

新ソフトウェア宣言

Declaration of New Software

従来のソフトウェアに関わる諸活動の呪縛から解放され、新たな世界を築いていくためには、新たな呪縛に捕われ、新たな中心を設定していかなくてはなりません。我々は、以下の関心の候補が有望であると確信しています。

In order to build a new world, freed from the curse of activities related to traditional software field, we shall be caught in a new curse, it does go to a new paradigm will set. We are strongly confident that the following are promising candidates for the matter of concern.

1. ソフトウェアは、数学的理論探求の上に成り立つ

Software is built upon exploration of mathematical theory.

2. ソフトウェアは、部分に還元することが不可能な全体である

Software can not be reduced to parts.

3. ソフトウェアは、実行可能な知識である

Software is executable knowledge.

4. ソフトウェアは、学びの副産物に過ぎない

Software is just a by-product of learning.

5. ソフトウェアは、制約条件下で創造される美しい人工物である

Software is a beautiful artifact, created under some constraints.

6. ソフトウェアは、富を生む経済活動の資源である

Software is a resource of wealth-creating economic activities.

7. ソフトウェアは、言語ゲームである

Software is a language game.

上記、基盤化、全体論、知働化、進化論、意匠論、経営論、遊戯論などの広範で、かつ、相互補完的アプローチが、新しい物語を生み出し、新たな地平を切り開いていくことでしょう。

Above, the mathematical infrastructure, Holism, executable knowledge / texture, evolution, design theory, management theory, and language game, those are broad and the complementary approaches, creating a new story, we will open up a new horizon.

大槻繁, 芝元俊久, 高野明彦, 竹内雅則, 時本永吉, 夏目和幸, 萩原正義,
羽生田栄一, 濱勝巳, 本橋正成, 山田正樹, 綿引琢磨

Shigeru Otsuki, Shibamoto Toshihisa, Akihiko Takano, Masanori Takeuchi, Eikichi Tokimoto, Kazuyuki Natsume, Masayoshi Hagiwara, Eiichi Hanyuda, Katsumi Hama, Masanari Motohashi, Masaki Yamada, Takuma Watabiki

新ソフトウェア宣言【概説】

Declaration of New Software <An Overview>

『新ソフトウェア宣言』は、通称『呪縛宣言』とも呼ばれています。2010年6月9日～11日に横浜開港記念館で開催されたソフトウェアシンポジウム2010の中のワーキンググループ「ソフトウェアエンジニアリングの呪縛WG」に集まった賢人たちが、これからのソフトウェア、および、ソフトウェアエンジニアリングの方向性について議論したものが元になっています。最終的には、「呪縛WG」の二人のコーディネータである大槻繁と濱勝巳氏によって、WG終了後議論が重ねられ、今の7項目からなる形に集約されました。

ここで言う「呪縛」とは、何かいわれのない思想や言葉などによって、行動や考えが制限されてしまうことを示しています。ソフトウェアの世界は、多くの思い込みや、論拠のない教えに満ちていて、研究、技術開発、ビジネスなどのあらゆる領域で閉塞感や限界を感じざるを得ません。次世代のソフトウェアの取り組みでは、まずこの呪縛から解放されることが必要です。ここで言う「呪縛」は、天動説から地動説へというコペルニクスの転回を説明する科学哲学の概念である「パラダイム（規範）」と言い換えてもよいでしょう。パラダイム論では、科学技術の進化が、不連続な変化であるとみなされます。パラダイムシフトを果たした後の世界は、過去とは異なるパラダイムです。つまり、新しいパラダイムは、古いパラダイムから解放されていますが、それは新しい呪縛でもあります。このことを、『新ソフトウェア宣言』の序文で「従来のソフトウェアに関わる諸活動の呪縛から解放され、新たな世界を築いていくためには、新たな呪縛に捕われ、新たな中心を設定していかなくてはなりません。」と、凝縮した形で述べています。

『新ソフトウェア宣言』は、7つの文から構成されています。それぞれの文は、それぞれ深い意味を持っています。また、相互に関連もしています。さらに、記述されている順序にも意味があります。新しい世界観を示すために言葉を厳選し、無駄を省き、シンプルで判りやすい宣言になっています。我々が学問や技術領域を語る場合には、一つの「関心」を持っていると言えるでしょう。それぞれの文は、新しいパラダイムを構成する「関心」の候補を掲げています。関心は、旧来の学問や技術領域にゆりく対応していることもあれば、複合的に対応している場合、さらには、全く新しい領域が示唆されている場合もあります。

1. ソフトウェアは、数学的理論探求の上に成り立つ

プログラムの動作原理は、アラン・チューリング、フォン・ノイマンといった偉人によって確立された計算理論に基づいています。関係代数、ラムダ計算、符号理論などのコンピュータサイエンスによって、エンジニアリングは支えられて

