

EntwicklerCamp 2012

Rediscover the *Power* of LotusScript In Notes/Domino 8.5!

By *Rocky Oliver*
Collaborative Technologies Geek

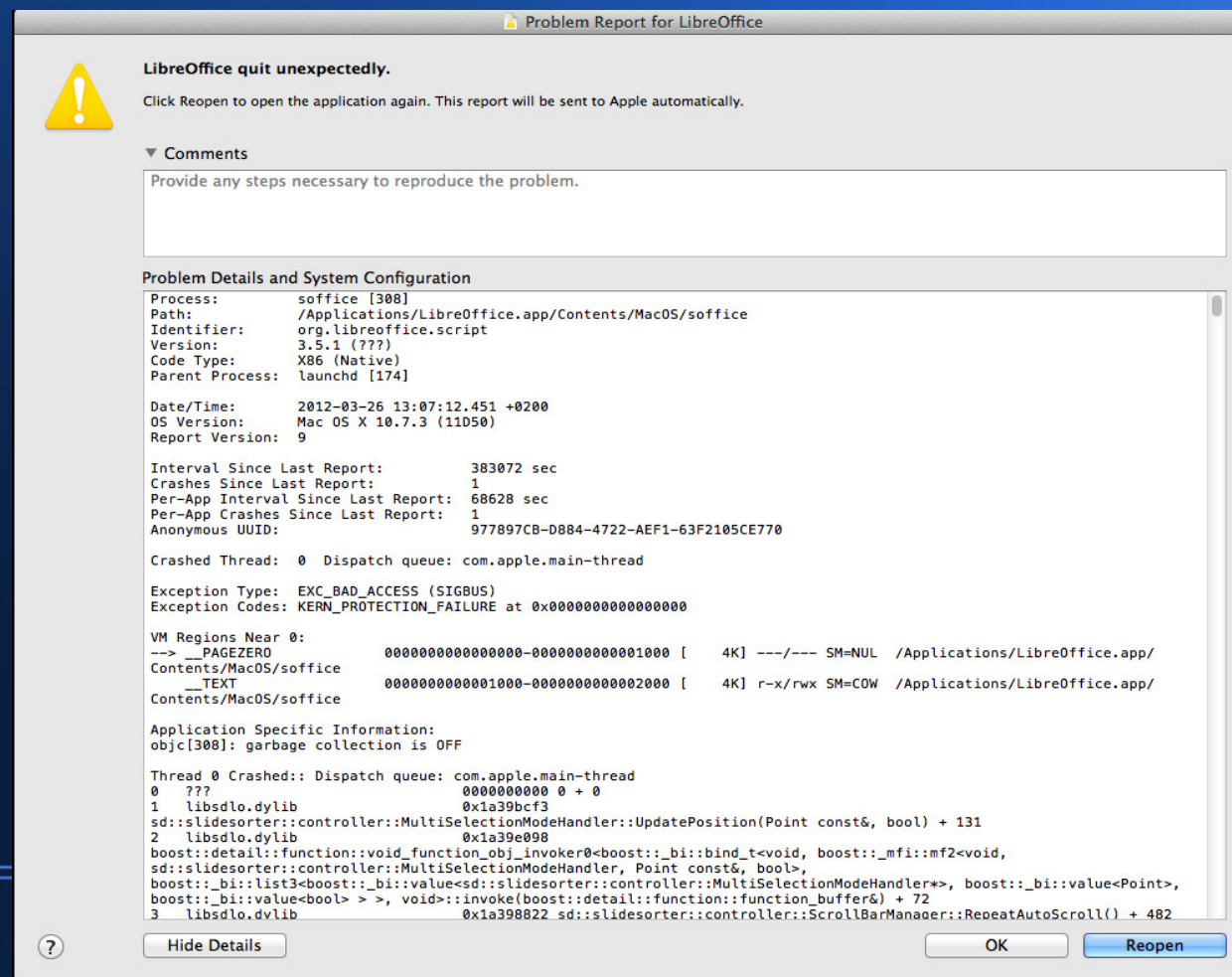
rocky.oliver@gmail.com
www.linkedin.com/in/lotusgeek

Introduction

- Began in Lotus-land in 1992
- Worked for large consultancies, pre-ipo startups, and out on my own
- Worked for Lotus/IBM – twice!
- Written a couple of books
 - Latest was “Notes and Domino 6 Programming Bible” (Wiley, 2003)
- I'm available – HIRE ME!

Apologies...

- I've had trouble putting this slide deck together...



