E-CELL2 User's Manual

Chapter 1 Setup Instructions

1 Introduction

E-CELL1 is a cell simulation software designed for Linux developed at the Tomita Laboratory of Keio University. E-CELL2 was designed to emulate every aspect of E-CELL1 in multiple platforms (MS-Windows, Solaris, Linux). This is an opensource project; as of now, MS-Windows (2000/NT/98/XP) version is under development.

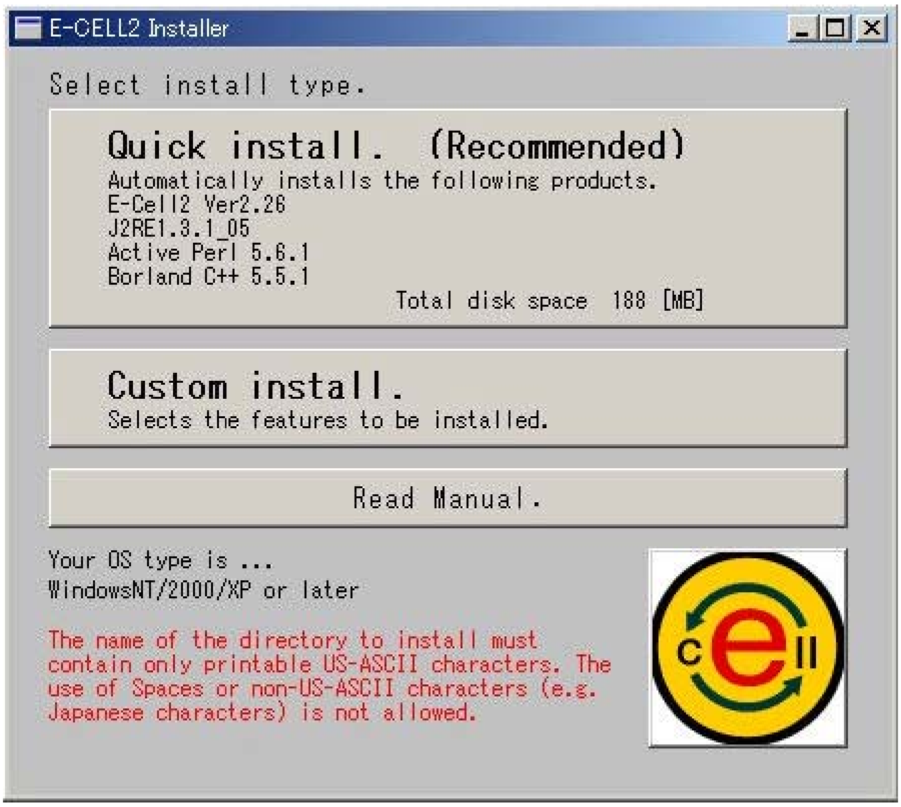
The simulation environment of E-CELL2 is a mixture of softwares including the following:

- Active Perl --- Perl Interpreter from ActiveState
  Required to create Rules and Reactors for E-CELL2.26. Modeling tools included in the E-CELL2 package require this software.
- Borland C++ Compiler -- Free C/C++ compiler from Borland
  Required to create Reactors for E-CELL2.26. This is not required for running E-CELL2 and performing simulations.
- Java(TM)2 Runtime Environment 1.3
  Required to use the GUI environment and to run E-CELL2.26. User are not required to install the software if JRE 1.3 is already installed in their computer. Advanced Users: install JDK 1.3 or higher to customize the E-CELL2 system.

2 Quick Install (CD-ROM version)

Performing Quick Install installs the above mentioned third-party software with the E-CELL2.26 software and tools. Since explanation here is aimed at the direction installed from CD, those who downloaded the installer of E-CELL2.26 need to refer to quick installation and custom installation of Section 4 or 5.

Let us now start with the installation procedures.



Select install type.

Quick install. (Recommended)
Automatically installs the following products.
E-Cell2 Ver2.26
J2RE1.3.1_05
Active Perl 5.6.1
Borland C++ 5.5.1
Total disk space 188 [MB]

Custom install.
Selects the features to be installed.

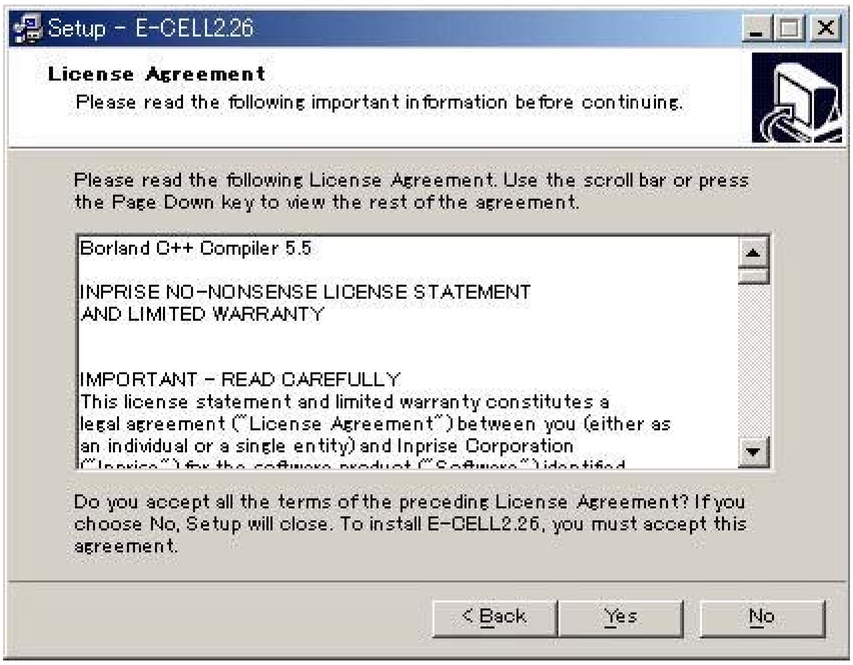
Read Manual.

Your OS type is ...
WindowsNT/2000/XP or later

The name of the directory to install must contain only printable US-ASCII characters. The use of Spaces or non-US-ASCII characters (e.g. Japanese characters) is not allowed.

Figure 1. E-CELL2.26 Installer

Select "Quick install. (Recommended)".



License Agreement
Please read the following important information before continuing.

Please read the following License Agreement. Use the scroll bar or press the Page Down key to view the rest of the agreement.

Borland C++ Compiler 5.5

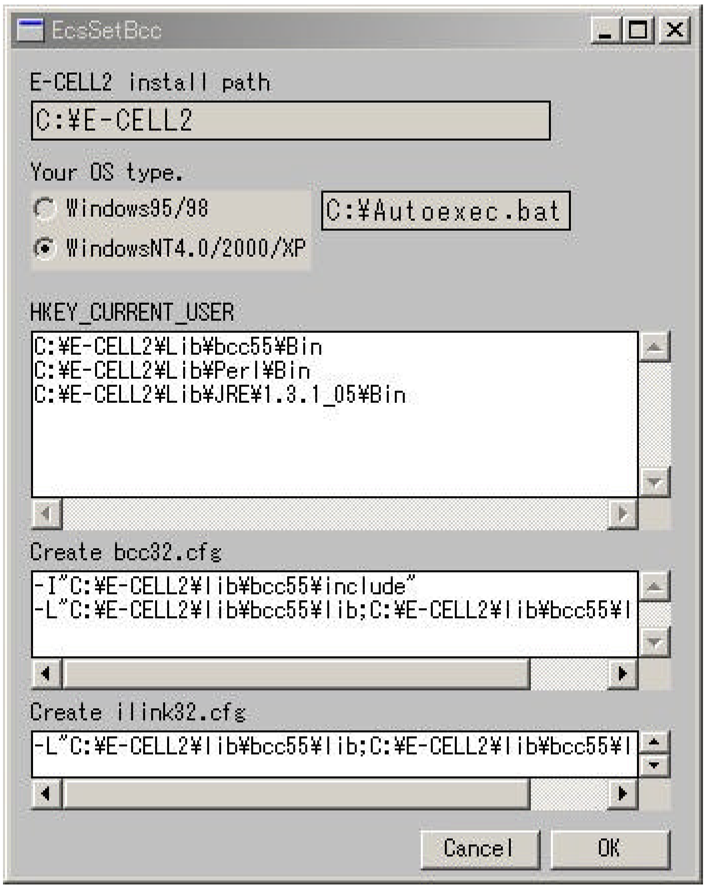
INPRISE NO-NONSENSE LICENSE STATEMENT
AND LIMITED WARRANTY

IMPORTANT - READ CAREFULLY
This license statement and limited warranty constitutes a
legal agreement ("License Agreement") between you (either as
an individual or a single entity) and Inprise Corporation
("Inprise") for the software product ("Software") identified

Do you accept all the terms of the preceding License Agreement? If you choose No, Setup will close. To install E-CELL2.26, you must accept this agreement.

Figure 2: E-CELL2.26 Installer

License Agreement for the third party software will be displayed. If you agree with the license, please click "Yes" and select the installation directory to go on with the install.



E-CELL2 install path

C:¥E-CELL2

Your OS type.

○ Windows95/98

● WindowsNT4.0/2000/XP

C:¥Autoexec.bat

HKEY_CURRENT_USER

```
C:¥E-CELL2¥Lib¥bcc55¥Bin
C:¥E-CELL2¥Lib¥Perl¥Bin
C:¥E-CELL2¥Lib¥JRE¥1.3.1_05¥Bin
```

Create bcc32.cfg

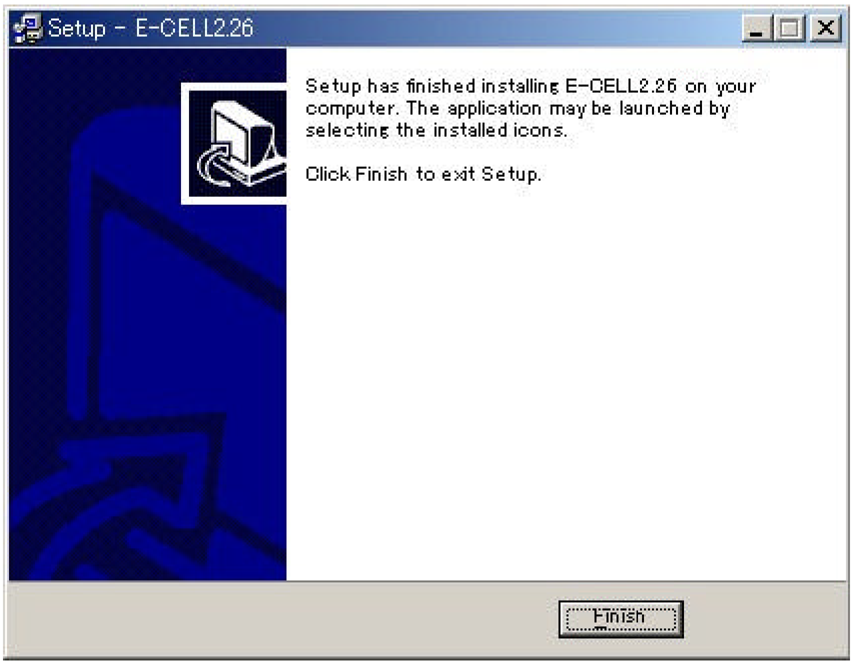
```
-I"C:¥E-CELL2¥lib¥bcc55¥include"
-L"C:¥E-CELL2¥lib¥bcc55¥lib;C:¥E-CELL2¥lib¥bcc55¥l
```

Create ilink32.cfg

```
-L"C:¥E-CELL2¥lib¥bcc55¥lib;C:¥E-CELL2¥lib¥bcc55¥l
```

Figure 3: E-CELL2.26 Installer

Set the path information in the HKEY_CURRENT_USER registry for Borland C++ Compiler. If the information is correct, click "OK" to proceed.

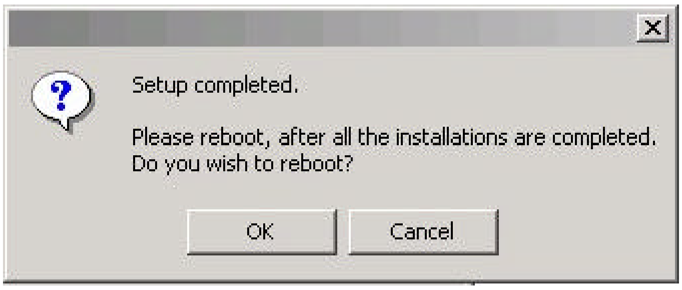


Setup has finished installing E-CELL2.26 on your computer. The application may be launched by selecting the installed icons.

Click Finish to exit Setup.

Figure 4: E-CELL2.26 Installer

Click "Finish".



Setup completed.

Please reboot, after all the installations are completed.
Do you wish to reboot?

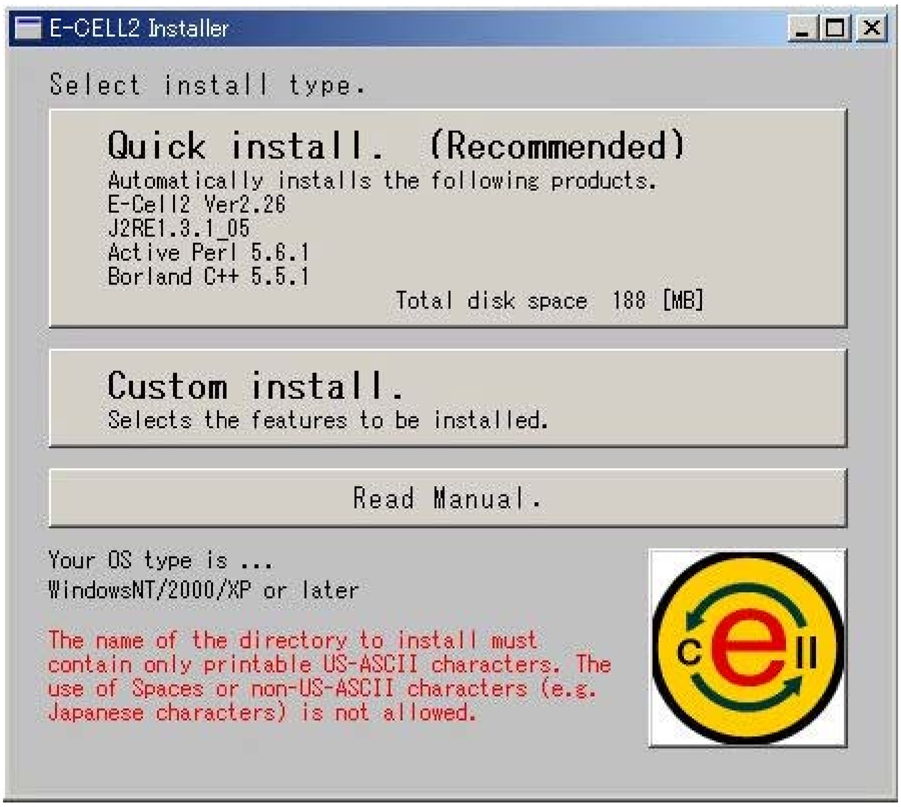
Figure 5: Restart Windows

Please restart Windows to reflect the settings.

This is the end of the installation procedure. For basic operations, please refer to Chapter 2.

3 Custom Install (CD-ROM version)

Custom Install is for users with some of the software above already installed. The E-CELL2.26 package to be installed will have the same function as installed by quick install.



Select install type.

Quick install. (Recommended)
Automatically installs the following products.
E-Cell2 Ver2.26
J2RE1.3.1_05
Active Perl 5.6.1
Borland C++ 5.5.1
Total disk space 188 [MB]

Custom install.
Selects the features to be installed.

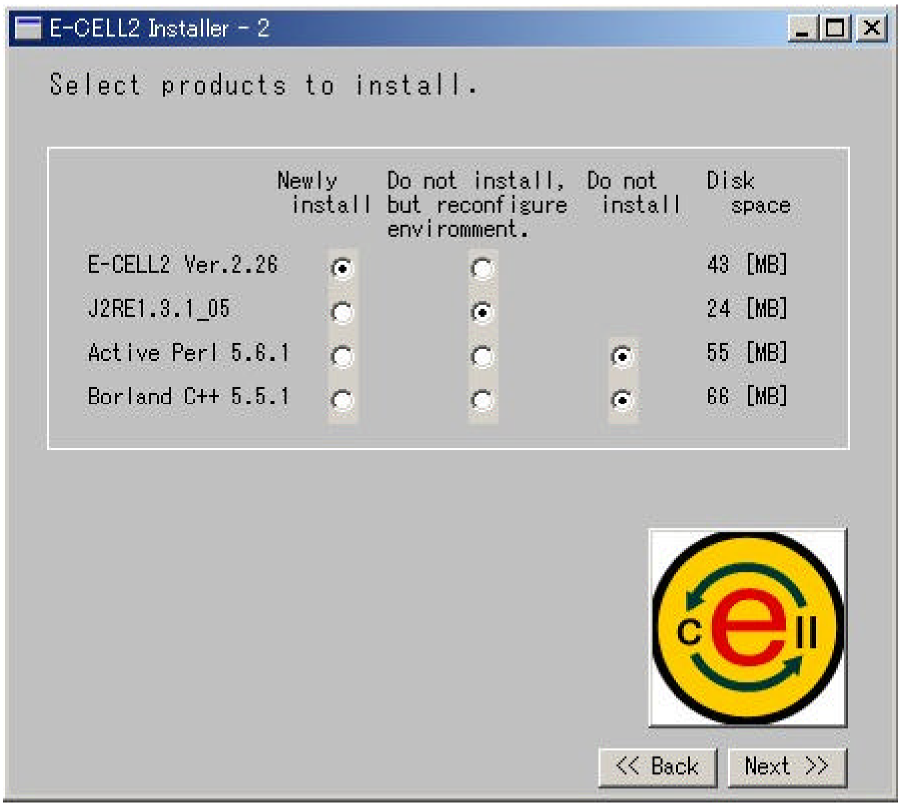
Read Manual.

Your OS type is ...
WindowsNT/2000/XP or later

The name of the directory to install must contain only printable US-ASCII characters. The use of Spaces or non-US-ASCII characters (e.g. Japanese characters) is not allowed.

Figure 6: E-CELL2.26 Installer

Click on "Custom Install". Select the third party software and the packages to install from below.



E-CELL2 Installer - 2

Select products to install.

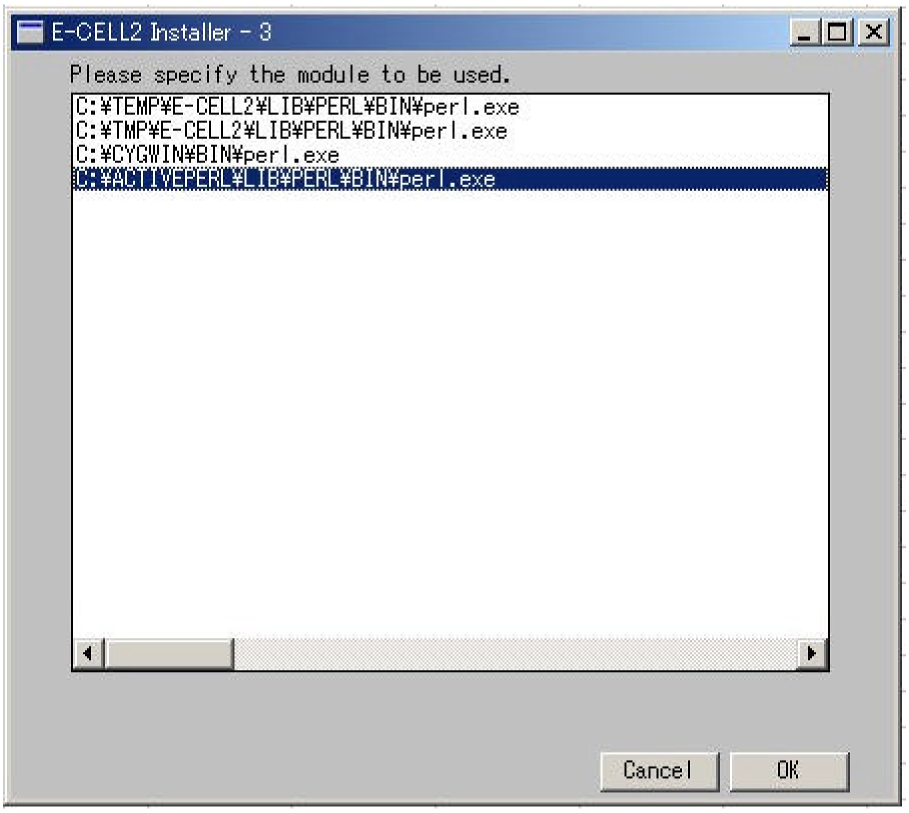
	Newly install	Do not install, but reconfigure environment.	Do not install	Disk space
E-CELL2 Ver.2.26	●	○		43 [MB]
J2RE1.3.1_05	○	●		24 [MB]
Active Perl 5.6.1	○	○	●	55 [MB]
Borland C++ 5.5.1	○	○	●	66 [MB]

Figure 7: E-CELL2.26 Installer

Select from below:

- Newly install
- Do not install, but reconfigure environment.
- Do not install

Click "Next" to go on. For "Do not install, but reconfigure environment.", please follow the instructions of the software.

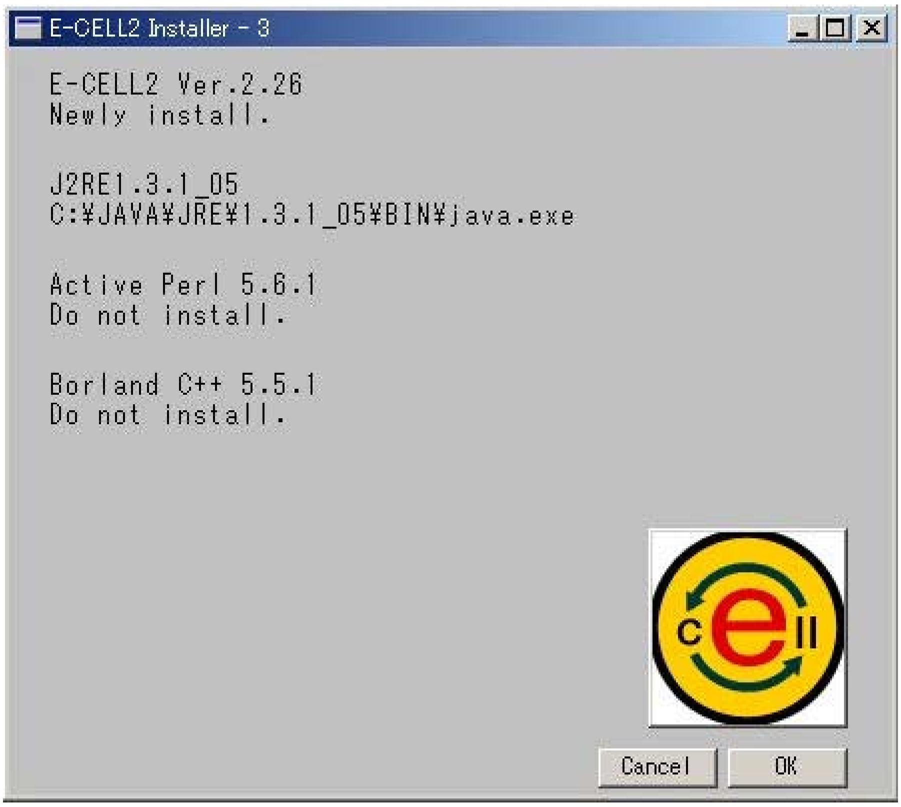


Please specify the module to be used.

```
C:¥TEMP¥E-CELL2¥LIB¥PERL¥BIN¥perl.exe
C:¥TMP¥E-CELL2¥LIB¥PERL¥BIN¥perl.exe
C:¥CYGWIN¥BIN¥perl.exe
C:¥ACTIVEPERL¥LIB¥PERL¥BIN¥perl.exe
```

Figure 8: E-CELL2.26 Installer

The following message will be displayed to check the list:

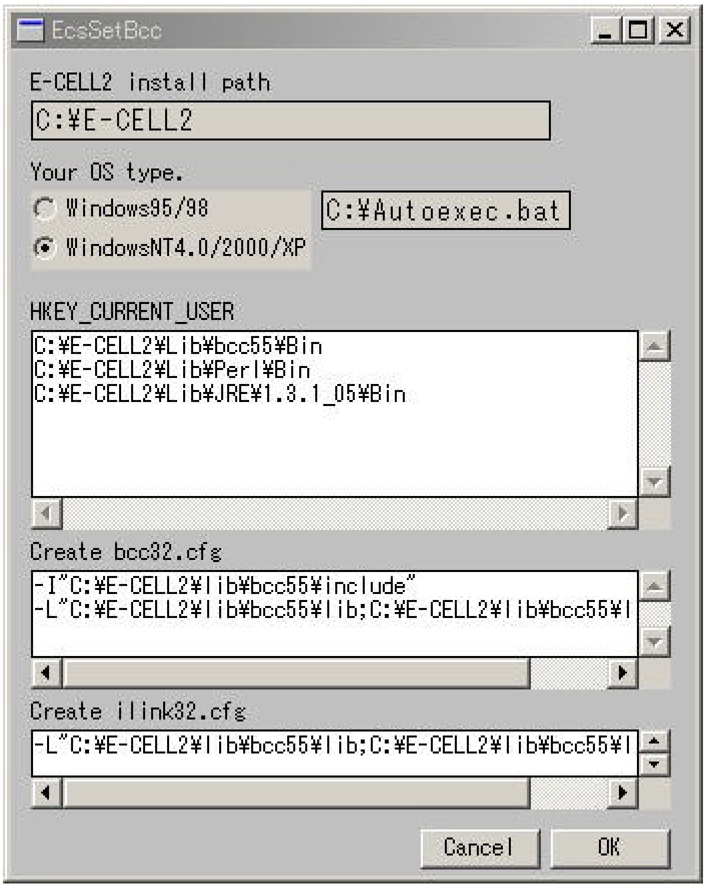


```
E-CELL2 Ver.2.26
Newly install.

J2RE1.3.1_05
C:¥JAVA¥JRE¥1.3.1_05¥BIN¥java.exe

Active Perl 5.6.1
Do not install.

Borland C++ 5.5.1
Do not install.
```

Figure 9: E-CELL2.26 Installer

If the above information is correct, click "OK" to proceed. Any new software will be installed at this point.

License Agreement for the third party software will be displayed for each software to newly install. If you agree with the license, please click "Yes" and select the installation directory to go on with the install.



E-CELL2 install path

C:¥E-CELL2

Your OS type.

○ Windows95/98

● WindowsNT4.0/2000/XP

C:¥Autoexec.bat

HKEY_CURRENT_USER

```
C:¥E-CELL2¥Lib¥bcc55¥Bin
C:¥E-CELL2¥Lib¥Perl¥Bin
C:¥E-CELL2¥Lib¥JRE¥1.3.1_05¥Bin
```

Create bcc32.cfg

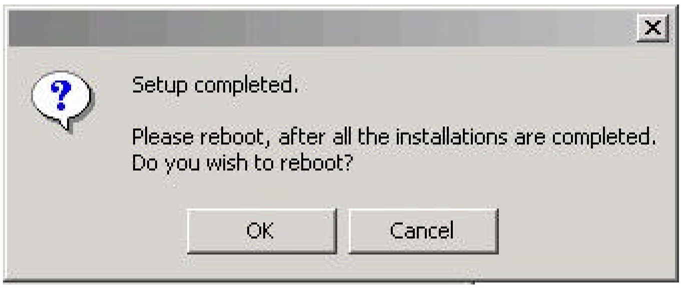
```
-I"C:¥E-CELL2¥lib¥bcc55¥include"
-L"C:¥E-CELL2¥lib¥bcc55¥lib;C:¥E-CELL2¥lib¥bcc55¥l
```

Create ilink32.cfg

```
-L"C:¥E-CELL2¥lib¥bcc55¥lib;C:¥E-CELL2¥lib¥bcc55¥l
```

Figure 10: E-CELL2.26 Installer

While installing BCC, the following will be displayed. In Windows NT/2000, perform path settings in HKEY_CURRENT_USER; in Windows 98, perform path settings in Autoexec.bat. Also creates bcc32.cfg and ilink32.cfg for BCC(editable). Click "Next" to proceed.



Setup completed.

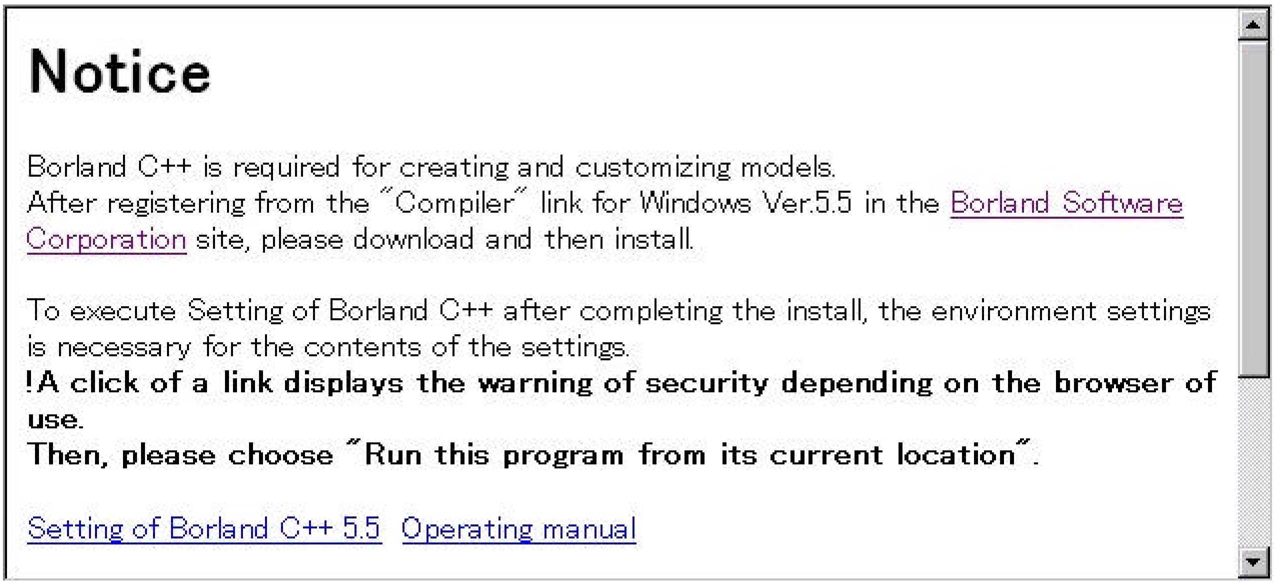
Please reboot, after all the installations are completed.
Do you wish to reboot?

Figure 11: Windows

Click "OK" and restart Windows to reflect the above settings. This is the end of the installation procedure. For basic operations, please refer to Chapter 2.

4 Quick Install (Network version)

Please download the E-CELL2 Ver2.26 Quick Install module(ecell226web-all-in-one.exe). Execute the module, and please install according to the directions of the wizard. If install is complete, the install guidance screen for Borland C++ will be displayed. Borland C++ is required for creating Reactors, so therefore, please download and install following the instructions.



Notice

Borland C++ is required for creating and customizing models.
After registering from the "Compiler" link for Windows Ver.5.5 in the Borland Software Corporation site, please download and then install.

To execute Setting of Borland C++ after completing the install, the environment settings is necessary for the contents of the settings.
!A click of a link displays the warning of security depending on the browser of use.
Then, please choose "Run this program from its current location".

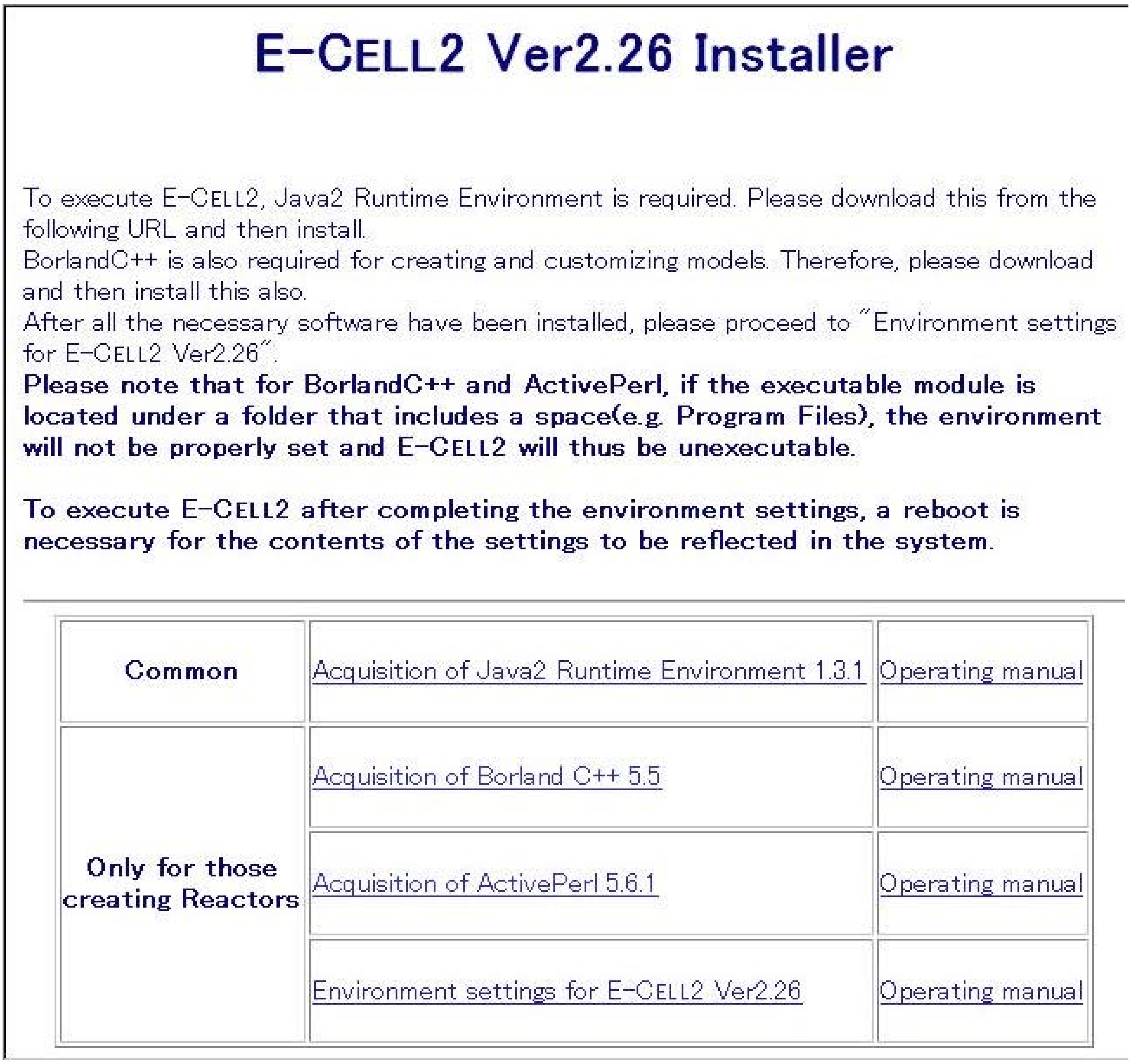
Setting of Borland C++ 5.5 Operating manual

Figure 12: Guidance window of the setup

This is the end of the installation procedure. For basic operations, please refer to Chapter 2.

5 Custom Install (Network version)

Please download the E-CELL2 Ver2.26 Custom Install module(ecell226web-minimal-set.exe). Execute the module, and please install according to directions of a wizard. If install is complete, install guidance for the third-party software and E-CELL2.26 will be displayed. Therefore, please install according to the guidance.



E-CELL2 Ver2.26 Installer

To execute E-CELL2, Java2 Runtime Environment is required. Please download this from the following URL and then install.
BorlandC++ is also required for creating and customizing models. Therefore, please download and then install this also.
After all the necessary software have been installed, please proceed to "Environment settings for E-CELL2 Ver2.26".
Please note that for BorlandC++ and ActivePerl, if the executable module is located under a folder that includes a space(e.g. Program Files), the environment will not be properly set and E-CELL2 will thus be unexecutable.

To execute E-CELL2 after completing the environment settings, a reboot is necessary for the contents of the settings to be reflected in the system.

Common	Acquisition of Java2 Runtime Environment 1.3.1	Operating manual
Only for those creating Reactors	Acquisition of Borland C++ 5.5	Operating manual
	Acquisition of ActivePerl 5.6.1	Operating manual
	Environment settings for E-CELL2 Ver2.26	Operating manual

Figure 13: Guidance window of the setup

This is the end of the installation procedure. For basic operations, please refer to Chapter 2.

Chapter 2 Tutorial

1 Features of E-CELL2

In this chapter, the features and operation methods of E-CELL2.26 will be outlined. For details regarding the theoretical background of E-CELL, please refer to the tutorial available at E-CELL.Org(http://www.e-cell.org).

To model a cell, the abstraction of its components is necessary. In E-CELL, the "**Substance-Reactor** Model" has been adopted for describing the structures of a cell and the chemical reactions inside a cell. This model has the following features.

- The state of a cell is regarded as a set of "quantity of substances (**Substance**)".
- The activity of a cell is represented by a "change in the quantity of a **Substance**", using a chemical reaction equation (**Reactor**).
- Nodes of **Substance** connected by a **Reactor** constitute a directed graph.

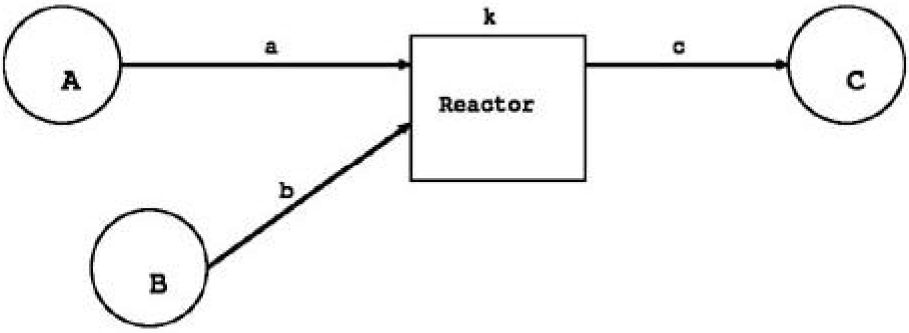


Figure 1. Substance-Reactor Model

- Cellular organelles are described via the location (**System**) of substances.

