取れた。

定率の減衰機構。
プロセスのバラツキ
結果のバラツキ

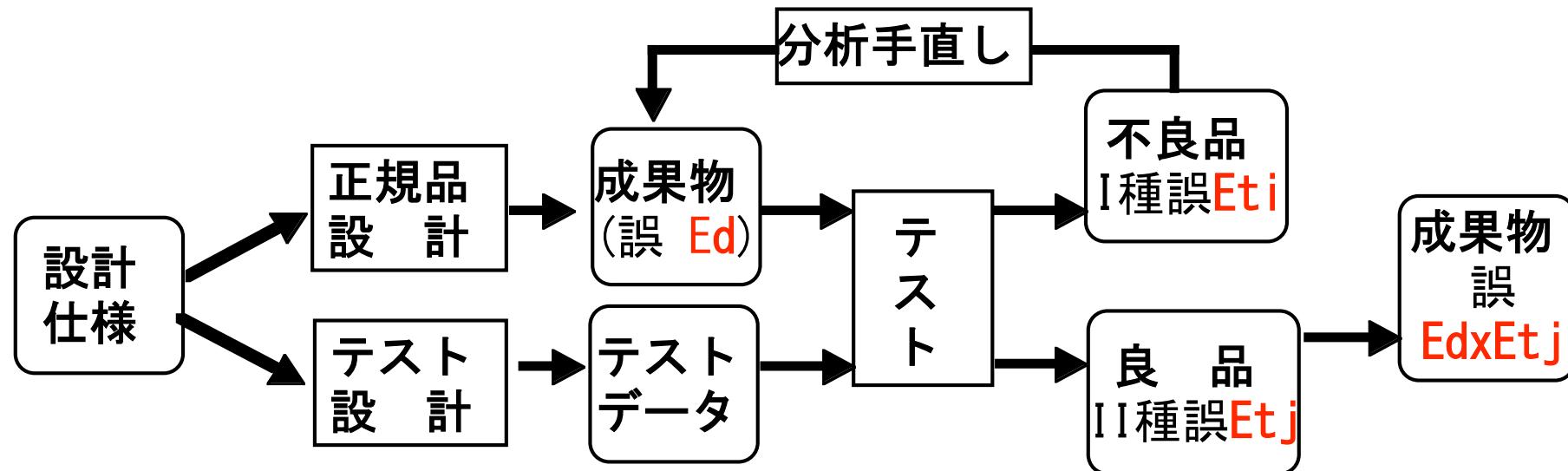
3. 2. 4 減衰特性の実測例



- ・横軸はテストした項目数を累計し、最終規模で除した累計テスト密度。縦軸は残留欠陥密度。
- ・傾向線は直線状。テスト項目数を増せば、残留誤率は低下する。
- ・勾配を「テストの有効度」と呼ぶ。テストの定量管理が可能になる。
- ・負の小さな勾配の線群はアセンブラー期システム、大きな勾配の線群はコンパイル言語、構造化設計、階層的文書のシステム。
- ・最右下点は出荷後残留欠陥密度、破線は実環境模擬テストでの抽出欠陥密度、後にはテストを細分化して最終テスト項目化して消した。最左上点の座標（0テスト、テスト開始時残留欠陥密度）から $\Delta X = \text{実行テスト数/規模}$ と $\Delta Y = \text{抽出欠陥数/規模}$ だけ右下に移行し、最後に（累計テスト密度、出荷後残留欠陥密度）に至る。

3. 2. 5 誤りとテストの構造

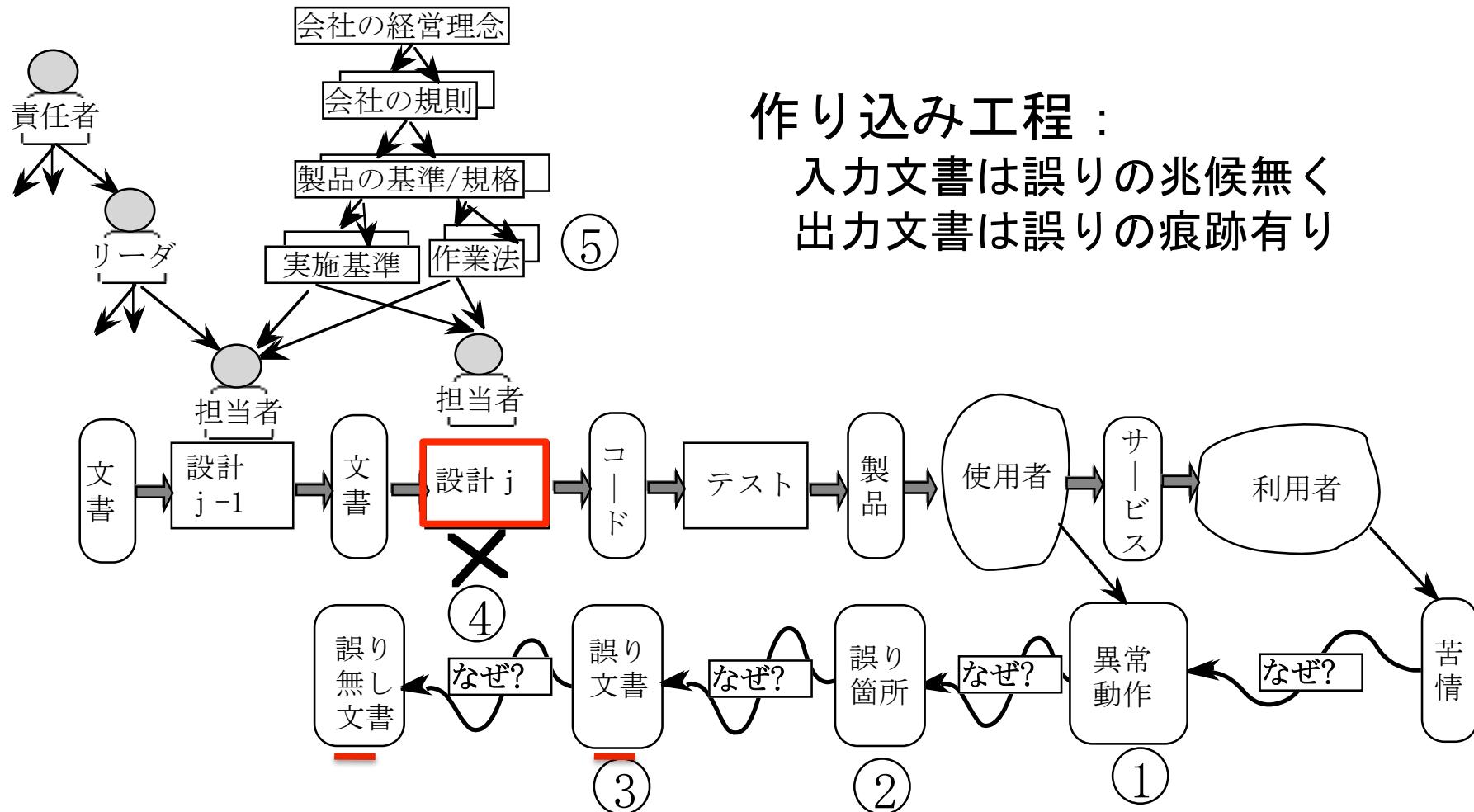
テストは誤りをEj倍に低下する定率減衰器である。
減衰を増すには、テストを増すか、関係する全作業
をより正確にする。



設計成果物は「作込による誤り率Ed」を含む。
テストは誤りを含む。II種(良→不), II種(見逃し)。
テスト成果物は誤認により「誤り率Ed × Ej」を含む。

3. 2. 6 作込み工程と再発防止

なぜ？なぜ？を工学的な手順にする



第3部 その他の特性

1. V型網

1. ソフトとハード開発と製造

2. 管理のV型網

2. 誤り（欠陥）とテスト

1. 誤りの作込み(Thayers資料)

2. 誤りの謎 1, 2

3. テストは定率の減衰機構

4. 減衰特性の実測例

5. 誤りとテストの構造

6. 作込み工程と再発防止

3. 対数正規分布

1. 対数用紙上の帯状分布

2. 規模も大きなバラツキ

3. 対数正規分布

4. プロセス(工程)の純化

5. プロセスの進化

4. 習熟効果

1. 対数習熟効果

2. 習熟(学習)効果は知の集積

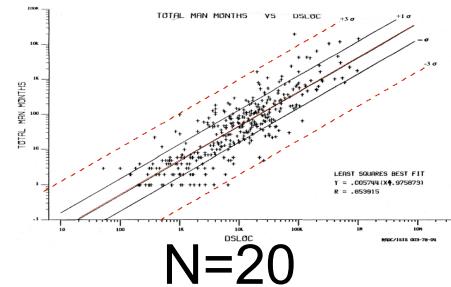
3. 効果を習熟工学で定量評価

4. 他の習熟効果例

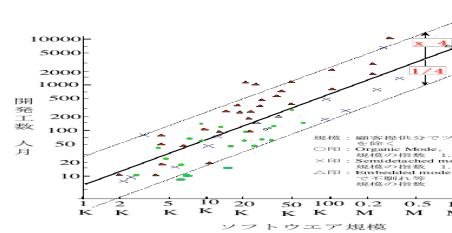
3. 3. 1 対数用紙上の帯状分布

対数尺度表示で帯状分布 分布幅 = 中央値×N ~ ×1/N

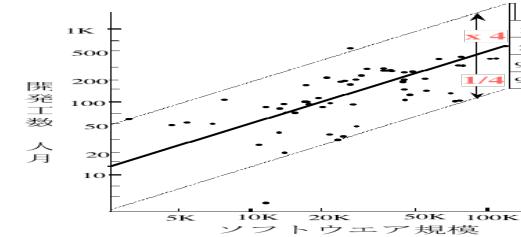
工數
關係



N=20

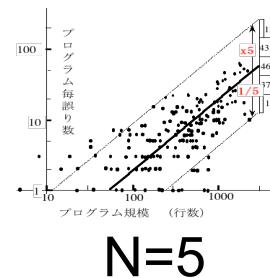


N=4



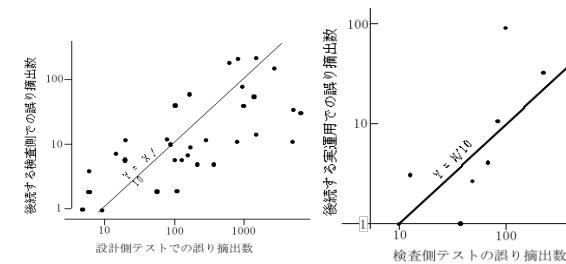
N=4

誤り 作込み

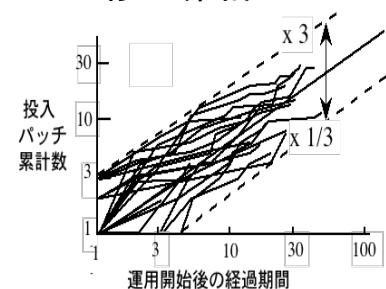


N=5

テスト
減衰

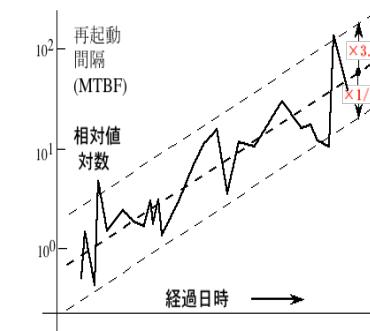


出荷後
品質



N=3

擬似負荷テスト でのMTBF推移

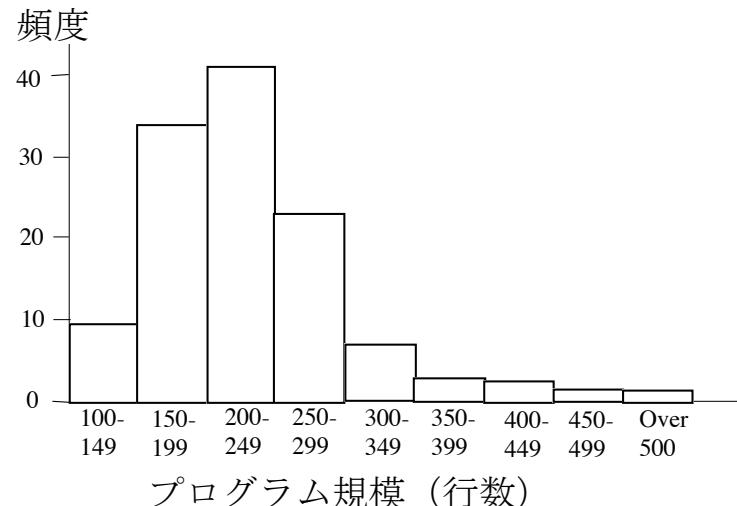


負の指數状減衰

3. 3. 2 規模も大きなバラツキ

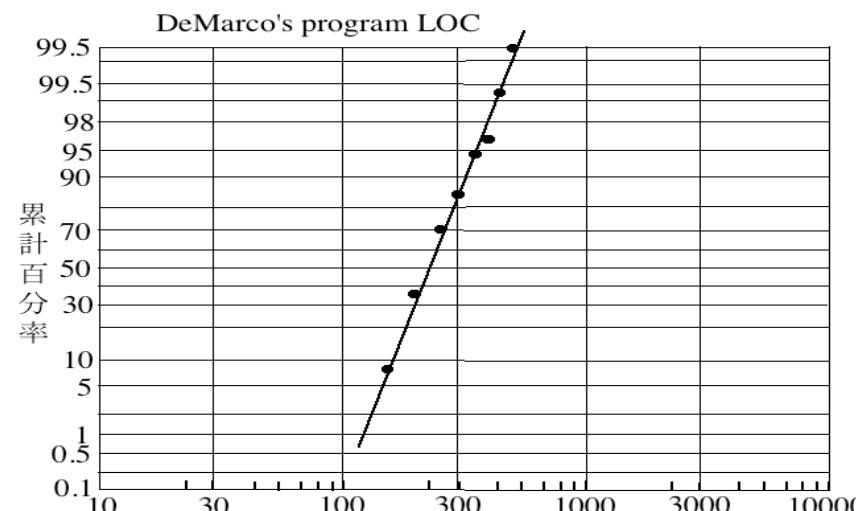
対数正規分布確率紙上の打点は直線状傾向、対数正規分布

DeMarcoの実験



プログラム行数の規模別分布

- 同一仕様でプログラムを作成しデバッグ
- プログラム行数（規模）は大きなバラツキ



対数正規分布確率紙：打点群が直線状の傾向なら対数正規分布。

T. DeMarco and T. Leister, Software Development; State of Thre Art vs. State of The Practice, The Eleventh International Conference on Software Engineering 1989, pp. 271-275, 1989.

河野善彌, 陳慧, ソフトウェア開発工程の定量的特性
(ソフトウェアの謎は有るか? 銀の弾丸は無いのか?
情処研報 2010-SE-No. 11, 2010.

3. 3. 3 対数正規分布

- ・確率変数の対数を取ると、対応する分布が正規分布になる分布.

- ・確率変数 X の確率密度関数 $f(x)$

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma x} e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}}, \quad 0 < x < \infty$$

- ・特性値が多数のランダムな因子 (>0) の相乗積で表される時に生ずる

例 疫学～株価（社会的現象）/自然現象など

- ・人の作業では、不可避的に多くの因子が関わる.

各因子のランダム性の為に対数正規分布になる.

