B-KH

Lotus Notes AdminCamp '06

Development for Administrators

Bill Buchan - HADSL

Introduction:1

- Bill Buchan
  - Director, HADSL. We develop administration tools.
  - Dual PCLP in v3, v4, v5, v6, and now v7 (Exam Junkie)
- Experience in large corporate environments
  - 100,000 + users
- Experience in migrating applications
  - 36,000 separate apps in the last five years!

Introduction:2

- This session leads you through best practices for dealing with applications in your environment
  - Avoid common pitfalls
  - Make your environment more secure
  - Prevent developers destroying your environment
  - Basic coding techniques in
    - @formula
    - LotusScript

Introduction: 3

- Administrators are from Mars – Developers are from Venus.....
- A good developer has some administration skills, and a good administrator has some development skills
  - This is why PCLP requires you to take one exam "from the other side"...
- This should give you a good kick start on that journey

B-KH

AdminCamp '06 Lotus Notes

Agenda

- Development, Test & Production Environments
- Deploying Applications
- Signing Agents
- Quick @Formula Language Agents
- Essential LotusScript
- Quick LotusScript Agents
- Summary

Development, Test & Production: 1

- You should have:
  - A developers “sandbox”
  - A User Testing Domain
  - A Production System
- Why ?
  - Proper change control
  - Allows developers to simulate system changes
    - Without destroying your production system!
  - Allows you to test version upgrades

Development, Test & Production: 2

- VMWare is your friend:
  - Workstation < $200
  - Server: Free
- Can create multiple partitions, and set breakpoints
- Can move partitions around between machines
- Easy to back up and roll back
- http://www.vmware.com

Development, Test & Production: 3

- Who has access?

	Administrators	Developers	Users
Development	N/A	Manager	No Access
Test	Manager	User	User
Production	Manager	User	User

Who deploys Templates ?

  – Administration Team

- How do I enforce this?

  – Special Admin-only “Signing ID”

Development, Test & Production: 4

- Isnt this a lot of work?
  - Yes – its more work than normal
  - Encourages less changes and more testing from Developers
  - Gives you a proper change control mechanism
- Prevents:
  - Unauthorised change
  - Developers having Manager access to environment

Development, Test & Production: 5

- Use different certifiers for these domains
  - And have different user ID's
  - Use multiple notes directories to run multiple clients
    - DONT use location documents!
  - DONT allow these environments to connect to each other
    - Its a security weakness

B-KH

AdminCamp '06 Lotus Notes

Agenda

- Development, Test & Production Environments
- Deploying Applications
- Signing Agents
- Quick @Formula Language Agents
- Essential LotusScript
- Quick LotusScript Agents
- Summary

Deploying Applications: 1

- Process:
  - Developers create/change applications
    - Create release notes
    - Sends template from Dev environment to Administrator
  - Administrator deploys to Test, asks users to test application
  - IF passed testing, deploys application to production
    - Else start again

Deploying Applications: 2

- Change/Release Notes
  - Created by Developers
  - Lists changes
  - Outlines deployment requirements
    - Dependencies on other databases, Platform
    - Client version issues
    - Any “restricted” agents
    - Any logging and monitoring requirements

Deploying Applications: 3

- What to watch for:
  - Hardcoded server name, application name or replica ID
    - Use “profile” documents and document how Administration team sets up application
  - Scheduled agents
    - What level of access is required
  - “Trusted Servers” requirement
    - Does this application access other servers?

Deploying Applications: 4

- Keep all versions and release notes in a database
  - A simple discussion database is a good start
  - Ensures that all templates are at least backed up
- Better:
  - Create a version control database
- Best:
  - Use a version control system such as Ciao! From Teamstudio

Deploying Applications: 5

- When deploying
  - Always use “Design, Replace”
  - Always use the “Signing ID”
  - Remember to set any scheduled agents
  - Access Control:
    - Should never contain names
    - Should always contain groups
    - Group name and/or description should point to application

