

ST/SD 入門セミナー・ワークショップ

(シリーズ1)

2014年3月~8月実施

目 次

1. 目的と構成	1
2. 方法論	2
3. ストックとフロー、システムの基本的な振る舞い	18
4. モデル構造の基本形	28
5. システム原型	35
6. モデル構築技法	65

日本未来研究センター
研究員
末武 透

1. 目的と構成

(1) セミナーの目的と趣旨

本稿は、日本未来研究センター及びシステム・ダイナミクス学会日本支部との共催で2014年3月~8月に実施した、SD入門コース(シリーズ1)の説明内容をまとめたものです。

本セミナー・ワークショップは、ST/SDの紹介、入門、そして、ST/SDに関心のある人が集まって、情報交換を行ったり、教え合ったりできる場として実施することにしました。ただ、このコースで使うSDのソフトは、ithink/Stella、Powersim、SimTaKN、Anylogなどいくつかある中で、Veism PLEに限定しています。

2つのコースを実施し、そのうちのこの「SD入門コース」は、システム・ダイナミクスのモデリング技法習得が目的で、講義中心の内容です。習得目標レベルとして：

- 簡単な問題をシステムとして把握できる
 - 簡単な定性モデルを作成して定性分析できる
 - 簡単な定量モデルを作成し、シミュレーションできる
- を目指していました。

(2) セミナーで使ったテキスト

テキストは、末武著、「Vensim PLEによるシステム思考及びシステム・ダイナミクス入門」、日本未来研究センター、2009を使用しました。

(3) 構成

第2章(第1回目実施)では、SDコースの概要も兼ねて、SDモデルを構築していく方法論について説明しています。その中で、いきなり定量モデルを構築するのではなく、モデルの目的やカバーする範囲、性格などをまず明確にし、それから定性モデル構築と定性分析、その後、定量モデル構築と定量分析に進むことで、手戻りが少なくなること、定量モデル構築がうまく行かない原因の一端には、前工程での検討の甘さがあることを指摘しました。定性モデル構築で、手戻りはある程度は避けられないことですが、あまり曖昧なままに作業を進めるのでは、結局、目的に合ったモデル構築がいつまでたってもできないこととなります。

第3章(第2回目実施)では、ストックとフロー、SDモデルの要素の振る舞いを説明しています。

第4章(第2回目実施)では、モデル構造の基本形ということで、2つのストックを含むモデル、そしてパイプライン構造のモデルについて説明しています。

第5章(第3回目及び第4回目実施)では、モデルの振る舞いの基本形、そして、それを表現した例として、システム原型とその定量モデルを紹介しています。

最後の第6章(第5回目実施)では、モデル構築におけるヒントや注意事項を記載しています。

なお、テキストでは、この後、定性分析、定量分析という2つの章があり、事例を通して、定性分析、定量分析を行うやり方を紹介していますが、セミナーでは、第2章の方法論の説明で事例に即し、定性分析、定量分析のやり方を紹介していますので、説明は行いませんでした。

本稿は、日本未来研究センターの研究員である末武透がまとめたものですが、本稿で述べた見解や意見等に関し、私が全面的に責任を負うものであり、日本未来研究センターやシステムダイナミクス学会日本支部の見解を表したものではないことを述べておきます。

2. 方法論

ここでは、

- ・SD とは何かの定義
- ・SD モデル開発方法論
- ・問題とは

を取り上げます。

(1) SD とは何かの定義

世の中では、なんとなく、「システム・ダイナミックス」とは、定量モデルと定量モデルを用いて行うシミュレーションを基に定量分析を行う方法と了解されているように思います。これに対して、「システム思考」というのが存在し、こちらは、システム類型などを用いて定性分析を行う方法と了解されているのではないかと思います。ただ、私個人はこの理解は間違っていると思っています。少なくとも、SD 関係者は、定性モデルも定量モデルも SD のモデルであると考えているし、SD モデルのコンセプトを説明するために ST モデルと呼ばれる定性モデルをよく使います。SD 関係者は、「システム思考」は、対象をシステムとして見る見方のことと考えています。注 2-1)

ここでは、定性モデルによって定性分析を行い、次いで、定量分析を行う必要があれば、定性モデルを基に、定量モデルを構築して、定量分析を行う一連のやり方を「システム・ダイナミックス」と定義し、話を進めたいと思います。また、「システム思考」と呼ばれているものは、システム・ダイナミックスの中の定性分析の部分の部分を指すとしします。後で、「システム思考」のコースになった時にこの定義は変わるかも知れません。

ここで、「モデル」という言葉が出てきましたので、モデルについても定義しておきましょう。モデルとは、「対象の持つ性格を抽象化し、簡略化したもの」です。モデルは、対象の特徴を表現したもので、従って、「モデル=対象」ではありません。

また、いろんなモデルがありうることとなります。モデルは、モデルを作った人の解釈を表現したもののなので、解釈が違えば同じ対象を扱っていても、モデルは違ってきます。

モデルは全体像を把握するために使うものですが、「全体像=第3者の見た姿」ではありません。なるべく客観的に、なるべく全体を捉えたものがモデルですが、人が作ったものである限り、完全に客観的に捉えることは難しいし、全体を完全に捉えることも難しいので、モデルが完全なものであるという意味ではありません。このことを、システム思考の主張者であったバリー・リッチモンドや、SD の権威者である MIT のスターマンなどは、「モデルは完全に正しいものではなく、仮説である」、「あらゆるモデルは必ずどこか間違っている」と言った言い方をして、モデルの限界性に常に留意するよう警告しています。

(2) SD モデル開発方法論

SD モデルの開発の方法の概要を説明するために、このコースの最初に方法論を取り上げます。「SD モデル開発方法論」とは、SD でモデルを作って分析する進め方のことです。初めての人には、ちんぷんかんの部分もあるでしょう。でも、これを心に泊めて作業を進めていくことは重要なことです。多分、最初は、ほとんど理解できないかも知れないけれども、こんなことを聞いた程度に覚えておいて下さい。後で、モデル構築で行き詰った時に、これを思い出してもらえば、多分、やり方が拙かったのだと気がつくでしょう。テキストの 53 頁から 66 頁に対応していますが、ここでは簡略化しています。

2. 方法論

どうシステム・ダイナミックスで分析を進めていくか：

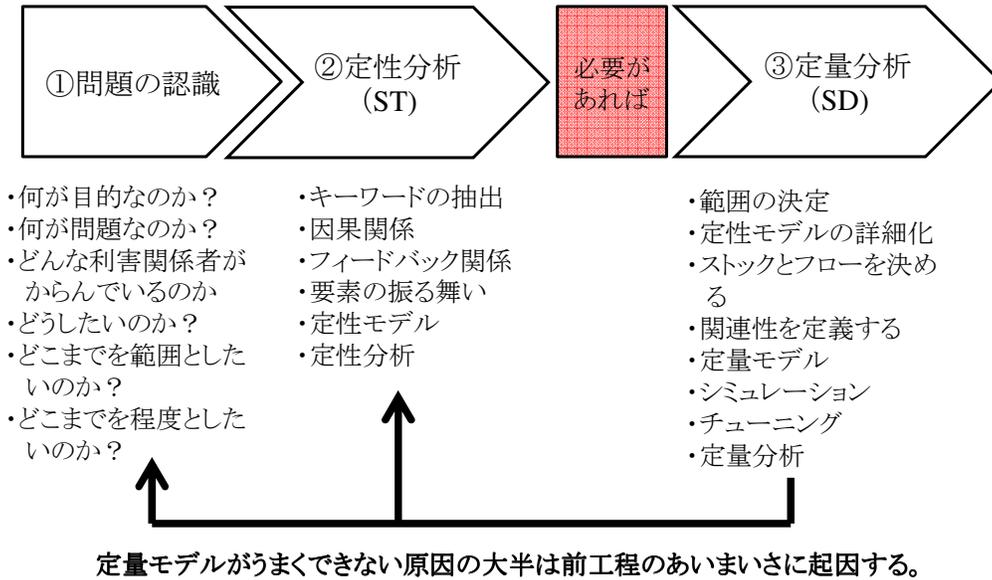


図 2-1：方法論

2. 方法論

何がアウトプットか：

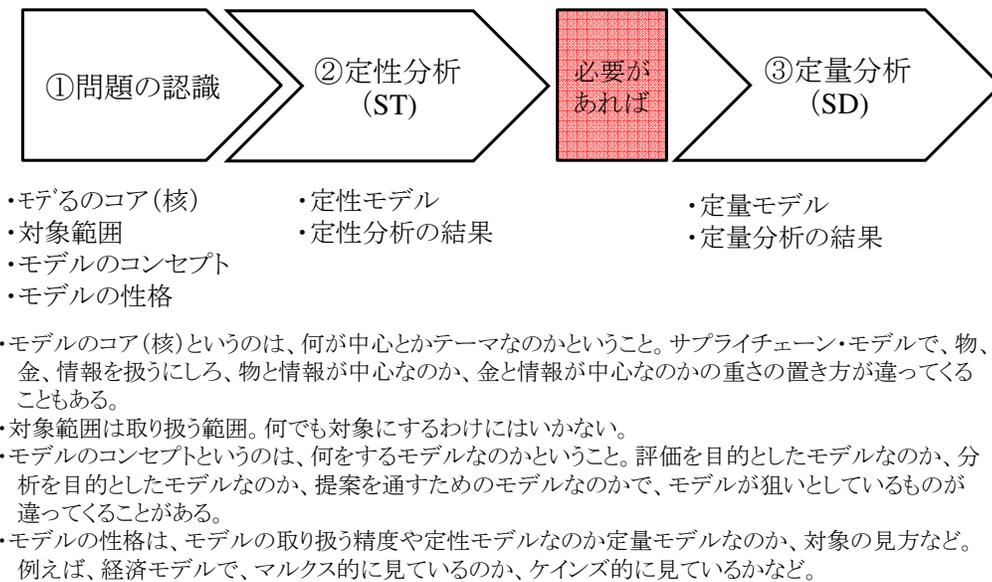


図 2-2：方法論

2. 方法論

どんな支援ツールや手法があるのか？：

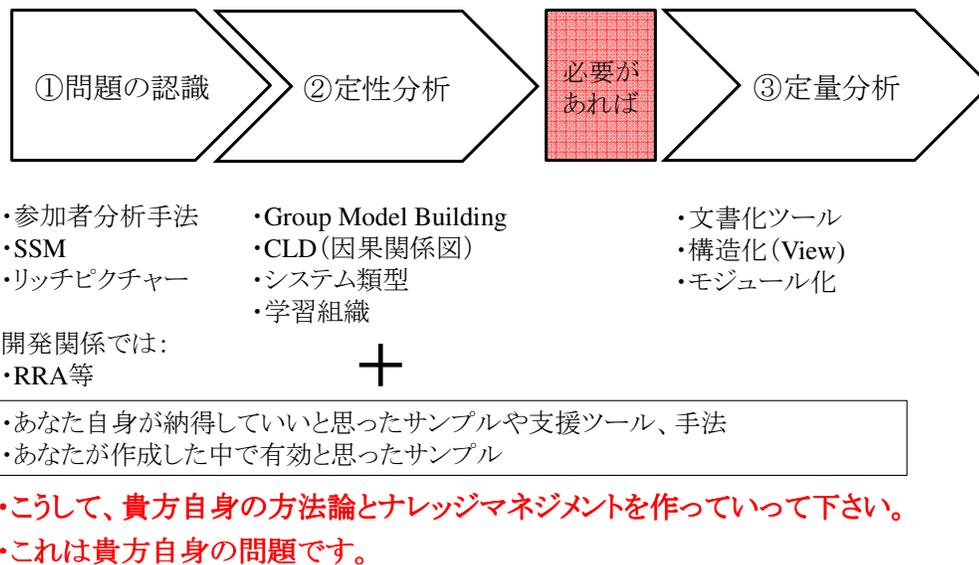


図 2-3 : 方法論

方法論及びナレッジ・マネジメントは、各個人が自分に合うものを構築していく必要がある性格のものであります。また、ここで紹介しているものは、経営コンサルの方法論を基にしているので、分野が違ってくれば、これとは違うものになるかもしれないし、こんな方法論などは不要なのかも知れません。例えば、社会科学の研究用に SD モデルを作る際には、こんな方法論は不要でしょうし、簡単な問題に取り組む際や小さなモデルを構築する際には不要でしょう。

図 2-1 から 2-3 の 3 つに分けて SD のモデル開発方法論を示しています。テキストの 53 頁から 66 頁に対応していますが、テキストでは、経営コンサルタントが使っている方法論をベースに記載しているのに対し、ここではそれをさらに簡略化しています。

この方法論では、3つのフェーズに分けてモデル開発を進めていくことを想定しています。まず、フェーズ1では問題の認識で、モデルの対象を決め、モデルの取り扱う範囲や、モデルの性格などを決めていきます。ただ、後のフェーズでの作業で、不都合があれば、取り扱う範囲を変えたり、対象にするものを変えたりといったことも発生しますので、ここで決めたことが確定され、後で変更がないという意味ではありません。ただ、このフェーズ1で、モデルの対象や、モデルの取り扱う範囲、性格などをある程度明確に決めておくと、作業の手戻りが少なくなると思います。

フェーズ2では、定性モデルの構築と定性分析を行い、さらに定量分析が必要な場合は、フェーズ3に進み、定量モデルの開発と定量分析を実施します。

これは、定量モデルを作る立場から見ての話ですが、定量モデルを作成しようとして困難に直面する原因の大半は、前工程のあいまいさが原因です。通常、定性モデルと定性分析を基に、定量モデルを構築しますが、その際に前提とする定性モデルは、定量モデルから見て、あまりにも曖昧すぎて、そのままの形ですっきり定量モデルに置き換えることは不可能です。また、範囲も広すぎて、定量モデルにその範囲を全て取り込むことが難しい

こともよくあります。これは私の意見ですが、定性モデル構築と定性分析だけでいいのであれば、無理に定量モデルを構築する必要はないと思います。定量モデルを構築する労力は、それまでの労力と比べけた違いに多いので、無駄なことはやめた方がいいでしょう。

ただ、私は定性モデル構築や定性分析が、定量モデル構築の上で無駄と主張しているわけではありません。定性モデルと定性分析によって、モデルの対象や性格を分析し、理解しておくことは非常に重要で、すつとばしていい作業ではありません。

定量モデルがうまくできない理由は主に3つあり、前工程での結果（定性モデル等）が定量モデルを作る上であいまいすぎるか、定量モデルでの表現技術が未熟であるか、あるいは、モデルの対象に対する知識不足・理解不足に起因します。知識不足というと、分かっていない問題を理解するためにSDで分析するのではないかと反論されそうですが、ここで言っている意味は、SDモデルはストック変数、フロー変数、変数、定数のたった4種類だけで、モデルが対象としているものを記述することでモデルを作るので、比喩的に言えば、どう対象を理解して、どう表現したかが問われます。また、理解には、モデル作成において必要かつ十分な、対象に対する知識が必要となります。SDモデルは、報告書と同じく、対象を、作成者が理解した通りに記述したもので、理解していないとモデルはできないし、理解が変だとモデルも変になります。

方法論には、豊富な支援ツールや参考例なども含まれます。あなた自身が、これは有効だと思ったツールや良いと思ったモデルを、こういった方法論の体系に沿って蓄積し、あなた自身のナレッジ・マネジメントを構築していくことを推奨します。

1) フェーズ1：問題の認識

モデルのコア（核）、対象範囲、モデルのコンセプト、モデルの性格を決める作業をここで行います。そのため、以下のような一連の間を設け、それに答える形でこれらを確定していきます。

- ・何が目的なのか？
- ・何が問題なのか？
- ・どんな利害関係者がからんでいるのか
- ・どうしたいのか？
- ・どこまでを範囲としたいのか？
- ・どこまでを程度としたいのか？

「何が目的なのか？」は、モデルの目的を尋ねている問ですが、分析するといったプロセスを聞いているわけではありません。「どうしたいか」という後の問とも関連しますが、モデルの性格、つまり、報告書に付けるモデル程度でいいのか、それとも、改善のために、費用効果なども含めてかなり緻密なモデルを構築する必要があるのかを聞いています。また、モデルの性格として、汎用的なものなのか、ある事項に絞ったものなのかを聞いています。

「何が問題なのか」は、相手が「何が問題の本質と思っているか」という、相手の「見ている姿」を尋ねているのですが、これも、人によって立場の違いによって、見ている対象の姿が違うので、場合によっては理解することが大変です。芥川龍之介の作品で、映画化もされた「羅生門」という小説がありますが、みんな、それぞれ見ている見方や、問題とと思っているものが少しずつ違っているので、「藪の中」になりやすく、それがモデル化を難しくしています。モデル化には、そういった違った見方を統合し、第3者的に見る理解が要求されています。あるいは、そうやって解釈したものが全体像をとらえたモデルになります。この問いで、モデルにするもののコアを仮決めします。

「どんな利害関係がからんでいるのか？」は、作るモデルの対象分野によっては関係ないこともあります。主に、問題の背景を理解するためにこの質問を行い、これに関しては、「関係者分析」、SWOT分析、リッチピクチャーなど、いくつか支援ツールがあります。モデルを作るために、いろんな関係者にいろんなことを質問すると思います。人によって違うのですが、相手は思惑を持って回答するので、場合によっては、回答そのものが本音や真実ではないことがよくあります。利害関係を十分理解した上で質問を行い、利害関係を十分理解した上で、回答を解釈することが重要です。

「どうしたいのか」は、場合によっては問わなくともいい質問です。問題に対してどうしたいかを尋ねていて、「何をどうしたいから ST/SD を使うのか？」という質問の方が適切かも知れません。うまく表現できないのですが、例えば問題解決には、根本的措置と対処的措置があり、どちらなのかとか、「分析して理解する程度でいいのか？」ということを知りたいのですが、もちろん、「そうではない」と言うにきまっています、しかし、実際には、「ふーん、そうか」で終わってしまうことが多く、疲れる質問と結果です。私の場合、相手の言葉による回答よりも、相手の態度を回答にしています。

「どこまでを範囲としたいのか？」は、対象にするのは、会社の全部の問題なのか、ある程度絞られた問題なのかといった範囲のことを聞いていて、さらには、作成するモデルの性格のことを聞いています。作成するモデルの性格というのは、汎用モデルなのか特殊モデルなのかということで、汎用モデルは、多くの問題を分析でき、当対象にしている問題だけでなく、将来の問題にも対応できるのですが、大きなモデルになるので、モデル作成には労力が必要である反面、何でも要素をぶち込むので、あまり考えなくともモデルができるという、「考える労力」という意味では楽なものです。しかし、汎用モデルにしたからと言って、本当に将来の問題に対応できる保証はありません。(まあ、その時になると、なんでこんな要素を見逃したのだ、使えないモデルだと思うことが多々あるでしょう。)

一方、特殊モデルは、対象としている問題に特化したモデルなので、他の問題には必ずしも対応しないものです。ただ、一般に小さなモデルなので、モデル作りにはあまり労力はかからない反面、モデルに取り込む要素を絞り込むので、考えることを必要とします。いずれにしても、汎用的なモデルにするにしても、私はこちらが先(原型)と考えています。汎用なんて嘘で、ただ、あまり考えずに、思いついた要素を全部取り込んだだけとしか見ていません。

「どこまでを程度としたいのか？」は、「かなり正確な数字が必要なのか？」という質問です。松本憲洋氏によれば、①定性モデル、②定量モデルだが、傾向を知ることを目的としたモデル、③定量モデルで、精度をある程度要求されるモデルの3つのモデルがあると、モデルを3分類しています。どれにするかでデータ収集などの労力が大きく違ってきます。私個人としては、全てのSDモデルは2番目のもので、3番目とは程度の問題、つまり、いくら精度を上げてても正確ではなく、傾向しか分からないと思っています。Starmanは、「全てのモデルはどこか必ず間違っている」と、モデルには限界があることを常に認識すべきであると言っています。モデルは対象の中から見たい要素を抜き出して作った人工物なので、実態と違ってくるのはむしろ当たり前だと私自身は思っていますが、でも、実態に合わない批判する人は多いのが現状です。

こうした一連の設問を介して、次のフェーズで開発するモデルのコア(核)、対象範囲、モデルのコンセプト、モデルの性格などを規定します。

2) フェーズ2：定性モデル構築と定性分析

List up of keywords: 要素となるキーワードをリストアップします。

一人当たり所得水準。	排水状況。	土質。	上水道施設整備。	①灌漑設備整備。
農業。	住血吸虫罹患病。	水質。	住血吸虫病対策。	①排水設備整備。
米の州内自給。	保健衛生状況。	農産物増産能力。	運輸インフラ整備。	①展示農場建設。
米の隣州からの移入。		開発計画。	組織開発。	②道路整備。
		灌漑設備の整備。	農村の居住環境。	②粘着場整備。
		農業インフラ。	生活上。	③上水道施設整備。
		農民の収入。		④住血吸虫病対策用排水路整備。
				④住血吸虫病対策保健衛生セミナー。
				⑤総合農業試験場改修。
				⑤農業近代化トレーニング。
				⑥農民組織化トレーニング。
				⑥地方自治体職員トレーニング。
				⑥中央官庁事業担当職員トレーニング。

お断り: 以下、例として作成したもので、詳細に見ると途中で要素名が変わっていたり、追加されたり、削除されたり、使われなかったりする要素やコネクションがあります。実務ではそんなものです。

図 2-4：キーワードの抽出（フィリピン、北サマール農業近代化プロジェクト）

Grouping: グループ化すると要素間の関連性を付けやすいので、要素をグルーピングします。

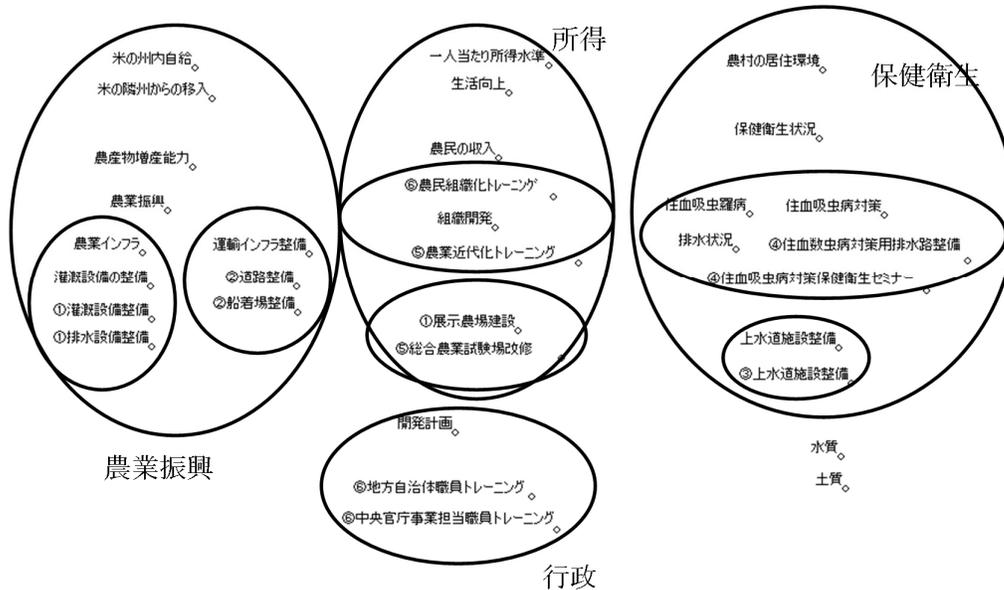


図 2-5：キーワードのグループ化（フィリピン、北サマール農業近代化プロジェクト）

- このフェーズでは：
- ・ キーワードの抽出
 - ・ 因果関係でキーワードを結び付ける

- ・フィードバック関係を発見する。
 - ・要素の振る舞いを分析し、フィードバック・ループの性格を決める
- という作業によって、定性モデルを構築し、構築した定性モデルを使って定性分析を行います。

「キーワードの抽出」では、モデルを作る要素（キーワード）を探し出します。ここが曖昧さの始まりになりやすいので、十分定義、あるいはキーワードの持つ意味を理解しておくことがキープポイントです。経営問題とか社会問題をモデルとして扱う際には、先に関係者分析を行い、背景の共通理解を行っておくことも有効です。人によってキーワードが意味していることが違うので注意が必要です。グループ・モデル・ビルディングで実施する場合は、提出した人に説明させて、議論を行うことで、言葉の定義を明確化するという手法も有効です。そして、キーワードを抽出し終わったら、似たようなものを集めてグループ化します。テキスト 76 頁の図 6-2、73 頁の図 6-3 を参照して下さい。

Make linkage in group: まずは農業振興のグループに含まれる要素の関係付けを行います。要素間の関係付けを行っていく作業で、先のリストで欠落している要素があれば追加します。



例: 整備では新規に設備を建設することも重要だが、既存の設備を保守管理していくことも必要。

図 2-6: キーワードの関係付け (フィリピン、北サマール農業近代化プロジェクト)

「因果関係」では、まず、グループ内で、抽出されたキーワードの原因と結果での因果関係を見出していきます。この際、正の関係と負の関係があり、正の関係は、原因側の因子の値が増加すると結果側の因子の側が影響を受けて増加し、減少すると、結果側の因子の値も減少するという関係です。正の関係は、「+」、「S」などの記号を矢印の先に目印としてつけるとか、矢印を青にするとといったことで区別します。一方、負の関係は、原因側の因子の値が増加すると結果側の因子の側が影響を受けて減少し、減少すると、結果側の因子は増加するという関係です。この関係には、「-」、「O」などの記号を矢印の先につけて目印にするとか、矢印を赤にするとといったことで区別を行います。

グループ内の因果関係の決定が終わったら、グループ間の相関を付けます。グループ間の相関を担う因子を特定し、その因子どうしを結び付けます。テキスト 79 頁、図 6-5 を参照して下さい。

Make linkage between groups: グループ間の関係付けを、要素の関係付けによって行います。

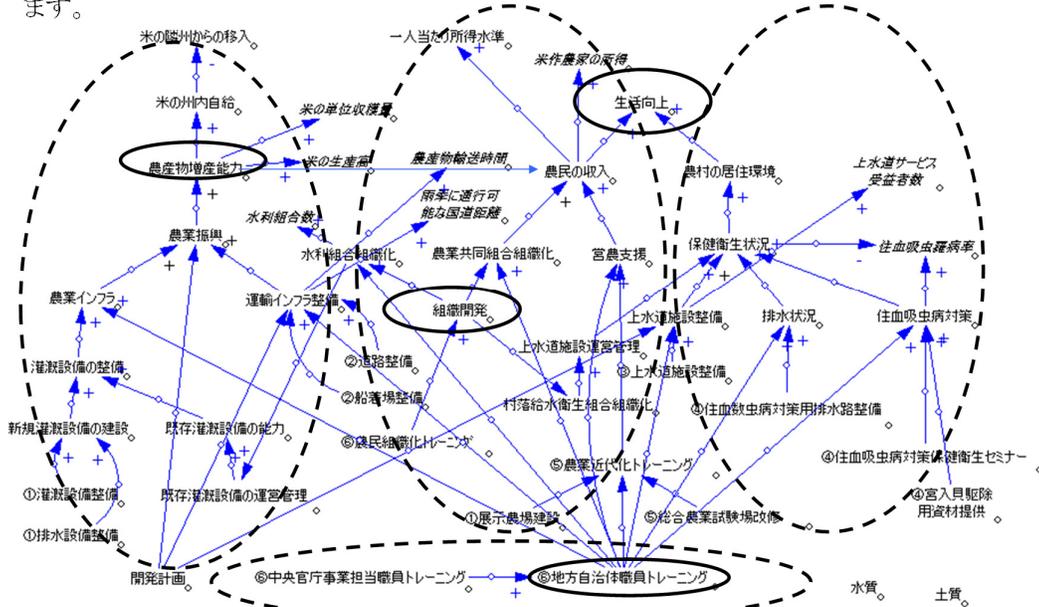


図 2-7: グループ間を結ぶ (フィリピン、北サマル農業近代化プロジェクト)

Find/make feedback loops: フィードバック・ループになっている、あるいはなるはずの部分を見出し、フィードバック・ループを作成します。

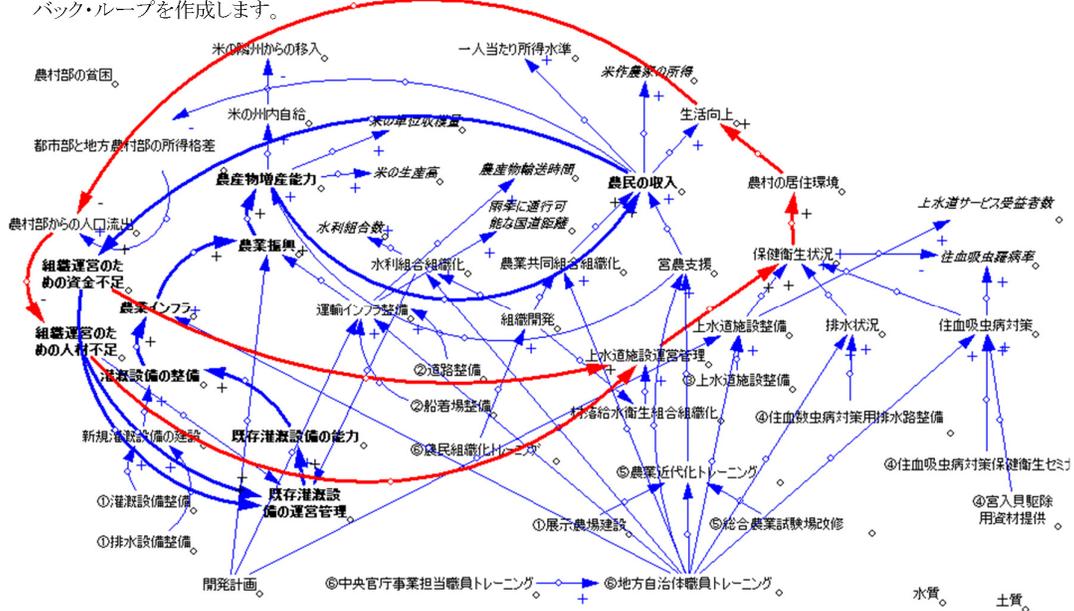


図 2-8: フィードバックの確定 (フィリピン、北サマル農業近代化プロジェクト)

「フィードバック・ループ関係」では、結びつきを見ながら、フィードバック・ループ構造にならないか考えます。やはり、正、負の関係があり、正はループ内の「-」の数が偶数であればフィードバックは増加、あるいは正の性格を持ち、フィードバックは増加すれ

