# Workflow for Developing Mechanism-Based Computational Biology Models

This is a major rewrite of the original JAG version.

Developing useful mechanism-based computational biology models requires discipline and progression through a series of logical steps. Typically, one starts from biological observations and gradually develop mechanism-based understanding that explain the biological system in a way that provides new insights in the normal and abnormal operation of the biological system.

By necessity, computational biology models leave out a great deal of complexity and therefore represent simplifications of reality. However, if a model includes the key components of the biological system, it can be both predictive and useful. Furthermore, identifying the key elements that govern the system greatly simplifies the knowledge needed to construct a model and provides critical insight into the controlling mechanisms in a biological process.

A critical aspect of developing useful computational biology models is to use a well-defined process that progresses from biological observation and understanding, through the development of physical and mathematical definitions of the key components and processes in the system, leading finally to a computational instantiation of that conceptualization. Many of the problems encountered in developing useful computational biology models arise from trying to short circuit these steps. In addition, to apply a computational modeling in, for example, a clinical domain requires that the model is credible in terms of both the biology represented and the computational implementation. (Mulugeta 2018a) The computational biology community is beginning to develop standards for judging the credibility of a model. (c.f. Mulugeta 2018b, Patterson 2017) However what is often missed is the need to document the entire trajectory of the model development process.

People often devote a great deal of effort to building mathematical models and implementing them in computer code without spending enough effort in defining what biological concepts and processes those equations or code represent or what the end goal of the simulation is. It is critically important that sufficient time is spent in defining and understanding the biology and goals of the modeling project. Only after that understanding and scoping has been completed should the modeler move on writing computer code and running and analyzing the simulations. ~~\*\*\*As a rough rule of thumb: 50% of your effort should be directed to understanding the biology and what you are trying to model and identifying the model elements, 15% to defining the quantitative relationships among the model elements, 10% to writing computer code and 25% to running the code, analyzing and interpreting results.~~

We will present this model development workflow as a linear process, but in practice modelers will iteratively refine their models, revisiting, revising and improving their descriptions at each level of abstraction as they proceed.

Our basic workflow is outlined in **Figure 1**. The workflow develops three levels of model; a *biological model*, a *mathematical model* and a *computational model*. Within each of these model levels there is a recommended set of steps for developing the model at that level. We start with a description of the three model levels.

**The Biological Model:** This model, also known as a “conceptual model”, is often created with the help of a domain expert such as a biologist or a clinician. This model is an attempt to explain an observable reality. It should include the parts of the reality that we know about filtered by what we (and the domain expert) believe are relevant to a particular question. This descrip­tion will include *physical objects* (cells, enzymes, tissues, …) and *processes* (cell proliferation, enzyme reaction, …). In addition, this description may include *spatial* and *temporal* information. Besides this list of objects and processes the biological model should also include a list of measurables and outcomes. For example, which of the physical objects are measurable in terms of count, or volume or concentration? Which of the processes are measurable such as in a time course? Finally, the biological model should identify *outcomes* of interest.

|  |
| --- |
|  |
| **Figure 1:**  Workflow for Developing Mechanism-Based  Computational Biology Models. |

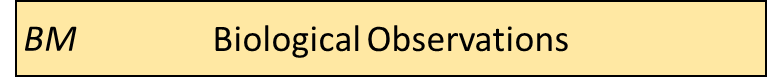
**The Mathematical Model:** The mathematical model is a mathematical description of the biological model. Here we create mathematical definitions, based on physics, chemical, or other physical models, of the objects and processes. Often the creation of the mathematical model requires making significant assumptions about the underlying mechanisms of the biological system. For example, when modeling transfer of a small molecule into a cell is the process simple diffusion (that can be modelled as a reversible first order ordinary differential equa­tion (ODE)) or is the process trans­porter mediated that may become saturated at high small molecule concentration? This process of converting a biological model into a computational model is the first point where the modeling process adds value to our understanding of the biological system. The mathematical model requires a level of understanding and specification that is rarely present in biological models. The modeler, often with the help of the domain expert, must coerce the available biological knowledge into a numeric framework and in that process decide how the available data can be used to select from a multitude of possible mathematical instantiations. At this stage it often becomes clear that certain aspects of the biological system that are measurable have indeed never been measured. Often, at this stage the modeler also encounters the problem that quantities required in the mathematical model are not directly measurable in the biological assays. This situation may require reformulating the mathematical model to avoid dependence on intrinsically unmeasurable quantities. On the other hand, at this point the mathematical model can also begin to offer new insights into the biological model. Quantities that are not directly *measurable* may instead by *calculable* by the model. This capability is one of the most attractive aspects of a mathematical model. A key aspect of the creation of the mathematical models is that it is the point where the model moves from a biological point of view to a chemical or physics-based description. This allows the modeler to use standard mathematical forms developed in those domains to define the model for the biological domain.

