insightsoftware

Portfolio Integration User Guide

LeaseAccelerator

Version 25.2

Document Information

Notices

Copyright

Disclaimer

This guide is designed to help you to use the LeaseAccelerator applications effectively and efficiently. All data shown in graphics are provided as examples only. The example companies and calculations herein are fictitious. No association with any real company or organization is intended or should be inferred.

Contents

About LeaseAccelerator Integrations

LeaseAccelerator is the leading enterprise lease lifecycle automation platform for equipment and real estate lease administration and management.

LeaseAccelerator ensures that all your equipment and real estate leases comply with the latest lease accounting standards. The application can quickly generate all the debits and credits needed for your general ledger as well as account for aircraft, trucks, computers, photocopiers, office buildings, retail stores, distribution centers, and health care facilities.

LeaseAccelerator is a cloud-based application designed and built to facilitate information exchange with both on-premise and cloud applications to provide a complete end-to-end solution for leasing business cycles. LeaseAccelerator is hosted on Amazon Web Services, the largest global web hosting service provider, with an additional provider for backup and disaster recovery protection.

This guide covers integration requirements and common business cases that are typically required for process and data integration between LeaseAccelerator and other systems. Two common integrations are between LeaseAccelerator and enterprise resourcing planning (ERP) systems that have accounting functions, such as accounts payable, and between LeaseAccelerator and other property management solutions. To ensure accurate accounting, LeaseAccelerator also integrates with business systems that maintain foreign exchange (FX) rates and other master data.

Accounting systems and ERPs

Typically, LeaseAccelerator generates financial entries for your accounting system or ERP system. LeaseAccelerator has multiple integration methods to extract transactions in different forms so they can be uploaded into your accounting system or ERP. To achieve the integration, LeaseAccelerator requires feedback from your accounting system or ERP.

Real estate and property management solutions

Some companies use vertical solutions to manage the operation, maintenance, inspection, and utilization of leased real estate or equipment. In such cases, LeaseAccelerator can exchange information with these systems.

Integrating with these systems aim to synchronize the lifecycle of leases between these systems and LeaseAccelerator. Hence while Real Estate and Association Management systems (AMS) cater to asset maintenance, utilization, operation, etc., LeaseAccelerator caters to all accounting aspects of leased equipment or assets.

This guide explains integration alternatives, approaches, use cases, process flows and data mapping with property management, fleet management, real estate, and asset management systems.

Integrating Portfolio and Configuration Data

The lifecycle of any lease contract (also called “deal”, “schedule” or “agreement”) consists of three phases:

1. Initiate a lease schedule including its assets (equipment or real estate) and all relevant information, terms and conditions, accounting information, variables, and accounting standards.

2. Update schedule information with modifications that occur throughout the lifetime of the lease agreement which may include renewal or re-negotiation on the leased assets such as termination, additions, buy-out, return.

3. Termination of the lease agreement, according to the End-of-Term (EOT), which may imply some assets will be bought or going into evergreen.

Some information in a lease agreement in Lease Accounting Manager is linked to “configuration” data maintained in master files. There are three categories of information that must be synchronized with Lease Accounting Manager:

Portfolio data: Representing individual lease schedules and all related information, including:

- Schedule
- Details
- EOT options
- Notifications
- Interim rent
- Step payments
- Payment adjustments
- Schedule-related expenses
- Invoice-related expenses
- Paid-related expenses
- Category Lease Rate Factor (LRF)

Configuration data: Representing master data that will be linked to and commonly used by lease schedules, including:

- Companies
- Addresses
- People

- FX rates
- Ledger segment values such as account or cost center

> **Note:** Information about integrating configuration data is discussed in the Enterprise Integration Guide.

Accounting configuration data: Representing various rules and values needed to calculate lease accounting per the proper accounting standard. Accounting configuration is not discussed in this guide as it is not subject to integration and must be defined from within Lease Accounting Manager.

Integration Methods

Lease Accounting Manager offers API and file-based integrations, both of which mitigate security risks.

APIs

For Inbound and outbound integration, Lease Accounting Manager offers a full library of well-documented and secured RESTful APIs to interface with ERPs, real estate administration systems, fixed asset systems, and any other system that can call RESTful APIs and process XML.

Lease Accounting Manager Restful APIs can be used for:

- Enterprise integrations - see Enterprise Integration guide for more details. Further information regarding enterprise API's can be found in the API Developer's guide.

  Synchronizing reference data such as lessee, project, property tax authority, address, business unit, entity, funder, geo, company, and contact information

  Generating ledger exports and asset-level detail reports

  Generating due payments to ERP accounts payable and importing actual disbursements

  Importing currency exchange rates

- Portfolio integrations – Portfolio use cases are discussed in detail in this guide. Further information regarding portfolio API's can be found in the API Developer's guide.

  Importing leased assets and deals as well as data for subledger entries such as disbursements and payment adjustments

  Making edits to leased assets

  Making modifications to leases

  Recording events or intentions for future events such as lease terminations, renewals, and buyouts

  Searching for leased assets, contacts, and schedules

- SSO – see Single Sign-On Integration guide for more information. Further information regarding SSO API's can be found in the API Developer's Guide.

Provisioning user privileges

Revoking user privileges

File-based

In addition to APIs, Lease Accounting Manager supports simple file transfer and file exchange.

Lease Accounting Manager includes support for importing data from standard spreadsheet-based templates and can generate reports in XML, CSV or XLSX (Microsoft Excel) formats. The Reporting Engine supports automatic transfer of generated reports via SFTP (secure file transfer protocol), allowing reporting to be leveraged as a data export mechanism.

For exports (outbound), Lease Accounting Manager users can export files manually at any time or can schedule files to be automatically generated and transferred to a designated folder on a client-provided SFTP server (and/or e-mail) at specific frequency (daily, weekly, monthly, etc.). Those files can then be processed in an automated manner on the client's ERP system.

For imports (inbound), the client's IT team extracts files at a specific time in a specific format and posts them to a specified folder on the client-provided SFTP server. Lease Accounting Manager can be configured to monitor the folder. Once files with a specific, agreed upon name are placed in this folder, Lease Accounting Manager processes the file by validating and importing the data into Lease Accounting Manager then sending a notice of failure or success via email. The processing can also be monitored using the System Operations Console in the LeaseAccelerator user interface (UI).

Integration Approaches

Point-to-point integration

Lease Accounting Manager supports point to point or "direct" integration through its RESTful APIs and file-based methods.

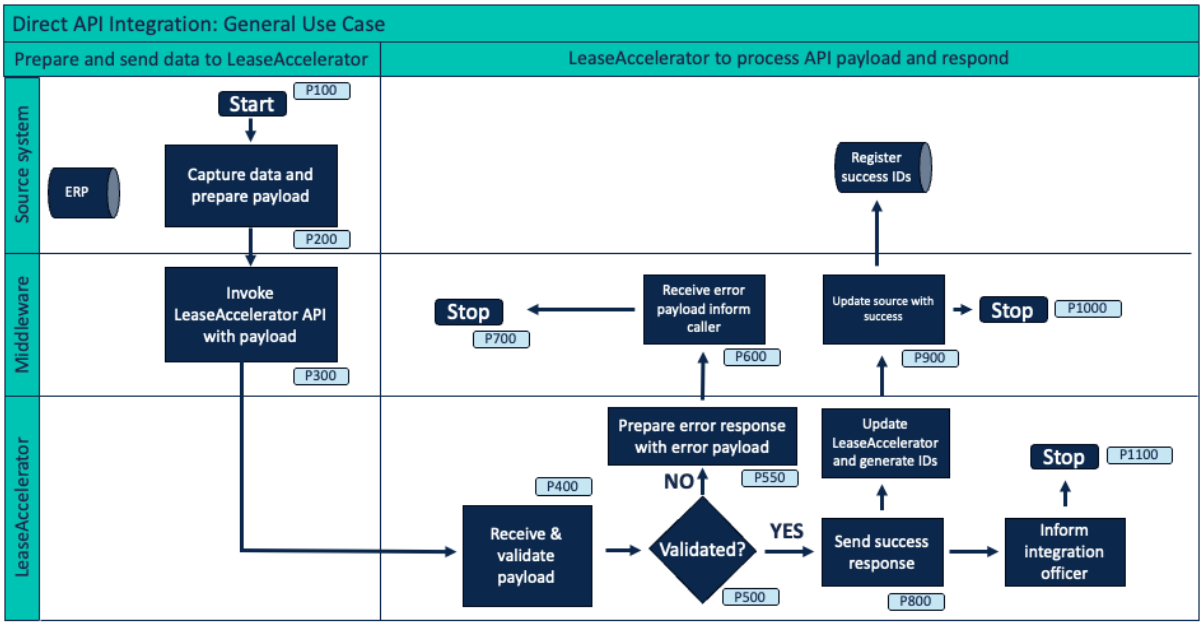
In all cases, there are three possibilities:

1. Data valid and complete.
2. Invalid data received will trigger rejection.
3. Data valid but missing information (gaps). In this case, Lease Accounting Manager will generate the control file with missing data. This file can be fed with missing data and reloaded directly.

Direct integration using APIs

In this use case, the user will extract data from the source system, prepare the workload as per the API XML format and call Lease Accounting Manager API though the Middleware. The user must send complete and correct data as Lease Accounting Manager will respond with success or failure response.

When calling Lease Accounting Manager API, the user can decide what Lease Accounting Manager should do for warnings and errors. Please refer to Lease Accounting Manager API.



