

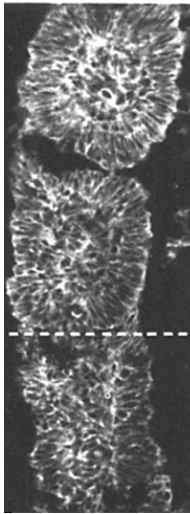
# The Cell Behavior Ontology: describing the biological behaviors of real and simulated cells seen as spatially active agents

James P. Sluka\*, Sherry G. Clendenon,  
Maciej Swat and James A. Glazier

Biocomplexity Institute, Indiana University,  
Bloomington, Indiana, USA

# Long Term Vision: Common annotation across multiple data sources (Somitogenesis Example)

## Image Data



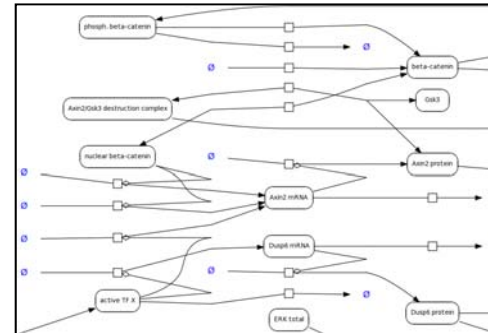
## uArray Data

Table 1

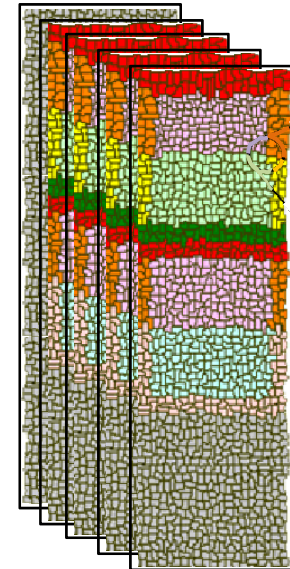
Genes differentially expressed between psm and somite I-V identified by microarray and independently confirmed.

PSM <sup>a</sup>	
Similar to KIAA0314	unknown
Unknown clone H3121H10	unknown
<i>Troy</i>	Signal Transduction
<i>Msg1</i>	Transcription
<i>Sm3B</i>	Transcription
<i>Fibronectin</i> <sup>b</sup>	ECM/Cytoskeletal
SOM I-V <sup>c</sup>	
RIKEN cDNA 1010001J06	unknown
<i>p8</i>	unknown function
<i>Mest</i>	unknown function
<i>Clim2</i>	Transcription
<i>Nab1</i>	Transcription
<i>Ech1</i>	Transcription
<i>Zfp217</i>	Transcription
<i>Ran BP8</i>	Nuclear Entry/Export
<i>Arl4</i>	Nuclear Entry/Export
<i>Gfp1</i>	ECM/Cytoskeletal
<i>Caveolin1</i>	ECM/Cytoskeletal
<i>Selp</i>	ECM/Cytoskeletal
<i>Collagen 2a</i> <sup>d</sup>	ECM/Cytoskeletal
<i>Collagen 9a</i> <sup>e</sup>	ECM/Cytoskeletal

## SBML Model

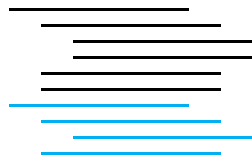


## CC3D Model



## Semantic Markup

Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis



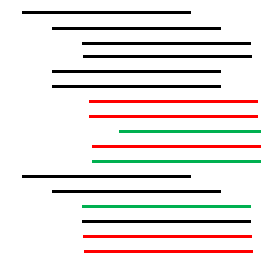
Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis



Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis

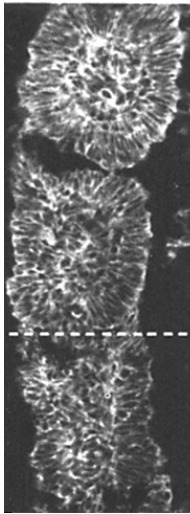


Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis



# Long Term Vision: Common annotation across multiple data sources (Somitogenesis Example)

## Image Data



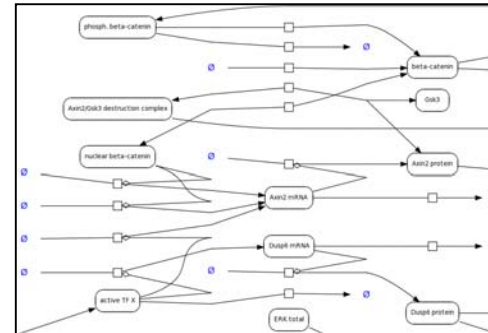
## uArray Data

Table 1

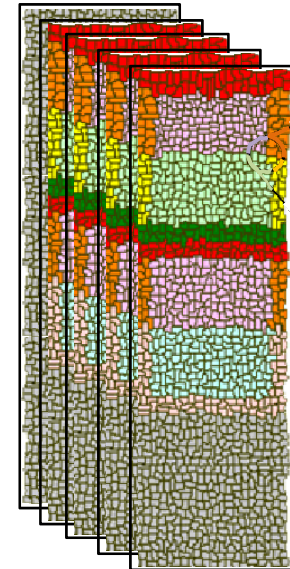
Genes differentially expressed between psm and somite I-V identified by microarray and independently confirmed.

PSM <sup>a</sup>	
Similar to KIAA0314	unknown
Unknown clone H3121H10	unknown
<i>Troy</i>	Signal Transduction
<i>Msg1</i>	Transcription
<i>Sm3B</i>	Transcription
<i>Fibronectin</i> <sup>b</sup>	ECM/Cytoskeletal
SOM I-V <sup>c</sup>	
RIKEN cDNA 1010001J06	unknown
<i>p8</i>	unknown function
<i>Mest</i>	unknown function
<i>Clim2</i>	Transcription
<i>Nab1</i>	Transcription
<i>Ech1</i>	Transcription
<i>Zfp217</i>	Transcription
<i>Ran BP8</i>	Nuclear Entry/Export
<i>Arl4</i>	Nuclear Entry/Export
<i>Gfp1</i>	ECM/Cytoskeletal
<i>Caveolin1</i>	ECM/Cytoskeletal
<i>Selp</i>	ECM/Cytoskeletal
<i>Collagen 2a<sup>d</sup></i>	ECM/Cytoskeletal
<i>Collagen 9a<sup>e</sup></i>	ECM/Cytoskeletal

## SBML Model

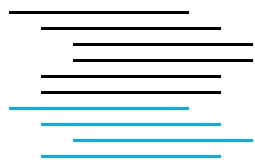


## CC3D Model

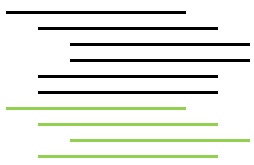


## Semantic Markup

Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis



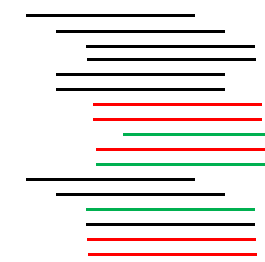
Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis



Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis



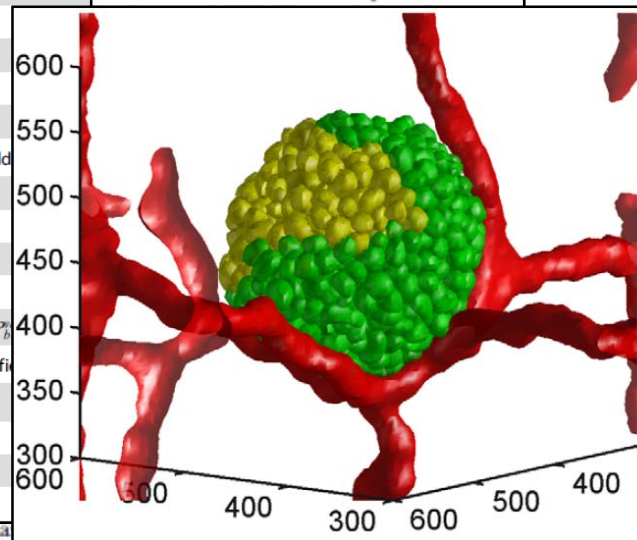
Species: chicken  
Process: embryogenesis  
Sub process: somitogenesis



