

# ビジネス・シミュレーション 第3回

December 16, 2008

松本憲洋 (POSY Corp.)

[matsu@posy.co.jp](mailto:matsu@posy.co.jp)

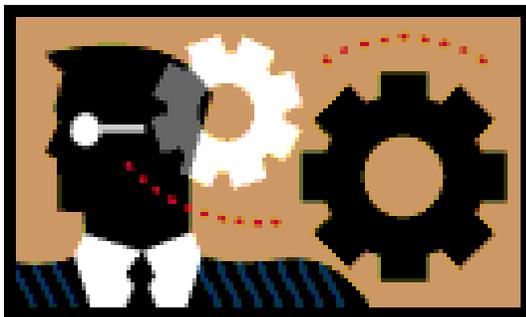
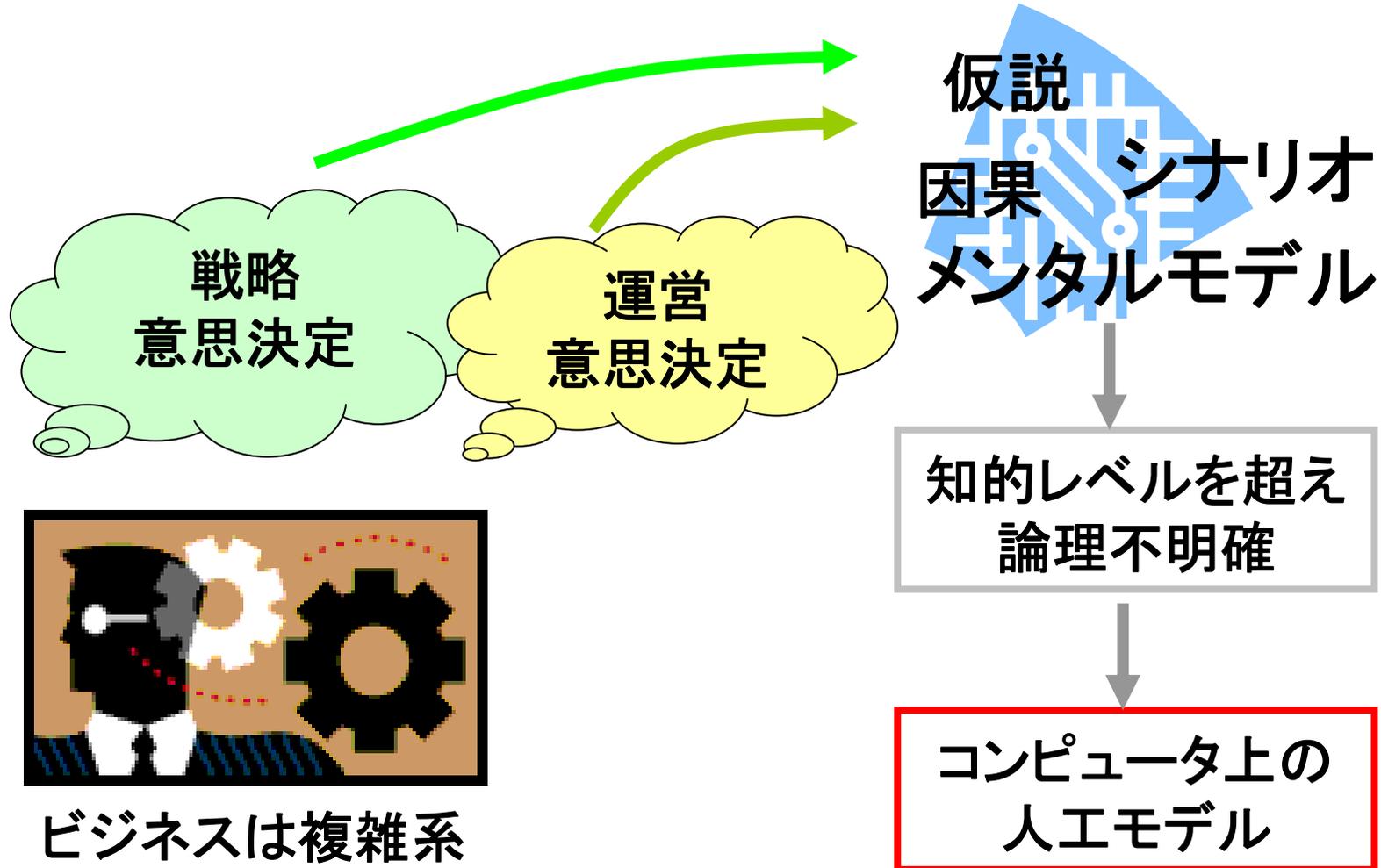
<http://www.posy.co.jp>

# 第2回目の復習

# SDモデルを使う必要がある、なぜ？



役割 : 業務分析    モデル構築    仮想経営(仮説検証)  
実世界に適用    実績の分析と適応策の策定

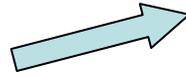


ビジネスは複雑系

# ビジネス・プロセスの設計の流れ

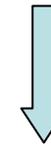
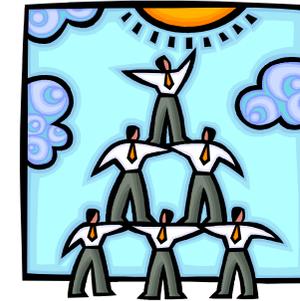


ミッション／事業目標



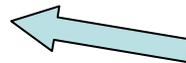
経営ノウハウに基づき仮説  
⇒モデルの構築  
⇒ミッション／事業目標を数値化  
⇒モデル構造と外生変数を  
操作し目標変数を最適化  
(Solver活用)

目標達成できる条件

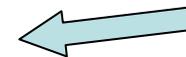


リスク要素を考慮した最適化  
リスク評価

最適条件化でのシミュレーション



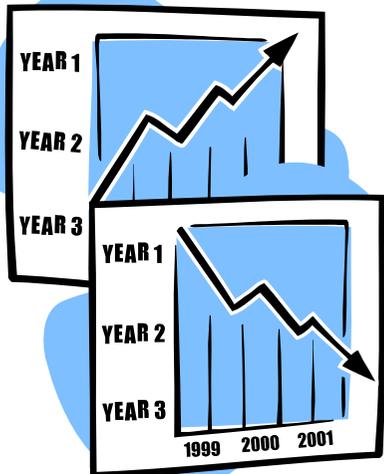
**BSCのKPIの目標値**



# 目標と実績値の乖離の分析と適応の流れ

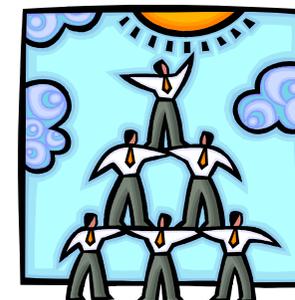


KPIの目標値からの乖離



適切な仮説を発見

経営ノウハウに基づき仮説を変更  
⇒モデルの修正  
⇒モデル構造と外生変数を操作し  
KPIの実績値等を説明できる仮  
説を探索 (Solver活用)



発見した仮説の下で、戦略の  
練り直しとBSCの見直し  
リスク要素を考慮した最適化  
リスク評価



**BSCの継続**



最適条件化での  
シミュレーション



# システムとは？



## システム

構成要素が互いに関連しあうことによって機能する、複雑な集合体

## システムの例

太陽系や水利系のような自然システム

生命体のような生物システム

機械やプラントのような人工システム

哲学体系のような抽象システム

会社や行政機関のような経営システム

地域自治会のような社会システム

## システム研究の方法論

還元論 : 全体に対する構成要素を重視する考え方

構成要素を特定し理解し、そこから全体の理解にたどり着こうとする。

システム論 : 構成要素に注目するだけでなく、構成要素どうしの関係性のネットワークが、全体と言う新たな存在をいかに発生させ、維持させるかという視点から、構成要素の関係性のネットワークに注目する。