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture Companies data and prepare payload	Source system user
P300	Invoke LeaseAccelerator API with payload	Middleware
P400	Receive & validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload & inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P850	Update LeaseAccelerator database and generate IDs	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform integration officer	LeaseAccelerator

File-based integration using SFTP

The real estate user will collect data from the source system into the Lease Accounting Manager-supplied template (could be XLSX, or XML) and then place the file into the agreed upon server in the SFTP.

Lease Accounting Manager will detect and process the file. If the data is complete and validated, Lease Accounting Manager will update its database and generate a **"Control file"** carrying;

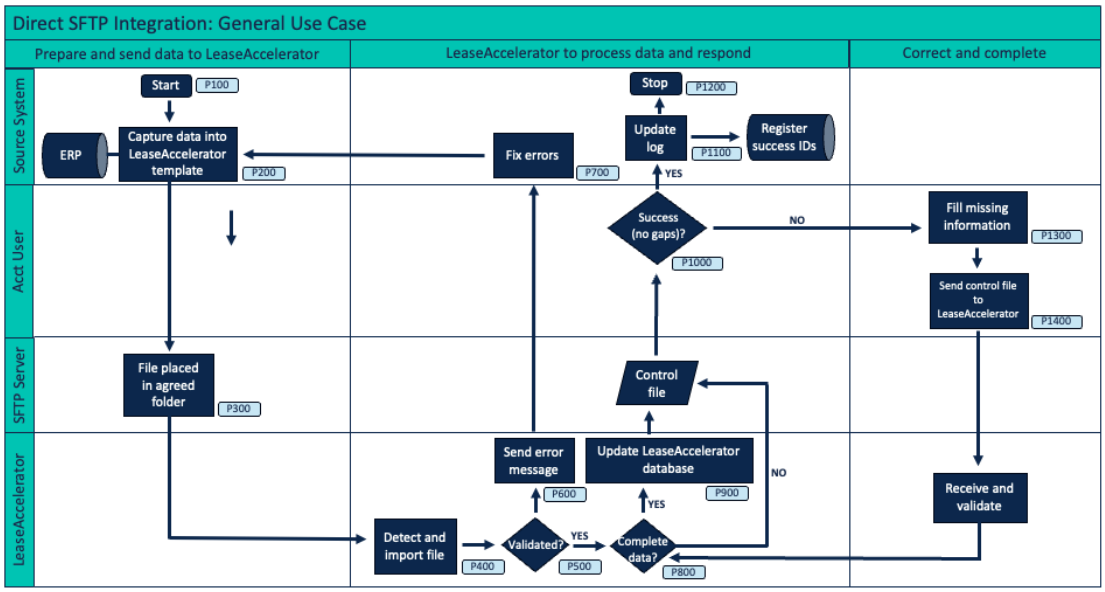
- the data imported and
- a list of the Lease Accounting Manager IDs

Lease Accounting Manager will pass the control file to the accounting user who should update the real estate system with the new Lease Accounting Manager- generated IDs.

If the data is found to be incomplete, Lease Accounting Manager will generate the **control file** highlighting missing data (gaps) but no IDs shall be generated.

Lease Accounting Manager will pass the control file to the accounting user who should input the missing data and send the file back to Lease Accounting Manager by either:

- Email callback



Upload through bulk import feature

Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture data into LeaseAccelerator template	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Update LeaseAccelerator database	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system

Process Step ID	Process Step Description	Phase
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

Which Data to Send

In some use cases, the data sent to Lease Accounting Manager as payload (via API or SFTP) must only include the changes (delta). Upon successful processing of the data, Lease Accounting Manager will return a set of success IDs so that the source data can be tagged to prevent resending the same data twice.

Depending on the use case, sending a full dump of data or resending previously processed data may cause errors or maybe ignored by Lease Accounting Manager. Resending previously processed data – even if does not generate errors - may trigger the accounting engine to re-run the calculations unnecessarily – even if the incoming data exactly matches the existing data which will negatively affect system performance and wait time for reports to be updated.

Integration Use Cases

1. Enter New Schedule

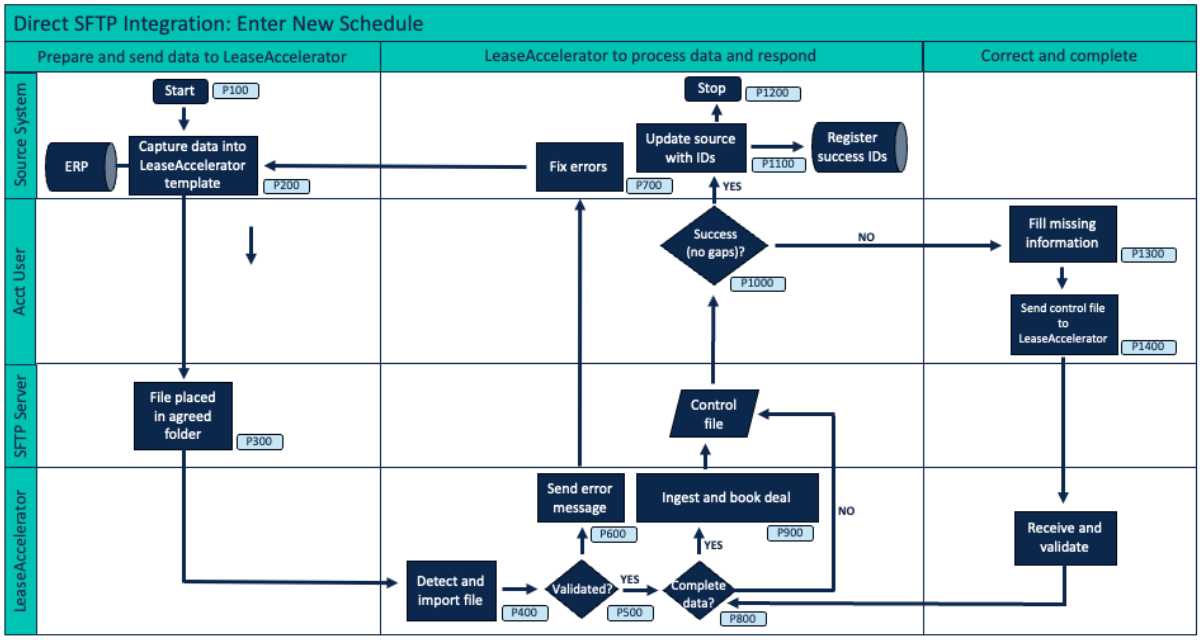
New lease schedule(s) have been captured in the external system and need to be communicated/transferred into LeaseAccelerator.

Data range/payload allowed: Delta and dump

LeaseAccelerator strongly recommends extracting only new deals from the source system so the payload sent to LeaseAccelerator only includes the new deals.

However, if the payload included deals that already exist in the database, LeaseAccelerator will ignore them.

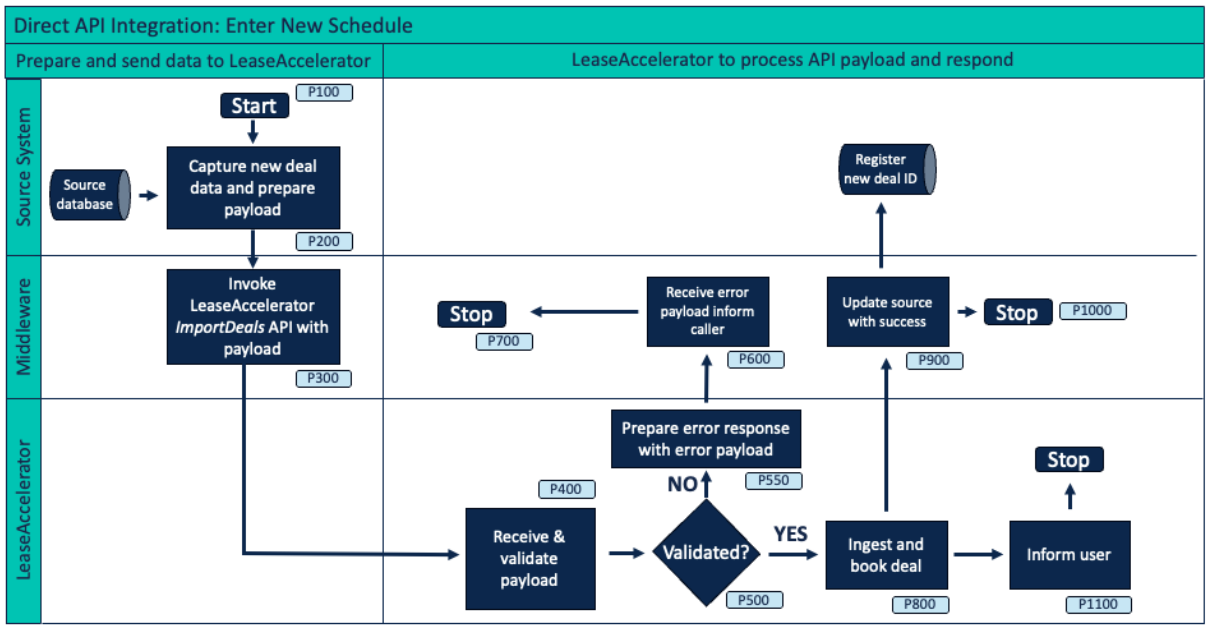
1.1 Enter new schedule SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture new deal data and prepare upload file	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Ingest and book deals	LeaseAccelerator

Process Step ID	Process Step Description	Phase
P1000	Success with no gaps?	Accounting user
P1100	Update source with IDs	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

1.1 Enter new schedule – API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture new deal data and prepare payload	Source system user
P300	Invoke LeaseAccelerator **ImportDeals** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Ingest and book deal	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

1.2 Data fields, format, and structure

Method	Format	Procedure/ Usage
API	XML	Please refer to API user guide for invoke process, request, and response parameters.
SFTP	XML	Same structure as API / XML payload. File name: ImportDeals +YYYYMMDDHHMM.xml
SFTP	XLSX	Data to be loaded into the PIW template. File name: ImportDeals +YYYYMMDDHHMM.xlsx. First row for column names must match the field names listed below.
NOTE – The data fields available via the XLSX format are listed in detail in the Client Implementation Guide (CIG) and can be found in Ask Alex in the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.		