**The Computational Model:** The computational model is an instantiation of the mathematical model in a computing system. In some cases, this instantiation can introduce implementation dependent parameters in a model. For example, if time or space is discretized in the computer simulation then that adds implementation-specific parameters that are not part of the biological system being modelled. At his point in the model development process the model, which has already moved from the biological to the mathematical/chemical/physics domains, moves again into the computational domain. Issues such as computability, simulation run time, memory and disk storage requirements become practical issues that must be considered.

There are two critical considerations in implementing a mathematical model in computable code that are often overlooked. The first is the choice of the computing infrastructure. In general, any mathematical model can be implemented in any modern computing languages. However, the modeler should try to reuse existing implementation frameworkswhenever possible. The reuse of existing frameworks has two major advantages compare to custom written code. First, it is much more programmer efficient to reuse existing code than it is to recreate code for problems that have already been solved. Secondly, reuse of existing code significantly reduces the effort needed to *verify* that the code implementation correctly solves the set of mathematical equations. This aspect of developing computational models is often overlooked. It takes a significant effort to verify that a code implementation correctly solves a mathematical problem. Reuse of existing code bases significantly reduces the effort need for verification. The second advantage of reuse of existing frameworks is that the model becomes much more shareable and, in some cases, much more easily annotated.

An important problematic aspect of the computational model is that it often unlinks the model from the biological concepts and descriptions in the biological model. This unlinking makes it extremely difficult for others to map biological concepts to computational implementation and greatly inhibits understanding of the model. This unlinking also makes it difficult for others to reuse the computational model. Some computational biology modeling communities, most notably that of the Systems Biology Markup Language (SBML) (Hucka 2003), have developed techniques for incorporating both the biological and mathematical description in the same computable code file format. This approach makes the executable code itself a repository of biological knowledge. That repository can be searched, mined, and reused much more easily than custom, one-of code. We hope that in the future other computational biology communities will implement similar methods of directly imbedding biological descriptors in their computational code. For all computational biology communities, a long-term goal should be to create technologies that allow for the recreation of the original biological model from the computational model.

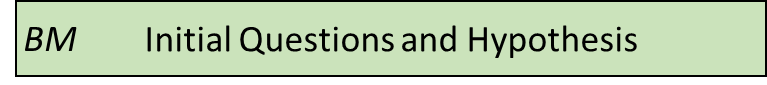
Now that we have laid out a three-level model structure we will examine in detail the workflow for each level.

 Begin by writing down a list of the key biological observations you wish to explore. Often the best starting point for a multi-cell scale model are the cartoons biologist commonly use to describe a tissue or organ. For example, the major cell types in the liver and a representation of the VEGF driven angiogenesis signaling pathway are described as cartoons in **Figure 2**. This simple viewpoint is used by biologists to graphically represent the key players and the key understanding of a biological process. This layout concisely lists the major components of the system.

Using the simple cartoon showing the key **objects**, define the key experimental observations you wish to explain. What **processes** are the **objects** involved in? Do the cells proliferate, carry out some key biological process such as metabolism, or die? Under what conditions do those processes occur? What key experiments and publications describe both the basic tissue structure as well as the (possibly abnormal) processes you are interested in? What are the core concepts discussed in the literature and how do those concepts contribute to the normal or abnormal behavior of the tissue? Can you can identify any observations that appear critical to the phenomena you are exploring? Can you identify any that seem irrelevant to your phenomena of interest?

While developing this initial list of objects and processes (cf. **Table xx**) it is useful to consider what is *visible* and what is *measurable* in an experiment. A cell expresses thousands of different proteins and is simultaneously carry out a huge number of processes. All that information cannot be included in the model. So how do you decide what to include at this stage of the modeling development process? One approach is to only include those things that are directly visible or that can be directly linked to a biological behavior. For example, in a multi-cell model cells are directly visible and measurable. Cells typically appear to be adhesive to each other suggesting the need for an adhesion process. If the cells are observed to grow or die, then those processes should be included. If those processes are not observed then they are not, and should not, be included. Subcellular processes are generally not directly visible and should not be included unless there is some direct measurable quantity (which might be at the cell or tissue level) that can be directly linked to the unmeasurable subcellular process.

Building this initial model description will often raise new questions and require new biological background as the process progresses. Therefore, as you proceed through the model building process it is expected that additional material will be added to this section in the form of new sources, new observations, etc. In addition, it is possible that components included initially will later be deemed less important and can be relegated to background material.

