30th Annual International IEEE EMBS Conference
Vancouver, British Columbia, Canada, August 20-24, 2008

A Platform for *in silico* Modeling of Physiological Systems II. CellML Compatibility and Other Extended Capabilities

Yasuyuki Suzuki, Yoshiyuki Asai, Toshihiro Kawazu, Masao Nakanishi, Yoshiki Taniguchi, Eric Heien, Kenichi Hagihara, Yoshihisa Kurachi, and Taishin Nomura

Abstract—**The number of biological models published in peer reviewed journals and complexity of each of those models are rapidly increasing, making it difficult to reproduce simulation results of the published models and to reuse the models by third persons. This paper is a continuation of our previous report on a software platform development as a solution to such difficulties. We describe progresses of our development. Those include improvement in functional capabilities to import and simulate published models in the CellML model repository, to browse and edit CellML models and then to export them as new models either with the CellML format or with a XML format defined for our platform (ISML), and to newly construct large scale models by connecting CellML/ISML models. Several advantages to use ISML in parallel with CellML are; 1) ISML can deal with geometry (morphology) of a model, enabling the user to perform geometry dependent modeling and simulations. 2) ISML can deal with time series data, both simulated and experimentally acquired data, for visualization of dynamics.**

I. Introduction

PHYSIOME and systems biology have been recognized as emerging and important research areas that can integrate growing experimental data at multiple levels and scales of the human body for better understanding of human physiological functions [1,2]. Mathematical models, in particular dynamical system models, of physiological functions play a key role to integrate the vast amount of pieces of knowledge. This can be argued with the following reasons. 1) Mathematical models are capable of describing time evolution of states of biological systems based upon physical and chemical principles or phenomenological logics governing behaviors of the system. Solutions of a given system can be associated quantitatively with changes in time of experimentally observed biological signals and images. 2) Biophysical models require morphology and geometry of biological organs and organelles with high-precision measurement in order to perform accurate numerical simulations of the model, by which structure-function linkage can be quantitatively revealed. 3) A mathematical system model can be divided into subsystems (*modules*), each of which may correspond to biological entities at different levels and scales, leading to understanding hierarchical mechanisms of physiological functions.

However, the number of mathematical models describing biological functions published in peer reviewed journals is rapidly increasing. Moreover, complexity of each of those published models increases as the computational performance increases. These make it difficult to reproduce simulation results of the published models and to reuse the models by third persons, apparently obstructing promotion of sciences and the knowledge integration. The pioneering effort to overcome this problem has been promoted by CellML project [3,4] as a part of IUPS Physiome Project. CellML is a XML based markup language which aims at describing mathematical models of biological functions such as electrical activities of cell membranes. Each model written in the CellML format can be downloaded from the CellML model repository, and includes all information necessary to reproduce simulated results described in the corresponding publication about the model. CellML project provides a software called PCEnv to simulate dynamics of the CellML formatted models.

This paper is a continuation of our previous report [5] on a comparable and complementary effort to CellML project. We have also proposed a XML based model description format referred to as *insilico*ML (abbreviated as ISML). For ISML, *insilico*IDE is used to support the user for model construction (generation of ISML formatted models) and for automatic generation of C++ source codes to numerically simulate dynamics of the models. The *insilico*IDE, together with model databases, is on the development to play a role as a platform for *in silico* modeling of physiological systems. ISML and *insilico*IDE are CellML compatible and have several additional unique features. In this paper, we describe progresses of our platform development. Those include improvement in functional capabilities to import and simulate published models in the CellML model repository, to browse and edit CellML models and then to export them as new

Manuscript received April 7, 2008. This work was supported in part by the Global COE Program "in silico medicine" at Osaka University. Y. Suzuki, T. Kawazu, M. Nakanishi, Y. Taniguchi, and T. Nomura are with Graduate School of Engineering Science and The Center for Advanced Medical Engineering and Informatics at Osaka University, Toyonaka, Osaka, 560-8531 Japan.(phone: +81-6-6850-6534; fax: +81-6-6850-6534;e-mail: suzuki@bpe.es.osaka-u.ac.jp, kawazu@bpe.es.osaka-u.ac.jp, nakanishi@bpe.es.osaka-u.ac.jp, ytaniguchi@bpe.es.osaka-u.ac.jp, taishin@bpe.es.osaka-u.ac.jp). Y. Asai is with The Center for Advanced Medical Engineering and Informatics at Osaka University, Toyonaka, Osaka, 560-8531 Japan.(e-mail: asai@bpe.es.osaka-u.ac.jp). E. Heien and K. Hagihara are with Graduate School of Information Science and Technology at Osaka University, Toyonaka, Osaka, 560-8531 Japan.(e-mail: e-heien@ist.osaka-u.ac.jp, hagihara@ist.osaka-u.ac.jp). Y. Kurachi is with Graduate School of Medicine and The Center for Advanced Medical Engineering and Informatics at Osaka University, Suita, Osaka, 565-0871 Japan.(e-mail: ykurachi@pharma2.med.osaka-u.ac.jp)

