Biocellion Model Specification in XML

Biocellion models can be represented using the XML tags and attributes discussed in this document. Use `biocellion.py` to translate the XML model specification into C++ for compilation and execution.

1 XML Tags

The following tags are currently supported. Some tags are required.

1.1 `<?xml?>`

Begin every model XML file with the XML versioning tag.

```
<?xml version="1.0" encoding="utf-8" ?>
```

1.2 `<model>`

Every model must be contained in the `<model></model>` tag, with exactly one model per XML file. Each `<model>` may contain the following tags.

1.2.1 Allowed Subtags

Tag	Required Count	Notes
`<simulator>`	1	
`<input>`	0 or 1	Not supported yet.
`<solute>`	≥ 0	
`<molecule>`	≥ 0	
`<particle>`	≥ 0	
`<interaction>`	≥ 0	
`<world>`	1	
`<reaction>`	≥ 0	
`<molecularReactions>`	0	Not supported yet.
`<solver>`	≥ 1	
`<agentGrid>`	1	
`<species>`	≥ 0	

1.2.2 Allowed Attributes

No attributes are allowed.

1.2.3 Allowed Text Values

No text values are allowed, only subtags.

1.2.4 Example

```
<model>
  ...
</model>
```

1.3 `<param>`

`<param>` tags are used for specifying various parameters to define the model. For example, how many simulated hours to run the simulation. Parameters have a name attribute, and a text value. Most parameters also expect a unit attribute. Many unit conversions are supported.

1.3.1 Allowed Subtags

No subtags are allowed.

1.3.2 Allowed Parameters

No parameters are allowed.

1.3.3 Allowed Attributes

Attribute	Required	Notes
`name`	Yes	Name of the parameter.
`unit`	Recommended	Units of the value.

1.3.4 Allowed Text Values

The text values is usually a floating point number, but could be an integer, or a boolean specification.

1.3.5 Examples

```
<model>
  ...
  <simulator>
    <param name="outputPeriod" unit="hour">0.1</param>
    <param name="agentTimeStep" unit="hour">0.1</param>
    <param name="numStateAndGridTimeStepsPerBaseline">1</param>
```

```
      ...
    </simulator>
    ...
</model>
```

1.4 `<simulator>`

Every `<model>` must contain one `<simulator>` tag. This tag is used to control
high-level time related settings for the simulation.

1.4.1 Allowed Subtags

Tag	Required Count	Notes
`<param>`		See required parameters below.
`<timeStep>`	1	

1.4.2 Allowed Parameters

Parameter	Required	Type	Units	Notes
`outputPeriod`	Yes	float	time	Simulated time between data output.
`agentTimeStep`	Yes	float	time	Simulated time delta per baseline timestep
`numStateAndGridTimeStepsPerBaseline`	Yes	integer		Internal timesteps per baseline timestep.
`restartPreviousRun`	No	boolean		Not supported yet.
`randomSeed`	No	integer		Not supported yet.
`chemostat`	No	boolean		Not supported yet.
`diffusionReactionOnAgentTime`	No	boolean		Not supported yet.

1.4.3 Allowed Attributes

No attributes are allowed.

1.4.4 Allowed Text Values

No text values are allowed, only subtags.

1.4.5 Example

```
<model>
  ...
  <simulator>
    <param name="outputPeriod" unit="hour">0.1</param>
    <param name="agentTimeStep" unit="hour">0.1</param>
    <param name="numStateAndGridTimeStepsPerBaseline">1</param>
    ...
```

```
    </simulator>
  ...
</model>
```

1.5   `<timeStep>`

The `<simulator>` must contain one `<timeStep>` tag. This tag is used to control the end of the simulation.

1.5.1   Allowed Subtags

Tag	Required Count	Notes
`<param>`		See required parameters below.

1.5.2   Allowed Parameters

Parameter	Required	Type	Units	Notes
`endOfSimulation`	Yes	float	time	Simulated time to end the simulation.
`adaptive`	No	boolean		Not supported yet.
`timeStepIni`	No	float	time	Not supported yet.
`timeStepMin`	No	float	time	Not supported yet.
`timeStepMax`	No	float	time	Not supported yet.

1.5.3   Allowed Attributes

No attributes are allowed.

1.5.4   Allowed Text Values

No text values are allowed, only subtags.

1.5.5   Example

```
<model>
  ...
  <simulator>
    ...
    <timeStep>
      <param name="endOfSimulation" unit="hour">1.0</param>
    </timeStep>
  </simulator>
  ...
</model>
```