社会・経済状況の分析  
ビジネス環境の掌握

企業・事業戦略の立案  
概略ビジネス・モデリング  
仮想経営による仮説検証で戦略の選択

ビジネス・プロセスの設計  
詳細ビジネス・モデリング  
ビジネス・プロセスと構造の仮説検証による選択

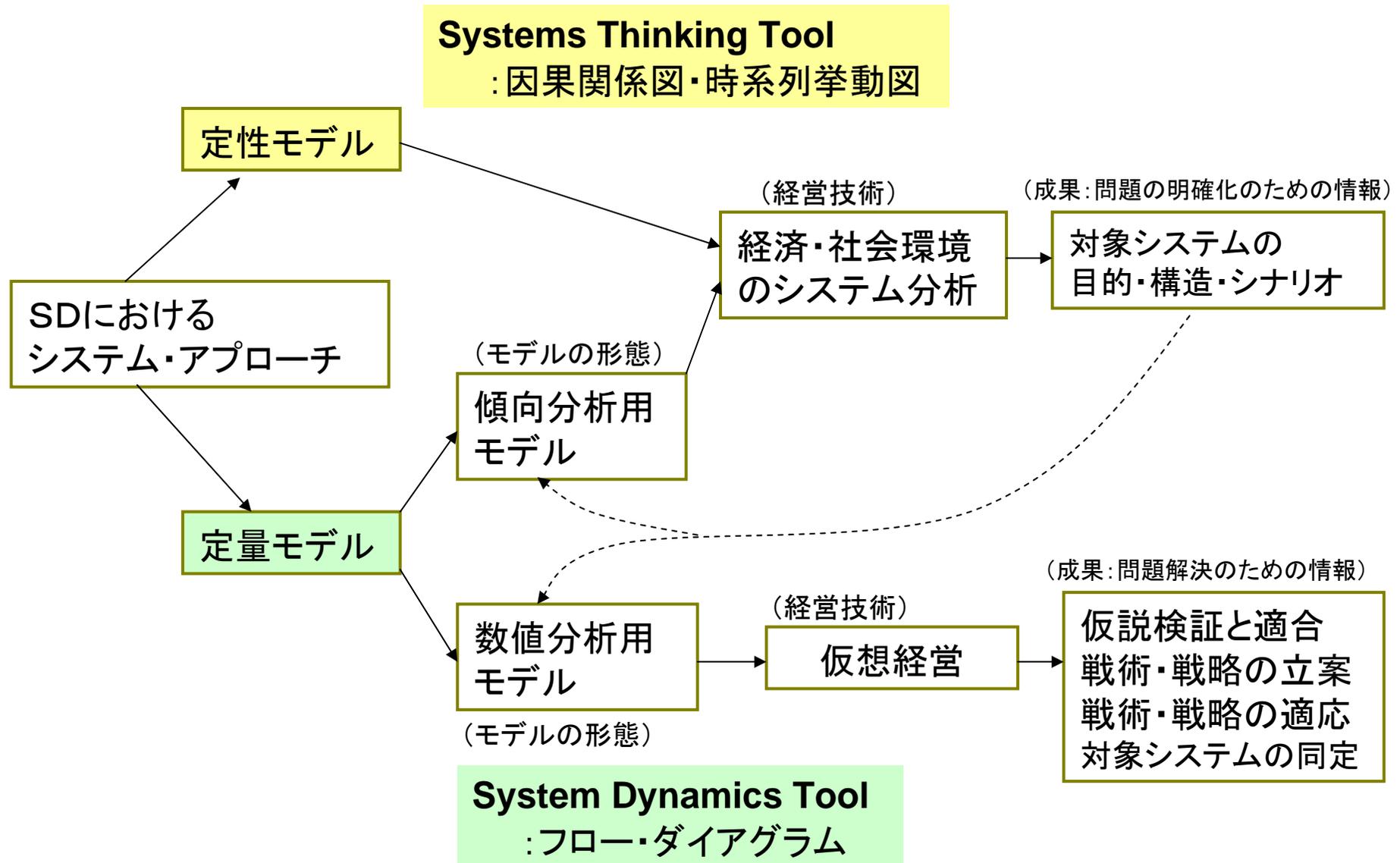
ビジネス・オペレーションの支援  
シミュレーションによる現状分析と戦術の選択

ビジネス経過の分析と変更・革新  
戦略とビジネス・プロセスの前提とした仮説検証  
変化したビジネス環境への適応

仮想経営における  
仮説検証

仮想経営による  
経営演習で  
マネージャの育成  
経営フライトシミュレーター

# SDにおけるシステム・アプローチの活用法

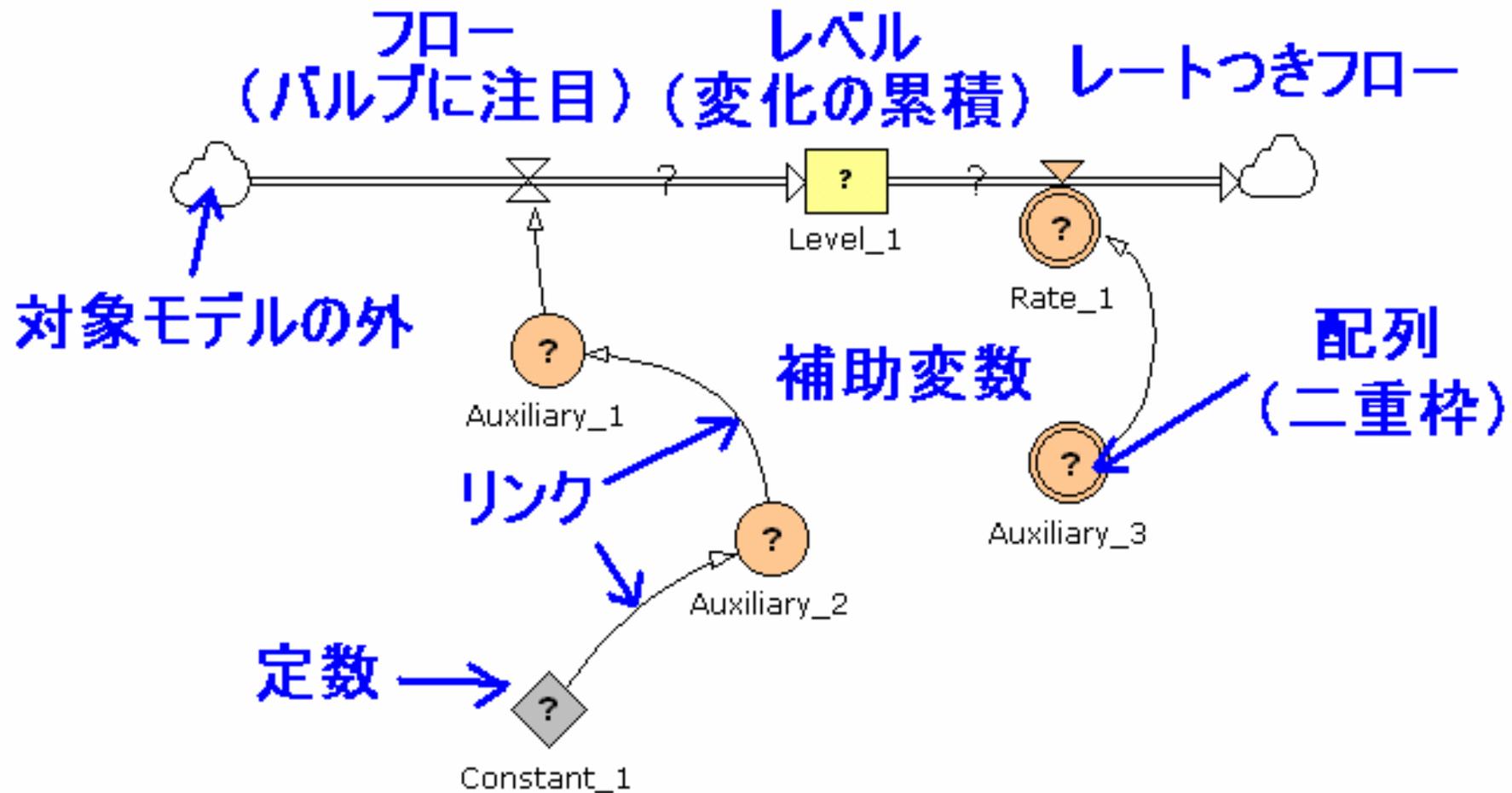


# SDの定性/定量モデルの経営における活用

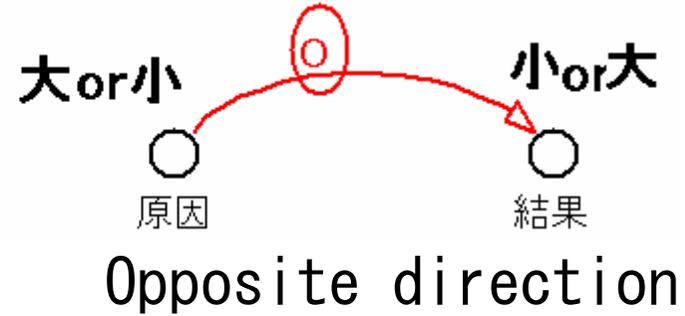
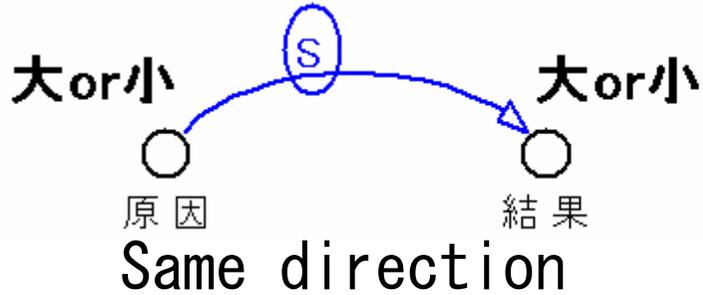


システム・アプローチ	Systems Thinkng	System Dynamics	
モデル	定性モデル	定量モデル	
SD活用の形態		傾向分析用	数値分析用
社会・経済状況の分析	◎	◎	
企業・事業戦略の立案	▲	◎	
ビジネス・プロセスの設計	▲	◎	▲
ビジネス・オペレーションの支援			◎
ビジネス経過の分析と変更・革新			◎
仮想経営における経営演習		▲	◎

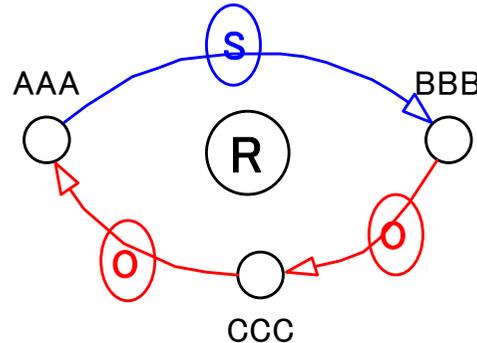
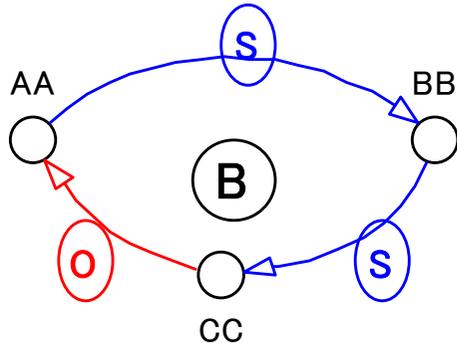
凡例 ◎:主に活用 ▲:副次的に活用



## 因果関係図

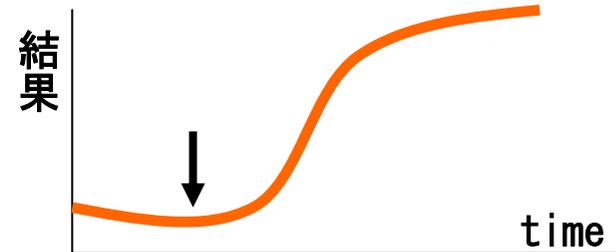
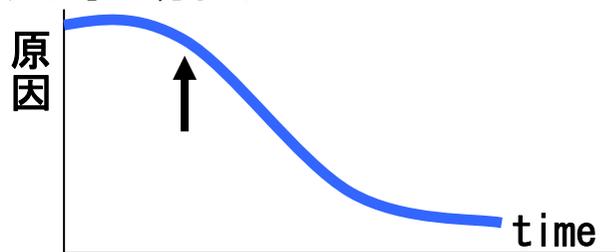


## フィードバック・ループ



⊙ B Balancing  
⊙ R Reinforcing

## 時系列挙動図



# 演習(1) CLDの適用 東京のタクシー減少



日本経済新聞 2008年11月22日

## 東京のタクシー 14年ぶり減少へ

規制緩和で増え続けてきた東京都内のタクシー台数が十四年ぶりに減ることが現実になった。関東運輸局の調べでは九月末は五万五千百四十一台と、半年前より七百四十三台減少。客足の不振でタクシー会社が減車に乗り出したうえ、政府が七

08年度

利用不振や規制強化響く

月から増車審査を厳しく円↓七百十円)の後、景したためで、二〇〇八年気後退もあって前年割れ度末に前年水準を上回るが続く。このためANZのは厳しい状況だ。

政府は〇九年度に新規板橋)が今年度内に四%参入などの規制をさらに減車し、飛鳥交通(同・強化する方針で、タクシー新宿)も年内に九%を削一会社の競争意識が薄減。日の丸交通(同・文れ、サービスが低下する京)は約三%の減車を実施した。タクシー需要の恐れがある。

都内のタクシーの運送多い都内が減少に転じた収入は昨年末の値上げことで、全国の台数も減少(中型初乗りで六百六十少に向かう見通しだ。

## タクシー業界の今後の対策

### (1) 目的

タクシー業界の現状を把握し、問題の打開策について、定性モデルを使って検討する。

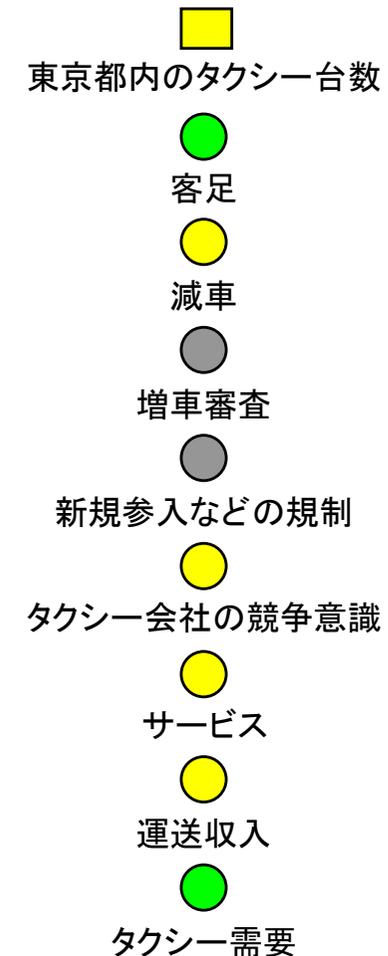
### (2) 方法

#### ①ステップ1

タクシー台数の増車／減車と客足の関係をCLDで表現する。

#### ②ステップ2

国交省は、「増車審査の厳格さ」を厳しくすることによって、「タクシー台数」と、「1台当たりの運送収入」が、どのような振る舞いをすると予測しているかを、時系列挙動図で説明する。



# 演習(1) CLDの適用 東京のタクシー減少



## ①ステップ 1

タクシー台数の増車／減車と客足の関係をCLDで表現する。

一般に、対象となるシステムは、過去の履歴に支配されたステータスを表すレベル要素と、現時点の状態を表すフロー要素などから構成される。

しかしCLDでは、それらを区別しないで、要素間の因果関係と遅れの関係のみを表現することが多いが、主要なレベルが明らかな場合には、それを表現するほうが分かり易いCLDが得られる。

ここで取り上げる対象では、タクシーの台数がレベルとして表現される。

まず、タクシーの台数に関係したレベルとフローとを以下のように配置して、それに関連して因果関係図を描く。

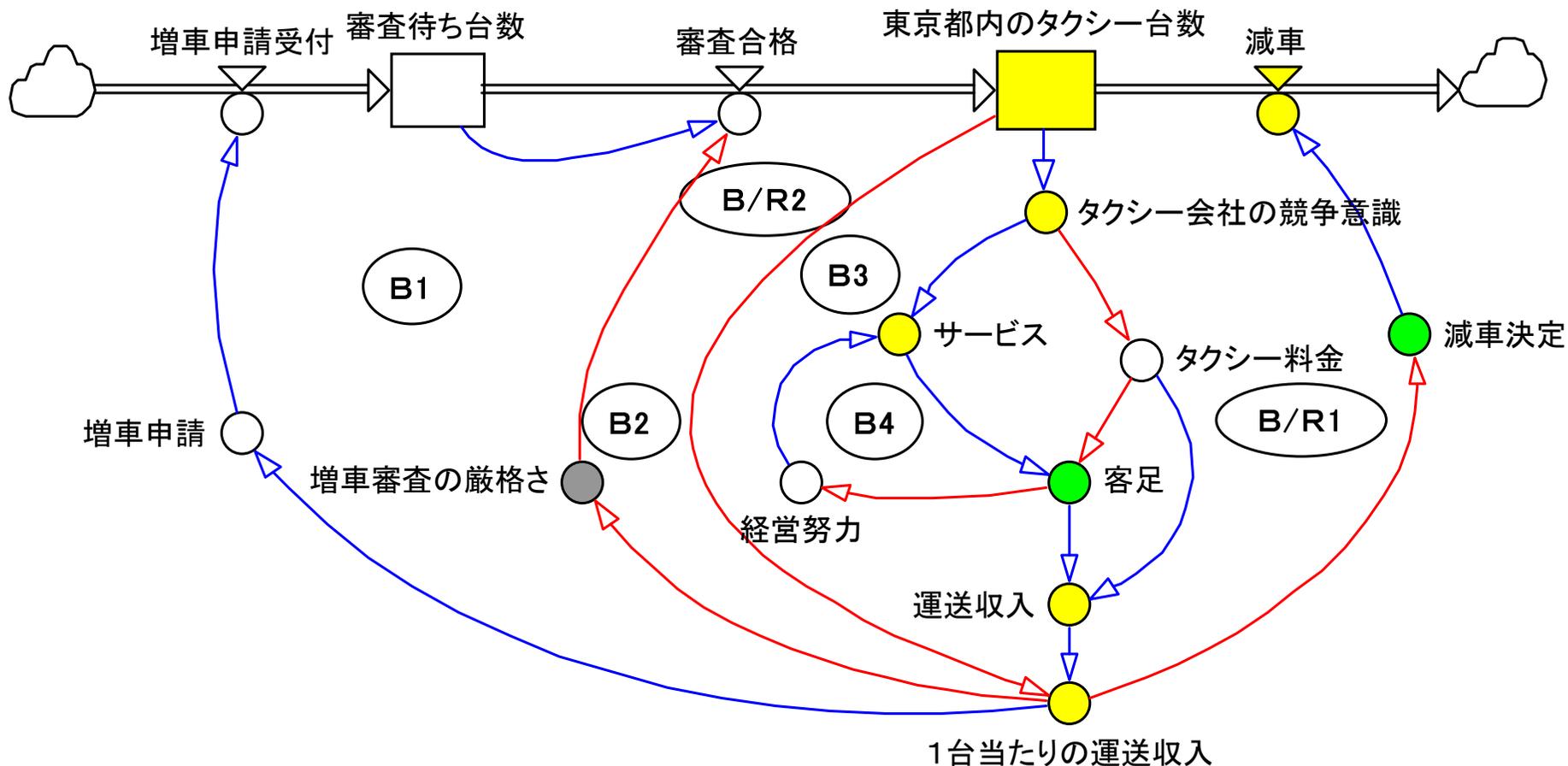


# 演習(1) CLDの適用 東京のタクシー減少



## ①ステップ 1

タクシー台数の増車／減車と客足の関係をCLDで表現する。



# 演習(1) CLDの適用 東京のタクシー減少



## ループの経路と特徴

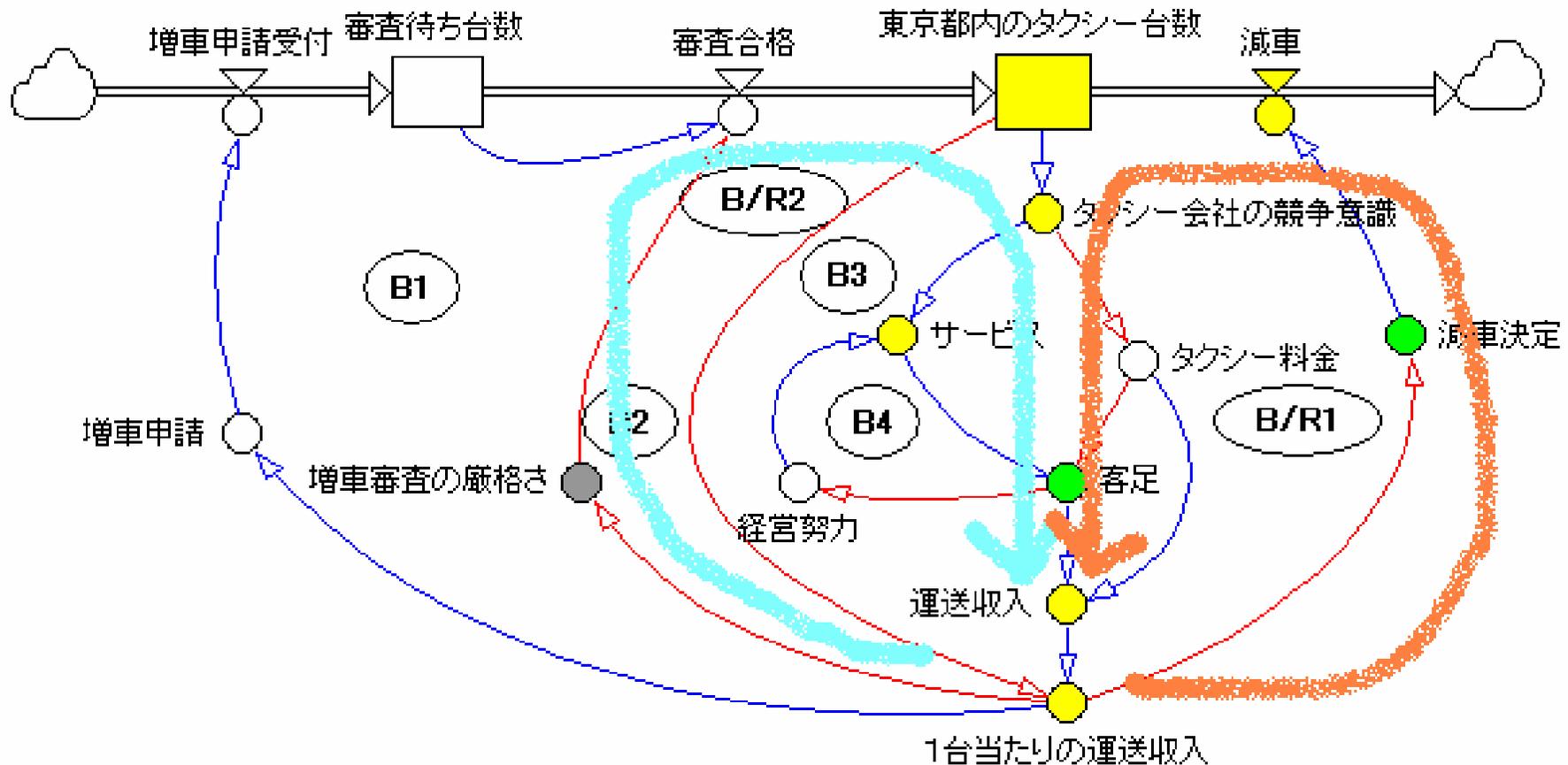
結論 : タクシー業界の経営の安定性において、B/R1と2が注目すべきループであると結論できる。

