

# デザインが生まれ出る瞬間<sup>とき</sup>

児玉公信<sup>†1</sup>

情報システムはどのようにデザインされるのだろうか。筆者は何度も情報システムをデザインしているにも関わらず、自らのデザインプロセスを言語化することは難しいと感じる。建築の世界では、デザイン論の一つとして「パタン・ランゲージ」が知られている。この手法は、形を変えて、情報システムやソフトウェアのデザインに適用されている。しかし、実際の建築デザインの場面でパタン・ランゲージが使われることはきわめて稀である。だとすれば、なぜ使われないのか。そして多くの建築家はどのような方法を使うのか。アーキテクトとの対話や文献調査を通して見えてきた課題を質的に整理し、情報システムデザインのあり方を考えてみる。

## The Moment When a Design is Emerged

KIMINOBU KODAMA<sup>†1</sup>

### 1. はじめに

#### 1.1 情報システムのデザイン論

情報システムはどのようにデザインされているのか。筆者はこれまでに、単体のソフトウェアから企業の基幹システムの再構築まで、多くの情報システムをデザインしてきた。ソフトウェアの設計法としては、かつてはワーニエ法やジャクソン法 (JSP, JSD) などを使い、やがて T. DeMarco の分析法や OMT を使った。UML による概念モデリングの経験<sup>1)</sup>と設計成果物が蓄積されるにつれて、近年は、ドメイン駆動設計<sup>2)</sup>やモデル駆動設計<sup>3)</sup>を行っていることになるのだが、実はそれは結果的にそう呼ぶのがふさわしい設計内容になっているからであって、始めから積極的にそうした設計法を使おうと意識しているわけではない。むしろ、世にある設計法は、基幹情報システムの設計法としては不十分であって、筆者独自の美意識に基づきプロセスを工夫している<sup>4)5)</sup>というのが実情である。しかし、その独自の設計法を他者に説明する、あるいは指導するという段になると、はたと困るのである。その美意識の裏にある基準は何か。

デザイン知識としては、構造化分析設計におけるモジュール設計の議論 (Cohesion and Coupling)、オブジェクト指向プログラミングの議論 (Law of Demeter, Open/Closed Principle, Design by Contract, デザインパターン<sup>6)</sup>)、オブジェクト指向モデリング技術 (Odell のモデル論<sup>7)</sup>、データパターン<sup>8)</sup>やアナリシスパターン<sup>9)</sup>が基本にあり、関係データベースについての議論、トランザクション制御の知識や SQL などの実装知識、それにソフトウェアアーキテクチャ<sup>10)</sup>と Web アプリケーション構築の知識が上乘せされる。だが、これらはデザインの過程で参照される要素であ

って、これらを使いさえすればよい情報システムのデザインができるわけではない。目的の情報システムのデザイン行為の中で、これらの要素をいつ、どこで、どのように適用すればいいかを知っていることは、デザインをする者の基礎的能力ではあるが、デザインプロセスを説明するものではない。

#### 1.2 デザインの科学の必要性

H. A. Simon は、その著書「システムの科学」<sup>18)</sup>の中でデザインの科学の必要性について言及している。デザインの学は、知的でなく厳密性が低いという理由によって、工学の世界から自然科学に追い出されてしまった。工学にデザインの科学を取り戻す必要があるという。

Simon は、デザインの過程は、「いかにして環境に手段を適合させるか」を考えることであるとして、定式化を試みる。人工物は外部環境と内部環境の接面にあり、そこでは外部から内部への求心的情報と内部から外部への運動的情報が交わされる。2つの情報は内部環境に蓄積され、それらが結びつけられる。外部環境への適合の目標は、内部の命令変数と外部の環境パラメタからなる効用関数の値を、制約条件の下で最大化する命令変数の組を見つけることとされる。しかし、最適化のアルゴリズムを提示するのは困難であり、デザイン行為にかかるコストも含めて、ヒューリスティックに代替案をいくつか出して比較する「満足デザイン」しかできない。

デザインの表現方法も重要である。たとえば、計算が本質的な関心であれば、ローマ数字よりもアラビア数字がよりよい。

#### 1.3 参照領域としての建築設計

残念ながら、現在の情報システムの構築と運用には問題があると言われる<sup>19)20)21)</sup>。情報システムのデザインを、歴史的に先行する建築のデザインと比較する<sup>13)16)</sup>ことで、そ

<sup>†1</sup> (株)情報システム総研  
Information Systems Institute, Ltd.

の行為の本質を明らかにできれば、デザインの評価やコンペティションが可能となり、よい情報システムの選別が進む。その社会的な効果は大きい。また、一棟の建築ではなく、都市計画やまちづくりに関する議論も先行しており、これと企業情報システムの中長期計画となぞらえる<sup>17)</sup>ことで、完備したデザイン論に挑むことができるかもしれない。

実はソフトウェアおよび情報システムのデザインに、建築の領域の知見がすでに使われている。C. Alexander のパターン・ランゲージ<sup>22)23)24)25)26)</sup>である。このパターン・ランゲージの考え方をソフトウェアに適用して<sup>28)</sup>、よいデザインを得ようという運動は、1993年にKent Beckたちが立ち上げたHillside Groupが中心となって、現在も活発に国際的なConference (PLoP, Pattern Language of Programming)を開催して、着実に成果を積み上げている。現在では、パターン・ランゲージをデザイン知識のシステムティックな表記法と見なして、ソフトウェアにとどまらず、組織設計<sup>11)</sup>などにも適用が拡大している。

しかし、実際の建築のデザインにおいてパターン・ランゲージはほとんど使われていないという。では、建築の領域では、どのような方法が有力なデザイン法なのであろうか。第一線で活躍されている建築家にインタビューし、文献調査も踏まえて、そのデザイン法を質的に整理する。ここから、デザイン法を大局的にとらえ直し、筆者がこれまで情報システム領域で実践してきた設計行為と対比しながら、本質的なデザイン論へと止揚する第一歩としたい。

本報告の構成は、第2章で、Alexander のパターン・ランゲージについて簡単にまとめる。第3章では、建築家のインタビュー結果を提示し、第4章において、一般的なデザイン論について述べる。第5章で企業情報システムの構築事例を述べ、第6章において総合的な検討を行って情報システムのデザイン論の構築を試み、第7章で全体をまとめる。

## 2. パタン・ランゲージ

ソフトウェアデザインのコミュニティでは、パターン・ランゲージをよいデザインのための、経験から導かれたノウハウの記述ととらえ、パタン記述のフォーマットについて議論し、実用的なパタンを集めることを重視している<sup>12)</sup>。しかし筆者は、文献(41)42)で述べたように、建築におけるパターン・ランゲージの本質は“時を超えた建築の道”を行くための「活動」であるととらえている。

### 2.1 建築を物語る

A Pattern Language<sup>23)</sup> (以下 APL) には 253 のパタンが収録されている。Alexander はこれをそのまま使えと言っているのではない。書名に不定冠詞が付いていることから分かるように、建築家やプロジェクトごとにパターン・ランゲージがありうることを示唆されている。実際、「オレゴン大学の実験」<sup>22)</sup>でも、「まちづくりの新しい理論」<sup>26)</sup>でも、APL

にはない新しいパタンが作られている。

APL によれば、パタン・ランゲージの使い方は、建築しようとするテーマに沿って、パタン・カタログの中から最初のパタンを探し、前後のパタンを参考にしながらパタンをいくつか拾い出し、必要に応じて追加・変更したものを「(the) パタン・ランゲージ」とする。これがそのプロジェクトのマスタープランとなる。そして、個々の建築物を表現するビジョンを、上に定めたパタン・ランゲージからパタンを語彙として用いて散文詩のように語る。これがプロジェクト言語 (Project Language) である。盈進学園東野高校のプロジェクト言語<sup>27)</sup>の一部を図1に示す。

