



# CURL in Detail

What CURL plugin can do for you.



# About CURL



- CURL command line tool
  - included with Mac OS X & Linux
  - available for Windows
- libCURL, the C library behind



# About CURL



- Transfers data in various protocols
- SSL with Certificates
- Proxies
- Cookies
- Authentications



# About CURL



- File Transfer Resume
- Proxy Tunneling
- HTTP POST/PUT
- HTTP Form based upload
- FTP upload



# Protocols



- DICT
- FILE
- FTP(s)
- Gopher
- HTTP(s)
- IMAP(s)
- LDAP(s)
- POP3(s)
- RTMP & RTSP
- SCP
- SFTP
- SMTP(s)
- Telnet
- TFTP



# LibCURL



- Portable C Library for Mac, Windows, Linux and others
- Thread Safe
- IPv6 Compatible
- Feature Rich
- Fast
- Well documented



# MBS CURL Plugin



- CURLSMBS class
- Built in CURL, SSH and SSL libraries.
- Background transfers with Threading
- Various simple & advanced interfaces



# MBS CURL Plugin



- Simple download:
- **dim c as new CURLSMBS**  
**c.CollectOutputData = true**  
**c.OptionURL = "http://www.mbsplugins.de/"**  
**dim e as integer = c.Perform**  
**dim h as integer = c.GetInfoResponseCode**  
**dim d as string = c.OutputData**  
**Break**
- See result in debugger.



# Options



- OptionURL - the target URL
- OptionUpload - upload or download?
- OptionPOST, OptionPUT, OptionCustomRequest
- OptionTimeOut
- OptionVerbose - enable debug messages
- OptionFollowLocation - follow redirects



# Input / Output



- Via Events
- Collect in memory and store in property (beware of size limits!)
- Provide input via variable in memory
- Provide input via form data
- Stream directly from/to file



# Receiving Debug Messages



- event DebugMessage(infotype as integer, data as string)
- CollectDebugData = true  
Later query DebugData property.
- CreateMTDebugOutputFile & CloseMTDebugOutputFile



# Receiving Header



- event Header(data as string) as integer
- CollectHeaderData = true  
Later query HeaderData property.
- CreateMTHeaderOutputFile &  
CloseMTHeaderOutputFile



# Receiving Data (Download)



- event Write(data as string) as integer
- CollectOutputData = true  
Later query OutputData property.
- CreateMTOutputFile & CloseMTOutputFile



# Data Upload



- event Read(count as integer) as string  
event RestartRead() as boolean  
event Seek(pos as Int64, whence as integer) as Integer
- property InputData as String  
method SetInputData(data as MemoryBlock)  
method SetInputData(data as string)
- OpenMTInputFile
- OptionPostFields - for POST
- FormAdd & FormFinish



# Perform transfer



- Perform
  - Can block GUI
  - YieldTime property
- PerformMT
  - run on Xojo Thread
  - Performs transfer on background preemptive thread
  - no read event!



# Information while transfer



- event Progress(dltotal as double, dlnow as double, ultotal as double, ulnow as double) as boolean
- Return true to cancel transfer. Or set Cancel property.
- Live data via DebugMessage, Write & Header events.



# Info after transfer



- GetInfoEffectiveURL - URL used
- GetInfoRedirectCount
- GetInfoResponseCode - HTTP Status codes
- various times
- content length uploaded/downloaded



# Build form data



- `d.FormAdd(d.kFormCopyName,"name",  
d.kFormCopyContents, "Mr. Tester")`
- `d.FormAdd(d.kFormCopyName,"email",  
d.kFormCopyContents, "test@test.invalid")`
- `d.FormAdd(d.kFormCopyName,"myfile",  
d.kFormBuffer, file.Name, d.kFormBufferPtr, buf,  
d.kFormBufferLength, lenb(buf))`
- `d.FormFinish`



# FTP Operations



- Run native FTP commands
- OptionQuote - instead of transfer  
OptionPreQuote - before transfer  
OptionPostQuote - after transfer
- SetOptionPostQuote array("DELETE Temp.txt")
- dim commands() As String  
commands.Append "RNFR Temp.txt"  
commands.append "RNTD MyFile.txt"  
d.SetOptionPostQuote(commands)



