Understanding Compartmental Modeling:
Synthesis Model
Conversion Model
Flow Model
Exchange Model
All Processes Model

Compartmental Modeling:

The following five short lessons will teach you about compartmental modeling. At the end of the first four lessons you should be able to do the following:
(1) Identify the sources and sinks for each substance in a complex reaction.
(2) Write correct mass balance equations governing the system.
(3) Code these equations in JSim's Mathematical Modeling Language (MML) along with other necessary information.
(4) Compile them into a JSim model, run the model, and save it.

To test your skill, Lesson five asks you to write a new model which combines everything you have learned in the previous four lessons.

Lesson I: Introduction to compartmental modeling and using JSim
1.   A restricted compartmental definition which covers a few simple cases:
A compartment represents a physical volume, V. In the volume, there is an amount of material, Q. The concentration, C, is given by the amount, Q, divided by the volume,V. The amount, Q, is given by the volume, V, multiplied by the concentration.

$$C = Q/V \quad \text{and} \quad Q = V \cdot C \ .$$

A compartment is assumed to be instantaneously well mixed. In physical terms, this means that the diffusional process is infinitely fast.
Sources are processes which increase Q. Sinks are processes which decrease Q. The rate of change of Q with respect to time, the change in the amount is given by

$$dQ/dt = +Sources - Sinks.$$

This is called an ordinary differential equation (ODE).
In the simpler models, the volume is usually constant, and the equation is written as either

$$V * dC/dt = +Sources - Sinks$$

or

$$dC/dt = Sources/V - Sinks/V.$$

The complete specification of an ODE requires an initial condition (IC). The initial condition

specifies the concentration at time equals 0. This is written as

$$C(0)=C_{0.}$$

2. Specifying units:

The units for the amount of material are usually moles (*mole*), millimoles (*mmol*), micromoles (*umol*), nanomoles (*nmol*) or picomoles (*pmol*). The units for concentration are usually molar (*M*), millimolar (*mM*), micromolar (*uM*), or nanomolar (*nM*), or picomolar (*pM*). The units for volume include liter (both *liter* and *L*), milliliter (ml), and cubic centimeters (cm^3). The units for time include seconds (both *s* and *sec*), millisecond (*ms* and *msec*), minute (*min*) and hour (*hr*). A 1 molar concentration is 1 mole of substance per liter. Water has a concentration of 55 moles/liter or 55 Molar.

3. A compartmental model for Decay, Constant Value, or Synthesis
The ODE and IC are given by

$$V \cdot dC(t)/dt = G \cdot C(t), C(0)=C_{0.}$$

The analytic solution is

$$C(t)=C_0 * \exp\left(\frac{G}{V} \cdot t\right).$$

If $G<0$, the right hand side of the equation is a sink and the solution is exponentially decaying. If $G=0$, the solution is a constant. If $G>0$, the right hand side of the equation is a source and the solution is exponentially growing. An example of a decaying solution is the decay of a radioactive isotope. An example of a growing solution is the exponential growth of yeast in a bioreactor.

4. Coding the model in JSim:
A comment can be in block form beginning with /* and terminated with */.
Other comments can be in the form // Comment. The CompDecayTutorial model is accessible from http://www.physiome.org/jsim/models/webmodel/NSR/CompDecayTutorial/ contains additional comments. This additional information is there so that the model will conform to NSR standards.

Below is a "stripped down" version of the CompDecayTutorial model:

```
import nsrunit;              // required for unit conversion and checking
unit conversion on;         // required for unit conversion and checking
math CompDecayTutorial {
realDomain t s; t.min=0; t.max=3; t.delta=0.1; // independent variable
real C0 = 1 mM,             // Initial Concentration
     V  = 0.05 ml,          // Volume
     G  = -3.0 ml/min;      // Decay rate
real C(t) mM,               // Numeric Solution concentration (dependent variable)
     analyticC(t);          // Analytic Solution concentration (dependent variable)
when(t=t.min) C=C0;         // Initial condition
C:t = (G/V)*C;              // Ordinary differential equation (ODE)
analyticC = C0*exp((G/V)*t); // Analytic Solution
}                           // END
```

You can run this model as an applet from the web site. If you have downloaded JSim to your own computer, you can download the code and run it on your own computer.

Save your projects by going to the Project tab and use the File tab.

Read model Notes (tab at bottom of run time page).