#### 2-3 玄関道

– 正門より内境界に向けて「**玄関道**」がある。玄関道の両側には壁か樹木が並び非常に静かである。

#### 2-5 中央広場

– **第2の門**の内側には、**中央広場**がある。この広場は**大講堂**とともに構成され、講堂の正面は庭に向かって



図1 プロジェクト言語の例<sup>27)</sup>

このプロジェクト言語は、アーキテクト (Alexander たち) と、施主 (学校の理事)、学生の共同によって作られた。この“詩”を読むと、まるで建物群の中を歩いているかのように感じる。これに基づいて配置を計画し、現地で原寸設計をしていったという。

### 2.2 設計原理

Alexander の建築プロジェクトでは、パターン・ランゲージだけが使われるわけではない。最初に、プロジェクトを実施するに当たっての計画原理が制定される。「オレゴン大学の実験」<sup>22)</sup>では、①有機的秩序、②参加、③漸進的成長、④“パタン”、⑤診断、⑥調整である。「パタンランゲージによる住宅の建設」<sup>25)</sup>では、①アーキテクトビルダ、②ビルダーズヤード、③共有地の共同設計、④個々の住宅のレイアウト、⑤一步一步の施工、⑥コストコントロール、⑦プロセスの人的なリズム、となっている。

そして、これらの計画原理を維持・改善する組織が作られる。「オレゴン大学の実験」では、パタンの制定、改訂を諮る組織「計画評議会 (Planning Board)」が設けられた。これは、施主 (大学)、利用者 (学生)、アーキテクトの三者からなる。施工者が含まれないのは、直営方式を取っているためである。オレゴン大学では、この体制が今も継続しており、漸進的成長が実現しているという。しかし、盈進学園でも、当初はこのような体制が組まれたが、ゼネコンを入れた時点で体制が崩れてしまったとされる<sup>37)</sup>。

### 2.3 柔らかいマスタープラン

都市や建築物は 100 年の時を越えて存在する。それゆえ、現時点で厳密な基本計画 (マスタープラン) を立てることには意味がない<sup>40)</sup>。基本計画を立てたととしても、柔らかい大筋の計画、すなわちガイドラインとして利用されるのが適当であろう。この観点から、Alexander の方法は、有機的秩

序をもたらすためのデザインコードであり、ガイドラインとして実効性を持つものと考えられる。パタン・ランゲージのガイドラインとしての活動を整理すると、図2のようになる。

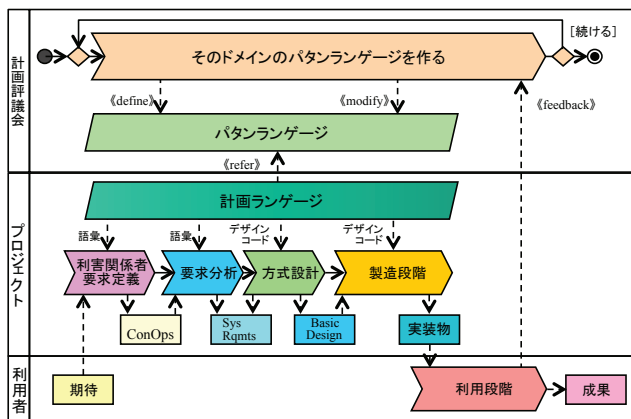


図2 ガイドラインとしてのパタン・ランゲージ

## 2.4 使われないパタン・ランゲージ

Alexander がパタン・ランゲージのアイデアを発表し、それに基づいてオレゴン大学<sup>22)</sup>やメヒカリのプロジェクト<sup>25)</sup>を精力的にこなしたのは、1970~80年代であり、本邦のアーキテクトの間でも「一世を風靡した」という。「オレゴン大学の実験」を読み、感銘を受けた盈進学園の当時の常務理事が直接 Alexander に依頼をして、東野高校を建てたのが1985年である。

しかし、実際のところ、Alexander 自身の建築プロジェクトは、工期延長、予算超過で必ずしも成功したとは言えない。盈進学園でも、Alexander は施主が建築を主導する「直営方式」の継続を主張して対立したが、工期が足りなくなり、しぶしぶゼネコンを導入し、突貫工事で仕上げたという<sup>37)</sup>。アーキテクトの間でも「あれでよくできたものだ」というのが率直な感想という。学校側も、職員室がない、運動場も体育館も大講堂も狭い、改築ができない、メンテナンスが困難といった問題を上げる。このような評価から、多くのアーキテクトはパタン・ランゲージに否定的である。一方、学生の応募数が移転前に比べて4倍となり、それが継続していることを見ると、学生の評判はよく、「生き生きと生きる」<sup>24)</sup>ことを目指す建築物としては成功と言えるかも知れない。

施主が主体的に設計と施工に関わる手法に共感するアーキテクトもいて、川越一番町の商店街の整備計画<sup>28)</sup>、谷中商店街の整備計画や真鶴町「美の条例」<sup>29)30)31)33)</sup>などの実践も行われている。しかし、施主を始めとしてステークホルダが主体的に関わる方法としては、L. Halprin<sup>34)</sup>が Alexander に先駆けて住民との「ワークショップ」手法を実践していた。

このように、建築の世界では有名でありながら、主流ではないパタン・ランゲージが、ソフトウェアの世界でもて

はやされるようになったのは、オピニオンリーダーである Beck によるところが大きいと思う。彼はオレゴン大学出身であり、1979年から8年間、まさにパタン・ランゲージの活動の中にいたのである。

## 3. アーキテクトの方法

建築の世界でパタン・ランゲージが一般には使われていないとすると、建築物はどのように設計され、都市計画はどのようになされているのだろうか。事例を基にこれを明らかにして、ソフトウェアや情報システムの設計においてその手法を参照したい。

事例は、札幌市が1986年から1990年にかけて、郊外の厚別区に作った工業団地である札幌テクノパークに、地元のIT企業が事業所として建てた二つの建物である。これを取り上げたり理由は、地理的に近接しており、気候、立地条件、使用目的が似ていることから、かえってデザインの本質部分についての比較がしやすいと思ったからである。それぞれの設計を担当した2名のアーキテクトには、2回のインタビューを行った。1回目は一般的なデザインに対する考え方、2週間程度の間があって、2回目はそれぞれの建築がどのようにデザインされたかを聞いた。以下に、インタビューメモからの要点をまとめる。

### 3.1 事例1: コンセプトワード

第2テクノパークにI社の社屋として建てられた作品。当時、ゼネコンF社の社員アーキテクトO氏による。

**第1回インタビュー** 設計の第一段階は「コンセプトワード」を決めること。コンセプトワードは、施主との合意事項を鮮明に表す1~2語のことで、施主の望んでいることを引き出す言葉。たとえば、「収穫」。アーキテクトは、なぜその語が適切なのかを説明できること。これを施主に植え付ける。次の設計段階で迷ったらコンセプトワードに戻って考え直す。コンセプトワードを得ることによりかなりのエネルギーを使う。そのために何をしたらが提案に現れる。

設計は、究極的にはアーキテクトが一人で決めることだが、常に代替案を考えておく。社内コンペティションはアイデアを引き出す仕掛け。何が施主にとっての価値なのかを考える。

シーンを想定して物語を読むように建物のイメージを浮かび上がらせる。鳥が降りていくように、大きな視野から徐々に拡大していくような見方で語る。

コンセプトと要件の2点が与えられて、それをどのようにつなぐかがデザインであり、そこに自分自身が外化する。制約とは、地勢、歴史、地形、雨量などであり、施主が重視する制約を考慮していないことがわかると、プレゼンからさせてもらえないこともある。