Agenda

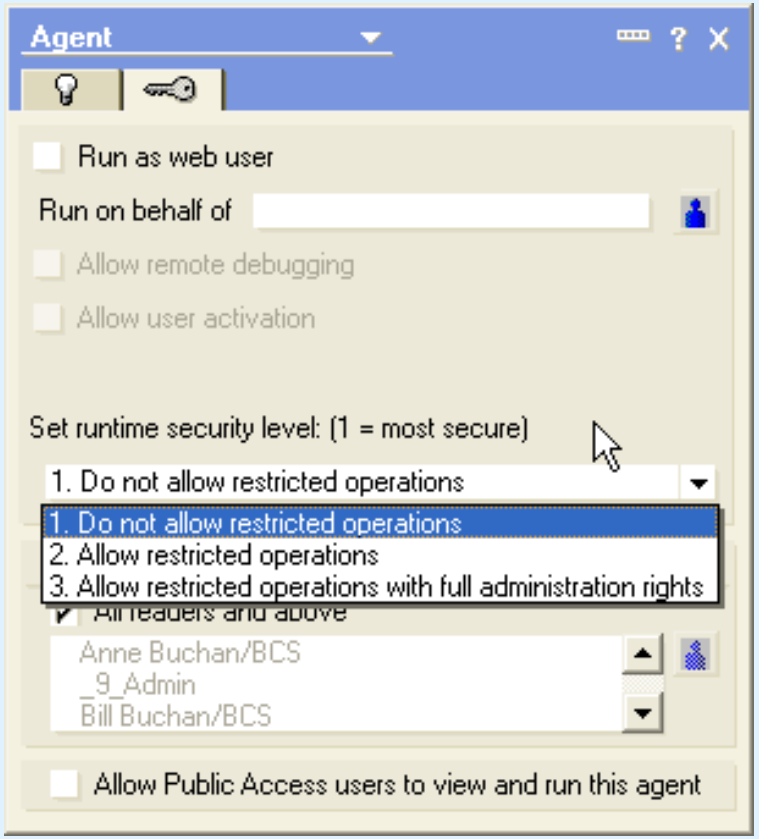
- Development, Test & Production Environments
- Deploying Applications
- Signing Agents
- Quick @Formula Language Agents
- Essential LotusScript
- Quick LotusScript Agents
- Summary

Signing Agents: 1

- All scheduled agents should be signed during deployment.
  - Remember: Developers only have “User” rights in production. Their ID may not have access to required databases
  - This means that you should use a special “Signing” ID
    - Such as “Template Development/CompanyName”

Signing Agents: 2

- Tighten up security
  - Only special ID allowed to run "restricted" agents
- Migrating to R6 ?
  - Remember that agents now have a security field which needs to be set



