**Table of Contents**

[**Introduction** 2](#_Toc528938257)

[**About Yammer Extraction Tool (YETI)** 2](#_Toc528938258)

[**Pre-requisites** 2](#_Toc528938259)

[**Download & Configure Yammer Extraction Tool** 2](#_Toc528938260)

[**Generate Yammer API Token** 3](#_Toc528938261)

[**Yammer Extraction Tool Database Creation** 4](#_Toc528938262)

[**Datasets & Usage** 7](#_Toc528938263)

## **Introduction**

This document describes the environment details (Servers etc..) related to Yammer eDiscovery (YETI) application.

## **About Yammer Extraction Tool (YET)**

**Introduction:**

Yammer Extraction Tool (YETI) is used to download, process and upload yammer data related to all groups (public and private) to SharePoint for future purposes. Yammer Export API is used to download data​. Data uploaded to SP is crawled and indexed and will be ready for eDiscovery searches​.

**Data Processing:**

Export API gives output Zip file for each day​. Data is loaded into SQL tables​ & extract attachments to File share​. User information is fetched using Users API​. We download missing attachments and notes (Pages). HTML file for each thread with related conversations and attachments is prepared​. Then folder is created for each Thread and Html file and attachments​ are placed.

## **Pre-requisites**

You must have below software’s installed in your machine to run Yammer Extraction Tool in your local machine.

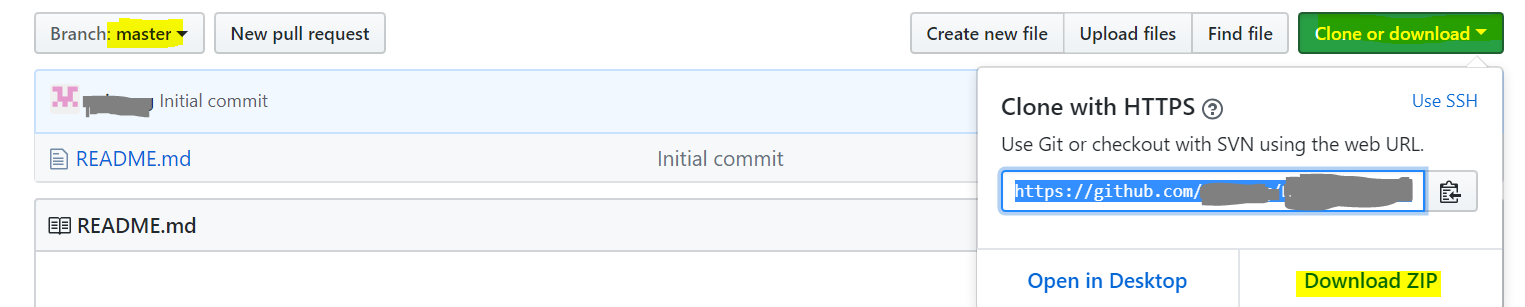
* Visual Studio any latest version
* SQL Server 2014 or above

## **Download & Configure Yammer Extraction Tool**

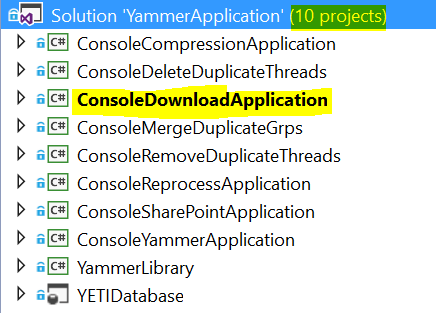
Follow below steps to download & configure YETI in your location machine.

1. Clone/Download the codebase from the below GitHub link.

<<GitHub Repository Link >>



1. Make sure that all the 10 Projects in the solution including database are loaded and try to build solution with no errors.
2. The project “**ConsoleDownloadApplication**” should be marked as startup project.



1. In every Console Application project there will be an App.Config file which is having key and value pair in the App settings section. The **need of each key is mentioned in the placeholder of value part.**

**Example**: <add key="DBNameURL" value="{Database connection string}" />

In real-time, those keys should be fetching their values from Azure Keyvaults.

1. There is a token to be generated for API calls in-order to download yammer data (ExportAPI). We need to make sure that the token will be active.