よい設計とは、設計者が何をやるうとしたかが伝わるものである。たとえその意図が成功していなくてもよい。

**第2回インタビュー** デザインは、たくさんいるステー

クホルダの認識を、筋道をつけて、あるいは杭を打って、囲い込んでいくイメージ。ちょうど、鶏があっちに行ったりこっちに行ったりしているのを、一か所に追い込んで行って、最後にカゴをかぶせるように、カゴはどこに置かれてもいいが、ステーキホルダを取り逃がしてはいけない。途中で、施主に「これは嫌い」と言われたらそこで終わりになってしまう。

I社の社屋は、コストも含めた設計要求事項が与えられてのコンペであった。要求事項には、(敷地条件、必要部屋数などのほかに)企業アイデンティティ、建築の狙い、デザインに対する姿勢なども含まれていた。さすが外資系と思った。設計者としては、とっかかりが多い方がやりやすい。ここから絞り込んで得られたコンセプトは「大地」。(内地から来た会社が)北の大地に根を下ろし、風雪に耐えるイメージ。カーテンが風に揺れるようにカーブを描くガラスウォールは、光をしっかりと取り入れる。

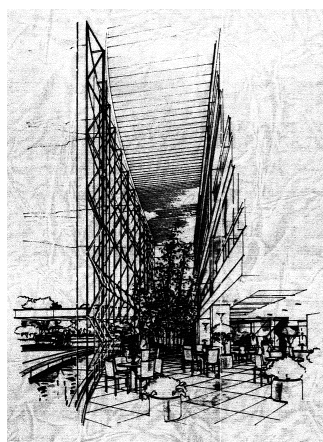


図3 I社社屋のインナーパス (O氏提供)

インテリアデザインは別の担当。そのコンセプトは「収穫」で、「大地」からの連想である。収穫のイメージからカラーリングを決めていった。

### 3.2 事例2: 理屈よりも五感

第1テクノパークの敷地が隣接するd社とB社、それぞれの社屋として建てられた作品。当時、新進気鋭のアーキテクト5人によるグループ設計。メンバーのM氏に聞いた。  
**第1回インタビュー** 自分は基本的にファンクションデザインの立場である。設計条件書を分析し、根拠を得て、そこからひらめいたイメージをスケッチに書き起こす。これがコンセプトスケッチ。

設計の理屈とコンセプトから来る五感をバランスする。自然を見て、自然に問いかけ (ask nature)、一定の作法を持って自然と関わっていく。デザインイメージ全体が、その関わり方のストーリーに一致すればいいだけ。

**第2回インタビュー** (筆者が、デザインはどうして決まるのかという問いかけに対して) 一般論として、3つの型がある。幾何学や自然にあるものの形をきっかけにする型、要求された機能を満たすことを考えた上で感性を加える型、生活や行動の観察から考える型がある。人の行動があるからアーキテクチャが存在する。これらの区別は厳密ではないし、同時に複数のやり方をすることもあつた。ファンクションデザインと言ったのは、この2番目のこと。

実際に建築を進めるには、意匠(好きな言葉ではないが)、

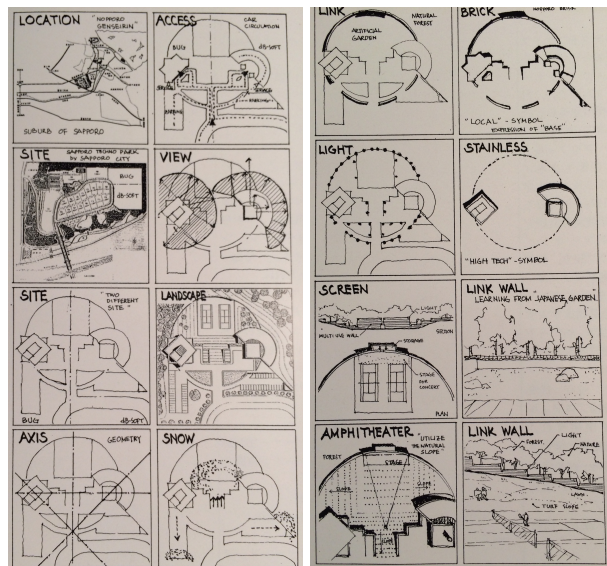
構造、(空調、水、電気などの)設備の3つの側面について、それぞれの専門の設計者と協働する必要がある。グループ設計と言う。意匠を担当する自分たちが全体を指揮することが多い。最近では環境への配慮が重視されるので、設備側からの制約も強くなってきている。

ビジネス的に関係があるd社とB社の経営者と何度かヒアリングを行った結果、2社の社員が気持ちよく働ける環境を希望していることがわかった。希望を叶えてあげたい。コンセプトは早い段階で「LINK」に決まり、互いに隣接する敷地の境界を取り去って、すぐ裏にある原生林を活かすことを考えた。景観の中に調和した一体感をマスタープランに盛り込んだ。ここからコンセプトスケッチ(図4)をいくつか描いて、2つの建物が向かい合うような形とし、材料は同じものを使うこと、周辺にテニスコートやステージを設けることとした。最終的に、作品名もLINKとした。

図4 d社とB社の社屋のコンセプトスケッチ(M氏提供)

外装に関して、下の部分は地元の土で作ったレンガを積み、上の部分の外壁は高級ではないアルミにあえて傷を付けてトーンを変えて並べることで、これもレンガを積み重ねたような印象にした。コストダウンになっている。建築基準法との兼ね合いやコストプランなどは、コンセプトワークの際に考慮しておく。

内装は、その当時では先進的な外断熱にして(容量を確保し)、建物中には日光を十分取り入れるようにした。柱は



なるべく使わない。2つの建物のアトリウムには、ともにイサム・ノグチ氏の彫刻を置き、インテリジェントな雰囲気や中和するようにした。B社の彫刻は「つくばい」で、彫刻の側面を水が伝って落ちて、その音がオフィス全体に聞こえるようにしてある。

コンセプトワードは重要だが、現実的な事情でゆがめられることもあり、こだわりすぎではいけない。

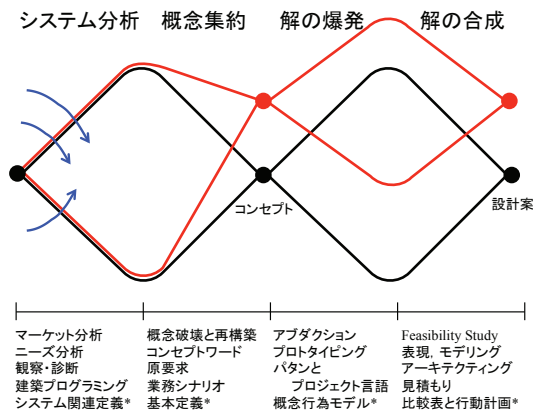


図5 デザインプロセスのモデル：Double Diamonds<sup>48)</sup>  
 注) 赤色および下部の作業例は筆者の解釈である

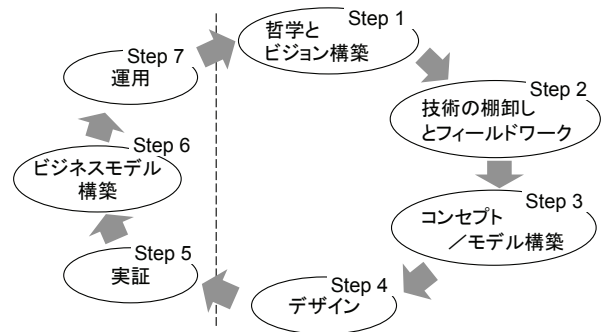


図6 デザイン思考における創造のプロセス<sup>52)</sup>

によって大きく異なることはないだろうが、集約されるコンセプト（ワード）が大きく異なることはありえる。この結果、設計案が大きく異なることになる。そのイメージを図5に赤色で示した。

