MODULE 25: INTERNET & NETWORKING

On Completion of this module you will be able to use your program in conjunction with Internet, send emails and establish a network and share data accordingly.

Subject Outcome 1: Introduction

Subject Outcome 2: Internet Introduction

Subject Outcome 2.1: Call WebSite

Subject Outcome 2.2: Download resource

Subject Outcome 2.3: FTP (upload/download)

Subject Outcome 2.4: XML data file (upload)

Subject Outcome 2.5: HTML Colour Code (HEX)

Subject Outcome 3: E-Mail

Subject Outcome 4: Networking

Subject Outcome 4.1: IP Address & Computer Name

Subject Outcome 4.2: Establishing a Network

Subject Outcome 4.3: MS3 Networker

Subject Outcome 4.4: Shared Common Drive sample

Subject Outcome 4.5: Chat Server

25.1   INTRODUCTION

A network is an integrated system between modules (computers and other devices) either wirelesses (much easier) or wired (limited and expensive hardware) so that they are able to communicate and share data. Internet must not be compared to a network. A network is whereby you connect computers with each other; internet is whereby you connect a computer to the World Wide Web system. You may connect computers to each other without having internet.

We have touched network sharing during module 21 (databases) whereby we had to establish a network to share a data file. Computer programs may communicate with each other either in the background (no set shared path or shared directory such as a T:\ drive) or via a common drive such as a shared path/directory. With Windows XP ® & 7 ® it was very simply as you only had to create an AD HOC network (wireless) , however with Windows 8 and 10 ® it is more complicated as you need a common internet connection to share computers. The important aspect to note regarding a network system is that one computer must be the identified server computer that establishes the network, the rest only joins the network.

25.2   INTERNET

The internet is the gateway to the World Wide Web that contains URL destinations. These URL destinations are actually websites and the medium of communication is via emails or FTP (upload/download files). Before we start we need to connect to internet. This is done bmo a cellular modem/router with data loaded onto it. You may also use a cellular phone, connected with an USB cable to a computer with **tethering activated** (not mobile hotspot, unless you wish to share your internet).

I will not teach you CGI coding as FREEQ can only master basic HTML coding and not CSS that is required by any browser. I will however teach you the basics required for and by Internet that will work with and required by your programs.

25.2.1 Calling a WebSite URL:

SNO	CODING	EXPLANATION
| 1 | ```
$TYPECHECK ON
$INCLUDE <RapidQ2.inc>
$include <qrun.inc>
dim runme as qrun
declare sub callwebber
CREATE Form AS QFORM
    Caption = "CALL WEBSITE"
    Width = 300:Height = 200:Center
    create callweb as qbutton
        left=10:top=10:caption="WWW"
        onclick=callwebber
    end create
END CREATE
SetWindowLong(Form.Handle, -8, 0)
SetWindowLong(Application.Handle, -8, Form.Handle)
Form.ShowModal
 sub callwebber
 runme.filerun("https://www.ms3system.co.za/",1)
 end sub
``` | Associate with default browser/program.<br><br><br><br><br><br><br><br><br><br><br><br>Call using default browser. |

25.2.2 Download a resource from a WebSite (an image, etc.)

You are able to download a resource directory from a Website as long as you have the URL address of that resource (image – right mouse click on the resource and select COPY ADDRESS – you will then use this address to download it – it can be done via QEDIT box or automatic for whatever purpose; updatement of program resources, copy verification, etc.)

SNO	CODING	EXP
| 1 | ```
$apptype gui
$typecheck on
$include "rapidq2.inc"
Declare Function URLDownloadToFile Lib "urlmon" Alias "URLDownloadToFileA" (ByVal pCaller As Long, ByVal
szURL As String, ByVal szFileName As String, ByVal dwReserved As Long, ByVal lpfnCB As Long) As Long
declare Function DownloadFile(url As String, LocalFilename As String) As integer
function DownloadFile(url As String, LocalFilename As String) As integer
    Dim lngRetVal As Long
    lngRetVal = URLDownloadToFile(0, url, LocalFilename, 0, 0)
    If lngRetVal = 0 Then result = 1
End Function
declare sub dlsub
create form as qform
    width=500:center
    create urledit as qedit
        top=0:width=400
        text="https://www.ms3system.co.za/gallery_gen/037507f8c3cfd1370cf40cfb05faded9_90x90.png"
    end create
    create path as qedit
        top=20:width=400:text="c:\ms3\~common\testd.png"
    end create
    create dlbtn as qbutton
        top=40:caption="Download":onclick=dlsub
    end create
end create
form.showmodal
Sub dlsub
if DownloadFile (urledit.text, path.text)=1 then
    showmessage "Download done"
    else
    showmessage "Error"
end if
End Sub
``` | Added functions to ensure the element works.<br><br><br><br><br><br><br><br>File to download<br><br>Download as ...<br><br><br><br>Download and if all in order or error. |
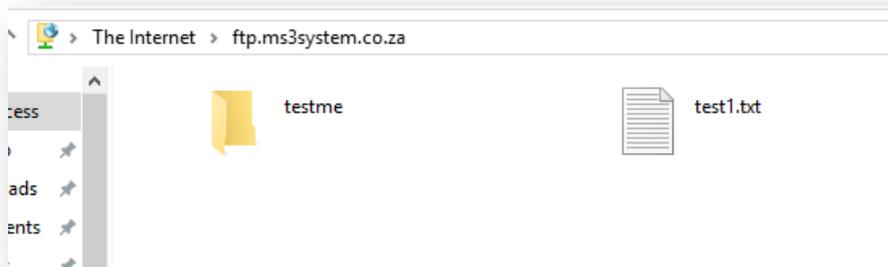
25.2.3   FTP Upload & Download

FTP Upload and download is extremely important for a program to be able to upload information/data/reports to a website and download again. It is used to display information on the website or to obtain information from the website. Say you have an athletics program and wish to upload the results of the athletes as they complete the races, for their viewing, this is the best and fasted method of getting the results to your clients. You however need the following to have this capability:

- A WebSite domain (website owner).
- Administrator password.
- FTP location (path).

Remember you may access the FTP account via Microsoft Explorer (ftp...) or via your CPanel WebSite administrator. Obviously FreeQ also now gives you access (read and write) to your FTP account.

