

Document AI for Telecom and Utility Bills: Invoice Processing in Salesforce

Split PDF master bills by account, extract header fields and line items, and write invoices to Salesforce without re-keying

CloudFiles Document AI is part of CloudFiles, Document Automation for Salesforce. This guide describes a pattern we see at companies that pay telecom and utility vendors across many accounts and sites, and track those bills in Salesforce. The work is invoice processing at volume: long master bills, many accounts per bill, and line items that someone has to key by hand. Intelligent document processing inside Salesforce runs it as one process: the bill arrives, is split, is read, is matched to accounts and vendors, and is written as records.

Why are telecom and utility bills so hard to process?

A single telecom or utility vendor often sends one master invoice that covers every account the customer holds with them. Each account section carries its own header, service address, charges, usage, taxes and fees. The finance team needs each account's bill as its own invoice in Salesforce, with its line items, tied to the right account and vendor.

Doing that by hand means paging through the master bill, cutting it into pieces, and re-keying header fields and line items for every account. Template-based OCR tools struggle here because vendors change their bill layouts, and a value that moves on the page breaks the template. The result is manual data entry that grows with every new account and every new vendor.

How do invoices get into the process?

In this pattern, CloudFiles Document Management gives every account and vendor its own folder in AWS S3, created automatically. Each vendor folder has an Invoices folder. When a bill is dropped into a vendor's Invoices folder, that drop triggers processing. There is no separate intake queue to watch.

The same process can start from other places:

- an invoice attached to an email, through Email-to-Case,
- a file uploaded to a Salesforce record,
- a screen Flow where a user uploads a batch of bills,
- a folder in SharePoint, OneDrive, Google Drive, Azure Blob, Box or Dropbox connected through Document Management.

Scanned and faxed bills are sent to OCR, and native PDF text is read directly, so paper and digital bills follow the same path.

How does a master invoice get split by account?

CloudFiles: Split Document using AI takes the master bill and splits it into one invoice per account. It names each output file from its content, so every piece carries the account it belongs to in its file name, and it saves the pieces as Salesforce Files.

This is a split PDF step driven by what the document says rather than by page counts or separator sheets. When the split finishes, a document-split event comes back, read with CloudFiles: Get Event Details, and the Flow hands each piece to the next step.

How are header fields and line items extracted and written?

Each per-account invoice runs through CloudFiles: Document Mapper. A saved mapper configuration reads the invoice and creates or updates an Invoice record plus its Invoice Line Item child records.

- **Header fields:** vendor, account number, invoice number, invoice date, billing period, due date, service address and totals.
- **Line items:** each charge line, with its description, quantity or usage, rate and amount, written as its own Invoice Line Item record.

Fields are described in plain language, with no layout templates, so a vendor that redesigns its bill does not break the process. Lookups are resolved per field: Get Existing finds the vendor and the account, Create New adds a record when the account is new, and Call Flow Action runs the org's own matching logic where it has one. A per-field option that protects existing values keeps fields that the finance team has already set from being overwritten.

Writes go through standard Salesforce DML, so validation rules and field-level security apply to every Invoice and line item.

Can finance review invoices before they are written to Salesforce?

Yes. In Document Mapper, the "When automation runs finish" setting decides what happens after reading. "Auto approve" writes the records. The review queue setting holds the run instead, and the run shows Pending Review. A reviewer opens it, edits cells, adds a missing line item row if needed, and chooses "Approve & Write to Salesforce" or Reject.

A lookup with no match or with several matches forces review regardless of the setting. If an account number on a bill does not match one account in Salesforce, a person decides before anything is written.

What is recorded for each bill?

Each run is recorded in Salesforce as a CloudFiles DocAI Run record, with its status: Processing, Completed, Pending Review, Failed or Rejected. Reprocessing a bill creates a new run and keeps the old one. Finance can report on runs the way they report on anything else in Salesforce: by vendor, by status, by month.

Which pieces does this invoice processing pattern use?

Step	CloudFiles piece
Folder per account and vendor, drop triggers processing	CloudFiles Document Management with AWS S3
Read scanned and native bills	OCR inside Document AI
Split the master bill into one invoice per account	CloudFiles: Split Document using AI
Write Invoice and Invoice Line Item records	CloudFiles: Document Mapper
Hold for review, approve and write	Mapper review queue
Track each run	CloudFiles DocAI Run records

Can we ask questions across all the bills?

Yes. CloudFiles: Create Dataset combines bill files with SOQL sources, such as the Invoice and Account objects, refreshed on a schedule. CloudFiles: Query Document/Dataset then answers plain-language questions across them, such as which accounts were billed a new fee this period, and returns a structured answer to the Flow.

Do our bills train the AI model?

No. CloudFiles does not train or fine-tune AI models on your documents. Document AI uses large language models and OCR. Security documents are available under NDA from the CloudFiles Trust Center.

Get started

Install CloudFiles Document AI from AgentExchange and start a free 14-day trial or a Test Drive, or book a demo at cloudfiles.io. A free sandbox proof of concept is offered, so you can run your own master bills through the process first.