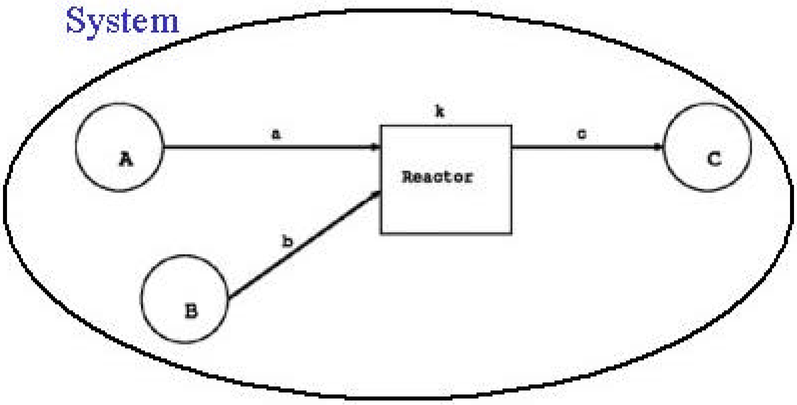


Figure 2. Substance-Reactor Model inside a System

Please note that the concept of a substance (**Substance**) includes not only chemical molecules, ions, and radicals, but also voluntary physical properties such as osmotic pressure and cell volume. Negative values however cannot be described.

In E-CELL, it is assumed that the intracellular concentration of a substance is uniform throughout the cell. To handle diffusion, it is therefore necessary to prepare multiple systems each with varying substance quantities.

To simulate various intracellular phenomena in E-CELL, both rate(flux)-based reactions such as enzymatic reactions, and rapid reactions such as chemical equilibrium can be implemented.

Flux-based (i.e. can be solved as differential equations) chemical reactions are handled inside E-CELL as follows.

- Changes the rate of chemical reaction equations using a standard (Regular) **Reactor**.
- Solves the ordinary differential equations using the Runge-Kutta method.

- Michaelis-Menten reactions are examples of rate equations.

Chemical equibrium-based (i.e. can be solved as algebraic equations) chemical reactions are handled as follows.

- Manipulates the quantity of **Substance**s directly using a back door (Postern) **Reactor**.
- Phenomena such as rapid equilibrium and changes in osmotic pressure are examples of this type of reaction.

From the standpoint of software engineering, E-CELL2 has the following features. The part written in C++ will be described first. This part, the so-called heart of E-Cell2, corresponds to the view of the world (Ontology) that shows how E-CELL2 describes a cell. Such components are called "classes" in the field of object-oriented modeling. The computation mechanisms necessary for simulation are implemented by the interactions between the classes.

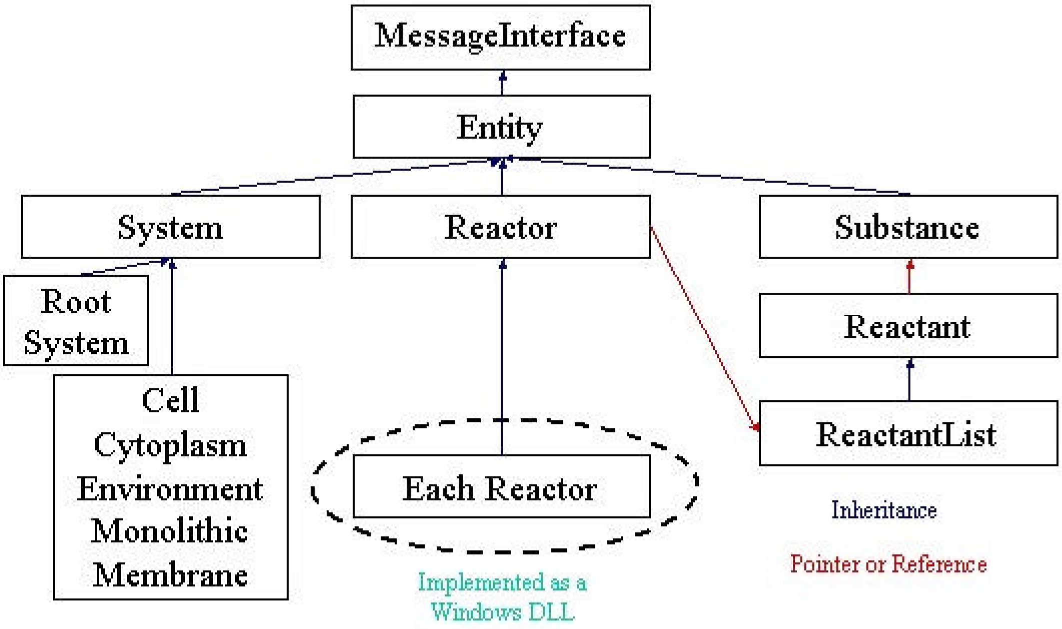


Figure 3. Internal structure of E-CELL2

The differences between E-CELL2 and the Linux-based E-CELL1 are listed below.

- Multiple class inheritance has been abolished in E-CELL2.
- Friend functions have also been abolished in E-CELL2.
- The "Integrator" and "Accumulator" classes have been integrated into the "Substance" class.
- The " Stepper"class has been integrated into the "System" class.
- A modeling launcher has been implemented

These changes were made to improve the portability of E-CELL1, which only works on Linux. The GUI (Graphic User Interface) and the modules for interpreting the script files for automating the operation of E-CELL2 are written in Java.

In E-CELL2, there is a GUI mode that allows for interactive operation, and a batch mode that supports command-line execution. This chapter provides explanations for only the former mode (GUI mode). Please refer to Chapter 3 Section 2.9 "Batch mode and Logger", for explanations on the latter mode(batch mode). Please note that the accuracy of the GUI mode and that of the batch mode is 64 bits and 80 bits, respectively. The inferior accuracy of the GUI mode is due to the limitations in numerical computation that exist in the Java Native Interface (JNI).

2 Demonstration of E-CELL2.26

E-CELL2 will be operated here using an actual model. In this section, a metabolic model of the Human erythrocyte will be simulated using E-CELL2. Why is an erythrocyte suitable as a target of simulation? The answers to this question are as follows.

- Mature erythrocyte cells do not undergo transcription, translation, or replication, and are therefore simple.
- Because erythrocyte cells are relatively easy to handle as experimental materials, there is a large accumulation of biochemical experimental data available.

The main metabolic pathways of an erythrocyte consist of glycolysis, pentose phosphate cycle, and nucleotide synthesis. In addition, large amounts of oxygen-carrying hemoglobin exist in an erythrocyte. An erythrocyte cell can therefore be likened to a bag full of hemoglobin that has self-regulating functions based on these metabolic pathways. Although this model aims at the reconstruction of only the metabolic pathways, it is expected that hemoglobin transport will be included in a future model.

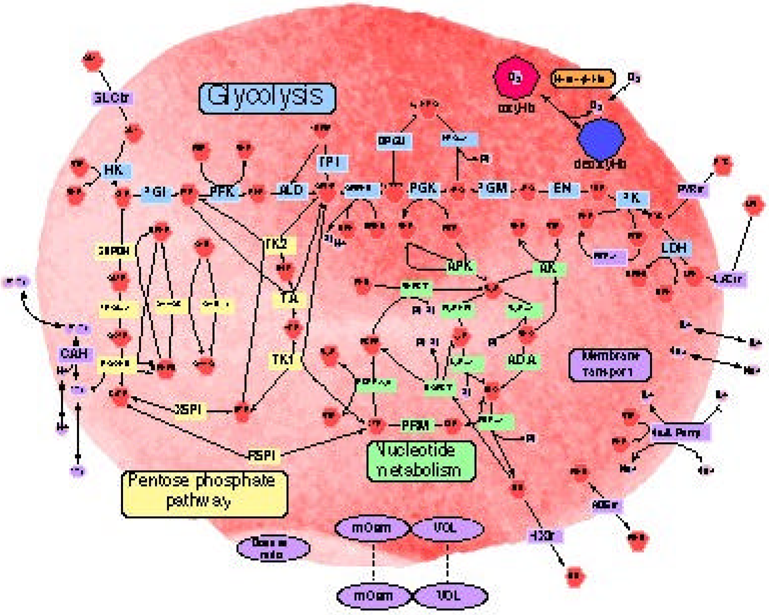


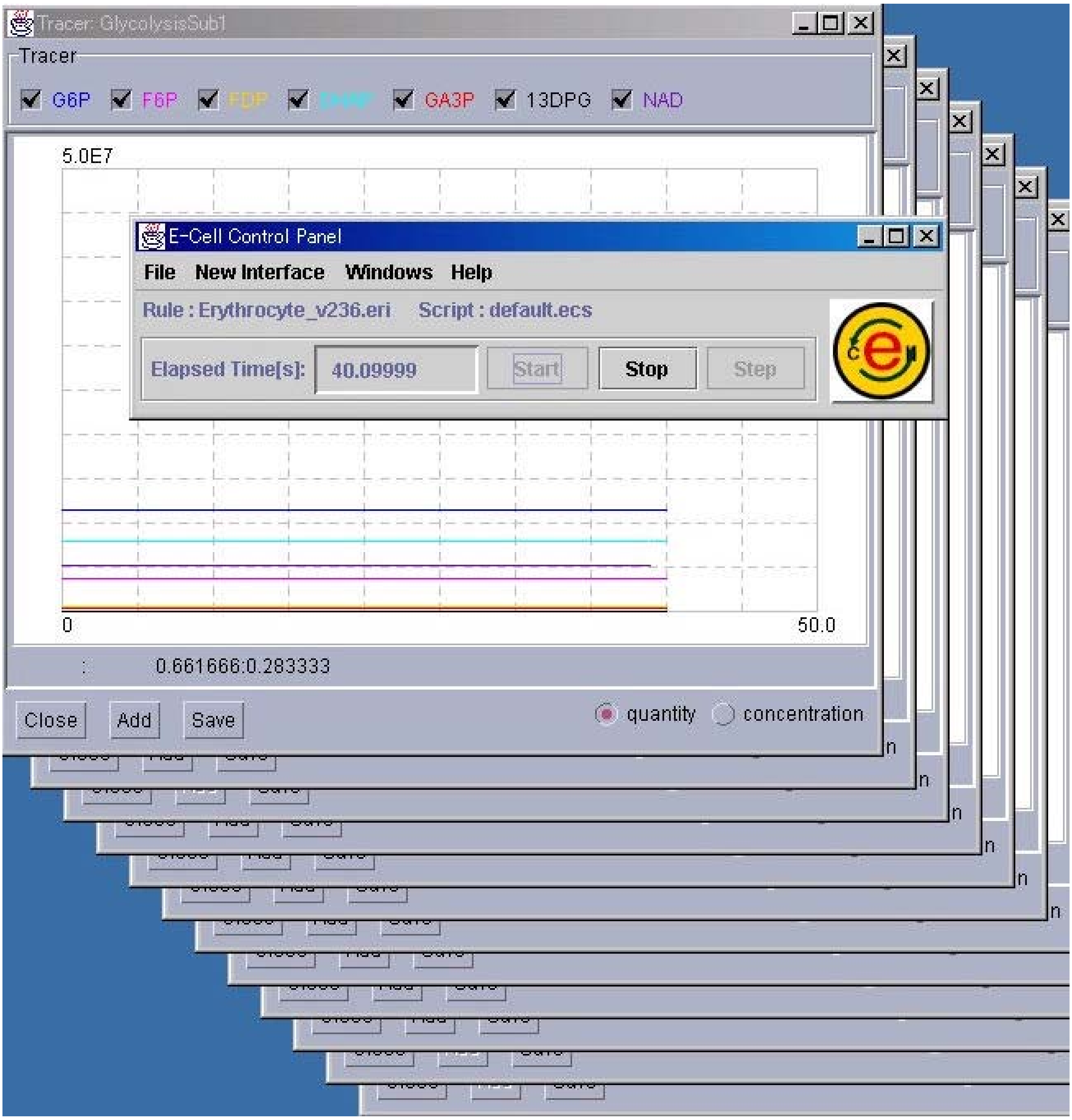
Figure 4. Model diagram of the Human erythrocyte metabolic pathway

This metabolic model of the Human erythrocyte cell was constructed by assistant professor Nakayama, professor Tomita, and their colleagues at Keio University. Although only the metabolic model of a normal human erythrocyte is introduced in this demonstration, models of anemia caused by genetic disorders are also currently being simulated. Please contact assistant professor Nakayama (ynakayam@sfc.keio.ac.jp) for further information.

To observe an actual simulation of the erythrocyte cell, please select [Program]-[E-CELL2]-[erythrocyte] from [Start], or click on the icon in your desktop.

E-CELL2, loaded with the erythrocyte model, will start. If E-CELL2 finds the file "default.ecs" in the same directory as ECELL2.BAT, it reads the file(default.ecs) and attempts to start simulation. Because a default.ecs file has already been prepared for the erythrocyte model shown in Fig. 4, the demonstration will automatically start. The file "Erythrocyte_v236.eri" describes the parameters and initial values for this simulation.

Files with the ".ecs" extension usually are scripts for automating E-CELL2, and include instructions for operating E-CELL2. For details, please refer to Section 2.3 and Section 3.4. The simulation of the erythrocyte model shown in Fig. 4 will start by executing this script. Using this script, simulation would stop after 500 seconds.



☑ G6P ☑ F6P ☑ FDP ☑ DHAP ☑ GA3P ☑ 13DPG ☑ NAD

● quantity ○ concentration

Rule : Erythrocyte_v236.eri Script : default.ecs

Elapsed Time[s]: 40.09999

Figure 5. Simulation of the erythrocyte model

To terminate E-CELL2 before the script stops the simulation, please press the "Stop" button in the Control Panel, and then select [Quit] from the [File] menu.

In the next section, the operating procedure for E-CELL2 will be briefly explained. After understanding the basic operation procedure, please reload E-CELL2 and experiment with it.

3 Creating Rules for E-CELL2

Before going into the details, a simple simulation model will be introduced. The figure below illustrates the so-called "Toy" model, which includes a simple feedback system.

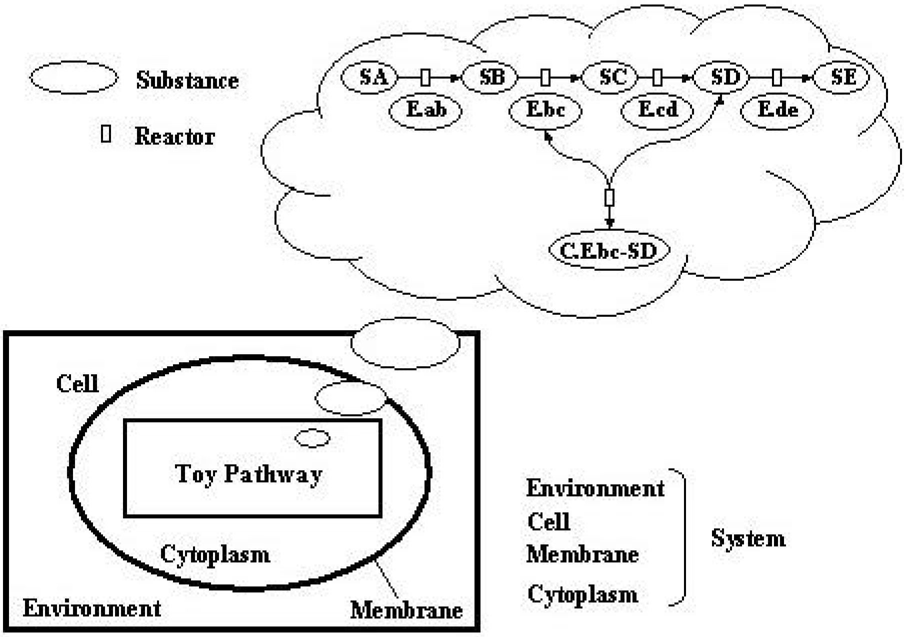


Figure 6. Toy model

5 **"S"ubstrate**s, 4 **"E"nzyme**s, and 1 **"C"omplex** constitute the model. The **Substrate**s are interconnected via 4 **Reactor**s (chemical reaction equations). **Reactor**s embody chemical reaction equations, such as those based on the Michaelis-Menten scheme. In the case of a cultured cell, these reactions would exist inside a Cell, surrounded by culture medium called **Environment**. A Membrane compartmentalizes the external world and the cell interior, which is filled with Cytoplasm.

The network of chemical reactions has now been depicted as static graph.

However, to actually start simulation, the rules that define the parameters for each reaction(e.g. initial values, rates),and how the reactions proceed(i.e. which chemical reaction equations to choose), have to be decided.

When creating models for E-CELL, simulation rules called Rule Files, which are distinguished by the ".er" extension, are described using text editors. Then, using tools included with E-CELL, files that can be directly read by E-CELL called Rule Intermediate Files are created. Rule Intermediate Files are usually distinguished by the ".eri" extension.

Because it is difficult for beginners of E-CELL to describe Rule Files from scratch, tools for creating Rule Files from spreadsheet files are also provided to support the Rule File creation process.

3.1 Creating Spreadsheets

Here, the methods for creating spreadsheets for simulation, and converting them to Rule Intermediate Files for E-CELL2 will be outlined.

The minimum operating procedures necessary for creating Rule Files using the modeling launcher, which is included with E-CELL2, will be explained.

To start the modeling launcher, please select [Program]-[E-CELL2]-[ModelingLauncher] from [Start], or click on the icon in your desktop.

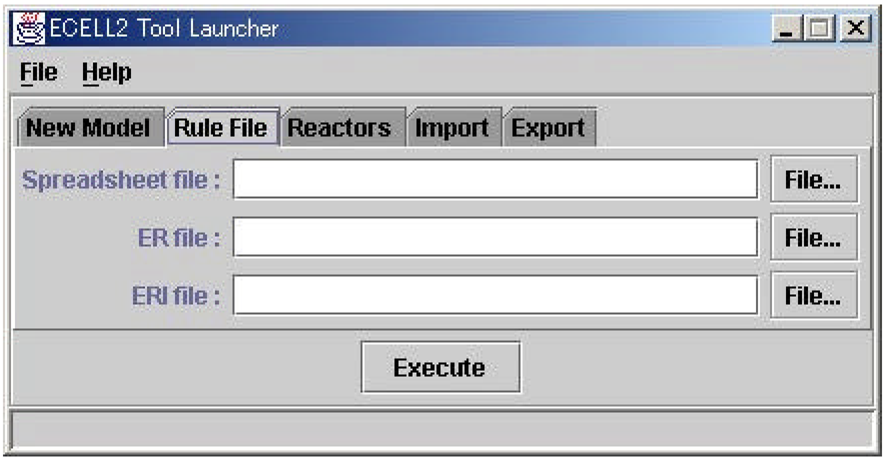
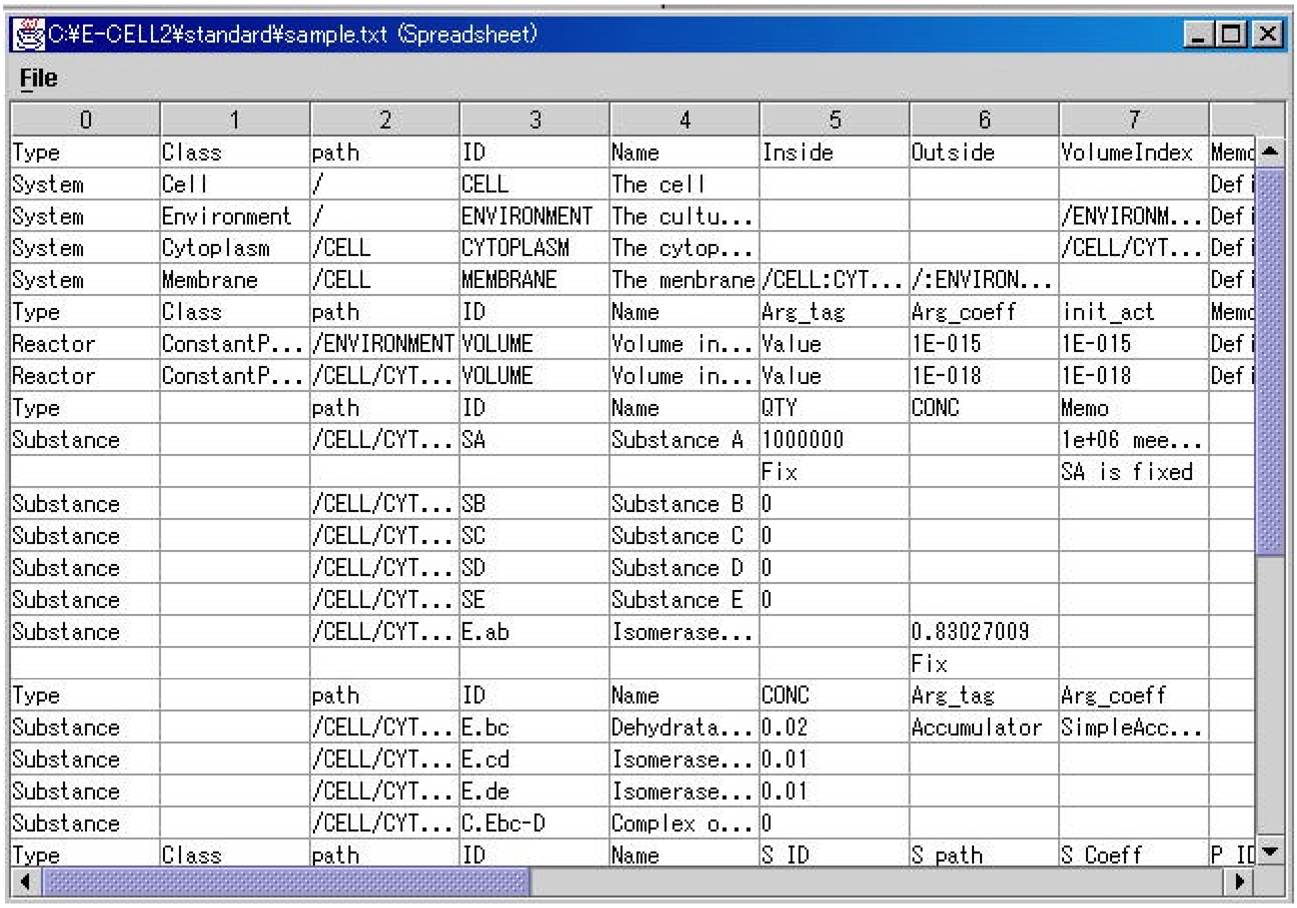


Figure 7. Window of the modeling launcher(1)

Please try loading the sample Rule File for E-CELL2. Please click on the File.. button in the Spreadsheet file column, select Edit, and open the file selection window. Then, please search and load "sample.txt".



0	1	2	3	4	5	6	7	
Type	Class	path	ID	Name	Inside	Outside	VolumeIndex	Memo
System	Cell	/	CELL	The cell				Def i
System	Environment	/	ENVIRONMENT	The cultu...			/ENVIRONM...	Def i
System	Cytoplasm	/CELL	CYTOPLASM	The cytop...			/CELL/CYT...	Def i
System	Membrane	/CELL	MEMBRANE	The menbrane	/CELL:CYT...	/:ENVIRON...		Def i
Type	Class	path	ID	Name	Arg_tag	Arg_coeff	init_act	Memo
Reactor	ConstantP...	/ENVIRONMENT	VOLUME	Volume in...	Value	1E-015	1E-015	Def i
Reactor	ConstantP...	/CELL/CYT...	VOLUME	Volume in...	Value	1E-018	1E-018	Def i
Type		path	ID	Name	QTY	CONC	Memo	
Substance		/CELL/CYT...	SA	Substance A	1000000		1e+06 mee...	
					Fix		SA is fixed	
Substance		/CELL/CYT...	SB	Substance B	0			
Substance		/CELL/CYT...	SC	Substance C	0			
Substance		/CELL/CYT...	SD	Substance D	0			
Substance		/CELL/CYT...	SE	Substance E	0			
Substance		/CELL/CYT...	E.ab	Isomerase...		0.83027009		
						Fix		
Type		path	ID	Name	CONC	Arg_tag	Arg_coeff	
Substance		/CELL/CYT...	E.bc	Dehydrata...	0.02	Accumulator	SimpleAcc...	
Substance		/CELL/CYT...	E.cd	Isomerase...	0.01			
Substance		/CELL/CYT...	E.de	Isomerase...	0.01			
Substance		/CELL/CYT...	C.Ebc-D	Complex o...	0			
Type	Class	path	ID	Name	S_ID	S_path	S_Coeff	P_ID

Figure 8. Window of the modeling launcher(2)

After finishing editing, please save the file by choosing [Save] or [Save As] from the [File] menu.

3.2 Creating Rule Files and Rule Intermediate Files

The procedures for converting spreadsheets into Rule Files and Rule Files into Rule Intermediate Files will be explained. Please refer to Chapter 4 for details regarding Rule Files.

Please start the modeling launcher and load "sample.txt " and "toy.txt ". The file "toy.txt " is an incomplete spreadsheet file for the Toy model. These files are located in the tutorial-related "standard" directory.

It has already been explained that "**Systems**", "**Substances**", and "**Reactors**" exist in the **Substance-Reactor** model of E-CELL. The leftmost column in the spreadsheet is called Type. Each field will include one of the above Types. Systems are used to show the structures of a cell or the localization of substances. Systems usually include "**Environment**", "**Cell**", "**Membrane**", and "**Cytoplasm**". Please refer to Section 4.2.2 for details regarding Systems. Please keep in mind here that embedded structures such as cells can be described by distinguishing "**Inside**" from "**Outside**".

A **Substance** implies a broad definition of a substance. The spreadsheet includes a description similar to the table shown below. In the "Toy" model, **Substance**s are the materials undergoing chemical reactions and the enzymes catalyzing the reactions.

Table 2.1 Example descriptions of Substances used as materials for chemical reactions

Type	**path**	**ID**	**Name**	**QTY**
Substance	/CELL/CYTOPLASM	SA	Substance A	1000
				FIX
Substance	/CELL/CYTOPLASM	SB	Substance B	0
Substance	/CELL/CYTOPLASM	SC	Substance C	0

Path is the location of the **Substance**. It is described in a fashion similar to the directory path in UNIX. **ID** is the **ID** of the specified **Substance**, and is an essential item. **Name** is the name of the specified **Substance**. Either the **Qty** or the **Conc** for a **Substance** has to described. **Qty** is the initial "**Quantity**" value of the **Substance**, whereas **Conc** is the initial "**Concentration**" value of the **Substance**. The unit of **Conc** is mol/l. If the **Quantity** and **Conc** should not change during simulation, the values may be fixed by adding a "**Fix**" tag to the row right beneath. Please refer to Section 4.2.3 for details regarding **Substance**s.

The following are descriptions regarding enzymes.

Table 2.2 Example descriptions of Substances that catalyze reactions as enzymes

Type	**path**	**ID**	**Name**	**CONC**
Substance	/CELL/CYTOPLASM	E.bc	Enzyme B	0.02
Substance	/CELL/CYTOPLASM	E.cd	Enzyme C	0.01
Substance	/CELL/CYTOPLASM	C.E.bc-SD	Complex of E.bc and SD	0

<Problems>

1. In "toy.txt", a few descriptions for Substance D and Substance E are missing. Please fill in the gaps by referring to "sample.txt".
2. Similarly, please complete the descriptions for Isomerase of A and Isomerase of D.

Reactors on the other hand are implemented chemical reactions. An example is defined below.

Table 2.3 Example descriptions of Reactors

Type	Class	path	ID	Name	S_ID	S_path
Reactor	MichaelisUniUniReactor	/CELL/CYTOPLASM	E.ab-0	A->B	SA	/CELL/CYTOP
Reactor	MichaelisUniUniReactor	/CELL/CYTOPLASM	E.bc-0	B->C	SB	/CELL/CYTOP
Reactor	MichaelisUniUniReactor	/CELL/CYTOPLASM	E.cd-0	C->D	SC	/CELL/CYTOP

Path is the location of the **Reactor**. It is described in a fashion similar to the directory path in UNIX. **S_ID** is the "**Substrate**", or the starting substance of the chemical reaction. **S_Path** is its location. **P_ID** is the "**Product**", or the resulting substance of the chemical reaction. **P_Path** is its location. **C_ID** is the "**Catalyst**", or the chemical substance that promotes or represses the reaction. **C_Path** is its location. In the **Arg_tag** column, the constant names that are defined differently according to the type of **Reactor** are inputted. In the **Arg_coeff** column, the constant values that correspond to the **Arg_tag** are inputted.

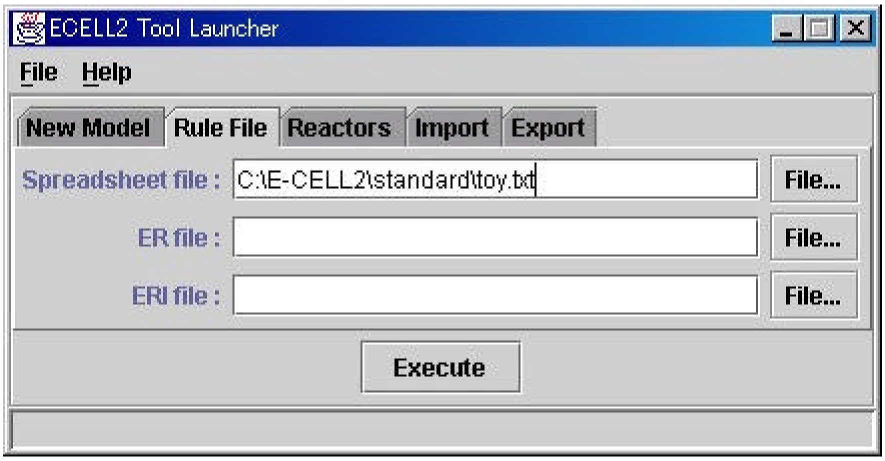
Reactors whose IDs begin with "!" are back door Reactors, used for describing reactions that cannot be described by differential equations. In the "Toy" model, "!EQ-Ebc-D" is the algebraic equation-based back door Reactor. Ebc+SD < - > C.Ebc-D is a reversible reaction that consists of a reaction that produces C.Ebc-D from Ebc and SD, and a reaction that decomposes C.Ebc-D. Therefore, it cannot be solved by differential equations. Thus, "**RapidEquibriumPReactor**" for rapid equilibrium is used here.

<Problems>
1. Please complete the descriptions for the reaction in Reactor D->E.

Please refer to Section 4.2.4 for details regarding **Reactor**s. If all of the above operations have been finished, please save "toy.txt", and start working on creating Rule Files.

The actual operations for creating Rule Files from spreadsheet files will be explained using the model launcher.

Please click on the "File..." button in the spreadsheet file column, and then open the file selection window by selecting "Choose...". Please search and load "toy.txt".



Spreadsheet file : C:\E-CELL2\standard\toy.txt

ER file :

ERI file :

Figure 9. Selection of toy.txt

If the Execute button is clicked, a Rule File "toy.er" and a Rule Intermediate File "toy.eri" will be created.

The creation of Rule Files and Rule Intermediate Files have been explained in this section. Please proceed to the next section on creating Reactors.

4 Creating User-defined Reactors for E-CELL2

To describe a chemical reaction that is not included in the standard reactors, one must design and create a Reactor for E-CELL2.

4.1 Outline of Reactors

- **Reactor**s calculate the temporal changes in the quantity of **Substance**s.
- The source code is described in C++ language.
- Writing a **Reactor** in E-CELL, means writing a Reactor-describing (Reactor Description, RD) file (e.g., filename.rd).
- In an RD file, the specifications of the **Reactor** and the contents of the processes to be executed are written.
- Using rd2ch.pl and rd2tex.pl, RD files are converted into tex files(i.e. Reactor Spec Sheets in LaTeX format) and ".dll" files(i.e. Dynamic link libraries in an E-CELL-loadable format).

RD files are created by arranging lines consisting of pairs of keywords and values.

RD files can roughly be divided into the following three parts;

- General information
- Reactor Spec Sheet
- Reactor Source Code

When creating Reactor Description files, please conform to the following rules.

- Keywords consist of capital letters or "_" , and do not contain any spaces.
- Keywords begin with "@" or "%". The contents of lines beginning with "@" are directly read. The contents of lines beginning with "%" are delimited by "," and processed as arrays.
- Lines beginning with # are treated as comments. To directly output the "#" in the

beginning of a line, please type "¥#" or place a space before "#". However, the "#" other than those located in the beginning of a line are directly outputted.

- Lines that do not include keywords at the beginning are interpreted as a continuation of the previous lines. (To start a new line when converting to Spec Sheets, please place a " ¥¥ " where the new line should start)
- Keywords may be omitted if they are not necessary.

An example RD file is shown below.

1. This is the file for the **MichaelisUniUniReactor**. This **Reactor** is based on the following rate reaction.

$$v = \frac{KcF[E][S]}{KmS + [S]}$$

The file for the **MichaelisUniUniReactor**

```
@CLASSNAME:MichaelisUniUniReactor

@BASECLASS: FluxReactor
@AUTHOR: E-CECLL Tutorial
@EMAIL: tutorial@e-cell.org
@DATE: 2000 12/12

%VERSION: ecs-v1, 0.1

@BRIEF_DESCRIPTION:Unireactor enzyme activity of which kinetics can be described by the Hen

@DESCRIPTION:A reactor class for unireactant enzyme activity where kinetics can be described
by the Henri-Michaelis-Menten equation derived from rapid equilibrium assumptions.
\vspace{0.2cm}

This reactor is applicable to the following reaction sequence:

\begin{center}
$E+S \rightleftharpoons ^{k_{1}}_{k_{-1}}ES
\rightarrow^{k_{p}}E+PS
\end{center}
\vspace{0.3cm}

@EQUATION:$$v=\frac{(K_{cF}[E] [S]}{K_{mS}+[S]}$$

%SUBSTANCE:Substrate, 1, 1
%SUBSTANCE:Product, 1, 1
%SUBSTANCE:Catalyst, 1, 1
%SUBSTANCE:Effector, 0, 0

%PARAMETER: KmS, Float, mol/l, Michaelis Constant of Substrate
%PARAMETER: KcF, Float, mol/l, Catalytic Constant (Forward)

@REACT_FUNC:
Float S = substrate(0)->concentration();
Float E = catalyst(0)->quantity();

Float velocity = KcF * E * S;
Float Den = KmS + S;

velocity /= Den;
process(velocity);
```

2. General information-related keywords
    - @CLASSNAME: The class name of this **Reactor** (the file name minus the .rd extension)
    - @BASECLASS: The name of the base class inherited by this class
    - @AUTHOR: The name of the author
    - @EMAIL: The e-mail address of the author
    - @DATE: The creation date of this **Reactor**
    - %VERSION: The version of the E-CELL System, the version of this **Reactor**
    - @BRIEF_DESCRIPTION: A brief description of this **Reactor**

3. Reactor Spec Sheet-related keywords
    - @DESCRIPTION: A detailed description of this **Reactor**
    - @EQUATION: The equation of this **Reactor**, described under the LaTeX displaymath environment in either of the following methods
        1. $$ <Description of a mathematical expression> $$
        2. ¥begin{displaymath}<Description of a mathematical expression>¥end {displaymath}
        3. ¥[<Description of a mathematical expression >¥]
    - %SUBSTANCE: The definition of **Substance**. Can either be **Substrate**, **Product**, **Catalyst**, or **Effector**
    - @NOTES: Notes regarding the implementation of this **Reactor**

4. Reactor Source Code-related keywords
    - %PARAMETER: The parameter name, parameter type, unit, and descriptions regarding the parameter are written, delimited by ","
    - @PRIVATE: The private members of this object(Member variables should be defined as private members)
    - @PROTECTED: The protected members of this object(Can be referred from derived classes)
    - @PUBLIC: The public members of this object
    - @INITIALIZE_FUNC: Executes the initial settings. Includes range check of parameters and calculation of values that are constant throughout the simulation
    - @REACT_FUNC: Specifies the processes executed in every simulation step. The following two-step process is described. 1. Calculation of the rate 2. Increase or decrease in the quantity of substances, based on the rate The argument for the Process() method is the number of reacted molecules per second, and is a Float type.

Reactors have been briefly explained in this section. Please refer to Chapter 5 for details regarding Reactors.

4.2 Creating Reactors

A simple example will be given to explain how to create **Reactor**s.

Please start the modeling launcher, and then click on the Reactor tab. Please click on the "File.." button in the RD file column, then select "Edit.." and open the file selection window.

