Analyses

Code analyses of existing notes applications

Code analyses regarding an UNIX server consolidation

Maßweiler, 20.07.2005
Author : Joachim Mutter

Preface

Inhaltsverzeichnis

Preface

1 Preface

The „MyCompany Deutschland GmbH & Co. KG" ordered an analyses of the applications on the following servers

and wants to scan them for the following criterias:

- are there any problems in the code of the applications regarding a later move to a UNIX server as hosting domino platform. If so, mark the applications with malicious code and point out the amount of time for the needed changes.

2 Working method of the Scanners[1]

The Analyses-Application „CodeGrabber" gets a list of servers, either form an users selection (has been used at "MyCompany") or out of a central Domino Names-And-Address book (NAB). On basis of this list, we scan all databases of the servers ordered be the appearance in the list, which **do not** match the applications "Exclude List" which could be found in the Technical Appendix
The so found notes applications will be opened and all Design Elements (Forms, Views, Libraries, Agents, etc) will be read in. We parse the code sections (in forms especially Actions, Hotspots, ...) for the given search patterns (Search Pattern for the UNIX Analyses), which depends on the used search mode (Cluster, UNIX, WEB, User), and identifies the elements with malicious code.
On the basis of found code segments and the kind of these elements, we classify the database regarding the "Decision Matrix Rules". The exact meaning of this classification will be described in more detail in chapter Explanation „CodeGrabber".
Every scanned database has its representation in a Notes Document within the CodeGrabber application. All found informations are presented in an easy readable way. Additionally to the classification, we put in every malicious code sequence with the information in which Design element we found it and in which function it's used (LotusScript). We lead a list of all different Design elements (all Fields, Actions, Links, Hotspots, etc.) which could be used later on to make further investigations and assumptions.

[1] The general fucntionallity of „CodeGrabber" you will find in chapter Technical Appendix

3 Criterias for the Analyses

During this scan we only want to have a statement regarding the ability to move all notes applications to a domino server hosted by a UNIX machine. The following describes the expected result set for the scan.

3.1 Analyses for UNIX Executability [2]:

1. List of all Applications, which will be hosted by the Servers defined in Chapter 1
2. Categorize of the Applications by used Design Templates
3. List of all used external Interfaces, as far as they will be bind in as Lotus Script Extension Module (LSX). Among them there are ODBC Access, NotesPump, LEI, SAP, etc.
4. List of all Applications, which use malicious LotusScript or Formula Code, which leads to errors during execution on an UNIX Platform.[3]

Really relevant in case of a UNIX analysis are applications with **Classification F.**
Only such applications are considered in the temporal view, which represents the approximate time for need adoptions. These time values may only be considered as reference points, not as absolute values, because we could only rate the Syntax during a programmatically scan, not the semantics.
This is especially hard to count, if you need to switch over to another technology in case of ODBC usage, which had to be adopted to LC

3.1.1 Data-Extracts

We made an Extract of all Applications, which are based on Templates and we made an Extract of all Applications, which had Replicas on different servers. These tables cold be found in Chapter C Replica Listing sorted by server: and Template Extraction. For the template extraction, we differentiate for the following rules :

- Applications, where we had real existing Templates
- Applications, which are based on the same Templates but we couldn't find the real Template Databases during the scan

3.1.2 Determination of all keys indicators (Server based Code Execution)

The result is a list of applications rated by the expected amount of redesign time. Really relevant are only applications classified as "ClassF", because only in such applications we found critical code in server based code execution units (normally scheduled/triggered or Web agents) and only these applications will be considered in the appropriate time views.
The malicious design elements of such applications will be rated with time factors (minutes) and summarized in the depending (UNIX) time views. Thus if a server agent uses 2 script libraries with critical code in it, we compute the time factor for libraries multiplied by 2 (two libraries) and add the time factor for the agent itself. Because of this proceeding, we could not consider the computed values as absolute values, more as a hint. Keep in mind, that we only see, that a library with critical code is used by the agent, if the code is really executed by the agent is not part of the scan!
But investigations of existing applications with malicious code and the adoption process brought us to a rating system, which is quite close to the real world.
However, there could be quite large differences in the computed values and the really needed times of a single application. But this relate itself because of statistic criterions by the quantity of scanned databases.

You must assume a factor of +- 20 percent of insecurity!

[2] The list for the search patterns you will find in chapter Search Pattern for the UNIX Analyses
[3] The list of applications which need adoption you will find in chapter Unix Applikationsliste

This insecurity is system based, because we could only rate the syntax not the semantic of the code. Besides, from the expenditure point of view there is probably a great difference, if you have to port 10 function calls to another LSX (I.e. ODBC to LCLSX) or if you only must exchange a windows LSX into a UNIX based LSX.

3.1.2.1 Restrictions

If we find applications, which uses special LSX extensions (i.e. ODBC), we must investigate if it also exist for the UNIX system (Solaris, Linux), where the applications should be moved too.

During the categorize process for design templates, there could be various groups, which are obviously based o a template, but we do not found the physical template database on the scanned servers. We actually assume, that we could create a design template out of one application from one group and use this for the adoption of the code. Afterwards all other applications of the group inherits the design from the new template. In detail this must be clarified by an experienced notes programmer, who is able to figure out, if all design of the applications in one group is the same.

4 Execution

The analyses tools „CodeGrabber“, which is able to do the above described investigations automatically, was installed and configured for the scan of the servers defined in chapter 1 Preface at 07. July 2005 in the office of “MyCompany” in Darmstadt. The scan was triggered on the same day and takes until 13. July 2005. The so determined data was sent to us on 14.July 2005.

The raw data (and logs) were investigated in detail for any abusive hints and searched for LSX and API usage. Afterwards we reduced the data to used templates (see Restrictions), so that during the time accumulation only one application (the Template) is considered.

- Raw
  Here we do not make any assumptions about inheritance. Every database was counted separately.

- Best-case
  Here we assume, that equal named templates, independent of the server where we found them, are also equal in terms of the used code and all inherited databases have also the same code. This means, that all databases of this group depends of only one design, the one of the template (physical found or not)

- Worst-Case
  Aggregation of applications with the same template only for one server. If there databases on other servers, they will be counted separately.

Keep in mind, that the best-case analyses must be examined by an experienced notes programmer, because you cannot necessarily assume that the Templates are regularly replicated, and/or if they are really inherited from the template!

5 Results

List of the single analyses, sorted by redesign time and server. Here we see, among other data, the servers with the highest amount of redesign time and the distribution of the several malicious applications to the servers regarding to the redesign time.

5.1 Unix Scan

5.1.1 Numbers

These are the total numbers of the scan.

	Count	Percentage
Total amount of databases[4]	2100	
Insufficient access	252[5]	12%
Scanned Databases	1848	88%
Hidden Design	86[6]	4,1%
ClassF Databases	59	2,8%

Additionally we found, that there exist 3556 replicas for the given servers.

[4] Here we donot count the replicas. This number represents the amount of unique databases
[5] The not scanned databases could detoriate the result, according to the amount of critical code which may exist! Obviously most of them are Virusscanner and Helpdesk applications, Finance stuff etc.
[6] Same like footnote 4

Results

5.1.2 Result list by server (RAW, Templates and Replicas are unconsidered)

Here we do not make any assumptions about inheritance. Every database was counted separately.
The results are not really surprising. In most of the cases, there are not really big amounts of databases which has to be changed because of UNIX critical code. However, the total number of approx.. 20 days is to be regarded only as rough estimated value, as mentioned under chapter 3.1.2 [Determination of all keys indicators (Server based Code Execution)].

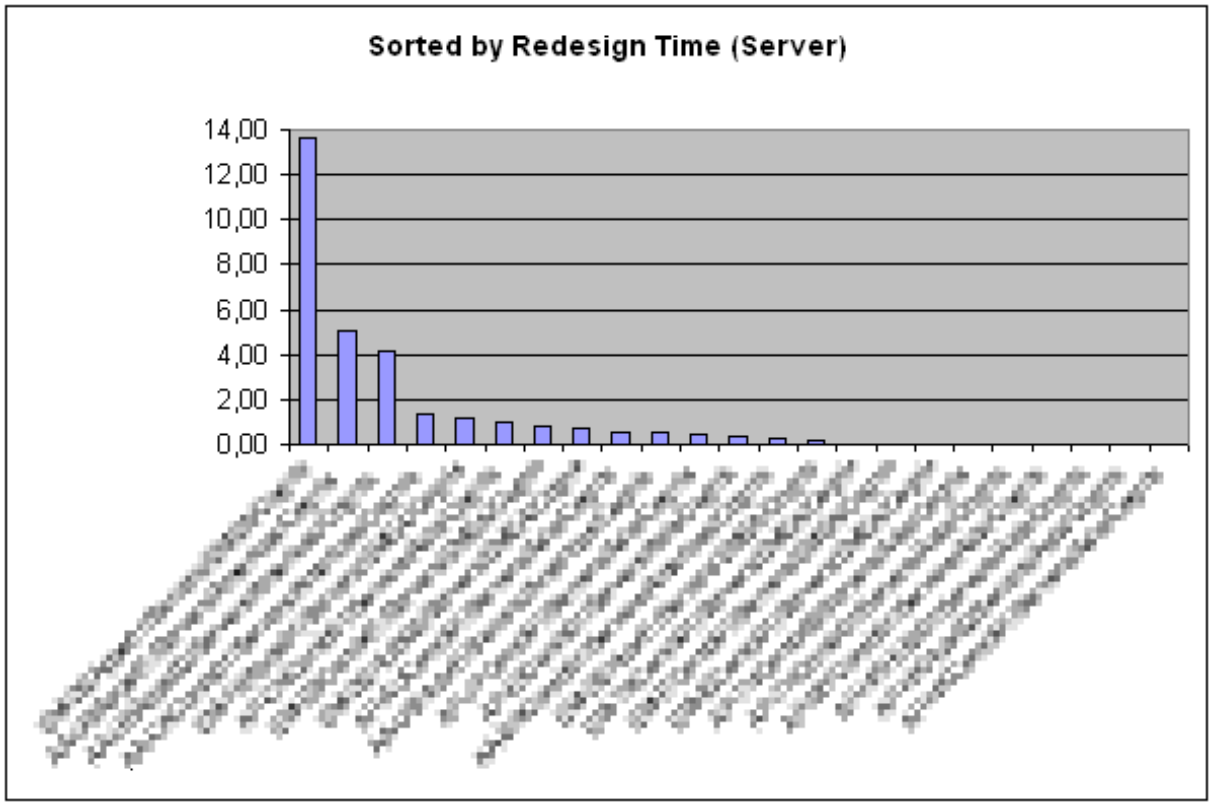
On the other hand, we see, that only 8% of all applications use critical code, which had to be investigated in more details and may be adopted.