## 4. さまざまなデザイン論

二人のアーキテクトのナラティブデータを理解するために、いくつかのデザイン方法論に照らして解釈する。

上述の2例から想定される共通の設計プロセスは、コンセプトを設計の根拠、あるいは発想の原点としている点である。事例1ではそのコンセプトを得て施主と合意する過程を重視する。コンセプトが合意されると、一気に設計イメージ（スケッチ）を爆発させる。これは、事例2でもLINKを原点とし、それを基に五感を研ぎ澄まして創造性を全開する過程ととらえられ、理屈ではないアブダクション<sup>49)</sup>の過程に見える。

### 4.1 Double Diamonds

このようすは、Design CouncilのAndrea Siodmokたちが提唱するデザインプロセス<sup>45)47)</sup>である“Double Diamonds”（以下、DDと略す）を彷彿とさせる。このモデルは、デザインプロセスにおいて爆発と集約が2度起きることを直感的に表している。まず、設計者は設計条件書を基に、マーケット分析やニーズ分析を行って要件を爆発させる（システム分析）。事例1のアーキテクトは、「次々と条件を取り込む」と表現した。建築の世界ではこの過程を「建築プログラミング」<sup>43)44)</sup>あるいは「ブリーフィング」と呼ぶことがある。

爆発した要件をごくわずかのコンセプトワードに凝集する（概念集約）。このコンセプトワードをひらめきの種にして、一気に解を生成する（解の爆発）。コンセプトワードは、解の爆発を誘発するほどの力強さを十分に持っている必要がある。そして爆発した解を検証しながら、一つの設計案に合成する（解の合成）。コンセプトを、あえて緩めに定める設計者もいる。最初の菱形が、より小さな診断の菱形と分析の菱形に分けられることもある<sup>47)</sup>。

DDモデルの対象範囲は設計案の提示までであり、そこからの施工設計、施工の実施、運用は含まない。

筆者の解釈では、「システム分析」の内容はアーキテクト

### 4.2 デザイン思考

IDEO社は製品の設計では世界の主導的立場にある。彼らは「デザイン思考」を標榜し、人の行動に着目し、ビジネスモデルを含めた設計を考えている。そこで叫ばれる「動詞のデザイン」という合い言葉が、それをよく表している。

奥出直人<sup>52)</sup>は、IDEO社の方法を拡張して、「創造のプロセス」を提案している（図6）。以下に簡単に説明する。

STEP1：その設計行為に関する哲学とビジョンを構築する。哲学とは設計者が設計に当たる態度と価値観であり、ビジョンとは何を作りたいかの欲求である。これらをチームで共有する。

STEP2：技術の棚卸しとフィールドワークを行う。ビジョンを実現できそうな技術を確認するのと並行して、それに関わる現象が起きている現場に行きエスノメソドロジックに観察する。真のニーズの発見につながる。

STEP3：フィールドワークの結果を基にアイデアを出し、コンセプトおよびモデルを構築する。アイデアを出すためにブレインストーミングやプロトタイプングが行われる。コンセプトとはビジョンを達成するための基本概念であり、一言で答えられるものでなければならない。モデルはその基本構造の記述であり、粘土や紙、映像などで表現される。

STEP4：デザインの過程で、上で考えたコンセプトやモデルを、実際に使えるものとして洗練し、確定していく。

STEP5：実際に制作して使用し、評価してもらう。

STEP6：その製品に関するビジネスモデルを描く。

STEP7：そのビジネスモデルを事業として運営し、設計社にフィードバックする。

DDモデルが設計案の作成までを述べているのに対し、創造のプロセスは、運営からのフィードバックまでを対象としている。これは、デザインが学習のプロセスであることを示している。また、一つのコンセプトに凝集するというよりも、複数のコンセプトとモデルを作って、明示的に選別するようにしている。

a 視覚だけでないことに留意。

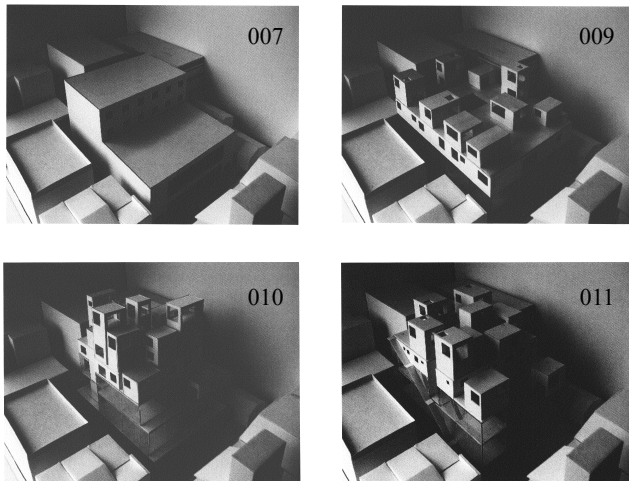


図7 超線型プロセスによる模型の変化<sup>38)</sup>

#### 4.3 超線型プロセス：迷走から洗練へ

藤村龍至<sup>38)</sup>は「超線型プロセス」と呼ぶ高速プロトタイプピングによる設計プロセスを実践している。これは、設計条件書から大雑把な設計案を模型で作り、その評価と設計案の変更を何度も高速に繰り返して最終案に到達しようとする方法である。模型は思考過程を外化したものであり、100案を越えることもあるという。

図7は、文献38)のBUILDING Kの事例から一部を引用したものである。写真の右肩に案のシーケンス番号を付けた。最初は、敷地面積から自動的に定めた形状（ボリュームスタディ）から出発し、施主やアーキテクト自身の評価に対応しながら徐々に形を変えていく。初期の模型の形は大きく変転する。図を見て分かるように、007と009の間には劇的な変化がある。構造設計者の意見が反映され、形が決まると、その後は小さな変化が繰り返されて最終案となる。

超線型プロセスを、DDモデルと創造のプロセスと照査してみよう。DDモデルに対して、超線型プロセスは、その名のおり二次元の膨らみを持たないので、要件の爆発は経時的な案の変転過程に置き換えられる。これは設計が定まらない迷走状態に見える。案が一定の安定を見た後、「解の爆発」に相当する案の変化は、「解の合成」と融合して進められ、順次的な洗練のプロセスとなる。

創造のプロセスに対して、超線型プロセスでは、STEP1と2が省略され、STEP3のモデル作りが何度も頻繁に繰り返されてデザインが完成する。

#### 4.4 再びパターン・ランゲージ：解の再利用

パターン・ランゲージを再度取り上げ、その手法を創造のプロセスに照らして定位してみる。STEP1は設計原理の制定に対応する。パターンは技術の棚卸しそのものと理解することができ、STEP2に対応する。プロジェクト言語と現場での原寸設計はSTEP3のフィールドワークとコンセプト化を行っていると思わせる。STEP5-7の実証し、設計者にフィードバックする過程は、まさに漸進的成長のことであ

る。パターン・ランゲージのプロセスは、DDのモデルのように明確なコンセプトの凝集を持たない。

多くのアーキテクトはパターン・ランゲージを使わない。理由は、事例2のM氏によると「パターン・ランゲージは関係性をわかりやすくしたものであり、創造の手がかりにはなるが、あくまでデザインの背景としてある。デザインそのものを生み出すわけではなく、マニュアル的に使うとつまらないものになる。創造はもっと別のひらめき」であるという。この理解は、コンセプトの凝集をどう捉えるかの違いに根ざしていると感じる。