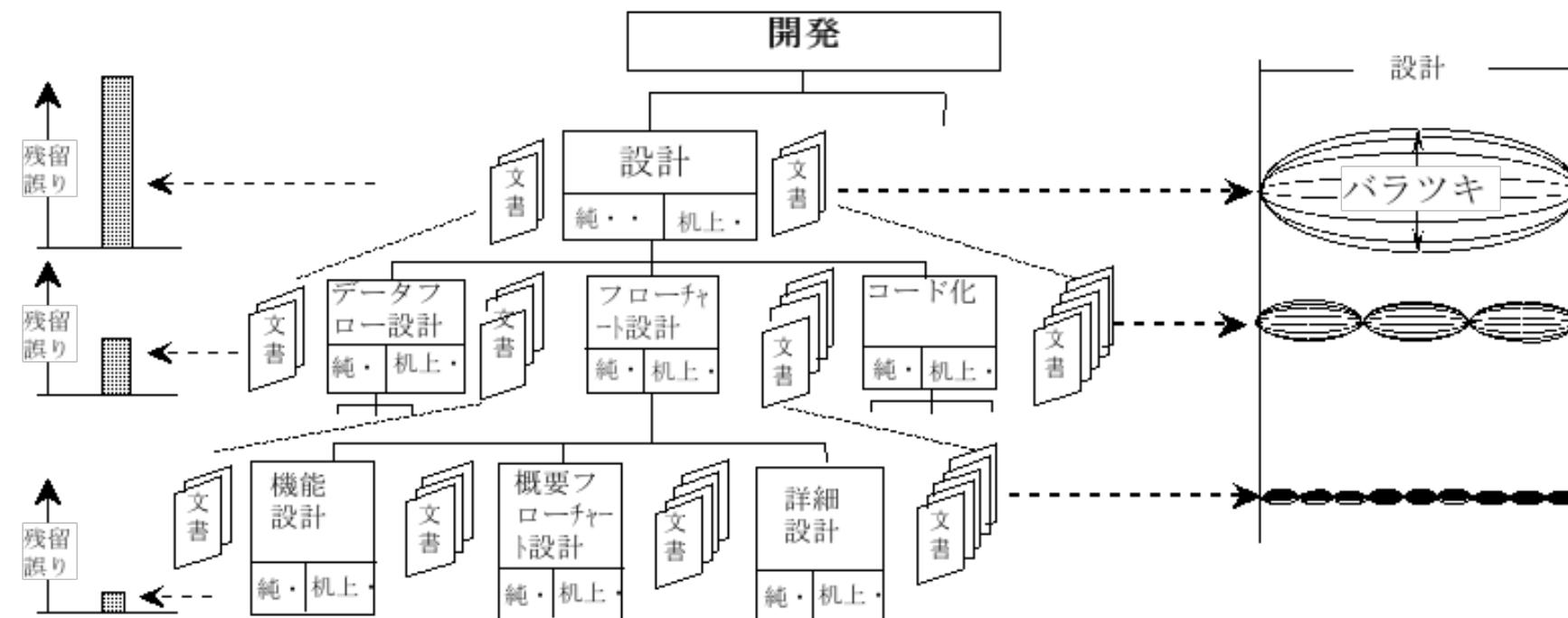
これは大きなバラツキを示す。平均値で正規化して

立上がり : $1/N \sim 1$ 減衰 : $1 \sim N$

これは謎ではなく、何も改善しない時に生じる特性.

3. 3. 4 プロセス(工程)の純化

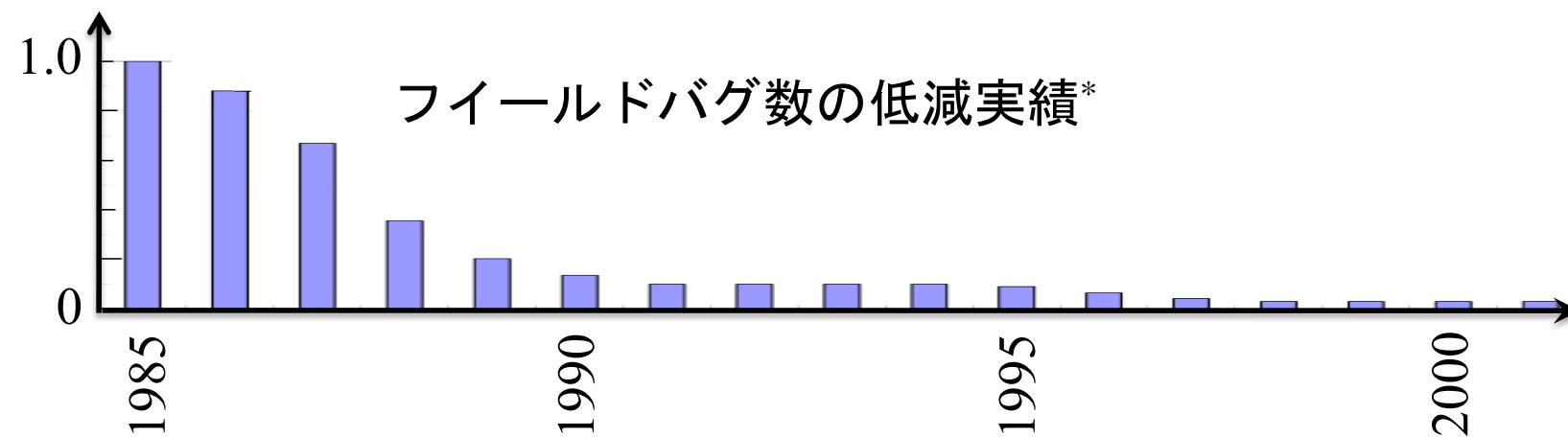
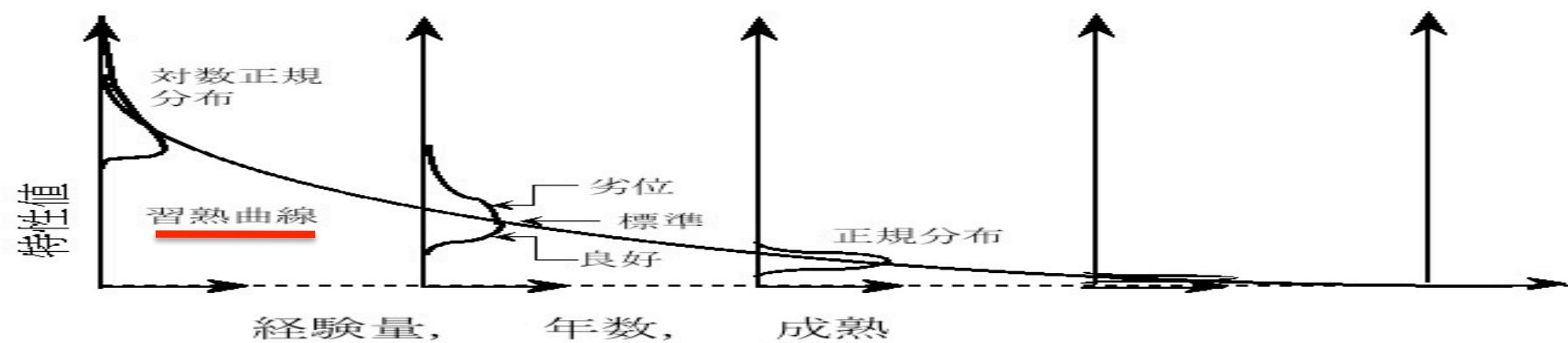
統制する工程を順次下位に伸ばす。図面/規格を明細化。
1/Mに細分化すると、原理上バラツキ幅は1/Mに減る。
同様に設計誤りも1/Mに減る。改善施策を打継け努力すると、
(蓄積された知・技術力が向上し) 習熟効果状に改善される。



河野善彌, Behrouz H. Far, ソフトウェア開発プロセスの構造 (Industrial Software Engineeringの立場から),
情処研報 ソフトウェア工学, 98-5, 1994.

3. 3. 5 プロセスの進化

改善の蓄積（習熟効果）により工程が進化～高度化する
定量化し具体的な改善をしなければ進化～高度化しない



*佐々木元, NECにおける品質革新の取組み, 日科技連ソフトウェア品質シンポジウム発表報文集2006, p. 15-35, 2006. のスライド16中のグラフを簡略化.

第3部 その他の特性

1. V型網

1. ソフトとハード開発と製造

2. 管理のV型網

2. 誤り（欠陥）とテスト

1. 誤りの作込み(Thayers資料)

2. 誤りの謎 1, 2

3. テストは定率の減衰機構

4. 減衰特性の実測例

5. 誤りとテストの構造

6. 作込み工程と再発防止

3. 対数正規分布

1. 対数用紙上の帯状分布

2. 規模も大きなバラツキ

2. 対数正規分

3. プロセス(工程)の純化

4. プロセスの進化

4. 習熟効果

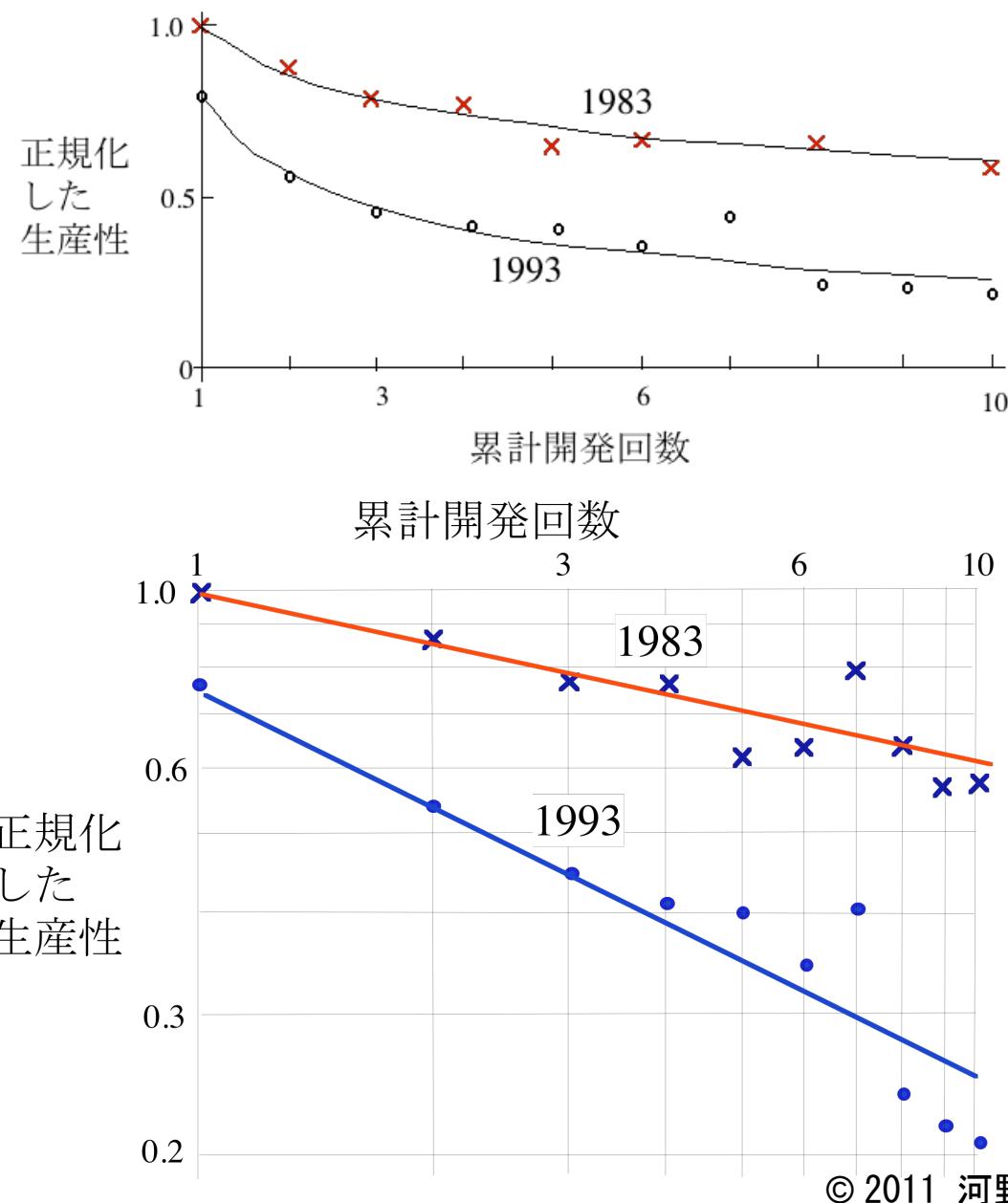
1. 対数習熟効果

2. 習熟(学習)効果は知の集積

3. 習熟工学で定量評価

4. 他の習熟効果例

3. 4. 1 対数習熟効果



・新しい行動を繰返すと、初めは急速に進歩する。伸びは次第に鈍化するが、何時迄も伸びる。従来、経験した知識を再利用すると想定されてきた。

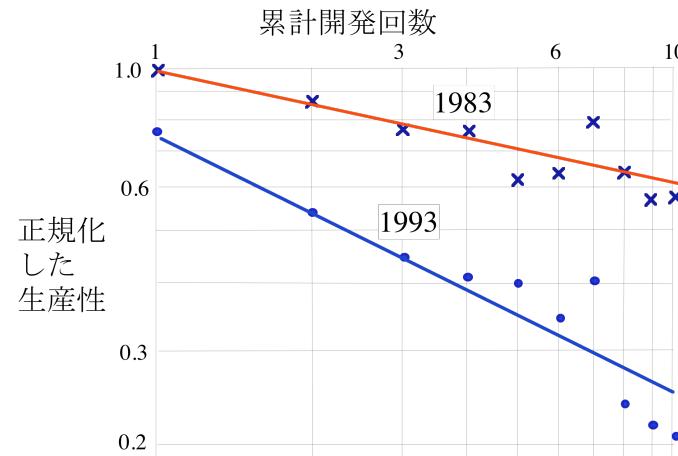
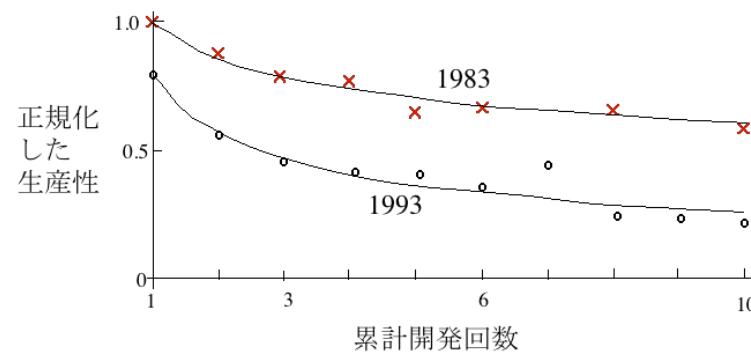
- ・左上図*はプログラム設計チームの実績、'93には自動設計が適用された。下図#は、作業量一定と仮定し打点した。
- ・累積経験回数vs指標値を両対数尺度で打点すると、直線状傾向線が表れる。対数習熟効果。ヒトの作業に表れる。

*森岡洋介、長野文子、大野治、EAGLE/Pを用いたプログラム開発の習熟、情処学会第43回全国大会、1K-3、pp. 5-385-6、平成3年後期。

#河野善彌、陳慧、人の設計知識構造と定量評価
(1/2)、信学技報KBSE2003-57(2004-3)

3. 4. 2. 習熟(学習)効果は知の集積

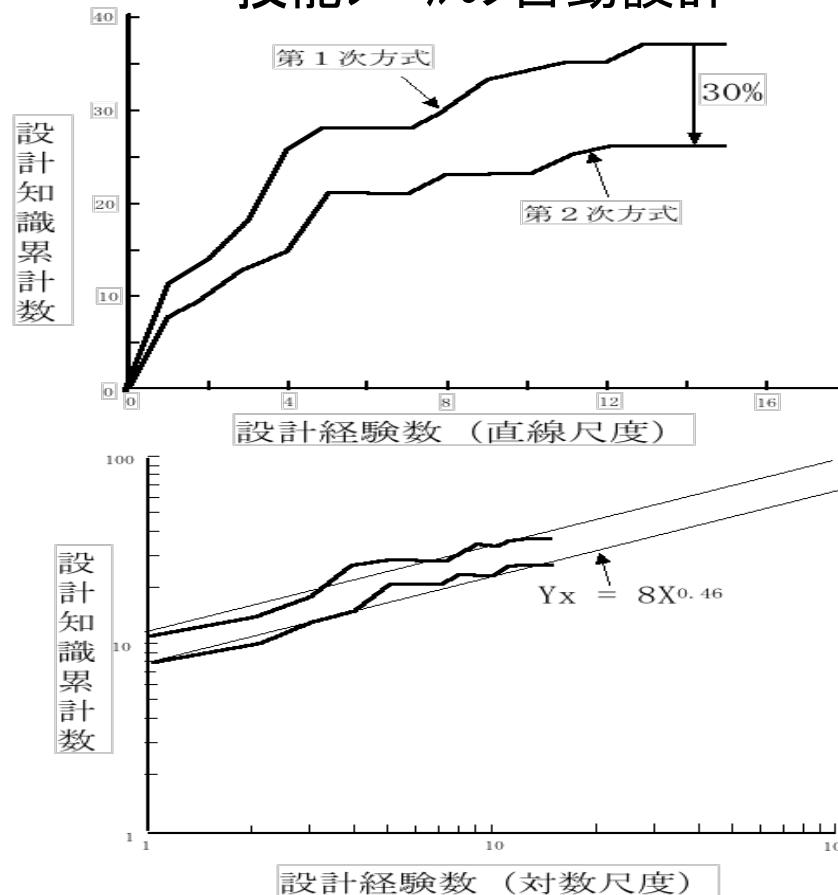
対数習熟効果



*Wright, Theodore Paul(February 1936),
"Learning Curve", Journal of the Aeronautical Sciences

