

Automatic Generation Between UML and Code

Fande Kong and Liang Zhang
Computer Science department

Outline

- The motivation why we need to do the generation between the UML and code.
- What other people have done or been doing to achieve this task?
- How do we achieve this task?
- More to go?

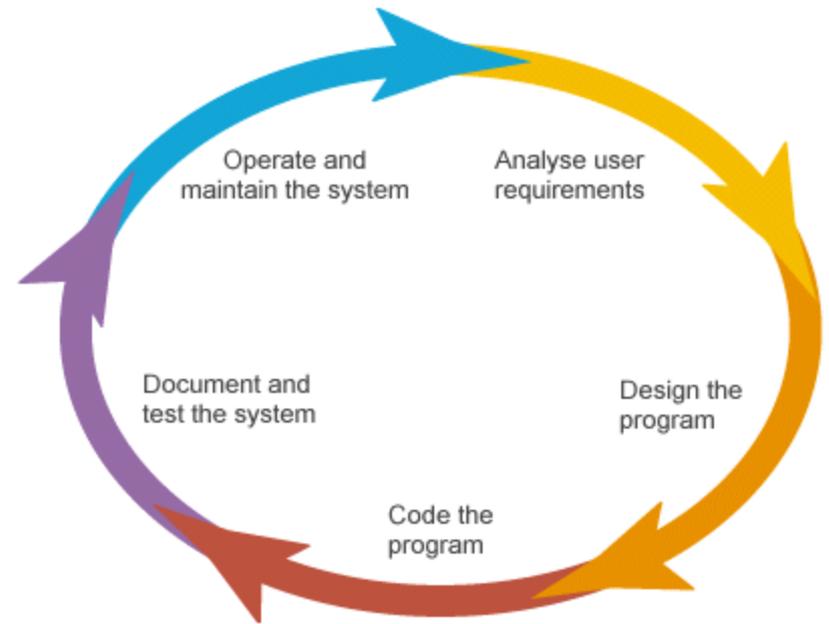
The software engineering life cycle

- What compose a cycle?

In the software development, the cycle is composed of analysis, design, code, test and maintenance.

- What relationships?

After the analysis, the design process accurately determine what you should code.



http://www.technologyuk.net/computing/software_development/software_development_life_cycle.shtml

Design determines code

The software engineering life cycle

- **How to describe the design process?**

In the design process, we can use Unified Modeling Language (UML) to and describe and portray the relationships between objects (classes), how the objects behave and what is the behaviors.

THUS, in other words, the UML diagrams can largely determine what your code looks like.

In the other side, the code should contain enough information with which we try to construct UML.

Problem and Motivation

- In during the software developing, after the analysis and design. The actual coding part is very time consuming, but does not need much thinking. As the above said, the results of the design can determine what your code look like.
- In other way, after the coding is complete, sometimes other people may want to test it. Then it is difficult to read the code to get idea.

Problem and Motivation

- **For the first problem**

As the coding part is decided by the analysis and design, we can somehow hope we can get the complete code automatically.

- **For the second problem**

As we have the complete code, but it is annoying to get the structure of the code by reading it. We hope we can generate the visual relationship of the Classes.

First Problem

- For the object oriented program design and analysis we got some UML diagram.
- Then the UML diagram determines what the code should be like.
- There are some general elements in UML that could be changed into code.
- Put the code together, we can get a draft code for the requirements.

Second Problem

- As we have the complete, then there are some relationship between the classes and what the class does.
- We try to visualize the relation and generate the UML diagram. Then we can clearly see the structure of the code and what it does.
- Using the derived UML diagrams, we can easily analyze the quality of the code and then make some improvement.