5.1.3 Ergebnisliste nach Server (Templates und Repliken unberücksichtigt)

Server	Time	Views	Agent	Forms	Web Agents	Amgr Agents	Critical Code	Critical Code ClassF	Count
[illegible]	13,63	16973	6720	10785	101	445	609	72	420
[illegible]	5,06	8057	3831	6544	146	294	320	18	246
[illegible]	4,19	2427	869	1321	6	87	90	18	40
[illegible]	1,31	2153	960	2231	51	181	64	6	126
[illegible]	1,19	9501	2841	5016	44	587	200	5	343
[illegible]	1,00	763	298	589	6	19	37	5	34
[illegible]	0,81	1932	264	527	0	14	12	3	80
[illegible]	0,69	489	319	563	7	23	27	1	20
[illegible]	0,50	918	702	1255	29	28	239	2	34
[illegible]	0,50	1075	598	862	9	28	127	2	26
[illegible]	0,44	1607	721	1382	27	121	22	2	101
[illegible]	0,38	699	270	707	6	29	17	2	48
[illegible]	0,25	762	307	661	10	50	43	1	45
[illegible]	0,19	1083	519	1576	22	30	64	1	23
[illegible]	0,00	1755	491	1283	5	76	75	0	111
[illegible]	0,00	311	80	189	5	8	1	0	6
[illegible]	0,00	2028	451	1079	6	82	19	0	29
[illegible]	0,00	308	26	160	0	2	2	0	22
[illegible]	0,00	800	95	215	2	15	4	0	37
[illegible]	0,00	716	91	359	0	5	3	0	25
[illegible]	0,00	79	37	79	2	3	2	0	6
[illegible]	0,00	15	8	12	1	2	1	0	1
[illegible]	0,00	391	216	507	16	18	6	0	25
	30,14	54842	20714	37902	501	2147	1984	138	1848

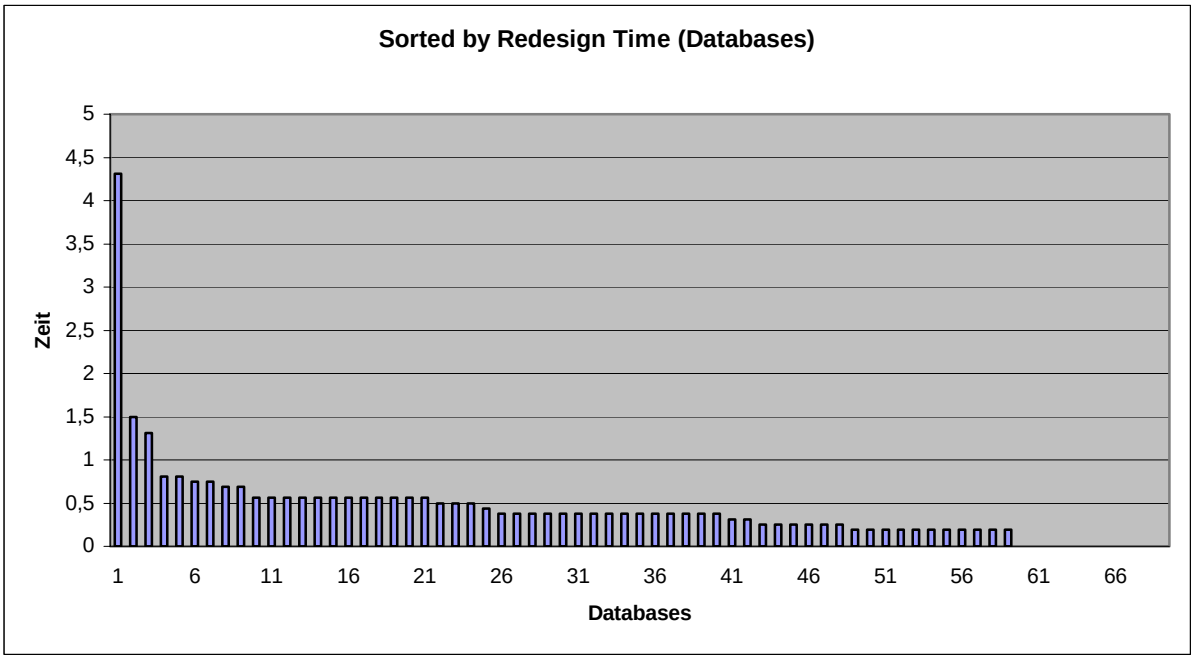
Accumulated redesign time for 1848 applications, which had to be moved to one or more UNIX servers is: **30,14 days .**

5.1.3.1 Graphic (Redesign Time by server)



Keep in mind, that you have to take a look onto the replica list, to relativate the amount of malicious databases per server. The amount of time will be added to the server, where a replica is first scanned!

5.1.3.2 Graphic (Redesign Time by databases)



You see here, that most of the database needed less than a half day for adoption. Only a few need more time!

Results

5.1.4 Result list by server (Inheritance Worst-Case)

Here we assume, that equal named templates, independent of the server where we found them, are also equal in terms of the used code and all inherited databases have also the same code. This means, that all databases of this group depends of only one design, the one of the template (physical found or not).

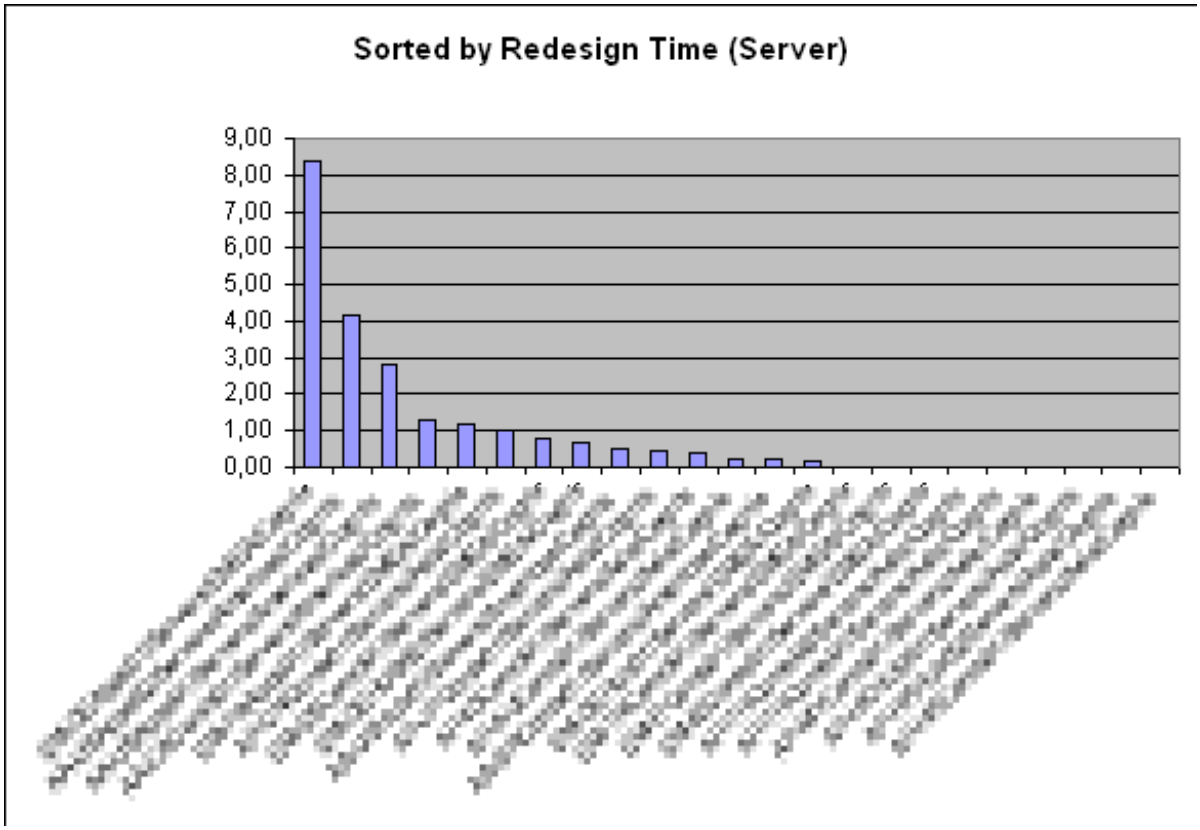
You see here, that the amount of time will be reduced by more than **25%!**