Common markup  
describing the  
*biological system*

# Typical *ad hoc* biomodel publication modality

Cells	Behaviors
Tumor cells	
Normal	-proliferate
	-consume oxygen
	-change to hypoxic
	-change to necrotic
Hypoxic	-proliferate
	-consume oxygen field
	-change to normal
	-change to necrotic
	-secrete long-diffusing proangiogenic field
Necrotic	-shrink
	-disappear
Endothelial cells	
Vascular	-consume oxygen field
	-supply oxygen field at partial pressure $P_b$
	-secrete short-diffusing chemoattractant field
	-chemotax up gradients of field $C(i)$
	-elastically connect to neighboring
	vascular and inactive neovascular cells
	-lose elastic connections, when $l > l_{max}$



```
class MitosisSteppable(MitosisSteppableBase):
    def __init__(self, _simulator, _frequency=1):
        MitosisSteppableBase.__init__(self, _simulator, _frequency)
    def step(self, mcs):
        cells_to_divide=[]
        for cell in self.cellList:
            if cell.type == 1 and cell.volume>64:
                cells_to_divide.append(cell)
            if cell.type== 4 and cell.volume>128:
                cells_to_divide.append(cell)
        for cell in cells_to_divide:
            self.divideCellRandomOrientation(cell)
        def __init__(self):
            parentCell=self.mitosisSteppable.parentCell
            childCell=self.mitosisSteppable.childCell
            parentCell.targetVolume=parentCell.targetVolume/2
            parentCell.lambdaVolume=parentCell.lambdaVolume
            childCell.type=parentCell.type
            childCell.targetVolume=parentCell.targetVolume
            childCell.lambdaVolume=parentCell.lambdaVolume
```

Zheng et al. [28] has a method to model solid tumor growth in combination with Anderson and Chaplain's hybrid model of angiogenesis [24]. Zheng's model treats tumor cells as a viscous fluid flowing through a porous medium obeying the Darcy-Stokes law. Zheng et al. have shown that both diffusional instability (competition of growth and surface tension) and co-option of the new anastomosed capillaries may be key glioma invasion mechanisms. Frieboes et al. [29] have used Zheng's level-set method in combination with Plank and Sleeman's hybrid continuum-discrete [30], lattice-free model of tumor angiogenesis to model the physiology and evolution of glioma neovasculature in 3D. Frieboes et al's model allowed them to correlate measurable tumor microenvironment parameters to cell phenotypes and potentially to tumor-scale growth and invasion. Cristini et al. [31] have also developed a continuum

assume that oxygen serves as the single limiting factor. 1) We neglect the distinction between veins and arteries, and the possible proliferation. 2) Since oxygen-dependent and energy-dependent processes are neglected, we assume that oxygen serves as the single limiting factor. 3) We neglect the local concentration of oxygen. 4) We neglect the consumption of oxygen by tumor cells. 5) We neglect the concentration in the capillaries is constant, neglecting vessel diameter, blood flow rate and external pressure. 6) We assume that oxygen concentration in the host tissue and tumor. 7) We caricature the signaling pathways as constant-rate secretion of a single long-diffusion-length isoform of VEGF. Since we do not model blood flow explicitly, we neglect its biomechanically important effects on vascular remodeling and the maturation of nascent blood vessels. 9) Rather than model tip-cell selection explicitly, we distribute a certain number of inactive

- Paper prose
- Paper figure
- Paper math
- Code
- Results

*Often don't agree*

# The view from 10,000m

## Fundamental physical processes

- Movement
  - translation
  - rotation
  - diffusion
- Shape change
- Volume change
  - expansion
  - contraction
- Creation
- Deletion
- Phenotypic change
- Barrier crossing

# What is an ontology *really*?

*An ontology is a particular view of reality, not necessarily complete nor comprehensive, that encompasses a defined set of objects, processes and relationships within that reality.*

“Ontological Commitment”

Controlled Vocabulary	Hierarchy of Terms (isA)	Full Ontology
Cell Hepatocyte Leukocyte Organ Heart Liver	1. Cell a. Hepatocyte b. Leukocyte 2. Organ a. Heart b. Liver	1. Cell a. Hepatocyte b. Leukocyte 2. Organ a. Heart b. Liver <i>← partOf</i>

# CBO

- CBO is an OWL Ontology in Protégé 4 (Reasoners!)
- Based on Basic Formal Ontology (BFO)
- CBO currently contains 210 terms and 42 properties
  - Physical objects (cell, basement membrane, ..)
  - Qualities of physical objects (location, volume, ...)
  - Processes (cell growth, apoptosis, ...)
  - Qualities of processes (equations, parameters, ...)

The cell behavior ontology: describing the intrinsic biological behaviors of real and model cells seen as active agents. Sluka JP, Shirinifard A, Swat M, Cosmanescu A, Heiland RW, Glazier JA. *Bioinformatics*. 2014 Aug 15;30(16):2367-74.

<http://cbo.biocomplexity.indiana.edu/cbo/>

[https://\*\*bioportal\*\*.bioontology.org/ontologies/CBO](https://bioportal.bioontology.org/ontologies/CBO)



# High-level classes in the CBO and corresponding BFO classes

CBO Class	Subclass of BFO Class
<b>Thing</b>	
<b>CBO_Object</b>	<b>BFO:snap (continuant)</b>
BioEntityType	object
▶ ExtracellularMatrix	
ExtraCellularFluid	
Molecule	
Cell	
▶ System	object
▶ Field	object
▶ PhysicalObjectQuality	quality
▶ RateFunction	quality
▶ SystemQuality	quality
▶ Energy	quality
▶ EnergyQuality	quality
▶ FieldQuality	quality
▶ CellState	quality
PhysicalEntityType	quality
CorpuscularEntity	
ContinuousEntity	
DiffuseEntity	
▶ GeometricalEntity	independent_continuant
▶ CellPart	specifically_dependent_continuant
▶ CellMembraneLocalStructure	specifically_dependent_cont.
<b>CBO_Process</b>	<b>BFO:span (occurent)</b>
▶ FundamentalPhysicalProcess	process
▶ CellProcess	process
▶ ExtracellularMatrixProcess	process
FieldProcess	process
StructuralProcess	process
▶ MoleculeProcess	process
▶ TransportProcess	process
▶ TemporalEntity	connected_temporal_region

CBO\_Object (BFO:snap) describes the physical entities and entity qualities of a biological model.

CBO\_Process (BFO:span) describes the processes in which CBO\_Object(s) participate.

Classes marked with ▶ have additional subclasses that are not shown



# CBO Object Properties and Data Properties

## Object Properties in CBO.

The screenshot shows the 'Object property matrix' window. The title bar indicates 'Object property matrix: has\_improper\_part'. The window contains a tree view on the left listing object properties, and a table on the right with columns 'Trans', 'Refl', and 'alt\_id'. The 'has\_improper\_part' property is selected in the tree, and its corresponding row is highlighted in the table.

Object Property	Trans	Refl	alt_id
topObjectProperty	<input type="checkbox"/>	<input type="checkbox"/>	
adjacent_to	<input type="checkbox"/>	<input type="checkbox"/>	"OBO_REL:0000012"
precedes	<input checked="" type="checkbox"/>	<input type="checkbox"/>	"OBO_REL:0000018"
preceded_by	<input checked="" type="checkbox"/>	<input type="checkbox"/>	"OBO_REL:0000017"
has_part	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	"OBO_REL:0000003"
has_improper_part	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	"OBO_REL:0000024"
has_integral_part	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
has_proper_part	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
contains	<input type="checkbox"/>	<input type="checkbox"/>	
derives_from	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
participates_in	<input type="checkbox"/>	<input type="checkbox"/>	
has_participant	<input type="checkbox"/>	<input type="checkbox"/>	
location_of	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
contained_in	<input type="checkbox"/>	<input type="checkbox"/>	
located_in	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
transformation_of	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
derived_into	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
is_a	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
instance_of	<input type="checkbox"/>	<input type="checkbox"/>	
has_Quality	<input type="checkbox"/>	<input type="checkbox"/>	
part_of	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

## Data Properties in CBO

The screenshot shows the 'Data property matrix' window. The title bar indicates 'Data property matrix: has\_Quality'. The window contains a tree view on the left listing data properties, and a table on the right with columns 'Func' and 'description'. The 'has\_Quality' property is selected in the tree, and its corresponding row is highlighted in the table.

