PRIASOFT™

KB Article 011515-01
Categorization and Batching

Categorization? Batching?

What is this topic of “categorization” and “batching”?

A key component of migration success is understanding what must be migrated and the demographics and distribution of the same across the organization(s) and the categorization and classification of those items. There are many ways to categorize and a balance must be met in order to not over-complicate things. However, it is equally important to sort and classify the related objects since such may create a need to change “how” those objects are migrated.

Batching is the process where, based on categorization, “batches” of objects are identified for migration and are then assigned a specific set of options that control how they are migrated. Each environment is different and the type and depth of the categories can vary, but the approach and effort is nearly the same each time. Batching without categorization becomes easy since all objects can be in a single batch, but this obfuscates potential issues that can affect end users, project duration, or even success. It is likely and should be expected that there will be at least a few different categories that cause more than a single batch for migration.

Categorization Mindset

Given that there are likely an infinite number of ways to classify objects for migration, the mental approach to this task becomes important. It is often a good idea to first analyze the different contexts of use [of Exchange and email], followed by rules or requests from the business like VIPs or other special cases. The idiom of “one cannot see the forest because of all the trees” applies here. Focusing too early on object details or attributes can make it difficult to identify the best higher-level categories.

Contexts of Use

When approaching the task of categorization, the context of use refers to looking at how groups of users access the information. The following is a list of common contexts that are typically important to identify.

- Traveling users – users that typically perform work from outside of the enterprise envelope. The way these users access information may require a different set of values than a typical “desk worker”. Consider that such users may need VPN access or may need to use Outlook Anywhere [for email]. They will typically be on a less reliable network and it may be a very highly latent connection, possibly over the Internet.

- High security users – depending upon the type of business, this category may or may not exist. However, it is wise to investigate. HiSec users often have special rules and tools that allow them to do their work. Special consideration, settings, and possible extra work may be needed to support a seamless transition for such users. In addition, HiSec users may have restrictions on time-of-day for which those users, or even admins and migration efforts, may access data. Very often, government contracts and financial trading are in this category in some fashion.
- Mail Consumers – this category would be users that generally only read email and either do not produce email, or produce very little. This category, if identified, can affect batching in a positive way since this group likely has the least concern on post-migration experience and expectation
- Web-only users – there may be users in an organization that only work with email through OWA (Outlook Web Access). The lack of a thick client app may simplify batching, but may also mean that these users are batched separately. Communications with these users is typically more difficult since they check email less frequently than an Outlook user might would do. This category might not cause a separate batch for migration, but may create a separate communication pattern of which is important for migration – there is often a need to inform users of system changes prior to an event. Also, it is likely that the URLs used by these users will change, and as such communication of this change should be distributed to them.
- Mobile clients – the business world is much more connected today than before. The saturation of mobile devices, phones and tablets, is very high and these devices typically have a different connection path than Outlook. There is, very often, a need to batch mobile users differently so that their mobile devices can be treated properly.
- Shared Mailboxes – (an Exchange 2010 and later sub-category) as the title implies, these are mailboxes that are "shared" between multiple users. There are typically 2 definitions for this term: the Microsoft technical definition, and the business usage definition. The Microsoft definition is one that requires the mailbox to be a disabled user account that is also not linked to a remote user. This definition is fine for cases where no user logs on to the mailbox directly. However, if the business definition is one where a user logs on to the shared mailbox directly, such would be treated differently during a migration.
- Room and Equipment Mailboxes – Exchange 2007 and later also support these additional sub-categories of a mailbox. They are very similar to that of a shared mailbox, but are more usage focused. These mailboxes have the ability of enabling an "auto accept agent" for booking of the resource. These mailbox types often need to be in a separate batch so that the tools can create those mailboxes at that specific type.
- Existing "Linked Mailboxes" – there can be cases in larger enterprises where some, most, or all of the mailboxes to be migrated are linked with a remote account. This is quite common in an organization that has grown by acquisition: they quickly ingest the mailboxes so that there is a common mail platform for the new organization, but the users continue to logon to their original domain. Priasoft has an ability to understand this case, but such mailboxes must be identified and migrated in a separate batch so that the setting can apply to that group. Mixing normal and linked mailboxes in a migration batch is not supported.
- Multiple-Mailbox users – there can be cases, especially in larger organizations, where a single user has more than one mailbox. The reason is not so relevant, but for the purpose of understanding can happen when a user needs to present 2 or more personas: private and internal, and public. Politicians and celebrities often have such. The case can also surface for enterprises that consist of multiple distinct email environments. A user may have multiple mailboxes across the entities for some reason and where all except one have an additional forward attached, so the user only needs to manage a single mailbox. The multi-mailbox scenario is a fringe case, but often is indicative of a very special user and as such should not be ignored.
- Journal Mailboxes – these mailboxes typically exist as one that receives a copy of all email based on some criteria, or potentially for an entire organization. It is very common for an application to also exist that processes the email delivered

