

Biology Foundation Class Library

Koji Kyoda¹

kyoda@mt.cs.keio.ac.jp

Hiroaki Kitano²

kitano@csl.sony.co.jp

¹ Department of Electrical Engineering
Faculty of Science and Technology, Keio University
3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223, Japan

² Sony Computer Science Laboratory
3-14-13 Higashi-Gotanda, Shinagawa, Tokyo 141, Japan

Abstract

In this paper, we describe the Biology Foundation Class Library, which enable us to develop the biological simulations efficiently. In nature, we can see the foundation constituent elements of organisms, for example, cell, nucleus, gene, etc. Object oriented programming language is well applied into such elements. We emphasize a necessity of the class library for biological simulations and propose a principle design.

1 Introduction

The computer simulation can reproduce the phenomena in actual world and predict the results. The phenomena can be simulated a numerical calculation and a symbol manipulation. Biological phenomena can be simulated by *object oriented modeling*. We subdivide the model into some “units” and establish some constituent elements being part-whole relationship as the inner attribute. The system of the simulation is also described as state transition of each model; transition of the inner attribute variables, and we can operate the inner variables by the methods of each object.

2 Modeling and Design

A cell of an organism consists of a nucleus, cytoplasm, mitochondria, etc. Chromosome consists of genes. This hierarchical structure can be directly mapped onto object class library. The simple hierarchy is shown at Fig.1. The class of C++ programming language enable us to make modeling the objects. *Gene*, *Chromosome*, *Cell* class and *Protein* class are major classes in the library.

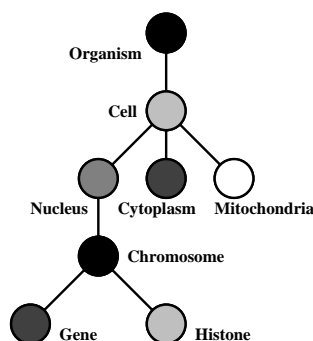


Fig.1: The simple hierarchy of the organism

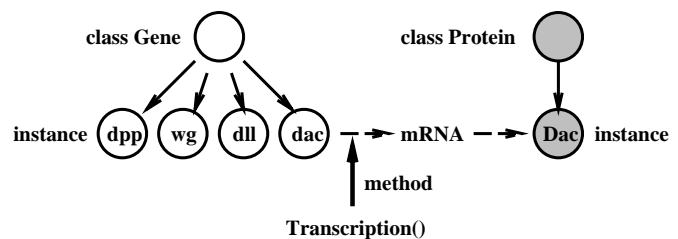


Fig.2: The relation between the class and the method

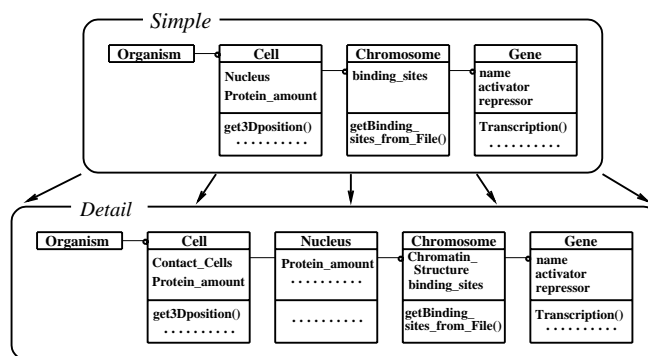


Fig.3: The simple and detail class

For example, we create many cells by making use of the *Cell* class. We can create many cells without programming all cell's information. Because we use the class and its methods. Fig.2 shows the relation between the class and the method. The *Gene* class creates the instance, and by the method it changes into protein which is the instance created by the *Protein* class. Fig.3 shows the simple and detail class. For example, the *Gene* class has name of gene and information of activators and repressors as the inner attribute. The information of activators and repressors is described as real type array, because there are many activators and repressors, and each element has the binding affinity.

A transcription from DNA to mRNA is described as the method of *Gene* class. The following is the description from the external functions.

```
SomeGene->Transcription();
```

Sending Transcription method to gene results in amount of transcribed gene products per unit time.

In regard to other classes, we construct the data (the inner state) and the method approximating faithful simulation of biological process.

3 Summary and Conclusions

This paper reported our attempt to develop the Biology Foundation Class, which can be a basis of broad range of simulation systems. The project have just started, and we wish to report the progress in future.

References

- [1] Gilbert, S. F., Developmental Biology (5th edition), Sinauer Associates, Inc, 1997.
- [2] Kitano, H., Hamahashi, S., Kitazawa, J., Takao, K. and Imai, S., "The Virtual Biology Laboratories: A New Approach of Computational Biology" *Fourth European Conference on Artificial Life*, 274-283, 1997.
- [3] Rumbaugh, J., et al., Object-Oriented Modeling and Design, Prentice Hall, Inc, 1991.
- [4] Stroustrup, B., The C++ Programming Language (2nd edition), Addison-Wesley, 1991. Addison-Wesley, Inc, 1991.