A Comparative Study of Applications in Academic Projects Using Reverse Engineering Tools

Varun Singh Chauhan[1] Dr. Abhay Saxena[2]

[1]Lecturer, Deptt. Of Computer Science, DSVV Haridwar,

[2] Associate Professor, Head, Department of C.S., D.S.V.V. Haridwar.

***Abstract* – Purpose- The aim of this paper is to introducing a well defined framework for software reverse engineering, applicable in academic projects, reverse engineering tools, steps, and concepts.**

Design/methodology/approach- This paper presents an abstract model with some practical applications, using reverse engineering at application level.

Findings- Application software reverse engineering framework established in this paper can lessen the well known gap of software industry and academics. Using these concepts students and fresher/programmer can get a real touch of project development and integration.

Practical implications- Design and development capabilities of academic students will definitely grow by applying practical approach of this paper. Automated reverse engineering tools and guidelines of this paper are directly applicable in academics.

Originality/value- oftware reverse engineering has been used for many years for maintenance and source code generation (if lost). This paper expands these classical concepts to a wider range so that they can be applied in academic project development. Besides concepts and designing framework introduction to some reverse engineering tools are presented in this paper

.--♦--

1. INTRODUCTION

Reverse engineering can be used in many ways in the area of software development. Whether we are creating a genuine new system or modifying existing one , proper use of reverse engineering tools and techniques would definitely help during project designing and coding process. Reverse engineering has many areas of application related to software industry. [**1**]

Many academic projects suffer poor designing and coding techniques, and don't have a clear understanding about objectives of the projects .this common problem forms the well known gap between software industry and academic projects. Using software reverse engineering concepts and tools this gap can be decreased effectively. Students can improve their level of development, designing and coding. Reversing process can be automated in the some extent with the tools available for modern programming languages such as c# .NET Dis#(http://netdecompiler.coml Reflector(www.reflector.com) [**2**]. Using these tools coding and designing information about the system can be easily obtained, and used in overall reverse engineering process. A complete framework has been established in this paper including concepts, steps and techniques for reverse engineering, which definitely lessen the gap between software industry and academics.

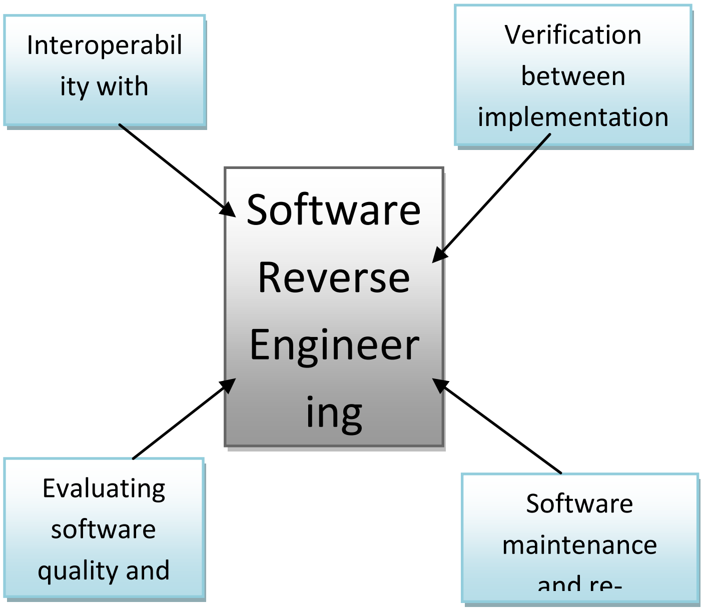
COMMON BELIEF

At first glance, it may seem that the need for SRE can be lessened by simply maintaining good documentation for all software that is written. If we provide the complete documentation of the system, it will definitely decrease need of that system. For example, even a company that has brought software to market may no longer understand it because the original designers and developers may have left, or components of the software may have been acquired from a vendor who is no longer in business. Need for reversing a project is never affected with documentation.