Data Property	Func	description
topDataProperty	<input type="checkbox"/>	
hasFloatValue	<input checked="" type="checkbox"/>	"A single floating point number."
hasPointsValueSetND	<input checked="" type="checkbox"/>	"Data is a list of n-tuples (one n-tuple/line) listing the coordinates of the elements within a set. For example, the list ..."
hasParameterValueSetND	<input checked="" type="checkbox"/>	"Data is a list of n-tuples (one n-tuple/line) listing values of various kinds. For example, the list of interaction energi..."
hasVTKfieldID	<input checked="" type="checkbox"/>	"hasVTKfieldID is a CBO data property that links a field prototype to the character string name used in the VTK file..."
hasIntValue	<input checked="" type="checkbox"/>	"A single integer number."
hasLocationValueSet2D	<input checked="" type="checkbox"/>	"2 Comma separated values (CSV). For example, the location of a point '3.2,-34.1'. Stored as a text string."
hasUnit	<input checked="" type="checkbox"/>	"A unit name, for example, 'meter', 'micrometer', 'hour'. Stored as a text string."
hasLocationValueSet3D	<input checked="" type="checkbox"/>	"3 Comma separated values (CSV). For example, the location of a point '3.2,2.4,-34.1'. Stored as a text string."
hasFloatValueTarget	<input checked="" type="checkbox"/>	"A single floating point number representing the target value of a quality or function."
hasIntValueTarget	<input checked="" type="checkbox"/>	"A single integer number representing the target value of a quality or function."
hasLocationValueSet1D	<input checked="" type="checkbox"/>	"1 value (included for completeness, probably best to just use hasFloatValue or hasIntValue). For example, the lo..."
hasVTKtypeID	<input checked="" type="checkbox"/>	"hasVTKtypeID is a CBO data property that links a cell prototype (or other physical entity prototype) to the integer ..."
hasExtentValueSet2D	<input checked="" type="checkbox"/>	"2 Comma separated values (CSV). For example, the extent of a 2D region (area) '5,10'. Stored as a text string..."
hasLocationValueSetND	<input checked="" type="checkbox"/>	"Comma separated values (CSV) with [CR] end of row markers. Dimension (rows and column counts) not explicit..."
hasExtentValueSet1D	<input checked="" type="checkbox"/>	"1 value (included for completeness, probably best to just use hasFloatValue or hasIntValue). For example, the 1 ..."
hasExtentValueSet3D	<input checked="" type="checkbox"/>	"3 Comma separated values (CSV). For example, the extent of a 3D region (volume) '5,5,10'. Stored as a text s..."

# Life on a lattice (or not)

**Multicell models are generally either:**

- Center based (cell is a point + radius)
- Lattice based (cell is a collection of pixels)

**Multicell models generally include:**

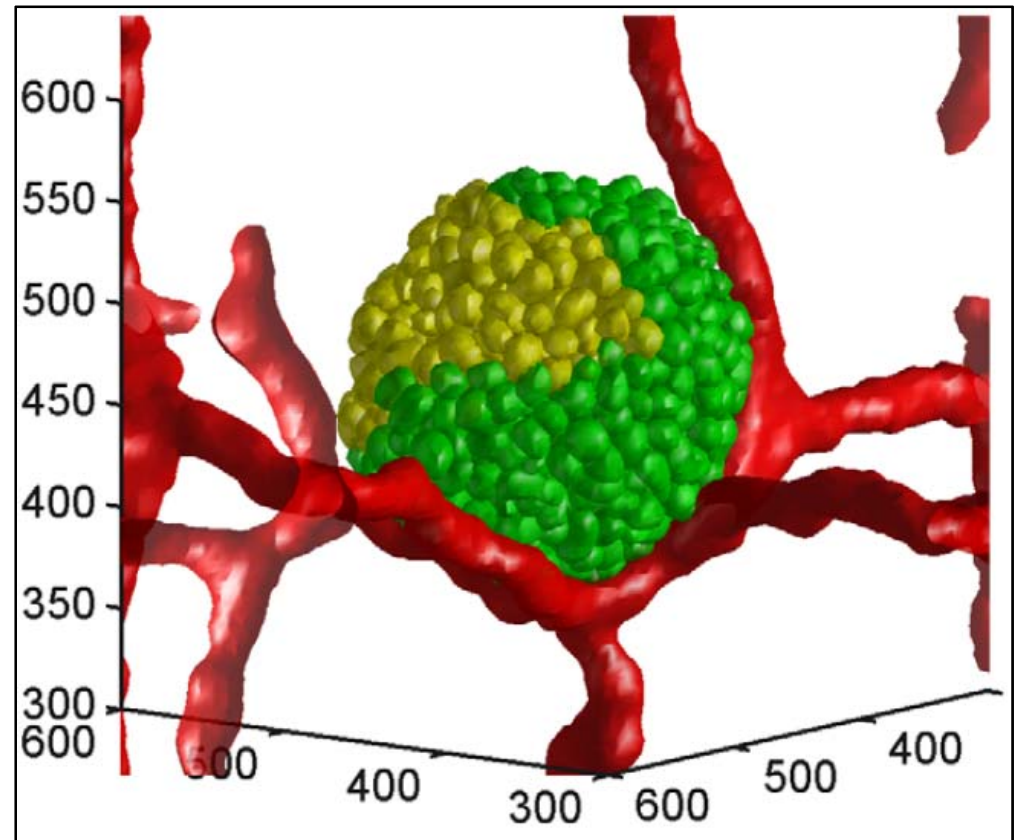
- Cells
  - Location, volume, surface, ...
  - Proliferation (creation)
  - Death (deletion)
  - Movement
  - Adhesion
- Non-cell materials
  - E.g. extracellular matrix
  - Chemical (diffusing) fields

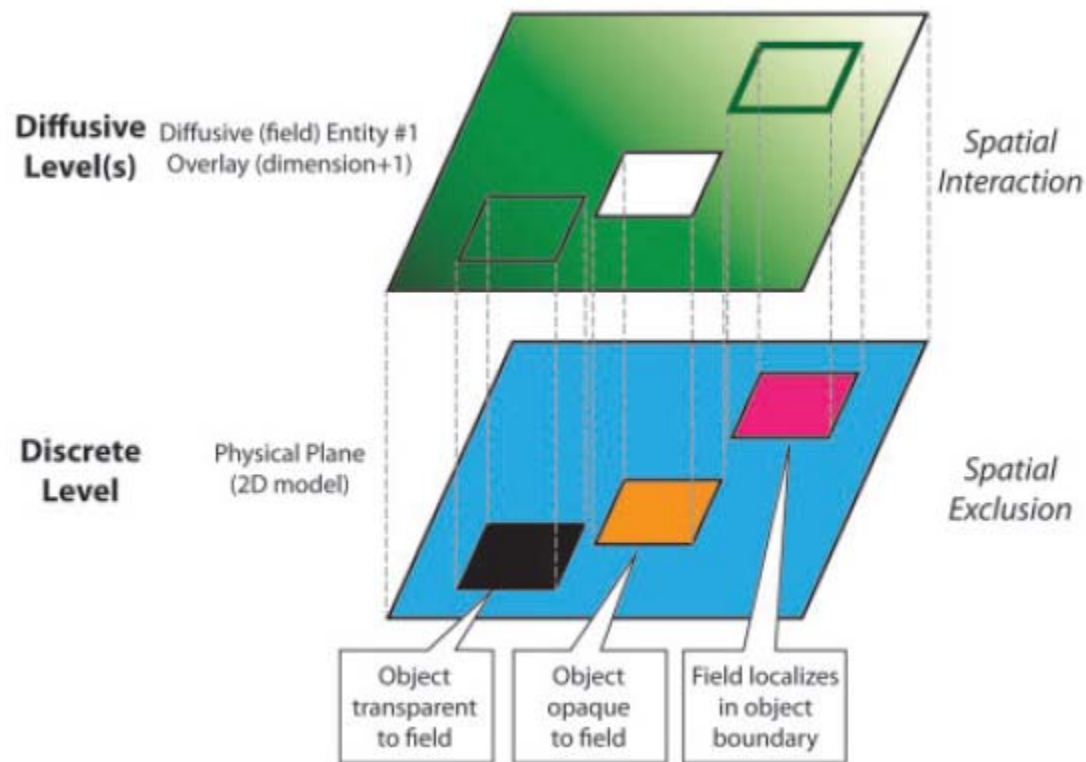
*Life doesn't exist on a lattice but relevant biological data is lattice based (e.g., microscope images)*

# Need to describe the *spatiality* of the system

- 2D and 3D cells and other objects
- Fields
- Time evolution of both

*Very large output data sets*



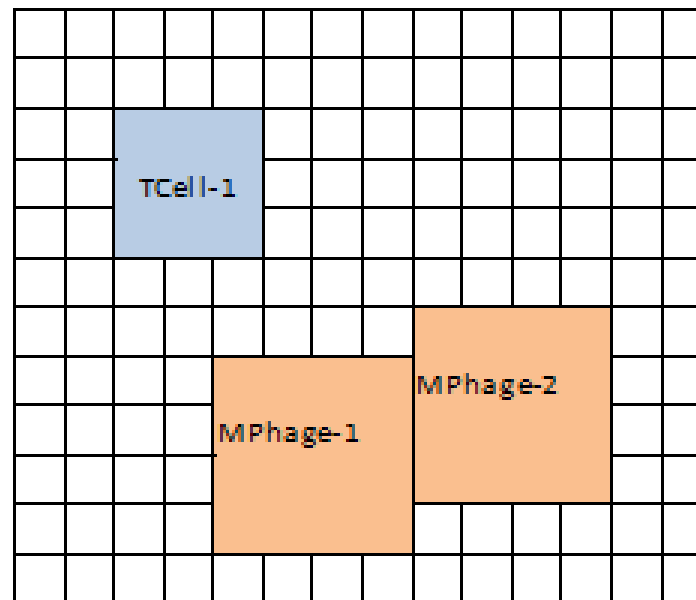


**2D Example:** The lower panel shows three CBO discrete physical objects (CBO:CorpuscularEntity) embedded in a blue field of medium (CBO:ContinuousEntity with the additional quality of a CBO:PhysicalObjectQuality:Fill). These objects are spatially exclusive. No two objects of these types can occupy the same region of space (pixel) at the same time.

