CHAPTER VI
EXPERIMENTS AND RESULTS
CHAPTER VI
EXPERIMENTS AND RESULTS

A key step in the formulation of any optimization problem is the assignment of performance measures that are to be optimized using GA, ACO, PSO, FL and NN. The success of any optimization result is critically dependent on the selection of meaningful performance measures. In many cases, the actual computational solution approach is secondary. Ways in which multiple performance measures can be incorporated in the optimization process are varied.

Softcode uses GA, ACO, PSO, FL and NN.

Visual basic with third party tools listbar and spread for the front end
MsAccess for the Reports

Hence the following are to be installed for site development

- Visual Basic 6.0
- MS SQL server 2005. This is required only when installation of MSDE becomes a failure.
- MSDE2000A, Dba Manager Spread, Listbar, Mdac2.7, DateControl.

6.1 SQL SERVER DATABASE

6.1.1 MS SQL SERVER AND DATABASE TOOLS

Soft code uses the following servers:

Ms Sql server-Sql Express 2005

Database manager –Ms Sql studio 2005

6.1.2 DATABASE TOOLS

SQL server management studio Express 2005 version-This is the most commonly used tool for performing the following operations.
- Attach database
- Detach database
- Backup Database
- Restore database
- Create tables, Views
- Enter, Edit data

**Sql query analyzer** – This is a handy tool for analyzing queries.

### 6.2 SOFTCODE DATABASE AND TABLES

#### 6.2.1 FORMS DATABASE – SOFTCODEFORMS

This database is the meta databases which contain all the form parameters. Forms are dynamically created by taking values from the tables mentioned below. The following are the primary tables and fields used.

**FORM MAIN**

This contains forms/Reports main information as follow.

- Formcode-alpha numeric code ex-FOMG460, ROMG470
- First digit ‘F’ for forms ‘R’ for reports
- OMG is short form for ERP module Order management
- Ftitle-Title of Rform or Report
- MenuCategory-Valid entries are stored in module master
- erpModule - Valid entries are stored in module master
- formname-this is the template name and the valid entries are
  - ✓ Newform2001 – master form
  - ✓ Newform2002 – master/detail form
✓ FlexiReport – Report Interface
✓ Hdr0 – name of header table
✓ Dtl0-name of detail table 1
✓ Dtl1-name of detail table 2
✓ Tab0-tab caption user preference

Remaining fields are not mandatory for forms.

6.3 MAIN DATABASE – SOFTCODE

All the client data is stored in this database. The following two tables are mandatory for running Softcode.

- **CompanyMaster**
  Contains Companycode, Report paths for various types of reports.

- **Combomaster**
  Contains list of commonly used values required for comb boxes in form.
  Example: Male/Female, Yes/No.
  Other data tables – As per Domain requirements.

6.4 POSTGRADE DATABASE

**VB VERSION**

Transfer options

- Uploading only

- Both Uploading and Downloading

**MODES OF TRANSFER**
• Data

• Transql – here sql statements are transferred and executed at destination.

REQUIREMENTS

• Internet or Dialup connectivity for remote databases.

• Fixed IP address at least on one side.

• If both sides have dialup ordinary connection then IP address at the time of exchange shall be obtained for transfer of data. This arrangement can be only a temporary.

CONNECTING TO SOFTCODE

Ensure that Softcode has set up, data transfer screens administration the following four tables are to be present in the Softcode database. Set up provides remote database connection information. Data transfer Tables (for uploading) select the required table for data transfer thro data transfer tables screen.

Data transfer download Tables (for downloading) – select the required table for data transfer through data transfer tables screen.

Data transfer – data is transferred through this table through sql statements.