- B1 : 1台当たりの運送収入 → 増車申請 → 増車申請受付 → 審査待ち台数 → 審査合格  
→ 東京都内のタクシーの台数 →  
1台当たりの運送収入が減ってきたとき、台数増加で自社だけでも売上を確保しようとしにくいこと。
- B2 : 1台当たりの運送収入 → 増車審査の厳格さ → 審査合格 → 東京都内のタクシー台数 →  
国交省が1台当たりの運送収入が減ってきたとき、増車審査を厳格にすると、タクシー台数と運送収入は安定する。
- B3 : 1台当たりの運送収入 → 減車決定 → 減車 → 東京都内のタクシー台数 →  
タクシー会社が1台当たりの運送収入が減ってきたとき、多くの減車決定をすれば、タクシー台数と運送収入は安定する。
- B4 : 客足 → 経営努力 → サービス  
客足が減れば、サービスを向上させるという一般的な商習慣である。
- B/R1 : 1台当たりの運送収入 → 減車 → 東京都内のタクシー台数 → タクシー会社の競争意識  
Bの場合 : → タクシー料金 → 運送収入 →  
Rの場合 : → タクシー料金 → 客足 → 運送収入 →  
運送収入は、客足の落ち方によって決まるために、社会・経済環境の中での料金改訂に対する客足の予測がタクシー業界の安定性の鍵を握る。
- B/R2 : 1台当たりの運送収入 → 審査合格 → 東京都内のタクシー台数 → タクシー会社の競争意識  
Bの場合 : → タクシー料金 → 運送収入 →  
Rの場合 : → タクシー料金 → 客足 → 運送収入 →  
運送収入は、客足の落ち方によって決まるために、社会・経済環境の中での料金改訂に対する客足の予測がタクシー業界の安定性の鍵を握る。

# 演習(1) CLDの適用 東京のタクシー減少



## タクシー業界の安定性の鍵を握る B/R-1とB/R-2

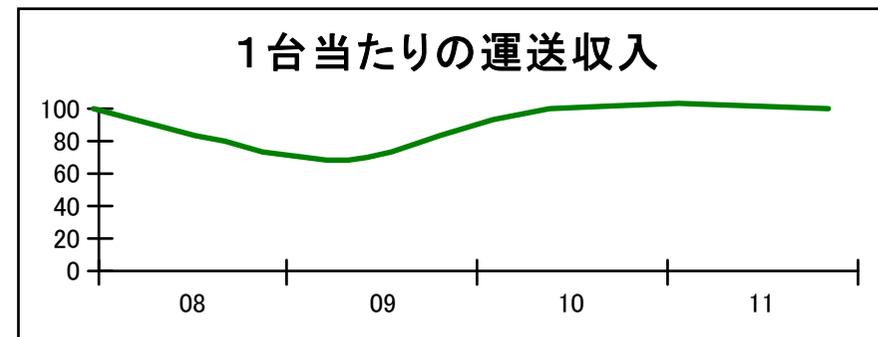
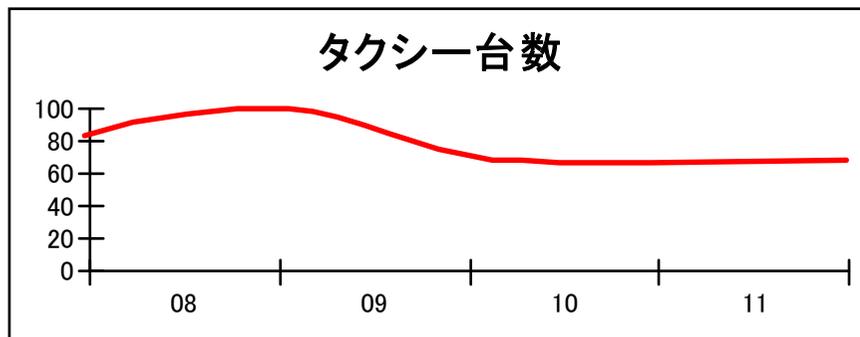
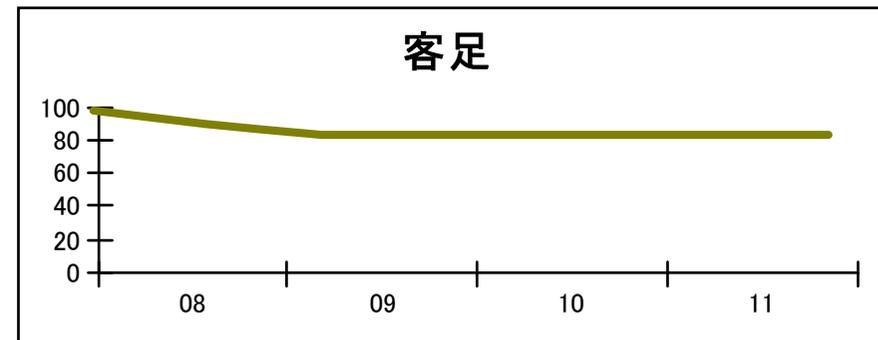
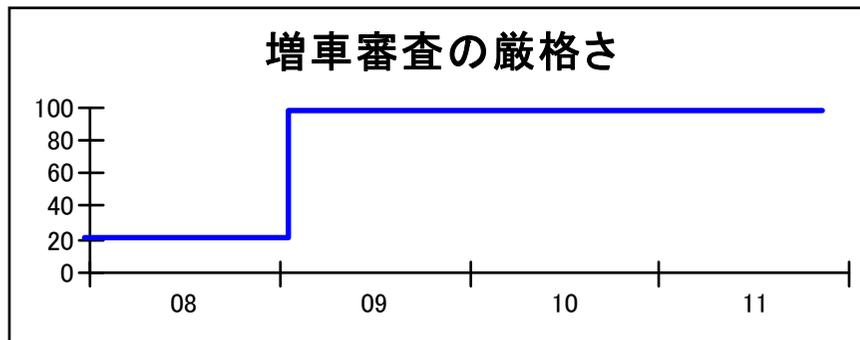


# 演習(1) CLDの適用 東京のタクシー減少



## ②ステップ 2

国交省は、「増車審査の厳格さ」を増すことによって、「タクシー台数」と、「1台当たりの運送収入」が、どのような振る舞いをすると予測しているかを、時系列挙動図で説明する。