SNO	CODING	EXP
1	```$TYPECHECK ON
$INCLUDE <RapidQ2.inc>
$escapechars on
$INCLUDE <QFTPCLIENT.INC>
Dim myFtp As QFtpClient

declare sub uploadfile
declare sub getfilenow
dim milda as string

CREATE Form AS QFORM
    Caption = "FTP - click form":Width = 640:Height = 480:Center
    create but as qbutton
      left=10:top=10:height=18:caption="Add files"
      onclick=uploadfile
    end create
    create but1 as qbutton
      left=200:top=10:height=18:caption="Get file"
      onclick=getfilenow
    end create``` | Add the element.

Establish the element.

Button to add the files to the FTP drive.
Button to retrieve the files from the FTP drive. |

SNO	CODING	EXP

	create info as qlistbox
	   left=10:top=40:width=300:height=300
	  end create
	END CREATE
	SetWindowLong(Form.Handle, -8, 0)
	SetWindowLong(Application.Handle, -8, Form.Handle)
	Form.ShowModal
	
	sub uploadfile
	form.caption="Awaiting response from FTP ..."
	  myFtp.Connect ("ftp.ms3system.co.za", "xmltools@tools.ms3system.co.za", "p@T}.,9SR6*0")
	    info.additems ftp.ErrorMsg
	  milda=myftp.getdir
	  myftp.getdir
	    info.additems "Set Dir: "+milda
	  myFtp.PutFile ("test.txt", "test1.txt")
	    info.additems ftp.errormsg
	  myftp.makedir ("testme")
	    info.additems ftp.errormsg
	  myftp.setdir ("testme")
	  myftp.getdir
	    info.additems "Set Dir: "+myftp.getdir
	    myFtp.PutFile ("test.txt", "test2.txt")
	myFtp.Close
	form.caption="FTP Actions done"
	end sub
	
	sub getfilenow
	form.caption="Awaiting response from FTP ..."
	  myFtp.Connect ("ftp.ms3system.co.za", "xmltools@tools.ms3system.co.za", "p@T}.,9SR6*0")
	    info.additems ftp.ErrorMsg
	  milda=myftp.getdir
	  myftp.getdir
	myFtp.GetFile ("test1.txt", "c:\ms3\~common\gotit.txt", True)
	myftp.close
	form.caption="Retrieved..."
	end sub
		Connect: FTP URL, user, password.
		Current Drive
		Activate dir
		
		Put file as **test1**
		
		Make dir
		Alter directory
		Activate dir
		
		Put file on new directory.
		CLOSE
		
		Connect to FTP.
		
		Set cur dir.
		
		Get file and save as **gotit.txt**
		CLOSE

2	$TYPECHECK ON
	$INCLUDE <RapidQ2.inc>
	$escapechars on
	$INCLUDE <QFTPCLIENT.INC>
	Dim myFtp As QFtpClient
	
	declare sub uploadfile
	declare sub getfilenow
	dim milda as string
	CREATE Form AS QFORM
	  Caption = "FTP - click form":Width = 640:Height = 480:Center
	  create but as qbutton
	    left=10:top=10:height=18:caption="Add files"
	    onclick=uploadfile
	  end create
	  create but1 as qbutton
	    left=200:top=10:height=18:caption="remove dir"
	    onclick=getfilenow
	  end create
	  create info as qlistbox
	    left=10:top=40:width=300:height=300
	  end create
	END CREATE
	SetWindowLong(Form.Handle, -8, 0)
	SetWindowLong(Application.Handle, -8, Form.Handle)
	Form.ShowModal
	
	sub uploadfile
	form.caption="Awaiting response from FTP ..."
	  myFtp.Connect ("ftp.ms3system.co.za", "xmltools@tools.ms3system.co.za", "p@T}.,9SR6*0")
	    info.additems ftp.ErrorMsg
	  milda=myftp.getdir
		Other functions include ==removing== a directory and ==deleting== a file (==REMEMBER== the directory must be empty to be deleted).
		
		Take note that you may also ==RENAME(oldname$,newname$)==

SNO	CODING	EXP
| | ```
myftp.getdir
    info.additems "Set Dir: "+milda
myFtp.PutFile ("test.txt", "test1.txt")
    info.additems ftp.errormsg
myftp.makedir ("testme")
    info.additems ftp.errormsg
myftp.setdir ("testme")
myftp.getdir
    info.additems "Set Dir: "+myftp.getdir
    myFtp.PutFile ("test.txt", "test2.txt")
myFtp.Close
form.caption="FTP Actions done"
end sub

sub getfilenow
form.caption="Awaiting response from FTP ..."
    myFtp.Connect ("ftp.ms3system.co.za", "xmltools@tools.ms3system.co.za", "p@T}.,9SR6*0")
        info.additems ftp.ErrorMsg
    milda=myftp.getdir
    myftp.getdir
    myftp.delete("test1.txt")
    myftp.removedir ("testme")
myftp.close
form.caption="removed ..."
end sub
``` | Delete file.<br>Delete directory. |

The following will list the FTP file list within the directory:

SNO	CODING	EXP
| 1 | ```
$TYPECHECK ON
$include "rapidq2.inc"
$INCLUDE "QFTPCLIENT.INC"
dim blank as qstringlist
Dim myFtp As QFtpClient
With myFtp
  DefStr sFile
  .Connect ("ftp.ms3system.co.za", "xmltools@tools.ms3system.co.za", "p@T}.,9SR6*0")
      sFile = .FindFirstFile ("*.*")
      While sFile <> ""
        If Instr(.fileAttributesAsString, "") = 0 Then
          blank.additems "———————————"
          blank.additems .filename
          blank.additems str$(.fileSize)
          blank.additems .filedate
          blank.additems .filetime
          blank.additems str$(.fileAttributes)
          blank.additems .fileAttributesAsString
          print:Print "FileName : ";.fileName
          Print "  File Size : ";.fileSize
          Print "  File Date : ";.fileDate
          Print "  File Time : ";.fileTime
          Print "  File Attributes: ";.fileAttributes
          Print "  File Attributes As String : ";.fileAttributesAsString
        End If
        sFile = .FindNextFile
      Wend
      Print "End Of Directory"
  .Close
End With
blank.savetofile "c:\ms3\~common\ftplist.txt"
run "notepad.exe c:\ms3\~common\ftplist.txt"
showmessage "ok"
``` | Use this program as is; load the data back into a STRINGGRID etc. as if you create a FORM, it just does not work due to internal classes. |

