

日本の合理性

京都大学大学院文学研究科 教授 林晋
平成19年9月18日 於 RIETI BBLセミナー

ちょっと自己紹介：林の研究歴

2

理学部

数学の研究所

理工学部, 工学部

文学部

70年代
数理論理学
(数学基礎論)



80年代-00年代
ソフトウェア科学
↓
ソフトウェア工学
形式的技法, プログラム意味論,
UML, Agile methods



00年代
情報技術の社会学
科学技術の社会学

00年代
数学史・論理学史

この話のテーマ

3

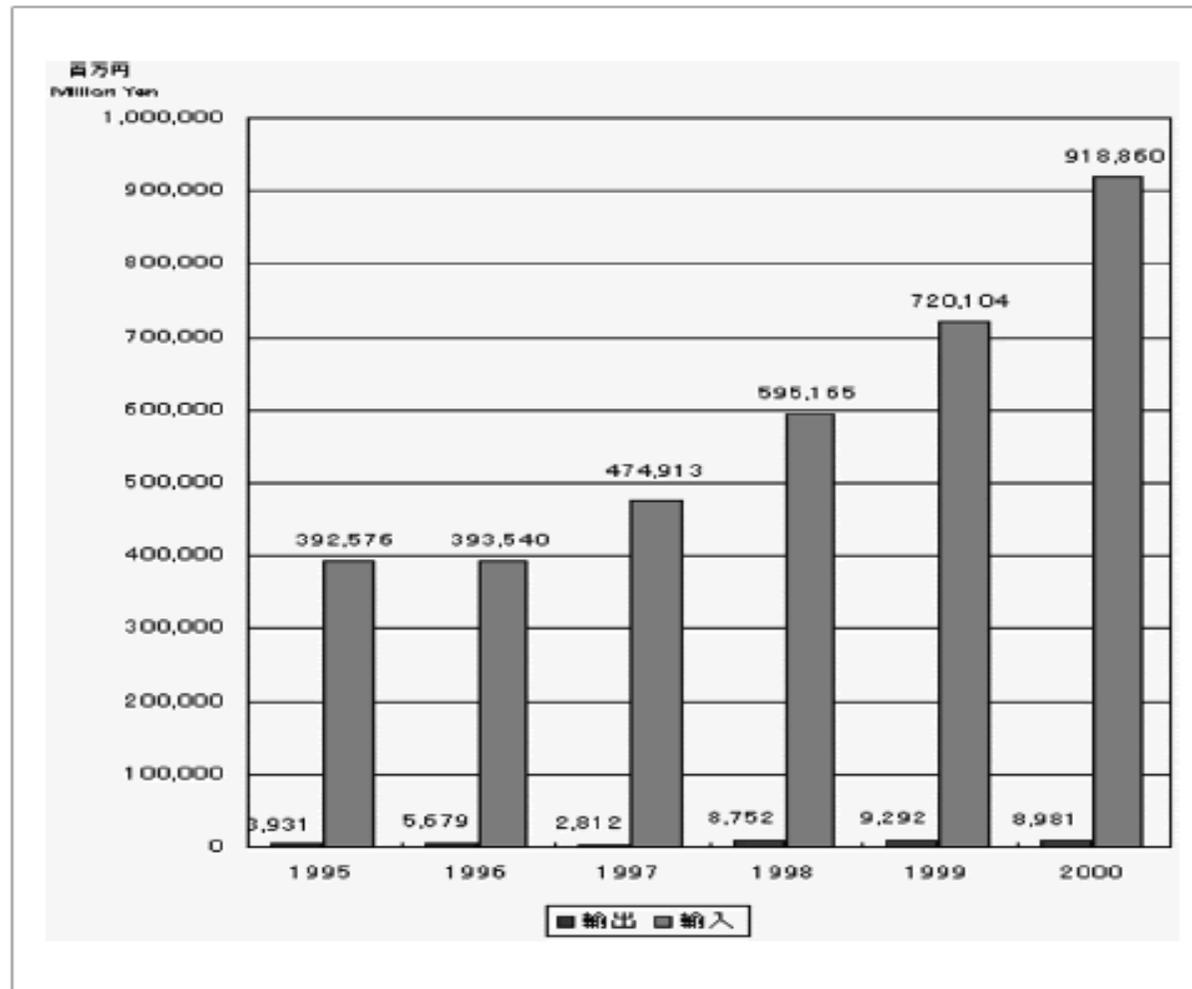
- 日本のソフトウェア分野(産業、研究)はハードウェアに比べれば遥かに弱い。それは何故か？
- これは良く問いかけられる問題
 - 産業構造が悪い。e.g. RIETI discussion paper 07-J-018
 - 日本人はモジュール型生産に弱い。(藤本隆弘)
 - 日本人が英語に弱いからだ。等々
- 林は、この問題を「日本の合理性」を鍵にして社会的アプローチで研究している。この未完の研究の現状を簡単に紹介。
- それに関連して、最近、中馬さんから投げかけられた問題について解決法ではない「答」を一言、最後に述べる。

