



The Iam Lotus User Group

From Frustration to Fascination: Dissecting Replication

Presented by Kim Greene & Ben Menesi



@iSeriesDomino



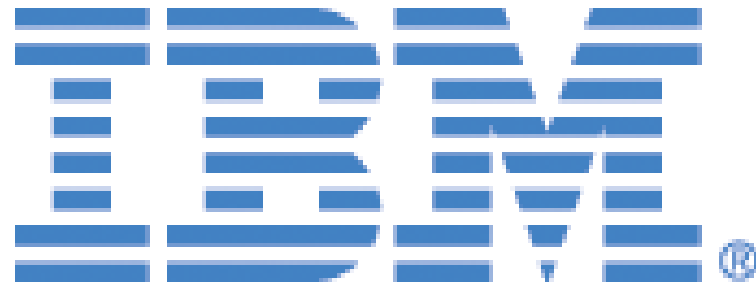
@BenMenesi



© 2013 by the individual speaker



The Iam Lotus User Group



IamLUG 2013 Sponsors

© 2013 by the individual speaker



Legal

This slide presentation may contain the following copyrighted, trademarked, and / or restricted terms:

- IBM® Lotus® Domino®, IBM® Lotus® Notes®, IBM® Notes® Domino®, LotusScript®
- Microsoft® Windows®, Microsoft Office®, Visual Basic®, Java®

Co-Author

Kim Greene

- Owner of Kim Greene Consulting, Inc.
- Over 15 years of experience with Domino and 24 years of experience with AS/400, iSeries and i5
- Services include System & Application performance optimization, Administration, upgrades, health, performance, security etc. checks, migrations, custom development, enterprise integration
- Blog: www.bleedyellow.com/blogs/dominodiva

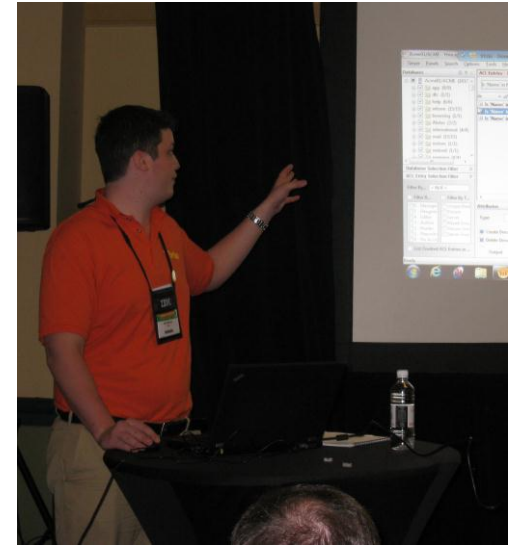


@iSeriesDomino <http://www.kimgreene.com>

Speaker

Ben Menesi

- Chief Product Officer at Ytria
- Administration and Development experience with various enterprise IBM clients
- Wearing both admin and dev hats
(this session may contain development references)
- Speaker at numerous LUG conferences (MWLUG, NLLUG, BLUG) plus IBM Connect 2013



@BenMenesi

<http://www.ytria.com>

Brief Company Description



- Advanced IBM Business Partner founded in 1999 in Montreal, Canada
- BLUG Silver Sponsor
- Released our newest product, replicationEZ in September 2012
- Idea is to manage all about replication -> loads of testing, tons of interesting stuff learned in the process.
- Promise: Not a Marketing session
- Let's cut to the chase!

Agenda

Replication is a huge subject, and is impossible to exhaustively cover in an hour. So we'll mostly be discussing the fun stuff!



Quick Introduction to Replication

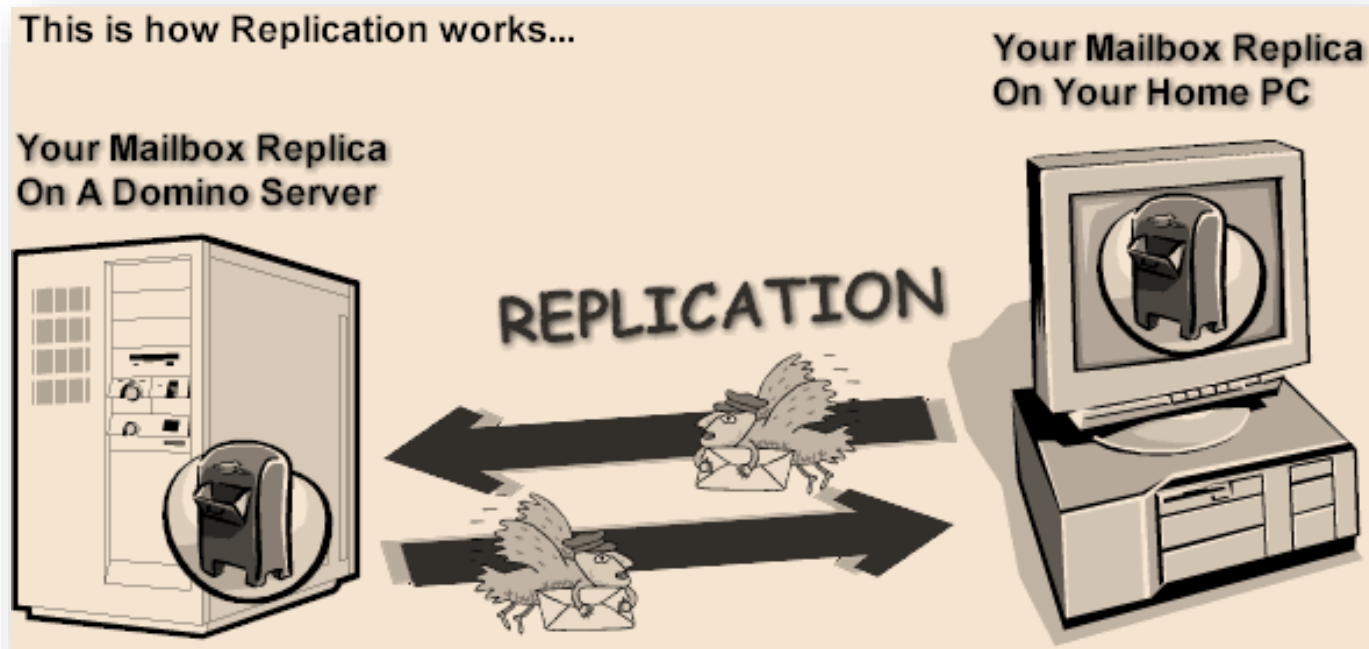
- ▶ Replicator Server task
- ▶ Connection documents
- ▶ Cluster Replication



What happens when two databases replicate?

- ▶ Security considerations, selective replication
- ▶ Document comparison, possible scenarios, problems
- ▶ Best Practices, stories, myths, mistakes, geek fun stuff

What is Replication?



- Replication is one of the main pillars of the IBM (Lotus) Notes / Domino platform
- Process that Notes and Domino use to keep replicas of databases synchronized

Who Can Create a New Replica?

- The **Create new replicas** field determines who can create a new replica database on the server

Server Access	Who can -
Access server:	<input checked="" type="checkbox"/> users listed in all trusted directories and */Kim Greene Consulting LocalDomainServers
Not access server:	
Create databases & templates:	Domino Administrators Help Desk Employees LocalDomainServers
Create new replicas:	Domino Administrators LocalDomainServers
Create master templates:	DominoAdministrators
Allowed to use monitors:	*
Not allowed to use monitors:	
Trusted servers:	

Who Can Create a New Replica

- Tip 1:
 - If you can't create replicas, but are allowed to create new databases on a given server, you can create an empty DB and change its Replica ID
 - You got around your Notes environment setup (True Story 😊)
- Tip 2:
 - If you create a new replica for a huge database manually, you might want to specify a fake Replication formula so that once it's evaluated no documents will be transferred. This will create a "Replica Stub". Then delete the formula and use the "Replicate" server task to synchronize the databases.

Quick Introduction to Replication

1.) Replication related tasks

2.) Connection documents

3.) Cluster Replication

Quick Introduction to Replication

1.) Replication related tasks

2.) Connection documents

3.) Cluster Replication

Replication Related Server Tasks

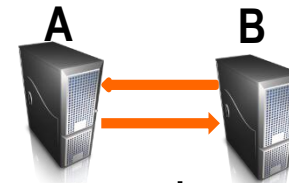
REPLICA

- Responsible for synchronizing source database with its replicas
- One instance loaded by default
 - To load multiple replica tasks:
 - Use REPLICATORS=n in notes.ini
 - 'load replica' at Domino console
- Replicator sits idle until connection document (or console command) triggers it
 - Replica task = muscle
 - Connection document = brain

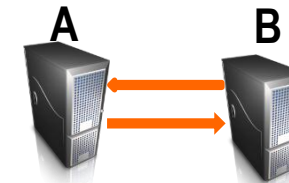
The replicator server task

Types of Replication

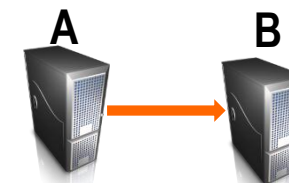
- **PULL-PUSH:** Server A contacts server B and PULLS new and modified information. Then it PUSHES all its new and modified information to server B.
 - Server A does all the work



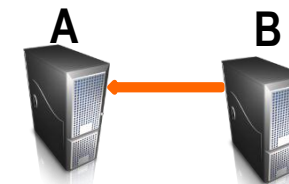
- **PULL-PULL:** Server A contacts server B and PULLS new and modified info. Then Server B contacts server A and does the same.
 - Work is shared by servers



- **PUSH-ONLY:** Server A contacts server B and PUSHES all new and modified info.
 - No updates received



- **PULL ONLY:** Server A contacts server B and PULLS any new and modified info.
 - No updates sent



The replicator server task

Console Commands

- **Rep x/ACME**
 - PULL-PUSH replication session with Server X/Acme
- **Rep x/ACME dbname.nsf**
 - PULL-PUSH dbname.nsf only on server x/Acme
- **PULL x/ACME**
 - PULL all databases with matching replica Ids from x/ACME
- **PUSH x/ACME**
 - PUSH all databases with matching replica Ids to x/ACME
- **PUSH / PULL x/ACME dbname.nsf**
 - PUSH or PULL replication of database dbname.nsf only

The replicator server task

It's replication time!