The following will list the FTP file list within the directory:

SNO	CODING TO NAVIGATE WITHIN FTP	REMARK
1	Myftp.setdir("..")	Return one level up from previous selected directory.
2	Myftp.setdir("/)	Refresh from original set directory.
3	newname=replacesubstr$(myftp.filename,":","-") newname=replacesubstr$(newname,"/","-") newname=replacesubstr$(newname,"\","-") myftp.rename (myftp.filename,newname)	If you wish to download an e-mail file from the FTP folder you will require the following replacements to download the file without illegal characters, otherwise the file will not even been seen.

4	myftp.Connect (ftpx, ftpuserx, passx)
	myftp.getdir
	myftp.setdir ("**mail**")
	myftp.getdir
	newwoord=replacesubstr$(ftpx,"ftp.","")
	myftp.setdir (newwoord)
	myftp.getdir
	myftp.setdir (mainuserx)
	myftp.getdir
	myftp.setdir ("cur")
	myftp.getdir
	
	sFile = myftp.FindFirstFile ("*.*")
	While sFile <> ""
	  If Instr(myftp.fileAttributesAsString, "") = 0 Then
	    newname=replacesubstr$(myftp.filename,":","-")
	    newname=replacesubstr$(newname,"/","-")
	    newname=replacesubstr$(newname,"\","-")
	    myftp.rename (myftp.filename,newname)
	    andre.additems myftp.filename
	    andre.addsubitem (i,myftp.filedate)
	    andre.addsubitem (i,myftp.filetime)
	    andre.addsubitem (i,myftp.fileAttributesAsString)
	    myFtp.GetFile (newname, "C:\ms3\ms3fitman\datalp\fitmailer\"+newname, True)
	    aniska.clear
	    aniska.loadfromfile "C:\ms3\ms3fitman\datalp\fitmailer\"+newname
	  End If
	  i=i+1
	  sFile = myftp.FindNextFile
	Wend
		FTPX being
		ftp.ms3system.co.za
		
		FTPUSERX is the **FTP server name (main)**
		
		PASSX is the **password.**
		
		E-Mail downloads. You will be required to know the location of the FTP MAIL (CUR and NEW) files. See CPANEL EXPLORER.
		
		ANDRE is a QLISTVIEW
		
		You need to download an email and then into a RICHEDITOR to view the content. Soft and Filter the content neatly.
		
		Remember there is a CUR and NEW directory for every EMAIL folder.

25.2.4   XML Data file

A XML data file is an Internet WebSite spreadsheet (table) that may be uploaded/downloaded using FTP upload/download. To create a XML data file you need to code the following format (please to use MS3XML creator for free available from the MS3 WebSite). You will basically create a QSTRINGGRID and use the following code to convert the file to XML format (**WOORD** is a **string** ;**BLANK** is a **QSTRINGLIST** and **MPARENT1** is a QEDIT that is the actual **heading** and **MCHILD1** represents the **CHILD HEADING** or **INDEX HEADING**) for instance **TOYOTA** is the **heading** and **HR** is the **child heading/department**.

```
sub populatexml
woord=""
blank.clear
woord="<?xml version="+chr$(34)+"1.0"+chr$(34)+" encoding="+chr$(34)+"iso-8859-1"+chr$(34)+"?>"
blank.additems woord
blank.additems "<"+mparent1.text+">"
x=1
for i = 1 to stringer.rowcount-1
   blank.additems chr$(9)+"<"+mchild1.text+" id="+chr$(34)+str$(i)+chr$(34)+">"
   for p = 1 to stringer.colcount-1
      if stringer.cell(p,0)<>"" then
         blank.additems chr$(9)+chr$(9)+"<"+stringer.cell(p,0)+">"+stringer.cell(p,i)+"</"+stringer.cell(p,0)+">"
      end if
   next p
   blank.additems chr$(9)+"</"+mchild1.text+">"
next i
blank.additems "</"+mparent1.text+">"
nuwenaam=replacesubstr$(menu0.caption,".xet",".xml")
blank.savetofile (kykhier+"\xml-files\"+nuwenaam)
run "notepad.exe "+kykhier+"\xml-files\"+nuwenaam
end sub
```

