Interview with Dr. Pierre Gradit
Researcher at LAAS

Date: March 23, 2001
Place: The Institute of Computer Based Software Methodology and Technology

【Curriculum vitae】

1971	Born in Toulouse, France
1994	Master (Magistere) of Fundamental Computer Science, Ecole Normale Superieure de Lyon, France.
2000	Awarded Doctorate in Computer Science of University Paul Sabatier, Toulouse III, France

Current Position	LAAS－CNRS(*1)	Researcher
	OLC(*2)	Computer Scientist

[Selected Publication]

"A Layered Δ-net Specification of a workshop"
"A workflow specification environment"
"Graph-based coordination of cooperative agent" etc.

*1 LAAS-CNRS（Laboratory for Analysis and Architecture of Systems）
A research establishment operated as part of C.N.R.S., the French National Organisation for Scientific Research with the total of 450 persons. Officially set up on 10 July 1967.

*2 OLC（Outils et Logiciels pour la Communication）
Software and Tools for Communicating Systems

Q: First, please tell us how you came to know Lyee.

A: Mr. Negoro and his staff came to Toulouse, France, a while ago and made the presentation about Lyee method. We began to discuss during the break in the middle of the presentation. Consequently, we decided to plan this trip. In addition, Mr.

Hamid has known someone in my group, OLC, and invited me to study about Lyee in Japan.

Q: Is it correct that you heard the explanation about Lyee methodology only once prior to coming to Japan?

A: Yes, that is right. I met Mr. Negoro and Mr. Hamid in Toulouse for the first time.

Q: When you participated in the Lyee presentation, did you find it interesting?

A: Yes, I found this stimulating. I was impressed by Lyee methodology. During the first part of the presentation, I was still not sure. On the second part of the presentation, however, I felt clearer about the key concept of objectification of an intention. This, to me, was a good process to deal with software engineering. I had never heard such an idea and it is a new process to manage software engineering.

Q: Could you explain more about managing software engineering?

A: Software engineering is a very difficult problem because it always has to deal with something incomplete. This concept proposes identification of difficulties of software engineering.

Q: Please tell us about the purpose of your visit to Japan this time.

A: I wanted to learn more about Lyee methodology because I participated in the conference in Toulouse, which was the first I heard about Lyee.

Q: Are you planning to be a part of the international network Mr. Hamid has organized?

A: Yes, I will join for a year. At the end of the year, I will have to discuss with my boss at LAAS. I want to discuss Lyee methodology during a year and propose a paper in workshops and meet all the people involved in the project. After a year, we'll decide whether or not we will continue to be involved in the project.

Q: Could you tell us about the institution you are working for in France? What kinds of research do you conduct there?

A: LAAS is the largest proper unit for research in France. There are mixed units in which a part of the unit is university, company, etc. Most of the research units are mixed units. At LAAS, there are 450 full-time researchers and more than 150 people are in Ph.D. programs. Therefore, it is hard to encompass the whole activity of such a huge unit. However, I can summarize that there are three major directions of research: automatics, informatics, and electronics.

Q: Do you mean machine automatics?

A: It is process identifications, control, including robotics and so forth.

Q: To which unit of the institution do you belong and what kind of research have you been engaged in?

A: I belong to the "Software and Tools for Communicating Systems" research group that deals with communication protocols and coordination issues in a generic software context. I am working specifically on cooperative applications.

Q: What are the cooperative applications?

A: Cooperative applications are applications in which software agents may appear and disappear in time. The general organisation scheme of the software may evolve in time.

Q: Is it an evolve software?

A: Not exactly, but it is composed of small pieces of software distributed over the network. As software can appear and disappear, small pieces of code existing everywhere can begin executing or stop executing while they are communicating. You have to manage the evolution of the organization. If a component appears or disappears, it is important that you keep global coherence. The key concept is appearance and disappearance, and evolution is a consequence.

Q: Speaking of communication, have you seen Japanese cell phone with the Internet accessibility called "i-Mode"?

A: I have heard of it but never manipulated it.

Q: That is the state-of-the-art technology in mobile field. What do you think of it?

A: It is not a matter only of communication medium but also of communication protocols. If you want to make an invoice, you will have to deal with global software and different software agents that have to cooperate to make it. I think it is important to focus on the way communication is taking place rather than using a specific medium.

Q: Which software method has become the standard in France?