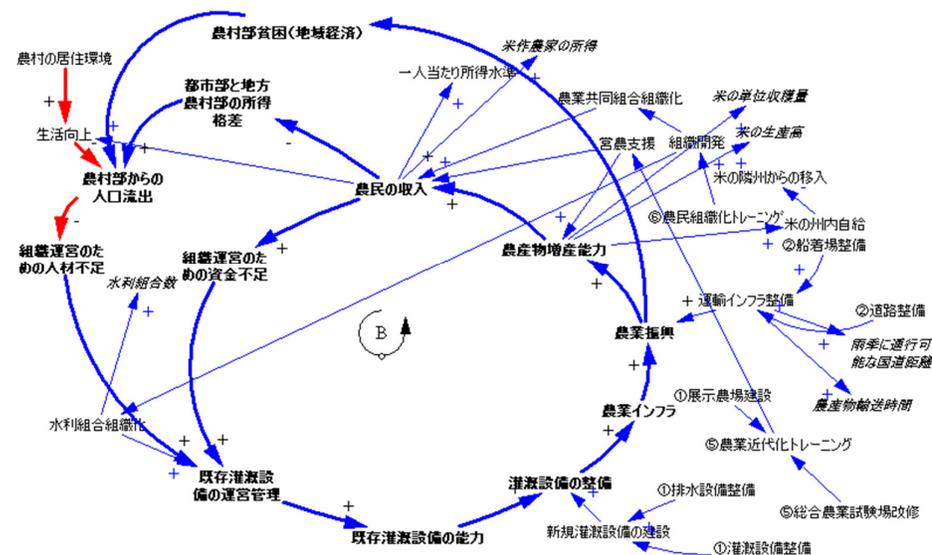


図 2-9：農業基盤整備関係者向けに変形した定性モデル

図 2-4 から図 2-9 までの、定性モデル構築方法で説明した図は、フィリピン、北サマールでの農業基盤整備プロジェクトの評価のために作成したものです。このプロジェクトでは、大きく、灌漑施設やアクセス・ロード、近代化農業普及のためのデモ農場などの施設の整備、農業試験場の整備などの農業基盤整備の部分と、当地で問題になっていた住血吸虫の撲滅や保健衛生状態改善のための保健衛生システム整備の大きく 2 つのコンポーネントがありました。前者がメインで、後者がサブの関係にあります。ただ、この 2 つのコンポーネントでは参加者が違い、前者の主体参加者である男性農民リーダーなどでは、農業基盤整備には関心があるものの、後者にはあまり関心がなく、突っ込んだ議論ができませんでした。後者に関して、主体参加者は、コミュニティ・ヘルス・ワーカーと呼ばれる、引退した看護婦や助産婦など、コミュニティ住民でコミュニティの保健衛生環境の改善に従事する人たちは、保健衛生システムの改善には関心が高いものの、農業基盤整備そのものの細かい議論にはあまり関心がなく、両方を合わせた議論は時間浪費と感じました。そこで、基本的には全く同じものですが、モデルをそれぞれのグループに合わせて変形し、変形したものを元に、グループを分けて現地でディスカッションを行っています。

ある因子の矢印の先が多くの因子に関係を持つような因子は、影響を与えるリバレッジ因子である可能性が高く、ある因子が多くの因子の矢印の先を受ける関係を持つような因子は、問題でボトルネックになっている因子である可能性が高いのですが、ただ、経営資源等の制約で、ボトルネックになっていると思われるその因子の改善が可能かどうかはまた別のことも多いので、これが改善可能なボトルネックという意味で、必ずというわけではありません。リバレッジも同じです。従って、改善可能という点からのリバレッジも、必ずしも、矢印の出が多い因子ではなく、ボトルネックの一つ先の因子を改善し、ボトルネックを改善するという手法もあります。例として、82 頁の図 6-9 では、「農村部からの人口流出」が問題ですが、これは直接的な方法では解決しにくい問題です。そこで、「農民の収入向上」で「農村の生活向上」を図っていくことで人口流出の問題を解決していくという方法がこのプロジェクトでは採択されています。

ボトルネックやリバレッジの発見は、定性モデルで、因子の振る舞いをトレースしてい

くことで発見していただけますが、ボトルネックの発生が、ビジネス・プロセス上で発生する性格のものであれば、業務フローチャートやPERTなどによる分析の方が確実に発見できるでしょう。定性モデルは通常かなりあいまいなので、ボトルネックやリバレッジの可能性のある因子を発見する程度で、改善できるまで特定できる精度での発見までは難しいのが現状です。逆に、定性モデルで発見されたリバレッジやボトルネックは、改善のためにさらに詳細化するためには、業務フローチャートやPERTなどによる分析が必要になってきます。

複数フィードバック・ループがある場合は、どのフィードバック・ループがメインなのかを、見出していきます。一番重要な因子が含まれているフィードバック・ループがメインのループである可能性が高いのですが、しかし、こういった因子に限って、複数のフィードバック・ループの要になっていることが多くあります。重要な因子が多く含まれているフィードバック・ループがメインのループであることが多いので、こういったループに絞って、先の因子の振る舞いをトレースして、メインなのかサブなのかを見出していきます。そして、サブのループを落として、メインだけのモデルにして、問題をうまく表現できているかどうかを確認します。

3) フェーズ3：定量モデル構築と定量分析

このフェーズでは：

- ・範囲の決定
- ・定性モデルの詳細化
- ・ストックとフローを決める
- ・関連性を定義する

ということで定量モデルを構築し、シミュレーションを行い現実との合致度合をチェックし、必要であればチューニングを行ってモデルを完成します。完成したモデルを使って定量分析を行います。

先のフェーズで完成された定性モデルは、それを忠実に定量モデル化できないことが多く、また、できても、モデルが大きく複雑になってしまうことが多くあります。そこで、分析する対象や範囲を定性モデル上で絞り込むことが必要になってきます。「何でも」はできないと最初から思った方がいいでしょう。定量モデル化の範囲としては、メインのループに絞ったものとするのが妥当です。これが、「範囲の決定」と呼ばれている第一作業です。

「定性モデルの詳細化」では、定性モデルを、定量モデルにできるように詳細化します。特に、定性モデルで安易に使われている合成因子には十分注意を払う必要があります。定量モデル作成には、さらなる分解が必要なことが多くあります。ここでは、フローとストックの区別がつくぐらいまで因子を細かくした定性モデルに落とし込みます。メインのストックを中心に、あとは単純化した、定性モデルと定量モデルの間のようなものを作ることも有効で、これができれば、あとはそれを詳細化していくだけの作業になります。

「ストックとフローを決める」では、定性モデルが詳細化できたら、ストックとフローの区別をつけていきます。ただ、何がストック、何がフローと言われても最初は難しいかも知れないし、間違えることも多いかも知れません。モデルを作りながら考える、あるいは変えていくことでも最初は構わないと思います。最初から明確にストック、フローの区別がつくものもありますが、どっちとも言えなく、モデルの作り方によるものもあるので、まずは、定性モデルのメインループの中で一番キーとなる因子から考え始め、それがストックであれば、そこから定量モデルの作成を開始するのがいいでしょう。

「関連性を定義する」では、ストックを中心にフローになる因子を決め、次いで変数及び

定数の関連を決めていきます。関連があれば矢印で結び付け、そして、関連について数式で定義します。ストックに関しては、Vensim が自動的に定義してくれ、定義式が出てきます。通常は初期値の設定ですが、Vensim が自動定義した式を修正することも可能です。こうしてストックをフローが取り巻く形のモジュールのようなものができるので、それらのモジュール間の関連を次に考え、関連があれば、関連する因子どうしを結び付けていきます。

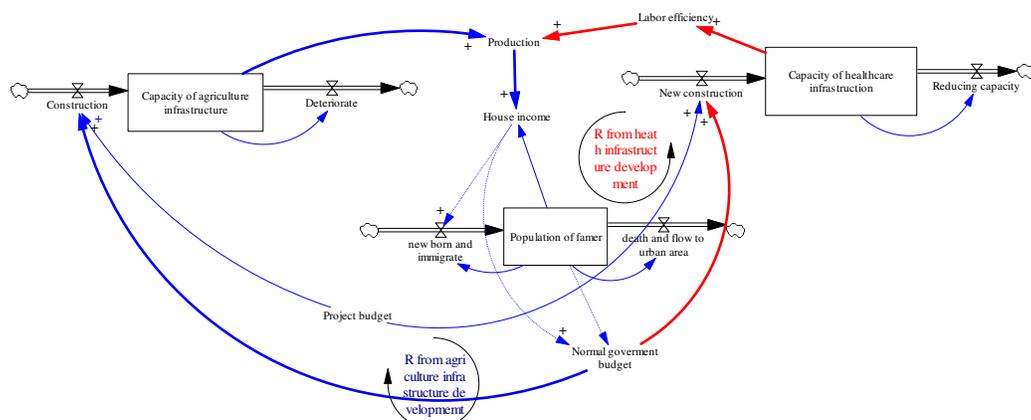


図 2-10：定量コンセプト・モデル

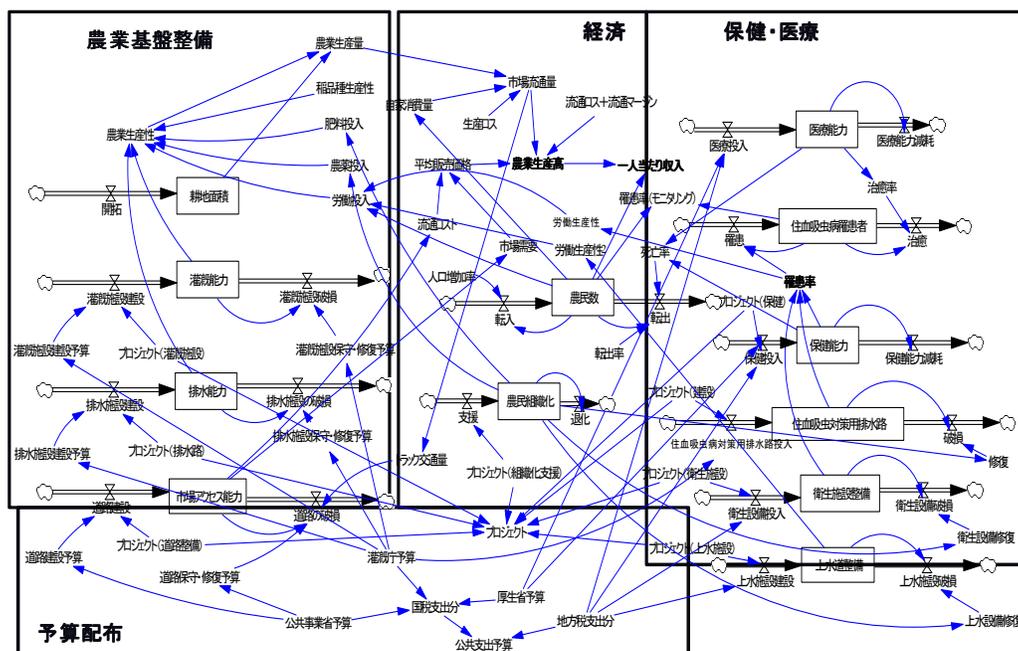


図 2-11：定量モデル（フィリピン、北サマール農業近代化プロジェクト）

大きく複雑な定量モデルの構築では、まず図 2-10 のようなコンセプト・モデルを構築し、そこでコンセプトを明確化し、その後、図 2-11 のような本格的な定量モデルに進化させていくとうまく定量モデルを構築できます。図 2-11 の定量モデルは非常に複雑に見えると思いますが、実は、図 2-10 のコンセプト・モデルでの左側の農業基盤整備のストックとフローをさらに 4 つのストックとフローに分解し、図 2-11 の農業基盤整備のモ

ジュールにしています。同じく、図 2-10 右の保健・衛生基盤のストックとフローを、6 つのストックとフローに分解し、図 2-11 の保健・医療のモジュールに詳細化しています。

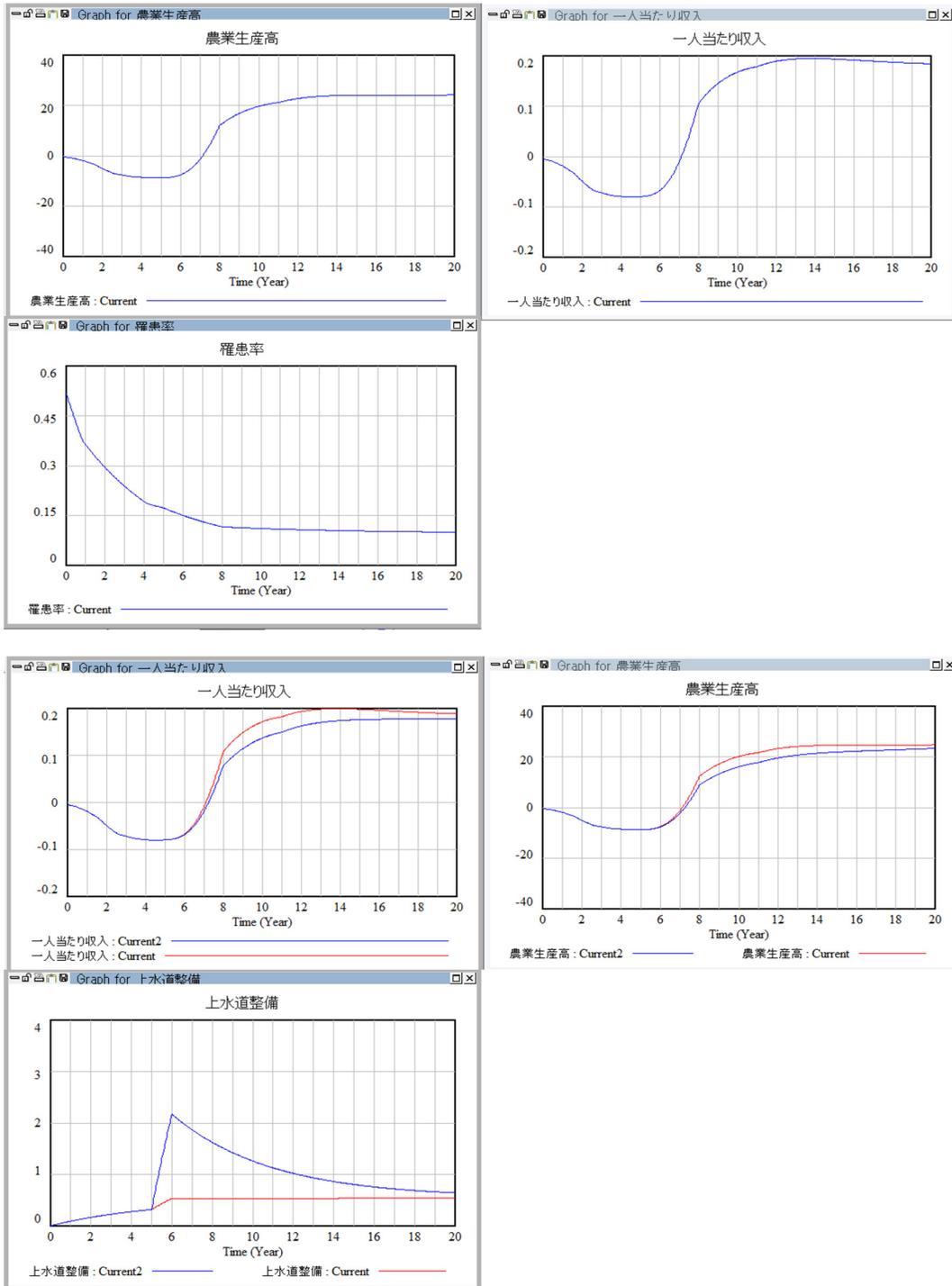


図 2-12：シミュレーション結果（フィリピン、北サマール農業近代化プロジェクト）

こうして、定性モデルとして完成させます。テキスト 90 頁図 7-1 を参照して下さい。完

成したモデルを使い、シミュレーションを行い、結果の表やグラフを見て、妥当性を検討します。おかしければ、妥当と思われるシミュレーション結果になるまでモデルを修正していきます。これをチューニングと呼んでいます。チューニングには、初期値や係数を微調整するものと、因子や変数を追加、あるいは削除して、モデル構造を変えるものがあります。モデル作成に慣れない間は、シミュレーション結果がおかしいのは、因子を見落としていること、あるいは因子の分解が不十分なことが多いので、後者の場合はモジュール追加が必要になってきます。

4) 設計論

よく、定性モデル構築や定性分析のステップを抜いて、定量モデル構築を行う人がいます。小さなモデルや簡単な構造のものではこれでもいいのですが、通常は、いきなり定量モデルに取り掛からない方がいいでしょう。

また、目的が明確ではないものを設計しないことも重要です。これも良くあることですが、分析をすることが目的だなどとプロセスを目的と誤解している人がいますが、ここで言っている意味は、前のフェーズ1で述べたことです。

複雑かつ巨大なモデルを作る前に、コンセプトとなる簡単なモデルを作り、大ざっぱなシミュレーションを行い、大ざっぱに結果を確かめておくことも有効な方法です。コンセプト・モデルでコンセプトを十分確認しておきましょう。複雑かつ詳細なモデルは、このコンセプトチュアルなモデルの設計コンセプトを基にただ複雑化、巨大化させただけであるはずなので、振る舞いやシミュレーション結果は概ね合致しないとおかしいはずですが、私は、大きなモデルや複雑なモデルの構築に反対しているのではなく、目的やコンセプトが不明確なものやあやふやなものを作ることに反対している点、誤解がないように願います。目的も、汎用的とか広く対応できるとか言うのは嘘だと思っています。モデルは必要かつ十分であればいいのであり、それ以上である必要はないと思っています。

以上の、SDモデルを作成する手順や心構えについての説明で、SDモデル構築の概要説明に代えたいと思います。ただ、モデル構築の手法、技術については説明していませんので、それは次回以降に順を追って説明していきます。

(3) 問題とは

すでにモデル開発方法論で述べていますので、繰り返しになりますが、ここでは3点ばかり問題とは何かについて述べます。

1) 問題が、違う立場の人間にはどう見えているか？

対象としている問題が、違う立場の人には違うように見えていることが分かっているといいと思います。立場の違いで見えている姿が違うこと、どのように見えているかを把握するには、リッチピクチャーを使うことで可能なこともあるので、リッチピクチャーは有効なツールです。モデルは解釈をした人の見た姿だということを常に念頭に置いておいてください。私の場合は、海外で仕事をしている関係上、言葉が通じないという事態に遭遇することがあり、絵でのコミュニケーションとして、リッチピクチャーを使っていますが、考えてみれば、同じ日本語でコミュニケーションを図って、意志や意味が通じているはずなのに、実は通じていないのではないかと思う人と出会うことがあります。同じ日本人同士でも、リッチピクチャーのようなツールを使ったコミュニケーションが必要なのかも知れません。でも、「言葉明確、意味不明確」な人はどんなリッチピクチャーを描くのでしょうか？

2) Problem と Issue の違い

これは今ある問題なのか、将来発生しうる問題なのかの区別をしておいて下さいという

意味です。

3) ST/SD では問題は構造を持っていると考える。

ST/SD では問題は、因果関係で構成されていると考えます。また、悪循環構造が存在することがあり、ボトルネックが問題を引き起こしていることが多く、リバレッジが存在する可能性があると考えています。

(4) 第3章のまとめ

この第3章では、SD コースの概要も兼ねて、SD モデルを構築していく方法論について説明しました。その中で、いきなり定量モデルを構築するのではなく、モデルの目的やカバーする範囲、性格などをまず明確にし、それから定性モデル構築と定性分析、その後、定量モデル構築と定量分析に進むことで、手戻りが少なくなること、定量モデル構築がうまく行かない原因の一端には、前工程での検討の甘さがあることを指摘しました。定性モデル構築で、手戻りはある程度は避けられないことですが、あまり曖昧なままに作業を進めるのでは、結局、目的に合ったモデル構築がいつまでたってもできないこととなります。

注 2-1) バリー・リッチモンドは、「システム思考」を以下のように定義しています。まず、対象を要素分解的に考えるのではなく、要素に因果関係があるシステム（メカニズム、しくみ）として捉えます。対象はメカニズムで動いていると考えるわけです。

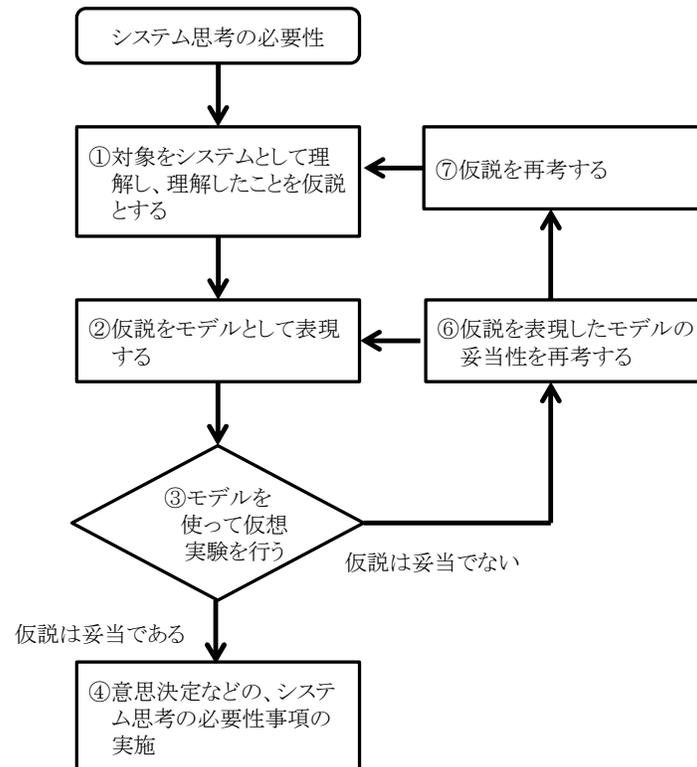


図 2-4 : システム思考とは (バリー・リッチモンドの定義)

さらには、そこには、フィードバック構造があるので、振る舞いとして、増大するとか減少するとか、均衡するとか、振動するというようになると考えます。そして、こういった振る舞いをするしくみを、モデルとして考え、その中にあるどのフィードバックがどの

ような振る舞いをするのか、そしてそれらの組み合わせでどういった挙動になるのかという仮説や、それであれば、どこをどうすればどう変わってくるかといった仮説を立てます。

次に、それらの仮説を、モデルを使って思考実験してみて、実験結果と比べ妥当であると判断できるのであれば、仮説は正しいと考えられるので、しくみや、そのどこをどう変えて改善するという先のアイデアを使って、対象を改善するなどの当初の目的を推進していけばいいでしょう。

しかし、思考実験の結果が仮説を支持しないようであれば、仮説が妥当ではないか、あるいは思考実験に使ったモデルが妥当ではないかどちらか、あるいはどちらも、でしょうから、再度、仮説を再考する、あるいはモデルを再考し、仮説やアイデアを再度思考実験で確かめるプロセスに戻ります。こういった物事の考え方、考える方法を「システム思考」と定義しています。

The “Thinking” in Systems Thinking: Seven Essential Skills, by Barry Richmond, Pegasus Communications, Inc. 2000,

3. ストックとフロー、システムの基本的な振る舞い

ここでは、ストックとフロー、SDモデルの要素の振る舞いについて説明します。

(1) ストックとフロー

定量モデルは、4つの種類の要素で作られます。その4つとは、ストック、フロー、変数、定数ですが、後の3つは同じ性格で、ただ、フローは、ストックに入出力という関係で結び付いているので、区別して、フローと呼んでいます。変数は、Stellaなどでは、コンバーターと呼ばれています。フローと変数、定数は同じ性格のもので、ただ、表現方法が違うだけなので、定量モデルは、ストックとフローだけでできていると言えます。

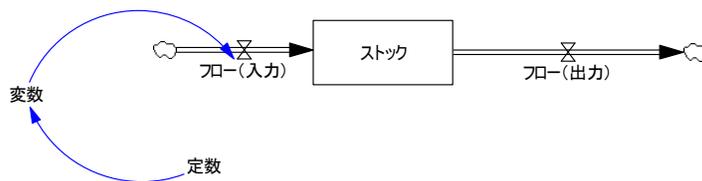


図 3-1：ストックとフロー

1) ストック

ストックは、入出力の差が量として蓄積されるものです。ストックのイメージとして、よく浴槽が例に使われます。浴槽がストックで、水やお湯の蛇口が流入のフロー、栓が流出のフローになります。今、栓を抜いたまま水の蛇口を空けた浴槽を考えた時、栓から抜けていく水よりも蛇口から入ってくる水の量が多いと浴槽に水が溜まっていきます。このように、時間と共に蓄積される量を計測するものとして、ストックがモデルの要素として使われます。Vensim PLEでは四角のボックスで表現されます。

2) フロー

フローはストックに変化を与えるもの、入出力となるものです。先の浴槽の例での水やお湯を浴槽に入れる蛇口や栓に相当します。フローは、ストックと違い、蓄積していきたくはなく、変化するだけです。よく、通過するだけというように表現されます。Vensim PLEでは、雲から出る（あるいは入っていく）三角が中央に2つ向き合って付いた矢印で表現され、必ずストックに矢印が接触する、あるいはストックから出てくるように表現されます。

3) 変数

変数は、ストックに直接に入力となるものではなく、間接的に入力となるものです。性質はフローと同じで、通常は関数や論理式で定義します。

4) 定数

定数は、変化しない値をフローや変数に与えるものですが、SDでは、モデルの作り方として、定数が与えられ、モデルがその後自動的に動くような設計にします。そのため、このような、きっかけを与える定数を政策変数とか、外部変数と呼んでいます。

ストックになる要因やフローになる要因は、先の浴槽の例のように、最初から明確な場合もありますが、モデル設計次第という場合もあります。また、シミュレーションで、ス

トックの値を求めることがSDの目的というわけではありません。ストックに量が蓄積されていく状態を見たい場合もあれば、フローでの変化を見たい場合もあるでしょう。先の例では、浴槽に溜まった水の量を知りたい場合もあれば、お湯と水が浴槽に流れ込むその流量を知りたい場合もあるでしょう。目的ではなく、表現方法と考えて下さい。

(2) 要素の変化のパターン

要素の変化は基本的に8つあります。ここで要素と言っているのは、ストック、フローや変数、定数だけではなく、ストックとフローの組み合わせを一つのシステムの要素と考えることもあり、それらも含んだ、システムの基本的な振る舞いというように考えて下さい。

- 1) 単純な増加（あるいは減少）－直線変化
 - 2) 指数的增加（あるいは減少）
 - 3) Sカーブ：立ち上がりがゆっくりで急成長し、収束する。
 - 4) パルス：ある時点で急に値が変わる
 - 5) 周期変化
 - 6) 収束、あるいは発散
 - 7) 無秩序：ランダム変化あるいはランダムに近い変動
 - 8) 無変化：定数など
- これらが組み合わさって複雑な変化を引き起こします。

1) 単純な増加（あるいは減少）－直線変化

直線変化と次の指数変化、及びSカーブは、基本的にストックを使って表現します。ストックを使った直線変化あるいは指数変化に対し、入出力にいろんな干渉を加えていくと、周期変化を引き起こしたり、収束、あるいは発散したり、無秩序的な動きになっていきます。これを意図的に引き起こすために、周期変化などを入力することもあります。

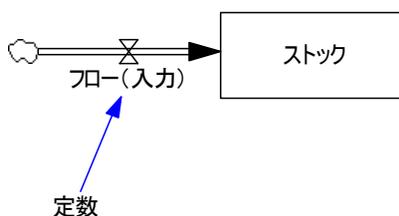


図 3-2：直線変化

図 3-2 は典型的な直線変化のモデルです。今、ここで、

ストックの初期値=3

フロー（入力）=定数

定数=2 としましょう。

これは、

$$y = a + bx$$

という式を表現したことの同じで、初期値 a をベースに b だけ増加していくことを意味し、

$a=3$ （ストック初期値）

$b=2$ （フロー値）

が与えられたことと同じです。こういったモデルの形で、こういった式で定義すれば、必ず直線的な変化になります。

「変化の形」という初等教育教材の中に、「友達を作ろうゲーム」という項目がありました
 が、クラスの生徒が 2 人ずつ、それぞれの友達を呼んで、中央のサークルの中に友達を増
 やしていく、その変化がこれに相当します。

モデル、図 3-2 に関し、注 3-1 参照

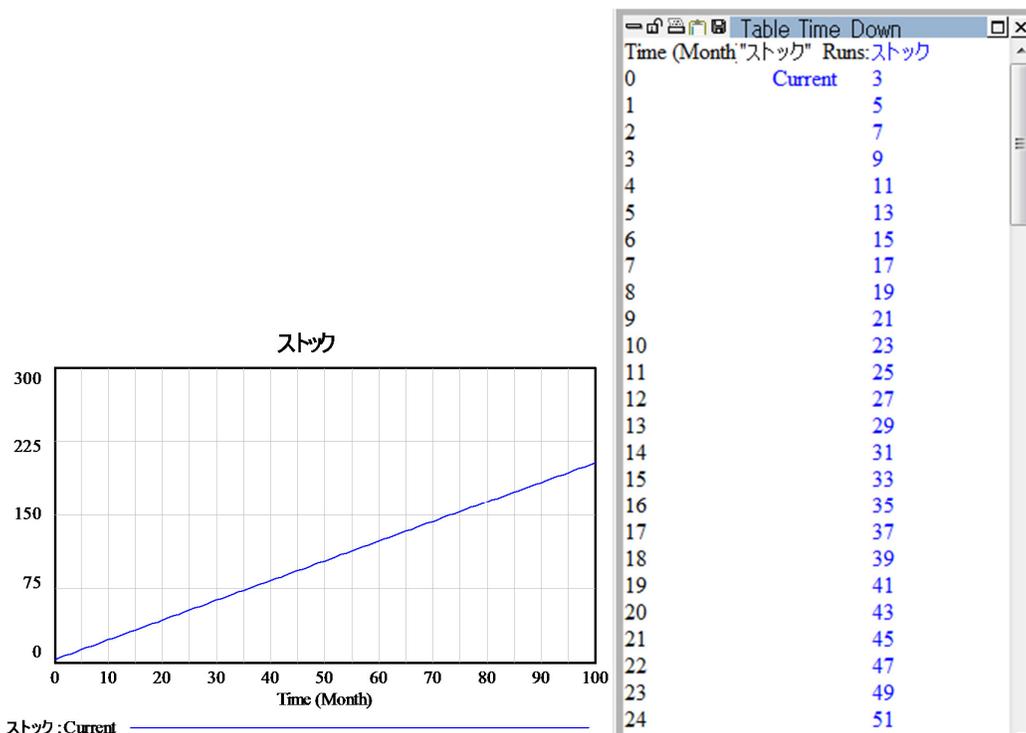


図 3-3 : 直線変化のモデルのシミュレーション結果

2) 指数的增加 (あるいは減少)

指数変化は、ストックからフローに値を戻すことで表現できます。図 3-43 で、ストック
 からフロー (入力) に矢印が引かれていることに注目して下さい。たったこれだけですが、
 これで指数変化を表現できます。

これも、数式的には、

$$y(x)=y(x-1)*b$$

で表現され、初期値 a をベースに b で乗された y が増加していきます。

ここでは :

$a=3$ でこれが $y(0)$ の値 (ストック初期値)

$b=2$ (フロー値)

とすると、初期値を 3 とする値が 2 乗され続けていきます。

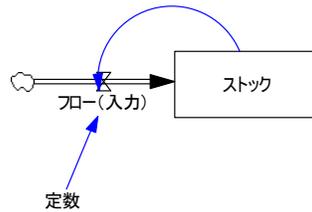


図 3-4：指数変化

このような、ストックからフローにフィードバック・ループがある構造で、先のような指数定義を行えば、システムは必ず指数変化を行います。

「変化の形」という初等教育教材の中に、「友達を作ろうゲーム」という項目がありました。クラスの生徒が 2 人ずつ、それぞれの友達を呼んで、中央のサークルの中に友達を増やしていく、呼ばれた、サークルの中に入った生徒が、それぞれも友達を呼んで、サークルの中に呼び入れるという変化がこれに相当します。