The actual XML file layout.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<MENU>
        <Main id="1">
                <FOOD>Steak</FOOD>
                <PRICE>99.00</PRICE>
        </Main>
        <Main id="2">
                <FOOD>Fish</FOOD>
                <PRICE>49.99</PRICE>
        </Main>
        <Main id="3">
                <FOOD>Chips</FOOD>
                <PRICE>20.25</PRICE>
        </Main>
        <Main id="4">
                <FOOD>Vegies</FOOD>
                <PRICE>12.87</PRICE>
        </Main>
</MENU>
```

25.2.5   WebSite HTML HEX colour code

You are not supposed to use RGB or the standard colour codes with HTML coding (CGI). Use this code to convert the colour code to HEX HTML Colour coding. This is applicable for any upload of colour coding.

SNO	CODING	EXPLANATION

1	`$TYPECHECK ON`
	`$include "rapidq2.inc"`
	`declare sub getcol`
	`dim Dial as QColorDialog`
	`   Dial.Style=cdNoFullOpen`
	`   Dial.left = 12`
	`   Dial.Top = 50`
	`create form as qform`
	`   onclick=getcol`
	`end create`
	`form.showmodal`
	
	`sub getcol`
	`   dim a$ as string`
	`   if Dial.Execute then`
	`     Form.Color=Dial.Color`
	`   end if`
	`   a$= hex$(dial.color)`
	`   dim rgb$ as string`
	`   RGB$="#"+mid$(a$,7,2)+mid$(a$,5,2)+mid$(a$,3,2)`
	`   showmessage rgb$`
	`end sub`
		The colour wheel.
		
		Convert HEX for HTML colouring code.

25.3   E-MAIL

The perfect database program is a program capable of sending an e-mail directly from itself, without the client having to open a browser, open the email client, compile an e-mail and having to search for the attachments – TIME consuming and TIME wasted.

I have created a full desktop e-mailing system (**ensure to add the MS3MFUL.EXE** application to your distribution of your software – find it within the **C:\SOFTWARE\FREEQ** directory - that may work as an independent program or load the information fields from your FreeQ program into the emailing program.

The information required by MS3MFUL to send an email directly from your program:

- Receiver E-Mail address.
- Sender E-Mail address.
- Sender Password.
- Out Port (port number).
- SMTP Host Server (gmail/etc.)
- SSL (secure line on/off).
- Subject (heading).

- Message (multi-line).
- Attachment (if required).

SNO	CODING	EXP

1	`$TYPECHECK ON`
	`$INCLUDE <RapidQ2.inc>`
	`dim i as single:dim p as single`
	`dim blank as qstringlist`
	`dim font as qfont:font.name="arial rounded"`
	`font.addstyles(fsbold):font.color=16777215`
	`dim openxlm as qopendialog:openxlm.caption="Attach Document ..."`
	`declare sub callattachment`
	`declare sub sendmailnow`
	`if direxists("c:\ms3")=0 then mkdir "c:\ms3"`
	`if direxists("c:\ms3\~common")=0 then mkdir "c:\ms3\~common"`
	`CREATE Form AS QFORM`
	`    Caption = "MAILER":color=16744576:borderstyle=1`
	`    width=300:height=560:left=screen.width-305`
	`    top=screen.height-605:FormStyle=fsStayOnTop`
	`    create lab1 as qlabel`
	`        left=10:top=8:caption="SEND TO:":font=font`
	`    end create`
	`    create lab2 as qedit`
	`        left=10:top=25:width=260:taborder=1`
	`        text="andre@ms3system.co.za"`
	`    end create`
	`    create lab3 as qlabel`
	`        left=10:top=55:caption="SUBJECT:":font=font`
	`    end create`
	`    create lab4 as qedit`
	`        left=10:top=72:width=260:taborder=2`
	`    end create`
	`    create lab5 as qlabel`
	`        left=10:top=102:caption="MESSAGE:":font=font`
	`    end create`
	`    create lab6 as qrichedit`
	`        left=10:top=119:width=260:height=100:taborder=3`
	`    end create`
	`    create lab6b as qlistbox`
	`        visible=0`
	`    end create`
	`    create lab7 as qlabel`
	`        left=10:top=228:caption="ATTACH FILE HERE":font=font`
	`        showhint=1:hint="click here to attach a file ..."`
	`        cursor=-21:onclick=callattachment`
	`    end create`
	`    create lab8 as qlabel`
	`        left=10:top=247:caption="(file)"`
	`    end create`
	`    create btn1 as qbutton`
	`        left=140:top=290:width=150:caption="SEND":cursor=-21:taborder=4`
	`        onclick=sendmailnow`
	`    end create`
	`    create btnpanel as qpanel`
	`        left=5:top=340:width=285:height=155:bevelinner=1:bevelouter=1:color=16761281`
	`        create lab10 as qlabel`
	`            left=10:top=5:caption="SMTP Host Server:"`
	`        end create`
	`        create lab11 as qedit`
	`            left=10:top=20:text="smtp.gmail.com"`
	`        end create`
	`        create lab12 as qlabel`
	`            left=150:top=5:caption="Out Port"`
	`        end create`
	`        create lab13 as qedit`
	`            left=150:top=20:text="587"`
	`        end create`
		Sending E-Mails (via **MS3Mailer ©**) Enter your details as sender. Then enter a receiver's detail and send the e-mail (remember internet connection).
		
		You must have the following to use this email system: C:\ms3\~common The same config (edit fields, etc.)
		
		MS3MFUL.EXE within the same directory.
		
		My advice would be use the creation code and executional code as is within your program.
		
		Subject
		
		Message
		
		Attachment file.
		
		SEND button.
		
		CONFIGURATION (where you code your email particulars – set in such a manner that it would save as you alter it for future use).

SNO	CODING	EXP

	create lab14 as qcheckbox
	    left=10:top=50:caption="SSL":checked=1
	  end create
	  create lab15 as qlabel
	    left=10:top=70:caption="User:"
	  end create
	  create lab16 as qedit
	    left=10:top=85:width=250
	  end create
	  create lab17 as qlabel
	    left=10:top=110:caption="Password:"
	  end create
	  create lab18 as qedit
	    left=10:top=125:width=250:passwordchar="X"
	  end create
	  end create
	END CREATE
	SetWindowLong(Form.Handle, -8, 0)
	SetWindowLong(Application.Handle, -8, Form.Handle)
	Form.ShowModal
	
	sub callattachment
	if openxlm.execute then
	   lab8.caption=openxlm.filename
	end if
	end sub
	
	sub sendmailnow
	if fileexists(application.path+"\ms3mful.exe")=0 then
	   showmessage "You need MS3MFUL.EXE to send ..."
	end if
	if fileexists("c:\ms3\~common\ms3mli.fet")=1 then
	   kill ("c:\ms3\~common\ms3mli.fet")
	end if
	if lab18.text="" then
	   if MESSAGEDLG("CONFIGURE your MAIL?", 3, mbok, 0)=mrok then
	     run (application.path+"\ms3mful.exe")
	     goto skipsending
	   end if
	end if
	blank.clear
	blank.additems lab11.text
	blank.additems "pop.gmail.com"
	blank.additems lab13.text
	blank.additems "995"
	blank.additems lab16.text
	blank.additems lab18.text
	blank.additems lab2.text
	blank.additems lab4.text
	blank.additems lab8.caption
	if lab14.checked=0 then blank.additems "False"
	if lab14.checked=1 then blank.additems "True"
	blank.savetofile ("c:\ms3\~common\ms3mli.fet")
	blank.savetofile ("c:\ms3\~common\ms3mlr.fet")
	blank.clear
	for i = 0 to lab6.linecount-1
	   blank.additems lab6.line(i)
	next i
	blank.savetofile ("c:\ms3\~common\ms3mlt.fet")
	p=0
	run (application.path+"\ms3mful.exe")
	lab2.text=""
	skipsending:
	end sub
		ID attachment file (all files) – no restrictions on file type or size as with other browsers.
		
		Check if MS3MFUL is installed.
		
		This is the file that MS3MFUL requires to compile the email and send.
		If no config is done, open the MS3MFUL program for configuration purposes (more options)
		
		Add all the file information as required to send the email.
		
		Store the data to be called by MS3MFUL.
		
		Run the MS3MFUL and send the email.
		Clear receiver to indicated that it was sent.

2	TO LOAD EMAILS – use a FTP Link within your program to download files and then convert for use within a STRINGGRID – You will require a CUSTOM EMAIL with CPANEL.	It must be fixed (WEB FORM) format.

25.4   NETWORKING

We have briefly touched networking with a computer system. Remember that a network is whereby you connect multiple (two or more) computers to a server computer. The server computer acts as the medium for all computers to connect with each other. The shared data is stored on the server. Printers may be shared even though you may print to printers on the sub-station.

Shared data may be shared as follows:

- Using a common drive (T:\ ) and the file is an actual file stored on the common drive –fast processing with large structured (tables) or unstructured files.
- An Internet network drive (XML file / FTP file) – slow processing.
- Network protocol files (not visible as with an actual file on a specific drive) – used for communication like a CHAT program or Index Key databases (SQL, Access based, QDATABASE files) – extremely fast processing however limited to the table columns/rows.

You will probably be requiring the following information regarding your computer to establish a network or connect to a network:

25.4.1   IP Address & Computer Name