**Export API URL:** https://export.yammer.com/api/v1/export?since="+ startDate+"&until="+endDate+"&access\_token="+ **token**;

1. Once everything is ready, you can create/Schedule task schedulers to run all above projects.

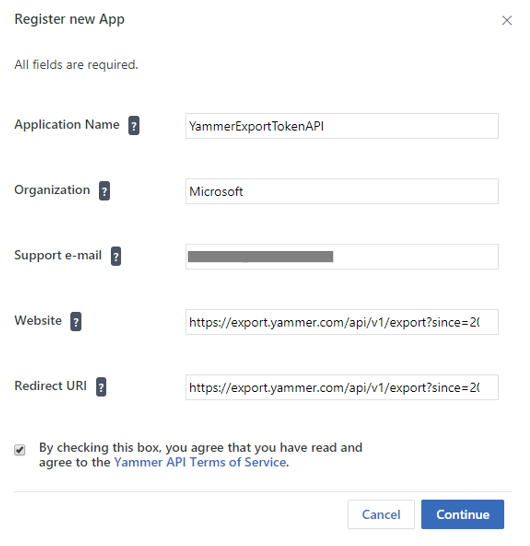
## **Generate Yammer API Token**

To generate a token for ExportAPI call, one should have a verified admin account that will be used while generation of the token. First, we need to Register our application in the Yammer Portal and continue the steps mentioned in below.

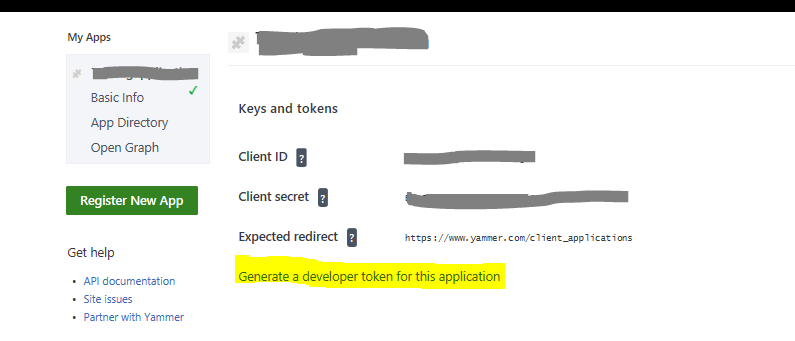
**URL :** <https://www.yammer.com/client_applications>

**New Registration Process**

* Go to <https://www.yammer.com/client_applications>
* Click on “**Register New App**” and fill the below form



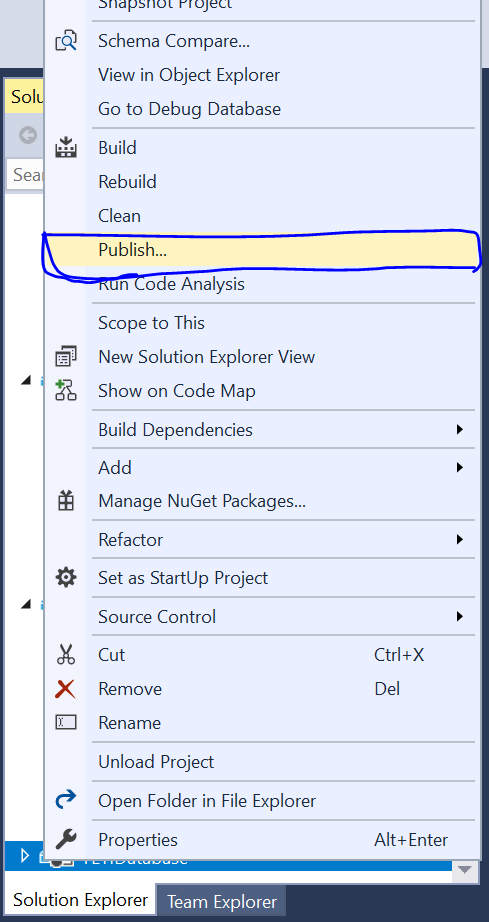
* Click on “**Continue**” button to create New App Registration. This action will generate and gives you a Client ID, Client Secret
* Click on “**Generate a developer token for this application**” link to generate Yammer API Token. Please copy & save this token local for future reference.



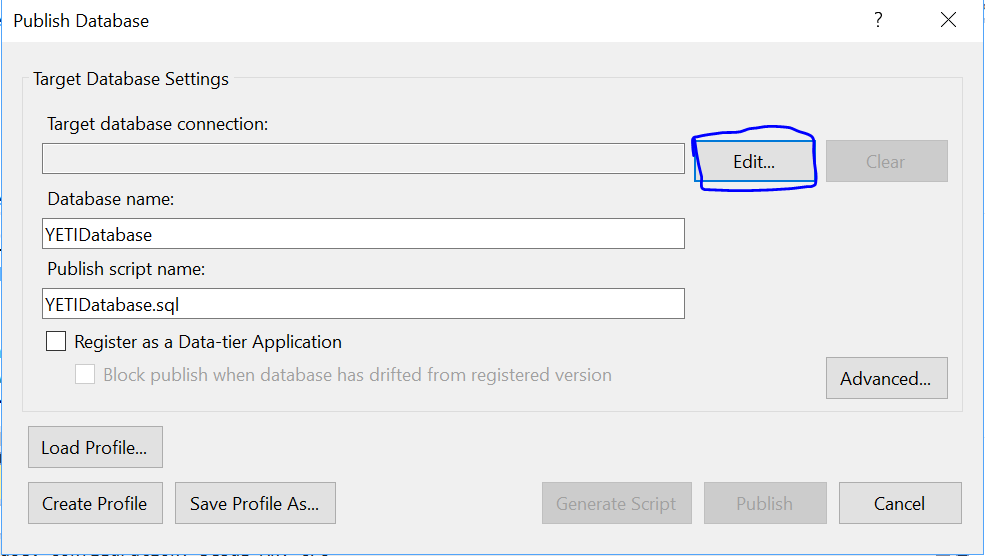
## **Yammer Extraction Tool Database Creation**

