Sage Intacct Connect Onboarding Package for Rillion Prime

Contents

Change Log

Version	Date	Changed by	Changes
1.0	2025/08/26	SA (Sarmad Abbas)	First draft created
1.1	2025/09/04	SA	Adding CW General Settings
1.2	2025/09/04	SA	Adding Payment Scenarios in Tips and Tricks section
1.3	2025/09/12	MZ (Michael Ziraschi)	First full review, multiple updates
1.4	2025/09/15	SA	Adding the
1.5	2026/01/05	MZ	Post Sage marketplace approval meeting updates
1.6	2026/01/07	MZ	Added Governance section
1.7	2026/01/19	MZ	Final edit before submission to Sage Marketplace

Introduction

This document describes in detail the Rillion Sage Connect integration in the areas listed below:

- Integration description
- Architecture
- Functional Description
- Integration Requirements and Assumptions
- Onboarding Process Checklist
- Onboarding Customer Questionnaire
- Installation Guide
- Tips and Tricks
- Governance

This document is meant for review by customer team resources including their f resources as well as internal Rillion resources. This document contains information for various stakeholders. It is the obligation of stakeholders to ensure they have reviewed the document for areas related to them and the project as whole.

Integration Description

Rillion uses its own proprietary integration module named Connect to integrate with various ERP platforms. The Connect module is designed to run behind the scenes as a service and execute on a scheduled basis using standard Rillion Prime functionality, not requiring any manual involvement by customers.

This Onboarding Guide is only for the Sage Intacct integration against Rillion Prime.

This integration is developed to support cloud Sage Intacct installations.

The integration app is built to Sage Intacct standards, REST API, methodology, and coding language, and can quickly be deployed by the customer themselves using this document if the Sage solution is an on-premise. The integration app is available in the Sage Intacct marketplace and has been approved and certified by Sage themselves.

The integration supports standard data imports and invoice creation based on invoices processed in Rillion as described in this document. At present, the integration does not include Purchase Order match in Phase 1. To see more information on the specification of what is supported as standard, see the Integration Specification section of this document.

The Sage Connect integration functions via our Connect platform calling the Sage API, not the other way around. It is a passive integration, Connect calls Sage, Sage does not call Connect.

Architecture

The figure below represents the data flow and components in play for this Sage/Rillion Prime integration.

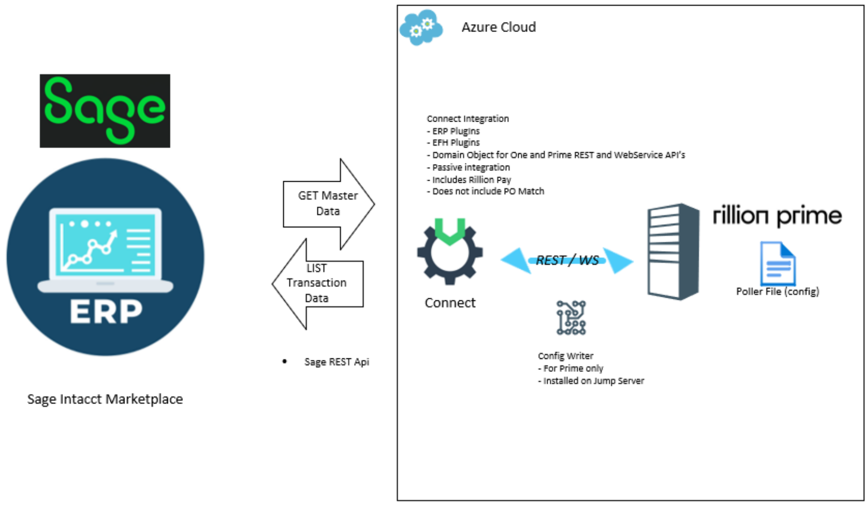
The Rillion parameters for this integration are set in Rillion Config Writer (CW), which is deployed on the Rillion PO Jump server. This configuration is performed by Rillion as customers do not have access to CW.

The parameter for pointing a customer's Rillion solution towards a Connect instance is determined in the Poller file configuration for each customer. More details on these Poller files can be found further down in this document in the Installation Guide. This file must be saved as per the instructions in this guide on the same customer server where Connect is deployed.

Connect for Sage is not updated automatically upon the release of a new version via the automatic update feature if the solution is an on-premise integration. Otherwise, if you are using the cloud solution, it will automatically update with the latest features in the new release.

If you have a Sage instance, you can generate the ClientID and Secret for the testing environment and test it prior to going to the live environment.

Connect calls and use the standard Sage REST Api.



Integration Functional Description

In the list below, the functions and features that are included in the latest version of the Sage Connect integration are indicated:

NOTE: All registers are company-bound.

Function/Feature	Included	Not Included	Company bound	Comment
Sage development partner	X			A Rillion Sage partner was used in the development of this integration.
Sage On-Prem solution	X			The integration can be used against an On-Prem Sage
Sage Cloud solution	X			The integration can be used against a Cloud Sage
Sage Marketplace	X			The integration is listed on the Sage marketplace
INBOUND DATA				
Registers/Basic Data/Master Data imported from Sage				
Company (Subsidiary)		X		Companies need to be manually added in Rillion
Calendar		X		

Accounts (GL Accounts)	X		X	Does not include a default Dimension values
Vendors (Suppliers)	X		X	Includes supplier bank accounts and days until payment (payment terms)
Currency	X		X	
Exchange rates	X		X	
VAT codes	X		X	
Tax code rates (%'s)	X		X	
Accounting periods		X	X	Accounting periods need to be setup manually in Rillion
Assets	X		X	
Asset type	X		X	
Dimensions/Objects	X		X	Does not include VAT %, includes only Dimensions. There are more than 10 Dimensions in SAGE, note that Rillion Prime has a maximum of 8 dimensions (Objects)
Amortization templates		X		Support for the Sage Prepaid Amortization Module will be in a future release
Posting line validations		X		Includes dimension relations on Accounts
Purchase Orders from Sage				
Purchase Orders (PO)		X		Support for import of PO and deliveries and PO matching process will be in future release
Purchase Order Receipts (PO Deliveries)		X		Support for import of PO and deliveries and PO matching process will be in future release
General				
Voucher number from Sage	X			Return of newly created invoice ID from Sage to Rillion Prime
Booking date	X			We are setting up the Creation, posting, due, and recommendation payment dates.
Payment return data (date)	X			Supported for Invoices
OUTBOUND DATA				
Invoices (Bills)				
Pre-booking (preliminary booking)		X		Sage Intacct does not have a preliminary booking process so this is not implemented
Pre-booking cancellations		X		Sage Intacct does not have a preliminary booking process so this is not implemented
Final invoice posting	X			

Final invoice posting w/ Periodization		X		Support for the Sage Prepaid Amortization Module will be in a future release
Invoice image link (URL)	X			Send a URL to the Rillion Prime image to Sage available in the comment field in Sage on the invoice
Image transfer (PDF)	X			
Payment Reference/OCR string	X			Bank payment strings from the invoice sent to invoice generated in Sage. (feks: Bank/Plus giro OCR strings)
Payment Block	X			
Posting line description text	X			
Rillion Pay settled payment	X			If applicable and the customer has the Rillion Pay service, payments settled by Rillion Pay will be exported to Sage
General				
Failure/Error message return	X			Error message return from Sage on failures or errors for final booking and Rillion Pay settled payments

Integration Requirements and Assumptions

The integration including the deployment is based on a standardized template which in turn is based on a set of requirements and assumptions. These requirements and assumptions are conditions for the onboarding process to stay within the standard offering and process.

