

to Master Your Domino Application Migration Challenges

NEW WAYS

EBOOK

Contents



	Way #2: Use the big picture of your code to know where to go	11
	Way #3: How to keep your project cost efficient from start to finish	16
	Way #4: Seeking application owners? Find application stakeholders instead!	20
	Way #5: About the many roadblocks in your source code	25
	Way #6: 3 New Ways to Automate Your Progress Reporting	34
	Way #7: Know before you go!" Using Facts to Plan Your Domino Application Migration	50
0 0 0 0	Bonus: Understanding Domino Application Data – Using Content Analysis for Migration Planning	53

Authors

Franz Walder

Product Manager at panagenda

Franz Walder is product manager of infrastructure optimization solutions in panagenda, an Austrian based IT management software solution provider. He has worked many years with companies in or around the financial sector in Europe in the field of IBM Lotus Notes Client and Domino Server administration. In addition to IBM software, his technical expertise from software spans development over Linux administration to being a virtualization enthusiast. Currently living in Vienna, Austria, Franz is a regular speaker at technical conferences in the US and Europe.



Florian Vogler

CEO at panagenda

From Commodore 64 Personal to Computers, AS/400 to S/36, DOS, Windows 3 to 8, OS/2, Apple MacIntosh to MacBooks and iPhones - Florian has always been excited by technology and especially software. After growing up in Germany, living in the United Kingdom, Austria and Spain, as well as spending a lot of time in the United States, Florian has settled for Germany as home country from where he travels extensively across the globe. Throughout his career in IT, he has worked in a number of managerial and executive positions in both the banking and information technology industry. Florian holds an executive MBA from private university PEF, Vienna (Austria).



Many companies around the world have been committed to HCL Notes/Domino* for years. They know the many benefits that come from that relationship. Additionally, Notes/Domino lies at the center of their processes and how they work. Despite all this, IT decisions makers around the world are starting to envision a future where Notes/Domino may play a reduced role or no role at all.

In this series of seven articles we will be discussing your toughest Notes/Domino migration challenges. The real-world solutions we will provide are not available from any other single source.

You will learn how taking a new approach to data analysis and presentation can revolutionize your ability to simplify the most complex Domino projects and deliver them efficiently, on time and on budget.

Helping you ride the crest of the technological wave of success is the motivation behind this series.

*formerly IBM Notes/Domino

Way #1

Safely Navigating Your Domino Projects With Confidence

The Migration Challenge

In many organizations, the Notes/Domino environment has organically grown over decades. It is deeply embedded in business processes and represent a big investment in the application landscape. In the 90s, many non-programmers created their own apps using templates. They even modified design and functionality without coding. Today, this is spoken of as citizen-developers doing low- or no-coding. Regardless of the term, the result was that database/application numbers sky-rocketed as more and more templates became available.

The market shows that in the last couple of years many companies have already migrated away from Notes/Domino to Microsoft Outlook or Microsoft 365. Others are in the process of doing so. In many cases the decision to migrate is a strategic one, not driven by technical issues or missing features.

Microsoft says, and our experience confirms, that migrating Mail is usually a straight-forward process. There are several tools on the market that can help with the planning and execution of this.

Application migration is much more complex. Most organizations have been developing applications for years. There can be tens of thousands of applications that need to be processed. They are all unique and have specific requirements. How valuable are they to the firm? How frequently are they used? By whom are they used? How complex is the code? These are just a few of the questions you need to answer. Having worked we hundreds of customers over the years, we know, how difficult it can be to find the internal expertise to be able to quickly and easily deliver precise answers for the entire organization. This is where the real challenge lies. It is, therefore, often dismissed by the Microsoft proponents as being as easy to implement as the mail migration. Unfortunately, it is anything but.

For most organizations, Notes/Domino is so deeply imbedded in their systems and processes, completely replacing all their Domino apps will take many months and often many years. Intensive application of time, people and money might shorten the process.

Like us, you'll probably know that even some of the largest organizations with the deepest pockets have taken more than 10 years to phase out their entire Notes/Domino infrastructure.

Getting Answers – Your First Steps for a Successful Project

Having right answers to the right questions from the start and being able to access that critical information in a central repository lays the groundwork for a successful project. With that information in hand, you can quickly develop an accurate migration plan that will minimize the time, resources, and costs involved in the project.

Getting it right can reconcile the conflict between the wish to simply 'get rid of Notes by the end of the year' and the actual implementation to be carried out by the project managers, Domino admins, developers and IT managers.

The first step is to understand the magnitude of the project:

- What's the structure and size of your environment?
- How many databases are unused?
- How many databases can be excluded from the project?
- How many users are actively using Notes?
- What are the most efficient migration paths for your applications?
- How does DB design complexity impact redevelopment costs?
- Who are the application owners who are stakeholders in the project?

Many of these questions must be asked continuously and answers will change over time as the project progresses and circumstances evolve.

The Usual Approach

As previously stated, many firms have little to no internal expertise to help navigate their way through the complexities inherent in a migration project. This leads them to bring in external consultants who will be paid to audit the existing environment, to deliver answers to the questions above, and to devise a project plan.

The benefits to such an approach would appear to be clear. The migration will be carried out by experts with lots of experience. This expertise should reduce the scope of the project, reduce the potential for errors in the implementation, and save costs.

Look a little deeper, however, and the benefits may well be outweighed by the disadvantages.

The most obvious disadvantage is that these projects are lengthy, and the cost of external consultants will often add up over time.

Less apparent is the inherent bias consulting companies have to prolong projects rather than to shorten them.

They will often use their own software tools during the preliminary, investigation stage of the project to find answers to the questions above. Once the



project is over, the software and access to this data leaves with them.

There is also the potential they will have access to their client's most sensitive data.

Once they develop a migration plan, they will often be retained to perform the implementation. In doing so, they naturally acquire an intimate understanding of the design and function of the new environment. Again, when the project ends, this knowledge is lost and there is no-one in-house with that expertise.

A New Approach with a Better Result

Is there an alternative approach? Is there a single source of information for you that is easy to use, comprehensive, accurate and unbiased? Is there a way to gain this information independently and continuously without bringing in a bunch of expensive external consultants?

Fortunately, there is.

iDNA Applications is designed to support you during what may be one of the most challenging IT projects of your career – migrating away from a longentrenched Notes/Domino environment.

- It gives you a completely new view of your application environment.
- With the click of a mouse, even the most complex environments are rendered in simple, easy to understand graphs and datapoints that give you the answers you need.
- Code complexity analysis combined with application usage gives you the ability to make value-based decision making for every step in your migration journey.
- Proprietary application rankings keep you focused on the most important project deliverables.

• These points and more ensure you will be able to deliver your project easier and faster than you ever thought possible.

Cutting Your Project Down to Size

You'll see a diagram of the complete Notes/ Domino environment in the Environment Overview dashboard – the starting point for your analysis. You can quickly determine the potential for scope reduction by comparing the Databases Deployed information with Databases Touched.



Figure 1: Environment Overview Dashboard

Reduce Project Sizes up to 90%

Now look at the 'Scope Focus Funnel' dashboard. Here is an easy to read graphical representation of the potential for scope reduction. Each section of the funnel describes for you the process used for eliminating applications that do not need to be migrated. Additionally, you can quickly see in the upper right-hand corner the potential for scope reduction displayed as a percentage.

The combined effect of eliminating these from your project has been shown to reduce project sizes by up to 90%!

This is a quick-win for you, and it is massive. Imagine the savings in time, cost and resources when you cut the project scope down to 10% of what it was.

Efficient Use of Your Programming Resources

Certain applications such as System DB's and mailfiles are not suitable candidates for migration. These are automatically excluded from your potential migration universe. The remaining applications are then analyzed for you. You can see their programming complexity and similarities. Armed with this information you'll know how to dedicate your programming resources most efficiently. Popular, complex applications will require more time and resources to migrate. Other applications can be simply and quickly handled with



Figure 2: ScopeFocus Funnel Dashboard

a minimal investment. Additionally, applications can be consolidated so you can reduce the scope of the migration even further. You'll have another quickwin by knowing which applications can be migrated quickly.

In addition to these two benefits, you can allocate your programmer resources efficiently. You'll have the people you need, where you need them to get those complex applications across the line in time.

Cost Savings and Roadblocks

Projects like these are expensive and they are highprofile. How would you like to be able to propose offsetting some of the cost by eliminating unused licenses? It's easy! The information you need to monitor end-user usage is at your fingertips. Unused licenses can be cancelled or better allocated.

The User Activity Overview dashboard gives lets you see quickly and, easily how actively end-users engage with applications. Armed with this knowledge, it is a simple decision to either reallocate or cancel unused licenses to maximize the value your company is receiving from its licensing spend.