LotusScript - Background

- Since 8.5, most Domino Developers have forgotten about the “old school” programming languages – LotusScript and @Formulas (tomorrow!)
- LotusScript (LS) has been around since Notes 4
- LS is based of the N/D core API
- LS is based on MS Visual Basic (sshhhh!)
- Unlike VB, LS is truly an object-oriented programming environment
- LS supports both front-end and back-end objects
 - NOTE: Front end (i.e. UI) for client ONLY

LS Background Continued

- LS is usually faster than Java or JavaScript when working with the N/D api, and as fast as a minimum
- You can write using pure OOP, or in traditional, “procedural” coding
 - Used to suck trying to write in OOP in the past, but DDE and the new LotusScript Editor makes this MUCH easier
 - The new LS Editor is much more robust when writing LS in general
- LS continues to be improved and kept current (but to be honest, this may change)
- Knowing LS object model will help you with xpages!
 - Know it! Live it! Love it!

LotusScript – Working with DDE

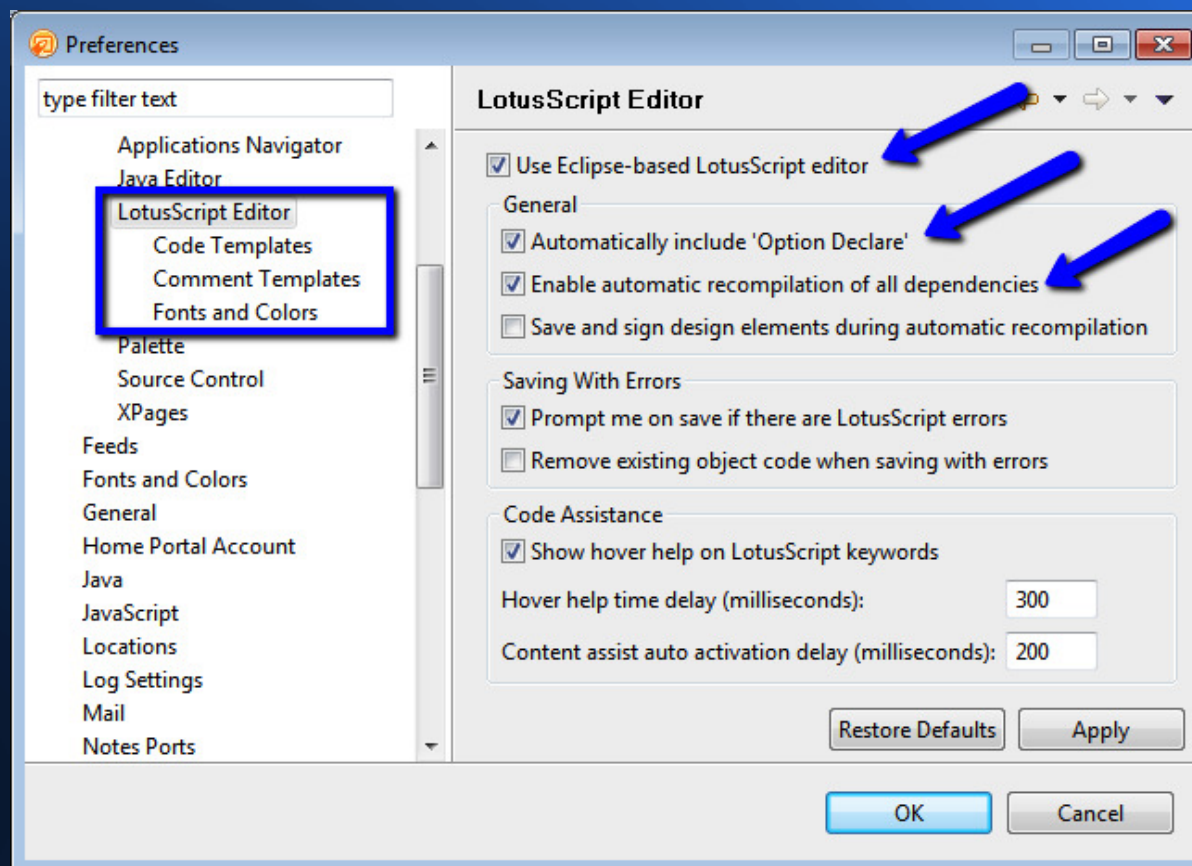
- Domino Designer for Eclipse (DDE) is a complete rewrite of the Domino Designer client
 - It modernizes and revolutionizes N/D development
 - Allows you to develop in a variety of ways, working with other platforms as needed
 - Easier script language development, including LotusScript, Java and JavaScript, as full-featured editors for each are provided by Eclipse
 - LotusScript is one of the biggest improvements
 - LEARN AND EXPLORE the DDE client – you'll be pleasantly surprised!

