Generating OpenModelica Web Books Including Mathematical Typesetting from OMNotebooks

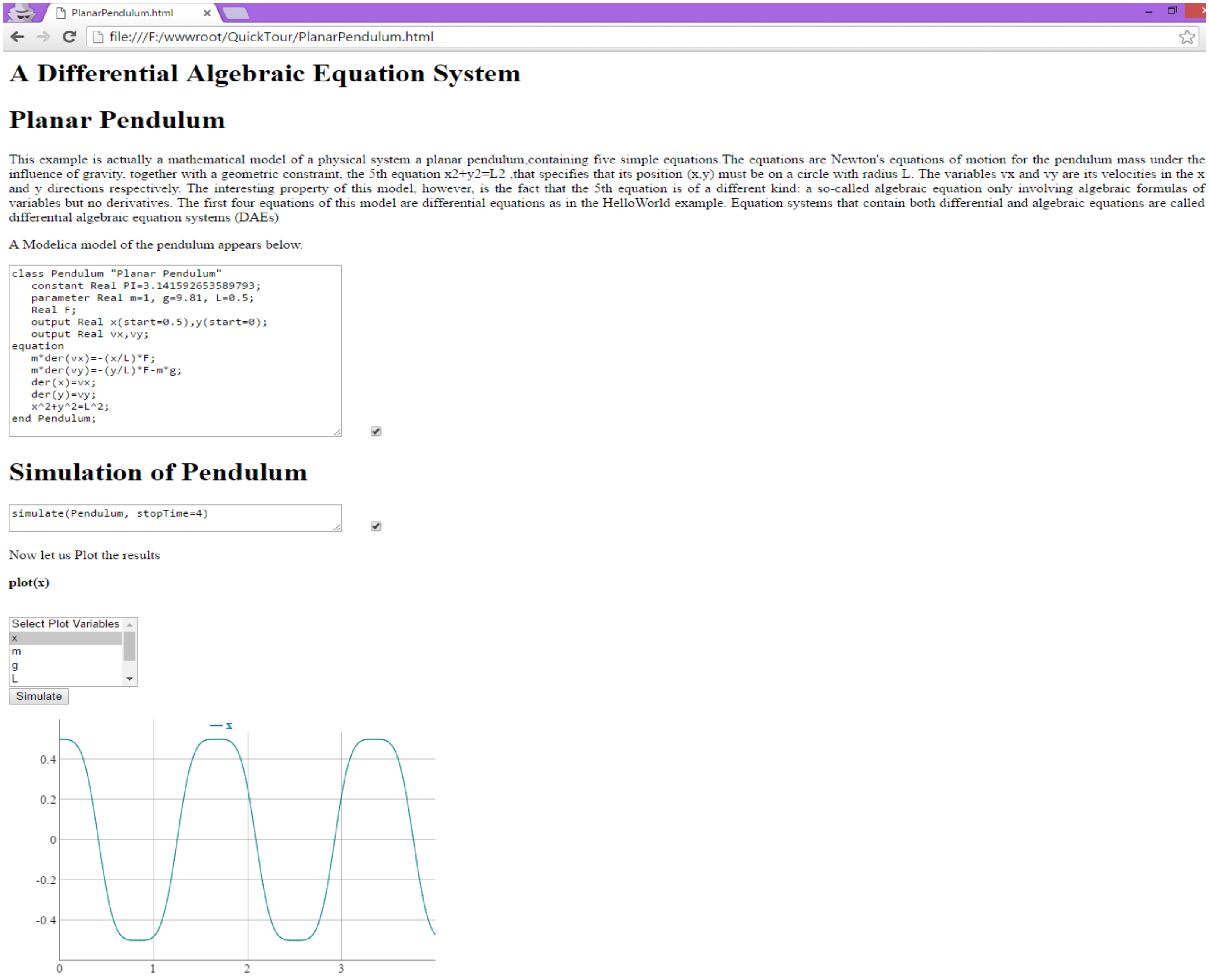
ARUNKUMAR PALANISAMY, MARTIN SJÖLUND,
AND PETER FRITZSON

LINKÖPING UNIVERSITY

Goals

- *Main goal*: **Easily accessible interactive Modelica course material on the web**
- *Interactive simulation* of Modelica models in a web browser.
- *Interactive editing* of Modelica models in a web browser
- *Automatic generation of web notebooks* from interactive OMNotebooks including text, mathematical formulae, and Modelica models

Example Web Book in Operation



A Differential Algebraic Equation System

Planar Pendulum

This example is actually a mathematical model of a physical system a planar pendulum,containing five simple equations.The equations are Newton's equations of motion for the pendulum mass under the influence of gravity, together with a geometric constraint, the 5th equation x2+y2=L2 ,that specifies that its position (x,y) must be on a circle with radius L. The variables vx and vy are its velocities in the x and y directions respectively. The interesting property of this model, however, is the fact that the 5th equation is of a different kind: a so-called algebraic equation only involving algebraic formulas of variables but no derivatives. The first four equations of this model are differential equations as in the HelloWorld example. Equation systems that contain both differential and algebraic equations are called differential algebraic equation systems (DAEs)

A Modelica model of the pendulum appears below.

```
class Pendulum "Planar Pendulum"
   constant Real PI=3.141592653589793;
   parameter Real m=1, g=9.81, L=0.5;
   Real F;
   output Real x(start=0.5),y(start=0);
   output Real vx,vy;
equation
   m*der(vx)=-(x/L)*F;
   m*der(vy)=-(y/L)*F-m*g;
   der(x)=vx;
   der(y)=vy;
   x^2+y^2=L^2;
end Pendulum;
```

