

さまよえるモデリング ゆるいソフトウェアを求めて

発表：2009年6月18日 / 資料：2009年7月17日

本橋 正成 <masanari@masanari.com>

8

はじめに：究極のモデルを求めて

「未来を予測する最善の方法は、それを発明することだ」

みなさんがご存知のアラン・ケイの言葉です。モデリングにおいて、どのような未来を作り出していくのかを探求者としての命題としてあげました。たとえ、今日時点ではファンタジーであったとしても、もし想像ができれば実現に向けてのきっかけになります。果たして、究極なモデルはどのようなものでしょうか。ぜひ、みなさまの意見をうかがいたいと思っています。



今回の発表では、若干のユーモアも混ぜ、究極のモデルとして「阿吽（あ・うん）」を例示しました。Wikipedia(2009年7月現在)の「阿吽」によると「悉曇文字（梵字）」において、阿は口を開いて最初に出す音、吽は口を閉じて出す最後の音であり、そこから、それぞれ宇宙の始まりと終わりを表す言葉」となっています。つまり、たった二文字で宇宙の始まりと終わりが記述されています。ただ、この「阿吽のモデル」は究極ですが、ちょっと規模が大きすぎて実生活やお仕事で使いにくいようです。ふたりがぴったり息を合わせて、やり取りしている「阿吽の呼吸」が、もうひとつ連想されます。今回は、この阿吽の呼吸を目指したソフトウェアやモデルを探求しました。

ご縁モデル：その1

モデリング・ワーキング・グループ・リーダの山田氏が、顔合わせのときに「ご縁」とおっしゃったのをヒントにして、ご縁モデルについて考察を試みました。

まずは、とても簡単な物理現象を取り上げてみましょう。発表時、手元にあった筆入れを落としてみます。



この筆入れの落下にかかる時間は、ニュートン力学の自由落下運動の公式（りんごですね！）を使って簡単に計算できます。ワーキング・グループが開催されていた10階会議室の窓から地上まで筆入れを落とした場合の落下時間は、風の抵抗も無視できなくなるでしょうから、この筆入れの投影面積を求め空気抵抗を入れて計算できそうです。しかしながら、実際に落としてみると、上昇気流のようなビル風の発生や、投げ方によってはくるくる回転（スピン）や障害物やビルへの衝突など状況が変化しますので、厳密な筆入れの落下時間の厳密な計算はおそらく難しくなるに違いありません。さらに、落下しているのが筆入れではなく、ひらひらしているA4の紙の場合はどうなるのでしょうか。もちろん計算できそうですが、さまざまな仮定をおく必要があります。つまり、実際の世界では単純な落下といえども、なかなか厳密な計算は難しそうです。簡単な物理現象で既知だと考えられていても厳密な計算は難しくなっていく例として、手元にあった筆入れの落下をあげてみました。もし、計算の難しさがまだまだ足りない場合には適切な問題が他にもたくさんあるかと思います。

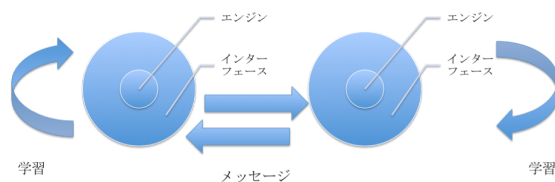
さて、筆入れの落下にかかる時間を予想する別の方法も検討できます。実際に落として落下時間を計測し、その結果を予想に使

うことです。筆入れを 10 回落とせば、おそらく 11 回目に落ちる時間も予想できそうです。筆入れを 100 万回も落としたら、100 万 1 回目の落下時間をよりの確に予測できそうです（けど、象に踏まれても壊れないぐらい丈夫な筆箱を使う必要がありますね）。100 万回の落下実験を行うと、そのつど、壊れ具合や状況などが変わってきますので、風の強さや筆入れの状態などの状況も記録しておくことによって、その結果、似たような状況での落下時間のある程度の精度で確率的に予想できそうです。

例の取り上げ方が少々乱暴ではありましたが、世界第二次大戦時代、ノーバート・ウィナーが自由に飛び回る戦闘機に打ち落とすため、軍艦から打つ砲丸の制御する必要性からまとめたサイバネティックスがこのような解決方法のヒントになっています。サイバネティックスは、出力した結果（の一部）をフィードバックされた入力として扱います。もちろん、サイバネティックスを使えば、シミュレーションの計算が不要になるわけでは決してありません。お互いに補完的な関係や、とらえかたによっては階層的な関係を構築できます。ただ、この計算結果を実際の世界に出力するだけでなく、実際の（実験）結果をフィードバックの入力として取り込む系を構築するところがポイントです。つまり、実社会の現象とのメッセージのやり取りがとても重要になってきます。

筆入れの落下時間を予測する際、実際の計測結果をフィードバックとして入力し、蓄積します。次の落下時間を予測するときは、その蓄積されたデータを参考にします。落下時間の実績を蓄積していくことを「学習」と表現しました。このシステムの構成は、受け取ったものを認識し、知識として取り込む学習部分と、知識から適切なものを出力する部分がエンジンとして構築されます。さらに、蓄えられた知識部分も構成されると考えています。

