Review on UML CASE Tools

WAN HASNIRA WAN HUSIN, SITI HAFIZAH AB HAMID, MOHD HAIRUL NIZAM
Faculty of Computer Science & Information Technology,
University of Malaya,
50603 Kuala Lumpur,
MALAYSIA

http://www.fsktm.um.edu.my

Abstract: - This paper presents a review on existing popular UML (Unified Modeling Language) CASE (Computer Aided Software Engineering) tools that have been used by many researchers, requirements analysts, system designers, and students. The review is done to look at the weaknesses they contain in order to come out with future and better UML CASE tool. There are several criterion have been considered, which are system boundary, exporting diagram, formal description support, robustness, auto-generation of UML diagram, place the user in control, reduce the user's memory load, and make the interface consistent. This review is important to give ideas to researchers on existing UML CASE tools in order to produce better UML CASE tool in the future.

Key-Words: - Unified Modeling Language, CASE tools

1 Introduction

Currently, there are several UML CASE tools on the market that describe the semantics, notations and constructs of UML. Due to this, research has been conducted on current UML CASE tools to gather useful information for the system development project. This review is important especially for UML CASE tools research and future development especially in the area of software requirements. The current UML tools that have been reviewed on are:

i. SmartDraw
ii. System Architect
iii. Rational Rose
iv. ArgoUML
v. Event Studio

To evaluate the five current UML tools as explained in the previous section, several criteria need to be considered. Thus, evaluation has been made by referring to a few features described by [1]. In addition, in order to evaluate the interface design of current UML tools, The Golden Rules that proposed by Theo Mandel [2] have been referred.

In April 2002, [1] presented a paper on evaluation of a range of current UML tools. The title of the paper is "Evaluation of Some of The Current UML Tools". He has evaluated the tools based on the features discussed below:

i.System boundary
Use cases do capture system requirements. Hence it is very important that the system's boundary is defined. A system boundary is the boundary between the system being modeled and the outside world.

ii.Exporting diagram
The UML tool should be capable of exporting diagrams into different formats such as .jpeg, .gif, .bmp, and many others, which may eventually be imported into a word processing document or web pages.

iii.Support formal description
UML tools must support a number of formal descriptions of construct such are associations, compositions, aggregations and generalization.

iv.Robustness
The UML tool must be with rock-solid reliability to prevent users from losing potential hours of productivity when a problem or crashes in the middle of a design session.

v.Auto-generation of UML diagram
The UML tool would be better if they could auto generate the diagram. The tool should facilitate the creation of a trace file during the execution of an

existing program in order to capture the requirement. After a trace file is created, the tool is then used to analyses the trace in order to find the pattern of the diagram. Then, the UML tool should be able to generate the diagram based on requirements.

In 1997, Theo Mandel has proposed The Golden Rules as a set of user interface design principles that is used to guide important software design activities [2]. The Golden Rules are:

i.Place the user in control
The interface of the system must support the user's understanding of a task and should not force the user to follow the computer's way of doing things.

ii.Reduce the user's memory load
The system must place all necessary information on the screen at the same time.

iii.Make the interface consistent
The system must have a consistency of form and behavior.

From the combination of the features explained by Mwaluseke and The Golden Rules proposed by Theo Mandel, the current UML CASE tools that have been discussed in the previous section are evaluated.

2 SmartDraw

SmartDraw helps users look like a graphic professional. No special skills are needed to draw charts and diagrams as the software uses the drag and drop technique. It has symbols and clip art images for users to use in their diagrams and charts. Aside from the built in components, they can also import their own symbols and clip arts.

Automatic alignments are provided for neat and crisp drawings. Users can use the templates and examples as reference. Anything done with this software can be printed or saved in gif, jpg or html format. This software can easily convert drawings made using other software. Furthermore, it works hand-in-hand with Microsoft Office which means drawings done in SmartDraw can be copy-pasted into Microsoft Words, Excel, PowerPoint, and various other programs.