Relation between two problems

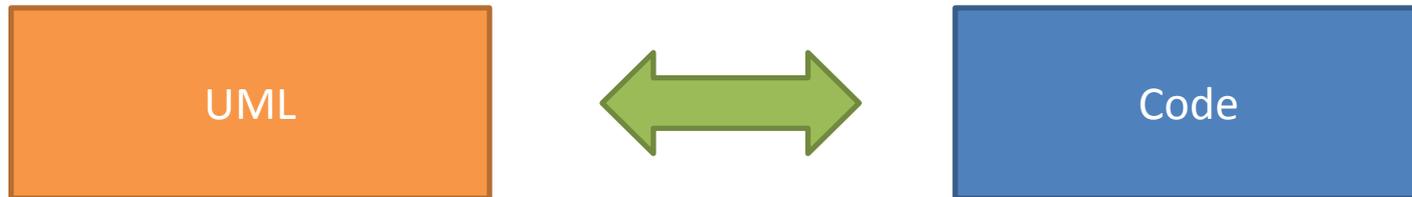
- **UML to Code**

After we got UML from the code, maybe we can see the design is a little different from the requirement. Then we can just change the UML we can directly correct the code.

- **Code to UML**

After we got the code from UML diagram, we need to check the code is what we want. Then we can just generate the UML from the code to verify the design is right.

Now the goal is:



Who are trying this goals

- **Model-driven architecture**

“The Object Management Group introduced the Model-Driven Architecture(MDA) initiative as an approach to system-specification and interoperability based on the use of formal models [MDA, MDA2, DSouza].” http://www.omg.org/mda/mda_files/Model-Driven_Architecture.pdf

In other words, they are trying to generate code from models.

Model-driven architecture

- **Some important standards for MDA**

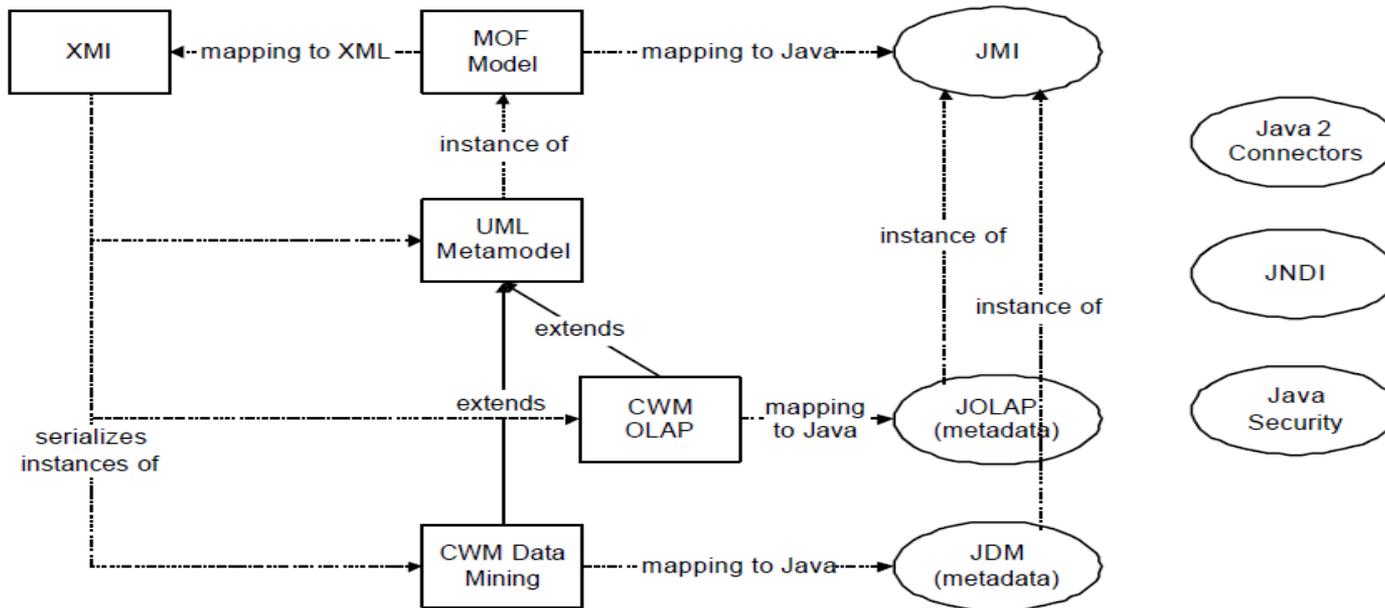
Firstly, “at the core of the MDA concept are a number of important OMG standards: The Unified Modeling Language (UML), Meta Object Facility (MOF), XML Metadata Interchange (XMI), and the Common Warehouse Metamodel (CWM). These standards define the core infrastructure of the MDA, and have greatly contributed to the current state-of-the-art of systems modeling [MDA2].”

http://www.omg.org/mda/mda_files/Model-Driven_Architecture.pdf

Model-driven architecture

- Relationships between standards

The interoperability and interactivities between these important standards contribute to the whole MDA system.



