

## EASY SOFTWARE DEVELOPMENT BY AUTOMATIC CODE GENERATION, BUSINESS LOGIC AND VALIDATION ENGINE INSERTION.

Mr. Sunil D Rathod

PhD Research Scholar, Computer Engineering Dept.  
JIT University, Jhunjhunu, Rajasthan, India Sunil2k\_r@yahoo.co.in

### ABSTRACT

In today's fast and progressive world is process optimization become essential in all fields of life and an IT industry is no exception to this. The Integrated Development Environment (IDE) based software development environments are popular for easier and faster project development using Drag and Drop (DND) tool box with all essential controls like text box, labels, buttons, etc. This helps developer to design Graphical User Interface (GUI) or User Interface (UI) just with mouse click. Though the designing of UI and Code generation as templates is easier using these tools but none of the existing tools are capable of generating complete working code with embedding "Business Logic" (BL) automatically in code. The vendor specific tools from ORACLE, Microsoft, IBM mainframe and some similar tools from other giant vendors are used to design and develop the products for analysis and reporting purpose but these tools need the proprietary DB and vendor specific environment to develop the project. Moreover these tools generate code without BL and developers later add BL code repeatedly for commonly used operations like Inset, Update, Delete, Display on the data of Database (DB). Our proposed tool named "Rapid Project Builder" (RPB) eliminates the time consumed for coding the same BL repeatedly. The RPB can also perform Automatic Code Generation (ACG) in specific language like C++, Java, etc of developer's choice. The tool helps developers to design the application's UI with DND along with the specification of fields used in UI form. After adding button control to UI with operation choice like add, delete, search or modify, RPB automatically embed code for BL to these controlling language specified by developer. All operations like ACG, Automatic DB Creation (ADC), table creation with all DML operations and Validation code insertion, are done using properties stored as XML semantics in automatically generated XML file by RPB.

**KEYWORDS-** UI, DND, XML, RPB, ACG, BL, ADC, OFMA, CRM.

### INTRODUCTION

Recent trends in Software Industry explore technology which meets dynamic demands of software projects with different stakeholders like client, analyst, developers and project manager focus on cost reduction, efficiency and reliability of product to be developed at time. Globally accepted standards and tools are devised to make the Software Development Life Cycle (SDLC) process more generic, platform independent, cost effective and automatic with respect code generation and transformation of specification into various other technologies [12],[13],[16],[17].

The Unified Modelling Language (UML) specifications focus on design of the software projects using DND and with forward engineering the designed classifiers like Class diagram, Use Case, State Chart and Sequence diagrams. UML based tools convert these classifiers to code using forward engineering [1],[2],[5],[9],[10]. For instance special efforts are taken for transferring Sequence diagrams into code using XMI representation in [19]. An automatic code generation from given UML classifiers are achieved using these tools but none of the approaches is able to generate complete language specific code with embedded business logic in it. The code generated for the any given classifier in UML is just a code template with its structure like class definition, attributes with data type and blank code template of methods [27].

In the generated code template, BL code is to be explicitly written by developer for operations like Addition, Change, Insert and Delete (ACID) on data stored in DB tables. The code of BL depends on the constraints specifications of domain of the project [27],[30],[36].

The front end and back end DB connectivity is another issue to be handled by developer at the time of project development which can either be done by data connectivity control of the development environment or using code

explicitly written in development language. Thus the entire SDL takes a lot of time in addressing the issues like code for UI, Code for different control likes text box, command button, radio button, check box, etc. along with programming different events like mouse click or key press events, DB creation with ODBC connectivity issues and last is embedding BL in code.

Our tool, Rapid Project Builder (RPB) addresses all the above issues together just by designing the project UI using DND along with the specifications of different properties of the controls like textbox, buttons, labels, etc. RPB stores the information given by developer as XML Meta data and automatically generates code, DB with tables, and BL embedded in the code.

The entire paper organization gives capabilities of RPB. In next section literature review as related work is given; subsequent chapters cover the architecture of proposed RPB, along with methods for ACG, ADC and automatic embedding of BL in language specific code along with XML file generation. The XML Meta file generation, ACG, ADC, automatic BL code generation and validation code creation is main focus of the paper. The works concluded with some experimental results along with future scope of the RPB tool. The RPB tool is designed and developed using Java programming language.