Server	Time	Views	Agent	Forms	Web Agents	Amgr Agents	Critical Code	Critical Code ClassF	Count
[illegible]	8,38	12136	4992	8364	62	336	415	53	355
[illegible]	4,19	2427	869	1321	6	87	90	18	40
[illegible]	2,81	6634	2566	3782	26	186	195	12	207
[illegible]	1,31	1969	712	1621	26	145	47	6	114
[illegible]	1,19	6302	1729	3158	31	244	150	5	222
[illegible]	1,00	763	298	589	6	19	37	5	34
[illegible]	0,81	1886	256	509	0	14	12	3	78
[illegible]	0,69	489	319	563	7	23	27	1	20
[illegible]	0,50	863	436	673	9	23	73	2	22
[illegible]	0,44	1476	652	1058	27	83	22	2	72
[illegible]	0,38	692	263	695	6	26	17	2	47
[illegible]	0,25	605	429	737	16	19	128	1	26
[illegible]	0,25	759	307	660	10	50	42	1	44
[illegible]	0,19	453	281	711	13	16	40	1	13
[illegible]	0,00	1558	482	1221	5	70	73	0	107
[illegible]	0,00	311	80	189	5	8	1	0	6
[illegible]	0,00	2013	448	1067	6	82	19	0	28
[illegible]	0,00	308	26	160	0	2	2	0	22
[illegible]	0,00	800	95	215	2	15	4	0	37
[illegible]	0,00	716	91	359	0	5	3	0	25
[illegible]	0,00	79	37	79	2	3	2	0	6
[illegible]	0,00	15	8	12	1	2	1	0	1
[illegible]	0,00	391	216	507	16	18	6	0	25
	22,39	43645	15592	28250	282	1476	1406	112	1551

Accumulated redesign time for 1551 applications, which had to be moved to one or more UNIX servers is: **22,39 days .**

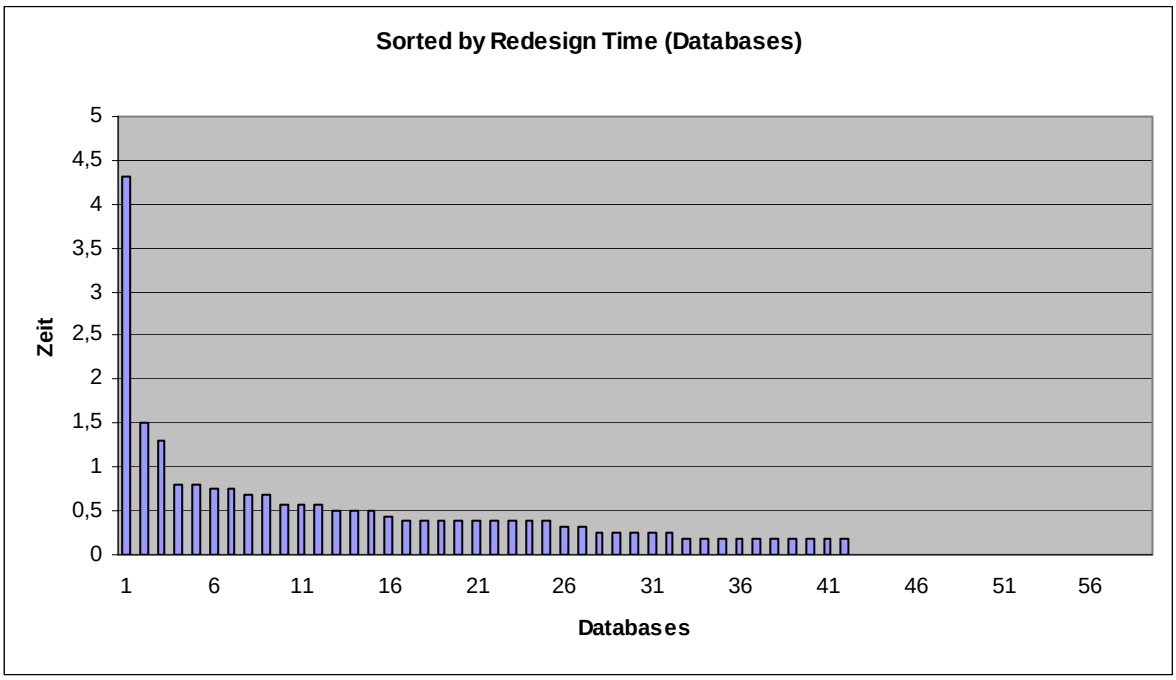
Results

5.1.4.1 Graphic (Redesign Time by server)



Keep in mind, that you have to take a look onto the replica list, to relativate the amount of malicious databases per server. The amount of time will be added to the server, where a replica is first scanned!

5.1.4.2 Graphic (Redesign Time by databases)



Results

5.1.5 Result list by server (Inheritance Best-Case)

Here we assume, that equal named templates, independent of the server where we found them, are also equal in terms of the used code and all inherited databases have also the same code. This means, that all databases of this group depends of only one design, the one of the template (physical found or not).

You see here, that the amount of time will be reduced additional by more than **20%.**

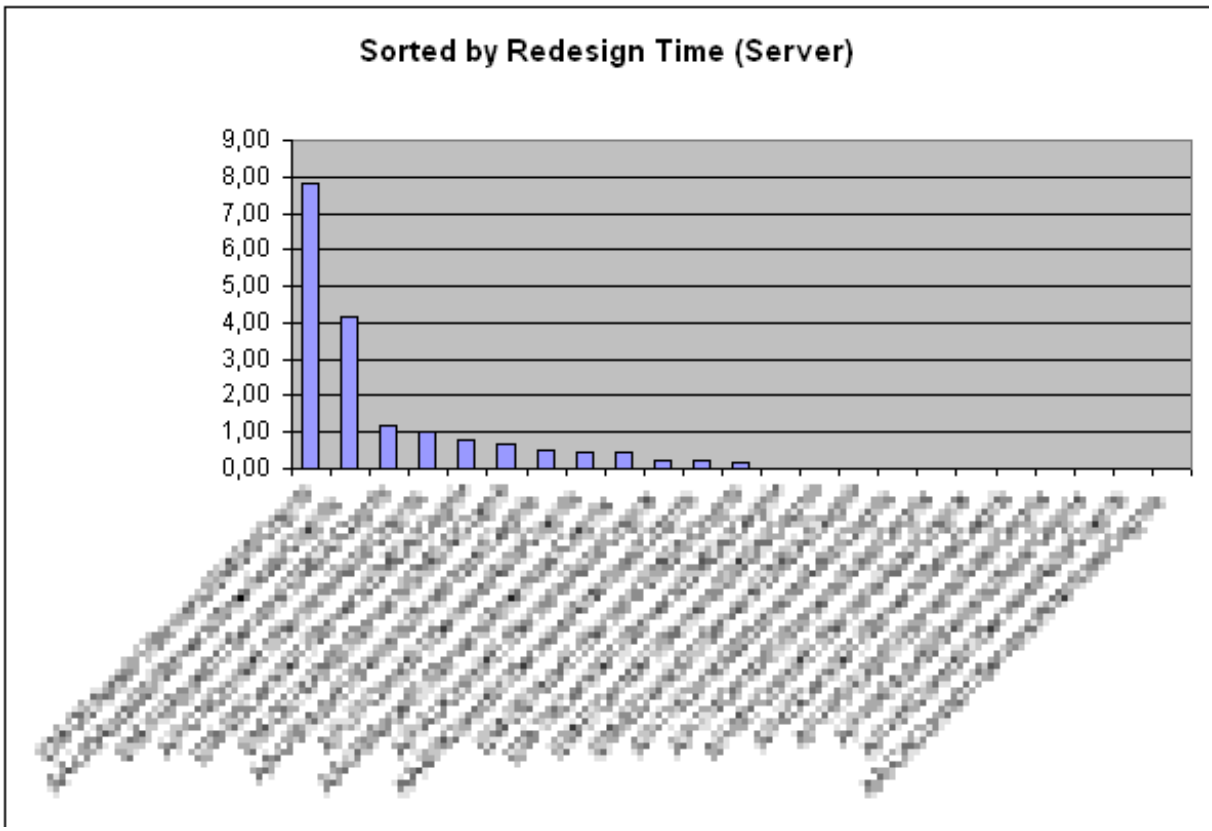
This mean an overall reduction by nearly 50% if we consider design-inheritance!