SET UP
Figure 6.1 Softcode-Set up

SET UP

SELECTING THE OPTION ORACLE/SQL SERVER

Comment the appropriate code for the option. This code is a part of softcode

Dim Clientrcset1 as RecordSet

Dim ClientConn as ClientServerConnection

Dim SqlCon as SqlConnection

Public Sub ConnectToMaster ()

Dim connstr As String

Dim Server as String

Dim UId as Number

Dim Pwd as String

Dim dbq as String
For Oracle

```
strSql = "Select * from SetUpTable"
Set clientrcset1 = ClientConn.Execute (strSql)
Connstr = Trim (clientrcset1 ("connection string"))
Server = Trim (clientrcset1 ("Server"))
Uid = Trim (clientrcset1 ("UID"))
Pwd = Trim (txtPassword)
dbq = Trim (clientrcset1 ("Database Name"))
Provider =Trim (clientrcset1 ("Provider"))
If UCase (clientrcset1 ("Connection Type")) = "ODBC" Then
    Connection Using ODBC
    Connstring = connstr & "server=" & Server & ";" & "Uid=" & Uid & ";" &
            "Pwd=" & Pwd & ";" & "dbq=" & dbq
Else
    Conection Using Provider
    Connstring = connstr
    Connstring = "Provider=" & Provider & "; server=" & Server & ";" & "User id=" &
        Uid & "; Password=" & Pwd & ";" & "data source=" & dbq
End If
```

For sql server

```
"MastConn.ConnectionString = <use the standard connection string>
"MastConn.Open" only for sqlserver
MastConn.Open connstring
'MsgBox "Connected To Central Database! "vbInformation, "Connection Established"
```

clientrcset1.Close
Set clientrcset1 = Nothing

Exit Sub

Transfer Logic

1. Uploading - From client to main database data is appended/updated
2. Downloading-deleting (optional) the destination table data and data transfer is done in specific cases.

6.5 UPGRADE SOFTCODE

Upgrading Softcode main the following parameters are specific to Client site.

CONNECTION INIS

Files there are two files giving connection info to main database and forms database respectively. These files are kept in the folder connectionInis folder inside Softcode folder. Hence existing file is to be retained during over writing the folder.

CUSTOM MODULE

This module is kept outside Softcode main folder. However after upgradation custom module to be reattached after removing the standard custom module in Softcode project.

REPORT DATABASE

This is an Mdb database having all the reports of the client and it is inside Softcode folder. Hence it is to be retained after upgradation.

Company master set the flag to False. A message box will appear indicating upgrade mode.

A back up of the full existing folder is desirable before updation.
6.6 FORM STRUCTURE

Using Softcode Administration the following types of forms can be built

- Master or Header forms
- Header/Detail forms

Header forms contain controls like text boxes, combo boxes and date Pickers.

Detail form contain only Grid.

Cells in the grid may contain Combo boxes, date pickers.

In general Header form data is saved in <abc> hdr table and Detail grid data is saved in <abc> Dtl table. Hence while building form these tables are selected respectively for listing fields for selection. Id is the common link field between header and detail table Forms automatically recognizes the common field for linking data. Hence presence of two common fields in above tables are prohibited to avoid confusion.

PROPERTY GRIDS IN FORM BUILDER

Property Grids are displayed when tables are selected in the admin form builder. In the Property Grids field properties can be defined.

Example

By default all fields are Textboxes/cells .They can be made as combo or Datepickers by giving a value ‘Yes’ to Combovar or Datevar field in the property grid.

6.7 HELPQRY, COMMAND BUTTON QUERY

These two are important parameters which are frequently used HelpQry, CommandButtonQuery are used for speeding up data entry in forms. Both serve the same purpose but used in different modes as explained below.
In helpqry values are selected from stored data in single record mode.

In commandbuttonquery a set of records are brought to the grid directly. Commandbuttonquery is applicable only for grid in the form where it can contain multirecords.

6.7.1 HELP QUERY

HelpQry - This is a field in FormHdr which provides List Of Values (LOV) help data for selection mainly from master tables for header form. Help query is also used for Id field for opening saved records.

Ex - Select employee name, designation, department, Employee No from employee Master.

Help query is used in conjunction with the following fields.

Mast - Name of the master table from where data is fetched.

Ex - Employee Master

Code - Name of the field name whose value to be selected and entered in form.

Ex - EmployeeName

MultiFieldHelpQry - In a number of Situations fields related to code are also required to be displayed.