## RELATED WORK

Since its emergence, many efforts have been put to make software development process as easy and fast as possible without losing the robustness, efficiency, user-friendliness along with cost effectiveness of software products. The user of the software has ever changing and dynamic requirement and so it is very much essential to use innovative methods and methodologies with different paradigm to meet these challenges. Experts also try to make software product more generic, platform independent, portable, scalable and robust with reduced cost of development.

MDA approach proposed OMG is very generic and is being adopted at large in many industries as standard practice of product development so as to meet all possible challenges [12],[13],[17]. The process of rapid and simplified project development with code reusability has already been achieved with the evolution of Object Oriented Programming (OOP) technologies which is again a part of UML standards [12],[27],[28].

The UML tools like IBM Rational Rose, star UML, and the like have been using evolutionary approaches to generate the code automatically by using state chart and class diagrams as classifiers. [1][5].

The authors in [3] have extend the functionalities of existing UML tools and developed a new tool called UJECTOR (UML to Java Executable Code generator) for generation of executable Java code from UML diagrams automatically. If a set of three UML classifiers namely class diagram, sequence diagrams and activity diagrams are given as input to the tool; it generates a complete executable Java code automatically. UJECTOR generates Java class structure from class diagram, method and its flow of control through sequence diagrams, and object manipulations code by using activity diagrams.

In [7] the authors proposed a system that allows user to specify his/her code specification using an easy-to-understand, simple-to-write and more or less unchangeable pseudo code specifications. The system checks errors, if any, in pseudo code and converts the algorithmic specifications into code of specific language like C, Java.

The concrete system, developed in .NET framework for automatic code generation, DB table creation, and some BL embedding in the code is seen in [23]. This system tool is vendor specific can only be used with Microsoft platform.

The giant vendor Microsoft has many language specific tools like VB6.0 or VB/ASP .Net framework which use DND approach for UI design. These tools generate code templates but no BL code or automatic DB table creation is possible by just DND [36].

The report generation tool of ORACLE Financial Management Analytics (OFMA) too lags in generating source code of specific language with BL is embedded in code [33].

The similar features like UI with report generation are available with tools like eclipse Windows Builder, Microsoft Dynamics Customer Relationship Management (CRM) [34],[35].

Our tool RPB is aimed to reduce the time and cost of product development in design and coding phase (frontend as well as backend code generation) of Software Development Life Cycle.

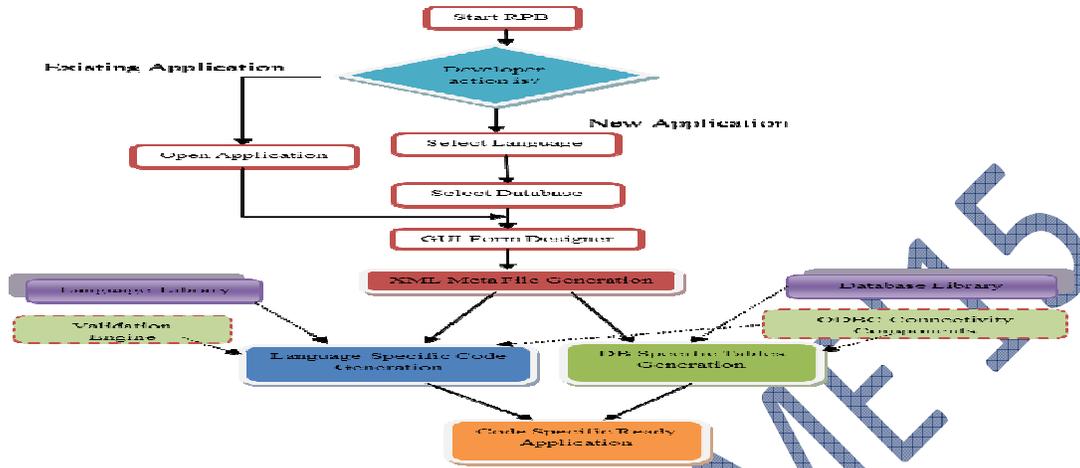
First kind of such an idea was proposed in [24],[25],[26] in which the authors have given the concrete frame work of tool like RPB.