Requirements:

1. All register data (master data we import from Sage) is company-bound by default
2. The customer themselves deploys Rillion Connect and Config Writer on the same customer server for On-premise Sage solution. If it is Cloud, then they do not deploy these modules and they are then responsible for providing Rillion with a valid ClientID and Secret for the Sage API
3. The customer needs an integration license in Sage Administration to connect the companies in Sage and Rillion Prime
4. Customer must subscribe to web services in their Sage solution and set up a web service user
5. Sage Instance should support the REST API development module
6. The customer must use Rillion Prime version 8 or later

Assumptions/Conditions:

1. The integration only supports the functionality and features as specified in this document
2. Functionality and features not specified as supported in this document are not supported
3. The customer TEST company contains a complete and accurate reflection of production quality data from the start of the project delivery

4. The customer has completed the questionnaire portion of this document and submitted it to Rillion prior to the start of deployment
5. The standard integration delivery is for one single company and country, additional companies and countries can be added however there can come time and materials charges for the additional time used to configure.
6. The customer has reviewed and fully understands all the information contained in this document prior to signing any agreement with Rillion.
7. If the customer server cannot be reached by Rillion GitHub, then the automated update function of Rillion Connect will not function, it is fully dependent on open communication between Rillion GitHub and Connect

Onboarding process checklist

Follow and signoff on the following checklist which clarifies the steps in the onboarding process in sequence. The Ownership of these tasks is split between multiple project stakeholders. Ultimately in the end, it is the responsibility of Rillion to ensure that the steps in the checklist are followed and indicated as complete.

Step #	Description	Ownership	Name	Initialled Complete	Comment
1	Opportunity is added in CRM with ERP system and solution provider information	Sales/Presales			
2	Send opportunity a copy of the complete onboarding document	Sales/Presales			This onboarding document must be sent to the prospect for them to review together with their Sage Expert resources prior to signing of any deal. Always send the latest version obtained from RCP or FreshDesk.
3	Opportunity confirms they have received and reviewed the onboarding document	Sales/Opportunity			Written confirmation from the customer they have reviewed and understood the onboarding document and have a chance to ask questions.
4	Produce an offer for delivery approved by Professional Services	Sales/Professional Services			
5	Opportunity completes the questionnaire	Opportunity (customer)			Accurate and complete answers, see Assumptions and Conditions.
6	Completed and signed questionnaire received from the opportunity	Sales			This step must be prior to any final agreement signature.

					The solution will be set up as per the Questionnaire. If the Sage environment actually differs from the completed questionnaire provided, then the project is outside the standard delivery and becomes an Agile type project delivery under time and materials.
7	Agreement is signed	Sales			
8	Opportunity is changed to a customer in Sage and the agreement is archived (in Rillion Prime Contracts)	Sales			
9	Professional services orders the Rillion Prime Online environment from Operations	Professional Services			The order for the online environment in FreshService ticket
10	Professional services orders the access to the webservice and Rillion environment credentials				
11	Professional creates and orders the Poller File for Connect for the customer	Professional Services/Online Operations			See the deployment step below
12	If the customer has an On-premise Sage, Professional services creates Fresh Desk ticket for 4th line to create the RCP tenant admin and provides the details as specified below in this document. ***Note:*** *Step 12-13 can be skipped if the customer has a cloud Sage solution*	Professional Services/4th Line Connect			Includes customer tenant being added to RCP and so that the customer can receive RCP credentials to download the Sage, Rillion Connect incl. Config Writer installation .zip packages for them to install on their own server.
13	If the customer has an On-premise Sage, Customer received the RCP credentials by email, downloads and deploys the downloaded integration packages as per this onboarding document.	Customer			

14	Professional services configures the integration as per this onboarding guide.	Professional Services			
15	Customer Training	Professional Services			
16	UAT	Customer/Professional Services			
17	UAT Signoff and agree go live date	Customer/Professional Services			
18	Deploy the Connect package and integration configuration in PRODUCTION	Professional Services			Note: If the customer is using their Sage for a sandbox AND production, there will be different ClientId and Secret Key associated with each of those.
19	Go live	Customer/Professional Services			
20	Handover to support	Professional Services			
21	Send customer the welcome letter, guide and credentials for access to FreshDesk Online Support	Support			
22	Onboarding document is continuously updated and available in MS Teams and FreshDesk	Integration Team	Product Owner – Michael Ziraschi	n/a	

Onboarding SAGE integration questionnaire

The questions in this section require complete and accurate answers from the customer. The purpose of providing these questions and gathering the answers is to ensure two things:

- That the customer scope and the most current available functionality and features supported by the Rillion Sage Connect integration are aligned before any final agreement is signed
- The minimum information required by the Professional Services/Delivery project team is provided by the customer before the project delivery begins so that the standard delivery package can be deployed according to the standard best practices and according to the timeline included in the standard delivery package.

The answers also ensure that there is no “moving target” scenario when deployment begins of and avoids scope creep. It is therefore of vital importance that the customer answer the questionnaire accurately to avoid scope creep of the delivery outside of the standardized delivery package and timeline.

The customer can add there answers directly below and add any comments they feel are required for consideration in the Comment field.

#	Item/Question	Answer	Comment
General information			
1	Customer Name		Complete legal entity name.
2	Customer technical resource name, email, telephone number		The name and contact information for the customers prime technical contact for all matters concerning IT and Sage. If there are two contacts, then provide both assuming we are allowed to contact them directly.
3	Customer technical resource location/physical address		We need this information to determine the timezone of this contact and any impact it may have on the project delivery (European vs. US time zone)
Company/Organization			
5	Number of companies (subsidiaries) to be included in this Rillion Prime solution		
6	Please List the companies (subsidiaries) by complete name that will be included in the Rillion Prime solution AND the Country for each company	• Xx - Country • Xx - Country • Xx - Country • Xx - County • Xx - Country	The integration is released for North American, Western and Scandinavian European regions. We do not guarantee the integration for regions/countries outside of these.
Sage environment			
7	Does your Sage TEST company accurately reflect data for all the companies included in this solution. (Y/N)		It is required that the data in the TEST company is up to date and an accurate reflection of real production data to ensure UAT testing is accurate and meaningful and is required to stay within the standard delivery package.

8	The customer has a Sage integration license (Y/N)		This is a requirement to allow the integration to establish communication between Sage and Rillion.
9	The customer has setup a Connect integration user in their Sage environment – provide details		Note: This is a webservice user, see steps below. This is not an actual user in Sage Intacct itself.
10	Do you have the Sage Prepaid Amortization Module installed (Y/N)		Support for the Sage Prepaid Amortization Module will be in a future release
11	Do you use Purchase Orders and Delivery Receipts in your Sage Environment, if so, for all companies? (Y/N)		Support for the PO and Delivery imports and matching process be in a future release
12	Will you use the Rillion Pay service in your Rillion solution (Y/N)		
13	Are the registers like Suppliers, Tax codes, Currencies set up as TOP LEVEL or ENTITY LEVEL in Sage? If it varies between registers, please provide details, if they are all set up the same please indicate the setup.		
Classifications/Dimensions/Objects			
14	Confirm the Sage dimensions used in this solution are the same as indicated in the next column and confirm they are on the posting lines Also confirm that these dimensions are used by all companies in this solution. If this is not the case, provide details on which dimensions are used by which companies. Furthermore, since Prime only support 8 dimension for now so we need to discuss which are required for the customer. Also Department and location must be in the Dim list they are mandatory in the Sage.	1. DEPARTMENT* 2. LOCATION* 3. PROJECT 4. CUSTOMER 5. VENDOR 6. EMPLOYEE 7. ITEM 8. CLASS 9. CONTRACT 10. TASK 11. WAREHOUSE 12. COSTTYPE 13. AFFILIATEENTITY 14. FIXEDASSET	Rillion supports a maximum of 8 dimensions (objects), so the customer needs to choose which 8 they want to support in Prime. * Location and Department are mandatory and must ALWAYS be used in each integration. So by default, there are only 6 more dimensions that can be used with Rillion Prime.
15	For each of the dimensions you use, are they set up as TOP LEVEL or ENTITY LEVEL dimensions in Sage? Please provide details for each if they vary or one answer if all are set up the same.		