Ex - Department and employeeNo are to be displayed or to be captured in database.

Syntax –
EmployeeName^staffName^Department^Department^employeeNo^staffNo.

DhelpQry - This is a field in FormDtl which provides LOV help data for selection mainly from master tables for Detail grid in the form. DHelp query is used in conjunction with the following fields.
Ex - Select emlyeename, designation, department, EmployeeNo from employee Master.

DtlMst - Same as Mast explained above but applicable for Detail table

KeyDtlmst - Same as code explained above but applicable for Detail table

dhelpfieldDisp - Same as MultiFieldHelpQry explained above but applicable for Detail table

Syntax – this differs from multiFieldHelpqry. Only target fields are listed as below

StaffName^Department^staffNo

CONDITIONS

- MultiFieldHelpQry and dhelpfieldDisp-Helpqry/DhelpQry should contain all fields listed in MultihelpQry/dhelpfieldDisp.

- Sequence of fields in DhelpQry should be same as dhelpfieldDisp list.

6.7.2 COMMAND BUTTON QUERY

Like help query in a number of situations Data from a previous transactions are to fetched into the detail grid. The sql query entered in CommandbuttonQuery field does this function.

Ex-Select itemcode, itemdesc from itemMaster

CommandButtonQuery is a field in formmain and is used in conjunction with the field QueryFields –here name of filter field is entered in the following syntax.

CategoryCode^groupCode - CategoryCode is the source field name and groupCode is the Target field name.
6.8 TABLES AND FIELD NAMING CONVENTIONS

The following conventions are used while designing new /editing existing tables.

TABLES

• First letter should be in caps.

• Spaces not allowed.

• Header table should suffix with ‘Hdr’

A typical master detail form will have a minimum of two tables. The master table is called Header table and the detail as detail table

Example: StudentAppHdr

• Detailed table should suffix with ‘Dtl’

Example: StudentAppDtl

FIELDS

• First letter should be in caps.

• Spaces not allowed.

• When combining two words first letter of second word should be in caps (e.g.: FatherName).

• Id should be created in first row

  ✓ Id Field should be created(Primary key in header table)

  ✓ Id field should be created(foreign key in detailed table)

• CompanyCode should be created in all header tables.

DATA TYPES

• For Text Field select-nvarchar.
• For Number field select float (also support integer).

• For date Field Select-Datetime.

VIEWS

View name will end with a suffix ‘View’.

Body will contain the type of view and the types are given below

• **Report views** – View name will have a prefix ‘Rpt’. Body will contain the Report title without spaces. Examples are given below
  
  ✓ RptPurchaseOrderView
  ✓ RptGrnView

• **Help Query Views** – View will have a prefix Hlp. Body will contain the description of Help in short. Examples are given below
  
  ✓ HlpExciseItemsView
  ✓ HlpPoPendingItemsView

• **Qty/Amount Balance views** - View will have a suffix ‘balanceView’. Body will contain balance description. Examples are given below
  
  ✓ BatchStockbalanceView
  ✓ RgStockbalanceView
  ✓ IndentbalanceView
  ✓ PObalanceView
  ✓ ProjectBudgetBalanceView

• **Transaction Union views** - Here tables are combined in union views. View name will contain transaction description. Examples are given below
  
  ✓ StockOpeningView
  ✓ StockReceiptView
StockIssueView

- Header Detail Link Transaction views (Optional) - these are supporting views used for all the above views. Transaction Description is prefixed.

REPORTS

Report name will be same as Ftitle (Report title) given in form main.

RepTable – This is a temporary data table in the MsAccess report database. The table will have a prefix ‘Rpt’.

REPORT BUILDING

Step 1: Creating Report Main Record

- This step is common for all types of reports. This can be achieved in two ways.
  
  ✓ Using report builder in Softcode Administration.

  A record is created in formmain with formcode starting with R.

  ✓ By directly creating a record in formmain

The following fields are entered with values.