O List	of users with details or	n their activity Explore in Me	tabase to filter			When were my users last active?		
Use	manie	Last Activity Category	Days Since Last Activity	DBs Used Category	Count DS Instances Used			
Aad	am Flynn/Acme	Less than a month ago	1	25 to 99 DBs	28			
Aad	il Bass/Acree	1 to 3 months ago	38	10 to 24 DBs	19	Less than a month age	b 19.0% Users	
Asli	yah Mellor/Acme	1 to 3 months ago	40	25 to 99 DBs	52	1 to 3 months ago	20.7% Users	205 11-1-1
Ажу	an Montoya/Acme	No activity	12	No usage	0	6 to 12 months ago	9,51% Users	305 Users
Abri	sham Coles/Acme	1 year ago or before	665	3 or less DBs	3	1 year ago or before	10.2% Users	
Abu Abu	bakar Brett/Acme	Less than a month age	1	100 or more DBs	336	No activity	38.0% Users	
Aga	tha Robbins/Acme	No activity		No usaga	0			
Abl	a Flower/Acme	Less than a month age	1	10 to 24 DBs	10			
Aist	a Perkins/Acme	Less than a month age	2	4 to 9 DBs	6			
Alar	tair Shea/Acme	1 to 3 months ago	40	100 or more DBs	461	How many applications do my user	s use?	
Alec	Saknas/Acme	No activity		No usage	0			
Alex	ia Croft/Acme	No activity	-	No usage	0			
y Lookep Alis	a Strong/Acme	3 to 6 months ago	101	3 or less DBs	2	100 or more DBs	11.8% Users	
Alis	hia Ratcliffe/Acme	1 to 3 months ago	40	25 to 99 DBs	59	25 to 99 DBs	12.8% Users	
Arts	alie Kumar/Acme	1 to 3 months ago	46	25 to 99 DBs	28	10 to 24 DBs	15.1% Users	305 Users
NEW ATTA	inah Tang/Acme	1 year ago or before	294	3 or less DBs	1	4 to 9 DBs	3.93% Users	TOTAL.
Artic	iyah Head/Acme	Less than a month ago	3	10 to 24 DBs	10	No usage	18.0% Users	
Ani	er Workman/Acme	6 to 12 months ago	207	3 or less DBs	2			
					Deres 1 10 of 201			
			LICENSE COST	TS	Hows 1-16 01 303 1 1			
E P	owered by Metabase				· · · · · · · · · · · · · · · · · · ·			
			Identify i		ers and			
			reallocate		unused			
			licenses to		e value of			
			neemaes to		s value or			

Figure 3: User Activity Overview Dashboard

Taking away someone's much loved Notes/Domino applications, can be a sensitive issue and have a real impact on the business. That's why the application owners are identified for you. You can smooth the path of your migration by approaching them in advance and eliminating potential roadblocks before they arise.

You can also use this information to rationally allocate project costs. Complex applications require more resources and, therefore, cost to migrate. Since you will know the owner and you can see the usage, you can see who benefits from migrating each application and you can proportionally allocate cost accordingly.

What's In It for You?

Who doesn't want to be able to make their projects simpler, smaller, quicker and cheaper? Who doesn't want to have a bigger budget, say in the form of reducing license costs? It all starts with having the tools to hand that will quickly and easily give you the answers you need in one place. From here you can devise the best possible strategy to deliver your project on time and on budget and enjoy the following benefits:

- Knowing which applications don't need to be migrated reduces your project to its true size for more accurate scoping
- Your timelines are shortened
- Your deadlines are more accurate
- Meeting deadlines means no knock-on effect for projects already planned for the future
- Programming resource requirements can be more accurately estimated and better allocated to where you need them to be
- Access to continuous information about your environment make your progress reporting easy, flexible and accurate

Way #2

Use the Big Picture of Your Code to Know Where to Go

Quo Vadis? Start With the Obvious

In way #1 we talked about how Notes/Domino environments have been growing organically in companies around the world over decades. We also outlined the role played by "Citizen Developers" in that growth. It's easy for non-programmers to create their own apps using templates. They can even modify design and functionality without coding.

Up to 97% of Code Duplication

The result was database/application numbers skyrocketing in the golden years of Notes. As more templates became available, that development was fueled even further!

The challenge we face in today's projects isn't just whether databases are used or not. Project managers and developers have to know the purpose these applications serve, what functions they provide and how they integrate into business processes. Quite the challenge since they probably only use a fraction of those applications personally. One way to tackle this challenge is by taking advantage of the similarity between databases to reduce the amount of effort required.

Only 5-10% of Your Source Code Might Be Unique

A very high rating of code duplication is very common in Notes environments. At panagenda we have seen up to 97% code duplication in customer environments.

There's good reason for such high numbers. Design templates in Domino applications are

extensively used. Standard templates like Mail, Discussion and Document libraries are favorites. It makes sense and it should be encouraged. It's more efficient to incorporate their core functions than it is



to continually develop them from scratch.

In fact, once you know where your duplicate code can be found, code duplication can be used to speed up your Notes migration and modernization efforts!

Using Cut-and-Paste to Speed Your Migration and Modernization Projects

Not a lot of people are aware of these numbers. How could they be? Information like this is almost impossible to gather. Once you have it though, it can be an extremely powerful tool.

Imagine your developers know all locations of a single code block right from the start. They could write one fix, one time, for one app or template. After that, a simple cut-and-paste can be used to apply that fix wherever it's needed!

Think of the potential reducing time spent coding when up to 90% of your coding can be done with cut-andpaste! Not only is time and effort reduced significantly, but the skill profile of part of your developers change.

Find Your Duplicate Code – the Key to Speed

The Source Code dashboard in iDNA quickly shows how much code is duplicated in your environment. You instantly start to have an idea of the potential to speed development in your own environment.

Imagine you are going through a mail migration project. You're looking for applications that could break when the mail system changes. How can you find that information?



It's not difficult. Just go to the Database Catalog. Click on the Filter button. Here you can select "Insights". Insights are potential issues we discovered for you during the source code analysis.

Now You Know Where Your Cut-and-Paste Needs to Go

Based on this principle of shared code, iDNA Applications automatically creates "Code Similarity Clusters". They show which databases share the same source code and how much of that source code they have in common.

5	Search in a	III fields	FILTER T 1 Default Colum	nnset 🗸 🕂	CSV EXCEL 2		
			Mail Send and Sign				
	Tit	tle	Filter databases by Insight — DEPENDENCIES ON OTHER DATABASES —	Filename	Replica ID	DB Туре	Last Access
	Pr	rocess Ma	@DbExists Formula Definite dependencies on other databases	f6e261f69695bccb1a.nsf	C12581DF:0075AF71	Application	Sep 23, 2019
⊕	FC	DE Centre	Possible dependencies on other databases —— IBM CLOUD READINESS ——	1b50915f71d752bafa.ntf	852580A3:0048BC5F	Application	Aug 13, 2019
#	N	FV- Netwo	What's new in Notes 8.0 What's new in Notes 8.5	dbd6fc42e5fb0058d4.nsf	C1257C43:005F0142	Application	Aug 13, 2019
⊞	Ca	areer Deve		8a5e55f2a36e818565.nsf	C1257600:0077D45C	Mailin DB	Sep 11, 2019
	Go	oal-Driven	Addressbook related code Calendar related code	347039502d02148e46.nsf	C1257D0E:0035C5C0	Mailin DB	Sep 13, 2019
	Pr	rocuremer	Mail Recipient Fields Mail Send	082849fac74e1e1475.nsf	C1257C31:004AA102	Application	Sep 13, 2019
	Bi	ig society	Mail Send and Encrypt	a54824604218f644b5.nsf	C1258398:0032C66F	Application	
	Н	ealth Deve	Mail Send, Sign and Encrypt	31296d5e90a06b198f.nsf	8525805F.002C2966	Application	
	Di	isability S	Todo / Tasks related code	bc3653c43caaf0baa9.nsf	C1257BAB:0057DA4C	Mailin DB	



Our experience in customer projects has highlighted three use cases where code duplication and similarity can be utilized for maximum impact:

- Finding the original templates used by applications that do not have an official configured "Master Template". Knowing which applications share the same internal structure and the degree to which that structure is shared, means you can design one, common roadmap for handling these applications.
- 2. Identifying design differences between template and database instances. Have your applications been modified from their original templates? You'll want to know! These may require more programming resources to properly process them within your project. Once you know, your developers are where they're needed, when they're needed.
- 3. **Separating applications based on IBM standard templates from those that are not.** Many applications will be based on standard templates. There will already exist developer tools that automate their handing in any project. They tend to be easier and quicker to process.

Don't worry. It's taken care of for you. It's more quick wins without lifting a finger.

Get to know your code before you hit the project road

Having a thorough understanding of your applications and their code creates so many opportunities for you. We've only been able to cover a few:

- Save time
- Improve development efficiency
- Reduce costs
- Speed your project
- Improving planning and resource allocation
- Minimize the risk of failure

Contact us to see the possibilities that await you or visit our <u>Online Knowledge Base for more</u> technical details.¹

Way #3 How to Keep Your Project Cost Efficient From Start to Finish

Know the True Scope of Your Project From the Start

We already talked about how to focus the project scope on what's most important and how having an overview of your code base can help you be more efficient and make better decisions for your project.

Now, we want to take on one of the big hurdles most of us know very well from personal and work related experience: "Finding a good place to start". It doesn't matter if your Notes/Domino project is modernizing your infrastructure, migrating to the Cloud or leaving the platform entirely.Consolidating your existing infrastructure is a crucial first step in the most successful projects. Finding and removing the deadwood in your environment will significantly reduce the overall project effort and save costs.

Ask the Questions That Reveal the Core of Your Project

Asking the right questions was a key factor in reducing the project scope in the <u>first part of this series</u>². Let's take a look at some of those questions again and concentrate on how we can use the answers to identify consolidation potential.

1. What's the Structure and Size of My Environment?

You must have a true understanding of the magnitude of the undertaking. You need KPIs to help evaluate consolidation potential and measure success:

Users

- How many users are registered vs active?
- When were the users last active?
- How many applications do my users really work with?

Servers

- How many servers are active?
- How many are application servers?
- How many databases are being used on a server?
- How many users are active on a server?

Applications

- How many applications are deployed vs used?
- How many DBs can be excluded from my project?
- How many databases are based on standard templates?
- How many replicas of each database exist?
- How many users are active per database?

2. How Many Databases Are Unused?

This is one of the most frequently asked questions we hear. It's an obvious starting point for consolidation.

The first instinct is often to simply remove all unused database instances and celebrate a quick win. That can make sense in some situations: e.g. – when one is only looking to reduce the number of servers, but it is very unreliable for judging which applications can be de-commissioned.

More interesting is to find out for how many applications no database instance is used. These groups of database instances which share the same Replica ID are what we call "Replica Sets". Only if the entire replica set is unused and activity does not occur on any server, can an application truly be considered unused.

The period over which activity is observed plays an important role in decommissioning. Not every database is used every day. You might have databases with seasonal usage which are only used once a year, quarter or month. To capture this variation, panagenda recommends a minimum observation period of three months to ensure a database is unused. The longer the observation period, the better decisions you will be able to make.