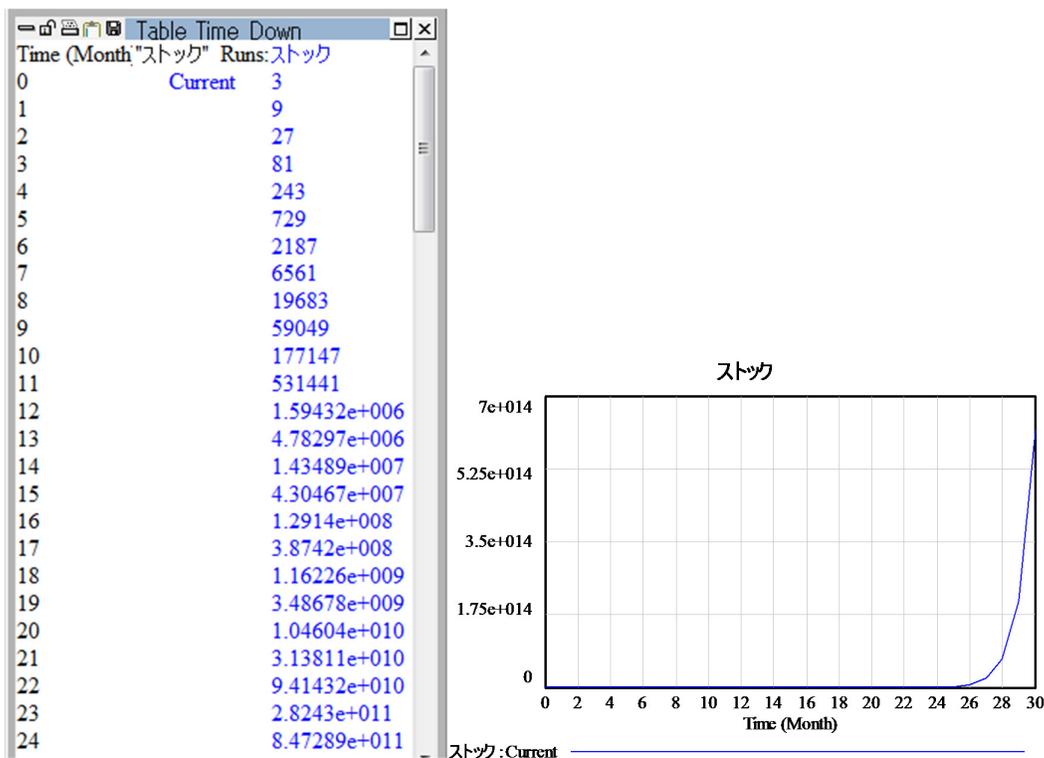


図 3-5：指数変化のシミュレーション例

この指数変化は、人間にとって予想しにくい性格であることが SD では言われています。例えば、池に 1 個の浮草があり、これが毎日 2 倍に増えていっても、池が浮草ですっかり覆われてしまう数日前までは気がつかなく、気が付いた時には、数日後に池が浮草ですっかり覆われてしまうといった例が挙げられています。また、池の大きさにもよりますが、1 週間以内にこうなってしまうと言う、短時間での大きな変化が特徴です。

こんな単純な指数変化でも人間には予想しにくいものなので、ここにいろんな変化が加

わった場合はまず普通は予想できなく、シミュレーションによって計算してみるしかないものです。

3) Sカーブ：立ち上がりがゆっくりで急成長し、収束する。

これは、もちろん、逆の変化、ゆっくり落ち込み、中間は急速に減衰、最後の減衰はゆるやかな変化でもいいわけです。

一般的には、普及モデルと呼ばれる、図 3-6 の例のような、ストックを 2 個、パイプラインで連勝した形で表現します。こうすると、既存顧客は、ロジスティック曲線と呼ばれる、立ち上がりがゆっくりで、中間で急成長し、最後は収束する形の変化を示しますし、製品普及では、釣鐘曲線と呼ばれる、中央で左右対称になる形の変化を示します。また、潜在顧客は、既存顧客を対象にしたような、ゆっくり下降し、中間で急速に下降、最後に下降スピードが落ちるとい変化を示します。

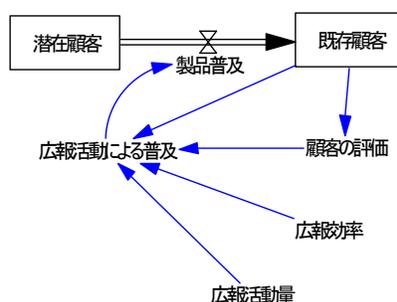


図 3-6：普及モデル例

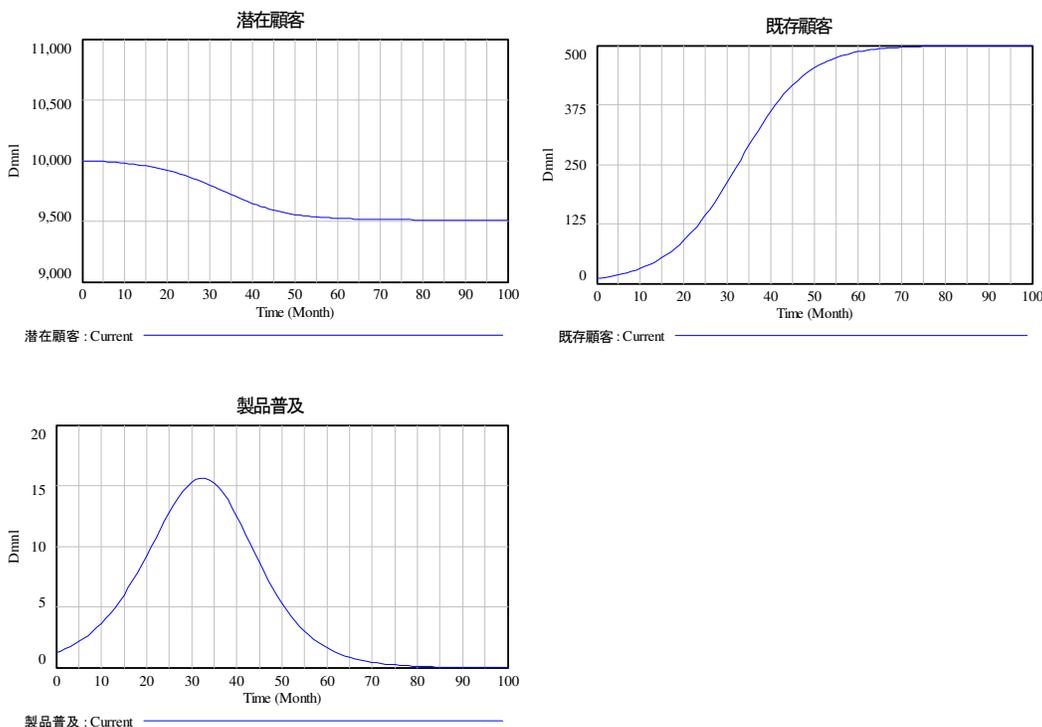


図 3-7：普及モデルシミュレーション例

逆に、こういった変化を与えるために、テーブル関数でこのような形を入力で指定することも多くあります。第 1 回目に、演習で、イースター島のモデルを取り上げましたが、食料の量と死亡率との関係を、既存顧客で示した曲線と同じ振る舞いにしたいので、イースター島のモデルでは、こういった変化の時点での値を食料の量に応じて、死亡率に当てはめています。従って、食料の量と死亡率との関係に限定すれば、イースター島モデルでは、死亡率はロジスティック曲線にそった形で変化していきます。ロジスティック曲線や成長曲線は、もちろん関数表現できますが、曲線の曲がり方などを指定するやり方が複雑なので、多くの場合、SD では、テーブル関数と呼ばれる方法で、直接グラフで指定するやり方を採択しています。

4) パルス：ある時点で急に値が変わる

これは、直線変化に直線変化を組み合わせた場合に現れるパターンです。3 つのタイプ、パルス、ステップ、ランプがあります。パルスは、ある期間、値が突然上昇（あるいは下降）し、期間が終わると基に戻るものです。ステップは、ある期間から急に値が上昇するものです。ランプは、ある期間から徐々に値が上昇していくものです。例えば、ある政策によってプロジェクトが起き、プロジェクト期間だけプロジェクト予算が予算として増えとか、都市開発によって、ある年から住民を受け入れるようになり、人口がある年から徐々に増えるといった変化を表現する場合に使われます。こういった、もともとは直線変化だったもの、あるいは無変化だったものに対し、何か政策のようなものが絡んで変化が起きた場合の振る舞いがこのパターンです。

こういった変化を起こしたい場合は、PULSE、STEP あるいは RAMP といった関数を使って入力値に変化量を与えていきます。

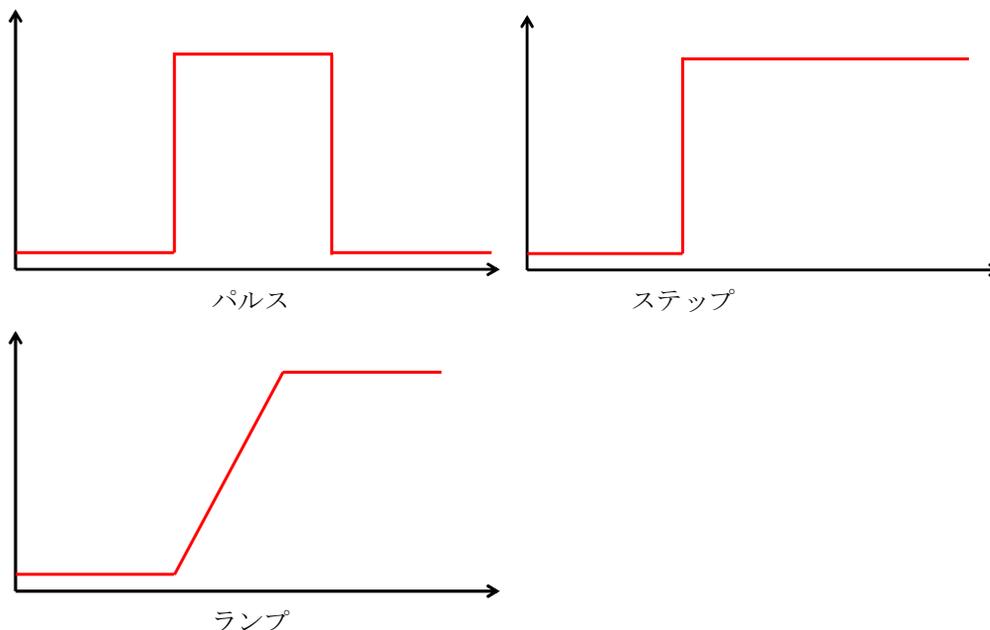


図 3-8：パルス変化

5) 周期変化

周期変化は、ストックに影響を与える入力と出力の間に遅れがある場合に発生します。

この際、入力と出力が均衡していることが重要です。均衡していないと、次に述べる収束あるいは発散していきます。

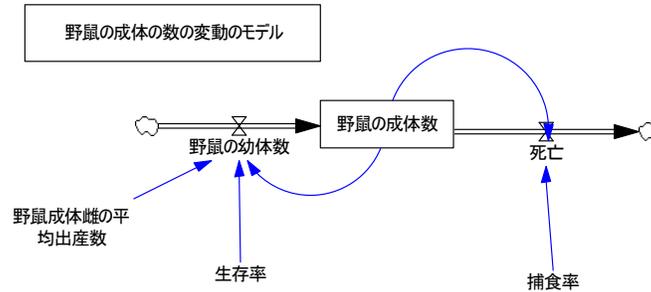


図 3-9：野鼠モデル

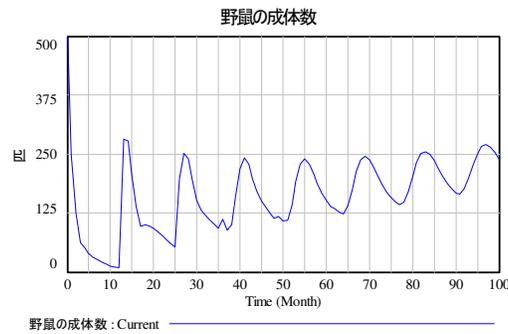
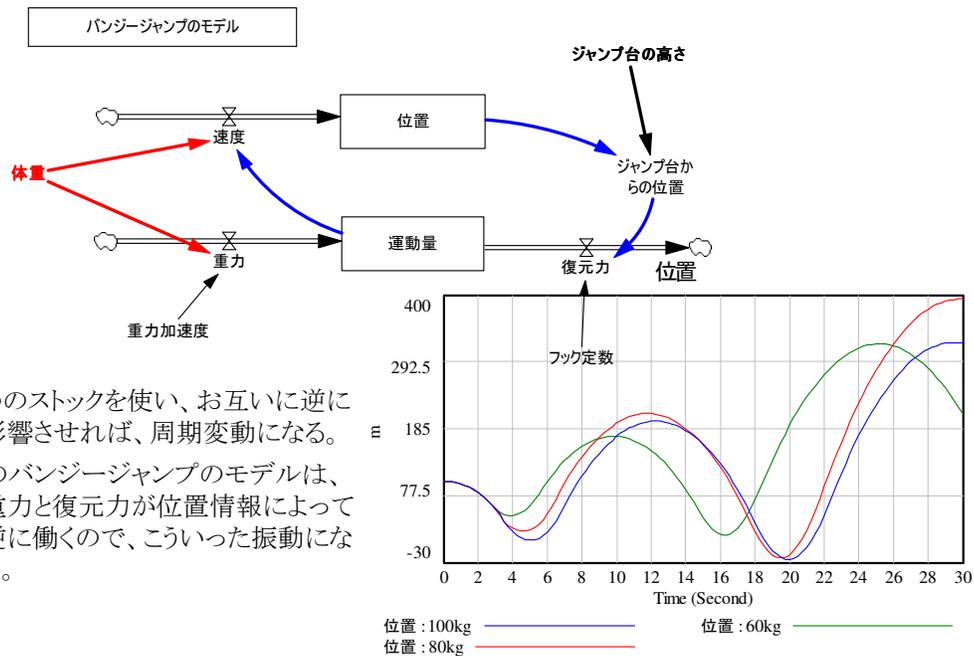


図 3-10：野鼠モデルのシミュレーション結果



- 2つのストックを使い、お互いに逆に影響させれば、周期変動になる。
- このバンジー・ジャンプのモデルは、重力と復元力が位置情報によって逆に働くので、こういった振動になる。

図 3-11：バンジー・ジャンプのモデルとシミュレーション結果

図 3-9 は、野鼠のモデルで、図 3-10 にシミュレーション結果も示していますが、野鼠の成体が子供を産み、それが成長するまでに遅れがあり、一方、死亡で野鼠の成体は減少するので、この遅れが振動を伴い、生誕率の関係で均衡しているのが、周期変化となっています。美しい形と言えない点はお詫びします。

図 3-11 はバンジー・ジャンプのモデルですが、重力と足を縛るゴム・ロープの力に関係するフック数とが位置との関係で逆に働くので、周期になります。

逆に、こういった周期変化を強制的に起こさせたい場合は、SIN 関数などを使って、周期を起こしていきます。ただ、Vensim PLE では COS や TAN などは使えませんので、SIN を基に合成していく必要があります。

6) 収束、あるいは発散

先にも述べたように、ストックに影響を与える入力と出力の間に遅れがあり、入力と出力が均衡していないと、収束あるいは発散していきます。図 3-12 は、図 3-9 に示した野鼠のモデルでの生誕数を変えてシミュレーションしたものを示しています。生誕数が 6 匹だと、出力（死亡）よりも生誕が多くなり、生育での遅れに起因する周期変化を伴いながら発散していきまますし、出産数が 1 匹だと、死亡していく方が多いので、絶滅に向かっていきます。

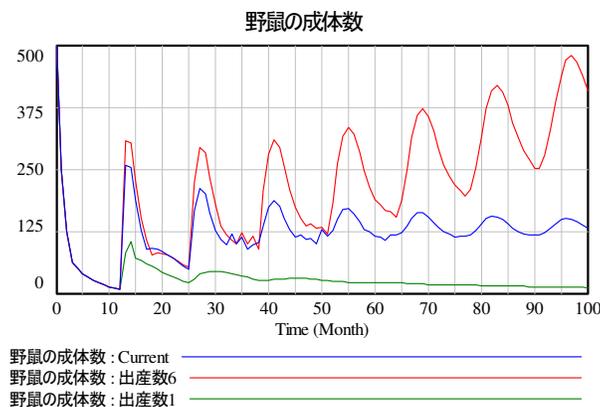


図 3-12：野鼠のモデルのシミュレーション結果

逆に、こういった変化を強制的に引き起こしたい場合は、SIN 関数などや、先のような周期変化を引き起こすものを、直線変化するストックに入力として干渉させていけばいいわけです。

7) 無秩序：ランダム変化あるいはランダムに近い変動

多くの場合、いくつもの変化が重なると、無秩序に近い変化になります。従って、意図的に設計しなくとも、フローにいろいろな変数が干渉するとか、いろんな変数を合成していくと、そのフローや変数は、無秩序状態になるのが普通と言えます。

乱数を発生させ、それを基に何かを決めたいということも良くある話です。例えば、小売店の SD モデルを作成する際に、来客を乱数にし、店を訪れる顧客数も時間間隔も無秩序としてシミュレーションしてみて、カウンターの数や店員数の最適な状態を求めるといったことが考えられますが、こういった場合に、乱数が用いられます。

図 3-13 は、シェクスピアの「ロミオとジュリエット」の最初の場面、第一幕第一場の、敵同士であるモンタギュー家とキャプレット家が市場で乱闘を繰り広げる場面をモデルに

したものです。両家の争いは、常に一発触発の関係にあるとは言え、いつ争いになるか分かりませんし、いつも乱闘になるわけでもありません。そこで、このモデルでは、市場の争いを、乱数によって、ある値以上及び以下で乱闘になるとしています。ある値以上はモンタギュー家の勝ち、ある値以下はキャプレット家の勝ちとしています。こういったやり方（モンテカルロ法）は、企業のマーケティング戦略のシミュレーションなどでもよく用いられる方法です。

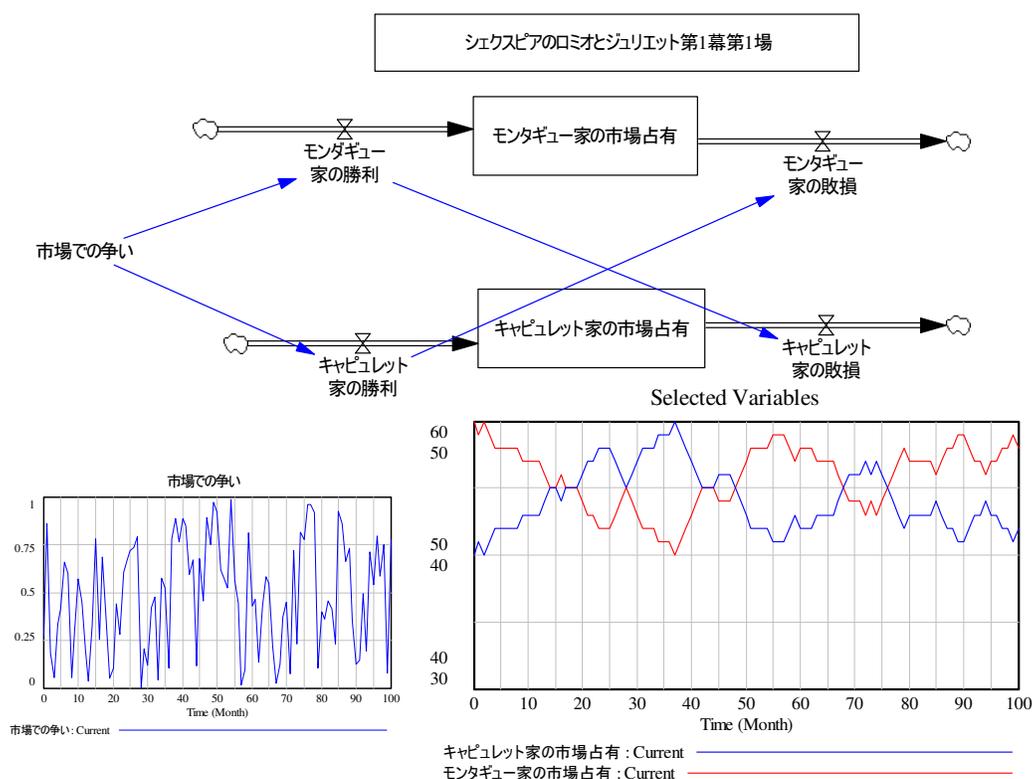


図 3-13：ロミオとジュリエットのモデルとシミュレーション結果

8) 無変化：定数など

無変化は、定数などで与えられるので、定数を一般に思い浮かぶかも知れませんが、モデルによっては、変化してはいけない要素が変化するかどうかをチェックするといった使い方も用いられます。例えば、現金だけの話で、売掛金や買掛金を含まない場合の話ですが、別々に計算した、売上と費用の差（利益）と、キャッシュフローでの現金が同じ金額でなくてはならないという原則がありますので、もし違っていれば大問題です。売上と費用の差（利益）とキャッシュフローでの現金の差をギャップという変数にし、この変数が常にゼロであれば問題はありませんが、シミュレーションしてみて、ゼロ以外の値を示した場合には、どこかにモデル、あるいはデータとしておかしい部分があります。こういったモデルのチェックやデータのチェックなどにも、バランスの原則を応用して、この無変化が使われます。

むろん、多くの場合は、定数は、政策変数や外部定数などとして、モデルにきっかけを与えるものとして用いられます。

モデルを構築していく際に、多くの場合は、意図などしないで、結果が先のような振る

舞いになってしまう部分もあるでしょうが、意図してそういった設計にする、あるいは、そういった振る舞いになるように設計する部分もあるはずで、SD モデルの設計に際し、要素の変化は、基本的に、この 8 つのパターンであり、それが組み合わせられていくだけで理解していただければと思います。

なお、私は制御工学に関しては全く何も知りませんので、ひょっとしたら、この説明は、制御工学的に間違っているかも知れません。

(3) 第3章のまとめ

定量モデルは、ストック、フロー、変数、定数で記載します。また、SD モデルの要素には基本的には 8 つの振る舞いしかありません。システムは、この 8 つの振る舞いが組み合わせられて複雑になっているだけというように極言できます。

注：

3-1) ストックに対するフローですが、入力も出力もプラスかマイナスだけの違いで、区別はありません。マイナス値をフロー（入力）に入れると、フロー（出力）に正の値を入れた結果と同じになります。

4. モデルの基本形

ここでは、モデル構造の基本形という意味で、2つのストックを含むモデル、そしてパイプライン構造のモデルを説明します。定量モデルは、基本的には、ストックとフローの組み合わせを最低 1 個持ちますが、システムと呼ばれるためには、そういったストックとフローの組み合わせが 2 個以上あり、それが独立していて、相互干渉し、その相互干渉の状態を表現するか、あるいは、パイプライン構造と呼ばれる、連結し、遷移状態を表現するものとなります。

(1) 2つのフローを含むモデル

2つのフローを含むモデルは、モアクラフトなどは、SD の基本構造と言っています。2つのストックを使って、基本的には以下の4つの振る舞いを表現することができます。

- 周期遅れの干渉（結果を遅らせる）
- 増強あるいは減衰を高める（早める）
- 打ち消し合い
- ランダムネス

1) 周期遅れの干渉（結果を遅らせる）

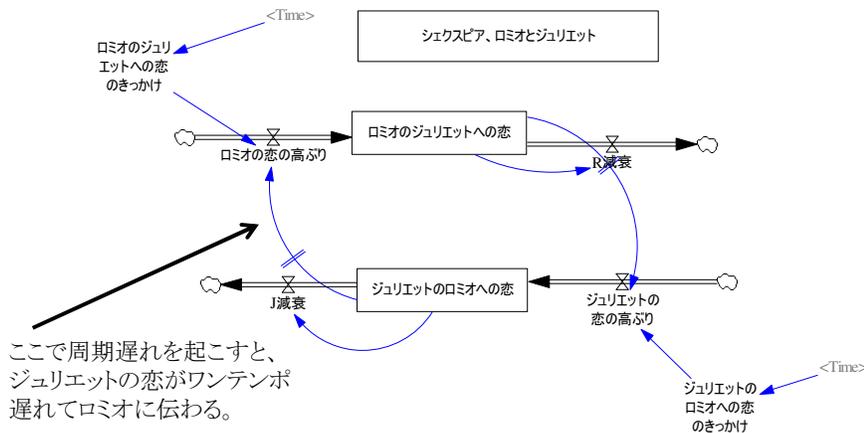


図 4-1：ロミオとジュリエットのモデル

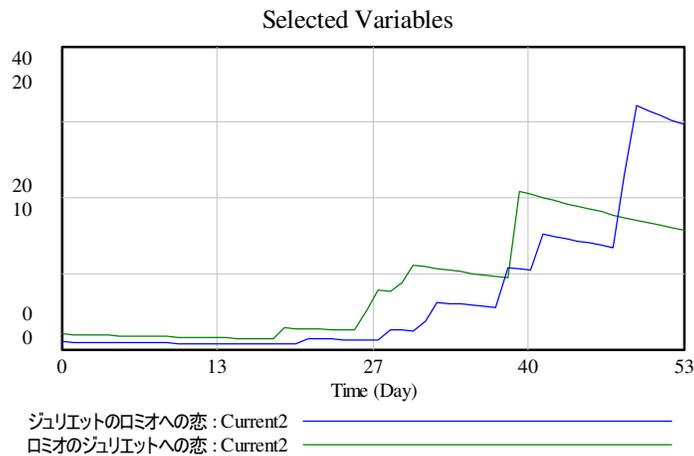


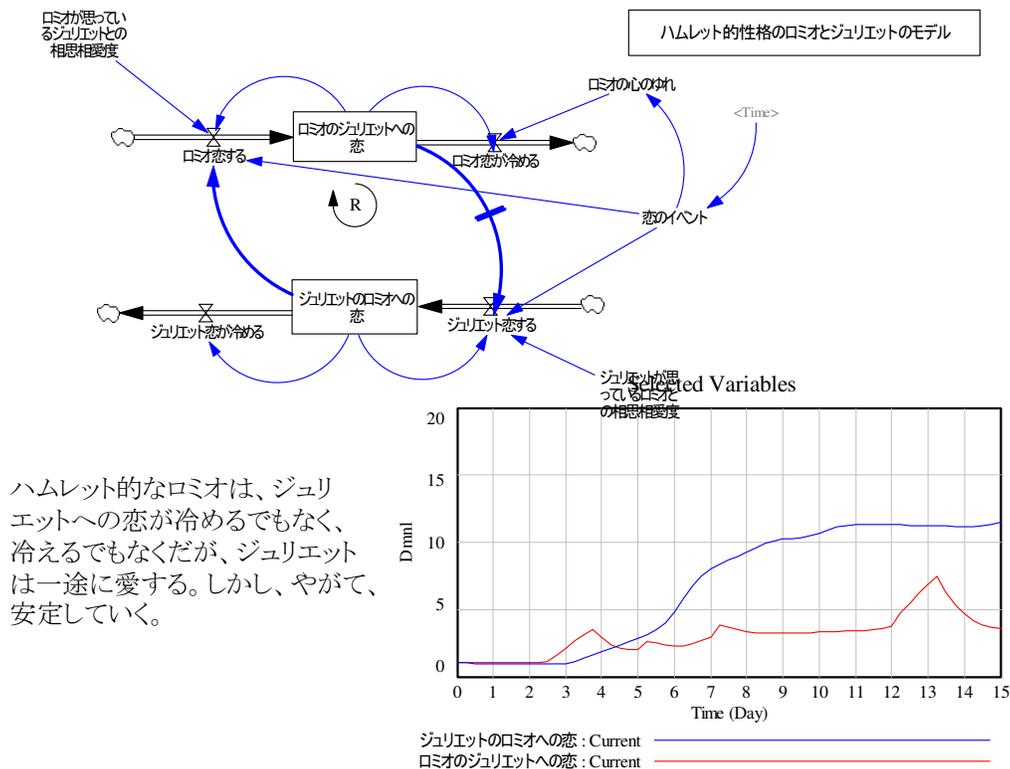
図 4-2:ロミオとジュリエットのモデル

一方のストックの値を他方のストックに周期を遅らせて入れてやる、あるいは引いてやることで、結果を遅らせる（遅延を増強する）ことができます。

図 4-1 と 4-2 は、かなりいいかげんなロミオとジュリエットのモデルで、ここでは、ロミオの愛の告白がジュリエットに少し遅れて伝わり、ジュリエットの愛の受け入れが、ロミオに少し鈍く受け取られるというモデルですが、2 人の愛は高まっていきませんが、遅れが伴うので、少しちぐはぐになっていきます。

2) 増強あるいは減衰を高める（早める）

あるいは振幅を増強するといった、変化を増強することができます。図 4-3 は、ロミオをハムレット的な性格にしたもので、これもいいかげんなモデルですが、ロミオがはっきりしないので、ジュリエットも最初は愛が高まりますが、次第にそれ以上高まらなくなっていくという結果を示しています。この場合は、増強を制限する例になります。



ハムレット的なロミオは、ジュリエットへの恋が冷めるでもなく、冷えるでもなくだが、ジュリエットは一途に愛する。しかし、やがて、安定していく。

図 4-3：ロミオとジュリエットのモデル

3) 打ち消し合い

相手の効果を打ち消し合うことができます。ゼロサム状態を表現する際に良く用いられる方法です。「図 3-13：ロミオとジュリエットのモデルとシミュレーション結果」で見ましょう。モンタギュー家の勝ち、キャプレット家の負けになり、相殺しあって、モンタギュー家の市場占有率とキャプレット家の市場千湯率の合計値は常に 100 であり、それを超えることも、下回ることもありません。マーケットのモデルなどでも同じような構造にして、競合相手と自社とのマーケット占有率の合計を 100% で変わらないように表現します。

4) ランダムネス

ランダムネスは、多くの変化が重なれば、自然にそうになってしまう面もありますが、最少の要素で引き起こすためには、乱数を使うか、最低 2 個のストックを使う必要があります。

(2) パイプライン構造

何かの遷移を表現する際に用いられる、SD では基本となるモデル構造です。図 4-3 と図 4-4 のモデルを比べてみましょう。

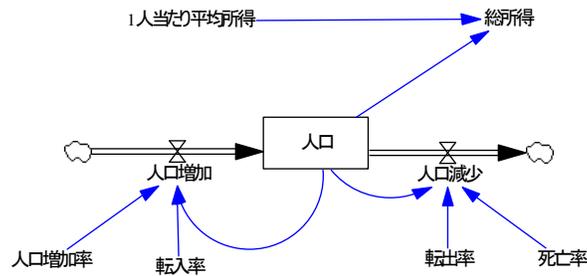


図 4-3：農村経済モデル

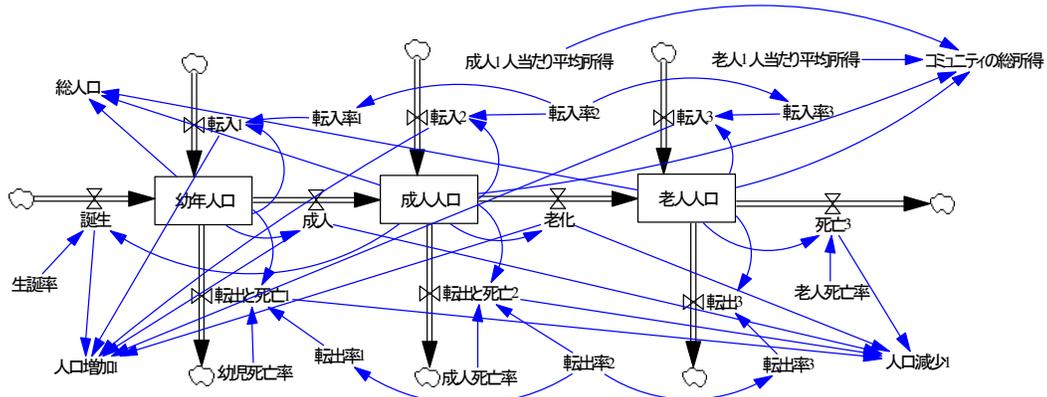


図 4-4：農村経済モデル

図 4-3 では、人口が、人口増加と人口減少で決まるようになっています。そこに、一人当たり平均収入を掛けて、村の総所得を計算しています。普通はこれでもいいのですが、もっと厳密に算出したい場合は、これでは粗すぎます。そこで、図 4-4 のように細分化します。4-4 では、誕生した赤ん坊は、幼年人口というストックに入り、18 歳までここで過ごします。18 歳を超えると、成人人口というストックに入り 27 年間過ごし、次いで、老人人口というストックに入り、徐々に死亡していきます。こういった人口の状態の推移を、ストックが連なる構造で表したものがパイプライン構造と呼ばれるモデルです。まず、幼年では子供を産みませんし、老年では子供は産めません。幼年者は所得がありませんし、老年者は、所得はあっても年金などが収入源で、成人者の稼ぐ所得よりは低くなります。こういった条件の差で、ここでは人口を 3 つに分けています。こうして、人口や所得を、図 4-3 で計算したものよりもっと厳密に計算します。図 4-3 では、人口は緩やかに減少するということしか分かりませんが、図 4-4 で計算すると、そういった変化を起こすのは成人人口だけで、幼年人口は激しく減少し、老人人口は逆に激しく増加していて、この農村は高齢化が進ん