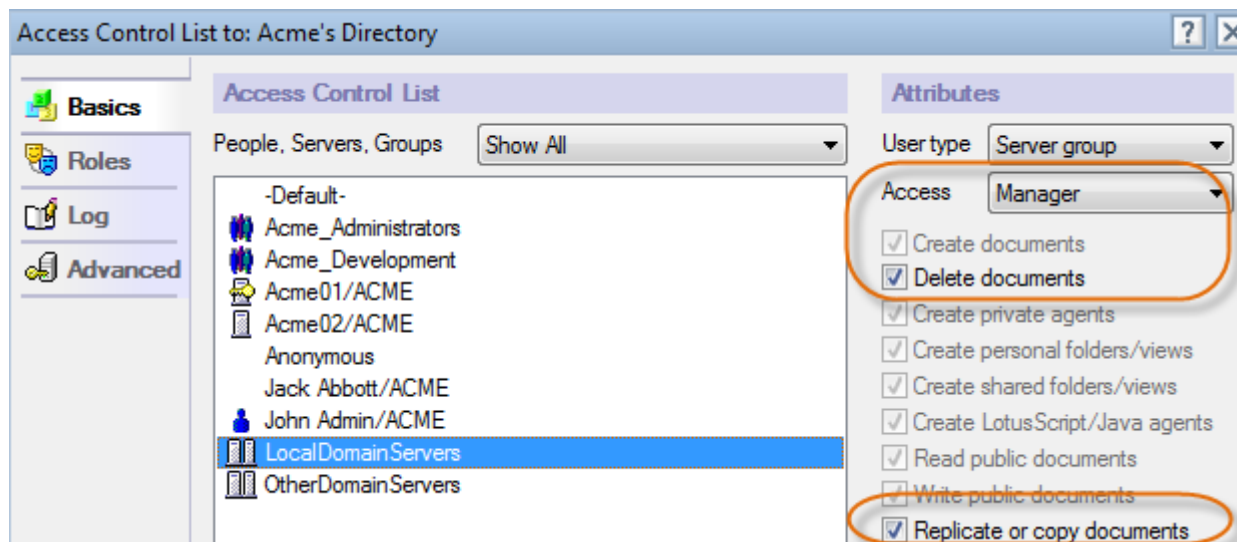
- **Server A contacts Server B**
 - Gotta have access!
 - Servers authenticate each other pretty much the same way they authenticate users when establishing a connection

Server Access	Who can -
Access server:	<input checked="" type="checkbox"/> users listed in all trusted directories and <input type="checkbox"/> LocalDomainServers *YTRIA ▾
Not access server:	<input type="checkbox"/> ▾
Create databases & templates:	<input type="checkbox"/> Acme_Administrators Acme_Development LocalDomainServers ▾
Create new replicas:	<input type="checkbox"/> Acme_Administrators Acme_Development LocalDomainServers ▾
Create master templates:	<input type="checkbox"/> Acme_Administrators Acme_Development ▾
Allowed to use monitors:	<input type="checkbox"/> * ▾
Not allowed to use monitors:	<input type="checkbox"/> ▾
Trusted servers:	<input type="checkbox"/> Acme01/ACME Acme02/ACME Acme03/ACME ▾

The replicator server task

It's replication time!

- **Building list of databases to process**
 - ACLs are used as a filter
 - No access to database = no replication



- **Replication of each database with matching Replica ID**
 - More on this later

Quick Introduction to Replication

1.) Replication related tasks

2.) Connection documents

3.) Cluster Replication

Connection Documents

Digesting connection documents

- Connection documents are definitions for server to server connections
 - Control replication and mail routing
 - Specify which databases, folders to replicate (or not!) and when
- Digesting a connection document
 - **Source Server**
 - Server doing work, REPLICA task responsible, connects to SERVER task on server

Basics	Replication/Routing	Schedule	Comments	Admin
Basics				
Connection type:	Local Area Network			
Source server:	Acme03/ACME			
Source domain:	Acme			
Use the port(s):	TCPIP			

Connection Documents

Digesting connection documents

- **What to replicate**
 - Replication type and files / directories to replicate, and avoid
 - Tip: ALWAYS replicate names.nsf, admin4.nsf and events4.nsf throughout the domain

Basics | **Replication/Routing** | Schedule | Comments | Administration

Replication

Replication task: Enabled ▾

Replicate databases of: Low & Medium & High ▾ priority

Replication type: Pull Push ▾

Files/Directory paths to replicate: Admin4.nsf, events4.nsf, names.nsf ▾ (all if none specified)

Files/Directory paths to NOT replicate: DBtoNotReplicate.nsf ▾

Replication time limit: ▾ minutes

- **Replication time limit**
 - Tip: Set the replication time limit less than the repeat interval

Connection Documents

Digesting connection documents

- **Repeat interval**
 - Starts when previous replication session ends
 - Tip: Easier to troubleshoot if using specific times

Basics	Replication/Routing	Schedule	Comments	Administr
Scheduled Connection				
Schedule:	Enabled			
Connect at times:	08:00 AM - 10:00 PM each day			
Repeat interval of:	90 minutes			
Days of week:	Sun, Mon, Tue, Wed, Thu, Fri, Sat			

Connection Documents

Interesting stuff

- How do you know where & how a given DB is supposed to replicate?
 1. Check replication schedule

The screenshot shows the IBM Domino Administration Center interface. The 'Replication' tab is selected and highlighted with a red box. The server information is 'Server: domserver1/akh' with 'Release 8.5.3 on OS400 V7R1M0'. The 'Replication Schedule' view is active, showing a grid of replication schedules for three servers: domserver2/akh, domtest1/kgci, and domtest5/akh. The grid has columns for days of the week and a 'Status' column. A yellow gear icon is visible above the grid, and a red double-headed arrow is positioned between the first and second rows of the grid.

Servers	6	6	Status
domserver2/akh			Idle
domtest1/kgci			Idle
domtest5/akh			Idle

Connection Documents

Interesting stuff

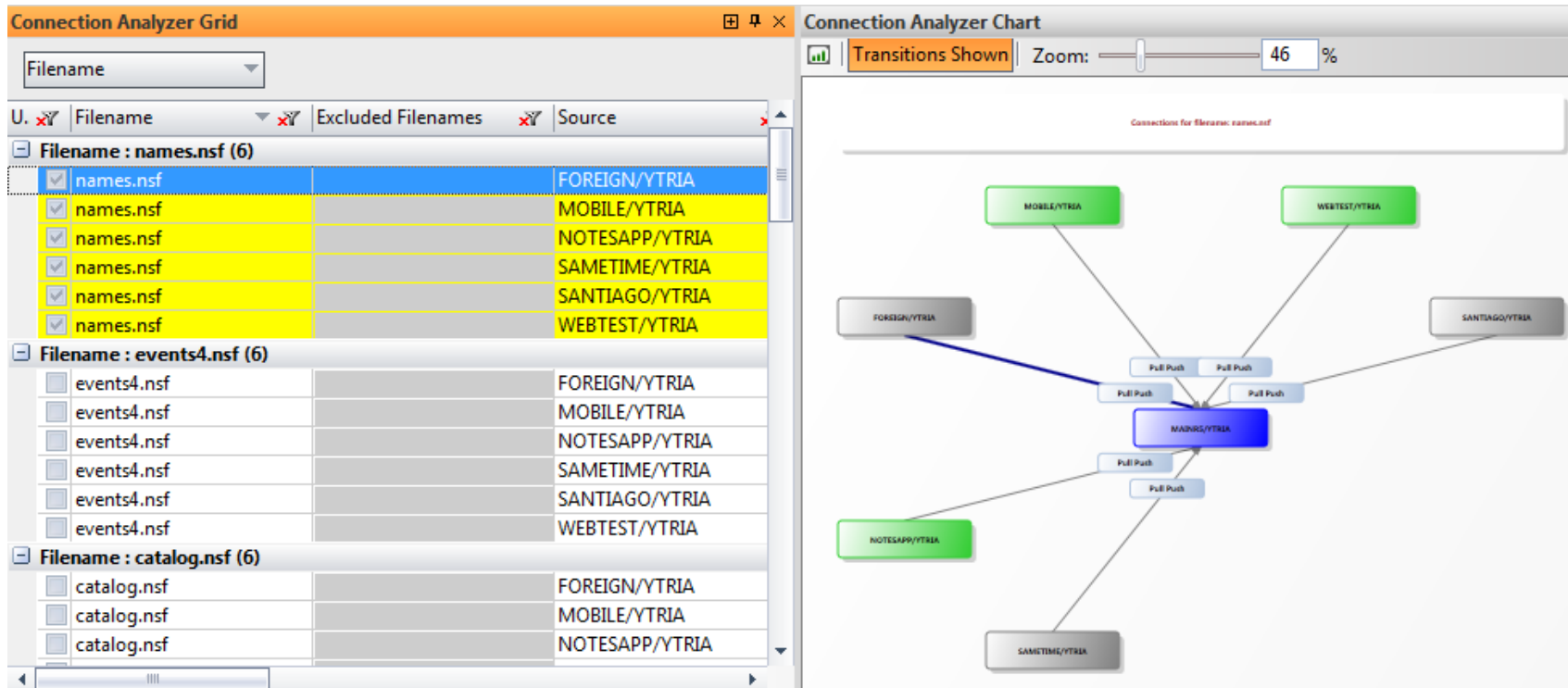
- **How do you know where & how a given DB is supposed to replicate?**
 2. Copy connection documents view and add a new column to display the 'filenames' item in the first categorized column