to this mailbox in order to control its size. These mailboxes may or may not need to be migrated and could contain an inordinate amount of items.

- Application Mailboxes – this is another non-user mailbox, but is typically very special. There are applications that require a mailbox in the organization. A common system that will have a mailbox is a help-desk platform. Special consideration is required for these mailboxes since they are directly related to the use of a service or application. Furthermore, there may be a group of users that work with that application and for which cannot, or should not be migrated unless the application mailbox and the application have also been migrated (or will be migrated at the same time).
- Third party contractors – these users are unique in that they can have more flavor than a typical employee. Contractors might be categorized as one or more of the following:
    - No user nor mailbox nor any other representation. This is easy in that there is nothing to consider.
    - User account exists, but has no mail attributes.
    - User account exists and has a forwarding address pointing externally. This is most common and are usually seen as mail-enabled user accounts. The contractor needs the user principal in order to access resources in the environment, but does not need a mailbox (he/she already has one at the parent company, like @Microsoft.com). When migrating to a new platform, these objects usually are migrated as a Contact in the new environment since the user likely does not need access to the target environment (yet). However, further consideration should be made for how the future may be. If there is a potential for the resources, for which the contractor accesses today, to be migrated to the target environment, then it may be wise to create a mail-enabled user in the target as well, but initially disabled.
    - User account exists and has a mailbox. Likely such a user should be migrated, but such may be wasted work if, when analyzing timelines, the user's contract expires 2 days after migration. It would likely be better to not migrate such a user and let normal attrition of the mailbox occur, unless, as mentioned before, data retention is required.
    - User account exists, has a mailbox, AND has a forwarder. This is the least common case, but can exist. This would be one where the contractor receives mail in the mailbox of the organization (for audit purposes perhaps), but has a forwarder in place to his/her parent company (like @Microsoft.com) so that he/she only manages a single mailbox.
- VIPs, CxO, Senior Management – without a doubt, the very senior executives of an organization receive special priority and treatment. These users often have attributes that a typical employee would not have. They may have a computer at home, possibly non-domain joined. They may have some special devices (tablets, terminals, etc). They may connect from atypical locations (private plane, boat, etc.). They may rarely be "at the office". They may have multiple mobile devices, including multiple phones of different brands. They may cross some or all of the previous categories. They may have enormous mailboxes, either by item count or size or both.
- High Influencers – similar to VIPs above, this category can include the CEO's wife or children (if they work at the same company), "big money" producers and the top 1% sales people, or wives, husbands, sweethearts, etc. of a VIP or other High Influencer. This group can also include those with very special access like certain security folks.
- Help-Desk users – this special group of users becomes important to categorize and batch in a very large organization as there may be many dozens or hundreds of such people. They are very often dependent upon a software platform that is likely tied to email. There can also be cases where some of these users are company employees and some are contractors.