でいくということが分かります。こうして、パイプライン構造の SD モデルにすることで、人口動態を詳細に分析していくことが可能になります。

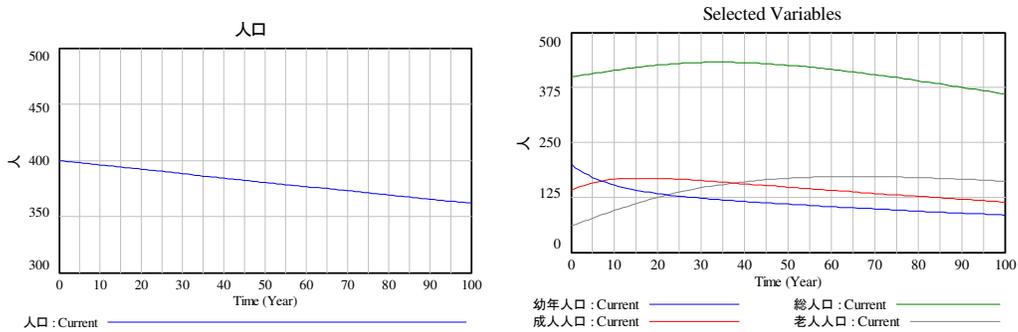


図 4-5：農村経済モデル
(左は 4-3 のシミュレーション結果、右は 4-4 のシミュレーション結果)

もう一つ例を挙げましょう。図 4-6 は人事モデルの例で、必要な採用人員を決め、新人採用し、年月によって係長、主任、課長、部長と昇進していく様子をモデルにしたものです。

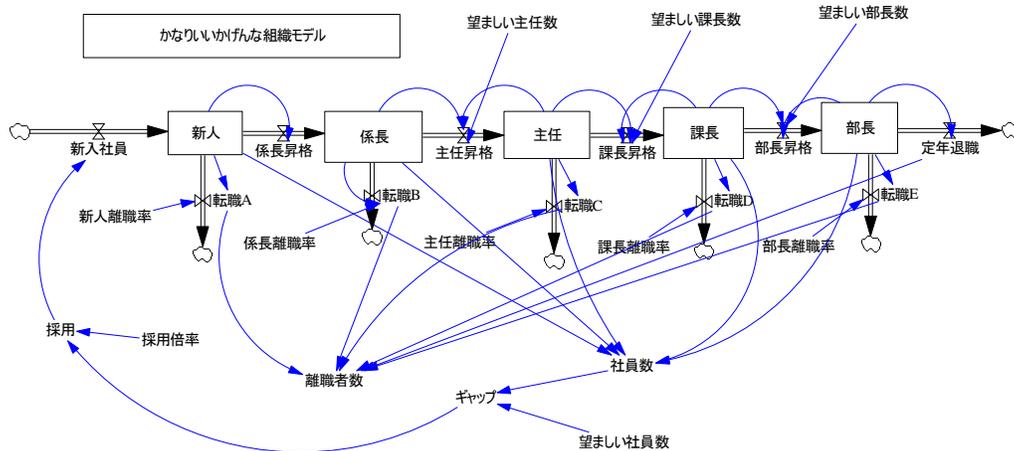


図 4-6：人事モデル

こういった遷移は、何も人口だけではなく、サプライチェーンでの物流などでも見られる事項で、遷移という状態に注目する、あるいは計測する必要がある場合は、モデルでこういったパイプ構造と呼ばれる構造でモデルを設計していきます。

また、図 3-6 で示した普及モデルも、潜在顧客から既存顧客へと状態遷移を起こしているので、これもパイプライン構造のモデルということが出来ます。

(3) Vensim PLE で使える関数

ここでは、代表的なものを説明していて、関数の全部は説明しません。上位版に誘発する気はありませんが、Vensim PLE では使える関数には制限があります。

- ①絶対値：マイナス値でもプラス値を返す
ABS(x)
- ②遅延

DELAY FIXED(遅れ値, 遅れ時間, 遅れまでの初期値)

これは、単純に遅れるだけのもので、パイプライン遅れなどと呼ばれます。

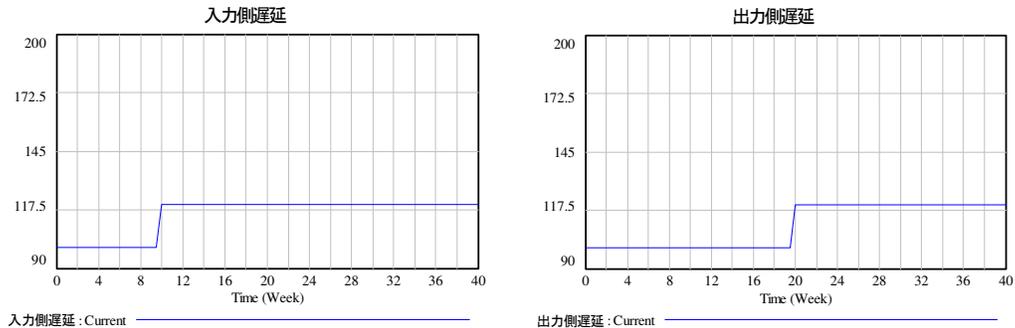


図 4-7 : パイプライン遅れ

図 4-7 は、

入力側遅延 = $100 + \text{STEP}(20, 10)$

出力側遅延 = $\text{DELAY FIXED}(\text{入力側遅延}, 10, 100)$

を表現したもので、遅延時間を 10 としているので、入力側で 10 週目に 100 から 120 になるという入力側の変化が、単純に 10 週遅れ、出力側で 20 週目に 100 から 120 になります。

これに対し、製造指示で製造個数を 100 個から 120 個に増やすという例を考えてみます。製造員の中には、すばやく対応して生産能力を上げられる人もいますが、なかなか新しい製造ペースについてこれなく、生産能力がすぐには上がらない人もいて、しかし、10 週もたてば、みんな増えた個数を生産するこつのようなものを習得し、120 個という目標が達成できるようになります。こういった変化を指数遅れと呼び、 DELAY 、 DELAY3 などの関数を使って表現します。

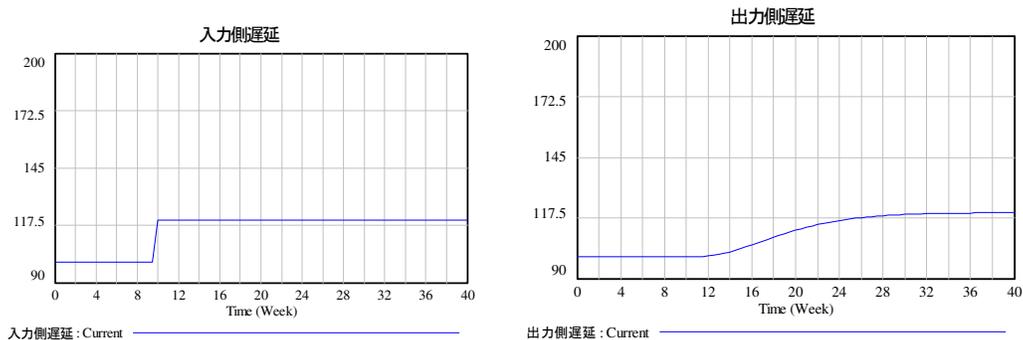


図 4-8 : 指数遅れ

図 4-8 は、

入力側遅延 = $100 + \text{STEP}(20, 10)$

出力側遅延 = $\text{DELAY3}(\text{入力側遅延}, 10)$

とし、入力側遅延で 10 週目に 100 から 120 になる変化は、出力側遅延では、遅延時間を 10 としているので、10 週から指数平滑化した変化になり、10 週目から徐々に変化し、約 30 週以降 120 に安定します。

③ 指数関数、対数

$\text{EXP}(x)$

LN(x)

⑤最小値、最大値（最少値、あるいは最大値を返す）

MIN(x1, x2)

MAX(x1, x2)

⑥除算の余り（小数点以下の数字を返す）

MODULO(分子、分母)

⑦整数出力（整数にして返す）

INTEGER(x)

⑧ゼロ除算防止（ゼロで割った時無限大になることを防止する）

XIDZ(分子,分母,ゼロ除算時に返す値)

ZIDZ(分子,分母)

ZIDZ はゼロで割った場合、ゼロを返す

⑨パルス、ステップ、ランプ

PULSE(開始時,継続期間)

STEP(高さ, 開始時)

RAMP(傾斜, 開始時, 終了時)

⑩乱数

RANDOM UNIFORM(最少値,最大値,疑似乱数初期値)

RANDOM NORMAL(最少値,最大値,平均値,標準偏差値,疑似乱数初期値)

⑪三角関数

SIM(x)

⑫平滑化

SMOOTH(平滑化値,平滑化期間)

平滑化関数は、使い方によっては、指数遅れの関数（DELAY）を使ったことと同じような効果になります。

⑬平方根

SQRT(x)

他に、関数定義には、論理式も使えます。

:AND:

:OR:

>

<

=

などが代表的なものです。

Vensim PLE では関数扱いされていますが、もちろん IF THEN ELSE(条件式,真の場合,偽の場合)は多用されます。ただ、IF THEN ELSE 関数の中では DELAY は使えないが、DELAY 関数の中では IF THEN ELSE は使えるなど、組み合わせの制限があるので注意が必要です。

マニュアルに詳しい解説があり、「Vensim PLE 入門」にも簡単に関数の説明があるので詳細はそちらを参照下さい。

(4) シミュレーションにおける計算方法

シミュレーションでは、コンピュータは無限を扱えないので、差分方程式にして計算します。その際に、一般には、オイラー法が使われますが、刻み値が荒い場合、誤差が問題になることがあります。これは刻み値を細かくすることで解消できます。オイラー法は少し計算方法が荒いので、それを解決する方法として、第 4 項まで展開したルンゲ・クッター法（RK4）も使えますが、ルンゲ・クッター法は、DELAY 関数などを使った場合にエラー

一になることがあります。通常はオイラー法で十分で、約 1,000 項目の大きなモデルを、刻み値をかなり小さくしてオイラー法でシミュレーションしたことがあります。ちょっと計算が遅い程度で、コンピュータが何分もフリーズしてしまうということはありませんでした。

(5) 4 章のまとめ

単純化すると、SD モデルの基本形は、ストック、あるいはモジュールの干渉の状態を表現したものか、あるいは、ストックやモジュールが直列し、パイプライン構造になり遷移状態を表すものになります。普通のモデルは、それらを組み合わせ、複雑化しただけというように極限できます。

5. システム原型

第2章で説明したように、SDモデル構築では、最初に、モデルにする対象を明確化し、定性モデルを構築します。構築した定性モデルに基づいた定量モデルを構築するのですが、定性モデルを元に定量モデルを構築する方法に何か決まった法則や方法のようなものがあるわけではありません。さらには、定性モデルの要素と定量モデルの要素は対一とは限らなく、定性モデルを詳細化したからと言って定量モデルにそのまま変換できるというわけではありません。簡単な定性モデルが複雑な定量モデルになることもあれば、複雑な定性モデルが比較的簡単な定量モデルになることもあります。そこで、定性モデルから定量モデルを構築していく際に、その中間のコンセプト・モデルを作り、肉付けしていくというやり方が行われています。定量モデルを構築していくためには、知っている定量モデルのパターンや型を、鉄筋コンクリートの鉄材のように、このコンセプト・モデルに当てはめて、作っていきます。この際に、第3章で紹介した、8つの変化や2つのストックが相互干渉するモデル、そしてパイプライン構造をモジュールなどの基本に据えて詳細化していくとモデルが作りやすくなると思います。8つの変化とは以下のような変化です。

- 1) 単純な増加（あるいは減少）－直線変化
- 2) 指数的增加（あるいは減少）
- 3) パルス：ある時点で急に値が変わる
- 4) 周期変化
- 5) 収束、あるいは発散
- 6) Sカーブ：立ち上がりがゆっくりで急成長し、収束する。（もちろん、逆の変化、ゆっくり落ち込み、中間は急速に減衰、最後の減衰はゆるやかというのもこれに含まれる）
- 7) 無秩序：ランダム変化あるいはランダムに近い変動
- 8) 無変化：定数など

これらの変化を意識し、モデルの対象要素に関し、そういった変化になるように設計します。複合要素では、あるいはそういったように最初に設計することは難しいので、複合される前の要素でそういった変化になるように設計すると共に、複合された要素の振る舞いが、計画されたような複合された振る舞いになるかを確認します。

(1) システム原型

最初は、「成長の限界」のパターンが多く他のシステムで見られるという発見から始まりました。セングなどが、他にもそういったパターンがあるのではないかと研究を始め、彼の研究成果は、学習する組織（Fifth Disciplines）に記載されました。これと並行して、1990年代前半に Wolstenholme などが、セングの主張するものも実は4つのパターンにまとめられると主張し、共著で論文を発表しています。彼らの研究発表や著作に対し、こんなパターンもあるとSD研究者からいろんな例が寄せ集められ、主に Systems Thinker 誌上で議論が進み、増加を除いた12の型が Fifth Disciplines Filedbook で認識されています。

システム原型はこれ以上あるかも知れないし、これだけなのかも知れません。ダニエル・キムは、「応急措置の失敗」、「問題の転化」、「成功の限界」の3つが基本的なもので、その他は、この組み合わせか精緻化したものという主張を行っています。ただ、彼の主張は、極論すると、

- ・増加 (Reinforce Loop)
- ・均衡 (Balancing loop)

の2つが基本形で、全てのシステム原型はこの2つを複数組み合わせたものという話になりかねなく、私自身は、STで単純化するとそう思うかもしれないが、それは、何でも単純化すると、増加ループか均衡ループになるという主張と同類で、極度に単純化しすぎると、

意味が無くなるように思っています。

センゲは、システムを単純化して考える上で便利なツールと主張していますし、ダニエル・キムも、複雑な物事を単純化して考えることで方針や大局的な対策ができる、そのためにシステム原型に慣れておくことが重要と主張していて、彼らのこの主張には同感します。

1) 基本的な変化：

- ・増強 (Reinforce Loop)
- ・均衡 (Balancing loop)

2) 学習する組織に紹介されているシステム原型 The Fifth Discipline (Peter Senge)

- ・遅延を伴う均衡 (Balancing loop with delay)
- ・応急処置の失敗 (Fixes that fail)
- ・成長と投資不足 (Growth underinvestment)
- ・問題の転嫁 (Shifting the burden)
- ・成功には成功を (Success to the success)
- ・成長の限界 (Limit of growth)
- ・エスカレート (Escalation)
- ・目標のなし崩し (Drifting goals)
- ・共有化の悲劇 (Tragedy of commons)

3) Fifth Discipline Field Book に紹介されている先以外のシステム原型

- ・昨日の友は明日の敵 (Accidental adversaries)
- ・魅力の基本 (Attractiveness principle)

私自身は、作成した定性モデルの中から、こういった振る舞いをすると思われる部分を発見し、それがコアであると考えられる場合に、一旦、そのシステム原型にまでそぎ落としてしまうことで、コアの振る舞いを理解するといった使い方をしていきます。これを、トップへの説明のための定性モデルとしています。そして、そこから、それをコアにして、本当に意味のあるループだけを付け加えた SD モデルを再構築していきます。

ただ、一般的、あるいはどの状況でも使えるような SD モデルの原型のようなものがあるのかはよく分かりません。今の所、そんなものは SD に関しては無く、システム原型を使ってどう定性モデルが構築されようとも、定量モデルにしていく際には、個別に SD モデルを構築していくしかないのではないかと考えています。SD の場合、構造という意味ではいくつかパターンがあります。これについては、テキストの第 4 章に記載しましたので御参考下さい。

また、システム原型の振る舞いに関し、東洋大学の池田誠先生によると、ある期間のみに見られる現象で、最初から最後までその振る舞いになるとは限らないとのこと。日本では、池田誠先生、山口薫先生が、システム原型について SD モデルを発表していて、山口先生のは、ことわざとの比較で理解させるという工夫が行われています。山口先生のは、JFRC の HP から、池田先生のもも、先生の HP からダウンロードできますので、興味がある方はご参照下さい。

先にも述べたように、ST モデルでは同じになりますが、SD モデルにした場合、対象にした事項によってモデルが違ってくるので、傾向としてはそれに似たようなものにはなりませんが、山口先生や池田先生、あるいはダニエル・キムやピーター・センゲなどが著書で解説したような振る舞いのグラフと同じになるとは限りません。ここで、私が示した SD モデルでのシミュレーション結果が、彼らのものと形が違っていると驚かないで下さい。別に間違っているわけではありません。元になっている SD モデルが違うのです。

A) 基本的変化

基本的変化として、増加と均衡について最初に簡単に触れておきます。

1) 増強 (Reinforce Loop)

これは、グラフで時間に対し y が増加しているという意味ではなく、原因となる因子の増加に対し、結果となる因子が同じく増加するという意味で、原因となる因子が減少すれば、結果となる因子も減少することもあります。これが、ループになった場合、途中反対、例えば一に動く因子があっても、打ち消し合えば、全体として、主因因子の増加により、ループ全体としては、増加し、まわりまわって、主因因子が増加する、あるいは減少するという変化になります。

増強ループの例として、図 5-1 左のようなケースを考えてみましょう。赤字になってしまいう予算を、政府は負債分を国債発行で切り抜けます。しかし、発行した国債に対し、利子を支払わなければならない、そのための支払利子が増えます。その支払利子と元本返済のために、政府の負債は減らなく、その負債を補うためにさらに国債を発行します。こうして、発行国債の額と支払利子がどんどん増えていくというケースです。

「政府の負債」、「国債発行による負債の補填」、「国債に対する支払利子」のパラメータはそれぞれ+の関係なので、これは増強ループになります。

この定量モデルの例が例えば、図 5-1 右で、国債発行額を見るとどんどん増えています。

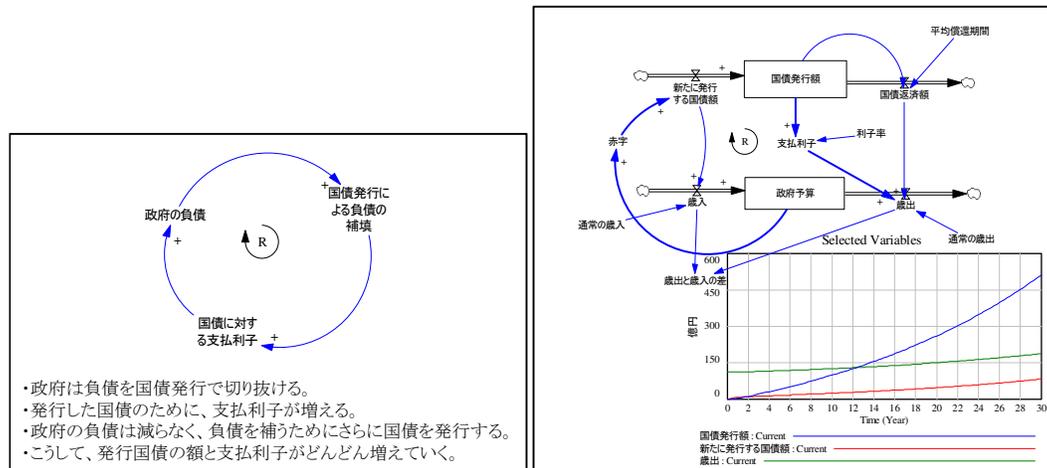


図 5-1 (左) : 国債の定性モデル、国債の定量モデル (右)

2) 均衡 (Balancing loop)

原因となる因子の増加に対し、結果となる因子が減少するようなものが奇数個ある場合を均衡ループとしています。均衡といっても、ループを構成する全ての因子の振る舞いが収束するという意味ではなく、部分や因子によっては増加することもあります。ループ全体では、均衡が保たれるので、増加してもどこかで減少し、あるいは減少しても、まわりまわって、いずれは増加しようとしません。その結果、一般的には、最終的にどこかに収束するような振る舞いとなるので、均衡ループと呼ばれています。

図 5-2 は、目標があり、その達成に向けて現状とのギャップを埋めるべく努力するというものです。例えば、冬で、室温が 10 度で、これを温かくし、20 度に保ちたいといった例を考えましょう。ストーブを付けて温かくなりますが、温かくなりすぎ、室温が 28 度にまで上昇します。こうなると暑いので、ストーブを止め、室温を下げます。今度は室温が下がり

過ぎ、15度まで下がって寒くなります。再びストーブを付け室温を上げといったことを行いながら、最終的に20度という快適な室温にしていきます。

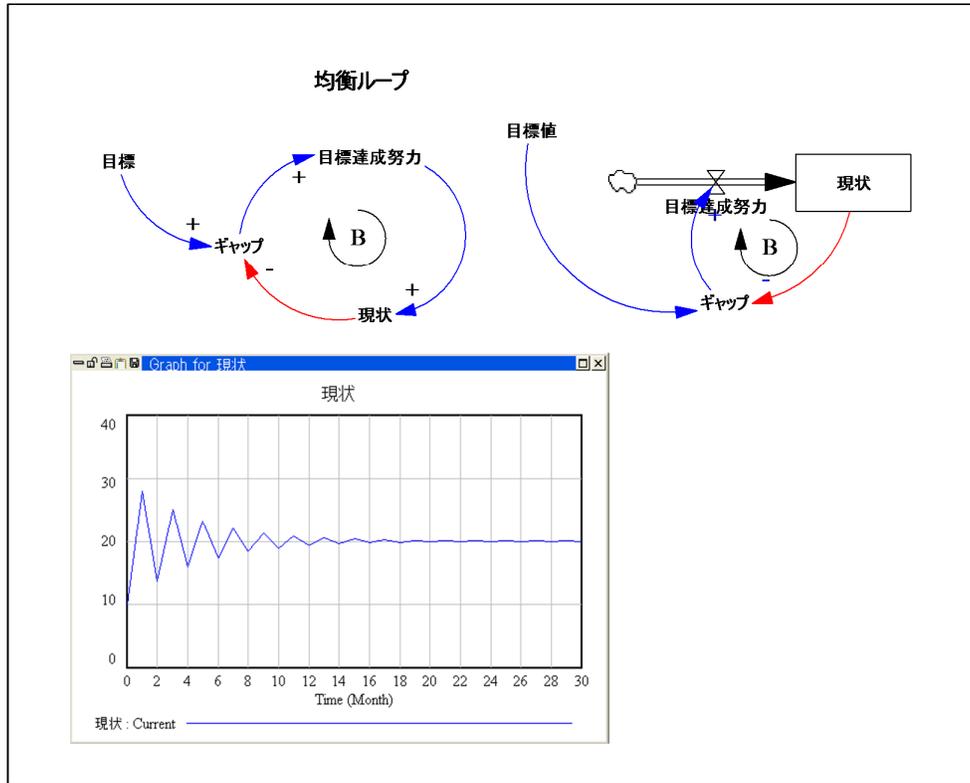


図 5-2：均衡ループ

B) 学習する組織で紹介されているシステム原型 The Fifth Discipline (Peter Senge)

3) 遅延を伴う均衡 (Balancing Loop with Delay)

通常は、遅れの誤差が積み重なり、最後には発散することが多いのですが、修正率が小さい場合は、むしろ目的探査型、従って成長曲線になり、収束します。

図 5-3 はこのような例で、今月額 100 万円の売上のある会社が、売上目標を 120 万円にしたいとします。そのための努力を必死にするのですが、認知が遅れ、実は目標を達成しているにも関わらず、達成していないと思って努力し、130 万円になってしまいました。その後、売上は落ちるのですが、目標を達成していると安心して努力を怠ってしまいます。遅れて、売上が落ちていることが認識され、再び努力します。こうした努力が、遅れにより加速され、どんどん月間の売上の振幅が大きくなっていきます。

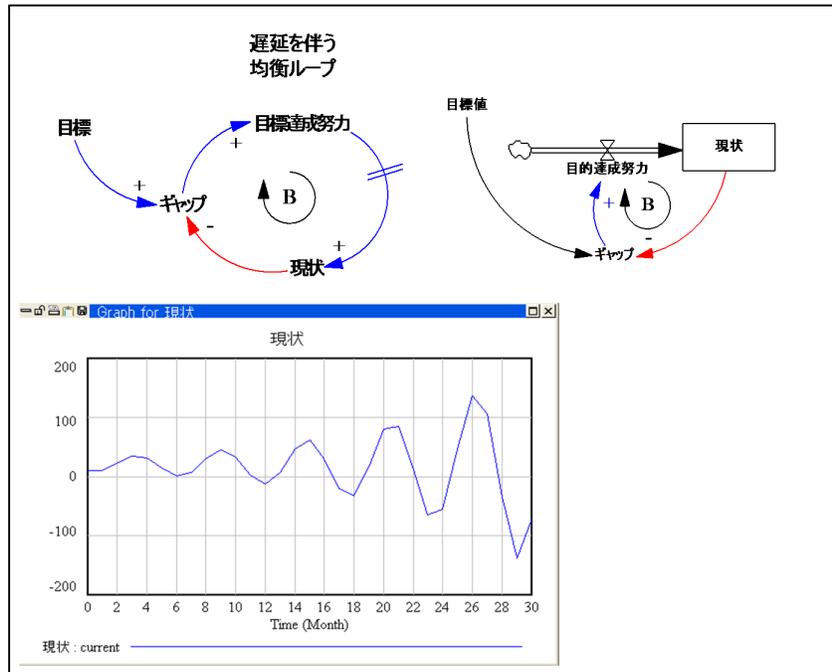


図 5-3：遅れを伴う均衡ループ

4) 応急処置の失敗 (Fixes that Fail)

問題に対して対処措置を講じるだけで根本的な措置を怠っているとやがて予期しなかった副作用が現れてくるというもので、遅延を伴う増加ループが均衡ループを包含しているような構造で起きます。

図 5-4 は、仕事のストレスをお酒で紛らわしているケースで、アルコールでストレスを紛らわせながら仕事をこなしていると、やがてアルコール中毒になってしまうというものです。

図 5-5 も同じく、応急処置の失敗の例を示しています。ある年、政府は予算が赤字になってしまいました。政府は負債を国債発行で切り抜けました。しかし、発行した国債に対する利子のために、また赤字になってしまいました。そこで、再び国債を発行し、切り抜けます。こうして支払利子が増え、政府の負債は減らなく、負債を補うためにさらに国債を発行し、発行国債の額と支払利子がどんどん増えていくという応急処置の失敗が起きます。図 5-6 左を見ると、ちゃんと、必要な歳入に見合う歳入が得られているので問題ないように見えます。歳出と歳入の差が一時的にマイナスになることもあります、概ねゼロなので、見かけ上の歳出と歳入には全く何も問題がありません。

しかし、実は、通常の歳入は 100 億円しかなく、30 年後には、国債の元本返済が 51 億、利子返済が 25 億とこれだけで通常の歳入の 76%の額になります。このことが図 5-6 右に示されています。この図 5-5 のモデル、どこかで見ませんでしたか？そう、前に増強ループを説明したモデルと全く同じなのです。

5) 生長と投資不足 (Growth Underinvestment)

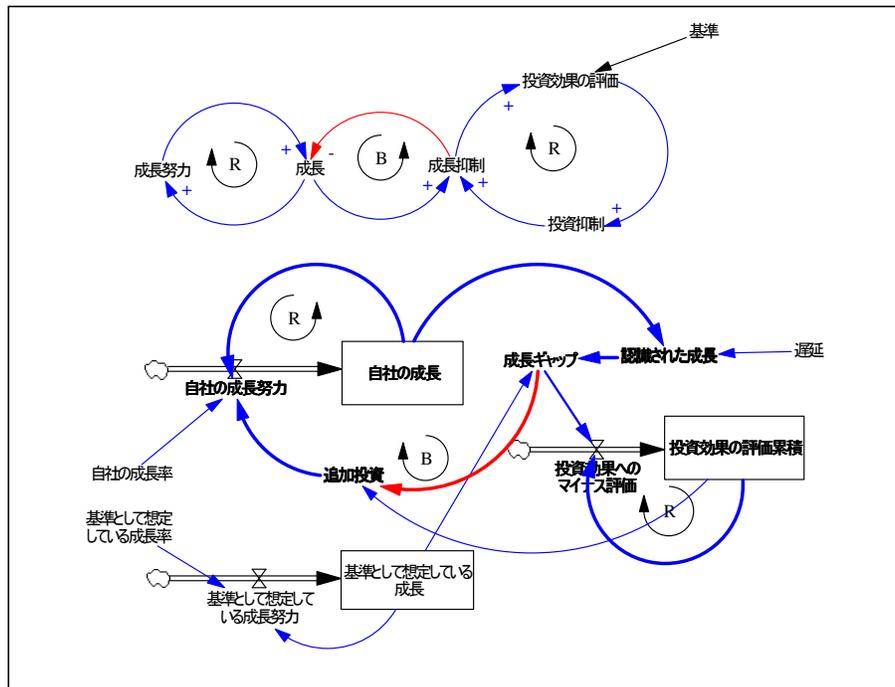


図 5-7：生長と投資不足

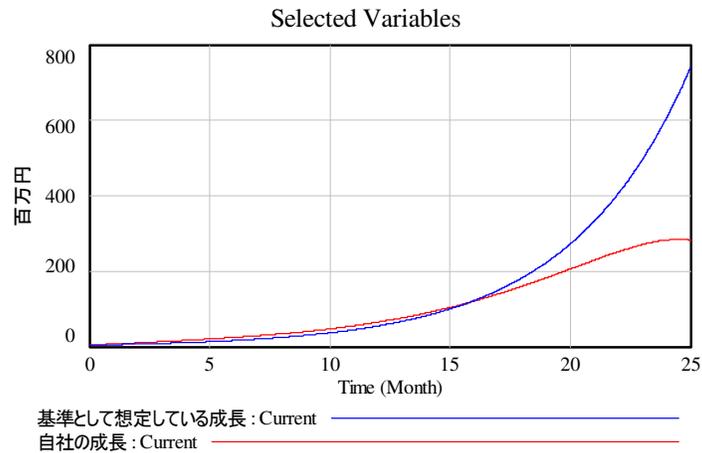


図 5-8：生長と投資不足

企業が成長を続けていくためには適切に経営資源の投入を継続する必要があるが、この適切な投入が行われないと、能力が限界に近づき、成長が停滞し、停滞した時点であわてて投入を行っても、その投入の成果が現れるには時間がかかります。その間に、競合相手に市場を抑えられたりすると成長がさらに低下し、やがて、投入効率性に関する疑念が起り、投入抑制が行われ、その結果さらに能力の低下が起こるといふもので、増加ループに均衡ループが近接している構造で起きます。

今、ある月額 500 百万円の売上有る会社を考えます。20%の成長目標が設定されていま

す。この成長目標達成のためには追加的に経営資源の投入を続けていくことが必要なのですが、経営者が得られる経営情報に 3 ケ月の遅延があるとします。遅れがあるので、目標と以前の実績のギャップで判断され、過少評価された値が投入されます。実は、この会社は順調に成長目標を達成していたのですが、過少評価された追加経営資源投入により、18 ケ月目あたりから成長に陰りが見られるようになります。

6) 問題の転嫁 (Shifting the burden)

問題に対してとりあえず対応的な措置を講じながら、同時に根本的な解決にも取り組むべきなのですが、対応的な措置が成功したので、根本的な問題に取り組むための方策や能力の開発意欲が失われ、問題への取り組み能力自体が開発されないというもので、2つの隣接する均衡ループを増強ループが取り囲む構造でよく見られる構造です。定性モデルを図 5-9 に示しています。