Model-driven architecture

- **PIM to PSM transformation**

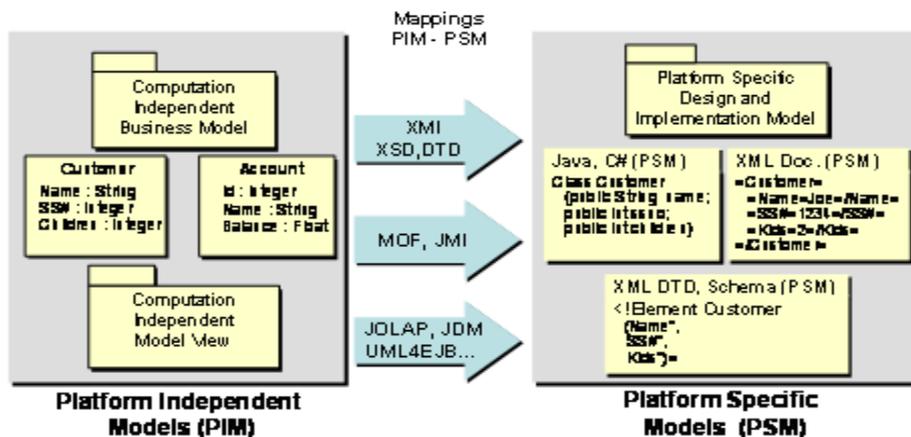
“The Model-Driven Architecture approach defines system functionality using a platform-independent model (PIM) using an appropriate domain-specific language (DSL) such as UML. Then given a platform model corresponding to CORBA, .NET, the Web, etc., the PIM is translated to one or more platform-specific models (PSMs) that computers can run.”

http://en.wikipedia.org/wiki/Model-driven_architecture

Model-driven architecture

- PIM to PSM transformation (continued)

We can use a picture to clearly display this process. In the picture, we try to transform some important standards for platform independent models to another platform specific models that can run at computer.



<http://www.ibm.com/developerworks/rational/library/3100.html>

Model-driven architecture

- **Some implementations for MDA**

The OMG organization only provides some rough specifications rather than implementations, often as answers to Requests for Proposals (RFPs).

One important implementation for OMG specifications is the Eclipse Foundation (EF). Most of implementations of OMG modeling standards can usually be found in the Eclipse Modeling Framework(EMF) or Graphical Modeling Framework (GMF), the Eclipse foundation is also developing other tools of various profiles as GMT.

More details you can see:

http://en.wikipedia.org/wiki/Eclipse_Modeling_Framework

Model-driven architecture

- **Some problems on MDA**

- (1) There are too many standards. The OMG makes too many standards to complex the problems.
- (2) They don't unify the platform specific languages, e.g. Java, .net, c++ and so on. Thus, they need to construct a transform for each platform dependent language.
- (3) They don't consider the philosophy of the transforms.
- (4) They don't pay many attentions on reverse engineering.

Our ideas on how to reach the goal

- We should only extend UML to represent every model for any domains. The current the UML have some limitations on how to describe some operators between UML elements.
- We should define an interface language for all exist programming languages.
- We should consider some math theories which could be used to do some transform mapping and constraining.
- We should pay more attentions on reverse engineering.

Now, start talking more about our ideas.

How to make code from/to UML?

- **Code generation**

From drawn UML diagrams, try to directly generate some useful computer language (Java, C, C++ and so on) codes which could run on platform. Or carry reverse process.

- **Is it possible?**

In some specific areas, there always have been a few cases, where they directly derive the usable and efficient based on domain models.

How to make code from/to UML?

- Is it possible (continued)?