2. Correct Asset Details

Asset details were incorrectly entered/captured in the external system and the corrected details need to be updated in LeaseAccelerator. Alternatively, the lessor provided revised asset detail information which has been updated in the external system and needs to be communicated to LeaseAccelerator.

Using an Asset Detail Import is the only way that a user can update all asset details listed on the Details tab of the Portfolio Intake Workbook (PIW), such as ShipTo (or Facility Code), Reference Number, and any financial information for the asset. Unlike the Asset Bundle and the Asset Management Bulk Update Import, which are used to update certain changed information about an asset only, the Asset Detail Import will replace all asset information previously captured in LeaseAccelerator. Essentially, the data is re-imported, not simply adjusted to only include new or changed asset information. Use caution when performing this action. If you forget to list an asset, it will be deleted from LeaseAccelerator. If you do not enter correct cost information, LeaseAccelerator will capture the incorrect cost information provided with this import.

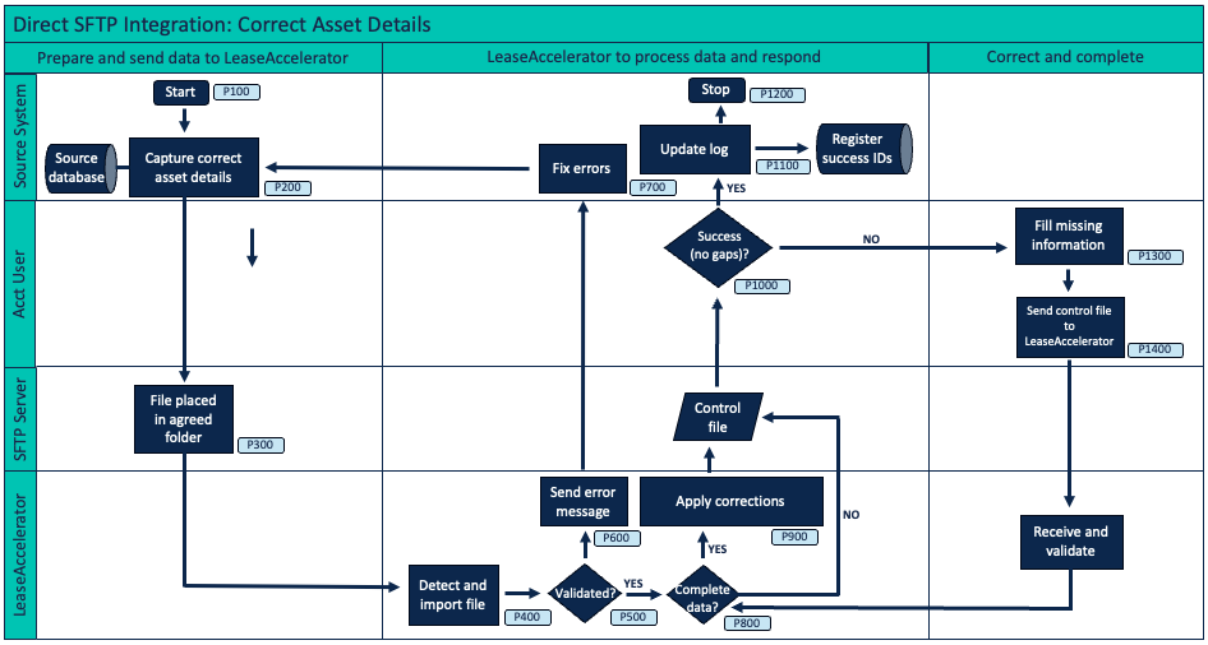
> **Note:** This scenario does not apply to adding or removing assets, nor to updating assets to capture a like-for-like swap. If the schedule has associated journal entries that have been closed in LeaseAccelerator, and the value or calculated rent for the updated assets do not match the value and calculated rent of the assets prior to update, then the update will be rejected, and any changes will need to be communicated to the responsible accounting staff for manual entry into LeaseAccelerator.

Data range / payload allowed: Delta and dump

LeaseAccelerator strongly recommends extracting only assets with corrected details from the source system so the payload sent to LeaseAccelerator includes only the updated assets.

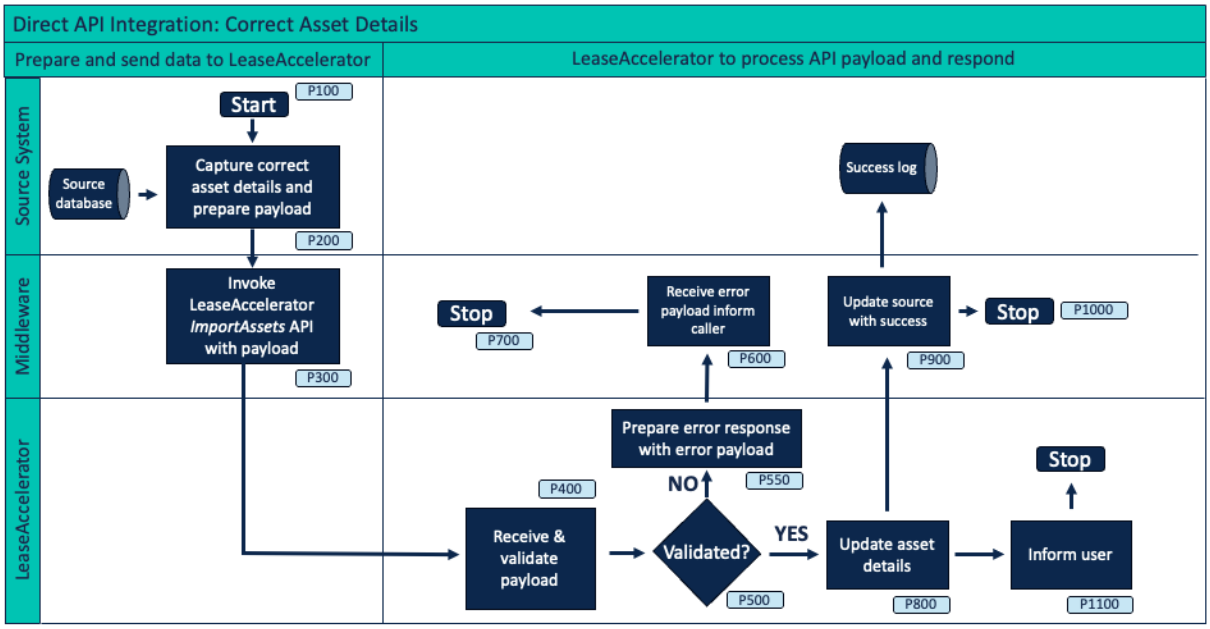
However, if the payload includes assets with non-changing details, LeaseAccelerator will ignore them but trigger the accounting engine to unnecessarily re-run the calculations, affecting performance and reporting availability.

2.1 Correct asset details – SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture correct asset details and prepare upload file	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Apply corrections	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update source with IDs	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

2.2 Correct asset details – API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture correct asset details and prepare payload	Source system user
P300	Invoke LeaseAccelerator **ImportAssets** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Apply corrections	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

2.3 Data fields, format, and structure

Method	Format	Procedure/ Usage
API	XML	Please refer to API user guide for invoke process, request, and response parameters.
SFTP	XML	Same structure as API / XML payload. File name: ImportAssets +YYYYMMDDHHMM.xml

Method	Format	Procedure/Usage
SFTP	XLSX	Workbook name: ImportAssets +YYYYMMDDHHMM.xlsx. Sheet name: Details. First row for column names must match the field names listed below.
SFTP	CSV	File name: ImportAssets +YYYYMMDDHHMM.csv. The first row of csv files must include the field names as headers separated by commas.
Note: *The data fields available via the XLSX and CSV formats can be found on the Details tab of the Portfolio Intake Workbook (PIW) found in Ask Alex in the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.*		