Server	Time	Views	Agent	Forms	Web Agents	Amgr Agents	Critical Code	Critical Code ClassF	Count
[illegible]	7,81	11319	4708	7922	55	315	379	51	337
[illegible]	4,19	2427	869	1321	6	87	90	18	40
[illegible]	1,19	5727	1522	2804	21	232	130	5	213
[illegible]	1,00	747	298	576	6	19	36	5	32
[illegible]	0,81	1883	256	508	0	14	11	3	77
[illegible]	0,69	430	229	459	4	21	12	1	16
[illegible]	0,50	694	391	458	9	19	70	2	18
[illegible]	0,44	1366	549	864	17	71	17	2	65
[illegible]	0,44	5176	1956	2870	19	123	145	2	179
[illegible]	0,25	595	425	721	16	17	126	1	24
[illegible]	0,25	759	307	660	10	50	42	1	44
[illegible]	0,19	267	222	522	13	15	31	1	7
[illegible]	0,00	1506	461	1128	5	65	70	0	102
[illegible]	0,00	280	42	151	0	5	0	0	4
[illegible]	0,00	2005	440	1046	6	81	18	0	26
[illegible]	0,00	240	15	88	0	1	0	0	20
[illegible]	0,00	797	95	214	2	15	3	0	36
[illegible]	0,00	271	70	202	0	4	1	0	22
[illegible]	0,00	474	207	541	6	20	9	0	43
[illegible]	0,00	79	37	79	2	3	2	0	6
[illegible]	0,00	15	8	12	1	2	1	0	1
[illegible]	0,00	336	163	433	9	13	5	0	21
[illegible]	0,00	1809	666	1482	26	131	41	0	108
	17,76	39202	13936	25061	233	1323	1239	92	1441

Accumulated redesign time for 1441 applications, which had to be moved to one or more UNIX servers is: **17,76 days .**

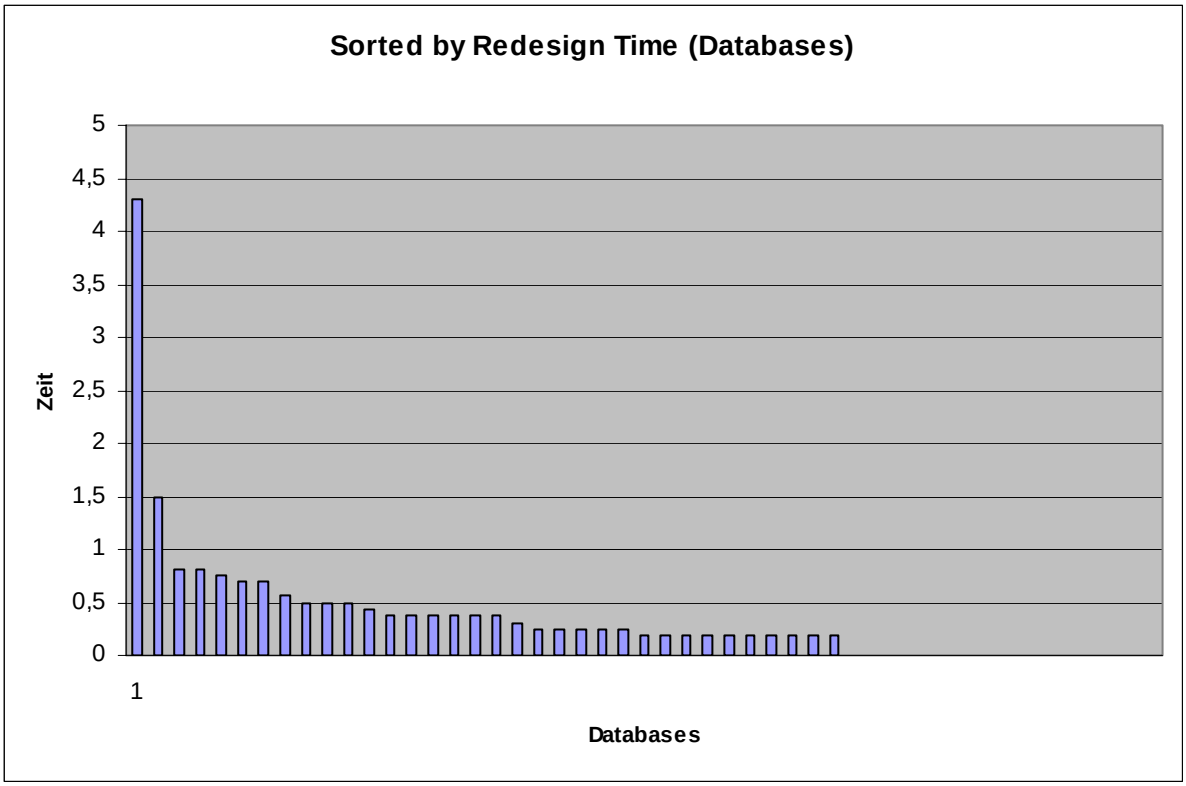
Results

5.1.5.1 Graphic (Redesign Time by server)



Keep in mind, that you have to take a look onto the replica list, to relativate the amount of malicious databases per server. The amount of time will be added to the server, where a replica is first scanned!

5.1.5.2 Graphic (Redesign Time by databases)



6 Applications which use LSX Extensions

The following applications use LSX Extensions, which are not part of the standard notes delivery. It must be checked very careful, if these extensions are available for UNIX:

As far as I can say, the lsxoracl.dll and *.rfc could be replaced by appropriate LCLSX technology, which is available for all UNIX platforms. But this had to be recoded, these LSX libraries could not be simply replaced by a fitting UNIX library!

7 Final comment

The complete data

- This document
- The reduced and accumulated database

will be sent to you in an electronic form (PDF) . The CodeGrabber database itself could be used as an data store after the scan and all evaluation and working functions are also usable. Only the scan functionality is blocked concerning the requested time license. So you have the possibility, after an intensive examination of the data, to remove additional applications from the class F state to have only enterprise critical databases left!
On the basis of these additional reductions you will get more extract data for the further decisions making process.

Additionally you can use this application to accompany the migrate process. There are actions and comment fields, which makes it possible to use the scanned data documents as a small documentation and workflow tool. So you have all relevant data for the migration process and could use the same document for documentation of the work, which had been done. After finishing the redesign work, you could switch over the document to an “processed state”.

Person to turn to

Joachim Mutter jmutter@sysarc.de 0172/7 38 35 25

Please contact us for any further questions regarding this analyses.

Best regards

Joachim Mutter

8 Technical Appendix

A) Expanded Exclude list

With these patterns we reduce the databases to be scanned and prevent the unnecessary scan of standard databases during the initial scan.

The red ones are explicitly defined by “MyCompany”

- backup*
- holistic-net*
- intelliwatch*
- help*
- iNotes*
- admin4.*
- alog*.*
- archlg50.*
- billing.*
- bookmark.*
- busytime.*
- cache.*
- catalog.*
- cca50.*
- certlog.*
- certsrv.*
- cldbdir.*
- clusta4.*
- csrv50.*
- da50.*
- dba4.*
- dblib4.*
- dblist.*
- decomsrv.*
- decsadm.*
- dircat5.*
- discsw50.*
- doclbs5.*
- doclbw50.*
- doclib4.*
- doladmin.*
- doldiscsw50.*
- dolres.*
- domadmin.*
- domcfg5.*
- domlog.*
- dsgnsyn.*
- events*.*
- group4.*
- headline.*
- help*.nsf
- help*
- inotes5.*
- journal5.*
- libact.*
- log.*
- loga4.*
- mail.box
- mail46.*
- mail50*.*
- Mail*
- mail*
- mailbox.*
- maillist.*
- setupweb.*
- siregw50.*
- siteact.*
- srchsite.*
- statmail.*
- statrep.*
- statrep5.*
- teamrm50.*
- webadmin.*
- mailtracker.*
- mtatbls.*
- *mtstore.*
- name*.*
- nntpcl5.*
- nntpdi50.*
- nntppost.*
- ntsync45.*
- pernames.*
- perweb50.*
- pubnames.*
- reports.*
- resrc??.*
- schema50.*

Technical Appendix

B) Search Pattern for the UNIX Analyses

- *Open * As *
- Close
- Unlock *
- Lock *
- Reset
- *=*FreeFile *
- *dir(*)
- *dir$(*)
- *ChDrive *
- *CurDir("*")<>*CurDir("")
- *CurDir$("*")<>*CurDir$("")
- *getFileDate(*)*
- *setFileAttr *,*
- *getFileAttr(*)*
- *"*:*"
- *"[A-Za-z]:*"
- date=*
- time=*
- * date=*
- * time=*
- Declare * Lib *
- *("ODBC"*
- *LCConnection(*)*
- *.EmbedObject(EMBED_OBJECT*
- *.EmbedObject(1452*
- *.EmbedObject(1453*
- *@DDE*
- *.verbs*
- *.doverbs(*)*
- *.activate(*)*
- *ActivateApp *
- *CreateObject(*)*
- *GetObject(*)*
- *isObject(*)*
- *isUnknown(*)*
- *.runonserver*
- *sendkeys *
- *%INCLUDE*<>*lsconst.lss*><*lsprcval.lss*><*lsxuierr.lss*><*lserr.lss*><*lsxbeerr.lss*><*orgconst.lss*

Technical Appendix

C) Explanation „CodeGrabber“

This database is designed to investigate databases filtered by self-defined criteria, so that a classification of the database script and code can be made (for a description of the different classifications see below). With the help of this classification a designer has got the possibility to see if the code has to be reviewed and adapted to ensure the correct execution of the application in a cluster or on a UNIX based system.