具体的にどうことで、図 5-10 のような、あるメーカーの新製品に対する事例を取り上げてみました。このメーカーでは、新製品のクレームが顧客から殺到したために、暫定的に製品についてよく知っている開発部門に対応をさせることにしました。その間に商品部できちんと問題対応できる能力を育成すべきだったのですが、開発部門の対応があまりにもうまくいっていたので、商品部でクレーム処理をする能力を早期に育成しなければならないという危機感が経営層に生まれませんでした。その間、開発部の商品開発の時間が問題対応に取られ、肝心の、次の新製品を開発することが十分できなく、競争相手の製品開発能力との間に大きな差ができてしまいます。16 ケ月までは、開発部のクレーム対応のおかげで問題のレベルが 5 を超えることがなく、このため商品部の問題対応能力を育成する努力が行われません。その間、開発部の新製品開発能力は停滞し、ついにはクレーム対応のために全く新製品開発努力に時間が避けなくなります。この問題の片がつくのは、競合他社との製品競争力の差が 22 ケ月を過ぎてから決定的になり、顧客からはもう自社製品が試みられなくなり、クレームが激減し、もう商品部にとっても開発部にとってもクレーム対応という行為に意味がなくなってからのことです。

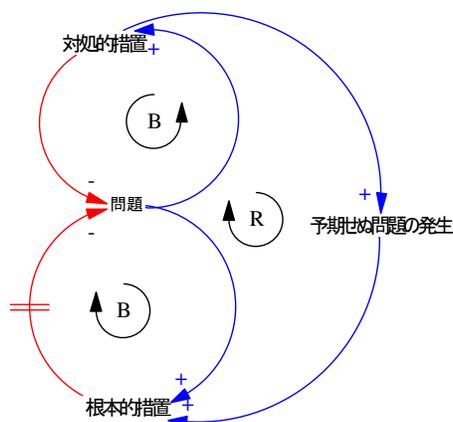


図 5-9：問題の転嫁 (定性モデル)

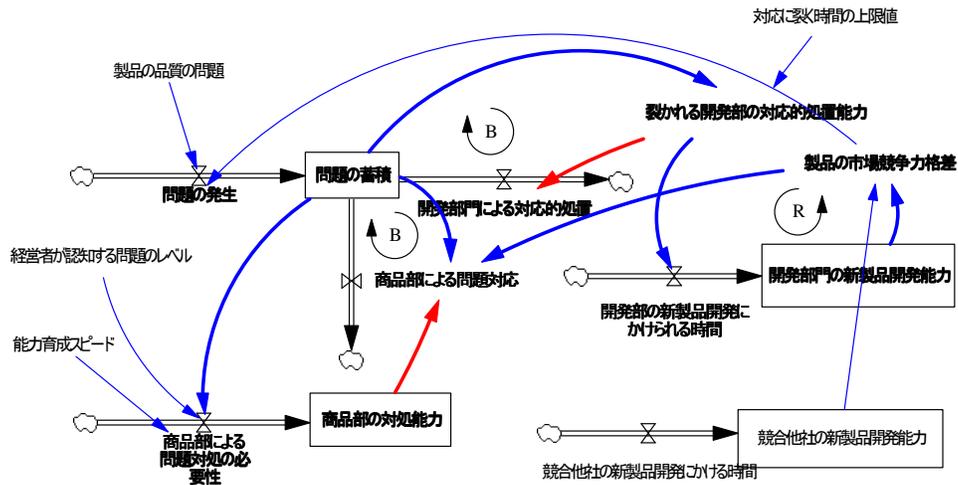


図 5-10：問題の転嫁 (定量モデル)

この結果を、図 5-11 の「問題の認識低感度」で示しています。先の問題に対して、うまくいった例としての 2 番目のシミュレーションとして、経営層の問題認識のレベルを上げ、早急に開発部のクレーム処理を商品部が代替できるようにするにしましたもので、開発部のクレーム対応も、それに避ける時間の 50% を上限にすることで、競合相手に大きく製品開発能力の差をつけられないようにしています。この結果は、「問題の認識高感度」で示しています。

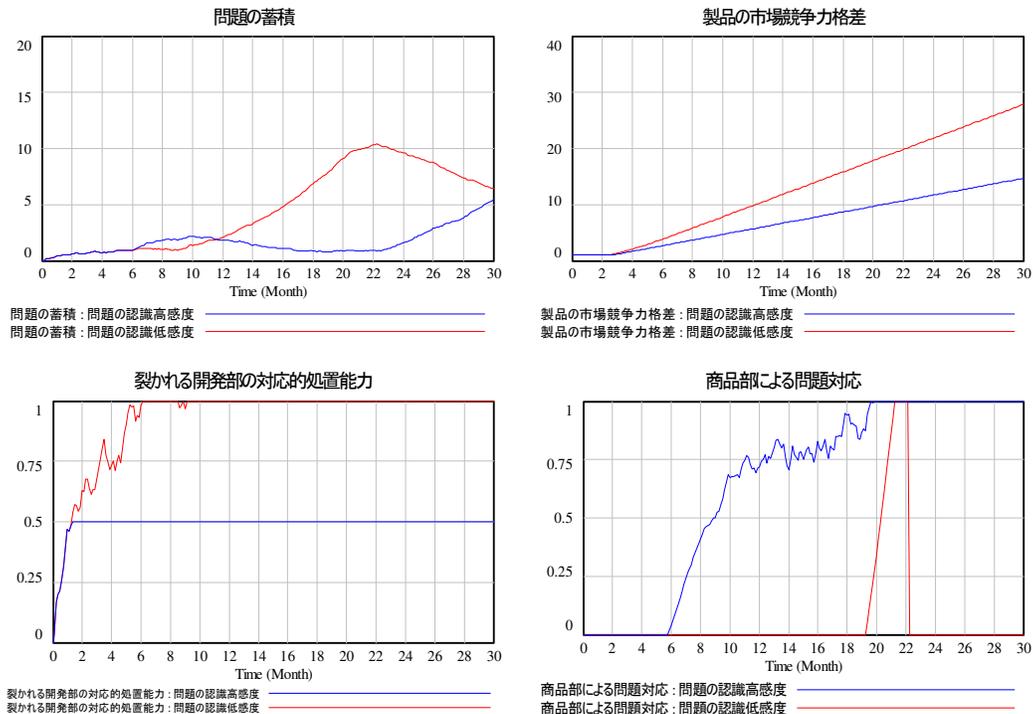


図 5-11：問題の転嫁 (シミュレーション)

こういった、問題の転換での失敗例は、いくつも見ることができます。例えば、日本のカメラ・メーカーは、戦前、戦中に培われた軍需産業での精密機械技術と工学技術を適用し、戦後、汎用カメラ市場で成功します。しかし、その成功も、2つの技術革新の波により、戦後、成功したカメラ・メーカーのほとんどは倒産、あるいは買収されてしまい、古くから残っているのは、ニコンなどのメーカー数社にしかすぎません。第一の波は、メカトロニクスという、機械だけではなく、電気・電子を使った駆動技術、制御技術をカメラに組み込むという動きの形で押し寄せます。オートフォーカスなど、モーター化、メカトロニクス化が行われますが、メカニックからメカトロニクス（機械技術+電子技術）に変化できなかった企業の多くは、ここで倒産していきます。ニコンは三菱グループだったので、三菱電機の技術支援を得て生き残ることに成功することができました。この時は、バカチョンで全焦点にするとか、機械化をもっと押し進めることでも対応でき、そのため、メカトロニクス化をやらなくとも、何とか生き延びられた企業もありましたし、まだフィルムを使い続けるプロ・カメラマンやセミプロ・カメラマン、愛好家もいましたので、プロ用製品に特化していくことで生き延びたメーカーもいました。

しかし、15年も経たないうちに、コンピュータ化という形で第2の波が押し寄せてきます。この変化は、メカトロニクスよりも、むしろカメラがコンピュータになったという変化で、フィルムもデジタルになってしまいます。ソニー、松下、カシオなどの、もともと電気・電子機器メーカーがカメラ市場に進出し、ドイツのツァイスなどのレンズ・メーカーと組んで高級カメラを作るソニーや、コピー機で成功し、高級カメラ市場に乗り出し成功するキャノンのようなメーカーも現れます。ここでも、機械式で、高級志向で生き延びようとしたメーカーもありましたが、プロですら、フィルムではなくデジタルに切り替えるようになり、この戦略はあまり成功していません。

当時活躍した技術者に話を聞くと、変化は何となく理解していたといいます。ただ、変化が激しすぎたこと、また、一番大きな原因は、機械屋のカルチャーが強すぎて、電子屋を技術開発に入れられなかったのだといいます。また、入れても、機械屋の雰囲気が強すぎて、すぐに辞められてしまったということで、話を聞いていて、この会社が成功できなかったのは、技術の問題ではなく、組織の問題というように感じました。

経営学では、過去の成功体験を忘れることの重要性が強調されています。成功してしまうと、問題に直面した際に、どうしても、その成功した方法で同じように問題に取り組もうとしてしまいます。毎回、白紙（ゼロ・ベース）で、情報収集→分析→ソリューションという方法を取らなく、情報収集でも、自社に都合のいい情報にしか聞く耳を持たないようになり、いきなり決めつけのソリューションを取りがちですし、ワンパターンの対処方法になってしまいがちです。これは特に競争戦略策定では致命的なことです。過去の成功体験に引きずられない常にチャレンジな対応が、技術革新が激しい競争市場を生き延びる上で必要なことです。

7) 成功には成功を(Success to the success)

これは、当初はほとんど同じ条件の競合者が、途中のちょっとした条件や結果の違いで、一方はどんどん成功するが他方はあまり成功できなくなるというものです。「あいつと俺とは同期入社なんだが、あいつだけ偉くなって、、、」なんていうせりふを、どこかで聞いたことがないですか？

このパターンは、2つの増加ループが接している構造の場合によく見られます。図 5-12 に定性モデルを示しました。また、図 5-13 に定量モデルとして、2つの支店、A支店とB支店の収益の差を示しました。支店・本店からの総利益の一割が、各支店に、売上に応じた、営業支援費という形で支給（再配分）されています。最初の10ヶ月目までは、A支店にもB支店にも均等に本店から約1割の営業開発費が支給されていて、順調に収益が上がっています。11ヶ月目に、あるきっかけで、この営業開発費がA支店に与えられ、B支店には与えられなくなります。収益は営業にかけられる開発費で決まるようになっているの

で、少しずつ、しかし指数的に差がつき、A支店の収益は指数的に上昇するのに対し、B支店の収益は指数的に悪化していきます。この差は、22ヶ月目まではそれほど大きく目立つ差ではありませんが、その後は大きく違ってきます。

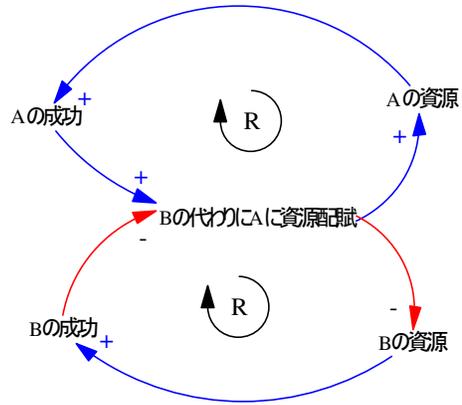


図 5-12：成功には成功を（定性モデル）

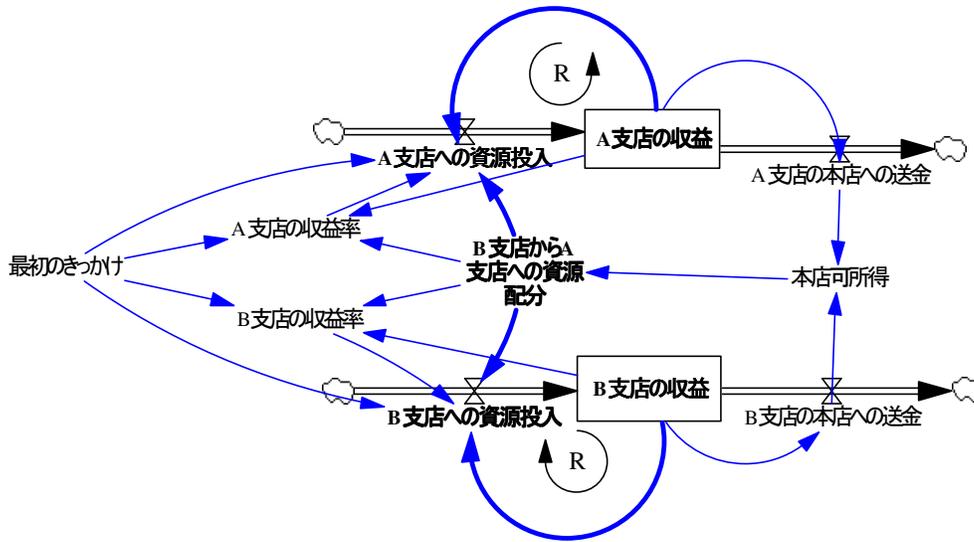


図 5-13：成功には成功を（定量モデル）

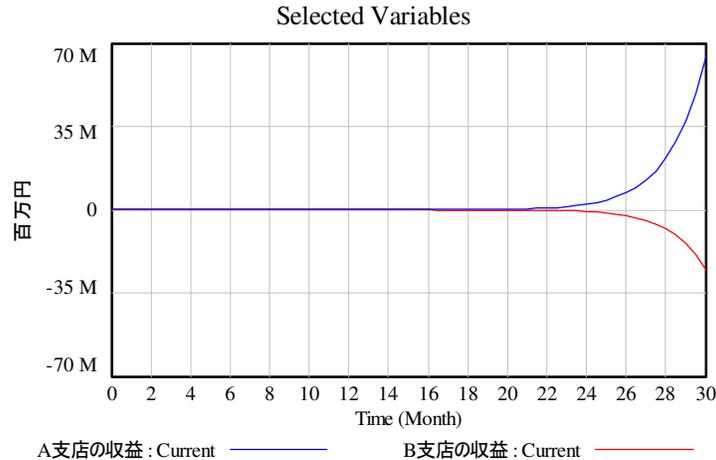


図 5-14 : 成功には成功を (シミュレーション)

こういったパターンは良く見られるもので、例えば、古代ローマ帝国の歴史では、ローマとカルタゴの争いが続きます。しかし、当初は、ローマは負け続け、有名なカルタゴの将軍ハンニバルが、アルプスを越え、ローマに攻め込んだ時などは、ローマは崩壊直前にまで陥ります。ローマは、こういった危機的な状況に何度も陥ります。ただ、違いは、ハンニバルは一度負けたとたんにカルタゴから支援を受けられなくなり、罪を問われますが、カルタゴと戦ったローマの将軍たちは数度負けても、以前にもまして全面的にローマから支援を受け続けられ、何とかカルタゴとの戦いを続けることができました。やがて、ローマはアフリカヌスの基に、スペインを占領し、カルタゴの補給路を断つ戦略でついにカルタゴを滅ぼします。一度の失敗で支援を与えなくなるか、それとも失敗しても支援を与え続けたかで、ローマの歴史は変わってきました。

IBM がパソコン市場に参入する際の話も有名です。IBM は従来であれば、全て自社開発で事業を行っていたのですが、この時は、開発部隊が IBM のプロパーではなかったこと、早急に PC 事業を立ち上げたいと考えていたので、同社のパソコン製品のための OS (オペレーション・システム) を外部調達することになり、適切な製品を探していました。当時、2つの候補、マイクロソフトの DOS とデジタル・リサーチ社の CP/M が挙がっていました。IBM の担当者がデジタル・リサーチ社の社長のゲイリー・キルドールを商談のために訪問した際、たまたまゲイリー・キルドールは、彼が趣味としている自家用の飛行機の操縦に夢中で留守で、IBM は商談を行うことができませんでした。

一方、IBM の担当者がマイクロソフト社を訪問した際は、社長のビル・ゲイツ以下社員が総出で出迎え、とんとん拍子で商談が成立してしまいます。実は、この時、マイクロソフトの DOS は完成していなく、受け入れられる状態ではなかったのですが、副社長の西和彦氏が、強固に主張し、マイクロソフトは、より完成度の高い製品を外部調達し、それに不足部分を自社開発し、急遽製品として完成させ間に合わせたそうです。IBM の戦略で、DOS を通じて、いろんな PC 周辺機器メーカーが IBM の PC に接続できるようにしたので、IBM の PC 製品も売れましたが、周辺機器メーカーや互換 PC を製造するメーカーも成長することができました。いずれも、マイクロソフトの DOS を使わないと稼働しないので、マイクロソフト自身が一番儲かり、さらには、PC の OS 市場を独占し続けることができました。

西洋の諺に、幸福の女神には後ろ髪がないので、女神が訪れたら前髪をしっかりとつかん

で離さないことだと言われています。チャンスは 2 度とないと思って、チャンスだと思った時には思い切ってチャレンジすることが成功の秘訣のようです。

8) 成長の限界 (Limit of growth)

デニス・メドーズらがモデル化した「成長の限界」がオリジナルで、社会・経済成長のためにいろんな活動を行い、成長は成功したかに見えるのですが、全体的な資源の制約のためにある所で成長は限界を迎えるというものです。

増加ループに均衡ループが接続している、あるいは増加ループを均衡ループが包み込んでいる構造でこの現象が見られます。定性モデルは図 5-15 のようになります。

図 5-15 に示した定量モデルは、AOL (America Online LLC) のケースを想定したものです。AOL は 1983 年に創立したオンライン会社で、1990 年代初頭に、大々的なキャンペーンを開始しました。このキャンペーンの一環で、値下げ、接続支援のための CD-ROM の無料配布などを行い、この CD-ROM 無料配布などで急速に顧客を増やすことに成功しました。しかし、同社の機械設備の性能増強が顧客の急激な増加による接続需要に追いつけなく、顧客としては、顧客になったもののなかなか AOL のオンラインに接続できないという不満が高じて、辞める顧客が続出し、さらには同社に対する不評も高まることになり、同社が想定していたようには顧客を増やすことができなかつたものです。

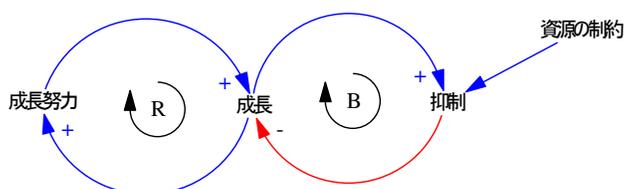


図 5-15 : 成長の限界(定性モデル)

モデルでは、「高い失望率」として、60ヶ月目に大々的なマーケティング活動を行い、それまでの 10 倍に顧客が増えます。こうして顧客が一気に増えるものの、同社のサービスに不満を持つ顧客の率を減らすことができなく、顧客数がサチュレートするという代表的な普及モデルにしています。一方、比較のために、「低い失望率」として、顧客の不満が高まらなく、安定しているようにします。ここでは、失望率が高いか低いかだけの単純化したモデルで、顧客の増え方によって失望率が変化するようににはしていません。また、普及モデルなので、どこかでいずれはサチュレートするのですが、潜在顧客を大きくしているのので、対象期間の範囲では、潜在顧客が理由でサチュレートするようにはモデルを作っていない。

経営では、成長の限界はニッチの問題として捉えられることが多いように思えます。うまいニッチ市場を発見して成功しても、ニッチという市場の限界によって成長が止まってしまう。次にどのニッチを見つけ、成功させるかが企業成長のカギになると言われています。ただ、こういった成長の限界は、ニッチ市場という市場の限界だけではなく、人材調達や人材育成の限界、資金調達の限界など、人、物、金といった経営資源の限界によっても引き起こされ得るものです。また、革新技術や発想の転換によって、ニッチの限界と思われていたものが払拭されるということもあります。例えば、カメラが機械ということだけに捉われていたのであれば、カメラという機械やカメラ製品市場での市場占有競争

で終わってしまいますが、画像記録という機能で捉えれば、PC市場やモバイル市場もあるし、画像処理市場もあると、市場が広がっていきます。

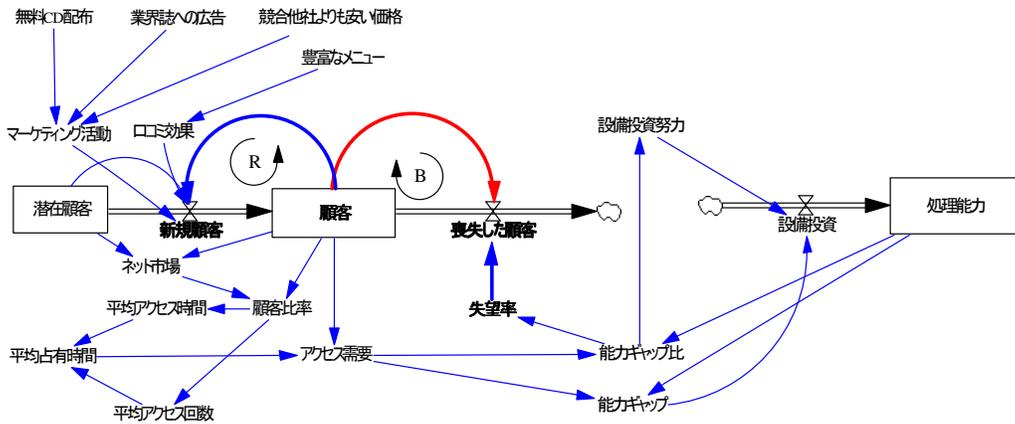


図 5-16：成長の限界(定量モデル)

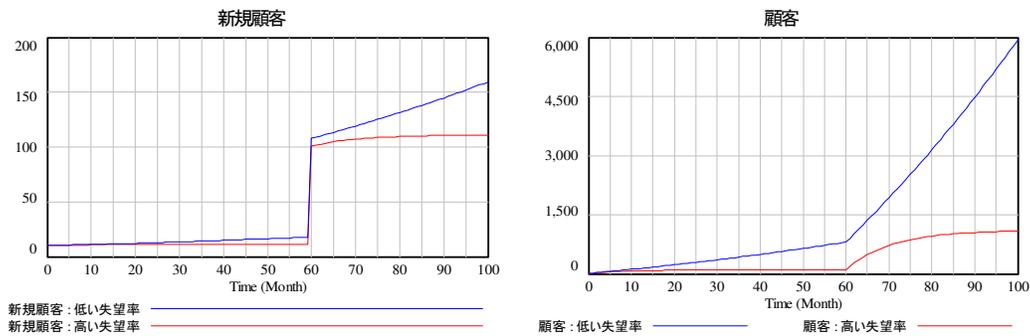


図 5-17：成長の限界(シミュレーション)

9) エスカレート (Escalation)

ライバル同士が争っていると、他方の行為に触発され行為を挑戦的に高め、それが相手を挑戦させ、この結果どんどん緊張が高まっていくというもので、冷戦時代の米ソの軍備拡張がよく、このエスカレーションの説明で取り上げられます。図 5-19 にこの定量モデルを示していますが、ソビエト連邦の軍事力の脅威から、米国は核武装も含む軍事強化を増加させ、米国の財務負担になっていく。一方、米国の軍事強化はソビエト連邦にも恐怖を与え、ソビエト連邦も無理な軍事予算増強を行う。どちらも財務的に苦しい状況に陥るが、恐怖ゆえに軍事強化レースをストップさせることができなかったというもので、2つの均衡ループが接している構造の場合にこの現象が見られます。

モデルでは、軍事力格差が2倍以下の場合は軍事予算を増やさないが、軍事格差が2倍以上になった場合、脅威を認識し、軍事予算を増やし、相手の軍事力に追いつこうとします。アメリカは議会承認で時間がかかるので3年で予算承認し、ソビエトは党独裁なので1年で予算承認するとしています。

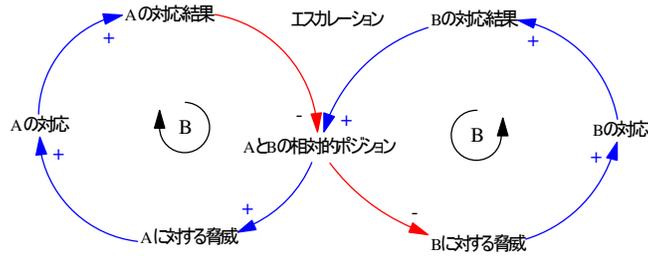


図 5-18 : エスカレート(定性モデル)

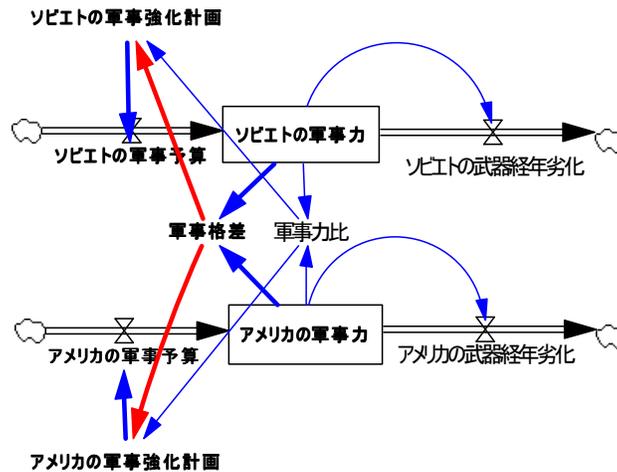


図 5-19 : エスカレート(定量モデル)

このようなエスカレートの例として、少し古い話ですが、スーパー業界でのダイエーと西友の争いが挙げられます。ダイエーがある地域に出店すると西友もすぐに近くに出店し、ダイエーと西友という 2 大スーパーマーケットが規模拡大を争いました。結局ダイエーは無理な拡張が原因で、経済不況に陥った時には経営破綻し、西友ストアーも、その後、経営拡大戦略から収益性を重視する戦略に変わり、採算性の悪い店舗はどんどん閉鎖するようになってしまいました。ダイエー、西友だけではなく、長崎屋など、多くのスーパーが出店競争を行いました。生き残ったのはサミットなどだけで、多くはかつてのようなエスカレート状態の出店競争はもうできなくなっています。無限に顧客を獲得でき、収益性も十分確保できる条件下ではエスカレーションにより競争を続けていくことに戦略的に意味があるのですが、多くは成長の限界という制約によって、いずれはエスカレートは終焉してしまいます。冷戦の米ソの軍事レースは国家経済の成長率に、ダイエーと西友の競争の場合、中級購買層という顧客の限界があるので、成長の限界のしくみが働き、顧客も収益性もサチュレーションを引き起こしてしまいました。永遠に競争を激化させ続けることはできません。

Selected Variables

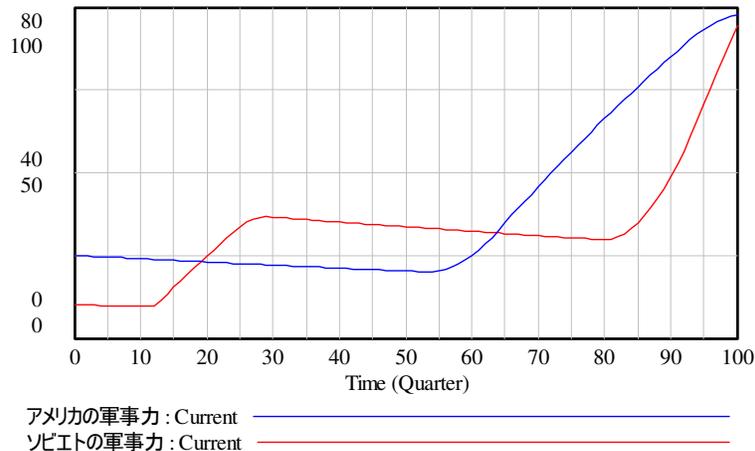


図 5-20 : エスカレート(シミュレーション)

これは私自身が従事した最近の ODA プロジェクトの例です。アフリカのケニアにあるザンジバル政府は、社会主義的な政策を行っていて、水道は無料でした。しかし、州都が世界遺産になり、美しい砂浜などを生かした観光産業開発の成功もあり、社会経済が発展し、人口が増加しましたが、人口増加に見合うようには十分に給水できなくなりました。ザンジバル政府は、十分に給水できない理由を、施設の老朽化と考え、施設の改善と、水源の拡張のための工事に、日本の支援を求めてきました。日本政府は、無償で、施設の修復と水源の拡張工事を行いました。利用者負担の原則から、その条件として、せめて、運営維持費の半分ぐらいは水道料金収入で経営して下さいということをもとめました。そこで、水道料金を徴収しようということになり、それまでは政府の部局(水道局)だった組織を、水道公社という形で独立させ、水道料金徴収の法律も制定し、水道料金を徴収することになりました。日本政府は、この料金徴収のための支援として、水道料金請求のための電算システム、電算機、プリンター、会計ソフト、人事ソフト、水道メータを提供し、公社経営のための職員訓練を実施しました。

水道は無料だったので、水道公社は顧客住所を把握していませんでした。そこで、水道料金を徴収するために、住民集会を開き、そこでアンケート調査を行い、アンケートに記載された顧客名と顧客住所をデータベースに登録し、それをベースに請求書を配達しようとした。ところが、請求書を印刷し、配布する段階になって、記載されている住所がいかげん、氏名もいかげんで、請求書をほとんど配達できないということが判明しました。配ることができない水道料金請求書で会議室が埋まってしまうほどでした。請求書が配達されず、水道料金収入がないので、当然財務は悪化します。水道公社は、困って、顧客をアトランダムに訪問し、水道料金の領収書を見せろという調査を行いました。顧客が水道料金の領収書を見せられないとその場で給水停止工事を行います。顧客はあわてて、その時だけ水道料金を払います。払ったことがある人の領収書をこっそり廻してもらってごまかした人もいます。あるいは、給水停止になっても、数日後、こっそり再接続してしまいます。少し金をもらって、再接続工事を行う人間もいます。ただ、こういったことをやられると、きちんと給水されなく、給水されない日があったり、水圧が低かったりといったサービスが悪いことも加わり、顧客は絶対水道料金なんか払うものかという気になっています。周囲が払わないので、もちろん当人も払いたくないという気持ちもあります。

- 500 個の水道メータを無償提供し、接続した。
- 全部で約 1,000 世帯あり、800 世帯が顧客。残り 300 個は水道公社がメータを供出してくれた。
- この地域では、検針し、検針結果に基づき請求書を発行し、請求書を配達。
 - ・顧客が水道料金を払わない場合は、督促状を出す。
 - ・それでも払わない場合は、給水停止。
 - ・毎月、違法接続していないか確認する。
 - ・給水停止になった場合、再接続を申請すれば、登録料、罰則金、遅滞金を取って、再接続する。
 - ・水が来ないという顧客のクレームは、本当か確認する。
- 給水時間に訪問し確認
- 近所の顧客がクレームしていないことも併せて確認
 - ・水道メータ設置を進めていく
- そのためのリボリングファンドを設け、水道メータ販売収入を全額、ここに充当する。
 - ・こうして、水が来ないから払わないというクレームを無くす。
 - ・確かに毎日水はこないかも知れないが、全く来ないということは無い。
 - ・水道料金が高いかどうかは、別の話だが、議会で認められた（公共）料金だということで、水道料金を、法律（官報のコピー）を使って説明。

水道メータも付いているし、改善の一番大きな原因として、きちんと顧客の住所が把握できていて、毎月請求書が配達され、顧客が水道料金を払ったかどうかきちんと把握できるようなしくみを確立しています。この時はうまくいき、この地域からの料金回収率は 80% を維持しています。しかし、うまくいったのは、このパイロット地区のみで、水道公社がこのやり方を他の地域にも拡大してくれることを希望していたのですが、予算不足を理由に実施してくれませんでした。その後、再度、日本政府の財務支援で、顧客調査を外注方式で実施し、このやり方を残りの地域にも広めようとしています。



図 5-23 : 航空写真 (左) とそこから起こした地図、顧客住所の番号付け (右)

10) 目標のなし崩し (Drifting goals)

ある目標を掲げたものの、達成できそうもないので少し目標を下げる、しかし努力してもやはり達成しそうもないので、また少し目標を下げる、と目標がなし崩しになる状況の構造です。均衡ループが接していて、一方の均衡ループが他方をさらに均衡させようとする構造でよくこういった現象が見られます。「問題の転化」に構造が良く似ていますが、増加ループがないので、均衡ループに歯止めがかかりにくくなっていて、こういった現象が発生します。図 5-24 にその定性モデルを示しています。