Having identified the deadwood, we take the first step to cleaning our environment, we continue to seek opportunities in slightly more demanding areas.

3. Where Can Costs Be Reduced?

A consolidation is an opportunity to find and eliminate the cost of unused licenses. Most organizations carry inactive users and under-utilized servers on their books. Acting on this information can save a lot of money with very little effort.

Re-evaluate 15 Year Old Designs

Finding and eliminating inactive DBs from your environment results in decreased server utilization. Consolidating several under-utilized servers reduces license, operations and administration costs.

The consolidation is also an opportunity to review the design of your Domino network. For example, environments designed 15 years ago with bandwidth considerations in mind should be re-evaluated. Modern infrastructure might have already made previously important redundancies or parts of the server de-centralization concept obsolete.

Many organizations choose to create hybrid environments with applications running on multiple solutions. Even if Domino remains a strategic application platform, certain generic services (e.g. mail, team rooms or file libraries) might become de-coupled from it over time. In such a migration scenario, user activity is also affected. Additional license savings potential is generated and can be predicted by understanding database design.

Even if you are not currently engaged in a particular project, the fast pace of change in dynamic working environments creates constant opportunities to reduce overall IT spend through continuous activity reviews.

Find the Low-Hanging Fruit – Standard Template-Based Applications

The concept of design inheritance will be an important topic throughout your project journey. The second way already introduced us to the general topic and the impact it has on the growth of Notes/ Domino environments. Design in general is of great significance during migration and modernization projects. Its importance may start as early as the consolidation phase.

Identifying databases that inherit design from standard templates is important for a couple of different reasons. From an organizational perspective, it's comparatively easy to understand the purpose they serve, even without knowing the business process they are involved in. For example, a document library will be used for exchanging files, archiving documents or something along those lines. No matter if they are used in departments Procurement, Controlling or Manufacturing. Getting that level of insight into custom developed



applications can be quite a challenge if those applications have been evolving over years. This is especially true if you are not just looking at one or two, but hundreds or thousands.

From a technical perspective, it's immensely valuable to know what technical features such standard databases provide. It will provide a perspective on whether it's feasible to use them via the web or make them available on mobile devices. Especially since HCL is investing a lot in modernizing classic templates, that might be a huge gain in a modernization project. In migration it can also be very profitable, because for most of the standard templates, there are ISVs who offer standard migration paths to other platforms. They can almost be excluded from the scope of manual migration work which in turn influences various KPIs like "users active" or "DBs used on server". Whatever the path, working with standard template-based applications will be way less expensive than transforming custom developed applications.

Now, Reap the Rewards

Let's combine those topics and apply them to your consolidation project to sum up how you can best

benefit from the findings. We can even create a small checklist to use as a step-by-step guide through the most important things to consider:

Task: identify unused applications

- Impact: prepare reduction of server utilization
- Impact: reduce project scope

Task: identify applications based on standard templates

- Impact: minimize migration/modernization effort
- Impact: discover user license cost reduction potential

Task: identify inactive users

- Impact: discover user license overspending
- Impact: prepare reduction of server utilization

Task: identify under-utilized servers

• Impact: discover server license cost reduction potential

Task: review Domino network design

 Impact: discover server license cost reduction potential Seeking Application Owners? Find Application Stakeholders Instead!

Stop Looking for Your Application Owners

Are you involved in a project where you are deciding the future of your Notes/Domino applications? If so, you know it's a daunting task where the decision you make will have repercussions for your entire organization.

It can seem impossible to find the information you need to make fact-based decisions. For most

organizations, the number of applications in the Notes environment is huge, having grown over years or even decades. As business structures and processes change, apps that were important 10 years ago, might not be relevant anymore. The original application users have moved on. It's often impossible to know who originally commissioned a given application and who, if anyone, is currently using it.

Many project managers in this situation try to find the application owners. The logic being these are the



people or departments who know the application. They should be able to decide the future of their applications. As owners, they might also be allocated any project development costs associated with processing them. Unfortunately, application owners can be hard to find, and they're often not much use when you do.

In way #4 we want to explore a better way to help you make decisions.

Owners Are Hard to Find and Probably Can't Help You

How do you find them your application owners? What if that department doesn't exist anymore, or the original owner has left the organization?

Even if you can find the name of the person or department who originally commissioned the application in some ancient records, you don't have a guarantee that will solve your problem. Business processes may have changed. That department might not use the application anymore, but a different one does.

Who Are Your Application Stakeholders?

Application stakeholders are the people who currently use and derive a benefit from an application. If you can answer the following questions, you'll know who they are:

- Who are the main users?
- To which departments do they belong?
- Where are they located?
- Who creates content in the application these days?
- Who's consuming that content?
- The answers to these questions are far more powerful than knowing the name of a department head or developer from 10 years ago.
- Once you know your stakeholders, you'll also know:
- Who will be impacted by any changes you make to an application?
- How will they be affected?
- Who should you consult before you start making changes?
- Who is receiving the benefit of the work you are doing?

Strengths and Weaknesses of Traditional Approaches

Here are four potential sources for finding your application stakeholders. Each has advantages and disadvantages.

1. Database ACL

This is often the first place many companies look. It has the advantage of being consistently available for all applications without additional configuration or software installation. In addition, the data from ACLs is relatively easy to gather and process.

Unfortunately, there are numerous disadvantages that can lead to faulty assumptions. There's the risk the ACL contains outdated information. Just having access rights doesn't imply actual usage. Knowing why someone is in an ACL is rarely available. The simplicity of use is outweighed by the superficiality of the information it provides.

2. Custom database inventory

Many companies have built their own applications to collect and maintain information on databases, owners, cost centers, etc. If well maintained, these can provide easy access to very useful information for you.

However, if not well maintained, information may be outdated or incorrect. If you are fortunate enough to have access to such an inventory, how do you know if the information it contains is accurate? Again, you face the risk of using faulty data for critical decision making.

3. Content metadata

Document metadata records are another possible source of information. Fields like "last modified by" can be very revealing when deciding on who are stakeholders or content creators.

Although these fields are dependable sources of accurate information, they are hard to collect and combine across multiple replicas. They're also limited to revealing only the editors. Notes doesn't store the last reader. You won't have an accurate picture of all the beneficiaries of the application. Without this information, you cannot understand the full potential impact of your decision making.

4. User activity

Finally, you can look at gathering usage from various sources like log.nsf or user activity logged by the database itself. If collected for an adequate time period, the quality of the data is high, and it is based off actual user activity. You can also distinguish between Read/Write activity and differentiate between Notes Client and web usage. This allows you to make the best possible decisions, because you will have a very good idea who are content creators and consumers and how they use the application. However, information is not trivial to collect and combine across multiple replicas. The barrier is substantially lower than the one for gathering content metadata, though. Thus, we see user activity as the most promising data source. It's a good compromise between the quality of the information you receive and the effort you have to invest into gathering it.

A Better Way to Find Your Stakeholders

When working with data from usage-based sources like user activity and content metadata, there are a number of KPIs that are available. Not all of them apply to every data source, but in general they are very helpful if you want to make decisions based off numbers. So, take what's available from the source you pick.

When looking at document Reads/Writes, you easily get a quick activity overview. You can differentiate between content creators and consumers. These are the people who understand the value of the application to the business. They are the ones who will be affected by any decisions you make, and you would be wise to consult them before forming an opinion. Additionally, understanding who derives the benefit from an application allows you to assign project costs accordingly.

Also, take a look at the sessions/transactions to understand the impact your decisions will have on server load. This will have a direct impact on your consolidation potential, which we discussed in way #3.



Overview 🕀	Replica Set Activ	vity Lookup by De	partment/Location	HI Explore							
Catalog 🌐	Title or Replica ID										
Usage 🛛 🕀	demo 🗡										
Sessions vs. Complexity	List of replica sets mate	hing the filter expression.									
Web vs. Client Usage	Database Title		User Access Days User	s Active Dep	partments Active Locati	ons Active Last Access	Last Write Access	Sessions Web	Sessions Client	Reads Client	Writes Client
Notes Client Usage	Process Management Dem	no (C12581DF0075AF71)	1,078	31	9	9 Less than a month ag	o Less than a month ago	0	72,936	3,872,933	10,442
Web Browser Usage											
🛤 Usage Overview	List of departments with	activity on the selected replic	ca sets. Hint: Export contains	additional inform	mation.	List of locations with activi	ty on the selected replica sets.	Hint: Export contai	ns additional info	rmation.	
M Usage Category KPIs	Replica ID	Department Name	User Access Days	Users Active L	ast Access	Replica ID	Location Name	User Access Days	Users Active	Last Access	
M Department / Location	C12581DF0075AF71	Controlling [ID: 4]	543	3 L	ess than a month ago	C12581DF0075AF71	Boston [ID: 25]	566	4	Less than a mo	nth ago
Overview	C12581DF0075AF71	HR US [ID: 10]	246	7 L	ess than a month ago	C12581DF0075AF71	The Hague [ID: 10]	186 🛑	3	Less than a mo	nth ago
M Department Activity	C12581DF0075AF71	PR Social Media [ID: 13]	91 🗖	7 L	ess than a month ago	C12581DF0075AF71	Amsterdam [ID: 24]	124 💼	5	Less than a mo	nth ago
M Location Activity	C12581DF0075AF71	HR Europe [ID: 9]	65	4 L	ess than a month ago	C12581DF0075AF71	Vienna [ID: 3]	74 •	3	Less than a mo	nth ago
Replica Set Activity Lookup	C12581DF0075AF71	Sales US [ID: 12]	61 •	4 L	ess than a month ago	C12581DF0075AF71	London [ID: 12]	71 •	6	Less than a mo	nth ago
M Replica Set Activity Lookup by User	Activity history for the s	elected replica sets. Hint	: Select a small number of replica	sets to make thi	is chart easy to read.						
M User Activity Overview	10 - 2010 - 2010 - 2010							and a second			
H User Activity Overview (Focus DBs)	8										
M User Activity Lookup	 ctive Use 9 										
Design 🕀	2						a anna				
Custom Insights ⊞	boot 1.2250 Boot A.A.D.	P Rotal Parts Barts 2019 Way 2 201	9 Way 12, 2019 Way 19, 2019 Way 20, 2019	un ^{2,2019} un ^{9,2019}	1 unth 2019 un22 2019 un90	.2519 1617.2519 1614.2519 16172.25	19 1478-2019 1556-2019 1567-2019	Puer 8,2019 puer 5,2019	549-1-7019 549-8-7019	500 ^{15,2019} 500 ^{22,7}	019

In addition to knowing who is using an application, it is important to understand how many people are using an application. This is particularly true if you can track usage over a period of time. Is the application becoming more or less popular? Is it only used once a year but to do something that is business critical? Which departments are proportionally consuming the application the most?