1. Formcode
2. Formname
3. ERPModule
4. Menucategory
5. Formname
6. Querystring
7. Reptable - This can be any value preferably with a prefix Rpt for identification.
8. This is a SQL query which can be created using enterprise manager. This query is stored in formmain in the field QueryString for the report formcode.

9. Repeatable – This can be any value preferably with a prefix Rpt for identification.

---

**Fig 6.2 Report Creation**

**Step 2: Assigning Report to user menu**

Softcode admin is used to assign the form/report to a particular user.
REPORT DESIGN IN MS ACCESS

Step 3: Report Design in Ms Access

- Create RepTable

A report table (Reptable) is created in access database for inserting records obtained in querystring (a field in formmain) built for the report. The following fields are to added to the query fields.

✓ ReportFromDate(Datetime)

✓ ReportToDate(Datetime)

The above table should be entered in formmain in Reptable field.

A report is created using access report designer using the above table - preferably using wizard for making grouping easier. The name of the report should be same as file in formmain entry made in step1.

SNAPSHOT VIEWER

Ms Access reports can be viewed in Snapshot viewer. The advantage of this method is the Report can be emailed directly this is to be installed in all clients.

In company master make the following entries.

- In AccessRunPath field enter the path of runtime msaccess.exe which gets installed during installation.

- Example.-C:\Program Files\Microsoft Office\OFFICE11\msaccess.exe.

- In ReportPath enter the access path where reports exist.

  Example
  
  C:\Report\Report.mdb
1. An entry in accessRunpath automatically changes the report to Access Run time mode.

2. In formmain make the following entries

   ✓ In Formname field enter ‘Flexi’.
   ✓ In ReportType field enter ‘access’.

3. For viewing Reports through snapshot viewer in formmain

   ✓ In Snapshot field enter ‘yes’.

REPORT PATH

MS ACCESS

Access reports can be run in two modes –

a. Normal mode - this requires Ms access to be installed (Normally available with Ms Office)

b. Run time modes – Doesn’t require access installation. Only Run time library to be installed.

Access in Run time

c. This is to be installed in all clients. Install Ms Access run time (ACCESSRT) available in the download menu.

d. Install ‘Remove Security warning (Security off’ (msecure10) in order to avoid Access Warning message boxes during report opening.

6.9 DATABASE ADMINISTRATION

6.9.1 TABLE DESIGN

For creating a form linking header and detail tables, there has to be common link field Softcode automatically recognizes this common field and links records accordingly.
Ensure that there are not more than one common field as more than one may result in error situation. Choose the right data type of primary id field based on type of serial coding system to be followed ex- auto serializing using either number or alpha numeric or composite structure combining form fields.

6.9.2 EXPORTING TABLES FROM ACCESS TO SQL SERVER

This can be done by three methods.

1. By using access export option (available on the top menu under file) – suitable for single table export

2. By using upsizing wizard (under Database utilities) available in MS access ideal for multi table uploading


6.10 COMMON FUNCTIONALITIES IN FORMS – BALANCE DISPLAY

6.10.1 BALANCE DISPLAY

Balance display in form Balance can be Quantity or Amount depending upon the situation. In a stock situation it's quantity and in a accounting situation it is Amount. Balance involves two transactions. First transaction adds to the balance value and the second transaction reduces the value. - Examples

QUANTITY

• Batch Stock balance

• RgStock balance (PoExciseInvoice)

• Indent balance

• PO balance
AMOUNT

- Budget Balance

Method 1 – Using multi update feature

Display of Balance in detail grid is explained below

- A field namely Balance is added to the first transaction table (IndentDtl, GRNdtl, Budget Dtl etc)
- This value is updated every time the second transaction takes place
- Updating is done through the feature ‘How to update values in Multitables’
- The balance value can be displayed in the second transaction form as a display field through helpQry / DhelpQry along with MultiFieldHelpQry / dhelpfielddisp.
- Validation for the second transaction can be tied up to the balance value through the feature ‘How to validate values in Header/Detail’.