Concluding thoughts about Contexts

Given the many example contexts above, it should be understood that this list is by no means definitive or complete for any organization. The above is a common list and can be used as a starting point. The goal in listing these contexts is to provide some mental insight as to "how" to think about contexts. The key point across all the above is that they deal with use and interaction with the system and such should be the starting point for "context". It can also be true that some objects cross more than one of the

above contexts. Such may create a separate batch in and of itself. It is very possible to have a batch of only a few users due to the fact that it crosses several contexts. For instance, consider a group of users that are High Security, work remotely, use an application that creates and/or receives email, and includes one or more VIPs or High Influencers. The point of this exercise is to reduce, as much as possible, changes in the end user experience after migration. The best chance of meeting this expectation is through categorization, even if it means that a batch contains a single user, since such a case would likely be VERY unique and likely equate to a very important use case.

As the rest of the topics following this one are analyzed, one should take a layered approach across all of them. The prioritization of categories is an important step in and of itself and is usually unique to each entity. It is for this reason that the caution was given in the opening section about balancing the number of categories with what is needed.

Relationships

The next approach to categorization involves looking at various relationships between users and other objects that a user might use directly or indirectly. One of the goals of this task is to help prevent cases where batching breaks relationships which could cause a negative perception for the users affected. The performance, options (like backfill/2-pass migration), and scalability of the Priasoft tools should allow for large, and or very large batches to be created. Doing so helps mitigate the chance of breaking such relationships. In addition, there are sometimes outside or even direct influencers that, on the surface, work against the creation of large batches. If possible, look for creative ways to control those influencers so that large batches can be used. For instance, in the past we have seen some customers that, initially, could not migrate in a single event due to a subset of users still using Outlook 2003. This version of Outlook could not be used to connect to Exchange 2013 and so the initial thought was to split the migration over several weeks until those users could be updated. However, once the understanding of the pains of coexistence and the likely negative perception for end users was received, it was chosen to accelerate the update of Outlook to a supported version so that coexistence could be avoided. It is often better, both from a monetary and time position, to reduce and/or avoid coexistence as much as possible, and when it cannot be avoided, to keep that period as short as possible.

The following list highlights most of the common relationships to track and for which may create a category that influences the batching process:

- Delegate relationships:
    - This refers to the native ability in Exchange to for a user to allow another user to have access in some way with their mailbox. Calendars are something that are very common to share between users.
    - This is one of the first and most visible relationships to track and analyze. By default, there is no ability for one user to open the folder of another user that has a mailbox in a remote forest. There is an ability in Ex2010 and Ex2013 to support cross-forest delegation, but it requires additional infrastructure to support it and then also means additional monitoring and maintenance of those systems. The additional complexity and maintenance shows that the case is best avoided, if possible.
    - There are multiple specific meanings of a delegate, as follows:
        - Folder delegate – permission granted to a user to access a folder in a mailbox or in Public Folders. The detail of this permission is stored directly on the folder in Exchange and as such is difficult to get since such detail is not indexed and is not in AD. Analysis of this information is valuable, but requires opening each user's mailbox and enumerating each folder. Such an operation, even if scripted, can take many hours or even days to complete.
        - Send-on-Behalf-of – this permission is normally granted automatically when using Outlook's Tools->Options->Delegates dialog. The SoBo permission is stored directly on the AD user account and should be

considered a “user permission”. The attribute in AD is “publicDelegates” and is a multi-valued attribute holding one or more AD DistinguishedName values.