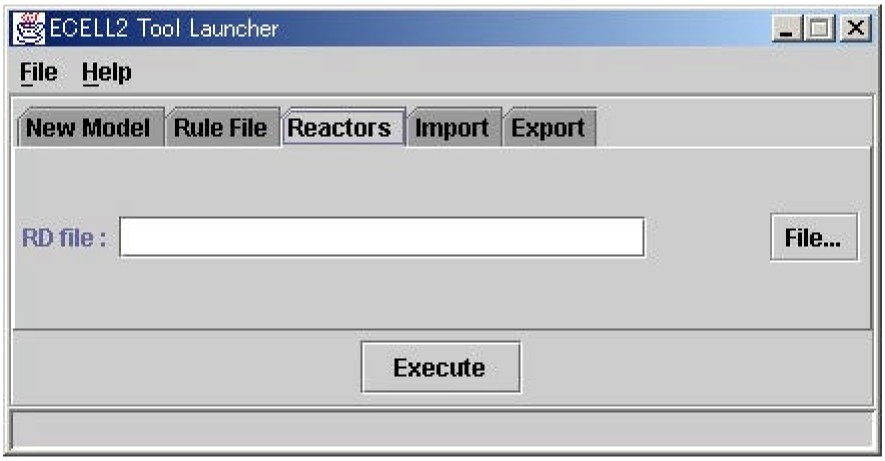
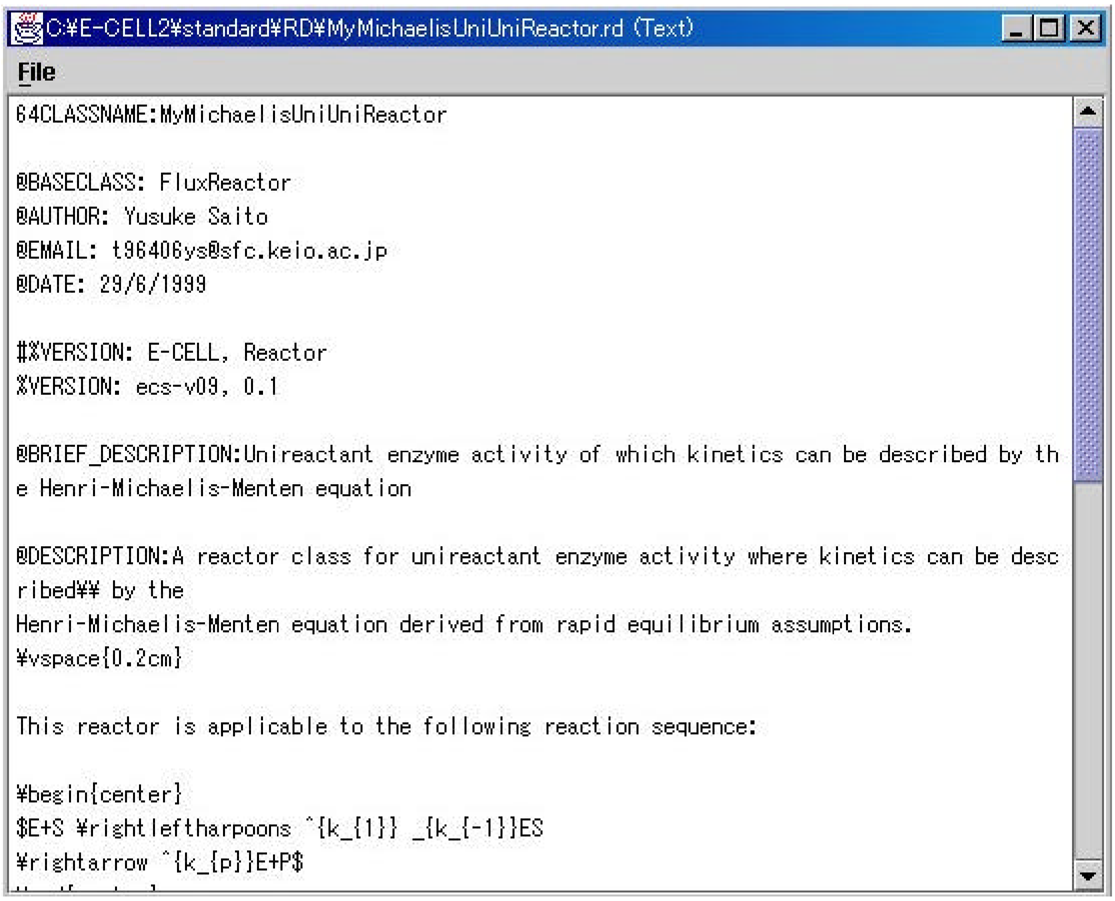


Figure 10. Reactor tab

Please open the file "MyMichaelisMentenUniUni.rd". Reactor Description files(RD files) are usually created as files with the ".rd" extension. Reactors are defined and described in these files.



```
64CLASSNAME:MyMichaelisUniUniReactor

@BASECLASS: FluxReactor
@AUTHOR: Yusuke Saito
@EMAIL: t96406ys@sfc.keio.ac.jp
@DATE: 29/6/1999

#%VERSION: E-CELL, Reactor
%VERSION: ecs-v09, 0.1

@BRIEF_DESCRIPTION:Unireactant enzyme activity of which kinetics can be described by th
e Henri-Michaelis-Menten equation

@DESCRIPTION:A reactor class for unireactant enzyme activity where kinetics can be desc
ribed¥¥ by the
Henri-Michaelis-Menten equation derived from rapid equilibrium assumptions.
¥vspace{0.2cm}

This reactor is applicable to the following reaction sequence:

¥begin{center}
$E+S ¥rightleftharpoons ^{k_{1}} _{k_{-1}}ES
¥rightarrow ^{k_{p}}E+P$
```

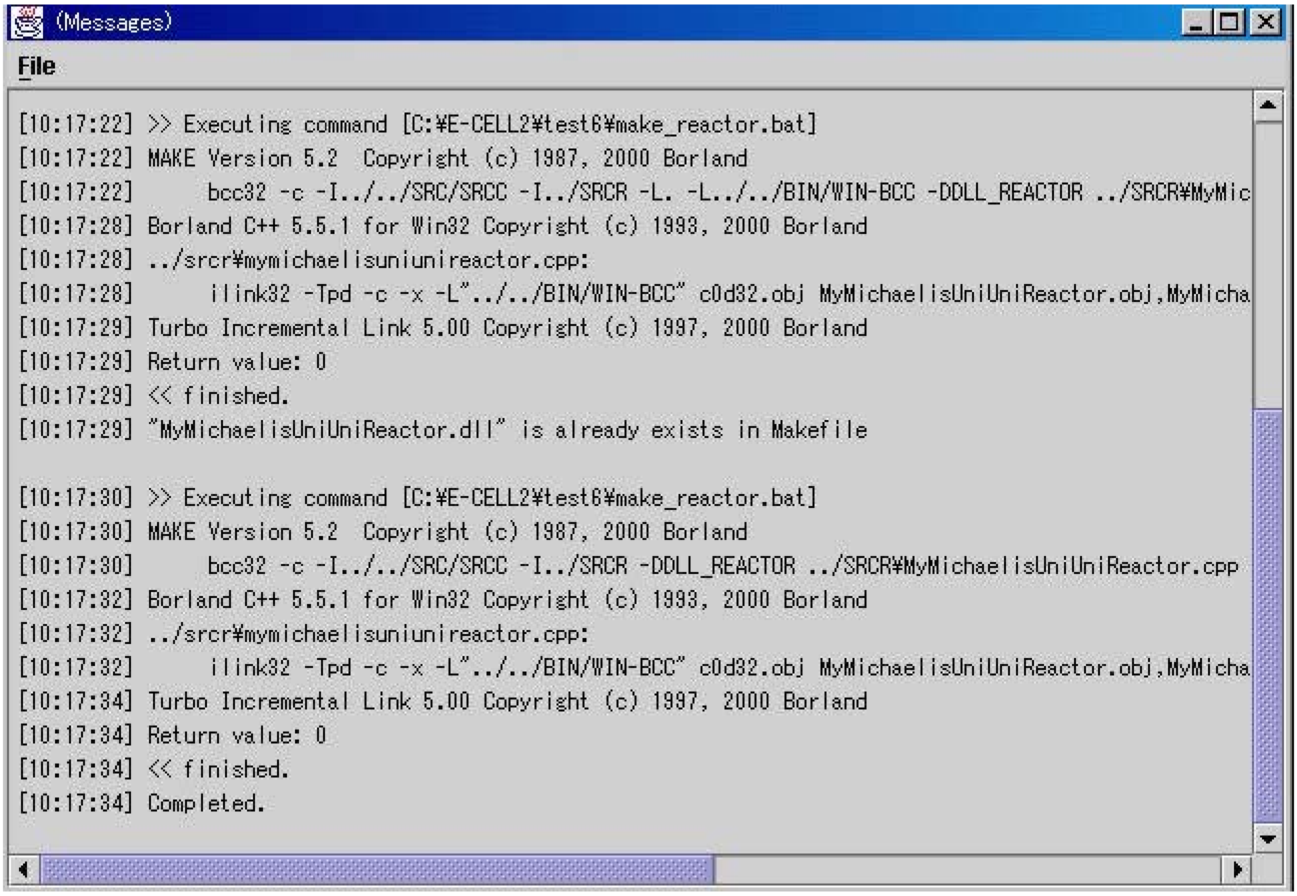
Figure 11. Example of a Reactor Description file

When creating Reactor Description files, please note that the file name minus the .extension has to directly become @CLASSNAME, as shown for this file. Except for their class names, the contents of MyMichaelisUniUniReactor.rd and MichaelisMentenUniUniReactor.rd are identical. The description files for the standard reactors would be located in "C:¥E-CELL2¥standard ¥RD" in a standard installation. When creating Reactor Description files, it would probably be helpful if the descriptions in these files are used for reference.

By using the modeling launcher to create C++ source codes and header files from Reactor

Description files, and then by compiling them, Reactor DLLs are created. Please press the "Execute" button to create the DLL.

If there are no syntax errors in the C++ code for the RD file, and if the Borland C++ Compiler has been properly set, a DLL will be created. Messages will be displayed in the message window during the compilation process.

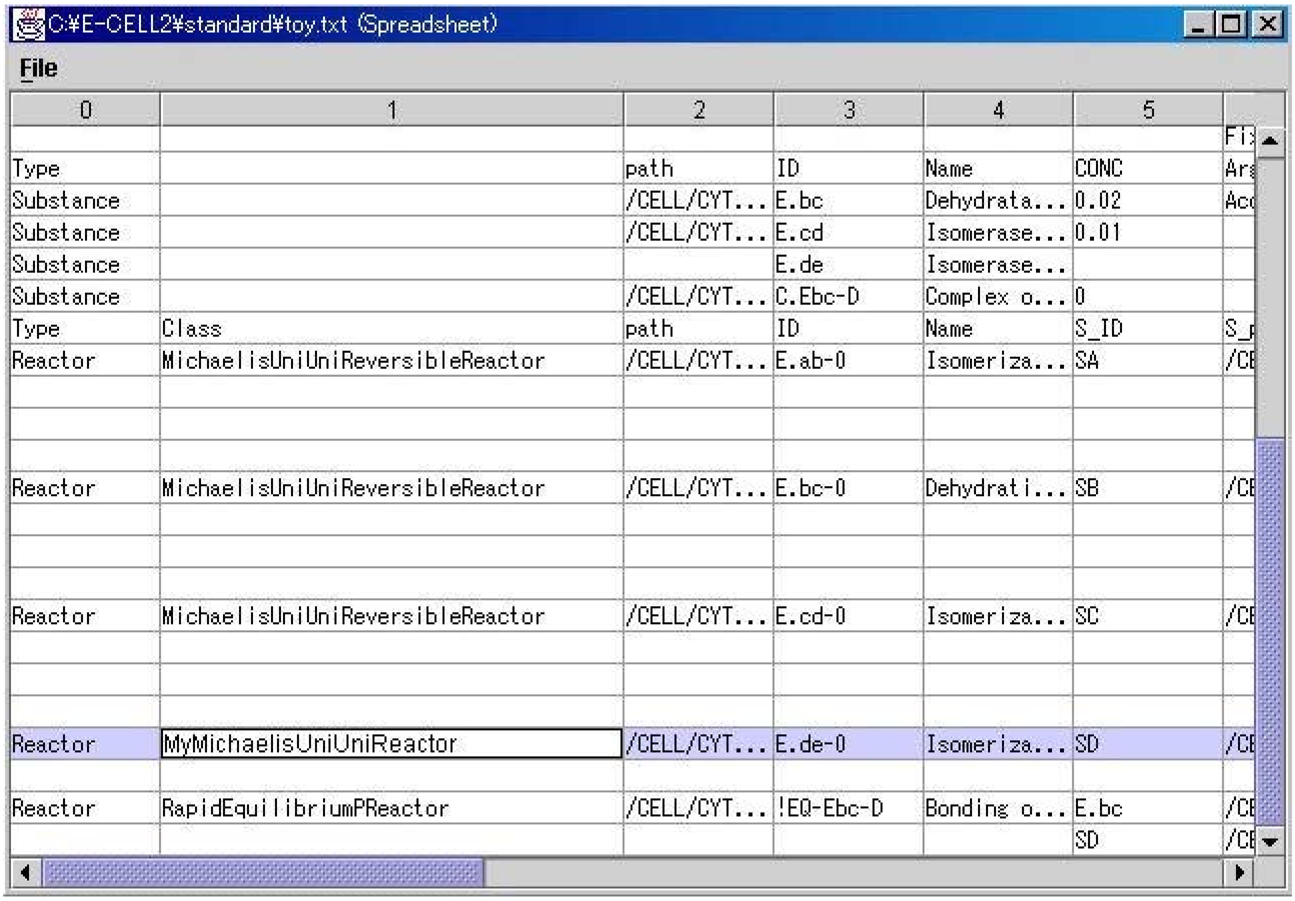


```
[10:17:22] >> Executing command [C:¥E-CELL2¥test6¥make_reactor.bat]
[10:17:22] MAKE Version 5.2  Copyright (c) 1987, 2000 Borland
[10:17:22]      bcc32 -c -I../../SRC/SRCC -I../SRCR -L. -L../../BIN/WIN-BCC -DDLL_REACTOR ../SRCR¥MyMic
[10:17:28] Borland C++ 5.5.1 for Win32 Copyright (c) 1993, 2000 Borland
[10:17:28] ../srcr¥mymichaelisuniunireactor.cpp:
[10:17:28]      ilink32 -Tpd -c -x -L"../../BIN/WIN-BCC" c0d32.obj MyMichaelisUniUniReactor.obj,MyMicha
[10:17:29] Turbo Incremental Link 5.00 Copyright (c) 1997, 2000 Borland
[10:17:29] Return value: 0
[10:17:29] << finished.
[10:17:29] "MyMichaelisUniUniReactor.dll" is already exists in Makefile

[10:17:30] >> Executing command [C:¥E-CELL2¥test6¥make_reactor.bat]
[10:17:30] MAKE Version 5.2  Copyright (c) 1987, 2000 Borland
[10:17:30]      bcc32 -c -I../../SRC/SRCC -I../SRCR -DDLL_REACTOR ../SRCR¥MyMichaelisUniUniReactor.cpp
[10:17:32] Borland C++ 5.5.1 for Win32 Copyright (c) 1993, 2000 Borland
[10:17:32] ../srcr¥mymichaelisuniunireactor.cpp:
[10:17:32]      ilink32 -Tpd -c -x -L"../../BIN/WIN-BCC" c0d32.obj MyMichaelisUniUniReactor.obj,MyMicha
[10:17:34] Turbo Incremental Link 5.00 Copyright (c) 1997, 2000 Borland
[10:17:34] Return value: 0
[10:17:34] << finished.
[10:17:34] Completed.
```

Figure 12. Messages during compilation

After the compilation process has finished, "**MyMichaelisMentenUniUniReactor**" will actually be used. Please open "toy.txt" using the modeling launcher.

Please replace "**MichaelisMentenUniUniReactor**" with "MyMichaelisMentenUniUniReactor", then save the file, and press the "Execute" button.



C:¥E-CELL2¥standard¥toy.txt (Spreadsheet)

File

0	1	2	3	4	5	
						Fi
Type		path	ID	Name	CONC	Ar
Substance		/CELL/CYT...	E.bc	Dehydrata...	0.02	Ac
Substance		/CELL/CYT...	E.cd	Isomerase...	0.01	
Substance			E.de	Isomerase...		
Substance		/CELL/CYT...	C.Ebc-D	Complex o...	0	
Type	Class	path	ID	Name	S_ID	S_
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYT...	E.ab-0	Isomeriza...	SA	/CE
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYT...	E.bc-0	Dehydrati...	SB	/CE
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYT...	E.cd-0	Isomeriza...	SC	/CE
Reactor	MyMichaelisUniUniReactor	/CELL/CYT...	E.de-0	Isomeriza...	SD	/CE
Reactor	RapidEquilibriumPReactor	/CELL/CYT...	!EQ-Ebc-D	Bonding o...	E.bc	/CE
					SD	/CE

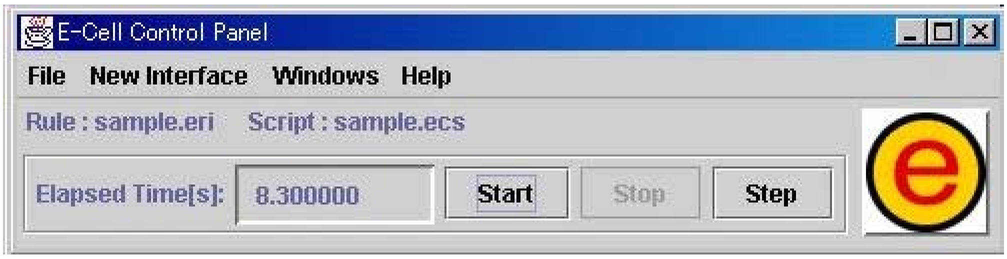
Figure 13. Rule File editing

Please start E-CELL2, then load "toy.eri" by selecting [Load Rule] from the [File] menu, and begin simulation. If the behavior is the same as that of "sample.eri", the Reactor has been successfully created.

Please continue working on Reactors so that in the future, you will be able to design and create new Reactors necessary for executing your own simulation model.

5. Operating E-CELL2

The goal of this section is to master the basic operation of E-CELL2, using the Rule Intermediate Files and Reactors that have been created. A Control Panel will first appear when E-CELL2 is started.



Rule : sample.eri Script : sample.ecs

Elapsed Time[s]: 8.300000

Figure 14. Control Panel of E-CELL2

The pull-down menus will be explained first.

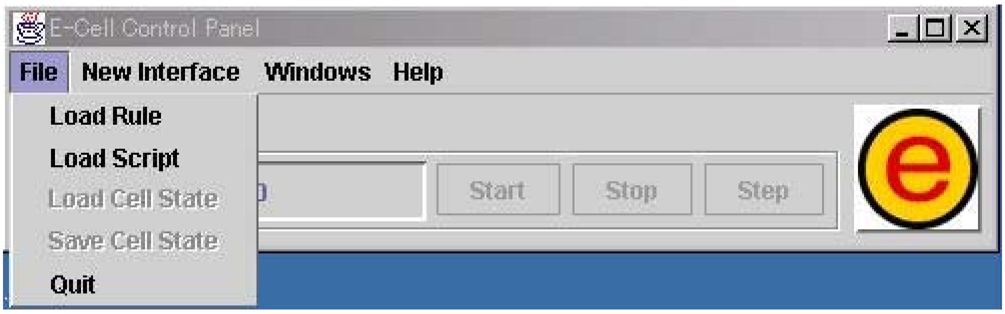


Figure 15. [File] menu

The items [Load Rule], [Load Script], [Load Cell State], [Save Cell State] and [Quit] are included in the [File] menu.

[Load Rule] is selected when loading Rule Intermediate Files.

[Load Script] is selected when loading scripts for automating the operation of E-CELL2.

[Load Cell State] is selected when resuming simulation by loading files that include the progress data of a cell state.

[Save Cell State] is selected when saving files that record the progress data of a cell state.

[Quit] is selected when terminating E-CELL2.

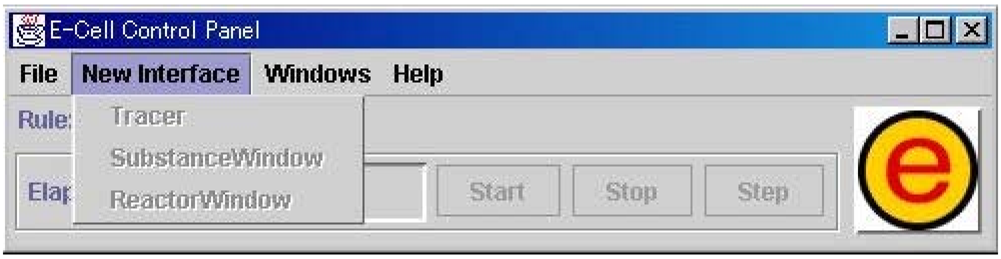


Figure 16. [New Interface] menu

The items [Tracer], [SubstanceWindow], and [ReactorWindow] are included in the [New Interface] menu.

[Tracer] is selected when creating a [Tracer] window, which represents the quantity of **Substance**s or the activity of **Reactors**.

[SubstanceWindow] is selected when opening windows for displaying and manipulating **Substance**s.

[ReactorWindow] is selected when opening windows for displaying **Reactor**s.

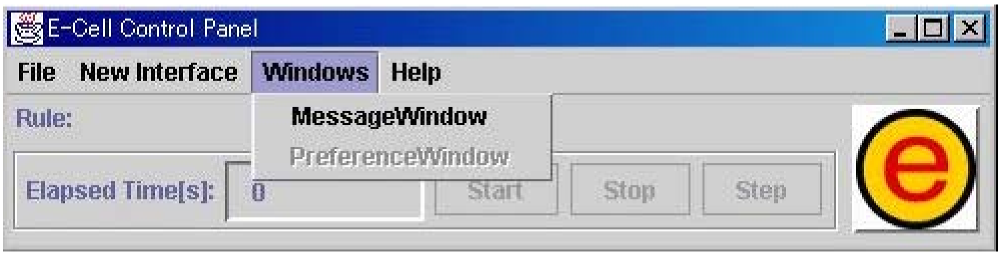
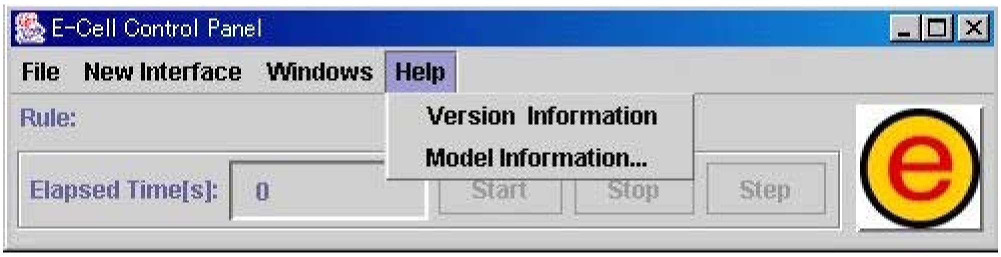


Figure 17. [Windows] menu

The items [MessageWindow] and [PreferenceWindow] are included in the [Windows] menu.

[MessageWindow] is used when opening windows for displaying messages from E-CELL2.

[PreferenceWindow] is selected when setting the width of a simulation time step and the timing of [Tracer] window updating.



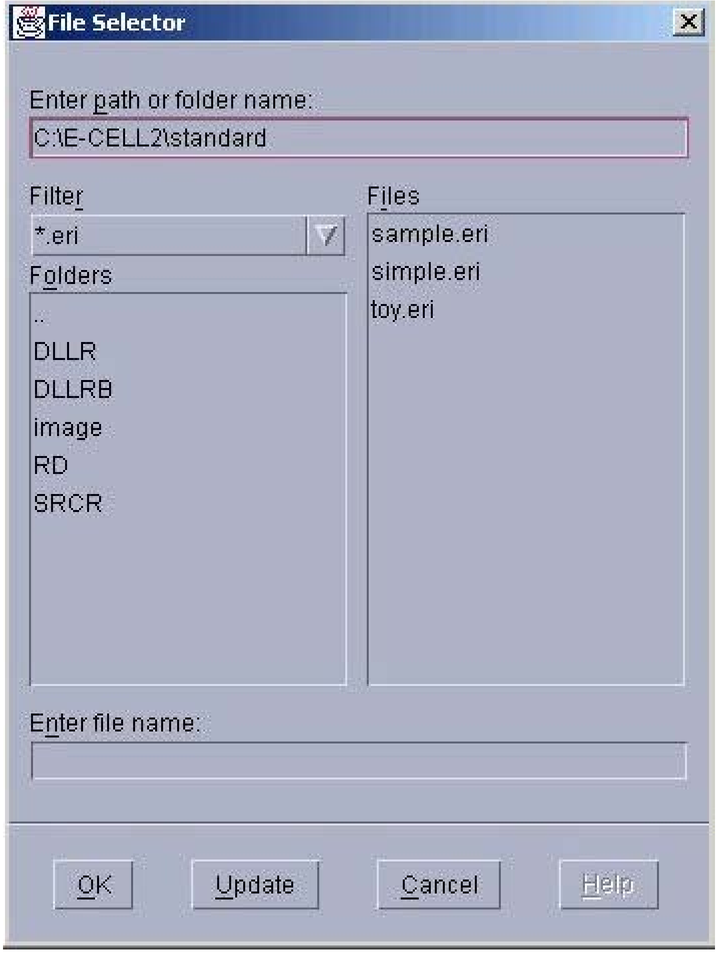
Elapsed Time[s]: 0

Figure 18. [Help] menu

The item [Version Information] is included in the [Help] menu. If selected, the version information will be displayed.

The menus have been explained above.

To load Rule Intermediate Files, please select [File]menu-[Load Rule]. A file selection window will be displayed. Please click on the name of the file that ends with the ".eri" extension, which indicates that it is a Rule Intermediate File, then select " Open ".



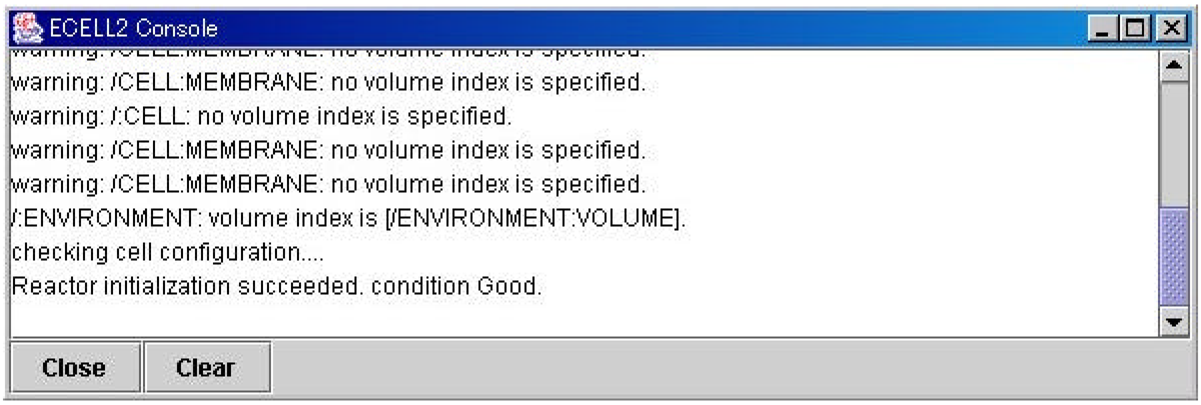
Enter path or folder name: C:\E-CELL2\standard

Filter: *.eri

Enter file name:

Figure 19. File selector window for Rule Intermediate Files

If the designated file is loaded normally, "Condition Good." will be displayed in the Message Window.



```
warning: /CELL:MEMBRANE: no volume index is specified.
warning: /CELL:MEMBRANE: no volume index is specified.
warning: /:CELL: no volume index is specified.
warning: /CELL:MEMBRANE: no volume index is specified.
warning: /CELL:MEMBRANE: no volume index is specified.
/:ENVIRONMENT: volume index is [/ENVIRONMENT:VOLUME].
checking cell configuration....
Reactor initialization succeeded. condition Good.
```

Figure 20. Messages during Rule Intermediate File loading

Next, please prepare for displaying the quantity of **Substance**s as graphs. If [New Interface] - [Tracer] is selected, a Tracer will be displayed.

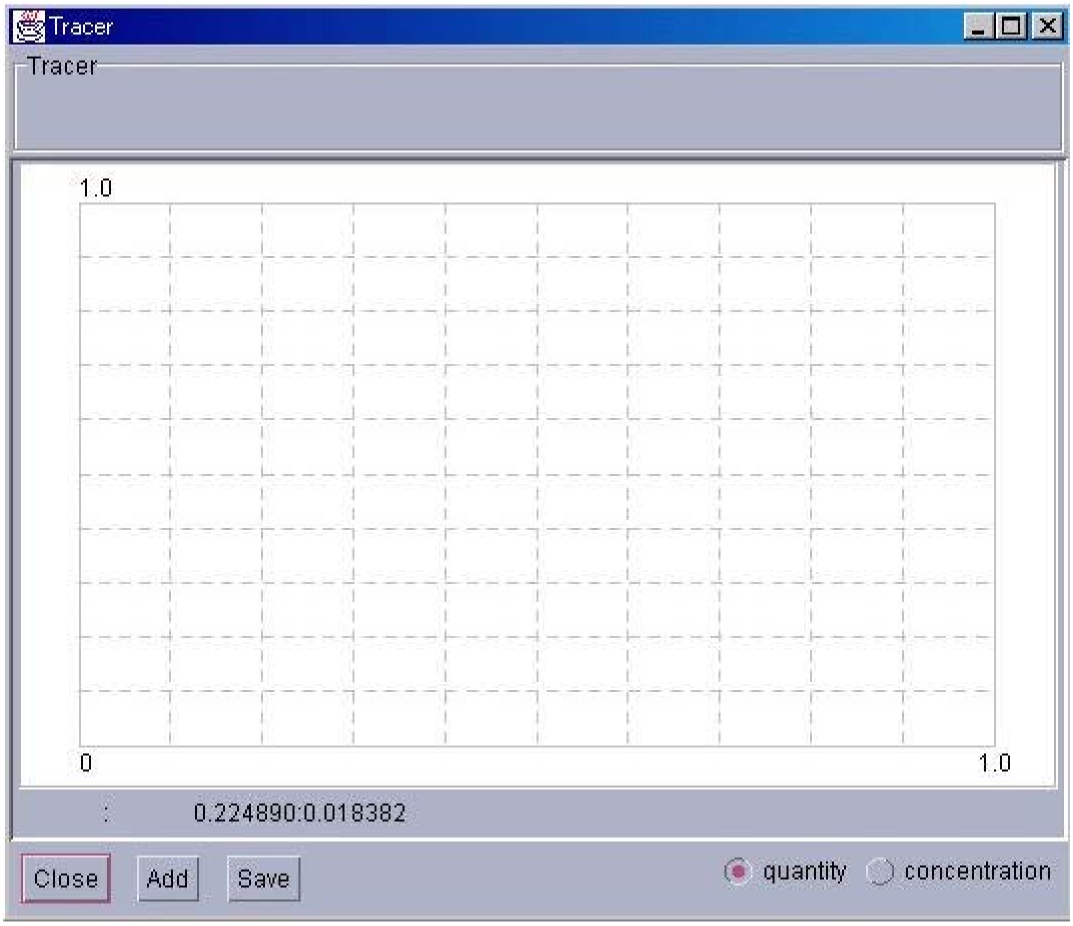
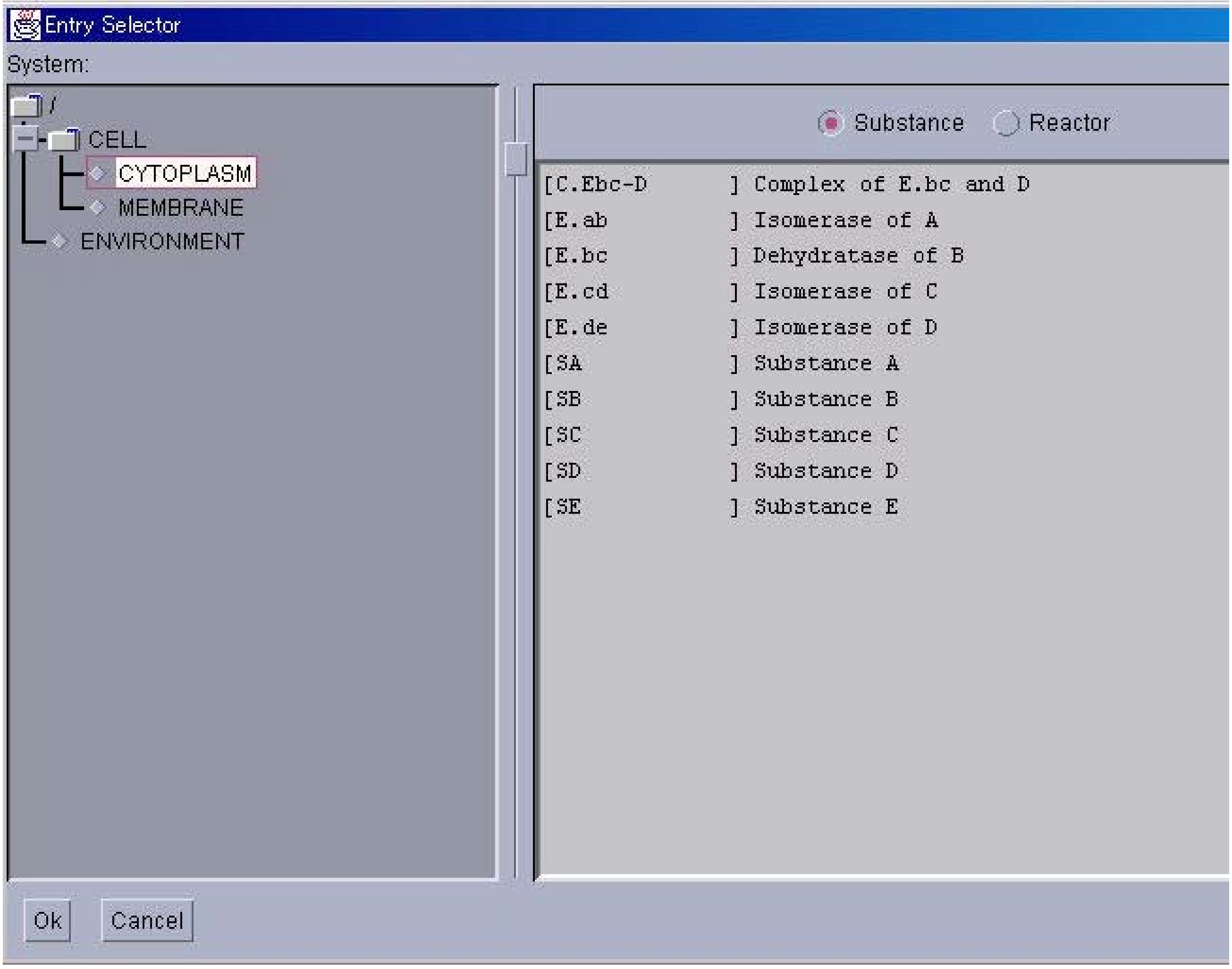


Figure 21. Tracer window

Please press the " Add " button in the lower left corner of the Tracer to display an [Entry Selector] window. Please click on the left panel, and select a **Substance** that is located in the **Cytoplasm**.



● Substance ○ Reactor

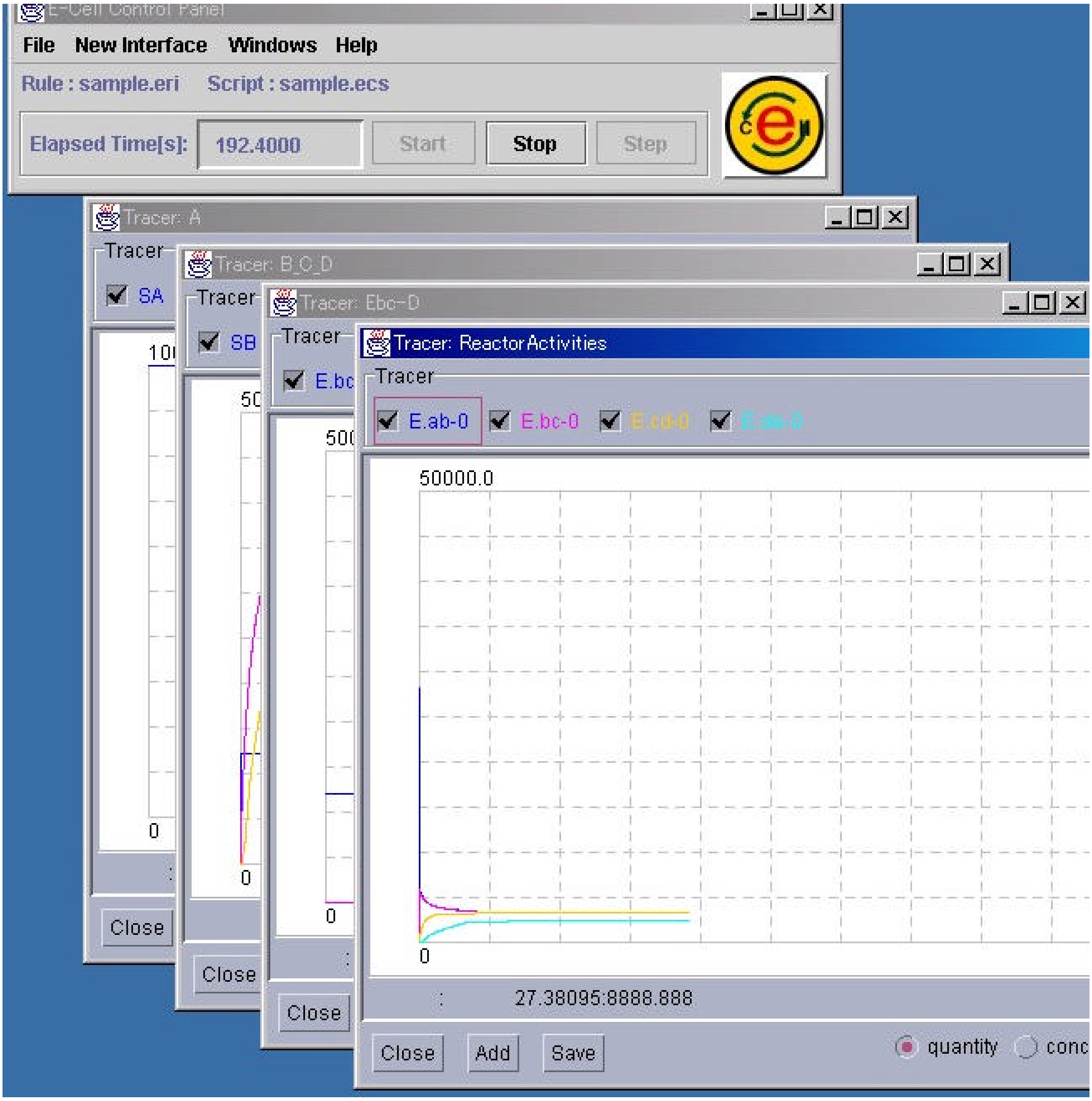
```
[C.Ebc-D       ] Complex of E.bc and D
[E.ab          ] Isomerase of A
[E.bc          ] Dehydratase of B
[E.cd          ] Isomerase of C
[E.de          ] Isomerase of D
[SA            ] Substance A
[SB            ] Substance B
[SC            ] Substance C
[SD            ] Substance D
[SE            ] Substance E
```

Figure 22. Entry Selector window (1)

Please click on the **Substance** radio button to display the Substances on the right panel. **Reactor**s can be similarly displayed.

The method for choosing two or more **Substance**s or **Reactor**s will now be explained. To select several entries, please click on the left mouse button while pressing the "Shift" key. After selection, please press the "Ok" button. Simulation can now be started by pressing "Start" in the Control Panel.

Up to 8 **Substance**s or **Reactor**s can be displayed on a single Tracer.