A: In France, object-oriented methodology such as UML is almost the standard but this is specifically for software engineering. For system engineering, many large-scale industries have their own methodology. They've developed their own methodology. The standard for the space application in Europe is called ECSS. Space application is very important in Toulouse.

Q: Was ECSS method developed by a private cooperation?

A: It is standardised by ESA, European Space Agency.

Q: In Japan, we had been developing system based on a method such as ADSG developed by IBM. In recent years, however, more and more C/S and Web developments are utilizing various kinds of tools, putting less importance on the method.

What’s the situation in France? Is method considered very important? I think that OOA is probably not a method but paradigm whereas Lyee is a method.

A: I am not a specialist to answer this software development question but I can say that we have several methods based on design, verification, and coding, which are applied in reengineering.

Q: In what aspects, do you think that Lyee has advantages?

A: I think that the most interesting feature is that Lyee says “software is a result of objectification of an intention” rather than “software is meaning.” This is very interesting idea.

Q: Lyee structure enables system development without considering about sequential order of program execution. Engineers don’t have to think about the sequential order when they develop programs. What do you think of this kind of approach?

A: It is an interesting point. In general, you have to think about the whole sequence when creating software. With Lyee structure, they are not obliged to think about the whole sequence but to think locally. This is interesting because a local point of view perfectly defines a local process. It is going into the right direction.

Q: Do you know any other method that is similar to Lyee method?

A: No. Lyee has some features that make it definitely original.

Q: How do you evaluate the mechanism of Scenario Function?

A: Scenario Function is built upon solid paradigms and has complex structures. Though we need a proof of this structure, it is hard to manage the proof of such complicated structure. Therefore, it is important that we consider how to prove it.

Q: Lyee program structure confirms validity of the user intention that user expressed as his/her development requirement by executing program. What do you think of this aspect?

A: With Lyee program structure, when values are set in data area of data items, testing is already done. In other words, testing is completed in the flow of Lyee program. And you also do general coherence testing at this stage. We can say that some kinds of local testing are all conducted here. Sometimes, when you execute program on LyeeALL, you'll see warning. Warning is raised when there is coherence problems somewhere in the program. I think the overall test is not necessary. Lyee is based on a principle of local coherence, which means if Scenario Function is correct, then, by the proof of local coherence, we have global coherence as well. In other words we don't need to test it.

Q: But you feel the need for the proof of Scenario Function?

A: At the moment, the proof of Scenario Function is out of reach. However, formalization of the general context (3-dimension-like space model) may achieve better confidence level upon Scenario Function. That is an intermediate goal I try to reach.

Q: Is it correct that you appreciate this method, at the same time, you feel the need for the proof from the academic point of view?

A: No, thinking in terms of formal proof is not the right way to proceed with Lyee methodology. Lyee proposes an image of how you transform or process your intention into software. We are not seeking the proof at this point. We are seeking to improve our confidence in Scenario Function and somehow characterize its scope of pertinence. That is the goal of formalization.

Say, you are in front of the mountain and you may say "I will go that way." But you'll immediately face the cliff in front of you. Thus, it is important to start from increasing your confidence level that may lead you to the final goal.

It seems difficult to prove validity of Lyee using existing logic and I can’ t say it cannot be or should be proved. This is one of the biggest questions for me. In one year, we will be able to consider if we need, can or should prove it.

Q: How do you conceive the approach in which independent program is created per word or data item?

A: That is a local way of programming and I like this idea.

Q: Conventional methods take an approach to break down and merge functions before actually creating programs so that we can avoid generating unnecessary things in advance. Meanwhile Lyee has completely opposite approach where program is made per data item without the process of merging functions beforehand. What do you think of this comparison?

A: Lyee methodology proposes thinking that is difficult to manage and it has dropped inheritance. That is the major difference compared with OOA. Lyee also proposes an intention and Scenario Function.

Q: With Lyee method, the size of program source is larger than that of conventional programs. What do you think of this aspect?

A: That is not to me a fundamental problem.

Q: Some people are concerned that it might require large resource.

A: It is the matter of optimization. We can find tools to optimize if we want to make the size of program source smaller. Any method has drawbacks and it is a matter of choice. Lyee has dropped inheritance and uses words. I don’ t know if it works completely but there are some interesting ideas to develop for designing software.