Stakeholders Give You Better Outcomes

Once the data has been gathered and the calculations are done, it's time to put that information to work for you.

Knowing your stakeholders and their application usage patterns allows you to identify the business units you need to speak with about the future of each application. They can tell you the value an application brings to a business. They will know the impact a change to that application will have on their ability to carry out their duties.

You will also be able to apportion and discuss the project costs the business units will incur for the applications they are using. They will be able to make an informed decision whether an application generates enough value to justify the expense of maintaining it. You will have a logical, evidencebased method to explain why departments will be assigned costs and how those costs are allocated according to the benefit received from them.

Involving application users in the decision-making process will ensure the needs of the business are being fully considered and incorporated into your project. The project will be more efficient. Outcomes will be better and relevant business units will be have bought into the necessity and logic used for determining the future of their applications. Way #5

About the many roadblocks in your source code

A Foreword for Non-Developers

The original title of this could be

"Don't end up in a ditch because you weren't aware of roadblocks in your source code".

We want to share **Why, what and how one should analyze** (in) your Domino applications.

If you are not a developer: The following introduction may help you better understand the rest of this post.

Every database in your environment contains design elements and code. This is to display all therein documents. Be it for creating, editing or reading them.

Views display your documents in list format. These may be sorted and/or categorized. For example, by date, username or whatever fits the respective application and view. Besides views, two more design elements are worth highlighting. These support you to create, edit or read documents in your databases:

- Actions in views (usually available via buttons or menu options at the top of views)
- and Forms, as well as fields on forms

A database can consist of many more types of design elements. Each of them can use one or many of the following programming languages:

@Formulas, LotusScript, JavaScript and Java.

That code again can have a variety of interfaces and dependencies:

- other Notes/Domino databases,
- other applications, such as Microsoft Excel or SAP,
- the file system be it on clients or servers –,
- the operating system,
- and many, many other dependencies.

An application may comprise one or many databases and be available on one or many servers. Replication keeps the same databases in sync across servers. This results in great performance, load balancing and high availability across geographies/networks.

- When looking at optimizing, modernizing or migrating an application, you must know:
- how complex an application is
- (think types and number of design elements, as well as lines of code),
- what all the code does,
- and whether the code will be easy to work with in your respective project.

Let's Start with Why

Most developers did not develop all applications by themselves. Even if they did (chapeau!), it most likely didn't happen "all yesterday", but over a couple of years. Do you still remember the code in all your applications well enough?

Also, some companies don't even have a Domino developer anymore. Let alone all who helped build today's application landscape throughout the years.

Next, knowing the source code of an application is great for running/maintaining it on Domino as is.

Finally, you want to find all the roadblocks, as well as helpful code, if not in general the good, the bad and the ugly. If you want to optimize, modernize or migrate your applications:

All in all, wouldn't it be great if you could save yourself valuable time, frustration and pitfalls?

Knowing the stakeholders of your applications, becomes far more powerful when combined with analyzing the design and code of your applications.

Once you know your stakeholders, you'll also know:

- Who will be impacted by any changes you make to an application?
- How will they be affected?
- Who should you consult before you start making changes?
- Who is receiving the benefit of the work you are doing?

Which Brings Us to the What

Before we dive into the "What of your code should you analyze", please remember:

There are many other important datapoints to consider before analyzing code:



- Which are your most used and least complex applications?
- These pay off fast. And you should not spend time with applications nobody uses! For more details, also see here and here.
- Which end users need local replicas for performance or offline usage reasons?
- Which applications do your VIPs or your most important profit centers use?

These are just a couple of examples to keep in mind.

Now What Code Should You Analyze, and What Should You Look for Therein?

All of it. Period. In *all* your applications. And for each of them, that means

- All code: @Formulas, Java, JavaScript and LotusScript
- All design elements (think "code containers"). These can be forms, subforms, views, columns, actions, agents, buttons, script libraries etc.

 The following design elements often deserve special attention: XPages, Java Classes, Applets, Jar files, Web Services, Composite Application features, and similar

Proper analysis of both application design elements and code answers two essential questions:

- 1. Where does all your code live? How much is where? And what type of code is it?
- 2. What does that code do?

The following two examples show the value of analyzing both design and code:

a) The more forms, fields and code an application has, the more time consuming it will be to modernize or migrate it

b) An application using Java Code is not an ideal candidate to migrate to SharePoint. Yes, this partly depends on what the corresponding code does. Yet, it helps you to rank your applications and prevent pitfalls

WHAT You Should Look for In Your Code

Searching in code can be anything from fun to frustrating.

After <u>exporting the design of your apps to DXL³</u> (Domino XML Language), here are three tips to get you started:

- Fire up notepad or similar and explore the result. To get a first feel, try and find some of your code by searching for parts of it. Also, search for usernames, servernames and similar
- 2. Try LotusScript Manager or Source Sniffer from OpenNTF
- (do *not* use Lotus Analyzer! It has hidden design and phones home)
- Pro-Challenge: Slice the code from the DXL into documents in a Notes database. This facilitates categorization, searching and post-processing (to count design elements, for example)

If you tried any of the above, you may have noticed a couple of shortcomings on the way:

a) Your searches also match comments/remarks in your code

b) Combined searches like @Db(Column OR Lookup) call for tackling above pro-challenge first

(=slicing the code into Notes/Domino documents or a SQL database). Separate searches lead to duplicate results. This in turn works very much against your aim to review as few code as possible.

c) Your searches also match code you were not looking for. For example:

- Searching for "Open" finds NotesDatabase[Object].Open and NotesStream[Object].Open
- Searching for "@DbLookup" includes lookups into the same database where the code lives.

Yet, you might only want lookups to other/ external databases. Or, your result also includes non-Notes lookups, like ODBC. Yet, you might only seek Notes lookups.

So, Before We Continue Our Search, We Must Optimize the Code

For good search results we must remove any and all comments from all code. Yes, comments are good to have in your code to better understand it again later. But they skew our search results.

The following pictures illustrate how a developer can comment in LotusScript and @Formulas:

- %REM ... %ENDREM
- (no)code '(apostrophe)
- REM
- (no)code : REM

In @Formulas comments begin with a REM, followed by any amount of whitespace (=blanks or tabs). Next comes the actual comment surrounded by either double quotes or curly brackets.

- REM "comments";
- REM "comments"
- REM {comments} ;

We've Already Prepared the Above for You to Test for Free

This entire article helps you understand and perform your own, independent application analytics. Yet, you may want to save yourself valuable time and get moving fast. All you have to do is register for our ready-to-play iDNA Applications sandbox online. As soon as you've registered, all you need to do is login and navigate to our ready-made instant code search. Granted, it's not showing your very own applications, but gives you a good idea of how code search should work.

For testing, use text search using "florian", "vogler", "server", "/acme", "/O=acme", and "workflow". Also, try a regular expression search like "(?iw)@db(lookup|column)".

Now Let's Really Dive Into WHAT to Look for *in* All the (Design and) Code of Your Applications

If you want to optimize

- Search for GetNthDocument it's slower than GetFirst/GetNextDocument
- Search for hardcoded servernames, usernames, database filenames, IP-Addresses, email addresses, replica ids, etc.

- Search for old code (@V2If, @V3Username, @V4UserAccess, @UserPrivileges, @IfError)
- Independent of code: Find your most used/ popular databases and give them a little love. Make them prettier and more modern!

If you want to modernize

- Check out whether an application heavily relies on NotesUI* classes. That doesn't work in web browsers and needs some rework Check for code that's not supported in browsers. Hint: search Designer help for "You cannot use this function in Web applications."
- Is there already a lot of code suggesting browser support? i.e. @WebDbName, @BrowserInfo, @ClientType, Domino @DbCommands, ...
- Does your application/code rely on printing? That can be challenging from a browser
- Analyze your code for whether it works on HCL Nomad
 - Does the application depend on XPages, Java or ODBC? This does not work on HCL Nomad.
 - Does the application use C-API calls? If yes, does the code also work on iOS and Android?
- Does your application require any 3rd party client extensions?
- Independent of code: find your most used/ popular databases and give them a little love. Make them prettier and more modern!



If you want to migrate

- How many forms, fields, views etc. does an application have? Don't pick the most complex first.
- Does your app depend on code or design elements that do not work well in your respective target platform?
 For example: Java <> SharePoint, too many Folders <> SharePoint. C-API. Private or public Addressbook, Mail (Send and) Encrypt. UseLSX, ODBC, DB2, DOS/cmd, OLE, files, directories, MIME, document links, etc.
- Does an application depend on other databases?

Think @DbLookups that connect to other databases (not using e.g. "":"" as server:filename, for example). Same goes for New NotesDatabase, GetDatabase, .Open, .OpenWithFailover, .OpenIfModified, .OpenByReplicaID, [FileOpenDatabase], [Compose] etc.

• Independent of code and migration: give one of your left-over databases a little love, make it prettier and more modern.

Finally, Let's Look at HOW to Best Search Your Code

The following is really, really creepy stuff. If you are not a developer, feel free to skip this chapter!