日本のソフトウェア業界は弱い？

ソフトウェア産業は弱い： JEITA 2000年度 統計

4

図表1 ソフトウェア輸出入統計調査



文部科学省 NISTEP
科学技術動向 2004
林・黒川「2つの合理性と日本のソフトウェア工学」より転載

※ 2000年度実績

<http://it.jeita.or.jp/statistics/software/2000/4.html> より

日本のソフトウェア研究は弱い？

- 現在の日本のソフトで海外でも評価を得ているものは？
- ゲームソフトを除けば、プログラミング言語Ruby位だろう。
- IEEE が1996年の50周年記念の際に作成した、computingの年表には、ETL Mark III, パラメトロン計算機を始め、相当数の日本のハードウェアが掲載されている。純粹にハードウェアだけでない掲載項目は MITI の第5世代コンピュータプロジェクト関連の話くらい。
- その分野の研究者の一人としても、「特別に良くはない」というのが実感。Turing award 受賞の可能性のある日本人の顔は全く浮かばない。ノーベル賞級の受賞が珍しくなくなってきた数学、物理学、化学、医学、生物学などに比べると明らかにスターがいない。

弱さの理由は？

- **産業構造の問題は非常に大きい。**
 - ▣ これは実に様々な問題を引き起こしている。
 - ▣ 例えば現場の技術者の多くが疲れきっており、スキルアップの時間もないのも、この構造が一因。
 - ▣ この産業構造に敢えて抗しているソフトウェア会社で、良い成果が出ているところがある。
 - ▣ ただし、この説明が該当するのは主に受注ソフトのみ。特に、研究の弱さの説明には全く使えない。
 - ▣ しかし、一番の問題は、皆が下請け構造の問題を知らず知らず改善しようという兆候がないという事実！

弱さの理由は？（続き1）

- 日本はインテグラル型製品に強く、モジュール型製品に弱く、ソフトウェアはモジュール型である、という藤本説
 - 確かにモジュール思考に弱い傾向が見られる。しかし、LISPプログラマかつUNIXハッカーの一人だった林の目からは、AI文化、UNIX文化における、ソフトウェアの作り方こそ、インテグラル型、摺り合せ型に見える。
 - ただし、機械とは摺り合せ実施の構造と方法が違う。
 - 現場のチームリーダーからは、ソフトウェア技術の「摺り合せ能力」としてのコミュニケーション能力の重要性と、実際の現場の技術者の「摺りあわせ能力」の不足への嘆きが聞こえる。