Agenda

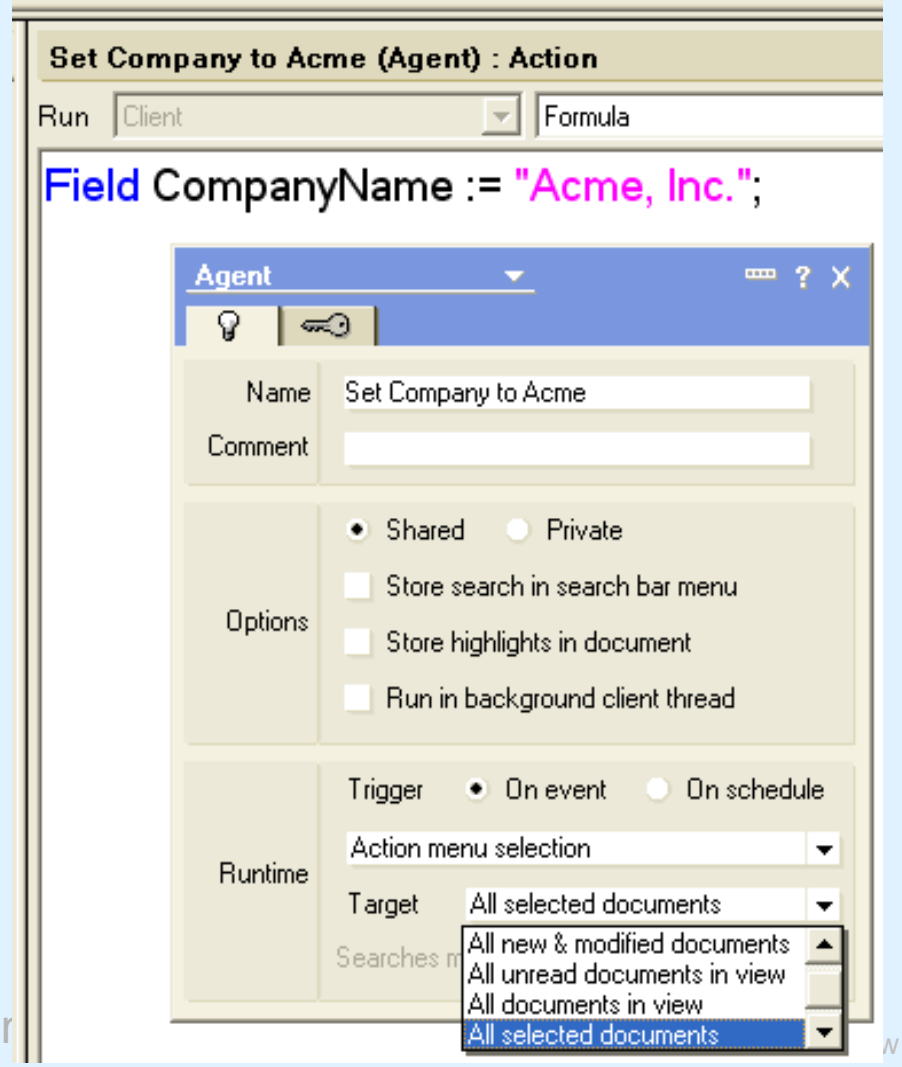
- Development, Test & Production Environments
- Deploying Applications
- Signing Agents
- Quick @Formula Language Agents
- Essential LotusScript
- Quick LotusScript Agents
- Summary

Quick @Formula Agents: 1

- What to use @Formula language for:
  - Changes to a number of documents
- What NOT to use @Formula language for:
  - User Input
  - Complex validation
  - Multi-document “relational” updates
  - If your @formula is more than 15 lines, consider LotusScript

Quick @Formula Agents: 2

- Example:
  - Set the CompanyName field to “Acme, Inc.”



Agenda

- Development, Test & Production Environments
- Deploying Applications
- Signing Agents
- Quick @Formula Language Agents
- Essential LotusScript
- Quick LotusScript Agents
- Summary

Essential LotusScript: 1

- LotusScript is better for:
  - Complex transactions, complex business rules
  - Creating Administration Requests
    - EG @Formula doesnt “sign” fields
  - Some form of UI
- Its harder to get into
- Its easier to Debug

Essential LotusScript: 2

- LotusScript is just like Visual Basic
- Accessing Notes objects is via the Notes.. Objects
- Backend code – document manipulation
- Frontend code – modifying User Interface documents
- Most of your work will be
  - Writing backend agents to modify documents
  - Call the Notes Administration tools

Essential LotusScript: 3

- Agents can be
  - Triggered by the user from the action bar
  - Scheduled and ran on workstations and servers
  - Agents can be passed “selected documents”
- Agents run as
  - The user who signed the code on the server
  - The current user on the workstation

Essential LotusScript: 4

- Lots of online help
- Lots of examples in the help
- If you dont understand it – dont do it
- Test in your development environment!
- Be careful when changing documents in scheduled agents
  - Replication storm!

B-KH

AdminCamp '06 Lotus Notes

Agenda

- Development, Test & Production Environments
- Deploying Applications
- Signing Agents
- Quick @Formula Language Agents
- Essential LotusScript
- Quick LotusScript Agents
- Summary

Quick LotusScript Agents: 1

- A simple agent:
  - Process all selected documents, and set the company name to "Acme, Inc."

```
Sub Initialize

    Dim sSession As New NotesSession
    Dim dbThis As NotesDatabase
    Set dbThis = sSession.currentDatabase

    Dim dc As NotesDocumentCollection
    Set dc = dbThis.UnprocessedDocuments

    If (dc Is Nothing) Then
        Print "Nothing selected"
        Exit Sub
    End If

    If (dc.count < 1) Then
        Print "No documents selected"
        Exit Sub
    End If

    Dim doc As NotesDocument
    Set doc = dc.GetFirstDocument

    While (Not doc Is Nothing)

        Call doc.ReplaceItemValue("CompanyName", "Acme, Inc.")
        Call doc.save(False, False)


        Set doc = dc.GetNextDocument(doc)
    Wend


End Sub
```