定量的なシステム・アプローチの手法

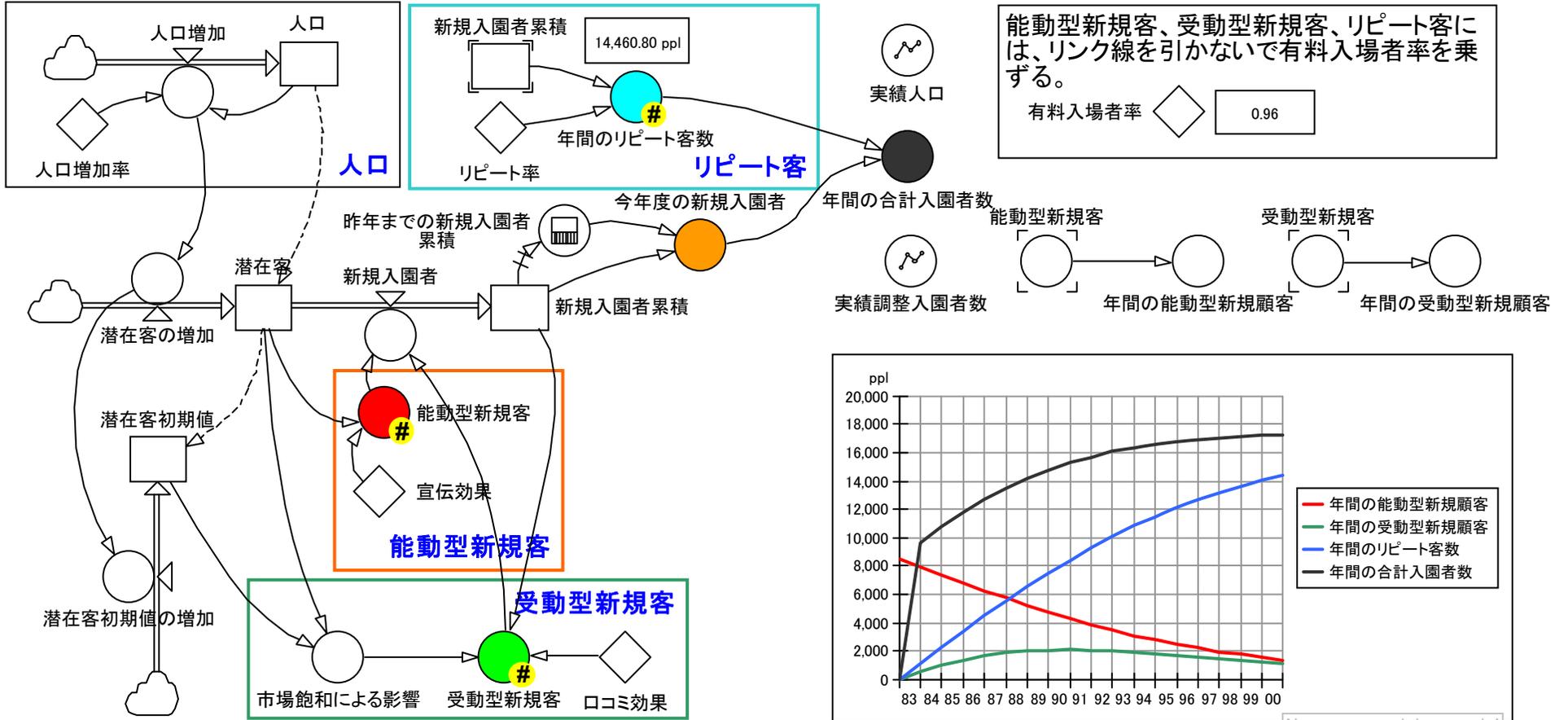
# System Dynamics “Ps Studio” の概要

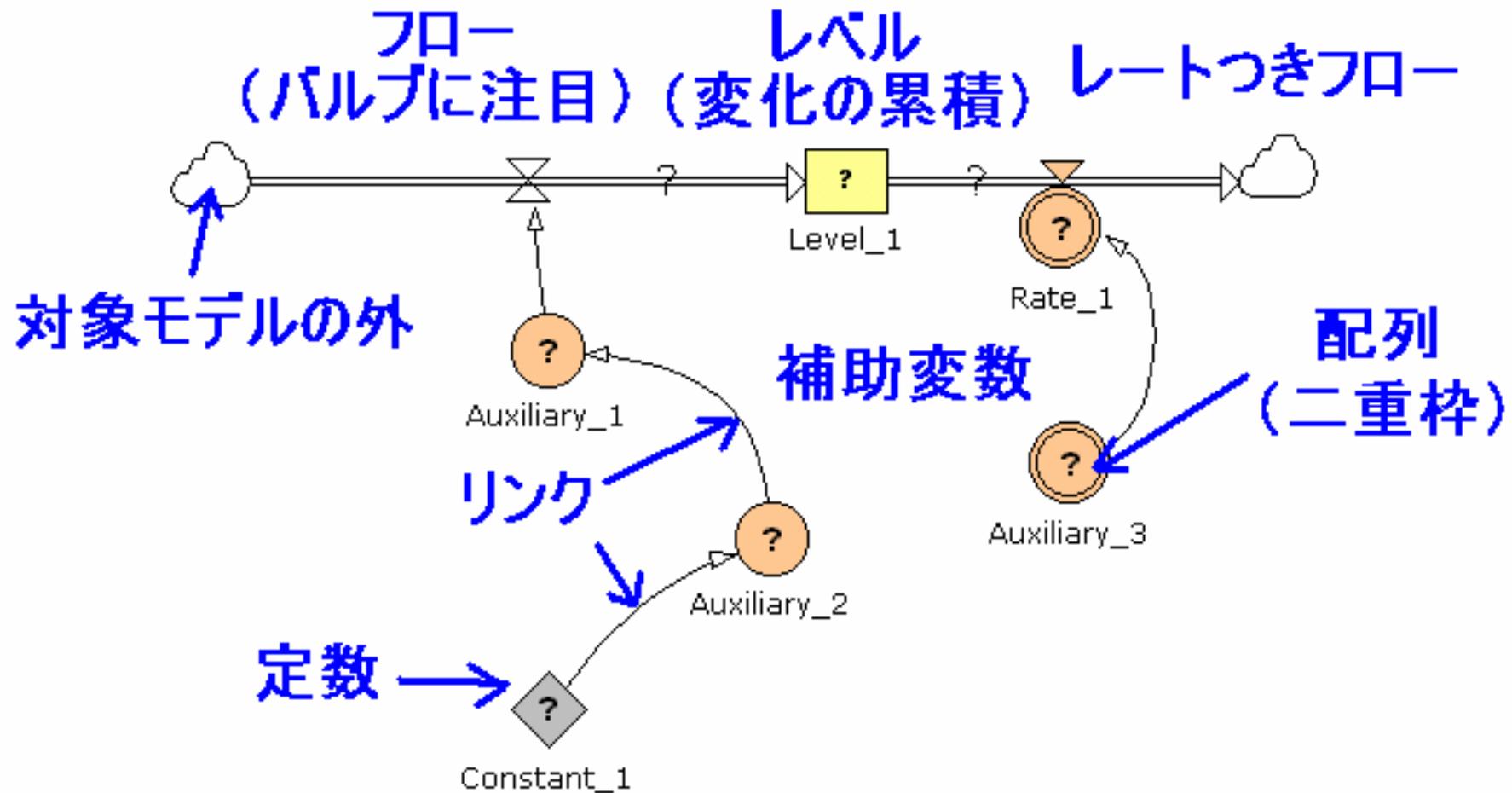
SDは、制御理論を社会系の問題に適用  
表記法としては、フロー・ダイアグラム

# TDLの入園者モデル



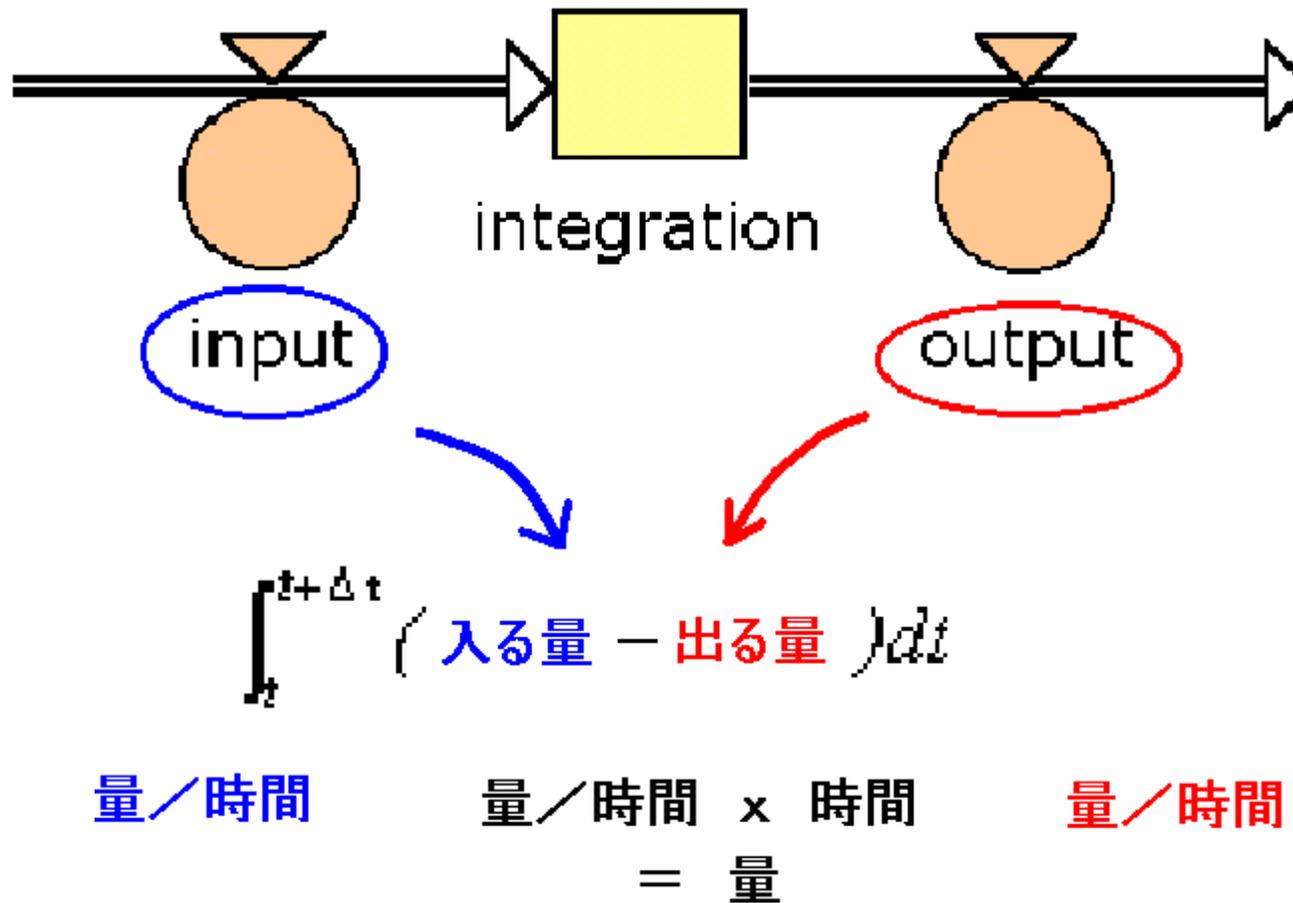
## ディズニーランドの入場客の分析





# レベルとフローによる積分

## → 積分の図式表現



# Ps Studio 7 新プロジェクト開始のウィザード



新しくプロジェクトを始める時、基本的な下記の事項を設定するための Wizardである。メニューバーのファイルの”New”またはツールバーの”New Simulation Project”アイコンから立ち上げる。

## ① Language

一般には日本語を選択

## ② Compatibility

Studio2005でも実行できる条件を選択(但し、機能は限定)

## ③ Calender Dependency

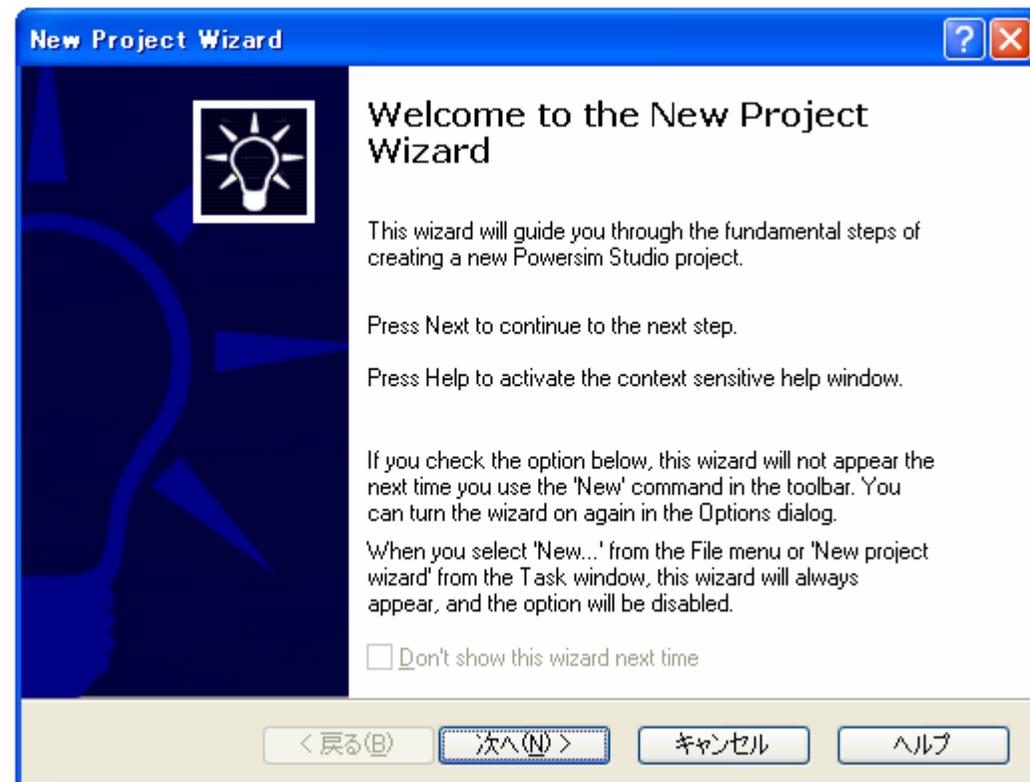
暦に帰属しない相対的な時間軸を選択

## ④ Calender & Time Unit

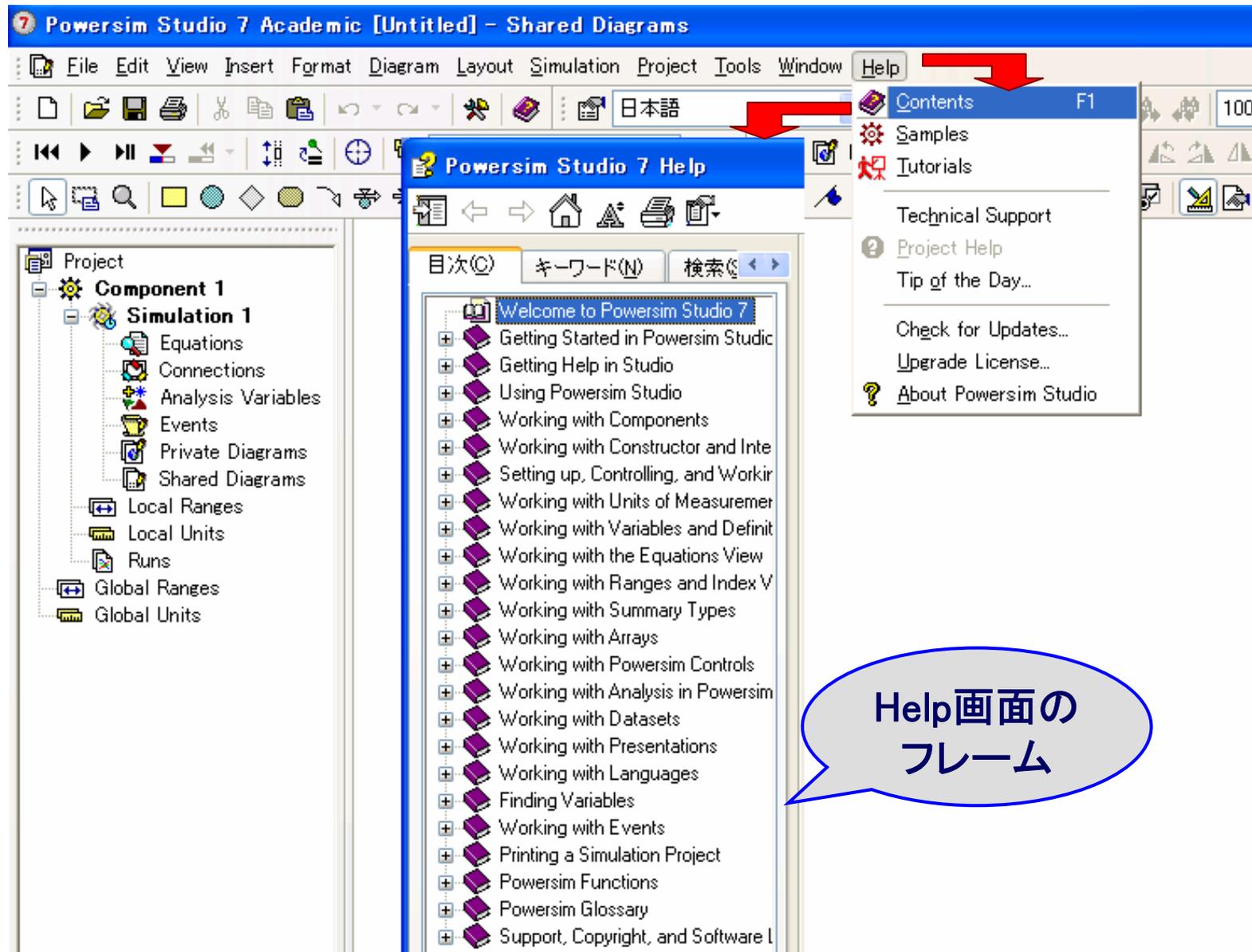
Project Settingと同じ

## ⑤ Simulation Time

Simulation Settingと同じ



# Ps Studio 7 オンライン・ヘルプの活用



Help画面の  
フレーム

機能については以下を参照して下さい。

## 1. Ps Studio 7 オンライン・ヘルプの活用

目次／キーワード／検索(探したい語句)

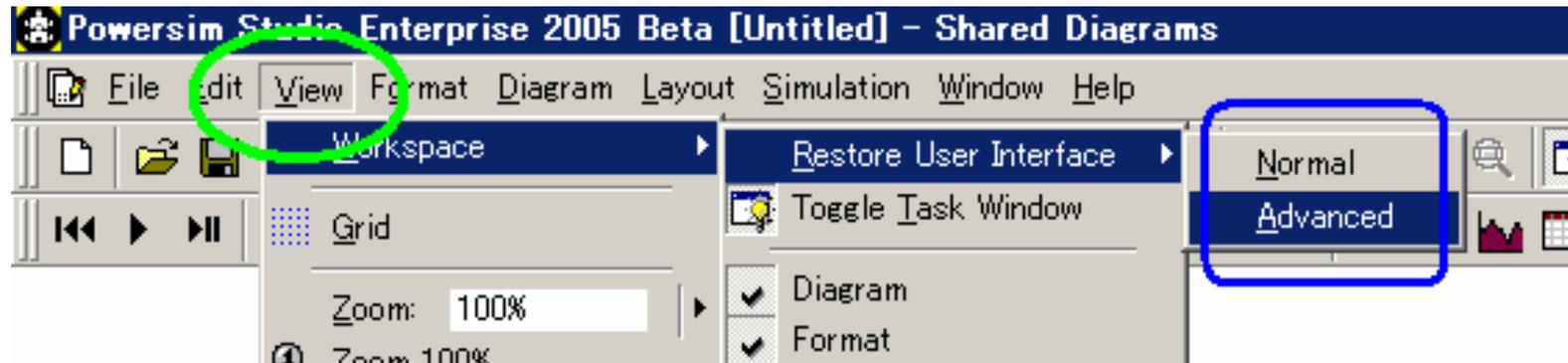
## 2. POSY社からの電子情報

<http://www.posy.co.jp>

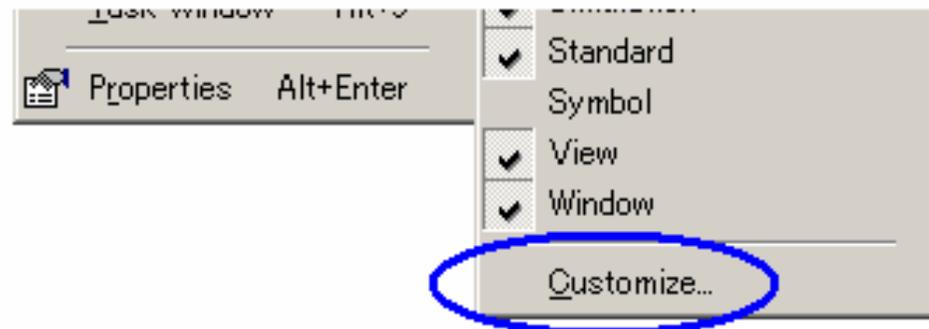
フレームの「Studioマニュアル」

⇒ Ps Studio 7/2005/2003 簡易マニュアル

# Ps Studio 7 でのトップ画面の選択



View⇒Restore User Interface⇒Normal/Advanced  
⇒Customize⇒コマンド毎の表示/非表示



- Normal : 簡単な一つのモデルを構築するための画面
- Advanced : 複数のモデルでプロジェクトを形成するための画面
- Customize: メニューバーとツールバーをカスタマイズできる機能

# Advanced :プロジェクト・ウィンドウ



The screenshot shows the Powersim Studio Enterprise 2003 interface. The main window is titled "Project 'SD&&インストール...'". The menu bar includes File, Edit, View, Insert, Format, Diagram, Layout, Simulation, Project, Window, and Help. The toolbar contains various icons for file operations and simulation control. The Project Window on the left shows a tree view of the project structure, including Component 1, Simulation 1, Equations, Co-models, Events, Private Diagrams, Shared Diagrams, Local Ranges, Local Units, Runs, Global Ranges, and Global Units. The Shared Diagrams window is open, displaying a list of terms and their definitions in Japanese. The diagram tab at the bottom is labeled "Diagram 1".