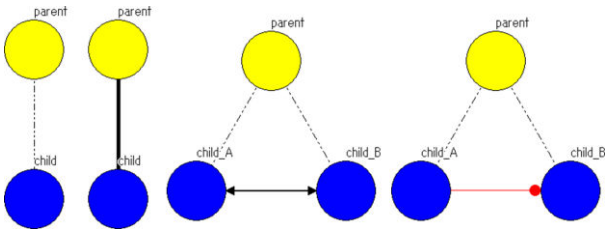

Fig. 1. Relationships among modules defined in ISML. Each ball is a module. Modules are connected by several types of lines. From left to right, it is the "include" like hierarchical structure, "constituent" like hierarchical structure, "attachment" like relationship, and "functional" relationship. The first two relationships represent hierarchical structure of modules. The attachment is represented by bidirectional arrow line. The functional relationship is uni-directional, and information flows from non-marked end to marked end.

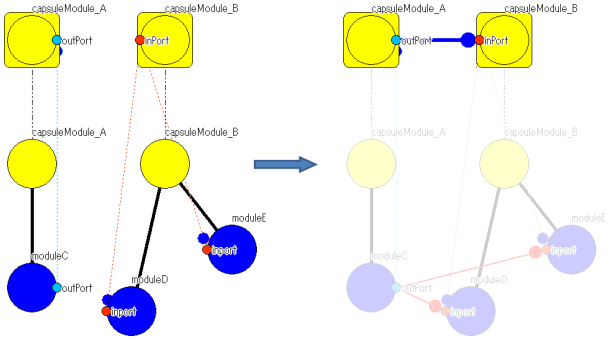

Fig. 2. Encapsulation. An encapsulated module is surrounded by a square. The user can connect encapsulated modules by a function link without taking care of detailed structure of the modules (the transparent part of the modules on the right panel).

models either with the CellML format or with ISML format, and to construct new large scale models by connecting numbers of CellML/ISML models. Several advantages to use ISML in parallel with CellML are also illustrated. Those are; 1) ISML can deal with geometry (morphology) of a model, enabling the user to perform geometry dependent modeling and simulations with the morphology of the models. 2) ISML can deal with time series data, both simulated and experimentally acquired data, for visualization of dynamics.

II. System Outline

Let us briefly summarize features of the platform that we have described previously[5]. Those include the ISML model description, the insilicoIDE (IDE), and the model database. Progresses on the CellML compatibility (functionalities to import and export CellML files) and the morphology database used with ISML are then illustrated in the following sections.

We consider a target biological system as an aggregate of elements referred to as *modules*. In ISML, each module is characterized by the XML tags such as *ID number*, *edges*, *state*, *parameters*, *dynamic-rules* and *morphology* among others. The *edges* represent the structural and functional relationships among modules. The structural relationships defined as the edge types are "*include*" and "*constituent*" to represent hierarchical relationships, and "*attachment*" to represent modules are glued each other (Fig. 1). A set of modules can form a module by declaring *encapsulation* which makes an encapsulated module independent and to be easily reused (Fig. 2). Two or more modules can be functionally related with each other by another type of the edge, the *functional link*. *Ports* of an encapsulated module are packed in a class object which has properties describing information necessary to use the module as a sub-module of larger models. The *state* and *dynamic-rules* of a module are responsible for dynamics of the module, where the *state* is time-updated using the *dynamic-rules*.

The insilicoIDE represents modules graphically as ball-like objects, where four kinds of relationships between modules are displayed as different types of lines (Fig. 1). Functional relationships connect between modules through the ports