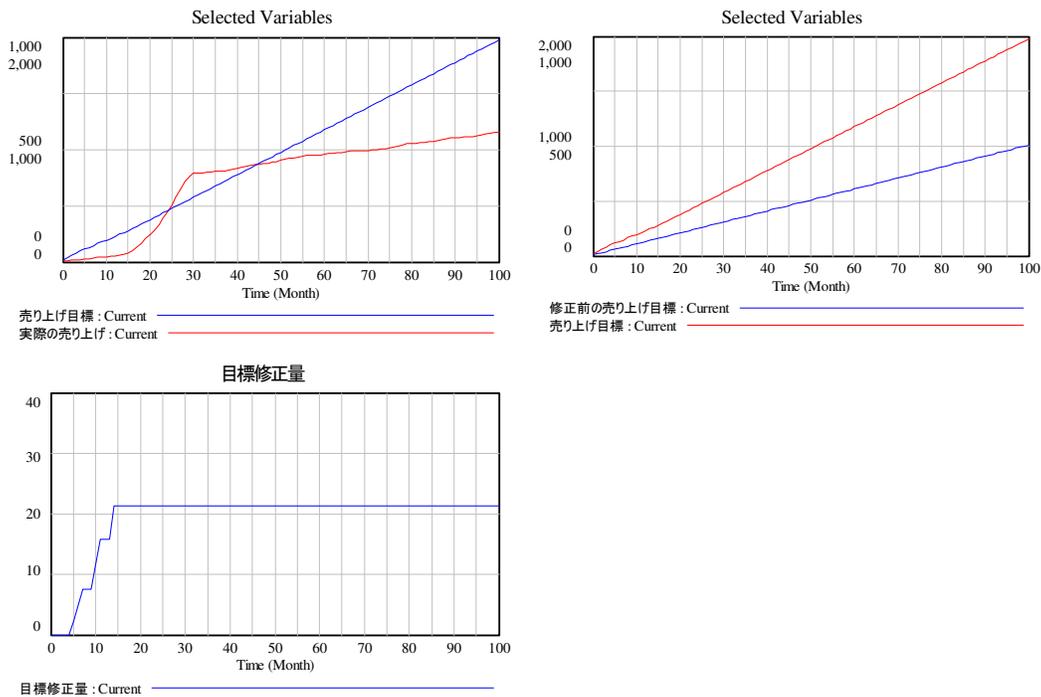


図 5-26：目標のなし崩し（シミュレーション）

適切に能力を把握し、適切に目標を設定することが一番重要ではありますが、適切に目標が設定されているのであれば、やはり初心貫徹のつもりで頑張ることが重要です。安易な目標設定、安易な目標変更は考え物です。私が昔、コンサルタントをやっていた頃に経験したことです。経営目標の設定がいいかげんな会社が多いにことにあせんとさせられました。あまり十分な市場分析とか自社能力分析をやっていなく、前年比 10%増といった経営目標が立てられます。こんなやり方では、問題が生じた場合、すぐに目標を下方修正せざるを得なく、この目標のなし崩し状態に容易に陥ることになります。

11) 共有化の悲劇 (Tragedy of commons)

資源を共有している場合に、ある利用者が、資源保全を考えない勝手な消費を図るととたんに資源が危機的な状況に陥るといふもので、増加ループが制約条件に影響される均衡ループを共有する構造、つまり増加ループを抑制する均衡ループの制約条件が共有される構造、いわば成長の限界が接して、成長の限界をもたらず制約条件を共有するような構造で見られる現象です。環境問題とか、生態系はほとんどこのパターンです。以前実施した、ST コースでの、Fish Banks の演習でお馴染みと思います。

この共有地の悲劇という名前は、Garrett Hardin が米国の科学雑誌、Science の 1968 年 12 月 13 日号に投稿した同名の論文によるもので、Hardin の論文は、核競争を抑制するような有効な技術はないという趣旨の論文を読んだことに触れ、技術開発がソリューションになるというのは幻想であること、生物学的な性向から人間の未来はないのではないかということを中心し、その例として、ある個人や団体が増殖行為を行い、それが持続性を破壊することを挙げ、解決は、このような性向を理解した上での、人間の良心的な行為のみにしか期待できないという趣旨のことを訴えたものです。そして、Hardin が挙げたサヘル地域における遊牧のような現象に対してこの名前が付けられるようになりました。

限界を持つ共有資源にだれもがアクセスできるようなオープンな場合に、Hardin が挙げたこの共有地の悲劇という現象が容易に発生することがその後の研究で知られて、今では、

解決策として、資源にアクセスができる者の限定やアクセスに制限を加え、クローズにすることでこの現象を防ぐことが行われています。

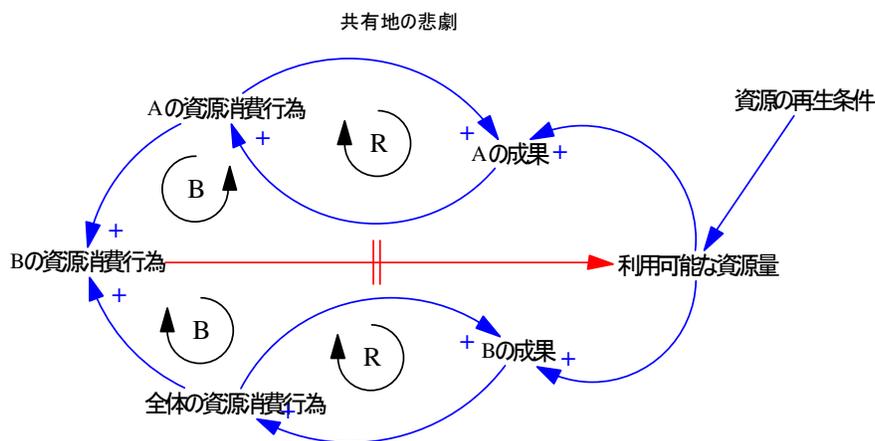


図 5-27：共有化の悲劇（定性モデル）

図 5-28 と 5-29 として 2 つの定量モデルを示しました。これは、アフリカ、サヘル地帯での、生態回復能力を無視した勝手な家畜の増加が遊牧民間で家畜増加競争をもたらし、それが原因で、家畜が草を食べつくし、乾燥化が進展し、砂漠化が始まり、砂漠が広がるといふ現象を示したものです。この地域は、サバンナ地帯で、降雨量が少なく、表土も薄く、それを草が守っている状態です。家畜が増える、特に羊は草の根まで食べつくすので、草が再生できなく、表土がむき出しになり、豪雨などが発生すると、簡単に表土が流されてしまいます。草が育つ土壌が失われてしまうと、後には石ころだけの土地が残され、こうして砂漠が広がっていきます。また、降雨量もますます少なくなっていきます。これを図 5-28 として示し、遊牧民に家畜を殖やしたいという願望があり、その願望に沿って家畜を殖やした場合をモデルにしています。

一方、図 5-29 には、遊牧民が昔ながらの持続性を考え、家畜を増やさない行為を守った場合をモデル化し、先のものと比較しています。2 つのモデルは基本的には同じで、違いは、図 5-29 のモデルでは、図 5-28 での理想としている家畜増加を計算する部分を除去し、家畜数の初期値を持続可能な数に抑えたことで、共に持続可能限界以下の数に家畜数を初期値として設定しています。それでも、自分勝手な増殖を行うととたんに環境が破壊され、家畜を養う草原が失われ、結局は家畜の数を減らさざるを得なくなります。

少し細かいのですが、草の量が不足し、栄養不足になり、その度合いが 0.5 以下であれば家畜は繁殖が不可能になり、0.3 以下になった場合、栄養失調で 3 ヶ月以内に死んでしまう、ただし、栄養度が 0.5 以上あれば、家畜は 1 年で子供を生み繁殖するとしています。また、家畜は 5 年生きるのですが、自家消費（肉）や販売などの消耗を考え、4 年にし、草の再生は 1 年の中で雨季のみで、雨季と乾季は 1 年に 1 度づつとしています。

図 5-30 にこの 2 つのモデルのシミュレーション結果を示しています。上は共有地の悲劇が起きた場合で、草が 15 年で種滅してしまいます。（済みません、単位を初期条件のままにしているので、「月」となっていますが、「半年」のつもりです。）一方、家畜の数を持続可能な数で抑えた場合は、草は次の年の雨季に再生し、無くなることはありません。

サヘル地域の砂漠化の原因として、地球温暖化による降水量の減少を強固に主張する学者いて、本当の理由ということではいろいろもめています。現地を訪れてみると、数年で砂漠が集落直前にまで押し寄せるといふ悲惨な状況です。こういった砂漠化の問題は、

サヘル地域だけではなく、いろいろな地域で起きていることは残念なことです。

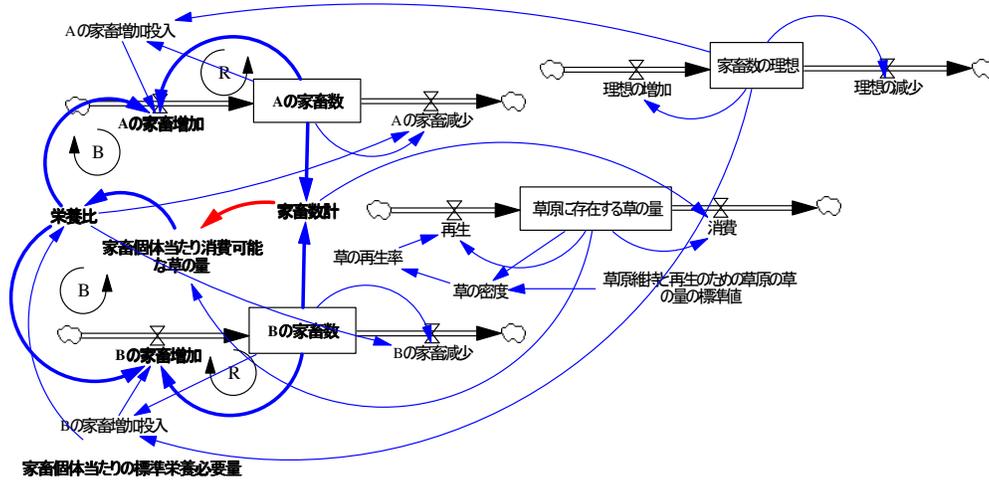


図 5-28：共有化の悲劇（定量モデル）

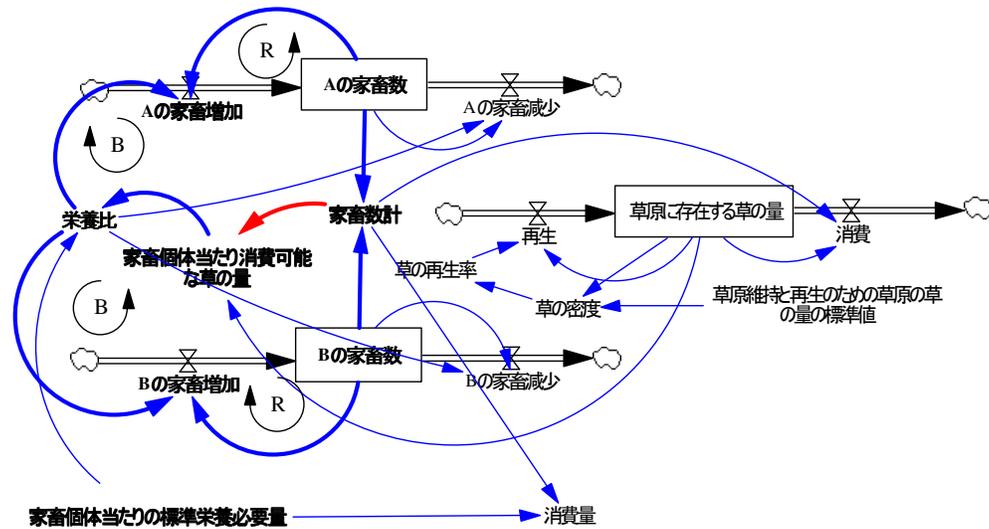
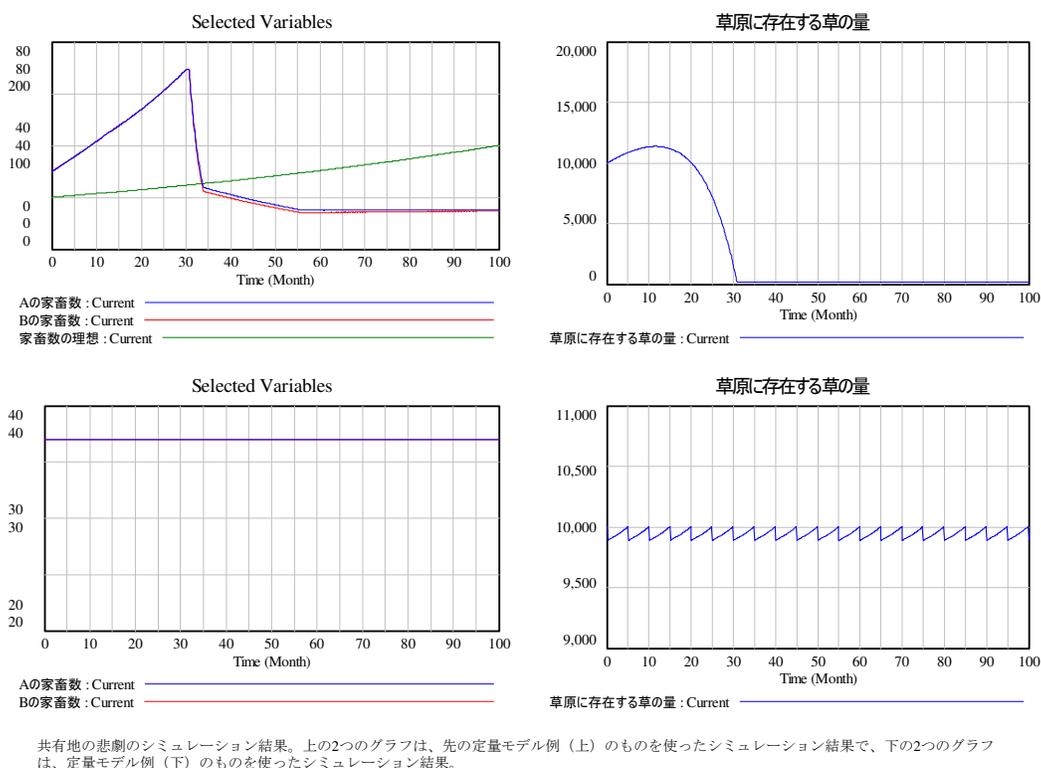


図 5-29：共有化の悲劇のない状態（定量モデル）

共有地の悲劇の例として、北海のタラ、日本海の花ハタ、ニシンなど、かつては無尽蔵と考えられ、制限なく漁獲されていた魚が、ある時突然捕れなくなるという現象がよく挙げられます。最近では、資源を有る程度保護し、魚が育成できる時間を与えることで持続的に漁業を営むという考え方が定着し、漁獲量や漁獲時期の制限を設けるクォーターが漁民に理解されていますが、初期には、クォーター破りの漁民が出て、なかなか漁獲量が回復できなかったこともあったようです。

また、旧ソビエト連邦時代、カスピ海沿岸のウクライナでは、カスピ海に流れる河川の水を利用して灌漑を図り、穀物や綿を生産しようとしてきました。確かに豊かな穀倉地帯を開拓できたのですが、その後徐々にカスピ海は干上がり、乾燥した湖底に積もった塩が風で

飛び散るといふ塩害をもたらすようになりました。さらには、灌漑を試みた地域も、乾燥によって地中にあった塩分が地表に出てきて、塩害などに悩まされるようになり、耕作ができなくなった土地も出ています。生態系に関する知識が不十分で、モデリング技術が発達していなかった当時、農業生産で西洋諸国に追いつくために、生態系のことなどを無視した無謀な農業開発計画を策定し、強引に開拓を押し進めていったため、河川の水が灌漑に収奪され、この収奪が旧来の生態系を破壊し、環境破壊をもたらし、砂漠化をもたらし、もともとの農業振興もうまくいかなかったものです。



共有地の悲劇のシミュレーション結果。上の2つのグラフは、先の定量モデル例（上）のものを使ったシミュレーション結果で、下の2つのグラフは、定量モデル例（下）のものを使ったシミュレーション結果。

図 5-30：共有化の悲劇（シミュレーション）

中国の新疆ウイグル自治区、内モンゴル自治区の例では、この地域の住民は、もともとは遊牧生活でした。中国共産党の政策で、遊牧民を農耕化させようとし、灌漑で農地耕作させようとしています。しかし、先住民は耕作農業に不慣れなこと、乾燥地帯なので、灌漑農業にすると塩化が起き、耕地として使えなくなり、こうして砂漠が広がり、先住民は貧困にあえぎ、土地を捨てて都会に流民するしかなくなるという現象が起きています。

ODA などでも、漁業技術支援を安易に行い、共有地の悲劇を招いたケースがあります。例えば、伝統的な旧式の打ち網漁法などで持続的に漁業が営まれている地域に、漁民の収益を向上させようとして、近代的なトロール漁法やトロール船などを導入すると、トロール網が海底の珊瑚礁や海草繁茂地を破壊したり、幼魚までも捕獲し、海洋資源にダメージを与え、漁獲量が減り、かえって前よりも漁民は貧しくなるということが起こっています。

私自身が参加したフィリピン、パラワンでのプロジェクトもこの共有地の悲劇のパターンでした。北部パラワン地域は、開発プロジェクトから取り残されたということもあり、貧しい漁民が、他の地方からの漁民によってそそのかされ、ダイナマイト漁法や青酸漁法などの破壊的違法漁業に手を染めるようになりました。魚は、サンゴ礁やマングローブ林などに卵を産みます。孵化した幼魚は、サンゴ礁やマングローブの根が入り組む林で外敵から守られ、大きくなります。ダイナマイトを海中に投げ込むと、爆圧で魚がショックを

受け、浮かび上がってくるので、簡単に魚を捕獲できます。また、青酸をサンゴ礁に振り掛けると、サンゴ礁に住む熱帯魚などが弱って簡単に生け捕りできます。なぜか、隣国では、別に刺身で魚を食べるわけではないのですが、生きた魚は珍重されます。美しい熱帯魚はペットとしてももてはやされます。こうして、青酸漁法も盛んになっていきます。問題は、ダイナマイト漁法も青酸漁法も、サンゴ礁を破壊し、ダイナマイト漁法ではサンゴ省は砕け、青酸漁法では、サンゴ礁は白化といって死んでしまった状態になります。こうしてサンゴ礁が破壊されると、魚も卵を産めなく、幼魚を育てるサンゴ礁が破壊され、漁獲高減少になっていきます。こうして、漁民はますます貧しくなっていきます。

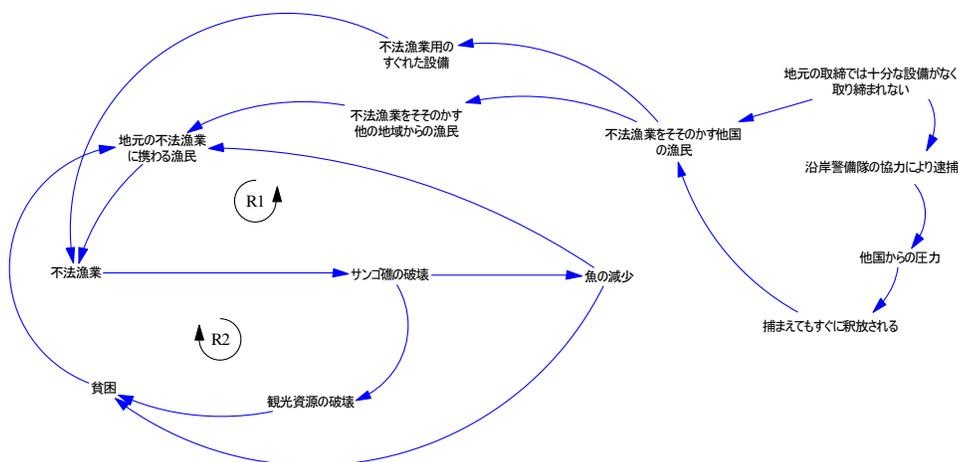


図 5-31：共有化の悲劇

このソリューションでは、エコマッピングと言って、保護区、漁業区を明確にし、それに基づき、パトロールで違法漁業の取締りを行います。保護区での漁業は禁止ですが、漁業区でも、捕獲してよい魚の品種には制限があり、例えばウミガメなどは、漁業区であっても捕獲は禁止されています。取締の一環として、港での水揚げにつきサンプルで、青酸漁法で捕れたかどうか検査しています。

一方、違法漁民の再生ということで、観光開発をやっているのので、観光客向けにも需要がある野菜や果物の栽培技術を伝授し、苗や種などを無料支給し、漁民が違法漁業に手を染めなくてもいいような支援が行われています。こうして、共有地であるサンゴ礁と海洋資源を守ろうとしています。

だが、問題はむしろ隣国からの漁民で、彼らは、高速な漁船を持ち、コンプレッサーや潜水用具などを提供して、地元漁民を手下に使うって青酸漁業などの違法漁業を行います。こうした隣国からの違法行為者を捕まえても、政治的圧力で、すぐ中央政府が釈放してしまいます。アキノ前大統領時代に、米国軍駐留契約を破棄し、米国は完全撤退したので、フィリピンは泣き寝入り状態でしたが、息子のアキノ現大統領になり、最近、米国が海軍を寄港するという契約を行い、それで、最近、隣国からの違法漁民を取り締まるようになったことは大きな前進です。

C) Fifth Discipline Field Book に紹介されている先以外のシステム原型

12) 昨日の友は明日の敵 (Accidental adversaries)

2人の関係を取り上げていて、2人の関係が、うまくいっている場合は、お互いの努力がお互いをサポートし、もちつもたれつ関係を保つ、しかし、他方にとっては悪意と受け取れるようなことがたまたま不幸にも発生すると、その対抗手段を講じ、それがエスカレ

ートしていき、もちつもたれつの良い関係から、お互いに相手の足を引っ張るような逆の関係になっていくというものです。勝手に「昨日の友は明日の敵」と名付けましたが、もともとの意味では、予期しなかったアクシデントで良好な関係が敵対関係になってしまうといった意味です。Jennifer Kemeny が投稿した、The Systems Thinker の 1994 年 2 月号の記事が最初で、センゲの The Fifth Discipline Field book にも取り上げられています。エスカレーションに構造的にも、システムの振る舞いも似ているのですが、エスカレーションでは均衡ループのせいでエスカレーションに歯止めがかからない状態なのに対し、このシステム原型では増加ループが独立した均衡ループをつなぎ、さらに増加ループがお互いの均衡ループを取り巻いているので、増加ループが均衡ループを強化する振る舞いになる点が違います。図 5-32 に定性モデルを示しています。

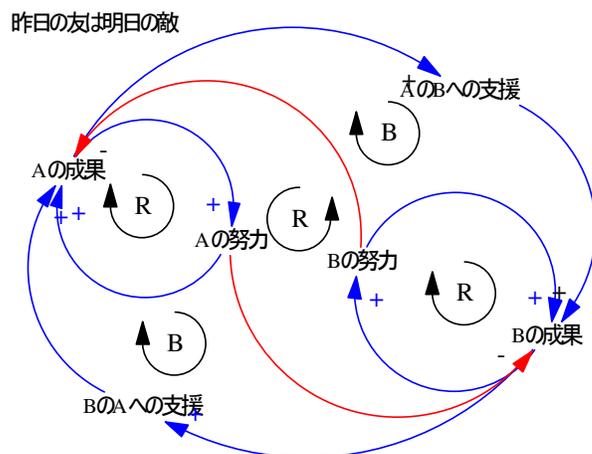


図 5-32 : 昨日の友は明日の敵 (定性モデル)

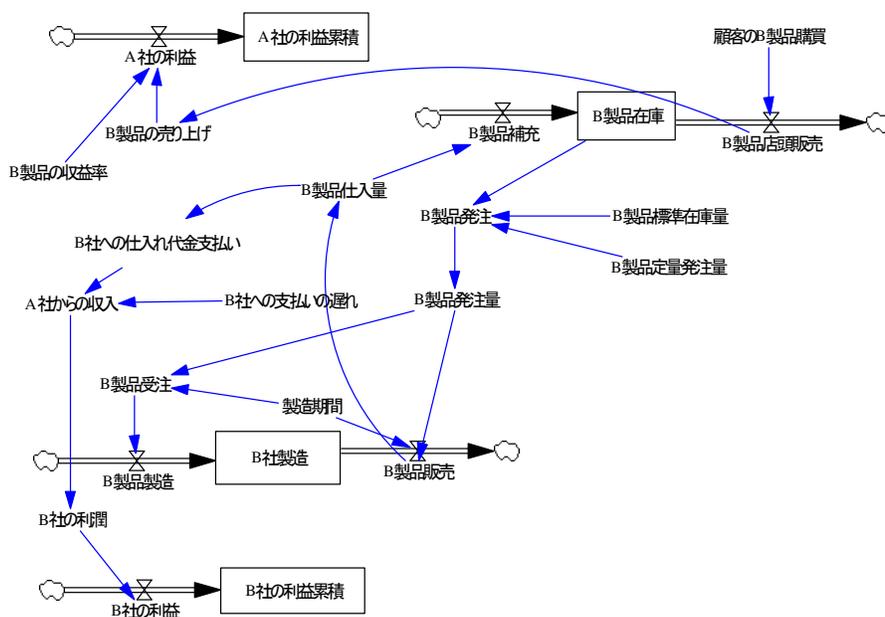


図 5-33 : 昨日の友は明日の敵 (定量モデル、ウインウイン関係)

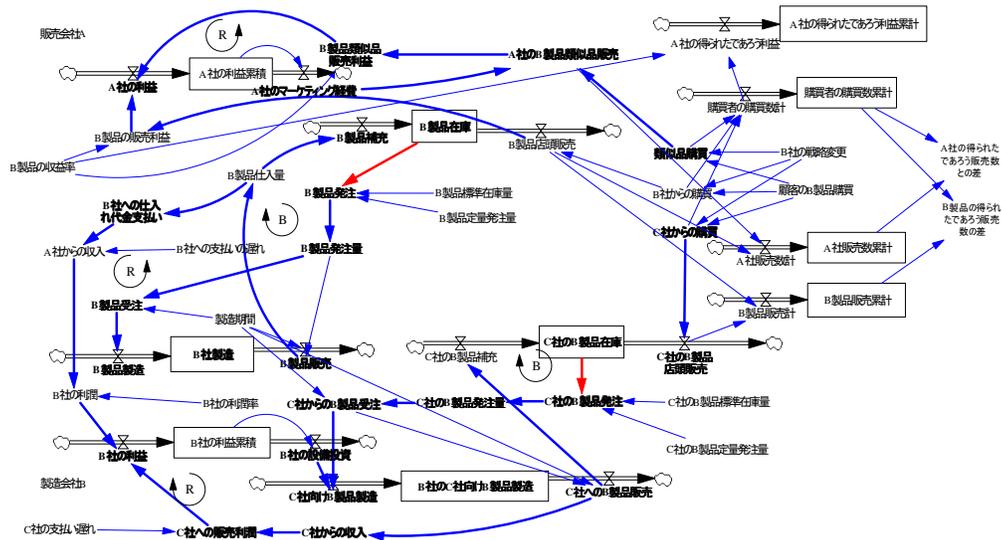


図 5-34：昨日の友は明日の敵（定量モデル、不幸な関係）

定量モデルでは、図 5-33、5-34 のような例を考えてみました。これは、製造業者である B 社が B 製品を受注生産方式で製造し、それを販売会社である A 社が販売するというもので、ウインウイン関係では、A 社は B 製品だけを販売し、B 社は A 社からの注文だけに基づいて B 製品を製造している状態です。この状態が図 5-33 です。

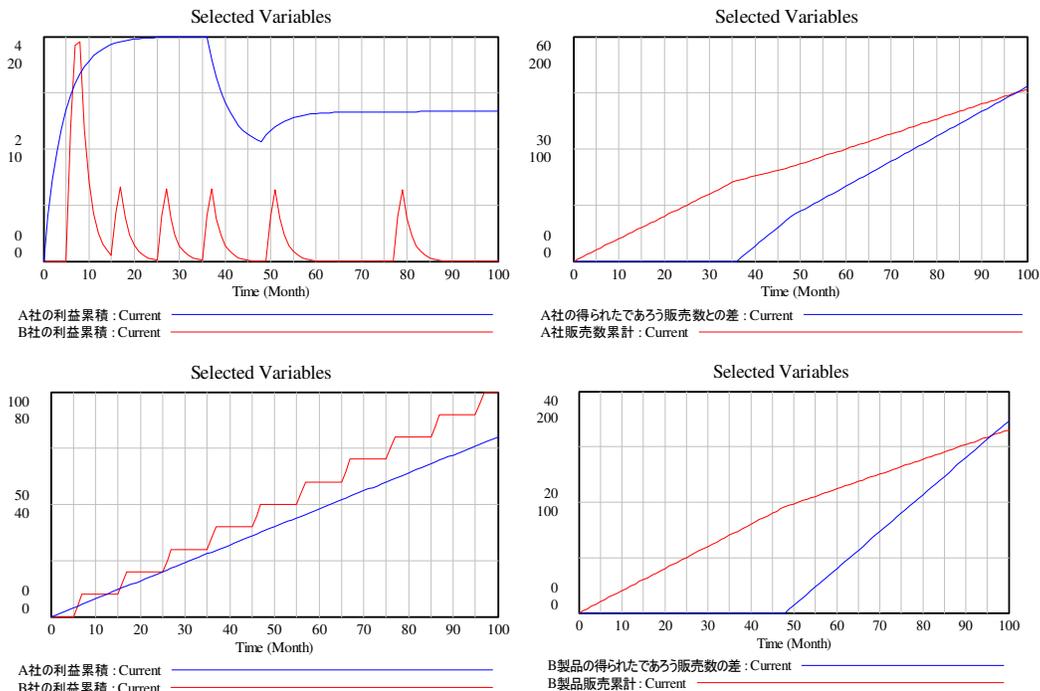


図 5-35：昨日の友は明日の敵（シミュレーション）

敵対の始まりとして、B 社は、もっと製品を販売し、利益を上げるために 3 年目に C 社

にも B 製品を販売することにしました。顧客は、今までは A 社からしか商品を購入できなかったのが、C 社からも購入するようになり、これを受けて、A 社における B 製品の販売数が半減してしまいます。A 社は、対抗手段として、4 年目から、B 製品類似品を販売することにします。

顧客は A 社から B 製品またはその類似品を購入するか、あるいは C 社から B 製品を購入するようになり、A 社は売り上げを半分から 3 分の 2 に戻せますが、B 社は販売が 3 分の 2 に落ちてしまいます。この状態を図 5-34 に示しています。2 つのモデルの違いは、C 社のモジュールがあるかどうかだけです。

このような不幸な出来事の例として、19 世紀後半に成立した日英同盟を考えてみましょう。英国は当時、インドの統治で手がいっぱい、南下し極東を植民地化しようとしていたロシアを抑制し、すでに確保していた香港や上海などの利権を守る余裕が十分ありませんでした。そこで、新興国だった日本と同盟を結び、日本を守り立てることで極東でのロシアからの自分たちが確保した利権を守ろうとしました。新興国だった日本も、英国との同盟は発展と世界に出て行く大きな支援になりました。この良好な関係は日露戦争で日本がかろうじて勝利するまで続きました。しかし、日露戦争後、日本が強くなると、英国がすでに確保していた中国での利権を日本が脅かすのではないかと疑うようになり、日本も、英国のサインを見誤るなどの不幸な事項が重なり、日英同盟はその改定されないことになり、結局解消されてしまいます。その後、日本は英国ではなく、英国に敵対していたドイツに接近し、中国の利権をめぐるはアメリカと衝突し、アメリカによる石油輸出制裁を受けて、石油確保のためにオランダ領だったインドネシアを占領することを考えるようになります。当時英国領だったビルマやマレーシアは石油が出るわけでもなく、特に資源確保のためには占領しなければならなかった土地ではありませんが、オランダと英国は同盟を結んでいて、英国はシンガポールを軍事基地にしていたこと、インドネシアの石油を日本に運ぶためにはシンガポールの英国軍事基地は脅威だったので、第 2 次世界大戦では英国と戦争をすることになってしまうという不幸な歴史の流れになってしまいます。