In part, we've already covered why searching for code with just (sub)strings isn't going to get you all too far. It matches too much or too little in too many cases. We also looked at why removing all comments is necessary before searching your code.

Also, the following is very, very, very important when searching for code:

Do NOT search for code thinking of how you would have developed it. Think broader and you will be quite surprised to learn how other people code.

So, what are better approaches for searching for code than just substring matches?

A fulltext-index supporting wildcards like "*" (asterisk) gets you somewhat further in the game. E.g., when searching for "@dbcolumn* OR @dblookup*" – but it lacks support for precise negation. Accurately negating code parts is vital to e.g., only find @DbLookups that do not point to the same database.

The following @DbLookup looks up data from the same database that the code is executed in. It does so through the "":"" as the second parameter:

@DbLookup("":"":"view":"lookupkey":2);

The next @DbLookup looks up data from the "local" addressbook (client or server, depending on where it runs):

@DbLookup("":"":"names.nsf":"view";"lookupkey";2);

Searching for any @DbLookup that looks up data from "names.nsf" is pretty easy. Even with just wildcards: @dblookup(*:"names.nsf"*). That's until you come across code like this:

```
@DbLookup("":
    "":"names.nsf";
    "view":
    "lookupkey";
    2);
```

Now we suddenly need to search for code across multiple lines – yes, we've seen code like that.

Where it gets even worse is when variables, let alone @functions, come into play as parameter:

The above code looks up data from the same database. The variable ht_filename is the result of @Subset(@DbName;-1). This again results in the filename of the database in which the code runs.

ht_server:=""; ht_filename:=@Subset(@DbName;-1); @DbLookup("";ht_server:ht_filename;"view";"lookupkey";2);

Similarly, the following code example looks up data from the local addressbook:

ht_server:=""; ht_filename:="names.nsf"; @DbLookup("";ht_server:ht_filename;"view";"lookupkey";2);

The best solution to searching code would be if one could parse the code. This would allow us to resolve the value of ht_filename in above examples. However, we have not found a smart solution for this.

What we did find a smart solution for is searching for code with precise negation:

We use regular expressions.

The following regular expression is a huge step forward. It allows us to look for any @DbLookup that does not lookup data from the same database where it runs from:

@dblookup\([^;]*;(?!""(:"")?).*

@dblookup(The open bracket needs to be escaped in regular expressions, hence the \(
[^;]*;	followed by "anything but a semicolon" until the next semicolon. Searching for ".*;" would be wrong since that expression would be greedy. It would search until the last semicolon.
(?!""(:"")?)	Next negating "", followed by another optional :"" – the second parameter can either be "" or "":""
.*	until end of line

This example still requires some fine tuning to

- also support any amount of whitespace pretty much everywhere,
- and cater for situations where @dbname or @subset(@dbname;1):@subset(@dbname;-1) use the same database
- and only find those @DbLookups for which the Class is "" or "Notes" (case insensitive)

As bonus points, you may also want to find @dblookups used in LotusScript Evaluates. Oftentimes, quotes need to be escaped then, too. The regular expression which matches all the above requirements (except for code parsing) is ... possibly the ugliest thing you will get to see today:

(?iw) @db(column|lookup)\s*? \((\s*?\\?"notes\\?"|\s*?\\?")[^;]*; (?!(?!\s*?\\?".+?\\?"|\s*?\S+\s*?:\s*?\\?".+?\\?")| \s*?\\?"\\?"(\s*?:\s*?\\?"\\?")?\s*?;|\s*?@dbname\s*?;| \s*?@subset\s*?\(\s*?@dbname\s*?;\s*?1\s*?\)\s*?: \s*?@subset\s*?\(\s*?@dbname\s*?;\s*?-1\s*?\)) [^;]*;[^;];[^\))*\)

I Could Go On and On for Hours

If you made it through the really creepy stuff in the last chapter: my hat is off to you! You are a superhero developer and regular expression survivor.

The good news for all readers is: We at panagenda have already done a good deal of the heavy lifting. And, we love sharing.

If you take a look at our iDNA Applications sandbox, you will find more than 70 ready-made regular expressions. These search for over 300 different code findings. You can use all those patterns in your very own code search solution for free.

And just in case you have a smarter idea than using regular expressions or come up with some new or improved regular expressions: Please let us know and we will in turn share with the community.

In case you don't have the time to build your very own code search solution:

There are a number of 3rd party solutions out there that can help. Some of them are available on OpenNTF. Some of them are commercial solutions like our very own panagenda iDNA Applications.



Summing it up

Properly analyzing your applications serves three primary goals:

- Saving (a lot of) time, frustration and money
- Focusing on the right things and being in the know
- Enabling the successful transformation of your Domino landscape

Optimization, modernization or migration

Whatever you do don't forget to give at least one of your Domino applications some love. It will pay you back. Many times. Way #6

3 New Ways to Automate Your Progress Reporting

Let Management Dashboards Replace Your Progress Reporting

Progress reports. They're a part of every project and they're not fun to produce. It's not just your progress you must share. You must also supply the project teams with the data they need to do their job. It's tough to coordinate and it never ends.

Let Analytics API's empower your teams with selfservice access to the data they need. A one-off set-up frees you from the drudgery of manual reporting and ensures your colleagues have continuous access to status reports that are always up to date.

Get back to the work you really need to focus on.

Business Intelligence and Data Exploration Tools Can Make Your Life Easier

We hear the same thing over and over from project managers: "Creating progress reports is an endless chore, takes a lot time and is usually required at the worst times imaginable!".

There is an easier way.

Using business intelligence and data exploration tools, you can gather, compile and make information available in dashboards tailored to the specific needs of various consumer groups.

Your entire organization will benefit from this selfservice approach.

The reports you create using this solution be:

- Less time consuming and cheaper to produce,
- automatically always up to date, and
- available on demand.



• You can present this information in a number of useful and interesting ways. You can use dashboards, charts, and even a new breed of interactive reports.

To create your reports, you could choose from a number of purpose-built tools readily available in the market. In practice, you will use something that's already available within your organization. If you are lucky, it will be one of the solutions mentioned in the <u>"Gartner 2020 Magic Quadrant for Analytics and Business Intelligence Platforms"</u>⁴

3 Ways to Produce Reports Using 4 Common Tools

Without needing a corporate purchase decision, you probably already have access to some combination of the software below. If so, read on!

Solution 1 – Microsoft Excel & PowerPoint

Admittedly, this combination is by no stretch of the imagination a BI or dashboarding tool. However, it's fair to say that with its Pivot tables, charts and data source capabilities, Excel has a lot to offer when it comes to data analysis and exploration. Combine that with the presentation capabilities of PowerPoint, and you might be surprised how close you get to automated reports.

This combination offers two major advantages:

- 1. It is readily available in most organizations, and
- 2. there are usually a number of people with a high level of expertise.

Data Source

In this example, we use an Excel file exported from iDNA Applications. However, the techniques we show will work for all kinds of other sources as well. Working with Excel sheets that are not connected to the original data source makes it significantly harder to keep the reports updated. Still, we'll use this example to showcase the simplest variant. N.B. – Recent versions of Excel and Power BI have a significant overlap in their feature sets and usability when it comes to importing data from external sources. If you would like to link your spread sheets to live data, please follow the instructions in the Power BI chapter before you continue to the next chapter.

Data Exploration

The content of our data set is a list of Domino database instances. Each has a variety of properties like title, server, file name and database type. Our goal is to find out how many of them are applications and how many are mail-in databases. We are looking to display the data using a visualization that will show the relative size between the two categories.

The first thing we do is insert a Pivot Table (PT) based on the data. To do that, we select a cell in the table and click "Pivot Table" on the "Insert" ribbon.

AutoSave 💽 🗄 りゃ 🦿	
File Home <mark>Insert</mark> Page	Layout Formulas I
PivotTable Recommended Table Pio	tures Online Shapes Icons Pictures ~ Illustra
PivotTable Easily arrange and summarize complex data in a PivotTable.	Process Manageme
FYI: You can double-click a value to see which detailed values make up the summarized total.	🗖 file, par
⑦ Tell me more	f6e261f
3 dev01/acme	abc2f46
1 appmail01/acma	hfc70-2

In the PT Fields configuration interface, we drag "database_type" to Rows to create our categories and "Replica ID" to Values to count it. The "Count of" operation is selected by default. That will show us the number of DBs per category.

	Search	2
	server_name file_name Title Replica ID size_bytes documentcount	
	template_inherits_from template_acts_as_master design modified Drag fields between areas below: Filters	▼ ■ Columns
	Rows database_type *	∑ Values Count of Replica ID ▼
· · · · · · · · · · · · · · · · · · ·	Defer Layout Update	Update

Since we are only interested in Focus Databases, we drag the "is_focus_db" field to the Filters box and then select "true" in the filters dialog above our results table. That will give us the results just for the database types we are interested in.



Data Visualization

Now, we have the data points we want. Next, let's create our first visualization. Select a cell of the Pivot Table and then click on the pie chart symbol on the Insert ribbon. There we have the chart we were looking for: a representation of how many Focus DBs are Applications and how many are mailin databases. Now, right-click the chart and select Copy to get this in the clipboard.

Pivot	Table Recommended Ta PivotTables Tables	able Illustrations Add- ins *	Recommen 2 Charts		Maps Pi	ivotChart	3D Map ~ Tours	Sparklines
A5	• : X	√ <i>f</i> _* Mailin Dat	abase					
	А	В	С	3-D Pie		*		
1	is_focus_db	true J	-			Pi	votTab	ole Field
2						Ch	oose fields	to add to repo
3	Row Labels 🛛 👻	Count of Replica ID		Doughnut		Sea	irch	
4	Application	529						
5	Mailin Database	47		(\mathbf{Q})			size_bytes	count
6	Grand Total	576	i				template_i	nherits_from
7				More Pie Cha	irts		template_a	icts_as_maste
8	_	_			-	┱║┣	design_mo	dified
9	o is focus db♥	0			-0		is templat	e candidate

In the next step we'll need to open PowerPoint as we are using it to compile our report. Here we paste our chart by right-clicking anywhere on the canvas and selecting an option with "Link Data". Whether you use the destination or source theme is up to you.

