

SharePoint Document Management for Energy and Project Developers

Project folders in SharePoint, created, named and secured by Salesforce Flow from the first Opportunity onward

CloudFiles is Document Automation for Salesforce. This guide describes a pattern we see among energy and project developers: SharePoint integration for project folders, run from Salesforce records instead of by hand. The same approach works when the files live in AWS S3, Google Drive or Azure Blob, and the guide notes where those differ.

What problem do energy and project developers have with project folders in Salesforce?

A development business tracks landowners, sites, grid applications and investors in Salesforce, while the documents behind each project (land agreements, surveys, permits, interconnection studies) live in SharePoint. The two drift apart. Someone creates a SharePoint folder by hand, names it slightly differently from the Opportunity, and pastes a URL onto the record. When the Account is renamed or the folder moves, the stored URL breaks. People work in two windows and nobody is sure which folder is the real one.

The fix is to let Salesforce own the folder structure, so every record gets its own folder, created by Flow and opened from the record page.

Which Salesforce objects are involved?

A typical energy developer uses three objects:

- **Account:** the landowner, offtaker or counterparty. Each Account gets a top-level folder in SharePoint.
- **Opportunity:** the development deal. Each Opportunity gets its own folder inside the Account folder.
- **Project (custom object):** the physical project. When a Project is linked to an Opportunity, it gets a project folder with a standard subfolder layout.

The CloudFiles file browser widget sits on the Lightning record page of each object, so a user on a Project record sees exactly that project's folder in SharePoint.

Where do the files live: SharePoint, AWS S3, Google Drive or Azure Blob?

Most developers in this pattern keep project files in SharePoint, because the wider business already works there. CloudFiles works on the live SharePoint files through Microsoft's own API, so there is one copy, in your storage. Files added in Salesforce appear in SharePoint, and files added in SharePoint appear on the Salesforce record once the folder is linked.

Teams that archive site data in AWS S3 or Azure Blob can connect those too, with event-driven sync turned on per connection, and Google Drive works on the live files just as SharePoint does.

How do we automate SharePoint folder creation from Opportunities?

To automate SharePoint folder creation from Opportunities, an admin builds a record-triggered Flow using CloudFiles Flow actions. No code is required, and Apex is available if you prefer it.

1. **Account created.** A record-triggered Flow on Account calls **Create Folder** to create the Account folder in the chosen SharePoint library. The folder is linked to the record.
2. **Opportunity created.** A second Flow calls **Get Connected Folder** on the parent Account to find its folder, then calls **Create Folder** to add the Opportunity folder inside it.
3. **Project linked.** When the Project lookup on the Opportunity is filled in, a Flow calls **Create Folder** for the project folder and its subfolders (for example Land, Permits, Grid, Engineering).
4. **Existing files moved.** If files were already attached to the Opportunity in Salesforce, **Copy Salesforce Files** places them in the new SharePoint folder and leaves a link on the record.
5. **Folder metadata stamped.** **Update Drive Columns** writes Salesforce values, such as project stage or region, into SharePoint columns so the folder is searchable in SharePoint too.

The Site Id, Drive Id and Folder Id a Flow needs are shown in the View Metadata dialog in the CloudFiles Content Library, each with a copy button. When a folder already exists, the library's conflict setting decides what happens: Rename, Replace, No Op or Fail.

Some actions are asynchronous. They return nothing, and completion arrives later as a CloudFiles event that a record-triggered or platform-event Flow handles.

How do we rename the SharePoint folder when the Account changes?

Landowner and counterparty records change: a company renames itself, or an address is corrected. To rename the SharePoint folder when the Account changes, a record-triggered Flow on Account runs when the Name or address fields change. It calls **Get Connected Folder** to find the linked folder and renames it to match the new Account name.

Because CloudFiles folder links survive record renames, the Account, its Opportunities and its Projects keep their folders throughout. Nothing depends on a pasted URL, so nothing breaks. This is the point where teams replacing Files Connect, Document Extractor or a sync tool that breaks on folder renames notice the difference first.

Who can see the project folders in Salesforce and SharePoint?

CloudFiles uses two identities. Flow automation runs as a service principal or an integration user. The widget and Content Library run as each user through their own storage connection, so people see what their SharePoint access allows.

For SharePoint, a service principal is the cleaner choice: it authenticates as the CloudFiles app in your Microsoft tenant, uses no license, has no password or MFA to break, and shows an "Active for automations" badge. Your Microsoft 365 admin can limit CloudFiles to only the SharePoint sites you choose, granting access site by site.

Permissions go from Salesforce to SharePoint by Flow:

- **Update Permissions** grants or revokes read or write on a folder by email, Salesforce user or group, for example when the Opportunity owner changes.
- **Create SharePoint Site Group** and **Update SharePoint Site Group Members** keep a project team's site group in step with the Opportunity team.

When an admin designs it that way, users need no SharePoint rights for the widget experience. External parties such as landowners or consultants can receive a share link from **Create Link**, protected by an email wall, password or verification code, with an expiry and view tracking.

Is SharePoint integration for project folders secure?

- ISO/IEC 27001:2022 certified, with ISO/IEC 27017:2015 and ISO/IEC 27018:2019.
- Independently audited for SOC 2 Type II.
- Compliant with HIPAA (BAA available). Compliant with GDPR.
- Hosted on AWS in four regions selected per customer account: United States, EU, United Kingdom and Australia.
- TLS 1.2 minimum in transit; AES-256 at rest on production data stores.
- Your files stay in your own storage. CloudFiles transfers them over encrypted connections and does not re-host them.

ISO certificates, the SOC 2 Type II report and a penetration test summary are available under NDA from the CloudFiles Trust Center.

Get started

Install CloudFiles Document Management from AgentExchange and start a free 14-day trial, or book a demo at cloudfiles.io.