	From	From	To
▼ admin4.nsf			
		Acme01/ACME	Acme02/ACME
		Acme01/ACME	Acme02/ACME
		Acme01/ACME	AcmeHUB/ACME
		Acme02/ACME	AcmeHUB/ACME
▼ ddm.nsf			
▼ International			
▼ mail			
▼ names.nsf			
▼ test			
▼ (Not Categorized)			

Connection Documents

Interesting stuff

- How do you know where & how a given DB is supposed to replicate?
 1. Use the replication log
 2. Use the replication console
 3. Use a 3rd party solution such as replicationEZ



Quick Introduction to replication

1.) Replication related tasks

2.) Connection documents

3.) Cluster Replication

Domino Clustering

Quick overview of clustering

- **What is a domino cluster?**
 - Collection of 2 to 6 Domino servers (6 is a recommended max) to provide high availability and / or workload balancing
- **How does it work?**
 - Each server contains replicas of the apps to be clustered
 - If user attempts to access a clustered application and it's not available, Notes opens a replica of the DB on a different cluster server
 - Domino continuously synchronizes databases
 - CLREPL server task
- **Requirements**
 - All cluster mates share same Domino Directory
 - A given server can only be a member of one cluster at a time

Domino Clustering

Quick overview of clustering

- **CLREPL**
 - Performs cluster replication
- **CLDBDIR**
 - Responsible for maintaining cluster database directory (clbdir.nsf)
- **Cluster Manager**
 - Tracks state of all servers in cluster, and maintains workload info on each server
- **Cluster Administrator**
 - Housekeeping tasks: starts CLDBDIR and CLREPL (+Administration process should it not be up automatically)
 - New in 8.5.x versions: clustering starts automatically

Domino Clustering

Important regarding clustering

- **Clustering is In Memory replication**
 - As changes occur they are queued and sent to the cluster server(s)
 - If server goes down, changes can be lost
 - Always use traditional replication as backup
 - Control with connection documents
- **Streaming cluster replication**
 - First introduced in Domino 8.0, enhanced cluster replication
 - Event-drive, changes pushed to cluster mates as they happen
- **Tip: Ping server using NSPINGSERVER API Call**
 - Much faster (3x according to our testing) than any other method in LS
 - Returns if server is busy (ERR_SERVER_UNAVAILABLE)
 - If server is restricted (ERR_SERVER_RESTRICTED)
 - pdwIndex – Availability index for server (0 if not participating in load balancing)
 - phList – Text list containing cluster name & all cluster mates

What happens when two databases replicate?

1.) Check if access rights sufficient

2.) Check replication history

3.) Build document list for processing

4.) Apply selection formulas

5.) Compare documents

6.) Make changes

What happens when two databases replicate?

1.) Check if access rights sufficient

2.) Check replication history

3.) Build document list for processing

4.) Apply selection formulas

5.) Compare documents

6.) Make changes

Access Considerations

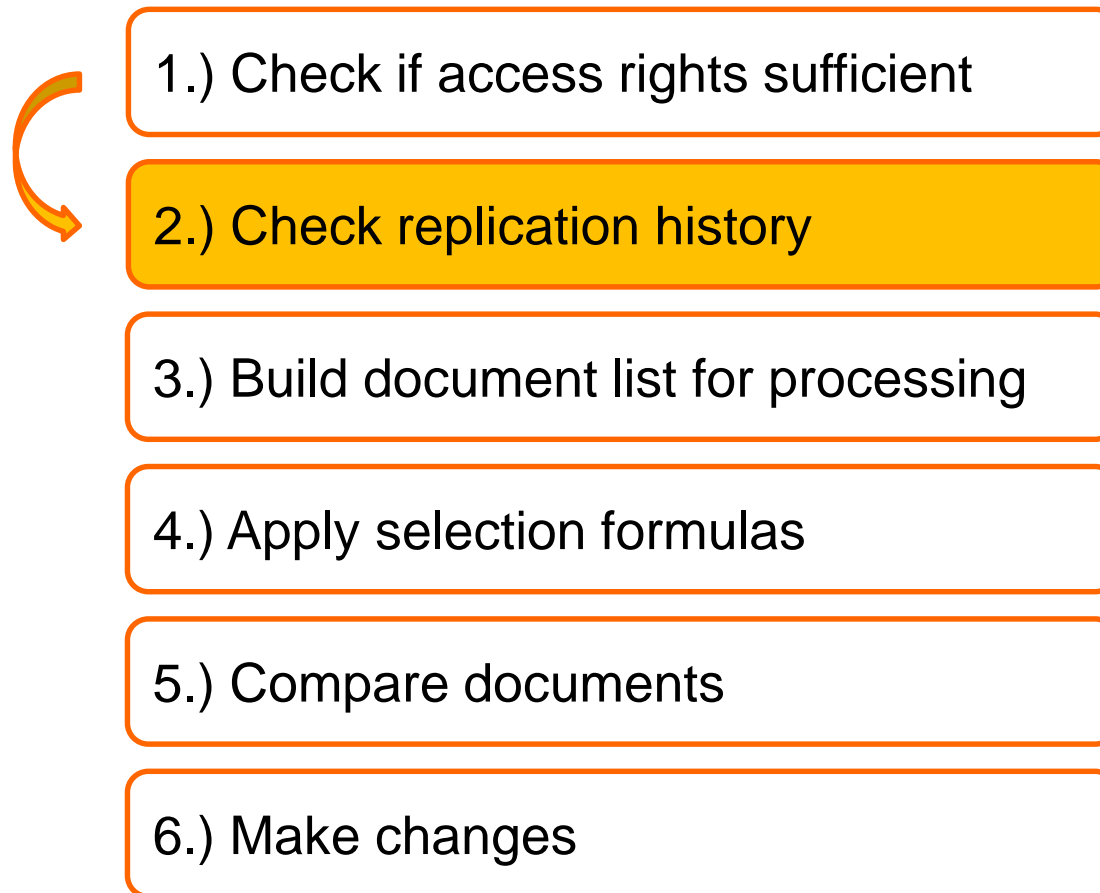
ACL level settings

- Makes sure the “source” server has access to the destination database and vice versa

Server level security considerations

- Controlled in ‘**Security**’ section of the Server Document
 - Who can ‘**Access server**’
 - If server isn’t listed, replication will fail
 - Who can ‘**Create new replicas**’
 - If server isn’t listed, new replica won’t create

What happens when two databases replicate?

- 
- 1.) Check if access rights sufficient
 - 2.) Check replication history
 - 3.) Build document list for processing
 - 4.) Apply selection formulas
 - 5.) Compare documents
 - 6.) Make changes

Replication History

How does it work?

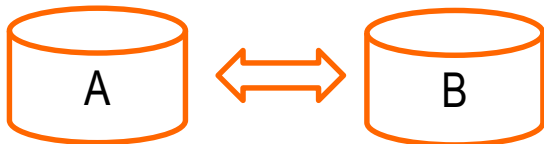
- When replicator is about to sync two DBs it looks at the history to find out what happened since their last “talk”



PUSH: Server A checks History in “Local” database (on server A)



PULL: Server A checks History in “Remote” database (on server B)



PULL & PUSH: Server A checks History in both “Local” and “Remote” DBs

Replication History

If there is no replication history or it's disregarded

- Calculation based on all documents: cleaner replication

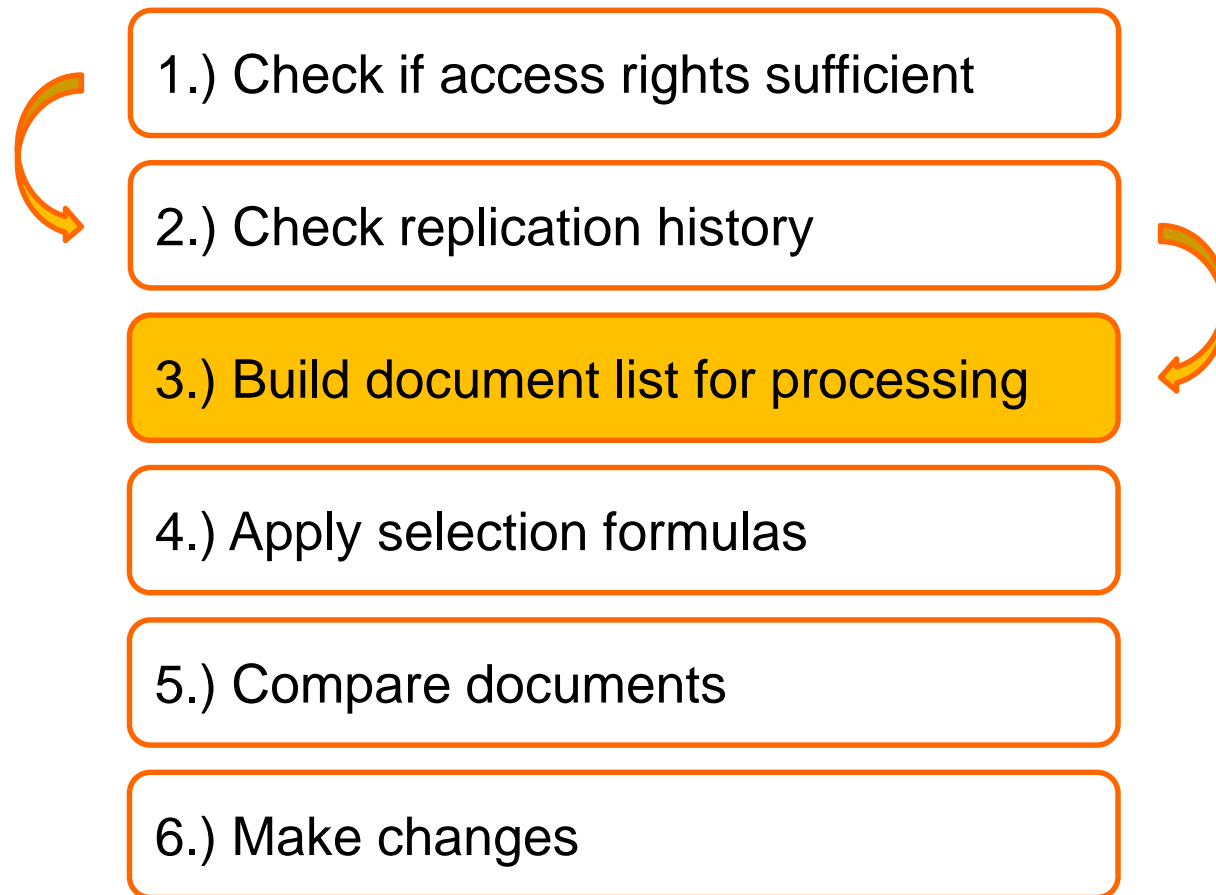
When is the replication History disregarded?