Rule : sample.eri   Script : sample.ecs

Elapsed Time[s]: 192.4000

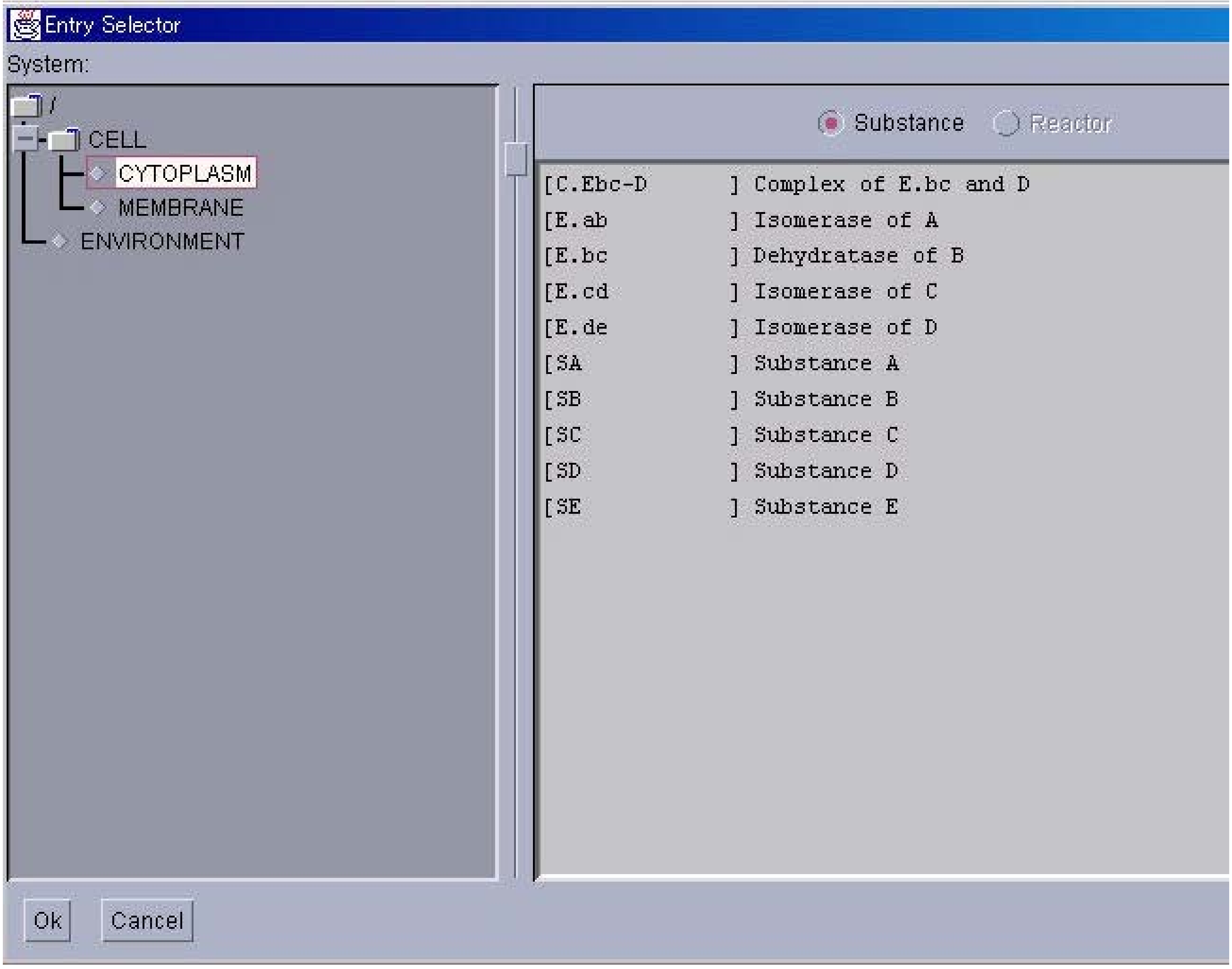
27.38095:8888.888

● quantity ○ conc

Figure 23. Example of a simulation using E-CELL2

Based on the rules, the quantity of **Substance**s or the activity of **Reactor**s will be displayed in the Tracer. The simulation results displayed in the Tracer can be saved by pressing the "Save" button in the Tracer. Please refer to Section 3.2.7 for details.

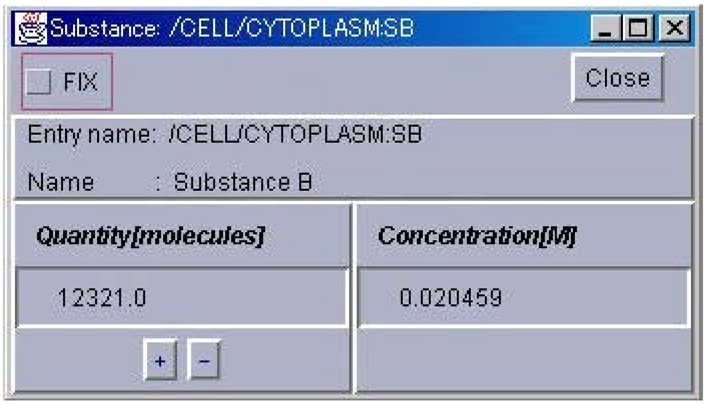
The quantity of **Substance**s can be changed manually to observe how the whole system is affected by such changes. Please select [Substance Window] from the [New Interface] menu.



```
[C.Ebc-D      ] Complex of E.bc and D
[E.ab         ] Isomerase of A
[E.bc         ] Dehydratase of B
[E.cd         ] Isomerase of C
[E.de         ] Isomerase of D
[SA           ] Substance A
[SB           ] Substance B
[SC           ] Substance C
[SD           ] Substance D
[SE           ] Substance E
```

Figure 24. Entry Selector window for SubstanceWindow

Because this Entry Selector is only for **Substance**s, **Reactor**s cannot be selected. If a **Substance** is selected and then the "Ok" button is pressed, a window displaying the individual details of the selected Substance will appear.



Entry name: /CELL/CYTOPLASM:SB

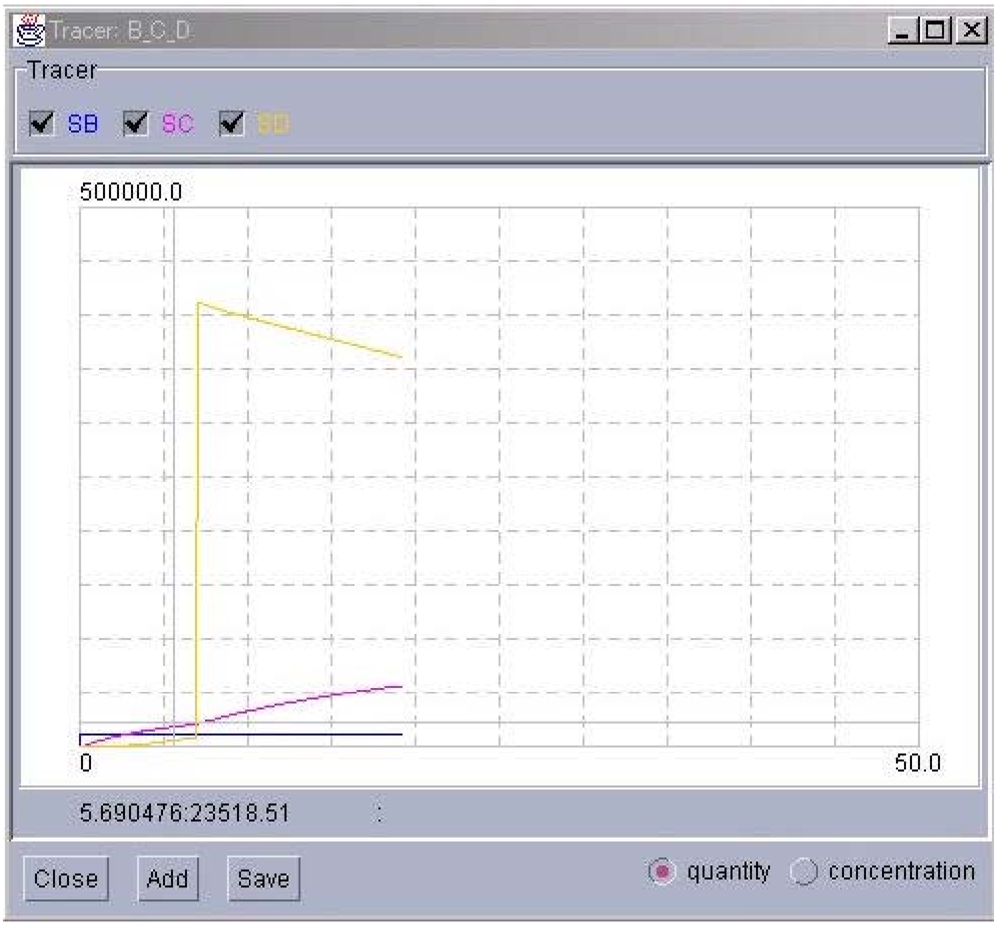
Name : Substance B

Quantity[molecules]	Concentration[M]
12321.0	0.020459

Figure 25. SubstanceWindow

The quantity of the designated **Substance** can be forced to increase or decrease by pressing the "+" and "-" buttons respectively, or by inputting a value in the "Quantity" or "Concentration" text field. If the "Fix" check button is checked, the quantity of the **Substance** can be made to ignore the computation results and maintain at a constant value. To terminate operation of the SubstanceWindow, please press the "Close" button.

Please try changing the quantity of a **Substance**, and then observe its effects on the system.

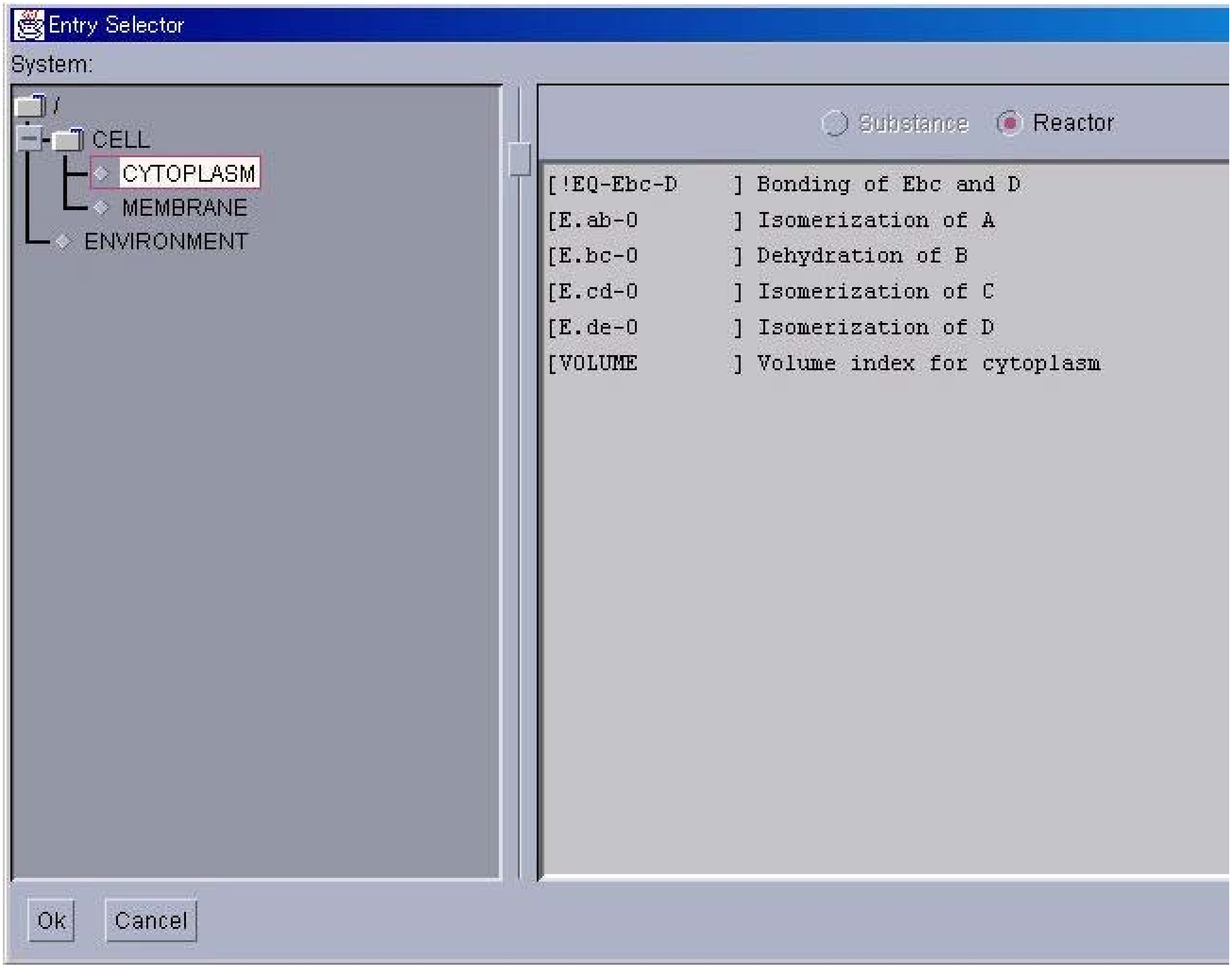


5.690476:23518.51 :

Figure 26. Behavioral change of a system caused by a change in the quantity of a Substance

In this example, "Substance D" has been suddenly increased. It can be observed that the increased **Substance**, compensated by the other reactions in the system, gradually decreases.

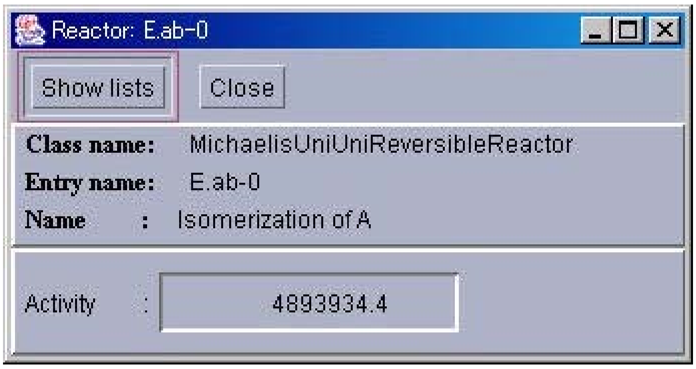
Similarly, the activity of **Reactor**s can also be displayed in detail. Please select [ReactorWindow] from the [NewInterface] menu in the Control Panel.



```
[!EQ-Ebc-D    ] Bonding of Ebc and D
[E.ab-0       ] Isomerization of A
[E.bc-0       ] Dehydration of B
[E.cd-0       ] Isomerization of C
[E.de-0       ] Isomerization of D
[VOLUME       ] Volume index for cytoplasm
```

Figure 27. Entry Selector window for ReactorWindow

As was the case with SubstanceWindow, please click on the System displayed in the left panel, and search for the target **Reactor**s. Please click on the "Ok" button after selecting the **Reactor**s.



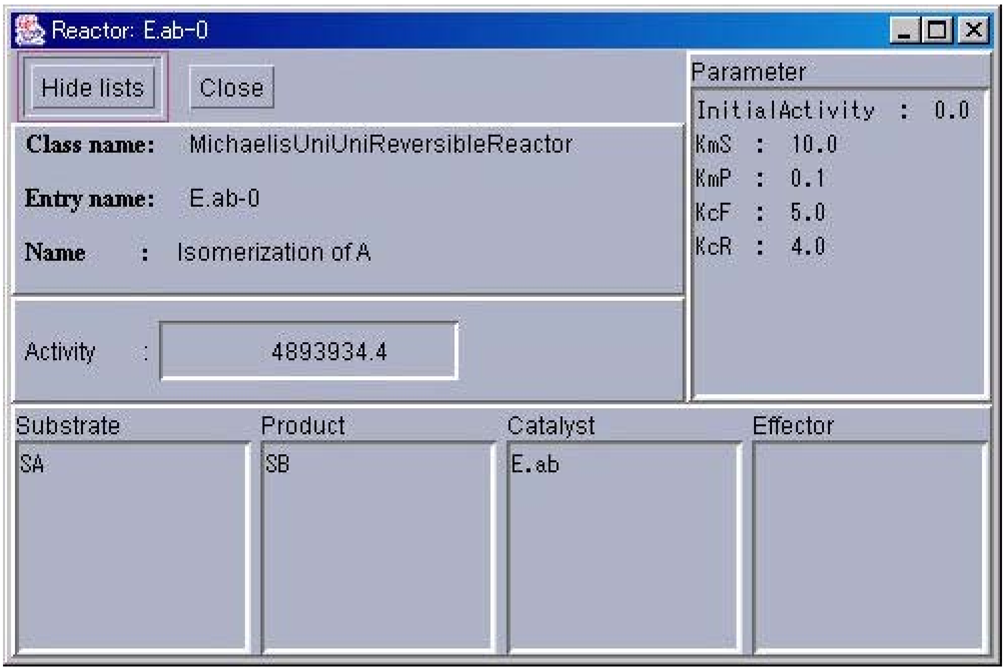
Class name: MichaelisUniUniReversibleReactor
Entry name: E.ab-0
Name : Isomerization of A

Activity : 4893934.4

Figure 28. ReactorWindow

In this window, the name of the selected Reactor, the type of reaction equation used, and the activity of the reaction is displayed. If the "Show lists" button is pressed, more details will be displayed. If the "Close" button is pressed, the window will close.

Please try pressing the "Show lists" button.



Class name: MichaelisUniUniReversibleReactor

Entry name: E.ab-0

Name : Isomerization of A

Activity : 4893934.4

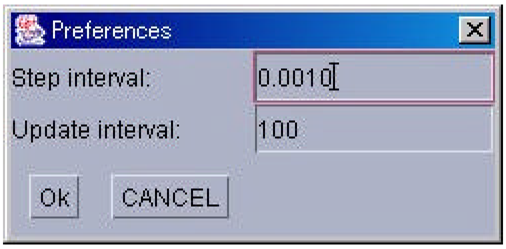
Parameter

```
InitialActivity : 0.0
KmS : 10.0
KmP : 0.1
KcF : 5.0
KcR : 4.0
```

Substrate	Product	Catalyst	Effector
SA	SB	E.ab	

Figure 29. ReactorWindow in the expanded state

Besides the information above, the initial values for each parameter and information regarding the substances participating in the reaction will be displayed. To return the window to the original size, please press the "Hide lists" button. To close the window, please press the "Close" button.



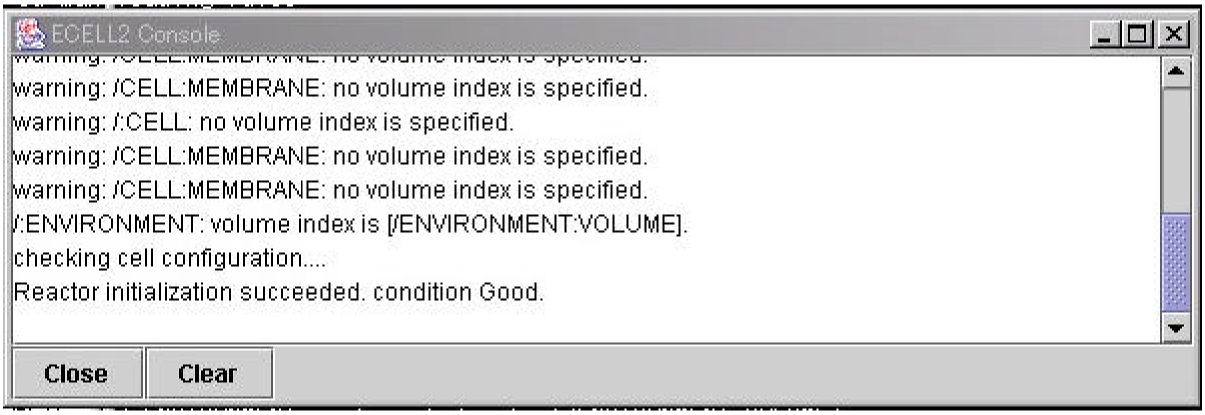
Step interval: 0.0010

Update interval: 100

Figure 30. PreferenceWindow

PreferenceWindow will be explained next. To display this window, please select [PreferenceWindow] from the [Windows] menu in the Control Panel. This window is provided to control the width of a simulation time step(Unit:sec.), and how often the Tracer window is updated. If the time step is short, the accuracy of computation increases at the expense of increases in the time necessary for simulation. On the other hand, if the time step is long, the time necessary for simulation decreases but the accuracy of computation also decreases which therefore makes its results less reliable. In the initial settings, a computation step is set at 0.001 seconds, whereas the frequency of window updating is set at once per 100 steps.

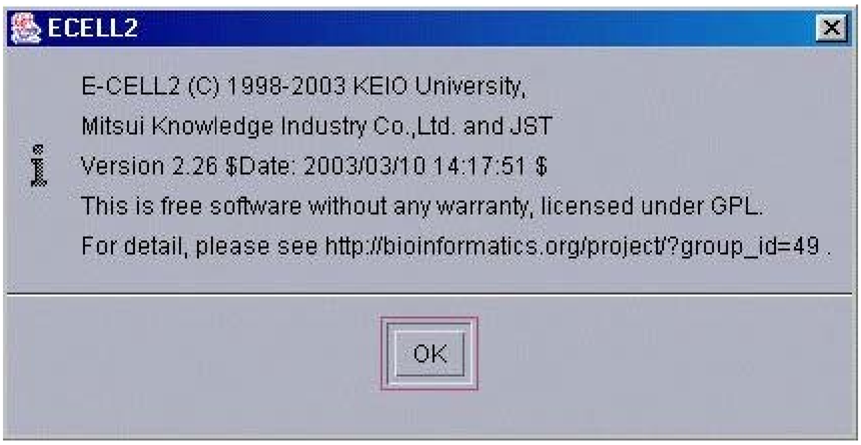
Because this window is a modal dialog, other windows may not be operated until "Ok" or "CANCEL" is pressed.



```
warning: /CELL:MEMBRANE: no volume index is specified.
warning: /CELL:MEMBRANE: no volume index is specified.
warning: /:CELL: no volume index is specified.
warning: /CELL:MEMBRANE: no volume index is specified.
warning: /CELL:MEMBRANE: no volume index is specified.
/:ENVIRONMENT: volume index is [/ENVIRONMENT:VOLUME].
checking cell configuration....
Reactor initialization succeeded. condition Good.
```

Figure 31. MessageWindow

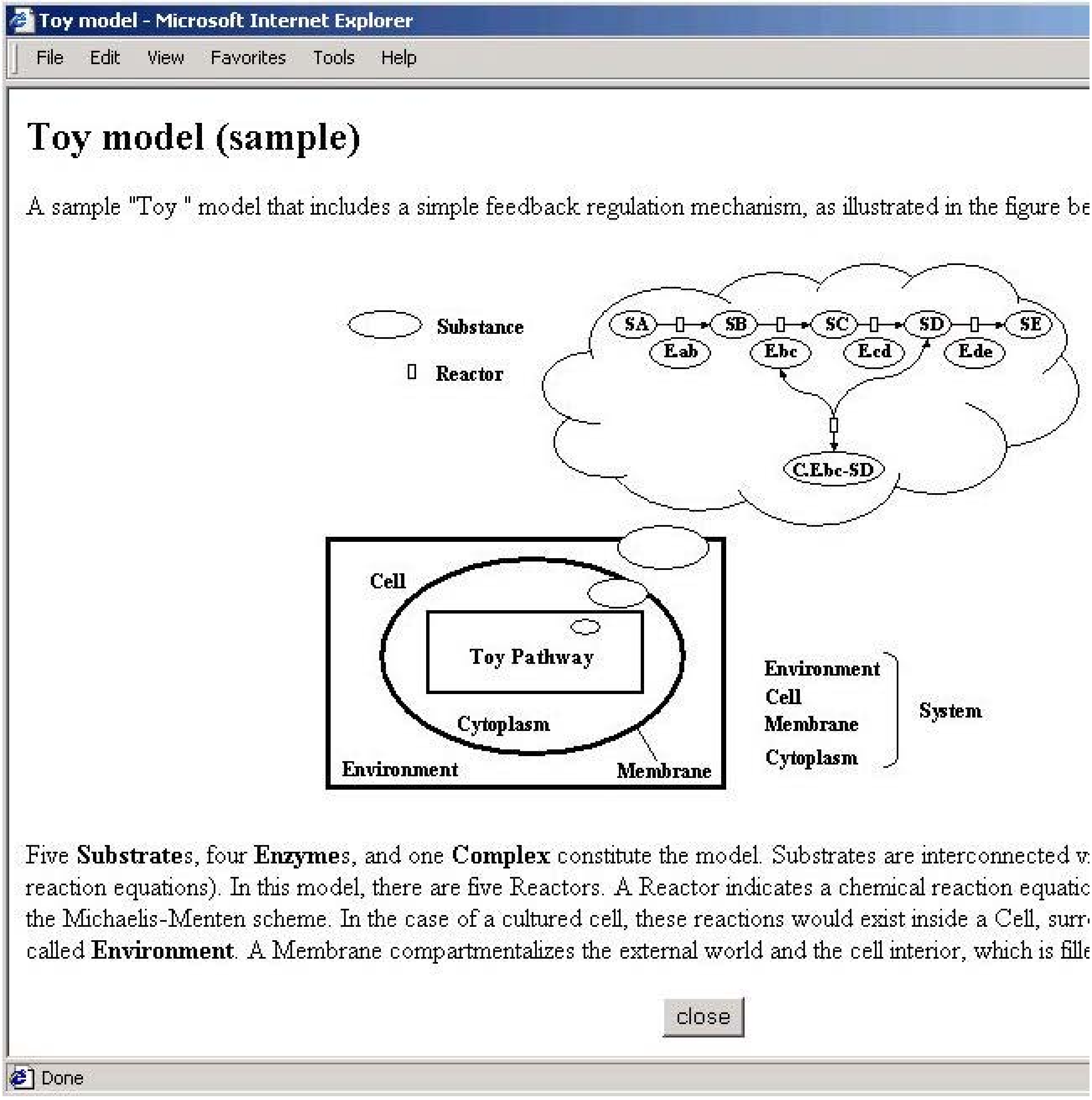
In the MessageWindow, messages from E-CELL2 will be displayed. To close this window, please press the "Close" button. If the "Clear" button is pressed, the messages displayed in the window will be cleared.



E-CELL2 (C) 1998-2003 KEIO University,
Mitsui Knowledge Industry Co.,Ltd. and JST
Version 2.26 $Date: 2003/03/10 14:17:51 $
This is free software without any warranty, licensed under GPL.
For detail, please see http://bioinformatics.org/project/?group_id=49 .

Figure 32. Version Information window

[Version Information] can be selected from the [Help] menu in the Control Panel. The version information for E-CELL2 will be displayed here.



Toy model - Microsoft Internet Explorer
File Edit View Favorites Tools Help

Toy model (sample)

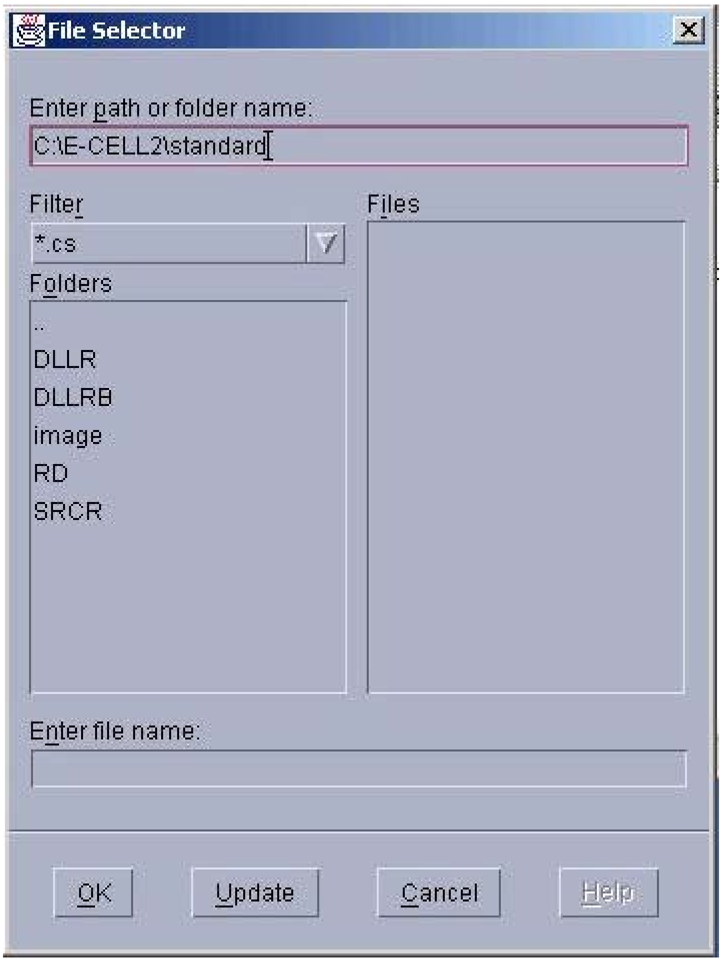
A sample "Toy " model that includes a simple feedback regulation mechanism, as illustrated in the figure be

Five **Substrates**, four **Enzymes**, and one **Complex** constitute the model. Substrates are interconnected v reaction equations). In this model, there are five Reactors. A Reactor indicates a chemical reaction equatic the Michaelis-Menten scheme. In the case of a cultured cell, these reactions would exist inside a Cell, surr called **Environment**. A Membrane compartmentalizes the external world and the cell interior, which is fille

Figure 33. Model Information (Model Information of the Toy Model)

Also, if [Model Information] is selected from the [Help] menu in the Control Panel, a description of the models will be displayed on a browser.

At the end of this section, the methods for saving simulation states will be explained. To save the progress of simulation, please press the "Stop" button in the Control Panel, and then select [Save Cell State] from the [File] menu.



Enter path or folder name: C:\E-CELL2\standard

Filter: *.cs

Figure 34. Saving a Cell State File

Files describing the state of the whole cell model have the ".cs" extension. Please save it under a suitable directory. Files in this format can be loaded by selecting [Load Cell State] from the [File] menu in the Control Panel. It is thus possible for simulation to be resumed from a saved state.

Automation using script files

In large-scale simulation, the manual operation of its various tasks is a troublesome process. In E-CELL, simulation can be automated by using a mechanism called "script", which loads the commands described in the script into E-CELL. Script files are usually distinguished by the ".ecs" extension.

The directions on how to write the script files will be explained briefly here. The format differences between the script files of E-CELL1 and E-CELL2 will also be explained.

First, please take a look at the following file(The numbers in the beginning of each line are line numbers).

```
01  TmpDir C:\E-CELL2\temp
02  LoadRule sample.eri
03
04  NewInterface Tracer A_B_C
05    AddTrace    Substance:/CELL/CYTOPLASM:SA
06    AddTrace    Substance:/CELL/CYTOPLASM:SB
07    AddTrace    Substance:/CELL/CYTOPLASM:SC
08    SaveAt      500
```

```
09
10  UpdateInterval 100
11
12  Run 501
13  SaveCellState sample-after500s.cs
14  Stop
```

```
  Line 1: Designates the directory for saving temporary files.
TmpDir is a directory for saving temporary files.
If the command does not exist, the "temp" directory under E-CELL2's installation directory is usec

  Line 2: Designates the Rule Intermediate File (".eri" file) to be loaded.

  Line 4-8: Information regarding a Tracer
  Line 4-7: "AddTrace" is declared only inside a Tracer.

         The Substances or Reactors written in the arguments are added for display by the Tracer.

         The arguments must include not only the ID, but also the System and Type that it belongs
  Line 8: The Substances displayed in this Tracer are saved after 500 seconds in E-CELL time.

          It is declared only inside a Tracer.
  Line 9: A blank line is necessary after a NewInterface.
  Line 10: Updates the Tracer every 100 steps.
  Line 12: Executes simulation for 501 seconds.
 Line 13: Saves the state of the cell model to the file "sample-after500s.cs".

 Line 14: Stops the simulation.
```

Please save the file above with a suitable name such as "toy.ecs". (However, please do not use non-US ASCII characters). Please save it under the directory "C:¥E-CELL2¥standard", which is the directory that includes "sample.eri". If E-CELL2 has been installed under a directory other than this, please choose the proper target directory accordingly.

Please start the standard reactor-version of E-CELL2. After it starts, please select [Load Script] from the [File] menu in the Control Panel, and load the .ecs file that has just been created.

The "Logger" mechanism, which has been newly implemented for E-CELL2, will also be introduced here. Logger is a mechanism that continuously records the states of a cell. It is more advanced than saving the cell states in a snapshot manner. It can be used by adding the following descriptions to the script file.

```
NewInterface Logger test-log
  AddTrace    Substance:/CELL/CYTOPLASM:SA
  AddTrace    Substance:/CELL/CYTOPLASM:SB
  AddTrace    Substance:/CELL/CYTOPLASM:SC
  AddTrace    Substance:/CELL/CYTOPLASM:SD
  AddTrace    Substance:/CELL/CYTOPLASM:E.bc
  AddTrace    Substance:/CELL/CYTOPLASM:C.Ebc-D
  AddTrace    Reactor:/CELL/CYTOPLASM:E.ab-0
  AddTrace    Reactor:/CELL/CYTOPLASM:E.bc-0
  AddTrace    Reactor:/CELL/CYTOPLASM:E.cd-0
```

Please add the descriptions regarding the Logger after the 9th line in the ".ecs" file that was first created, and load it to E-CELL2.

Loggers have the following features.

1. A directory called "Logger" will be created under the current directory. Here, the files for each Substance and Reactor will be recorded in a format similar to the one created when

using the "SaveAt" function of Tracers.
2. Because Loggers continuously record data throughout simulation execution, the recorded data will remain regardless of crashes in the E-CELL2 system.
3. Because Loggers do not use virtual memory, they record data faster than Tracers.

After simulation stops, please try opening the .ecd file located under the Logger directory using tools such as notepads. The file names correspond to each Substance and Reactor. Information on the "time", "instantaneous value", "average value", "maximum value", "minimum value", "instantaneous concentration", "average concentration", "maximum concentration", and "minimum concentration" are listed.

The procedures for carrying out simulation have been explained so far in this chapter. Please refer to the chapters that follow Chapter 3 for details regarding each item.

Chapter 3 Basic Operation

1 Introduction

In this chapter, the basic operation of the E-CELL2 System will be explained. It will be assumed that users have already mastered the overall operating procedures for E-CELL2, which have been explained in "Chapter 2 Tutorial". Please refer to Chapter 2 if necessary. Two modes, a GUI mode and a batch mode, exist in E-CELL2. In this chapter, most of the explanations will be on the former mode(GUI mode). In "Section 2 Basic Operation", the startup and shutdown of the E-CELL2 System in GUI mode will be explained briefly first. Operations in batch mode will be explained in "Section 2.9 E-CELL2 Batch Mode and Loggers". Loggers, newly implemented for the E-CELL2 System, are mechanisms that continuously save simulation results in real-time. Details of the operation procedures will be explained in "Section 3 Interfaces". Script files, which improve the operation of E-CELL2, will be explained in detail in "Section 4 E-CELL2 Script Files".

The E-CELL2 System requires data files called rule files for running simulations. Rule files will be explained in Chapter 4. Also, to create new models, new **Reactors** need to be constructed. **Reactors** will be explained in detail in Chapter 5. After reading through these chapters, one should be able conduct basic modeling using the E-CELL2 System. Please note that the guaranteed accuracy of calculation for GUI mode and batch mode is different; it is 64 bits and 80 bits respectively.

2 Basic Operation

2.1 Starting the E-CELL2 System

To start the E-CELL2 System, please click on the "Standard" shortcut in the desktop, or select [Start]-[Program]-[E-CELL2]-[standard]. E-CELL2 for the standard reactor model will start.

After the E-CELL2 System starts, the E-CELL2 ControlPanel will first displayed first.

If a script file called "default.ecs" is located under the E-CELL2 startup directory (The model directories under C¥:E-CELL2, in a standard install), the file will automatically be loaded. However, please note that if the "LoadRule" command does not exist in the script file, while the

"NewInterface" command or the "Run" command does exist, an error will occur and the file will be non-executable. The details regarding script files are explained in Section 4 of this chapter.

It is also possible to load settings with the file by adding options when directly starting ECELL2.BAT from the command line. The options are shown below.

-a *classname*	Designates the default Accumulator class (Refer to Chapter 4).
-c	If this option is designated, and if a cell state file is loaded when the simulation elapsed time is 0, the simulation elapsed time recorded in the cell state file will be set to the present simulation progress time. In all other cases, the simulation progress time will not change even if a cell state file is loaded. For details regarding cell state files, please refer to Section 2.7 in this chapter.
-d	Starts in debug mode.
-f *filename*.ecs	Loads the designated scriptfile.
-h	Displays a list of options.
-t *directory_name*	Designates the target directory for writing the temporary files.
-u *integer_value*	Designates the number of times Postern Reactors are called at the time of initialization. For details, please refer to the explanation of Postern Reactors in Chapter 5.

2.2 Loading Rule Files

Rule(eri) files, which are data files describing information that is necessary for the E-CELL2 System to run simulations, **must be loaded prior to executing simulation**. To load Rule(eri) files, please select [Load Rule] from the [File] menu in the ControlPanel, and then select the appropriate Rule(eri)file from the newly displayed FileSelector window.

Please note that if Rule files are loaded more than 2 times, E-CELL2 will display an error message.

2.3 Loading Script Files

If a file called "default.ecs" is located under the E-CELL2 startup directory, the file will automatically be loaded at the time of startup. Please note that if an error occurs during the loading process, there is a possibility that E-CELL2 will abnormally shut down.

Script(ecs)files are files for controlling and automating simulation and interfaces. Script(ecs)files need not be loaded. To load a Script file, please select [Load Script] from the [File] menu in the ControlPanel, and then select the appropriate Script(ecs)file from the newly displayed FileSelector window.

Please refer to the startup option "-c".

2.4 Executing Simulations

If the "Start" button in the ControlPanel is clicked after loading a rule file, the time progress within the simulation system will be displayed in the time counter. To stop simulation, please press the "Stop" button. Also, if the "Step" button next to the "Stop" button is clicked, a single time step-worth of simulation will be executed.

2.5 Changing Step Sizes

A step refers to a single cycle that first calculates the rates of reactions, and then updates the quantities of the affected **Substance**s. At default, the size of an integration time step is 1/1000 seconds. To change the size of this time step, please select [PreferenceWindow] from the [Windows] menu in the ControlPanel, and then input a numerical value in the text box that will be displayed. This can also be designated using ecs files.

If the step size is increased, the time necessary for simulation will be shorter, but will result in an increase in error. On the other hand, if the step size is decreased, the time necessary for simulation will be longer, but will result in a decrease in error.

2.6 Displaying Simulation Results

In the default settings, user interfaces are updated once every 100 integration calculations. This interval can be changed in the ecs file.

2.6.1 Using Tracers

Tracers display the simulation results as a 2-dimensional graph. In the vertical axis, the changes in the quantity/concentration of **Substance**s, or the changes in the activity of **Reactors** are displayed. In the horizontal axis on the other hand, the time is displayed. To display a new TracerWindow, please select [Tracer] from the [NewInterface] in the ControlPanel. To display a new **Substance** or **Reactor** in the Tracer, please click on the "Add" button at the bottom. An EntrySelector will then be displayed. Please select the appropriate **Substance** or **Reactor**.

Please note that what **Substance**s describe does not necessarily refer to "molecules" of strict chemical terminology. It may be voluntary particles such as ions and radicals. However, in this manual we will use the term "molecule" for convenience.

The selected **Substance**s and **Reactor**s will be displayed with the colored square checkboxes in the Traces display area. If simulation starts in this state, the state of the Substance or Reactor will be represented by a graph of the same color as the square next to the checkbox.

2.6.2 Using SubstanceWindows

To display or change the quantities of substances, SubstanceWindows are used. To display a SubstanceWindow, please select [SubstanceWindow] from the [NewInterface] menu in the ControlPanel. An EntrySelector will then be displayed. Please select the appropriate **Substance**. If the "OK" button is clicked, a SubstanceWindow will be displayed.

In the SubstanceWindow, the ID and the substance name are displayed from the top. Below this, the number of molecules is displayed on the left, and the concentration on the right. If simulation is in a stopped state, these two values can be directly changed. After clicking on the displayed values to be changed and inputting new numbers, please press the return key to actually change the values. The number of molecules can be changed also by using the up and down buttons on the bottom.

2.6.3 Using ReactorWindows

To observe the rates of reactions, and the substrates and products that are involved, ReactorWindows are used. To display a ReactorWindow, please select [ReactorWindow] from the

[NewInterface] menu in the ControlPanel. An EntrySelector will then be displayed. Please select the appropriate **Reactor**. If the "OK" button is clicked, a ReactorWindow will be displayed.

In the ReactorWindow, the **Reactor** class, entry name, and the name of the reaction will be displayed from the top. Below these, the activity per second will be displayed.