Project : 一つのファイルとして保存する複数のモデル (Component) 群とその関連情報

Component : 一つの独立したモデル

Simulation : 各Componentは計算条件を変えた複数のシミュレーションを実行できる

Equation : シミュレーションモデルのダイナモ方程式

Event : シミュレーションをインタラクティブに実行するための割り込み

Private Diagram : 各シミュレーションに個別のダイアグラムで、  
ユーザーインターフェースのみ構築可能

Shared Diagram : 各Componentの下で共有なダイアグラムで、  
あらゆる構造を構築可能

Run : 実施されたシミュレーションの実行結果

Global Range : 配列の次元の定義に使われる列の定義

Global Unit : 数量の単位

メニューバー

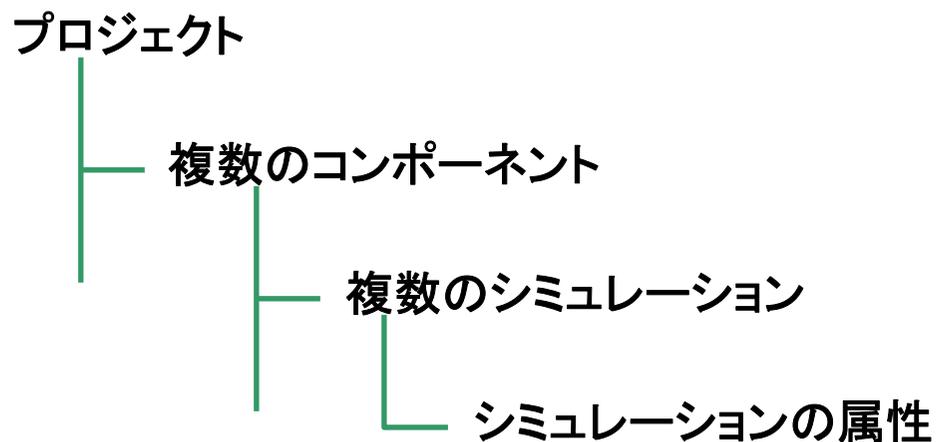
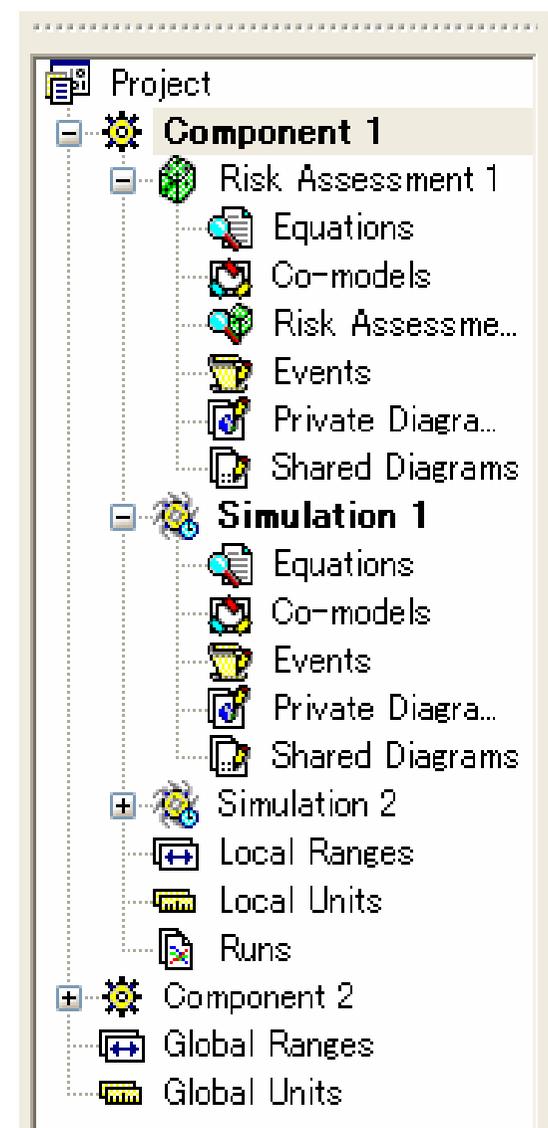
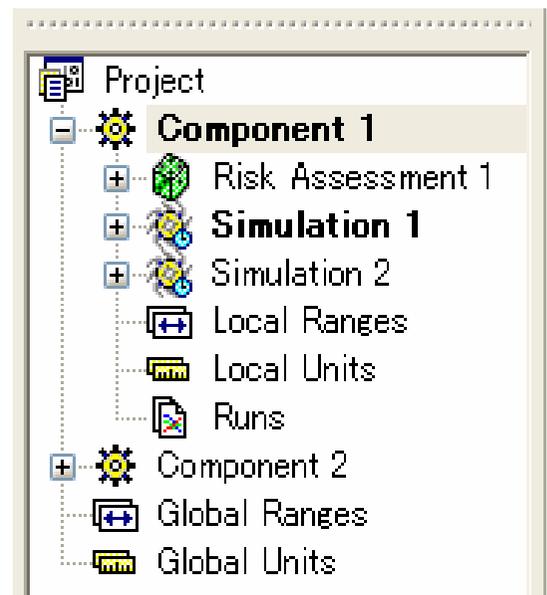
ツールバー

プロジェクト・ウィンドウ

ダイアグラム

ダイアグラム・タブ

# Advanced :プロジェクトウィンドウの構造



# メニューバーとツールバー



**メニューバー:** File, Edit, View, Insert, Format, Diagram, Layout, Simulation, Project, Window, Help

**メニュー項目:** プロジェクト条件設定, シミュレーション条件設定, 使用言語 (マイクロソフトの多言語対応), プレゼンテーションモードのテスト

**ツールバー (上部):** プロジェクトウィンドウの表示/非表示, 変数名の表示/非表示, シンボル検索

**ツールバー (下部):** グラフの変数軸の自動調整, 定数とレベルの固定の解除, スナップショット (ダイアグラム内のシンボルのコピー), 現ダイアグラムへ既存変数の持込, モデルのサブモデル化, ゲージ, スライダー, スイッチ回路, ダイアグラムの設定

**ツールバー (下部):** 実行, リセット, 1ステップずつ実行, 標準カーソル, イメージコピー, 左ボタン+囲う, レベル, 補助変数, 定数, サブモデル, フロー, レート付フロー, 情報リンク線, フレーム, 自由曲線, バックマーク, ハイパーリンク, テーブルコントロール, チャートコントロール, 時系列テーブル, 時系列グラフ, デザインモード/プレゼンテーションモード

# シミュレーション条件の設定



The screenshot shows the 'Simulation Settings' dialog box with the 'Integration' tab selected. The 'Time Settings' section includes a 'Calendar' dropdown set to 'Bank', a 'Timestep' of '1.00 da', a 'Start Time' of '2004/01/01', and a 'Stop Time' of '2005/01/01'. The 'Simulation Speed' section has 'Maximize Speed' selected and a speed of '1,000 ms/Time Step'. The 'Simulation State' section shows the 'Time' as '2004/01/01'. A callout box explains the settings: '設定の意味: カレンダーはバンク暦(1月=30日、1年=360日) シミュレーション計算の刻み時間=1da=1日 計算開始期日(Starttime)=2004/01/01 計算終了期日(Stoptime)=2005/01/01 計算の速さ=このPCの最大の速さで'. A red arrow points from the 'Integration' tab to the callout box. Below the dialog box, the text '積分法の選択' is written in red.

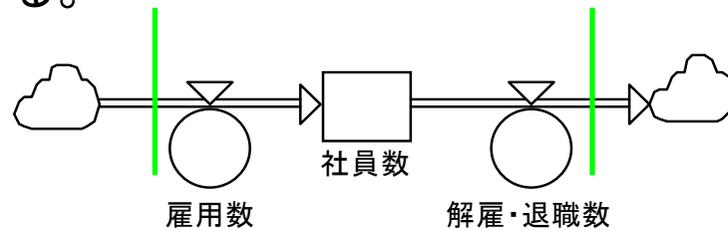
**積分法の選択**

設定の意味:  
カレンダーはバンク暦(1月=30日、1年=360日)  
シミュレーション計算の刻み時間=1da=1日  
計算開始期日(Starttime)=2004/01/01  
計算終了期日(Stoptime)=2005/01/01  
計算の速さ=このPCの最大の速さで

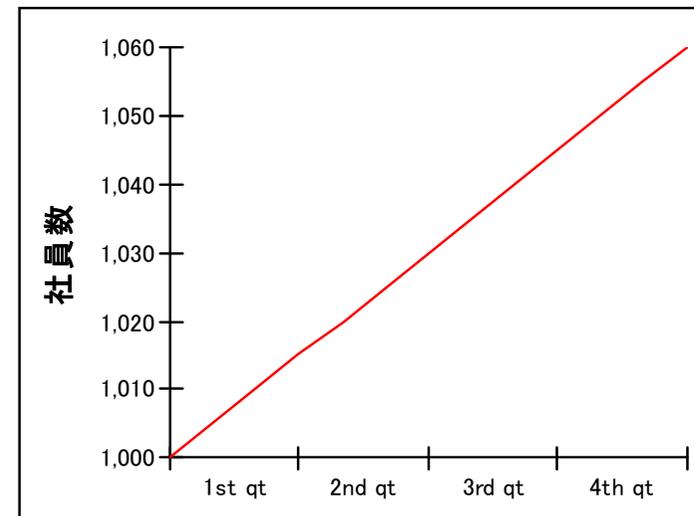
# 構築ダイアグラムの変数：レベルとフロー



社員の数は短時間では一定の値だが、ある程度の時間が経過したり、情報産業のような労働流動性の高い産業では、時間が短くても変動する。簡単なモデルを作ってみる。



当初の社員数 = 1000人  
雇用数 = 10人 / 月  
解雇・退職数 = 5人 / 月



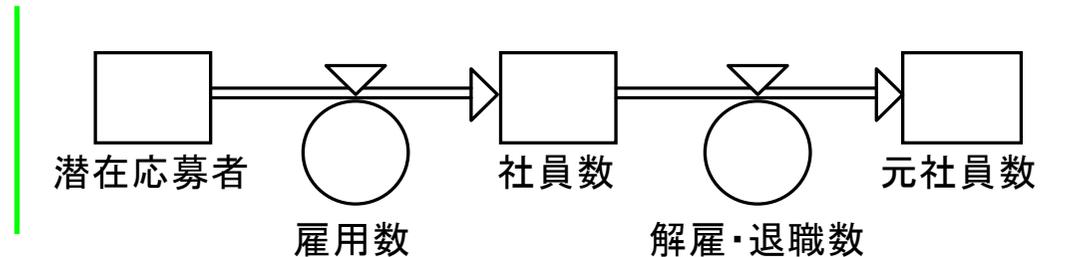
両端の雲の絵は、この構造のソースとシンクであり、無限大を意味する。ここでは、人材はいくらでもいて、解雇・退職に制限はないことを意味している。また、雲はモデルの外を意味する。したがって、ここでは緑の縦線の内側がモデリングの範囲である。

# 構築ダイアグラムの変数：レベルとフロー



しかし、人材の数に限りがあり、労働組合との関係で解雇・退職には受け皿を用意しなければならないとなったら、この雲の部分もモデルの中に含めて検討の対象にしなければならない。

すなわち、モデルの領域として、両端まで含める必要がある。このように、モデルの領域を設定する過程は、モデリングの初期段階で大変重要な作業である。



当初の社員数 = 1000人  
雇用数 = 10人/月  
解雇・退職数 = 5人/月  
潜在応募者 = 200人  
元社員 = 100人

# 構築ダイアグラムの変数：レベルの積分



SDモデルでは、システムの構造は数学的に表現される。

フローの累積はレベルに貯まる。

これはフローの値が積分されてレベルの値になることを意味している。

SDツールの積分法としては、棒状の面積を積算する**オイラー法**が先ず使われる。

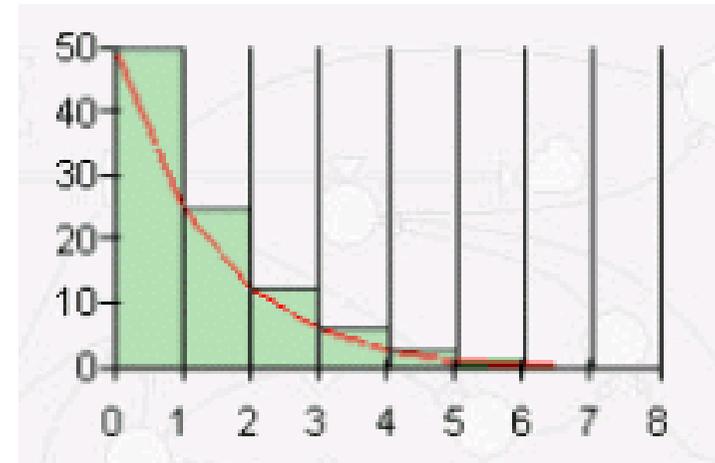
それ以外に高次の**ルンゲ・クッタ法**も選択できる。

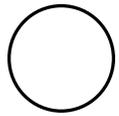
Studioでは、メニューバーのシミュレーションのプルダウンメニューからSimulation Settingを開くと積分法を設定できるタブがある。