知識数

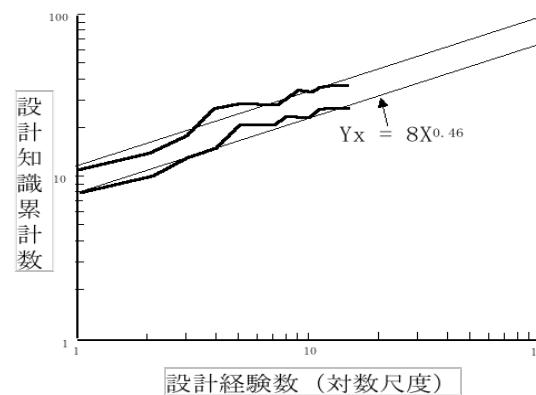
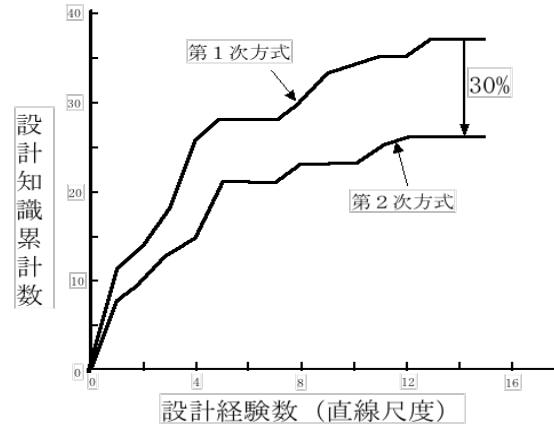
技能レベルの自動設計*



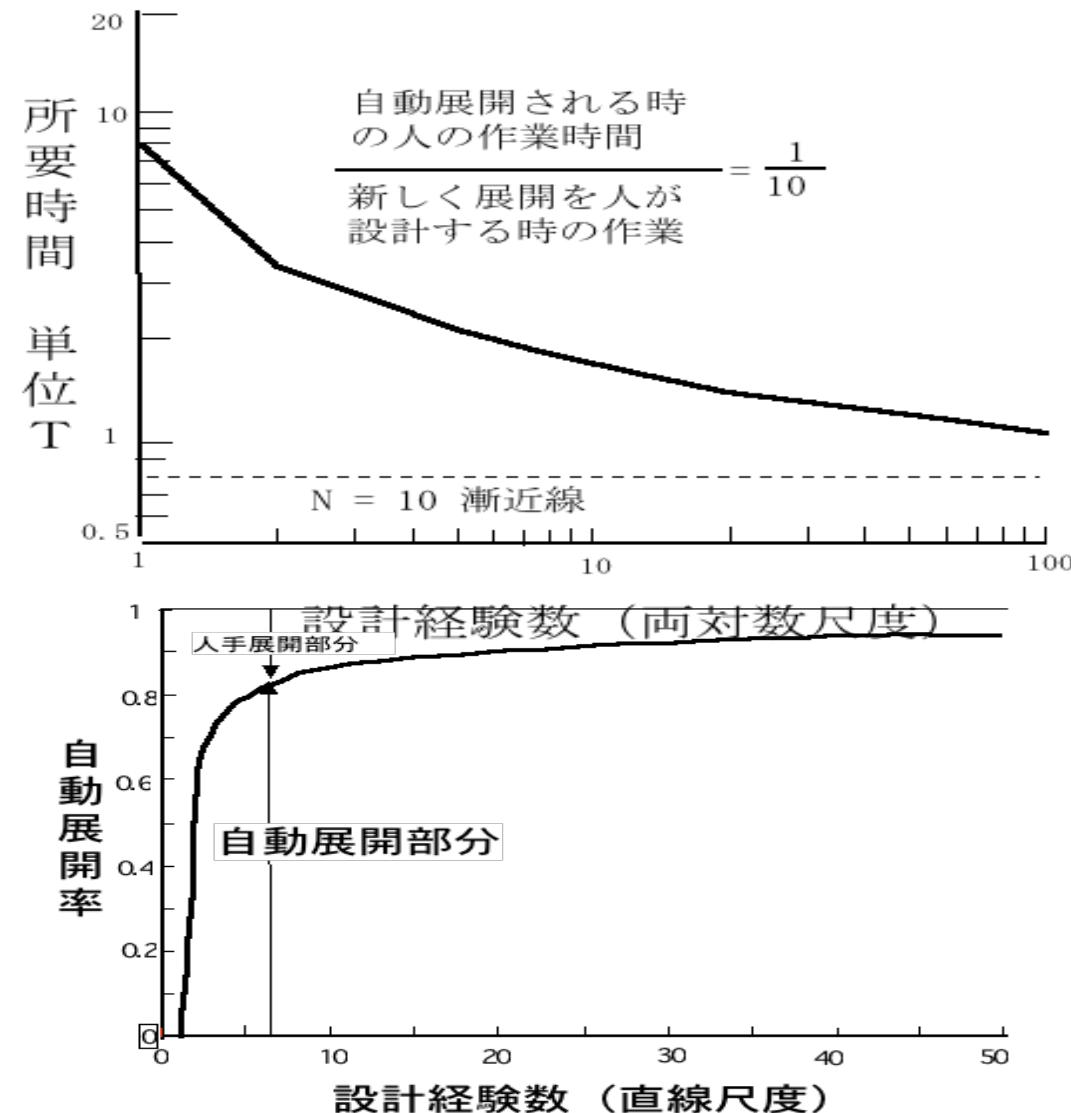
*Chen H., Tsutsumi N., Takano H. and Koono Z., Software Creation: An Intelligent CASE Tool Featuring Automatic Design for Structured Programming, JIEICE, Vol. E81-D, No.12, pp. 1439 – 1449, Dec 1998.

3. 4. 3 効果を習熟工学で定量評価

・自動設計 技能レベルの動作

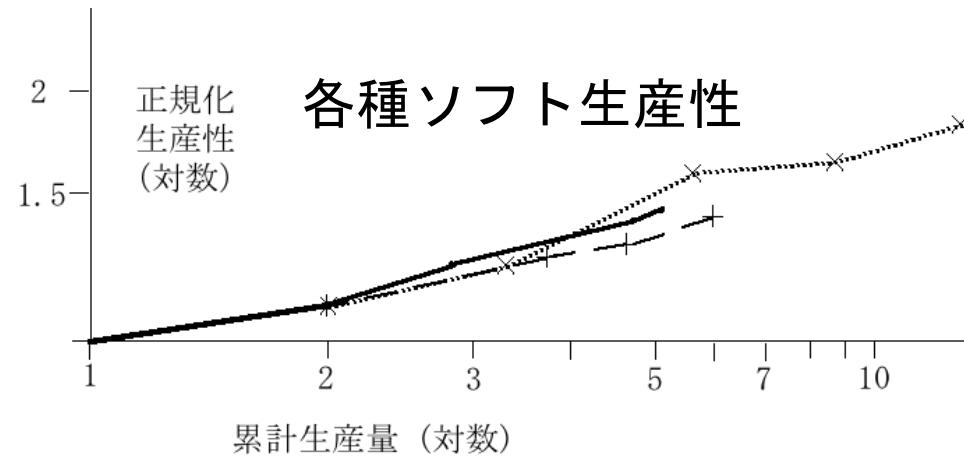


習熟傾向
傾向線の式を得る

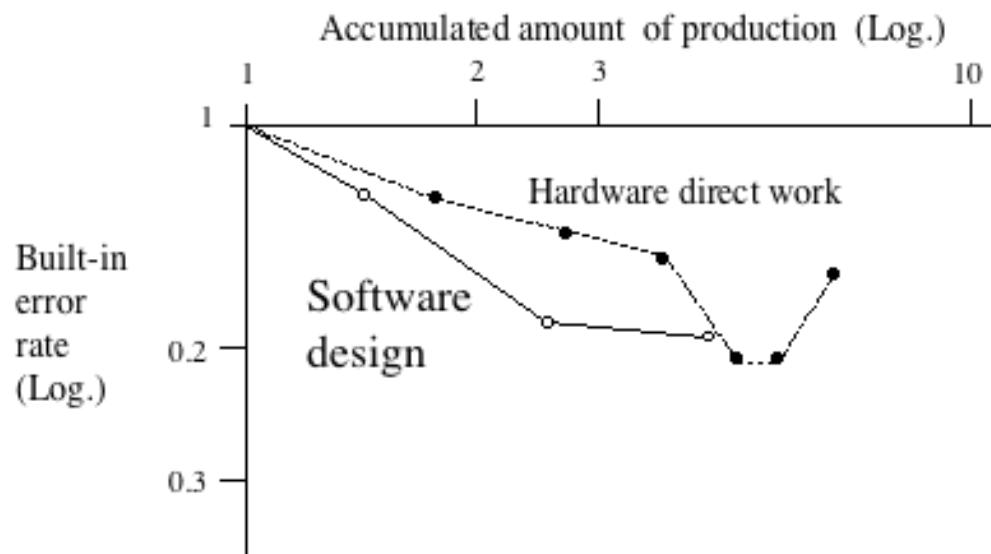


*Chen H., Tsutsumi N., Takano H. and Koono Z., Software Creation: An Intelligent CASE Tool Featuring Automatic Design for Structured Programming, JIEICE, Vol. E81-D, No.12. pp. 1439 – 1449, Dec 1998.

3. 4. 4. 他の習熟効果例



Z. Koono, H. Tsuji and M. Soga, Structural Way of Thinking as Applied to Productivity, IEEE COMSOC ICC '90, 204.2.1-7, 1990.

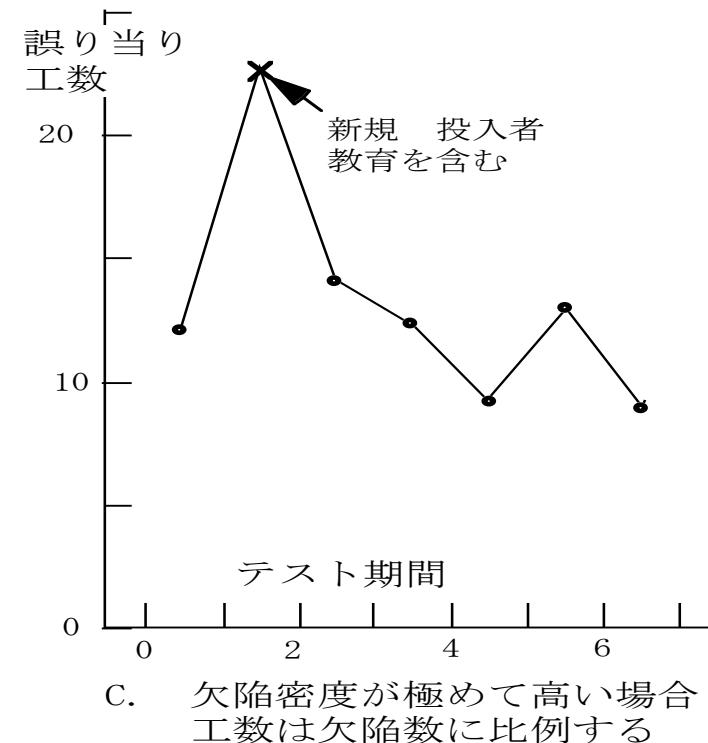


Z. Koono, K. Igawa, and M. Soga, Structural Way of Thinking as Applied to Improvement Process, IEEE COMSOC GLOBECOM '88, 40.1.1-5, 1988.

現実/現物/現場が全て！

現場で遭遇する謎

デバッグ総工数/期間 (件数)



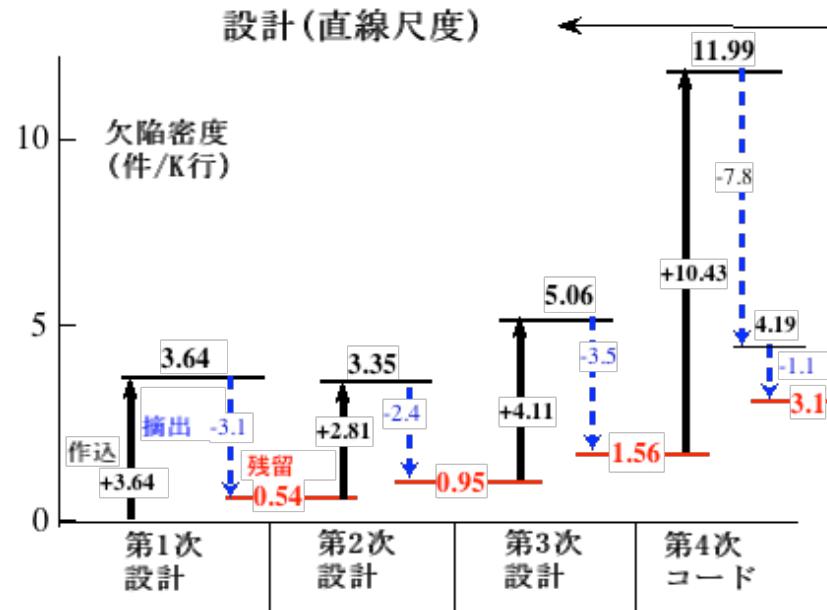
Z. Koono, H. Tsuji, and M. Soga, Structural Way of Thinking as Applied to Productivity, IEEE COMSOC ICC 90, 204.2.1-7, 1990.