This is in a simple display mode. To observe the substrates and products involved in the reaction, please click the "ShowList" button on the top left. A new display area will then be added to the bottom. Starting from the left, the list of substrates, products, enzymes, and effectors will be displayed.

2.7 Saving Simulation Results

To save the simulation results displayed in Tracers, please click the "Save" button on the bottom left corner of the TracerWindow. A directory called "Data" will be created under the directory that the E-CELL2 System has been executed. In addition, subdirectories will be created according to the structure of the **System**. Also, data files in ecd format will be created for each Substance or Reactor. A data file in **ecd** format is usually composed of a header, more than 5 "-"(dashes) lined up that represent the end of the header, the data body, and "///"(3 slashes) that represent the end of the file. What has been improved compared to E-CELL1 is that the time, instantaneous quantity, average quantity, maximum quantity, minimum quantity, instantaneous concentration, average concentration, maximum concentration, and minimum concentration can now be outputted in E-CELL2.

Header of an **ecd** file

key word	**value**
DATA	name of data
SIZE	number of columns and rows
LABEL	legend of each column
NOTE	optional comments

Example of an **ecd** file for E-CELL2

```
DATA: /CELL/CYTOPLASM:E.ab-0:Quantity
DATA: /CELL/CYTOPLASM:P:Quantity
SIZE: 5 102
LABEL: time current-quantity mean-quantity max-quantity min-quantity
NOTE:

----------------------
0       0       0       0       0
0.10000000000000064     624     314.66  624     6
0.20000000000000016     1247    938.65  1247    630
0.30000000000000192     1871    1562.5  1871    1254
0.40000000000000032     2495    2186.19 2495    1877
0.50000000000000032     3118    2809.73999999999968     3118    2501
0.60000000000000384     3742    3433.15 3742    3125
0.70000000000000512     4365    4056.4  4365    3748
0.80000000000000064     4988    4679.5  4988    4371
0.90000000000000064     5611    5302.47999999999936     5611    4994
1.00000000000000064     6234    5925.27 6234    5617
1.0999999999999896      6856    6547.95 6856    6240
1.19999999999997856     7479    7170.48 7479    6862
1.29999999999996768     8101    7792.86 8101    7485
```

```
(Omission)
///
```

To suspend simulation and save the states of the substances for the whole model, please select [Save Cell State] from the [File] menu in the ControlPanel. Please save the file with an appropriate name, with the ".cs" extension. The states of the models saved here can be loaded by selecting [Load Cell State] from the same [File] menu. Simulations can thus be resumed from the states that have been saved.

2.8 Shutting Down the E-CELL2 System

If [Quit] is selected from the [File] menu in the ControlPanel, a confirmation dialog will be displayed. If the "OK" button is clicked, the E-CELL2 System will shut down.

2.9 E-CELL2 Batch Mode and Loggers

E-CELL2 provides a batch mode that is executable from command lines. Please start the command prompt(MS-DOS prompt), and then move to the directory that E-CELL2 has been install to. In a standard install, "C:¥E-CELL2¥standard" is the location of the startup batch of E-CELL2 for the standard reactor model. The E-CELL2 startup batches for each model are located in "C:¥E-CELL¥model name".

If "ECELL2BB.BAT -h" is typed, a list of the options will be displayed. The options are the same as in GUI mode(Please refer to Section 2.1). ECELL2BB.BAT is the batch file for MS-DOS, and describes the PATH of the directory that the dynamically linked Reactor libraries(DLL) are housed in. In PATH, the designation of a DLL-housing folder(e.g. DLLRB) for batch mode is necessary. To execute simulation in batch mode, it is necessary to include the "-f" option when loading the script file(ecs). If simply typed "ECELL2BB.BAT", it will load "default.ecs". For details on script files, please refer to Section 4 in this chapter.

```
ECELL2BB.BAT -f filename.ecs
```

The following are the contents of ECELL2BB.BAT.

```
SET PATHSAV=%PATH%
PATH=../BIN/WIN-BCC;DLLRB;%PATH%
ecell2b.exe -r DLLRB %1 %2 %3 %4 %5 %6 %7 %8 %9
PATH=%PATHSAV%
```

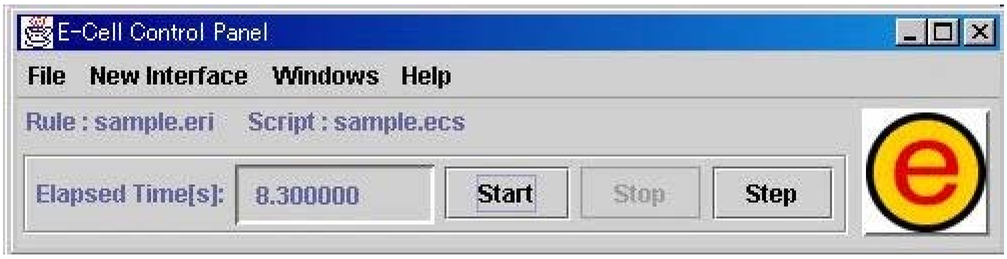
If the "Logger" mechanism implemented from E-CELL2 is used, it is possible to record the cell state continuously during simulation. This function was implemented for use in batch mode. For details on its usage, please refer to "Section 4.5 List of Script File Commands".

3 Interfaces

3.1 ControlPanel

The ControlPanel, which is the main interface for the E-CELL2 System, is used for operation of tasks such as reading and writing files, operating on other interfaces, and executing and stopping simulation.

If a rule file is loaded, the name of the loaded rule file will be displayed after the "Rule:" under the menu. If a script file is loaded, "Script:" and the script file name will be displayed following the rule file.



Rule : sample.eri Script : sample.ecs

Elapsed Time[s]: 8.300000

Figure 1. ControlPanel

Pull-down menu list

File	Load Rule	Loads a rule file.
	Load Script	Loads a script file.
	Load Cell State	Loads a Cell State.
	Save Cell State	Saves the Cell State.
	Quit	Shuts down the E-Cell2 System.
NewInterface	Tracer	Displays a TracerWindow.
	Substance Window	Displays a SubstanceWindow.
	Reactor Window	Displays a ReactorWindow.
Windows	Message Window	Displays a MessageWindow.
	Preference Window	Displays a PreferenceWindow.

Start Button
Executes simulation.

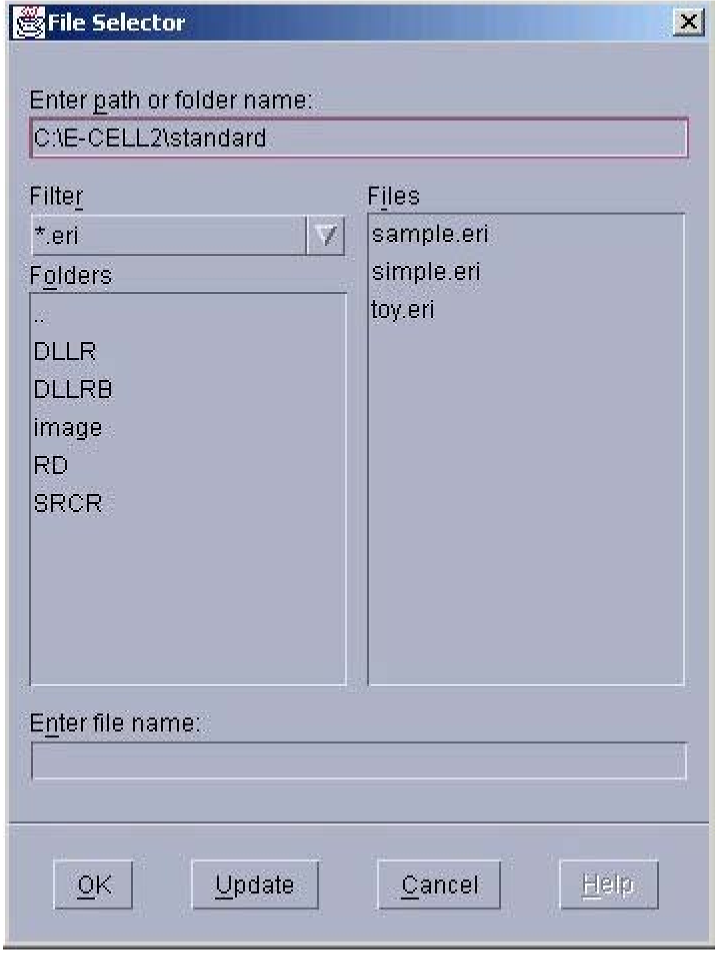
Stop Button
Stops simulation.

Step Button
Executes a step of simulation.

Time Counter
Displays the elapsed time inside the simulation system, in seconds.

3.2 FileSelector

FileSelector windows are displayed for selecting files when reading or writing files. Files can usually be selected by first clicking the directory name until the file display area becomes the appropriate directory, and then by clicking the appropriate file and the "OK" button on the bottom. To write in a new file, please click on the file name display area, then directly input the file name and click on the "OK" button.



File Selector

Enter path or folder name: C:\E-CELL2\standard

Filter: *.eri

Folders:
..
DLLR
DLLRB
image
RD
SRCR

Files:
sample.eri
simple.eri
toy.eri

Enter file name:

Figure 2. FileSelector

File Display Area

Displays a list of the files and directories that are currently selectable. By clicking on the displayed file name, the file name in the file display area on the bottom becomes that of the clicked one. By clicking on the "OK" button, the file can be selected. Files can also be selected by double-clicking the file names. If a displayed directory name is clicked, the directory displayed in the file display area on the top becomes that of the clicked one.

File Name Display Area

Displays the file name currently selected. By clicking on the displayed area, the file name can be directly inputted. If the return key or the "OK" button is clicked after input, The file will be selected.

File Type List

Displays the regular expression for restricting the files that are displayed in the file display area, using filters. In the default settings, only the files with the appropriate extensions such as *.ecs. will be displayed. To change filters, please click on the file type list and select a different file type.

OK Button

If clicked, the file currently displayed in the file name display area will be selected.

Cancel Button

If clicked, file selection will be canceled and the window will close.

3.3 Tracer

Tracers display the quantity of **Substance**s or the activity of **Reactor**s, using 2-dimensional graphs. Up to 8 series of data can be displayed in a single Tracer window.

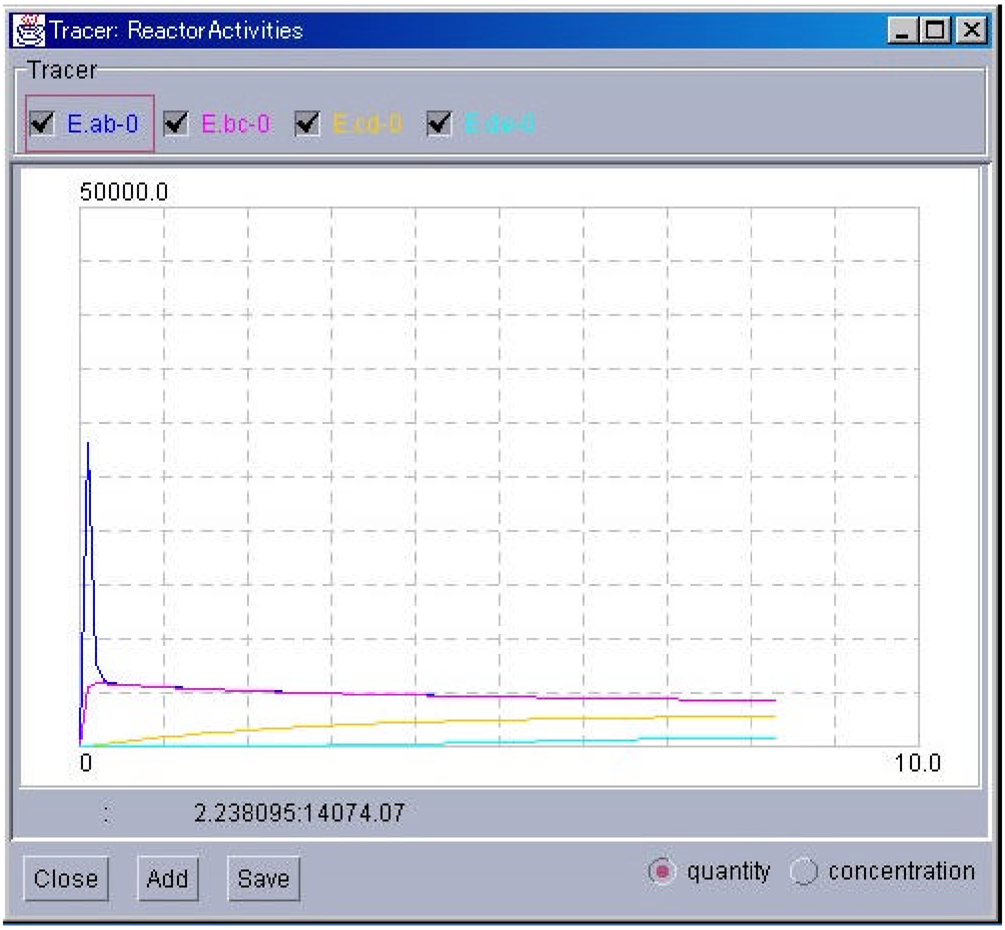


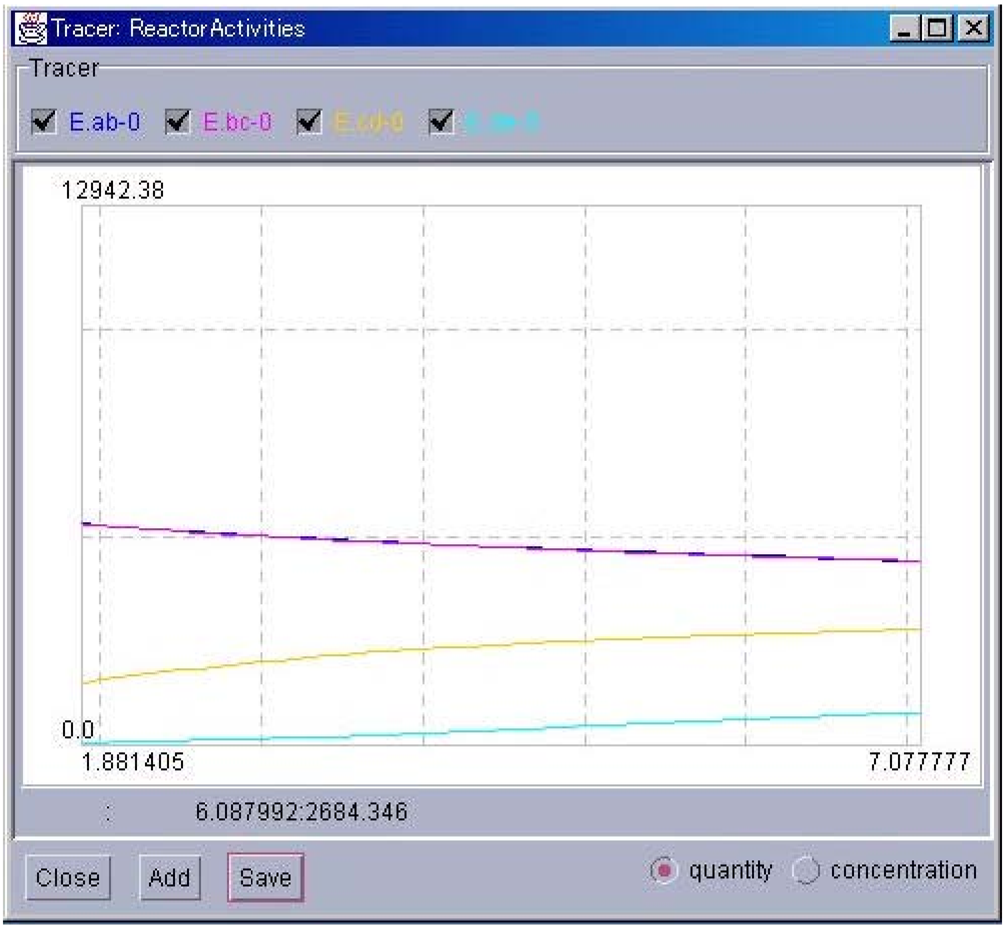
Figure 3. Tracer

Tracer

A list of the **Substance**s and **Reactor**s displayed in the Tracer will be displayed. The color of the radio button on the left of the name, corresponds to the color of the graph on the bottom. Switching between the display/non display of the graphs is possible by left-clicking the button. By right-clicking the button, a SubstanceWindow will be displayed if it is a **Substance**, and a ReactorWindow if it is a **Reactor**. In a single Tracer, up to 8 Substances or Reactors can be displayed.

Graph Display Area

Simulation results will be displayed as graphs. The vertical axis represents the number of molecules(or concentration), whereas the horizontal axis represents time. If left-clicked in the graph area, a gray cross with its vertex at the point that was clicked will be displayed. If there are multiple TracerWindows being displayed, the gray crosses will be displayed in the same coordinates for all of them. Please note that there are a few points to be careful about regarding synchronous cross cursors. For example, if the scales of 2 windows differ by a lot, even if a cross is displayed in one of the Tracers, the coordinates of the cross may be near 0 in the other Tracer and therefore may not be displayed. In this case, the coordinates on the bottom of the window will also not be displayed. However, if the area is enlarged, the cross will appear. Also, if the Tracer in the enlarged state is clicked, crosses will appear in the corresponding points. To clear the gray cross, please click anywhere outside the graph display area. Also, regions can be enlarged by dragging and creating rectangles with the right mouse button. To cancel the enlarged state and return to the default state, please double-click on the right mouse button.



● quantity ○ concentration

Figure 4. Tracer(Expanded display)

Coordinates Display Area

The coordinates of the gray cross that is displayed when the graph is left-clicked will be displayed on the left. On the other hand, the coordinates of the mouse cursor will be displayed on the right.

Close Button

Closes the TracerWindow.

Add Button

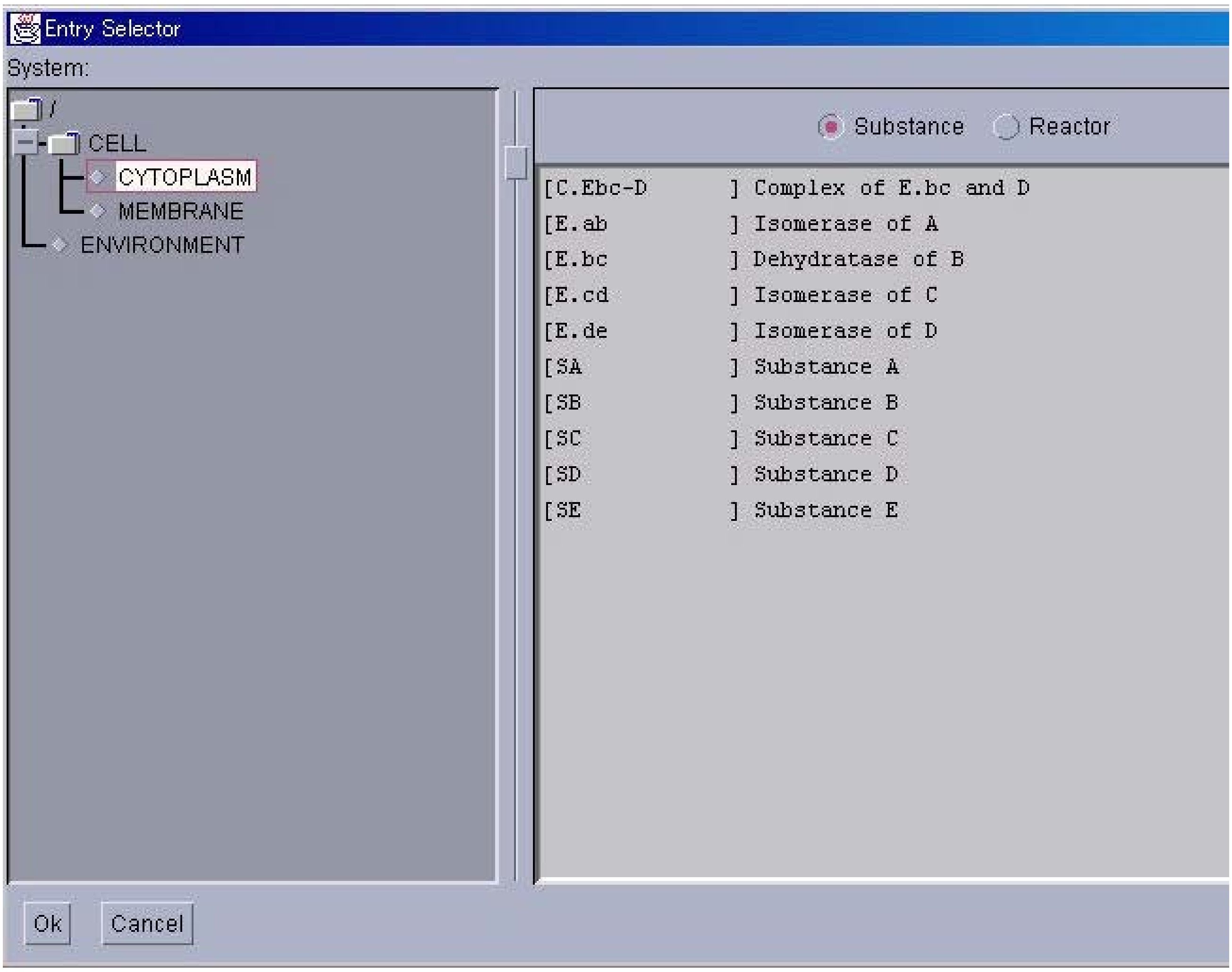
Displays a new EntrySelector, and then displays the newly selected Substance or Reactor in the Tracer.

Save Button

Saves the time-series data that is displayed in the Tracer, into a file. The saved data is composed of 5 columns; "time", "instantaneous quantity", "average quantity", "maximum quantity", and "minimum quantity".

3.4 EntrySelector

EntrySelectors are used when selecting **Substance**s or **Reactor**s. Please click on the folder icon in the **System** selection dialog on the left. If the appropriate **System** has been displayed, please select the appropriate items from the Substance and Reactor selection dialog, and then click on the "OK" button on the bottom. By using the switch button on the top right, what is selected by the Tracers(**Substance** or **Reactor**), can be selected beforehand.



```
[C.Ebc-D       ] Complex of E.bc and D
[E.ab          ] Isomerase of A
[E.bc          ] Dehydratase of B
[E.cd          ] Isomerase of C
[E.de          ] Isomerase of D
[SA            ] Substance A
[SB            ] Substance B
[SC            ] Substance C
[SD            ] Substance D
[SE            ] Substance E
```

Figure 5. EntrySelector

Switch Button

Used for switching between **Substance** and **Reactor** when called by Tracers.

System Selection Dialog

The **System** in a hierarchal structure is displayed. To display the lower levels, please click on the folder icon.

Substance, **Reactor** Selection Dialog

Substances or **Reactor**s will be displayed. If dragged, multiple selections are possible. Also, if clicked while the Ctrl key is pressed, separately located items can be selected.

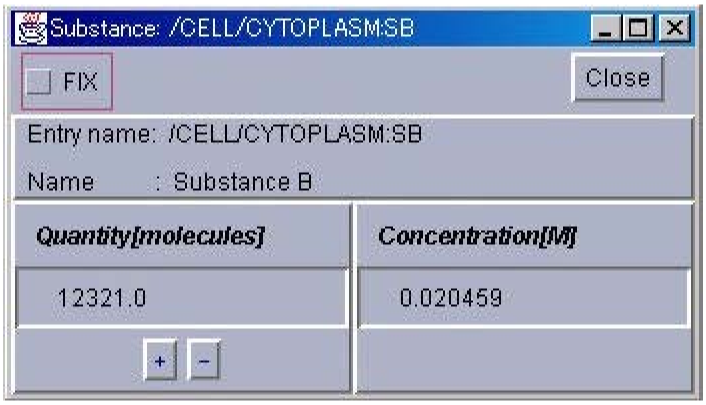
OK Button

Finalizes the selection.

Cancel Button

Cancels the selection, and closes the window.

3.5 SubstanceWindow



Entry name: /CELL/CYTOPLASM:SB

Name : Substance B

Quantity[molecules]	Concentration[M]
12321.0	0.020459

Figure 6. SubstanceWindow

FIX Button

If this button is checked, the quantity will not change during simulation.

Quantity Display Area

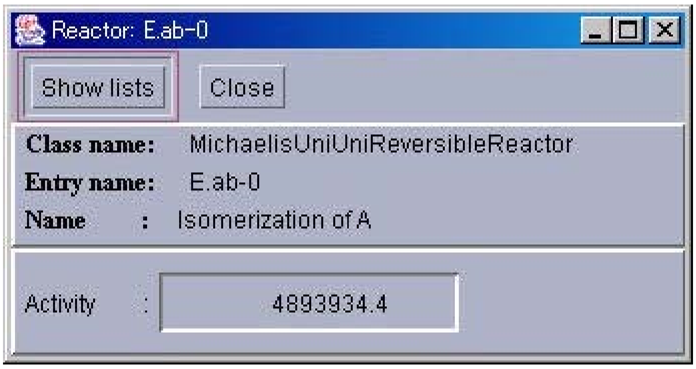
Displays the quantity(number of molecules) of the **Substance**. When simulation is in a stopped state, its numerical value can be changed by clicking this area.

Concentration Display Area

Displays the concentration of the **Substance**. When simulation is in a stopped state, its numerical value can be changed by clicking this area.

3.6 ReactorWindow

Normal Display Mode



Class name: MichaelisUniUniReversibleReactor

Entry name: E.ab-0

Name : Isomerization of A

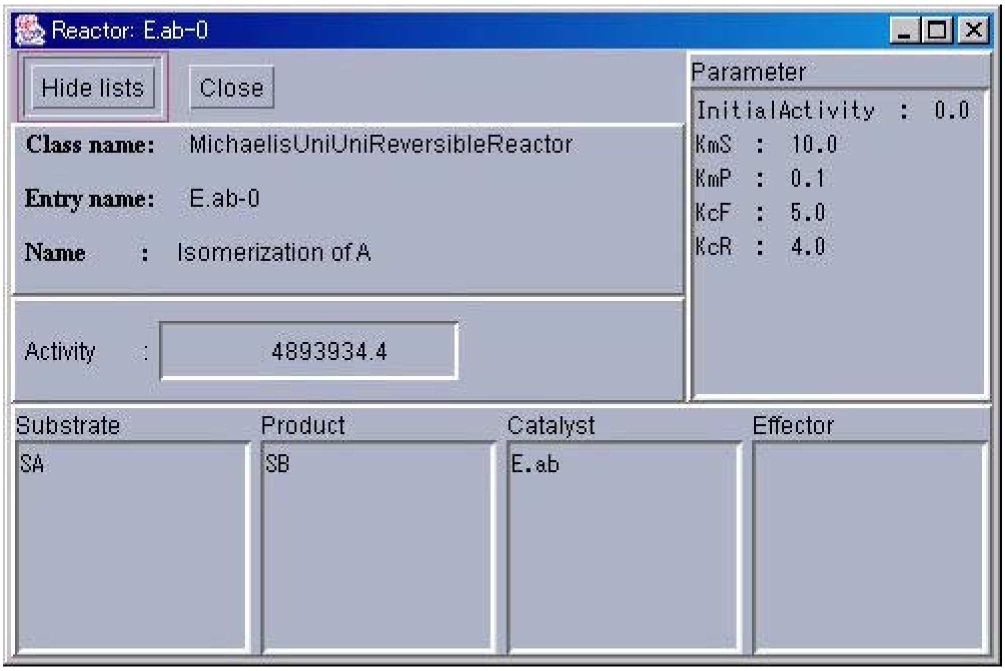
Activity : 4893934.4

Figure 7. ReactorWindow(Normal Display)

Detailed Display Mode Switch Button

Switches display to detailed display.

Detailed Display Mode



Class name: MichaelisUniUniReversibleReactor

Entry name: E.ab-0

Name : Isomerization of A

Activity : 4893934.4

Parameter

```
InitialActivity  :  0.0
KmS  :  10.0
KmP  :  0.1
KcF  :  5.0
KcR  :  4.0
```

Substrate	Product	Catalyst	Effector
SA	SB	E.ab	

Figure 8. ReactorWindow(Detailed Display)

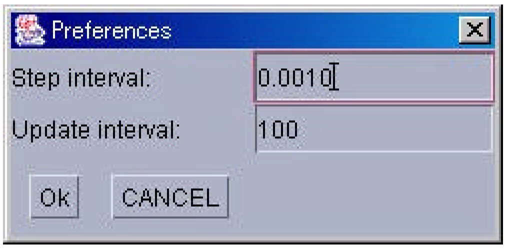
Normal Display Mode Switch Button

Switches display to normal display.

Reactant Display Area

Displays a list of the parameters on the right, and a list of the substrates, products, catalysts, and effectors on the bottom(from the left).

3.7 PreferenceWindow



Step interval: 0.0010

Update interval: 100

Figure 9. PreferenceWindow

Step Interval

Changes the interval of a computation step.

Window Update Interval

Changes the interval of Tracer updating.

4 E-CELL2 Script Files

4.1 What are Script Files?

Script files are files for batch-processing the startup and shutdown of the E-CELL2 System, the control of simulation such as the recording of its results, and the control of its interfaces that display results.

Script files are text files, and can therefore be created using normal text editors.

4.2 Script File Syntax

Script files are loaded one line at a time, from the top. In script files, one command is described per line, except for the NewInterface command. The insertion of a new line in a single command is not allowed. To declare a command inside a NewInterface such as AddTrace, please describe it directly under the NewInterface command that it belongs to, inserting a space or a tab at the beginning of each line.

The text between the "#" symbol until the end of the line are comments that will be ignored.

4.3 Sample Script File

A sample script file is shown below.

```
TmpDir C:\E-CELL2\temp
LoadRule sample.eri

NewInterface Tracer trace1
  AddTrace Substance:/CELL/CYTOPLASM:A
  AddTrace Reactor:/CELL/CYTOPLASM:R1
  SaveAt 45

NewInterface SubstanceWindow test-sw
  Substance /CELL/CYTOPLASM:A

Run 500
Stop
```

There have been a few changes made from the E-CELL1 System. What indicates the path of the Reactors, is included in the startup batch file ECELL2.BAT or ECELL2BB.BAT. It has been made to describe the PATH when loading. Therefore, the "ReactorPath" command cannot be used.

TmpDir is a directory for saving temporary files. If the command does not exist, the "temp" directory under E-CELL2's installation directory is used. If the directory designated for TmpDir does not exist, the E-CELL2 System will terminate when executing simulation.

4.4 Explanation of Sample Script File

The first line designates the target location for writing the temporary files, whereas the second line designates the rule file to be loaded.

Lines 4 to 7 display a TracerWindow called Trace1, and are carry out a few settings. Line 5 designates the addition of a Substance called A to the Tracer, whereas line 6 designates the addition of a Reactor called R1. Line 7 designates the saving of the state of the Tracer after 45 seconds.

Lines 9 and 10 open a SubstanceWindow.

Line 12 designates the execution of simulation for 500 seconds, whereas line 13 designates the termination of the simulation.

4.5 List of Script File Commands

Command	Contents
Accumulator *class name*	Designates the default Accumulator class. For details, please refer to Chapter 4 Section 2.4.. Has to be described before LoadRule.
TmpDir *directory*	Writes the temporary file generated during simulation, to the path designated in the argument. ../temp is designated at default. Has to be described before LoadRule.
LoadRule *filename*	Loads the rule file designated in the argument.
StepInterval *second*	Designates the integration time interval in seconds. The default value is 0.001. Has to be described after LoadRule and before Run.
UpdateInterval *count*	Designates the user interface update interval in natural numbers. The default value is 100. The user interface is therefore updated once every 100 integration steps. Has to be described after LoadRule and before Run.
SaveCellState *file_name*	Saves the simulation state in the designated file. Has to be described after Run.
LoadCellState *file_name*	Loads the simulation state that is saved in the designated file. Has to be describe after Run.
Run *seconds*	Executes simulation for the duration(time within the simulation system) designated in the argument.
Stop	Stops simulation. Has no arguments.
Exit	Shuts down E-CELL2. Has no arguments.
NewInterface *interface, name*	A command for creating new interfaces. Requires 2 arguments; the type of the interface, and the name of the interface.

Commands Declared within NewInterface

Command	Contents
Geometry *(X1)x(Y1)+/-(X2)+/-(Y2)*	Designates the size and the display coordinates of the interface. However, because it only takes one argument, it is described like the following. *(X1)x(Y1)+/-(X2)+/-(Y2)* However, *(X1)* and *(Y1)* denote the width and height respectively. Also, *(X2)* and *(Y2)* denote the x coordinate and y coordinate for the top left corner of the window. If it is +, the top left becomes the base. If it is -, the bottom right becomes the base. The difference between E-CELL1 is that if the location of the window is designated in the above rule, but does not fit in the Windows screen, the location of the screen will automatically be fixed so that it does fit in the Windows screen.

Commands for Each Interface

- Tracer

Command	Contents
AddTrace *FQPN*	Adds the Substance or Reactor written in the argument, to the Tracer for display. The argument has to include not only the ID, but also the System and Type that it belongs to, and uses FQPN.
SaveAt *seconds directory*	Saves the state of the Tracer at a certain time. Has the same effect as clicking the Save button in the TracerWindow. In the first argument, the time in which the state will be recorded is provided, in seconds. In the second argument, the directory for saving the data is designated. If omitted, becomes "./Data".

- SubstanceWindow

Command	Contents
Substance *FQEN*	Is declared only within the SubstanceWindow. Displays the Substance in the argument, in the SubstanceWindow. The argument needs to not only include the Substance ID, but also the System that it belongs to, and uses FQEN.

- ReactorWindow

Command	Contents
Reactor *FQEN*	Is declared only within the ReactorWindow. Displays the Reactor designated in the argument, in the ReactorWindow. Similar to the Substance command, the argument has to not only include the Reactor ID, but also the System that it belongs to, and uses FQEN.

- Logger

Command	Contents
AddTrace *FQPN*	Records the time-series data of the Substance or Reactor written in the argument. The argument has to include not only the ID, but also the System and Type that it belongs to, and uses FQPN. A directory called Logger is created under the model directory that E-CELL2 has been executed. Files for each Substance or Reactor will be recorded under the Logger directory, in a format similar to the one created using the SaveAt function of Tracers. Because file recording is done continuously during execution, records will remain even if E-CELL2 happens to crash. In this file, information on the "time", "instantaneous quantity", "average quantity", "maximum quantity", "minimum quantity", "instantaneous concentration", "average concentration", "maximum concentration", and "minimum concentration" is listed.

Chapter 4 Creating Rule Files

1 Introduction

Rule files are data files used to define the models to simulate in E-CELL2. In this file, enzymatic and substance related data are included with their inner-relations (reactions) network and environment settings such as the cell volume. Users can also edit Reactor files to define the kinetic equation of the reaction (Details in Chapter 5) and define the necessary substance and parameters in that Reactor inside the Rule file.

There are three types of Rule files:

- ss file
- er file
- eri file

Rule files can be written in spreadsheet format (ss) or directly as text (er). While writing rules as ss files under general spreadsheet programs or the model launcher are visually and simplicity-wise advantageous, er files can easily be batch-manipulated under script programs and macros. In any case, they will eventually be converted to eri files (where ss files must first be converted to er files before eri).

Rules files are divided into 4 sections: the System section (define the model environment for simulation), the Substance section (define substance name and value), the Reactor section (define reactions and inner-relation networks), and the Include section (to import other er files). These sections are not required to be in a single file or in this order. This chapter describes the methods in creating rule files in spreadsheet format.

2 Editing spreadsheet files (ss)

In this section, the required 4 sections with their lemmatizing lines will be described step-by-step by actually creating a rule file.

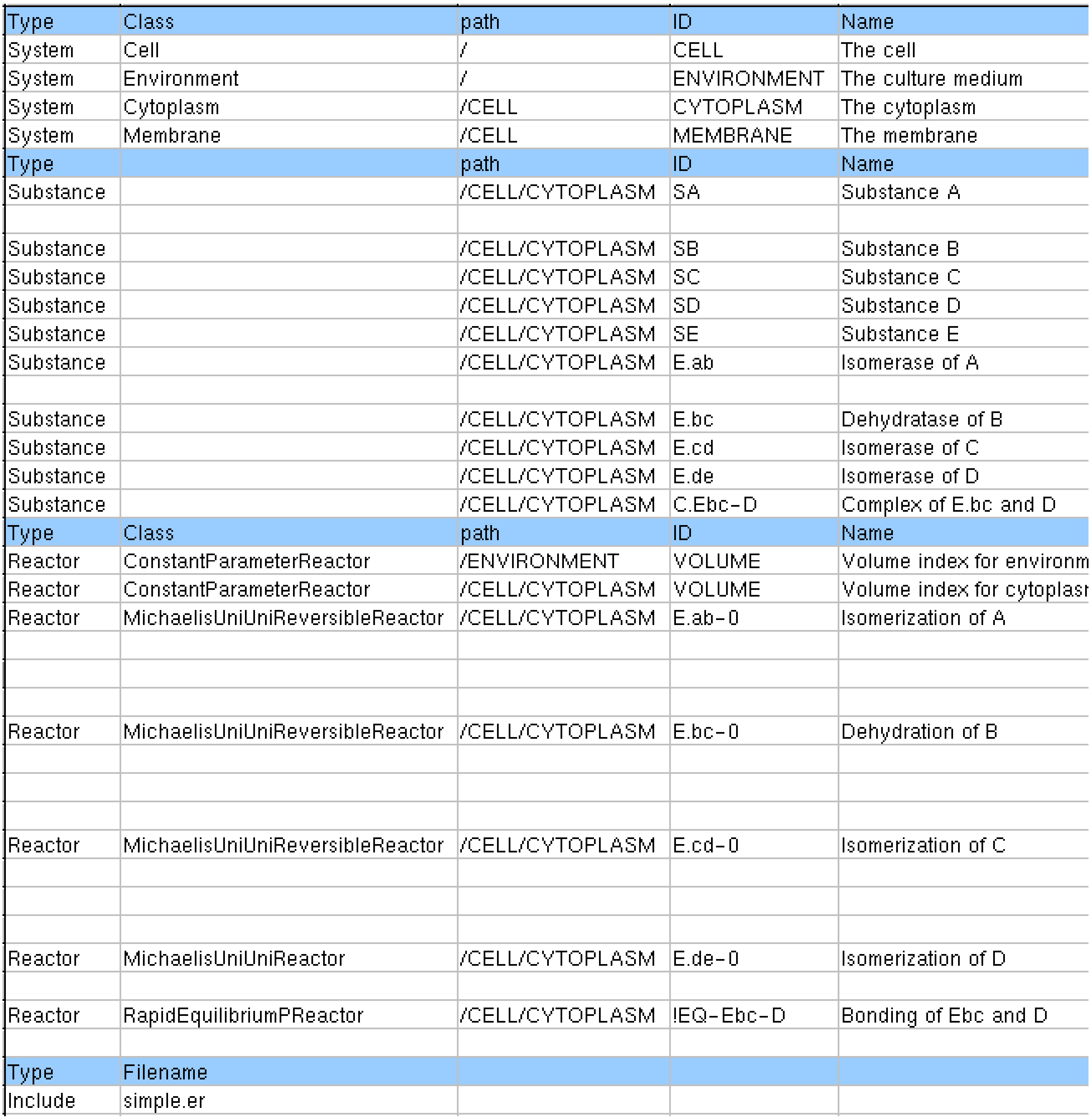
Note here that using the following in the columns are prohibited, for these are used to separate the items in the spreadsheet program.

