

Microsoft Store App Update Automation: Technical Flow Documentation

Sai Aditya Kukudala

June 9, 2025

1 Introduction

This document describes the technical flow for automating Microsoft Store app updates using the `autoPublisher` CLI. It covers the API endpoints used, the structure of payloads, and the sequence of operations performed to update an app submission.

2 Update Flow Overview

The update process for a Microsoft Store app via automation consists of the following high-level steps:

1. Authenticate and obtain an access token.
2. Delete any existing in-progress submission.
3. Create a new submission.
4. Prepare and update submission metadata.
5. Upload package and media assets.
6. Commit the submission.
7. Poll for submission status.

3 APIs Used

All API calls are made to the Microsoft Store Dev Center API (<https://manage.devcenter.microsoft.com/v1.0/my/>). Authentication uses Azure AD OAuth2.

- **Token Endpoint:** `https://login.microsoftonline.com/{tenantId}/oauth2/token`

- **Get App Data:** GET applications/{productId}
- **Delete Submission:** DELETE applications/{productId}/submissions/{submissionId}
- **Create Submission:** POST applications/{productId}/submissions
- **Get Submission Metadata:** GET applications/{productId}/submissions/{submissionId}
- **Update Submission Metadata:** PUT applications/{productId}/submissions/{submissionId}
- **Commit Submission:** POST applications/{productId}/submissions/{submissionId}/commit
- **Get Submission Status:** GET applications/{productId}/submissions/{submissionId}/status

4 Authentication

OAuth2 client credentials flow is used. The following payload is sent to the token endpoint:

```
grant_type=client_credentials
client_id=<client-id>
client_secret=<client-secret>
resource=https://manage.devcenter.microsoft.com
```

The resulting `access_token` is used as a Bearer token in all subsequent API requests.

5 Submission Metadata Payload

The main payload for updating a submission is a JSON object matching the `ApplicationSubmissionRequest` schema. Only a subset of fields are required for updates. Example structure:

```
{
  "ApplicationCategory": "HealthAndFitness",
  "Pricing": { ... },
  "Visibility": "Public",
  "TargetPublishMode": "Immediate",
  "TargetPublishDate": "1601-01-01T00:00:00Z",
  "Listings": {
    "en-us": {
      "BaseListing": {
        "Description": "...",
        "Images": [
          {
            "FileName": "Screenshot1.png",
            "FileStatus": "PendingUpload",
            "ImageType": "Screenshot"
          }
        ]
      }
    }
  }
}
```

```

        }
      ],
      ...
    }
  },
  "ApplicationPackages": [
    {
      "FileName": "myapp.msixbundle",
      "FileStatus": "PendingUpload"
    }
  ],
  "Trailers": [
    {
      "VideoFileName": "Trailer_demo.mp4",
      "TrailerAssets": {
        "en-us": {
          "Title": "demo",
          "ImageList": [
            {
              "FileName": "TrailerImage_demo.png",
              "Description": null
            }
          ]
        }
      }
    }
  ]
}

```

Note: Fields like `ApplicationPackages` and `Images` must be managed carefully. Existing assets are marked `PendingDelete`, and new assets are added with `PendingUpload`.

6 Sequence of API Calls

1. **Authenticate:** Obtain access token.
2. **Get Current Submission:** GET `applications/{productId}` to check for pending submissions.
3. **Delete Submission:** If a pending submission exists, DELETE it.
4. **Create Submission:** POST to create a new submission.
5. **Get Metadata:** GET submission metadata for the new submission.
6. **Update Metadata:** PUT updated metadata payload.

7. **Upload Assets:** Use the `fileUploadUrl` from metadata to upload the package via Azure Blob Storage.
8. **Commit Submission:** POST to commit the submission.
9. **Poll Status:** GET status until published or failed.

7 Error Handling

Common errors include:

- **InvalidState:** Submission is not in a modifiable state. Solution: Delete and recreate the submission.
- **Schema Errors:** Payload does not match expected schema. Solution: Validate JSON structure and field types.
- **Authentication Errors:** Invalid credentials or expired token.

8 References

- Microsoft Store Submission API Overview
- Create and manage app submissions