Q: Some people are concerned that Lyee’s recursive structure in which iteration occurs until value is set in all data items is interrupting response time. What do you think about this?

A: It is not very important problem to me. Because there were also problems associated with Java at the beginning such as execution speed. As the speed of computer improves everyday, it is not the problem.

Q: Do you see any problems structure-wise?

A: Lyee methodology can create any kinds of software writings. In terms of speed and optimization, VisualBasic is probably not the best target language for Lyee methodology as it is not fast.

Lyee is a methodology that has been developed over a single language but that methodology shall be applied to different languages. Thus, the problem of performance will appear as the secondary problem. Also, it may be a matter of application.

Q: What do you think is the difference between Lyee and OOA?

A: OOA is based on the certain concept and Lyee methodology is based on a different one. These two are created on completely different base. Thus, it is not relevant to compare these two, though these are both paradigm and method.

Q: From business user's point of view, both Lyee and OOA are one of their choices. Either of them can be used but not at the same time. In that sense, user needs to compare these two.

A: It depends on the needs of the customers. Since I am not a specialist in business applications, I cannot answer this question. OOA has been there for several years by now and some evaluations are already made whereas Lyee is still very young. This is why it is difficult to compare these two.

Q: When we introduce Lyee to potential customers, we often receive the question regarding the difference between Lyee methodology and OOA. How would you answer to this question if you were in this situation?

A: Key concept of OOA is inheritance as a software relation meanwhile Lyee proposes the concept of objectification that is relation between intention and software. People should take a look at Lyee because it has very interesting concept in terms of system development. Lyee proposes another way of thinking. This is very exciting and interesting. To develop good software, you have to form a group where people have different roles and ideas. I think Lyee is suitable for that purpose. Perfection is out of reach but something can be grasped from that.

Q: Do you think productivity increases with Lyee method compared to conventional methods?

A: I think any language can increase productivity including Lyee and OOA. If you propose a language for people to talk about work, productivity of the group will increase.

Using basic concept of Lyee, communication will be promoted. Lyee concept fits well for communication purposes because objectification is simpler than inheritance. Lyee concept makes communication among people much simpler.

At this stage, I am not sure if the process of an intention of Lyee itself has been completed. So it is hard to say whether or not it increases productivity. In terms of paradigm, I think objectification is more productive for communication between human than inheritance.

Q: Is there anything that you noticed such as advantages and disadvantages, or problems of Lyee?

A: Of course, there are things that could be improved in Lyee methodology. However, if it is used specifically in business applications, there will be no major problems. But if it tries to deal with any kind of program or language, some problems might be raised. It is still a young idea and some improvements are necessary. As for the development language, I am still not sure if Lyee can be applied to any language. It is a challenging issue and research shall be conducted from various aspects.

Q: You said that as long as Lyee method is used for business applications disadvantages will not be evident. Do you want to use Lyee method for your specialty area, communication system and cooperative application?

A: I think that Lyee paradigm will probably influence me in many ways. I, however, don't directly deal with user intention at work. Again, my work is not directly connected but perhaps there may be connection in the future.

Q: Do you deal with "words" in the course of your research?

A: Lyee makes programs or text by objectifying an intention. I am dealing with text part, not with intention. Since an intention is very abstract word, I could even write a paper on that single topic.

Lyee is significant in grasping logic that will give new interpretation on what I am currently working on. For me, the key concept of Lyee is an idea that modifies your perception of the reality.

For some people, a program itself is a proof. We can say that the program is the proof of Lyee structure, however, it is impossible to logically and rationally prove the intention of the program itself. If you define something, you will need to define upper level, etc. It may continue and never finish.

Q: How would you like to be a part of Lyee project in the future?

A: First, I have to talk with my boss. Personally, I am going to develop bridges between logic theory and Lyee methodology. I hope these bridges will bring questions and even answers.

I am confident in the fact that Lyee method can bring interesting themes. But I am sure that we are not able to image what kind of themes, at the moment. We will probably be able to foresee them as a result of contact between Mr. Negoro's ideas and the academic world. In the academic world, there are people like Mr. Negoro who propose something completely new. Perhaps there are other people in the world who have the same idea at the same time. It happened often in the past.

We can not yet know where this international network is going. I am trying to make bridges between common language for logic and Lyee method, and translate what I learned here in an acceptable language for the mathematician community, though I am not sure I will succeed. That is my objective.