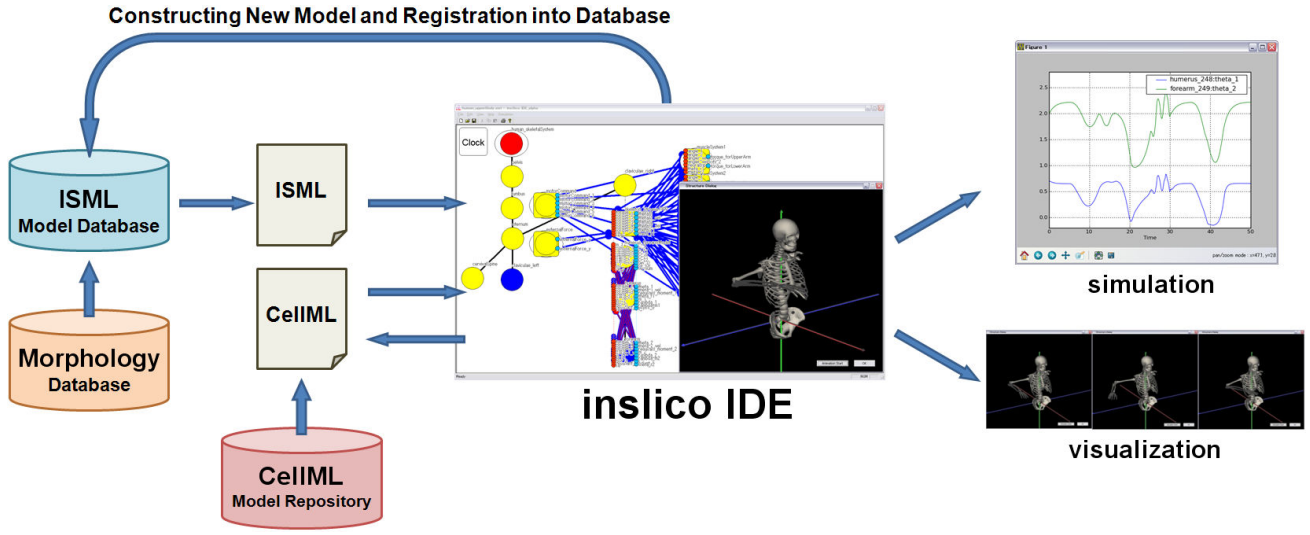

Fig. 3. Outline of the platform. The insilicoIDE can load both ISML and CellML formatted models from the model databases. The users can develop their own models on insilicoIDE and export them as either ISML or CellML models. The models developed on insilicoIDE can be simulated by automatically generating C++ (MPI C++) source code files. Simulated data or experimental data (time series) can also be loaded for visualization. Note that, when an ISML model uses geometry of biological organs, the corresponding morphological models are taken from the morphology database.

which are depicted as the small circles on the modules. Each encapsulated module is indicated by the square surrounding the ball-like module. The user can edit modules and relationships (hierarchical structure of modules) using mouse-operated GUI. For example, as shown in Fig. 2, the aggregate of modules that are appropriately encapsulated can be simply treated as a single object, and the user can use it without taking care of detailed structure and complexity of underlying modules. Once the model construction is completed, the insilicoIDE can export the model either as a C++/MPI-C++ source code or as a CellML/ISML file. The exported ISML file may be registered, after publication, to the model database to be used by other users. With the source code files, the user can perform model simulation either during IDE runtime without apparent compiling or off-line with manual compiling and execution (Fig. 3). Note that, as described in [4], ISML and IDE are capable of describing and simulating multi-agent type models as well as models described by a set of ordinary differential equations (ODEs).

III. CellML Compatibility

The CellML model repository that has been in the public domain provides more than 140 models in CellML format in the public domain. The insilicoIDE is fully compatible to CellML in the following senses. The insilicoIDE can import any model files with CellML 1.1 specification as examined by the models in CellML model repository as of March 2008, and most of the models can be simulated to exhibit reasonable dynamics. The insilicoIDE can export newly constructed models described by ODEs as CellML 1.1 formatted files. The exported model files by the IDE are validated by performing dynamic simulations of the models on PCEnv, which is the model-simulator provided by the CellML project. Figure 4 shows simulated waveforms on PCEnv for a model of neuro-musculo-skeletal system constructed on the IDE. A CellML formatted file used for this simulation on PCEnv is not from the CellML model model repository but it is

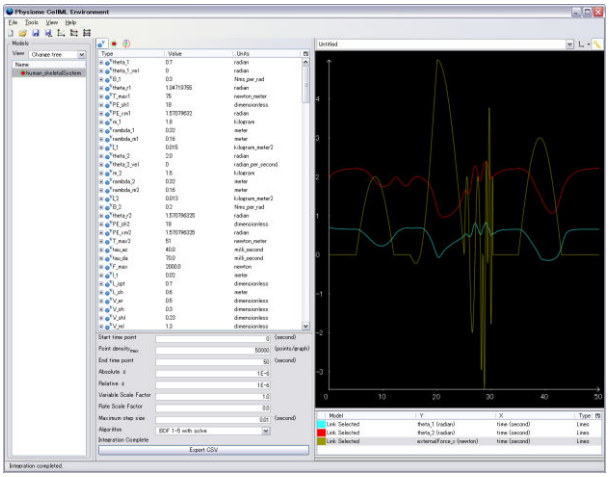

Fig. 4. Dynamic simulation of a CellML formatted model exported by the insilicoIDE on PCEnv. This graph shows a simulated result for the neuro-musculo-skeletal system. The blue line represents the rotating angle of shoulder, red line the rotating angle of elbow.