- Full-Access – this permission is a “mailbox permission” and when set allows the named user or group to have access to the entire mailbox. This permission is commonly used to allow a user to add a shared mailbox to his/her Outlook profile as a secondary mailbox. Although this detail can be inspected from the AD attribute “msExchMailboxSecurityDescriptor”, it should be noted that this attribute is not synchronized with the mailbox in the Exchange database. Changes to mailbox permisssions require use of a supported API (powershell or cdoexm) and such causes the change to happen on both the mailbox entry in the database and the attribute on the AD user. However, this attribute can be used for analysis to determine relationships.
- Send-As – this permission is a “user permission” and is separate and distinct from Full-Access. It used to be true in early versions of Exchange 2003 that Full-Access implied Send-As, but such was changed in a later Ex2003 service pack to separate the permissions. Send-As, being a “user right”, can be inspected and managed from the AD user account’s ‘NTSecurityDescriptor’.

  - The most valuable “delegate” information, in the context of relationship, is typically the Folder delegate permission. However, as mentioned above, this information is the slowest information to retrieve. Send-on-Behalf-of is likely the easiest to retrieve, but might not show the full scope of relationships.

- Group relationships
  - Group and Distribution List membership can be valuable to establish how objects are related.
  - Note that a group can be a member of one or more other groups and can be nested very deep. Expansion of groups would be needed to identify all members.
  - A misleading idea with groups is to assume that owners or members completely define the relationship. This error comes from the realization that mail-enabled groups can be used as a recipient of email. There is no way, by analyzing a group, to determine all users that send mail to the group. There is no requirement that one be a member to be able to send to the group. There are restrictions that can enforce such, but those restrictions are not enabled by default.
- Department, Office, Business Unit, etc.
  - There are many attributes on objects that can be used to identify relationship.
  - However, few of these attributes are required and as such may not show the entire relationship.
  - Human Resource data combined with Active Directory data may help provide a better mapping of relationships with such attributes.
- Mailbox databases
  - In some organizations, strict guidelines exist regarding how mailboxes are distributes across mailbox databases. Such information may provide some level of relationship suggesting that all mailboxes on a specific database have some relationship.
- Physical Location
- Marital, Sibling, Kinship
  - It is not uncommon to find entire families working at the same company.
  - Often these relationships can be assumed by looking at surnames and physical locations, unless the surname is very common.
  - Such relationships should not be casually ignored especially in cases of VIPs or High Influencers.

Active Directory Attributes

Active Directory has an extensive set of attributes for users, contacts, and distribution lists. There are opportunities to analyze the attributes to infer possible relationships. Since this is merely data attached to objects, the opportunity to use scripting and technology to extrapolate such does exist.

The following list will show some examples of how AD attributes could be used to identify relationships. However, this list is only an example and is meant to provide a mindset to show how relationships might be inferred.

- Surname
  - This attribute, when sorted alphabetically, will align objects with the same value next to each other. Such can provide an opportunity to identify kinship and genealogical relationships.
  - This attribute can also show possible cases of ambiguity, perhaps by misspellings. Example "Smith, Jon" and "Smithe, Jon". Such a case could be 2 distinct users, the same user, or one could be a test account or some other issue.
- Department, Office, Address, etc.
  - When populated, such attributes can quickly provide grouping of likely related objects.
- Custom and Extension Attributes
  - There may be values in these attributes that come from automated systems or applications that can show a relationship between objects.
- EmployeeID
  - There is often some format to Employee IDs similar to a postal code. If such a pattern exists, this value can show relationship when only the duplicated portion of the attribute is analyzed. For example, if an EID is something like AMS1130034 and "AMS" is a 3 letter code for the geographic business unit (Amsterdam perhaps) and the next 3 digits "113" is a cost center or some department code, there may be an opportunity to use such components to form groups and relationships.

Caution should be exercised when attempt to analyze AD attributes. The large quantity and near infinite ways of comparing details can lead to "analysis paralysis" where too much time is spent on this task. However one VERY valuable exercise involving analysis of AD attributes is described in the next section, "Exceptions and Oddities".

Exceptions and Oddities

This topic is a different approach to categorization than the previous sections, but may have the most important role. Exceptions and Oddities, when identified, often show cases that would have otherwise created and issue and typically require input from business leaders and managers to classify and define. Since such tasks involve ***waiting on non-technical persons***, identifying these cases is important to find as early as possible to give time for those leaders and managers to respond.