- If access rights have changed
- If Replication Formulas changed

Watch out!

- If local <> server replication, no trace in server Replication History
- If no changes detected = no trace in replication history
 - How come my DB that's supposed to replicate every 5 minutes hasn't replicated in 1.5 hours??

What happens when two databases replicate?

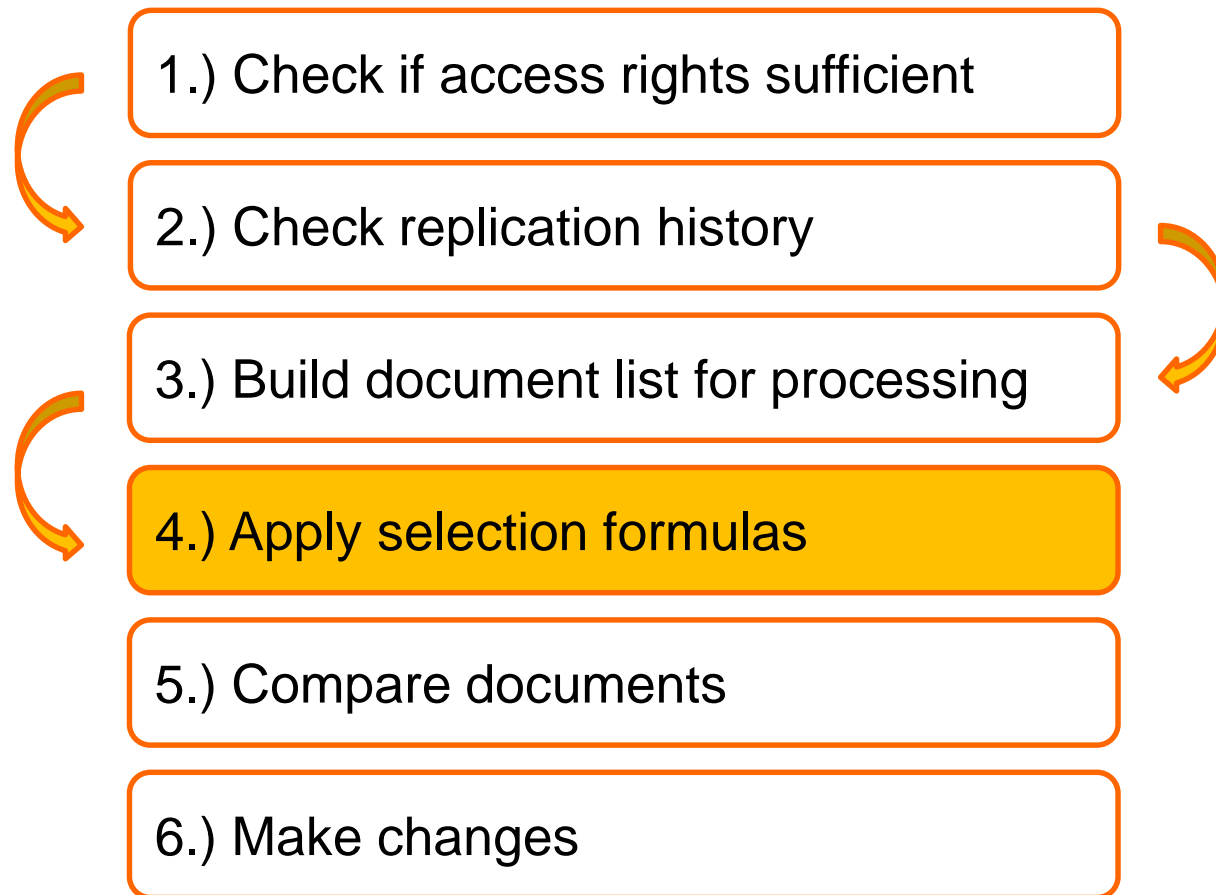


Document level security – Readers & Authors fields

Managing selective replication using readers fields

- When building document list, basic Notes security applies
 - Server needs access to documents
 - If this is not true, the server won't see document
 - (Much like how servers disregard replicating apps they have no access to)
- This is something you can play with. It's actually a technique that a lot of folks prefer over using replication formulas
- Make sure servers involved in replication are in readers fields

What happens when two databases replicate?



Selective Replication Formulas

What are Selective Replication Formulas?

- Selective Replication Formulas are one way to replicate a subset of your database contents
 - Use case
 - If you use multiple replicas on the same server (which you probably shouldn't) you want to use replication formulas instead of readers fields

Where do I find them?

- It's pretty tough to **REALLY** oversee them
 - Some think this is a database level setting, but you can have as many scenarios as you'd like and there are rules you have to consider

Selective Replication Formulas

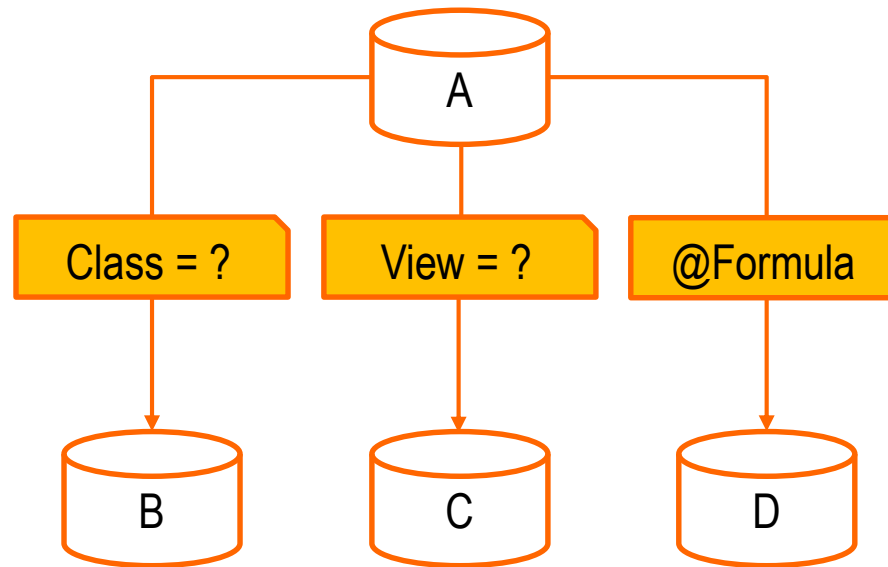
Where to set Replication Formulas?

The screenshot shows the 'Replication Options for Benedek MENESI' dialog box. It is divided into several sections:

- Basics**: Includes 'Receive only a subset of information from other replicas' with dropdowns for 'When computer' (MAINR5/YTRIA) and 'Receives from' (-Any Server-). A callout 'Scenario Configuration' points to these dropdowns.
- Advanced**: Includes checkboxes for 'Documents in specified views or folders', 'Documents by selection formula', and 'Selected documents'. A callout 'Which documents?' points to these checkboxes.
- Folders and Views**: A tree view showing folders like 'Inbox', 'Drafts', 'Sent', 'All Documents', 'Calendar', 'Grouped Entries', 'To Do', 'Follow Up', 'Junk', 'Chat History', '00 - Ytria Communication', and '01 - Ytria Colleagues'.
- Receive these elements from other replicas**: Includes checkboxes for 'Design elements', 'Agents', 'Access control list', and 'Deletions'. A callout 'Which design elements?' points to these checkboxes. There are also checkboxes for 'Replication formula' and 'Fields' with a 'Define...' button.

Selective Replication Formulas

How does it work?



- Caution! If you have a –All– Type selective replication formula note, the note that specifies the actual server will **OVERRIDE** the one specified for all scenarios

Selective Replication Formulas – Fun Facts

How are they stored in a Notes DB?

- In a pretty funky way. You'd imagine one note for each relationship / scenario, but in fact there is one note per source server
- Scenarios involving the current "Local" database are stored in one note, using multi-value fields

Name	N.	S..	D.	I..	Value
\$ReplClassMasks	4	3	0	5	30719; 1535; 1983; 513
\$ReplFields	4	3	0	13	:::
\$ReplForm	4	3	0	7	0; 0; 0; 0
\$ReplFormula	1	3	0	6	@All
\$ReplFormula	1	3	0	8	@All
\$ReplFormula	1	3	0	9	@All
\$ReplFormula	1	3	0	10	@All
\$ReplPrivateFol...	4	3	0	12	:::
\$ReplSrcServers	4	3	0	4	-; CN=Acme03/O=ACME; CN=Acme02/O=ACME; CN=Acme04/O=ACME
\$ReplVersion	1	3	0	3	2
\$ReplView	4	3	0	11	:::
\$TITLE	1	3	0	1	CN=Acme01/O=ACME
\$UpdatedBy	1	3	0	2	CN=John Admin/O=ACME

Selective Replication Formulas – Fun Facts

Do Replication Formulas always take affect?