☑

Simulation of Pendulum

```
simulate(Pendulum, stopTime=4)
```

☑

Now let us Plot the results

plot(x)

Current Status

- Ongoing Work , started in Mid November-2015
- Python script that automatically translates existing Notebook files to HTML files for the Web notebook
- Dynamic creation and evaluation of cells in a browser similar to OMNotebook
- Setup of local webserver and start simulate Modelica models in a browser

Technologies Used

- Client interface

  Javascript and HTML

- Server

  Python, OMPython, Python CGI

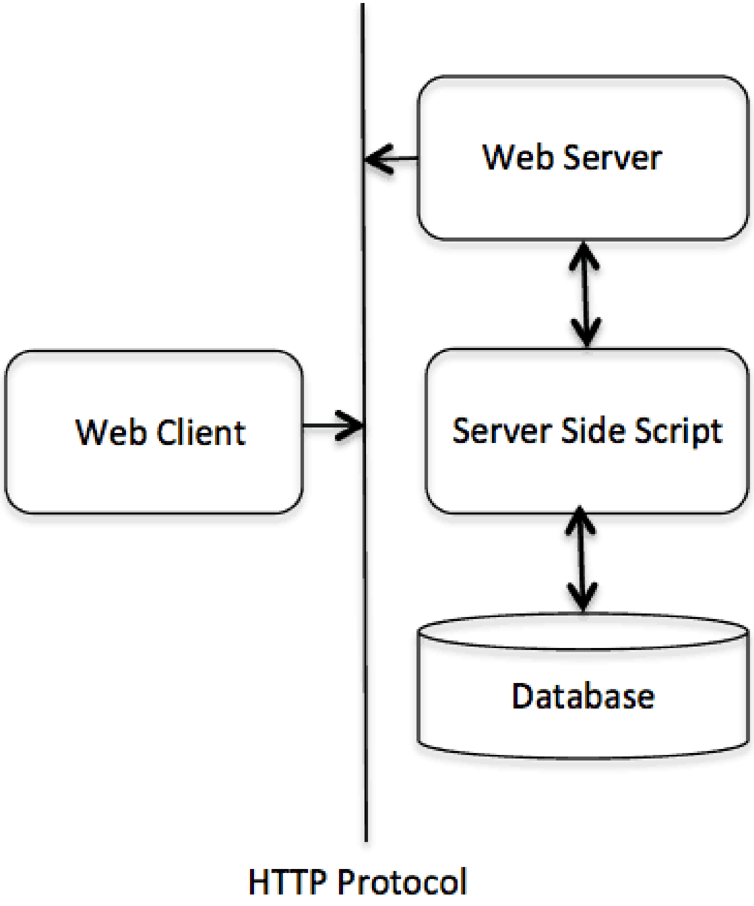
OMPython

- OMPython – OpenModelica Python API is a free open source library
- Interactive session handling, parsing, interpretation of commands and Modelica expressions for evaluation, simulation, plotting, etc.
- Interface to all OpenModelica API calls
- Easy access to the library and testing of OpenModelica commands
- Not a standalone package, depends on OpenModelica Compiler