Integration Specification

SAGE - Number of API calls for inbound and outbound

Entity Name	Nr of calls	Description
Authentication	1+n	First call to obtain the authentication token, and then subsequent calls when the token expires. Usually, the token expires after 36 hours.
Accounts	1 + (Dim Configured max 8-10)	1 for the account Others depend on dimensions configured
Supplier	1+1+1	1 supplier 1 payment terms 1 supplier bank accounts
Currency	1+1	1 for Currency 1 ExRates
VatCodes	1+1	1 VATs 1 VATRates
Assets	1+1	1 Asset 1 Asset Type
Dimension	1+n	Based on the Configurations. Max 8-10
Payment Date	1	Get Invoices/Bills from SAGE
Account Period	1	Get the Accounting period from SAGE
Def Booking	1+1+1	1 Getting Accounts 1 Creating Attachment 1 Create an Invoice/Bill
Cancellation Invoice	1+1	1 Getting Bills 1 Delete the Invoice/Bill
Create Payment	1+1	1 Getting Bills 1 Add Payment for Invoice/Bill

Register data mappings

InBound

Accounts

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
Key	Number	Account	Account
Name	Name	Name	Name
Status (active, inactive)	Active		
	ActiveTo If Active = true, then datetime Max value	ValidTo	ValidTo
TaxCode	VatCodeNumber	VatCode	VatCode
		VatCodeMandatory = true if VatCodeNumber is defined	VatCodeMandatory
Id	ExternalId	ExternalId	ExternalId
	DefaultDimObject1Number	DefaultObject1	useObjects[]
	DefaultDimObject2Number	DefaultObject2	useObjects[]
	DefaultDimObject3Number	DefaultObject3	useObjects[]
	DefaultDimObject4Number	DefaultObject4	useObjects[]
	DefaultDimObject5Number	DefaultObject5	useObjects[]
	DefaultDimObject6Number	DefaultObject6	useObjects[]
	DefaultDimObject7Number	DefaultObject7	useObjects[]
	DefaultDimObject8Number	DefaultObject8	useObjects[]

VATCodes

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
Description	Name	Name	Name
Description	Description		
Key	Number	VatCode	VatCode
TaxPercent	VatPercentage	Percentage	Percentage
GlAccountId	VatAccountNo	Account	Account
Status	Active		
Id	ErpInternalId	ExternalId	ExternalId

AssetType

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
Name	Name	Name	Name
Key	Number	AssetType	AssetType
AssetGLAccountId	Account	Account	Account
Id	ErpInternalId	ExternalId	ExternalId

Asset

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
Name	Name	Name	Name
Key	Number	Asset	Asset

AssetTypeKey	AssetType	AssetType	AssetType
Id	ErpInternalId	ExternalId	ExternalId

Currency

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
FromCurrency	BaseCurrencyCode	BaseCurrency	BaseCurrency
ToCurrency	Code	Currency	Currency
EffectiveStartDate	RateDate	FromAccountCodingDate	FromAccountCodingDate
Rate	BuyingRate	BuyingRate	BuyingRate
ExchangeRateTypeName	Description		
	Active = true		
Id	ErpInternalId	ExternalId	ExternalId

DimObject

Dimensions:

Department, Location, Project, Customer, Vendor, Employee, Item, Class, Contract, Task, Warehouse, CostType, AffiliateEntity, FixedAsset

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
Id	ErpInternalId	ExternalId	ExternalId
Key	Number	Number	Object
Name	Name	Name	Name
Status	Active		

	ObjectType = DimNr configured in CW	ObjectTypeNo	ObjectTypeNumber
Description	Description	Description	

Supplier

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
Id	ErpInternalId	ErpInternalId	ExternalId
Key	OrgNumber	OrgNumber	CorporateIdentityNumber
Name	Name	Name	Name
Key	Number	Number	
ContactDefaultMailingAddressLine1	Address1	Address1	AddressLines[0]
ContactDefaultMailingAddressLine2	Address2	Address2	AddressLines[1]
ContactDefaultMailingAddressLine3	Address3	Address3	AddressLines[2]
ContactDefaultEmail1	Email	Email	Email
$"{sageSupplier.ContactDefaultFirstName} {sageSupplier.ContactDefaultMiddleName} {sageSupplier.ContactDefaultLastName}"	ReferenceName		
ContactDefaultPhone1	PhoneNumber1	PhoneNumber1	PhoneNumbers[0]
ContactDefaultPhone2	PhoneNumber2	PhoneNumber2	PhoneNumbers[1]
IsOnHold or DoNotPay	Blocked	Blocked	Blocked
DefaultExpenseGLAccountId	CostAccountId	CostAccountId	CostAccount
TaxId	VatRegNo	VatRegNo	VatNo

$"{sageSupplier.ContactDefaultFirstName} {sageSupplier.ContactDefaultMiddleName} {sageSupplier.ContactDefaultLastName}"	Contact	Contact	ContactPerson
Currency	CurrencyCode	CurrencyCode	Currency
ContactDefaultURL1 || ContactDefaultURL2	Website	Website	WebAddress
If blocked, then DateTime.Min else null	ActiveTo	ActiveTo	ValidTo
Notes	Note	Note	Note
ContactDefaultMailingCity	City		AddressLines[4]
ContactDefaultMailingCountry	Country	Address5	AddressLines[5]
ContactDefaultMailingPostCode	PostalCode		
PaymentTerm.DueDays	DaysUntilPayment	DaysUntilPayment	
		GraceDays = true if DaysUntilPayment HasValue	
SupplierBankAccount			
VendorAccountNumber	Name	Name	Name
VendorAccountNumber	Number	BankAccount	BankAccountNo
false	Default	Default	DefaultAccount
Id	ErpInternalId	ExternalId	ExternalId

Period

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
Name	Name	Name	Name
StartDate.Year	Year	Year	Year
StartDate	StartDate	StartDate	StartDate
EndDate	EndDate	EndDate	EndDate
Status	Active	Closed	Closed
Id	ErpInternalId	ExternalId	ExternalId
StartDate.Month	PeriodNo	Period	Period
ColumnHeader1	String1	Group1	Group1
ColumnHeader2	String2	Group2	Group2
ReportingPeriodType	String3	Group3	Group3

OutBound

Invoice

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
BillNumber	VoucherNo	VoucherNo	
Vendor	ErpSupplierID	Supplier	
ReferenceNumber	PayReference	PayReference	
Description	Note	Note	
CreatedDate	DateTimeOffset.UtcNows		
PostingDate	PostingDate		
DueDate	DueDate	DueDate	
RecommendedPaymentDate	PaymentDate	PaymentDate	
PaymentPriority	“Normal”		
IsOnHold	BlockPayment	Blocked	
BaseCurrency	ArenaCompany.BaseCurrency		
TxnCurrency	Currency	Currency	
ExchangeRate	“null” SAGE handles the exchange rate in its system, provided the currency is configured differently.		
Contacts	“null”		
IsTaxInclusive	IsTaxLiable		
Attachment	All the attachments from the Prime side will send to SAGE		

Invoice Lines

Sage Field Name	Connect Domain Object Name	Rillion Prime Webservice Name	Rillion One REST Name
GlAccount	AccountNo	AccountNo	
TxnAmount	Amount	Amount	
TotalTxnAmount	Amount	Amount	
Dimensions	All the configured Dimensions	Object1 Object2 Object3 Object4 Object5 Object6 Object7 Object8	
Memo	Note	Note	

Payments – When using Rillion Pay

Sage Field Name	Connect Domain Object Name (Rillion Pay)	Rillion One REST Name
FinancialEntity	accountNumber From the configuration, see section: BankAccountMapping in CW General Settings	
PaymentDate	PaymentDate	
Action	"submit"	
State	"approved"	
DocumentNumber	PaymentReferenceId	
DocumentId	$"PR#{Prime.PaymentReferenceId}-BR#{Sage.ReferenceNumber}-VN#{Prime.VoucherNo}"	
Description	Sage invoice Description	
PaymentRequestMethod	PaymentRequestMethod From the configuration, see section: Payment Request Method in CW General Settings	
BaseCurrency	Currency (Sage invoice currency)	
TxnCurrency	Currency (Sage invoice currency)	
PaymentMethod	PaymentMethod From the configuration, see section: Payment Method Mapping in CW General Settings	
Vendor	Vendor Id (From Sage invoice)	
TxnCurrency	Amount	
Bill	Id (Sage invoice Id)	

Installation Guide

The following are the detailed steps for deployment, installation, and configuration of the Sage Connect integration for Rillion Prime.