SOFTWARE DEVELOPMENT AND REVERSE ENGINEERING:

Although revering has its great use in designing and developing genuine systems, yet it can be used in harmful purposes like password cracking, unauthorized access, network intrusion and website hacking etc.

Software Development process can be related with reverses engineering in the following cases.


Interoperabil ity with
Verification between implementation
Software Reverse Engineer ing
Evaluating software quality and
Software maintenance and re-

- **Interoperability with proprietary Software:** Develop applications or device drivers that interoperate (use) proprietary libraries in operating systems or application. During the development of a new system there may be situations when we have to interact with third party software for which no code is available now, this situation gives rise to the need of software reverse engineering .Effectively using reversing concepts codes are created to interact with proprietary software.[**3**]

- **Verification between implementation and design:** Verify that code produced during the forward development process matches the envisioned design by reversing the code back into an abstract design. Since software development is a complex process, chances are exist whether the goals meets design and implementation .After implementation a project there may be need to evaluate its actual performance and design goals.

- **Evaluating software quality and robustness:** Ensure the quality of software before purchasing it by performing heuristic analysis of the binaries to check for certain instruction sequences that appear in poor quality code. Many antivirus companies use this technique to ensure the binary pattern of a virus. Security checks can also be evaluated with the help of dissembler; low level memory operations can also be verified. Based on some previous experience quality indicator can be checked at low level.

- **Software maintenance and re- engineering:** Maintenance requires more effect than developing the system itself. If the source code is lost, we have no options but reversing the whole project. Reverse engineering process can **recover** the design of legacy software module when source is not available to make possible the maintenance, evolution and reuse of the modules.

IMPORTANCE OF SRE IN ACADEMIC PROJECT DEVELOPMENT:

Reverse Engineering concepts can easily be applied in the development of a new academic project or in the maintenance of a legacy system**.** We can understand the design and functionality of the system while source code is not available. Although reverse engineering can effectively be used in many industry based projects, yet the concept of reverse engineering is completely applicable in academic projects.

HINDRANCE IN ACADEMIC PROJECT DEVELOPMENT:

Development of a project is never an easy task .The hindrances comes in many forms and flavor .While development of any project, it is a common challenge almost for the development final year students and the beginners. Some of the common problems can be stated as …

1) Projects don't have a clear objective.
2) Projects are persistent, repeatable nature .
3) Hardly a project implements innovation.

4) Short time schedule forces more concentration on coding rather designing of the project.

5) Probably 50% projects never completed during academic project schedule.

6) Almost every academic project has some serious bugs and design weakness.

7) Hardly an academic project posses industry standard in its design and coding.

A Study shows that 51% academic projects have some of the above mentioned problems. These problems form the well known gap between industry and academic [**4**]. Intelligent use of reverse engineering can remove many of the above mentioned problems of academic projects. Proper use of reverse engineering tools and techniques can lead good design and coding process. While developing new projects a lot of designing and internal working information can easily be obtained by use of reversing a legacy product carefully .According to our requirements the extracted information can be applied intelligently to the new projects.