For example, in numerical computation, the FEniCS Project has been developing a collection of free software which could be used to produce computer language codes based on given mathematics models (formulas). More details please see

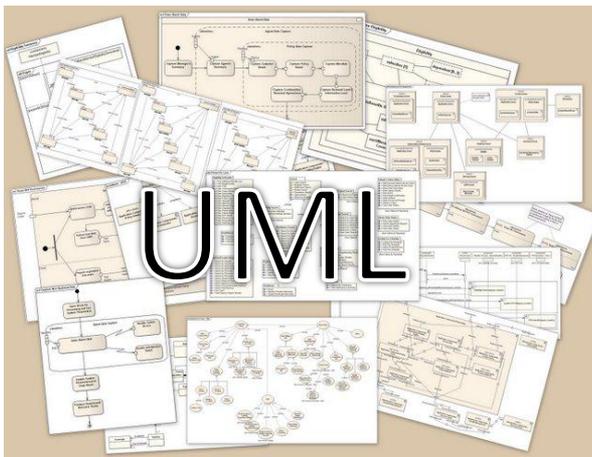
<http://fenicsproject.org/>



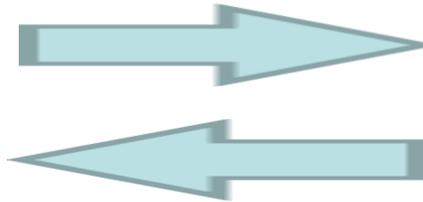
How to make code from/to UML?

- But it is still very hard

For a specific domain, maybe we can come up some ideas on how to produce some specific language codes. BUT, we still could not derive the general codes from UML diagrams for general domain.



How to /from....?



```
4 // Message relates the operation based on the message.
5 // Return an Message object can be used to transform the possible relation
6 // To make an SQL table structure.
7 // Create a message.
8 // Add the Message object to the message.
9 // Add the Message object to the message.
10 // Add the Message object to the message.
11 // Add the Message object to the message.
12 // Add the Message object to the message.
13 // Add the Message object to the message.
14 // Add the Message object to the message.
15 // Add the Message object to the message.
16 // Add the Message object to the message.
17 // Add the Message object to the message.
18 // Add the Message object to the message.
19 // Add the Message object to the message.
20 // Add the Message object to the message.
21 // Add the Message object to the message.
22 // Add the Message object to the message.
23 // Add the Message object to the message.
24 // Add the Message object to the message.
25 // Add the Message object to the message.
26 // Add the Message object to the message.
27 // Add the Message object to the message.
28 // Add the Message object to the message.
29 // Add the Message object to the message.
30 // Add the Message object to the message.
31 // Add the Message object to the message.
32 // Add the Message object to the message.
33 // Add the Message object to the message.
34 // Add the Message object to the message.
35 // Add the Message object to the message.
36 // Add the Message object to the message.
37 // Add the Message object to the message.
38 // Add the Message object to the message.
39 // Add the Message object to the message.
40 // Add the Message object to the message.
41 // Add the Message object to the message.
42 // Add the Message object to the message.
43 // Add the Message object to the message.
44 // Add the Message object to the message.
45 // Add the Message object to the message.
46 // Add the Message object to the message.
47 // Add the Message object to the message.
48 // Add the Message object to the message.
49 // Add the Message object to the message.
50 // Add the Message object to the message.
51 // Add the Message object to the message.
52 // Add the Message object to the message.
53 // Add the Message object to the message.
54 // Add the Message object to the message.
55 // Add the Message object to the message.
56 // Add the Message object to the message.
57 // Add the Message object to the message.
58 // Add the Message object to the message.
59 // Add the Message object to the message.
60 // Add the Message object to the message.
61 // Add the Message object to the message.
62 // Add the Message object to the message.
63 // Add the Message object to the message.
64 // Add the Message object to the message.
65 // Add the Message object to the message.
66 // Add the Message object to the message.
67 // Add the Message object to the message.
68 // Add the Message object to the message.
69 // Add the Message object to the message.
70 // Add the Message object to the message.
71 // Add the Message object to the message.
72 // Add the Message object to the message.
73 // Add the Message object to the message.
74 // Add the Message object to the message.
75 // Add the Message object to the message.
76 // Add the Message object to the message.
77 // Add the Message object to the message.
78 // Add the Message object to the message.
79 // Add the Message object to the message.
80 // Add the Message object to the message.
81 // Add the Message object to the message.
82 // Add the Message object to the message.
83 // Add the Message object to the message.
84 // Add the Message object to the message.
85 // Add the Message object to the message.
86 // Add the Message object to the message.
87 // Add the Message object to the message.
88 // Add the Message object to the message.
89 // Add the Message object to the message.
90 // Add the Message object to the message.
91 // Add the Message object to the message.
92 // Add the Message object to the message.
93 // Add the Message object to the message.
94 // Add the Message object to the message.
95 // Add the Message object to the message.
96 // Add the Message object to the message.
97 // Add the Message object to the message.
98 // Add the Message object to the message.
99 // Add the Message object to the message.
100 // Add the Message object to the message.
```