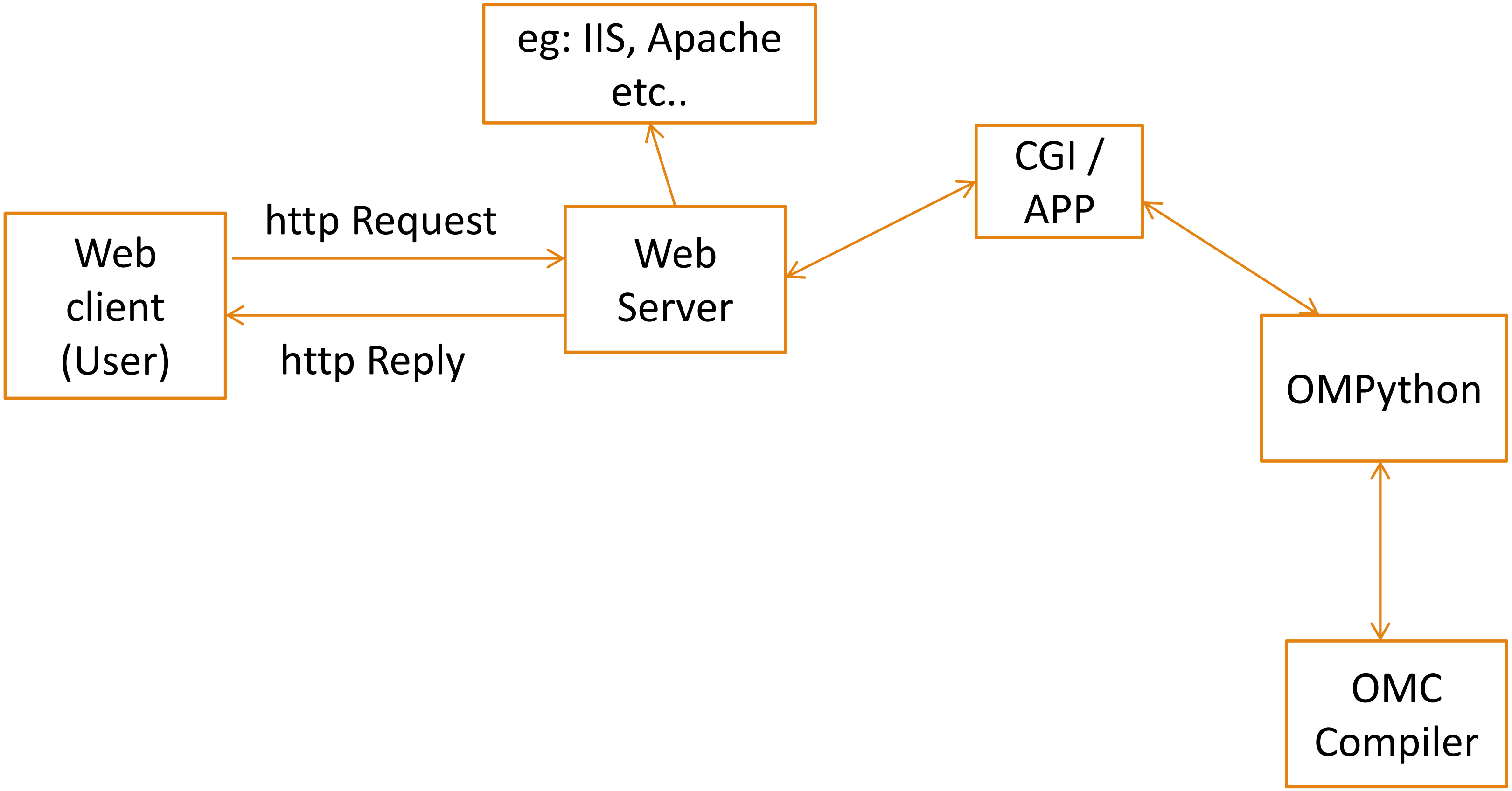
Python-CGI

- The Common Gateway Interface, (CGI)
- Allows webservers to launch external application that creates pages dynamically
- CGI scripts supports many programming languages.

eg. Python Script, PERL Script, Shell Script, C or C++ program, etc.



Client Server Communication



Web client (User)

http Request

http Reply

Web Server

eg: IIS, Apache etc..

CGI / APP

OMPython

OMC Compiler

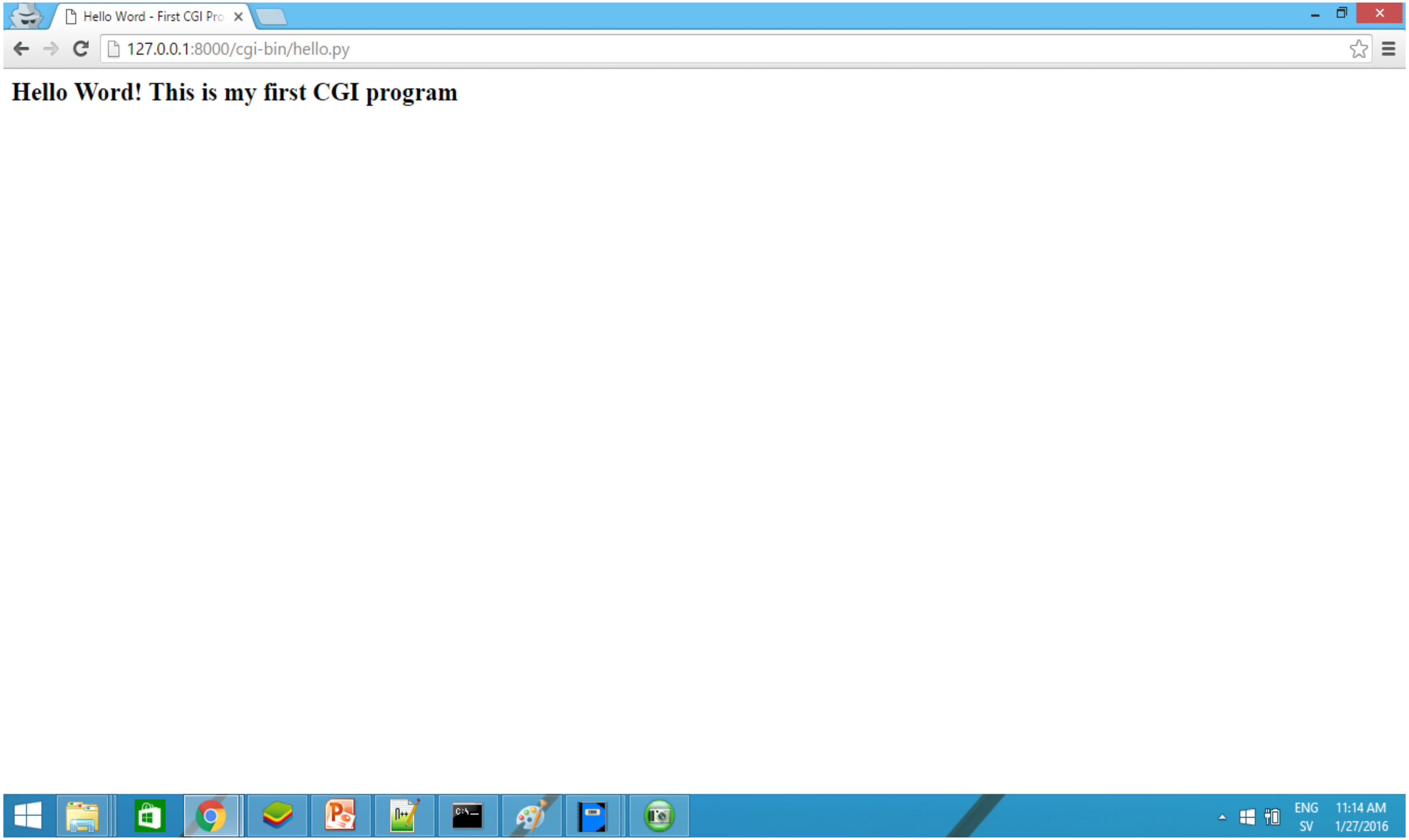
CGI Application Configurations

- Make sure that your Web Server supports CGI and is configured to handle CGI Programs.
- All the CGI Programs to be executed by the HTTP server are kept in a pre-configured directory mostly “cgi-bin”
- Setup the access permission for cgi-bin directory and cgi programs inside the directory
- By convention, CGI files have extension as.**cgi,** but you can keep your files with different extension depending on the languages you use, in this case we use python extension **.py** .