3. Change Key Operational Attributes for Assets

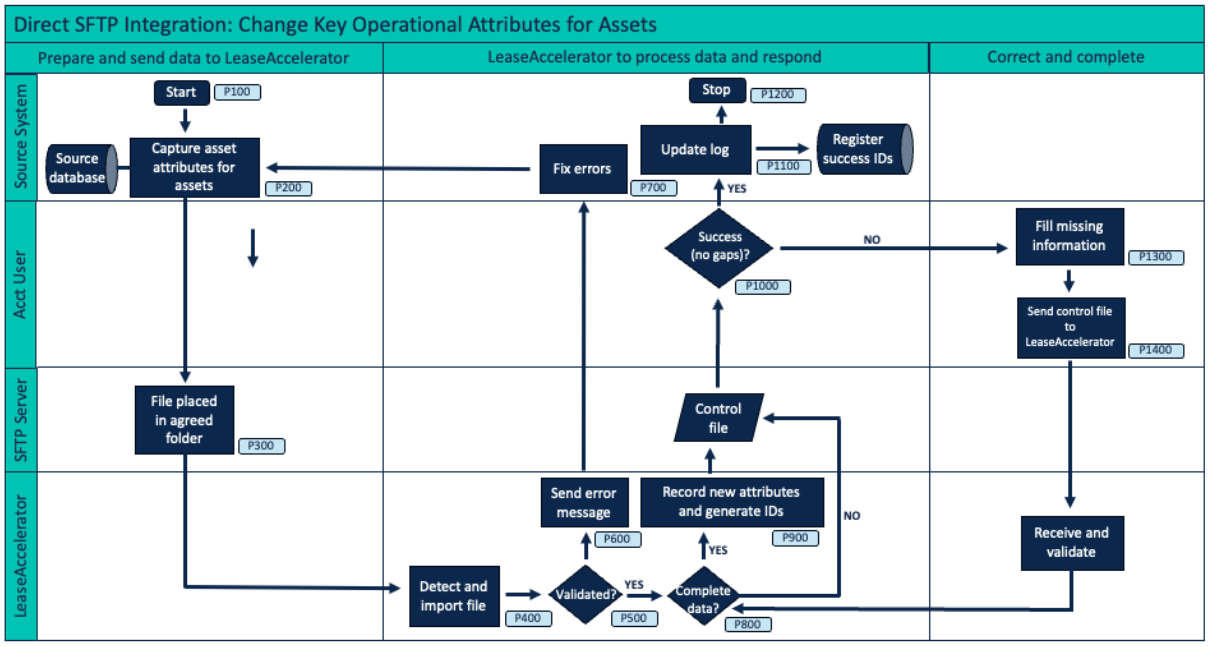
Key operational attributes (asset owner, asset user, serial number, cost center, IP address, etc.) have changed for one or more assets in the external system and need to be communicated to LeaseAccelerator.

Data range / payload allowed: Delta or dump (with warning)

LeaseAccelerator strongly recommends extracting only assets with corrected attributes from the source system so the payload sent to LeaseAccelerator includes only the updated assets.

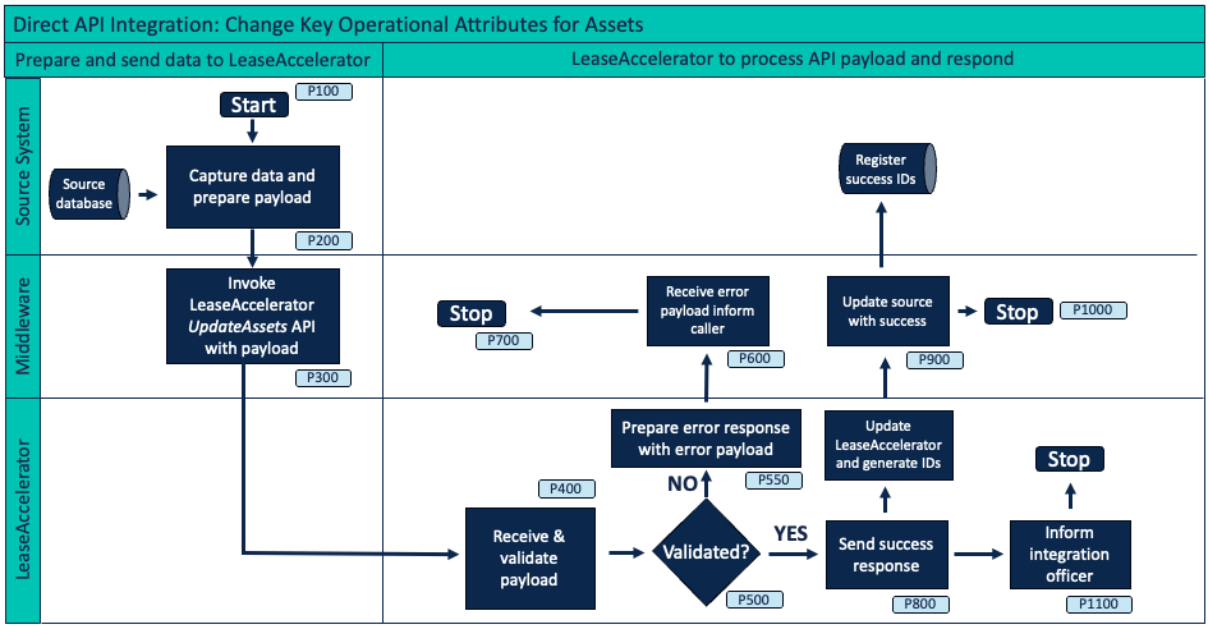
However, if the payload includes assets with non-changing attributes, LeaseAccelerator will ignore them but will trigger the accounting engine to unnecessarily re-run the calculations, affecting performance and reporting availability.

3.1 Change key operational attributes for assets using SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture asset attributes for assets and prepare upload file	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Record new attributes and generate IDs	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

3.1 Change key operational attributes for assets using API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture data and prepare payload	Source system user
P300	Invoke LeaseAccelerator **UpdateAssets** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

3.2 Data fields, format, and structure

Method	Format	Procedure/ Usage
API	XML	Please refer to API user guide for invoke process, request, and response parameters.
SFTP	XML	Same structure as API / XML payload. File name: UpdateAssets +YYYYMMDDHHMM.xml

Method	Format	Procedure/ Usage
SFTP	XLSX	Workbook name: UpdateAssets +YYYYMMDDHHMM.xlsx. Sheet name: Asset Report. First row for column names must match the field names listed below.
SFTP	CSV	File name: UpdateAssets +YYYYMMDDHHMM.csv The first row of csv files must include the field names as headers separated by commas.
Note: *The data fields available via the XLSX and CSV formats can be found in the Asset Management Update (AMU) file in Ask Alex on the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.*		

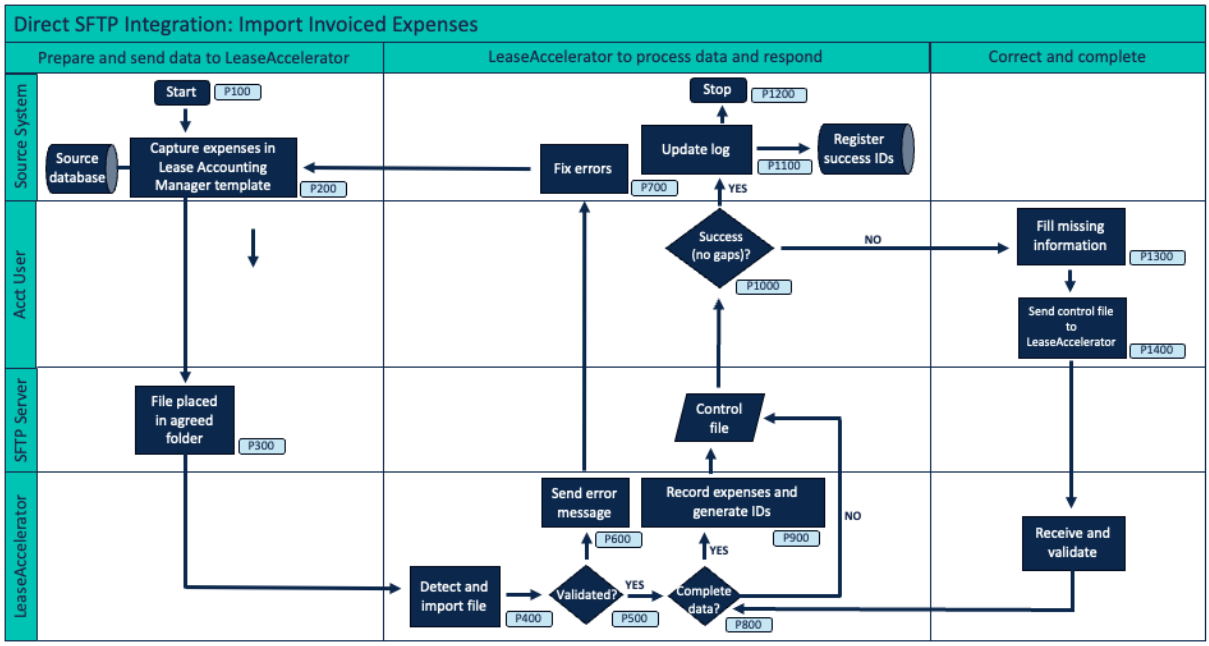
4. Import Invoiced Expenses

An invoice has been received for a Common Area Maintenance (CAMS) charge or other related expense. The invoice has been recorded in the external system but needs to be communicated to LeaseAccelerator. Accounting entries are based on the specific expense type; for items such as CAMS, the journal entries created are expense accruals.

Data range / payload allowed: Delta ONLY

Data should only include unprocessed invoices expenses. Resending previously processed invoices will cause errors.