Follow below steps to create/publish database in your local machine.

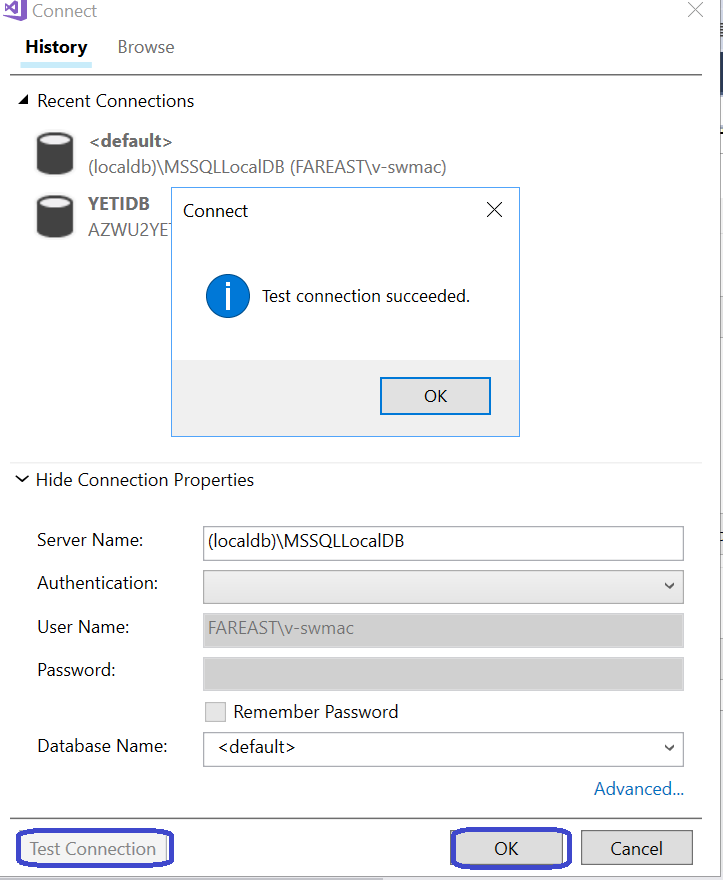
* Right click on “YETIDatabase” Project from solution and click on “Publish”.