As a first step, this tool is incorporated with the functionalities to develop a concrete prototype for standalone applications with backend support but in future this tool can be extended with the functionality to develop web applications.

## RPB ARCHITECTURE

The RPB architecture with process flow is as shown in figure fig-1 below.

The RPB tool has standard IDE for developer to design and develop the desired application using DND. The tool provides similar project form design facilities as that of VB .Net or Net beans using which a developer can easily design UI. The developer can either create a new application from the scratch or open an existing one to customize it.



**Figure 1. RPB Architecture with Flow**

**METHODOLOGIES**

The methodologies used to design and develop software using RPB with its various components are discussed in this section. Form Creation Mechanism using DND The developer creates UI by DND using controls in tool box. Through property window developer specifies various properties of controls like Label, text field, text area, buttons, etc. The properties like data type, size, name, fonts set for each of the fields help in creation of CLASS with attributes and corresponding fields for DB table. Each control has separate property window using which the properties for each of these controls can be set appropriately. GUI from designer component of RPB is used to achieve this task. Figure fig-3 & fig-4.

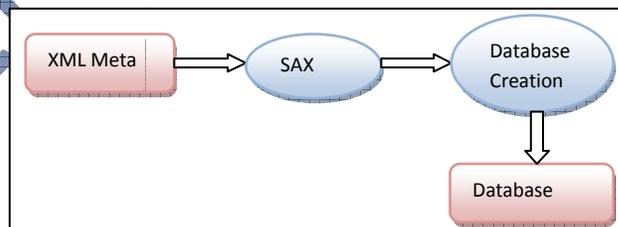
**Conversion of GUI to XML Meta information**

Once the UI is designed, RPB automatically generates XML Meta data file. The UI which is saved as an object is given to a converter module of RPB. This module extracts proper information from UI object to generate XML file. The XML file creation is depicted in figure fig-5.

**Conversion of XML Meta class to Database creation**

DB tables in backend DBMS are created using XML Meta information. SAX parser extracts proper information for table creation from XML Meta File. This extracted information is used in Conversion Module of RPB and a DB Table is automatically generated using DML syntax of specific DBMS like MS-Access as follows.

CREATE TABLE Customer (Name Text, Age Int, Sex Char(5), Pin Long); Figure fig-2 shows this. When developer creates workspace for project Database with workspace name and .mdb extension is automatically created.



**Figure 2. XML to DB Table creation process**

**CONVERSION OF XML METACLASS TO JAVA CLASS**

The RPB automatically creates a JAVA class from the above generated XLM Meta file using Language library syntax file which has all the relevant language key words, conditional/loop control, class, interfaces, design pattern,

and syntax of specific language. Each language has own separate file with all necessary information so as to generate proper error free code.

The SAX PARSER scans the XML Meta file and parses TAGs defined by RPB application development tool. It transforms the XLM TAGs into JAVA class definition as follows:

Class Customer

```

Private String Name;
Private Age;
Private List Sex []={"Male", "Female"};
Private String City;
Private int Pin;
Public Ok_onClick()
Business Logic for saving the fields
In DB table is embedded here.
Public Cancel on Click()
Logic for Cancellation of the action.
    
```

The JAVA is the default language RPB supports but developer can specify any OO language like C++ or C# supported by RPB.

**CAPTURING BL FROM UI**

RPB dynamically captures the details of the BL to be generated from UI. When developer drags and drops the button control from standard tool box, the property window of button control helps developer to specify all the operation specific BL like ADD, DELETE, SHOW or any other specific operation.

As these properties are stored properly in XML Meta file as depicted in fig-6, RPB BL insertion module will parse the essential details from XML Meta file and creates the appropriate BL query as per the BL operation so that proper action is performed by application at run time.

The RPB BL insertion module will parse the details like the data entered by user at run time, action to be performed on button click and the table in which the data is to be synchronized. Here data synchronization means performing operations like INSERT, DELETE, UPDATE, DISPLAY, CANCEL, etc. depending on BL. The module then generates complete code for operation to be performed on button click as shown below.