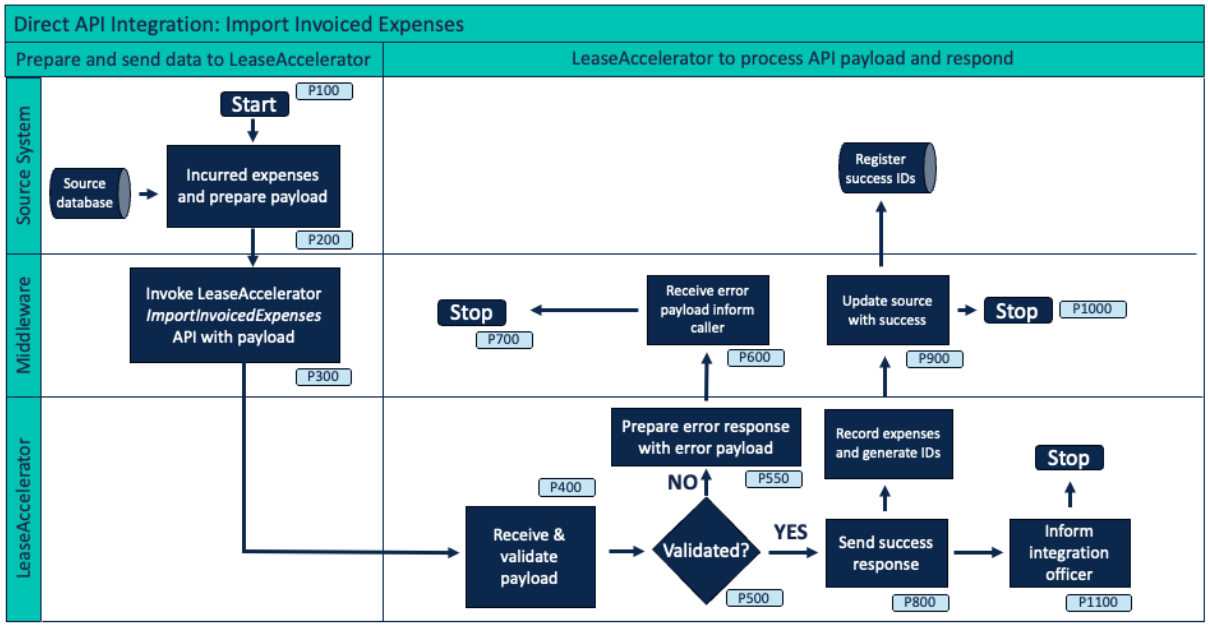
4.1 Import invoiced expenses using SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user

Process Step ID	Process Step Description	Phase
P200	Capture expenses in LeaseAccelerator template	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Record expenses and generate IDs	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

4.2 Import invoiced expenses using API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture data and prepare payload	Source system user

Process Step ID	Process Step Description	Phase
P300	Invoke LeaseAccelerator **ImportInvoicedExpenses** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

4.3 Data fields, format, and structure

Method	Format	Procedure/ Usage
API	XML	Please refer to API user guide for invoke process, request, and response parameters.
SFTP	XML	Same structure as API / XML payload. File name: ImportInvoicedExpenses +YYYYMMDDHHMM.xml
SFTP	XLSX	Workbook name: ImportInvoicedExpenses +YYYYMMDDHHMM.xlsx. Sheet name: Invoice Related Expenses. First row for column names must match the field names listed below.
SFTP	CSV	File name: ImportInvoicedExpenses +YYYYMMDDHHMM.csv. The first row of csv files must include the field names as headers separated by commas.
Note: *The data fields available via the XLSX and CSV formats can be found on the Invoiced Related Expenses tab of the Portfolio Intake Workbook(PIW) in Ask Alex on the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.*		

Invoice related expenses

This tab enables a client to import the actual invoiced amounts for Other Related Expenses. Based on the lease type, these invoiced amounts may generate accounting entries to record expense adjustments. Accounting entries for this tab are based on the specific expense type; for items such as CAMS, the journal entries created are expense accruals.

Please note that the accounting treatment for ARO and End-of-Life is as follows: When Expense Type is ARO-End-of-Life, please note that for SubTypes of Equipment-Refurbishment and Real Estate - Return to Original Condition, LeaseAccelerator will generate Asset Retirement Obligations (ARO) accounting. For SubType of Cost to Dismantle or Remove (per Agreement), LeaseAccelerator will generate End-of-Life accounting which is added to the payment schedule.

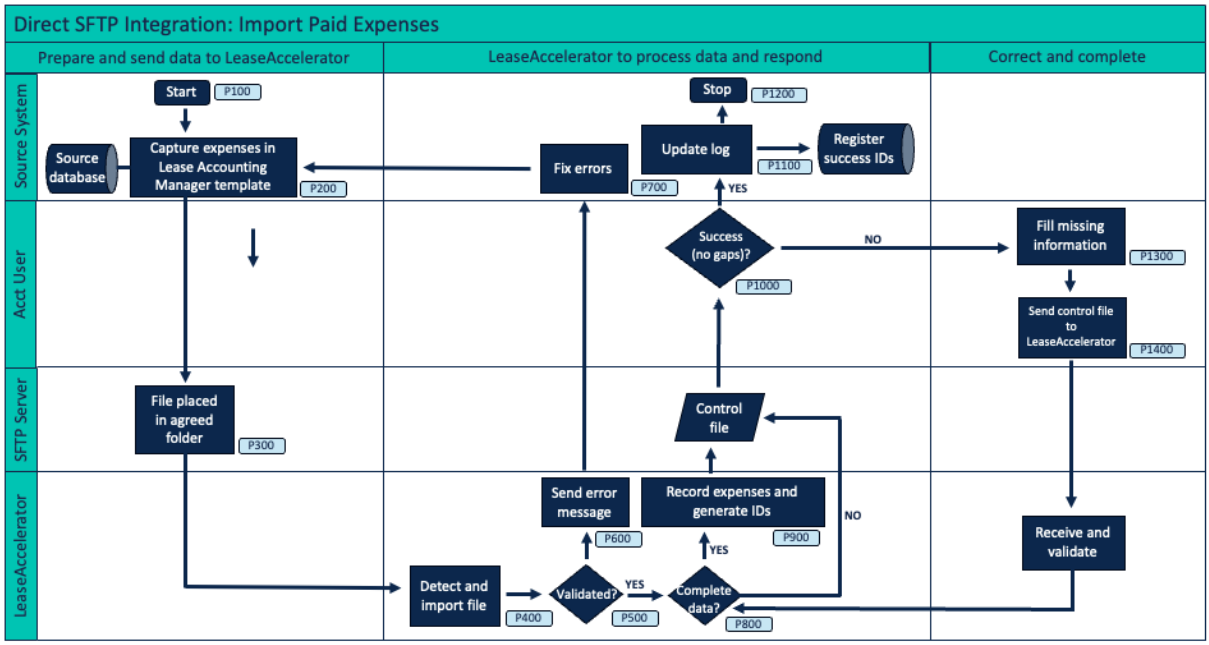
5 Import Paid Expenses

An invoice has been paid for CAMS charge or other related expense. The payment has been recorded in the external system but needs to be communicated to LeaseAccelerator. The journal entries will reverse the expense accruals created from the Invoiced Related Expense tab (if matched based on Invoice Number and Invoice Date) and record the expense recognition based on the amount approved for payment.

Data range / payload allowed: Delta ONLY

Data should only include unprocessed invoices expenses. Resending previously processed invoices will cause errors.

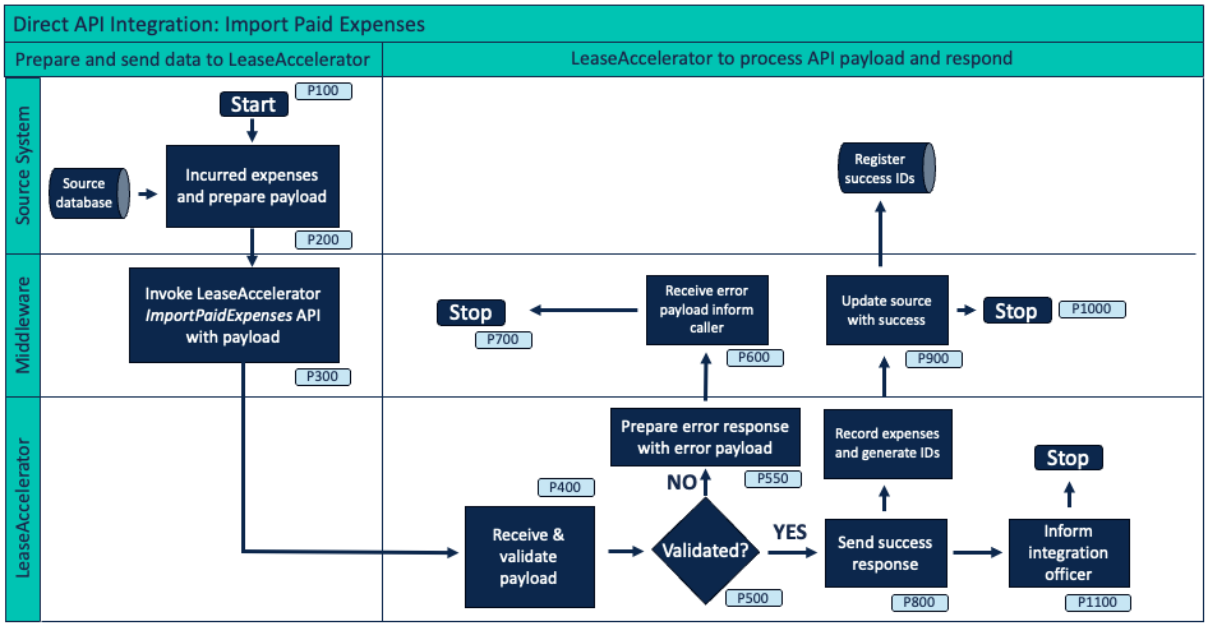
5.1 Import paid expenses using SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture expenses in LeaseAccelerator template	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Record expenses and generate IDs	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system

Process Step ID	Process Step Description	Phase
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

5.2 Import paid expenses using API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture data and prepare payload	Source system user
P300	Invoke LeaseAccelerator **ImportPaidExpenses** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

5.3 Data fields, format, and structure

Method	Format	Procedure/Usage
API	XML	Please refer to API user guide for invoke process, Request and response parameters.
SFTP	XML	Same structure as API / XML payload. File name: ImportPaidExpenses +YYYYMMDDHHMM.xml
SFTP	XLSX	Workbook name: ImportPaidExpenses +YYYYMMDDHHMM.xlsx. Sheet name: Paid Related Expenses. First row for column names must match the field names listed below.
SFTP	CSV	File name: ImportPaidExpenses +YYYYMMDDHHMM.csv The first row of csv files must include the field names as headers separated by commas.
Note: *The data fields available via the XLSX and CSV formats can be found on the Paid Related Expenses tab of the Portfolio Intake Workbook (PIW) in Ask Alex in the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.*		

This tab enables a client to import the actual paid amounts for Other Related Expenses, which are approved for payment and sent to Accounts Payable. Based on the lease type, these invoiced amounts may generate accounting entries to record expense adjustments. Accounting entries for this tab are based on the specific expense type; for items such as CAMS, the journal entries will reverse the expense accruals created from the Invoiced Related Expense tab (if matched based on Invoice Number and Invoice Date) and record the expense recognition based on the amount approved for payment. Please note that the accounting treatment for ARO and End-of-Life is as follows: When Expense Type is ARO-End-of-Life, please note that for SubTypes of Equipment-Refurbishment and Real Estate - Return to Original Condition, LeaseAccelerator will generate Asset Retirement Obligations (ARO) accounting. For SubType of Cost to Dismantle or Remove (per Agreement), LeaseAccelerator will generate End-of-Life accounting which is added to the Payment Schedule.