The upper panel shows a CBO: Field, which is not spatially exclusive and by default can overlay any other object. A CBO:Field can interact with the discrete level.

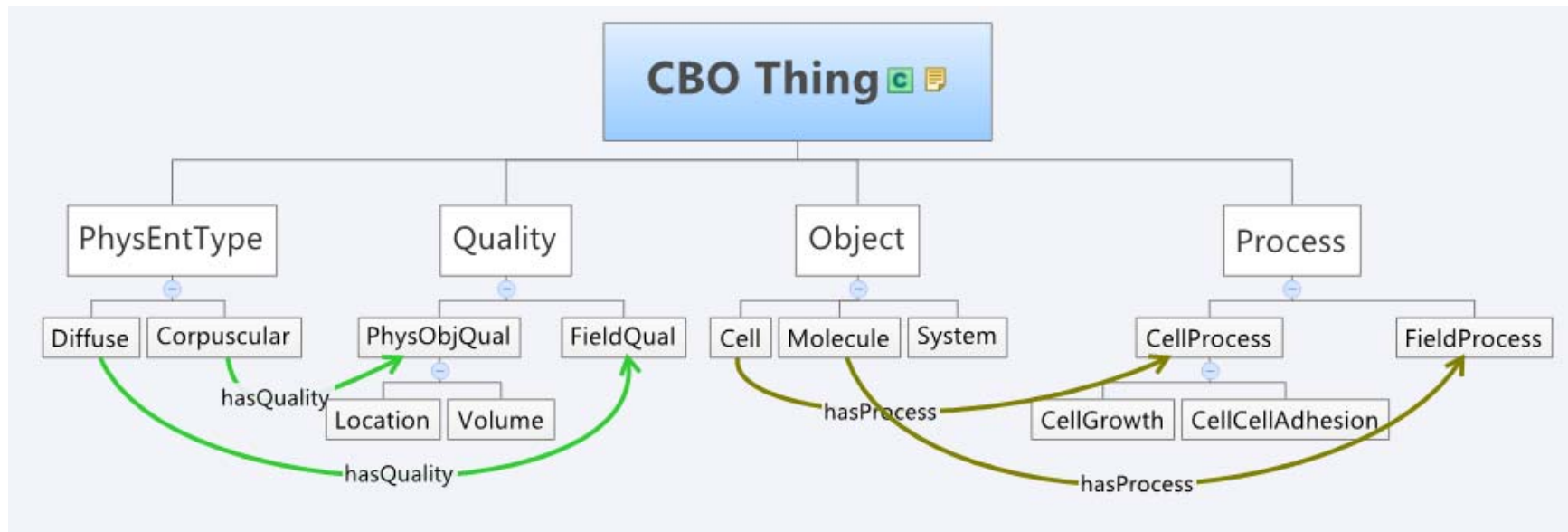
# Instantiate a simple model in CBO

Describe a model with two cell types and three cell instances including cell phenotypes, locations, volumes and adhesion.

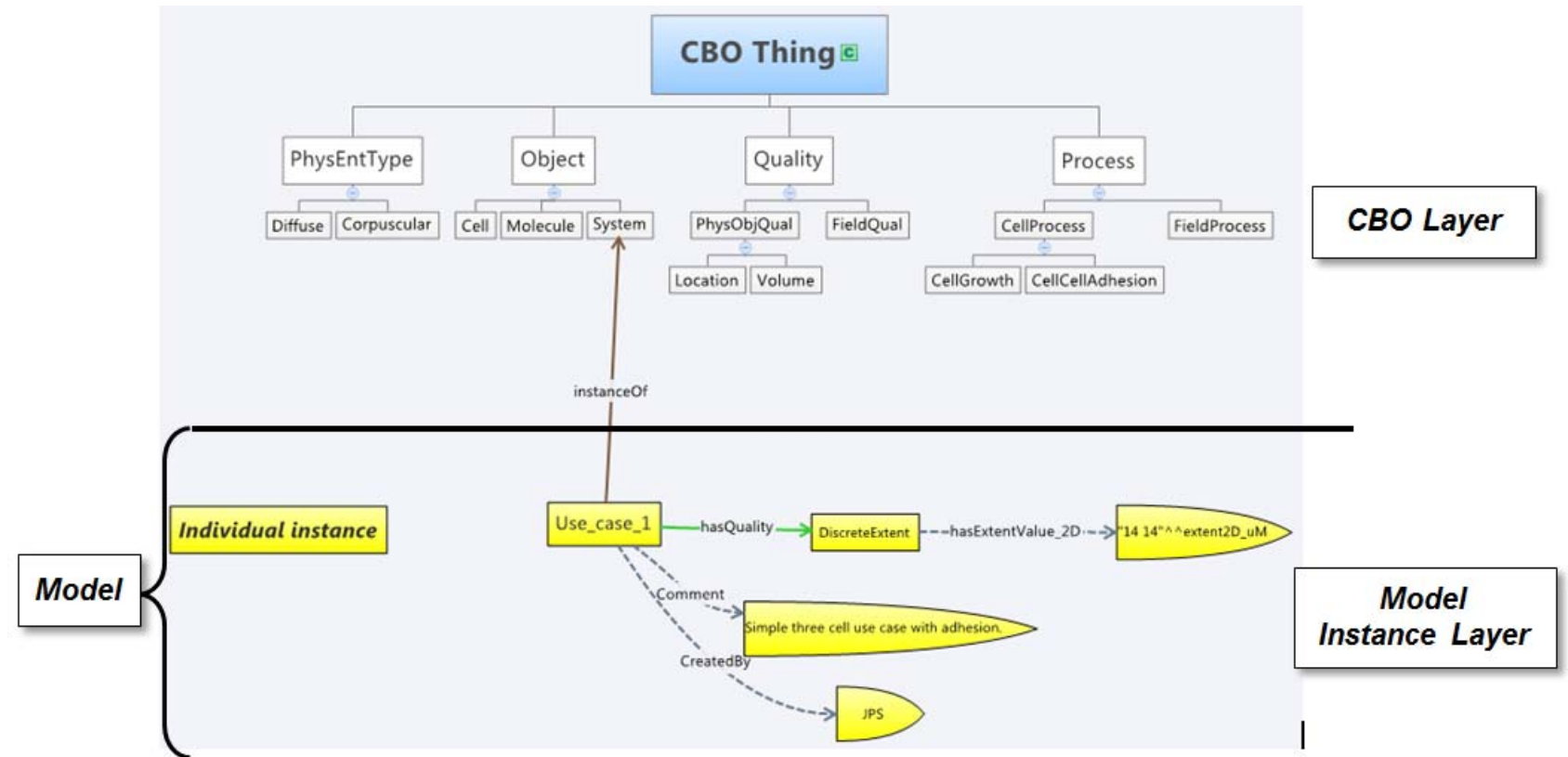


# Use Case:

- Simplified view of the CBO (17 of 194 classes shown)
- Unlabeled relationships are "isA".
- Indicated relationships may or may not be "hard coded" into the CBO.