CODE

How to make code from/to UML?

- **Need to do three things**

First, we could construct a language to describe UML diagrams. The Extensible Markup Language (XML) is already very close to this-like demanded language. And the language should have some mechanism to do some inter-operations between UML elements. BUT, unfortunately, the XML can't carry these operations. THUS, we indeed a absolutely new language to help us do some operations and communication between UML elements. We call the language, extensible UML (EMUL).

How to make code from/to UML?

- **Need to do three things (continued)**

Second, we need to define an interface language for all computer languages which are used widely among computer world. Today, there are many popular computer programming languages in our world, e.g. Java, C, C++, Fortran, Python and so on. If we develop a generator for each programming language, it would cost a lot. THUS, we must define an interface language to unify the exist programming languages. We call it unified interface language (UIL).

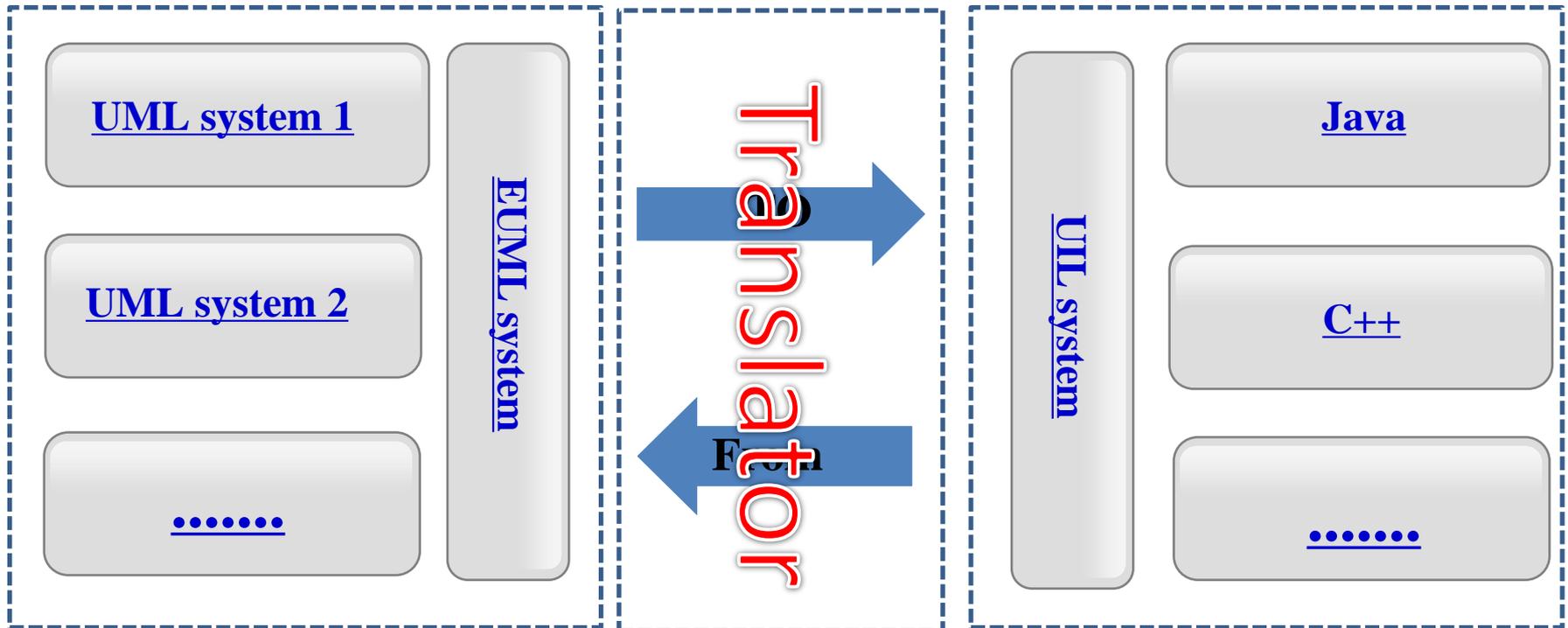
How to make code from/to UML?