All versions of SmartDraw act as an OLE server. This means that user can embed SmartDraw drawings into Microsoft Word, Excel, PowerPoint and any other OLE Client program by simply copy-pasting. User can also run the SmartDraw from inside any of these programs by double-clicking on the embedded drawing. Furthermore, in SmartDraw, users are allowed to export and import images in popular graphics formats including wmf, bmp, jpg, gif, tif and many others.

However, SmartDraw also has a disadvantage in which it is rather vague in expressing the parts of the UML diagrams apart from having a background mark on the images made.

3 System Architect

System Architect provides all of the tools necessary for development of successful enterprise systems from Popkin Software [3]. It is a tool to integrate, in one multi-user product, industry-leading support for all major areas of modeling, including business modeling, object-oriented and component modeling with UML, relational data modeling, network architecture design and structured analysis and design. All functionality is harnessed within System Architect's extensible repository with native support for Microsoft VBA.

System Architect is a repository-based visual modeling tool that supports the following methodologies in a single product:

- Enterprise Modeling (Strategy and Planning & Business Requirements Capture)
- Data Modeling
- SSADM
- Business Enterprise Modeling (Catalyst)
- Business Process Modeling (IDEF)
- Object and Component Based Design with UML

Similar to various other application, System Architect has the menu bar and toolbar. Additionally, the toolbar consist of symbols used for drawing diagrams. At the left-hand side is the browser to view the various design such UML, Structured, Data Modeling, Business Direction, and etcetera.

The advantages of System Architect:

1. Users can provide a very detailed and complete design. Each UML element can be assigned with behaviors, actions, triggers, child or parent.

2. This CASE tool is very suitable for large scale projects for experienced designers.

3. Codes can be automatically generated by System Architect in Java, C++, CORBA and VB.

4. System Architect can generate a HTML report for the whole system design or selected diagrams with just a few mouse click.

5. It has built in encyclopedia that can be used as example, reference or starting point.

The disadvantages of System Architect:

1. It is a complex application to use. One would have to go through the tutorial and do a lot of practices to gain experience or take a formal class to be able to use this software.

2. It is not suitable for beginners in designing as one would have to understand software design fully before taking full advantage of this tool.

3. The diagrams, codes, reports and documentations really look professional.

4 Rational Rose

Rational Rose is an object-oriented Unified Modeling Language (UML) software design tool intended for visual modeling and component construction of enterprise-level software applications [4]. In much the same way a theatrical director blocks out a play, a software designer uses Rational Rose to visually create (model) the framework for an application by blocking out classes with actors (stick figures), use case elements (ovals), objects (rectangles) and messages/relationships (arrows) in a sequence diagram using drag-and-drop symbols. Rational Rose documents the diagram as it is being constructed and then generates code in the designer's choice of C++, Visual Basic, Java, Oracle8, CORBA or Data Definition Language [3].

Two popular features of Rational Rose are its ability to provide iterative development and round-trip engineering. Rational Rose allows designers to take advantage of iterative development (sometimes called evolutionary development) because the new application can be created in stages with the output becoming the input to the next. (This is in contrast to waterfall development where the whole project is completed from start to finish before a user gets to try it out.) Then, as the developer begins to understand how the components interact and makes modifications in the design, Rational Rose can perform what is called "round-trip engineering" by going back and updating the rest of the model to ensure the code remains consistent.

Despite the fact that Rational Rose is widely accepted as the leading UML tool, it lacks very important aspects as explained below:

- Code generation is not automatic. It does not generate code as one draws a diagram.
- No consistency checks are done. It does not check on the consistency of classes and relationship.

5 ArgoUML

ArgoUml is an open source application which uses the UML to model the design of computer software. The application runs on most platforms since it is implemented in Java. It is distributed under the BSD license. It provides support for most diagram types of the UML standard and includes cognitive support [5]. ArgoUML provides the following features:

- Runs on any platform with Java 1.2
- Standard UML Meta-Model
- XMI-Support
- UML design editing
- OCL support
- Database support
- Several diagram export formats
- Code generation (partially implemented)

Using ArgoUML, a user has the capability to draw a use case diagram by clicking “Add” on the appropriate icon in the toolbar.

