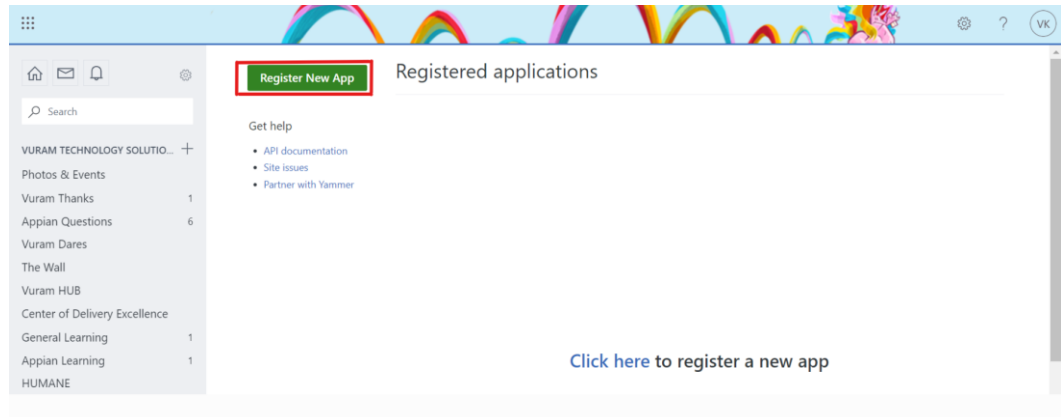# Yammer Migration Through API

Yammer migration is the process of transferring the Yammer data from one domain (ex. xyz@abc.com) to another domain (ex. xyz@def.com). The domain of the source and the target network varies and all the posts, groups, messages that are in the source domain must be present in the target domain. This process of Yammer migration can be done on a small scale (100 users). The bot connects to the Yammer via API, and performs the actions of creating groups and posting messages.

Prerequisites:
- The user has to export the network data and user data from the source network
- Create users using bulk CSV upload in the target network
- Export the user and network data from the target network after creating the groups in it
- Extract the access token of the users in the network domain and store it in a separate CSV file

How to Generate Access Token:
- After creating the user accounts in the target Yammer domain, we have to generate the access token for each of the users. It can be done by the following steps.
  - Login to the yammer account
  - Navigate to the link: https://www.yammer.com/client_applications
  - Click on "Register New App"

- ○ Fill in the register form as follows:
  - ■ Application Name - Preferred by user (ex. Access Token)
  - ■ Organization - Vuram
  - ■ Support email - User's email (abc@vuram.com / abc@wns.com)
  - ■ Website - https://www.vuram.com/ (Provide organization's website)
  - ■ Redirect URL: https://www.yammer.com/vuram.com/ (Yammer URL)
  - ■ Click on the Checkbox and click continue

- ● You'll be redirected to the page where the access token is generated
- ● Click on "**Generate a developer token for this application**"
- ● The Access Token will be generated

Workflow:

- The workflow of the bot consists of 2 modules
  - Group creation
  - Adding posts from the respective groups

Group Creation:

- Before posting the contents into the Yammer, Groups (or) Communities have to be created. There will be a separate CSV file named "**groups.csv**" from the export
- The columns of the CSV file will be in the following format

id	name	description	private	moderated	api_url	created_by	created_by	created_at	updated_at	deleted	external	office_grou	group_tags
111111	All compan	Post for all	TRUE			10				FALSE			
222222	Appian Lea	Posts relate	TRUE			20				FALSE			

- After creating the groups in the target domain, we need to export data from the target network. This process is done because the group ID differs in both the networks. Hence group ID mapping must be done
- The bot compares the "groups.csv" files from both the source and target network and creates a new column "**Updated Group id**" which contains the group id of the group from the target network. Also include the access token of the individual users
- After the updation of group ID, the resulting CSV column looks as follow:

id	name	description	private	moderated	api_url	created_by	created	create	upda	deleted	extern	office_grou	group_tags	Access Token	updatedGroupId
111111	All compan	Post for all	TRUE			10				FALSE				79580348416-4dPukhbHZc	10000
222222	Appian Lea	Posts relate	TRUE			20				FALSE				79580348416-4dPukhbHZc	20000

Add Posts:

- After creating the groups, the next action is the adding posts
- The bot reads the following parameters from the **messages.csv** file which contains the information such as the posted message, posted user, posted under the group id, posted at (time and date) etc.
- Inorder to post message in the yammer, we need only the following parameters:
  - senderName
  - accessToken
  - Body
  - groupId
  - messageType
  - uniqueId

- The bot get these values from the messages.csv file and connects to the yammer via API using the access token of the following users and post the messages respectively

```
[STAT] 2023-10-30 06:10:03 - - <ACTION_START> - Start action 'YammerMigration:v0.0.1:addPost' - 'Add Post' // dur: - - avg: - etc:
[INFO] 2023-10-30 06:10:04 01s (0h) - Map Size: 1
[INFO] 2023-10-30 06:10:04 01s (0h) - Map Size: 2
[INFO] 2023-10-30 06:10:04 01s (0h) - Map Size: 3
[INFO] 2023-10-30 06:10:04 01s (0h) - Map Size: 4
[INFO] 2023-10-30 06:10:04 01s (0h) - Map Size: 5
[INFO] 2023-10-30 06:10:05 02s (0h) - Response Code: 201
[INFO] 2023-10-30 06:10:05 02s (0h) - Message posted successfully! for 1
[INFO] 2023-10-30 06:10:06 04s (0h) - Response Code: 201
[INFO] 2023-10-30 06:10:07 04s (0h) - Message posted successfully! for 2
[INFO] 2023-10-30 06:10:08 05s (0h) - Response Code: 201
[INFO] 2023-10-30 06:10:08 05s (0h) - Message posted successfully! for 3
[INFO] 2023-10-30 06:10:09 06s (0h) - Response Code: 201
[INFO] 2023-10-30 06:10:09 06s (0h) - Message posted successfully! for 4
[INFO] 2023-10-30 06:10:09 06s (0h) - Response Code: 201
[INFO] 2023-10-30 06:10:09 06s (0h) - Message posted successfully! for 5
[STAT] 2023-10-30 06:10:10 07s (0h) - <ACTION_END> - End action 'YammerMigration:v0.0.1:addPost' - 'Add Post' (transition through
```

Time Constraints:

- Upon testing, it has been noted that the time taken to post 100 messages from 100 different users was approximately **110 seconds / 1 minute & 50 seconds**, resulting in the inference that it takes approximately 1 second to post a content from a single user
- These automations are suitable to carry out only in small scale migration (Less than 200 Users)

Variable Used:

- GroupCSVFilePath: Filepath of the groups.csv file, which is used for creating groups in the target network
- MessagesCSVFilePath: Filepath of the messages.csv file that is used to posts the contents in the yammer

▼ Variables		➕
Name	Parameter	Multiple
GroupCSVFilePath	✓	
MessagesCSVFilePath	✓	