Relative to Rillion Prime, it is important that the Rillion professional services delivery specialist does not import dimension/register data from the standard Rillion eCompany sheet into the Rillion Prime solution. The only data that should be imported into Rillion Prime should come from via the Rillion Sage Connect integration, not the eCompany sheet.

Each step and include an indication with a label as shown below as to which steps are to be performed by the customer and which are to be performed by Rillion.

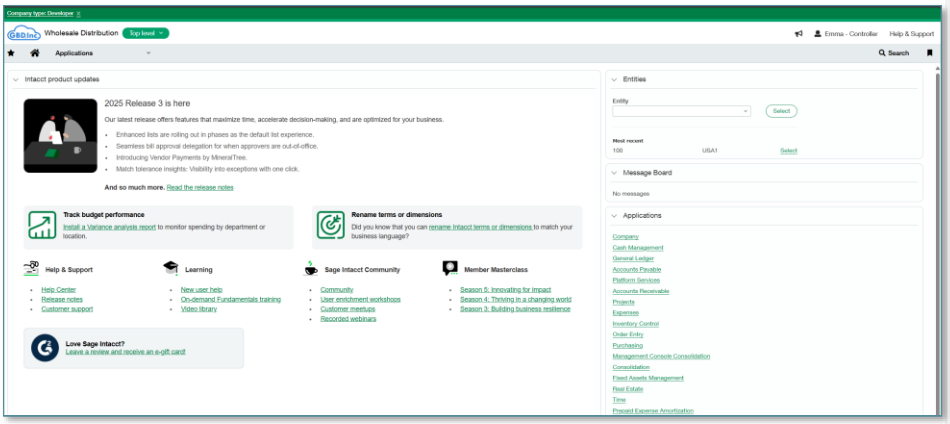
- **CUSTOMER** This can be the customer themselves or their Sage resource
- **RILLION** This will be the Rillion

Installing the Rillion integration using Cloud

Prep of Sage – CUSTOMER

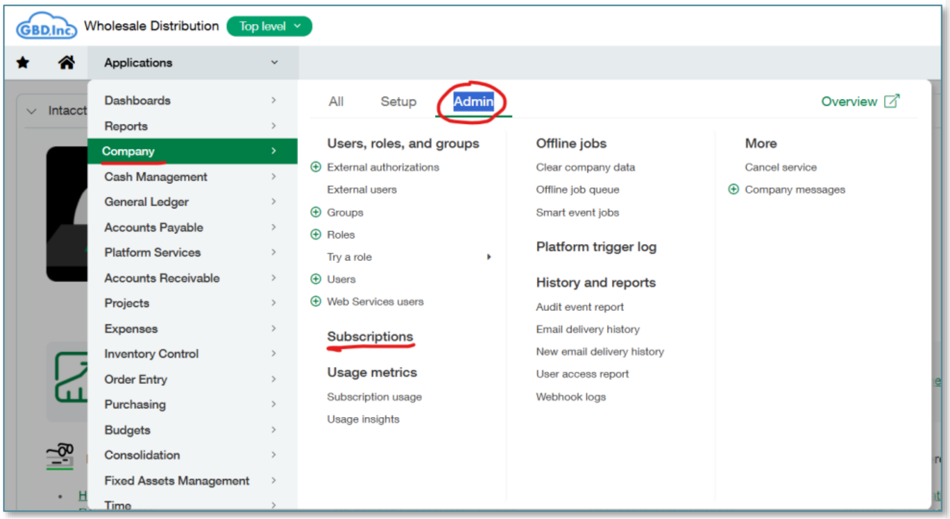
Step 1 – Login to the Sage portal as Admin User

Log in to the Sage portal with Admin User

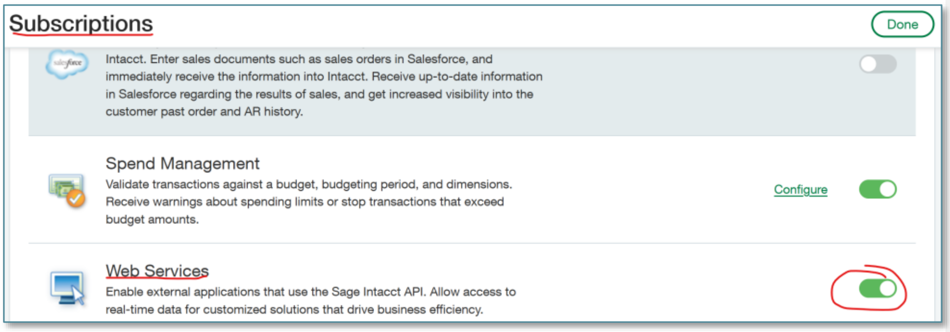


Step 2 – Checking the Subscription

From the main menu, click on Company → Admin → Subscription page.

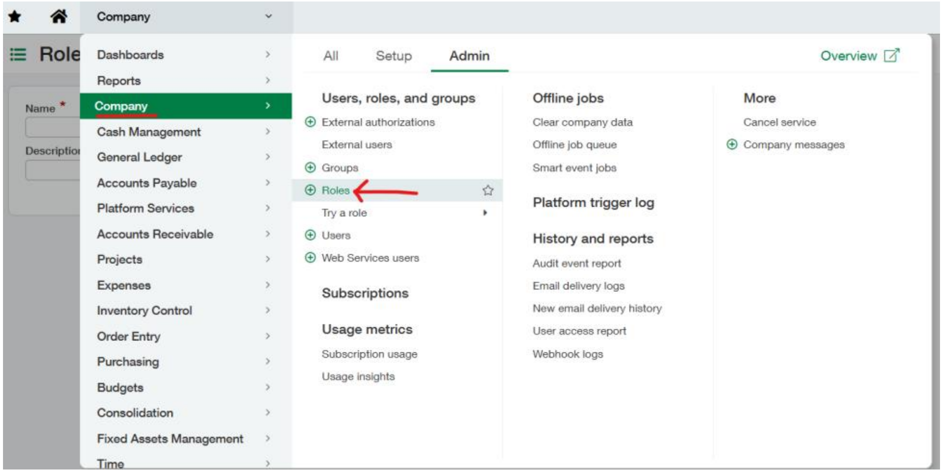


In the Subscriptions list, the Web Services option should be enabled to use the Sage API to communicate with Rillion.