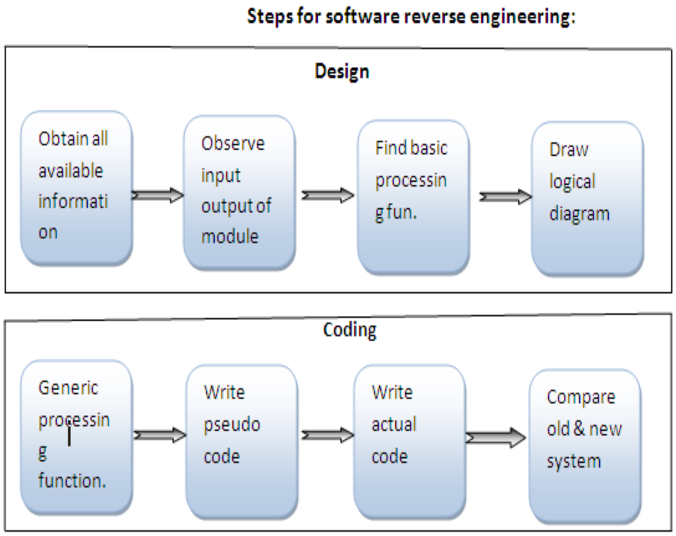
SOFTWARE REVERSE ENGINEERING: HOW IT IS DIFFER FROM SOFTWARE ENGINEERING

Software reverse engineering is differ from software engineering as-

Software Reverse Engineering	Software Engineering
Applies in modify or maintenance of the existing system	Applies in creating a new system
Evaluate goals and objectives of the system after implementation	Goals and objectives are implemented in the system
No use of life cycle methods	Various development life cycles are available
Extensive use of automated tools	Less use of automated tools
Documentation is not available.	Provide well documentation ,client reviews etc

Steps for reverse engineering a software module:

Reverse engineering in and of itself does not involve changing the subject system. It is a process of examination, not change or replication. **[5]** Before we go for reengineering targeted system it is good idea to clearly identify all the modules of the application. Software reengineering requires some prerequisites steps such as collecting all the possible information about the system itself, searching for documentation (if available) etc.


Steps for software reverse engineering:
Design
Obtain all available informati on
Observe input output of module
Find basic processin g fun.
Draw logical diagram
Coding
Generic processin g function.
Write pseudo code
Write actual code
Compare old & new system

1. Obtain all the available information about the system to be reengineered.-

The very first step is to obtain all the possible information about the project to be reengineered .Preliminary information includes documentation, third party information about the project, user views and inbuilt help module. All this information should be collected in an easily understandable format**. [6]**

2. Carefully observe the input and output of the given module or software -

After collecting all the related information the second step should be carefully observe the input and output data flow of the of the current module we want to reengineered . I/O sequence can help to better understand the processing functions of the module.

3. Try to understand the basic processing function of the given module.

Analyzing input output sequence carefully, note down basic model of input output data flow certainly helps in understanding the processing function of the module. Generic processing function can be designed for data flow.

4. **Draw a logical design diagram from the observations of the module.**

Having done with preliminary analysis of the module we can design a visual presentation for that module, diagram may show data flow of the module in graphical format. Understand and extract [5]the design of targeted system. **[7]**

5. **Construct a generic processing function from the logical design.**

Now pseudo code can be written for every function of the module. There is no need till now to choose a specific language, generic types and methods are very useful to understand generic processing function.

6. **Write the function or algorithm in pseudo code.**

The underlying algorithm can be coded in pseudo code at this level[9]. This later can easily be translated into any programming language. Functions are coded without using any specific type or syntax; it works just like a blueprint of the actual function.

7. **Implement the algorithm in a suitable programming language.**

Any suitable programming language can be selected to code the real working module or function obtained from the above steps .Higher level language can provide some support when we go to write actual code.

8. **Carefully compare the working of new module with old one.**

As a final step some comparison is needed between the module created by reverse engineering and the original working module module , if found some changes ,then the process would have to be repeated for better implementation of reverse engineering concepts.

Obviously some intuition is required for proper implementation of reverse engineering concepts. Beside all these steps there are also some tools available which can help during reversing software.

A CASE STUDY:

Objective: Understanding the design of an application program and hence changing the code according to our requirements or using this knowledge in creating a new legacy system. Our targeted system is ***Magazine Subscription management*** developed in C#.NET 2005 and MS-Access as backend database.

This simple module manages the subscription data of a Magazine. This module saves data of customers and prints some simple reports based on the selection. Many tools are available for browsing class information, and source codes for c# .NET.

TOOLS:

***.NET Reflector**:*

.net reflector is a very powerful class browser, analyzer and decompiler for .net environment

Reflector can be used for showing the assembly, class and function information of the .NET module

The latest version of the Reflector can be obtained from: *http://www.aisto.com/roeder/dotnet*.

Dis#:

Dis # is also a class browser, analyzer and decompiler for c# .net environment. *Dis#* allows editing names in decompiled code and persisting changes in project file. Dis # can be obtained from the following link: http://netdecompiler.com/download.html.