Explication

A list of all databases on the selected servers will be created according to the search criteria defined in the setup profile. These databases are scanned by the CodeGrabber. It searches for following features and characteristics: all scheduled and new mail agents, forms which contains readers and authors fields, file or OLE/COM operations and generally for critical code which is not working in a failover action. After the scan a document is created, which contains the name of the database, the number of replicas (if any exist on other servers), the used template and some other useful general information. If corresponding design information is found, the database is classified into one of the following classes:

Class A	**no special actions are necessary**
Class B	**author fields and/or**
Class C	**reader fields and/or**
Class D	**critical code and/or**
Class E	**scheduled agents (or matched rule in "User defined search" and enabled ACL lookup)**
Class F	**critical code in scheduled agents in a UNIX search or "User defined Search" and matched Rule in selected design elements**
Class X	**databases, which have got a hidden design. Here an analyses cannot be made, because the source code cannot be read This is also true for Script-Libraries, where the Code item is missing!**

In addition to this class the information which design elements contain the appropriate characteristics is shown in the document.

Criteria for the Classification

This section describes the problems regarding application development and redesign, which can occur at a migration. Procedures are shown, which are necessary to

- ensure the correct execution of the application on a clustered domino server
- migrate applications from R4.x to R5.x
- ensure the correct execution of the application for a server consolidation from one platform (NT) to another (UNIX)

In order to do this, the databases are bound to a classification, so that the staff responsible for the analyses is able to develop a standard procedure for the migration/consolidation. This is essential, if a large amount of data is affected.

This document is designed as a first analyses for a consolidation of the applications for clustering and for the server consolidation, respectively. Classifications of domino applications are shown, which are necessary for a consolidation of these applications on a domino cluster. This classification organizes the applications into a small number of classes according to different specifications, so strategies can be developed, with which a class can be made cluster-compatible or consolidated.

Technical Appendix

A short description what happens inside a cluster:

All clustered databases, reside on the servers of the clusters. If a user works on the cluster server A and edits a document in a database X, the changes are automatically replicated to the database X on cluster server B. If cluster server A crashes or is not available any more, the user will be relocated automatically to cluster server B and can continue his work without any loss of data. high availability and load balancing are further concepts regarding this context.

Notice

Following points have to be taken into account for a consolidation of applications:

- Readers Fields (cluster servers in field)
- ACL changes for groups
- Execution of scheduled agents (polling of availability?)
- Accesses to the operating system (existing directory structure? temporary files?)
- Workflow applications (storage of documents if the server is not available? before a mail is sent?)
- Accesses to other databases (views, etc., ("Open with failover in Lotus Script"))

Explanations

1. **Databases with Script Functionality**
   In general application developers use the command NotesDatabase. Open(Server, Databasename). This is normally working without any problem. But in the case of an error at the server A inside a cluster the request is forwarded to server B. Then this function is causing an error, because the database on server A cannot be opened. An alternative Lotus Script command exists, NotesDatabase.
   OpenWithFailover(Server,Database name). This command does not cause an error, but tries to find a replica on a different available server. Only if all severs inside a cluster are not available any more, an error is created. Therefore the commands NotesDatabase.Open() have to be exchanged with NotesDatabase.OpenWith Failover() in all clustered applications. A second method to open a database is the New() method of the NotesDatabase class. The same problem as above exists here. These code fragments have to be exchanged with NotesDatabase.OpenWithFailover().

2. **Scheduled Agents**
   It might be the case that applications detach files on the server, e.g. for further processing with third-party tools or cyclic editing of documents. In order to keep such an application executable in the case of a failover, the process has to be executed on all servers. This might cause multiple editing of the same source data, which might not be the same because of the asynchrony of the processes (for example, documents are changed or deleted during the process). This creates replication conflicts inside the cluster and causes a violation of the data integrity. As there is nearly no possibility (for example, only with a code which uses polling, etc, which would be difficult to develop) to prevent the execution of the agent, if another server already has executed the agent, the limits are reached soon.

   Independent of all other problems first it must be ensured that the agents are triggered on another cluster server. To achieve this the wildcard "*" has to be entered in the "server"-field in the agent options. NewMail or NewDocument agents, which do not have this property, should run without any changes, but it must be taken into account that the justification to run this agent is determined by the domino directory. This is controlled by the home server, found in the currently used location document.

The source data might cause more problems. If they are taken from the documents in the concerning databases, this is not a problem because of the direct replication inside the cluster - the databases are always identical, nearly simultaneously. If data from other files or non-notes-applications are used, it must be ensured that the data are available on all servers inside the cluster.

3. **Applications with Setup and Profile Documents**
   Problems result from applications, where servers and paths of clustered databases are referenced in profile and setup documents and if applications are executed with @commands. According to the Lotus Notes Knowledge Base the failover process is not able to process @commands in version 4.6x. To determine the behaviour in version 5.x tests must be made. If this problem then continues to exist, the server name from the profile document must not be used. Instead of this the name of the current server must be determined by the @DBName command. The path of the database can be taken from the profile document.

4. **Applications which contain Authors and Readers Fields**
   It must be ensured that in all documents of the cluster servers these fields are included. Otherwise the replication of these documents would not function properly. The server's replicator does not realize documents if it is not entered in the authors and readers fields.

5. **Applications with hidden Design**
   One has to take into account what happens with applications with hidden design (third party tools, commercial applications, etc.), where the information which database has to be opened, is received by setup documents. If correct commands are not used in this case, there is not yet a strategy for a solution at the moment.

Conclusion

- **Applications which work in conjunction with a setup document/profile, must be reviewed concerning the problems mentioned in matter 4. This means a complete review of all functions!**

- **Applications which contain server agents must be analysed concerning agent code. See matter 2.**

- **All Applications which use other databases must be analysed completely, if they contain critical code [.OpenDatabase(..), new NotesDatabase(...), getDatabase(...)]. See matter 1.**

Technical Appendix

Classification

As a consequence of the matters mentioned above a relative simple distribution of the applications into six groups follows. These groups are sufficient to initiate appropriate examinations:

- Class A : simple applications (e.g., discussion databases, etc.)
- Class B : applications with readers fields
- Class C : applications with authors fields
- Class D : applications with critical script functionality
- Class E : applications with time and event triggered agents
  or for a matched rule and enabled ACL lookup in a "User defined Search"
- Class F : applications, which contain a combination of D and E (this group represents a combination and is therefore not a "real" new class)
  or a matching rule in a enabled set of design elements in a "User defined Search"

Strategies for a Solution

First a list of all applications which will be clustered, has to be created. Then the application scans all databases according to this list to find critical characteristics of the script code like authors and readers fields. The result of this scan is a list which contains all databases with a classification to enable further treatment, automatically and/or manually.

Notice:
If a scanner is not available, the code review must be made manually. This, of course, would involve the danger of missing completeness and reliability.

- Databases with critical source code must be searched for and changed manually
- Server groups in the domino directory must be completed with the cluster servers. This could be achieved if special cluster groups are created which contain the cluster servers and the servers are removed from the other server groups.
- Authors and readers fields can be completed automatically if they do not contain groups whose members are editable in the domino directory.
- Mail databases and other applications whose designs are controlled by a template, which contains critical code, can be changed by changing the templates. After the design task has run, the status is correct.

For a migration from R4.6x to R5.0x no serious problems should occur for applications which are using only internal programming functions. Formula Language and Lotus Script are downward compatible, so all code fragments developed for an R4 application must run in an R5 environment. Sporadically problems occurred in versions prior to 5.0.2c, which can be solved easily by creating a new copy of the database.

API functionality of separate libraries, programs, add-ins and server tasks must be taken into consideration separately, because the API versions are differing in some aspects. In general a transfer should not cause serious problems, but all sources have to be reviewed precisely, because the differences mentioned below need not cause syntax errors, but are sometimes semantic.

Technical Appendix

Domino as an application is developed, sold and used for many different platforms. All applications which are developed in Lotus Script, Formula Language and Simple Actions, can be used without any changes on different platforms. But if an application communicates directly with the operating system like reading, changing or deleting data, invoking other applications from a domino application or using program interfaces, which are only available on Windows platforms (OLE, DDE, ODBC), problems might and will arise. These problems are limited to the fact that the actions have to be executed in the server context. This occurs only for agents which are started by a task which is executed directly from the operating system of the computer. This refers generally to the agent manager which executes scheduled and event triggered agents and to the http task which executes QueryOpen- and QuerySave agents. In addition to that used LSX modules have to be tested if they exist for the various platforms. Standards like DECS etc. are not a problem, but all used modules from third party services or manufactures must be get new for the different platform versions.

For all other agents and code fragments like GUI event scripts (QueryOpen, PostOpen, QuerySave, etc.), validation formulas in fields, button scripts etc., which can only be executed from client interfaces, the restrictions do not have any influence, because the executing program (the client) runs under the Windows platform (no other platform is supported for version 5).