6 Adjust Payment Amount

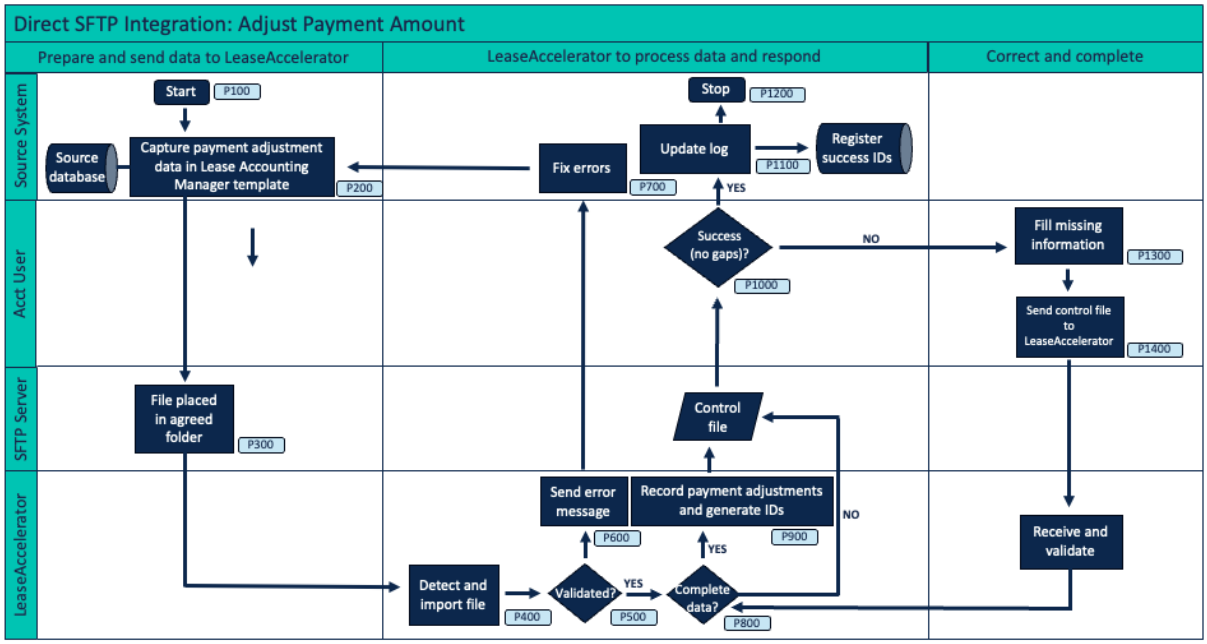
The payment amount for a schedule has changed, either because of a routine adjustment due to fluctuations in the underlying index rate, resolution of a contingency, or a scheduled step payment (e.g. automatic escalation clause). This change in payment needs to be communicated to LeaseAccelerator.

Data range / payload allowed: Delta or Dump (with warning)

LeaseAccelerator strongly recommends extracting only new payment adjustments from the source system so the payload sent to LeaseAccelerator includes only the unprocessed payment adjustments.

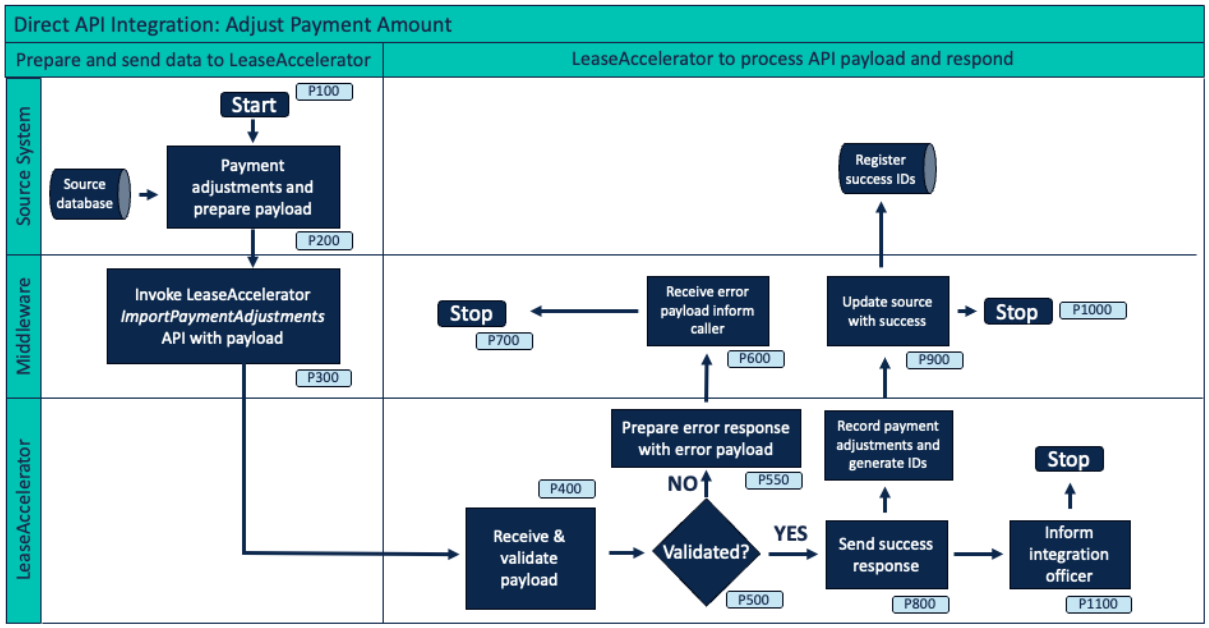
However, if the payload includes assets previously processed through payment adjustments, LeaseAccelerator will ignore them but will trigger the accounting engine to unnecessarily re-run the calculations, affecting performance and reporting availability.

6.1 Adjust payment amount using SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture payment adjustment data in LeaseAccelerator template	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Record payment adjustments and generate IDs	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

6.2 Adjust payment amount using API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture data and prepare payload	Source system user
P300	Invoke LeaseAccelerator **ImportPaymentAdjustments** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

6.3 Data fields, format, and structure

Method	Format	Procedure/ Usage
API	XML	Please refer to API user guide for invoke process, Request and response parameters.

Method	Format	Procedure/Usage
SFTP	XML	Same structure as API / XML payload. File name: ImportPaymentAdjustments +YYYYMMDDHHMM.xml
SFTP	XLSX	Workbook name: ImportPaymentAdjustments +YYYYMMDDHHMM.xlsx. Sheet name: Payment Adjustments. First row for column names must match the field names listed below.
SFTP	CSV	File name: ImportPaymentAdjustments +YYYYMMDDHHMM.csv The first row of csv files must include the field names as headers separated by commas.
Note: *The data fields available via the XLSX and CSV formats can be found on the Payment Adjustments tab of the Portfolio Intake Workbook (PIW) in Ask Alex of the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.*		

7 Swap Like-for-Like Assets

One or more assets have been exchanged for assets of comparable value on a like-for-like basis, and the updated asset information needs to be communicated to LeaseAccelerator.

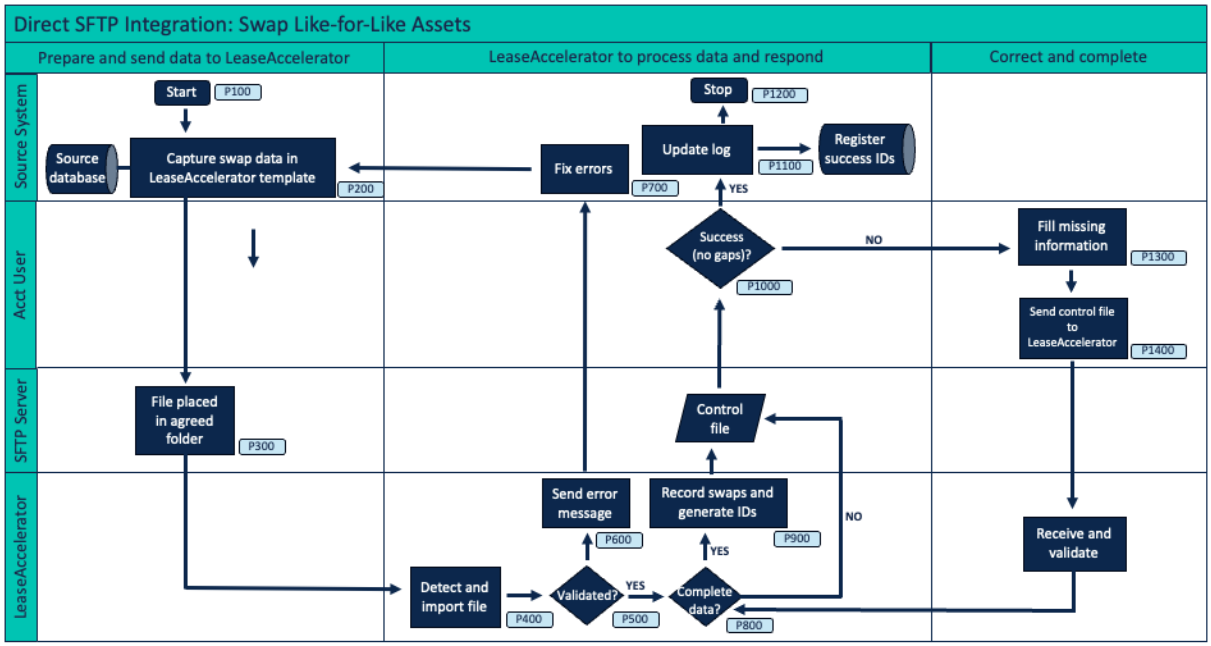
Note: If the schedule has associated journal entries that have been closed in LeaseAccelerator, and the value or calculated rent for the updated assets do not match the value and calculated rent of the assets prior to update, then the update will be rejected, and any changes will need to be communicated to the responsible accounting staff for manual entry into LeaseAccelerator.