# IMAP Email access



- List emails:  
OptionURL = "imap://" + Server + "/INBOX"  
OptionCustomRequest = "FETCH 1:\* FLAGS"
- Download Email with ID:  
OptionURL = "imap://" + Server + "/INBOX;UID="+UID
- Create Folder:  
OptionURL = "imap://" + Server + "/"  
OptionCustomRequest = "CREATE MyFolder"



# Send Email via SMTP



- First build email source code  
e.g. with EmailMessage class
- Use email source as input data for transfer
- Set OptionMailFrom and OptionMailRecipients
- Use URL with SMTP/SMTPS prefix
- Run upload transfer



# Send Email via SMTP



- `dim em as new EmailMessage`  
`em.AddRecipient ToName.text`  
`em.FromAddress = FromName.text`  
`em.Subject = Subject.text`  
`em.BodyPlainText = content.text`
- `d.InputData = em.source`  
`d.OptionMailFrom = FromName.text`  
`d.SetOptionMailRecipients array(ToName.text)`  
`d.OptionURL = "smtp://" + server.Text`  
`d.OptionUpload = true`



# Send Email via SMTP



- CURL Email MBS class
- Text, RichText and HTML content
- Add Receipts TO/CC/BCC, From and ReplyTo
- Set subject
- Add attachments
- Supports inline graphics for html



# Send Email via SMTP



- Plugin takes care about encoding
  - Default UTF-8 encoding
  - Alternative encoding possible.
  - Quoted Printable Encoding for text
  - Base64 encoded attachments
  - Plugin encodes names and subject!



# Send Email via SMTP



- Set SMTP Server
- Enable SSL/TLS
- Pass email to CURLSMBS object for upload
- Get detailed debug log from CURL



# Example, setup email



```
dim email as new CURLEmailMBS
email.SetFrom FromEmail.Text, FromName.Text
email.AddTo ToEmail.Text, ToName.text
email.SMTPUsername = Username.Text
email.SMTPServer = server.Text
email.SMTPPassword = Passwort.Text
email.Subject = Subject.Text
email.PlainText = content.text
```

```
dim pic as Picture = LogoMBS(500)
dim jpegData as string = PictureToJPEGStringMBS(pic, 80)

email.AddAttachment jpegData, "mbs.jpg", "image/jpeg"
```



# Example, send email



```
dim d as new CURLSMBS
call d.SetupEmail(email)
d.YieldTime = true
d.OptionVerbose = true
d.CollectOutputData = true
d.CollectDebugData = true

dim e as integer = d.Perform
if e = 0 then
    d.ProgressMessage = "Result: OK"
else
    d.ProgressMessage = "Result: " + str(e)
end if
```



# SSL



- Without security:
  - `d.OptionSSLVerifyHost = 0`  
`d.OptionSSLVerifyPeer = 0`
- With security:
  - get cacert.pem file or use own certificate
  - `d.OptionSSLVerifyHost = 2`  
`d.OptionSSLVerifyPeer = 1`  
`d.OptionCAInfo = cacert.NativePath`



# Resume Upload/ Download



- Set OptionResumeFromLarge before transfer
- Provide data starting given position (Upload)
- Receive data starting given position (Download)



# iOS?



- Plugin SDK not yet available
- No CURL on iOS for declares
- NSURLConnection -> MBS Xojo Network Kit



# MBS Xojo Network Kit



- Wrapper for Bonjour classes
  - Wrapper for NSURLConnection
  - For OS X and iOS
- 
- \$99 USD with full source code. Part of iOS Kits



# MBS Xojo Network Kit



- HTTP and HTTPS support.
- FTP download support.
- Synchronously and asynchronously transfers
- Load to memory or stream to file
- HTTP Upload



# MBS Xojo Network Kit



- Supply HTTP Header fields
- Supply HTTP Body for upload
- Supply HTTP Method like GET, PUT, POST or others.
- Control about cache handling
- Can use pipelining with HTTP 1.1
- Define if to use cellular network or not.



# Questions?

- Questions?