For INSERT/ADD operation following BL code is generated by RPB.

```

"INSERT INTO Customer(Name, Age, Sex, City, Pin) VALUES(" + new String(Name.getText()) + "," + new
Byte(Age.getText()) + "," + new String(Sex.getText()) + "," + new String(City.getText()) + "," + new
Long(Pin.getText()) + ")";
    
```

The values entered by user in text field are converted into the DB format of underlying DB automatically.

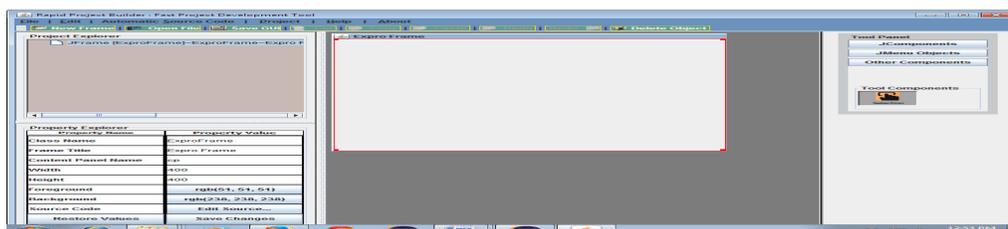
**VALIDATING CODE INSERTION**

This component performs automatic form validation. Validation engine makes validation of the text fields, text area, etc. of the frontend UI forms and the code generated for this is automatically inserted on event specific controls.

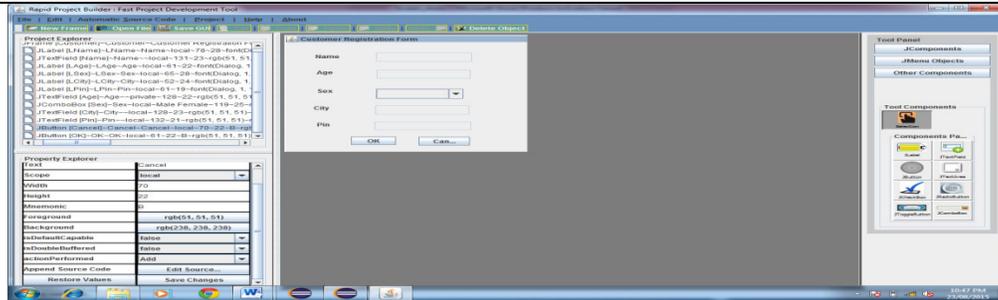
The proposed tool also provides facility to even customize the code and make the application redesigned as per the new project requirements.

**EXPERIMENTAL RESULTS**

The implementation of some of the modules of RPB is as shown in figures below.