Data range / payload allowed: Delta or dump (with warning)

LeaseAccelerator strongly recommends extracting only swapped assets from the source system so the payload sent to LeaseAccelerator includes only the swapped assets.

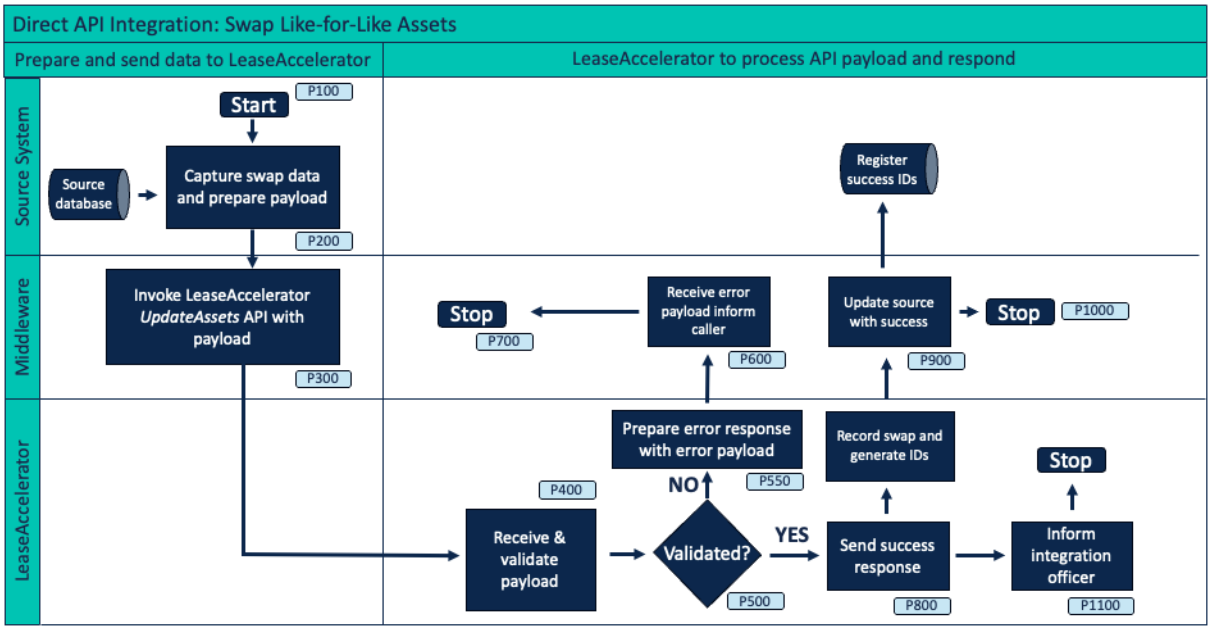
However, if the payload includes previously processed swapped assets, LeaseAccelerator will ignore them but will trigger the accounting engine to unnecessarily re-run the calculations, affecting performance and reporting availability.

7.1 Swap like-for-like assets – Using SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture swap data in LeaseAccelerator template	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Record swap and generate IDs	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

7.2 Swap like-for-like assets – using API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture swap data and prepare payload	Source system user
P300	Invoke LeaseAccelerator **UpdateAssets** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

7.3 Data fields, format, and structure

Method	Format	Procedure/Usage
API	XML	Please refer to API user guide for invoke process, Request and response parameters.
SFTP	XML	Same structure as API / XML payload. File name: UpdateAssets +YYYYMMDDHHMM.xml

Method	Format	Procedure/ Usage
SFTP	XLSX	Workbook name: UpdateAssets +YYYYMMDDHHMM.xlsx. Sheet name: Asset Report. First row for column names must match the field names listed below.
SFTP	CSV	File name: UpdateAssets +YYYYMMDDHHMM.csv????? The first row of csv files must include the field names as headers separated by commas.
Note: *The data fields available via the XLSX and CSV formats can be found in the Asset Management Update (AMU)file in Ask Alex on the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.*		

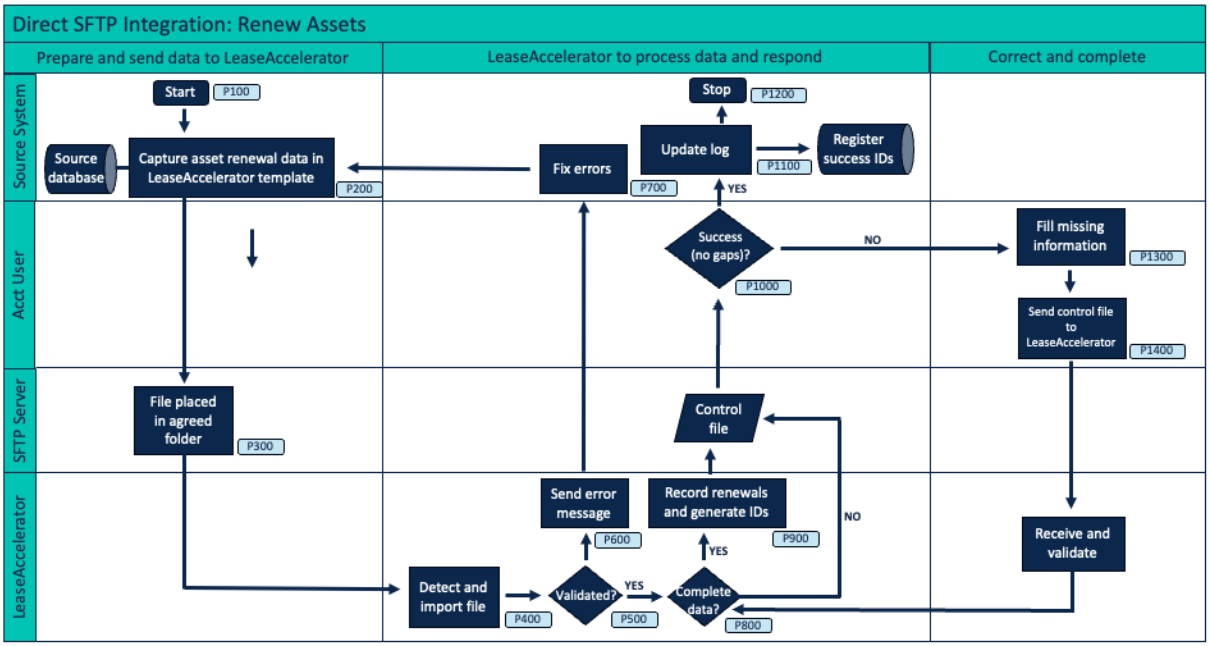
8 Renew assets

The lease obligation for one or more assets has been extended, either for a fixed term, or indefinitely (evergreen/month-to-month). This extension has been captured in the external system but needs to be communicated to LeaseAccelerator.

Data range / payload allowed: Delta ONLY

Data should include unprocessed renewed assets only. Resending previously processed renewed assets will cause errors.