As we know c# is a popular language for application development now a day's .it is widely used in academic and professional environment for developing many database driven applications.

Procedure for dissecting a .net application using Dis#

1) Analyze and identify all the modules of the application

2) Disassemble the module using .NET Reflector

3) Understand and change the hidden code of the module [function]

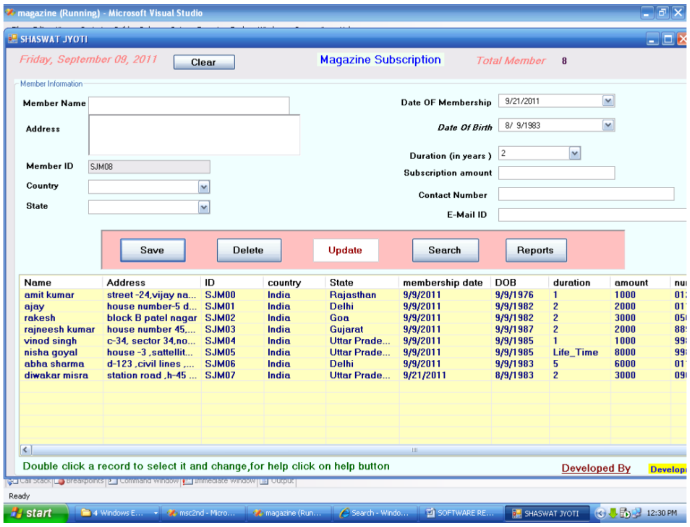
4) Recompile and understand the functioning of the code to use it.

Many tools are available today for decompiling and analyzing code written in high level languages such as c# .These tools are powerful and sophisticated to work as class browser, decompiler and are very helpful in software reverse engineering **[8]**. Following two tools are very popular for c# .net project reverse engineering.

Using Dis# for c#.NET project decompilation-

Dis # can translate c#.net assembly into human readable formate.it can decompile classes, methods, emus, and other elements of c# .NET compiled project. It has a very impressive interface and for decompiled classes and methods.

The primary interface for the application is shown in the figure 1 below.



Friday, September 09, 2011

Magazine Subscription

Total Member 8

Member Information

Member Name		Date OF Membership	9/21/2011
Address		Date Of Birth	8/ 9/1983
		Duration (in years)	2
Member ID	SJM08	Subscription amount	
Country		Contact Number	
State		E-Mail ID	

Name	Address	ID	country	State	membership date	DOB	duration	amount	nu
amit kumar	street -24,vijay na...	SJM00	India	Rajasthan	9/9/2011	9/9/1976	1	1000	01
ajay	house number-5 d...	SJM01	India	Delhi	9/9/2011	9/9/1982	2	2000	01
rakesh	block B patel nagar	SJM02	India	Goa	9/9/2011	9/9/1982	2	3000	05
rajneesh kumar	house number 45,...	SJM03	India	Gujarat	9/9/2011	9/9/1987	2	2000	88
vinod singh	c-34, sector 34,no...	SJM04	India	Uttar Prade...	9/9/2011	9/9/1985	1	1000	99
nisha goyal	house -3 ,sattellit...	SJM05	India	Uttar Prade...	9/9/2011	9/9/1985	Life_Time	8000	99
abha sharma	d-123 ,civil lines ,...	SJM06	India	Delhi	9/9/2011	9/9/1983	5	6000	01
diwakar misra	station road ,h-45 ...	SJM07	India	Uttar Prade...	9/21/2011	8/9/1983	2	3000	09