### 5. 情報システム設計の質的事例

筆者がアーキテクトを務めたA社の基幹システムの再構築の案件（事例3）を振り返って、そこで起こった設計イベントを中心に概略を述べる。A社は業種トップの製造業である。守秘義務の関係で、開発時期や具体的な設計成果物の提示はできない。

#### 5.1 提案期

本件は、A社のCIOからの提案依頼に対して、生産管理の一般モデル<sup>50)</sup>をA社の業態に適合するように拡張し、アジャイルプロセスで実装することで応じた。この提案が採用され、設計作業が始まった。

#### 5.2 第1期

ある事業所のローカルな生産管理システムの再構築が最初の課題であった。A社のビジネスは、製品の製造ではなく、製品の検査を行って検査報告書を提出するものであった。最初のモデルは、検査報告書を製造するプロセスを考えた。しかし、検査報告者は必要に応じて何度も印刷される。悩みながらも、プロトタイプシステムを開発した。

あるとき、製造プロセスはいくつかの工程を経て、指示された製品仕様を実現するプロセスであり、検査プロセスは製品仕様の値を計測によって明らかにするプロセスであることに気がついた。生産管理の一般モデルは、製品仕様値を明示することが可能になっていたことと、ものを作らないプロセスも扱えるものだったので、仕様値を埋める処理を追加することで検査管理のモデルはほぼ完成した。

#### 5.3 第2期

経営トップからの構想が提示され、ローカルシステムではなく、全社システムとして再構築することが決まった。アーキテクトは、検査管理ドメインだけでなく、注文管理ドメイン、検査対象の製品管理ドメイン、工場ごとの検査実施（MES）のドメインを想定し、IEC 62264の業務階層モデルを参考に、管理階層に沿ったドメインの分割と、それらが相互作用して動作するアーキテックチャ原案を施主に提案した。

施主はこの案を採用し、ドメインごとにモデルを作成していった。同時に、ドメイン間のインターフェースの設計原則を設計した。

機能設計をB社に依頼したが、モデルを理解しないまま、聞きとりによるユースケース記述を行い、As-isに引きずられた設計となったため、施主はこれを良しとしなかった。

#### 5.4 第3期

経営トップがオーナーとして指揮を執ることになった。オーナーは、ドメイン駆動開発<sup>2)</sup>に準拠することを明確に宣言し、A社の担当が、アーキテクトの書いたモデルに整合するようにユースケース記述を一部やり直した。

A社のリーダは、設計されたアーキテクチャの妥当性を早期に検証するため、重要なユースケースを取り上げて先行的に実装することを提案し、C社とその開発・稼働フレームワークを採用した。C社は記述されたアーキテクチャとモデルに忠実に実装した。結果は満足できるものであり、アーキテクチャもモデルも妥当性が示されるので、本格開発に移ることとなった。

#### 5.5 第4期

本格開発の体制が取られ、C社のほかに、2社が参加することになった。2社はC社のフレームワークの採用を嫌ったが、最終的にオーナーの裁定に従うことになった。3社でドメイン別に開発担当を分けたが、後発の2社は、モデルの理解、フレームワークの理解に時間がかかり、その担当分の実装は遅れた。

実装に向けて要求が詳細化するにつれて、主要なユースケースが実装されてくると、曖昧な仕様が明確になり、モデル図、アーキテクチャ図は何度か補正された。この結果、設計はほとんど安定するものとなった。

それでも2社の開発は芳しいものではなかった。開発が遅れると、その隙を突いてAs-isからの揺戻しや、一旦はあきらめた細かい要求が吹き出してくる。ペンドは、遅れを取り戻すために弥縫策を重ね、設計が崩されていった。

#### 5.6 第5期

問題を残しつつも開発はほぼ完了し、稼働準備の段階に入った。顧客への説明、契約の切替え、製造現場の体制、対顧客窓口の体制、データの準備など、稼働開始に向けての準備が進められた。

### 6. 総合的検討

#### 6.1 真の施主の関与

基幹システムの再構築を伴うほどのビジネス改革を実行するには、真の施主が設計に積極的に関与することが必須である。しかし、いきなり経営トップの直接の関与が可能になるわけではない。少し時間をかけて、プロジェクト組織への介入のあり方を模索する必要がある。

アーキテクトも、これと並行してキーパーソンとの信頼関係の構築、啓蒙活動、活動成果物の提示などの努力が必要である。キーパーソンの社内的地位が弱いときは、社内的地位がきわめて強いスポンサーシップが必要である。

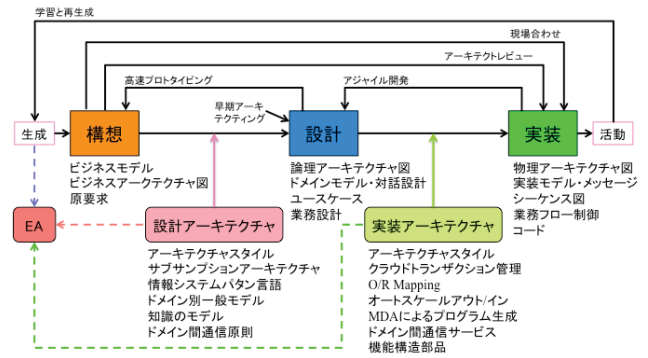


図8 情報システムの設計と実装

#### 6.2 デザインプロセスの設計

情報システムの設計において、かつてDeMarcoは、情報システムの設計を機能 (function)、データ (retained data)、状態 (state transition) の3つ視点をモデル化するのがよいとした<sup>53)</sup>。これらは現代的には、機能的、静的、動的側面のモデルとなる。しかし、これだけでは現代の情報システムは動作しない。これらは、建築で言う意匠に相当する設計行為でしかなく、情報システムが、要求される品質や性能で動作するためには、構造設計や設備設計に相当する、適切なサブシステム (ドメイン) への分割とその間のインタフェース、サーバの能力計画と実際の割付け、バックアップの方式などの設計の取込みが必要である。

図8は、ビジネスモデルの生成から運営の活動までの設計成果物の変換過程を明示するために、情報システムサイクル<sup>4)</sup>を線形に展開した設計・実装のプロセスである。この左半分は創造のサイクルのSTEP1~4のデザイン生成の過程で、右半分がSTEP5~7の実証から評価までの過程に對比できる。中央の「設計」は、設計内容の質を高め、収束を早期化するために、プロトタイプを積極的に活用している。デザインは、創造のサイクルという施主との合意の結果であることを意識して、早期段階で構造設計、設備設計の観点を取り入れるために、各専門家がより積極的に協働できるような体制を確保することが必要である。

#### 6.3 フィールドワーク

デザイン思考でもDDモデルでも、システム分析やプロダクトの適切なコンセプトを得るために、フィールドワークが重視されている。IDEO社では「師匠/弟子」と呼ばれるフィールド観察のプラクティスがある<sup>52)</sup>。日本では、KJ法というフィールドワークの手法が知られている。近年では、「質的アプローチ」と総称される観察対象の内観や、フィールドの観察を整理する手法が理論化されてきた。