弱さの理由は？（続き2）

- 同じく藤本説の情報転写説による記憶媒体への書込み難易度に基づく議論。
 - 日本人は情報を書き込みにくい媒体（自動車の鋼板等）を相手にしたときは強いが、ソフトウェアの媒体（CD, DVD）のように書き込み易いものを相手にする産業では十分力を発揮できない、という説。
 - ソフトが書き込まれた媒体が製品ならば、ダウンロードされたソフトは何なのか？ CD, DVDなどの媒体は機械製品を運ぶコンテナ、梱包に対応するもので製品自身ではないから、それへの書込み易さを論ずるのは無意味。

「日本人論」に基づく議論の危なさ

- 藤本氏の議論は80年代ころ(?)からの「もの作り神話」のように、日本人論をベースにした産業論と理解できる。
- これは一見心地よい議論だ。「私は日本人である。ゆえに本来的に、もの作りに優れている (^_^)」。
しかし、危険も孕む。
- もの作りは重要でない、という環境ではこの議論はどうなるか？「私は日本人である。ゆえに本来的に、もの作りができるだけだ。だから駄目だ (T_T)」

「日本＝もの作り」として良いのか？

- それに日本人は、本当に他と比べて本来的かつ格別に、もの作りに秀でているのだろうか？たとえばアメリカは日本より劣っているのか？
- また、「もの作り」と「ソフト作り」を対比する態度は正しいのか？
- さらに言えば、今後、「情報産業」は、急速に、その中心が「ソフト作り」から「サービス作り」に移行するはずである。そこではどうか？
- 問題は、それほど簡単ではない。実際、すでに、常識的な「日本/米国＝もの作り/情報産業」という図式からは説明できないことが起きている...

線形から非線形へ、そして...

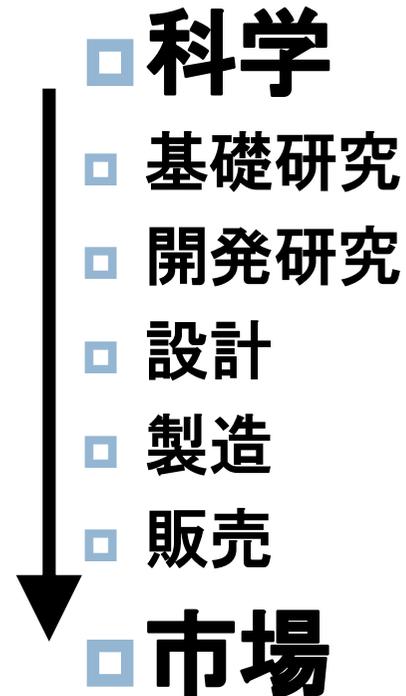
11

- S. Kline の線形モデル批判の翌年、Boehm が「ソフトウェアの線形モデル」ウォーターフォール・モデルを批判。
 - Kline,S.J.(1985). "Innovation is not a Linear Process"
 - **Linear model → Chain-Linked model**
 - Boehm B. (1986), "A Spiral Model of Software Development and Enhancement"
 - **Waterfall model → Spiral model**
- Boehm の Spiral model は、さらに2000年代になって Agile (アジャイル)と呼ばれる方法論に進化した。spiral model では、線形開発を螺旋状に繰り返すが、Agile では、さらに要求とプログラムが、常に動的に変化するネット状の関係で結び付けられる。(Chain-Linked model に類似)
- そして、...