第4部 システム

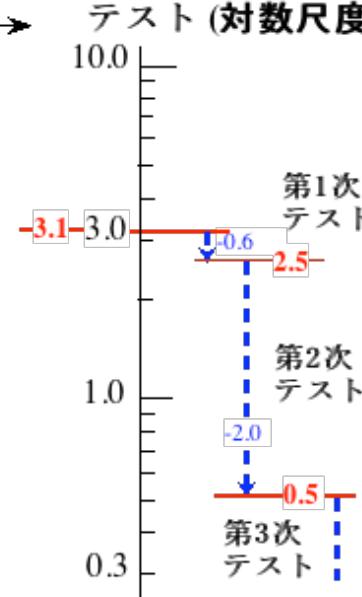
1. 作業自体は高度化できる
2. 自主製品の事業計画
3. 設計技術 ソフト規模
4. 設計技術 生産性
5. 意図的行動の枠組
6. 設備的ソフトの投資計画
7. 依頼開発なら

4. 1 作業 자체は高度化できる

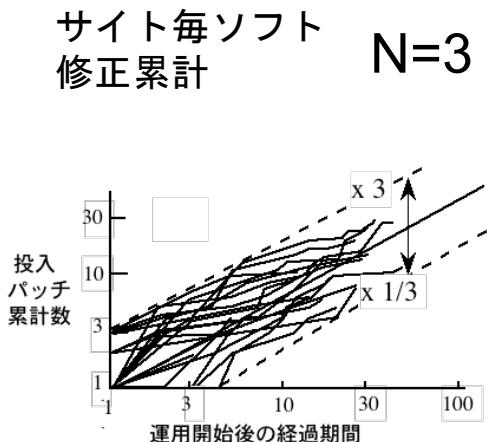
管理の基本はハードと略同様だが、更に高度化可



設計は誤り作込↑と
机上チェック↓



テストは
減衰機構



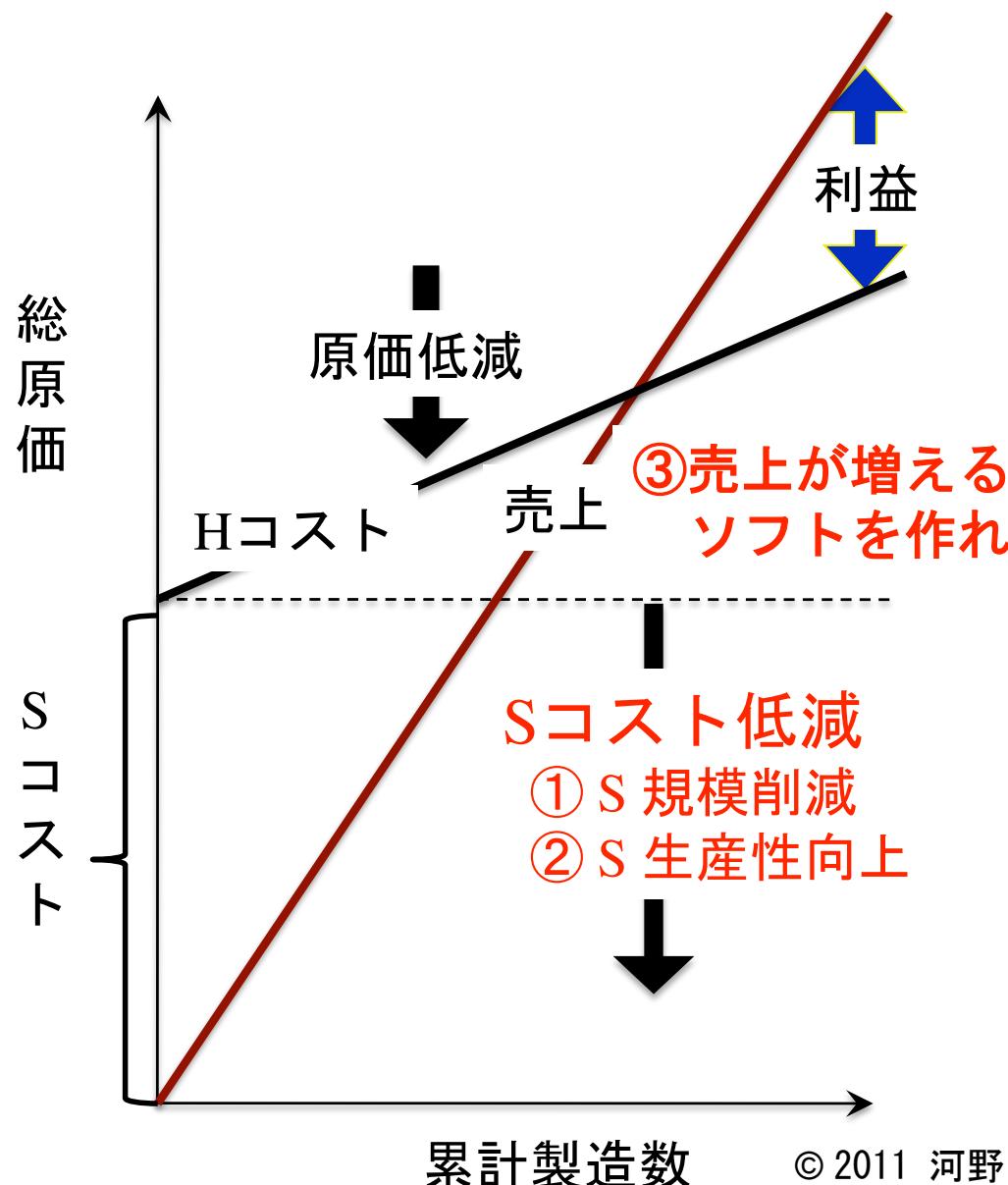
出荷製品の
残留欠陥率は
対数正規分布
工程の能力
平均値×3
平均値の推移
負の指數減衰

1970年代末米GTE PBXの事例*, 通信で巨大開発力を持つBTLに対抗し、エースを集中投入してPBXを開発し、評価した。

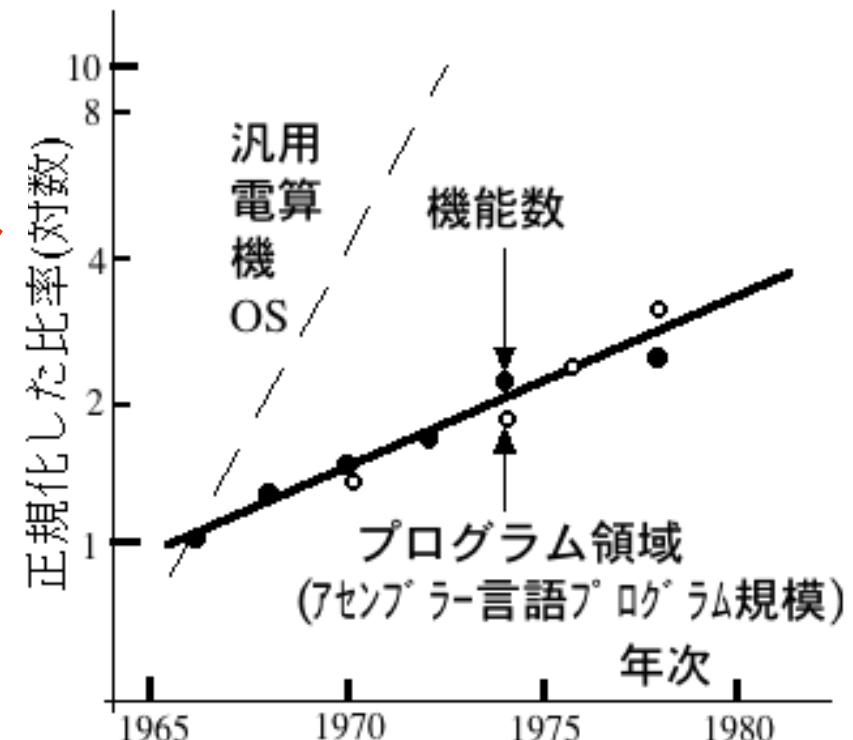
* E. B. Daly and D. A. Mnichewicz, The management of large software development for stored program switching systems, International Switching Symposium '79, pp. 1287-1291, 1979. および第2著者より頂いた内部資料から作成。

4. 2 自主製品の事業計画

事業の成功は、経営トップ、製品企画～開発の技術力に依存

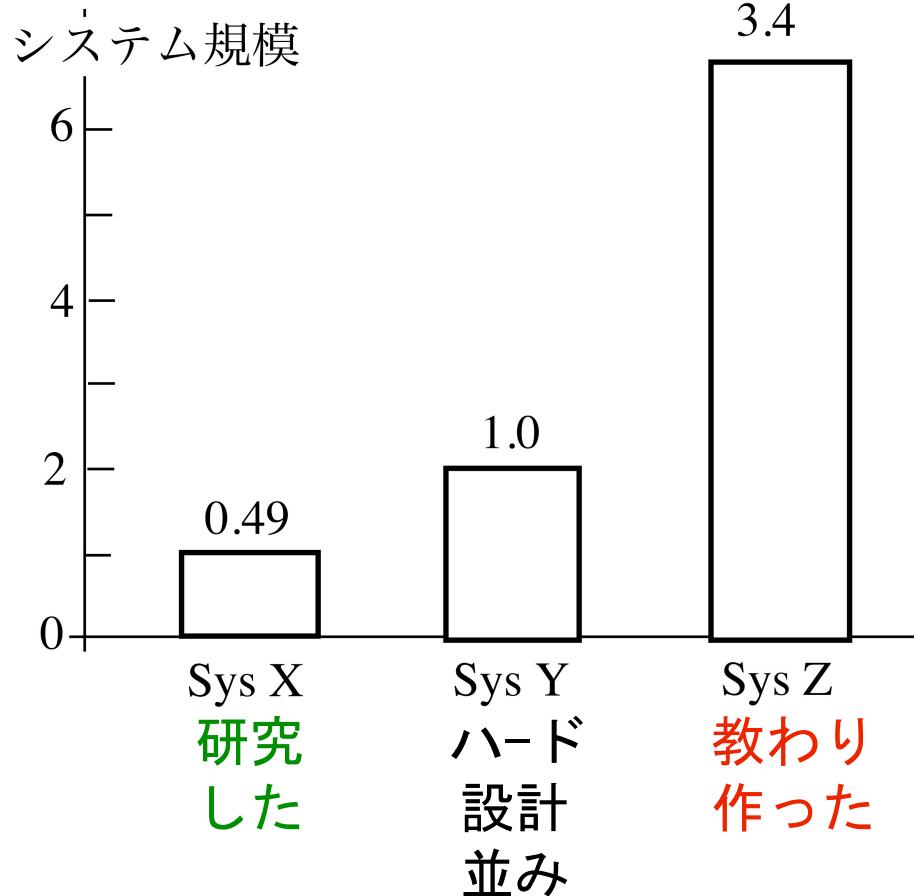


一度作ったソフトの
規模は約10%/年増える
①ソフト規模の低減

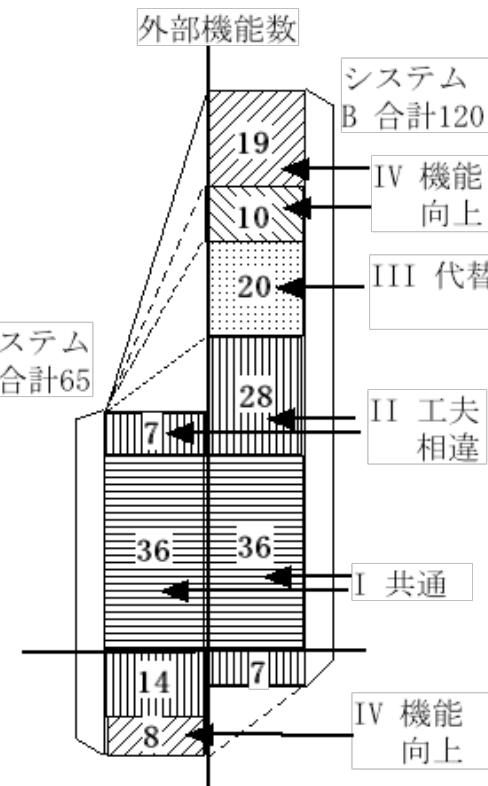


河野善弥、高集積時代の処理系装置、昭和54年
電子四学会連合大会、No. 27-3.

4. 3 設計技術 ソフト規模



規模の違いは工夫の違い



多くの中小IT系～ソフト受託開発企業の考え方
云われたようにドンドン作れ！ 最適化などするな！
弁明：顧客仕様遅れ/変更多発/出来る時に作らねば
発注側と受注側のもたれあいでは？

Z. Koono, Kondo T., Igari M. and Ohtsu K., Structural way of thinking as applied to good design (Part 1. Software size), IEEE Global Telecommunications Conf. 1991, pp. 24. 3. 1-8, 1991.

4. 4 設計技術 生産性

SWEng 公開資料による実力評価

	1970~'80	2000+α
COCOMO 1981	$\frac{1200\text{MM}}{100\text{KL}}$	$\frac{11\text{KMH}}{100\text{KL}} = \frac{73\text{MM}}{100\text{KL}}$
RADC 1978 (×1/20～×20)	$\frac{350\text{ MM}}{100\text{KL}}$	$\frac{1}{5}$
IPA SEC データ白書2008 p. 160 新規 COBOL 150 MH = 1MM	$\frac{1}{16}$	

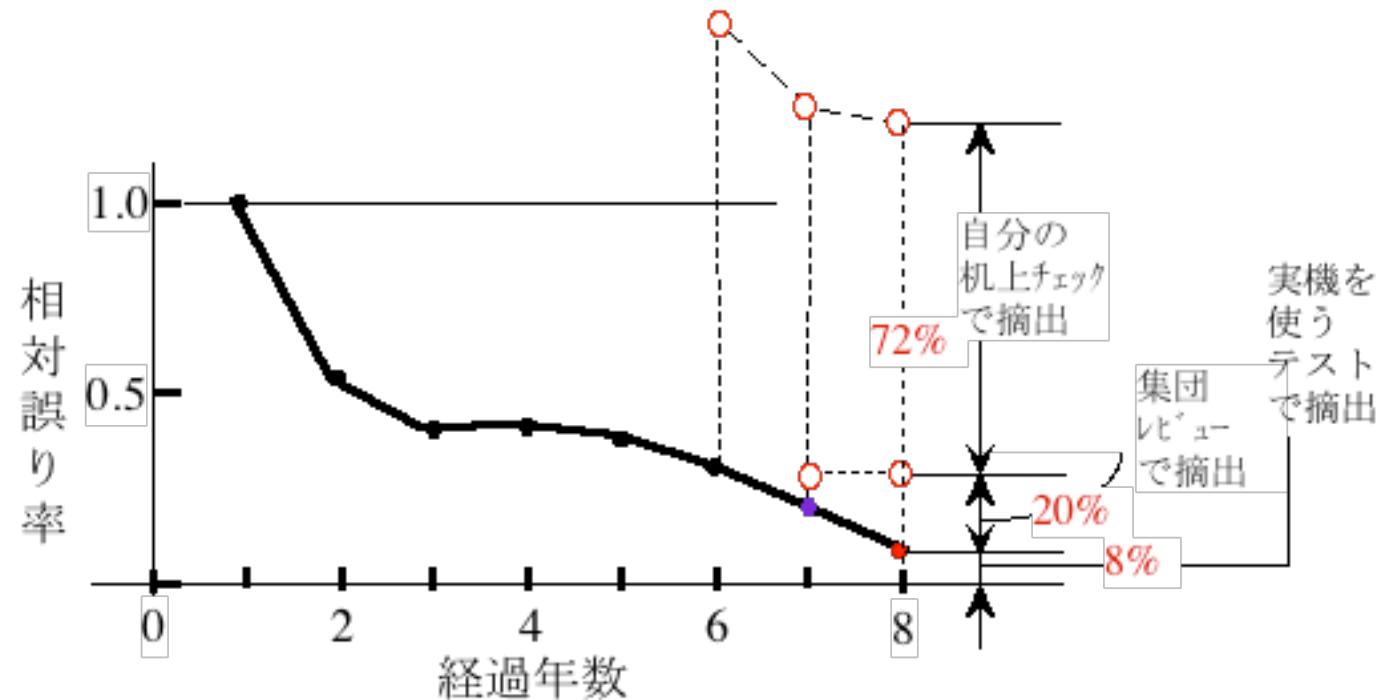
1世代（30年間）に5~16倍とは低すぎる。

生産性向上は売上高確保に有害なのか？

ハード論理設計は少なくとも1000倍化した。

決め手は再利用と自動化。

長期間の品質向上努力



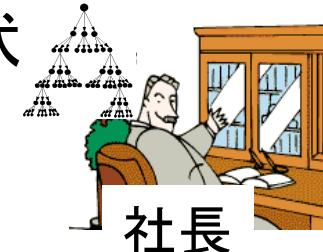
日立のあるソフト部門の成果TQCによる約8年間の改善
テスト摘出欠陥率 < 1/10
リーダの率先垂範
期毎に品質向上の新具体策
終期 事業所随一の精銳部隊