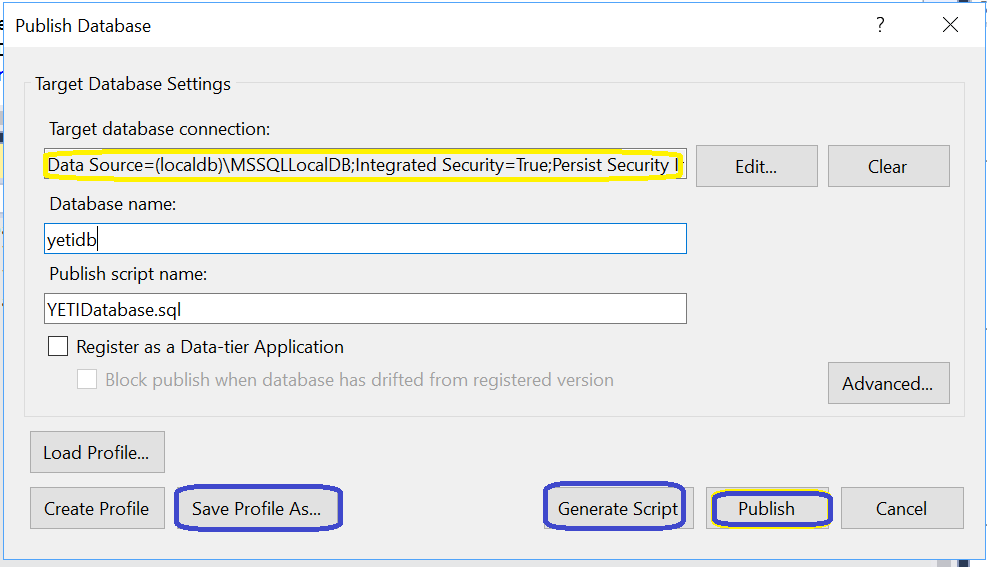


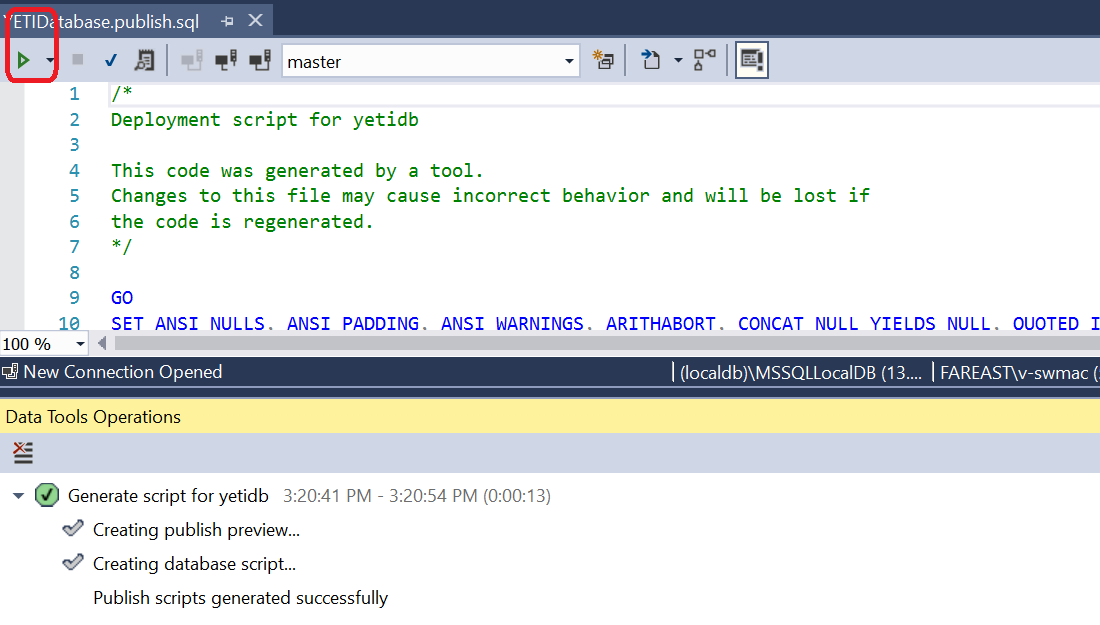
* Click on “Edit” button to provide your local database connection string.



* Provide the server name, authentication type and database name. Click on “Test Connection” to ensure the given information is correct. then press OK.



* We can save a profile using “Save Profile As” button - so that we don’t have to re-enter connection related settings next time. Now, we can generate the script by clicking on “Generate Script” to manually run the script in our database. Or else, click on “Publish” button to publish the database in our local machine. 
* If we use “Generate Script” button, we can see compared script in visual studio’s new tab. We can check the script and after that press Execute



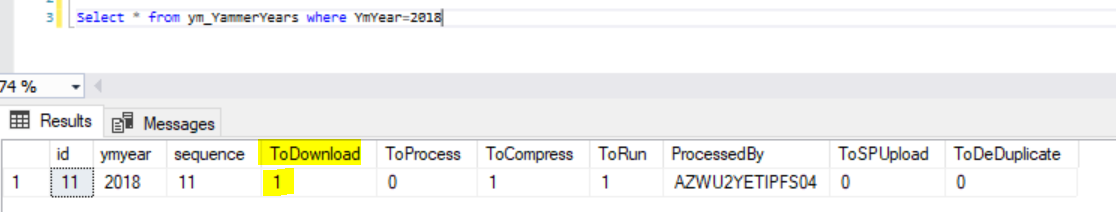
After clicking execute, changes are published successfully to the local database.

Check Database using SQL Server Management Studio now with local settings. You will be having a database created YETIDB. Finally, run the script which is available in “Scripts” folder under “YETIDatabase” project to have master data available in your database.

## **Datasets & Usage**

In Database section, we have few major tables that can be used frequently for monitoring.

[dbo].{Ym\_YammerYears} – This table used to verify which stage (Download,Process, Compress & SharePoint upload) is current being running. Initially, please make sure that “ToDownload” flag set to 1.



**[dbo].[YM\_EventLogs]** – We can check the logs of different Console application processes running through task schedulers. We can also check the error logs immediately in this table.

**[dbo].[YM\_ExportDetails]** – Used to monitor different status from “Download from yammer” to “Sharepoint Upload” status where the file stands.

After downloading the yammer data to Fileshare servers, data will be processed and saved into below tables in our database.

* + [dbo].[YM\_Messages]
  + [dbo].[YM\_Files]
  + [dbo].[YM\_Pages]
  + [dbo].[YM\_Topics]
  + [dbo].[YM\_Users]