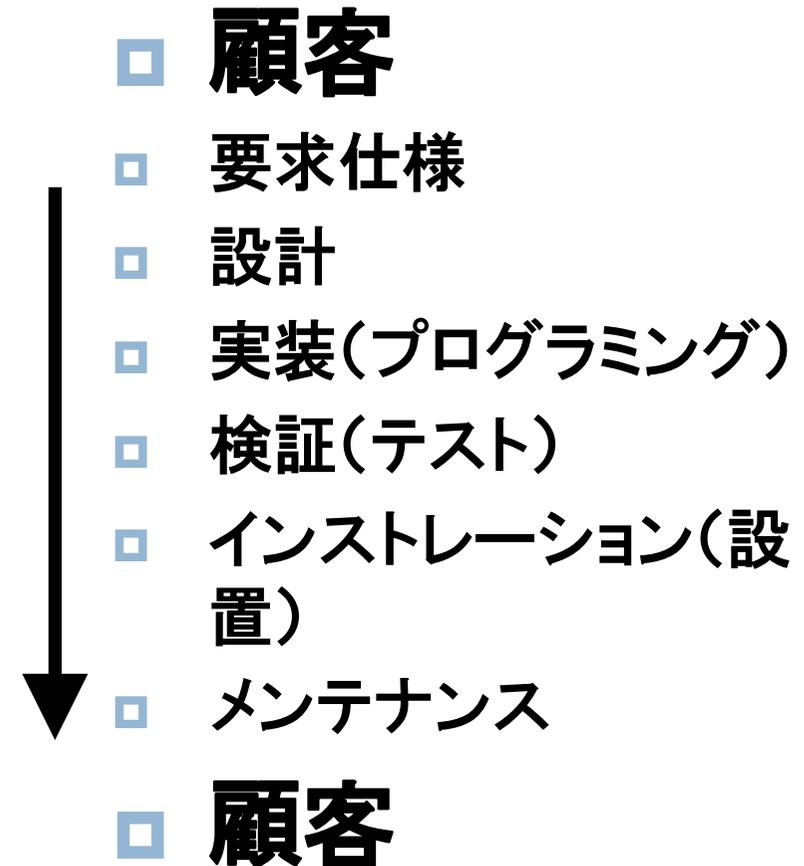
二つの線形モデル

12

□ Linear model



□ Waterfall model



Agile 方法論

13

- Agile (アジャイル)は、機械関連の生産工学(IE)、医療関係に詳しい人なら珍しい言葉ではないだろう。
- 80年代に米国が日独の競争力に注目して行った研究(lean はその一つ)から生まれた概念。
- 一つのものではなく、Agile Alliance (ソフトウェア開発法の連合)
 - Lean Software Development
 - Scrum
 - eXtreme Programming (XP)
 - Crystal
 - 等々

「日本型思考」に基づく最新の米国のソフトウェア工学

14

- Agile alliance の多くは、何がしかの日本的生産法、日本的経営学の影響を受けているか、意識していた。
 - Lean Software Development: その教本はトヨタ生産法の説明から始まる (Ohno, Software kanban, muda)
 - Scrum 野中郁次郎経営学からとった名前
 - eXtreme Programming (XP)
 - 提案者 Kent Beck, XP 2003 Panel で Lean Production の muda の消去と自分の方法を比較
 - Crystal:提案者 Cockburn は宮本武蔵の五輪の書を引用してソフトウェア作りを説明
- この方法は米国で大いに流行し、現在、日本でも流行しつつある。(Google で検索すると、沢山でてくる。特に平鍋健児氏の活動に注目！)

これをどう理解すべきか？

15

- 林は、これをどう理解すべきかを模索している。
- 理論的裏づけが弱く、仮説の段階だが、現状の理解は次のようなもの：
 1. ソフトウェアは論理的存在である。合理的存在といった方がさらに良い。その合理性には、社会学者Max Weber の合理性理論における「実質合理性：形式合理性」という一対の合理性が深く関わっており、Waterfall, Spiral, Agile という流れは、ソフトウェア開発において、実質合理性の重要度が増加していることを示す。

NISTEP 科学技術動向 2004年9月号 「二つの合理性と日本のソフトウェア工学」

http://sv20.wadax.ne.jp/~shayashi-jp/ws2005/hayashi_SEandHSS.pdf

2. この一対の合理性は相補的性格を持ち、特に環境の変化が激しいときは、十分な実質合理性に支えられない限り形式合理性は機能できない。また、実質合理性のみでは、巨大システムを動かすことができない。(20名以上のチームでは、Agileは十分機能しないとも言われる。)
3. 日本社会は伝統的に実質合理性の発揮に巧みらしい。それが自動車や家電製品という比較的安定した「自然物ともいえる形式」をもった製品の生産に十分発揮されたのが Toyota などの80年代までのケースである。