Active Directory attributes are a very good way to identify oddities and exceptions. The following will show examples of how one can identify these cases.

- Changes in pattern of how data is viewed.
  - For instance, if 90% of objects that have a particular attribute are in UPPERCASE, the remaining 10% should be scrutinized and determined why the rest do not conform to the majority. Often such a distinction means that the exceptions were created outside of automation and may be ok, but it can also mean that those objects are special in some way.
  - If an attribute has a pattern of length, words, or other pattern for which there is a large majority that conform, any differences should be looked at.
  - Missing data can be just as much as an indicator of an exception as a change in pattern. If 90% of objects have a value on an attribute, those objects that do not may be special in some way.
- Identification of non-User mailboxes
  - Most user mailboxes will follow a pattern in the display name of some sort. "First M Lastname", "Lastname, First", etc. A non-user mailbox might only show as "Projector#6".
  - Application mailboxes will typically have a display name related to the application name: "Techsoft Service Desk"
- Duplicates
  - DisplayName and most other attributes in AD can be duplicated.
  - Multiple objects sharing the same name should be analyzed and notes taken as to why the case exists.
- Special characters
  - When migrating from one version of Exchange to another, there can be cases where characters that were accepted in one version (perhaps only because of no enforcement) may not be accepted in the other.
  - Identifying objects that contain non-alphanumeric values can potentially show an issue early and allow proper business reaction and rectification.

The examples above hopefully show ways to look at data to quickly sort in to 2 groups: normal and "needs investigation". In many cases, the "normal" objects equate to 80% or more of the total objects. Quickly getting to the "needs investigation" list allows more time for the analysis of such. The more the data is reviewed and looked at, the more likely that other oddities and exceptions can be found. These cases WILL be found, one way or another, either early or late when these oddities impact users or admins.

Batching

Batching is the process of taking the results of the categorization to build "batches" of objects that should be migrated together or at least with the same settings. Batching has additional value of providing justification for how to scale the migration effort. It is often difficult in the early stage of a migration plan to determine exactly how many migration servers are necessary to meet a goal. Batching provides the answers.

Successful batching comes from a combination of good categorization and good understanding of options available with the migration solution and options available (or not available) with additional tools, services, or infrastructure. For instance, the decision on whether co-existence components are necessary (cross-forest free/busy, delegation, etc.) can be driven and determined based on how batching results. If batching efforts shows that there are few batches needed and batches are able to be large enough to migrate the organization in a single event, the additional complexity of co-existence can be removed. Or, conversely, if the batching shows that many batches are necessary and combined with business processes, policies, or directives, co-existence is required, one can plan for duration of those additional systems and to what scale they must be employed.

Batching can take considerable time, even after categorization completes. This task involves comparing list sizes with performance numbers, available time windows, overlapping events from other work streams, tool options, and goals. Due to the amount of data needed to perform this task, both in categories and quantities of objects, a tracking system is best used to manage such.

The following specifics about Priasoft's tools can help with the mental preparation for batching. Priasoft's team can be available to help with batching efforts and when needed must be scheduled with our team in advance.

Mailbox Migrations

Batches in the mailbox migrator consist of tool settings and a list of accounts to be migrated. User lists can be created and saved in the tools so that preparation can be done days before the start of a migration. However, a saved user list is static by nature and when loaded at a later time is not re-evaluated for correctness. This is so that user can quickly move thru the wizard and start a batch. The side effect is that if changes have been made in AD or Exchange for any of the saved users, those will likely fail to migrate. It is best to generate such user lists only a few days before use, unless some other control over the users is made to ensure no changes occur.