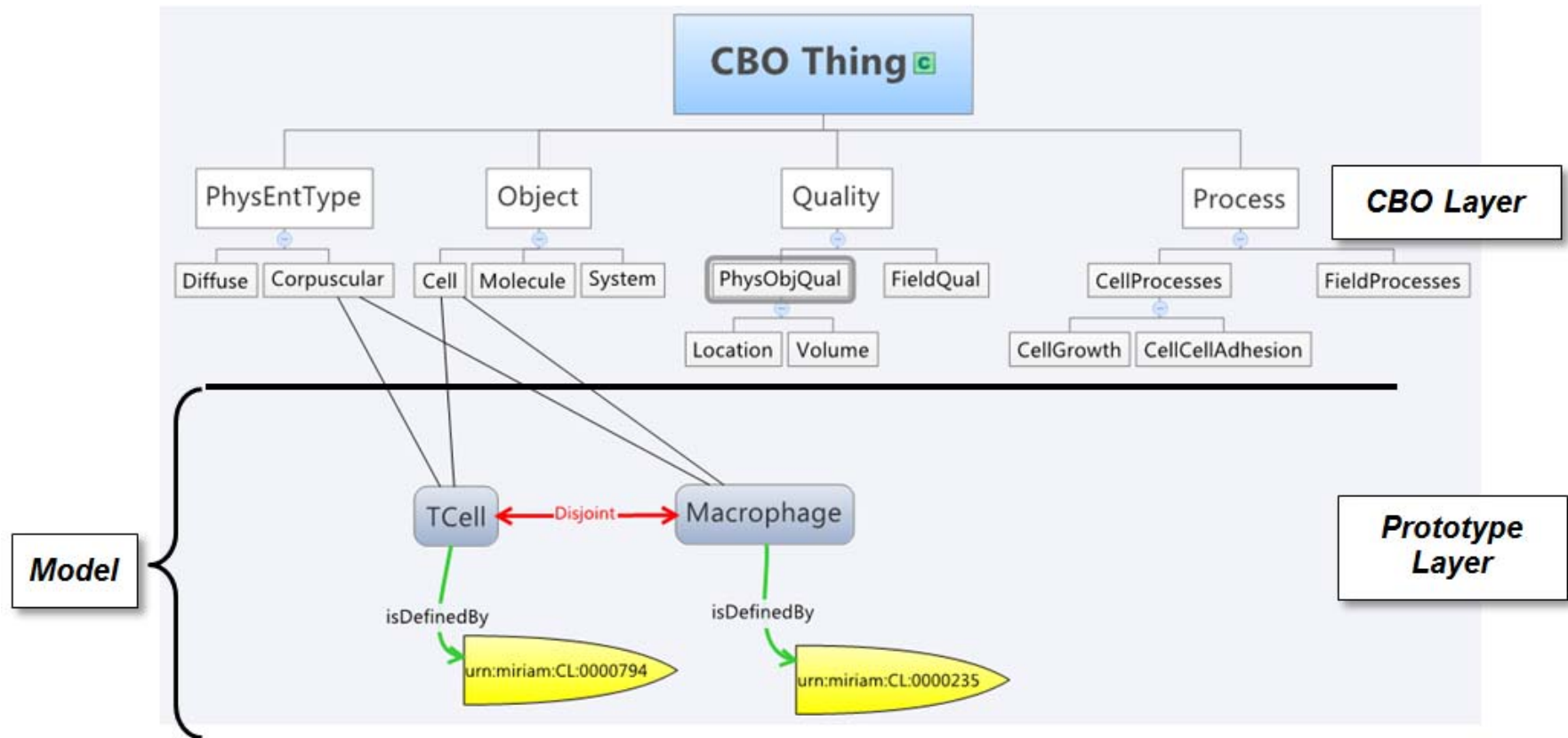


# Define the "system"

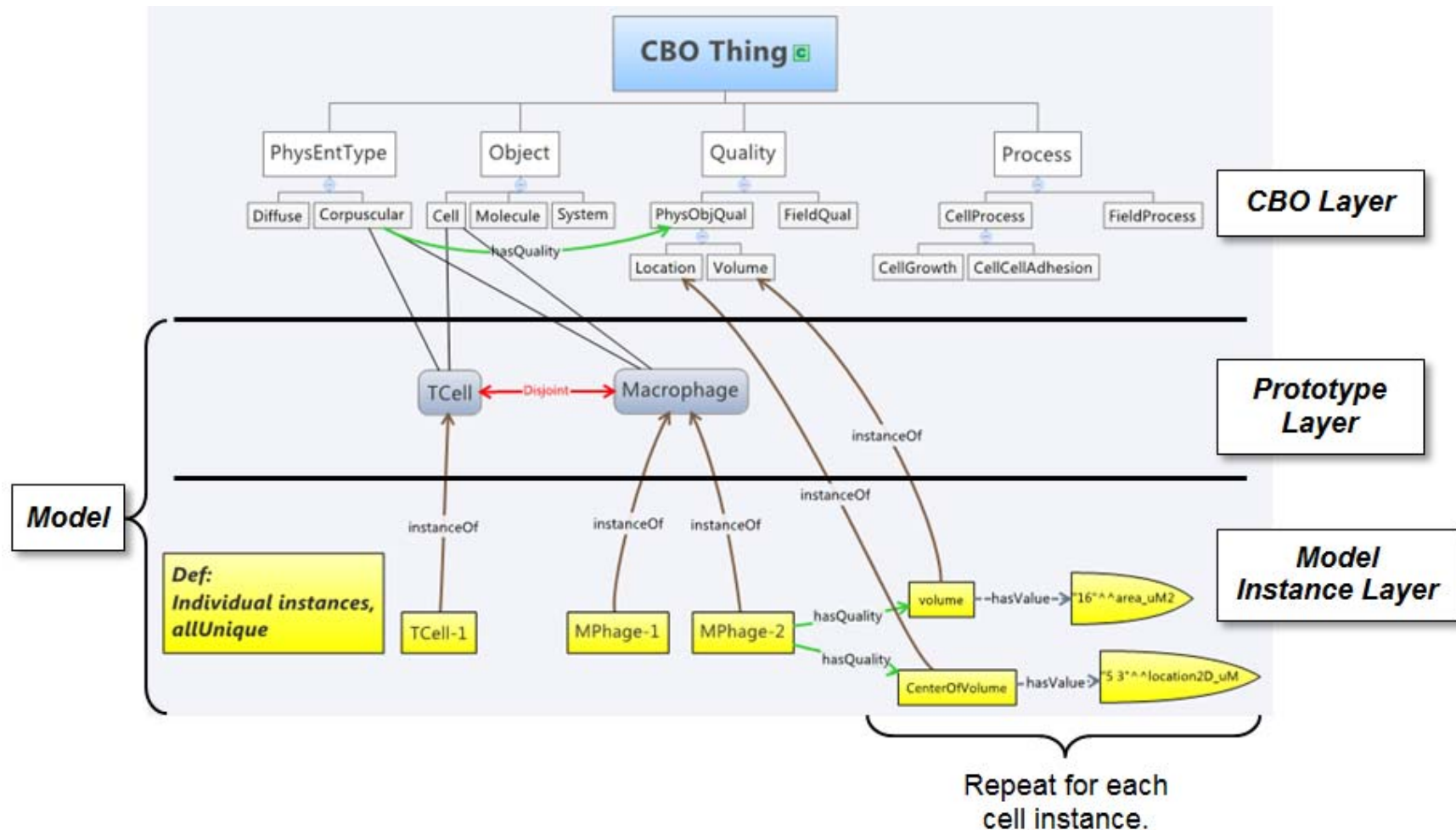




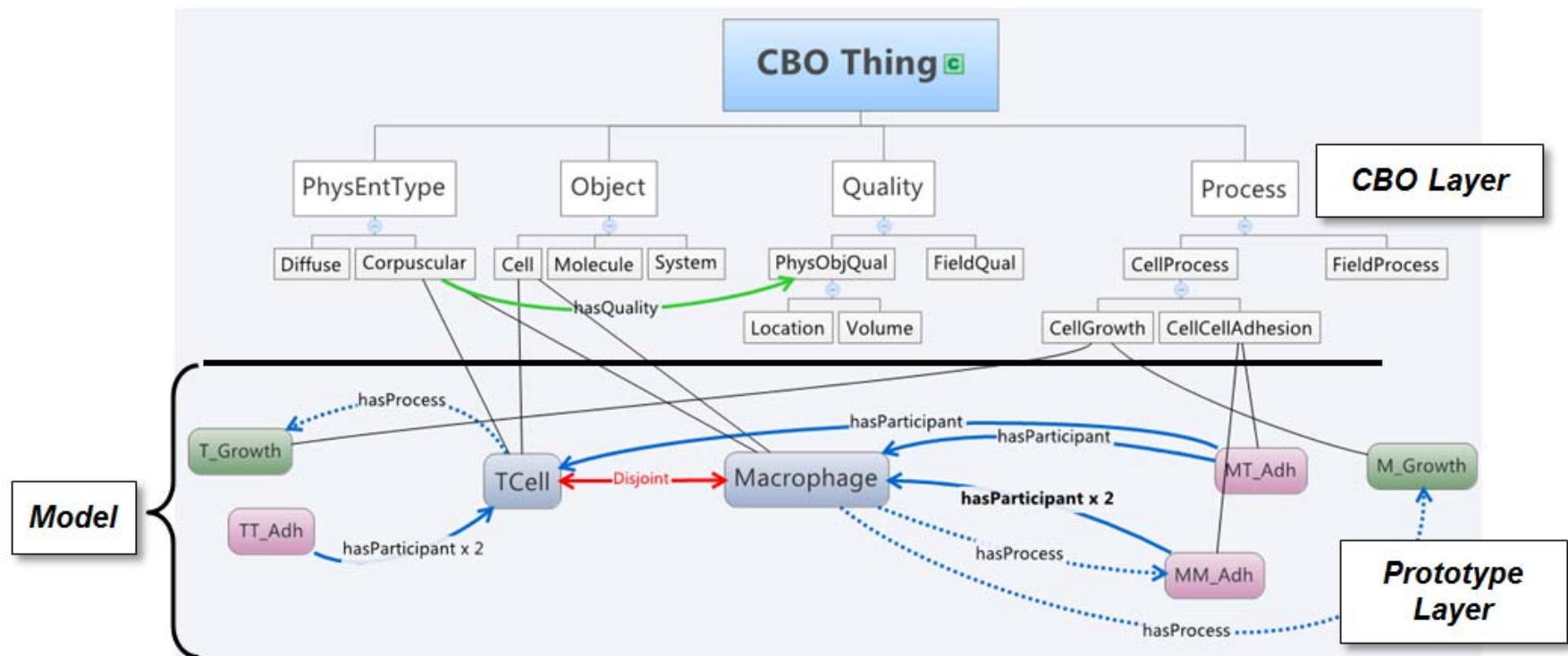
# Define prototype cell classes



# Define individual cell instances



# Processes are defined for prototype classes



# CBO annotation of the tumor growth with angiogenesis model

Cells	Behaviors	CBO Term
<b>Tumor cells</b>		isA Cell isA CorpuscularEntity
<b>Normal</b>	-proliferate	participates_in CellGrowth participates_in SymmetricCellDivisionRandom-CleavagePlane
	-consume oxygen field	participates_in MoleculeDeletion
	-change to hypoxic	participates_in PhenotypicChange
	-change to necrotic	participates_in PhenotypicChange
<b>Hypoxic</b>	-proliferate	participates_in CellGrowth participates_in SymmetricCellDivisionRandom-CleavagePlane
	-consume oxygen field	participates_in MoleculeDeletion
	-change to normal	participates_in PhenotypicChange
	-change to necrotic	participates_in PhenotypicChange
	-secrete long-diffusing proangiogenic field V	participates_in MoleculeDeletion
<b>Necrotic</b>	-shrink	participates_in CellVolumeChange
	-disappear	participates_in Necrosis
<b>Endothelial cells</b>		isA Cell isA CorpuscularEntity
<b>Vascular</b>	-consume oxygen field	participates_in MoleculeDeletion
	-supply oxygen field at partial pressure P	participates_in MoleculeCreation
	-secrete short-diffusing chemoattractant field C	participates_in MoleculeCreation
	-chemotax up gradients of field C	participates_in Chemotaxis
	-elastically connect to neighboring vascular and inactive neovascular cells	participates_in CellCellAdhesion has_part CellPart:CellMembranePart:TightJunction
<b>Fields</b>		
<b>Oxygen</b>		isA Molecule, isA DiffuseEntity, isA MolecularField participates_in Diffusion, participates_in Creation participates_in Deletion