10年間の努力の蓄積により、品質向上は 10~30 倍

4.5 意図的行動の枠組

製品開発/設備導入は、現ハードの枠組にソフトも含める

階層展開網状



階層的に

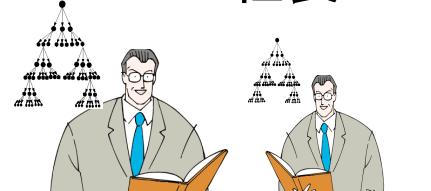
財務諸表 → 改善した財務諸表

投資額

達成効果額

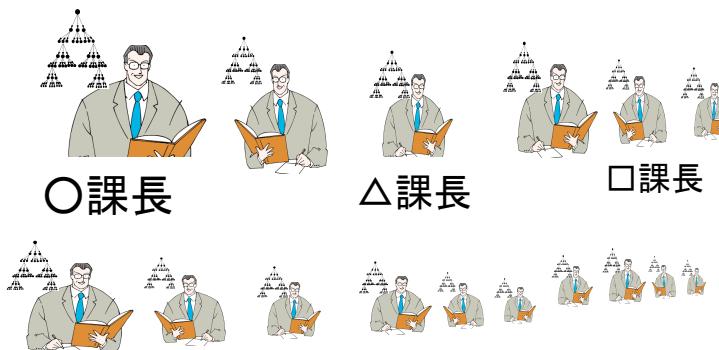
達成時期

具体化



経理部長

する



投資案A
投資案B
投資案C

社長意図

B設備新営

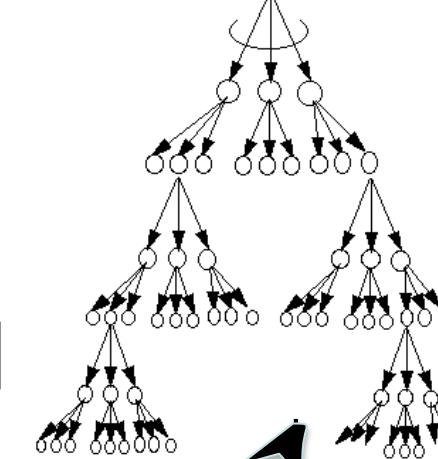
β設備

γ設備

△仕様

ε仕様案

φ仕様案



階層的に統合

設計仕様書

ソースリスト

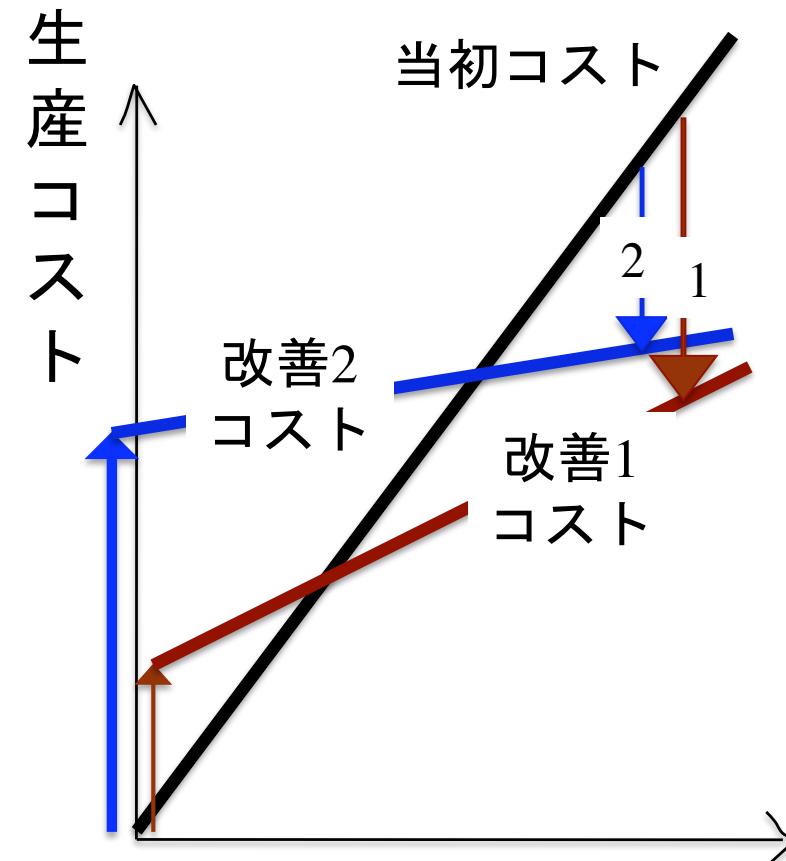
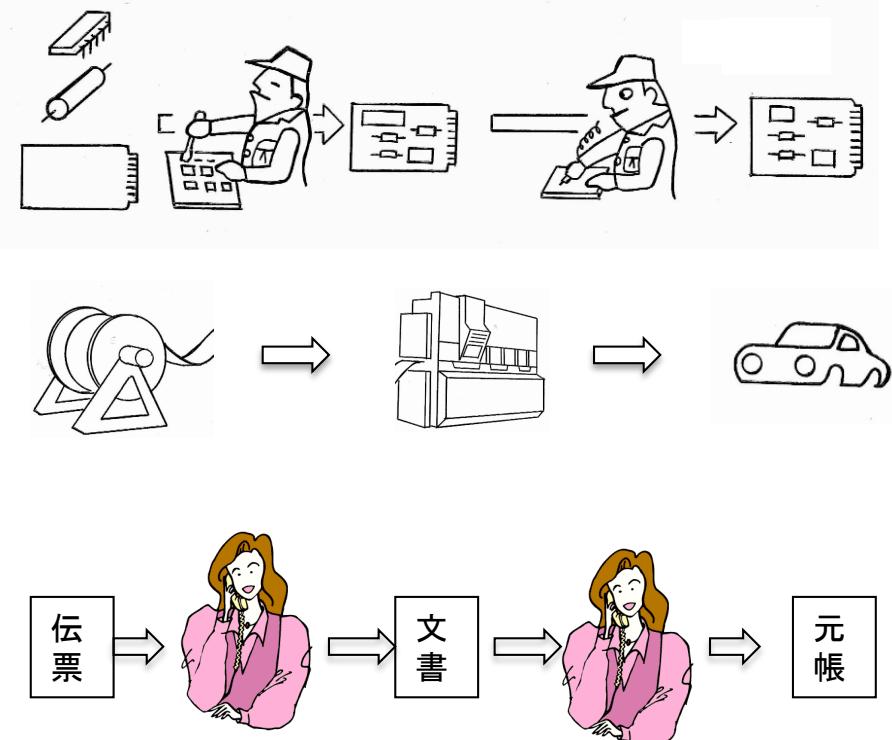
現状はこの辺から出発では?

© 2011 河野

4. 6 設備的ソフトの投資計画 前

直接的コストを節減する設備投資も間接的コストを節減する
ソフト的設備投資も同じ。同一ルールで厳しく統制せよ！

原理

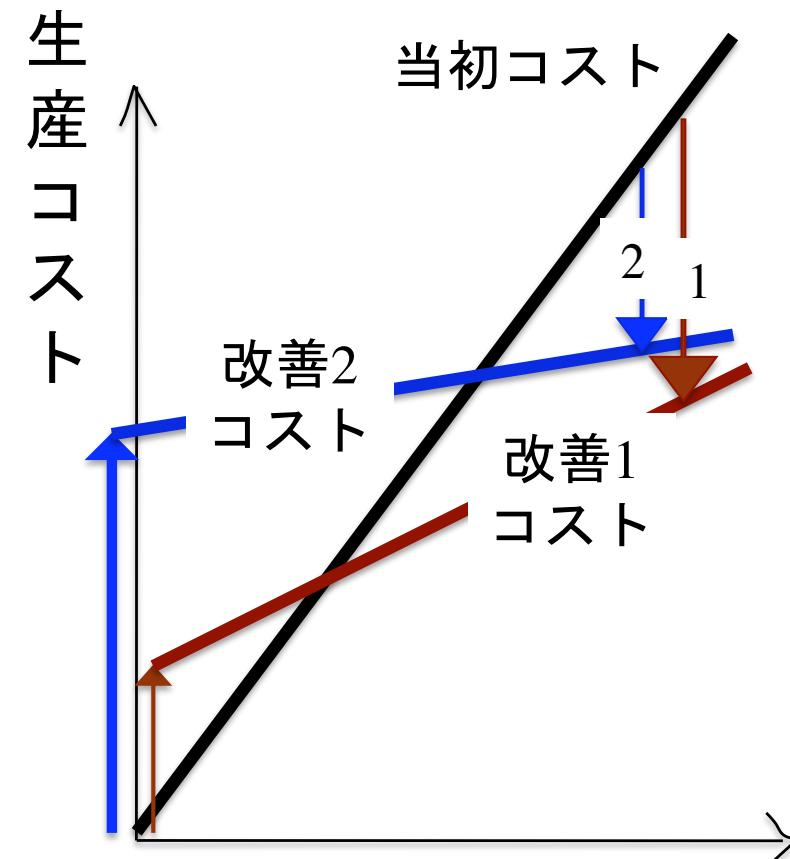
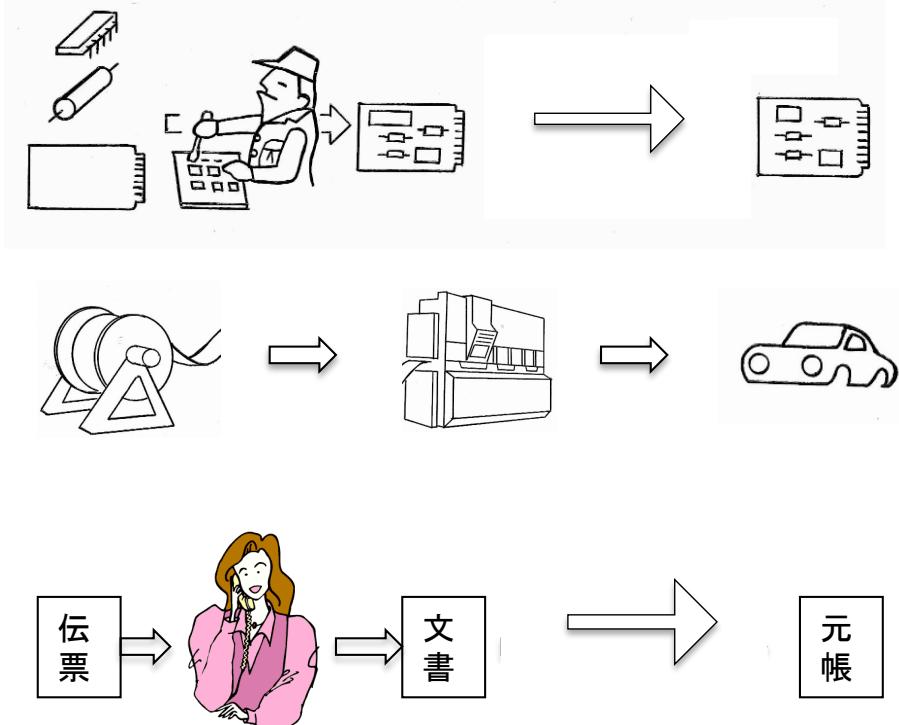


自主製品の場合に加えて、設備投資の約束効果—自動化対象—
ソフト開発費のトレードオフを自ら確認する必要がある。

4. 6 設備的ソフトの投資計画 後

直接的コストを節減する設備投資も間接的コストを節減する
ソフト的設備投資も同じ。同一ルールで厳しく統制せよ！

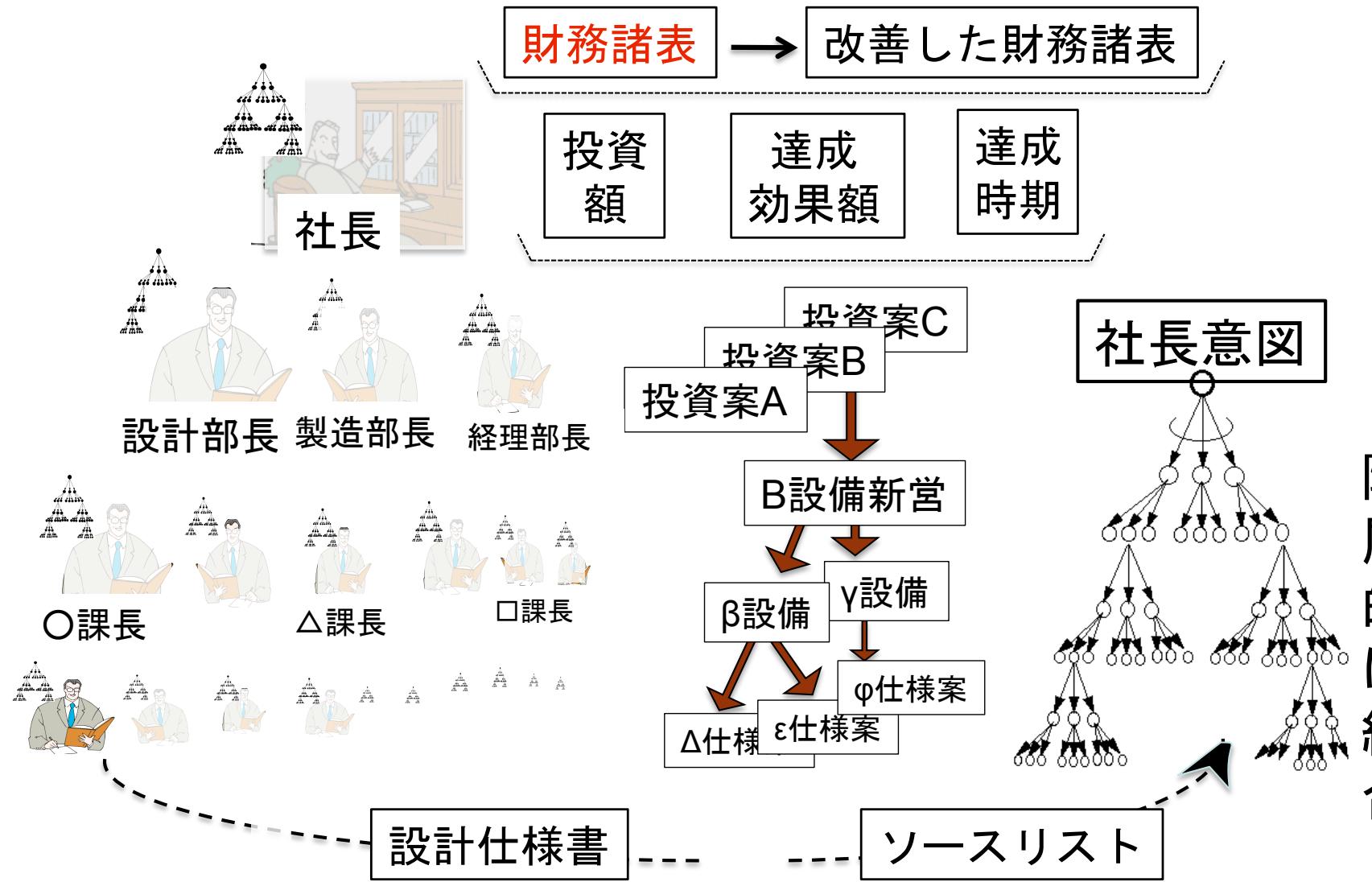
原理



自主製品の場合に加えて、設備投資の約束効果—自動化対象—
ソフト開発費のトレードオフを自ら確認する必要がある。

4.7 依頼開発なら

依頼元の全関係者に成り替わって心遣いする。日本流。



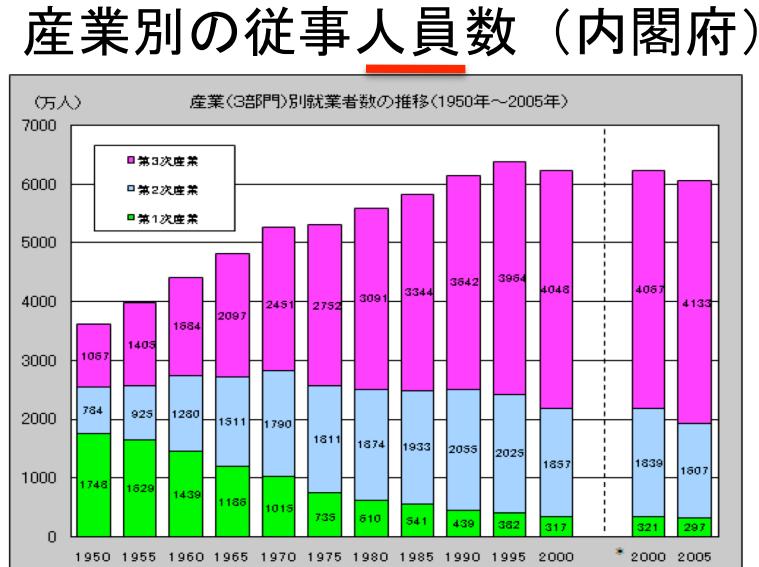
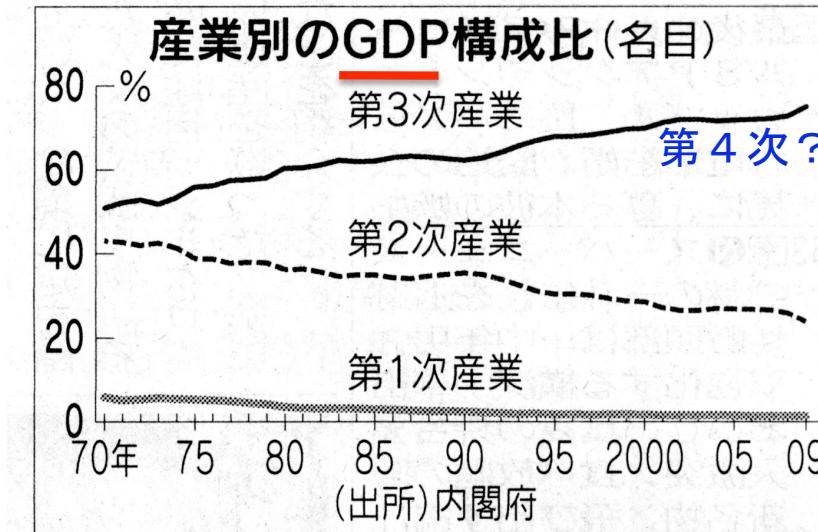
「要求工学」は成功できるか？