オイラー積分法を図化して右に示す。

フローをレベルに接続すれば、ツール内部で自動的に積分式が定義される。

なお、モデルの中の変数であるレベル、補助変数、定数の方程式は、Excelの定義式によく似ている。

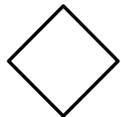




補助変数

## 補助変数

フローのバブルに接続した補助変数を特別にフローレートと呼ぶ。フローレートが、フローのバルブの絞り具合を操作して、流出入の量を決定する。この決定に至る諸計算を実施するのも補助変数である。補助変数間の情報伝達は情報リンクにより行なわれ、瞬時に情報が伝達される。

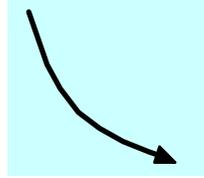


定数

## 定数

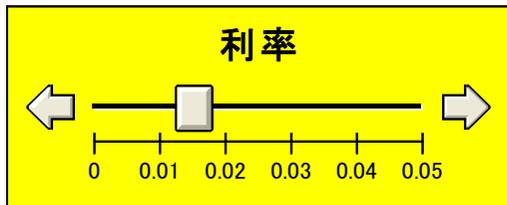
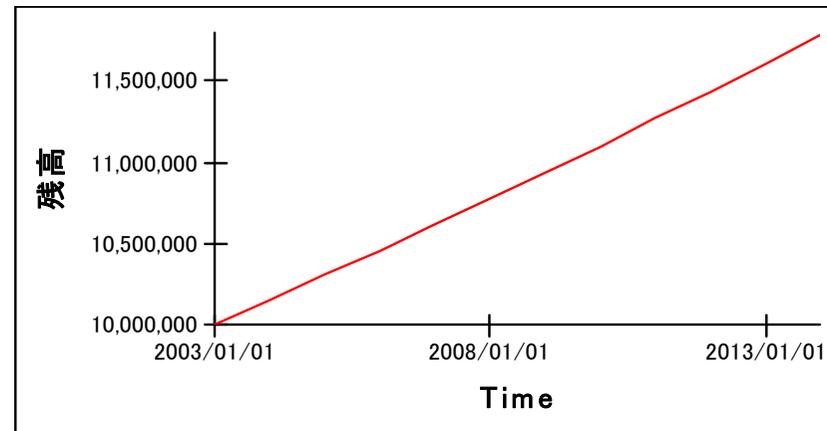
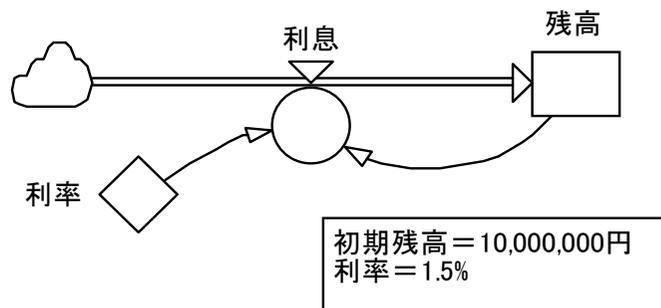
ひし形で表す定数は、シミュレーションの期間中一定な値を持つ。しかし、スライダーバー、テーブルコントロールなどからの入力により、シミュレーションの実施中に、その値を変更することができる。

# 構築ダイアグラムの変数：情報リンク



情報リンクは、通信線とみなされる。すなわち、フローと違って、情報が発信元から発信先に流れても、発信元の情報が増えることはない。

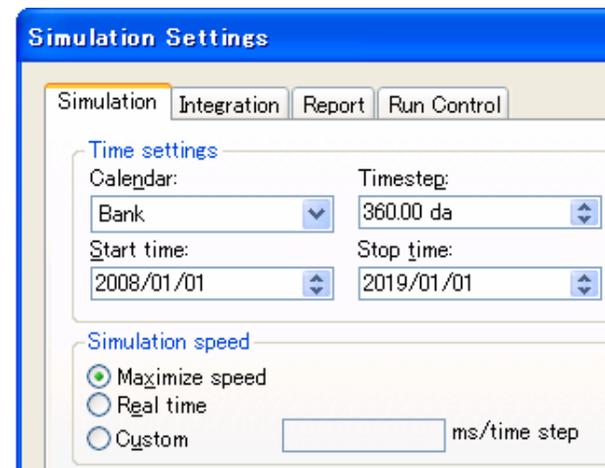
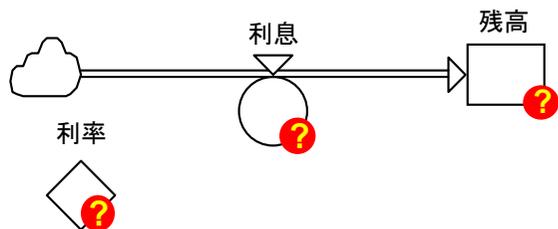
逆に、レベルにレート付きのフローが接続していても、レベルの情報はレートには伝わらない。情報を伝達するためには、情報リンクを新たに配置する必要がある。簡単な例として、定期預金の複利計算のモデルを以下に示す。



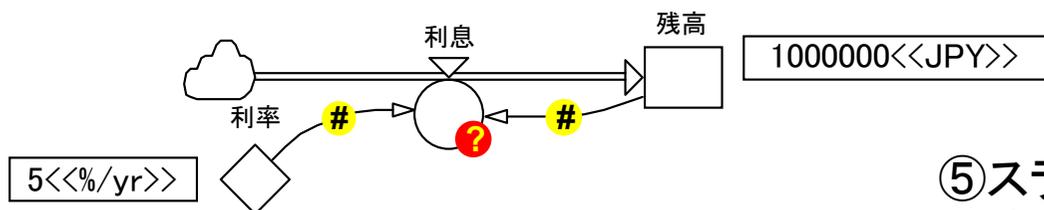
# 複利計算のモデルを作ってみよう



- ①レベルを配置⇒フローを配置⇒定数を配置      ④メニューバーのSimulationから Simulation Setting を開く

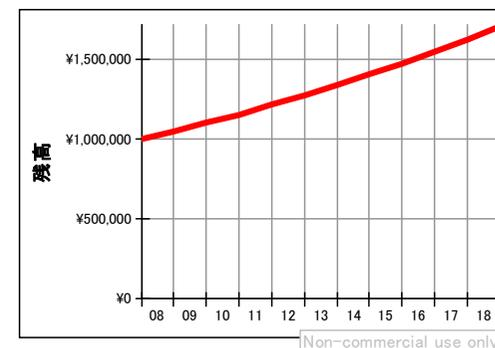
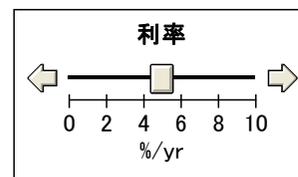
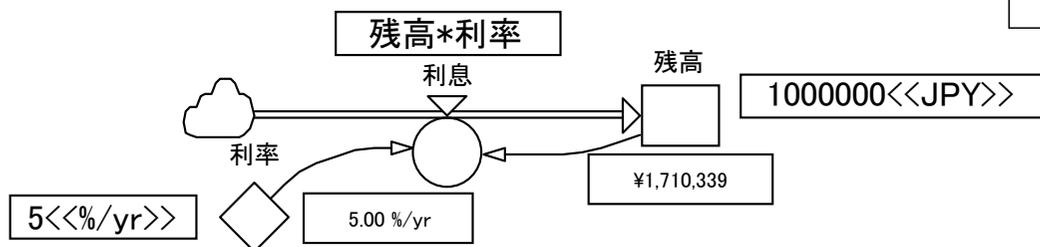


- ②リンク線を配置⇒利率を設定 ⇒残高の初期値を設定



- ⑤スライダーを設定して利率を引き込む 時系列グラフを設定して残高を引き込む

- ③利息の定義式を設定 ⇒利率と残高にShow Auto Reportを設定



# モデリング事始

## 卸販売モデル

# モデルを作ってみよう:卸販売モデルの説明



## 1. 概要

商品を仕入れて卸販売する簡略化した卸販売モデルを構築する。商品の種類は複数であるから最終的には配列による複数表現が必要であるが、このモデルは商品数が一つとして簡略化している。

シミュレーション期間は3年間とし、1日ごとに計算する。

## 2. 需要予測

卸販売先(顧客)の数は複数ある。

注文される需要数が在庫としてあれば販売するが、不足すればある分だけ販売する。

営業部員は注文される需要数を集計するが、顧客は1~2ヶ月の販売実績からその月の注文量を決めているようで、それらを集めた需要予測は、平均遅れ時間(情報遅れ時間)が1ヶ月(30日)程度あるようだ。

毎日の需要は、平均80個、季節変動(年周期)の振幅は50個のサイン関数、1週間周期の週間変動の平均振幅は10個のサイン関数で表現する。

最後の週間変動は、純粹の定常的なサイン関数ではなく乱れている。その乱れを毎日の平均発生数5個のポアソン分布による乱数を掛け合わせることで表現する。

ポアソン分布 :  $POISSON(5 \llbracket \text{product/da} \rrbracket, 0.89)$  ここで、0.89はseed

## 3. 在庫管理

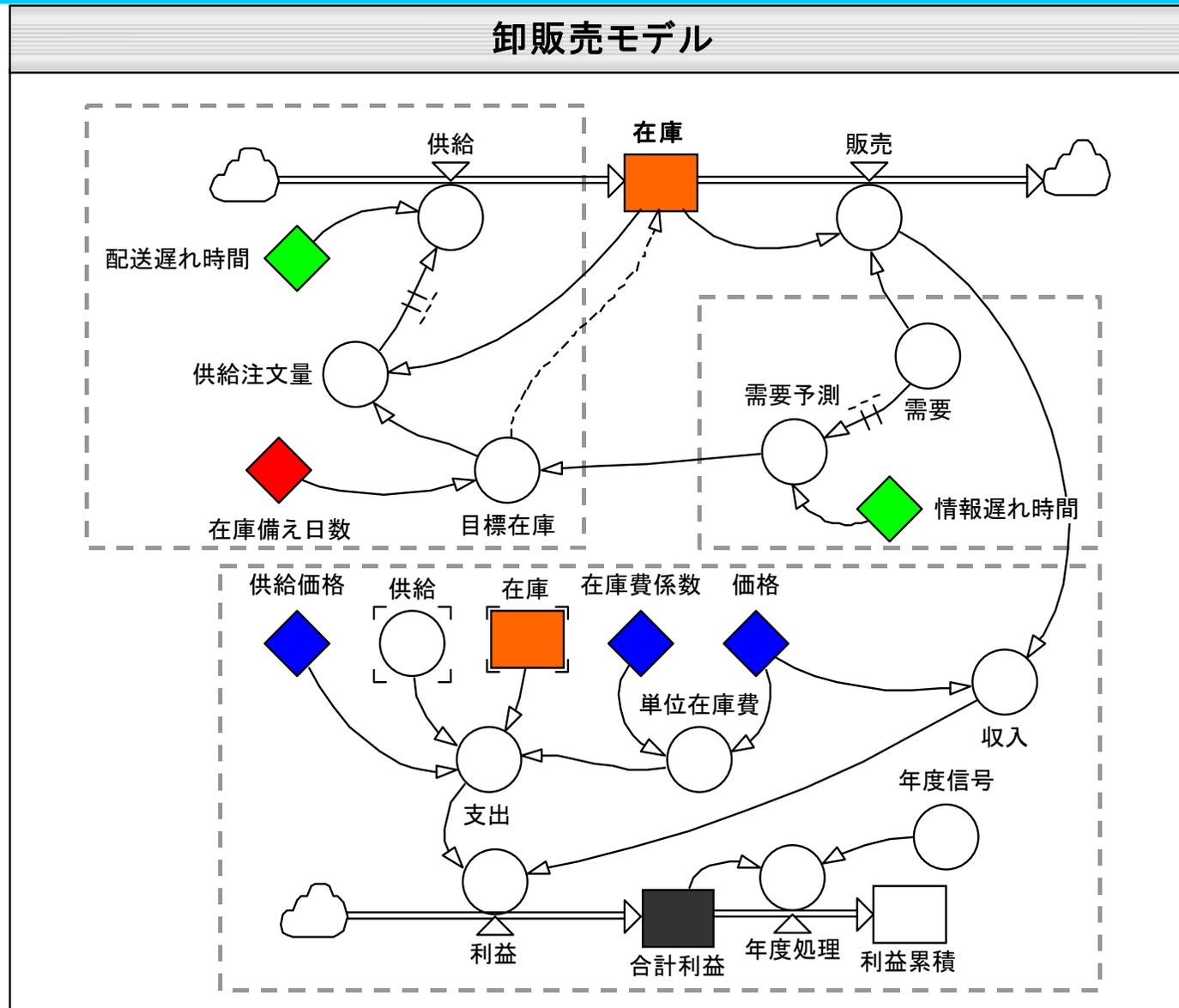
毎日の需要予測に対してその「在庫備え日数」倍した数を目標在庫数とする。目標在庫分を満たすように仕入れて倉庫に商品を供給するが、供給量を注文して実際に供給されるまでの平均遅れ日数は8日である。