DDE – Feature Highlights

- There are quite a few features that make writing LS easier
- Many of these are controlled in Preferences
- Check these settings the first time you enter DDE
- Recommended settings
- Here are a few Preference suggestions...

DDE – Recommended Settings

- Open the Preferences, type filter text, LotusScript Editor...



DDE – Recommended Settings

- Templates
 - No, not the NTF kind – the LS kind!
 - Templates insert “predefined” text into your Subs, Functions, Design Elements, and even Comments
 - Never have to type that stuff you type for every sub, function, and element (almost)
 - With this, there is NO EXCUSE for not using proper error handling (at a minimum), proper comments
- Quick Tour and Demo

One Final Recommendation

(for now)

- The fonts used in DDE are not that great, especially when fixed font size is needed
- Free programming fonts are available – and they are GREAT (you just saw them)
- Proggy Fonts
 - www.proggyfonts.com

LotusScript – Ground Rules

- There are a few “mandatory” requirements for LotusScript development (you should know these already; if not, here's a primer)
 - Use Option Declare or Option Explicit
 - Added by default in DDE, if enabled
 - Use readable variable names
 - Comment your code *intelligently*
 - Use Error trapping, including logging, where feasible and appropriate
 - OpenNTF – OpenLog. Highly recommended
- NOTE: Designer Templates make many of these MUCH easier

Enhancements in 8.x

- Bug fixes AND enhancements continue to be added to LS
- For instance...
 - Native support for working with Lotus Symphony from LotusScript (*%INCLUDE SymphonyLSX.lss*)
 - Class browser – for your own custom classes!
 - Typeahead for your custom classes as well!
 - Save your code, even if it has errors in it
 - Source Control – even for N/D design elements, including LS
 - Xpage and Form/View perspectives (quick demo)

Enhancements in 8.5.3

- There are 3 new “Perspectives” in DDE
 - Window, “Open Perspective...”
 - Domino Designer – this is the one that is the “standard”
 - Forms/Views – this is for the developer who is working on a traditional N/D application and doesn't want or need the extra Xpage stuff
 - Xpages – loads everything, plus it configures the environment to make it easier to add Controls and manage Custom Controls
 - Quick Tour/Demo

New Stuff in LS

- `NotesDocument.getItemValue` is much more efficient for large group processing
 - OLD: get doc, get item, get value(s), keep item in memory until doc is destroyed OR script ends
 - NEW: get doc, get item, get value(s), DESTROY item after access completed, which makes more mem available
 - Could affect performance a small amount, but prevents “LSXBE: Out of Backend Memory” error

New Stuff in LS

- Need an empty NotesDocumentCollection?
- There are 2 ways:
 - Create it by using something like `Set emptycol = myview.getAllDocumentsByKey("Some Nonexistent Key")`
 - Create it by using undocumented method – `Set mycol = mydb.createDocumentCollection`

Slight Tangent...

- Technique for getting the size of any NotesDocumentCollection very quickly...
 - Create your source NotesDocumentCollection
 - Make sure your database has an “All Documents” view, and that it has a column with document sizes
 - Create a NotesViewEntryCollection containing all documents from your All Documents view, i.e.
`Set allEntries = AllDocsView.AllEntries`
 - Use `NotesViewEntryCollection.Intersect`
 - NOTE: It WILL work with a ViewCollection intersecting with a DocCollection!

Troubleshooting LotusScript

- You can enable NSD to capture the current condition and memory usage for LotusScript
- Add **DEBUG_LS_DUMP=1** to ***NOTES.INI***
- You can enable this for both the Notes 8.5.3 client and the Domino 8.5.3 server

Troubleshooting LotusScript

- NSD Stack Dump:

LotusScript Call Stack:

```
<@@ ----- LotusScript Interpreter -> Call Stack for  
[ nlnotes: 0ef8: 1494] (Time 10:38:14) ----- @@>  
[3] COUNTVIEWS  
[2] GETDBINFO @ line number 7  
[1] COUNTDOCUMENTS @ line number 12  
[0] INITIALIZE @ line number 2
```

- This is interpreted as...

Troubleshooting LotusScript

- The trace stack dump is pretty easy to read:

Sub Initialize was at line 2, which called CountDocuments

Sub CountDocuments was at line 12 when it called GetDbInfo

Sub GetDbInfo called CountViews from line 7

There was a crash in CountViews

StampAllMulti Method

- New method for *NotesDocumentCollection*, *NotesViewEntryCollection*
- Like StampAll, but allows you to add/replace multiple items on all docs/entries in a collection – at once, no save required
- Syntax: *StampAllMulti*(**document**), where document is a Notes back-end doc containing the item/value pairs you wish to write to the documents/entries in the collection