N.B. – Linking is very important here, because embedding will break the connection between the Excel sheet and the PowerPoint presentation. This disconnect might be desirable in some situations, but it won't provide the functionality we need.



The chart is now a PowerPoint object you can freely style and format according to your taste, needs or corporate design.



The most important thing about this combination though, is that data changes in the Excel sheet will influence the visualization in PowerPoint. If you now want to create a monthly report, for example, all you need to do is update the excel sheet with the correct database list. The pivot chart will reflect those changes and so will the chart in PowerPoint (a manual refresh might be required). If you created a dozen charts and tables based on the Excel sheet, all of them will update with the data changes. That's why this solution is so powerful.

The question then becomes how you share your report. We have seen customers create PDF files from the PowerPoint presentation and share them in HCL Connections or SharePoint. Of course it's an option to simply send the file per email, but that would put you in the situation again where you have to "deliver" the reports to someone instead of putting them in a central location where your stakeholders can go and get the information from.

Solution 2 – Microsoft Power Bl

With Power BI we have a great tool in hand for the job we're trying to accomplish. Its capabilities around data analysis and exploration, as well as dashboarding, are top notch. They are leaders in Gartner's Magic quadrant on Analysis for a reason.

Microsoft managed to strike a beautiful balance between powerful functionality and ease of use. A lot of its data source integration and data manipulation features are very similar to Excel. This makes for an exceptionally smooth learning curve.

"Power BI Desktop" is free of charge. There aren't a lot of limitations compared to the Pro version. However, you cannot share your content with coworkers unless you share the whole Power BI file. Even when using the pro version, information consumers need a Power BI license unless your dashboards are publicly available on the internet. For many organizations, that is not an option. Still, it's still a great solution for our needs, even though we have to share the result as if it was just another Office document.

Data Source

We'll gather our data from panagenda iDNA Applications again. This time around though, we'll create a live connection to the data using DataMiner, iDNAs analytics API. Our whole dashboard will update automatically if data changes in the backend. The same method can be used in Excel, since the data source features are very similar.

🝱 panagenda	DataMiner × +						
\leftrightarrow > c	🕈 🔂 🖉 🔒 htt	:ps://ifa-sandbox .panagenda.com /dm					
≡							
		G		► ► <mark>5</mark> 2	Copy I	Public API URI (by Id)	
panage	enda DataMiner	QUERY INFO	EXECUTION OPTIONS	Copy to clipboard QUERY C	c <>	JSON 3	GRID
Query View	er	Options School uto this quark for outcomption	aightly data proporation	1 select * from cq		CSV	
	Samples ^	Schedule this query for automatic i	ightiy data preperation		Copy I	Public API URI (by Name)	
<>	DB activity by department 😙	Materialized View			<>	JSON	
					=	CSV	
<>	List of databases 🕲	Generated on: March 26, 2020 5:35 PM (22 days a	go)		Copy	Tableau WDC API URI	
	A simple demo query Simple query on cax data	Last update on			<>	by Id	
<>	List of replica sets	March 26, 2020 5:35 PM (22 days a	<u>go)</u>		:=	by Name	
		€ REFRE	SH NOW		Copy I	Materialized View URL	
<>	List of users				<>	JSON	
					:=	CSV	
<>	Query with parameters A demo for using placeholders						

In DataMiner, select the query you want to link to and click the clipboard icon to copy the URL to the query result in JSON format. Both Power BI and Excel can work with it natively and you don't have to worry about delimiters in DB titles, like you would have to with CSV.

8 9 9		Untitled - Power Bl Desktop	
File Home Ir	nsert Modeling View Help		
Paste Copy Paste Format painter Clipboard	Get Excel Power BI SQL Enter Recent data v datasets Server data sources	Transform Refresh sv datav Queries Insert Calculations Share	
	Excel Power Bl datasets Power Bl dataflows SQL Server Analysis Services Text/CSV Web OData feed	From Web • Basic O Advanced URL https://ifa-sandbox.panagenda.com/dm/v1/api/query/byld/2/json 3 (Cancel	
	More	Anonymous Windows Basic 5 Web API Organizational account Back Back Content Con	

Back in Power BI click "Get data" in the Home ribbon and select "Web" as the source. You will be prompted to enter a URL next and we can simply paste the one we just copied in DataMiner. After selecting Basic authentication and entering credentials, we can finish connecting to the data source.

After the connection was established successfully, the Query Editor will open in a new window. The first thing we need to do is click the "To Table" button in the top left corner to convert the list of records into a table we can later work with. Just accept the default options in the pop-up window and click OK.

	PROPERTIES
File Home Insert Modeling View Help	Database List
	All Properties
List Tools Paste Pas	2 OK Cancel
3 Record 4 Record	



Now that we have a table, let's give it a proper name, so we can identify it later, in case we have multiple sources. Then we click the small icon on the top-right of our first column that looks like to arrows pointing

File	Но	me li	nsert	Modelin	ng N	View Hel	p					
ے۔ aste	Cut	 🖯	- P ∓ Unt	itled - Pow	er Query	Editor	A			ſĿ I	. 4	
	Serm 🖓	File	Home	Transfo	orm	Add Column	View To	ols Help				
ป	A The	Close &	New	Recent	Enter	Data source	Manage	Refresh	ditor Ch	oose Remo	J C E E E	eep
≣		Close	Source	New Query	Data	settings Data Sources	Parameters • Parameters	Query	Colu	imns • Colum anage Columi	ns∓ Ro as P	leduc
		Querie	s [1]		<	ABC 123 database	id 👻	ABC 123 server_name	ABC 123 set	rver_id	-	A5C 123
		🛄 Data	abase List		1		3935	appmail01/acme			19) f6e
					2		3942	dev01/acme			32	abc

away from each other. With it we can expand the JSON record into individual columns. I recommend unchecking the option "Use original column name as prefix", as all our columns have unique names and we are not expanding more than one JSON field.

Once that is done, we'll see our data in individual columns for the first time and could change data types or even manipulate data already. It's not necessary in our case though, so we simply click "Close & Apply" in the top left corner.

Data Exploration and Visualization

Back in the dashboard view we click "Pie chart" on the Visualizations area. After selecting the new object, drag the field "database_type" from the Fields list on the right to the Legend box (2) to specify the categories for our chart. Within those categories we want to count items. We accomplish that by dragging the "replicaid" field to the Values box (3).



We have the same issue that we had in Excel though. We see all database types, but we only want to see Focus databases. To accomplish that, drag the "is_ focus_db" field from the Fields list into the "Filter on this page" box in the filters area (might be minimized by default). After that is done, we now have the option to select which items we want to see and we select "true" to see only Focus DBs.

Voilà, there we have our pie chart again. We can see the ratio between applications and mail-in databases!



Of course, that's an extremely simple example of Power BI's capabilities. You can find more information in the <u>webinar that accompanies this article⁵</u>. There we use an additional data set to bring user activity by department into the mix. Power BI allows linking the two data sets and exploring them together.

In this example, we used data from panagenda's own solutions. Of course we could have connected to any compatible data source or even simply started from an Excel file. Regardless of the data source, Power BI makes it very easy to create dashboards and takes visualization and presentation to a new level. Users who open my dashboard will be able to explore not just my dashboard but the data behind it. They will be able to filter, format and drill down into what's important for them. Even though the free version has its limitations and I have to file-share it like the PowerPoint presentation we created earlier, it's an immense step up from "just" a couple of slides with static content.

Solution 3 – Metabase

Here we find a drastically different approach to accomplish something similar to what we did in Power BI.

Metabase is an open-source data exploration and analytics solution that "lets you ask your own questions". It is completely web based and has multiple ways to share your questions and dashboards publicly or with individuals.

One of the key characteristics that we are looking for is the ability to run it on-prem. Even though a hosted Metabase offering is in the works and an Enterprise version is available, for our scenario we can use what the free on-prem edition has to offer.

Data Source

Metabase differs from Excel and Power BI by being deeply intertwined with the database system it sets out to analyze. When you ask your questions in the slick web interface, it creates SQL queries on the fly so the database server does the data crunching. That's a great way of exploring data, especially since the tables and columns can be enriched with metadata at a central point.

Here's an example how you can use this. You would create user friendly names for columns (like "DB Type" instead of "database_type_text") or set up relations between tables. The end-user can easily benefit from SQL JOINs without having to know the data or even what a JOIN is!

There is a down-side to this though. Metabase cannot work with simple files like CSV or Excel as a data source unless you first import them into a relational database. That's not a tough hurdle to cross, but it adds another layer of complexity. If a simple file is your only data source, Power BI might be the more convenient solution for you.

🔅 Metabase Admin				Databases	
DATABASES 🕻 ADD DA	TABASE				
Database type:					
PostgreSQL					
Name					
How would you like	e to refer to t	his databa	ise?		
Host					
localhost					
Port					
5432					
Database name					
birds_of_the_world					



That said, we at panagenda have looked long and hard for a web-based solution like Metabase. Nothing we have found has come close to the wealth of features, simplicity and elegance that it offers as an open-source solution. Beyond that it has a very active community and a healthy developer participation. It is an amazing package!

N.B. – If you use iDNA Applications, Metabase comes pre-installed, configured and ready for you to use! If you don't, a discussion on how to set up Metabase is beyond the scope of this blog. If you wish to know more, you can look here:

- Setup and database connection
- Getting started and firs steps
- (Optional) Enriching meta data

Data Exploration and Visualization

To start out in Metabase we can simply click the "Ask a question" button in the top menu bar. It will present us with three options:

- Simple question: an easy way to explore data with simple grouping, filtering and visualization functionality.
- Custom question: a notebook-like editor to join data, create custom columns and perform advanced operations like calculating custom columns.
- Native query: allows entering your own SQL queries for even more advanced operations like e.g. working with window functions.