 Defining the initial questions and hypotheses is the most important step in the model development process and yet it is the step most often treated superficially. What are the main questions you wish to answer? Can the questions be described precisely and succinctly? For example, in cancer biology it is known that lack of nutrients in a tumor can lead to tumor cell death. Cells receive most of their nutrients from the blood supply, suggesting the importance of blood flow in and around the tumor. Based on that assumption, a hypothesis might be that “increased blood supply reduces tumor cell starvation resulting in increased tumor size”. Overall this is a *selection* step in which you must decide what parts of the Biological Observations will be included in the model.

|  |
| --- |
|  |
| **Figure 2:** A typical biological cartoons for a tissue and a signaling pathway. Left: Shown is a small section of the liver showing the major cell types (hepatocyte, endothelial cell, …), the sinusoidal blood vessel lumen, and a greatly simplified representation of the spatial relationships among the key components (not to scale). (From: [http://www.eclinpath.com­/­chemistry/­liver/liver-structure-and-function/liverlobule/](http://www.eclinpath.com/chemistry/liver/liver-structure-and-function/liverlobule/)) Right: The VEGF signaling pathway leading to angiogenesis (new blood vessel formation). (From: [http://www.genecopoeia.com/product/search/pathway/­h\_vegfPathway.php](http://www.genecopoeia.com/product/search/pathway/h_vegfPathway.php)) |

**Table 1:** Initial list of objects based on the simplest description of the biological system of interest.

|  |  |  |  |
| --- | --- | --- | --- |
| **Objects** | **Properties** | **Property Type** | **Reference** |
| Hepatocyte | Location Volume | Parameter |  |
| Endothelial Cell | Location Volume | Variable |  |
| etc. |  |  |  |
|  |  |  |  |
| **Processes** | **Properties** | **Property Type** | **Reference** |
| Blood Flow | Velocity | Parameter |  |
| Blood - hepatocyte O2 exchange | Rate | Variable |  |
| etc. |  |  |  |

A model takes a set of Objects with their Behaviors and/or processes and predicts how the State will change given a specific set of Initial and Boundary Conditions. So, you will typically ask whether a specific set of model elements are sufficient to reproduce an observed system behavior. Many questions you might ask may not be addressable by a model, so you will need to frame your question in a way that will allow your model to answer the question. What are specific hypotheses concerning how model elements determine the macroscopic result(s) of interest that you could test? Your models will only be useful if they begin with experimentally measurable states, and/or states inferable from experiment or a limited number of hypothetical alternatives and then predict experimentally measurable states. If we try to build a model that requires us to know the position of every atom in an organism or if we predict something we could never verify experimentally we will not be doing useful science.

Remember that **a model can usually only show the sufficiency (or insufficiency)** of your hypothesized elements to explain an observation. The probability exists that a different set of model elements could explain the same observation equally well or better.

Express your initial questions as a brief list. Each question should connect an experimentally measurable model element to an observable result. Ideally, express these questions in the form of assertions (hypotheses), though you may initially want them to have an exploratory structure.

You should also begin to think about how you might compare model outputs to experimental results to check if the model reproduces the phenomenon of interest. What are the observables we want our model to reproduce? Sometimes we may need to invoke “hidden observables,” states that we must infer indirectly from directly observable quantities.

Based on your reading and hypotheses, generate an initial set of hypotheses concerning model elements. Hypotheses can define either model structure (*e.g.* the nature and type of interactions) or the values of parameters for a given model structure. Express this in the form of a list about Objects, Behaviors, and Interactions (e.g. “Access to blood borne nutrients is needed for healthy cells”, “Hepatocytes seem to be crucial players,” “Organization into epithelia seems to determine the rate of metastasis,” “Apoptosis driven by contact signaling seems important,” “The amount of growth factor in the medium seems critical,” “Initial cell density determines if the cells differentiate into bone or neuron” etc.)

 Build a formal structure describing the model elements that captures the domain-specific knowledge that you have identified as significant to your problem. The Qualitative Model embodies your hypotheses about the needed Model Elements and their relationships that lead to the phenomenon of interest. This is the second most important step in model building and (again) is often neglected.

Based on the general observations in the previous step, define the Objects you will initially include in your model (you can always come back to this later and add or eliminate them). Put these Objects into a table and assign them the Properties that you think are clearly going to be important. Remember that nothing exists in your model until you assert it! If you assert an Object, it does not have qualities like position or volume unless you say it does!

In the following, begin by focusing on the names and identities of the Objects, Behaviors and Interactions, then think through the Properties to characterize them. Don’t worry too much if initially the identity and names of the Parameters are not so clear. These are often implementation dependent and can be refined later.

|  |  |  |
| --- | --- | --- |
| **Object** | **Properties** | **Property Type** |
| Environment | Spatial-extent | Parameter |
| Cell | Position | Variable |

Now, based on your analysis of the biology of the problem, begin to define the Behaviors (processes) and Interaction for your Objects. For Behaviors, list the Objects they affect and the Object Properties that they involve.

|  |  |  |
| --- | --- | --- |
| **Behavior** | **Participant Object** | **Properties** |
| Movement | Cell | Object Velocity |

Each time you include a Behavior, you will likely find that you need to add additional Properties to the appropriate Object definition so that the model is self-consistent.

|  |  |  |
| --- | --- | --- |
| **Object** | **Properties** | **Property Type** |
| Environment | Spatial-extent | Parameter |
| Cell | Position Velocity | Variable Variable |