います。これらは、数学的理論といってもよいでしょう。

エンジニアリングとサイエンスというのは、車の両輪です。お互いに切磋琢磨して進歩していくものでしょう。一般的には、ソフトウェアをつくるということは、抽象機械を構築することを意味しています。人間が自由に発想し、新しい概念を生み出し、定式化し、計算を実行する機械として実現します。

クラウドコンピューティングや、進化型のソフトウェア、さらには、人間の認識や脳のモジュール化などを扱っていくための、数学的なアプローチ、科学的な基盤はまだまだ不足しています。ですから、「探求」という言葉で数学的理論整備の取組みも同時並行的に進めていかななくてはならないことを宣言しています。

2. ソフトウェアは、部分に還元することが不可能な全体である

大きなソフトウェア、大きな問題に対処する有効な方法が「分割統治 (divide and conquer)」です。モジュール化、段階的詳細化といった方法があり、これらが有効な場面も多いのは確かですが、単純な理論や構造を拠り所としているため、現実世界の問題では限界があります。これは裏を返せば、全体が部分から構成されることを前提とした対象に絞って、できることしかやらないと言っているようなものです。

こういった制約された良構造を探求すること自身は悪いことではありませんが、少なくとも限界を知る必要があります。科学の世界でも、旧来のデカルト的な世界観や還元主義、機能主義、決定論からの脱却の必要性が広く唱えられてきています。

近年台頭して来ている複雑系、カオス、進化論などの研究領域では、全体は部分の総和以上の何かがあるとする立場をとっています。全体と部分とが循環しているといってもよいでしょう。企業やチーム、さらには、ソフトウェアそのものを生命体と見なすことは、自然な発想です。複雑系の研究領域では、ニュートン以来の決定論的な考え方脱却し、組織化された生物学的なアプローチをとっています。

3. ソフトウェアは、実行可能な知識である

ソフトウェアに対する新しい見方をしようという活動である「知働化研究会」の主要コンセプトが「実行可能な知識」です。従来のソフトウェアづくりでは、プログラムコードが中心で、それを工業的、組織的に開発するための文書があり、正しさを検証するための膨大なテスト、プロジェクトマネジメント手法による精密な段取りを要していました。出来上がったプログラムは、その稼働環境、社会の中での役割や位置づけが比較的安定していました。

こういった状況を打開し、ソフトウェアとは何かを見つめ直し、社会の中でダイナミックに変化していくソフトウェアやそれへの関わり方を、探求していく必

要があります。

我々は、知識主導社会に生きており、「知識」はその中心です。ソフトウェアに関わる活動は、知識活動です。それは、作り方/使い方の知識であり、それを糸や布のように紡いでいくことであり、実世界の知識を実行可能な知識に埋め込み/変換していくことであり、知識の贈与と交換とが行われる世界でもあります。

4. ソフトウェアは、学びの副産物に過ぎない

ソフトウェアは、人が作り、人が使うものです。人、組織、社会が関わっているということは、そこには、認識の進展、知識の獲得、ノウハウの蓄積があり、これらが実は中心であり、ソフトウェアは知的活動の副産物に過ぎないというのが、この文の主張です。

常に学び続けることが人の本性であり、常に組織知能を高めていくことが組織経営の基本です。経営手法でよく使われるバランス・スコア・カードでも、主要な観点の一つに「学習と成長」を掲げています。収益を上げることを財務的な目標としつつも、顧客の観点、内部プロセスの観点との関連づけが必要ですし、最終的には組織力というのは、人材や組織の学習と成長無しには向上させることができません。

ソフトウェアの実行は、実世界に影響を及ぼします。ソフトウェアも変化していくでしょうし、それに関わる人の認識や、システムやビジネス環境も動的に変わっていきます。このダイナミズムを中心にすえて、「学ぶ」ということの本質を明らかにしていくことが、実は次世代のソフトウェアづくりを考える近道なのかもしれません。

5. ソフトウェアは、制約条件下で創造される美しい人工物である

ソフトウェアは、人間が創造する「人工物 (artifact)」であることは間違いありません。概念的で目に見えないこと、最終的にはコンピュータによる実行を伴うことが特徴です。

クラウド・クリッペンドルフは、著作『意味論的転回』の中で人工物のデザイン論を提唱しています。「科学」というのは、過去に起こったことを分析して何か法則を見いだしたり証明したりするのに対して、「デザイン」というのは将来について意思決定していく行為であり、「人間中心」の「意味」を扱う体系でなくてはならないと主張しています。この視点は、サイエンスとエンジニアリング、そして、デザインとの関係を新しい世界観で構成していくことを意味しています。

一方、フレデリック・ブルックスの著作”The Design of Design”では、創造的活動としてのソフトウェアのデザイン手法について、随想的に本質を述べています。全体を見ること、ユーザの不確実性、制約と美、チームによる創造活動、専門家集団のコラボレーションなどについて述べています。特に、「制約と美」は、

関係が深く、自由度が高過ぎると美しい人工物は生まれにくいようです。メモリ制約、計算量の限界、予算やさまざまなリソースに制約条件が、想像力をかきたて、真に創造的な成果を生み出す原動力になることが多いと言えるでしょう。