Example – Issue cannot be done when stock balance is 0

Method 2 – Using Views (Dynamic balance-preferred method)

Transaction values are summed up in a View and the balance is calculated at view level. This method is error free as it takes care of changes and cancellations without adjustment process as in the case of multiupdate method.

SELECT CustomerCode, itemcode, RG23DNo, SUM (QtyReceived) AS QtyReceived, SUM (IssuedQty) AS QtyIssued
FROM dbo.V_SaleExciseInvoice
GROUP BY CustomerCode, RG23DNo, itemcode
6.10.2 BALANCE DISPLAY – INVENTORY (PURCHASE BALANCE)

SELECT dbo.povw.itemcode, dbo.povw.qty as PoQty, (dbo.povw.qty-DerivedTable.ReceivedQty) as BalQty
FROM dbo.povw Left Outer JOIN
( SELECT pono, itemcode, SUM(qty) AS ReceivedQty
FROM dbo.grnview
GROUP BY pono, itemcode) AS DerivedTable
ON dbo.povw.pono=DerivedTable.pono

6.10.3 BALANCE DISPLAY – INVOICE

SELECT dbo.SalesInvoiceHdr.InvNo, dbo.SalesInvoiceHdr.InvDate, ISNULL(dbo.SalesInvoiceHdr.GrandTotal, 0) AS InvAmt, ISNULL(dbo.SalesInvoiceHdr.GrandTotal,0)-ISNULL(PaidAmt.ReceivedAmt,0) AS InvBalance, dbo.SalesInvoiceHdr.PartyAccountCode
FROM dbo.SalesInvoiceHdr LEFT OUTER JOIN
(SELECT InvNo, SUM(ISNULL(Current Amount, 0)) + SUM(ISNULL(Advance Adjusted, 0)) AS ReceivedAmt
FROM dbo.VoucherDtl
GROUP BY InvNo) AS PaidAmt
ON dbo.SalesInvoiceHdr.InvNo = PaidAmt.InvNo
WHERE(ISNULL(dbo.SalesInvoiceHdr.GrandTotal,0)-ISNULL(PaidAmt.ReceivedAmt0)> 0)

Invoices are generally made in the Order management and purchase modules. They differ from company to company Hence the following parameters are to be changed.

6.11 DOT NET FORM CONVERSION
But for custom Procedures Dotnet version of Softcode is supposed to open all forms operated in VB. However error can happen in the following situation

1. When first field is string and composite key is used for creating auto generation code, remove autogen Yes from formmain. Retain only hdrAutogen “yes” in formHdr.

2. In controlTable null values or string data type in the following fields will produce errors. Change the datatype and values. Delete the records and reformat using format builder.

3. Use flexi report interface for Ms Access reports.

4. Hard coded Vb procedures are to be converted to custom procedures if the features are not covered in dotnet parameters.

6.11.1 COMPANY CODE IMPLEMENTATION

In new menu2004 table form and report menu should be assigned for each company code. Then company code will be available in login form for selection.

All header tables and hence header forms will have company code text field. Value for company code is defaulted from login screen. Only data pertaining to the company alone will be available for selection in the form for Viewing and editing. However by including the table in the Common Masters will make available the data pertain to all companies. This feature will be helpful for sharing certain master data like Item Master etc.

REPORT DATA

QueryString must have companycode in the selected field.

In formmain in addcompanycode parameter give the value “yes”.

6.12 SOFTWARE DESCRIPTION

6.12.1 MICROSOFT SQL SERVER 2005
Microsoft SQL Server is a relational database server, developed by Microsoft: it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). There are at least a dozen different editions of Microsoft SQL Server aimed at different audiences and for different workloads (ranging from small applications that store and retrieve data on the same computer, to millions of users and computers that access huge amounts of data from the Internet at the same time). True to its name, Microsoft SQL Server's primary query languages are T-SQL and ANSI SQL.

SQL Server 2005, Microsoft's next-generation data management and analysis solution, represents a huge leap forward. It comes with a myriad of changes that deliver increased security, scalability, and power--making it the complete data package. Used properly, SQL Server 2005 can help organizations of all sizes meet their data challenges head on.