- NO!
 - Cluster Replication will DISREGARD any replication formulas in place
 - It's #1 target is to ensure the DBs taking part in cluster replication are 100% the same
 - TIP: Don't use selective replication formulas for databases that are set to take part in cluster replication
 - Scheduled replication will use selective replication formulas
 - CLREPL will disregard them
 - Result: Confusion, increased processing time (+ numerous view index updates!)

Selective Replication Formulas – Fun Facts

How to deploy selective replication formula settings?

- You need to know about the structure of replication formula notes for this
 - Use the ReplicationInfo object from the LS NotesDatabase class (Datatype: NotesReplication) – this was introduced in R5
 - Use the “getEntry” method from the NotesReplication object to set values
- Example:
 - Script loops through all mailboxes and sets replication formulas to make server Acme01 only receive “Memo” type documents from server Acme02

Selective Replication Formulas – Fun Facts

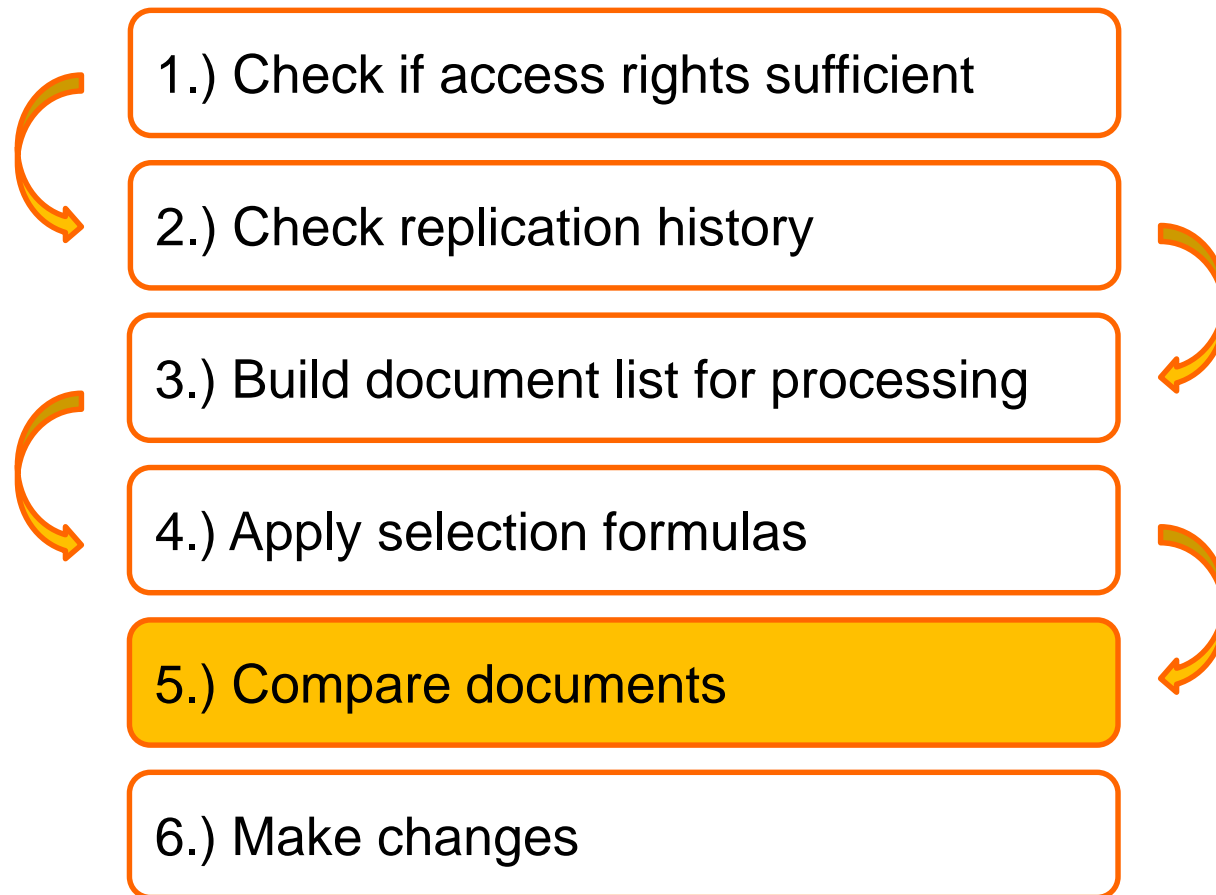
How to deploy Selective Replication formula settings?

```
repformula = "SELECT Form=" & Chr(34) + "Memo" & Chr(34)
svr1 = "Acme01/ACME"
svr2 = "Acme02/ACME"

Set sname1 = New NotesName(svr1)
Set sname2 = New NotesName(svr2)

Set dbdir = New NotesDbDirectory("Acme01/ACME")
Set db = dbdir.GetFirstDatabase(DATABASE)
While Not db Is Nothing
    'Skip databases which you don't have access to
    On Error 4060 GoTo Error4060
    'Check to see if this database is in the mail directory
    pos = InStr(db.FilePath, "mail")
    If pos = 1 Then
        Call db.Open(svr1, db.FilePath)
        'Get replication note, create if doesn't exist.
        Set repinfo = db.Replicationinfo
        Set repnote = repinfo.Getentry(sname2.Abbreviated, sname1.Abbreviated, True)
        repnote.Formula = repformula
        Call repnote.Save
    End If
GetNextDb:
    Set db = dbdir.GetNextDatabase()
Wend
Exit Sub
Error4060:
    'If the code reaches here then the user does not have access rights.
    Resume GetNextDb
```

What happens when two databases replicate?



Anatomy of Document & Item Properties

Document Identifiers & Properties

- Document Universal ID
 - 16-byte identifier that uniquely identifies a document across replicas
 - @DocumentUniqueID in formulas
 - NotesDocument.UniversalID in LS
- Note ID
 - Uniquely identifies documents in ONE notes database
 - @NoteID in formulas
 - NotesDocument.NoteID in LS

Anatomy of Document & Item Properties


Document Identifiers & Properties

- Sequence Number
 - Both documents and items have them
 - Increases every time a document is saved
 - Devs, careful with ctrl+s!
 - The doc.Sequence number = highest item's sequence number
- Ways to get:
 - \$Elements(\$Revisions)
 - Risky because the limit might have been hit, and in this case it's incorrect
 - NSFNoteGetInfo API Call: SQ number part of the returned header information
 - Ytria scanEZ

Anatomy of Document & Item Properties

Document Identifiers & Properties

- \$Revisions
 - Standard Document field that contains 8 byte entries for each time a doc has been saved.
 - CAUTION! There can be a DB level limit here.



The screenshot shows a configuration dialog box with three settings:

- Limit entries in \$UpdatedBy fields: 0
- Limit entries in \$Revisions fields: 0
- Allow Domino Data Service: Never

- \$Updatedby
 - Stores a list of editors (Canonical names = > 40 bytes). First entry is usually creator, but if DB limit hit, won't be accurate.
 - Computed when composed created by field is advised.
 - BEWARE! Not necessarily in sync with \$Revisions!
 - If there are multiple modifications by the same person after each other, it'll only record ONE entry.

Anatomy of Document & Item Properties

Document Identifiers & Properties

- \$UpdatedBy item: No duplicate entries when same person saves document multiple times in a row (space saving considerations)

\$Revisions	4	5	0	29
\$UpdatedBy	3	5	0	28

Note Info Readers/Authors Field Value

Type: Date list Name: \$Re

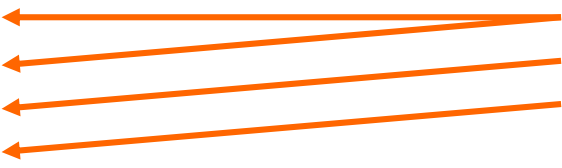
04/26/2013 12:52:40 PM
04/26/2013 12:53:06 PM
04/26/2013 12:53:58 PM
04/26/2013 12:54:37 PM

\$UpdatedBy	3	5	0	28
ab AltFrom	1	1	0	8

Note Info Readers/Authors Field Value

Type: Names Name: \$Up

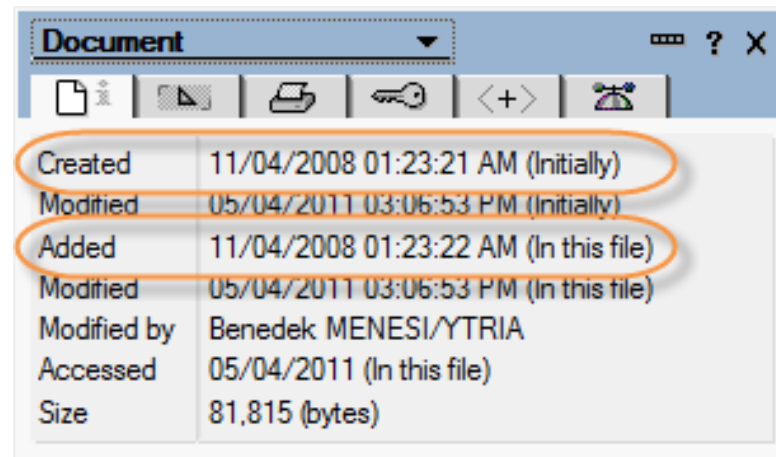
CN=John Admin/O=ACME
CN=Dorothee Parker/O=ACME
CN=Joe Gowlan/O=ACME



Anatomy of Document & Item Properties

Document Identifiers & Properties

- Creation date
 - @Created in formulas, NotesDocument.Created in LS
 - The initial creation date (more on this later)
- Created Initially & Added in this file
 - Sooo important! Two different date stamps
 - @Created in formulas and NotesDocument.Created
 - Returns initial creation timestamp
 - Added in this file
 - When the document was created in this file



The screenshot shows a 'Document' properties dialog box with a toolbar at the top containing icons for file operations. The main area displays a list of properties with their corresponding values. Two rows are circled in orange: 'Created' and 'Added'. The 'Created' row shows the date '11/04/2008 01:23:21 AM (Initially)'. The 'Added' row shows the date '11/04/2008 01:23:22 AM (In this file)'. Other properties include 'Modified' (05/04/2011 03:06:53 PM), 'Modified by' (Benedek MENESI/YTRIA), 'Accessed' (05/04/2011), and 'Size' (81,815 bytes).