『新ソフトウェア宣言』の中で、この文は、言葉遣いの中に修飾語も多く、「美」という主観的な価値観を連想させる言葉をいれることによって、メッセージ性を高めようとしています。

6. ソフトウェアは、富を生む経済活動の資源である

ピーター・ドラッカーは、晩年の著作『ネクスト・ソサイエティ』で、知識主導社会へのシフトを唱えました。経営論の主題は、組織と組織、人と人との間の関係性です。「経営」という概念そのものもドラッカーが造り出した概念です。経済活動や社会活動の目的は、富を創出すること、適切な配分を行うこと、協調して生き延びる仕組みを造ることです。

人、金、もの、情報が経営資源と言われていますが、ソフトウェアそのものも「資源」とみなすことができます。工業的パラダイムでは、手順化して役割を定義し、人を割当てるという属人性を排除したプロセスを構築することが、経営の一つの目標になっていました。知識主導社会では、創造的活動での人の役割や、役割分担の方法、そして、これ等の経営の方法も新しいものが必要になってくるでしょう。また、社会の制度も、アルビン・トフラーの言うような地産・地消、富の分配の方法などが変わってくるものと思われれます。

この文では、「幸福」という言葉を使っていないところに明確な意思があります。どうしても「幸福論」という見方ですと、不条理の中で幸せと思えるような考えが入りこんでしまうので、これを排除しています。あくまでも「富」なのです。「経済活動」というのも、「社会/経済活動」とすべきかもしれませんが、あえて、「経済活動」と限定してみました。無論、金銭や金融の問題のみを対象としているわけではありません。

7. ソフトウェアは、言語ゲームである

『新ソフトウェア宣言』を一言に集約するならば、「ソフトウェアエンジニアリングの世界で言語ゲーム的転回を進めよう」ということです。

ヴィトゲンシュタインは近代分析哲学の基礎付けを行った哲学者です。大きく前期と後期に成果が大別され、前期と後期では全く異なった世界観を構成しています。前期の成果は『論考』（論理的・哲学的論考, 1921）として出版され、後期哲学は『探求』（哲学的探求, 1949）としてまとめられました。

『論考』では、言語による記述（命題）の意味を、世界の事態の成立として定義しています。このことは、世界が言語の記述とは独立に（たぶん神が創造して）在り、それを語るために言語による記述があるという考え方です。世界の事実と

は関係のない、信念や形而上学的な事項については、語ってはならない、つまり無意味だという主張です。いわゆる、命題論理の基礎付けを与えている哲学といえるでしょう。この「意味の対象説」は、20世紀初頭の論理実証主義を支える哲学になりました。

ヴィトゲンシュタインは『論考』の世界観を捨て、後期哲学では「意味の使用説」として「言語ゲーム」という考え方を提唱しました。「物の世界」でも「事の世界」でもない、全ては「言語的事象の世界」であるという主張です。この「言語の使用」が即ち、「言語の意味」になっているという、後の言語行為論や状況意味論の基礎付けを行っています。

ソフトウェアエンジニアリングというのが、プログラムコードの記述に限らず、あらゆる活動を<言語活動>や<言語現象>であるとみなすという立場は、とても有望です。無論、無意識や心理的な事項もありますが、最終的には言語として現れることとなります。こういった『探求』とか『言語ゲーム』の哲学に基づいて、新しいソフトウェアの世界を構築していくことは、新たな地平を見るために必要なアプローチであると確信しています。

以上で、7つの文の概要を説明しました。もう一度、列挙してみることにしましょう。

1. ソフトウェアは、数学的理論探求の上に成り立つ
2. ソフトウェアは、部分に還元することが不可能な全体である
3. ソフトウェアは、実行可能な知識である
4. ソフトウェアは、学びの副産物に過ぎない
5. ソフトウェアは、制約条件下で創造される美しい人工物である
6. ソフトウェアは、富を生む経済活動の資源である
7. ソフトウェアは、言語ゲームである

なんとなく、ふわっと新しい世界が見えてくるとは思いませんか。一つ一つの文は独立ではありません。この順序にゆっくりと噛み締めながら読んでみると、学問、技術、デザイン、ビジネス、人、世界、認識など、全体として何か新しいものが見えてくると思います。7つをもっと強引に「**論」や「**化」として集約した記述が、『新ソフトウェア宣言』の最後にある「基盤化、全体論、知働化、進化論、意匠論、経営論、遊戯論」です。この宣言全体自身が部分に還元できませんし、語ってしまうとそれではなくなってしまうようなものです。

今回の宣言は、2010年の夏のスナップショットです。とりあえず果たし得ぬ夢の方向性は示すことができたと思います。この文書のような解説も、人それぞれあると思います。一つの主観的な解釈として参考にさせていただければ幸甚です。

大槻 繁

2010年7月15日 新横浜にて