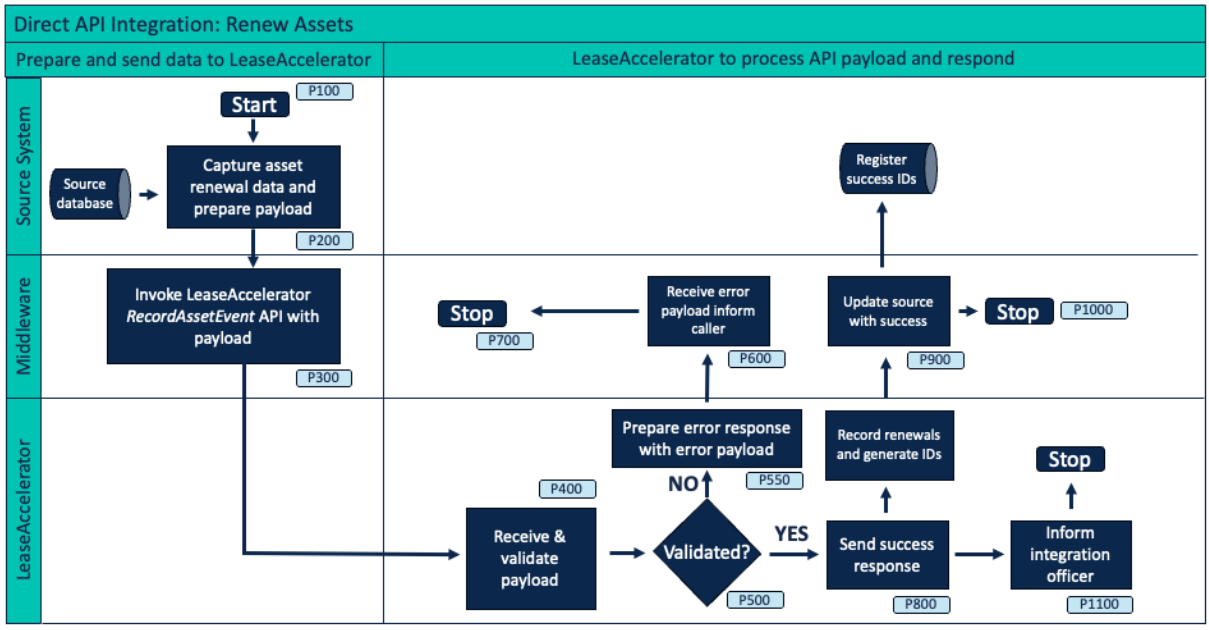
8.1 Renew assets – using SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user

Process Step ID	Process Step Description	Phase
P200	Capture asset renewal data in LeaseAccelerator template	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Record renewals and generate IDs	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

8.2 Renew assets – using API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture renewal data and prepare payload	Source system user
P300	Invoke LeaseAccelerator **RecordAssetEvent** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator

Process Step ID	Process Step Description	Phase
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

8.3 Data fields, format, and structure

Method	Format	Procedure/ Usage
API	XML	Please refer to API user guide for invoke process, Request and response parameters.
SFTP	XML	Same structure as API / XML payload. File name: RecordAssetEvent +YYYYMMDDHHMM.xml
SFTP	XLSX	Workbook name: RecordAssetEvent +YYYYMMDDHHMM.xlsx. Sheet name: Advanced Form. First row for column names must match the field names listed below.
SFTP	CSV	File name: RecordAssetEvent +YYYYMMDDHHMM.csv The first row of csv files must include the field names as headers separated by commas.
Note: *The data fields available via the XLSX and CSV formats can be found in the Bulk Import Record Asset Event template in Ask Alex on the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.*		

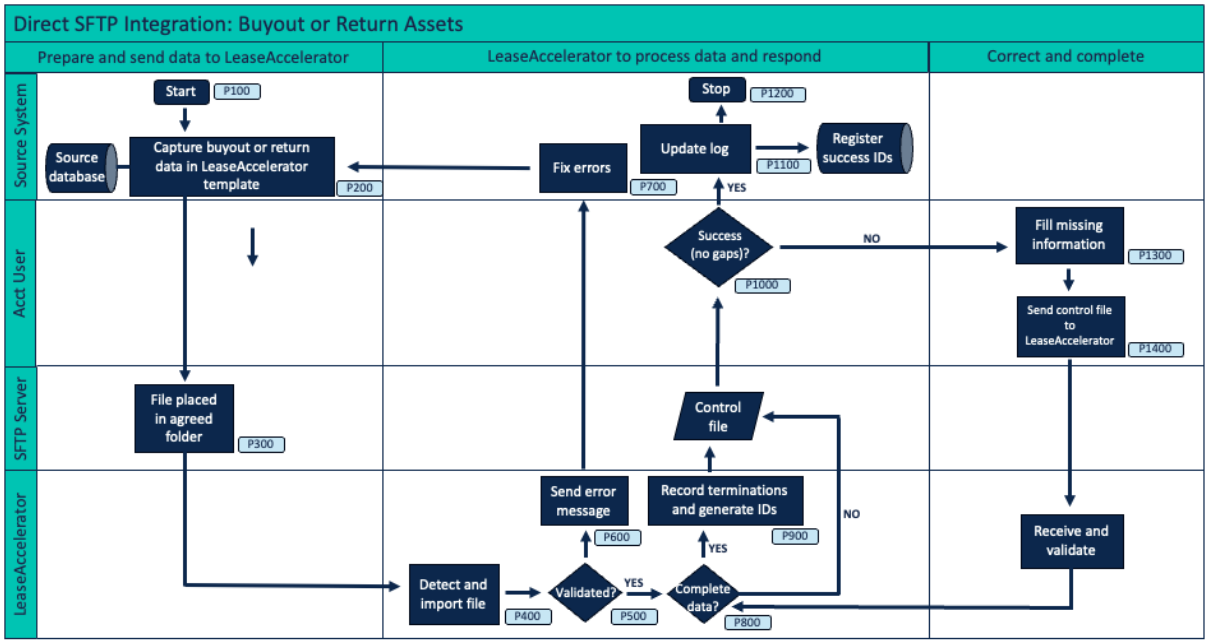
9 Buyout or Return Assets

One or more assets have been returned or bought out, thus ending the lease obligation for those assets. This termination has been captured in the external system but needs to be communicated to LeaseAccelerator.

Data range / payload allowed: Delta ONLY

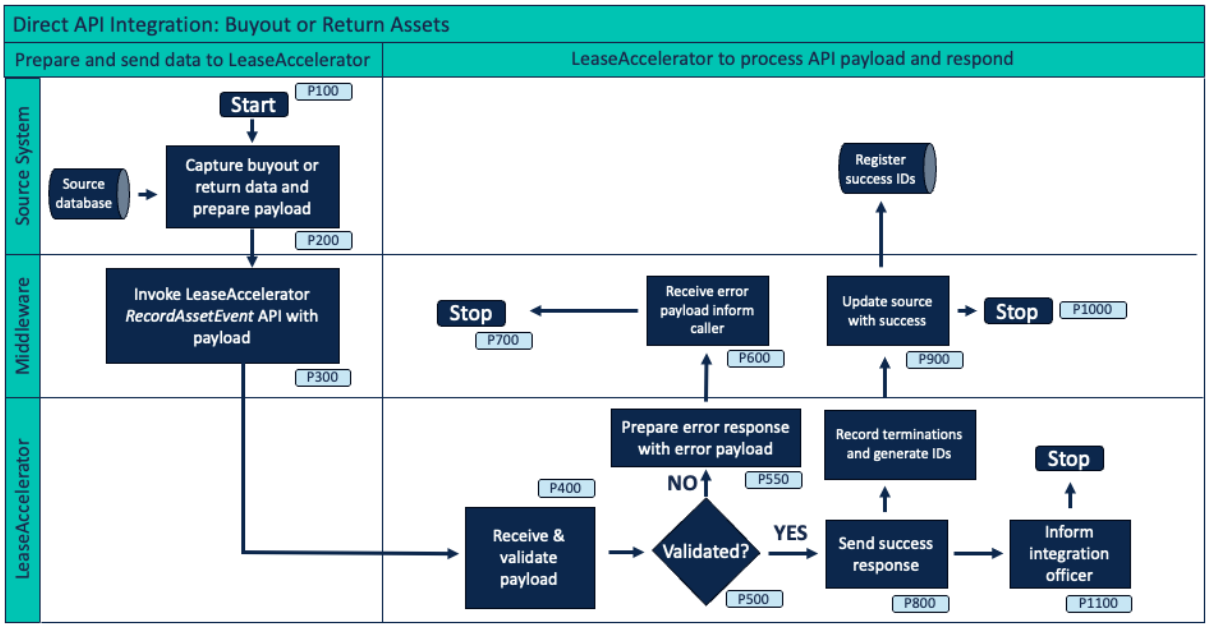
Data should include unprocessed assets only. Resending previously processed assets will cause errors.

9.1 Buyout or return assets – using SFTP



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture asset buyout or return data in LeaseAccelerator template	Source system user
P300	File placed in agree folder	SFTP server
P400	Detect and import file	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P600	Send error message	LeaseAccelerator
P700	Fix errors	Source system
P800	Complete data?	LeaseAccelerator
P900	Record terminations and generate IDs	LeaseAccelerator
P1000	Success with no gaps?	Accounting user
P1100	Update log	Source system
P1200	Stop	Source system
P1300	Fill missing information	Accounting user
P1400	Send control file to LeaseAccelerator	Accounting user
P1400	Receive and validate	LeaseAccelerator

9.2 Buyout or return assets – using API



Process Step ID	Process Step Description	Phase
P100	Start	Source system user
P200	Capture buyout or return data and prepare payload	Source system user
P300	Invoke LeaseAccelerator **RecordAssetEvent** API with payload	Middleware
P400	Receive and validate payload	LeaseAccelerator
P500	Valid?	LeaseAccelerator
P550	Prepare error response with error payload	LeaseAccelerator
P600	Receive error payload and inform caller	Middleware
P700	Stop	Middleware
P800	Send success response	LeaseAccelerator
P900	Update source with success	Middleware
P1000	Stop	Middleware
P1100	Inform user	LeaseAccelerator

9.3 Data fields, format, and structure

Method	Format	Procedure/Usage
API	XML	Please refer to API user guide for invoke process, request, and response parameters.

Method	Format	Procedure/Usage
SFTP	XML	Same structure as API / XML payload. File name: RecordAssetEvent +YYYYMMDDHHMM.xml
SFTP	XLSX	Workbook name: RecordAssetEvent +YYYYMMDDHHMM.xlsx. Sheet name: Advanced Form. First row for column names must match the field names listed below.
SFTP	CSV	File name: RecordAssetEvent +YYYYMMDDHHMM.csv. The first row of csv files must include the field names as headers separated by commas.
The data fields available via the XLSX and CSV formats can be found in the Bulk Import Record Asset Event template in Ask Alex on the LeaseAccelerator UI. Fields for XML are very similar and are discussed in detail in the API Developer's Guide.		