Notice:
The backslash and slash problem concerning file names do not need to be taken into consideration, because the server ensures that only the correct separators are shown to the operating system. Even if using a DBDir command (Notice: DBDir lists all file names below the notes data directory) which is started by a scheduled agent on a domino server on a Solaris platform, all file names are processed with a backslash, not with a slash as it might be expected. Problems exist if using lower and upper case letters, because UNIX systems are case sensitive. Further on the use of absolute paths like c:\notes\data\help\help.nsf is not possible.

Technical Appendix

Domino Applications

For the development and the use of applications only the normal restrictions concerning different notes classes, functions, properties and some standard Lotus Script functions and @formulas apply to make them executable on UNIX systems. This applies for the platform OS390, too. In general the context of the OS390 platform which runs the domino server is the same as for UNIX based systems. It is not a R5 standard, but this has only consequences for the operating system commands and the corresponding parameter files (RC files) of the domino server.
Below follows a list of these elements which cause problems and have to be taken into account for a migration of the platform.

ODBC (Database layer)

Doesn't work per default in a UNIX environment, because the whole architecture (3 tier) is Windows standard and had to be reprogrammed on other platforms. There are third party companies who offer driver managers for UNIX, with which the LS:DO architecture would work without changing the LotusScript code, if the DBMS driver is supported under UNIX.

LSX (LotusScript Extensions)

Such modules provide external functionality as LotusScript classes. For this to work, there are shared libraries necessary, which will be loaded by the script interpreter, but these libraries are platform dependant. For example, the uselsx "*lsxlc" command makes the NotesConnector classes available. By the way, these classes are available under UNIX!
But there are third party companies, who deliver LSX modules with their applications, which may not be available for UNIX. (i.e. LDAP)

DOM (DominoObjectModell)

NotesEmbeddedObject.activate,	
.doVerb	
.Object	Doesn't work in UNIX environments at all
NotesRichtextItem.EmbedObject	
.GetEmbeddedObject	
.EmbeddedObjects (Property)	Works with attachments, but not for OLE objects and OLE-Links
NotesUIDocument.GetObject	irrelevant, because of the front-end character of the command
NotesDocument.HasEmbedded	Works with attachments, but not for OLE objects and OLE-Links
.EmbeddedObjects	Doesn't work in UNIX environments at all, because this command returns only OLE objects, no attachments.
NotesInternational.IsCurrencyZero	Meaningless under UNIX

Formulas

@Command([ObjectDisplayAs])	Is not supported
@Command([EditInsertObject])	Is not supported
@DDETerminate(conversationID)	Is not supported
@Command([ObjectOpen])	Is not supported

Technical Appendix

Lotus Script

ActivateApp	Not supported. Generates a run-time error.
ChDir	A run-time error is generated if LotusScript cannot interpret the argument to ChDir, for example if a drive letter is contained in the argument.
ChDrive	Generates a run-time error unless the drive argument is the empty string (""), signifying the default drive.
CreateObject	Not supported. Generates a run-time error.
CurDir, CurDir$	Generates a run-time error unless the drive argument is the empty string (""), signifying the default drive.
CurDrive, CurDrive$	Return the empty string (""), since there are no drive letters on UNIX.
Date, Date$	For reasons of security and system integrity, only the superuser can change the date on a UNIX system. Attempting to change the date under any other username will generate a run-time error. Attempting to change the date while logged in as superuser will change the date system-wide.
Declare	The Pascal calling convention for external function calls is not supported. All external function calls must use the CDECL calling convention. Specifying an ordinal number (using the Alias clause) is not supported. This will return a run-time error at the point of the call to the illegally declared function.
Dir, Dir$	Ignores the optional *attributeMask* argument. These functions behave as if all files have the attribute Normal. **Returns all files for "*.*", not just those containing ".". Returns only those files ending with a period for "*.", not every file without an extension.**
FileLen, Len, LenB, LenBP, LOF	Strings containing line terminators are smaller than on DOS/Windows platforms. The line terminator is one character (linefeed), not two. Therefore the return value of these functions will be smaller for strings on UNIX than on Windows.
GetFileAttr	Generates a run-time error if a drive letter is included in the argument. Does not return the following attributes: ATTR_HIDDEN, ATTR_ARCHIVE, ATTR_VOLUME, ATTR_SYSTEM.
GetObject	Not supported. Generates a run-time error.
Input #, Input, Input$, InputB, InputB$, Line Input, Print, Write #	Compiled scripts using these constructs may be platform-specific, since file data is stored in a platform-specific manner. UNIX character set, byte order, line terminator, and numeric precision specifics may affect the portability of scripts using these functions.
IsObject, IsUnknown	See "Other differences," below.
Name oldfile as newfile	You can rename a directory, but you can't move it except under UNIX.
Open, Lock, Unlock	No explicit or implicit file locking is supported on UNIX. This implies the following: • LotusScript for UNIX allows the user to copy, open, etc., a file that is already opened for reading. Thus, the Name statement works differently on UNIX. • The Open statement may specify only Shared as its lock status. Lock Read, Lock Write, and Lock Read Write will cause a run-time error. • The Lock and Unlock statements will cause a run-time error.
SendKeys	Not supported. Generates a run-time error.
SetFileAttr	Ignores the attributes ATTR_HIDDEN, ATTR_ARCHIVE, and ATTR_VOLUME.
Shell	Window styles are ignored.
Time, Time$	For reasons of security and system integrity, only a superuser can change the time on a UNIX system. Attempting to change the time under any other username will generate a run-time error. Attempting to change the time while logged in as superuser will change the time system-wide.

Technical Appendix

General (Excerpt from the Notes Help)

File system differences

LotusScript respects all aspects of UNIX file system security. This difference affects Kill, Open, and RmDir.

There are no drive letters on UNIX. All devices reside under the root directory. If you use a pathname containing a drive letter, LotusScript may return an error. For the %Include directive, this is a compiler error; for all other uses, this is a run-time error. (Note that since UNIX allows ":" in file names, the statement Dir$("a:") is legal. It searches the current directory for a file named a:.)

UNIX uses the "/" character (slash) as the directory separator while DOS/Windows platforms use "\" (backslash). LotusScript supports the use of slash and backslash, with the following restrictions:

- String literals. If a slash is used in a string literal that is a pathname argument, the .LSO file generated will not run on other platforms, unless that platform supports slash (for example, the UNIX platform).
- String variables. If you assign a string literal containing a slash to a variable, and then pass the variable as a pathname argument, a run-time error occurs if the platform does not support slash pathnames (for example, the DOS/Windows platform).

UNIX allows a wider variety of characters in pathnames than DOS/Windows platforms. For example, more than one "." may appear in a valid UNIX pathname.

LotusScript cannot use UNIX filenames (as opposed to pathnames) that contain the "\" character, since this character is always a path separator on other platforms.

UNIX uses the linefeed (ASCII 10) character as the line terminator. Other platforms use other characters. This difference means that files manipulated with the same LotusScript code, but executed on different platforms, may have different sizes. For instance, the MacIntosh platform uses the carriage return character as the line terminator, so text files written on that platform have the same length as files written on UNIX. Since the Windows platform uses a two-character sequence, text files written there are larger than text files written on UNIX, given identical source code.

Other differences

- Function aliasing with ordinal numbers (using the Alias clause in the Declare statement) is not possible on UNIX, because UNIX has no notion of numbering the routines in a shared library.
- Where wildcards are permitted in file path strings, LotusScript supports the use of UNIX regular expressions in addition to the "*" and "?" characters. However, using regular expressions in file path strings makes the script platform-dependent.
- The Like operator does not use the same regular expression syntax as the UNIX shell. It uses LotusScript regular expressions.
- OLE is not supported on LotusScript Release 3.0 for UNIX platforms. This difference affects CreateObject, GetObject, IsObject, and IsUnknown. The CreateObject and GetObject functions will raise run-time errors when executed on UNIX platforms. The IsObject function tells if a variable refers to a native or product object, but not an OLE object, since OLE objects don't exist on the UNIX platform. The IsUnknown function always returns FALSE on UNIX, since there is no way for a Variant expression to receive the V_UNKNOWN value.
- Date--A date in the format [mm/dd/yy]. The year is optional and defaults to the current year. Use yy to specify a year in the 20th century (yy is 50 or greater) or the 21st century (yy is less than 50); use yyyy to specify any year. The validity of a date format depends on the date separator that users choose in their operating system control panel. The default separator for Windows, UNIX, and Macintosh is a slash (/). The default separator for OS/2 is a hyphen (-).