## 4. 会計処理と利益

会計処理では、販売収入、在庫費、仕入れ原価により、粗利益を計算し、その累積を合計利益とする。

販売価格は1個100ドル、仕入れ価格は1個40ドル、在庫費用は1個につき販売価格の7.5%が毎日かかる。

# モデルを作ってみよう:卸販売モデルの説明

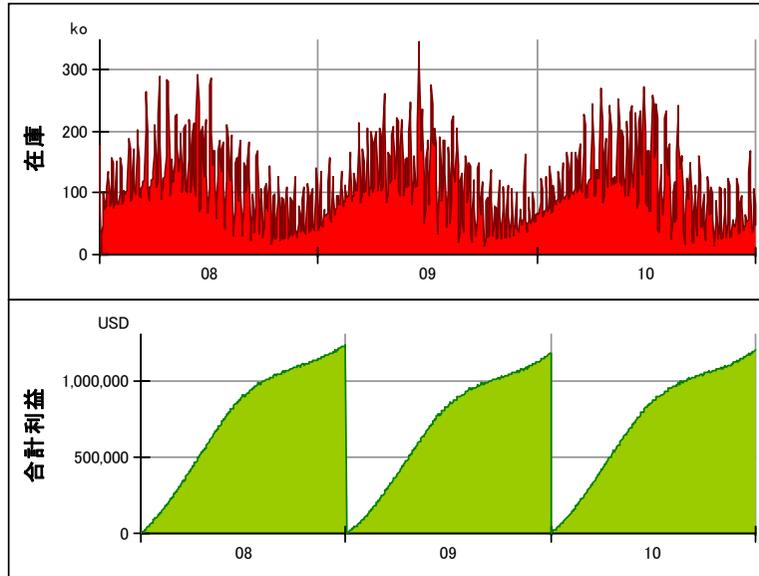


# モデルを作ってみよう:卸販売モデルの説明



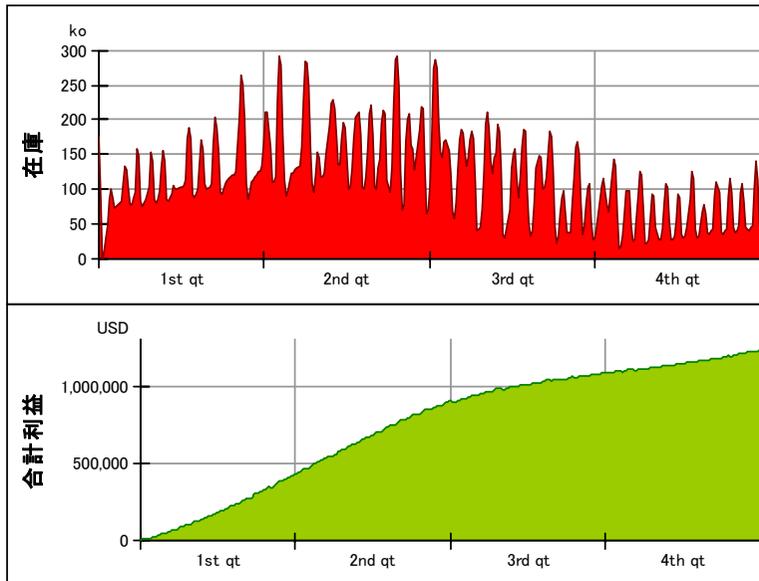
Name	Dimensions	Unit	Definition
供給		ko/da	DELAYMTR(供給注文量, 配送遅れ時間)
供給価格		USD/ko	40<<USD/ko>>
供給注文量		ko/da	MAX(0<<ko>>, 目標在庫 - 在庫)/TIMESTEP
価格		USD/ko	100<<USD/ko>>
利益		USD/da	収入 - 支出
利益累積		USD	0<<USD>>
年度処理.in			年度処理
単位在庫費		USD/(da*ko)	在庫費係数 * 価格
収入		USD/da	販売 * 価格
合計利益		USD	0
利益.in			利益
年度処理.out			年度処理
在庫		ko	目標在庫
供給.in			供給
販売.out			販売
在庫備え日数		da	2.2<<da>>
在庫費係数		%/da	7.5<<%/da>>
年度信号			TIMECYCLE(STARTTIME,1<<yr>>,1<<da>>)
年度処理		USD/da	IF(年度信号,合計利益,0<<USD>>)/TIMESTEP
情報遅れ時間		da	30<<da>>
支出		USD/da	在庫 * 単位在庫費 + 供給 * 供給価格
目標在庫		ko	需要予測 * 在庫備え日数
販売		ko/da	MIN(需要, 在庫/TIMESTEP)
配送遅れ時間		da	8<<da>>
需要		ko/da	80<<ko/da>> + SINWAVE(50<<ko/da>>, 360<<da>>) + SINWAVE(10<<ko/da>>, 7<<da>>) * POISSON(5<<1/da>>, 0.89)
需要予測		ko/da	DELAYINF(需要, 情報遅れ時間)

# モデルを作ってみよう: 卸販売モデルの説明



パラメーター	数値
価格	USD100.00 per ko
供給価格	USD40.00 per ko
在庫備え日数	2.20 da
在庫費係数	7.50 %/da
情報遅れ時間	30.00 da
配送遅れ時間	8.00 da

3年間のシミュレーション



パラメーター	数値
価格	USD100.00 per ko
供給価格	USD40.00 per ko
在庫備え日数	2.20 da
在庫費係数	7.50 %/da
情報遅れ時間	30.00 da
配送遅れ時間	8.00 da

1年間のシミュレーション

# 合計利益に及ぼす影響の感度分析



## 1. 設計変数

在庫備え日数 : 2.2 1.0 1.5 3.0

配送遅れ時間 : 8 1 3 16

情報遅れ時間 : 30 1 15 60

各変数の左端の数値が基準値である。

## 2. 感度分析

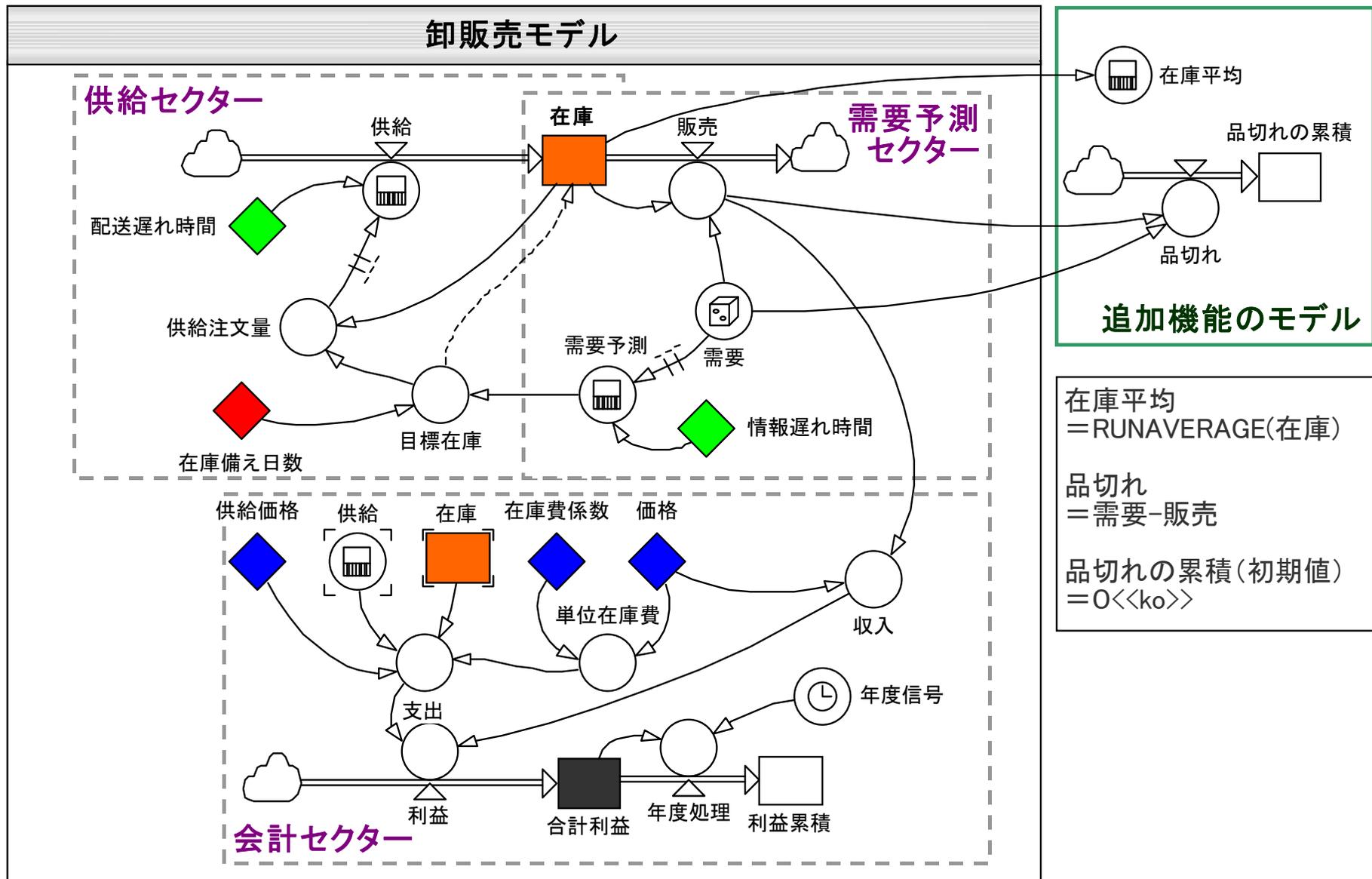
二つの変数は基準値にして、残りの変数を変化させて以下の評価変数を計測し、集計テーブルを作った後に、グラフ表示する。

評価変数 : 在庫平均 品切れ 1年後の合計利益

## 3. 考察

三つの評価変数それぞれに対して顕著な効果を及ぼす設計変数は何か？

# 感度分析のためにモデルに機能の追加



# 1年後の合計利益を最大化する“在庫備え日数”は？



1. 1年後の合計利益が最大になる在庫備え日数を1日から5日の間で求める。
2. その結果をRun Tableに登録して、referenceデータと設定する。
3. その結果をprivate diagram上の時系列グラフに表示する。
4. 最適として求めた在庫備え日数の値と異なる値を設定した場合に、合計利益がreferenceデータより小さいことを確認する。

Name	Value	Type	Apply Time	Deviation
Assumptions				
Decisions				
在庫備え日数	2.20 da		Start	
Minimum Value	1.00 da			
Maximum Value	5.00 da			
Objectives				
合計利益		Max	Stop	<input type="checkbox"/>

# 供給価格変動と在庫費係数変動の合計利益への影響は？

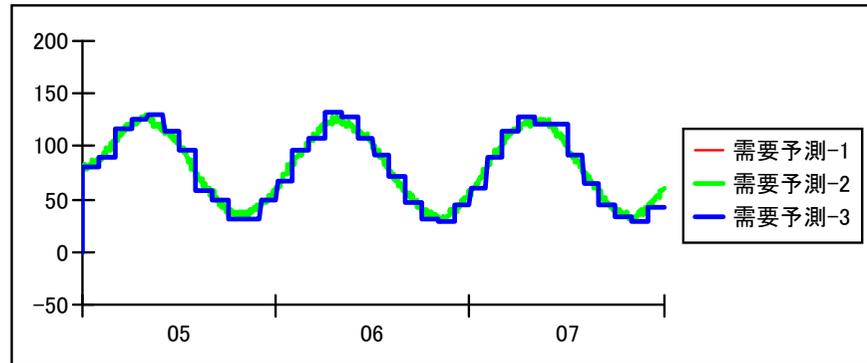
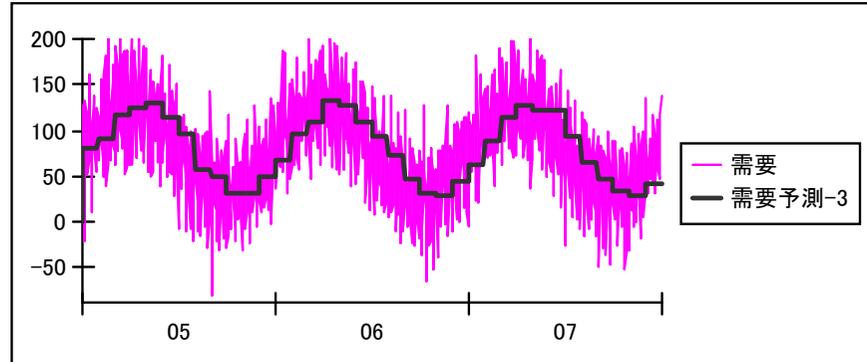
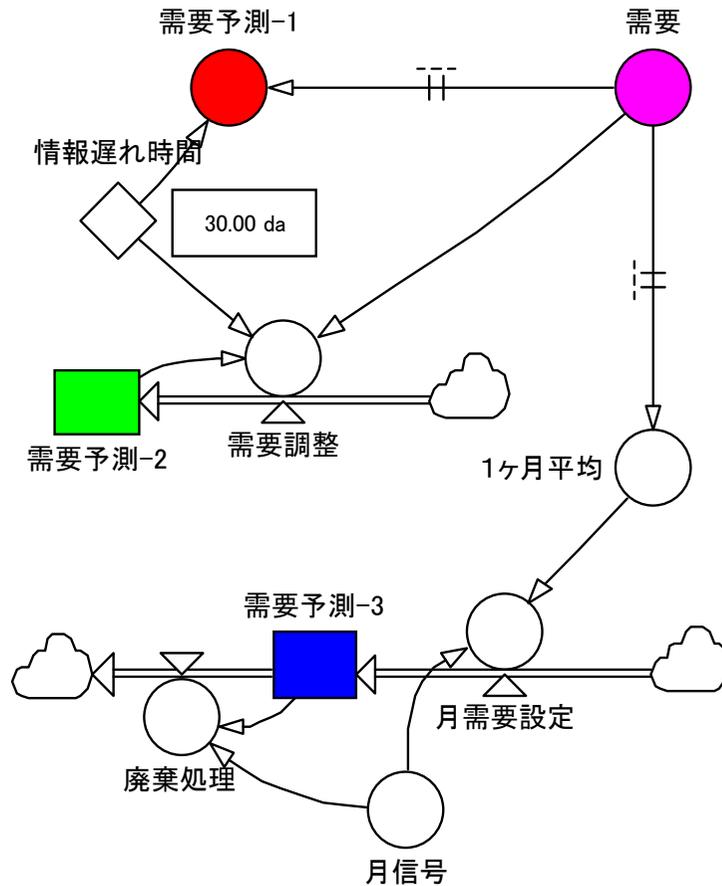


- 1年間の内に、供給価格と在庫費係数は以下の変動が予想される。  
供給価格 : 三角分布乱数 (最低 = 36USD、最大 = 48USD、ピーク値 = 40USD)  
在庫費係数 : 正規分布 (期待値 = 7.5%、標準偏差 = 1.5%)
- このとき、1年後に、10%、25%、平均、75%、90%の確率で得ることが期待できる合計利益を、Latin Hypercubeを100回実行して求める。

Name	Value	Type	Apply Time	Deviation
Assumptions				
供給価格		Triangular	Start	<input type="checkbox"/>
Minimum	USD36.00 per ...			
Maximum	USD48.00 per ...			
Peak	USD40.00 per ...			
在庫備え日数		Fixed Value	Start	<input type="checkbox"/>
Fixed Value	2.20 da			
在庫費係数		Normal	Start	<input type="checkbox"/>
Expected Value	7.50 %/da			
Standard Deviat...	1.50 %/da			
Decisions				
Objectives				
Effects				
合計利益		First		
Average				
Standard Deviat...				
10 Percentile				
25 Percentile				
75 Percentile				
90 Percentile				