We'll just go ahead and select a simple question. We can convert it into a custom question later or view the SQL behind it if we need to.

Q Search		+ Ask a question 🏭 Browse Data + 🔚
		0
		1 SHLECT 2 avg(d(st) as 'Pistance',
) (d 4 FROM dest_table 5 WHERE (dest_value + "qvinate") 4 ORDELTY (d. REC
Simple question	Custom question	Native query
Pick some data, view it, and easily filter, summarize, and visualize it.	Use the advanced notebook editor to join data, create custom columns, do math, and more.	For more complicated questions, you can write your own SQL or native query.

After selecting the "Database" table as our data source, Metabase shows us some of the content to get a feel for what the data looks like.

Q Search	Q Search			+	Ask a question	Browse Data 🕂 🕻	3
Pick your data	panagenda	• cqx_data • Da	tabase		Save \Xi Filter	+ Summarize	Ċ
^	Database ID 🗸	ReplicalD ~	File Name \vee	Collection State 🗸	Database Type 🔻	✓ Design Complexity Sco	ore
🕻 panagenda	3935	C12581DF0075AF71	f6e261f69695bccb1a.nsf	Full (Insights)	Application	276,	,908
Q Find	3942	C1257AB7004AA24F	abc2f463aefb8912e5.nsf	Limited (Complexity)	Application	15,	,630
	3157	C1257C0B003F068D	bf678a3079e3273360.nsf	Limited (Complexity)	Application		4
Code Cluster	3093	C12582C900798414	dae44c275848b59d08.nsf	Full (Insights)	Application	20,	,408
Code Element	3114	C1258266003DCF96	5904b188f60e8b4d8c.nsf	Limited (Complexity)	Application		3
Code Element Summary	511	C125818100691F44	c8f617e52c575da0d7.ntf	Not Collected	System Database		
Code Hash Set	4016	85257EEC0046812B	44524bd5ea19f1e723.nsf	Limited (Complexity)	Application	18,	,345
Code Similarity Link	3207	C12582CD003CD938	d881f44ceb906d13ee.nsf	Full (Insights)	Application	6,	,867
Dashboard KPIs	487	85257E4B003E8ECF	a7de3108a2cc2157c6.nsf	Limited (Complexity)	Application	4,	,177
🗄 Database	3191	C1257FF1003893F7	d50d8f8bf26c277a1c.nsf	Limited (Complexity)	Application	15,	,642
Database Basic	441	C12576C8003181DE	8e73b8ab5449202c91.nsf	Limited (Complexity)	Application	40,	,657
Database Inventory	2898	852580A30048D1EB	e196cdf4dbc1c26b17.ntf	Not Collected	Application		
DB Instance Usage Overview	527	C12582EC0029183A	d96a43edb7485fb2a7.nsf	Limited (Complexity)	Application	11,	,514
Department	3108	C1257C31004AEB79	dfa83aefd7bdd909b8.nsf	Full (Insights)	Application	7,	,275
Department Application Usage	5172	C12576C80030865E	c3cba9d30dcd82de95.nsf	Limited (Complexity)	Application	161,	,641
• • • • • • • • • • • • • • • • • • •	<						
	Visualization	Settings			Shov	wing first 2,000 rows 📣 .	<u>م</u> رأ

Q Search.						🕂 Aska que	estion 🏢 Browse Data 🕂 🗔 🌣	
panagenda •	cqx_data • Dat	tabase				Save	Ţ Filter + Summarize # Ĉ	
Database ID 🗸	ReplicalD >>	File Name 🗸	Collection State 🖂	Database Type \vee	V Design Complexity Score	Design Modified	Summarize by	
3935	C12581DF0075AF71	f6e261f69695bccb1a.nsf	Full (Insights)	Application	276,908	November 21, 20	Summarize by	
3942	C1257AB7004AA24F	abc2f463aefb8912e5.nsf	Limited (Complexity)	Application	15,630	November 15, 20	Count of rows X	
3157	C1257C0B003F068D	bf678a3079e3273360.nsf	Limited (Complexity)	Application	4	July 6, 2014, 1:48	+ Add a metric	
3093	C12582C900798414	dae44c275848b59d08.nsf	Full (Insights)	Application	20,408	July 14, 2018, 1:0		
3114	C1258266003DCF96	5904b188f60e8b4d8c.nsf	Limited (Complexity)	Application	3	April 5, 2018, 1:1	Group by	
511	C125818100691F44	c8f617e52c575da0d7.ntf	Not Collected	System Database		October 1, 2017,	Q Find	
4016	85257EEC0046812B	44524bd5ea19f1e723.nsf	Limited (Complexity)	Application	18,345	October 28, 2015		
3207	C12582CD003CD938	d881f44ceb906d13ee.nsf	Full (Insights)	Application	6,867	July 17, 2018, 1:1	DATABASE	
487	85257E4B003E8ECF	a7de3108a2cc2157c6.nsf	Limited (Complexity)	Application	4,177	May 20, 2015, 1:2		
3191	C1257FF1003893F7	d50d8f8bf26c277a1c.nsf	Limited (Complexity)	Application	15,642	January 2, 2018, :	Database ID	
441	C12576C8003181DE	8e73b8ab5449202c91.nsf	Limited (Complexity)	Application	40,657	February 7, 2019	Aa Database Ture	
2898	852580A30048D1EB	e196cdf4dbc1c26b17.ntf	Not Collected	Application		January 9, 2017, :	Ad Database Type Add grouping	
527	C12582EC0029183A	d96a43edb7485fb2a7.nsf	Limited (Complexity)	Application	11,514	November 7, 201	Design Complexity	

Here we select "Summarize" (1) and pick "Database Type" (2) as the field we want to group or summarize by.



The result we get is a bar chart the number of entries within each database category. Let's go ahead and change that to a pie chart by clicking "Visualization" (1) at the bottom and then choosing "Pie" (2) as chart type.

We're back at the point where we see the distribution of DBs across database categories in a pie chart (or donut in this case). Again we want to display only Focus DBs. We can accomplish that by clicking "Filter" (1) on the top navigation bar and then selecting the "Is Focus DB" (2) in the field list.



Yay! We have accomplished our goal and see the distribution of DBs between the two database types. From here there's a couple of things we can do with our question. If we click on "Show Editor" (1) we will be taken to the same editor we would have gotten if we had selected "Custom question" initially.



Here we can continue to refine our question or view the SQL (1) that was generated for us by Metabase. We also have a bunch of advanced techniques we utilize here: joining with other data tables, sorting, limiting or even creating custom columns.

Q Search	🕂 Ask a question 💠 Browse Data 🕂 🖸 🖨
Count by Database Type	Save IP
Data	
Database	•
• H ⁺	
Filter	
Is Focus DB is true ×	
Summarize	
Count X + by Database Type X +	
Ξ Σ 3 4 Filter Summarize Join data Sort Row limit Custom column	

Now how do we share this with our stakeholders? It all starts with saving that question (1) and optionally adding it to a dashboard.

N.B. – We won't go into dashboarding in this blog post, but if you would like to see more on the topic or advanced Metabase use cases, please let us know in the comments.

Now that we have a saved question, we get an additional option on the bottom row (2) to share and embed our question.

🕂 Ask a question : 🔛 Browse Data 🕂 💽 🌣		
Save Ţ Filter + Summarize I ⊂ C		
	576 TOTAL	
	E C	& A 11 <mark>2</mark>



With this dialog, we can share (1) our question or dashboard publicly or embed it in a site where we control access. You can also publish (2) a dashboard of overviewdata on the intranet without additional access control if employee intranet is limited. More advanced statistics are embedded (3) in an area with advanced access control.

Even though this method of sharing is great, it's not really needed. You can simply give your co-workers access to the Metabase installation and create custom dashboards for whatever information you want to provide them with.

Make Your Life Easier and More Productive

These are just 3 ways you can simplify your life by automating your reporting. There's so much more you can do with the right tools and a little know-how.

If you'd like to learn more how to make your workday more productive, take a look at our <u>webinars</u>⁶ and <u>blogs</u>⁷. There's a world of information waiting for you.

Way #7 Know Before You Go!

Using Facts to Plan Your Domino Application Migration

Migrating your business process applications off the Domino platform is not the straightforward, cookiecutter process that some might suggest. There are many complexities involved, and pitfalls await those IT groups that move forward too quickly without first understanding all the facts from an in-depth investigation of their Domino environment.

Let us introduce you to a set of best practices that we have learned over the years from assisting customers through detailed assessments of their Domino application deployments. Prior to any risky, high-profile migration planning project it is imperative that the team leading the effort does their due diligence to understand all the facts. Application migration and modernization projects are hugely expensive and can quickly run off he rails by making assumptions and underestimating complexity. So how can you stack the deck in your favor for an accurate Domino application migration project estimate? Read on and you will see.

Step 1 -Shrink the Number

When Domino application assessments begin there is usually a huge number of databases flagged for analysis. Sometimes that number is daunting and scary to consider as part of a complete redevelopment effort. However, that first number is NEVER the reality. The first phase of any assessment



will greatly reduce the number of applications by following this three-pronged approach. More details in the chapter <u>'Navigating your Domino projects</u> with confidence'⁸.

Archive and Decommission Inactive Apps

By analyzing usage and activity data the project team can quickly identify applications that are no longer needed. From years of experience we have found that this number is usually quite large. Most organizations that have run Domino for 10+ years will have 40% or more of their applications in this inactive state. These applications should be archived off and removed from consideration. More details in our previous blog on 'Keeping your cost estimates efficient from the start'.

Group Simple Applications for Easy Migration

The second analysis pass for Domino applications is to catalog the simplest apps that can be quickly transitioned to another platform by migrating the data. There are many target platforms available where these types of applications which can be migrated with minimal effort involved. The other option is to simply leave these applications in place on Domino, accessible through a Web Browser interface. You can even add a little beautification for the UI to spruce up these applications which were built from old, well-known templates such as Document Libraries, Threaded Discussions, or Mailin Databases. More details in in the chapter <u>'Most successful assessment projects start with a</u> <u>consolidation'</u>?.