- tab
- quotation
- double quotation

Including these may cause an adverse effect to the eri converting process.

2.1 Lemmatizing Words

Lemmatizing words are required for starting a new section. The columns with these lemmatizing words are called the "Lemmatizing Columns" where they must be included before starting a new section. The blue columns in Figure 1 are the lemmatizing columns, where the items are case/order-insensitive. Users can add a "memo" item to write down comments which will not be included in the eri conversion process.



Type	Class	path	ID	Name
System	Cell	/	CELL	The cell
System	Environment	/	ENVIRONMENT	The culture medium
System	Cytoplasm	/CELL	CYTOPLASM	The cytoplasm
System	Membrane	/CELL	MEMBRANE	The membrane
Type		path	ID	Name
Substance		/CELL/CYTOPLASM	SA	Substance A
Substance		/CELL/CYTOPLASM	SB	Substance B
Substance		/CELL/CYTOPLASM	SC	Substance C
Substance		/CELL/CYTOPLASM	SD	Substance D
Substance		/CELL/CYTOPLASM	SE	Substance E
Substance		/CELL/CYTOPLASM	E.ab	Isomerase of A
Substance		/CELL/CYTOPLASM	E.bc	Dehydratase of B
Substance		/CELL/CYTOPLASM	E.cd	Isomerase of C
Substance		/CELL/CYTOPLASM	E.de	Isomerase of D
Substance		/CELL/CYTOPLASM	C.Ebc-D	Complex of E.bc and D
Type	Class	path	ID	Name
Reactor	ConstantParameterReactor	/ENVIRONMENT	VOLUME	Volume index for environm
Reactor	ConstantParameterReactor	/CELL/CYTOPLASM	VOLUME	Volume index for cytoplasr
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.ab-0	Isomerization of A
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.bc-0	Dehydration of B
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.cd-0	Isomerization of C
Reactor	MichaelisUniUniReactor	/CELL/CYTOPLASM	E.de-0	Isomerization of D
Reactor	RapidEquilibriumPReactor	/CELL/CYTOPLASM	!EQ-Ebc-D	Bonding of Ebc and D
Type	Filename			
Include	simple.er			

Figure 1: Lemmatizing columns

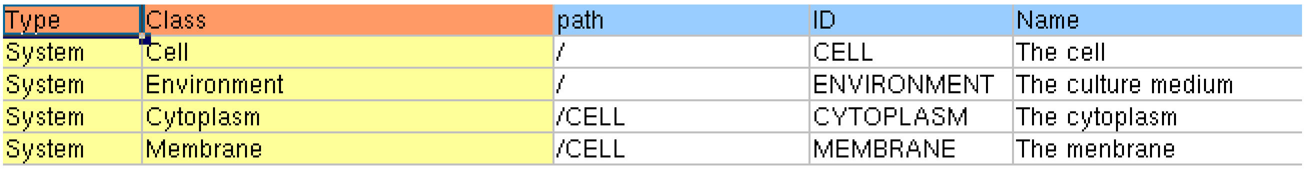
2.2 System Section

The System Section sets the simulation system architecture and the environment which are required to perform a simulation: the destination and the volume of the simulation target (System). Below are the descriptions:

Names	Definition	Default (if left blank)
Type	[System]	required
Class	Type of the simulation for	required
Path		required
ID	ID of the	required

Name	Name of the	ID
Inside	Define inner space (In)	required (Systems with membranes)
Outside	Define outer space (In)	required (Systems with membranes)
Volumeindex	Designate	required for Systems with Volume

Figure 2: System Section



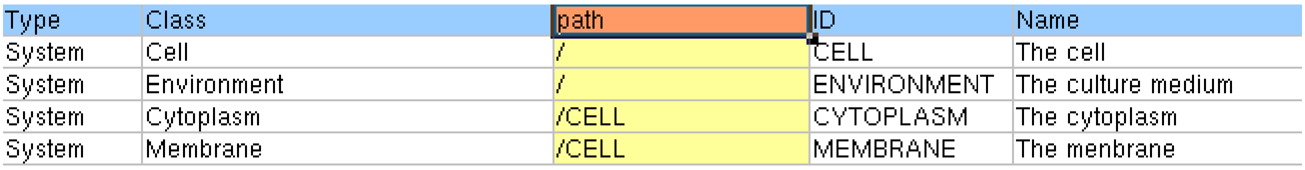
Type	Class	path	ID	Name
System	Cell	/	CELL	The cell
System	Environment	/	ENVIRONMENT	The culture medium
System	Cytoplasm	/CELL	CYTOPLASM	The cytoplasm
System	Membrane	/CELL	MEMBRANE	The menbrane

Figure 3: the System Section [Type, Class]

Input [System] in the Type row, and the destination of the system and class in the Class row. Cell, Environment, Cytoplasm, and Membrane can be defined as Class in this row. While these Classes are able to handle Substance, Reactor and Volume inside itself, Environment and Membrane cannot create a System under itself. Each Class is able to hold the following items:

Class	Substance	Reactor	Volume	System under itself
Cell	O	O	O	O
Environment	O	O	O	X
Cytoplasm	O	O	O	O
Membrane	O	O	O	X

Figure 4: Classes and their types



Type	Class	path	ID	Name
System	Cell	/	CELL	The cell
System	Environment	/	ENVIRONMENT	The culture medium
System	Cytoplasm	/CELL	CYTOPLASM	The cytoplasm
System	Membrane	/CELL	MEMBRANE	The menbrane

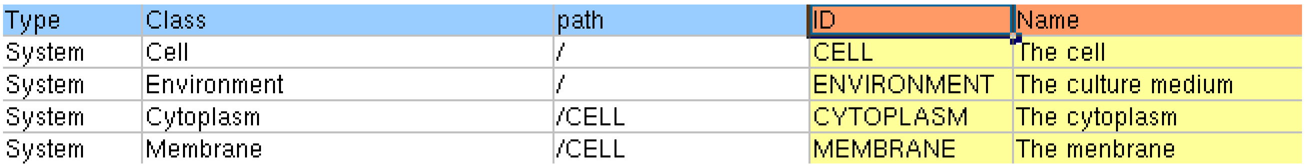
Figure 5: Systems Section [path row]

The path row defines the destination of the System created, acting as an address. Paths start with a RootSystem (/), the uppermost directory of the E-CELL2 System, and continue with user defined directories. Paths are divided up with single front-slashes (/system1/system2/system3 and so forth) and must be written as absolute paths. This section cannot be left blank. Note also that this section is case-sensitive.

FQEN, FQPN

FQEN (FullyQualifiedEntryName) is a method to define the destinations for , and , Using this method, the destination can be written as "/system0/system1/system2:ID", where the path and the ID are separated with a colon (:). If the path is "/CELL/CYTOPLASM" and the ID is "ATP", the FQEN would be "/CELL/CYTOPLASM:ATP".

FQEN cannot tell the difference among the System, Substance and Reactor items. FQPN adds these item name into the statement. In FQPN, the target is described as [Type]:[FQEN]. In the former example of ATP, the target in FQPN will be "Substance:/CELL/CYTOPLASM:ATP". Note that FQPN is case-sensitive.

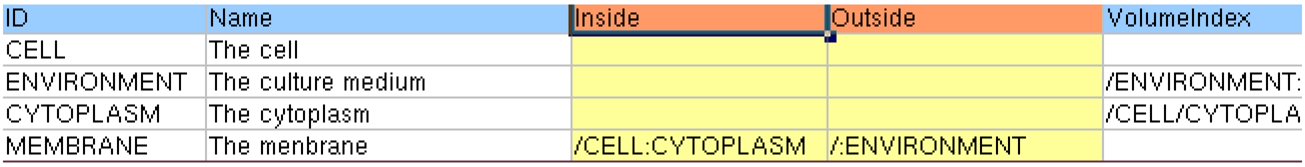


Type	Class	path	ID	Name
System	Cell	/	CELL	The cell
System	Environment	/	ENVIRONMENT	The culture medium
System	Cytoplasm	/CELL	CYTOPLASM	The cytoplasm
System	Membrane	/CELL	MEMBRANE	The menbrane

Figure 6: System Section [ID row, Name row]

IDs are used as process orders inside the E-CELL2 simulation software. Thus, if duplicate IDs are defined inside a System, a warning message will be displayed and the former defined ID will be used only. Note that ID must be defined as case-sensitive.

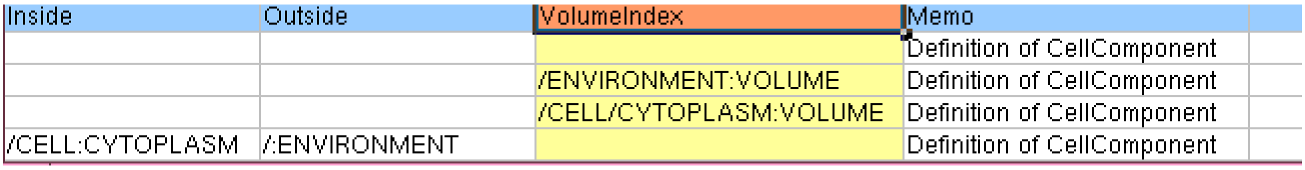
Define the ID of the System(location) in the ID row and the name of the System in the Name row.



ID	Name	Inside	Outside	VolumeIndex
CELL	The cell			
ENVIRONMENT	The culture medium			/ENVIRONMENT:
CYTOPLASM	The cytoplasm			/CELL/CYTOPLA
MEMBRANE	The menbrane	/CELL:CYTOPLASM	/:ENVIRONMENT	

Figure 7: System Section [Inside row, Outside row]

inside/outside rows are used in defining Systems inside or outside of the membrane (if the defined Class is Membrane). Although current E-CELL2 System cannot process these information, it will be dealt with in further versions by limiting the accessibility for each System. Figure 7 is an example of this, a System of MEMBRANE framed by CYTOPLASM and ENVIRONMENT.



Inside	Outside	VolumeIndex	Memo	
			Definition of CellComponent	
		/ENVIRONMENT:VOLUME	Definition of CellComponent	
		/CELL/CYTOPLASM:VOLUME	Definition of CellComponent	
/CELL:CYTOPLASM	/:ENVIRONMENT		Definition of CellComponent	

Figure 8: System Section [VolumeIndex row]

In the E-CELL2 System, the System volumes are defined as Reactors. When destinating FQEN of the Reactor as System's VolumeIndex, the activity value of the Reactor will be used as the volume of the System (in Liters). When destinating VolumeIndex of the System, the Reactor FQEN in correspondence must also be defined in the Reactor part.

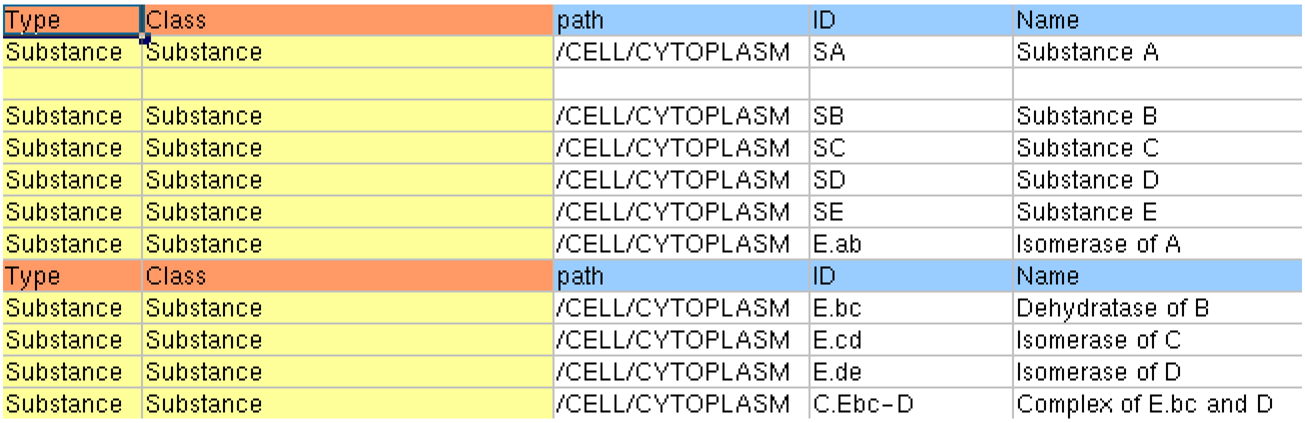
Although System may or may not have a volume (Membrane in the above example do not have a volume), their volume and the VolumeIndex is required in cases where Reactors use the substance concentrations for calculations or the Substance section uses Conc as the substance value(to recalculate the molecular amount)

2.3 Substance Section

Substance Section in the rule file is used to define all of the Substances involved in the simulation. The definition for each item is as follows:

Name	Definition	Default (if left blank)
Type	[Substance]	required
Class	[Substance] (can be omitted)	optional
Path	( )	required
ID	ID of the in target	required
Name	Name of the in target	ID
Qty	) (	0
Conc	( ) /	Qty
Arg_tag	value in orders: [Accumulator]	optional
Arg_coeff		required if Arg_tag is included

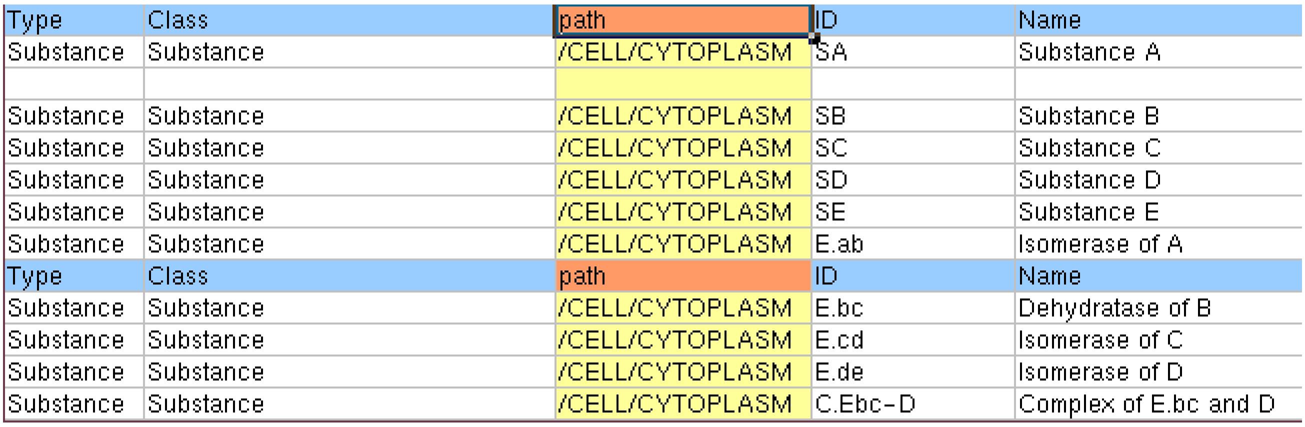
Figure 9: Substance Section Lemmatizing Names



Type	Class	path	ID	Name
Substance	Substance	/CELL/CYTOPLASM	SA	Substance A
Substance	Substance	/CELL/CYTOPLASM	SB	Substance B
Substance	Substance	/CELL/CYTOPLASM	SC	Substance C
Substance	Substance	/CELL/CYTOPLASM	SD	Substance D
Substance	Substance	/CELL/CYTOPLASM	SE	Substance E
Substance	Substance	/CELL/CYTOPLASM	E.ab	Isomerase of A
Type	Class	path	ID	Name
Substance	Substance	/CELL/CYTOPLASM	E.bc	Dehydratase of B
Substance	Substance	/CELL/CYTOPLASM	E.cd	Isomerase of C
Substance	Substance	/CELL/CYTOPLASM	E.de	Isomerase of D
Substance	Substance	/CELL/CYTOPLASM	C.Ebc-D	Complex of E.bc and D

Figure 10: Substance Section [Type row, Class row]

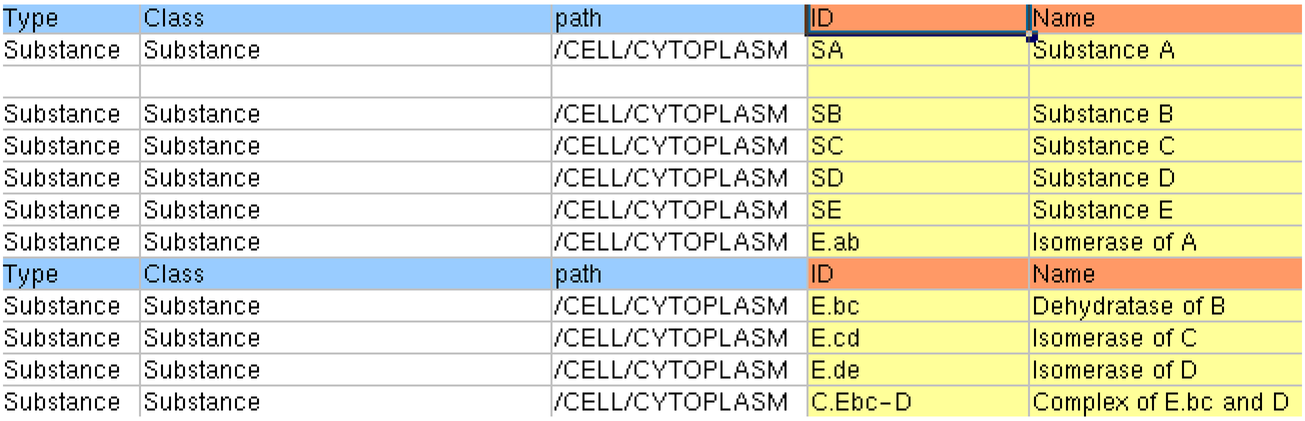
Define "Substance" in the Type row with the type format in the Class row. For current version only supports "Substance" as Class, always input "Substance" as Class (it will be automatically inserted if left blank)



Type	Class	path	ID	Name
Substance	Substance	/CELL/CYTOPLASM	SA	Substance A
Substance	Substance	/CELL/CYTOPLASM	SB	Substance B
Substance	Substance	/CELL/CYTOPLASM	SC	Substance C
Substance	Substance	/CELL/CYTOPLASM	SD	Substance D
Substance	Substance	/CELL/CYTOPLASM	SE	Substance E
Substance	Substance	/CELL/CYTOPLASM	E.ab	Isomerase of A
Type	Class	path	ID	Name
Substance	Substance	/CELL/CYTOPLASM	E.bc	Dehydratase of B
Substance	Substance	/CELL/CYTOPLASM	E.cd	Isomerase of C
Substance	Substance	/CELL/CYTOPLASM	E.de	Isomerase of D
Substance	Substance	/CELL/CYTOPLASM	C.Ebc-D	Complex of E.bc and D

Figure 11: Substance Section [path row]

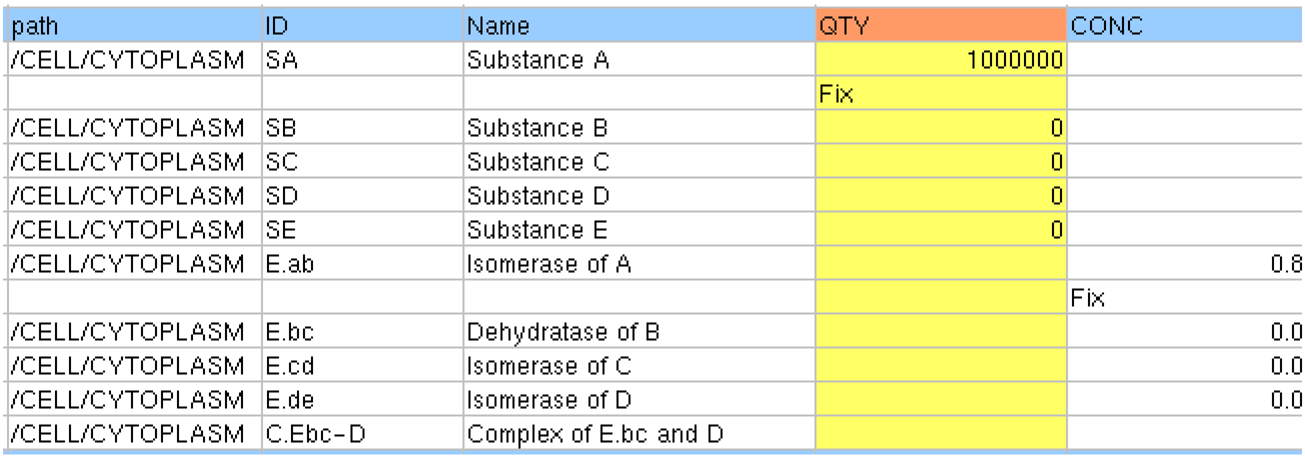
Define the location(System) of the Substance in the path row. The format is identical to the System section with front-slash seperators (/system1/system2/system3). Note that Substances with the same ID and Names in different Systems are treated as different elements in E-CELL2 System. Note also that this section is case-sensitive.



Type	Class	path	ID	Name
Substance	Substance	/CELL/CYTOPLASM	SA	Substance A
Substance	Substance	/CELL/CYTOPLASM	SB	Substance B
Substance	Substance	/CELL/CYTOPLASM	SC	Substance C
Substance	Substance	/CELL/CYTOPLASM	SD	Substance D
Substance	Substance	/CELL/CYTOPLASM	SE	Substance E
Substance	Substance	/CELL/CYTOPLASM	E.ab	Isomerase of A
Type	Class	path	ID	Name
Substance	Substance	/CELL/CYTOPLASM	E.bc	Dehydratase of B
Substance	Substance	/CELL/CYTOPLASM	E.cd	Isomerase of C
Substance	Substance	/CELL/CYTOPLASM	E.de	Isomerase of D
Substance	Substance	/CELL/CYTOPLASM	C.Ebc-D	Complex of E.bc and D

Figure 12: Substance Section [ID row, Name row]

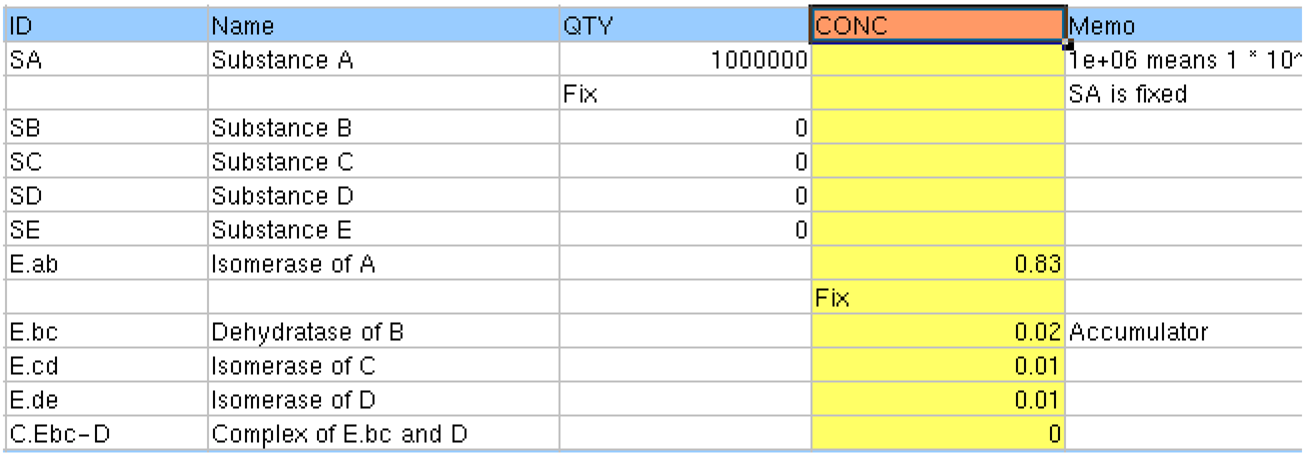
, (-)



path	ID	Name	QTY	CONC
/CELL/CYTOPLASM	SA	Substance A	1000000	
			Fix	
/CELL/CYTOPLASM	SB	Substance B	0	
/CELL/CYTOPLASM	SC	Substance C	0	
/CELL/CYTOPLASM	SD	Substance D	0	
/CELL/CYTOPLASM	SE	Substance E	0	
/CELL/CYTOPLASM	E.ab	Isomerase of A		0.8
				Fix
/CELL/CYTOPLASM	E.bc	Dehydratase of B		0.0
/CELL/CYTOPLASM	E.cd	Isomerase of C		0.0
/CELL/CYTOPLASM	E.de	Isomerase of D		0.0
/CELL/CYTOPLASM	C.Ebc-D	Complex of E.bc and D		

Figure 13: Substance Section [QTY Section]

Qty defines the initial molecular amount of the substance in the user defined volume. Qty cannot coexist with Conc; displays a Warning message and adopts the Qty value. If neither values are inputted, Qty will start as 0. It is also possible to fix the Qty value at a certain amount (refer to the later sections).

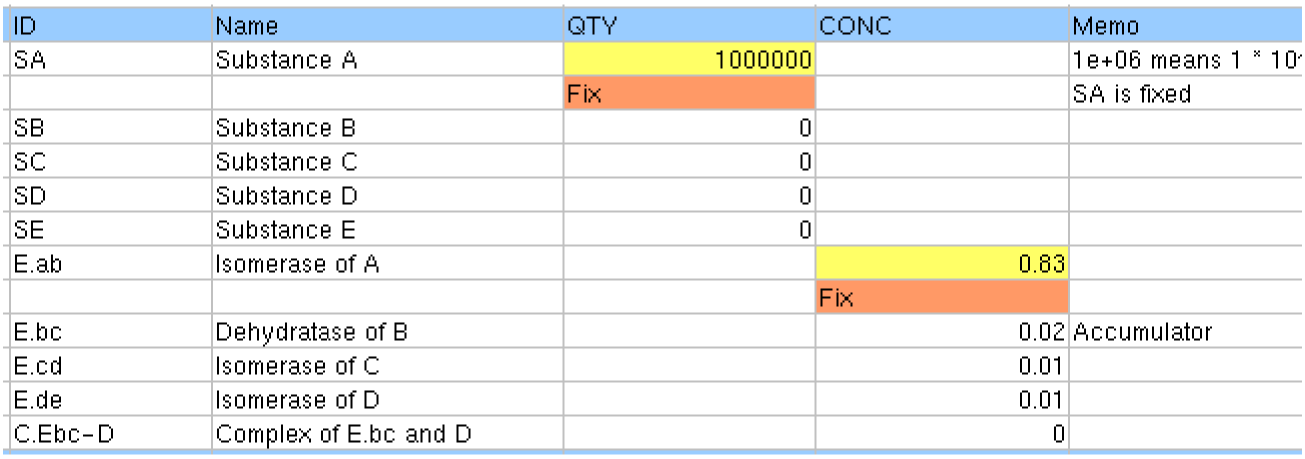


ID	Name	QTY	CONC	Memo
SA	Substance A	1000000		1e+06 means 1 * 10
		Fix		SA is fixed
SB	Substance B	0		
SC	Substance C	0		
SD	Substance D	0		
SE	Substance E	0		
E.ab	Isomerase of A		0.83	
			Fix	
E.bc	Dehydratase of B		0.02	Accumulator
E.cd	Isomerase of C		0.01	
E.de	Isomerase of D		0.01	
C.Ebc-D	Complex of E.bc and D		0	

Figure 14: Substance Section [Conc row]

For Conc, define the concentration values in M(mol/l). While converting er files to eri, these values will be used for the "init_act" of Reactor defined in the Volume part of the System section with the Avogadro's constant (6.0221367e23) to be converted to molecular amount. Conc cannot coexist with Qty; Qty value will be used in the eri if none is inputted in Conc. Note that having no VolumeIndex for System sections with substances result with the System Volume equal to 0. Conc value can be fixed.

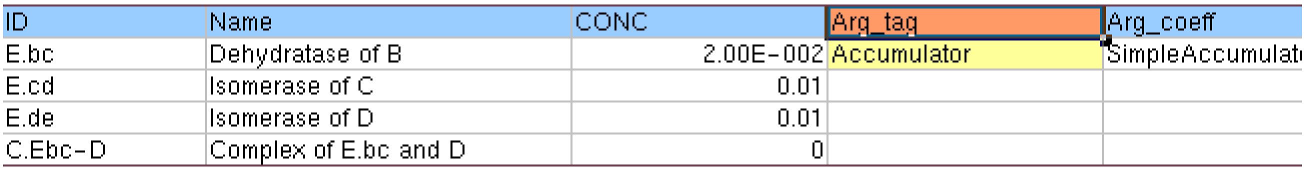
Fix function



ID	Name	QTY	CONC	Memo
SA	Substance A	1000000		1e+06 means 1 * 10
		Fix		SA is fixed
SB	Substance B	0		
SC	Substance C	0		
SD	Substance D	0		
SE	Substance E	0		
E.ab	Isomerase of A		0.83	
			Fix	
E.bc	Dehydratase of B		0.02	Accumulator
E.cd	Isomerase of C		0.01	
E.de	Isomerase of D		0.01	
C.Ebc-D	Complex of E.bc and D		0	

Figure 15 Fix function

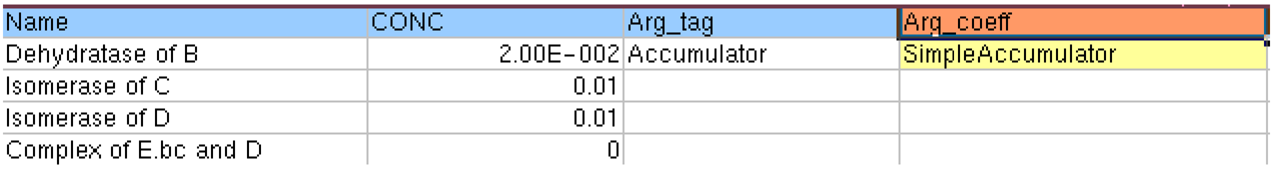
Fix function manipulates the Substance Column so that the substance will not get the influence of the Reactor to fix the inputted Qty or the Conc value. To fix these values, input "Fix" under the Qty/Conc value.



ID	Name	CONC	Arg_tag	Arg_coeff
E.bc	Dehydratase of B	2.00E-002	Accumulator	SimpleAccumulat
E.cd	Isomerase of C	0.01		
E.de	Isomerase of D	0.01		
C.Ebc-D	Complex of E.bc and D	0		

Figure 16: Substance Part [Arg_tag row]

Accumulator describes the method in rounding off the decimal points. Defining "Accumulator" for Arg_tag selects the method. Arg_coeff describes the specifics in selecting the actual rounding method; choosing multiple methods for calculations have no proof in accuracy.



Name	CONC	Arg_tag	Arg_coeff
Dehydratase of B	2.00E-002	Accumulator	SimpleAccumulator
Isomerase of C	0.01		
Isomerase of D	0.01		
Complex of E.bc and D	0		

Figure 17: Substance section[Arg_coeff row]

Select the capsulated Accumulator Class for rounding decimal points in Substance and define in Arg_coeff. Below are the selectable choices:

Accumulator Class	**Rounding Decimal Points**
ReserveAccumulator	Integer and decimal parts are stored separately. Only the integer part will be used for the results; while under calculations, the decimal parts will be added up until the sum crosses the 1.0 border to be added on to the integer part. Variation Integer Decimal Initial 0 0 0.8 0 0.8 0.8 1 0.6 0.8 2 0.4
SimpleAccumulator	Use real values for calculations.(i386:18 digits, alpha 15 digits (in 10s))
RoundDownAccumulator	Round down decimal points.
RoundUpAccumulator	Round up decimal points
RoundOffAccmulator	Round off decimal points
MonteCarloAccumulator	Store separately integer and decimal parts and handle the latter as random numbers and round off. Ex: if the actual change is 3.4, The change will be 60% wise 3 and 40% wise 4.

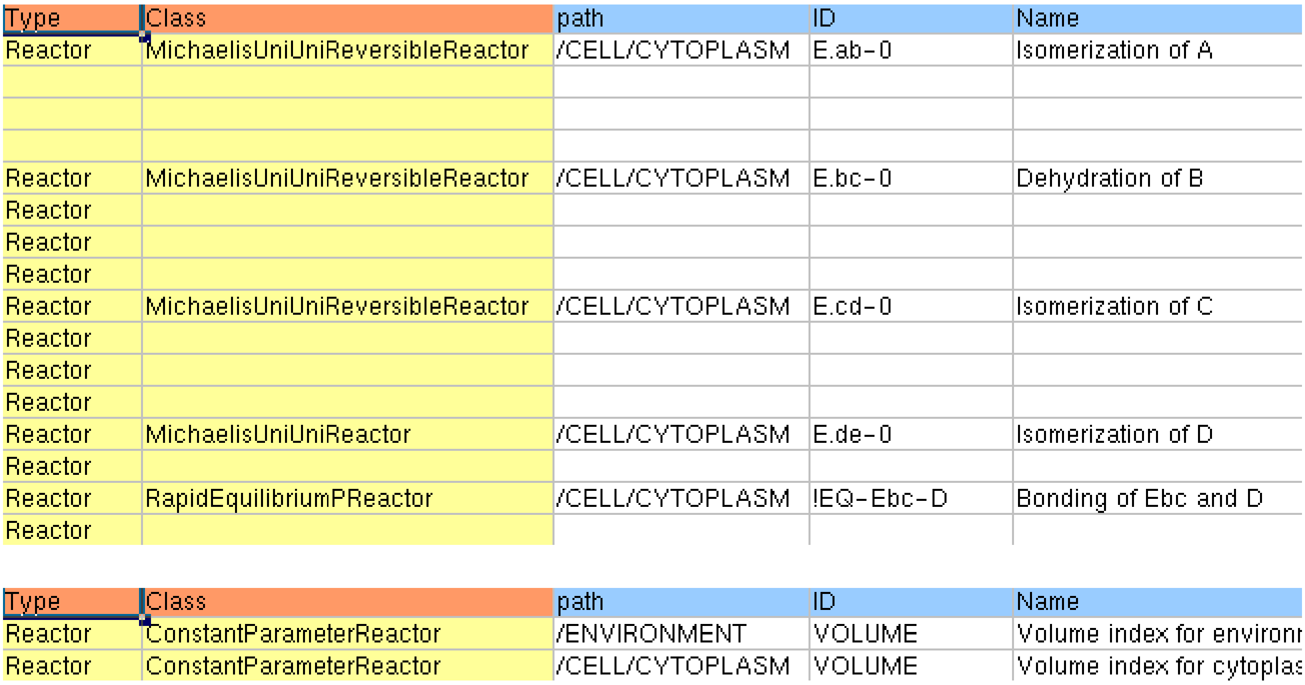
2.4 Reactor Section

The Reactor Section is to select the environment variables such as the reactions(using substances defined in the Substance sections), inter-relationships, and volume. In this section, the Reactor will be stoichiometrically defined with the involved Substances. Below are the specifications.

Name	Definition	Default if left blank
Type	[Reactor]	required
Class	(kinetic equations)	required

Path	Reactor Path( )	required
ID	Reactor ID	required
Name	Name of the Reaction	ID
init_act	Initial Reactor Activity	omittable unless defined in VolumeIndex
S_ID	Substrate ID	required
S_Path	Substrate Path(System)	Path
S_Coeff	Substrate Stoichiometric Coefficient	1
P_ID	Product ID	required
P_Path	Product Path(System)	Path
P_Coeff	Product Stoichiometric Coefficient	1
C_ID	Catalyst ID	required
C_Path	Catalyst Path(System)	Path
arg_tag	Constant name (for Reactor use)	required for certain reactions
arg_coeff	Reactor value of the constant	required for arg_tag reactions
E_ID	Effector ID	required unless specified
E_Path	Effector Path (system)	Path
E_Coeff	Effector Coefficient	1

Figure 18: Reactor Section Specifications

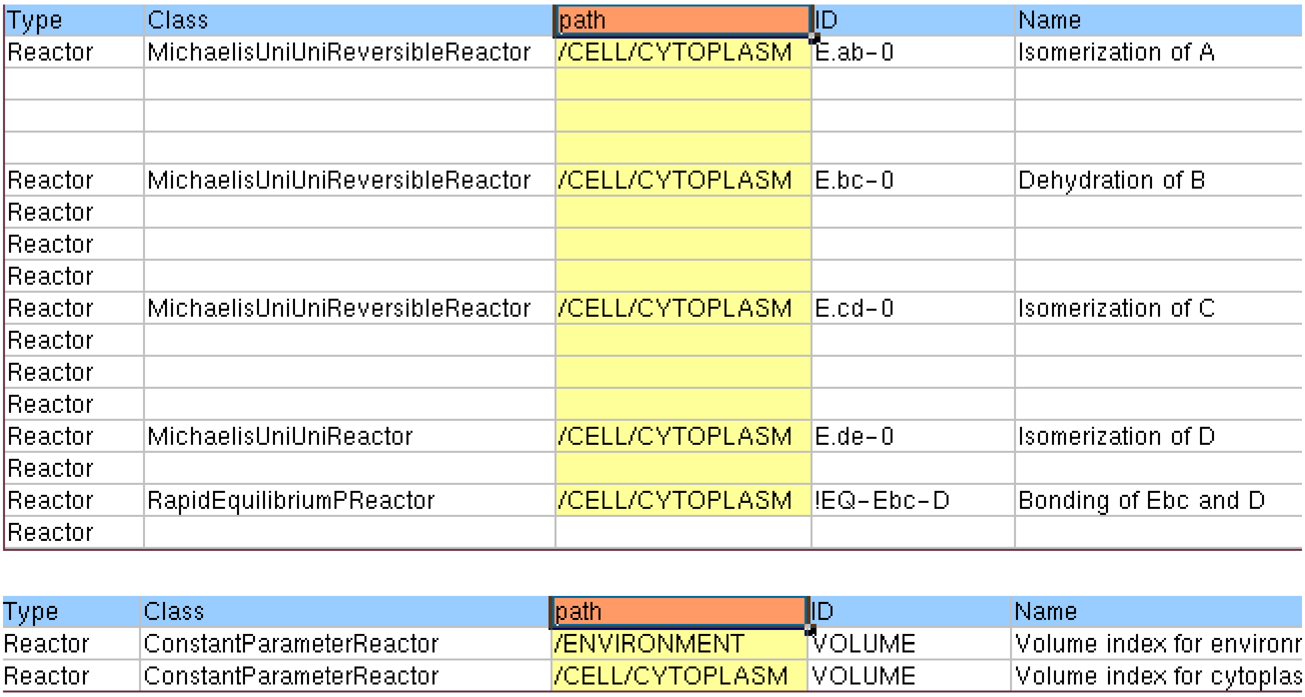


Type	Class	path	ID	Name
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.ab-0	Isomerization of A
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.bc-0	Dehydration of B
Reactor				
Reactor				
Reactor				
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.cd-0	Isomerization of C
Reactor				
Reactor				
Reactor				
Reactor	MichaelisUniUniReactor	/CELL/CYTOPLASM	E.de-0	Isomerization of D
Reactor				
Reactor	RapidEquilibriumPReactor	/CELL/CYTOPLASM	!EQ-Ebc-D	Bonding of Ebc and D
Reactor				

Type	Class	path	ID	Name
Reactor	ConstantParameterReactor	/ENVIRONMENT	VOLUME	Volume index for environr
Reactor	ConstantParameterReactor	/CELL/CYTOPLASM	VOLUME	Volume index for cytoplas

Figure 19: Reactor Section[Type row, Class row]

Reactor Types (Class: Kinetic Equations) are defined in the class row. In the row, the Reactors must be written as ???Reactor name format. (The Example below describes the MichaelisMenten related equations, and thus are written as Michaelis???Reactor, each defining Reactor items with Volume) Please refer the spec sheet for other predefined Reactors. Define the ID for the reaction (or the interrelations). Note that the items must be written case-sensitive.