Simple CGI program

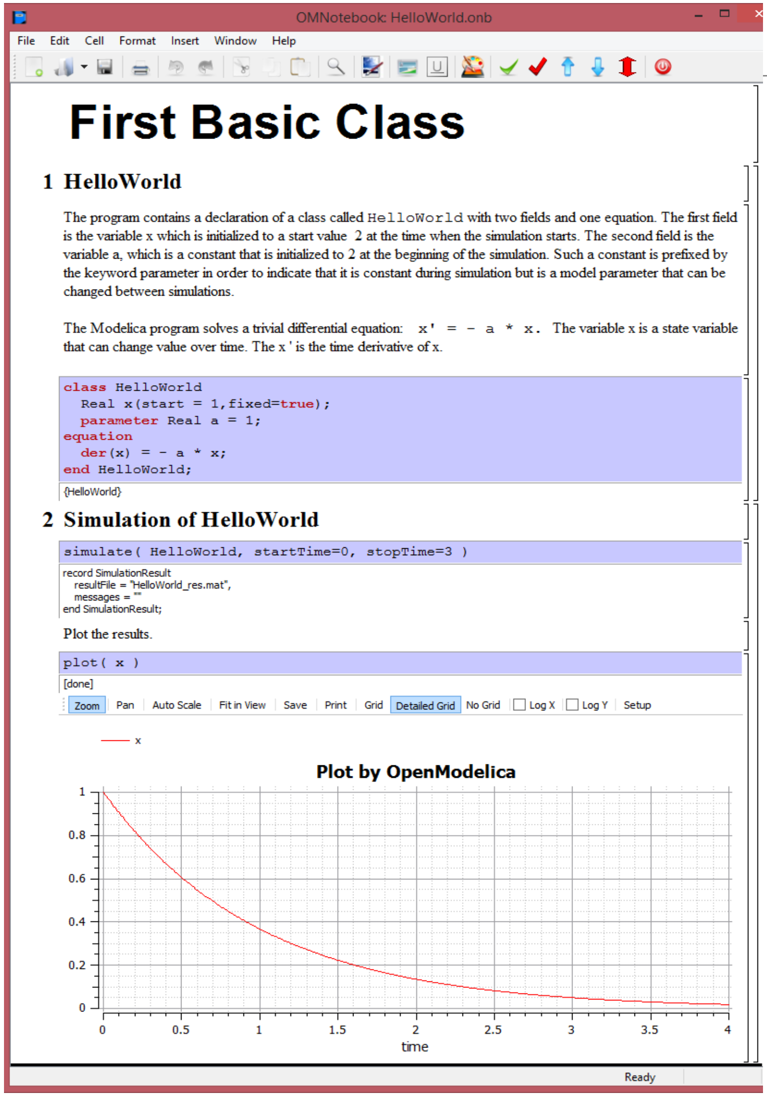
```
#!/usr/bin/python
print "Content-type:text/html\r\n\r\n"
print '<html>'
print '<head>'
print '<title>Hello Word - First CGI Program</title>'
print '</head>'
print '<body>'
print '<h2>Hello Word! This is my first CGI program</h2>'
print '</body>'
print '</html>'
```

Output in Browser



Hello Word! This is my first CGI program

Translated Notebook to Webbook



First Basic Class

1 HelloWorld

The program contains a declaration of a class called `HelloWorld` with two fields and one equation. The first field is the variable x which is initialized to a start value 2 at the time when the simulation starts. The second field is the variable a, which is a constant that is initialized to 2 at the beginning of the simulation. Such a constant is prefixed by the keyword parameter in order to indicate that it is constant during simulation but is a model parameter that can be changed between simulations.

The Modelica program solves a trivial differential equation: `x' = - a * x.` The variable x is a state variable that can change value over time. The x ' is the time derivative of x.

```
class HelloWorld
  Real x(start = 1,fixed=true);
  parameter Real a = 1;
equation
  der(x) = - a * x;
end HelloWorld;
```

```
{HelloWorld}
```