For Interactions (or processes), list the participating Objects and the relevant properties that affect the interaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Interaction** | **Participant Objects** | **Properties** | **Property Type** |
| Cell-Cell Adhesion | Cell 1 | Cadherin Level | Parameter |
|  | Cell 2 | Cadherin Level | Parameter |
| Oxygen Uptake | Cell | Uptake Rate | Parameter |
|  | Oxygen Field | Concentration | Variable |

Note that we now need to go back and define the Oxygen Concentration Field Object

|  |  |  |
| --- | --- | --- |
| **Object** | **Properties** | **Property Type** |
| Environment | Spatial-extent | Parameter |
| Cell | Position | Variable |
|  | Velocity | Variable |
| Oxygen Field | Concentration | Variable |
|  | Diffusion Constant | Parameter |
|  |  |  |

If we have an Oxygen Field, we know that the supply of Oxygen available to the cells will depend on the diffusion rate of Oxygen, so we need to add the Diffusion Behavior to the Field object.

|  |  |  |
| --- | --- | --- |
| **Behavior** | **Participant Object** | **Properties** |
| Diffusion | Oxygen Field | Diffusion Coefficient |

Now we need to define the dynamics. In general, discrete objects move based on their current velocity and the forces applied to them, and Fields evolve due to diffusion.

|  |  |  |  |
| --- | --- | --- | --- |
| **Dynamics** | **Participant Objects** | **Participant Behaviors/Interactions** | **Property Affected** |
| Movement | Cell | Movement (Velocity) | Position |
| Diffusion and Uptake | Oxygen Field | Cell (Positions), Diffusion | Field (Concentration), Cells (Amount of Oxygen) |

We should review our qualitative verbal model a few times to make sure it is consistent and complete and anything we call for in one place is defined somewhere in the model.

The above set of tables and descriptions defines a Model template. To convert it into something that can describe a biological situation, we will need to create a Model Instance, that specifies our initial and boundary conditions. We can do that now, if we are comfortable with it, otherwise we can come back to them as we continue to develop our model.

You can write a paragraph defining where the objects are initially, what their states are and what the boundary conditions are. For example, we need to specify the number, location and size of the cells. What the initial Oxygen concentration is everywhere. What happens at the boundary of the Environment when it is encountered by cells (rigid wall, absorbing, periodic) or Oxygen (absorbing, impermeable, source…). Are there any events or processes that add or remove objects such as cells or Oxygen?

One of the key aims of science is communication. Because models are developed by assertion, we run the risk of using terms that are meaningless to the colleagues with whom we want to communicate. If we are going to follow best practice in developing models other people can understand, we should *annotate* our model using accepted reference ontologies and controlled vocabularies. The biological literature is riddled with alternative and often incorrect terminology. Incorrectly named biological objects and process obscures the knowledge in a model (be it laboratory or computational) and inhibits understanding and reuse of models. Therefore, you should attempt to link all the objects and concepts in your model to terms in a controlled vocabulary. This minimizes the chances of using nonstandard nomenclature and greatly increases the chances the model can be understood by others. For example, a model that includes “hepatocytes” you should link that object to the appropriate term in the cell ontology; *hepatocyte - Cell Ontology (CL) CL:0000182*. Often a term will occur in more than one ontology and it is reasonable to include multiple links. For example, “hepatocyte” is also in the Foundational Model of Anatomy (FMA) as fma14515. While annotating your model you should of course insure that the definition in the controlled vocabulary matches your expectation.

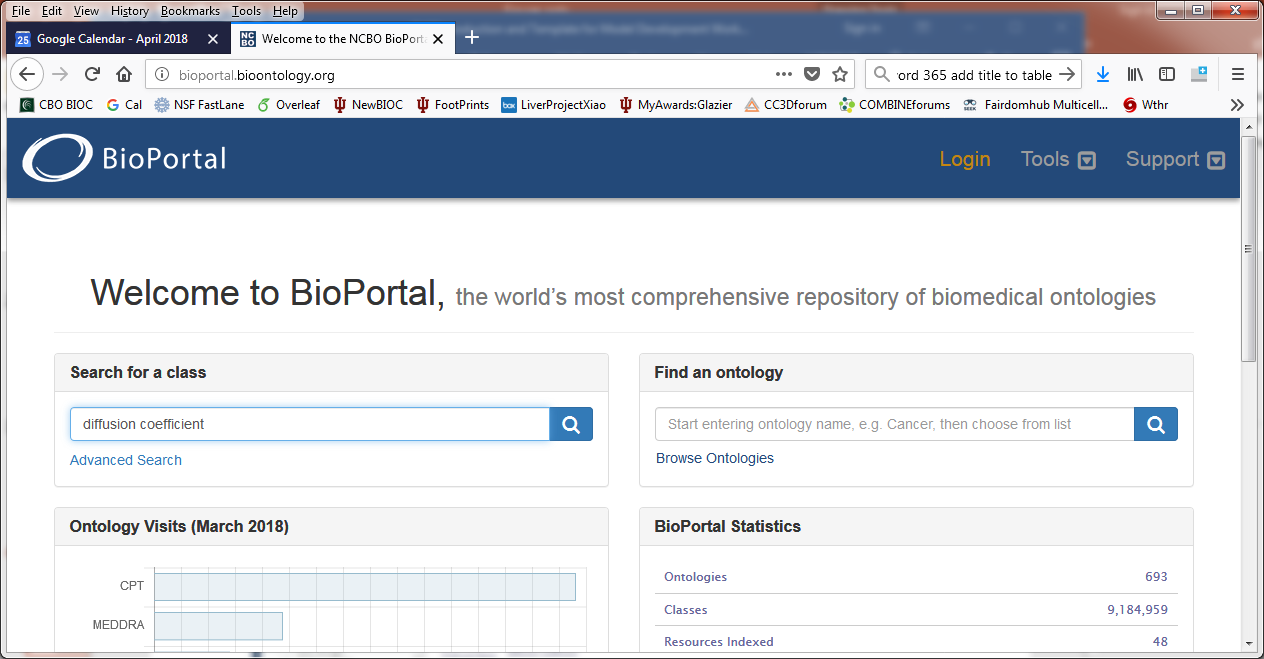
While annotating your model with links to controlled vocabularies and ontologies you can also collect a list of common acronyms and alternative names since these are often listed in the definition of the term. These alternative names can be of great value when searching the literature. (Unfortunately, most journals do not require the use of standard nomenclature nor do they require linking terms and concepts to controlled vocabularies or ontologies.) For example, “Estrogen Receptor 1” (the official name) is also known as “Estrogen Receptor alpha”, ESR1 (the official symbol), ESRa, ERalpha, etc. Indeed, the historical name is just “estrogen receptor”. This list of alternative names can be very useful when searching the literature and in addition, including the alternate names makes your model more findable.