Lesson II. Multiple substances in a compartment
1. Compartmental Model with reaction:

```
+--------------------------+
|                          |
|          Ga2b            |
|      A <--------> B      |
|                          |
|    V                     |
+--------------------------+
```

       This is a model where substance A can become substance B irreversibly. The reaction is governed by a single rate constant, G_{a2b}. This model is available at

http://www.physiome.org/jsim/models/webmodel/NSR/CompReactionTutorial/

A "stripped down" version of the model is presented here.

dA/dt has a single sink term, -(Ga2b/V)*A. dB/dt has a single source term, +(Ga2b/V)*A.

```
import nsrunit; unit conversion on;

math CompReactionTutorial {

 realDomain t s; t.min=0; t.max=3; t.delta=0.1;
 real A0    = 1 mM,           // Initial concentration of A
      B0    = 0.3 mM,         // Initial concentration of B
      V     = 0.05 ml,        // Volume
      Ga2b  = 5 ml/min;       // Reaction rate for A->B
 real A(t) mM,                // Concentration of A
      B(t) mM;                // Concentration of B
 when(t=t.min) {A=A0; B=B0;}// Note the use of brackets
 A:t = -(Ga2b/V)*A;          // Decay equation for A
 B:t =  (Ga2b/V)*A;          // Synthesis equation for B
 real analyticA(t) mM, analyticB(t) mM; // Analytic solutions
 analyticA=A0*exp(-Ga2b*t/V);
 analyticB=B0+A0*(1-exp(-Ga2b*t/V));

}
```
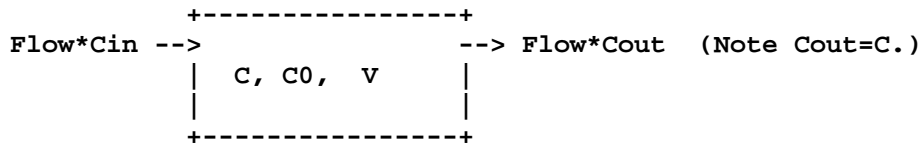
Explore the model by changing values for the various parameters and running solutions. You may have have noticed that the volume is missing from this model. Locate the ODE solvers page (Pages Tab pull down menu on Run Time Page.) Try the various
solvers and also vary t.delta with them. How does the error in Euler-1 step change with changes in time step? What about RK4 (Runge-Kutta fourth order)?

Lesson III: Compartment Model with flow:

1. Inflow and Outflow

```
                +----------------+
  Flow*Cin -->                      --> Flow*Cout   (Note Cout=C.)
                |   C, C0,   V    |
                |                 |
                +----------------+
```

The next model is the inflow of a concentration and the outflow of a concentration in a single compartment. Recall that in a compartmental model, mixing occurs instantaneously. The governing ODE and IC are:

$$dC/dt = Flow/Volume * (Cin(t) - Cout(t));$$

where *Cin(t)* is the inflowing concentration and *Cout(t)* is the outflowing concentration, and

$C(0)=0$; What is the source? What is the sink?

Since the compartment is instantaneously well mixed, $Cout(t) = C(t)$

and we can rewrite the ODE as

$$dC/dt = Flow/Volume * (Cin(t) - C).$$

The analytic solution involves an integral. For a constant inflowing concentration, *Cin(t)=Cin*, the analytic solution is given by

$$C(t) = Cin - (Cin - C_0) * \exp(-Flow * t/Volume).$$

The time constant for the system is defined as $\tau = Volume/Flow.$

This model is available at

http://www.physiome.org/jsim/models/webmodel/NSR/CompFlowTutorial/

A "stripped down" version of the model is presented here.

```
 import nsrunit; unit conversion on;

 math CompFlowTutorial {

 realDomain t sec; t.min=0; t.max=40; t.delta=0.1;
 real C0      = 1 mM,            // Initial Concentration
      Volume = 0.05 ml,          // Compartment Volume
      Flow   = 1 ml/min    ;     // Flow rate
extern real Cin(t) mM;           // Inflowing concentration (defined with
                                 //    function generator
 real C(t) mM;                   // Compartment concentration
 when(t = t.min) C=C0;            // Initial condition
 C:t   = (Flow/Volume)*(Cin-Cout); // ODE
 real analyticC(t) mM;            // Analytic solution when Cin is a constant
 real tau sec;
 tau = Volume/Flow;
 analyticC=Cin-(Cin-C0)*exp(-t/tau);
}
```

Vary **C0** and **Cin**.

Follow the instructions in the JSim Notes (tab at bottom of Run Time Page).

3. The function generator

The function generator for **Cin(t)** can be accessed by the small sine wave enclosed in a circle **(~)** button to the right of **Cin(t)** . Selecting this button will bring up a menu for creating a function generator. You can accept the default name fgen_1 or create a new function generator with a different name. Select "Use existing fgen_1." Selecting the Pulse 1 button gives you access to all of the predefined functions. Note that the analytic solution based on constant input will no longer be valid as a comparison. You may wish to increase t.max to 30. Compartment flow models will be revisited when flow compartment in series are considered.