2 Simulation of HelloWorld

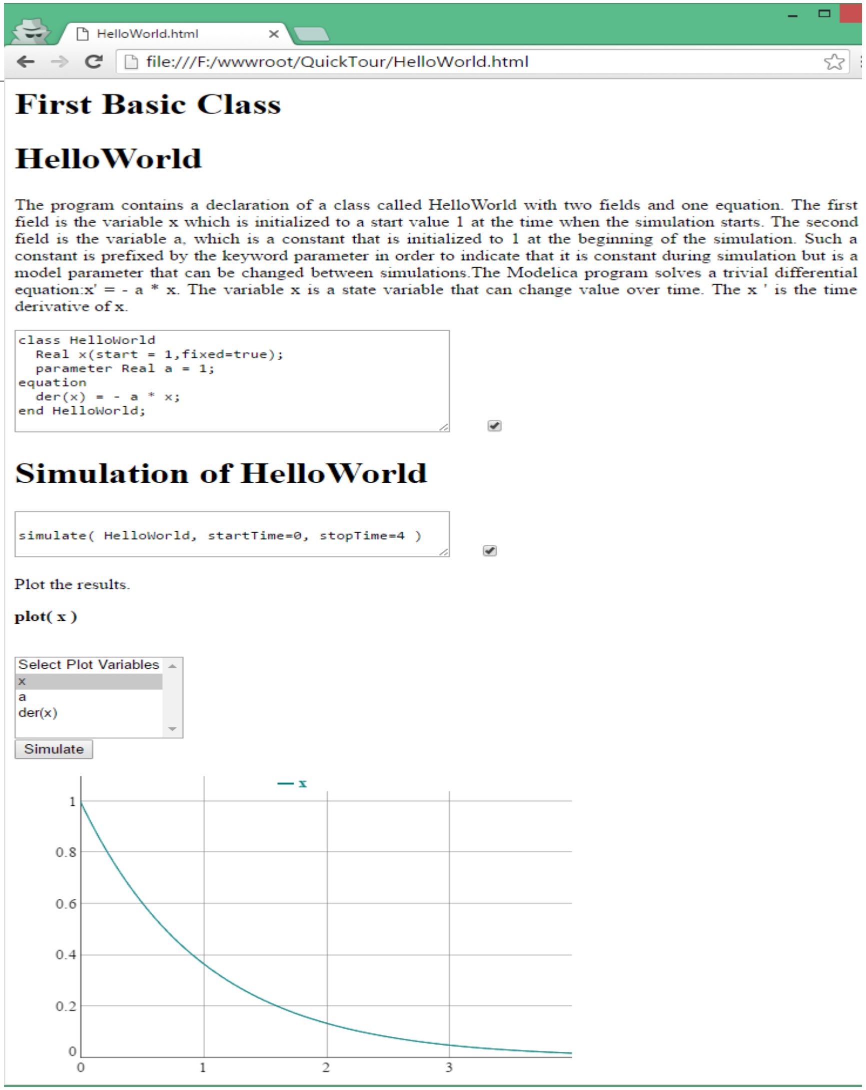
```
simulate( HelloWorld, startTime=0, stopTime=3 )
```

```
record SimulationResult
   resultFile = "HelloWorld_res.mat",
   messages = ""
end SimulationResult;
```

Plot the results.

```
plot( x )
```

```
[done]
```



First Basic Class

HelloWorld

The program contains a declaration of a class called HelloWorld with two fields and one equation. The first field is the variable x which is initialized to a start value 1 at the time when the simulation starts. The second field is the variable a, which is a constant that is initialized to 1 at the beginning of the simulation. Such a constant is prefixed by the keyword parameter in order to indicate that it is constant during simulation but is a model parameter that can be changed between simulations.The Modelica program solves a trivial differential equation:x' = - a * x. The variable x is a state variable that can change value over time. The x ' is the time derivative of x.

```
class HelloWorld
  Real x(start = 1,fixed=true);
  parameter Real a = 1;
equation
  der(x) = - a * x;
end HelloWorld;
```

☑

Simulation of HelloWorld

```
simulate( HelloWorld, startTime=0, stopTime=4 )
```