Technical Appendix

Decision Matrix

Using all mentioned facts of this chapter (with the exception of the C-API) a matrix can be created to facilitate the decision to classify an application. Starting from this point further actions can be performed. These actions are code review, perhaps further programming, the use of tools for the automatic conversion and other administrative tasks (e.g., server and cluster groups, etc.). This matrix applies to clusters and server consolidations to UNIX based systems, too. Notice that In the case of a UNIX consolidation other criteria for the code review have to be used as for the clustering:

Classification Clustering

- Class A: simple applications (e.g., discussion databases, etc.) which are controlled by a standard template
- Class B: applications with authors and readers fields
- Class C: applications with critical script functionality
- Class D: applications with time and event triggered agents

Classification Server Consolidation

- Class A: simple applications (e.g., discussion databases, etc.) which are controlled by a standard template
- Class B: applications with time and event triggered agents, HTTP agents

These classes are combined to the complete classification

- Class A: simple applications (e.g., discussion databases, etc.) which are controlled by a standard template
- Class B: applications with authors and readers fields
- Class C: applications with readers fields
- Class D: applications with critical script functionality
- Class E: applications with time and event triggered agents, HTTP agents

Technical Appendix

	Class A	Class B	Class C	Class D	Class E	Class F
Scheduled Agents - monthly, - weekly, - daily, - more than once a day					X	X (Unix)
Event triggered Agents - before new mail arrives - after new mail has arrived - if documents have been created or modified					X	X (Unix)
Internet Applications - WebQueryOpen - WebQueryClose Agents					X	X (Unix)
Critical Code - concerning clustering - concerning different platform				X	X	X (Unix)
Selected designelements - matching rule in "User defined Search"						X
ACL - matching rule in"User defined Search"					X	
Readers Fields* - refers to forms only			X	X	X	
Authors Fields* - refers to forms only		X		X		
Application is controlled by a Standard Template - does not contain any critical features of the classes B,C,D and E	X					

This matrix has to be read from the upper right to the lower left corner. This means that the most critical problems refer to the class E. In addition to that problems of the class D (critical code) can occur in the context with file operations and OLE. Failover problems like "OpenWithFailover()" cannot appear in this case, because for security reasons server agents are only allowed to open databases on the same server where the agent is executed. Readers and authors fields refer only to forms, so they cannot appear in this context.

Technical Appendix

Description of the Classes and Tasks to do after the Classification

Class A **(not critical)**

A simple test, if the application is working, is generally enough. There should be no problem.

Class B **(not critical)**

It has to be ensured that the cluster servers have got at least editor rights in the ACL of the database. If this is not possible, they have to be typed in all authors fields to enable the cluster replication of the documents and to enable the server to edit all documents. In this case the administrator should create a suitable group which must be added into these fields (if roles are not used).

General

Readers and authors fields limit the access for editing and reading documents. These limitations apply to every kind of user, so for servers, too. Because of this it is required to review the contents of the authors and readers fields in a cluster accurately.

Authors Fields

Authors fields define the access for editing documents by the members of an author group defined in the ACL. Authors are allowed to create documents, but only to edit the documents where they are listed in authors fields

Authors fields do not change the access rights as defined in the ACL but complete them. Users with editor rights and above are allowed to edit all documents of the database. They are not affected by authors fields. Further on authors fields can not grant editing access for users with readers rights and below.

Implications for a Cluster

In a cluster readers and authors fields can cause problems if the cluster servers do not have the necessary rights to access documents. The basic access rights of the cluster servers are implemented in the ACL. But wrong or missing entries in authors and readers fields might prevent the correct execution of applications.

Consequences of Authors Fields

Authors fields in a cluster must only be taken into account if the server has author rights on the database. In this case problems exist in the following situations:

- Not all cluster servers are directly (by name) or indirectly (by group) listed in the author fields of a document
- The servers execute agents which edit the documents

Technical Appendix

Class C **(critical but relative rarely)**

It must be ensured that the cluster servers are listed in all readers fields to allow the server to read the documents at all. This is important for the cluster replicator and for agents. In this case the administrator should create a suitable group which must be added into these fields (if roles are not used).

General

Readers and authors fields limit the access for editing and reading documents. These limitations apply to every kind of user, so for servers, too. Because of this it is required to review the contents of the authors and readers fields in a cluster accurately.

Readers Fields

Readers fields are used to limit the access to a document. If a document contains a readers field, only users have access to this document which are listed in this field. All other users are not allowed to read or edit the document. Further on then the document is not visible in views and for not suitable for actions (like agents, event triggered actions, etc.) which are started by the user.

Implications for a Cluster

In a cluster readers and authors fields can cause problems if the cluster servers do not have the necessary rights to access documents. The basic access rights of the cluster servers are implemented in the ACL. But wrong or missing entries in authors and readers fields might prevent the correct execution of applications.

Consequences of Readers Fields

If entries for the cluster servers are missing in readers fields, a user can create documents on the server, but the server is not able to view or replicate these documents. This implies the following:

- The server can not process these documents by the use of agents
- The server can not replicate these documents, so replicas in the cluster are not synchronous
- All other cluster servers can not replicate these documents, so the user is able to see the documents on the server where they are created but not an all other cluster servers. The failover function does not work.

Class D **(critical)**

It must be ensured that in the case of a failover the work can be continued. This might cause considerable intervention to the program logic if, for example, applications are used where server names and database paths are taken from a profile document. @Commands cause problems, too, because they are not designed for failover actions. When using Lotus Script it must be ensured that OpenWithFailover() is used to open other databases. For LSX includes file operations and OLE/COM operations have to be observed

In more complex applications it is sometimes necessary to leave the notes/domino environment and to access external data sources and programs. Here several possibilities exist.

1. **Access to Relational Data with the ODBC Interface**

   The ODBC interface allows an access to relational data in WIN32 systems. In Notes/Domino applications it can be addressed by using the commands **@DBLookup("ODBC",...), @DbColumn("ODBC",..), @DbCommand("ODBC",...)** in the formula language or with the LSX - Extension **LS:DO**. Of course Java is able to access the ODBC interface, too.

2. **Access to Notes Databases with NotesSQL**

   NotesSQL is the standard ODBC driver for Notes/Domino databases. It allows external programs access to data within Notes/Domino applications. If NotesSQL is used, for a migration of Notes/Domino applications it has to be taken into account that external accesses to these applications consider the characteristics of the cluster.

3. **Access to Non-Notes Applications with COM / OLE / CORBA**

   With the use of the COM/OLE interface it is possible to manage external programs from Notes if they are COM compatible. The connection is established with Lotus Script by using the commands **CreateObject** and **GetObject** and the method **CreateObject** from the class NotesUIDocument. The resulting object resembles a COM compatible program. It enables methods which can be invoked from the Notes/Domino application.

   CORBA is supported starting with version R5, so it needn't be taken into account for a migration.

4. **Use of Notes/Domino Standard Connectors**

   1. Lotus Enterprise Integrator LEI (former NotesPump)

      The "Middleware" LEI allows data management between different servers. It opens connections and transfers and synchronizes data. It can be started triggered by an event or according to a schedule. Primarily LEI is used for the transfer of huge amounts of data in the background and therefore is an extension of the Domino Enterprise Connectivity Services DECS.

   2. Domino Enterprise Connectivity Services DECS

      The Domino server task DECS allows to connect Notes/Domino applications with relational data sources in real time. DECS monitors events on the server and starts predefined actions in the relational data sources if particular events happen (e.g., opening of a form). To establish a connection several possibilities exist:

      1. Creation of connection and activity documents on the server. In this case the use of DECS in the database can not be determined unambiguously.
      2. Use of the Lotus Script Extension LSX LC

         In this case the following code segment must be in the options of a Lotus Script event code:

         USELSX "NLSXLC"

         This interface is another possibility besides ODBC for a connection from Domino to relational database which is much faster and more secure than ODBC. Additional connectors have to be used, like DBS Oracle, etc.

      3. Domino "ERP" Connector

         The Domino Enterprise Resource Planning Connector is used to connect domino applications to ERP systems.

5. **Access to the Operating System**

   Notes/Domino applications can directly access the file system with different Lotus Script commands and within Java. They can invoke programs and commands directly to the operating system. Because in a cluster the application is installed on different servers, these commands access in the case of a failover different file systems.

Technical Appendix

Implications for a Cluster

The main problem is the guarantee that the failover is functioning while simultaneously the connection to external data sources and programs must be maintained. Following problems can occur:

1. **Use of the ODBC Interface**

   The ODBC interface allows the access to data outside of the notes cluster. If the data source is not present on the client in the case of a failover the following problems can occur:

   The connection to the relational data will be cut off, because it does not change the location of the Notes/Domino application dynamically

   The relational data source is not available, because it resides on the server which is not available any more

2. **Use of NotesSQL**

   An external program uses NotesSQL to access a Notes/Domino application. If the access is bound to a particular server the process will fail in the case of a failover

3. **Use of the COM/OLE Interface**

   The COM/OLE interface allows the access to external programs outside the Notes cluster. If the application is not installed on the client, following problems can occur in the case of a failover:

   1. The connection to the external program will be cut off, because it does not change the location of the Notes/Domino application dynamically
   2. The external program is not available, because it resides on the server which is not available any more

4. **Use of Connectors**

   For the use of connectors normally a special architecture of the Notes/Domino application must be present. For the operation inside a cluster further matters have to be taken into account.

   1. The used connectors must be installed and be available on all servers of the cluster. Adjustments on a server must be made for all cluster servers.
   2. The connection established by the connector must established in the case of a failover, too.
   3. The failover process will be prevented if the target of a connection with a connector resides on a cluster server.
   4. If load balancing is used, it must be tested, how the simultaneous connection from two replicas to the same target acts.

5. **Access to the Operating System**

   Accesses in a cluster to the operating system occur from different points. If the access occurs to the operating system of the client, no problems should be found. Otherwise the following matters have to be taken into account:

   1. The locations which the application accesses in the server file system, must be present on all cluster servers.
   2. If the file system is accessed not only in read mode, but in edit mode, it has to be planned how the data will be transferred to the other cluster server in the case of a failover.