第5部 総括

1. 産業の長期的推移
2. 受託開発企業群は労働集約産業
3. 統一的な体系
4. むすび

5. 1 産業の長期的推移

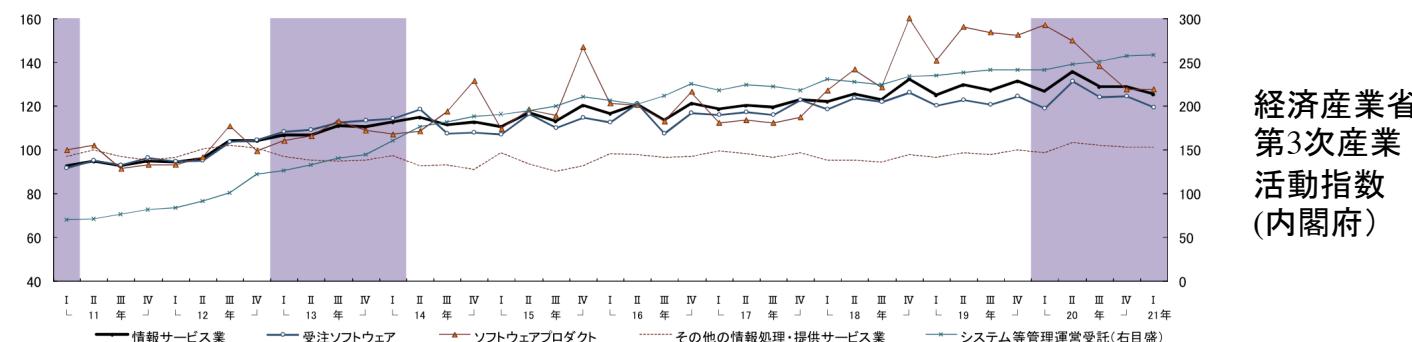
第2次産業はピークを越し、第3次産業は続伸中



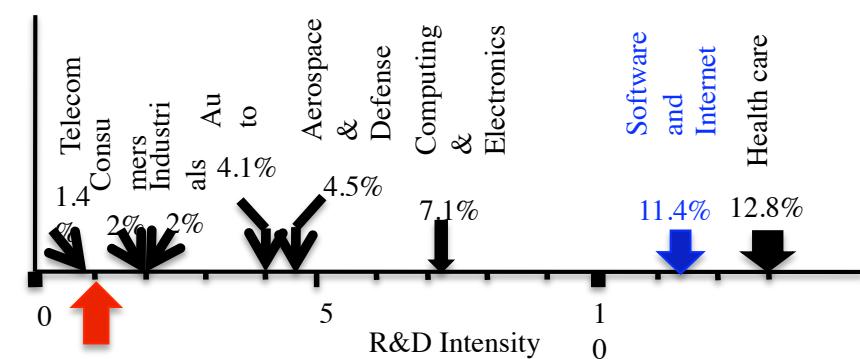
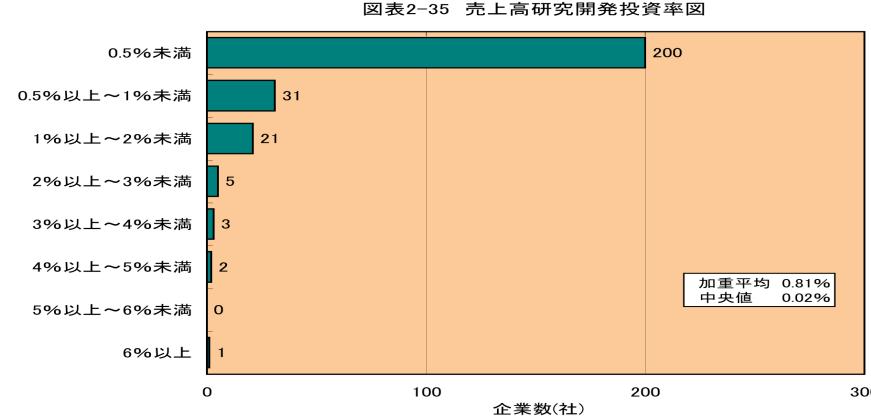
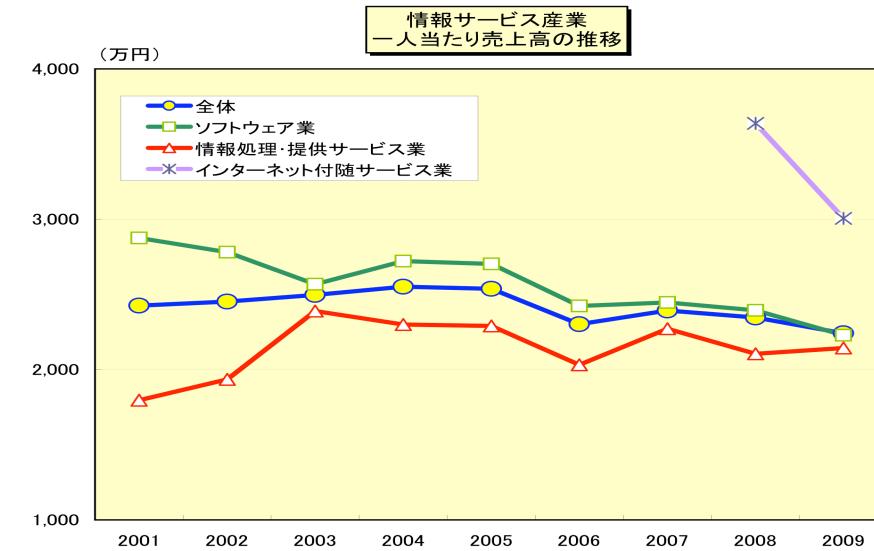
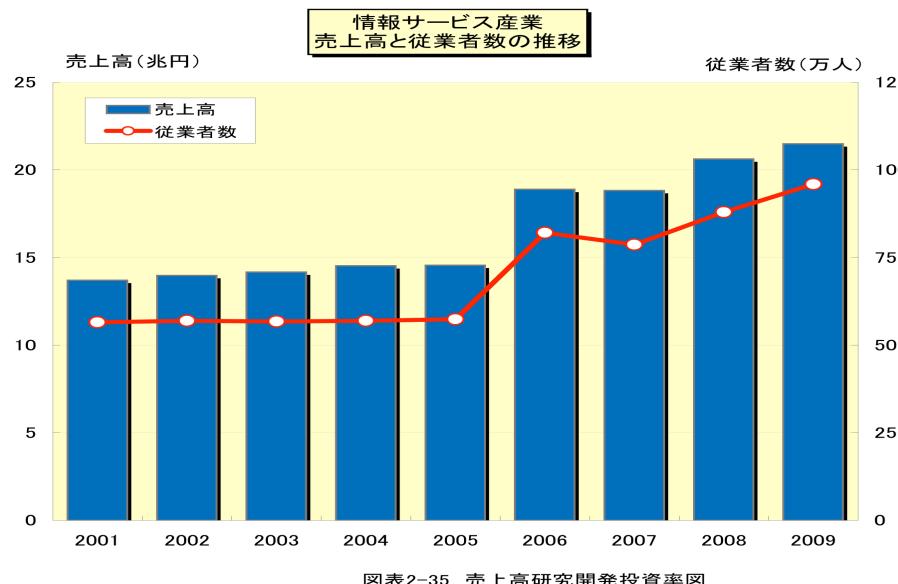
日経新聞、2011.8.17 朝刊23面経済教室

第3次産業で伸びる業界

第I-2-6図 情報サービス業活動指数の推移(12年=100、季節調整済)



5. 2 受託開発企業群は労働集約産業



Year	'94	'95	'96	'97	'98	'99	'00	'01	'02	'03	'04	'05	'06	'07	'08	'09
R&D Index (%)	1.10	1.04	1.33	1.58	1.62	1.72	1.2	1.1	1.0	1.0	1.1	1.02	1.15	0.84	0.47	0.82

右中央の図を除く全出典 JISA, 日本情報サービス産業協会, 各年統計資料類

5. 3 統一的体系

1. 現場実績資料から経験式を得て、理論で裏付けた。これらで実績を説明できる。(大学初期段階 程度)
2. IE～経営工学での広義のハード生産管理体系を(S/Hを含む)システムを扱う統一的体系が出来る。これは合理的定量的科学的な横断～共通的な工学。
3. 経営等の目的に向け、これら各要素技術を組合せ合目的な活動体系(例：日本的～TQM)を推進すると産業的に高度化でき、国～産業とその成果を受ける大衆/これを支える人々に大きく貢献できる。
4. 意図的行動はヒトの公式的行動の殆どを占める。全国民の教育体系の基礎とする価値が有る。
5. 経営工学会の各位は上記に参画すべき基幹人材で、本学会はこれらを推進～牽引する母胎であろう。

5. 4 むすび

1. ヒトの知の基盤は、言葉→文法→意図的行動と進化した。
ハード作業用の定量化/評価/改善等を延長した統一的体系ができる。
自主製品～S/H含むシステムは自社中心で！。プロダクション段階も自動化して世界トップを目指せ。
2. 世界は知を集積し発揮する総合知の時代(第4次産業説)
に入っている。我々も社会の役に立とうではないか！
3. 従来の改善～管理技術は企業対象。この体系は全てに共通な知の向上策。単にハードの製造/品質技術でなく、
ハードも含めた全ての、研究開発/製品計画/各種の改
善/向上用の設備投資、これら経営の上段に当たるMOT等
も含めた全てに改善～向上が実現します。
4. 知の向上策だから、企業/組織で働く各個人は知の蓄積
と向上を要します。中学/高校と目的を建てて実行させfeedbackして自己修正する教育をする。
大学は各領域毎の色彩を入れて教育し簡単なシステム仕様作成からプログラム化する。
修士ではシステムの計画管理も学ぶ。本学会はこの理想を目指しましょう。
関心をお持ちの方方に、思想/計画/構想等を報告して、異見/意見
を頂き、試行し/評価し/更に改善向上いたしたい。

本体終了

ご清聴有難うございました.

忌憚無いご意見/異見を賜りたく

付録 人に倣ったソフト自動設計

動機： 最高に有効なソフトウェア生産手段

方向付け： 部品・技術・管理・経営等の就業経験から、
知が人の行動の中核と推定し、ヒトに倣った。

経過： 1901年大学に転職してプロジェクトを開始。

1. 暗中模索

人工知能は駄目

2. 知的CASEツール

Computer Aided Software Engineering tool

3. 強化

Zipf's law

2001年定年退官に先立ち原理段階を終え、特許を出願

出願：JP WO2002/097727 再公表（優先権主張 2001. 5. 28）

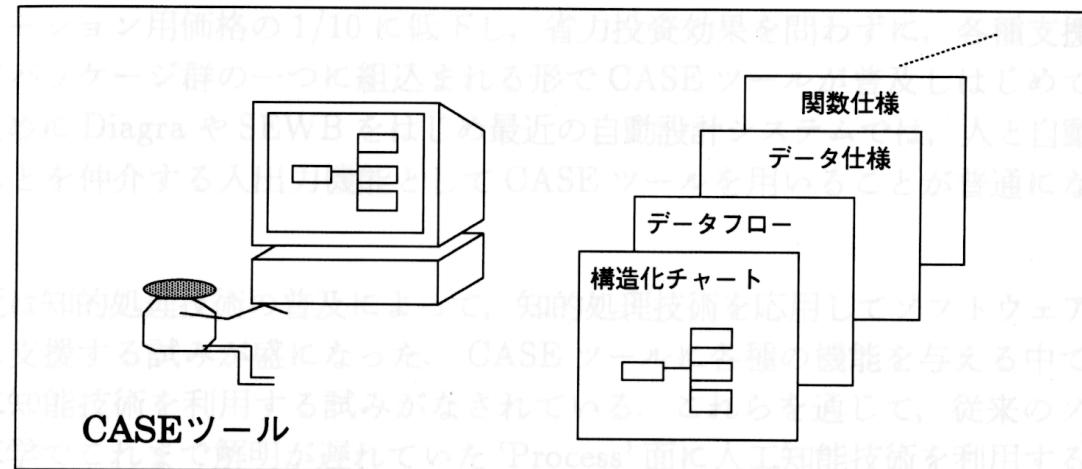
中国：ZL 02810859.0 (2006. 7. 26)

米国：US 7480642B2 (Jan. 20, 2009)

基本思想

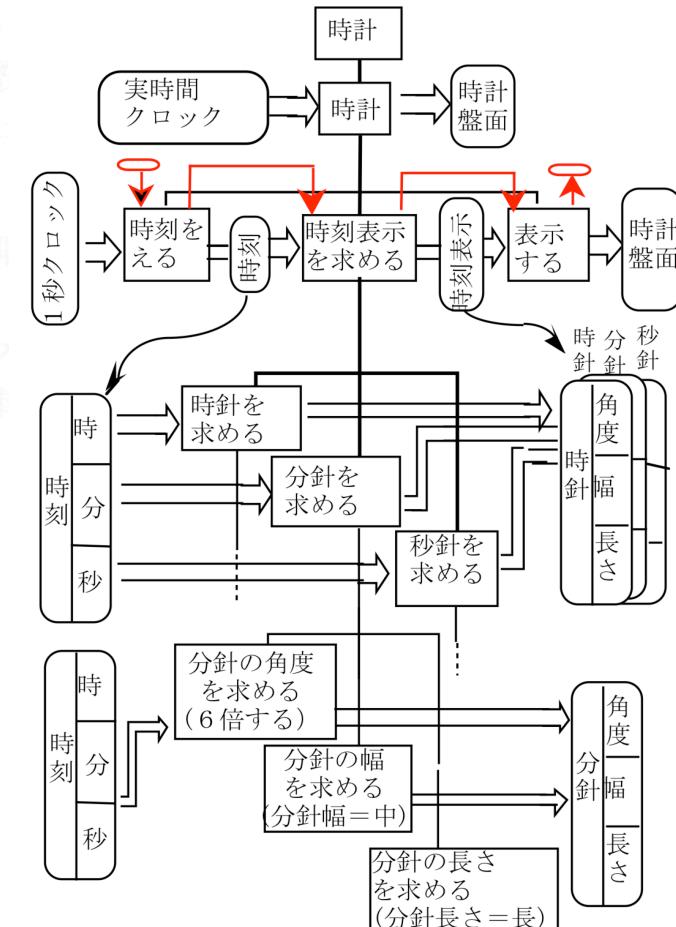
- ・設計＝意図的行動の親子概念展開連鎖 親一子対を単位知識
- ・設計＝図面作成プロセスと看做す 図面作成の手続き処理
- ・複数エンジン：Zipfの労力最小化，Rasmussen説 技能/ルール/知識
- ・制御 上位：有限状態抽象機械(FSM) 下：状態遷移フワーム

Computer Aided Software Engineering ツール



ソフトウェア開発用に特化した
対人入出力 ⇔ 設計情報のツール

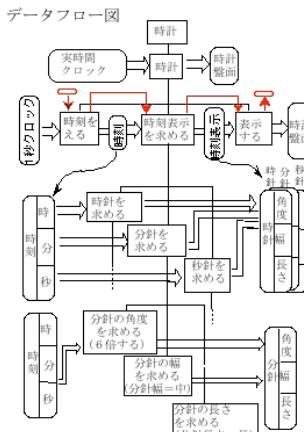
1. データフロー(図)
2. 構造化チャート類
(フローチャート)
3. 階層図類
 - ・機能(モジュール)階層図
 - ・ファイル～データの階層図



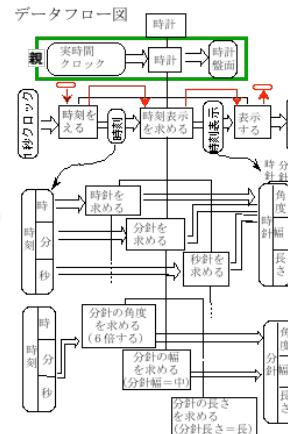
データフロー(図)

知識抽出

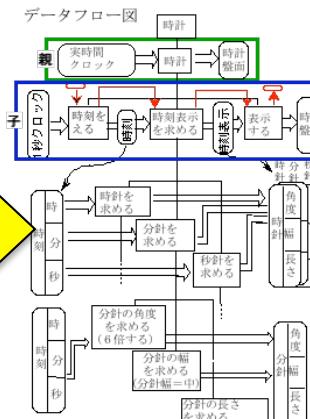
下記により 親単位データフローと詳細化データフローの対を得ます.
制御のフローチャートも基本は同じです.