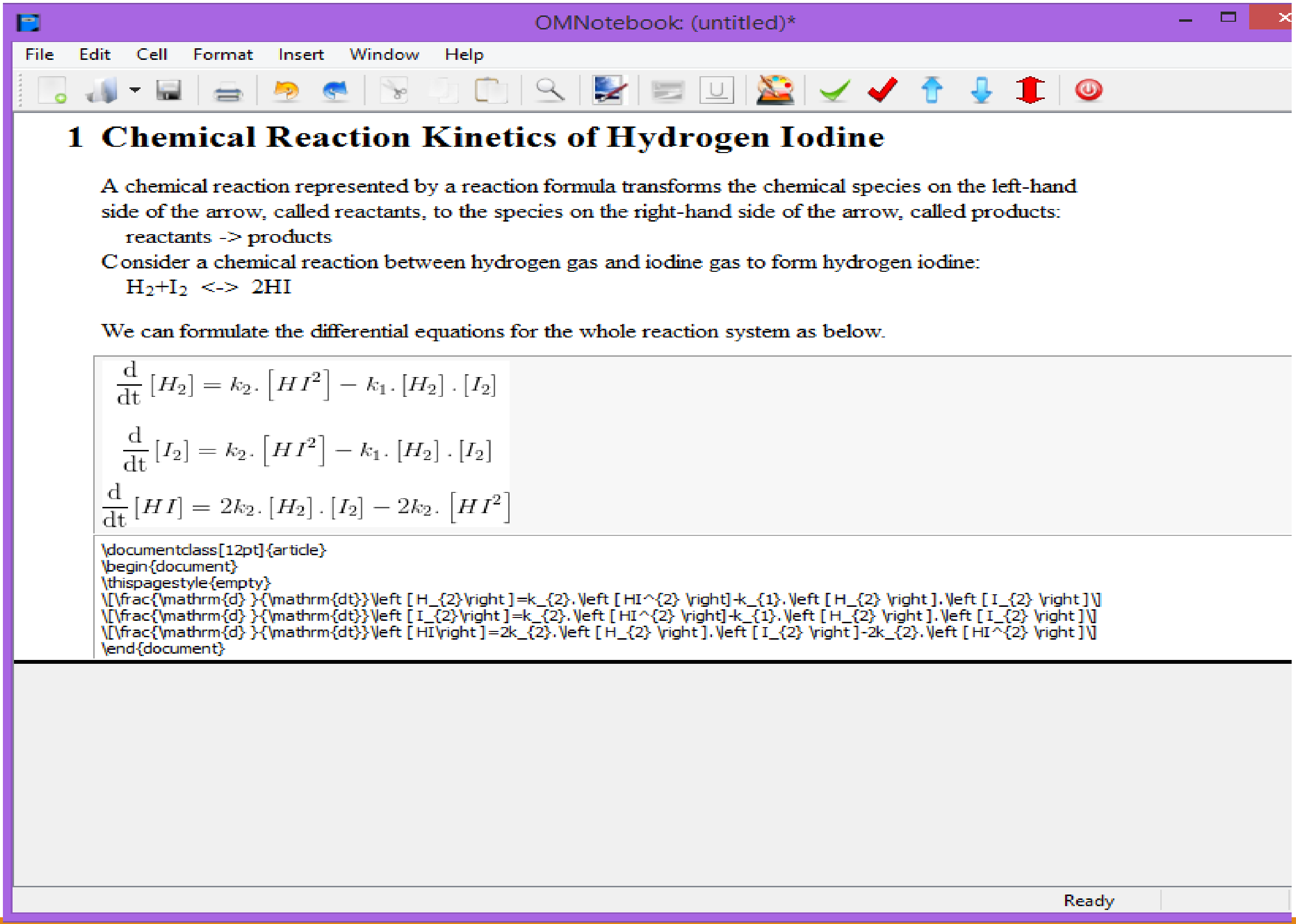
☑

Plot the results.

plot(x)

Mathematical Typesetting

- OMNotebook supports Latex formatting evaluation for mathematics



1 Chemical Reaction Kinetics of Hydrogen Iodine

A chemical reaction represented by a reaction formula transforms the chemical species on the left-hand side of the arrow, called reactants, to the species on the right-hand side of the arrow, called products:

reactants -> products

Consider a chemical reaction between hydrogen gas and iodine gas to form hydrogen iodine:

H_2+I_2 <-> 2HI

We can formulate the differential equations for the whole reaction system as below.

$$\frac{\mathrm{d}}{\mathrm{dt}}\left[H_{2}\right]=k_{2}.\left[HI^{2}\right]-k_{1}.\left[H_{2}\right].\left[I_{2}\right]$$

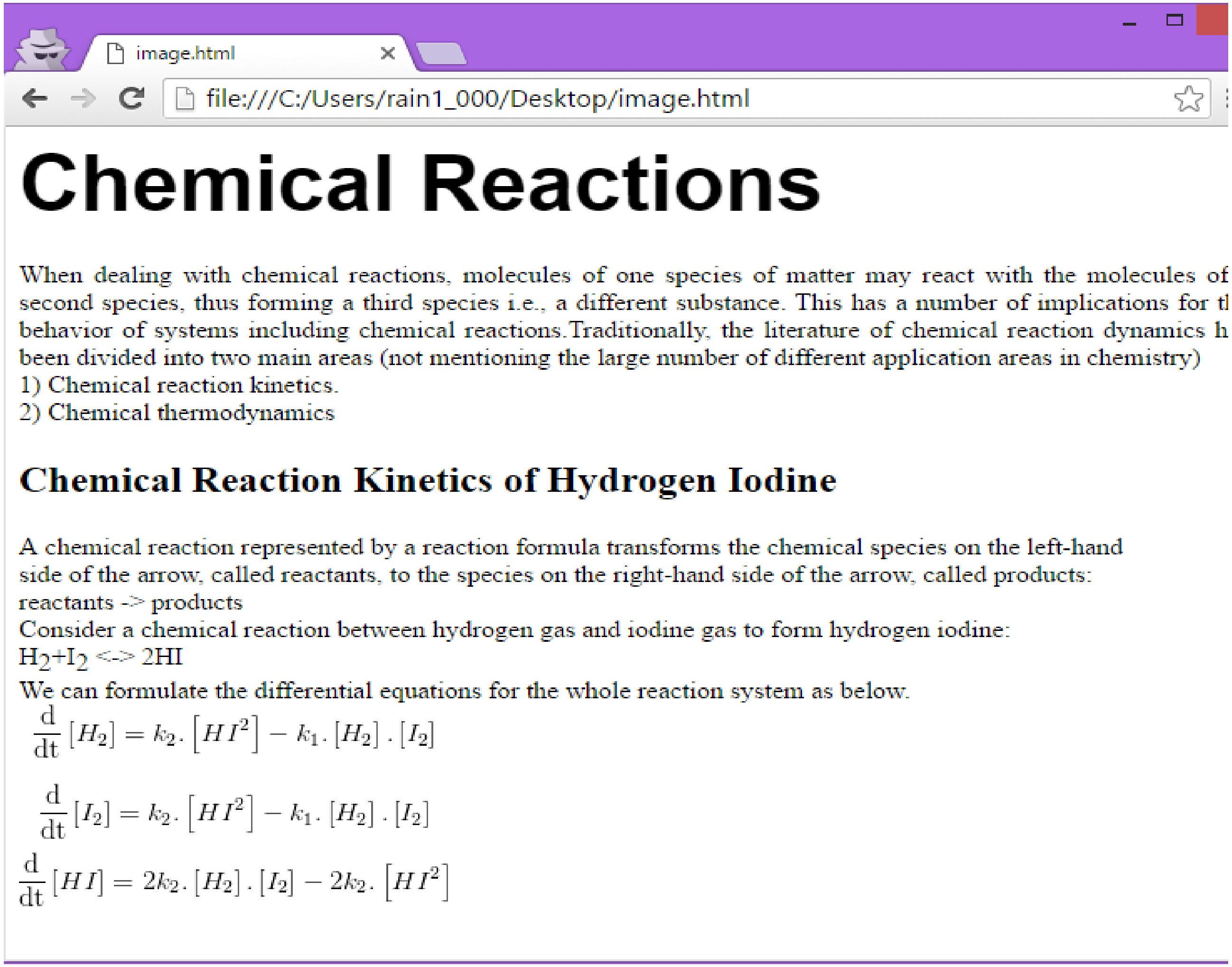
$$\frac{\mathrm{d}}{\mathrm{dt}}\left[I_{2}\right]=k_{2}.\left[HI^{2}\right]-k_{1}.\left[H_{2}\right].\left[I_{2}\right]$$