```
$TYPECHECK ON
DIM Sock as QSocket
DIM SockNum AS INTEGER
SockNum = 0

Showmessage Sock.GetHostIP                          'your IP address
Showmessage Sock.GetHostName                        'your computer's name
sock.Open(80)
SockNum = sock.Connect(Sock.GetHostName, 80)
showmessage str$(socknum)                           'socket number
```

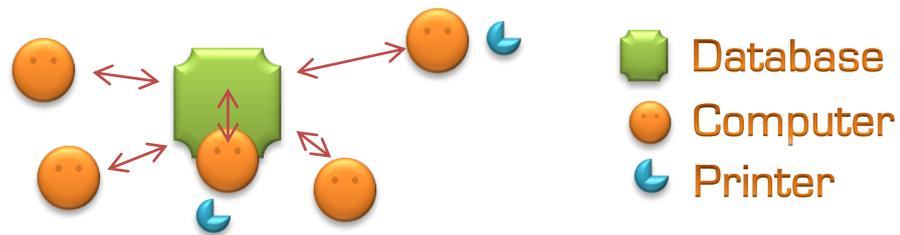### 25.4.2   ESTABLISHING A NETWORK

The first step is to establish a network. You need to decide either wired or wireless. Wired obviously is faster and much more stable, however you are limited by a router or only two computers (direct link) – the setup is also a bit complicated as it is not automated (Windows Xp & 7 manual ad hoc) – Windows 8/10 may only create a network if an internet router/modem/hot-spot is available. See MS3 Create a Network yourself.

A wireless network is rather simple (**Windows XP** – manual setup required, **Windows 7** – use MS3NETWORKER – if it works then you don't have to establish a network, you simply

need to connect, **Window 8/10** establish **an internet router/modem**) and you may connect a few computers easily – speed is effected slightly, however you really don't need anything more than this – printer sharing is also made easy using this method. Just remember that one computer must be the server (with the main shared database file(s)) – however the server has the database, BUT also acts as a sub-computer itself – THE DATABASE is the server, not the actual computer.

The structural layout should be seen as follows:

25.4.3   MS3Networker

The actual coding for establishing a wireless network (remember you must run this program as ADMINISTRATOR and the password must consist out of 5 or 13 characters):

SNO	CODING	EXP
| 1 | ```
$TYPECHECK ON
$include "rapidq2.inc"
$Include "QVolumeInfo.inc"
$include "qshfileoperation.inc"
$INCLUDE "XP_Manifest.inc":$XP_MANIFEST
$APPTYPE gui
declare sub updatenetworkadd
Declare Function WinExec Lib "kernel32" Alias "WinExec" (ByVal lpCmdLine As String, ByVal nCmdShow As Long) As Long
Dim sapi as QOLEObject
    sapi.CreateObject("SAPI.SpVoice")
FUNCTION desktopdir(dir AS STRING) AS STRING
    DIM dirsys AS QREGISTRY
    DIM v$ AS STRING
    dir = "desktop"
    dirsys.rootkey = &H80000001
    dirsys.openkey("Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders" , 0)
    result = dirsys.readstring(dir) + "\"
END FUNCTION
dim userhier as string
dim dir as string
userhier = desktopdir(dir)
IF DIREXISTS(desktopdir(dir) + "MS3") = 0 THEN MKDIR desktopdir(dir) + "MS3"
dim blank as qstringlist
dim milda as string
declare sub saveedit1
declare sub startnetwork
declare sub stopnetwork
declare sub visitms3
dim font as qfont:font.name="arial rounded":font.addstyles(fsbold):font.size=9
``` | |