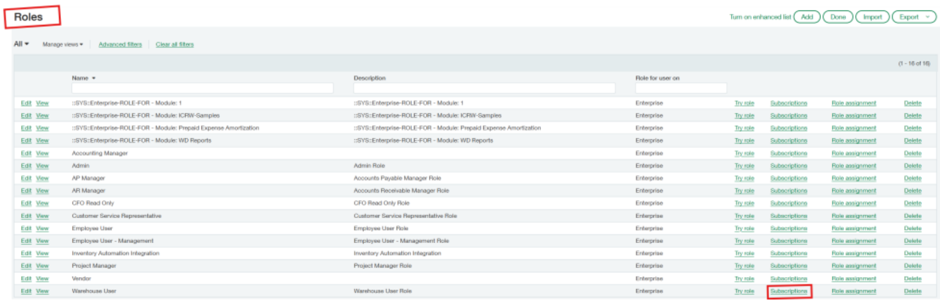


Step 3 – Creating a Rillion Role to be used by Web Service User

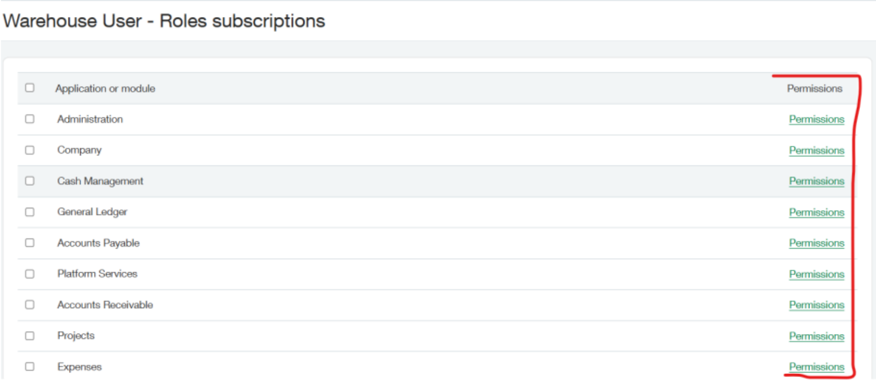
The Sage Intacct admin user needs to create a Rillion-specific role to be assigned to the Web Service User in the next step and used only by the Rillion Sage Intacct integration. To do that navigates to the following UI



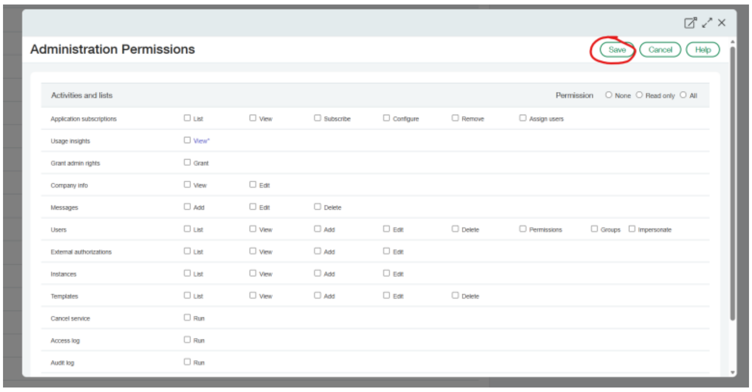
After adding the new role, you can see the newly created role in the "Roles" listing. Click on the "Subscriptions" link to add the permissions to the role.



By clicking the permission link for each module, you can assign the permissions to the role.

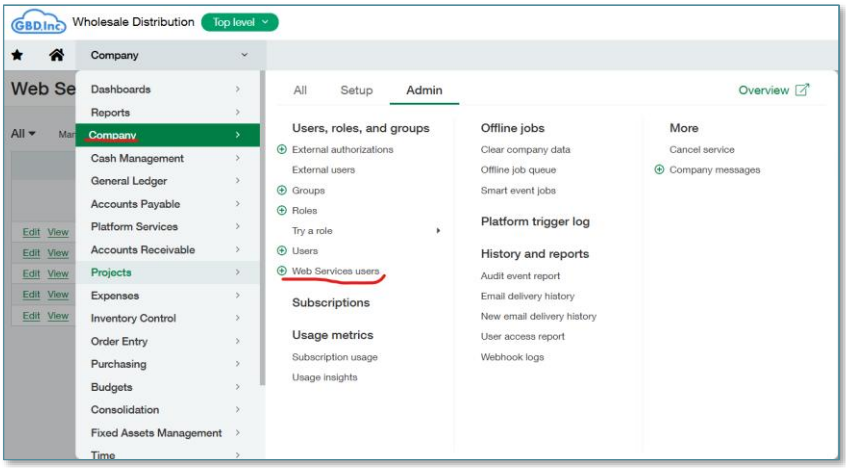


By clicking the checkbox, assigning the permissions, and clicking the save button, you can save the role.

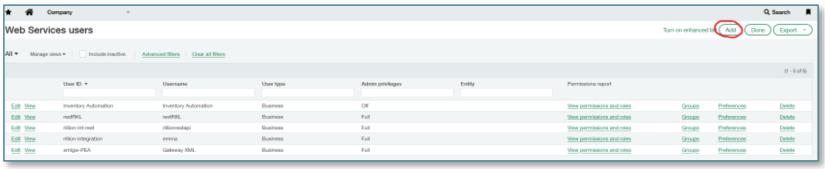


Step 4 – Creating the Web Service User

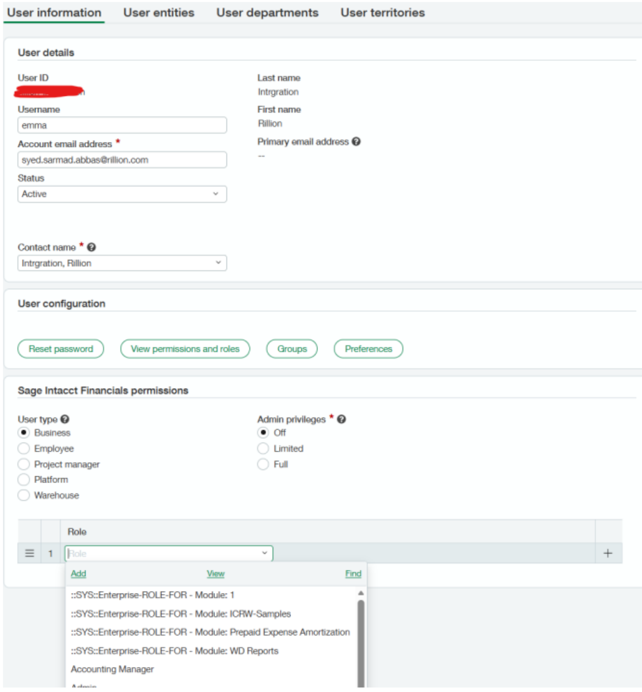
Create a Web Service User - Click on the main menu → Company → Admin Tab → Web Service users.



Now click on “Add” to add a new Web Service user.

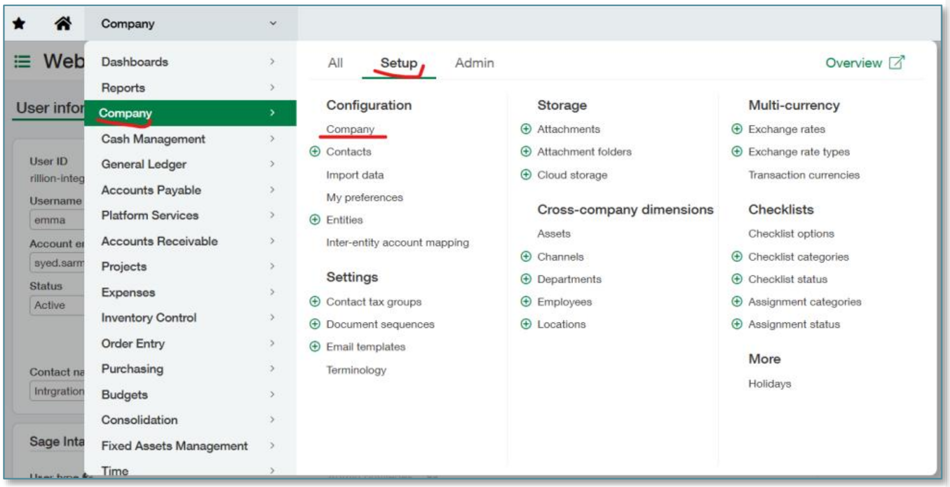


Now fill the fields, and permissions selection should be as shown below in the image. Please note the USERNAME does NOT need to be "Emma", this was purely for demonstration purposes. It can be any meaningful username as determined by the customer. "Rillion" or "RillionConnect" can be used. Leave the Admin privileges to off. And select the role that we created in step 3.