$$\frac{\mathrm{d}}{\mathrm{dt}}\left[HI\right]=2k_{2}.\left[H_{2}\right].\left[I_{2}\right]-2k_{2}.\left[HI^{2}\right]$$

```
\documentclass[12pt]{article}
\begin{document}
\thispagestyle{empty}
\[\frac{\mathrm{d} }{\mathrm{dt}}\left [ H_{2}\right ]=k_{2}.\left [ HI^{2} \right]-k_{1}.\left [ H_{2} \right ].\left [ I_{2} \right ]\]
\[\frac{\mathrm{d} }{\mathrm{dt}}\left [ I_{2}\right ]=k_{2}.\left [ HI^{2} \right]-k_{1}.\left [ H_{2} \right ].\left [ I_{2} \right ]\]
\[\frac{\mathrm{d} }{\mathrm{dt}}\left [ HI\right ]=2k_{2}.\left [ H_{2} \right ].\left [ I_{2} \right ]-2k_{2}.\left [ HI^{2} \right ]\]
\end{document}
```

Latex instructions can be hidden by double clicking the Cell in tree view

Mathematical Typesetting in Web Browser



Chemical Reactions

When dealing with chemical reactions, molecules of one species of matter may react with the molecules of second species, thus forming a third species i.e., a different substance. This has a number of implications for t behavior of systems including chemical reactions.Traditionally, the literature of chemical reaction dynamics h been divided into two main areas (not mentioning the large number of different application areas in chemistry)
1) Chemical reaction kinetics.
2) Chemical thermodynamics

Chemical Reaction Kinetics of Hydrogen Iodine

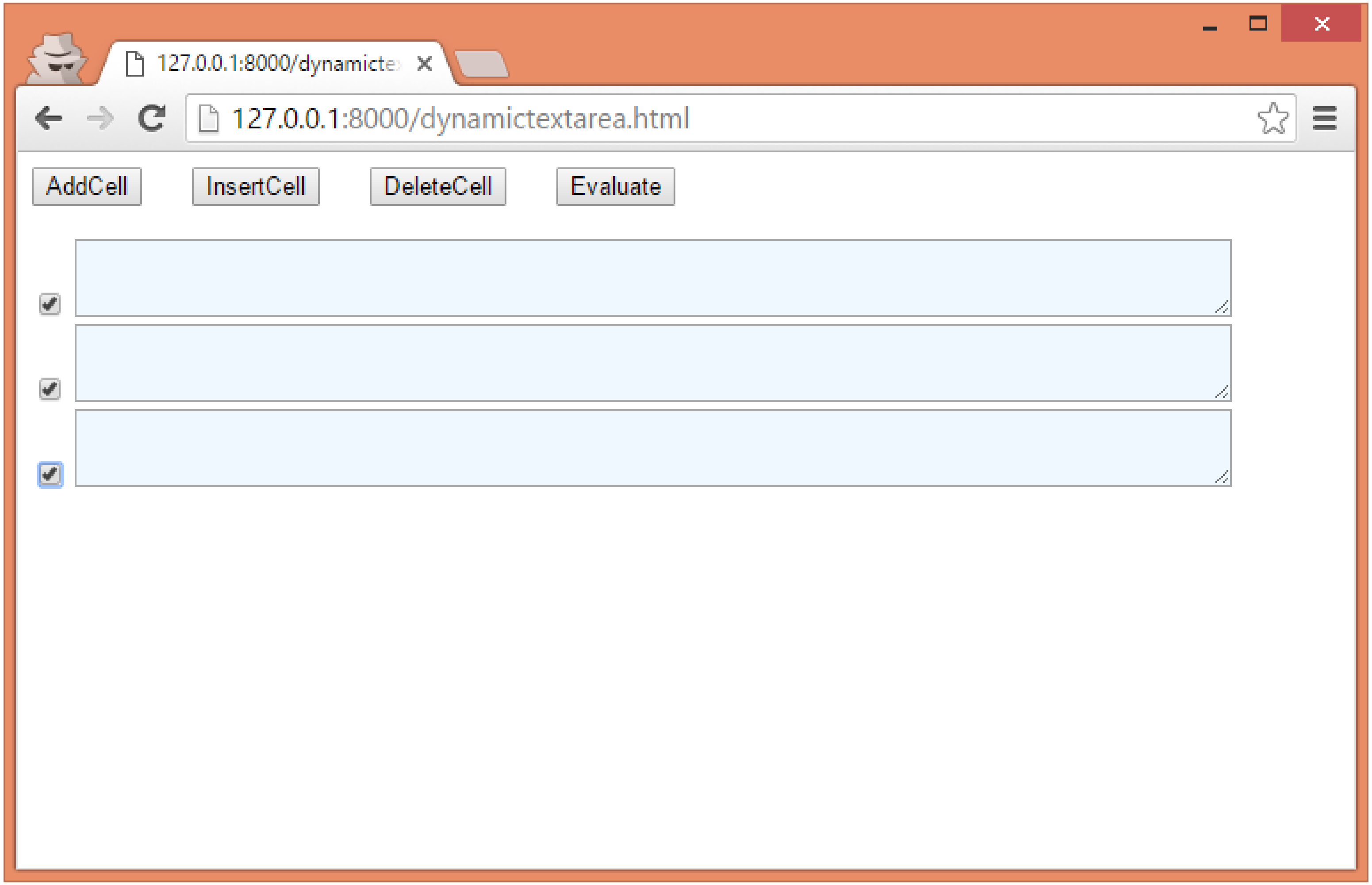
A chemical reaction represented by a reaction formula transforms the chemical species on the left-hand side of the arrow, called reactants, to the species on the right-hand side of the arrow, called products:
reactants -> products
Consider a chemical reaction between hydrogen gas and iodine gas to form hydrogen iodine:
H_2+I_2 <-> 2HI
We can formulate the differential equations for the whole reaction system as below.