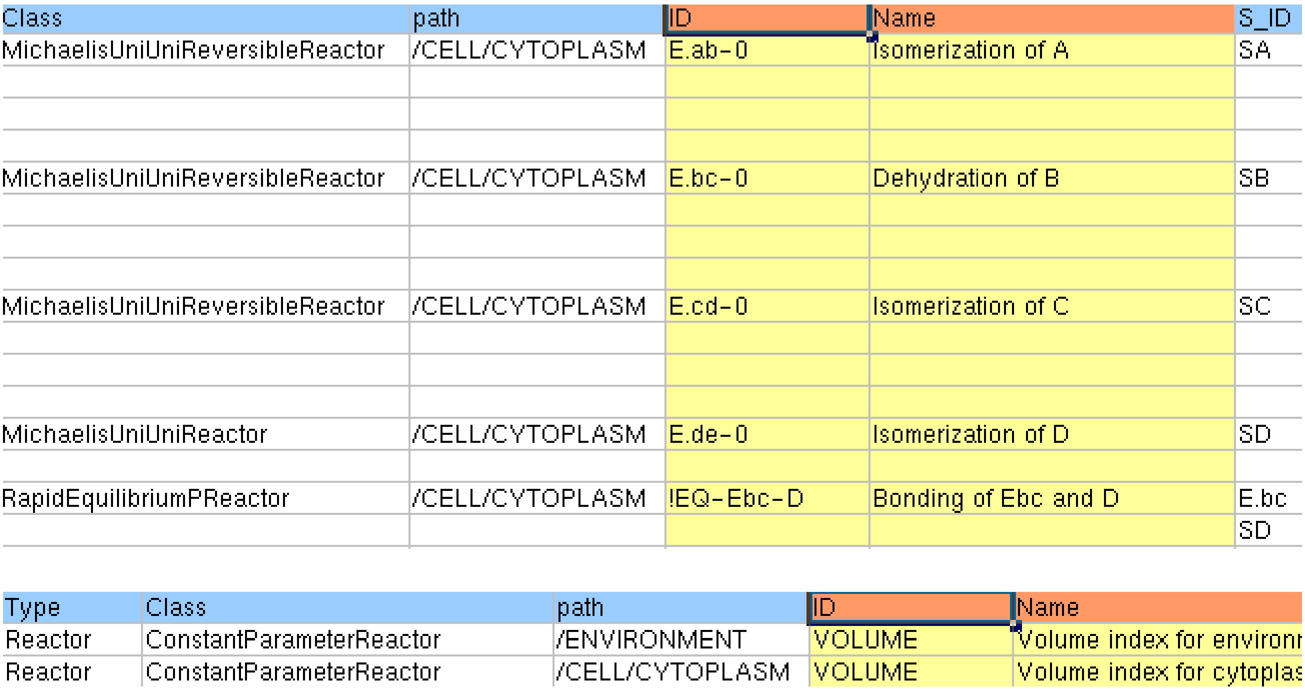


Type	Class	path	ID	Name
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.ab-0	Isomerization of A
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.bc-0	Dehydration of B
Reactor				
Reactor				
Reactor				
Reactor	MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.cd-0	Isomerization of C
Reactor				
Reactor				
Reactor				
Reactor	MichaelisUniUniReactor	/CELL/CYTOPLASM	E.de-0	Isomerization of D
Reactor				
Reactor	RapidEquilibriumPReactor	/CELL/CYTOPLASM	!EQ-Ebc-D	Bonding of Ebc and D
Reactor				

Type	Class	path	ID	Name
Reactor	ConstantParameterReactor	/ENVIRONMENT	VOLUME	Volume index for environr
Reactor	ConstantParameterReactor	/CELL/CYTOPLASM	VOLUME	Volume index for cytoplas

Figure 20: Reactor Section [path row]

Define the System Path for the reactions or the interactions in the Path row. Similar to the paths for the Substance section, it can be observed below that the Reaction occurs under /CELL/CYTOPLASM. Note that the texts are case-sensitive.



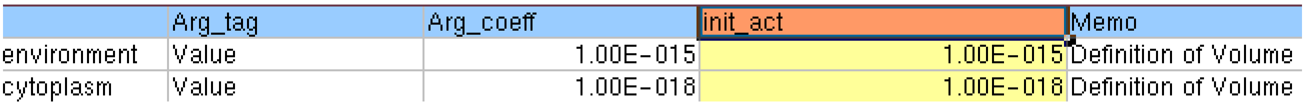
Class	path	ID	Name	S_ID
MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.ab-0	Isomerization of A	SA
MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.bc-0	Dehydration of B	SB
MichaelisUniUniReversibleReactor	/CELL/CYTOPLASM	E.cd-0	Isomerization of C	SC
MichaelisUniUniReactor	/CELL/CYTOPLASM	E.de-0	Isomerization of D	SD
RapidEquilibriumPReactor	/CELL/CYTOPLASM	!EQ-Ebc-D	Bonding of Ebc and D	E.bc
				SD

Type	Class	path	ID	Name
Reactor	ConstantParameterReactor	/ENVIRONMENT	VOLUME	Volume index for environr
Reactor	ConstantParameterReactor	/CELL/CYTOPLASM	VOLUME	Volume index for cytoplas

Figure 21: Reactor Section [ID row, Name row]

Name rows are to define the names for the reactions or the interactions. The ID will be the name for the eri file if this is left blank. For Reactors with volume, it is desired for that Reactor to have names that can be understood as volume-related (see chapter 3 for details). However, for

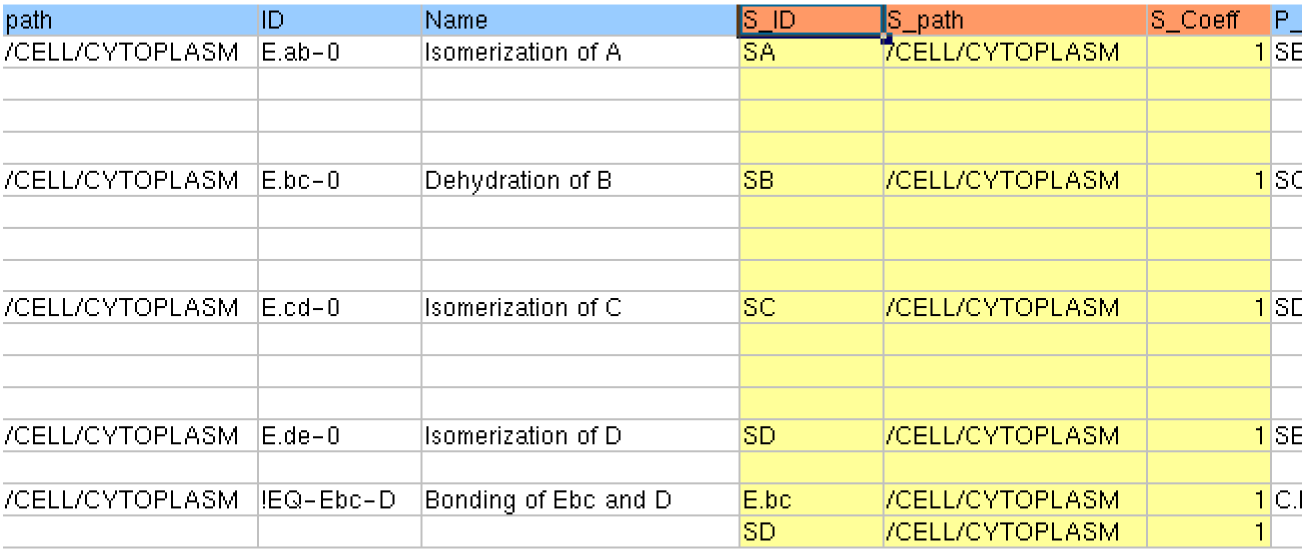
FluxReactor (or in any case where the IDs start out with an "!"), the behavior is non-guaranteed.



	Arg_tag	Arg_coeff	init_act	Memo
environment	Value	1.00E-015	1.00E-015	Definition of Volume
cytoplasm	Value	1.00E-018	1.00E-018	Definition of Volume

Figure 22: Reactor Section [init_act row]

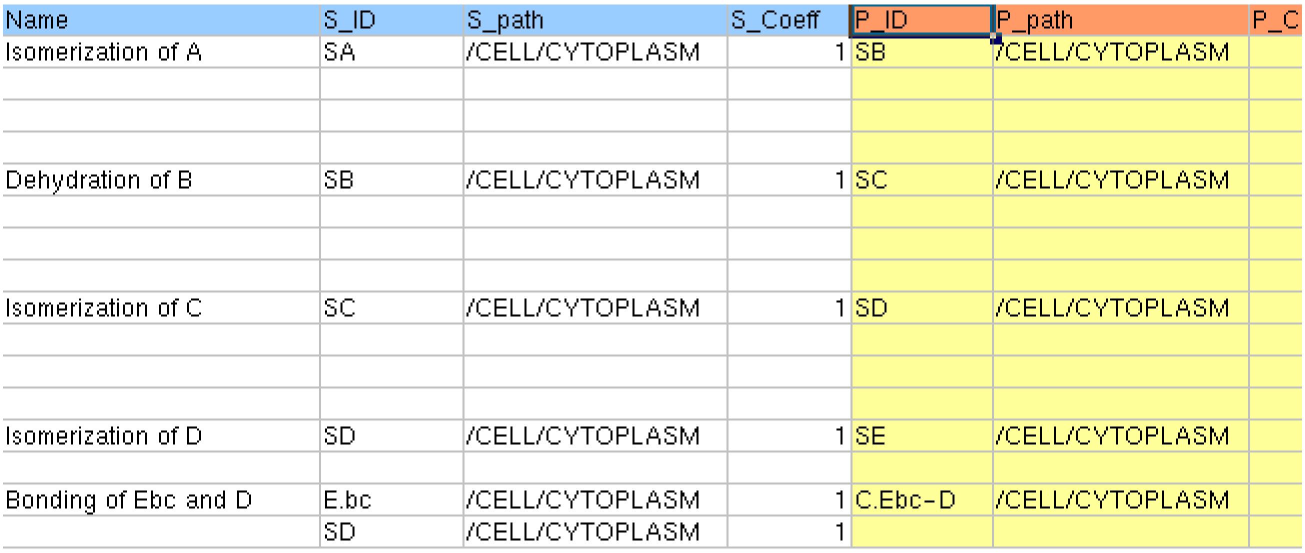
For Reactor activities that get calculated 1 step after the another, these Reactor activity values need to be set at the start of the simulation. init_act row is to input those initial values, where in defining Reactors pointed in the VolumeIndex of the System section it is always required. This init_act will be used initially as Reactor activity values before starting the simulation. The Volume to be used in every reaction inside the model will be set as 0 at the first step initial point of the simulation, if no value exist here --- disabling the simulation. The initial Volume in the example below is defined in the ConstantParameterReactor of the Reactor (defining volume) inside the System.



path	ID	Name	S_ID	S_path	S_Coeff	P_
/CELL/CYTOPLASM	E.ab-0	Isomerization of A	SA	/CELL/CYTOPLASM	1	SE
/CELL/CYTOPLASM	E.bc-0	Dehydration of B	SB	/CELL/CYTOPLASM	1	SC
/CELL/CYTOPLASM	E.cd-0	Isomerization of C	SC	/CELL/CYTOPLASM	1	SD
/CELL/CYTOPLASM	E.de-0	Isomerization of D	SD	/CELL/CYTOPLASM	1	SE
/CELL/CYTOPLASM	!EQ-Ebc-D	Bonding of Ebc and D	E.bc	/CELL/CYTOPLASM	1	C.I
			SD	/CELL/CYTOPLASM	1	

Figure 23: Reactor Section [S_ID row, S_path row, S_Coeff row]

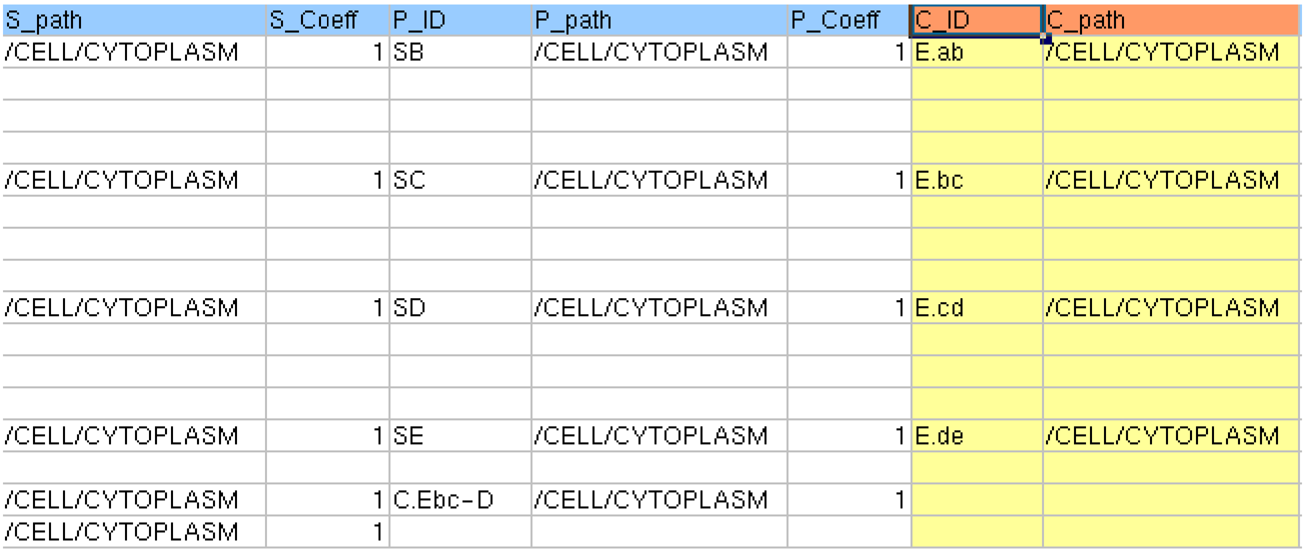
S_ID and S_Path defines the Substrate ID and its path of the System (location). In the reaction below, ATP and Glc are the substrates (under CYTOPLASM) in the reaction. Define the stoichiometric coefficient of the Substrate in S_Coeff.



Name	S_ID	S_path	S_Coeff	P_ID	P_path	P_C
Isomerization of A	SA	/CELL/CYTOPLASM	1	SB	/CELL/CYTOPLASM	
Dehydration of B	SB	/CELL/CYTOPLASM	1	SC	/CELL/CYTOPLASM	
Isomerization of C	SC	/CELL/CYTOPLASM	1	SD	/CELL/CYTOPLASM	
Isomerization of D	SD	/CELL/CYTOPLASM	1	SE	/CELL/CYTOPLASM	
Bonding of Ebc and D	E.bc	/CELL/CYTOPLASM	1	C.Ebc-D	/CELL/CYTOPLASM	
	SD	/CELL/CYTOPLASM	1			

Figure 24: Reactor Section [P_ID row, P_path row, P_Coeff row]

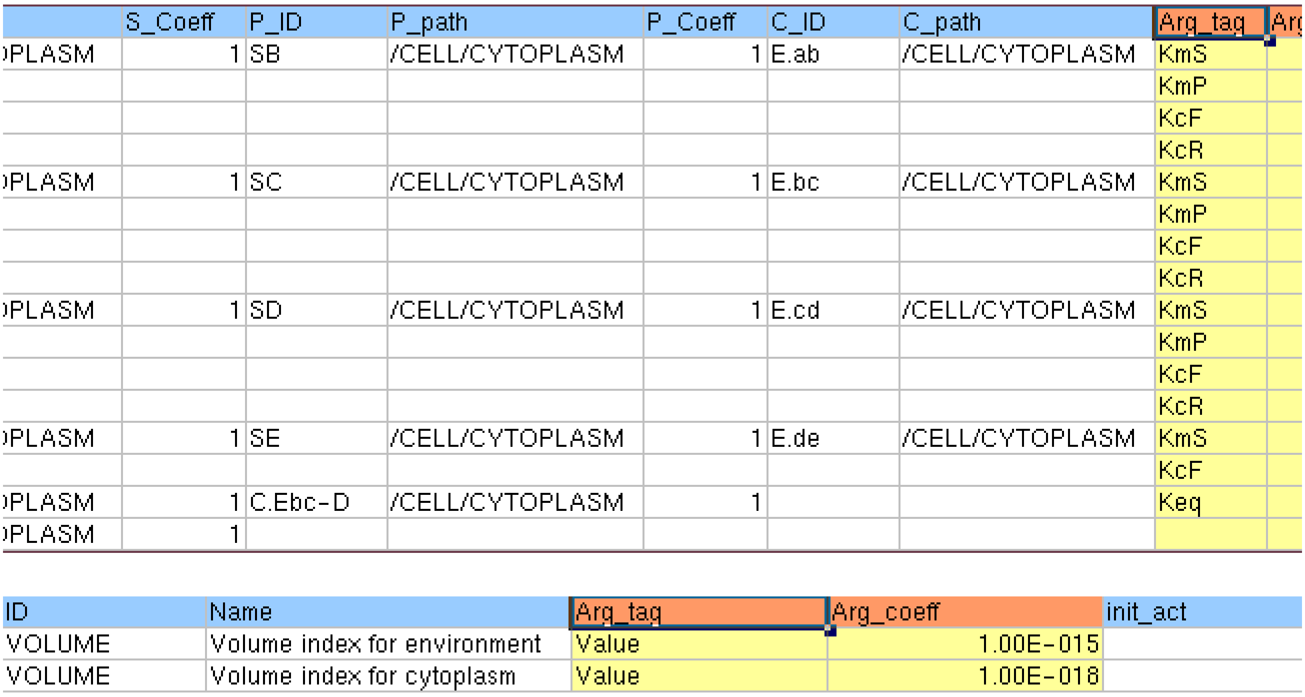
P_ID and P_Path defines the Product ID and its path of the System (location). In the reaction below, ADP and G6P are the products (under CYTOPLASM) in the reaction. Define the stoichiometric coefficient of the Product in P_Coeff.



S_path	S_Coeff	P_ID	P_path	P_Coeff	C_ID	C_path
/CELL/CYTOPLASM	1	SB	/CELL/CYTOPLASM	1	E.ab	/CELL/CYTOPLASM
/CELL/CYTOPLASM	1	SC	/CELL/CYTOPLASM	1	E.bc	/CELL/CYTOPLASM
/CELL/CYTOPLASM	1	SD	/CELL/CYTOPLASM	1	E.cd	/CELL/CYTOPLASM
/CELL/CYTOPLASM	1	SE	/CELL/CYTOPLASM	1	E.de	/CELL/CYTOPLASM
/CELL/CYTOPLASM	1	C.Ebc-D	/CELL/CYTOPLASM	1		
/CELL/CYTOPLASM	1					

Figure 25: Reactor Section [C_ID row, C_path row]

C_ID defines the catalysts influencing Substrates and Products inside the reaction. C_Path defines the Catalyst System (location) Path. In default, the ID and the Path for the catalysts will be identical with the reaction's ID and Path.



	S_Coeff	P_ID	P_path	P_Coeff	C_ID	C_path	Arg_tag	Arg
)PLASM	1	SB	/CELL/CYTOPLASM	1	E.ab	/CELL/CYTOPLASM	KmS	
							KmP	
							KcF	
							KcR	
)PLASM	1	SC	/CELL/CYTOPLASM	1	E.bc	/CELL/CYTOPLASM	KmS	
							KmP	
							KcF	
							KcR	
)PLASM	1	SD	/CELL/CYTOPLASM	1	E.cd	/CELL/CYTOPLASM	KmS	
							KmP	
							KcF	
							KcR	
)PLASM	1	SE	/CELL/CYTOPLASM	1	E.de	/CELL/CYTOPLASM	KmS	
							KcF	
)PLASM	1	C.Ebc-D	/CELL/CYTOPLASM	1			Keq	
)PLASM	1							

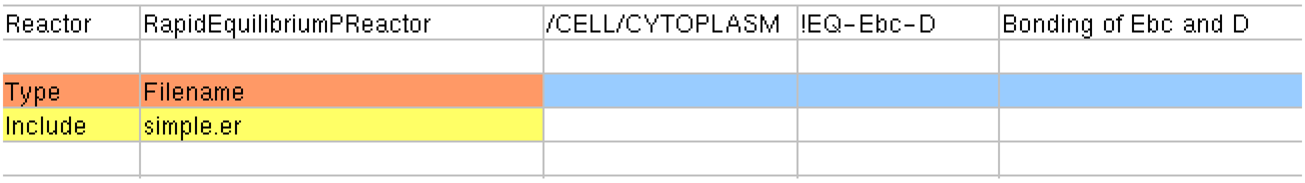
ID	Name	Arg_tag	Arg_coeff	init_act
VOLUME	Volume index for environment	Value	1.00E-015	
VOLUME	Volume index for cytoplasm	Value	1.00E-018	

Figure 26: Reactor Section [Arg_tag row, Arg_coeff row]

In the arg_tag row, the constant name that the kinetic equation will use as a parameter must be defined. This constant name is must exist for each Reactor; in the example above, the second-last Reactor uses "MichaelisUniUniReactor" as the kinetic equation which make the arg_tag "KcF and KmS". (Refer to the spec sheet for specifications on general reactors) If multiple parameters exist, write them in the row under the another, and no other data must be defined for that row (see example). Note that this section is also case-sensitive. For ConstantParameterReactor (For calculating the Volume of the Reactor), insert Value in the Arg_tag.

Also, users are able to define any Effectors that exist in the reaction where the amount itself is not influenced by that reactor. Define the Substance ID and its System(location) path under E_ID and E_Path. Define the stoichiometric coefficient of the Effector in E_Coeff. Refer to the Reactor chapter for more information.

2.5 Include Section



Reactor	RapidEquilibriumPReactor	/CELL/CYTOPLASM	!EQ-Ebc-D	Bonding of Ebc and D
Type	Filename			
Include	simple.er			

Figure 27: Include Section

By including an Include Section in the rule, it is possible to import other er files into the ERI conversion. As similar with other sections, the Include section can be placed many times at any position. This function is usable in cases where general lines that must be included. The Include Section need not be included if there are no files that need to be imported. The lemmatizing words are "Type" and "Filename". Define [Include] as Type and the actual filename under Filename. The above example is importing the file "simple.er".

3 Saving and Converting Spreadsheets

3.1 Saving Spreadsheets

To convert to ERI file which can be read in the E-CELL2 System, the input file must be in a tab separated text format that the converting software can recognize. To create such a text file, it is desired to use the spreadsheet creating function of the ModelingLauncher or general software such as Microsoft Excel. A sample for the ss file can be found under the standard directory with the .txt extensions. In the basic install, the directory will be C:¥E-CELL2¥standard. ModelingLauncher which can be used for creating rule files can be executed with the ModelingLauncher shortcut or from Programs->E-CELL2->ModelingLauncher in the Start menu. Refer to Chapter 6 for more details.

3.2 Converting to ERI format

In order to perform simulations with the E-CELL2 System, it is required to convert the ss files (or the er files) into eri format. Launch the ModelLauncher, select the Rule tab, and select "Choose..." from the "File..." button on the right.

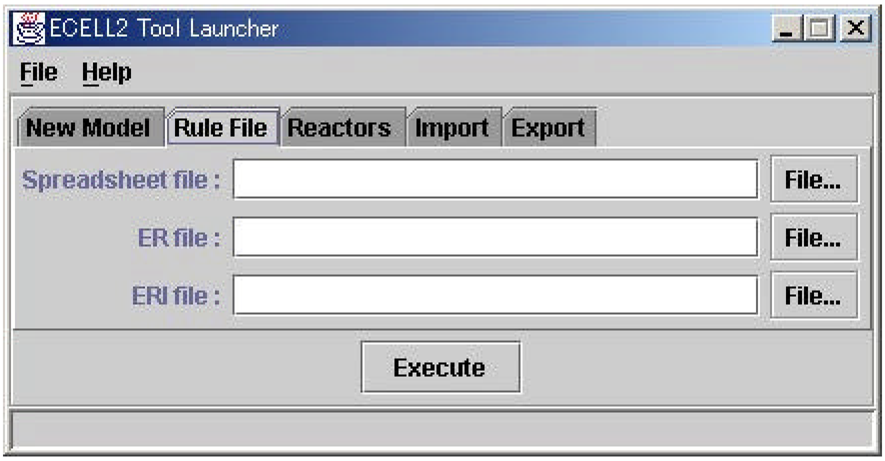


Figure 28: Modeling Launcher

After selecting the input file from the file window, eri file will be created from pressing the Execute button. In the GUI version, By selecting "Load Rule" from the "File" menu loads up the file for input. Refer to chapter 3 on loading E-CELL2.

Chapter 5 Creating User Defined Reactor Files

1 Introduction

In the E-CELL2 System, cellular models can be created with the combination of three object types: Substance (to describe substance amount), Reactor (describe the reaction and the reactor system path), and System (to describe certain functions). Reactor describes the time-change of the Substance amount which plays an essential role under an E-CELL2 model.

Reactor objects can be freely manipulated by the user. Reactor objects must be compiled in separate with the E-CELL2 System and therefore must be converted to a format that E-CELL2 can read; if correctly written, the E-CELL2 System loads up the defined Reactor. For Reactor objects are written in C++, users can create almost any kind of a kinetic model.

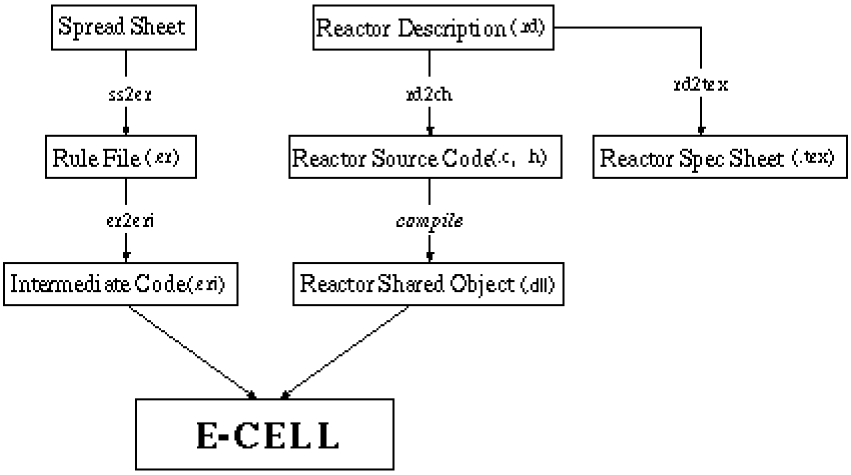

Spread Sheet
ss2er
Rule File (.er)
er2eri
Intermediate Code(.eri)
Reactor Description (.rd)
rd2ch
Reactor Source Code(.c, .h)
compile
Reactor Shared Object (.dll)
rd2tex
Reactor Spec Sheet (.tex)
E-CELL

Figure: The scheme of files used in E-CELL2 System

The graph above describes the files used in E-CELL2 System. As it can be seen, Reactors are created and compiled in separate of the E-CELL2 System itself. In order to create a Reactor object, the Reactor Description file (RD file) must be written with the actual commands and the required data(such as the equation and the parameters) inside. The RD file gets converted to C++ and .dll modules executable with E-CELL2 System. RD files can also be converted to a Reactor Spec Sheet in Latex format.

1.1 Postern Reactor

Two kinds of Reactors exist: Regular Reactors and Postern Reactors. While regular reactors use integrating functions to solve multi-derivative equations, postern reactors are separated from these integrating functions and used in cases where derivative equations are not suitable for calculating the Substance amount. The two reactors realize due to the multi-step calculation architecture; after the React step in the regular reactor is executed, the calculation will be performed in the Postern Step. Postern Reactors can be used in the following case:

- In cases where the substance amount is controlled as discrete. (in cases where the substance amount is very low)
- In cases where the substance amount can be defined directly without the integrating functions. For reactors describing the transcription start point or in cases where all substance value reduce to half because of a mitosis
- In cases the reaction is described as an algebraic function. Suitable for RapidEquilibrium Reactions or for Osmotic Calculations.

While regular reactors calculate the reaction speed in order to define the substance amount, postern reactors do not consider the substance flux and handle all of the substance involved . Thus, while regular reactors use the velocity(Float v)method in performing calculations, postern Reactors use the SetQuantity(Float Q)method in returning calculation results.

Postern reactors have "PReactor" on the end of their name. (ex: "**ABCPReactor**")

Each postern reactor must and can only handle a single substance where the substance cannot be defined more than once (if so, the simulation accuracy cannot be guaranteed). If multiple reactions effect a single substance, each reactions must be inside a singular postern reactor such as the GeneralRapid EquilibriumPReactor.

2 Reactor Description File

Though several General Reactors for simulation is already included in the E-CELL2 System Package (refer to section 5), in order to create non-standard reactions or to perform higher leveled simulations, it is required for the user to create their own Reactor files.

2.1 Creating a Reactor Description File

RD files can be created by including the keyword - value pair lines. Note for the following while creating RD files:

- Keywords must be Alphabets (uppercase only) or underbars without spaces.
- Keywords must start with @ or %. While lines that start with @ will simply be included into the target, the lines starting with % will be treated as hash separated by dots (".").
- Keywords that start with "#" will be created as comment lines. In order to use the # character at the start of the line as a non-comment line, use ¥# or include a space before it. #s at non-starting positions will be treated as normal characters.
- For line headers without these keywords, the line will be treated as continuance from the previous line. (While converting to Spec Sheet, in order to add a new line, type "¥¥")
- Certain Keywords can be omitted if they are not required.

General Keywords

- @CLASSNAME:
- @BASECLASS:
- @AUTHOR:
- @EMAIL:
- @DATE:
- %VERSION:
- @BRIEF_DESCRIPTION:

@CLASSNAME line is to define the Reactor Classname. The filename of the Reactor MUST BE ***CLASSNAME*.** **in any case.**

@BASECLASS line is to define the base class for that class. In most cases, FluxReactor is defined.

@AUTHOR, @EMAIL, @DATE lines are to define those information into the reactor.

@BRIEF_DESCRIPTION line is a single-lined definition of the Reactor. The details of these keywords are passed on to the E-CELL2 System.

Keywords for the Reactor Spec Sheet

- @DESCRIPTION:
- @EQUATION:
- %SUBSTANCE:
- @NOTES:

@DESCRIPTION line is to define the full description of that Reactor.

@EQUATION line is to define the equation in LaTeX displaymath format. Thus:

1. $$ Write Equation$$
2. \begin{displaymath} Write Equation \end{displaymath}
3. \[Write Equation\]

choose any of the above.

%SUBSTANCE line is to define the Substance from Substrate, Product, Catalyst, and Effector, its maximum amount, minimum amount and their comments separated by commas ",". The max/min amounts can be defined in decimal integers, but it is also possible to set the maximum as "Inf" for infinite. Comments can be used as a section to describe special Substances. As of now, the information here will only be reflected to the Spec Sheet.

Ex: "%SUBSTANCE:Substrate, 1, 10" means that the amount of the substrate is between 1 and 10.

@NOTES line is to define any comments that need to be noted.

Reactor Source Code Keywords

- %PARAMETER:
- %INCLUDE_FILE_H:
- @PRIVATE:
- @PROTECTED:
- @PUBLIC:
- %INCLUDE_FILE_C:
- @INITIALIZE_FUNC:
- @REACT_FUNC:
- @OPTION_C:

%PARAMETER line is to define the parameters that become the argument for that Reactor in parameter name, type, unit, comment format separated by commas ",". Input "Int" or "Float" as parameter type according to the value (which are NOT identical to the "int"s and "float"s used for C++). "Int"s in E-CELL2 Windows version are 32bit integers and "Float"s are 80bit firm decimals (but because of JNI(Java Native Interface), the accuracy is 64 bit in the GUI version and 80bit in the batch version)
%INCLUDE_FILE_H line is to define the .h filename to be included in the reactor. Do not embrace file names with < > or " ".
@PRIVATE line is to define private items used.
@PROTECTED line is to define restricted items used.
@PUBLIC line is to define the public items used.

If FluxReactor.h is included in the file, Reactor.h, Reactant.h, and RootSystem.h are all automatically included in the file are need not to be included. Note that Stepper.h used in E-CELL1 is not used in E-CELL2. If not including FluxReactor.h, including StandardHeaders.h is desired in ordered to include the former three header files.

@OPTION_C line is to define new methods and macros. The data here will be appended to the top of the .cpp file.

@INITIALIZE_FUNC line is to define the C++ code to be executed only once at the initialization point. This line can be used to set the non-changing elements and to check parameter ranges.

@REACT_FUNC line is to define the process that must be done for each calculation step. As default:

1. Calculate Reaction Velocity
2. Alter the substance value according to the Velocity

if including the FluxReactor, **process() method can be used to easily perform the 2. task. Argument for the process() method is the molecular reacting amount per second(in Float).**

The activity that the reactor handle is the value per single step, but the value shown in the reactor window are values per second.

KEYWORD	.tex	.cpp	.h	format	required
@CLASSNAME	O	O	O	.	O
@BASECLASS	O	O	O	.	O
@AUTHOR	O	(*1)	(*1)	.	.
@EMAIL	O	(*1)	(*1)	.	.
@DATE	O	(*1)	(*1)	.	.
%VERSION	O	O	O	.	.
@BRIEF_DESCRIPTION	O	.	O	.	.
@DESCRIPTION	O	.	.	TEX	.
@EQUAION	O	.	.	TEX	.
%SUBSTANCE	O	.	.	.	O(*2)
%PARAMETER	O	O	O	.	O(*2)
@NOTES	O	.	.	TEX	.
%INCLUDE_FILE_H	.	.	O	.	.
@PRIVATE	.	.	O	C++	.
@PROTECTED	.	.	O	C++	.
@PUBLIC	.	.	O	C++	.
%INCLUDE_FILE_C	.	O	.	.	.
@INITIALIZE_FUNC	.	O	.	C++	O
@REACT_FUNC	.	O	.	C++	O
@OPTION_FUNC	.	O	.	C++	.

Figure: Keywords (target file, format)
(*1) will be treated as comments
(*2)Dependent on the **Reactor**

Method	Definition
N_A	Avogadro amount
deltaT()	Step range (E-CELL1: supersystem()->stepper()->deltaT)
numSubstrate()	Substrate amount
numProduct()	Product amount
numCatalyst()	Catalyst amount
numEffector()	Effector amount

substrate(i)->coefficient()	*i*th Substrate stoichiometry
substrate(i)->concentration()	*i*th Substrate concentration in M
substrate(i)->quantity()	*i*th Substrate molecular amount
substrate(i)->substance().supersystem()->volume()	System Volume of the *i*th Substrate
product(i)->coefficient()	*i*th Product stoichiometry
product(i)->concentration()	*i*th Product concentration in M
product(i)->quantity()	*i*th Product molecular amount
product(i)->substance().supersystem()->volume()	System Volume of the *i*th Product
catalyst(i)->concentration()	*i*th Catalyst concentration in M
catalyst(i)->quantity()	*i*th Catalyst molecular amount
catalyst(i)->substance().supersystem()->volume()	System Volume of the *i*th Catalyst
effector(i)->concentration()	*i*th Effector Concentration in M
effector(i)->quantity()	*i*th Effector molecular amount
effector(i)->substance().supersystem()->volume()	System Volume of the *i*th Effector

Figure: Method list

2.2 Reactor Description Example

In order to create a Reactor file for the kinetic equation such as one below:

$$v = \frac{(K_{cF} K_p [S] - K_{cR} K_s [P])[E]}{K_s [P] + K_p [S] + K_s K_p}$$

we shall name the Reaction "MichaelisUniUniReversibleReactor" and the keyword definition will be such as the following:

```
@CLASSNAME:MichaelisUniUniReversibleReactor

@BASECLASS: FluxReactor
@AUTHOR: Kouichi Takahashi
@EMAIL: shafi@sfc.keio.ac.jp
@DATE: 1999 2/22

%VERSION: ecs-v08, 0.1

@BRIEF_DESCRIPTION:Simple Henri-Michaelis-Menten UniUni Reversible kinetics.
@DESCRIPTION:Simple Henri-Michaelis-Menten UniUni Reversible kinetics.
@EQUATION:$$v=\frac{(K_{cF} K_p [S]-K_{cR} K_s[P])[E]}{K_s[P]+K_p[S]+Ks Kp}$$

%SUBSTANCE:Substrate, 1, 1
%SUBSTANCE:Product, 1, 1
%SUBSTANCE:Catalyst, 1, 1
```

```
%SUBSTANCE:Effector, 0, 0

%PARAMETER: Ks, Float, mol/l, Michaelis Constant of Substrate
%PARAMETER: Kp, Float, mol/l,Michaelis Constant of Product
%PARAMETER: KcF, Float, mol/l, Catalytic Constant (Forward)
%PARAMETER: KcR, Float, mol/l,Catalytic Constant (Reverse)

@PRIVATE: Float Ksp;

@INITIALIZE_FUNC:
Ksp = Ks * Kp;
@REACT_FUNC:
Float S = substrate(0)->concentration();
Float P = product(0)->concentration();
Float E = catalyst(0)->quantity();

Float velocity = (KcF * Kp * S - KcR * Ks * P) * E / (Ks * P + Kp * S + Ksp);

process(velocity);
```

For Ks and Kp are both parameters in the equation, The product (Ks * Kp) will always be constant. Thus, the product of this function need only be calculated once at the initialization point. Declaring Ksp as a @PRIVATE function:

```
@INITIALIZE_FUNC
Ksp = Ks * Kp;
```

In the @REACT_FUNC line, variable S must be declared to describe Substrate concentration. Thus:

```
@REACT_FUNC
Float S=Substrate(0)->concentration();
```

P and E must identically be declared.

3 Conversion to Reactor Source Code (.C .h file) and DLL

To create Reactor source and executable (DLL) modules for E-CELL2 system, first press the Reactor tab of the ModelingLauncher. By selecting the .rd for input and pressing the "Execute" button, the launcher will automatically create the source code and DLL files by compiling. Refer to the ModelingLauncher tutorial in Chapter 6 for more details.