労使協調でも、それまでうまくいっていた会社と労働組合の関係が、労使担当者が変わったため、労働組合との話し合いをおざなりにするようになり、会社は社員のことを考えていないと労働組合が疑うようになり、会社側の担当者も業績や計画などのことを労働組合に説明しなくなり、次第にぎくしゃくした関係になっていき、ついには不信感をつのらせて対立するようになるという例をいくつか見えています。

中長期的なビジョンを成立させている要因というものをしっかり分析し、不幸な関係に陥らないように、せつかく確保したウインウインの関係を維持し続けることが重要です。

13) 魅力の基本 (Attractiveness principle)

人口もほどほどで、緑もまだ多く、文化的で仕事の通勤にも便利な都市があるとすると、みんなこぞってその都市に移り住むようになり、人口が急増するが、都市のインフラ整備などはその人口増加に追いつけなく、とたんに緑は少なくなる、インフラは未整備で道路は渋滞し、物価も高くなり、治安も悪くなる、といったように、かつてはその都市の魅力であったものが、損なわれてしまう現象で、フォレストターの **Collected Papers** にある同名の論文からのものです。意味合いからは、「損なわれた本来の魅力」といったニアンスでしょうか。システム原型は、増加ループを資源制約から均衡ループが増加抑制に働き、それを解決しようとする、別の資源制約を持つ均衡ループが働くというもので、構造的には成長の限界と似ているのですが、違いは均衡ループが複数あることで、このために制約条件を克服して成長を続けていくことが難しくなります。図 5-36 に定性モデルを示しています。

定量モデル例として、美味しいと評判のレストランがビジネスを拡大し、2 号店、3 号店

と店を開こうとすると、店の味を保持できるシェフの数の増加や育成のために必要な修行期間などの制限から味の良さを確保できなくなり、またこれを何とかレシピの標準化や資質の優れたシェフを雇用して短期間に育成できたとしても、今度は食材の確保の限界や店のテーブル数などの立地条件の制約など次々に問題が発生して、最初に思い描いていたようにはレストラン・ビジネスはうまく成長しないという例を挙げてみましょう。

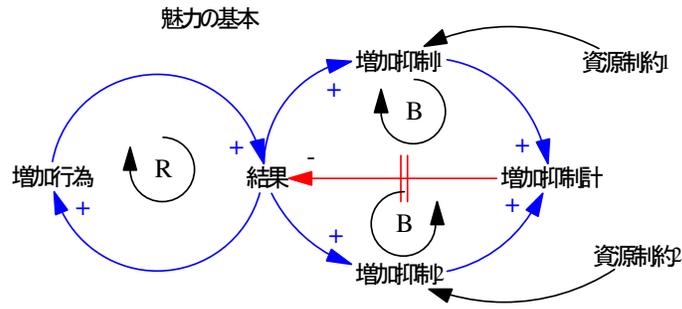


図 5-36：魅力の基本（定性モデル）

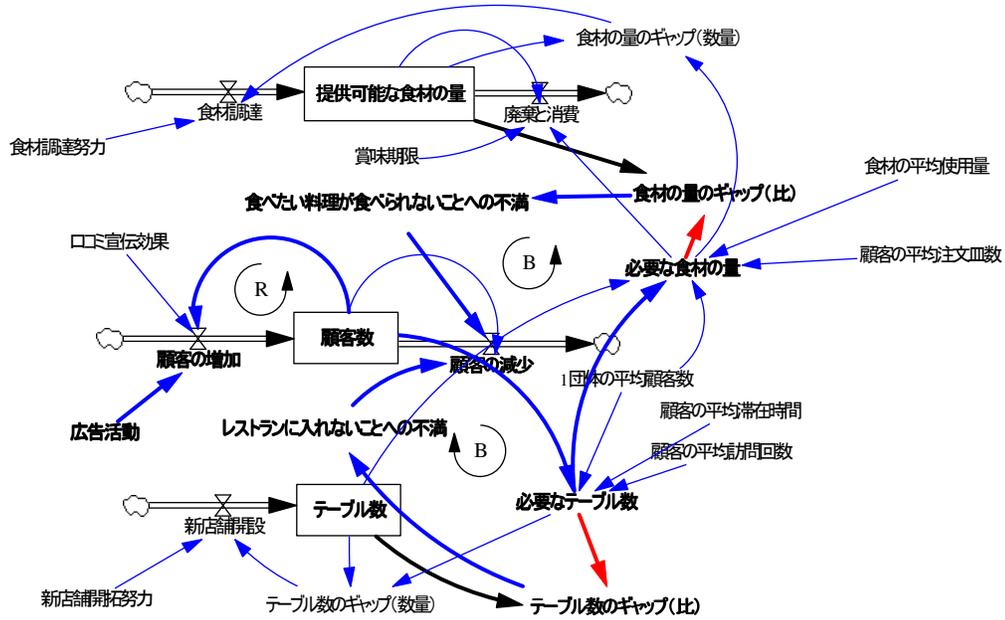


図 5-37：魅力の基本（定量モデル）

モデルでは、店舗拡大の制約及び食材調達の制約の 2 つの制約があります。店舗を急速には拡大できない制約条件から、いつ訪れても入れない、予約しようとしても断られるという頻度が多くなり、顧客を失うことで成長できないようになっていきます。また、適切な食材確保の制限に関しては、原産地から必要な量を確実に仕入れられるような交渉や、鮮度を保ちながら運ぶための方法、食材を生産している農家などから協力を得ることなどのロジスティック整備を早急にはできない制限から食材の量が不足し、せっかく美味しい料理が目当てで入ったのに、注文すると、「あいにく本日はもう売り切れてしまいました」と

断られてしまうという頻度が多くなり、食べたい料理が食べられないという不満で顧客を失うことで成長できないという設定にしています。

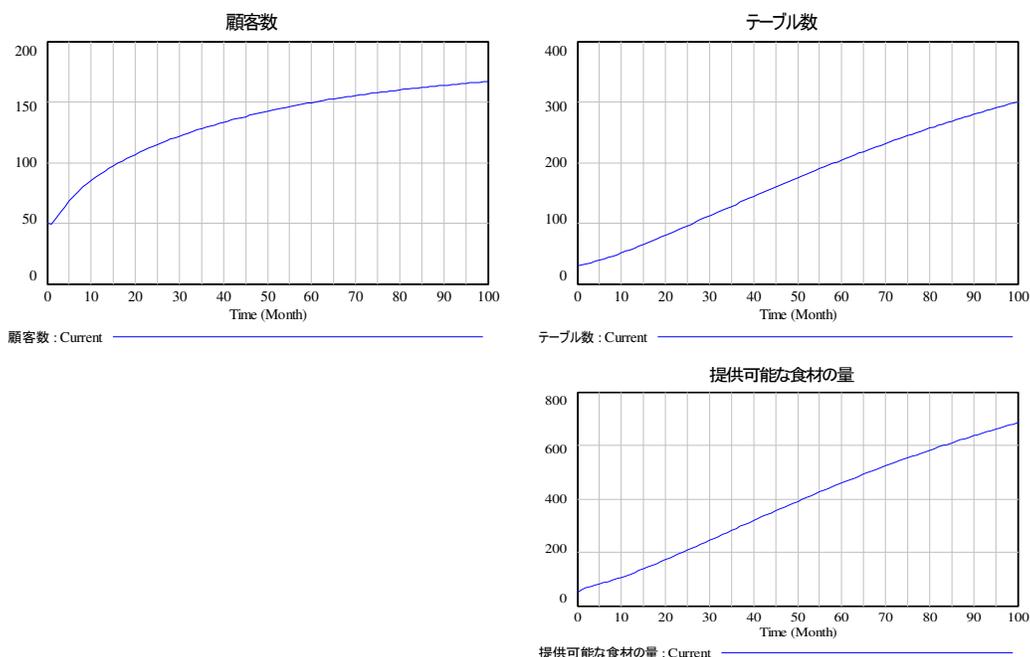


図 1-8-3：魅力の基本（シミュレーション）

こういった失敗に陥らないために、成功の要因、あるいはビジネスの売りは何なのかをしっかりと把握し、その要素はきちんと確保することが重要です。よく製造業などで、事業多角化を行い、失敗すると本業復帰が言われますが、売上を伸ばすことだけが至上命題になっている会社で見られる失敗です。自社は、本当は一体何をしようとしていたのかを機会あるごとにしっかりと再考しながら事業を拡大していかないとこのシステム原型の失敗に陥ってしまいます。

(2) システム原型の適用

重複し、あるいは内蔵する複雑なフィードバック・ループ構造で表現されるシステムには、全体として、あるいは部分として、システム原型に示したこういった典型的な振る舞いを示す部分があるので、そういった部分を探し出し、そこからシステム全体の振る舞いを理解する糸口とすることができます。ただ、ループには干渉力の違いがあり、干渉力の違いで、同じ構造であっても違った振る舞いになることもあり、定性分析だけではなかなか分かりにくいので、定量的に分析することが必要になるように思っています。

システム原型の可用性の評価に関してはさまざまな意見があり、極端なものでは、意味がないと主張する人もいますが、私自身は有用と考えています。モデルを構築する際に 2 つのやり方があります。ボトム・アップ・アプローチとトップ・ダウン・アプローチです。

ボトム・アップ的なアプローチは、モデルをゼロから作って行って、あるモデルが完成した場合、その動作を、モデルを構成するループを 1 つずつ、動作をトレースし、全てのフィードバック・ループの振る舞いが妥当であればモデルも妥当であろうとするボトム・アップ的なチェックを行うものです。その際に、フィードバック・ループを単独で見終らせるのではなく、隣接するフィードバック・ループの関係をチェックすることも必要で、この隣接するフィードバック・ループの関係をチェックするツールとして、こういっ

た振る舞いになるのではないかという観点から振る舞いを調べ、妥当性をチェックする手法として、システム原型を使うことが有効です。

トップ・ダウン的なアプローチは、最初から、先のシステム原型の振る舞いを念頭にシステムを構築し、それを追加、あるいは拡大してモデルを作るもので、モデルのコアになる部分は、システム原型で示された振る舞いになることが保証されているので、安心してコアからシステムをどんどん拡張していくことができます。

例えば、ある事業やプロジェクトの評価を行う際に用いられる手法では、現象なり振る舞いなどが最初から分かっている、焦点は、何ゆえにそのような問題が発生し、何が解決に有効なのかを探ることにあります。このような場合、1から白紙でモデルを構築することが必ずしも効率的ではないし、時間的制約から、このやり方を採択した場合、モデルを作って現象をトレースすることで終わりになりがちです。そこで、最初から、想定したシステム原型を、起きている問題に対する仮説として捉え、考えている問題構造をシステム原型で置き換えてモデルを作成していくことで、短期間にモデルを構築できます。また、ある変更を加えた際に、もし想定しているシステム原型と違った振る舞いをした場合、その理由を探っていく、いわば仮説検証、仮説が正しくない可能性、といったように考えていくことができます。

(3) 5章のまとめ

システム原型とよばれるシステムの振る舞いのパターンがあり、対象を単純化して考える際には有効な方法だというピーター・センゲの主張には同感します。モデル構築の期間短縮のため、対象のコアとなるフィードバックはこういったシステム原型のあるパターンのように振る舞いと決めつけ、そういった振る舞いをするようなモデルを作成するという方法もあるとは思いますが、むしろ、無心に対象をモデル化し、その振る舞いを調べたら、単純化すればシステム原型のある振る舞いになっていたというように理解する方が自然だと思います。

6. モデル構築のヒント

この章では、SD モデル構築におけるヒントと、構造のパターンについて説明します。

(1) モデル構築のヒント

A) モデルを構築していく手順

SD モデルを構築していくやり方、手順ですが、オーソドックスな方法は、1)定性モデルを構築し、それを基に定量モデルを設計する方法で、途中で力つき、定量モデルに行きつかない場合でも、定性モデルはできるという利点があります。また、上記の方法の途中にコンセプトual・モデルを挟む、2)定性モデル→コンセプトual・モデル→定量モデルというステップで設計する方法があり、筆者が薦めている方法です。定性モデルを開発しないで、いきなり定量モデルを開発するというやり方もあり、バリー・リッチモンドなどがやっていた方法ですが、3)コンセプトual・モデル→定量モデルというステップで設計するという、コンセプトual・モデルから始めるやり方があります。途中で力尽き、シミュレーション可能な定量モデルを開発できなくとも、コンセプトual・モデルはでき、コンセプトual・モデルでも十分に定量モデルの指針や考え、コンセプトは明確なので、これを使ってやりたいことをかなり説明できる利点があります。そして、最後は、4)いきなり、定量モデルを開発する方法で、十分にモデル化する対象が分かっている場合、小さなモデルで、コンセプトが単純な場合ではよく行われる方法です。ST という概念が確立する前までは、SD モデルはこの方法で開発されてきました。

2) 定性モデル→コンセプトual・モデル→定量モデルというアプローチ

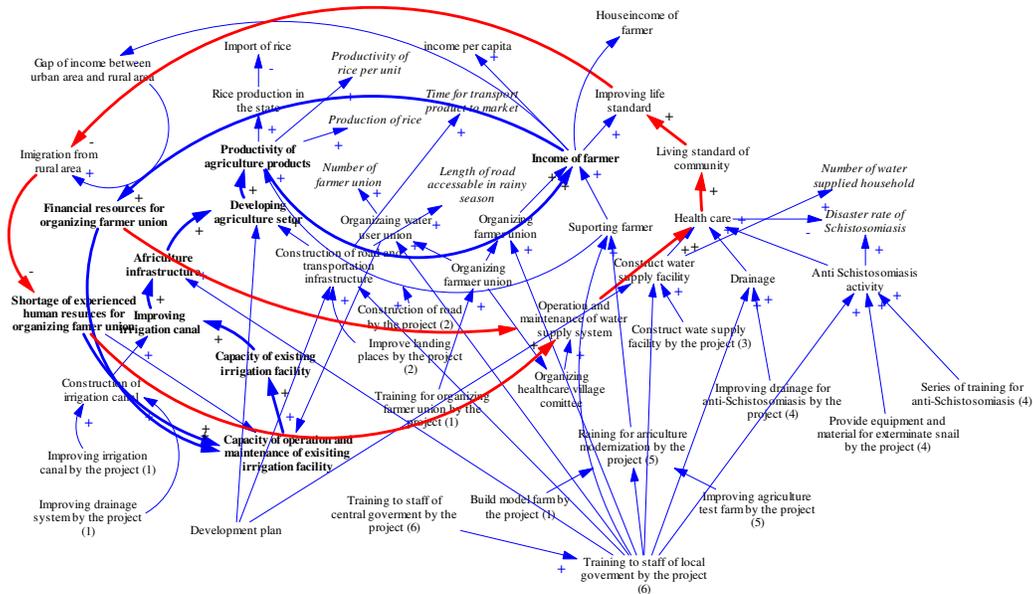


図 6-1：フィリピン、北サマールでの農業インフラ整備事業プロジェクトの ST モデル

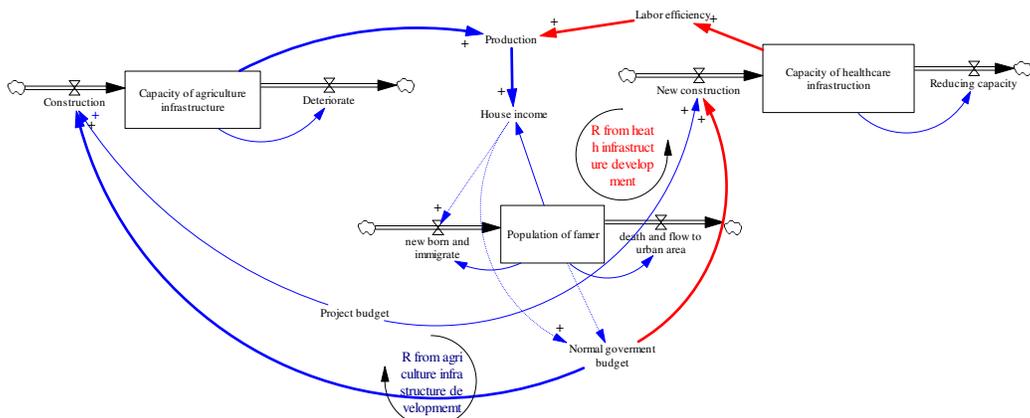


図 6-2 : コンセプチュアル・モデル

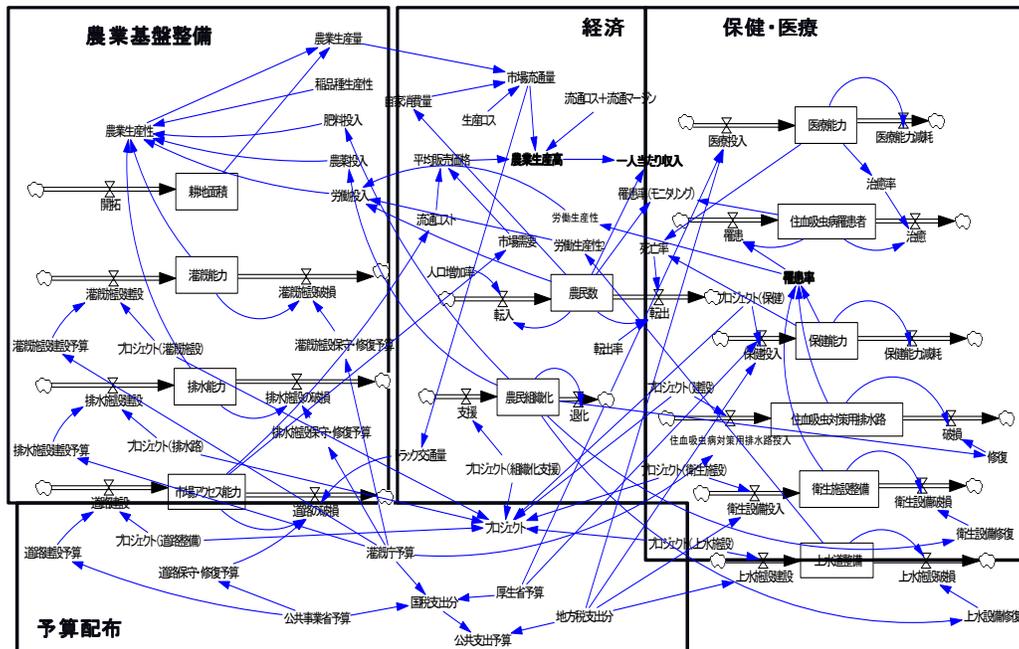


図 6-3 : SD モデル

図 6-1 から 6-3 は、定性モデル→コンセプチュアル・モデル→定量モデルの流れの例を示したものです。図 6-1 のような ST モデルが構築され、定性分析が行われます。ここからいきなり図 6-3 のような SD モデルを構築する前に、図 6-2 のようなコンセプチュアル・モデルを構築し、図 6-3 のようなモデルに詳細化する前に、モデルの基本的な構造を検討します。図 6-3 の複雑そうに見える定量モデルも、実は、図 6-2 のコンセプト・モデルを基に、左の農業インフラと右の保健・医療インフラの部分の詳細化し、展開していただけです。この図 6-2 のコンセプチュアル・モデルでは、単に農業インフラになっていますが、これを、図 6-3 では、農業基盤整備モジュールの中で、灌漑施設、道路整備、河川輸送港湾整備、マーケット整備に詳細化しています。同じく、図 6-2 のコンセプチュアル・モデルでは、単に保健・医療インフラになっていますが、図 6-3 では、保健・医療モジュールの中で、病院や診療所などの医療能力、住血吸虫患者、住血吸虫防止用排水路整備、トイレなどの村落単位の衛生施設整備、そして上水道整備のストックに細分化しています。こうして、いき

なり複雑な SD モデルを構築する前に、簡単なコンセプチュアル・モデルで、おおまかにどのような構造やシステムにするかを検討し、モデルを構築していくことで、複雑化の中でつつい見失いがちになりやい全体構造をきちんと認識しておきます。

3) コンセプチュアル・モデル→定量モデルというアプローチ

バリー・リッチモンドなどがよくやっていた手法ですが、いきなり、コンセプチュアル・モデルから始め、必要に応じて詳細化していく方法で、詳細化された SD モデルが完成しなくとも、大ざっぱではあっても、途中でシミュレーション結果を示すことができるという利点があります。彼は、議論しながら議事録代わりに SD コンセプチュアル・モデルを構築し、適当な値を入れて、議論の途中でシミュレーション結果を参加者に示し、議論をそのシミュレーション結果に基づき進めるというやり方をしていて、その迫力ある定量的な説明に感心したことがあります。SD モデルを作り慣れてきた人にはお勧めの方法です。

自制への介入

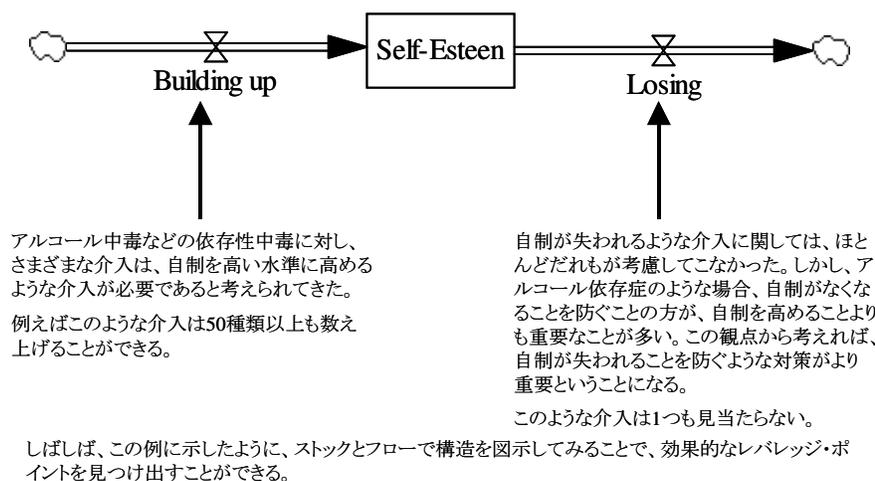


図 6-4：リッチモンドのコンセプチュアル・モデル（注 6-1）

図 6-4 は、リッチモンドの最初のモデルで、ST でのアルコール依存症をトピックスに取り上げたディスカッションでのものです。議論の途中で、アルコール依存症の問題では、自制という観点で SD 的に見ると、入力、つまり自制を入力側で高めるとい議論ばかりされているが、自制を出力側で見るものもあっていいのではないかと、そうでないとシステム的に考えておかしいという爆弾発言を行ったものです。確かに、アルコール依存症の危険性を理解させるビデオを見せるとか、アルコール依存症者の会合に参加させ、体験を聞かせるなどの研修で自制を高めることで、アルコール依存度を減らせるでしょうが、何かのきっかけで蓄積された自制が失われてしまい、アルコール依存に戻ってしまうという側面の議論も重要です。ストックは何か、ストックを増加させるものがあれば、減少させるものも当然あるだろうという考え方、見方は、ST で考えているとつつい見失われがちです。

彼は、常に、こういったストックとフローが数個ある非常に簡単なコンセプチュアル・モデルを議論のメモ代わりに描いて、それを基に議論し、議論に基づいて SD モデルを完成させています。

B) 比較的小さなモデルや構造が単純なモデルの構築方法

比較的小さなモデルや構造が単純なモデルを構築する上でのヒントのようなものを述べ

ていきます。このような場合、モジュールで考える必要はなく、いきなりモデルを構築すると思います。それでも、5)フィードバック・ループを中心に考えて設計する方法と、6)定性モデルを細かくしていった定量モデルにしていく方法、7)メインのストックを中心にしたアプローチが考えられます。

5) フィードバック・ループを中心に考えて設計する方法

一種のボトム・アップ・アプローチですが、考えられるフィードバック・ループは全て取り入れてモデルを作り、パラメータに仮の数値や等式を入れ、その後、それぞれのループについて振る舞いや強さをチェックして妥当性を検討しながら、ループの追加や削除、パラメータに設定する値の最適値を決定していきます。もし、フィードバック・ループを分解し、独立したフィードバック・ループにすることができるのであれば、独立したフィードバック・ループを表現する定量モデルを構築し、そこでシミュレーションで振る舞いを確認し、その後、それぞれのフィードバック・ループを表現する定量モデルを結合させていくといいでしょう。

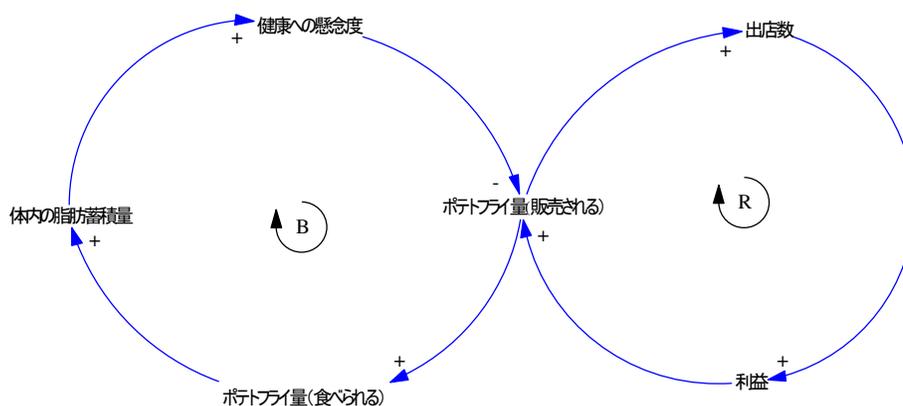


図 6-5：ポテトフライの定性モデル

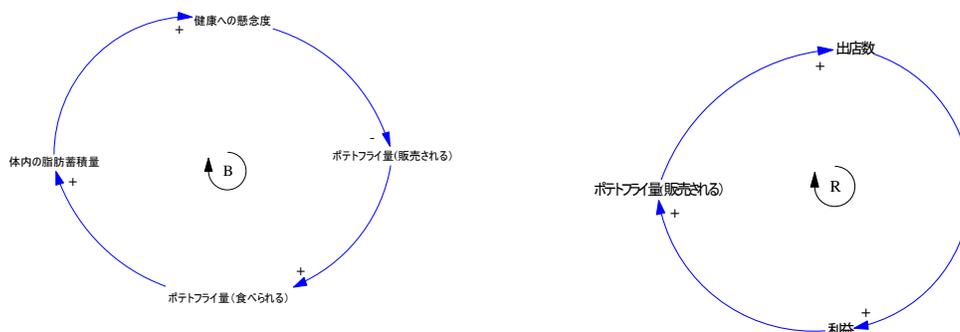


図 6-6：分解した要素としてのポテトフライの定性モデル

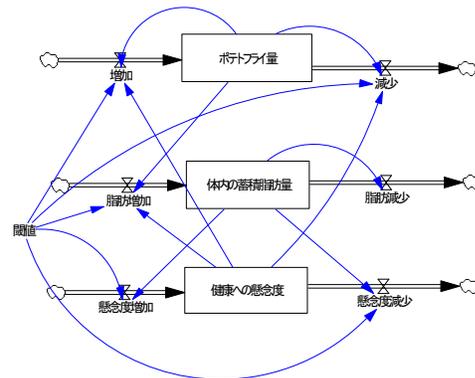
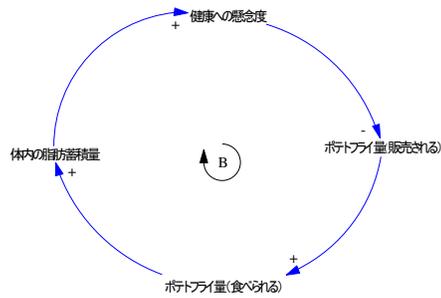


図 6-7：分解した要素としてのポテトフライの定性モデルと定量モデル

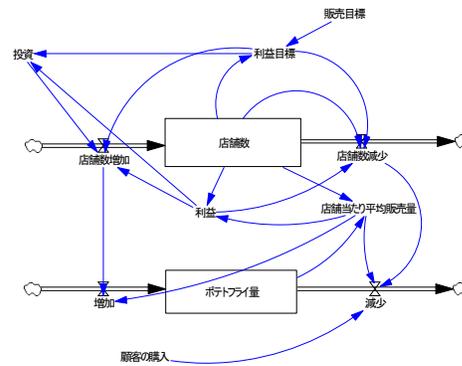
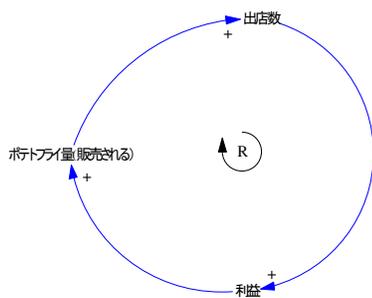


図 6-8：分解した要素としてのポテトフライの定性モデルと定量モデル

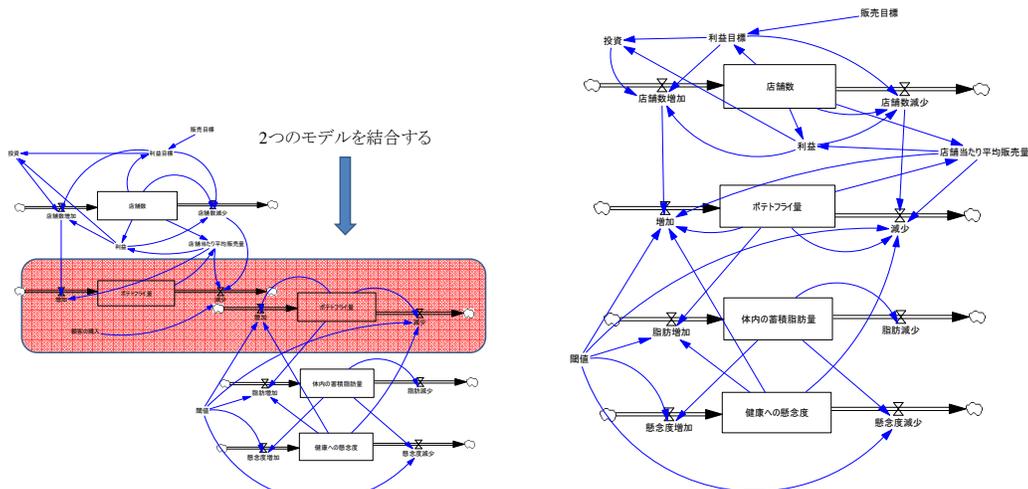


図 6-9：分解した要素としてのポテトフライの定量モデルの合成

図 6-5 は完成した定性モデルです。このモデルは、図 6-6 に示したように、2つのフィードバック・ループで構成されています。そこで、それぞれの定性モデルを独立した定量モデルで表現したものが、図 6-7 及び図 6-8 です。こうして、定量モデルが完成したら、図 6-9 のように 2 つの定量モデルを結合します。ポテトフライ量というストックとそれを取り巻

くフローは、もともと、2つのフィードバック・ループの共有部分だったので、この共有部分を結合します。こうして、図 6-9 の右に示したような定量モデルが完成します。(注 6-2)

6) 定性モデルを細かくしていった定量モデルにしていく方法：

これは、定性モデルをとにかく細かく定義していくことで、定量モデル構築に必要なパラメータを全て列挙する方法です。定量モデル構築に必要なパラメータが全て列挙されているので、後は、定性モデルのパラメータをストック、フロー、変数、定数に置き換えてしまえば、定量モデルになります。とは言っても、定量モデル構築に向けて詳細化するということがなかなか分かりにくいかもしれません。定性モデルのフィードバック・ループの中で、仮定義でもいいからストックにするものを決めると、ストックには必ず入力と出力が伴うものなので、入力になるパラメータと出力になるパラメータが決まります。これだけでも決まれば、シミュレーション結果はともかくも、シミュレーション可能な、コアとなる定量モデル構築に必要なパラメータは列挙されたこととなります。以下の手順で細かくしていきます：