$$\frac{\mathrm{d}}{\mathrm{dt}}[H_2] = k_2 . \left[HI^2\right] - k_1 . [H_2] . [I_2]$$

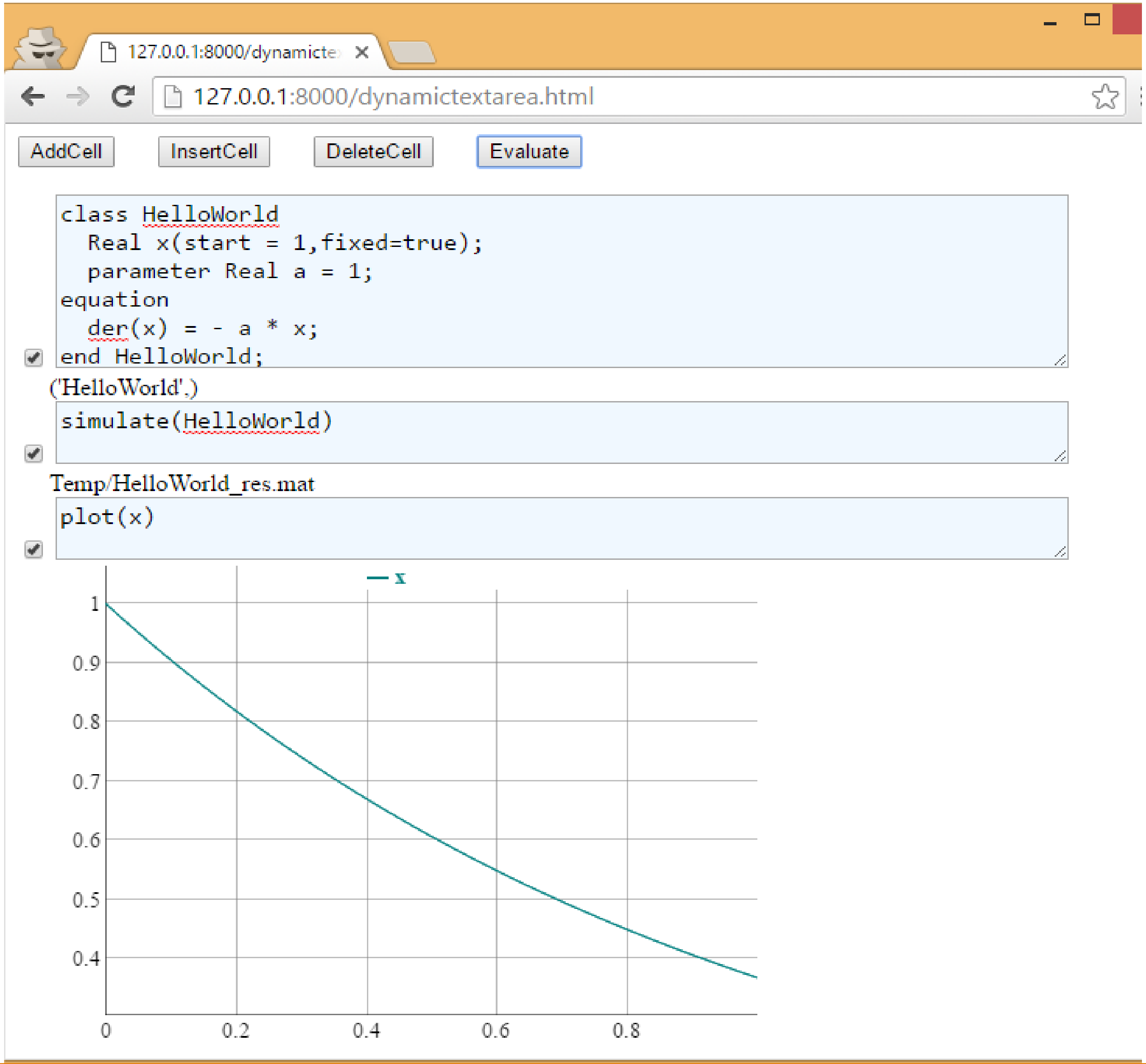
$$\frac{\mathrm{d}}{\mathrm{dt}}[I_2] = k_2 . \left[HI^2\right] - k_1 . [H_2] . [I_2]$$

$$\frac{\mathrm{d}}{\mathrm{dt}}[HI] = 2k_2 . [H_2] . [I_2] - 2k_2 . \left[HI^2\right]$$

Dynamic Editable Webbook



Interactive Simulations and Plots



☑
```
class HelloWorld
  Real x(start = 1,fixed=true);
  parameter Real a = 1;
equation
  der(x) = - a * x;
end HelloWorld;
```

('HelloWorld',)

☑
```
simulate(HelloWorld)
```

Temp/HelloWorld_res.mat

☑
```
plot(x)
```

Future Work

- Set up a Sandbox-Server and make browser simulation of Modelica models tool independent and secure.
- Improve the client (user) web book features

Demo

Thank you

Questions ?