SNO	CODING	EXP

	`$resource back_1 as "c:\ms3\~gx\back5b.bmp"`
	 Dim s as QSocket
	 DIM SockNum AS INTEGER:SockNum = 0
	 dim ip$ as string
	 dim name$ as string
	 CREATE Form1 AS QFORMex
	    Width = 500:Height = 270:AutoScroll = False
	    BorderStyle = 4
	    Caption = "MS³ Networker © ... MS³ (Pty)"
	    Parent = Form1:Center
	    create image as qimage
	      left=0:top=-50:autosize=1:bmphandle=back_1
	    end create
	    create info as qlabel
	      transparent=1
	      left=10:top=160:caption="Set as Administrator Rights."+chr$(13)+"Create desktop shortcut of EXE (ensure administrator rights)"+chr$(13)+"Run the desktop short-cut"+chr$(13)+"This will create a virtual network (with Win XP, Vista && 7)"+chr$(13)+"The Network is accessible by all Win O/S formats (incl Win 8 && 10) and Andriod devices."
	    end create
	    create label0 as qlabel
	      left=5:top=5:transparent=true:font=font:caption="(computer's detail ...)"
	    end create
	    CREATE Label1 AS QLABEL
	      Left = 5:Top = 30:Width = 103:Height = 23:Caption = "NETWORK NAME:":Transparent = True
	      font=font
	    END CREATE
	    CREATE Label2 AS QLABEL
	      Left = 5:Top = 67:Caption = "PASSWORD:":Transparent = True:font=font
	    END CREATE
	    CREATE Edit1 AS QEDIT
	      Left = 115:Top = 28:Width = 300:showhint=1:hint="enter network name ... avoid using characters ..."
	      onkeyup=saveedit1
	    END CREATE
	    CREATE Edit2 AS QEDIT
	      Left = 115:Top = 65:Width = 100:passwordchar="X"
	      showhint=1:hint="enter password, must be either 5 or 13 characters ..."
	      onkeyup=saveedit1
	    END CREATE
	    CREATE Butn1 AS QBUTTON
	      Left = 15:Top = 110:Width = 109:Caption = "START NETWORK":font=font:height=25
	      onclick=startnetwork
	    END CREATE
	    CREATE Butn2 AS QBUTTON
	      Left = 150:Top = 110:Width = 90:Caption = "Stop Network":font=font:height=20
	      onclick=stopnetwork
	    END CREATE
	    CREATE Butn3 AS QBUTTON
	      Left = 405:Top = 2:Caption = "Visit MS³":font=font:height=20
	      onclick=visitms3
	    END CREATE
	    FormStyle=fsStayontop
	 END CREATE
	 if direxists("c:\ms3")=0 then mkdir "c:\ms3"
	 if direxists("c:\ms3\~common")=0 then mkdir "c:\ms3\~common"
	 SetWindowLong(Form1.Handle, GWL_HWNDPARENT, 0)
	 SetWindowLong(Application.Handle, GWL_HWNDPARENT, Form1.Handle)
	 xpbutton(butn1.handle)
	 xpbutton(butn2.handle)
	 xpbutton(butn3.handle)
	 Ip$ = s.GetHostIP
	 Name$ = s.GetHostName
	 s.Open(80)
	 SockNum = S.Connect(name$, 80)
	 label0.caption="NAME : "+ Name$ + " ... IP : "+ip$
	 blank.clear
	 if fileexists("c:\ms3\~common\ntwr.fet")=1 then
	    blank.loadfromfile ("c:\ms3\~common\ntwr.fet")
	    edit1.text=blank.item(0)
	    edit2.text=blank.item(1)	

SNO	CODING
	end if
	Form1.ShowModal
	sub saveedit1
	blank.clear
	blank.additems edit1.text
	blank.additems edit2.text
	blank.savetofile ["c:\ms3\~common\ntwr.fet"]
	end sub
	sub visitms3
	WinExec "C:\Program Files\Google\Chrome\Application\chrome.exe http://ms3system.co.za/Home.php",10
	end sub
	sub startnetwork
	if len(edit2.text)=5 then goto startnetwork2
	if len(edit2.text)=13 then goto startnetwork2
	sapi.speak["Check Password.",1)
	IF MESSAGEDLG("Password must be either 5 or 13 characters long only!!!" , 1 , mbok , 0) = mrok THEN goto startnetworkend
	startnetwork2:
	milda="netsh wlan set hostednetwork mode=allow ssid="+edit1.text+" key="+edit2.text
	winexec milda,10
	milda="netsh wlan start hostednetwork"
	winexec milda,10
	sapi.speak("The network has been established successfully.",1)
	updatenetworkadd
	startnetworkend:
	end sub
	sub stopnetwork
	milda="netsh wlan stop hostednetwork"
	winexec milda,10
	sapi.speak("The network has been stopped.",1)
	updatenetworkadd
	end sub
	sub updatenetworkadd
	label0.caption="updating ..."
	sleep 2
	Ip$ = s.GetHostIP
	Name$ = s.GetHostName
	s.Open(80)
	SockNum = S.Connect(name$, 80)
	label0.caption="NAME : "+ Name$ + " ... IP : "+ip$
	end sub

Now establish the network. Now you must decide will you use a Common drive (T:\) **or direct communication** (via a Chat Client, etc.) This will depend on the data that you will share and the method used (QStringGrid / QDatabase / Access based, etc.)

Remember that if you use a common drive, **you need to log onto it via MS Explorer** whereby the direct communication method don't require a log on.

25.4.4   Shared Common Drive sample program

We will now code a simple example of how two programs share a file and use it. Let's create a file whereby two computers will share mouse control – a small game of move around the mouse on a screen. Compile program 1 (server computer) then program 2 (sub/slave computer) – you need to establish a network with a common drive (t:\). Once

you run the programs, click the form and select where to store the PLAY file (common drive).

SNO	CODING	EXP

1	`$TYPECHECK ON`
	`$INCLUDE <RapidQ2.inc>`
	`dim kykhier as string`
	`declare sub checkmousepos`
	`declare sub stopthis`
	`declare sub storeposition`
	
	`DIM fDialog as qfiledialog`
	`fDialog.Caption = "Find location and click SAVE to store PLAY1"`
	`fDialog.FileName= "play1.fet":fdialog.mode=1`
	
	`dim timer1 as qtimer:timer1.enabled=0:timer1.interval=20:timer1.ontimer=checkmousepos`
	`dim blank as qstringlist`
	
	`CREATE Form AS QFORMex`
	`    Caption = "Prog 1 - click form to start": Width = 500:Height = 100:center`
	`    onclose=stopthis:onclick=storeposition`
	`    create lab1 as qlabel`
	`       left=10:top=10`
	`    end create`
	`END CREATE`
	`SetWindowLong(Form.Handle, -8, 0)`
	`SetWindowLong(Application.Handle, -8, Form.Handle)`
	`Form.ShowModal`
	
	`sub stopthis`
	`kill kykhier+"\play1.fet"`
	`end sub`
	
	`sub storeposition`
	`IF fDialog.Execute THEN`
	`    kykhier=replacesubstr$(fdialog.filename,"\play1.fet",""): lab1.caption=kykhier+"\play1.fet"`
	`    timer1.enabled=1`
	`end if`
	`end sub`
	