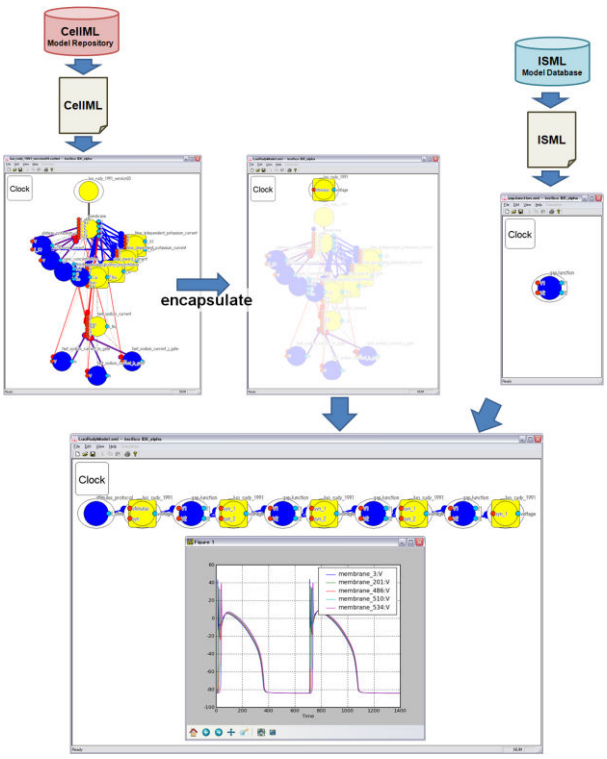

Fig. 5. Construction of a coupled cell models on insilicoIDE. A cardiac cell model and a gap-junction model are imported from, respectively, CellML and ISML databases. Copying and connecting the models allow the user to simulate the constructed model.

generated and exported by the IDE, validating the CellML file export functionality of the IDE.

Figure 5 exemplifies a collaborative simulation between CellML and ISML. In this example, a cardiac cell model (Luo and Rudy[6]) and a model of gap-junction are downloaded from, respectively, the CellML model repository and the ISML model database, and they are imported on the IDE as modules. The cell model is then encapsulated with setting its membrane potential at an output port and the stimulation current at an input port. By copying those modules and connecting between ports of the cell and gap-junction models, the user can easily simulate dynamics of the diffusively coupled cardiac cell models (a model with five cells coupled by the gap-junctions in Fig. 5).

IV. Use of Morphology and Time Series Data

In this section, we illustrate other extended capabilities of ISML and the insilicoIDE, motivating the complimentary use of this platform with CellML and its APIs. The extended capabilities include the use of a morphology database and time series data obtained in physiological experiments and/or numerical simulations of models.

Figure 6 shows snapshots of the insilicoIDE in which dynamics of a model of human neuro-musculo-skeletal system [7] are simulated. The model includes three-dimensional geometry of the skeletal system of human body, models of six muscles for an upper arm attached on the skeletal system, and neural motor controllers for the muscles, each of which is described by ISML as a reusable module and registered in the

ISML model database (see Fig. 3). The morphology of each skeletal link, in this case, is represented by a surface data (VRML), and separately stored in the morphology database to be reusable for general purposes. The ISML file of this neuro-musculo-skeletal system model is described as a set of different modules including the arm and the muscles, and has links specifically to the morphological models in the morphology database. The morphological models that can be registered in the morphological database include surface and volume data of three dimensional objects. Each morphological model is registered with a metadata file describing the metric unit of the model and keywords to characterize the model. The morphological models in the database are used not only for visualization purposes but also for dynamic simulations if a model utilizes morphometric information such as in multi-agent simulations.

Another new functionality of the insilicoIDE is to relate time series data obtained either by numerical simulations or physiological experiments to models on the IDE. When dynamics of a model are simulated using IDE, the obtained time series data usually represent changes in the *states* properties of all modules comprising the model. For experimental data that are not obtained directly by simulations on the IDE, the user specifies modules (properties of modules) that can be underlying models of the data. In order to facilitate describing relationships between a given set of time series data and the corresponding modules on the IDE, a XML based markup language, referred to as TSML, is designed, and time series data are maintained with the TSML format.