企業情報システムにおいて、観察すべきフィールドはコンピュータを使って行われる作業の現場だけではない。人が関わるすべての業務の現場がフィールドである。製造業でいえば、製品の設計、製造、営業、受注、出荷、請求の

b 画面設計も含むが、中心となるのは、この情報システムで何をするのかの定義。

業務、仕入、原価管理、改善活動など、すべての領域に渡る。こうした情報システムの現場に適したフィールドワークの手法を確立する必要がある。

そして、情報システムにおいて何よりも重要なフィールドは経営者や管理者の意思決定の現場である。企業秘密が絡むので、その観察は困難であり、長期間の観察もできない。さらに、意思決定という目に見えない状況を観察し理解することも困難である。経営者に *thinking aloud*<sup>51)</sup> をお願いすることもはばかれる。経営者への短時間のインタビューは可能なので、凝縮した意思確認と信頼関係の構築に務める。情報システムの再構築を決断した経営者はイノベーションを期待している。インタビューから、オペレーション要求をしっかりと獲得して、その場で構想を共有する。

#### 6.4 コンセプト化とモデル

第4章で紹介したでサイン論は、どれもコンセプトワークを重視していた。

事例1のアーキテクトは、一つのコンセプトワードに凝集する手法を用いた。対象システム (system-of-interest)<sup>59)</sup> を一言で表現する手法は、たとえば経営トップとの面談におけるきっかけのプレゼンや合意の手法としての有効性を感じる。試みに、筆者の方法論のコンセプトワードで表現すると「企業が変わり続けるために」となる。

創造のプロセスでは、「コンセプト」の意味が上とは少し異なる。創造のプロセスのSTEP3では、技術要素も含めた複数のコンセプトとモデルを作って競わせている。情報システム全体についてコンセプトとモデルを作成することは実務上できないので、その中心となる業務についてのみ、複数のコンセプトとモデルを用意する。しかし、プロトタイプ作成に相当の時間とコストがかかるので、複数のコンセプトを評価したい場合は、業務シナリオ<sup>54)</sup>で代替する。

創造のプロセスで言うモデルは、モックアップのような見えるモデルである。情報システムにおけるモデルは、プロトタイプに当たる。

#### 6.5 アーキテクチャ

上述のデザイン論のうち、パタン・ランゲージを除いて、「アーキテクチャ」という言葉は出現しない。その理由は、建築デザインにおいてアーキテクチャの存在は議論を待たないからであろう<sup>55)</sup>。建築はアーキテクチャであるといっている文献<sup>56)</sup>もある。事例2のM氏は「人の行動があるからアーキテクチャがある」と言いきった。Alexanderは、「すべてのアーキテクチャは、人が活力を持って生きる状況を提供するためにある。アーキテクチャの中心的課題は、そうした仕組みや社会的状況を創造して、心地よさや深い満足のための励ましと支えを提供し、人が生きがいを感じられるようにすることである(拙訳)」<sup>64)</sup>と述べている。建築のアーキテクチャは、人の営みと密接に関わる。これは建物が量産品でないこと、基本構造が確立されていることなどに起因するのだろう。

近年、自動車やPCなどの「製品のアーキテクチャ」が議論されている<sup>57)</sup>。ここでいうアーキテクチャとは、製品を標準化された部品モジュールを組み立てるのか、製品ごとにすり合わせて部品から作るのかの“ものづくり”の方針を指す。これは、顧客との関係、工場設備、製造技術、会社の組織などを、時間をかけて構築し、時代に合わせて変化させてきた総合的なシステムである。量産型の製品は、顧客の買い換え行動に対応する製品の進化がアーキテクチャの形成を促してきた。

情報システムにおける「アーキテクチャ」は、「システムの基礎的な概念または特性で、システムの要素、要素間の関係であり、その環境において具現化され、設計と進化の原則となる」<sup>59)</sup>とされる。これには設備や製造技術が含まれておらず、工業製品ではないことを反映している。これを除けば、情報システムにおけるアーキテクチャは製品アーキテクチャに近いと言える。最近になって、利用者の行動cまでをデザインの対象とする「人間中心設計」<sup>60)</sup>などの提唱がされているが、それ以前に情報システムのアーキテクチャに関する決定的な議論と実践がなされていない現在では、情報表示のデザインにとどまらざるを得ないように感じる。

#### 6.6 デザイン

建築デザインにおいて、アーキテクトはさまざまな制約の下で、一定の設計思想に基づいて、施主の期待(用)を形(美)に変換することを知恵を絞る。プロダクトデザインにおいても同様だが、想定する顧客(マーケット)に与える価値(用美)であることと、想定するズレを修正するための設計の繰り返しが可能であること<sup>63)</sup>が異なる。

情報システムのデザインについては、これまであまり議論されて来なかった。80年代に「方法論」と呼ばれたものの多くは、設計成果物の記法や体裁を決めることしかできなかった。

現代の情報システムのデザインを考えるために、事例3を一般化して議論する。

##### 6.6.1 構造のデザイン

事例3では、生産管理の一般モデル<sup>50)</sup>から出発して、経営者との対話や業務要求を聞きながら、まず、その業種の特性に応じた製造領域の計画・実行相のドメインモデルを設計していった。これには、知識相の製造部品表(BOM)のドメインモデルの設計も含む。次いで、顧客注文のライフサイクルを管理するドメインのモデルを書いた。

さらに、製造実行ドメインのモデルを作成したところで、3つの異なる管理階層を構成することに気がついた。これらが、計画・実行相であるならば、それぞれに知識相<sup>9)</sup>があり、同様に報告相もあるだろう。こうした大きな枠組み

c IDEO社のいう動詞のデザインに対応する。

d そして、美を成り立たせるためには構造(強)が備わってなければならない<sup>55)</sup>。



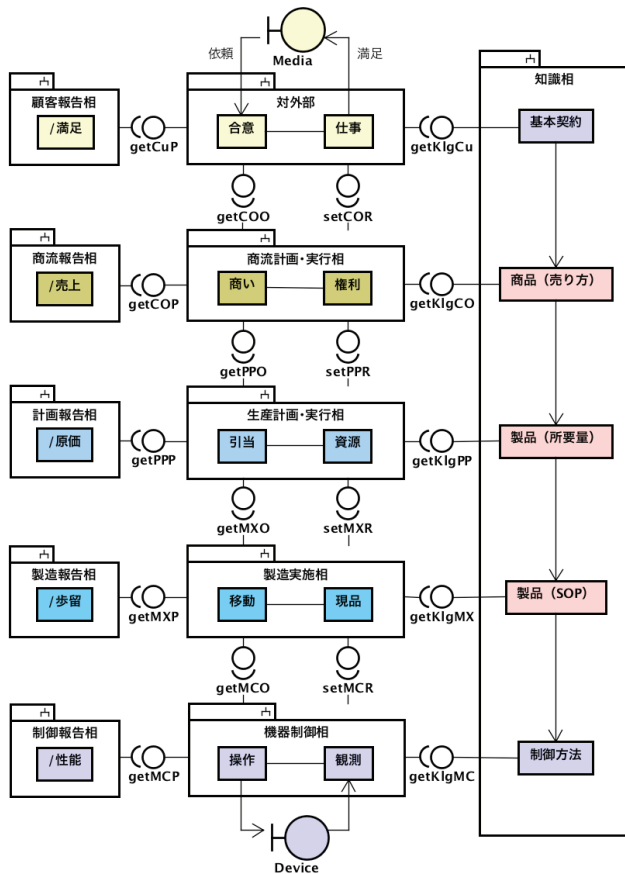


図9 論理アーキテクチャ図

が見えてきたところで、各ドメインの意味づけを調整し、概念モデルを調整した。

次いで、ドメイン間のインタフェースを設計した。当初は文献(61)にならって push 型の双方向のインタフェースとしていたが、(非公式の)構造設計レビューによって単方向のインタフェースに変更した。下層における処理のインスタンス数が上層よりも多いことに着目して、下から上への単方向参照が有利と考えたからである。