一般化して、自システムから出たメッセージを、他システムへ送り、他システムにおいても相互に学習するモデルにしました。



なお、発表では、コミュニケーションやメッセージなどの「関連」の重要性を訴えるために、ユーモアをこめて存在（いいやつ）は中身とコミュニケーションの二乗である $e=mc^2$ というエイプリル・フール・ネタも合わせて示しました。

ご縁モデル：その2

現実には多システム間での連携もしています。ここでのシステムとは「相互に影響を及ぼしあう要素から構成される、まとまりや仕組みの全体。系。(Wikipedia より)」を示しているため、必然的に多システムの連携は再帰的・階層的なシステムやパターンになります。

多システムを、たとえばベイジアン・ネットワークやマルコフ連鎖を使用して、不確実で複雑な構成を（主観）確率の連鎖で記述できそうです。ベイジアン・ネットワークでは、それぞれのノードの確率の相互作用から、ある事象における「もっともらしさ」を求められます。

ベイジアン・ネットワークやフィードバックの利用は、系の方向性などから階層構造や再帰構造など別のレイヤーで構成する必要があるかもしれません。

ゆるいソフトウェアに向けて

今まで、ビジネスやソフトウェアの典型的な進め方は、できる限り形式化し、それを計画に沿って厳密に構築、実行していく方法が多かったように感じます。しかしながら、現実の世界では相互作用や刻々と変わる流動的な現象であり、事前に計画することは困難です。計画が実行される時に状況はすでに変化しているため、事前にビジネスやソフトウェアを計画すること自体がも

ろさに結びつきやすいのではないかと考えています。

「強い」ビジネスやソフトウェアには、刻々と変わる現象を柔軟に扱う必要があります。また、計画時点では ぼんやりしているもの（暗黙知 **tacit knowing**）もできる範囲で取り込んでいきたいです。今回はご縁と表現しましたが、相互作用を行い、いかにシステムを育てていくのか、という枠組み作りの視点がより重要になります。そのための解や方向性のひとつとして「ゆるさ」の観点の重要性を提案します。「ゆるい」は多義的ですので、大槻氏にまとめていただきました。

ゆるい#	定義名称	説明	典型的対象分野
ゆるい1	抽象的	設計自由度	全般
ゆるい2	不確実	確率・非決定的	予測、フィルタ
ゆるい3	曖昧	未認識・未定義	全般
ゆるい4	相対的	言語ゲーム的	社会、対話
ゆるい5	進化的	メタ・状況適応	パターン認識
ゆるい6	計算量限界	近似解	レイアウト、検索
ゆるい7	未解明項	調整・フィードバック	制御、ログ解析

事前に厳密に文書化された計画には限界があります。いろいろなモノやコトは厳密な認識は困難であるため、ゆるい、という視点を持つことによって、変化する状況に対し、より柔軟で的確に進むことを期待しています。ゆるさの視点を適用することにより、目の前の要件や要求を直接的にアプローチするだけでなく、より本質的で柔軟な間接的なアプローチを取れるのではないかと目論んでいます。その結果、価値が高く「強い」ソフトウェアやビジネスの構築に貢献できると考えています。

ご縁モデルの例として紹介したサイバネティックスやベイジアン・ネットワークは、かなり古くから研究されており、いくつかのエンジンもあります。また、すでに「ゆるさ」を必要としている分野や部品があり、すでに実装され、実際に利用されています。発表では、某所で開発中の製品の例から説明も行いました。しかし、開発中ということもあり、この資料では割愛させていただきます。

現実の社会には、実際のビジネスやソフト

ウェアの世界に「ゆるさ」がフィットする分野や問題があり、そこには「ゆるいソフトウェア」を導入していきたいと考えています。学習するご縁モデルなどによって、状況や相手に応じて「ほしいところに手が届く」、さらには「阿吽の呼吸」の状態に近づいていくと期待しています。ただ、阿吽の呼吸であっても、たまに外します。そこは確率ですのでご容赦くださいませ♪

知働化に向けて

今までは、ウォーターフォール型開発プロセスに象徴されるように計画段階で知識を形式化し、それを構築することが主でした。今後は、偏在する知識や知恵と、それをダイナミックにつむぎだすことによって、より強いシステム（系）やビジネスの構築を目指します。トヨタ生産方式にもあるカイゼン（改善）の知恵を機械やシステムに織り込み、持続的に行うためにも人が関わる「自働化(にんべんの働)」の考え方を、より明確にして推進できる知働化が重要であると考えています。

天動説から地動説へのパラダイム変化のように、今後はソフトウェアやビジネスのパラダイムも変わっていくと考えています。

おわりに

今後は、実際にゆるさや知働を組み込むチャレンジを行い、経験を共有させていただけたら幸いです。その中で、ゆるさ自体のパターンを抽出するとともに、ビジネスやソフトウェアを構築するために、ゆるさや知働をマッチするパターン（と、アンチ・パターン）を抽出し、知見としてまとめたいと考えています。

ゆるい発表と文章ではございましたが、皆様とのご縁のおかげと強く感じております。

どうもありがとうございました。