Box Introduction

Box connector will crawl content from Box repository. The connector will retrieve the supported elements using the RESTful API (Content API Basics 2.0 version), for authentication will use Box API (that uses OAuth 2).

Features

Some of the features of the Box connector include:

- Ability to perform either full or incremental crawling (so that only new/updated documents are indexed)
- Possibility of exclude a folder or a set of folders and their content.
- Possibility of exclude or include elements (folders or files) by file name or folder name using regular expression (regex patterns)
- Metadata extraction
- it is search engine independent
- Runs from any machine with access to the given Box account
- Fetches access control lists (ACLs)

Content Retrieved

The Box connector retrieves several types of documents. Listed below are the inclusions and exclusions to these documents.

Include

- Folders
- Folder's collaborations
- Files
- Box Note
- Bookmark
- Google Doc
- Google Spreadsheet
- Word document
- PowerPoint document
- Excel Spreadsheet
- File's comments
- File's tasks
- Task's assignments
- Users and Groups (memberships)
- Events (for Incremental crawls)
    - ITEM_CREATE
    - ITEM_UPLOAD
    - COMMENT_CREATE
    - ITEM_MOVE
    - ITEM_COPY
    - TASK_ASSIGNMENT_CREATE
    - ITEM_TRASH
    - COLLAB_ADD_COLLABORATOR
    - ITEM_RENAME

Exclude

- Example Doc Type

Limitations

Due to API limitations, Box connector has the following limitations:

- Box connector crawls only the latest version of files.
- Box connector does not crawls any Trash items (folders or files)
- Incremental limitations
    - When there is change on a Box Note will not reflect in the incremental crawl.

- Box API request limitation
    - In certain cases, Box needs to enforce rate-limiting in order to prevent abuse by third-party services and/or users. In the event that an excessive level of usage is reached, a standard 429 Too Many Requests error will be returned, with an indication of when to retry the request. In the event that back-to-back 429s are received. RETRY HEADER HTTP/1.1 429 Too Many Requests Retry-After: {retry time in seconds}.
    - The connector has a Back-off implementation in case of connection problems. For instance, in case of 429 Http error, you should add the pattern .*429.* in order to retry this error.

Please see more information at, https://developers.box.com/docs/#rate-limiting

Future Development Plan

The following features are not currently implemented, but are on the development plan:

- Example future plan

Anything we should add? Please let us know.