最終的に、図9のような論理アーキテクチャ図<sup>5)</sup>になった(設計成果物は、このほかに、各ドメインの概念モデル、インタフェース設計原則と基本設計書があるが、これは非公開とする)。これらの設計成果物は、のちに数十回にわたる小さな設計を重ねたが、大きな構造は変化しなかった。

この思考過程は、建築のアーキテクトがするような創造的なデザイン行為というよりも、分析と発見、仮説と検証のような理詰めのプロセスであったと思う。ただし、どこか美しくないと感じる図やモデルは、何かが間違っている可能性があると考えているので、気になった部分に対しては積極的に揺さぶりをかけて分析しなおすという態度は維持した。これが感性に依存する部分と言えるかもしれない。

これらの成果物を振り返ってみると、そこには論理的あるいは技術的な合理性があると感じる。たとえば、大規模なシステムを設計するには、適切なドメインへの分割戦略

が必要だし、計画・実行相が扱う時間は“今”であり、報告相は“過去”，知識相は“未来”である。管理階層は、最上層と最下層で実世界とつながっている、などは何度か書き直していく中で発見された後づけの理屈である(が、これは今後、再利用できる理屈となる)。しかし、技術的な合理性があるからといって、誰もがこの設計に辿り着けるわけではないこともわかる。

### 6.6.2 業務のデザイン

業務デザインは常に As-is の圧力に晒される。理想と現実の狭間に立たされるからである。真の施主による改革視点での設計方針の徹底を図り、デザインレビューによる設計方針への準拠性、決定された概念モデルとの整合の検査を行う必要がある。

しかし、経営トップの構想を実現するように業務のデザインを行うのは、業務担当者が行うことが多く、彼らにデザイン指向はない。実際、事例3ではアーキテクトレビューの必要性が認識されず、一部の業務と機能については再設計が必要となった。設計方針への準拠性やモデルとの整合を問う以上、業務担当者に対する一定の訓練とアーキテクトレビューは必要であった。

### 6.6.3 機能のデザイン

機能設計では、業務設計に基づき、業務ごとに必要となる機能をユースケース記述で定義する。自然言語による文書という形態を取るために、恣意性が混入することは避けられない。恣意性の混入は、アーキテクトの設計意図を歪めかねないので、これを避けるために、モデルの参照、用語の統制、形式の統制を義務づけ、設計レビューが必要である。

事例3では、あるベンダにユースケース記述の作業を依頼した。ベンダは業務担当者に“要件”を聞いて、ユースケースの形式で記述するという従来手法を用いたため、モデルの参照、用語の統一が不十分となり、再作成が必要となるものがあつた。上述のように、設計方針への準拠性やモデルとの整合を問う以上、作業者に対する一定の訓練と設計レビューは必要であった。

### 6.6.4 性能のデザイン

処理性能を上げるため、トランザクションの安全性を確保するためなどの技術的な工夫は、ソフトウェア製造段階での創造的な設計行為である。これも同様に、感性による問題検知と、理屈による解の生成過程が絡み合っている。

事例3では、C社のフレームワーク製品を導入した。このリード設計者は優秀で、アーキテクトとの連携も密接に行われた。フレームワークの問題の解決は、アーキテクトと相談しながら、C社が自律的に行うことができた。

### 6.6.5 意匠のデザイン

ユーザインタフェース、特に対話画面の設計は、設計者の感性が本質的に問われる創造的行為であると言える。しかし、画面設計からの分析を積み重ねても、基幹システム

の設計はできない。事例3のアーキテクチャでは、意匠設計は構造設計に従属させるために、ユーザインタフェースを意図的に切り離してある。現在の情報システムの設計では、構造が未熟なため、構造設計を優先せざるを得ない。

ユーザインタフェースは、HTMLで切り離すことができるため、当該システムが稼働した後に、総合的に改善していければよい<sup>60)</sup>。

#### 6.6.6 デザインのポイント

デザインのポイントは、デザインプロセスを決めることではない。Brooks<sup>62)</sup>はデザインプロセスを設定することは、デザインの自由度を制約することであり、レビューに拒否権を与えることであるという。レビュー組織は官僚機構化して、リスクを負わなくなる。それはデザイナーの創造性を奪うことになる。デザインをその人に任せよと叫ぶ。O氏は、インタビューの後のメールのやりとりで、デザインは個的なもの。公式レビューの前に、本人がチェックリストに基づいて自己レビューするしかないとのコメントをくれた。

情報システムのデザインプロセスは未熟であり、アーキテクトと各領域の設計者がどのように分業し、どのように協業するかは定まっていない。これからも、模索が続く。

## 7. おわりに

デザインが生まれ出る瞬間をとらえることはできなかったが、そこに近づくことはできたと思う。これを基に、デザインの過程の探求と研鑽を積み重ねていこうと思う。

Alexanderは、2012年に“The Battle for the Life and Beauty of the Earth—A Struggle Between Two World-System”という本<sup>64)</sup>を出版した。これは1985年の盈進学園東野高校の建築の場に何が起きていたかを、Alexanderの側から振り返ったものである。これをあえて否定的にとらえて、本稿のまとめに代えたい。

Alexanderは、この世界には二つの価値システムがあるという。System-Aは人々が生命にあふれ、美を愛で、有機的につながることを求めるシステム、System-Bは人々が時間や金で管理され、規制や機械に支配されることを求めるシステムである。盈進学園の建築でもこの戦いは起こった。あとき、Alexanderは教師や学生が自らの手で“学校”を作り続け、変わっていく世界をイメージしていたのだろう。しかし、開校の期限が迫っている、予算も残り少ない。結局、学校はSystem-Bを選んだ。それ以来、あの校舎は変わることができなくなってしまったという。これはAlexanderにとって都合のいい二項対立の構造である。しかし、多くのアーキテクトはこのような対立軸の上にはいない。これこそが、パタン・ランゲージが一般に使われない理由なのであろう。

企業は変わり続けなければならない。情報システムも、企業が変わり続けることを支えなければならない。これは

必ずしもAlexanderがいう共同設計を意味しない。むしろ、真の施主である経営者が設計に参画すべきだと思う。それが経営のハンドルを握るといことだろう。ここに情報システムのデザインの意味があると感じている。

**謝辞** インタビューに応じていただいた滋賀県立大学／アーキテクトシップの松岡拓公雄氏、アーキテクトの大牟礼真氏とその仲介の労をとっていただいたフジタの深澤道子氏、並びにデザインプロセスの議論に参加して下さった一級建築士の村山隆司氏に感謝します。