To do so, we add two columns to our tables, source reference and annotation terminology

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Object** | **Properties** | **Property Type** | **Source Reference** | **Annotation Terminology** |
| Environment | Spatial-extent | Parameter | [New Concept] |  |
| Cell | Position | Variable | http://www.red3d.com/cwr/boids/ |  |

In the first column, annotate the journal article or other source for each concept. Notate concepts NOT in your source! In more developed models, with mathematical forms or parameter values, specify the source in more detail (*e.g.* page number, table number and line, equation number…) and, this is critically important, the units of the value.

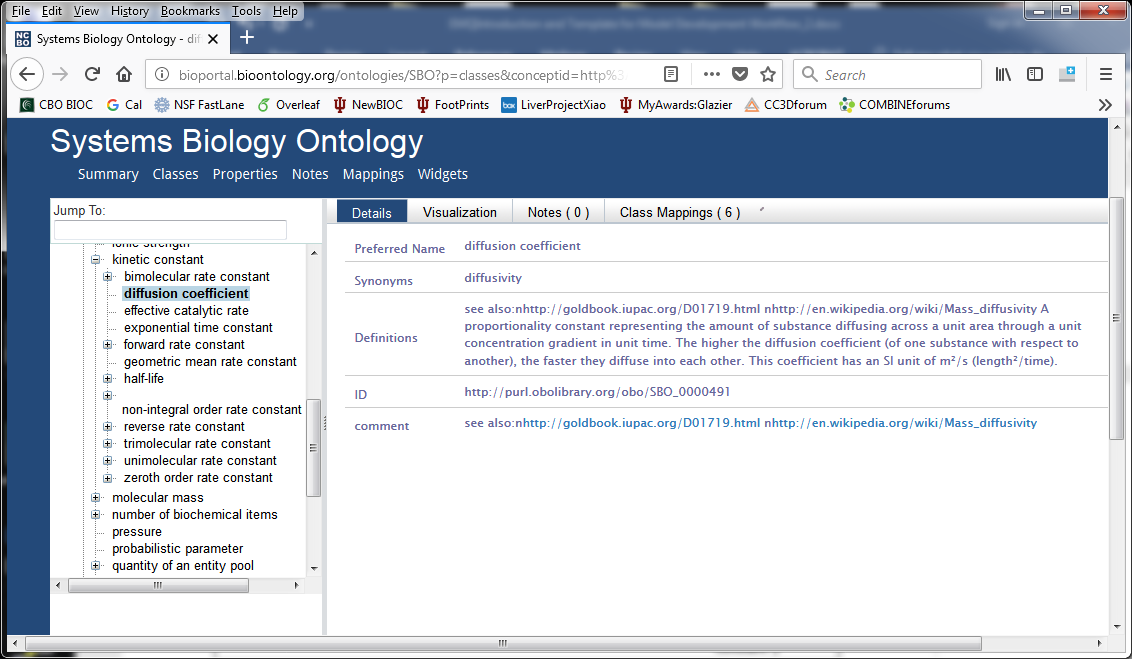
The second column will be filled in from a reference ontology. I will illustrate this in class, but as a reminder, to assign a unique meaning to our terms, search in <http://bioportal.bioontology.org/> (Whetzel 2011). You will see a search dialog box



Type the terms you are defining into the search box. BioPortal will return a list of options for common terms, each corresponding to a different reference ontology. Ontologies (or naming authorities) exist for genes (Gene Ontology (GO), proteins, biological process, chemicals, …), you’ll have to dig a bit to see which one is the one that makes the most sense for your application. If your search returns nothing, see if a synonym or similar idea works.