# Describe a molecular field

## Class: VEGF1

### Annotations:

dc:description "Endothelial-derived short diffusing VEGF isoform.",  
rdfs:isDefinedBy "http://purl.bioontology.org/ontology/MSH/D042461",  
rdfs:isDefinedBy "http://purl.obolibrary.org/obo/PR\_000017284"

### SubClassOf:

MolecularField,  
Molecule,  
derives\_from some VEGF1\_creation,  
derives\_from some VEGF1\_deletion,  
DiffuseEntity

vascular endothelial  
growth factor A

## Class: VEGF1\_creation

### Annotations:

dc:description "Process that describes the creation of the VEGF1 field.",  
rdfs:isDefinedBy "http://purl.obolibrary.org/obo/GO\_0010573",

### SubClassOf:

MoleculeCreation,  
has\_participant some (NeoVascular or Vascular)

vascular endothelial  
growth factor production

## Class: VEGF1\_deletion

### Annotations:

rdfs:isDefinedBy "http://purl.obolibrary.org/obo/GO\_0010573",  
dc:description "Process that describes the deletion (consumption, degradation, clearance)  
of the VEGF1 field."

### SubClassOf:

has\_participant some ( (not (NeoVascular)) and (not (Vascular)) ),  
MoleculeDeletion

# Storing Lattice Data

For efficiency, data storage can be separate from the CBO-OWL meta-model.

Need a file format that:

- Can store arbitrary lattice data (many “color” channels, multidimensional, time)
- 2D and 3D with Time
- Would also like to be able to store shape primitives (line, plane, polygon)
- Store center models
- Has existing tools for visualization and manipulation
- Has a database option for storage

One solution has been implemented and one in the works:

- Visualization Tool Kit (VTK)
- Open Microscopy Environment (OME, OMERO & OMEtiff)