- 定性モデルの中で定量モデルにする部分や、そのループの性格（増加ループ、均衡ループ）や要素の正負を再確認
- なるべくループを構成する要素を細かくする。例えば、企業モデルを構築しようとしていて、「収益」→「営業促進への投資」という要素があるとする。定性モデルではこの荒さでも十分だが、「売上」→「営業費用」→「営業促進への投資」と細かく再定義することで、定量モデルに置き換える際にやりやすくなる。
- ループの中でストックになる要素をまずストック変数で置き換える。
- 残りの要素の中で、ストックに直接関係するフロー変数で入出力になるものを決める。
- フローが決まった後の、残った要素はこのストックの入出力になるフローに関して入出力になる変数か定数だけになる。
- 要素間の正負の関係を確認しながら、ループが表現できているか確認する。もし、うまくループが表現できないようであれば、2)に戻って、要素を分解できないか検討する。ただ、定性モデルがそのまま定量モデルになるわけではないので、それでもうまくいかないようであれば、1)に戻って、定量モデルにする範囲を再検討してみる。また、うまくループが表現できればいいのであって、場合によれば要素を変えても構わない。
- 等式や初期値を設定し、シミュレーションしてみて、定性モデルで予測していた定性的な結果がうまく定量的に表現できているか確認する。

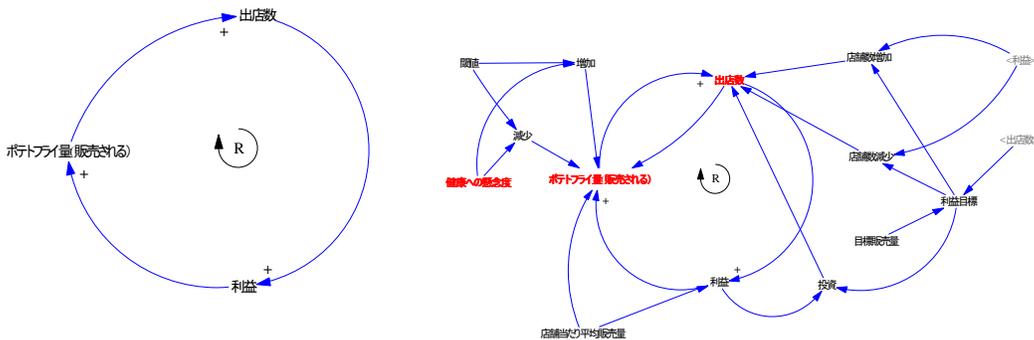


図 6-10：分解した要素としてのポテトフライの定量モデルの合成

図 6-10 は、図 6-8 の定性モデルを詳細化し、最終的には定量モデルにする方法を示したものです。図 6-10 の左の定性モデルで、「ポテトフライ量」、「出店数」という2つのキーワードをストックとします。すると、この2つのストックに関する入力と出力を司るフローがあるはずですが、ストック「店舗数」に関しては、「店舗数増加」と「店舗数減少」という

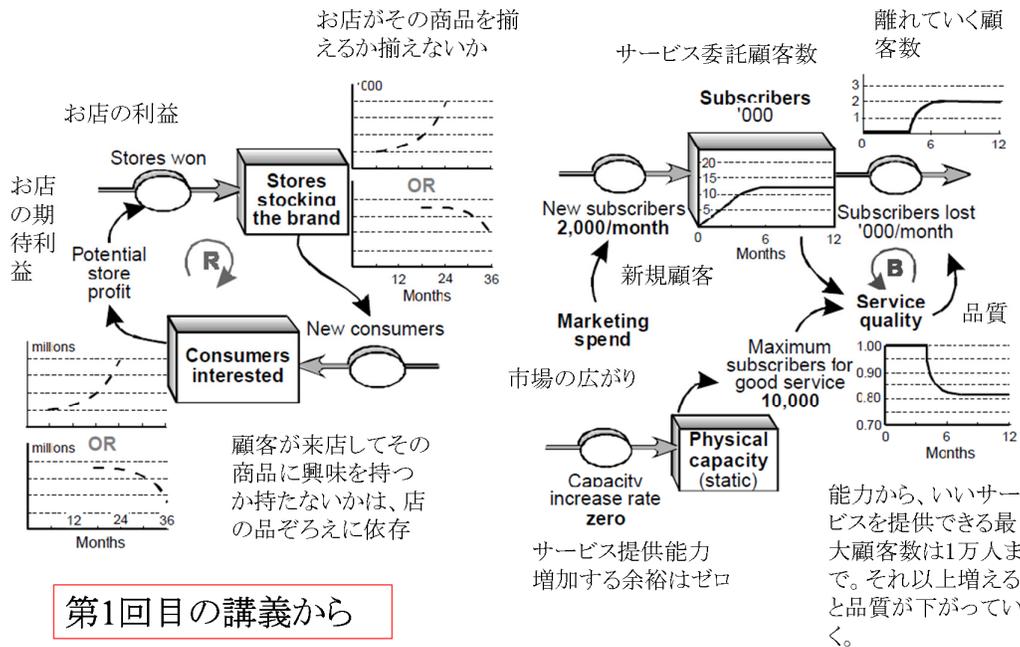
フローがあるはずで、「店舗数増加」、「店舗数減少」があれば、増加あるいは減少させる判断基準を示す閾値のようなものと、利益があるはずで。つまり、例えば、利益を閾値と比較し、閾値よりも多いのであれば、店舗を増加させ、下回っていれば店舗を減少させると考えます。さらには、店の数を増加させる量、例えば、「投資」が必要なパラメータとして考えられます。こうしたロジックによってモデルを詳細化していきます。定性モデルでは区別しませんが、それでも、詳細化し、どの要素がストックであり、どれがフロー、そしてどれが変数ということが分かるまで詳細化すれば、この定性モデルを定量モデルに置き換えることは簡単です。このようにして図 6-8 右のような定量モデルを作成できます。(注 6-3)

7) メインのストックを中心にしたアプローチ

これは、キム・ウォーレンが多用している方法で、最も重要なもの、最も注目したい部分から作っていくというものです。いろいろフィードバック・ループがからむ定性モデルが完成するのですが、フィードバック・ループ構造は一旦置いておいて、一体何を知らりたいのかを考えます。すると、メインになるストックが決まってきます。そこで、メインになるストックに付属するフローを追加し、ループを確認しながら変数、定数を追加し、ループの振る舞いをチェックします。メイン(コア)となるシステムがこれで決まります。メインとなるストックは一個だけとは限りませんが、少なくとも、一番重要とモデル構築者が考えている要素が取り入れられた定量モデルが完成します。ここがコアであれば、コアの範囲でのシミュレーションが可能なので、これでも十分本質的な部分は分かるはずで

す。その上で、さらに詳細化していく必要があれば、同じように、サブとなるシステムを作り、先のコアのシステムにサブとなるシステムを結合し、振る舞いをチェックするということを繰り返していきます。

Kim WarrenのBusiness Strategy Review掲載論文から: 上がったらどうなるか、下がったらどうなるかを、因果関係ループでトレースしながら戦略を考える。



第1回目の講義から

図 6-11 : メインのストックを中心にしたアプローチ (注 6-4)

図 6-11 は第 1 回目の説明で紹介したのですが、例えば左のモデルなどでは、ストックにフローがついているだけの単純な構造を 2 つ結合しただけでのモデルです。ここで、店の品ぞろえ、つまり商品数を増やすというシナリオと減らすというシナリオについて、顧客が増えるか減るかを示しています。彼のモデルは、こういったストックに入出力のフローが付いただけの単純なストック・フローを数個組み合わせただけのモデル、いわばコアだけのモデルで、本質のような部分だけを説明するというやり方を採択しています。モデルを必要最低限のものにすることで、コンセプトが非常に明確になります。

C) 大きなモデルや構造が複雑なモデルの構築方法

大きなモデルや構造が複雑なモデルを構築していくためには、モジュール構造として考えて、モジュール毎に構築していくほうがいいでしょう。いきなり大きなモデルや構造が複雑なモデルを構築すると、とかく全体像がぼけてしまいがちです。

8) モジュールを中心に考えて設計する方法

モジュール間の構造については、次の節で説明しますが、モジュールでモデルを構築する場合、既存のモデルがあり、それが使える場合は、それをモジュールにして、モデルに組み込むのがいいでしょう。

コアを固め、そこにモジュールを追加し、妥当性を検証しながらモデルを膨らませる、あるいは、モジュールの中にストックとフロー、変数を追加する、あるいは削除し、他のモジュールと結び付けていきます。

既存のモデルを部品として使う利点として、システムの動き（振る舞い）をすでに良く知っているという安心感があることが挙げられます。少なくとも、その部分は確実に動くことは実証済で、どう振る舞うかということも納得済みなので、モデル構築作業を、自信を持って進めることができるでしょう。私は、この方法が、モデルを構築する上で一番効率的だと考えています。

9) コンセプトチュアル・モデルを拡張していく方法

この方法は、常にコンセプトを意識しながらモデルを構築していく方法です。定性モデルからあまり重要ではないと考えたフィードバック・ループを削ぎ落とし、単純な、コア・モデルを作ります。これを表現するストックが 1 個だけあり、そこにフローが付属しているモデルを作ります。そして、さらに変数を追加し、ループを表現した定量モデルを作ります。先の最も基本的なフィードバック・ループの定性コア・モデルを表現する上で必要であれば、さらにストックやフロー、変数を加えます。こうして、コンセプトチュアル・モデル（定量モデル）を作ります。この時点ではこのモデルは動かなくともいいでしょう。

このコンセプトチュアル・モデルを拡張していくことで、モデルが大きく、複雑になっても、常にコンセプトは確保できます。

(2) モデルの構造

大きなモデルや構造が複雑なモデルを構築する方法として、以下のようなモデル構造の基本形に関する考え方が参考になるのではないのでしょうか。

最低 1 個のストックと最低 1 個のフローを持つ塊りが、最小のシステムであり、最小のモジュールであると考えられます。図 6-12 は、野鼠と天敵の 2 つのモジュールと考えられ、この 2 つを結び付てシステムとします。つまり、大きなシステムのモデルでは、こんな単純なものでもいいし、もっと複雑なものでもいいのですが、一つのシステムで最低 1 つの機能を持つものをモジュールと呼び、まずミジュールで独立させ、それを結合していくというシステム構築の考え方を導入します。

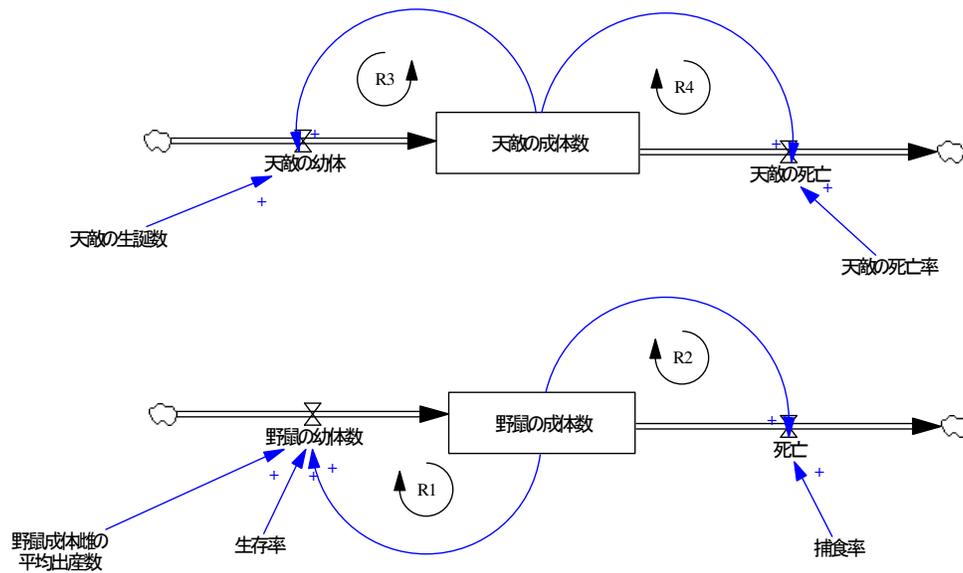


図 6-12：最小のモデル

こうして、モジュールを組み合わせてモデルを作ると、過去に作ったモデルを部品として使うことができるという利点があります。つまり、部品のストック、モジュールのストックを使ってモデルを作ることができ、モデル構築の時間短縮を図ることができます。大きなモデルの場合、一つの画面でモデル全体を表示できないので、Vensim PLE では View でモジュールに切り分けることができます。

Sterman の Business Dynamics にも載っているし、いろんな SD の本にも載っているし、自分でも人口モデル、マクロ経済モデル、製造、在庫（サプライチェーン）、販売などを作った経験があると思います。学習や経験を兼ねて、そういったモデルを自分自身で作成して、保管してあるのではないのでしょうか。こういったモデルはモジュールや部品として使えます。ただ、セクター・モデル、環境モデルなどは、過去のモデルの一部を手直ししたり、追加したりする作り方が多いので、部品としては使いにくいでしょう。

大きなモデルを構築する際に、以下のようなモジュールの関係の基本形が存在します。あるいは、大きなモデルを作るのであれば、コントロールという観点から、以下のようなことを意識して作っていくのがいいのではないかと思います。

- ・パイプライン構造
- ・ネットワーク構造
- ・単純な支配構造
- ・マクロをモジュールとして独立させた構造
- ・多重のコントロール構造

1) パイプライン構造

状態が遷移する構造で、人口モデルなどのように基本的にはストックが遷移しますが、産業連関、付加価値連鎖などのように、モジュールが遷移するものもあります。この構造では、特にコントロールが意識されているわけではありませんが、最初の遷移モジュール

で初期条件が決まるもで、そこでコントロールされているというように考えることもできます。

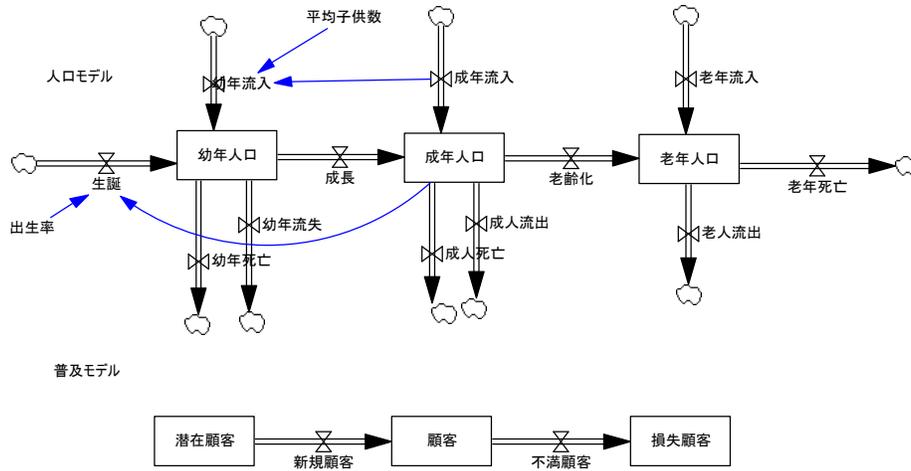


図 6-13 : パイプライン構造

2) ネットワーク構造

これは、モジュールが対等の関係で結び付くもので、特に何かが何かを支配するのではなく、情報などを与えるという関係になります。多分、モジュールのコントロールを考えると、この構造に自然になるでしょう。

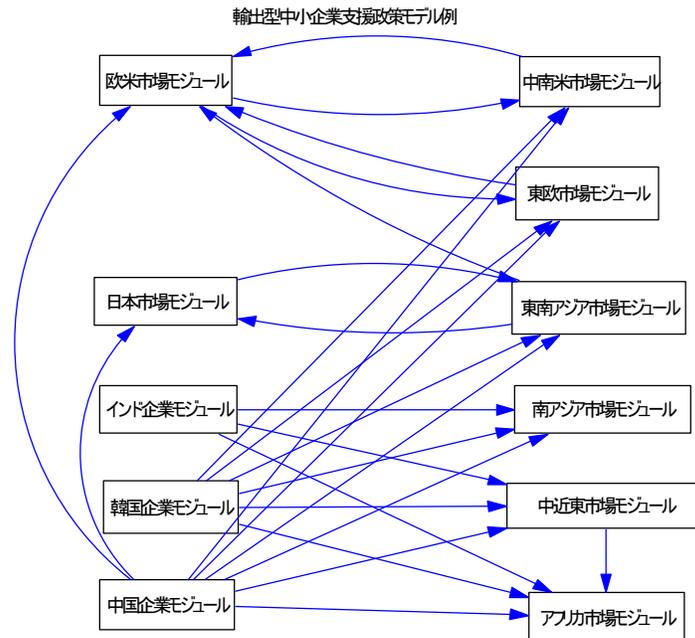


図 6-14 : 貿易システムの全体構造 (ネットワーク型の例)

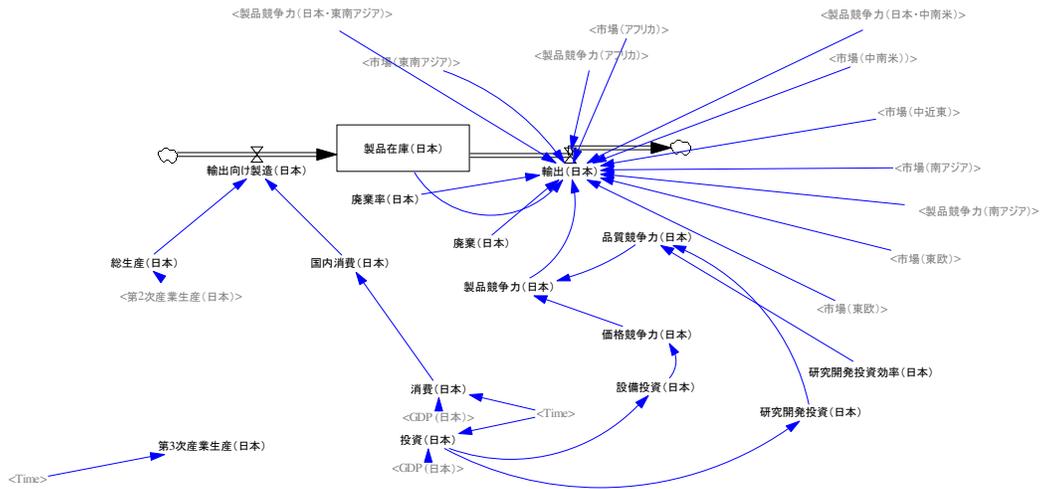


図 6-15 : 貿易システムのある国のモジュール (ネットワーク型の例)

3) 単純な支配構造

予算配分モジュールがその他のモジュールを支配している

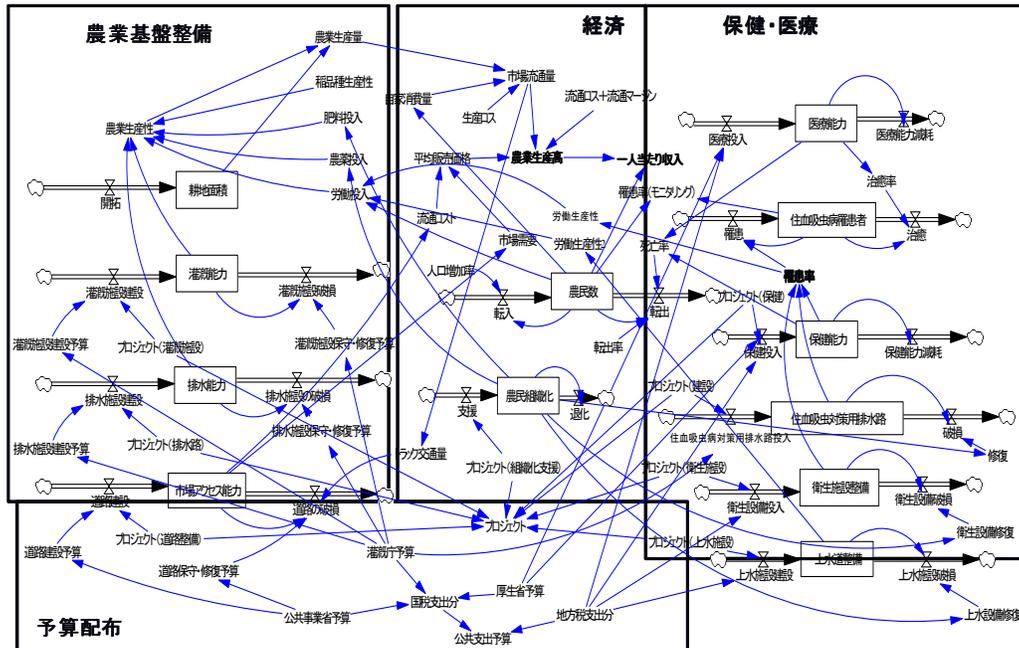


図 6-16 : 単純な支配構造

あるモジュールが一方的にデータを与え、それが他のモジュールにとって外部政策変数のように振る舞うもので、特に、マクロ経済モジュールがあって、それが他のセクターに影響を与えるものについては、別建てとしますが、基本的には同じです。例に挙げたものは、政策モジュール、予算配分モジュールがあり、それに沿ってインフラ整備やプロジェ

クトが実施される構造になっています。この例では、経済モジュールの中に人口モジュールがあり、それを基に税収が計算され、税収の一部がプロジェクトで建設されたインフラの保守に使われます。従って、経済モジュールと予算配賦モジュールにフィードバックが無い分けではないのですが、予算配賦モジュールからのインフラ建設予算の方が圧倒的に金額が大きいこと、経済モジュールからの予算配賦モジュールへのフィードバックをあまり意識していなく、予算配賦の変化のインフラ建設とその経済効果の測定に注目しているため、このモデルの場合、予算配賦モジュールの一方的な支配と考えられます。

プロジェクト期間+10年がターゲットなので、地域経済が伸び、一般会計予算が増加し、農業基盤や医療・保健がさらに充実するというフィードバックはあまり重要視していない。一般会計予算はプロジェクト予算に比べそもそもあまり大きくないのでさらなるプロジェクトは無理。プロジェクトで作った施設が維持管理できればいいといった程度。

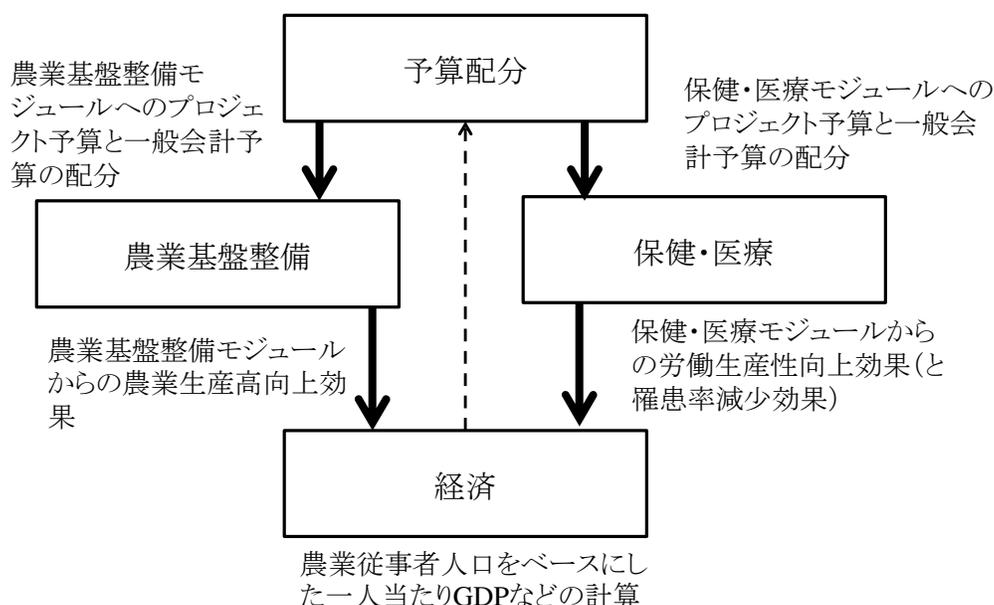


図 6-17 : 図 6-16 の単純な支配構造の模式化

4) マクロをモジュールとして独立させた構造

比較的よくある構造で、マクロ経済の状況が、他のセクターを決めます。例に挙げたものは、高速道路モデルです。この例では、マクロ経済モジュールが一方的にマクロ経済予測を行っています。道路建設モジュールからマクロ経済モジュールへのフィードバックも考えられなくはなのですが、計算や効果が難しいので、ここでは考慮していません。マクロ経済モジュールで決まる道路建設需要を満たすものと想定しています。従って、マクロ経済モジュールからの一方的なコントロールと見なせます。

参考までに、高速道路の経済効果は：

- －実地測定での交通量から、高速道路を作った時の交通量の増加を予想
 - －他の高速道路プロジェクトでの交通量の増加を外挿
 - －物品の輸送量増加に伴う経済効果（取引量増加）を計算
 - －物品の輸送時間短縮に伴う経済効果（取引額補正）を計算
 - 移送される商品の価格と取引量で経済効果を計算
 - －人の移動量増加に伴う経済効果（労働機会、労働賃金）を計算
- といった手順で求めるのですが、ただ問題は、こういった経済効果をマクロ経済モデルに

整合的に組み込むのは難しいので、通常、こういったインフラ整備プロジェクトのためにSDモデルを作る場合、既存の計量経済モデルなどをそのまま使うなどしてマクロ経済モジュールを先に作成し、それを前提にその他のモジュールを構築していきます。

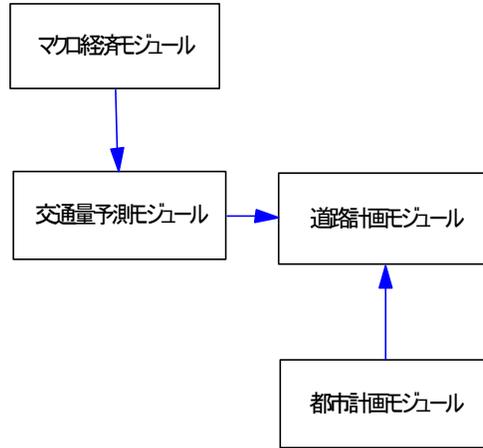


図 6-18 : マクロ経済モジュールが他のモジュールをコントロールする例

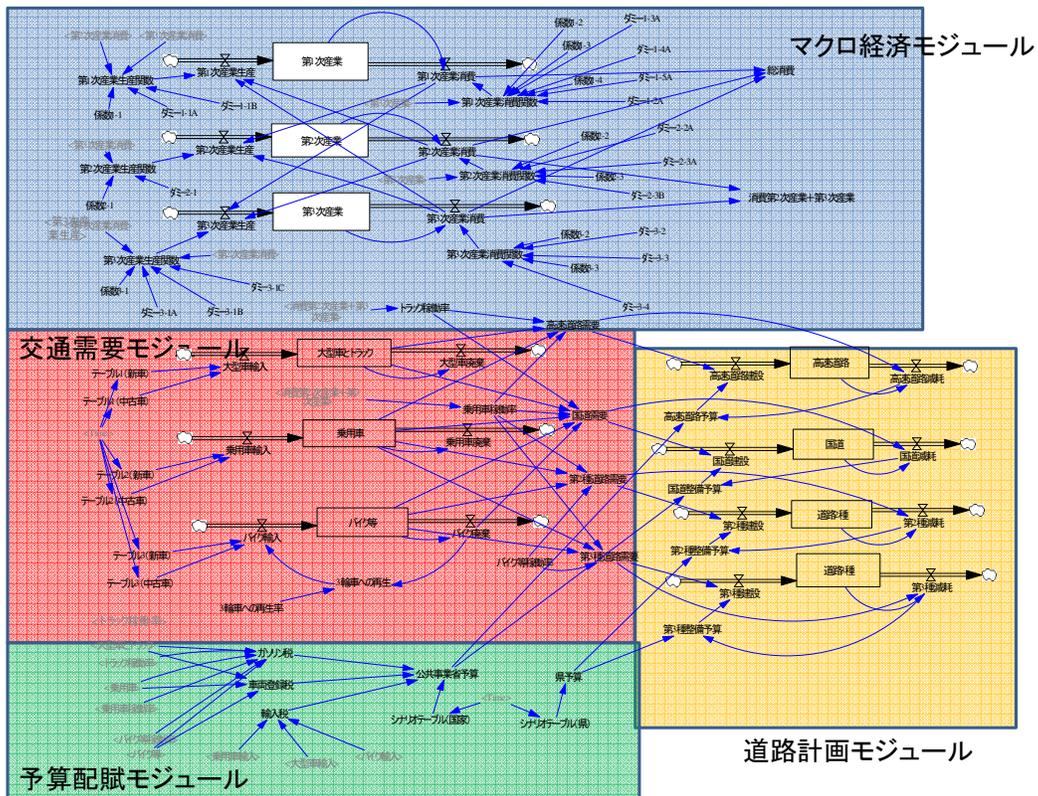


図 6-19 : マクロ経済モジュールが他のモジュールをコントロールする例

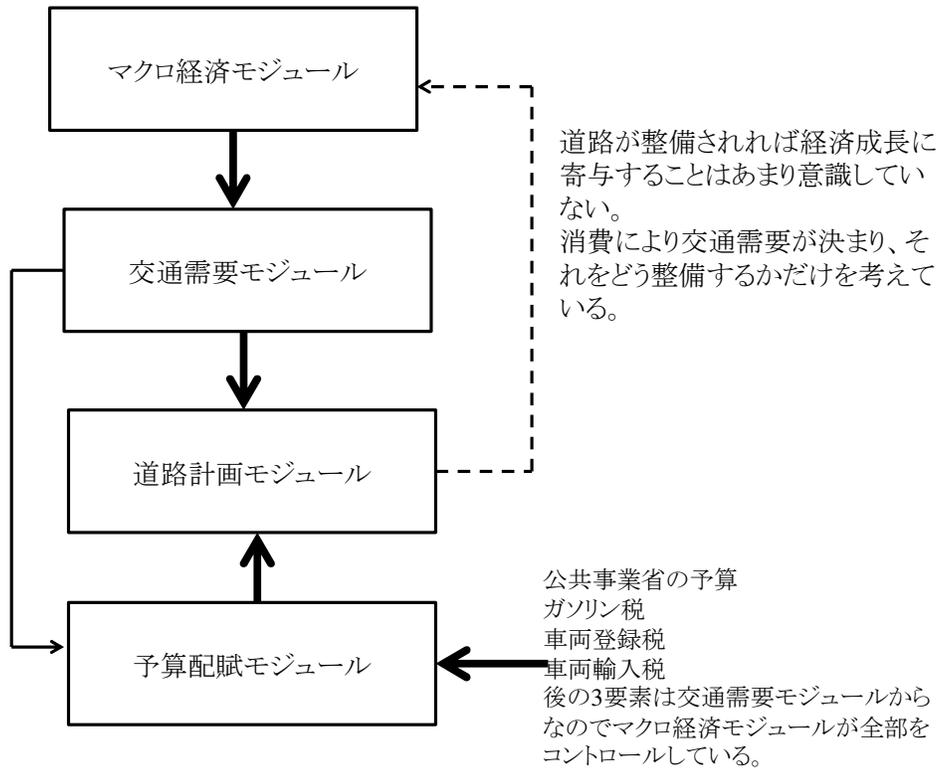


図 6-20 : 図 6-18 の模式化

5) 多重のコントロール構造

重層的にコントロールされる構造で、最上層のモジュールは最下層のモジュールに対して、一般的に一方的なコントロールとなります。



図 6-21 : 重層的なコントロール

(3) 6章のまとめ

大きなモデルを構築する場合は、モジュールに分け、モジュール間のコントロール構造

を意識して作成することが有効です。

また、コンセプチュアル・モデルをまず固め、それを元にモデルを進化させていく方法が有効です。

注：

6-1) The “Thinking” in Systems Thinking: Seven Essential Skills, by Barry Richmond, Pegasus Communications, Inc. 2000

6-2) 演習 06 ポテトチップス

6-3) 演習 06 ポテトチップス

6-4) The Dynamics of Rivalry, by Kim Warren, Business Strategy Review, 1999, Vol. 10 Issue 4

(以上)