Quick LotusScript Agents: 2

- Its not as scary as it looks:
  - Everything is an Object
  - You drill down through logical structures
  - You can loop
- Remember:
  - On-line help
  - Lots and lots of online examples
    - http://www.notes.net

Quick LotusScript Agents: 3

- Lets ask for some user details, and then register that user.
  - This is EXAMPLE CODE
  - Dont use this in production!

```
Sub Initialize

    Dim ws As New NotesUIWorkspace

    Dim lastName As String, firstName As String, MI As String
    lastName = ws.Prompt(3, "Enter a last Name", "New User's Last Name")
    firstName = ws.Prompt(3, "Enter a first Name", "New User's First Name")
    MI = ws.Prompt(3, "Enter Middle Initials", "New User's Middle Initials")

    If (firstName = "" Or lastName = "") Then
        Print "User pressed Cancel"
        Exit Sub
    End If

    ' Now go off and register this user

    Const CertifierIDFileName = "c:\temp\acme.id"
    ' Please do NOT use this in production. Hardcoding passwords
    ' is NOT recommended!
    Const CertifierPassword = "password"
    Const targetIDFile = "c:\temp\newID.id"

    Dim sSession As New NotesSession
    Dim dbThis As NotesDatabase
    Set dbThis = sSession.CurrentDatabase

    Dim admin As New NotesRegistration

    admin.CertifierIDFile = CertifierIDFileName
    Call admin.RegisterNewUser(LastName, targetIDFile,  dbThis.Server, _
    FirstName,   MI,  CertifierPassword)

End Sub
```

B-KH

AdminCamp '06 Lotus Notes

Quick LotusScript Agents: 4

- Lets use AdminP to add a user to a group
  - Prompt for the new username using the address dialog
  - Prompt for the group

```
Sub Initialize
    Dim sSession As New NotesSession
    Dim dbThis As NotesDatabase
    Set dbThis = sSession.CurrentDatabase

    Dim ws As New NotesUIWorkspace
    Dim newUser As String
    newUser = ws.PickListStrings(0, False)

    Dim group As String
    group = ws.PickListStrings(3, False, dbThis.Server, _
    "names.nsf", "Groups", "Select a Group", "Please select a group", 1)

    If (newUser = "" Or group = "") Then
        Print "User pressed cancel"
        Exit Sub
    End If

    Dim reg As NotesAdministrationProcess
    Set reg = sSession.CreateAdministrationProcess(dbThis.Server)

    Dim noteID As String
    noteID = reg.AddGroupMembers(group, newUser)
    If  noteID <> "" Then
        Print "The request to add: "+ newUser + " to group: " +_
        " succeeded - the Admin4.nsf Note ID is: "+ noteID

        Dim dbAdmin As NotesDatabase
        Set dbAdmin = sSession.GetDatabase(dbThis.Server, "admin4.nsf")
        Call ws.EditDocument(False, dbAdmin.GetDocumentByID(noteID))
    Else
        Print "Request to add: "+ newUser + " to group: " + group + " failed"
    End If
End Sub
```

Core code

B-KH

AdminCamp '06 Lotus Notes

Agenda

- Development, Test & Production Environments
- Deploying Applications
- Signing Agents
- Quick @Formula Language Agents
- Essential LotusScript
- Quick LotusScript Agents
- Summary

Summary

- Trust no-one!
- Proper change control saves your career
- Programming isnt hard
  - (After all, developers do it..)
- Be cautious
  - The debugger is your friend

B-KH

AdminCamp '06 Lotus Notes

Thank you

Worst Practices needs you!

Confess your stories….

www.theworstpractices.com

Bill Buchan www.billbuchan.com

hadsl www.hadsl.com