入出力データを付加
親 単位データフロー化
緑内図形を一筆書き
親情報再現する

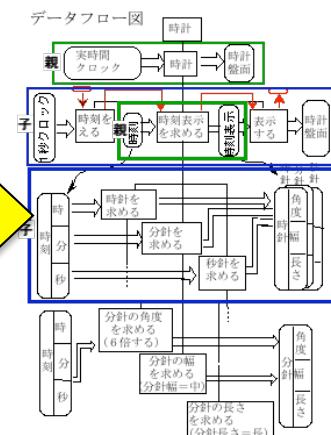
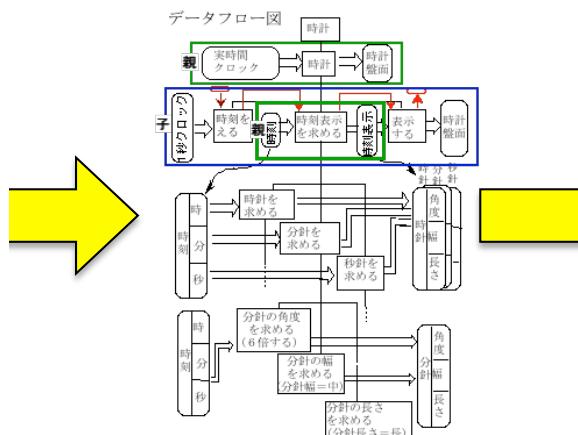