**Figure 3. RPB Editor Environment with Form Design, Property window & Tool box**



**Figure 4. Customer Form desing using RPB editor**

The Source code and XML Meta information generated by RPB can even be saved separately in desired work space. These files can be used at any point of time by developer for further use. More specifically the source code file is file containing complete working code which can be executed on any machine having proper compiler of corresponding language.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<FORM>
<FORM_NAME>Customer Registration Form</FORM_NAME>
<CLASS>
<CLASS_NAME>Customer</CLASS_NAME>
<TEXTFIELD>
<TEXTFIELD_NAME>Name</TEXTFIELD_NAME>
<TEXTFIELD_DATATYPE>STRING</TEXTFIELD_DATATYPE>
<TEXTFIELD_DATASIZE>255BYTE</TEXTFIELD_DATASIZE>
</TEXTFIELD>
<TEXTFIELD>
<TEXTFIELD_NAME>Age</TEXTFIELD_NAME>
<TEXTFIELD_DATATYPE>INTEGER</TEXTFIELD_DATATYPE>
<TEXTFIELD_DATASIZE>1BYTE</TEXTFIELD_DATASIZE>
</TEXTFIELD>
<COMBOBOX>
<COMBOBOX_NAME>Sex</COMBOBOX_NAME>
<COMBOBOX_ITEM>Male Female</COMBOBOX_ITEM>
<COMBOBOX_ITEM>TestItem</COMBOBOX_ITEM>
<COMBOBOX_DATATYPE>STRING</COMBOBOX_DATATYPE>
</COMBOBOX>
<TEXTFIELD>
<TEXTFIELD_NAME>City</TEXTFIELD_NAME>
<TEXTFIELD_DATATYPE>STRING</TEXTFIELD_DATATYPE>
<TEXTFIELD_DATASIZE>255BYTE</TEXTFIELD_DATASIZE>
</TEXTFIELD>
<TEXTFIELD>
<TEXTFIELD_NAME>Pin</TEXTFIELD_NAME>
<TEXTFIELD_DATATYPE>INTEGER</TEXTFIELD_DATATYPE>
<TEXTFIELD_DATASIZE>4BYTE</TEXTFIELD_DATASIZE>
</TEXTFIELD>
<BUTTON>
<BUTTON_NAME>Cancel</BUTTON_NAME>
<BUTTON_MNEMONIC>B</BUTTON_MNEMONIC>
<BUTTON_OPERATION>Add</BUTTON_OPERATION>
</BUTTON>
<BUTTON>
<BUTTON_NAME>OK</BUTTON_NAME>
<BUTTON_MNEMONIC>B</BUTTON_MNEMONIC>
<BUTTON_OPERATION>Add</BUTTON_OPERATION>
</BUTTON>
</CLASS>
</FORM>
```

**Figure 5. XML Meta data of Customer Form**

**CONCLUSION**

The existing research done by various researchers and the tools developed by various global vendors like IBM, ORACLE, and Microsoft are limited to the design of UI and generation of code template in some vendor specific language and in platform dependent manner. The work presented in this paper is an innovation towards making the software project development simpler, efficient, and robust with low cost through our open source tool RPB.

The RPB generates code automatically in various languages like Java, C++, C#. If more language libraries are added in RPB, more will be the language support by RPB. At initial level DB support is given only for MS-Access and MySQL. The sole efforts of this research are to reduce software development cost thus contributing towards the

simplification of some of the processes of SDLC. The scope of RPB can be extended for development of web application as a future scope.

## REFERENCES

- [1] Abdeslam Jakimi and Mohammed Elkoutbi, Automatic Code Generation From UML Statechart, International Journal of Engineering and Technology Vol. 1, No. 2, June, 2009, 1793-8236, 165-168.
- [2] Yingxu Wang Xinming Tan, Cyprian F. Ngolah, Design and Implementation of an Autonomic Code Generator based on RTPA, 44 International Journal of Software Science and Computational Intelligence, 2(2), 44-65, April-June 2010
- [3] Muhammad Usman, and Aamer Nadeem, Automatic Generation of Java Code from UML Diagrams using UJECTOR, International Journal of Software Engineering and Its Applications Vol.3, No.2, April, 2009
- [4] George Edwards ,Yuriy Brun, Nenad Medvidovic ,Automated Analysis and Code Generation for Domain-Specific Models 2012 Joint Working Conference on Software Architecture & 6th European Conference on Software Architecture.
- [5] Prajkta R. Pawde, Vikrant Chole, Generation of Java Code Structure from UML Class Diagram, International Journal of Innovative Science and Modern Engineering (IJSME) ISSN: 2319-6386, Volume-2, Issue-7, June 2014
- [6] Herve Kabamba Mbikayi, Visual Composition and Automatic Code Generation for Heterogeneous Components Coordination with Reo. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 1, Issue 8, October 2012
- [7] Suvam Mukherjee, Tamal Chakrabarti, Automatic Algorithm Specification To Source Code Translation,
- [8] Amit Barve & Brijendra Kumar Joshi, Automatic C Code Generation for Parallel Compilation