Group Remaining Custom Applications with Similar Designs

After identifying the inactive and simple apps, the remaining applications will move into this next phase of the assessment. Most organization will have loads of applications that are very similar in design. They were originally built from a series of templates that were morphed over time, but their actual design DNA such as navigation UI, forms, views, security roles, data storage, and custom code segments, plus their business process functions are still very close. More details in the chapter <u>'Get the big picture by comparing your application code'¹⁰</u>.

During a comparative design pass these applications should NOT be considered unique, complex apps which would each require a custom redevelopment effort to migrate them to a new platform. Instead these should be counted as grouped application types to reduce the overall number of apps further. Only one of each design type will require scoping and estimation for redevelopment in the next step.

Step 2 -Use Experts and Facts to Scope Custom Applications for Migration

Now that you have greatly reduced the actual number of applications to consider for a redevelopment and migration effort, it is time to dig in and understand these remaining custom apps completely. Get all the facts and learn the true nature of these applications and exactly what business processes they support.

More details in the chapter <u>'About the roadblocks</u> <u>in your source code'11</u>.

This is best served by combining two separate efforts. Firstly, interview the application owners and power users. More details in the chapter <u>'Finding application stakeholders'¹²</u>.

They will provide all the insights on the complexity of the data, security requirements and process workflows included within the applications. And secondly, perform a detailed forensic analysis of the application designs and datastores by using an automated solution like iDNA Applications. Only by gathering the facts from these two methods will your team acquire the full visibility and knowledge to target the best matching platform for migration and accurately scope the level of effort involved to perform the redevelopment and data transfer.

**Note: when scoping the redevelopment and migration effort be sure to leverage consulting groups with experience on both sides of the fence (Notes/Domino and the Target Platforms).

Recommendation: Keep Complex Applications on Domino until Ready

Custom built, business process applications running on Domino are full of complicated workflows, tailored forms and views, intricate security roles, and multi-faceted reporting functions. They should be scoped individually for redevelopment and migration effort along with the assistance of the department owner and power users. Until the budget and resources are available to modernize these complex applications and incorporate the updated business processes, workflow, security, and reporting needs, they should remain on Domino.

Conclusion: Get the FACTS First, then Scope Your Migration Effort

Can you handle the truth? To make accurate, wellinformed migration decisions for your Domino application environment it is important to get the facts, all the facts. They will provide you with a complete understanding of what is involved. These migration projects are highly visible from upper management (more details here: <u>Ways to provide</u> <u>accurate progress reports</u>⁽¹³⁾), especially for critical business process applications that drive revenue for the company. Make sure you analyze these from all angles, so you understand the truth.

If you are interested in finding out more about our iDNA Applications solution and how it can help you get the facts about your Domino application environment, please visit our overview page online, or sign-up for a trial at www.panagenda.com/idna.

Bonus

Understanding Domino Application Data – Using Content Analysis for Migration Planning

There are two parts to every migration planning estimate for a production Domino application. Number one is to know the business processes, workflow, and functional design of your application so you can redevelop it on the best target platform. That part was covered in detail by the seven-part blog series earlier this year. You can read all about it here, Master Your Domino Application Migration Challenges. Now we are going to discuss the second part of the story in this blog, which should be titled, "Don't forget about the data!"

Successfully migrating your business applications off Domino will require transitioning the data to the new platform. Maybe not all of the data, but most likely a big chunk of it. Understanding what needs to be moved to the new repository, and how to migrate it with complete data fidelity is a key component for every migration planning project. The same is true even if the platform does not change. New UI technologies or 3rd party solutions will have their own restrictions that need to be validated.

Application migration and modernization projects are hugely expensive and can quickly run off the rails by making assumptions and underestimating the complexity of moving the historic data. So how can you pull back the curtains and inspect the data stored in your Domino apps so you can complete an accurate migration project estimate? Let's see how our customers address the challenge with our iDNA Applications solution.

Topic 1: How many attachments? And how big are they?

One of the biggest mistakes that migration rookies make is to forget about all the attachments stored within Rich Text fields. They just count the number of records in the Domino application and don't bother about the thousands of PDF, JPEG's, or Excel files that require a separate migration effort to move them into the appropriate target storage format or document management system.

Deciding where to transition these files is an important consideration during the migration planning effort. And that decision might rest on different factors, such as the type of files, or the size of the files. Domino doesn't have a file attachment size limit, so video files or large CAD drawings could be attached to a Rich Text field inside a record.



Example Report: File Attachments by Size

Understand your attachment types

iDNA Applications maps out the file attachments by document type and size, as you can see in the associated example reports above and below. By analyzing their extension name you will quickly be able to determine the use cases for these files and where they might best be stored for simple integration with the newly designed application.



Example Report: File Attachments by Extension Name

Topic 2: How many records are we migrating? And how old are they?

The next important item for data migration planning is the number of records you need to transition. Included in this topic is a common decision point for a date cutoff to migrate records. Most organizations would like to migrate only a subset of historic records based on the date they were created. The remaining records can be kept in archive.

It's important to be able to judge content age from multiple viewpoints (e.g. creation or modification history). This empowers your migration planning team with exact details on the amount of information they will need to migrate, along with their size references for the total data volume.



Example Report: File Attachments by Size

Topic 3: Identifying the Correct Forms In-Scope for Redevelopment Effort

Another benefit of analyzing the data stored within a Domino application is that it identifies exactly what Forms are being used in production to store and reference the records. Each record in a Domino application dictates what Form is used for that record. This provides the planning team with specific insights on what Forms and Views need to be redesigned to display, create and edit the data records.

This applies to both modernization and migration projects. It doesn't matter if you are preparing your move to a new platform or if you are putting a web or mobile interface on your applications. When a Domino application could have 100 Forms included in the design, but only 10 are being used to interact with the data, then you can reduce your redevelopment scope considerably.

Topic 4: Are there Special Rich Text Objects that Complicate Redevelopment?

Rich text fields in Domino are very flexible data types. Anything from simple text to attachments to embedded objects can be found in these fields. That same flexibility also makes it hard to display their data in an interface other than native Notes/ Domino, let alone migrate the content intact to a new repository.

Seemingly trivial things like nested table structures can add a lot of complexity to redevelopment. Other things like Doc-, View- or Database-Links may require 3rd party software to make a transition seamless to the new target platform. Embedded content like OLE objects might even be a complete no-go when it comes to migration or modernization, since very few target platforms or modernization solutions can work with them. Identifying how widely these different elements are used throughout the content of a Domino application can be pivotal when choosing the right target platform and estimating the redevelopment effort.



Example Report: Special Object in Rich Text Fields

Topic 5: Are there any special security considerations or access restrictions?

Finally, there are some additional considerations related to data security that can be answered through content analysis. Domino allows for several extra levels of data security beyond the simple access rights defined in the Access Control List (ACL). The two most important ones which cause issues for migration teams are Encryption and "Reader Fields".

Encryption completely scrambles specific fields in documents, making their content inaccessible unless the key for decryption is present in the Notes ID accessing the document. "Reader Fields" work differently, but with a similar result. If you are not included within the "Reader Names" list for a specific document, then it will be hidden from view. These special security challenges should be identified before migration planning and will need to be evaluated during data export testing to assess the migration tools and ID files used for processing.



Example Report: Security Requirements for Reader Names Fields

CONCLUSION: Understand Your Domino Data, then Scope Your Migration Effort

Can you handle the truth? To make accurate, wellinformed migration estimates for your Domino application environment it is important to get all the details. Don't forget about the data. Combining the knowledge about the application design with the details about the content stored within the application will provide you with a complete understanding of what is involved for the successful transition to a new platform. These migration projects are highly visible to upper management, especially for critical business process applications that drive revenue for the company. Make sure you analyze these from all angles, so you understand the truth.

If you are interested in finding outmore about our iDNA Applications solution, now with content analysis, please visit our overview page online, or sign-up for a trial at <u>www.panagenda.com/products/idna/</u>

Disclaimer

panagenda, panagenda product names and all related logos are trademarks owned by panagenda. All other names of products and enterprises in this knowledge base are the property of their respective owners.

panagenda reserves the right to update this document without being obliged to announce the changes or revisions.

Although all due care has been taken in the preparation and presentation of this ebook, the corresponding software may have changed in the meantime. panagenda therefore disclaims all warranties and liability for the accurateness, completeness, and currentness of the information published, except in the case of intention or gross negligence on the part of panagenda or where liability arises due to binding legal provisions.

Sources:

¹https://www.panagenda.com/kbase/display/IA

²https://www.panagenda.com/2019/10/safely-navigating-your-domino-projects-with-confidence/

- ³https://help.hcltechsw.com/dom_designer/9.0.1/appdev/H_NOTESDXLEXPORTER_CLASS.html
- ⁴https://powerbi.microsoft.com/en-us/blog/microsoft-named-a-leader-in-gartners-2020-magic-quadrant-for-analytics-and-bi-platforms/
- ⁵https://www.panagenda.com/2020/04/3-new-ways-to-automate-your-progress-reporting/#video
- ⁶https://www.panagenda.com/webinars/
- ⁷https://www.panagenda.com/news-events/blog/
- ⁸https://www.panagenda.com/2019/10/safely-navigating-your-domino-projects-with-confidence/
- ⁹https://www.panagenda.com/2020/02/most-successful-projects-with-consolidation/
- ¹⁰https://www.panagenda.com/2020/02/most-successful-projects-with-consolidation/
- ¹¹https://www.panagenda.com/2020/03/about-the-many-roadblocks-in-your-source-code/
- ¹²https://www.panagenda.com/2020/03/seeking-application-owners-find-application-stakeholders-instead/
- ¹³https://www.panagenda.com/2020/04/3-new-ways-to-automate-your-progress-reporting/

Copyright © 2020, panagenda All Rights Reserved.