- **Need to do three things (continued)**

Third, we need to construct a translator which could translate to/from information coded by EUML from/to information coded by UIL. How to construct the translator is most hard part of our project. We can extract some abstract models from general domains, and then construct a math system to explain and describe these models. THUS, we can use some math mechanism to construct the translator.

How to make code from/to UML?

- System architecture

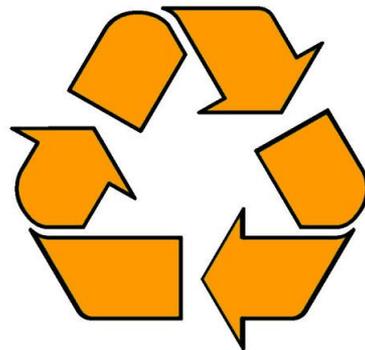


How to make code from/to UML?

- **Some ideas for translator**

How to construct an efficient and usable translator? It is the most hard part of the system. It is sure that we can not directly produce code/UML from UML/code once. But we can do some iteration, that is, we give an initial framework for code /UML and then we repeat the translating process until the results meet our requirements.

Iteration

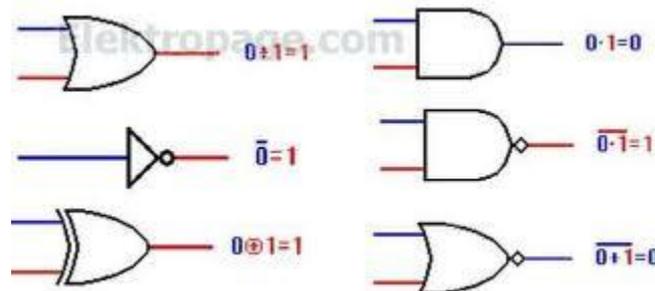


How to make code from/to UML?

- **Some ideas for operators between UML elements**

There are a lot of operators/relationships between UML elements , e.g. aggregation/composition, inheritance, association, qualification and interfaces. We could construct a new math algebra system to carry out these operators and describe these relationships. That is, we want to extract some math abstract ideas to deal with complex UML operators and relationships.

Algebra

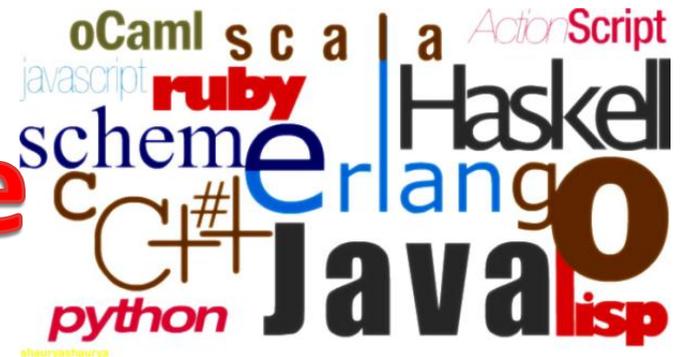


How to make code from/to UML?

- **Some ideas for interface language**

Today, we can make some interchanges between a lot of programming languages, e.g. convertor for Fortran to C, convertor for Java to C and so on. But nobody did some interfaces to which all other programming languages are adapted . According to the computer mechanism all exist programming languages satisfy, we define a new interface language.