緑の親単位データフローから下位の詳細化データフローを一筆書きでなぞり情報を再現する



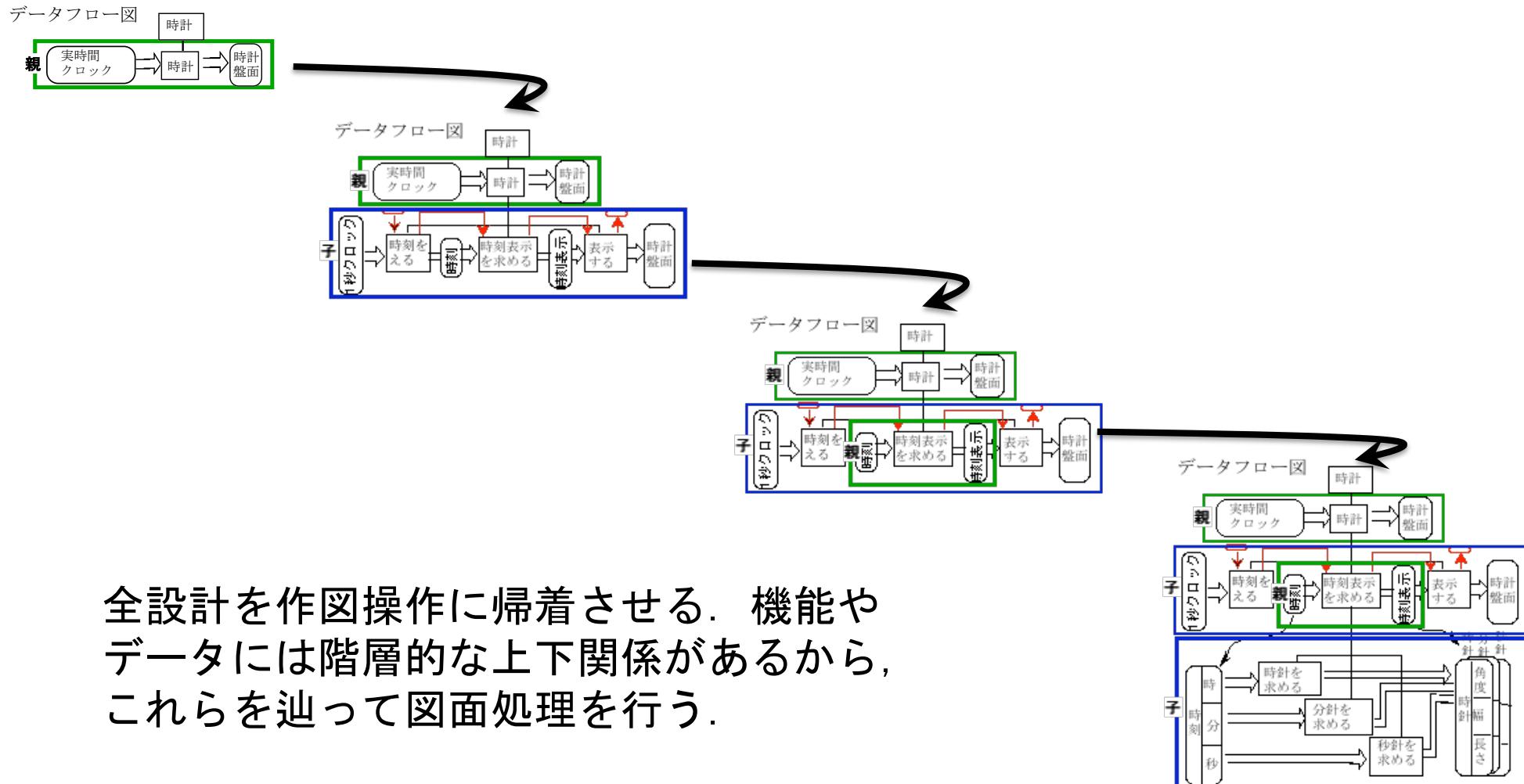
前記の動作を繰返し、青色の新しい詳細化データフローを取出す。以下繰返す。

青色の詳細化データフローは
3単位データフローに分かれる。
各単位データフローについて



自動設計

緑色の親をエンジンに送り、対応する子（青色）を得て、下位に貼付けることを繰返す。



全設計を作図操作に帰着させる。機能やデータには階層的な上下関係があるから、これらを辿って図面処理を行う。

親概念を詳細化する多種のエンジン

- Zipfの「労力最小化の原則」：

問題の解決にはまず、最簡単な解法を試みる。不可なら少し高度化を試み、以後順次に高度化～複雑化する。

∴ヒトの知的処理は複数エンジンである

- Rasmussenの3レベル

- 技能レベル～：瞬発～技能的反応

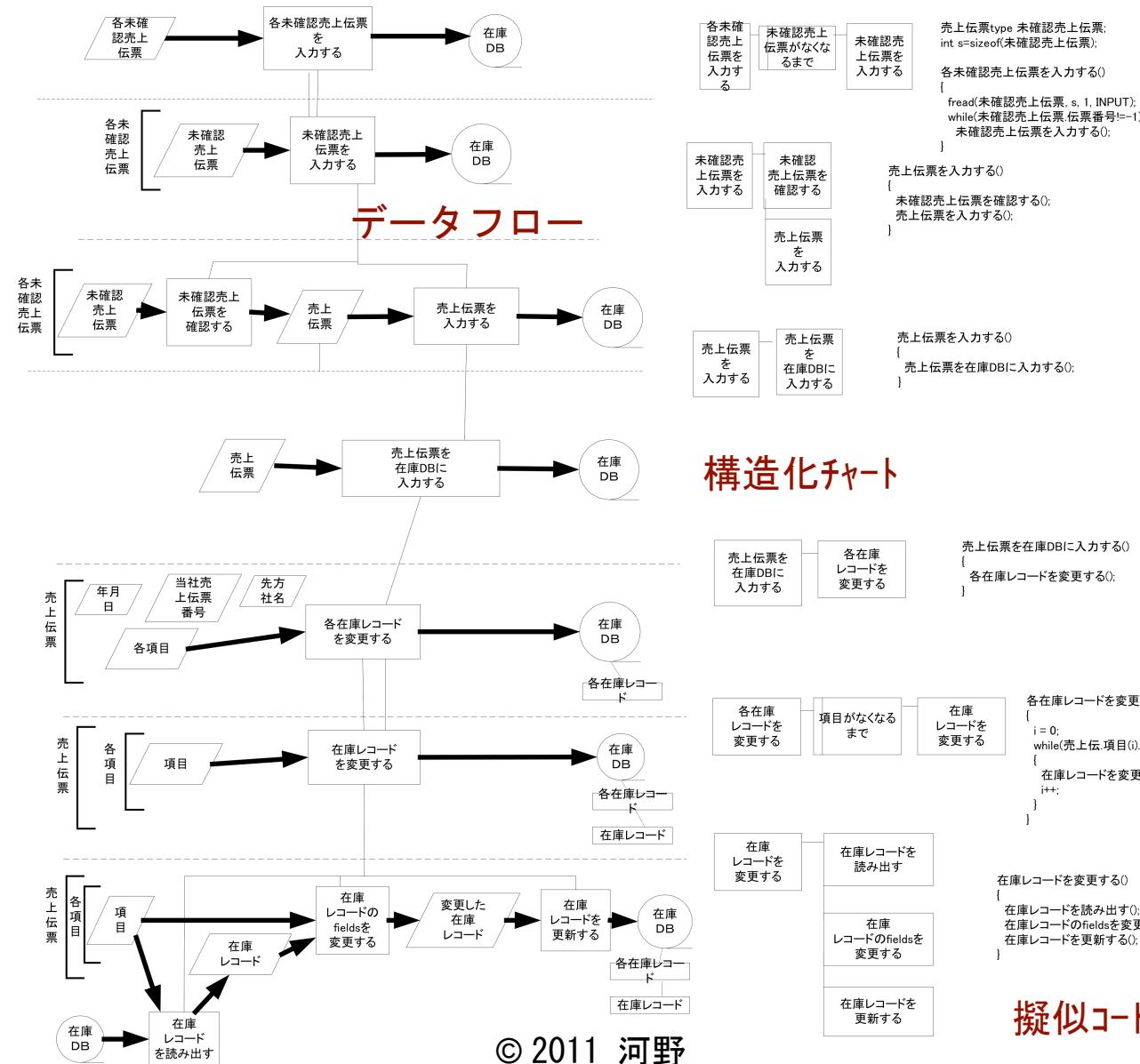
- ルール レベル～：ルールで処理する

- 知識レベル～：基礎的なレベルから積上げる

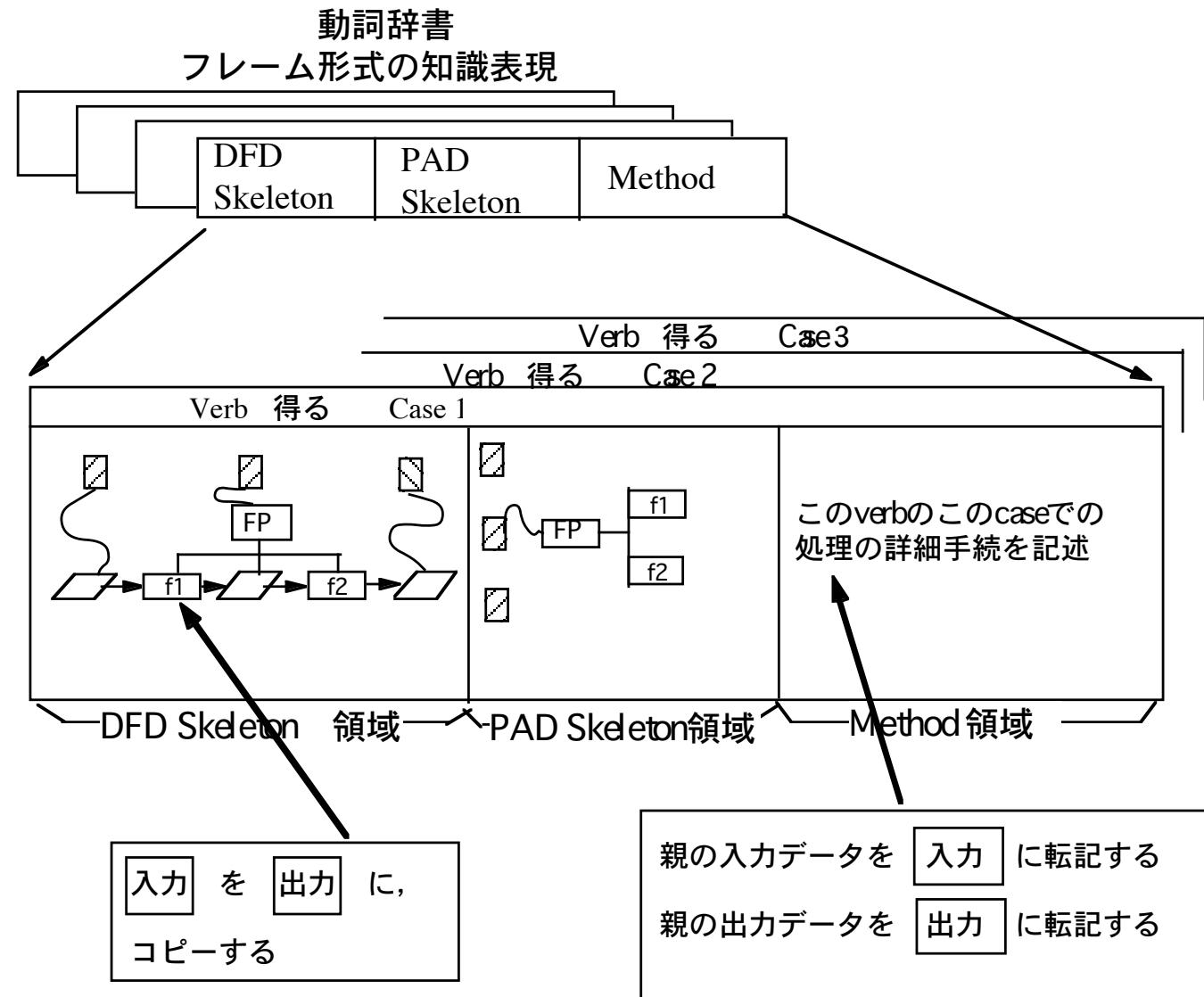
模範設計例から規則を抽出

在庫管理の基本的機能 + 第1次機能拡張

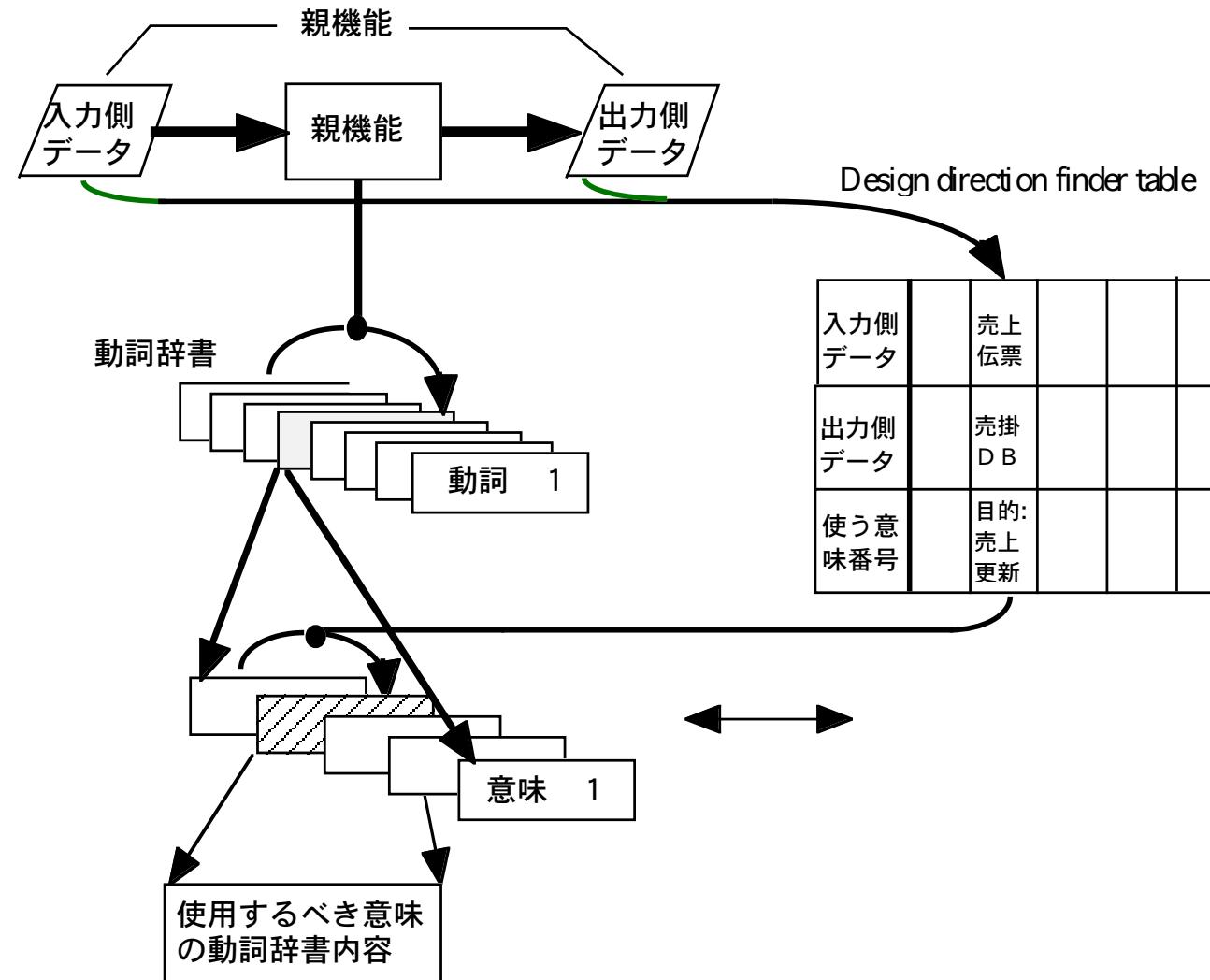
約700行 10本余り



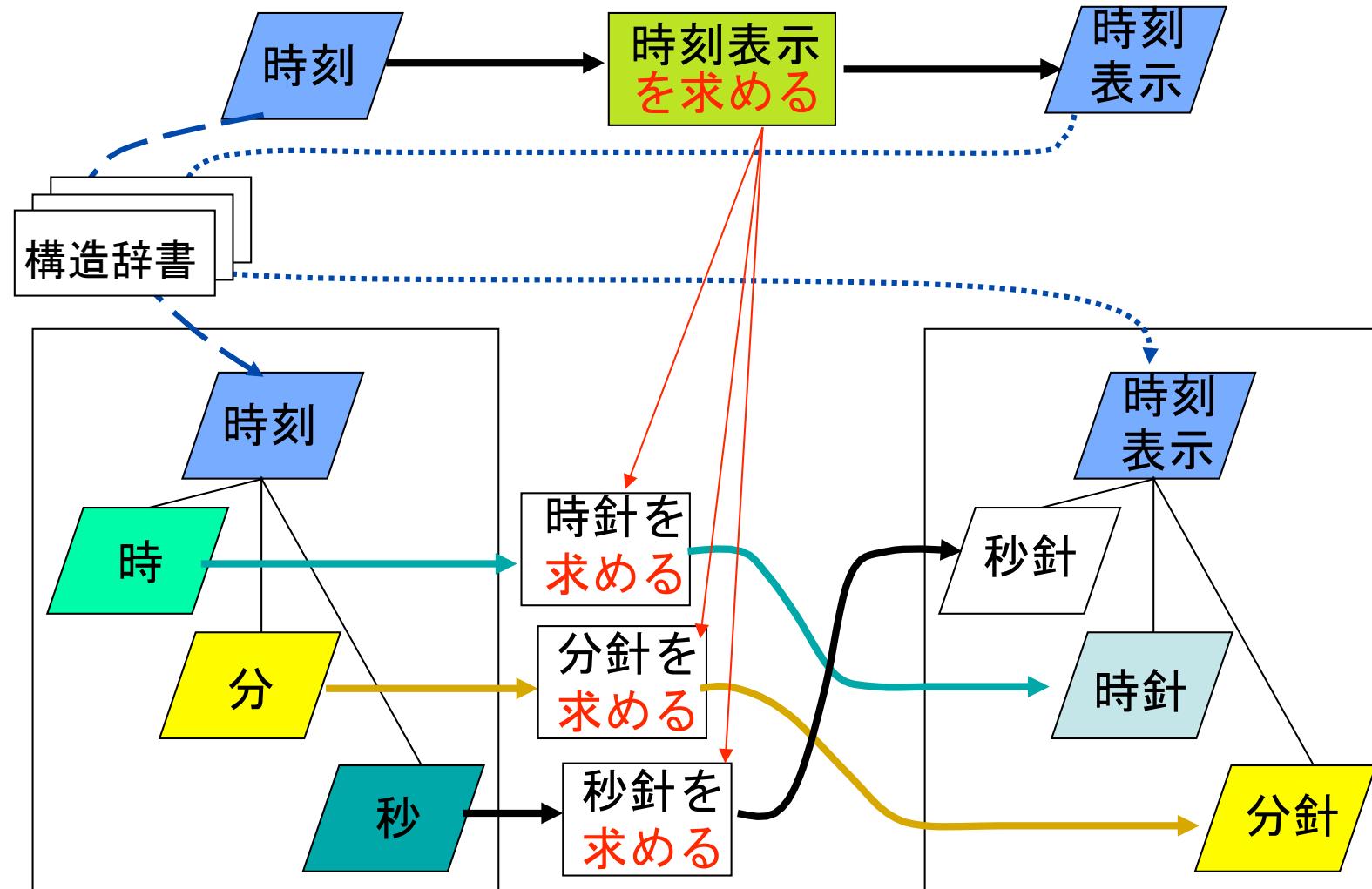
動詞辞書



動詞辞書の意味に応じた自動選択

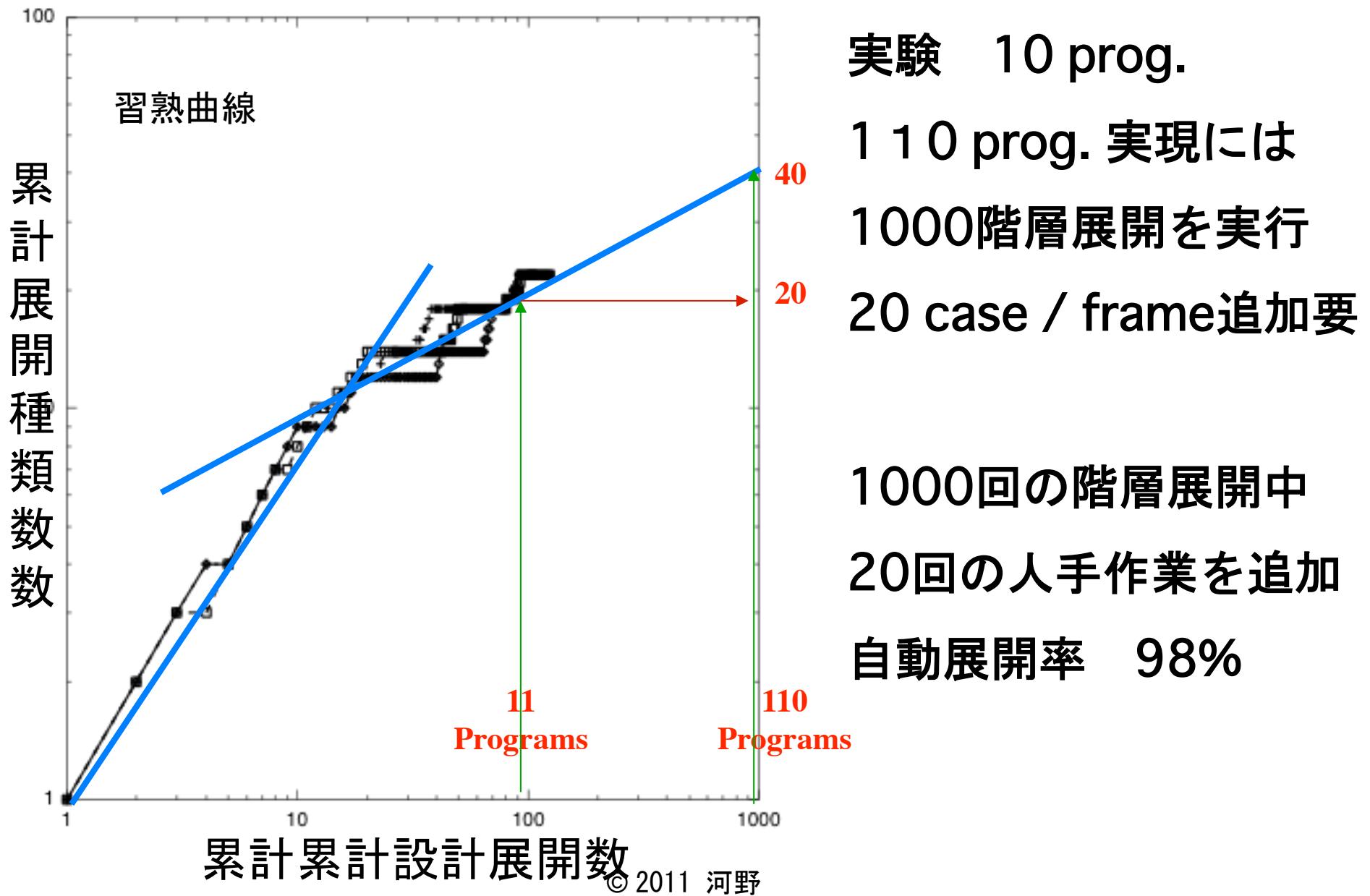


動作例 (JSP) と例外対策 (知識レベル)



例えば出力データの子が太短針, 太長針, 細長針であると動作できない。子の単位的名詞毎に基礎概念辞書を用いて太短針⇒時針と読み替え、全組合せを試みる悉皆的（人工知能的）処理を実行させれば解決できる。これは知識レベルの例である。

Creator '00の性能評価



Creator '00のコスト評価

実験総量	(87) symb	192 行	(97) symb	61 行	1.2K 行	332 行	
原単位当たり	4.6 <u>Symb</u> case	10.1 <u>行</u> case	9.7 <u>Symb</u> 帳票	2.3 <u>condx</u> entry			
区分	スケルトン	メソッド	構造辞書	DDFT	図形処理部品	前処理	
	動詞辞書						
	領域毎	適用毎	共通				
基礎合計 約 10K行		専用CASEツール (小計 約6K行)	専用CASEツール (小計 4K行)				

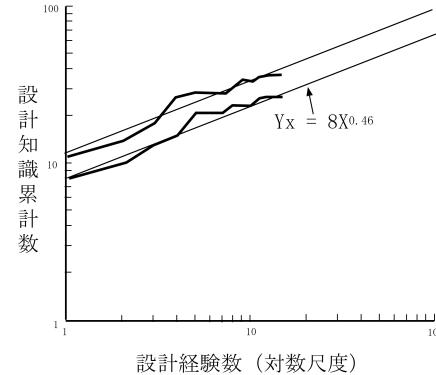
特徴

2.8K

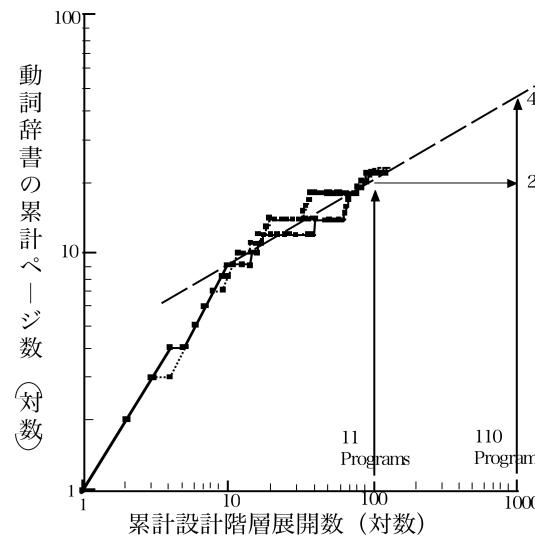
- 必要な知識量が少ない
 - 図形処理で簡単化
 - 単純再利用上に立つ

更に、大規模な確認実験、
異種の適用の実験要

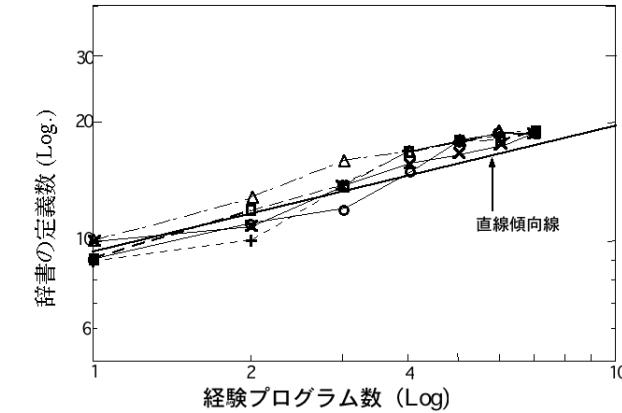
人に倣ったソフトウェア自動設計



技能レベル



ルールレベル



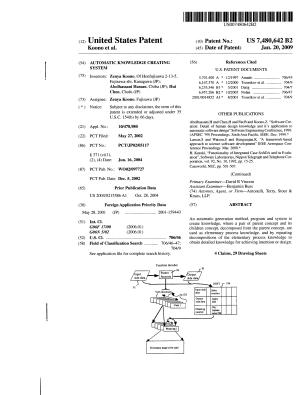
知識レベル

●本格的なソフトウェア
自動設計を
初めて実現した.
●中/米で特許を認可済.

一般技術者が原理～構成を理解
して実現可能で効果効用有り.



中国特許
権利証



米国特許
公報

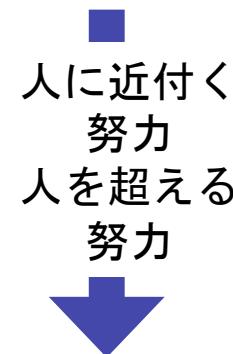
おわりに

- 10年掛かりでBreak throughに成功. P_{erformance}/C_{ost}大
プロジェクトメンバー達に深く感謝します.
- 今後の課題：高度化/広度化と試行/適用の継続拡大
 - ・再開：プロジェクト組織， 適用普及の拡大 ポスドク，社会人
日本システムの興隆

第5世代コンピュータ費用 500億円の1/10が欲しい !!

- ・子/詳細化したデータフローの他種生成法
- ・シーケンス性/組込システムへの拡張
- ・中流→上流での判断（設計の評価）
- ・他分野のDesign Automationとの連携

例：論理設計はプロパティ言語の1種に帰着
道路や橋梁の設計 基準が公式化



全体終了

ご清聴有難うございました.

忌憚無いご意見/異見を賜りたく.