Created	11/04/2008 01:23:21 AM (Initially)
Modified	05/04/2011 03:06:53 PM (Initially)
Added	11/04/2008 01:23:22 AM (In this file)
Modified	05/04/2011 03:06:53 PM (In this file)
Modified by	Benedek MENESI/YTRIA
Accessed	05/04/2011 (In this file)
Size	81,815 (bytes)

Anatomy of Document & Item Properties

Document Identifiers & Properties

- Difference between the Initial Creation & Added in this file = time it took to replicate document
 - This time date value can ONLY be retrieved using the AddedToThisFile API call, and not from Formulas

What's taken Into Consideration When Docs are Compared?

How are documents compared?

- If replication History is present
 - Documents are compared where
 - Last Modified Date was later than Last Replication history describing the last time the two DBs talked
- If replication History has been cleared or does not exist
 - All notes will be compared based on their UNIDs
 - Results in a longer replication time
 - 'Cleaner' replication
 - Very helpful for solving replication issues
 - Tip: use scanEZ's Replication Auditor to imitate replication
 - Will look behind the curtain (even if replication history present) and calculate what would happen if replication would happen with no replication History present

What's taken into consideration when docs. Compared?

How are documents compared?

- There are three major scenarios:

I. Note exists in both DBs. In this case, sequence number will be compared, and either a conflict is created or doc with lower sequence number is updated.

II. Note only exists in one of the DBs. Note will be created in the database where it is not present.

III. Note exists in both DBs, but in one of them it's a deletion stub. In this case, the sequence numbers are still compared, and whichever wins will be replicated.

- YES, this means that a frequently updated note can win replication over a stub with a lower sequence number!

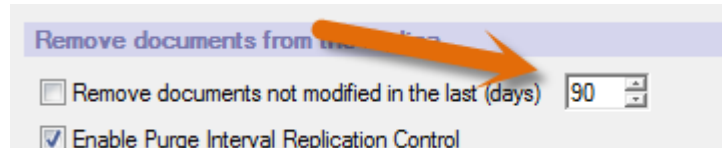
What are Deletion Stubs?

- Deletion Stub definition
 - When a document is deleted from the database, a light place holder is created that contains little information
 - In case enabled, Soft Deletions are an interim step in this process
 - Soft Deletions have a different class, and are only different from deletion stubs because they actually do contain all their items
- What do deletion stubs contain?
 - Document Replica ID
 - Creation Time (Initial)
 - Deletion Time (Both initial & in this file)
 - Sequence number

Deletion Stubs

What are Deletion Stubs?

- How long are they kept in the DB?
 - By default they will be deleted after 90 days
 - Can be changed on Space Savers tab of replication settings window. (Not very obvious, but field value takes affect even if removal is not checked)



- Note: Purging of deletion stubs interval will be the 1/3rd of this value!
 - In the default case this means it'll happen every 30 days, which will possibly leave you with deletion stubs that are kept in the database for <120 days!

Deletion Stubs

How to locate to them?

- NOT displayed in views (in fact, they aren't even documents – different class)
- TIP: When using GetdocumentbyUNID in LS, object returned by a Deletion Stub will raise an error when trying to work with it
 - doc is nothing is NOT enough protection here!
 - When getting document object, once “Is Nothing” check has been done, you should do an “ If TestDoc.Size = 0 Then” type check to handle any errors due to deletion stubs.
 - Favorite comment below an article regarding this:

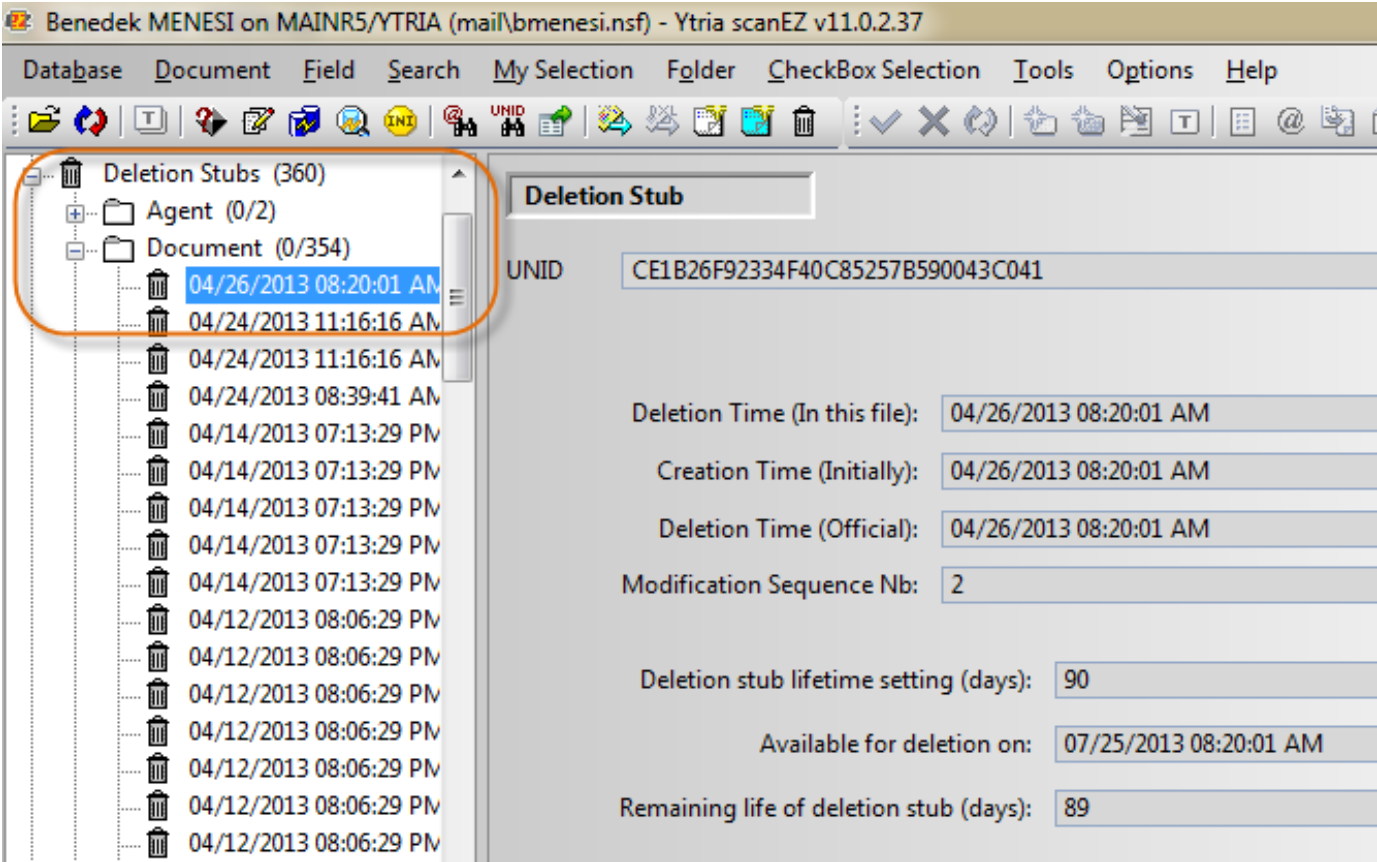
COMMENTS

1 - Holy crap. I have some code to go patch!

Deletion Stubs

How to locate to them?

- Use scanEZ (using the Lite version this is free!)



Deletion Stubs

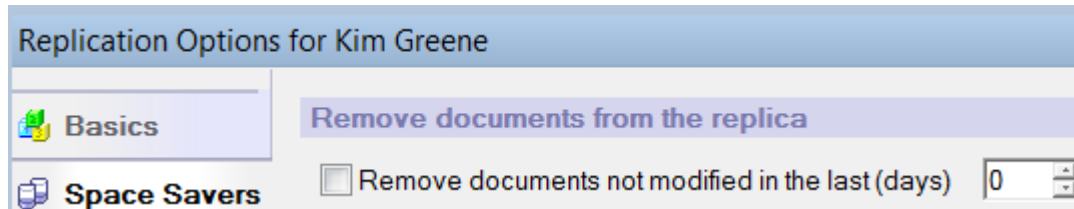
How to locate to them?