Double click a record to select it and change,for help click on help button

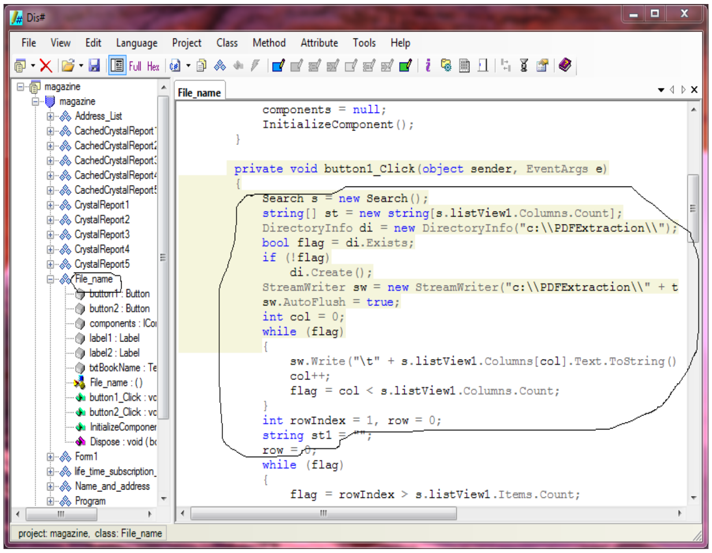
Developed By Develop

Figure 1

When we open this ***magazine subscription management t*** project developed in c# .NET and MS Access, with Dis# firstly it looks like as following figure:

This simple application works with MS Access database as background. Subscription detail for magazine customers can be saved searched, and some reporting features. This small application is a good example to work as a module of large project. Dis# is capable to browse all the information about classes, forms and reports of the module.

By selecting any class, forms or reports (browsed in a clear way) we can see the code behind it (as in Figure 2) .Properties classes and forms are well arranged in a hierarchal manner so it is easy to browse and locate the desired object. When we select an object in the left pane of Dis# tool it can unhide the background code of that object. As we can see in the following snap-



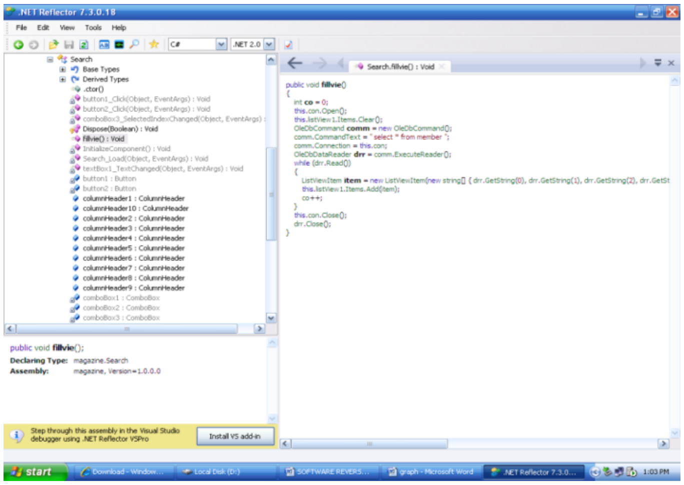
```
            components = null;
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Search s = new Search();
            string[] st = new string[s.listView1.Columns.Count];
            DirectoryInfo di = new DirectoryInfo("c:\\PDFExtraction\\");
            bool flag = di.Exists;
            if (!flag)
                di.Create();
            StreamWriter sw = new StreamWriter("c:\\PDFExtraction\\" + t
            sw.AutoFlush = true;
            int col = 0;
            while (flag)
            {
                sw.Write("\t" + s.listView1.Columns[col].Text.ToString()
                col++;
                flag = col < s.listView1.Columns.Count;
            }
            int rowIndex = 1, row = 0;
            string st1 = "";
            row = 0;
            while (flag)
            {
                flag = rowIndex > s.listView1.Items.Count;
```

Figure2

The above screen shot of Dis# shows the code of method "file_name" .All the extracted code are well formatted and self described, and can easily be understood by reading carefully .Classes ,functions and properties can be documented manually during the code reveal process.