- [9] Sukhvir Singh, Neeraj Kumar, Effort Reduction by Automatic Code Generation,
- [10] Harshal D. Gurad, Prof. V. S. Mahalle, Transformation Of Uml Sequence Diagram To Java Code.
- [11] Jeff Tsay, Christopher Hylands, Edward A. Lee, A Code Generation Framework for Java Component-Based Designs.
- [12] EL BEGGAR Omar, BOUSETTA Brahim, GADI Taoufiq, Automatic code generation by model transformation from sequence diagram of system's internal behavior, International Journal of Computer and Information Technology (ISSN: 2279 – 0764) Volume 01– Issue 02, November 2012.
- [13] Abdelouahed Kriouile, Najiba Addamssiri, Taoufiq Gadi, An MDA Method for Automatic Transformation of Models from CIM to PIM, American Journal of Software Engineering and Applications, 2015.
- [14] Harshal D. Gurad, V. S. Mahalle, An Approach to Code Generation from UML Diagrams, International Journal Of Engineering Sciences & Research Technology.
- [15] Model Based Software Development: Issues & Challenges, N Md Jubair Basha, Salman Abdul Moiz & Mohammed Rizwanullah, Special Issue of International Journal of Computer Science & Informatics (IJCSI), ISSN (PRINT) : 2231–5292.
- [16] Moskitt Framework and Bizagi Process Management Suite, Oskeol Gjoni, Comparison of Two Model Driven Architecture Approaches for Automating Business Processes, Mediterranean Journal of Social Sciences MCSER Publishing, Rome-Italy, Vol 6 No 2 March 2015
- [17] Andrés Muñetón Carlos Zapata, Dyna, Definition Of A Semantic Platform For Automated, Code Generation Based On Uml Class Diagrams And Dsl Semantic Annotations , year 79, Nro. 172, pp. 94-100. Medellin, april, 2012. ISSN 0012-7353
- [18] Jostp Maras, Automating Reuse In Web Application Development Automating Reuse In Web Application Development, 2014, Printed By Arkitektkopia, Västerås, Sweden
- [19] George. T. Vadakkumcheril, , M. Mythily, M. L. Valarmathi, A Simple Implementation of UML Sequence Diagram to Java Code Generation through XMI Representation.
- [20] Divya.A, and V. Renuka, A Novel Approach for Business Logic Evaluation Model in Web Service Environment, IJISSET - International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 9, November 2014
- [21] Stephen Cullum, Walton Hall, Milton Keynes, The Effect of Automatic Code Generation on Developer Job Satisfaction
- [22] Z. Hemel, L.C.L Kats, E. Visser, Code Generation by Model Transformation. A Case Study in Transformation Modularity Software Engineering Research Group, Delft University of Technology, The Netherlands

- [23] Mr. Sandeep Agarwalla, Ms. Priyanka Roy, Auto Generation Of Code And Table Tool, IJCSMC, Vol. 4, Issue. 4, April 2015, pg.487 – 492.
- [24] Chandan P & Rathod S D, Automatic Generation of Business Logic using DND, MJRET, 288-294, 1(3), October 2014
- [25] Chandan P & Rathod S D, An Approach towards Automatic Source Code Generation and Embedding Generic Business Logic Using DND, IJARCSSE, 513-521, 5(6), June 2015
- [26] Rathod S D & Joshi S D, A Fast Project Development Tool with Generic XML generation, NCAC-2015(7), 1-5 December 2015, IJCA, 0975-8887
- [27] Unified Modeling Language User Guide, The Grady Booch James Rumbaugh Ivar Jacobson, Addison Wesley ,ISBN: 0-201-57168-4, 512 pages
- [28] UML 2.0 in a Nutshell, By Dan Pilone, Neil Pitman, Publisher: O'Reilly, ISBN: 0-596-00795-7, Pages: 234
- [29] Java 6 Platform Revealed, John Zukowski, Apress, ISBN-13 (pbk): 978-1-59059-660-9, Pages 239
- [30] The Definitive Guide to Java Swing, John Zukowski, Apress, ISBN (pbk): 1-59059-447-9, Pages 913
- [31] Java and XML Data Binding, Brett McLaughlin, O'Reilly, ISBN 0-596-00278-5, Pages 214
- [32] [www.oracle.com/us/solutions/.../financial-mang-analytics-ds-501409.pdf](http://www.oracle.com/us/solutions/.../financial-mang-analytics-ds-501409.pdf)
- [33] <http://www.oracle.com/us/solutions/business-analytics/performance-management/financial-close-reporting/financial-management- analytics/resources/index.html>
- [34] <http://www.microsoft.com/en-in/dynamics/crm.aspx>
- [35] <http://www.interdynbmi.com/microsoft-dynamics-crm>
- [36] <https://msdn.microsoft.com/en-us/library/2x7h1hfk>.