最適化で求めた値を入力する。

# 需要予測の説明(1/2)



## 需要予測モデルの説明

### 需要予測-

3 : 月末に先月の需要(販売注文)の平均を求め、翌月の毎日の注文数とする場合の需要予測。

### 需要予測-1と-

2 : 毎日の需要が需要予測と違っていった場合に、そのギャップを情報遅れ時間(1ヶ月)で除して、その量だけの需要予測調整を行う場合の需要予測。

-1は情報遅れ関数(delayinf)を使う場合、-2は原始的モデル表現で表した場合である。

卸販売モデル(完成). sip需要の説明

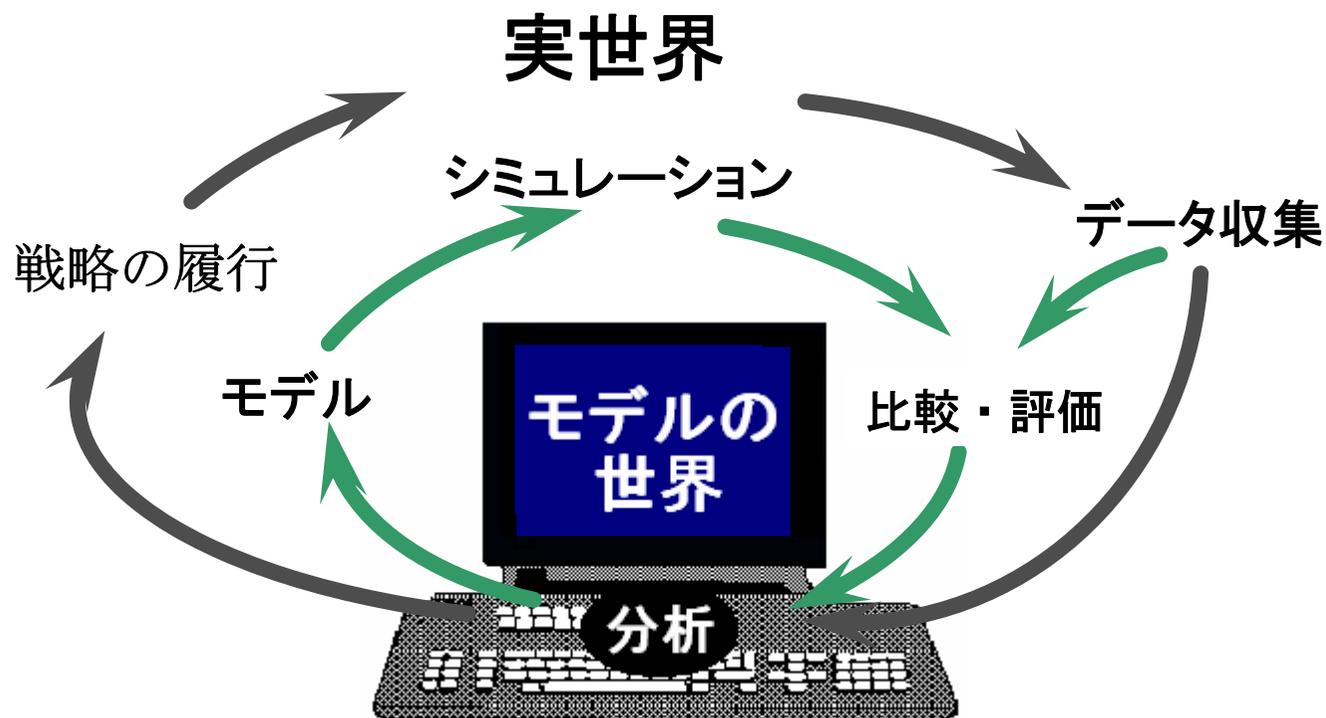
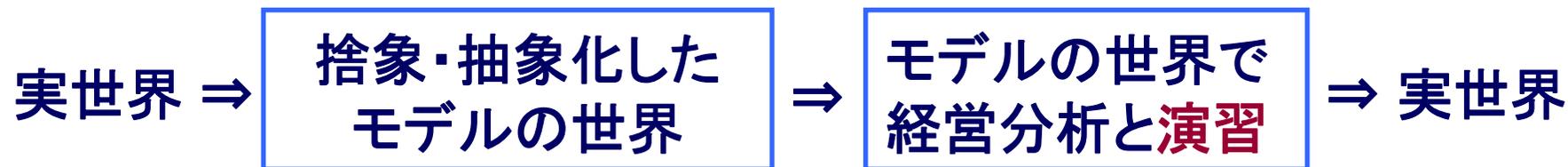
# 需要予測の説明 (2/2)



Name	Dimensions	Unit	Definition
廃棄処理		da <sup>-1</sup>	IF(月信号,'需要予測-3',0)/Timestep
情報遅れ時間		da	30<<da>>
月信号			TIMECYCLE(STARTTIME,30<<da>>,1<<da>>)
月需要設定		da <sup>-1</sup>	IF(月信号,'1ヶ月平均',0)/Timestep
需要			80+SINWAVE(50,360<<da>>)+SINWAVE(10,7<<da>>)*POISSON(5/1<<da>>)
需要予測-1			DELAYINF(需要,情報遅れ時間)
需要予測-2			80
需要調整.in			需要調整
需要予測-3			0
廃棄処理.out			廃棄処理
月需要設定.in			月需要設定
需要調整		da <sup>-1</sup>	(需要-'需要予測-2')/情報遅れ時間
1ヶ月平均			SLIDINGAVERAGE(需要,30<<da>>)

# モデリングを 適用する前提条件

# モデリング & シミュレーションの基本プロセス



Yes

No

対象についてメンタルモデルがあるか？

モデリングする意味がない。

↓  
モデルによる仮想経営で問題解決の可能性が  
あるか？

→ モデリング & シミュレーション以外の手法で問題解決を  
図る。

即ち、システム思考による問題の整理、部分モデルの組み合わせによる全体システムの挙動、仮説に基づくシステム構造に伴う挙動、初期条件および途中経過に対する条件変更に伴う挙動などの理解が問題解決に結びつく可能性があるか？

→ モデリング & シミュレーション以外の手法で問題解決を  
図る。

↓  
モデリング開始

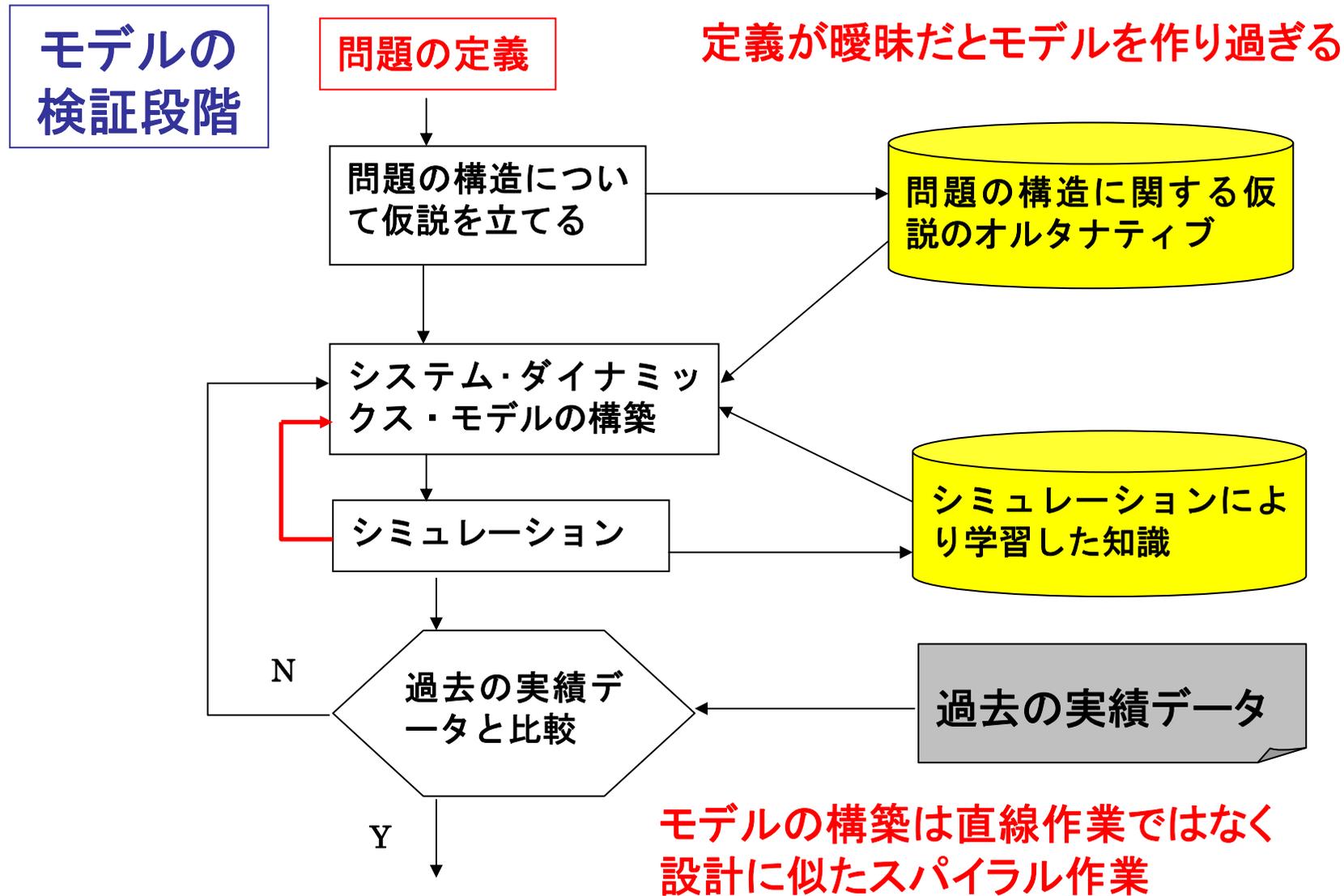
# モデリングの原則



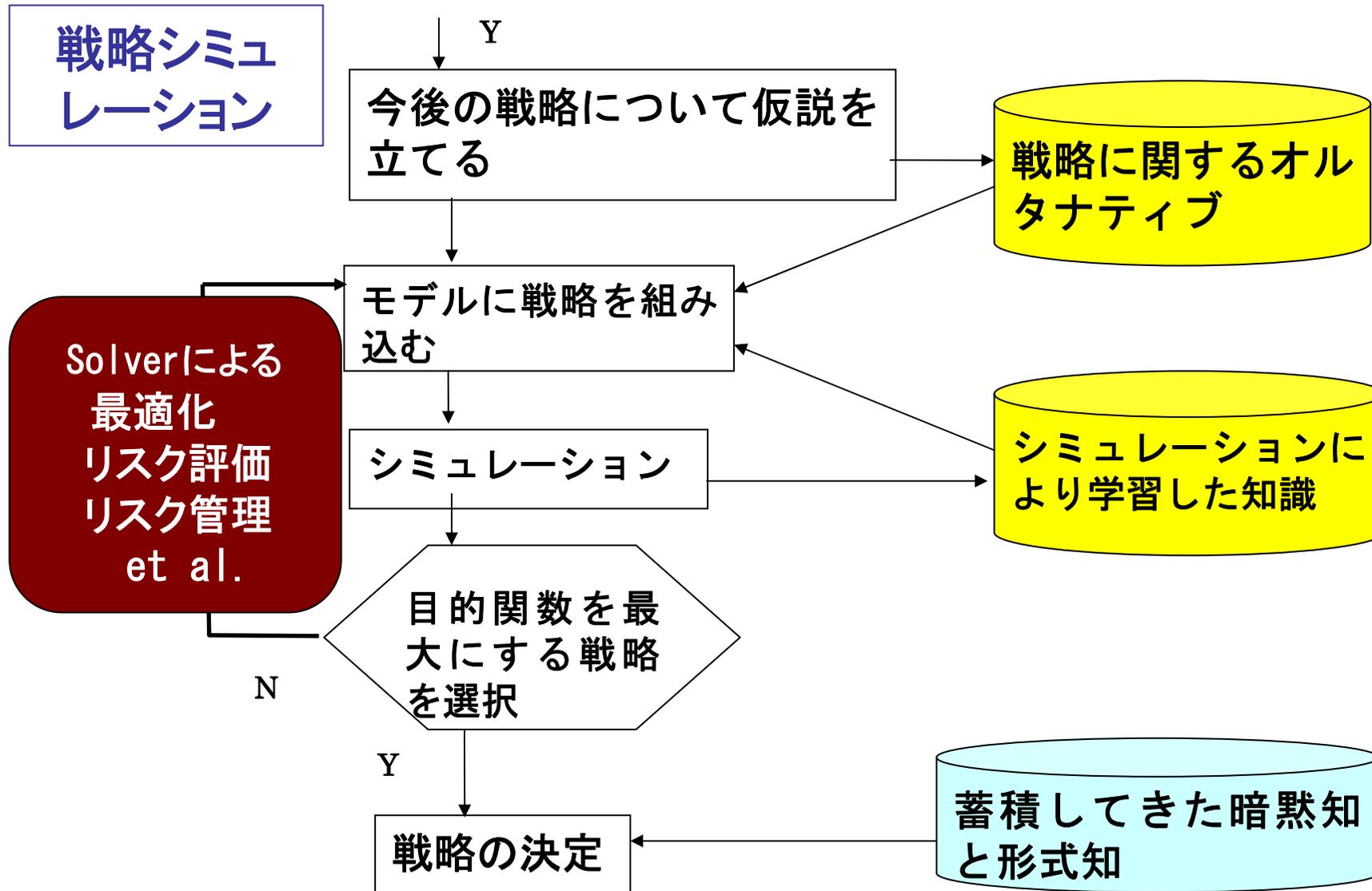
- ① 対象とするシステムを最もよく表現するステータスを最初に配置するレベルとする。  
ステータスとはフローではなく、過去に蓄積されたストックであり、そのシステムの価値表現に直接関係している要素のことである。
- ② 最初のレベル要素の周りにフロー要素を配置して、中心から外に向かって、モデルの構造を構築する。要素の名前は構築しながら記入する。
- ③ ダイアグラム1枚程度のまとめり毎に、要素のプロパティの中に挙動を定義する数式を入力する。  
定義の順は、最初に外側の定数、次にレベルの初期値、その後は、外から内側に向かって補助変数の数式を定義する。
- ④ 単位は、定数とレベルの初期値では入力するが、他では入力せず、Studioによる論理チェックを兼ねた単位の自動計算に任せる。  
ただし、プロパティ画面で、意識して単位を変更することは可能である。
- ⑤ 検証の終わっていないモデルは、実用には適用できない。  
部分的または全体的な検証を実施し、実用に当ってはその信頼の程度を説明できなければならない。

# ST/SDによる 問題分析と解決の 標準的プロセス

# 問題分析／解決の標準プロセス(1/2)



# 問題分析／解決の標準プロセス(2/2)



# “モデリング課題に取り組む第一歩”でのつまずき



モデリング & シミュレーションに取り組む第一歩として、以下に記す踏み出し方が効果的です。

## 準備1

自分が解決したい問題あるいはその一部をピックアップする。  
その課題が、SDの活用により解決できるかどうかを判断します。

## 準備2

SDに基づくモデリング & シミュレーションにより問題を解決できる可能性が高いと判断できたら、以下の基本方針を作成することから始めます。

## SDに基づくモデリング & シミュレーションの基本方針

- ① 問題は何か？
- ② 問題をどのように認識しているか？
- ③ 予想している解決策は？
- ④ 問題に関係する要素は？

# The END

松本 憲洋

**POSY Corp.**

〒102-0092 東京都千代田区隼町2-12-104

藤和半蔵門コープ 1F

**Tel.& Fax. 03-3512-5358**

**Mobile 080-5047-3849**

**matsu@posy.co.jp**

**<http://www.posy.co.jp>**