Interface language

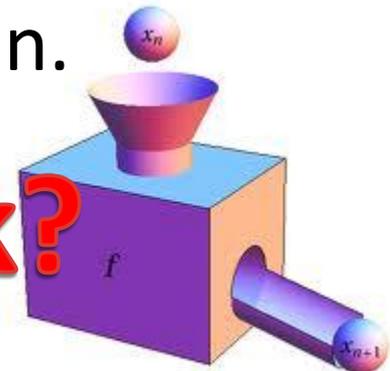


Some issues

- **How to pick an initial framework up?**

While translator doing some repeat iteration processes, in order to converge to a satisfying result, we need to pick up a reasonable initial guess framework for code/UML. BUT, how to choose is very hard, that should depend on some experience on related domain.

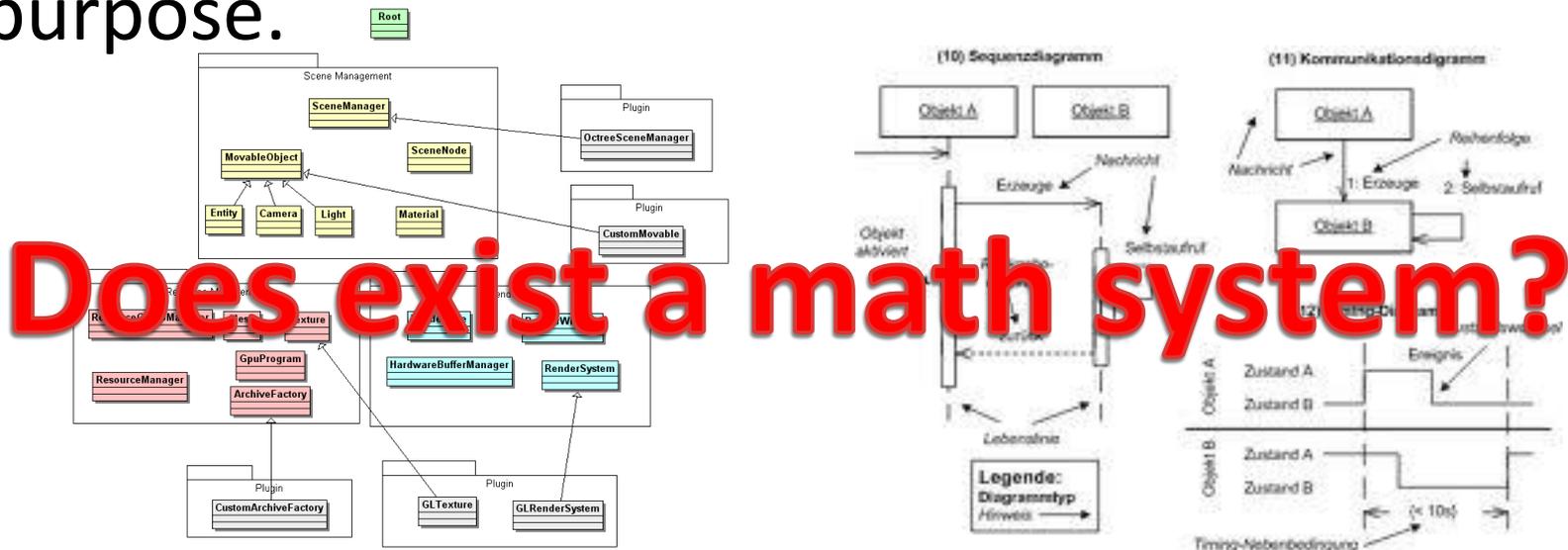
An initial framework?



Some issues

- Does exist a math system in UML diagrams

Extract some math system from UML is very hard. Nobody can prove it exist or not today. There are a lot of works we need to do for this purpose.



More to go

- We can use different code to achieve the same task.
- But definitely we can only get one set of code from the given UML.
- Then how do we decide the one we got is the best or nearly the best?

More to go

- After we got the code or the UML, how do we know the code is right?
- How to update the code or the UML dynamically.

Any questions?

Thank you!