Each Microsoft SQL Server 2005 component is designed to have a unique architecture and life cycle. The two primary areas that can affect an upgrade path. Some SQL Server 2005 components build on a solid foundation to augment, optimize, and help stabilize existing functionality. Microsoft has performed an extensive reworking of other SQL Server features to enhance performance and application programmability. SQL Server 2005 also incorporates completely overhauled components and additions designed to meet a new generation of data management needs.

Preparing for a SQL Server 2005 upgrade involves understanding some basic principles that enable administrators to make appropriate decisions and help ensure success. As with any upgrade, the keys to success are appropriate planning and testing for the needs of the specific environment. This article explores the overall upgrade path for SQL Server 2005 components and how the SQL Server 2005 Upgrade Advisor tool
can help identify areas that require special attention. Specific upgrade considerations for certain SQL Server 2005 components—the database engine, Integration Services, Analysis Services, and Reporting Services—are also examined.

6.12.2 PROGRAMMING IN MS SQL SERVER 2005

Programming SQL Server 2005 can help you:

- Build, deploy, and manage enterprise applications that are more secure, scalable, and reliable
- Maximize IT productivity by reducing the complexity of building, deploying, and managing database applications
- Share data across multiple platforms, applications, and devices to make it easier to connect internal and external systems

Because the goal of Programming SQL Server 2005 is to introduce all facets of Programming SQL Server 2005, it's beneficial to programmers at all levels. The book can be used as a primer by developers with little experience with SQL Server, as a ramp up to the new programming models for SQL Server 2005 for more experienced programmers, or as background and primer to specific concepts.

Any IT professional who wants to learn about SQL Server 2005’s comprehensive feature set, interoperability with existing systems, and automation of routine tasks will find the answers in this authoritative guide.

6.12.3 KERBEROS WITH SQL SERVER

SQL Server 2005 (and 2000) supports Kerberos indirectly through the Windows SSPI interface when using Windows integrated authentication (as opposed to SQL authentication). However, Kerberos will only be used under certain circumstances as SQL Server allows SSPI to negotiate the authentication protocol to use; if Kerberos
cannot be used, then Windows will fall back NTLM authentication. Kerberos
authentication is far more desirable than NTLM from a security (and, to a lesser degree,
performance) point of view and I think it’s important to understand how to ensure
Kerberos is used for remote connections when possible. If you're using integrated auth,
you need to make sure that the following things are done:

1. Both the client and server machines must be part of the same Windows domain,
or else trusted domains.
2. The server's Service Principal Name (SPN) must be registered with the Active
Directory (I'll explain this in more detail below)
3. The client must be connected to the server using TCP/IP. Assuming that the
server has TCP/IP enabled, this can be accomplished by either placing TCP/IP at
the top of the client's protocol order or else prefixing the connection string with
"tcp:"

6.12.4 VISUAL BASIC.NET

Visual Basic .NET provides the easiest, most productive language and tool for
rapidly building Windows and Web applications. Visual Basic .NET comes with
enhanced visual designers, increased application performance, and a powerful integrated
development environment (IDE). It also supports creation of applications for wireless,
Internet-enabled hand-held devices. The following are the features of Visual Basic .NET
with .NET Framework 1.0 and Visual Basic .NET 2003 with .NET Framework 1.1. This
also answers why should I use Visual Basic .NET, what can I do with it?

POWERFUL WINDOWS-BASED APPLICATIONS
Visual Basic .NET comes with features such as a powerful new forms designer, an in-place menu editor, and automatic control anchoring and docking. Visual Basic .NET delivers new productivity features for building more robust applications easily and quickly. With an improved integrated development environment (IDE) and a significantly reduced startup time, Visual Basic .NET offers fast, automatic formatting of code as you type, improved Intelligence, an enhanced object browser and XML designer, and much more.

BUILDING WEB-BASED APPLICATIONS

With Visual Basic .NET we can create Web applications using the shared Web Forms Designer and the familiar "drag and drop" feature. You can double-click and write code to respond to events. Visual Basic .NET 2003 comes with an enhanced HTML Editor for working with complex Web pages. We can also use IntelliSense technology and tag completion, or choose the WYSIWYG editor for visual authoring of interactive Web applications.