4. 日本は形式合理的巨大システムの運用が苦手らしい。これは藤本氏がモジュール型と呼んだものに対応する。一方で「自然に」形式が与えられたときに、それを運用することは、むしろ日本は上手だが、それは柔軟に実質合理性を発揮するからだろう。
5. しかし、「合理的形式」を作り上げる、たとえば、神学や哲学のような膨大な思想のシステムを作り上げるということは、日本は弱いらしい。ソフトウェアは「論理的存在」「形式合理的存在」であり、かつ、自明で自然な「形式」は作り難いか存在しないのが、現代のソフトウェア生産の現場である。このために日本はソフトウェア分野で苦戦をしている可能性が高い。

7. しかし、大野耐一たちが、TPS という方法と思想のシステムを作り上げたことは確かである。それは哲学や宗教の概念システムにまで続くものでは無かったが、明らかに倫理観などと結びついており、日本の形式合理性構築能力が、それほど低いものではない、むしろ生産の現場では、かなりのレベルに達している証拠と考えるとよいのではないか。
8. これらの合理性は、実は日本特有のものというよりは、西洋社会も本来もつ、普遍的な合理性の一つであり、明治の開国以来の西洋文明の一要素を日本が完全に消化しきり、さらには「実質合理性の運用に秀でる」というローカルな特性を生かすことにより、それを次の段階にまで進めたのだと理解した方がよい。
 - JITの大野の実現法は米国のスーパーマーケットにヒントがあった。
 - 自動化などのTPSの多くの要素は大野が語っているように豊田織機以来のものであり、戦前の繊維産業は日本を代表する輸出産業、つまり、国際競争にさらされた「近代への窓」であった。

9. 日本がソフトウェア分野に弱いことは確かなようだが、その意味を理解し、産業構造、社会通念、自己イメージの改革を推し進めれば、決して弱いままではないだろう。日本にも形式合理的巨大システムを作り上げる能力を持つ人々は存在する。その人たちを育てる(潰さない)努力をすべきである。そのためには産業構造の改革も必要だろう。
10. これらの合理性に関する能力は、一種の技術であるが、「理系」「文系」という分類からすると、むしろ「文系」の能力といえる。現場のチームリーダーたちが強調する情報技術者の能力が「文系」的な文章力やコミュニケーション能力であることは、その一つの現れである。

11. 中馬氏の「サイエンス型産業におけるイノベーション・プロセス調査」でも同様の傾向が見て取れるように、実はこの傾向はソフトウェア産業のみならず、多くの他の産業でも起こりつつある、あるいは、起こる可能性があるらしい。(E.g. システム・ナノ?)
12. その理由は、まだ良く分からない。あるいは、ソフトウェア産業のような sequential innovation, chain-linked model, agility 等を要求する産業での競争力の最大の根源は、急速に消費される製品にではなく、次々と新しい製品を作り出すシステムにあるからなのかもしれない。

13. そうならば多くの製造業は、むしろ音楽産業や映画産業、教育産業、医療産業、出版業、サービス産業のようなものになって行くのだろう。ソフトウェア産業内部でもその傾向が見える。米国の現代の花形ソフトウェア技術者の生活形態は、ますます作家や音楽家、医師、弁護士、経営コンサルタント、教育者、あるいは宗教家などに近づきつつある。彼らの中には、xxxxx evangelist という職名で企業に在職するものもいる。(evangelist = 福音伝道者。xxxxx の部分は企業が社内外で広めたい技術・アイデアの名前などになる。)
14. その時には、特許などの産業社会の基本的構成要素でさえ無意味なものになるだろう。たとえば、Bessen-Maskin の「特許制度が sequential innovation に悪影響を与える」という議論は、その表れの一つだろう。