The migrator also allows for the settings to be saved to a file that can be loaded at a later time. Unlike the user list, the detail saved is specific to options in the application and can be created days or weeks in advance. In very large migration efforts, it is not uncommon to have an exercise to establish several settings files for the different contexts necessary for migration. Such effort provides an opportunity to provide auditing and detailed control over how a migration actually occurs. There could be several separate settings files, each named descriptively as to their purpose. Then, combining those settings with several user lists, is what can create a batch. It is a good methodology to provide a "check in/check out" pattern for such files for later auditing.

Directory Sync

Batches in the Priasoft Collaboration Suite consist of "agreements" that define the scope of the objects to sync and the options to apply to the same. In contrast to the mailbox migrator, these agreements use filter patterns to define the list of objects. This means that as the business operates over time, the list can react properly to business changes.

It is likely that multiple agreements are used in a migration, at a minimum one for each source object classification: contact, mail-user, mailbox user, and distribution list. However, the categorization work presented earlier in this document may require subtle distinctions between objects in a single class of objects. The following list shows some likely cases:

- Non-migrating mailboxes in source
  - If categorization shows that there will be some mailboxes that will NOT migrated (never and no chance at all), such should likely be created as Contacts in the target.
- Contacts
  - There are likely 5 sub-categories just for contacts:
    - Normal External Contacts – these are the common object that have an email address outside the organization like Johan@microsoft.com
    - Legacy Co-existence Contacts – these are contacts that exist as a representation of a mailbox in another organization that is considered part of the enterprise.
    - Non-mail contacts – the object class of "Contact" is required by LDAP and is a default class from Active Directory. It is possible to create contact objects that do not participate with Exchange and are invisible to Exchange processing.
    - Application Contacts – these contacts may exist to support an application in some way.
    - Migration Contacts – these are contacts created by a migration effort (Priasoft has an option for this) to provide a forwarder pointing to the target mailbox.
  - Normal contacts can simply be synchronized as-is and would be contacts in the target.
  - Legacy Co-existence contacts present a unique issue. If the mailbox related to the contact (in the other mail domain) will be migrated to the same target as others, these contacts should be created as mail-enabled users and not contacts since at some point a mailbox will be needed in the target for it.
  - Non-mail contacts will not be seen by the dir-sync. If there is some need for these in the target, such will require other tooling or scripting.
  - Application contacts can be sync'd, but special consideration should be made and its relationship and dependency on the application analyzed.
  - Migration contacts will be explicitly ignored by dir-sync.
- Mail-enabled users
  - These are user accounts with a forwarding address like a contact.
  - These should either be created as mail-users or contacts in the target. If the target environment is destined to be a mail-only platform (no user logons now or ever), then the source mail-users should be created as contacts since those users should have no reason to access the target environment.
- Mailbox-users
  - These are user accounts with mailboxes
  - The dir-sync can create these as contacts, mail-users, or mailbox-users.
  - Most implementations create them in the target as mail-users and the mailbox migrator will convert them to mailbox-users during migration. However, there may be specific reasons to create them as contacts or mailbox-users.
- Distribution lists
  - The dir-sync process uses the linking established by other agreements to understand how to manage group members.
  - In most cases, this type of agreement should run after all other agreements.

Given the various object classes combined with categorization, the number of agreements could be less than 10 or many dozen. It is important to understand how dir-sync works and what options are available in order to be effective in creating appropriate agreements.

Conclusion

Categorization and Batching have as much to do with success as the many technical options and features provided by Priasoft's tools. Time spent on categorization helps identify potential conflicts, issues, and oddities and influences scaling and capacity planning efforts.

Time is an important component in that the size of the organization(s), number of categories, size of oddities and exceptions all affect how long these tasks can take to complete. When consideration is made that some of this work can prompt interaction with non-technical business leaders and managers, time becomes more important. It can be frustrating to discover an exception late, and then to have to wait for a response before proceeding. Furthermore, such frustration can lead to assumptions about data and treatment of data to the point of creating a cascading issue that requires much rework after the fact.