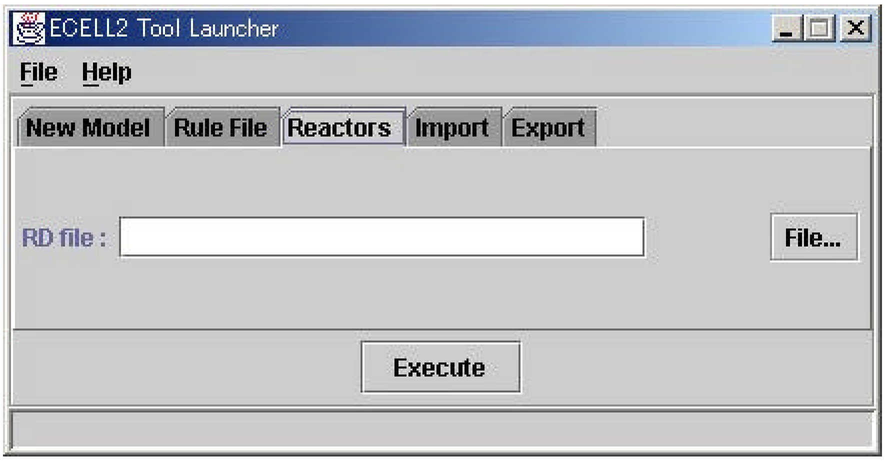


Figure: Modeling Launcher

4 Load the E-CELL2 System

In order for E-CELL2 to be executed, the following precautions must be noted. The Reactor (dll files) defined in the Rule must be placed under the Reactor Directory (As default, DLLR and DLLRB are set in ECELL2.BAT and ECELL2BB.BAT). Normally the created reactor dlls will be placed in these directories. E-CELL2 will terminate with an error message if the designated Reactor itself or the path do not exist.

5 General Reactors from the E-CELL2 Project

Reactor Classname	Reversible/Irreversible	Reaction
ZeroReactor	I	zero order reaction
MassActionReactor	I	1st order reaction
MichaelisUniUniReactor	I	follow MichaelisMenten
MichaelisUniUniReversibleReactor	R	follow MichaelisMenten
RapidEquilibriumReactor	R	rapid equilibrium
ConstantParameterReactor	I	Constant
OrderedUniBiReactor	R	.
OrderedBiUniReactor	R	.
OrderedBiBiReactor	R	.
PingPongBiReactor	I	.
PingPongBiBiReactor	R	.
RandomUniBiReactor	R	.
RandomBiUniReactor	R	.
RandomBiBiReactor	R	.
RapidEquilibriumReactor	R	.
IsoUniUniReactor	R	.
CatalyzedMassActionReactor	R	.
DecayReactor	I	.

GeneralRapidEquilibriumPReactor	-	.
RapidEquilibriumPReactor	-	.

Figure: Standard Reactor

?

- Standard Reactor Specifications (Adobe Portable Document Format(PDF))

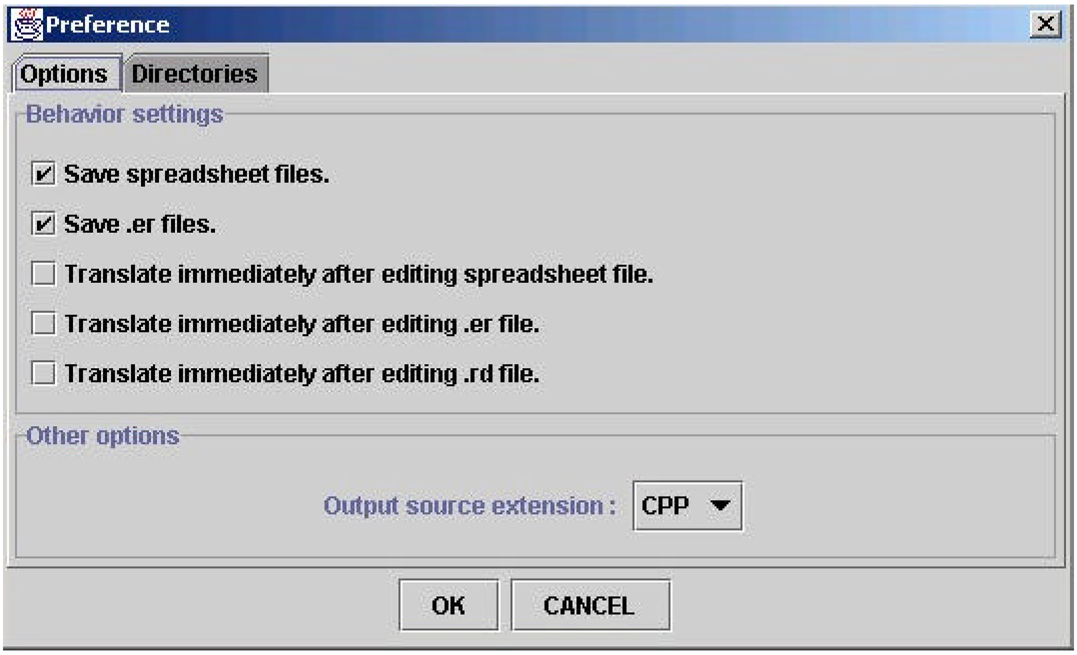
Acrobat Reactor is required to view a pdf file.

Chapter 6 Modeling Launcher

1 Initial Settings (Preference)

1.1 Options Tab (Options)

1.1.1 Behavior settings



Preference

Options | Directories

Behavior settings

☑ Save spreadsheet files.

☑ Save .er files.

□ Translate immediately after editing spreadsheet file.

□ Translate immediately after editing .er file.

□ Translate immediately after editing .rd file.

Other options

Output source extension : CPP

Figure 1: Preference Options

- Save spread sheet files.
  If checked, spreadsheet files will be saved when creating .er files (Deleted if not checked)
- Save .er files.
  If checked, .er files will be saved when creating .eri files (Deleted if not checked)
- Translate immediately after editing spread sheet file.
  If checked, .eri creation process will start immediately after the spreadsheet file is created.
- Translate immediately after editing .er file.

If checked, .eri creation process will start immediately after the .er file is created.
- Translate immediately after editing .rd file.
  If checked, Reactor source files and DLLs will be created immediately after .rd files are created.

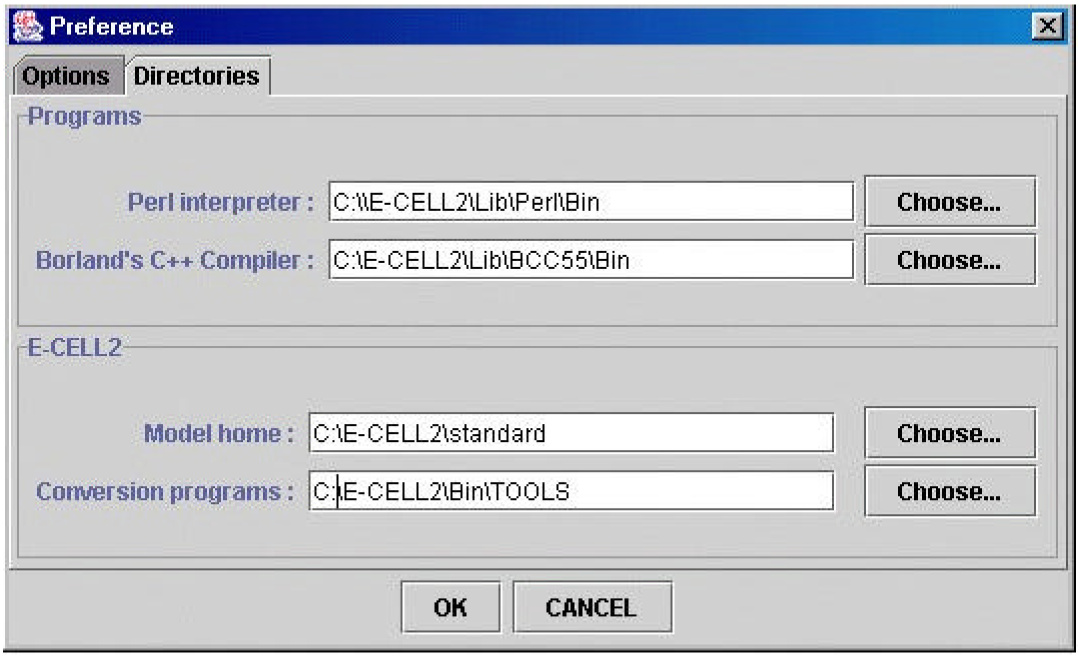
While *.er files are the RULE files of E-CELL2 written in text format, *.eri files are in E-CELL2 executable format. Details can be found in Chapter 4 of the user's manual (Creating Rules section).
*.rd files are the Reactor files of E-CELL2. Details can be found in Chapter 5 of the user's manual (Creating Reactors section).

1.1.2 Other settings

- Select the Reactor source file extension from CPP or C. *In default ,use CPP in Windows.

1.2 Directory Settings (Directories)

Sets the paths to the programs and scripts, and the directories for saving the files. A warning message will be displayed if the target directory do not exist. If left blank, the directory of this launcher program will be used.



Programs

Perl interpreter : C:\\E-CELL2\Lib\Perl\Bin

Borland's C++ Compiler : C:\E-CELL2\Lib\BCC55\Bin

E-CELL2

Model home : C:\E-CELL2\standard

Conversion programs : C:\E-CELL2\Bin\TOOLS

Figure 2: Preference Directories

1.2.1 Path settings (Programs)

- Perl interpreter
  Set the path to Perl interpreter.
- Borland's Make
  Set the path to Borland's C++ Compiler.

1.2.2 Path settings for the Model and Script files (E-CELL2)

- Model
  Set the directory to create Rules and Reactors. In default, STANDARD under E-CELL2's

installation directory is used.
- Conversion programs
  Set the directory where the conversion programs are located. In default, BIN/TOOLS under E-CELL2's installation directory is used.

2 Model wizard(New Model) tab

Creates a new model directory.

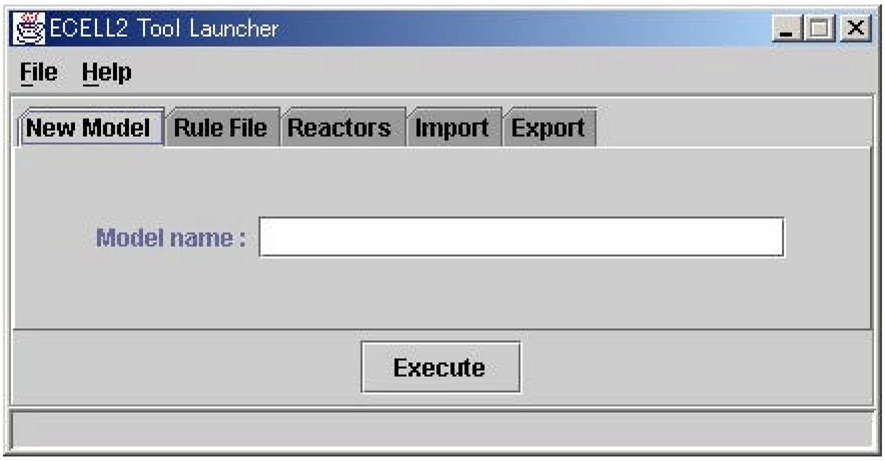


Figure 3: New Model Wizard

Input the name of the model to create in the input field and press Execute to copy the temporally stored data (under Template directory) to that directory. The Model directory set at the Preference window will be changed with the execution of this process.

3 Rule file wizard (Rule file) tab

Creates an ERI file from a spreadsheet or an ER file.

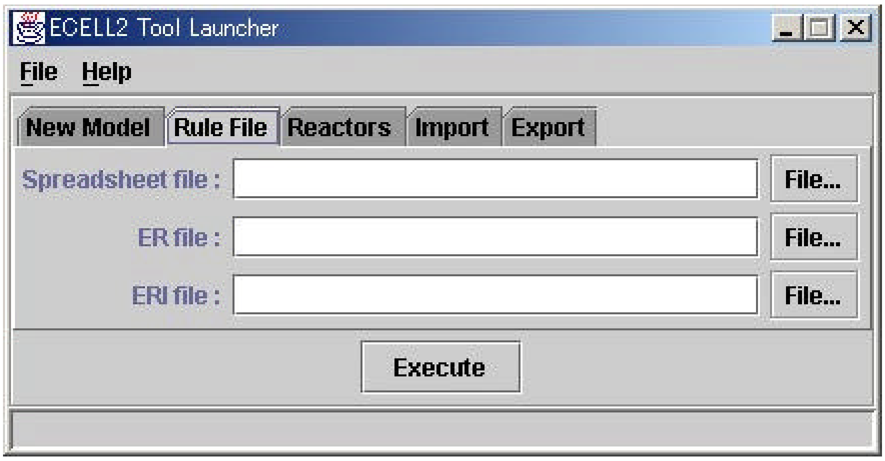


Figure 4: Rule File Wizard

3.1 Required Input

3.1.1 Spreadsheet file

Select the spreadsheet file via its path by directly typing its location or by browsing for the target by the "File..." button on the right. If the path is omitted, the Model directory set at the Preference window will be used. The process will terminate with a warning message if the target file do not exist.

- "File..." menu
  The following can be performed:

1. Select the target file(Choose...)
   A file selection dialog will open and users are to select the target file to open. The selected file will be displayed in the input field.
2. Edit existing files (Edit...)
   A file selection dialog will open and users are to select the target file to edit. The selected file will be displayed in the input field.
3. Create new file (New...)
   This will create new spreadsheet files to be edited. . The selected file will be displayed in the input field.

3.1.2 .er file(ER file)

Select the .er file via its path by directly typing its location or by browsing for the target by the "File..." button on the right. If the path is omitted, the Model directory set at the Preference window will be used.

- "If the spread sheet is selected"
  Spreadsheet -> .er -> .eri conversion will be performed. .er files do not need to be selected. If blank, the output files in spreadsheet filename+.er will be created under the Model directory set at the Preference window. A warning message will be displayed on overwriting existing files.
- "If the spreadsheet is not selected"
  .er -> .eri conversion will be performed. .er files need to be selected. Conversion process will terminate with a warning message on assigning files or directories that do not exist.

- "File..." menu
  The following can be performed:

1. Select the target file(Choose...)
   A file selection dialog will open and users are to select the target file to open. The selected file will be displayed in the input field.
2. Edit existing files (Edit...)
   A file selection dialog will open and users are to select the target file to edit. The selected file will be displayed in the input field.
3. Create new file (New...)
   This will create new spreadsheet files to be edited. The selected file will be displayed in the input field.

3.1.3 .eri file (ERI file)

Select the .eri file or the path itself by directly typing its location or by browsing for the target by the "File..." button on the right. If the path is omitted, the Model directory set at the Preference window will be used. If only the path is designated, files will be created with the input spreadsheet (or the .er) file name with an .ERI extension. Model directory set at the preference window will be used if the field is left blank. A warning message will be displayed on overwriting existing files. The process will terminate with a warning message if the set directory do not exist.

- "File..." menu
  The following can be performed:

1. Select the target file(Choose...)
   A file selection dialog will open and users are to select the target file to open. The selected file will be displayed in the input field.

3.2 Process Scheme

3.2.1 Select the spreadsheet or .er file for input

Choose the input file under the following rules:

1. If the Spreadsheet is selected
   Spreadsheet -> .er file -> .eri file conversion will be executed. A warning message will be displayed if the selected spreadsheet do not exist.
2. If the Spreadsheet is not selected
   .er file selection is required for it will become the input for the scheme.
3. Selecting .er files or their paths
   - If the Spreadsheet is selected
     Spreadsheet -> .er file -> .eri file conversion will be executed according to the data inputted. The output will be directory name + spreadsheet name + .ERI extension. A warning message will be displayed if the selected spreadsheet do not exist.
   - If the Spreadsheet is not selected
     .er file is required for it will be the input for the scheme. .er file -> .eri file conversion will be executed. A warning message will be displayed if the selected .er file do not exist.
4. If neither .er files nor the directory are selected
   Spreadsheet -> .er file -> .eri file conversion will be executed according to the data inputted. The output will be directory name + spreadsheet name + .ERI extension under the Model directory set at the preference window. If neither the spreadsheet file nor .er files are selected, a warning message will be displayed.

If either "Translate immediately after editing spreadsheet" or "Translate immediately after editing .er file" is checked, the .eri conversion process will start immediately after each editing window is closed.

3.2.2 Set output rule file and directory

Set the filename or the directory name for the output .eri files according to the data inputted. .ERI files with inputted filename and directory (if any) will be created. If left blank, the file will be created under the Model directory set at the preference window. If the selected filename do not exist, a new file will be created with that name. A warning message will be displayed if the selected directory do not exist.

3.2.3 Batch Process

Clicking the Execute button starts the conversion process. The log of the process will be displayed in the message window during the process with errors if any (where the process will be terminated at that point). If the error occurred while script execution, the error details will be in the message window.

- If "Save spreadsheet file" in the Preference window is not checked, the spreadsheet file of input will be deleted in the spreadsheet -> .er conversion process.
- If "Save .er file" in the Preference window is not checked, the .er file will be deleted in the .er file -> .eri file conversion process.

4 Reactor wizard tab (Reactors)

Creates the Reactor source files (C/C++ and header files) and DLL from .rd files.

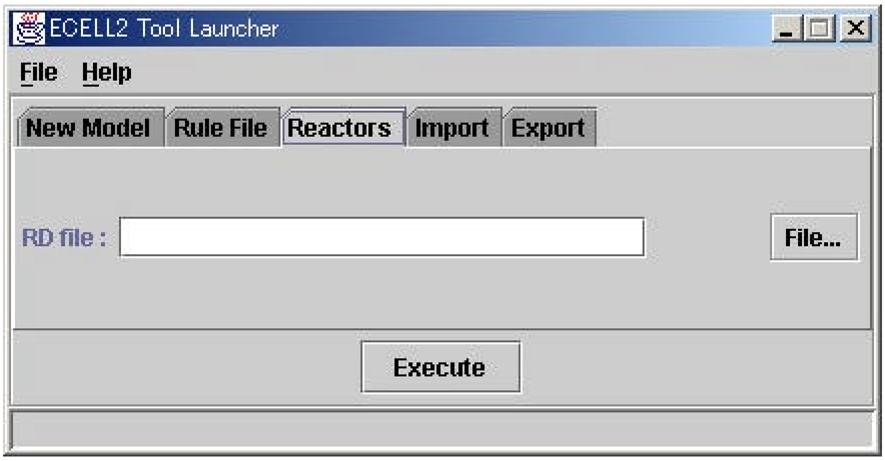


Figure 5: Reactors Wizard

4.1 Required Input

4.1.1 .rd file (RD file)

Select the .rd file via its path by directly typing its location or by browsing for the target by the "File..." button on the right. If the path is omitted, the RD directory under Model set at the Preference window will be used. The process will terminate with a warning message if the target file do not exist. If the filename is omitted, all .rd files under the target path will be converted.

- "File..." menu
  The following can be performed:

1. Select the target file(Choose...)
   A file selection dialog will open and users are to select the target file to open. The selected file will be displayed in the input field.
2. Edit existing files (Edit...)
   A file selection dialog will open and users are to select the target file to edit. The selected file will be displayed in the input field.
3. Create new file (New...)
   This will create new spreadsheet files to be edited. The selected file will be displayed in the input field.

4.2 Process Scheme

4.2.1 Designating the .rd files for input

Designate the input .rd file via its path. A warning message will be displayed during the conversion process if non-existing files are selected. The RD directory under the Model directory designated under the Preference window will be chosen if the path information is not included in the input.

If a directory name is selected, all .rd files under that directory will be converted. If no input, the RD directory under the Model directory in the Preference window will be the target.

If "Translate Immediately after editing .rd file" is checked in the Preference window, Reactor file conversion process will start immediately after closing the editor window.

4.2.2 Batch Process

Conversion Process will start when the Execute button is pressed. Log will be displayed in the message window during the conversion process. The process will terminate with a warning message if an error occurs during the process. If the error occurred when loading script, error details will be displayed in the message window.

Note that the Makefile for creating Reactor modules is required for this conversion process to succeed. A warning message will be displayed if this Makefile do not exist under the DLL output directory set at the Preference window. Please also make sure that the permission settings for the Makefile is set in writable mode, for the list of the Reactor modules (DLL) to be created will be appended to the file under the DLLFILESR section.

This process creates the Reactor source files in the format set at "Output source Extension" in the Preference window. The extensions for the header files are always .h.

Reactor Source files and DLL files will be saved under the directories set at the Preference window.

5 Import wizard tab (Import)

Imports models that are created externally to this system.

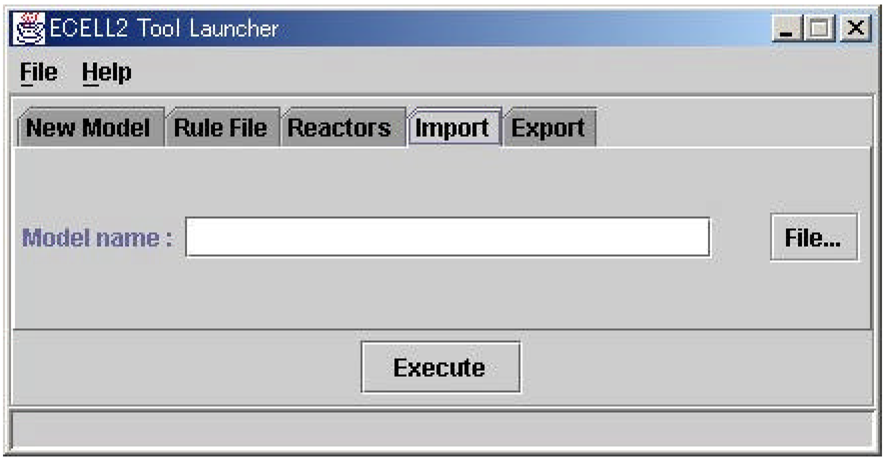


Figure 6: Import Wizard

5.1 Required Input

5.1.1 Model name

Designate the location of the .jar file of the model to import via its path, by directly typing its location or by browsing for it using the "File..." button on the right. The installation directory of E-CELL2 will be chosen if only a file name is designated.

- "File..." button menu
  If this menu is selected, a directory selection dialog will appear with directories that can be selected and designated. The selected directory name will be directly reflected in the input field on the left.

5.2 Process Scheme

5.2.1 Designating the .jar file (Java Archive file) for imput

Designate the input .jar file via its path. If a nonexistent file name is designated, a warning message will be displayed upon execution, and the import process will not be executed.

5.2.2 Batch Process

The import process will start when the Execute button is clicked. A log will be displayed in the message window during the import process. If an error occurs during the import process, a warning message will be displayed and the process will be terminated.

This process creates a new directory with the name of the imported model, under the installation directory of E-CELL2.

6 Export wizard tab (Export)

Exports models that are created by this system so that they can be loaded by other E-CELL2 users.

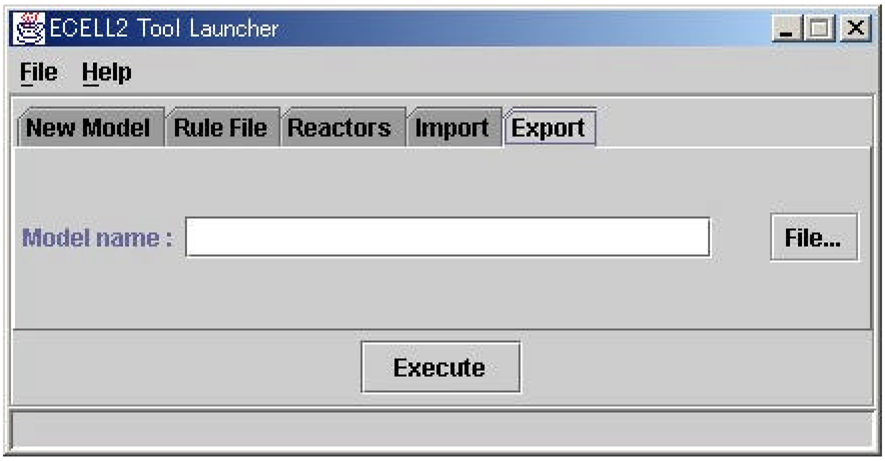


Figure 7: Export Wizard

6.1 Required Input

6.1.1 Model name

Designate the directory of the model to export by directly typing its location or by browsing for it using the "File..." button on the right.

- "File..." button menu
  If this menu is selected, a directory selection dialog will appear with directories that can be selected and designated. The selected directory name will be directly reflected in the input field on the left.

6.2 Process Scheme

6.2.1 Designation of the model to export

Designate the directory of the model to export. If a nonexistent file name is designated, a warning message will be displayed upon execution, and the export process will not be executed.

6.2.2 Batch Process

The export process will start when the Execute button is clicked. A log will be displayed in the message window during the export process. If an error occurs during the export process, a warning message will be displayed and the process will be terminated.

This process creates a model archive file in ".jar" format under the installation directory of E-CELL2.

7 Editor/Viewer Window

Two simple editors and one viewer window is included in this launcher.

- Spreadsheet Editor
- Text Editor (for .er and .rd files)
- Message Viewer

Each window has a common command set of the following:

1. Save File

Saves the edited (or displayed) text onto a file. If the file was newly created, a file selection dialog will open during save. The editing file will be the save target elsewise. When in message viewer mode, the displayed text file will be saved. A file selection dialog will always appear in this mode. A warning message will be displayed on overwriting existing files.

2. Refresh

Performs an "undo" and refreshes the data back to its saved or last opened position. When in message viewer mode, any displayed message will be cleared. If any changes were to be made to the saved data, A warning message will be displayed before altering the data.

7.1 Spreadsheet Editor

Simple editor to edit CSV files in tab format.

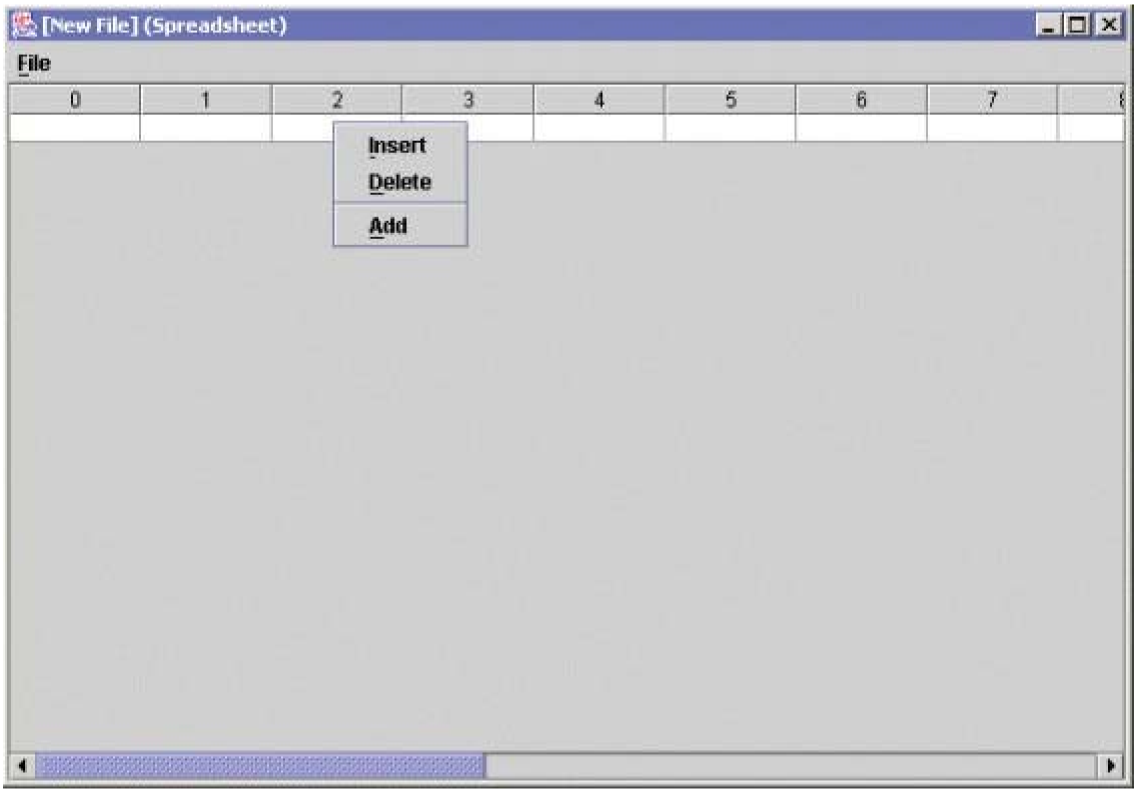


Figure 8: Spreadsheet Editor

Right-clicking inside the cell displays the following menu:

- Insert (Insert row)
  Inserts a new row onto that position.
- Delete (Delete row)
  Deletes the row of that position
- Add (Add new row)
  Adds a new row under the selected row.

7.2 Text Editor

Simple editor for editing files in text format.

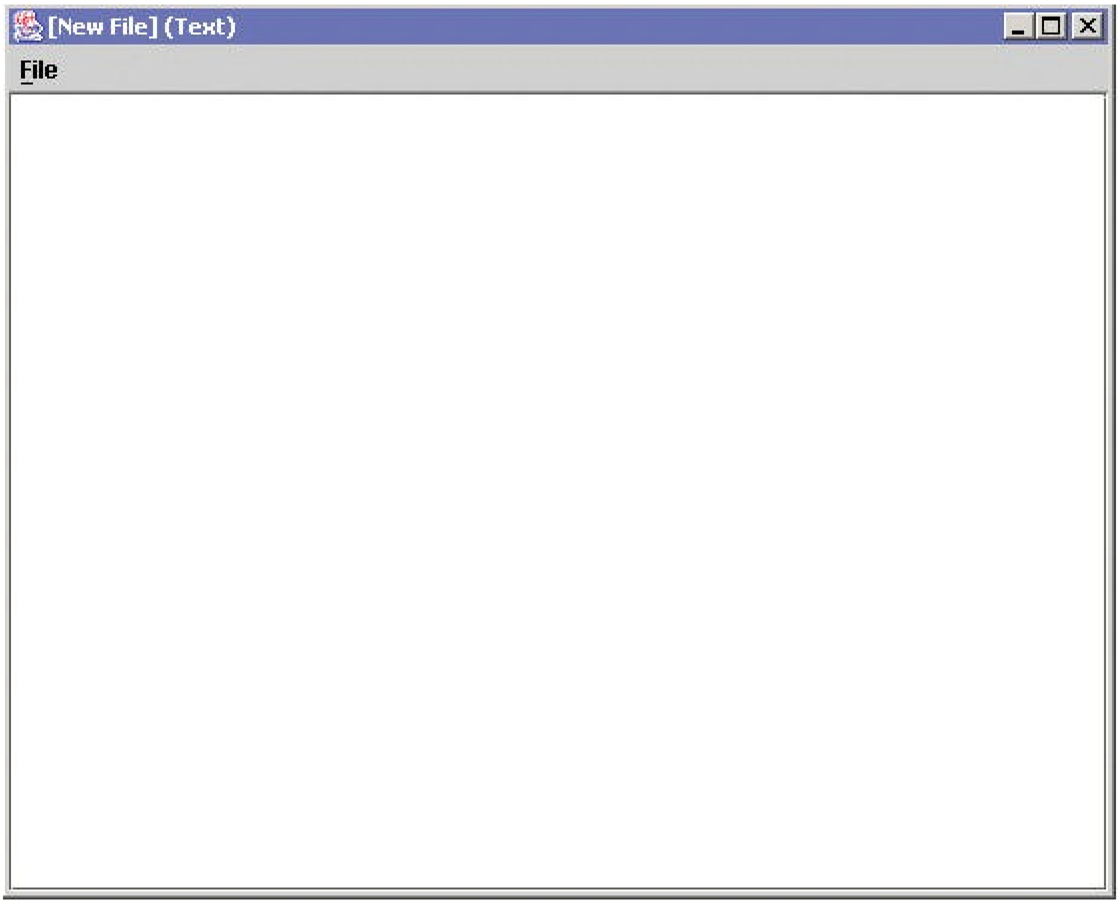
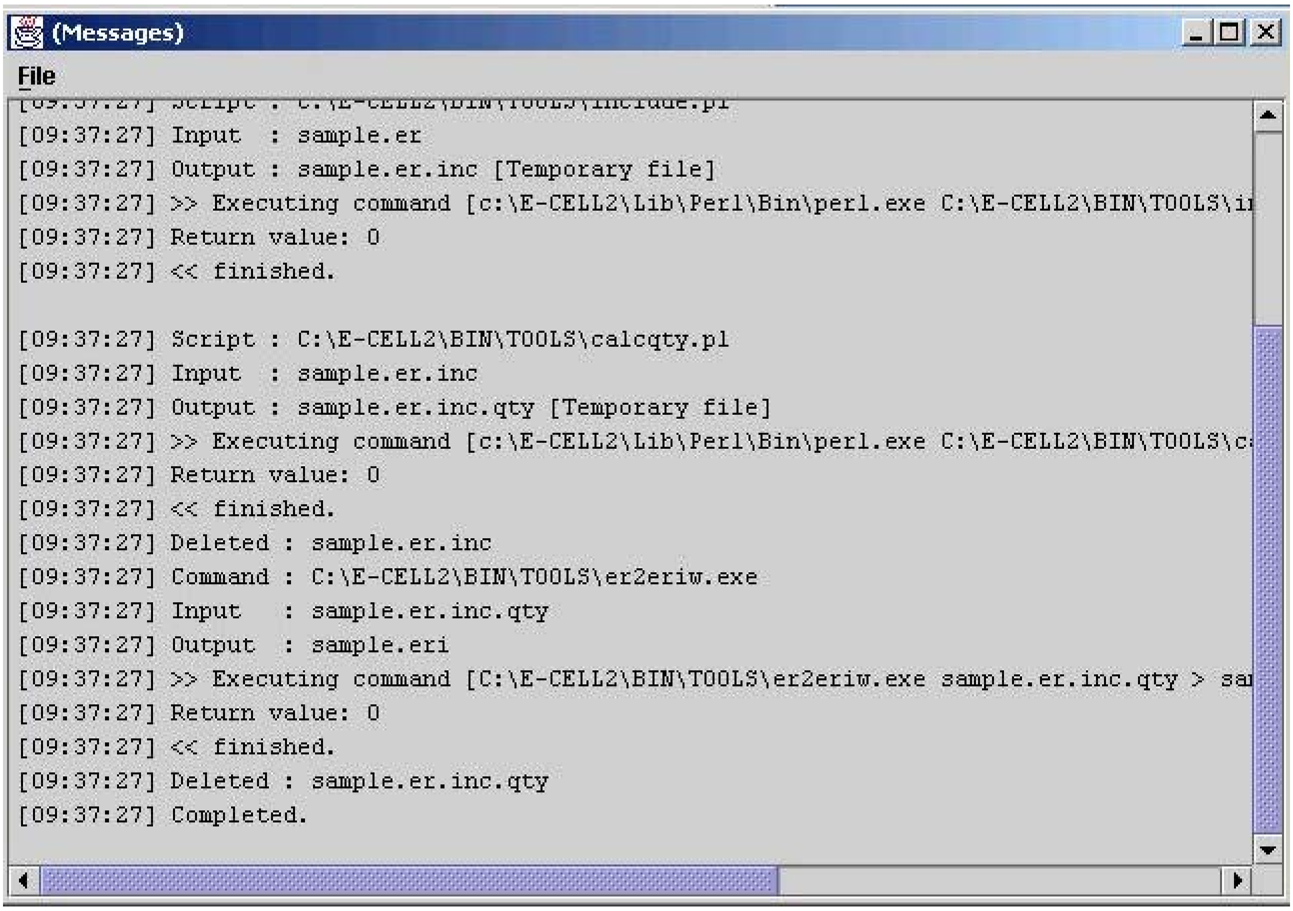


Figure 9: Text Editor

7.3 Message Viewer

Message window to display errors and process logs. Read only.



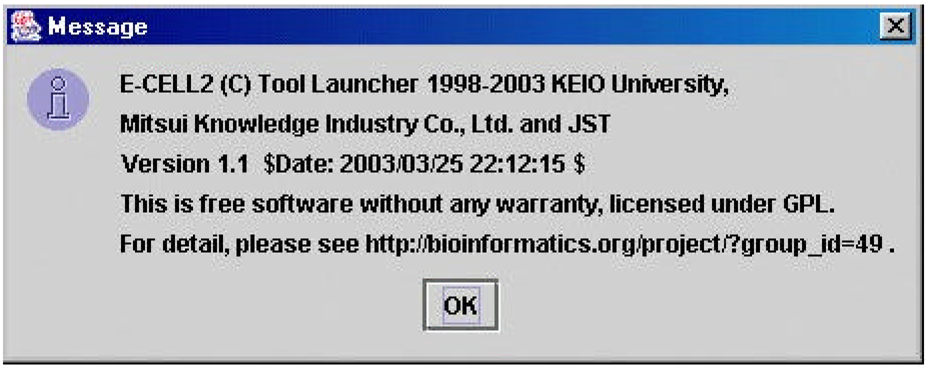
```
[09:37:27] Script : C:\E-CELL2\BIN\TOOLS\include.pl
[09:37:27] Input  : sample.er
[09:37:27] Output : sample.er.inc [Temporary file]
[09:37:27] >> Executing command [c:\E-CELL2\Lib\Perl\Bin\perl.exe C:\E-CELL2\BIN\TOOLS\i
[09:37:27] Return value: 0
[09:37:27] << finished.

[09:37:27] Script : C:\E-CELL2\BIN\TOOLS\calcqty.pl
[09:37:27] Input  : sample.er.inc
[09:37:27] Output : sample.er.inc.qty [Temporary file]
[09:37:27] >> Executing command [c:\E-CELL2\Lib\Perl\Bin\perl.exe C:\E-CELL2\BIN\TOOLS\c
[09:37:27] Return value: 0
[09:37:27] << finished.
[09:37:27] Deleted : sample.er.inc
[09:37:27] Command : C:\E-CELL2\BIN\TOOLS\er2eriw.exe
[09:37:27] Input   : sample.er.inc.qty
[09:37:27] Output  : sample.eri
[09:37:27] >> Executing command [C:\E-CELL2\BIN\TOOLS\er2eriw.exe sample.er.inc.qty > sa
[09:37:27] Return value: 0
[09:37:27] << finished.
[09:37:27] Deleted : sample.er.inc.qty
[09:37:27] Completed.
```

Figure 10: Message Viewer

8 Other Settings

8.1 Current Version

Displays the version of this launcher.



E-CELL2 (C) Tool Launcher 1998-2003 KEIO University,
Mitsui Knowledge Industry Co., Ltd. and JST
Version 1.1 $Date: 2003/03/25 22:12:15 $
This is free software without any warranty, licensed under GPL.
For detail, please see http://bioinformatics.org/project/?group_id=49 .

Figure 11: Version Information

E-CELL2 User's Manual
Last Update $Date: 2003/3/25 $.