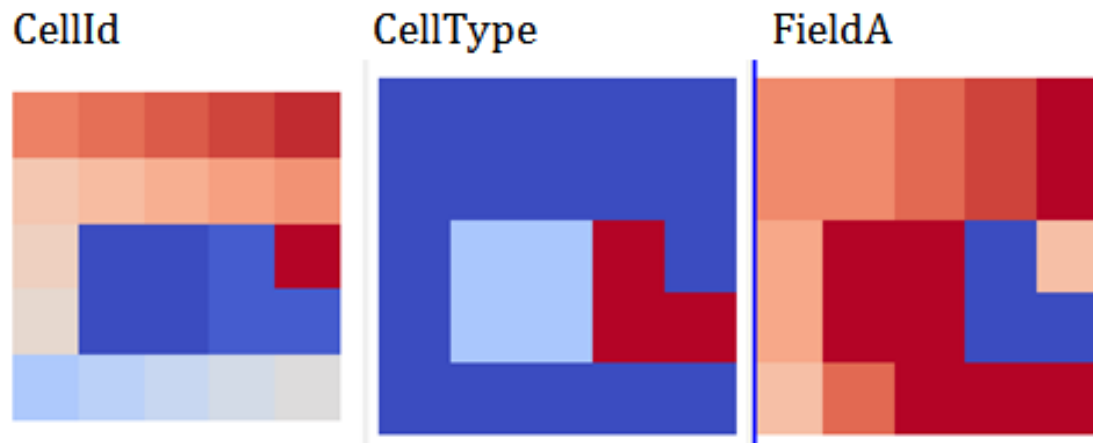
<http://www.vtk.org/>

<http://www.openmicroscopy.org>

# Storing Lattice Data in VTK

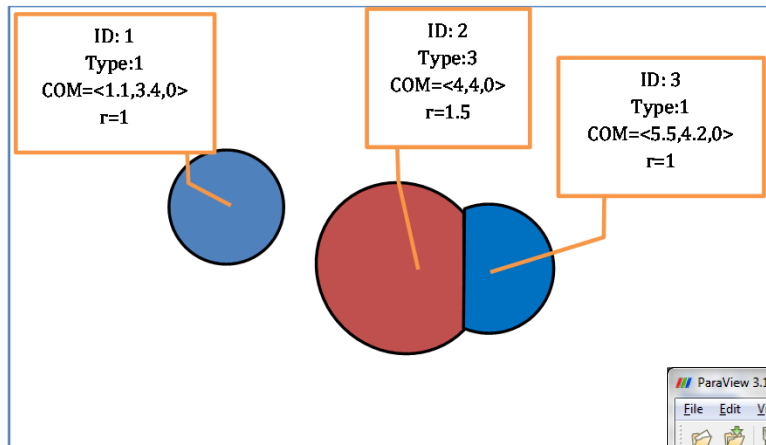
Cell ID Field "CellId"	Cell Type Field "CellType"	Field "FieldA"																																																																											
<table><tr><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td></tr><tr><td>15</td><td>1</td><td>1</td><td>2</td><td>2</td></tr><tr><td>16</td><td>1</td><td>1</td><td>2</td><td>27</td></tr><tr><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td></tr><tr><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td></tr></table> <p>NOTE: &lt;0,0&gt; is bottom left</p>	10	11	12	13	14	15	1	1	2	2	16	1	1	2	27	17	18	19	20	21	22	23	24	25	26	<table><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>4</td><td>4</td><td>12</td><td>12</td></tr><tr><td>0</td><td>4</td><td>4</td><td>12</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	4	4	12	12	0	4	4	12	0	0	0	0	0	0	0	0	0	0	0	<table><tr><td>1</td><td>1.3</td><td>1.5</td><td>1.5</td><td>1.5</td></tr><tr><td>1.1</td><td>1.5</td><td>1.5</td><td>0.1</td><td>0.1</td></tr><tr><td>1.1</td><td>1.5</td><td>1.5</td><td>0.1</td><td>1</td></tr><tr><td>1.2</td><td>1.2</td><td>1.3</td><td>1.4</td><td>1.5</td></tr><tr><td>1.2</td><td>1.2</td><td>1.3</td><td>1.4</td><td>1.5</td></tr></table>	1	1.3	1.5	1.5	1.5	1.1	1.5	1.5	0.1	0.1	1.1	1.5	1.5	0.1	1	1.2	1.2	1.3	1.4	1.5	1.2	1.2	1.3	1.4	1.5
10	11	12	13	14																																																																									
15	1	1	2	2																																																																									
16	1	1	2	27																																																																									
17	18	19	20	21																																																																									
22	23	24	25	26																																																																									
0	0	0	0	0																																																																									
0	4	4	12	12																																																																									
0	4	4	12	0																																																																									
0	0	0	0	0																																																																									
0	0	0	0	0																																																																									
1	1.3	1.5	1.5	1.5																																																																									
1.1	1.5	1.5	0.1	0.1																																																																									
1.1	1.5	1.5	0.1	1																																																																									
1.2	1.2	1.3	1.4	1.5																																																																									
1.2	1.2	1.3	1.4	1.5																																																																									

If we rendered these three fields in a tool such as ParaView (<http://www.paraview.org/>) top left corner in ParaView) we get graphical representations of the data in each field:



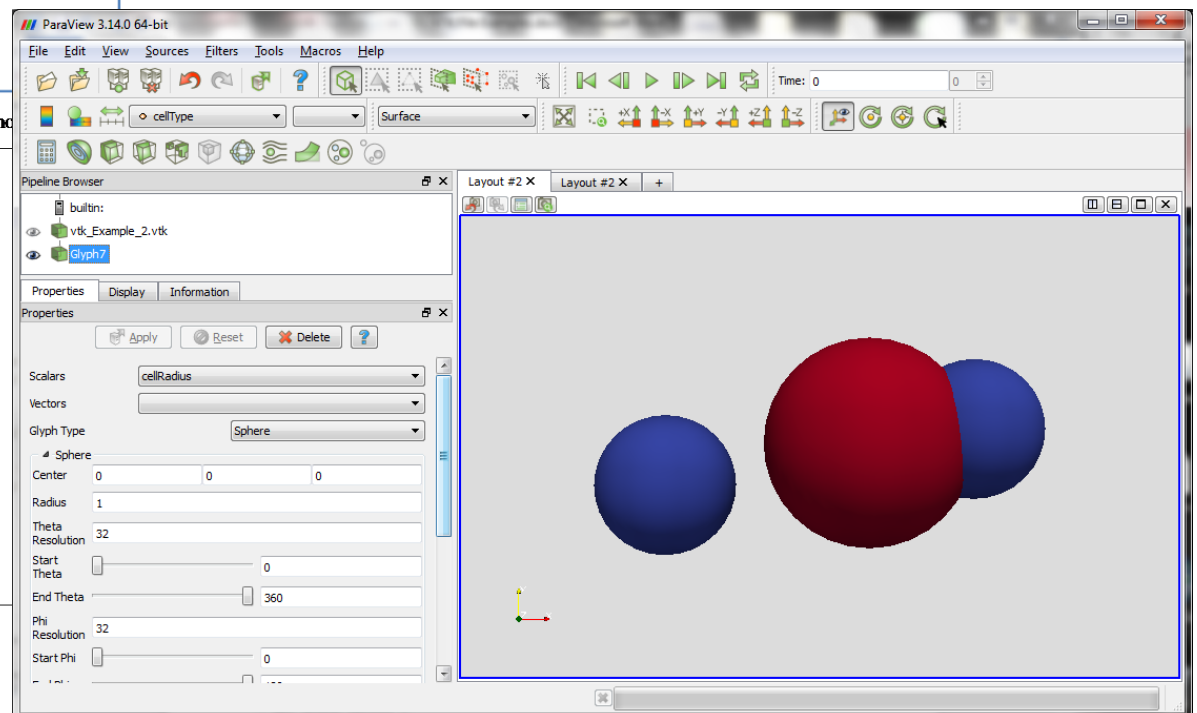


# VTK can also handle “center” data

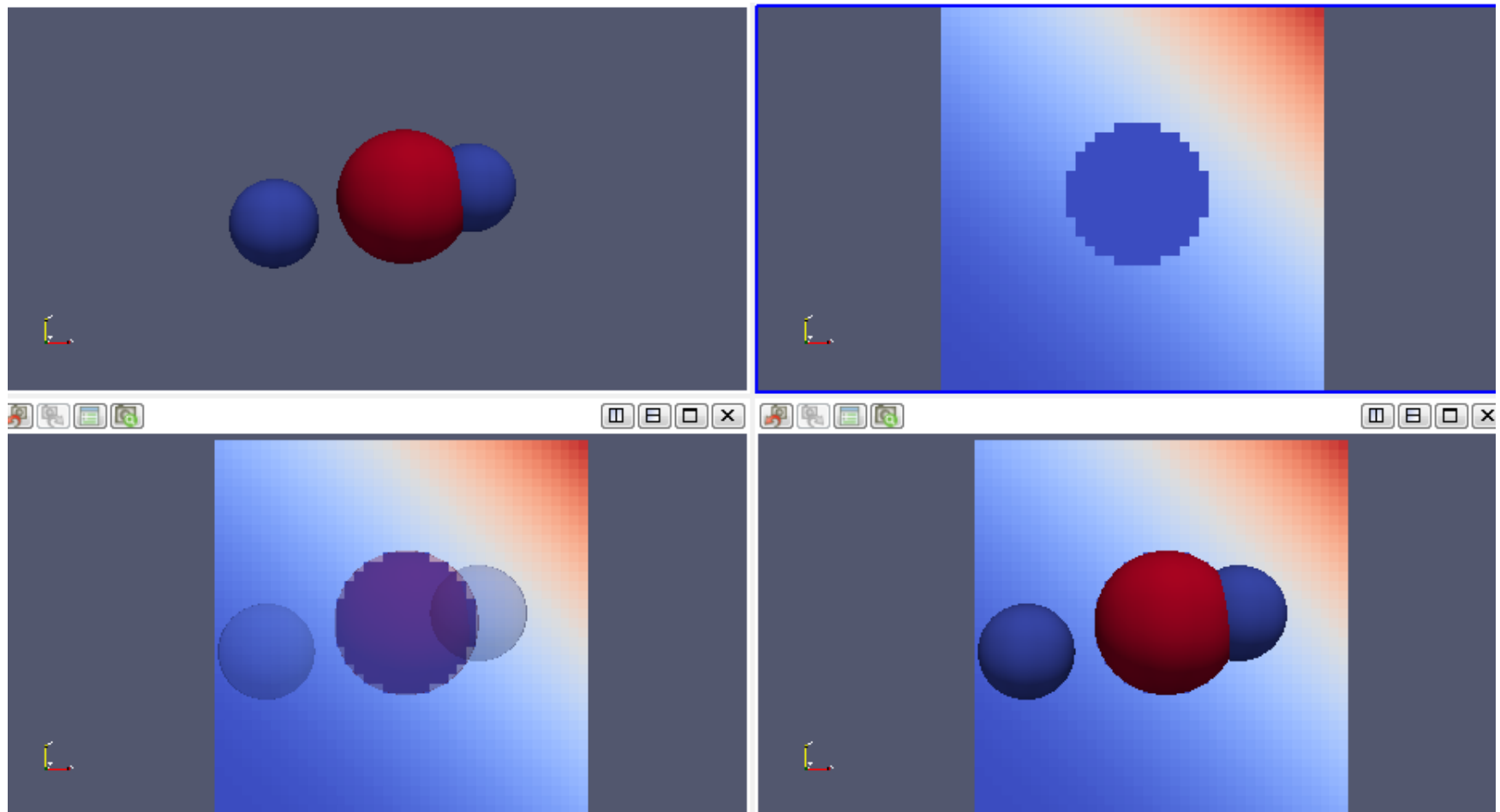


A VTK file for this model uses the "DATASET POLYDATA" file type and is shown below:

```
# vtk DataFile Version 2.0
Example 1: Lattice model with cell types and a field
ASCII
DATASET POLYDATA
POINTS 3 float
1.1 3.4 0.0
4.0 4.0 0.0
5.5 4.2 0.0
VERTICES 3 6
1 0
1 1
1 2
POINT DATA 3
SCALARS cellType int
LOOKUP_TABLE default
1 2 1
SCALARS cellID int
LOOKUP_TABLE default
1 2 3
SCALARS cellRadius float
LOOKUP_TABLE default
1.0 1.5 1.0
```