6 EventStudio

EventStudio is a CASE tool for distributed system design in object oriented as well as structured development environments. EventStudio supports multiple scenario use case and sequence diagram modeling. EventStudio is particularly suited for Message Sequence Charts (MSCs), real-time and embedded system design, use case development, object sequence diagram development, protocol design and documentation, process flow diagrams, distributed system design and business process workflows [3].

Unlike SmartDraw or any other CASE tools, this software does not use the drag and drop technique to draw the required diagrams. Instead, it designs the software and draws the diagrams based on requirements entered by users. These requirements are entered using the Featured Description Language (FDL). FDL is a software development which involves the design state followed by coding. After inputting the requirements in FDL, users can choose to draw use case diagrams, interface use case diagrams, interaction use case diagrams, message filter use case diagrams, unit test procedure, summary and statistics, collaboration diagrams, interface collaboration diagrams or message filter collaboration diagrams documents for the intended software or system. The documents are presented in the .pdf format.

Using this software, users do not have to design their software thoroughly. As long as they have an overview of the system, this software is able to design their system by producing the required diagrams. All they have to do is type in the system requirements in the accepted format to produce the desired diagrams.

However, users have to learn the format and syntax of FDL in order to enter the requirements into the system. These necessitate users to learn prior to before using this software.

Table 1. Comparison between five UML CASE tools

UML Tools / Criteria	System boundary	Exporting diagram	Support formal description	Robustness	Auto-generation of diagram	Place user in control	Reduce the user's memory load	Make the interface consistent
SmartDraw	Not provided	Capable to export diagram in many formats	Supported	No. Once the tool give problems, users need to start a design session again	No because it uses drag and drop techniques	Yes	Yes	Not consistent
System Architect	Provided	Capable to export diagram in many formats	Supported	No. Once the tool give problems, users need to start a design session again	No because it uses drag and drop techniques	No, because it was a complex system.	No	Consistent
Rational Rose	Not provided	Not able to save diagram in other formats	Not supported	No. Once the tool give problems, users need to start a design session again	No because it uses drag and drop techniques	Yes	Yes	Not consistent
ArgoUML	Provided	Capable to export diagram in many formats	Supported	No. Once the tool give problems, users need to start a design session again	No because it uses drag and drop techniques	Yes	Yes	Not consistent
EventStudio	Not provided	Capable only in PDF and word picture (EMF)	Not supported	No. Once the tool give problems, users need to start a design session again	Yes but based on Feature Description Language	No because user need to study a format and syntax of FDL before using it	No	Not consistent

Discussion

In general, the existing UML CASE tools assist their end user to draw any UML diagram easily and effectively. In order to do that, the end users need to know his/her requirements and all the important elements of the diagrams. For example, actors, use cases, classes and actions. It is only Event Studio allowed the end user to enter the requirements by using FDL in order for them to be generated as diagrams. There is no such tool that allowed the end user to simply enter any requirements without using any specific language to be generated to any diagram. System analysts or any project developer will cut down a lot of time if such tool is available. In fact, more proper documentation will be produced without the effort of the project developer.

References

[1] Mwaluseke, G.W (2002). *Evaluation of Some of the Current UML Tools.* [Internet] Available

from: http://myweb.lsbu.ac.uk/~mwalusgw [Viewed 17/06/2005].

[2] Roger S. Pressman (1997). *Software Engineering, a Practitioner's Approach 4th Edition*. System Testing (p. 507-509). Singapore: McGraw Hill.

[3] Meyer, J., (2005). *Comparison of UML Modeling Tools*. Dunstan Thomas Consulting. [Internet] Available from: http://consulting.dthomas.co.uk [Viewed 11/01/2006].

[4] Wendy B., and Michael B., (1999). *UML with Rational Rose*. Use Cases and Actors (p. 97-147). Sybex.

[5] Alejandro R., Vanpeperstraete, P., Rueckert, A., Kunle Odutola, Jeremy Bennett, Linus Tolke, Van Der Wulp, M. (2004). *ArgoUML User Manual. A tutorial and referencedescription.* [Internet] Available from: http://argouml.tigris.org/documentation/defaulthtml/manual [Viewed 18/08/2005].