## 参考文献

- 1) 児玉公信, 「UMLモデリングの本質」, 日経BP (2004)
- 2) Evans, E., 著, 和智ほか訳, 「エリック・エヴァンスのドメイン駆動設計」, 翔泳社 (2011)
- 3) Hruby, P. 著, 溝口ほか訳, 「ビジネスパターンによるモデル駆動設計」, 日経BPソフトプレス (2007)
- 4) 児玉公信, 「情報システムサイクルと原要求の記述」, 日本情報経営学会誌, Vol. 28(2), pp.77-87 (2007)
- 5) 児玉公信, 「企業情報システムのための早期アーキテクティングの一方法」, 情報システムと社会環境研究会報告, IS120-3, 2012/6/4
- 6) Gamma, E., et al 著, 本位田ほか (監訳), 「デザインパターン (改訂版)」, ソフトバンクパブリッシング (1999)
- 7) Odell, J. J., “Advanced Object-Oriented Analysis and Design Using UML,” Cambridge SIGS (1998)
- 8) Hay D. C., “Data Model Patterns—Conventions of thought,” Dorset House Publ. (1996)
- 9) Fowler, M., 著, 児玉ほか訳, 「アナリシスパターン」, ピアソンエデュケーション (1998)
- 10) Buschmann, F., et al 著, 金澤ほか訳, 「ソフトウェアアーキテクチャ」, トッパン (1999)
- 11) Coplien J. O. 著, 和智訳, 組織パターン, 翔泳社 (2013)
- 12) 羽生田栄一監修, 「ソフトウェアパターン入門」, ソフト・リサーチ・センター (2005)
- 13) Fowler, M. 著, 長瀬監訳, 「エンタープライズアプリケーションアーキテクチャパターン」, 翔泳社 (2005)
- 14) Hillside Group, URL <http://hillside.net/>
- 15) Zachman, J. A.: “A Framework for information systems architecture,” IBM SYSTEMS JOURNAL, Vol. 26(3), pp.276-285 (1987)
- 16) Swell, M. T. ほか著, 倉骨訳, 「職業としてのソフトウェアアーキテクト」, ピアソンエデュケーション (2002)
- 17) 南波幸雄, 「企業情報システムアーキテクチャ」, 翔泳社 (2009)
- 18) Simon, H. A., “The Sciences of the Artificial,” 稲葉ほか訳, 「システムの科学 第3版」, パーソナルメディア (1999)
- 19) 日経コンピュータ (編), 「動かないコンピュータ—情報システムに見る失敗の研究」, 日経BP社 (2002)
- 20) Carr, N. G. 著, 清川訳, 「ITにお金を使うのは、もうおやめなさい」, ランダムハウス講談社 (2005)
- 21) 日経コンピュータ (編), 「システム障害はなぜ二度起きたか—みずほ12年の教訓」, 日経BP社 (2011)
- 22) Alexander, C. 著, 宮本訳, 「オレゴン大学の実験」, 鹿島出版 (1977)
- 23) Alexander, C. et al 著, 平田訳, 「パタン・ランゲージ」, 鹿島出版 (1984)
- 24) Alexander, C. 著, 平田訳, 「時を越えた建築の道」, 鹿島出版 (1993)
- 25) Alexander, C. et al 著, 中埜監訳, 「パタンランゲージによる住宅の建設」, 鹿島出版 (1991)

- 26) Alexander, C.著, 難波訳, 「まちづくりの新しい理論」, 鹿島出版 (1989)
- 27) 環境構造センター, 「盈進学園のボタン・ランゲージ」, 新建築, 新建築社, 1985年, 第6号, 182 (1985)
- 28) 川越市, 「蔵造りのまちづくり」, in 国土交通省都市・地域整備局, 「まち再生事例データベース」,  
[http://www.mlit.go.jp/crd/city/mint/html\\_doc/table.html](http://www.mlit.go.jp/crd/city/mint/html_doc/table.html)
- 29) 真鶴町, 「美の基準—デザインコード」, 真鶴町まちづくり条例 (1993)
- 30) 中埜 博, 「ボタン・ランゲージによる住まいづくり」, 井上書院 (1988)
- 31) 五十嵐敬喜, 「美の条例—いきづく町をつくる」, 学芸出版社 (1996)
- 32) 秋田典子, 「真鶴町まちづくり条例の『美の基準』の実効性に関する研究」, 日本建築学会大会講演梗概集, 431-432 (2002)
- 33) 五十嵐敬喜, 「美しい都市をつくる権利」, 学芸出版社 (2002)
- 34) Halprin, L., 伊藤訳, 「都市環境の演出—装置とテクスチュア」, 彰国社 (1970)
- 35) Vitruvius 著, 森田慶一訳: ウィトルーウィウス建築書, 東海大学出版会 (1979)
- 36) 「盈進学園東野高校」, 新建築, 新建築社, 1985年, 第6号, 163-181 (1985)
- 37) 福島ちあき, 「ボタン・ランゲージの実践と理論の比較分析: 盈進学園東野高等学校の分析を通して」, 大阪市立大学大学院修士論文 (2005)
- 38) 藤村龍至, 「プロトタイピング—模型とつづやき」, LIXIL 出版 (2014)
- 39) 守口守, 「入門 都市計画」, 森北出版 (2014)
- 40) 箕原 敬ほか, 「これからの日本に都市計画は必要ですか」, 学芸出版社 (2014)
- 41) 児玉公信, 水野忠則, 「情報システム学的ボタンランゲージの再発見」, 情報処理学会ソフトウェア工学研究会, 2007/5/28
- 42) K. Kodama and T. Mizuno, “Rediscovery of Pattern Language from the Information Systems Viewpoint,” The 51st Annual Meeting of the ISSS, International Society for the Systems Sciences, 2007/8/9
- 43) Peña, W. M. and Parshall, S. A.著, 溝上訳「プロブレム・シーキング—建築課題の発見・実践手法」, 彰国社 (2003)
- 44) Cherry, E.著, 上利訳, 「建築プログラミング」, 彰国社 (2003)
- 45) Bason, C. (Ed.), “Design for Policy,” Gower Pub. (2014)
- 46) Sauder School of Business, “Design Processes,” The University of British Columbia, <http://dstudio.ubc.ca/toolkit/processes/> (2015.1.11 accessed)
- 47) “Introducing Design Methods,” Design Council Blog, <http://www.designcouncil.org.uk/news-opinion/introducing-design-methods> (2015.1.11 accessed)
- 48) Gardiner, E., “Are behavioural science and design the building blocks of innovation?” Design Council, <http://www.designcouncil.org.uk/news-opinion/are-behavioural-science-and-design-building-blocks-innovation> (2015.1.11 accessed)
- 49) 米盛裕二, 「アブダクション—仮説と発見の論理」, 勁草書房 (2007)
- 50) 児玉公信, 水野忠則, 「少量多品種生産管理システムの一般モデル CHARM の提案」, 情報処理学会論文誌, Vol. 49(2), 902-909 (2008)
- 51) Ericsson, K., & Simon, H., “Verbal reports as data.” Psychological Review 87 (3), 215-251 (1980)
- 52) 奥出直人, 「デザイン思考の工具箱」, ハヤカワ文庫 NF398, 早川書房 (2013)
- 53) DeMarco, T., “Controlling Software Projects,” Yourdon Press (1982)
- 54) IEEE Std 1362:1998 “Guide for Information Technology System Definition -- Concept of Operation Document.”
- 55) O’Gorman, J. F., “ABC of Architecture,” PENN (1998)
- 56) Unwin, S.著, 重枝ほか訳, 「建築デザインの戦略と手法」, 彰国社 (2005)
- 57) 藤本隆宏ほか, 「ビジネス・アーキテクチャー—製品・組織・プロセスの戦略的設計」, 有斐閣 (2001)
- 58) 田子 学ほか, 「デザインマネジメント」, 日経 BP (2014)
- 59) ISO/IEC/IEEE 42010:2011, Systems and software engineering—Architecture description.
- 60) 長谷川敦士, 「IA100—ユーザーエクスペリエンスデザインのための情報アーキテクチャ設計」, ビー・エヌ・エヌ新社 (2009)
- 61) 高橋達也ほか, 「標準技術の相互活用による工場内情報連携 (MESX プロトコルによる製販一体化)」, 日本機械学会論文集 (C編), Vol.76, No.772, pp.58-63 (2011)
- 62) Brooks, Jr. F. P.著, 松田ほか訳, 「デザインのためのデザイン」, ピアソン (2010)
- 63) 山中俊治, 「デザインの骨格」, 日経 BP 社 (2011)
- 64) Alexander, C., et al, “The Battle for the Life and Beauty of the Earth—A Struggle Between Two World-System,” Oxford Univ. Press (2012)