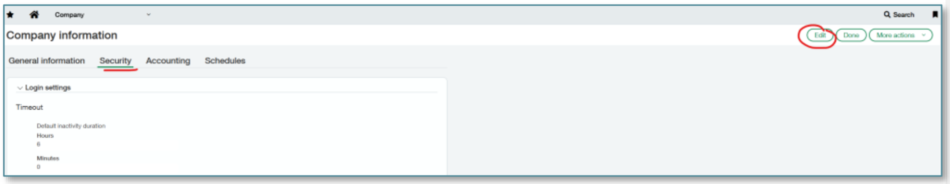


Step 5 – Adding the configurations to the Company

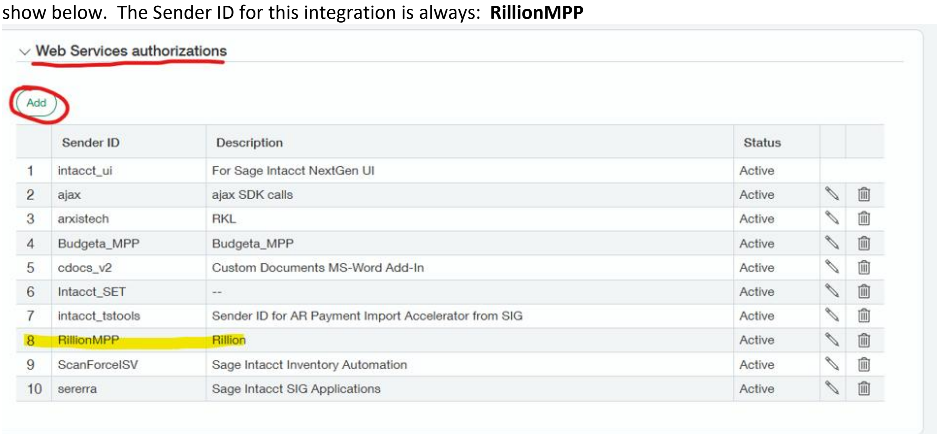
Authorize the Rillion Sender ID to authorize the Web Services User - Click on the main menu → Company → Setup Tab → Company



Now go to the Security Tab and click on the “Edit” button.



Now scroll to the Web Service authorizations section and click “Add” the Sender ID for the Rillion integration as show below. The Sender ID for this integration is always: **RillionMPP**

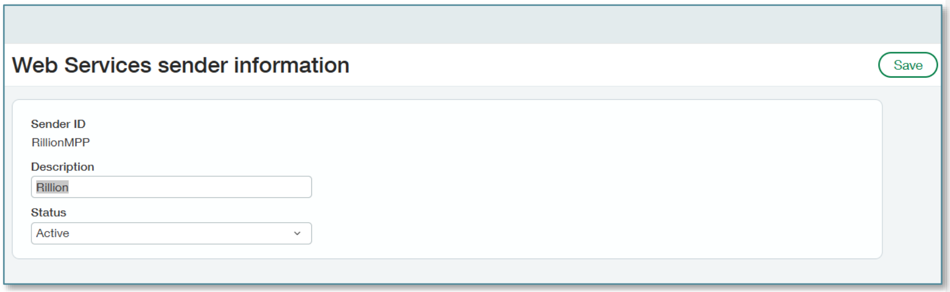


Web Services authorizations

Add

	Sender ID	Description	Status
1	intacct_ui	For Sage Intacct NextGen UI	Active
2	ajax	ajax SDK calls	Active
3	arxistech	RKL	Active
4	Budgeta_MPP	Budgeta_MPP	Active
5	cdocs_v2	Custom Documents MS-Word Add-In	Active
6	Intacct_SET	--	Active
7	intacct_tstools	Sender ID for AR Payment Import Accelerator from SIG	Active
8	RillionMPP	Rillion	Active
9	ScanForceISV	Sage Intacct Inventory Automation	Active
10	sererra	Sage Intacct SIG Applications	Active

Fill the details as follows, and the Sender ID must be the same that you received in the email. And now click on save



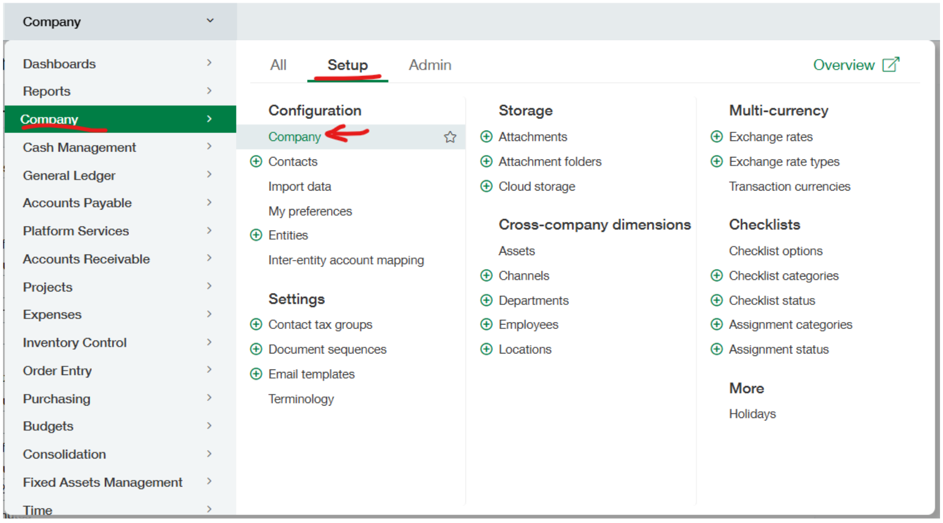
Step 6 – Setting up the API Keys

This step is where the customer adds the Client ID which is

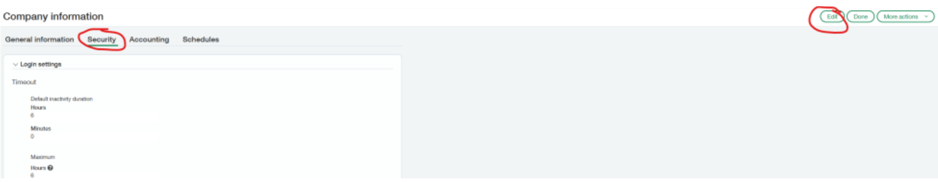
- 55524a851771f75b6640.app.sage.com

for the Rillion Sage Intacct integration in their Sage Intacct environment.

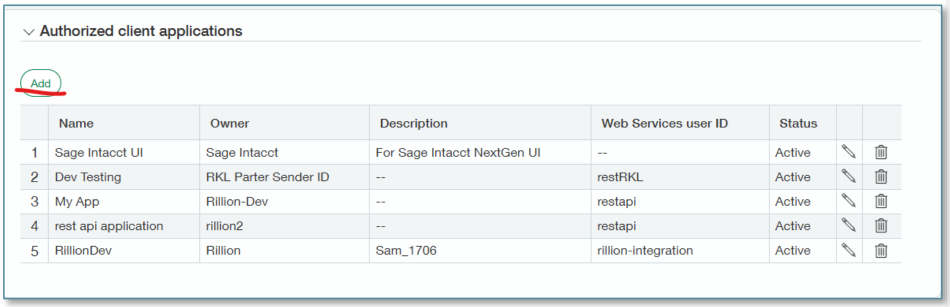
Go to main menu and then Company → Setup → Company



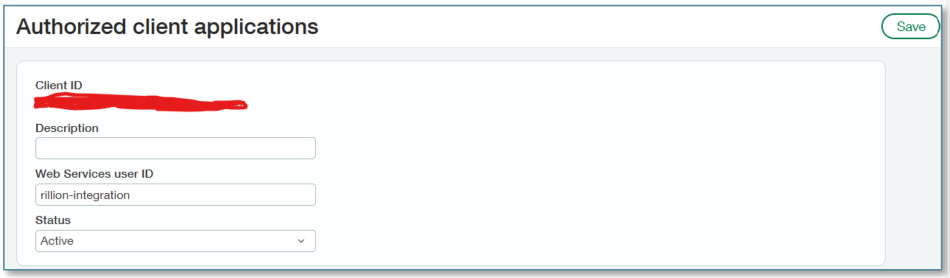
Go to the "Security" tab and click the "Edit" button



Once you clicked the edit button the page allowed to edit the Company information and now, go to the section Authorized client applications and click on the "Add" button.