- How to get their sequence number?
 - Only way is to use the Notes API
 - Both NSFDbGetNoteInfo and NSFDbGetNoteInfoByUNID calls return the note header without attempting to open it, i.e. will let you gather the sequence number too

Deletion Stubs

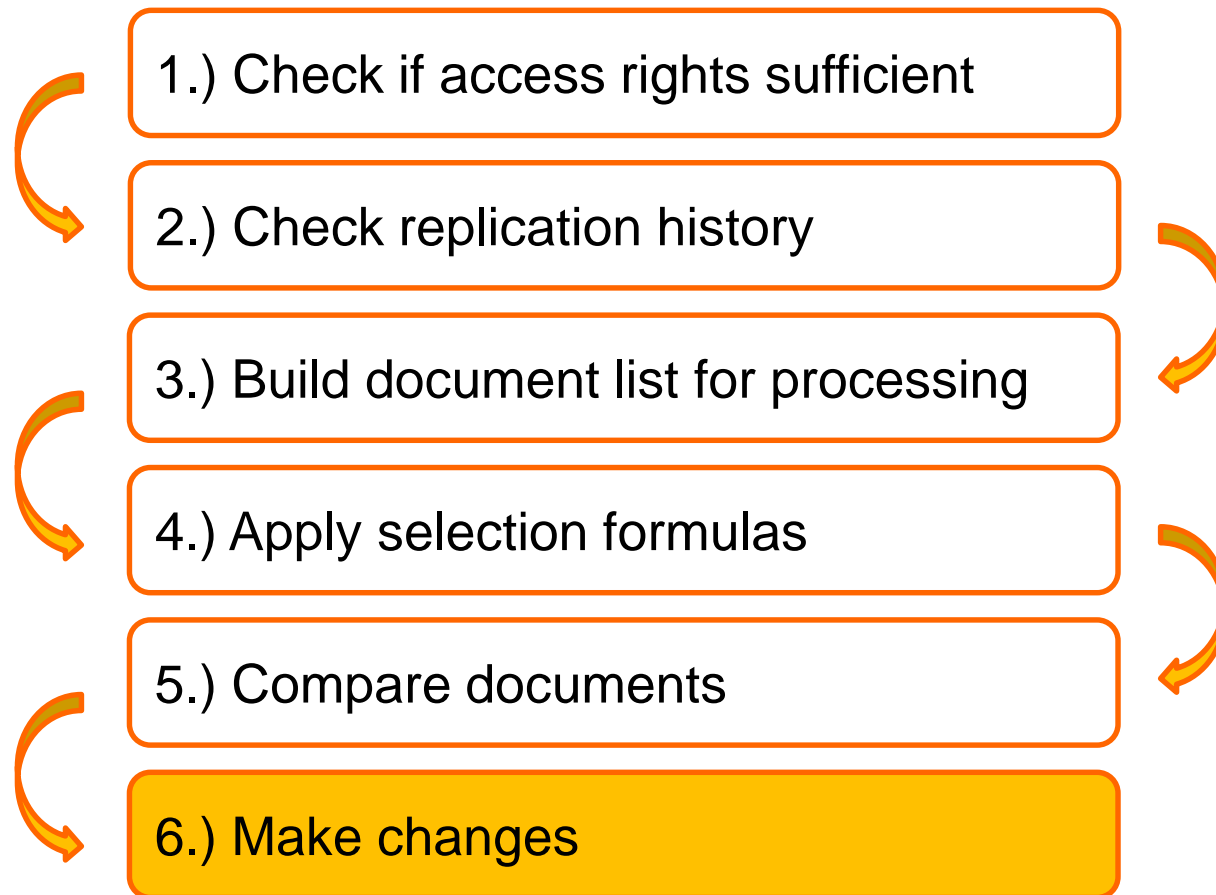
How to remove them?

- Reset Purge interval
 - Unless the customer has a tool to get rid of deletion stubs manually (like scanEZ), changing the database replication settings -> space savers, and setting the parameter 'Remove documents not modified in the last (days)' to 0, closing and reopening the database clears the deletion stubs



- The TRICK, is to NOT put a checkmark next to 'Remove documents modified in the last (days)' as that will delete documents from the database that were not intended to be deleted!!!
- Clearing deletion stubs programmatically ONLY works through the C API

What happens when two databases replicate?



Problems after replication

What can go wrong?

- Replication / Save conflict documents
 - We'll take a look at what exactly they are, and the best ways to prevent / resolve them
- Ghost Documents
 - Whatever that means: two usual scenarios, we'll discuss how to identify and deal with them

Replication or Save Conflicts

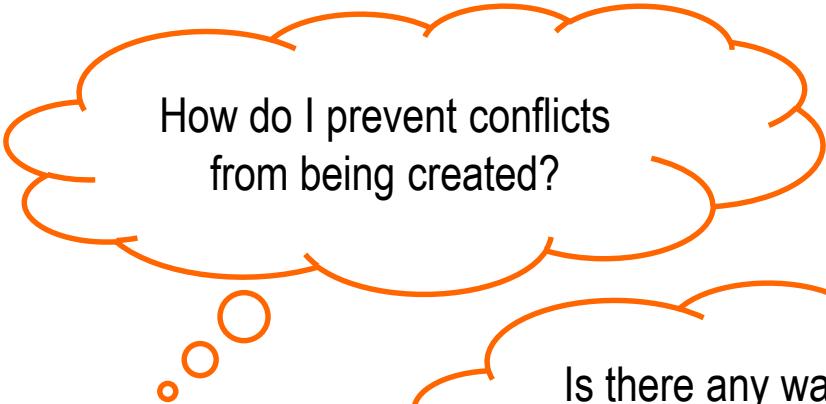
What are replication / save conflicts?

- Two documents are in conflict if they either have diverged at some point, and changes have been made past the point of divergence independently to multiple copies.
- Created when two or more users edit a document at the same time on the same / different servers.
- Replicator examines the two docs. By looking at the Originator ID (OID) and \$Revisions item. (OID contains the sequence number)
- What is the parent document?
 - Parent document = winner. No change on the backend items.
- What is/are the conflict document(s)?
 - Loser copies that are transformed into response documents. (\$Ref [notes.ref list type] item pointing to parent, plus presence of an empty \$Conflict [text type] item)
- Sooo Important! Conflicts are NOT the problem, in fact they are the solution!

Replication or Save Conflicts

What are replication / save conflicts?

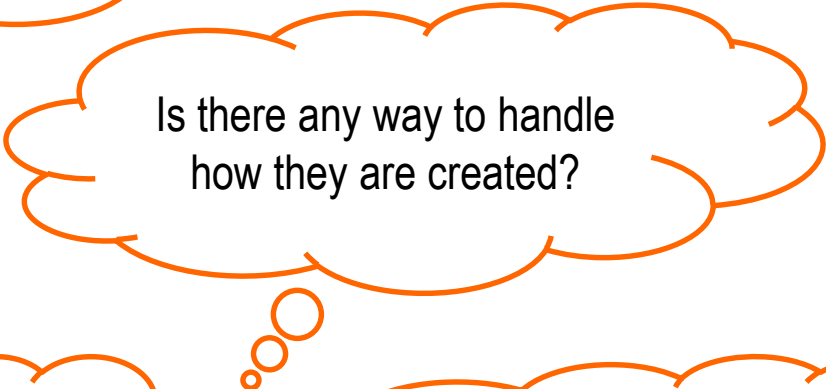
- We might be asking ourselves the following questions...




How do I prevent conflicts from being created?



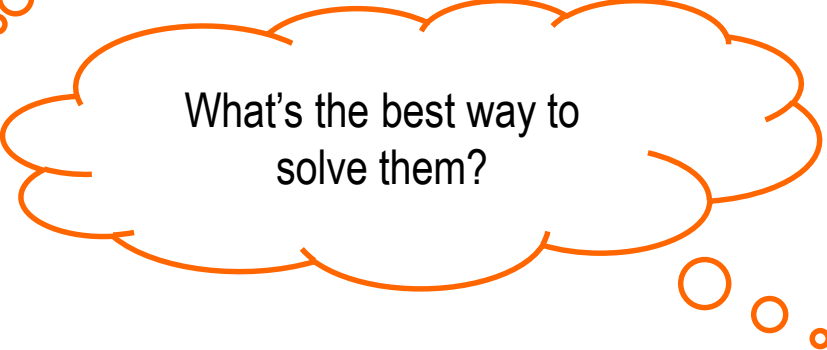
How do I find all conflicts?



Is there any way to handle how they are created?



Anything else to watch out for?



What's the best way to solve them?

Replication or Save Conflicts

Conflict Handling options

- Specified by the developer on the form properties, instructs notes to:
 - **Create Conflicts**
 - No \$ConflictAction item on the backend document.
 - If Replication / Save conflicts happen, winner turns into parent and loser will become conflict document.
 - **Merge Conflicts**
 - \$ConflictAction = "1" -> backend document item value
 - With this setting conflicts are only created in case the same item has been modified across the documents. (i.e. If item A and B modified on doc. 1 and item C and D modified on doc. 2, theoretically no conflict)
 - This usually FAILS: Last modifier updated in most cases, hence ALWAYS conflict.
 - **Merge / No Conflict**
 - \$ConflictAction = "3" -> backend document item value
 - Merge if different items modified.
 - If same item modified, silent deletion of conflict doc = SUICIDE!

Replication or Save Conflicts

Conflict Handling options

- **No Conflicts**
 - `$ConflictAction = "2"` -> backend document item value
 - Winner takes it all (Almost guaranteed data lose)

So what do we advise?

- **Create Conflicts**
 - Make sure you keep their volume as low as possible
 - To be discussed later
 - Ensure you can quickly find them
 - Implement a special folder in the DB
 - Ensure you solve them quickly and efficiently without losing data

Replication or Save Conflicts

How do I prevent them from being created?

- There are many-many different tactics and best practices
- You should know of a few general rules...
- ... And a few specifics about save or replication conflicts

- **Prevent Conflicts in general**

- Manage author items

- Dynamic author list

- Ensure only a limited subset of people are authorized to edit documents

Replication or Save Conflicts

How do I prevent them from being created?