Figures 6 and 7 exemplify use cases of the functionalities of both morphology database and time series data. In Fig. 6, the simulation data obtained from the neuro-musculo-skeletal system used for Fig. 4 is represented by TSML which is imported to the IDE to show the three-dimensional human arm movement. In Fig. 7, a motion captured data of human body movement is represented by TSML. More specifically, changes in the three-dimensional positions of several markers are observed experimentally, and the motion data of the markers are saved by TSML format. The human skeletal system with markers is selected (or constructed) on the IDE as a model to be related to the data. This TSML and the model are merged on the IDE to visualize the motion.

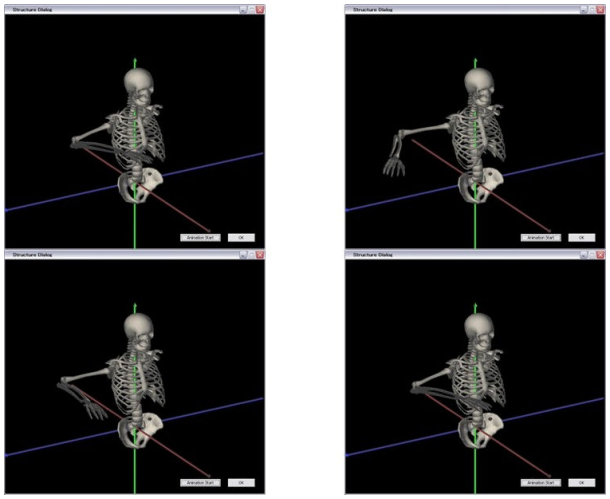

Fig. 6. Three-dimensional visualization of simulated dynamic behavior of a model using the insilicoIDE.

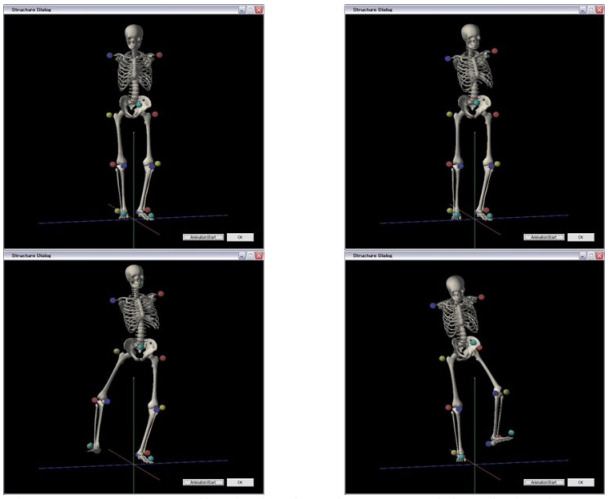

Fig. 7. An experimentally obtained motion captured data during human movement is represented by TSML (time series markup language) and imported to the insilicoIDE to visualize the motion with a three-dimensional of skeletal system model.

V. Conclusion

Several functionalities such as importing and simulating published models in the CellML model repository, browsing and editing a CellML model and then exporting it as a new model either with the CellML or ISML format, and newly constructing large scale models by connecting numbers of CellML/ISML models are demonstrated. Benefits to use both ISML and CellML are described. ISML can deal with morphology of models, enabling the user to perform geometry dependent modeling and simulations. ISML with the IDE can deal with time series data, both simulated and experimentally acquired data, for visualization of dynamics.

References

[1] J. B. Bassingthwaighte. Strategies for the Physiome Project, *Annals of Biomedical Engineering*, Vol. 28, pp. 1043-1058, 2000

[2] P. J. Hunter and Thomas K. Borg. Integration from proteins to organs: The Physiome Project. *Nature Review Mol Cell Biol*, 4(3), pp. 237-243, 2003

[3] http://www.cellml.org/

[4] C. M. Lloyd, M. D.B. Halstead, P. F. Nielsen. CellML: its future, present and past. *Prog Biophys Mol Biol*, Vol. 85, No.2-3 pp. 433-450, 2004

[5] T. Kawazu, M. Nakanishi, Y. Suzuki, T. Nomura. A Platform for in Silico Modeling of Physiological Systems. Proceedings of the 29th Annual International Conference of the IEEE EMBS Cité Internationale, Lyon, France, August 23-26, 2007, pp. 1394-1397, 2007

[6] C. H. Luo and Y. Rudy. A model of the ventricular cardiac action potential. Depolarization, repolarization, and their interaction. *Circulation Research*, Vol. 68, No. 6, pp. 1501-1526, 1991

[7] S. Stroeve, "Impedance characteristics of a neuro musculo skeletal model of the human arm", *Biological Cybernetics*, Vol. 81, pp. 475-494, 1999