Lesson IV. Multiple compartments

1. Compartmental Model with exchange:

```
+----------+              +----------+
| C1       |    PS        | C2       |
|          |<-------->|             |
|       V1 |              |       V2 |
+----------+              +----------+
```
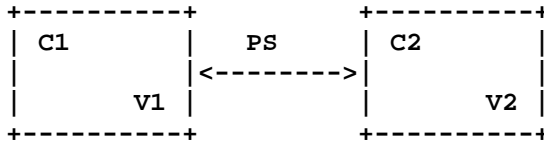
      This is a model with exchange between two compartments. The exchange process is governed by an exchange coefficient. For reasons which will be explained in a later lesson, the exchange coefficient is designated *PS*. The governing equations are

$$V_1 \cdot dC_1/dt = PS \cdot C_2 - PS \cdot C_1$$

$$V_2 \cdot dC_2/dt = PS \cdot C_1 - PS \cdot C_2 \quad,$$

which emphasizes the mass balance. They are usually written in the form

$$dC_1/dt = (PS/V_1) \cdot (C_2 - C_1)$$

$$dC_2/dt = -(PS/V_2) \cdot (C_2 - C_1) \quad.$$

Note that the right hand side (*RHS*) of the first equation is divided by V_1 and the RHS of the 2nd equation is divided by V_2. The two volumes are not necessarily equal. Identify all sources and sinks.

This model is available at

http://www.physiome.org/jsim/models/webmodel/NSR/CompExchangeTutorial/

A "stripped down" version of the model is presented here.

```
import nsrunit; unit conversion on;

math CompExchangeTutorial {
realDomain t s; t.min=0; t.max=4; t.delta=0.1;
real C10 = 1 mM,            // Initial concentration in first compartment
     C20 = 0 mM,            // Initial concentration in second compartment
     V1  = 0.05 ml/g,       // Volume of first compartment
                            //   units are milliliters per gram of tissue
     V2  = 0.15 ml/g,       // Volume of second compartment
     PS  = 5 ml/(g*min);    // Exchange rate
real C1(t) mM,              // Time dependent concentration in first  compartment
     C2(t) mM;              // Time dependent concentration in second compartment

when(t=t.min) {C1 = C10;
               C2 = C20;}
C1:t= (PS/V1)*(C2-C1);
C2:t=-(PS/V2)*(C2-C1);
real analyticC1(t) mM,
     analyticC2(t) mM,
```

```
      Exponential(t) dimensionless;
 Exponential   = exp(-PS*(V1+V2)*t/(V1*V2));
 analyticC1(t) = (1/(V1+V2)) * (C20*V2+C10*V1
                               + V2*(C10-C20)*Exponential);
 analyticC2(t) = (1/(V1+V2)) * (C20*V2+C10*V1
                               - V1*(C10-C20)*Exponential);

 }
```

Question: Ordinary Differential Equations are solved numerically by many different methods.
Access the ODE solvers from the **Run Time** GUI by using the **Pages**
tab and selecting **Solvers**. Change the ODE method to **Euler** and set **nstep** to 1.
Estimate the maximum error as a function of t.delta by changing t.delta (Run Time GUI) from 0.1
to 0.01and 0.001. Do the same for RK2 and RK4. Interpret your results.

Lesson V: Putting it all together:

1. Creating a new model

```
   Source       +--------------------------------------+   Sinks
   Flow*Alin --> A1, B1, V1, A10, B10                  --> Flow*A1
                 |                   (Sink) -G1B*B1     --> Flow*B1
                 |                                       |
                 |                                       |
                 |                                       |
                 +--- v ----- ^ ------ V ----- ^ --------+
                       |       |        |       |
                      Psa*A1 PSa*A2   PSb*B1  PSb*B2
                       |       |        |       |
                 +--- v ----- ^ ------ V ----- ^ --------+
                 |                                       |
                 |                                       |
                 |            Ga2b-->                    |
                 |         A2 --------> B2               |
                 |                                       |
                 |  V2,        A20, B20 (Initial         |
                 |                       Concentrations) |
                 +--------------------------------------+
```

We next construct a model for two species with inflow of species A, exchange of A between two compartments, irreversible reaction converting A to Bin non-flowing compartment, exchange of B between the two compartments, consumption of B in the flowing compartment and outflow of A and B.

Assignment: Write the governing equations. You will need to use everything you have already learned. Code the Mathematical Model Language version in JSim, compile and run it. Send project file to garyr@u.washington.edu for feedback.

How many species will you need? The exchange coefficients for A and B are different. You should be able to use the units from the previous problem.