Here we can create dummy application by using RAD (rapid application development) tools and can obtain a lot of information about designing a coding of the application. **[8]**

Using .NET Reflector: When we open the current module with another tool, it looks like -


.NET Reflector 7.3.0.18
File
Edit
View
Tools
Help
C#
.NET 2.0
Search
Base Types
Derived Types
.ctor()
button1_Click(Object, EventArgs) : Void
button2_Click(Object, EventArgs) : Void
comboBox3_SelectedIndexChanged(Object, EventArgs) : Void
Dispose(Boolean) : Void
fillvie() : Void
InitializeComponent() : Void
Search_Load(Object, EventArgs) : Void
textBox1_TextChanged(Object, EventArgs) : Void
button1 : Button
button2 : Button
columnHeader1 : ColumnHeader
columnHeader10 : ColumnHeader
columnHeader2 : ColumnHeader
columnHeader3 : ColumnHeader
columnHeader4 : ColumnHeader
columnHeader5 : ColumnHeader
columnHeader6 : ColumnHeader
columnHeader7 : ColumnHeader
columnHeader8 : ColumnHeader
columnHeader9 : ColumnHeader
comboBox1 : ComboBox
comboBox2 : ComboBox
comboBox3 : ComboBox
Search.fillvie() : Void
Step through this assembly in the Visual Studio debugger using .NET Reflector VSPro
Install VS add-in
start
Download - Window...
Local Disk (D:)
SOFTWARE REVERS...
graph - Microsoft Word
.NET Reflector 7.3.0...
1:03 PM

```
public void fillvie()
{
    int co = 0;
    this.con.Open();
    this.listView1.Items.Clear();
    OleDbCommand comm = new OleDbCommand();
    comm.CommandText = " select * from member ";
    comm.Connection = this.con;
    OleDbDataReader drr = comm.ExecuteReader();
    while (drr.Read())
    {
        ListViewItem item = new ListViewItem(new string[] { drr.GetString(0), drr.GetString(1), drr.GetString(2), drr.GetSt
        this.listView1.Items.Add(item);
        co++;
    }
    this.con.Close();
    drr.Close();
}
```

public void fillvie();
Declaring Type: magazine.Search
Assembly: magazine, Version=1.0.0.0

Findings of the case study:

Results are comparable in the following cases

1. Since reflector supports different version of .net framework (including framework 1.0 to latest version i.e. 4.0) so if the targeted system updated, changes can be reflected easily.

2. Reflector can decompile different languages selected by the user.

Depending upon the requirements any of the tools can be selected to decompile the c#.net application. If the project is complex and developed using different .NET supported language then Reflector would preferred.

RESULT

This simple case study shows the traditional power of reverse engineering in the area of academic project development. Carefully using these tools and techniques student can get a real touch of industry based projects which can definitely improve their level of knowledge and understanding the working of a good software project. These concepts can lessen the gap between software industry and academics, as well as student can learn the complex process of designing and integrating industry oriented applications using available tools.

REFERENCE

[1] http://reversingproject.info/ as retrieved on 27th September 2011.

[2] *http://netdecompiler.com/ “owner of Dis# decompiler”* as retrieved on 27th September 2011.

[3] CEM KANER, J.D., (1998). Ph.D. ARTICLE 2B and REVERSE ENGINEERING, http://www.kaner.com/pdfs/ RevEngShort.pdf as retrieved on 27th September 2011

[4] www.it-cortex.com (The Robins-Gioia survey, 2001), as retrieved on 25 September 2011.

[5] Chikofsky, E.J.; J.H. Cross II (1990). "Reverse Engineering and Design Recovery: Taxonomy in IEEE Software". *IEEE Computer Society*: 13–17

[6] SOFTWARE—PRACTICE AND EXPERIENCE, VOL. 21(12), 1349–1364 (DECEMBER 1991)

[7] SOFTWARE—PRACTICE AND EXPERIENCE, VOL. 21(12), 1349–1364 (DECEMBER 1991) Software Reverse Engineering: A Case Study, ERIC J. BYRNE, Department of Computing and Information Science, Kansas State University, Nichols, Hall, Manhattan, Kansas 66506, U.S.A.

[8] Paolo Tonella and Alessandra Potrich (2005), Reverse *Engineering of Object Oriented Code*, Springer, New York, USA, ISBN: 0-387-40295-0.