Technical Appendix

Class E **(very critical)**

For a server consolidation to a UNIX based system first a code review has to be made to look for sequences with critical commands and classes. If this is the case, this code has to be deleted or changed. This might be time-consuming, because the program logic could be destroyed. But this situation is not certain, the probability is rather low.

Internet applications which contain QueryOpen and QueryClose agents, must be reviewed concerning critical code. For the execution in a cluster these agents are not important, because they are executed on the server, but their invocations are only made on the server which the user has been accessed or addressed with an url. Synchronization problems might occur, but these should be covered in the non-clustered work anyway .

Applications which are designed for the work in a cluster must be adapted in all known cases. According to the different and individual program logics extensive modifications of the agents have to be made, because with "normal" notes based methods no synchronization can be obtained. This refers to documents which are affected by server agents (because all agents see the same source data and change them and this causes replication conflicts) as well as to other actions. It can not be controlled that the agent on a server A runs normally and on server B only in the case of a failover.

General

Agents execute repeating tasks at different places in a Notes/Domino application. They are started either manually by a user or automatically in the background. In a cluster only the agents can cause problems, which run in the background on the server. These are:

- event triggered agents and
- scheduled agents

Event triggered Agents

Event triggered agents can be invoked with several ways

- Before new Mail arrives

  The agent is executed before the mail arrives into the mailbox of the user. This is possible by taking the mail router as the trigger for the agent. Therefore the agent only runs on the server where the mail router resides. This function is new to version R5.

- After new mail has arrived

  The agent is executed after the mail is deposited into the mailbox of the user. The agent only runs on the home mail server of the user who has signed the agent. It does not execute on other servers. The name of the home mail server is entered in the person document of the agent signer. This will be determined by a lookup in the domino directory.

- If documents have been created or modified

  The agent is executed after a document in the database has been created or modified. With the use of replication these documents can be transferred to other servers, so it is possible that the agent for a new document runs more than once.

- If documents have been pasted

  The agent is executed if documents are pasted into a database. Again a multiple execution of the agent can not be excluded.

Technical Appendix

Scheduled Agents

Scheduled agents run on a schedule. The server on which the agent should run can explicitly be selected. It is possible to start the agent on a single server or on a group of servers.

Implications for a Cluster

The Use of event and scheduled agents can cause problems in two ways:

1. The agent runs on one single server
   In this case the failover does not work. If the server on which the agent is intended to run is not available, the agent is not started, because the cluster manager does not have a failover for agents
2. The agent can run on one more than one server.
   In this case the replicas of a document might be changed simultaneously causing a violation of the data integrity.

Without any problem in this sense is only the use of a agent which runs before new mail arrives. These agents are triggered by the mail router and are fully compatible with a cluster and therefore with the failover process.

Class F (very critical)

To investigate server consolidations to UNIX based systems an additional class F is implemented which is used for applications which are a combination from class 4 (critical code) and class E (scheduled agents). The reason for this is to have the possibility to analyse the application's output arranged by scheduled agents and critical code. Otherwise one would have to search for the critical code sequences inside the application's output for the scheduled agents which might result from script libraries of the agent, too. Therefore it is not really a new classification but a combination of classes.

Class X (not possible to analyse)

If the design of a application is hidden by a DesignReplace, this database can not be analysed any more, because no reference to the source code can be found. Normally this is not a serious problem because the template should reside on the server, too. If the template is not available, the application can not be used anyway, because a change of the code would be impossible.

Data Appendix

A) Scanned servers

1. [illegible]
2. [illegible]
3. [illegible]
4. [illegible]
5. [illegible]
6. [illegible]
7. [illegible]
8. [illegible]
9. [illegible]
10. [illegible]
11. [illegible]
12. [illegible]
13. [illegible]
14. [illegible]
15. [illegible]
16. [illegible]
17. [illegible]
18. [illegible]
19. [illegible]
20. [illegible]
21. [illegible]
22. [illegible]
23. [illegible]

Data Appendix

B) Replica Listing sorted by server*:*

Server	Database	Replica Count
Server1		**183**
	DB1.nsf	5
	doclbm50.ntf	13
	dpa.ntf	3
	dpd.ntf	3
	dpv.ntf	3
	GSSS_Config.ntf	10
	GSSS_Log.ntf	11
	GSSS_Update.ntf	11
	imapcl5.ntf	13
	intadrbook.nsf	10
	pubweb50.ntf	13
	telefone.nsf	4
	userreg.ntf	13
	Databases/data1.nsf	4
	Databases/data2.nsf	4
	Databases/data3.nsf	4
	Databases/data4.nsf	4
	Databases/data5.nsf	4
	Databases/data6.nsf	4
	Databases/data7.nsf	4
	Databases/data8.nsf	4
	Databases/data9.nsf	4
	Spain/Purchaseorder.nsf	4
	Finance/Service.nsf	4
	McAfee/GSD53_Config.nsf	3
	McAfee/GSD53_Help.nsf	10
	McAfee/GSD53_Log.nsf	3
	McAfee/GSD53_Update.nsf	3
	Mail/ProjectTracking.nsf	4
	Regensburg/DDDAtrib.nsf	4
Server2		**57**
	BinaryTree.ntf	1

Data Appendix

	ACL Template.ntf	1
	ModACL.nsf	1
	mail_inotes_en.ntf	1
	international.nsf	8
	DDD/Pipeline/Pipeline.nsf	10
	DDD/Pipeline/CC.nsf	10
	DDD/Pipeline/CO.NSF	10
	QA/Technical Service Requests.nsf	3
	CQA/DDENADGB.NSF	1
	QA/DDNADMGB.NSF	1
	CCI/HHHGB.NSF	1
	CCI/Desktops.nsf	1
	CCI/ForDomains.nsf	1
	EGG/MARKTSEG.NSF	1
	EGG/TEXITIN.NSF	1
	Intranet/tracking.nsf	1
	MMM//ServerL.ntf	1
	PDS/Web/probe1.nsf	1
	PDS/Web/probe2.nsf	1
	test/Functional/surface.NSF	1
Server2		**94**
	….	2

Data Appendix

C) Template Extraktion

Here we see an extract of the template view but only the details for ClassF applications. All others are not expanded due to size and clearness reasons. For details please refer to the view “Admin/Analyses/Templates” which gives you the detailed information about all applications.

Class Category	Server	Inherited Design - Template	Database	Class	Count	Overall time (hours)	Class F Code
					2314	**91,38**	**419**
A					**860**	**0,00**	**0**
D					**544**	**29,25**	**134**
E					**765**	**30,13**	**138**
F					**59**	**30,13**	**138**
	Server1				1	0,25	1
		.... No references			1	0,25	1
			Spain/Purchaseorder.nsf	DEF	1	0,25	1
	Server2				1	1,31	6
		T_IMSMain			1	1,31	6
			DDD/Pipeline/Pipeline.nsf	XDEF	1	1,31	6
	Server3				13	5,06	18
		.....					
X					**86**	**1,88**	**9**
					2314	91,38	419

Data Appendix

D) Unix ClassF application list

Data Appendix

Server	Database	Template - DesignTemplate	Replicas	Views	Agents	Forms	DB Count
[redacted]			4	68	6	44	1
	[redacted]		4	68	6	44	1
[redacted]			4	76	22	35	1
	[redacted]	[redacted]	4	76	22	35	1
[redacted]			39	1039	224	361	13
	[redacted]	[redacted]	3	119	46	60	1
	[redacted]	[redacted]	3	67	14	17	1
	[redacted]	[redacted]	3	48	19	15	1
	[redacted]		3	59	11	75	1
	[redacted]		3	6	6	4	1
	[redacted]	[redacted]	3	62	6	13	1
	[redacted]	[redacted]	3	62	6	13	1
	[redacted]	[redacted]	3	62	6	13	1
	[redacted]	[redacted]	3	62	7	13	1
	[redacted]	[redacted]	3	62	6	13	1
	[redacted]	[redacted]	3	62	6	13	1
	[redacted]	[redacted]	3	62	6	13	1
	[redacted]	[redacted]	3	306	84	99	1
[redacted]			0	1246	339	436	6
	[redacted]	[redacted]	0	62	7	13	1
	[redacted]	[redacted]	0	288	79	94	1
	[redacted]	[redacted]	0	288	77	97	1
	[redacted]	[redacted]	0	306	80	99	1
	[redacted]	[redacted]	0	119	46	60	1
	[redacted]	[redacted]	0	67	14	17	1
	[redacted]	[redacted]	0	48	22	21	1
	[redacted]	[redacted]	0	75	17	34	1
[redacted]			16	275	63	93	1
	[redacted]		16	275	63	93	1
[redacted]			0	24	46	51	1
	[redacted]	[redacted]	0	11	23	28	1
	[redacted]	[redacted]	0	13	23	23	1
[redacted]			16	181	71	89	1
	[redacted]		16	8	4	4	1
	[redacted]	[redacted]	6	176	70	88	1
[redacted]			3	139	31	63	1
	[redacted]	[redacted]	3	139	31	63	1
[redacted]			0	189	59	90	1
	[redacted]	[redacted]	0	189	59	90	1
[redacted]			203	4400	1137	1653	22
	[redacted]	[redacted]	1	7	10	19	1
	[redacted]	[redacted]	0	4	10	19	1

Data Appendix