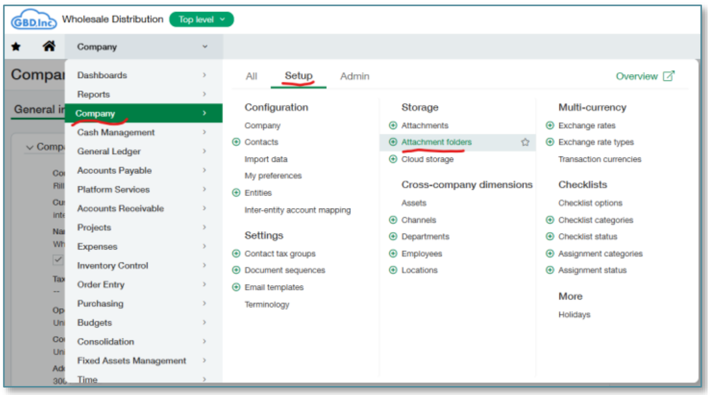
	Name	Owner	Description	Web Services user ID	Status		
1	Sage Intacct UI	Sage Intacct	For Sage Intacct NextGen UI	--	Active		
2	Dev Testing	RKL Parter Sender ID	--	restRKL	Active		
3	My App	Rillion-Dev	--	restapi	Active		
4	rest api application	rillion2	--	restapi	Active		
5	RillionDev	Rillion	Sam_1706	rillion-integration	Active		



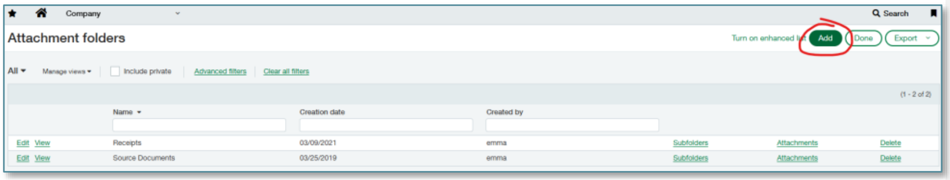
Note, the "Web Services user ID" in the screen shot above refers to the web services user the customer created in Step 4 previously in this section.

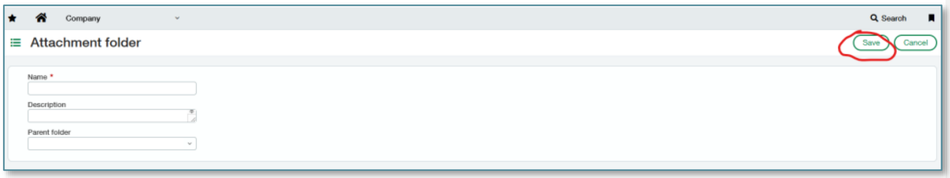
Step 7 – Add a folder

Now, we need to perform one more action on the Sage UI. Go to Company → Setup → Attachment folders.



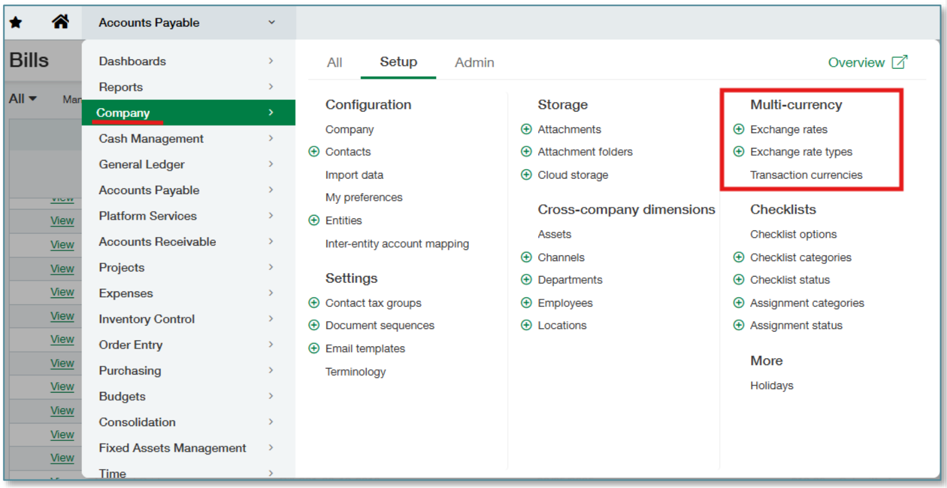
Add a new folder entity and name it "**Receipts**" and click save. We will use this newly created folder entity later in the process.





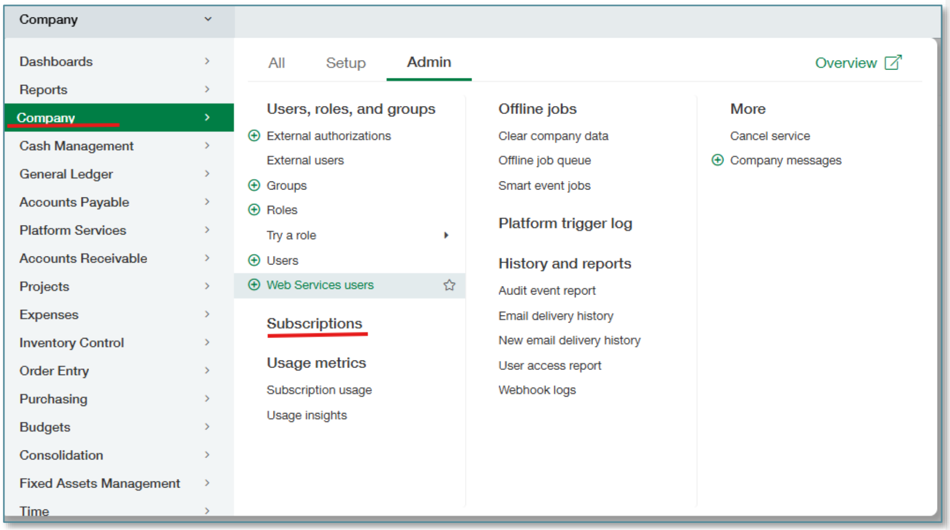
Step 8 – Enabling the Currency Module

By default, Sage did not allow us to work around multiple currencies and define custom exchange rates. To do that, we need to follow the steps shown below.

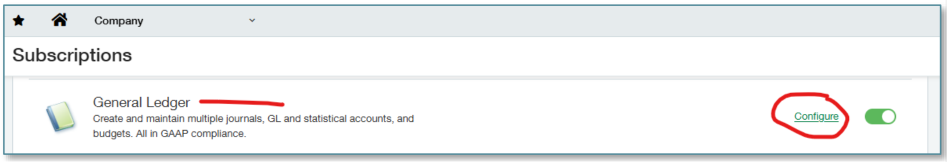

Accounts Payable
Bills
Dashboards
Reports
Company
Cash Management
General Ledger
Accounts Payable
Platform Services
Accounts Receivable
Projects
Expenses
Inventory Control
Order Entry
Purchasing
Budgets
Consolidation
Fixed Assets Management
Time
All
Setup
Admin
Overview
Configuration
Company
Contacts
Import data
My preferences
Entities
Inter-entity account mapping
Settings
Contact tax groups
Document sequences
Email templates
Terminology
Storage
Attachments
Attachment folders
Cloud storage
Cross-company dimensions
Assets
Channels
Departments
Employees
Locations
Multi-currency
Exchange rates
Exchange rate types
Transaction currencies
Checklists
Checklist options
Checklist categories
Checklist status
Assignment categories
Assignment status
More
Holidays