Select one of the offered ontologies that seems sensible. **Try to use the same ontology for as many terms as possible and double check each term by opening the definition and reading it.** After a little exploration you will find that a few of these ontologies have almost everything you need.

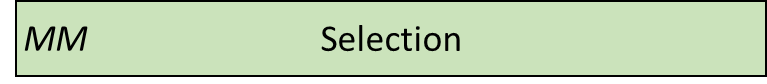
If the offered definition seems sensible, adopt it by copying the ID field and pasting it into your table.



This formalizes your definition of the concept and clarifies the meaning to others.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Object** | **Properties** | **Property Type** | **Source Reference** | **Annotation Terminology** |
| Environment | Spatial-extent | Parameter | [New Concept] | [MESH:D004777](http://purl.bioontology.org/ontology/MESH/D004777) |
| Cell | Position | Variable |  | [PATO:0000140](http://purl.obolibrary.org/obo/PATO_0000140) |
| Molecular Field | Diffusion Coefficient | Parameter |  | [OPB:01306](http://bhi.washington.edu/OPB#OPB_01306) or [SBO:0000491](http://purl.obolibrary.org/obo/SBO_0000491)  (Aka “diffusion constant”) |

If you paste the annotation term “MESH:D004777” into the Google search bar, it should pull up your definition.

 As in the selection step for the Biological Model, we must decide which components of the Biological Model can and will be included in the Mathematical Model. In general, it is unlikely that everything in the Biological Model will be representable in the Mathematical Model. It is likely that simplification will need to be made, or perhaps multiple concepts in the Biological Model will be aggregated in the Mathematical Model. In any case this selection step decides which components will propagate into the Mathematical Model.

 In the next stage of model building, we refine the Qualitative Model into a specific mathematical representation, our Quantitative Model. We need to represent all the concepts we have defined in our qualitative model. We define the level of detail of our description and which specific aspects of processes we will include in our model. In making these choices, we decide what numeric formulism and parameters we will use in the model. In the case of a chemical reaction, we would decide if a reaction rate obeyed Mass-Action, Michaelis-Menten, Hill or some other rate law. For spatial objects, if we are describing a cell Object, a model might specify its position and volume, but not its specific shape, or might define an elliptical cell with specific major and minor axes, but arbitrary orientation, or might specify a cell with a specific volume and membrane area, or a detailed but static shape for a complex cell like a neuron. A chemical in a Field might be present everywhere and its changes in concentration depend only on diffusion (in which case we would need to define a diffusion constant) or it might be carried (advected) by flow of the fluid, and/or the movement of cells. Diffusion might be uniform everywhere (in which case we would have a single diffusion constant) or it might be reduced at cell membranes or inside cells. If the diffusion occurs in extracellular matrix, it might differ in rate depending on the orientation of the matrix fibers. For cell motility, we might decide if the motion is directed or random (and if so, how we will describe the velocity profile). Usually, we will start with the simplest, most generic assumptions, and add complexity only when a simpler model fails to reproduce our observations or if we have specific experimental evidence of the importance of a complex microscopic mechanism. In the latter case, we should always compare the consequences of the more complex mechanism with a simpler one.

At the end of this step of model building, the quantitative model should be fully defined and should have no missing information or parameters. In addition, the application of this quantitative model should be mathematically precise—that is, anyone implementing the model should be able to obtain the same model outputs.

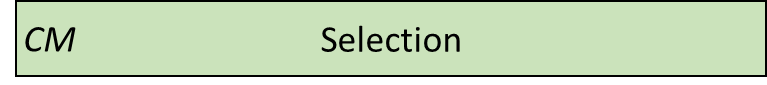
These decisions are properly the domain of mathematical chemistry, biology and biophysics, and in a mathematical biology course, we would spend a good deal of time discussing the specific choices and tradeoffs that specific mathematical representations entail. Here, to begin with, we will accept the choices of mathematical representation that CompuCell3D makes and revisit them as we become more comfortable with the methodologies and the practice of developing multiscale models.

Time scale separation considerations?

 Now that we know what parameters we need to specify in our quantitative model, we should look to see which parameter values are known in the literature. Examples of such parameters might be the typical cell diameter in microns, the time between cell divisions, the speed of movement of a cell, the diffusion rate of a molecule in water (or in a biological material in the rare cases it is known), or the compressibility or viscosity of material components, etc. If possible, we should provide ranges for these values, references to their sources and the units. In some cases, the specific quantities needed are not available (or cannot be found). In these cases, attempt to assign a reasonable range for the parameter based similar quantities and basic physical considerations.