	`sub checkmousepos`
	`blank.clear:blank.additems str$(screen.mousex):blank.additems str$(screen.mousey)`
	`blank.savetofile kykhier+"\play1.fet"`
	`end sub`
		==PROGRAM 1 (run on server computer)==

| 2 | ```
$TYPECHECK ON
$INCLUDE <RapidQ2.inc>

dim kykhier as string
declare sub checkmousepos
declare sub stopthis
declare sub storeposition

DIM fDialog as qfiledialog
fDialog.Caption = "Find location and click SAVE to store PLAY2"
fDialog.FileName= "play2.fet":fdialog.mode=1

dim timer1 as qtimer:timer1.enabled=0
timer1.interval=800:timer1.ontimer=checkmousepos
dim blank as qstringlist
create formp as qform
    borderstyle=0:width=20:height=20:color=255
    FormStyle = fsStayontop
end create
CREATE Form AS QFORMex
    Caption = "Prog 2 - click form to start"
    Width = 500:Height = 100:center
    onclose=stopthis:onclick=storeposition
    create lab1 as qlabel
        left=10:top=10
    end create
END CREATE
SetWindowLong(Form.Handle, -8, 0)
SetWindowLong(Application.Handle, -8, Form.Handle)
formp.visible=1
Form.ShowModal

sub stopthis
kill kykhier+"\play2.fet"
end sub

sub storeposition
IF fDialog.Execute THEN
    kykhier=replacesubstr$(fdialog.filename,"\play2.fet","")
    lab1.caption=kykhier+"\play2.fet"
    timer1.enabled=1
end if
end sub

sub checkmousepos
blank.clear
blank.additems str$(screen.mousex)
blank.additems str$(screen.mousey)
blank.savetofile kykhier+"\play2.fet"
if fileexists(kykhier+"\play1.fet")=1 then
    blank.loadfromfile (kykhier+"\play1.fet")
    formp.left=val(blank.item(0))
    formp.top=val(blank.item(1))
end if
end sub
``` | **PROGRAM 2 – run on sub computer.** |
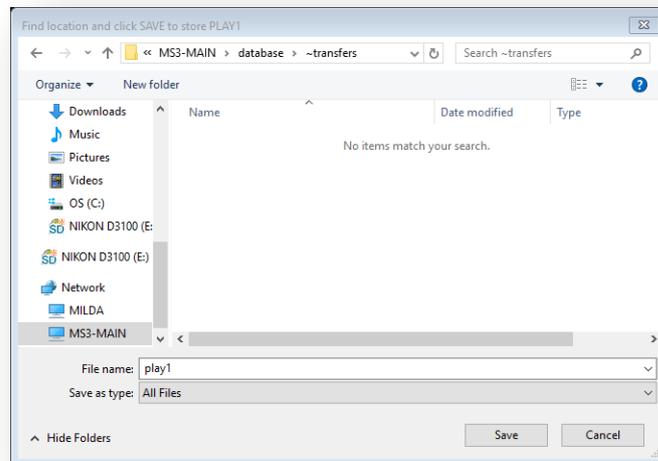
25.4.5   CHAT SERVER/CLIENT

Chat is a program capable of instantly (99.9999% instant) transferring a message to another computer via a network (no-common shared drive required – only need to establish a network and **same port number – 0 to 5000**) The **port number is like a channel**.

You will again require two computers that have been established via a network. You will

require a server computer (**server program** – log on as **LOCALHOST**) and unlimited slave computers – remember the server computer must also use the **client program** to log on. Whenever a computer logs onto the server, it is given the **next user index number**.

SNO	CODING	EXP
| 1 | ```
$APPTYPE GUI
$TYPECHECK ON
CONST DELAY = 10
CONST PortNum = 5000
CONST False = 0
CONST True = NOT False
DECLARE SUB TimerExpired
DIM Socket AS QSocket

CREATE Form AS QForm
  Height = 310:borderstyle=4:width = 500
  Caption = "Chat Server"
  Center
  CREATE ListBox AS QListBox
    Top = 80:Height = 200
    Width = Form.ClientWidth
    AddItems "Chat server", STRING$(30,"-")
    ItemIndex = 0
  END CREATE
  CREATE Label1 AS QLabel
    Top = 10
    Left = 10
    Height = 14
    Caption = "Server: " + Socket.GetHostName
  END CREATE
  CREATE Label2 AS QLabel
    Top = 29
    Left = 10
    Height = 14
    Caption = "Port: "+STR$(PortNum)
  END CREATE
  CREATE Label3 AS QLabel
    Top = 48
    Left = 10
    Height = 14
    Caption = "Connections: 0"
  END CREATE
END CREATE

DIM Timer1 AS QTimer
DIM MasterSocket AS INTEGER, NumClients AS INTEGER
DIM Client(1 TO 50) AS INTEGER
DIM SwearCount AS INTEGER
DIM Swears(1 TO 50) AS STRING
    Swears(1) = "ASS"
    Swears(2) = "BASTARD"
    Swears(3) = "HELL"
    SwearCount = 3

Timer1.Interval = DELAY
Timer1.OnTimer = TimerExpired
NumClients = 0
MasterSocket = Socket.Open(PortNum)
IF MasterSocket = -1 THEN
  PRINT "Server error: could not make connection, maybe try another port..."
  END
END IF
Form.ShowModal

SUB SwapClients(N AS INTEGER)
  DIM I AS INTEGER
  FOR I = N to NumClients
    Client(I) = Client(I+1)
``` | The SERVER program (only on the server computer). The SERVER program must be on (even if it is minimized) to enable the communication network to work). |

SNO	CODING	EXP

	``
	    NEXT
	  END SUB
	
	  FUNCTION Filter(Message AS STRING) AS STRING
	    DIM I AS INTEGER, N AS INTEGER
	    DIM X AS INTEGER, Y AS INTEGER
	
	    FOR I = 1 To SwearCount
	      N = INSTR(UCASE$(Message), Swears(I))
	      WHILE N
	        Message = LEFT$(Message, N-1) + STRING$(LEN(Swears(I)), "*") + _
	              MID$(Message, N+LEN(Swears(I)), LEN(Message))
	        N = INSTR(UCASE$(Message), Swears(I))
	      WEND
	    NEXT I
	    Filter = Message
	  END FUNCTION
	