▪ Prevent Save Conflicts

- Document locking
- Avoid the LS
`doc.computewithform()`
- Avoid `doc.save()` in
`querysave` and
`postsave` form events
without closing form

▪ Prevent Replication Conflicts

- Ensure scheduled agents don't run on both servers (Ytria `agentEZ` or `agent aggregator` thanks to Thomas Lindberg)
- Replicate more often, consider clustering
- Keep number of replicas to a minimum

Replication or Save Conflicts

Ok, so it happened. Now what? Let's find them!

- Displayed as [Replication or Save Conflict] in views
 - WATCH OUT! Some views use @IsUnavailable(\$Conflict) formulas!
 - e.g. Connection documents
 - If displayed in views: do you want your users to find (and edit) them?
- You might want to consider implementing a \$Conflict folder, something like what's described in [this](#) IBM Tech note
- Once found, how do we go about dealing with them?

Replication or Save Conflicts

What's the best way to solve conflicts?

- ~~It's easy! Delete conflict document!~~
- Seriously. Three scenarios (but you'll need to compare docs in any case)
- **Winner document contains all necessary information and loser is expendable.**
 - In this case it would be safe to just delete the Conflict document
- **Loser document contains all the necessary information**
 - This is where it gets a little tricky. You can't just delete parent, because the conflict doc will still have both the \$Ref & \$Conflict items, which will make it an orphan, so you'll have trouble finding it.
 - In this case, consider using the following script to turn loser into winner (thanks to breakingpar.com)
 - Essentially this means you get rid of the \$Conflict and \$Ref items before removing the winner (=parent)

Replication or Save Conflicts

Turn loser into winner

```
If winner Is Nothing Or loser Is Nothing Then Exit Sub
```

```
Set parent = Nothing
```

```
If winner.HasItem("$Ref") Then
```

```
    On Error Resume Next
```

```
    Set parent = db.GetDocumentByUNID(winner.ParentDocumentUNID)
```

```
    On Error Goto 0
```

```
    If Err <> 0 Then
```

```
        Err = 0
```

```
        Exit Sub
```

```
    End If
```

```
End If
```

```
Call loser.RemoveItem("$Conflict")
```

```
Call loser.RemoveItem("$Ref")
```

```
If Not parent Is Nothing Then Call loser.MakeResponse(parent)
```

```
On Error Resume Next
```

```
Call loser.Save(True, False)
```

```
Call winner.Remove(True)
```

```
On Error Goto 0
```

```
If Err <> 0 Then
```

```
    MsgBox "Error removing original winner: " & Error$, 16, "Error"
```

```
    Err = 0
```

```
End If
```

```
End Sub
```

What's the best way to solve conflicts?

- **Both winner and loser contain necessary information and need to be merged.**
 - Several solutions out there, but to be honest scanEZ is just the BEST way to go about it.
 - You have to compare all conflicts with the parent document (Yes, there might be more) and ideally – select the values you'd like to keep.

Replication or Save Conflicts

What's the best way to solve conflicts?

Conflicts Solver

Conflict Documents... 1. Document Show only fields with different values between do...

Name	2342 - Parent	2330 - Conflict
UNID	D36C57799457387A85257B330078A66B	4C3B5ED50DFA9AA585257B21006F1686
Modified By	Benedek MENESI/YTRIA	Benedek MENESI/YTRIA
Created Initially	03/19/2013 05:57:49 PM	03/01/2013 03:13:22 PM
Modified Initially	03/20/2013 07:03:41 AM	03/20/2013 07:04:28 AM
<input checked="" type="checkbox"/> \$RefOptions		<input checked="" type="checkbox"/> 1
<input checked="" type="checkbox"/> Body	<input checked="" type="checkbox"/> [rich text cannot be viewed in list] CRC: 7D765321	<input type="checkbox"/> [rich text cannot be viewed in list] CRC: 0405BA5
<input checked="" type="checkbox"/> Categories	<input type="checkbox"/> Sales	<input checked="" type="checkbox"/> Miscellaneous
<input checked="" type="checkbox"/> DocumentRead...	<input checked="" type="checkbox"/> CN=John Admin/O=ACME	<input type="checkbox"/>
<input checked="" type="checkbox"/> Scope	<input type="checkbox"/> Private	<input checked="" type="checkbox"/> Public
<input checked="" type="checkbox"/> Subject	<input type="checkbox"/> First Doc	<input checked="" type="checkbox"/> First Doc (revised)

Print... Export ... Switch select. Move all response(s) from Conflict(s) to the Pa... Solve Close

Replication or Save Conflicts

What to watch out for?

- DON'T let your users edit Conflict docs! They'll turn into duplicates which is definitely not something we want.
 - Protect your conflicts: Either hide them from the views, or use the QueryModeChange event.
 - Below an example of how to do this:

```
Sub QueryOpen(Source As Notesuidocument, Continue As Variant)
    Dim doc As NotesDocument
    Set doc=source.Document
    If doc.Hasitem("$Conflict") Then
        MsgBox "This is a conflict document. Editing
        conflicts is not permitted in this database", 16, Error
        continue=False
    End If
End Sub
```

Replication or Save Conflicts

What to watch out for?

- Don't let the view fool you!
 - 61/104 Views in the 8.5.3 NAB hide Conflicts!
 - Story: Customer having issues with replication, came down to conflicts in connection documents
 - Tip: There is a Conflicts folder

Ghost / Resurrected Documents

What are they?

- Two kinds of issues referred to as “GHOSTS”
 - Ghosts are not an “official” definition. Usually stands for two things
 - *Replication Ghosts*
 - Response Document with no parent (nor it’s parent’s stub) replicated to a new DB
 - Notes cannot handle the lack of parent Doc referenced by the \$Ref item located on the response document
 - Empty, ghost document created, which physically exists, but contains no items
 - Makes it complicated to find orphan responses, as these responses won’t be orphans per say
 - *Resurrected ghost documents*
 - Let’s take a look!

Ghost / Resurrected Documents

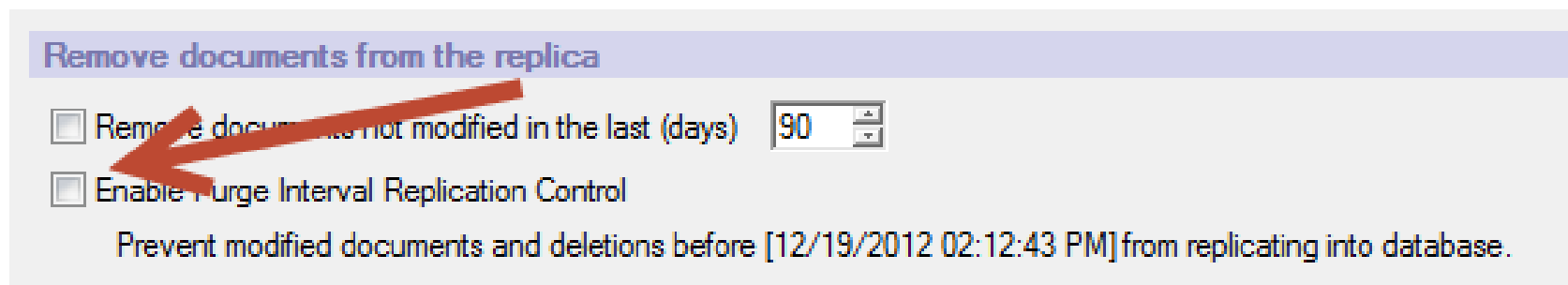
Resurrected Ghost document symptoms

- You leave the office with an internationally replicating database that has 60k documents
- You come to work in the morning, and all of a sudden, there are 75k documents
- What happened?
 - You have a replica somewhere (local OR server) which has not been replicating with your DB since more than the deletion stub lifetime setting
 - This means:
 - Documents have been deleted from the production version of the DB, but even their deletion stubs have been purged
 - These documents will be considered as NEW documents when replication happens

Ghost / Resurrected Documents

Preventive actions – PIRC

- Use the PIRC (Purge Interval Replication Control) feature
 - If enabled, replication runs a background check to only let documents in when the difference between their initial creation and the time of replication is less than the Deletion stub lifetime setting
 - New with 8.5.3
 - Gotta be 8.5.3 or later to work with this!



Remove documents from the replica

Remove documents not modified in the last (days)

Enable Purge Interval Replication Control

Prevent modified documents and deletions before [12/19/2012 02:12:43 PM] from replicating into database.

Ghost / Resurrected Documents

Preventive actions – PIRC

- Set notes.ini to show more information
 - `DEBUG_REPL_PIRC`
 - = 1 (for general info)
 - = 2 (for more details)
- Instruct the replicator server task to disregard PIRC
 - `Replicate –NOPIRC`
 - Will replicate the selected databases with no regards to the PIRC settings for the given session

Ghost / Resurrected Documents

What if it already happened?

- Use the Initial creation & created in this file dates
 - Like we said, the second one is quite tricky to get. If you implement the API Call, you could have an agent stamp all docs using a special field, so you can display the following in a view:
 - Initial creation
 - Creation in this file
 - Difference between these two dates
- ... And categorize by something like `@If(Createdinthisfile->@Created>Delstublifetime;"Problem";"OK")`
- Use scanEZ's Post Replication Auditor (instant identification)
- Or take a look at this openNTF project
 - <http://www.openntf.org/Projects/codebin/codebin.nsf/0/300F25985BCB5CA38625737900608E54>

Thank you

Thank you for attending!!

- Please fill out your evaluation forms
 - Your feedback is very important!
 - Feel free to get in touch with us:

Contact – Ben Menesi



ca.linkedin.com/in/benedekmenesi



@BenMenesi

Contact – Kim Greene



www.linkedin.com/in/kimgreeneconsulting



@iSeriesDomino