In rare instances when we can analyze the equations in a Quantitative Model analytically, determine the predictions of our Quantitative Model for a specific set of parameters, and Initial and boundary conditions. More generally though we must translate the mathematical description in the Quantitative Model into a numerical approximation that can be solved on a computer. We face the same problem when we do numerical approximation of a definite integral in calculus or solve a set of ordinary differential equations (ODEs) numerically. Sometimes these choices can produce radical differences in results (*e.g*., in chemical reactions choosing between deterministic ordinary differential equation solvers or stochastic Gillespie solvers). Often these numerical implementations will require us to define additional parameters beyond those required by the Quantitative Model that specify details of the numerics (*e.g.* the integration time step) or which require us to translate a concept in the Mathematical model into the representation of the numerical approach.

We are now at a point in the model development process where we need to move from the biological and mathematical domains into the third model domain, the computational domain. As mentioned earlier, it is best if we make this transition into an existing code base suitable for the class of problem we are working on. Custom code can be written, and in some cases that is the only option, but in general our model development processes will be easier and more robust if we use an existing code base.

 As in the two previous selection steps, we must decide which components of the Mathematical Model will be included in the Computational Model. At this step we must map mathematical concepts into a computing framework. In general, it is possible that some aspects in the Mathematical Model will be not be directly representable in the Computational Model. It is likely that simplification will need to be made, or perhaps multiple concepts in the Mathematical Model will be aggregated in the Computational Model. In any case this selection step decides which components will propagate into the Computational Model. In addition, this step directly influences the next step where we decide upon a computational modality. If we can reduce the model to a set of ODE’s than that directly impacts our choice of computational modality. Alternatively, if our selection includes spatial characteristics then our choice for a computational modality is different than it would be for a set of ODE’s.

 There are many different ways to represent cells computationally—popular representations include lattice-based cellular automata (which represent cells as single voxels in a lattice), lattice-free center models that represent the cell as the positions of their centers, with a potential field around their centers, lattice-free subcellular element models (which represent cells as clouds of center-model blobs), lattice-free vertex models (which represent cells only by the points at where three or more cells touch), lattice-free finite element models (which apply the technology of finite-element materials modeling to describe the boundaries of cells by triangulation meshes and/or their volumes by tetrahedralization meshes) and the lattice-based Glazier-Graner-Hogeweg (GGH)/Cellular Potts model (CPM) models (which represent cells as sets of lattice sites). Each of these methodologies requires different translations of biological and mathematical concepts and adds different Methodology-Dependent parameters. *E.g.*, lattice-based methods, require the specification of the lattice while lattice-free methods do not; cells in a center-model do not have well-defined volumes or surfaces; finite-element methods require rules for mesh updating as the cell configuration changes, etc.

In this course, we will generally use CompuCell3D and its GGH/CPM spatial representation of cells. As we will see, the GGH/CPM representation has its specific set of advantages, disadvantages and complications. Because it is a lattice method, with a fixed lattice size, we will need to specify the size of the lattice unit (*voxel*) (in microns), the size of the modeled Environment (in voxels), the lattice type (square vs. hexagonal) and the lattice interaction range. All these parameters reflect the numerical solution method and not any reality of the biology. In addition, GGH/CPM specifies many properties and behaviors in terms of constraints, which mean that we must specify a target and constraint strength, for each. For example, instead of specifying the volume of a Cell, we will specify its target volume and inverse compressibility. If we select these Methodology-Dependent parameters appropriately, and if the translation of Quantitative Model parameters to Methodology-specific parameters is correct, then our model predictions should usually be independent of the translation. *E.g.*, if we change the lattice unit from 0.25 microns to 0.5 microns and correctly rescale my other parameters, the simulation results should be essentially unchanged.

While the ideal prediction of a quantitative model is unambiguous, the approximations we obtain for different numerical approaches may differ. These differences are artifacts of the numerical method and do not reflect the underlying behavior of the quantitative model. In an ideal world, we would be able to use different numerical solution methods of the same quantitative model to check for these artifacts. Unfortunately, we do not have general multimethod tools available to do so.

The paper of Osborne *et al.* (Osborne 2017) attempts such a comparison. It is well worth reading but note that there are some errors in the translation of the Quantitative Model into computational instantiations which mean that some of the differences in results reported are due to mistakes by the authors rather than numerical artifacts.

As with our previous model development steps, we should define the method and method dependent parameters of our particular instantiation of the computational model.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Object** | **Properties** | **Property Type** | **Source Reference** | **Annotation Terminology** |
| Model Environment | Glazier and Graner model, extended large-q Potts model |  | [Swat 2012] | [MAMO:0000183](http://identifiers.org/mamo/MAMO_0000183) |
| Model Extent | ThreeDimensionalSpatialRegion |  |  | [BFO:0000028](http://purl.obolibrary.org/obo/BFO_0000028) |
|  |  |  |  |  |

 Once we have defined the solver methodologies and selected the Methodology-Dependent parameters, we need to write the simulation code that will run on a computer. For the GGH/CPM methodology, there are multiple software packages that implement the same numerical methods (including CompuCell3D, CHASTE, Tissue Simulation Toolkit, Morpheus and others). Each one of these will require the model to be implemented using different syntax, just the way you would need to write different code to solve the same problem using MatLab versus Mathematica.