SIMPLIFIED DEPLOYMENT

With Visual Basic .NET we can build applications more rapidly and deploy and maintain them with efficiency. Visual Basic .NET 2003 and .NET Framework 1.1 makes "DLL Hell" a thing of the past. Side-by-side versioning enables multiple versions of the same component to live safely on the same machine so that applications can use a specific version of a component. XCOPY-deployment and Web auto-download of Windows-based applications combine the simplicity of Web page deployment and maintenance with the power of rich, responsive Windows-based applications.

POWERFUL, FLEXIBLE, SIMPLIFIED DATA ACCESS
We can tackle any data access scenario easily with ADO.NET and ADO data access. The flexibility of ADO.NET enables data binding to any database, as well as classes, collections, and arrays, and provides true XML representation of data. Seamless access to ADO enables simple data access for connected data binding scenarios. Using ADO.NET, Visual Basic .NET can gain high-speed access to MS SQL Server, Oracle, DB2, Microsoft Access, and more.

IMPROVED CODING

A multitude of enhancements to the code editor, including enhanced Intelligence, smart listing of code for greater readability and a background compiler for real-time notification of syntax errors transforms into a Rapid Application Development (RAD) coding machine.

DIRECT ACCESS TO THE PLATFORM

Visual Basic developers can have full access to the capabilities available in .NET Framework 1.1. Developers can easily program system services including the event log, performance counters and file system. The new Windows Service project template enables to build real Microsoft Windows NT Services. Programming against Windows Services and creating new Windows Services is not available in Visual Basic .NET Standard, it requires Visual Studio 2003 Professional, or higher.

FULL OBJECT-ORIENTED CONSTRUCTS

We can create reusable, enterprise-class code using full object-oriented constructs. Language features include full implementation inheritance, encapsulation, and polymorphism. Structured exception handling provides a global error handler and eliminates spaghetti code.
**XML WEB SERVICES**

XML Web services enable you to call components running on any platform using open Internet protocols. Working with XML Web services is easier where enhancements simplify the discovery and consumption of XML Web services that are located within any firewall. XML Web services can be built as easily as you would build any class in Visual Basic 6.0. The XML Web service project template builds all underlying Web service infrastructure.

**MOBILE APPLICATIONS**

Visual Basic .NET 2003 and the .NET Framework 1.1 offer integrated support for developing mobile Web applications for more than 200 Internet-enabled mobile devices. These new features give developers a single, mobile Web interface and programming model to support a broad range of Web devices, including WML 1.1 for WAP—enabled cellular phones, compact HTML (cHTML) for i-Mode phones, and HTML for Pocket PC, handheld devices, and pagers. Please note, Pocket PC programming is not available in Visual Basic .NET Standard, it requires Visual Studio 2003 Professional, or higher.

**COM INTEROPERABILITY**

COM interoperability enables you to leverage your existing code assets and offers seamless bi-directional communication between Visual Basic 6.0 and Visual Basic .NET applications.

**REUSE EXISTING INVESTMENTS**

Windows Forms in Visual Basic .NET 2003 provide a robust container for existing ActiveX controls. In addition, full support for existing ADO code and data binding enable a smooth transition to Visual Basic .NET 2003.
UPGRADE WIZARD


6.13 SUMMARY

Softcode, using the algorithm such as PSO, GA, FA and NN, and with help of BI claim to resolve all the problem than an organization faces in its administrative as well as technical methodology in the face of severe constraints, thrown out by in the delay in decision making process. Softcode instruction into minor area like material management and its onward journey to other little area like stock, creating intent and delivery of goods to the customer, brings more perfection in all areas. So the beginning to end, business houses are ruled by its golden touch and we are relieved off constant noggin whether all things are going on smoothly. So business houses managers are rescued from mental stress and are left peace of mind and a lot of time to concentrate on quick decision making process.