VTK can also handle “center” data  
and pixel based field data



# Open Microscopy Environment's OMERO

- Store:
  - Very large “image” datasets
  - Meta-data
- Rendering
- Analysis
- Access control and publication



The screenshot shows the OMERO web interface. At the top is the OMERO logo. Below it is a navigation bar with links: Home, Products, Documentation, Community, and About. A breadcrumb trail indicates the current location: Home → Products → OMERO. On the left is a sidebar with a 'Navigation' section containing links to OMERO, Feature List, Big Images Support, Volume Viewer in OMERO.web, Screenshots, OMERO Downloads, Legacy downloads, and Security Vulnerabilities. The main content area is titled 'About OMERO' and 'What is OMERO?'. The text describes OMERO as a secure central repository for microscopy images, accessible from various platforms, supporting over 140 image file formats.

Home Products Documentation Community About

You are here: Home → Products → OMERO

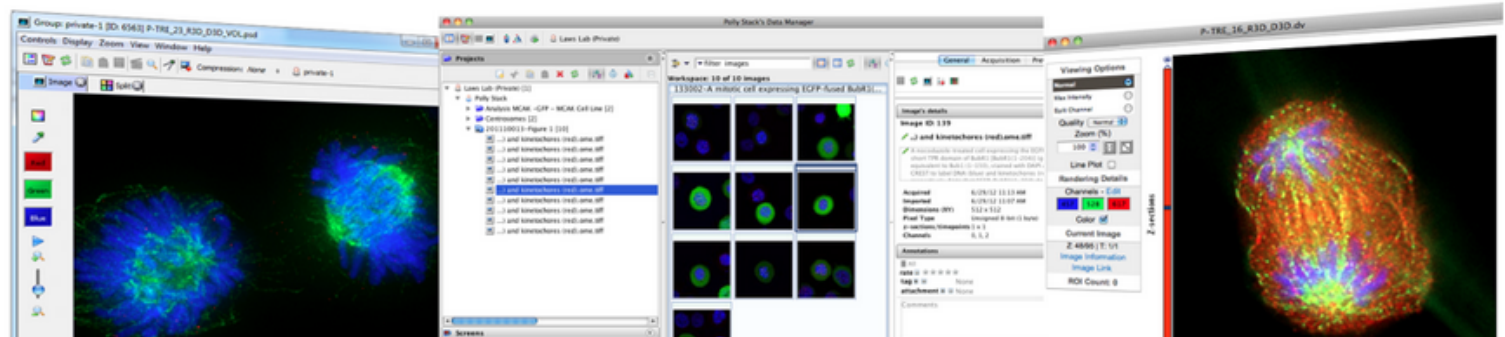
Navigation

- OMERO
- Feature List
- Big Images Support
- Volume Viewer in OMERO.web
- Screenshots
- OMERO Downloads
- Legacy downloads
- Security Vulnerabilities

## About OMERO

### What is OMERO?

From the microscope to publication, OMERO handles all your images in a secure central repository. You can view, organize, analyze and share your data from anywhere you have internet access. Work with your images from a desktop app (Windows, Mac or Linux), from the web or from 3rd party software. Over 140 image file formats supported, including all major microscope formats.



# CBO OWL Meta-models can be queried with SPARQL

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?s ?o
  WHERE { ?s rdfs:isDefinedBy ?o . }
ORDER BY ?s
```

s	
AdhesionMolecule	"urn:miriam:obo.fma:67214"@
Apoptosis	"urn:miriam:obo.go:0006915"@
Apoptosis	"PATO:0000638"@
AutophagicDeath	"urn:miriam:obo.go:0048102"@
BasalPart	"urn:miriam:obo.go:0045178"@
BasalPart	"urn:miriam:obo.fma:72558"@
BasementMembrane	"urn:miriam:obo.go:0005605"@
BodyShape	"urn:miriam:opb:'bounded Volume Shape'"@
BrushBorder	"urn:miriam:obo.go:0005903"@
BrushBorder	"urn:miriam:obo.fma:70977"@

# Extract all of the cells in the meta-model.

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX CBO: <http://cbo.biocomplexity.indiana.edu/svn/cbo/trunk/CBO_1_0.owl#>

SELECT ?subject ?pred ?idVal
WHERE {
    ?subject rdfs:subClassOf CBO:Cell .
    ?subject ?pred ?idVal .
    ?subject owl:equivalentClass ?idVal . }
```

Returns the prototype cells defined in the metamodel along with their VTK ids:

subject	pred	idVal
Vascular	equivalentClass	● hasVTKtypeID value 3
Proliferating	equivalentClass	● hasVTKtypeID value 1
Medium	equivalentClass	● hasVTKtypeID value 0
NeoVascular	equivalentClass	● hasVTKtypeID value 4
Necrotic	equivalentClass	● hasVTKtypeID value 2

# “Simple” Python programs can extract information from the CBO-OWL + simulation snapshot VTK files

Name Spaces found in the OWL file:

CBO_1_0	http://www.semanticweb.org/ontologies/2013/1/7/untitled-ontology-286#
CBO_1_02	http://cbo.biocomplexity.indiana.edu/svn/cbo/trunk/CBO_1_0.owl#
dc	http://purl.org/dc/elements/1.1/
oboInOwl	http://www.geneontology.org/formats/oboInOwl#
owl	http://www.w3.org/2002/07/owl#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema#

Searching for system parameters:  
Looking up unit names:  
http://purl.obolibrary.org/obo/UO\_0000017  
http://www.w3.org/2000/01/rdf-schema#label = micrometer  
http://purl.obolibrary.org/obo/UO\_0000031  
http://www.w3.org/2000/01/rdf-schema#label = minute

In Main():

Cell Type ID numbers

ID	Name
0	Medium
1	Proliferating
2	Necrotic
3	Vascular
4	NeoVascular

Field Names:

Glucose  
VEGF1  
VEGF2

System Parameters:

SystemPixelDistanceScale = 4.0  
SystemPixelDistanceScale unit = micrometer  
VTK\_FileTimeStep = 1.0  
VTK\_FileTimeStep unit = minute

Finished parsing OWL file: ... p/VascularTumor/AngioTumor.c

Processing VTK files, found 10 VTK files  
Processing VTK File = Step\_0000.vtk MCS= 0 Processing fields...

VTK 'fields' in the VTK files:

CellId  
CellType  
CellVolume  
ClusterId  
Glucose  
VEGF1  
VEGF2

Total Cell Volumes by cell type (micrometer^3):

MCS	minute(s)	Medium	Proliferating	Necrotic	Vascular	NeoVascular
0	0.0	12645120	41600	0	96576	16704
600	600.0	12537024	154048	0	91776	17152
1200	1200.0	12494912	187648	768	92352	24320
1800	1800.0	12448512	229888	0	92800	28800
2400	2400.0	12374784	297792	0	93120	34304
3000	3000.0	12288256	372160	0	92864	46720

Average field pixel values:

MCS	minute(s)	Glucose	VEGF1	VEGF2
0	0.0	4.79804	0.00008	0.00003
600	600.0	8.81348	0.00717	0.00651
1200	1200.0	9.51673	0.00753	0.00710
1800	1800.0	9.26159	0.00803	0.00795
2400	2400.0	8.48257	0.00838	0.00991
3000	3000.0	7.49264	0.00908	0.00988

Cell Counts:

MCS	minute(s)	Proliferating	Necrotic	Vascular	NeoVascular
0	0.0	64	0	24	4
600	600.0	64	0	24	4
1200	1200.0	63	1	24	5
1800	1800.0	87	0	24	5
2400	2400.0	102	0	24	6
3000	3000.0	111	0	24	10

# Thank You!

Support:

- US EPA
- NIH/NIGMS
- Indiana University
- NSF