	  SUB SendToClients (Message AS STRING)
	    DIM I AS INTEGER, N AS INTEGER
	    FOR I = 1 to NumClients
	      N = Socket.WriteLine(Client(I), Message)
	    NEXT
	  END SUB
	
	  SUB TimerExpired
	    DIM I AS INTEGER
	    DIM S AS STRING
	    Timer1.Enabled = False
	    Timer1.Interval = DELAY
	    IF Socket.ConnectionReady(MasterSocket) > 0 THEN
	      NumClients = NumClients + 1
	      Client(NumClients) = Socket.Accept(MasterSocket)
	      ListBox.AddItems "Client "+STR$(NumClients)+": Connected to server."
	      IF Socket.WriteLine(Client(NumClients), "Welcome to Andre Botha's Chat Server!") = -1 THEN
	        ListBox.AddItems "Client "+STR$(NumClients)+": Disconnected from server."
	        NumClients = NumClients - 1
	      ELSE
	        Label3.Caption = "Connections: "+STR$(NumClients)
	      END IF
	    END IF
	    FOR I = 1 to NumClients
	      IF Socket.IsClientReady(MasterSocket, Client(I)) > 0 THEN
	        S = Socket.ReadLine(Client(I))
	        ListBox.AddItems "Client "+STR$(I)+": "+S
	        IF Socket.Transferred = -1 THEN
	          ListBox.DelItems ListBox.ItemCount-1
	          ListBox.AddItems "Client "+STR$(I)+": Disconnected from server."
	          NumClients = NumClients - 1
	          Label3.Caption = "Connections: "+STR$(NumClients)
	          SwapClients(I)
	          SendToClients("Client "+STR$(I)+": Disconnected.")
	        ELSE
	          S = Filter(S)
	          SendToClients("Client "+STR$(I)+": "+S)
	        END IF
	      END IF
	    NEXT
	    Timer1.Enabled = True
	  END SUB
	``	

The sub computers will require the <mark>LOG ON Port number</mark> and LOG ON computer name (server name).

SNO	CODING	EXP
| 1 | ```
$APPTYPE GUI
$TYPECHECK ON
CONST DELAY = 10:cONST False = 0:CONST True = NOT False
DECLARE SUB SendButtonClick
DECLARE SUB ButtonClick
DECLARE SUB TimerExpired
DIM Socket AS QSocket
DIM SockNum AS INTEGER
   SockNum = 0

CREATE MainForm AS QForm
  Height = 295:Width = 400
  Caption = "Chat Client - by Andre Botha"
  Center:borderstyle=4
  CREATE Label1 AS QLabel
    Top = 12:Left = 5:Caption = "Server:"
  END CREATE
  CREATE Label2 AS QLabel
    Top = 39:Left = 5:Caption = "Port:"
  END CREATE
  CREATE Edit1 AS QEdit
    Left = 50:Top = 10:Width = 220
    Height = 20:Text = "localhost"
  END CREATE
  CREATE Edit2 AS QEdit
    Left = 50:Top = 37:Width = 220
    Height = 20:Text = "5000"
  END CREATE
  CREATE Button AS QButton
    Left = 50:Top = 65:Height = 20
    Width = 220:Caption = "Connect"
    OnClick = ButtonClick
  END CREATE
  CREATE ListBox AS QListBox
    Top = 95:Left = 50:Width = 220:Height = 140
  END CREATE
  CREATE Label3 AS QLabel
    Top = 98:Caption = "Message"
  END CREATE
  CREATE Label4 AS QLabel
    Top = 242:Left = 6:Caption = "Text:"
  END CREATE
  CREATE Edit3 AS QEdit
    Left = 50:Top = 240:Width = 170:Height = 20
  END CREATE
  CREATE SendButton AS QButton
    Left = 225:Top = 240:Height = 20:Width = 50:Caption = "Send":OnClick = SendButtonClick
  END CREATE
END CREATE
DIM Timer1 AS QTimer
   Timer1.Interval = DELAY
   Timer1.OnTimer = TimerExpired
SUB TimerExpired
  Timer1.Enabled = False
  Timer1.Interval = DELAY
  IF SockNum <= 0 THEN
    Timer1.Enabled = True
    EXIT SUB
  END IF
  IF Socket.IsServerReady(SockNum) THEN
    ListBox.InsertItem 0, Socket.ReadLine(SockNum)
    IF Socket.Transferred  < 0 THEN
      SockNum = 0
      ShowMessage "Server disconnected!"
``` | The CLIENT program – all slave computers (including server to be used as a client). |

SNO	CODING	EXP

	END IF
	  END IF
	  Timer1.Enabled = True
	END SUB
	
	SUB SendButtonClick
	  IF SockNum <= 0 THEN
	    ShowMessage "You must be connected to a server first!"
	  ELSE
	    IF Socket.WriteLine(SockNum, Edit3.Text) = -1 THEN
	      Button.Caption = "Disconnect"
	      ShowMessage "ERROR: Disconnected from Server!"
	    END IF
	  END IF
	END SUB
	
	SUB ButtonClick
	  IF Button.Caption = "Disconnect" THEN
	    Timer1.Enabled = False
	    Button.Caption = "Connect"
	    Socket.Close(SockNum)
	    SockNum = 0
	    Timer1.Enabled = True
	    EXIT SUB
	  END IF
	  SockNum = Socket.Connect(Edit1.Text, VAL(Edit2.Text))
	  IF SockNum > 0 THEN
	    'ShowMessage(Socket.GetPeerName(SockNum))
	    Timer1.Enabled = False
	    Button.Caption = "Disconnect"
	    ListBox.InsertItem 0, Socket.ReadLine(SockNum)
	    Timer1.Enabled = True
	    edit3.text=""
	  ELSE
	    SockNum = 0
	    ShowMessage "Unable to connect to: "+Edit1.Text+" on port "+Edit2.Text
	  END IF
	END SUB
	MainForm.ShowModal	

Instant networking is much faster (direct communication method – it is lost once you close the game/session/program), however it is for gaming, databases should be controlled from a real file stored on a server.