In CompuCell3D (**CC3D**) we define the static properties of the model (cell types, behaviors, interactions,) and parameters that apply to all members of a given Object category, and complex dynamics (like cell growth and division and differentiation) as well as events and parameters that are unique to each instance of an Object. CC3D uses Python scripting to define initial conditions and to track, plot and log simulation results.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Object** | **Properties** | **Property Type** | **Source Reference** | **Annotation Terminology** |
| Three-dimensional region | X-axis Y-axis Z-axis | Parameter |  | [OPB:01211](http://bhi.washington.edu/OPB#OPB_01211) |
|  |  |  |  |  |
| Progression with Fixed Time Step |  | Parameter |  | [KISAO:0000108](http://www.biomodels.net/kisao/KISAO#KISAO_0000108) |
| Simulation Duration |  | Parameter |  | [SBO:0000585](http://purl.obolibrary.org/obo/SBO_0000585) |

 Once a computable model is ready we proceed to calibrating the model based on experimental data. In general, the model will contain parameters for which no value is available, and we will need to determine those parameters based on fitting the model to experimental data. Even though we did not have initial values for these parameters they still should have been listed in our model description and should have been assigned an expected value range.

In many cases, because tissue evolution and our model are stochastic, we will need to run the same simulation multiple times for a given set of parameters, or we will run many simulations with different parameters to explore the parameters’ effects on the results, or we may run related simulations embodying different choices of hypotheses.

We have now reached a point in the modeling workflow where there are two key issues; First we should revisit the entire model development workflow and second, we should determine if the computable model can answer our initial hypothesis and questions.

For the first question, we should revisit the entire workflow processes and review any assumptions we have made. Are the assumptions made during the early part of the model development process still viable in the final model? Has the final model raised critical issues such as unobtainable parameters or poorly defined initial, boundary or final condition? Each of the three model levels has its own set of issues that should be revisited. Working backwards through the development process we should consider;

* Computational Model and Implementation dependence:
  + Effect of discretization step size in both time and space
  + Implementation specific parameter reliability and sensitivity
* Quantitative model:
  + Are the time scales for the various processes comparable? Is a time scale separation possible that would simplify the model?
  + Are the quantitative model parameters obtainable or calculable?
* Biological Model:
  + Are there missing or extraneous components in the model

For the second question we should determine if the computable model is *capable* of answering or initial hypothesis. Note that here we are not asking if the model *does* answer the question, instead we are just asking if it *can* answer the question. We should examine the model outputs and determine if they are *comparable* to some experimentally observed data. If there are no model outputs that can be directly mapped to experimental data, then we have no way to verify the basic functioning of the model.

# Conclusion

We have outlined a workflow for developing defensible, mechanism-based computational biology models. Our approach requires discipline and progression through a series of logical steps. If followed, this orderly approach can directly couple biological concepts to their instantiation in computer code. Furthermore, the workflow greatly simplifies the task of explicitly defining what is, and is not, included in a model. Simplifications and assumptions made in the process are recorded. Furthermore, this approach greatly facilitates establishing the credibility of a model.

References

Hucka M, et al. (2003). The Systems Biology Markup Language (SBML): A medium for representation and exchange of biochemical network models. Bioinformatics, vol. 19, no. 4, pp. 524–531.

Mulugeta L, Drach A, Erdemir A, Hunt CA, Horner M, Ku JP, Myers JG Jr., Vadigepalli R and Lytton WW (2018a) Credibility, Replicability, and Reproducibility in Simulation for Biomedicine and Clinical Applications in Neuroscience. Front. Neuroinform. 12:18. doi: 10.3389/fninf.2018.00018

Mulugeta L, Erdemir A, Eckmann DM, Peng GCY, Vadigepalli R, An G, Hunt CA, Steele MJ, Lytton WW, Morrison TM, Myers JG, Pradhan GN, Lochner DR, Walton ME, Horner M, and Ku JP. (2018b). “Guidelines for Credible Practice of Modeling and Simulation in Healthcare” <https://www.imagwiki.nibib.nih.gov/sites/default/files/Poster%20IMAG_MSM2017_2.pdf> Retrieved from the web 23 April 2018.

Osborne, JM, Fletcher, AG, Pitt-Francis, JM, Maini, PK, & Gavaghan, DJ. (2017). Comparing individual-based approaches to modelling the self-organization of multicellular tissues. *PLoS computational biology* **13**, e1005387.

Patterson, E. & Whelan, MP (2017). A framework to establish credibility of computational models in biology. Progress in Biophysics and Molecular Biology, 129, pp.13–19.

Whetzel PL, Noy NF, Shah NH, Alexander PR, Nyulas C, Tudorache T, Musen MA. (2011) BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. Nucleic Acids Res. 2011 Jul;39(Web Server issue):W541-5. Epub 2011 Jun 14.