Step 9 – Enabling Multicurreny Subscription

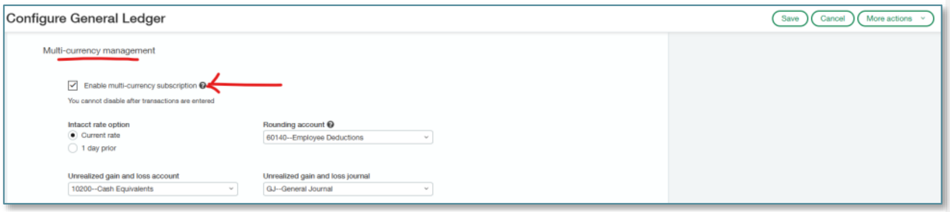
Go to Company → Admin → Subscription



Go to the General Ledger and click on Configure Link

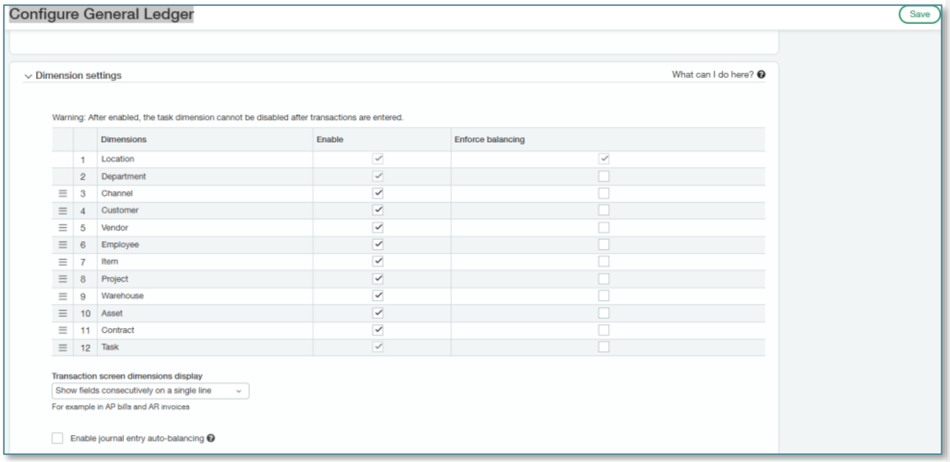


Scroll down to the “Multi-currency management” section and enable the option by checking the checkbox, fill out the accounts feilds and click on the save button.



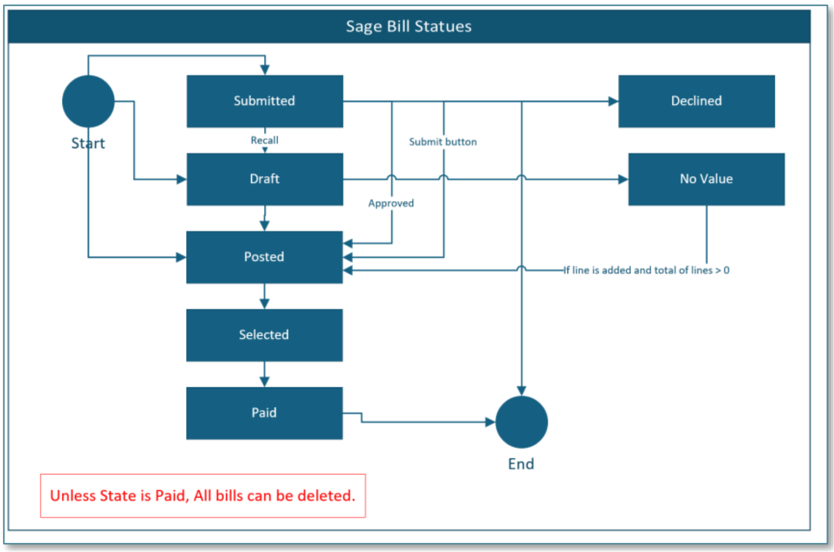
Step 10 – Enabling the Dimension settings

If you are not able to see some dimension that you want to see, then go to the Configure General Ledger, as shown above, and enable the option as needed.

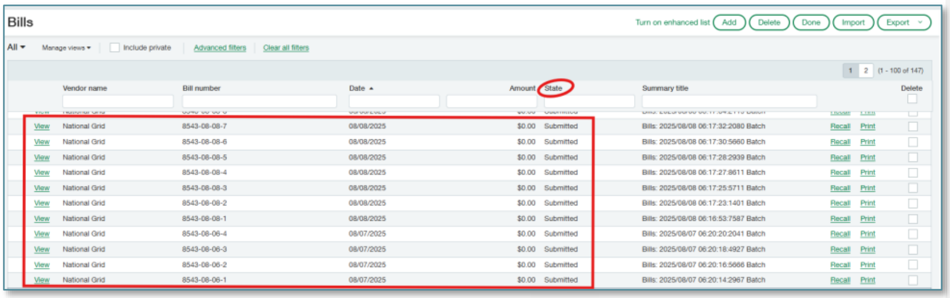


Step 11 – Sage Invoice / Bill Workflows

There are different states of the invoice or bill in Sage. You can see the flow diagram below to see the state and how the bill moves between different states.

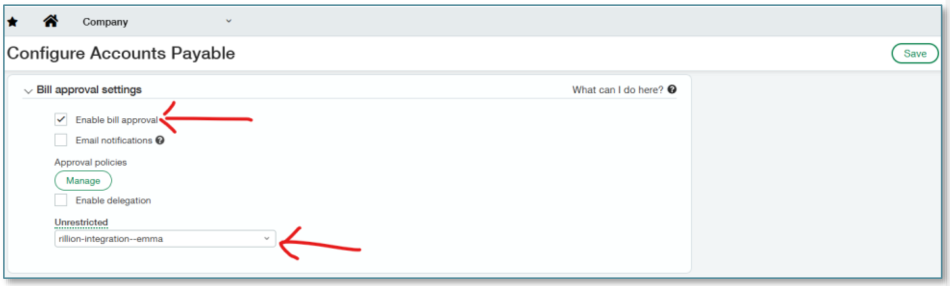


We need to give the Web Service user the right to post the invoice/bill in the Posted state so that the customer doesn't need to approve the bill in the Sage environment to proceed



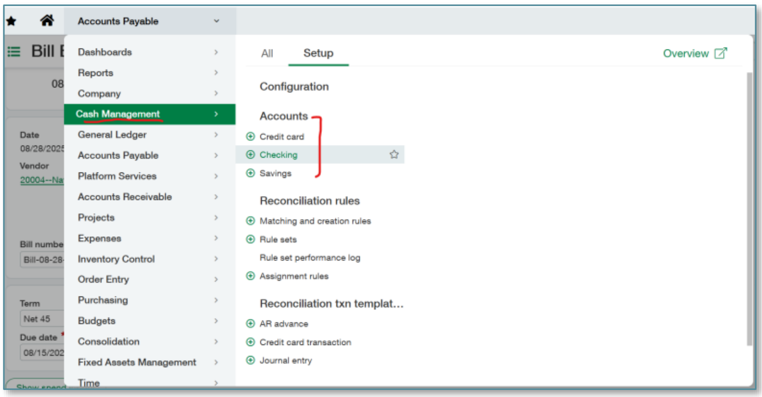
To do that, we need to go to Company → Subscription → Account Payable and scroll to the “Bill approval settings” section.

We need to enable the option as shown below. In the unrestricted dropdown, select the web service user, which was configured during client ID setup. After filling in the details, click the save button to save the changes.



Step 12 – Sage Bank Accounts

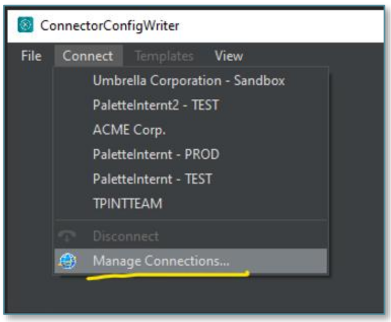
To pay the invoice, the Prime account needs to be configured before initiating the payment process. For different methods of payment like check, credit card, and savings, we need to define the banks in the respective sections as shown below.



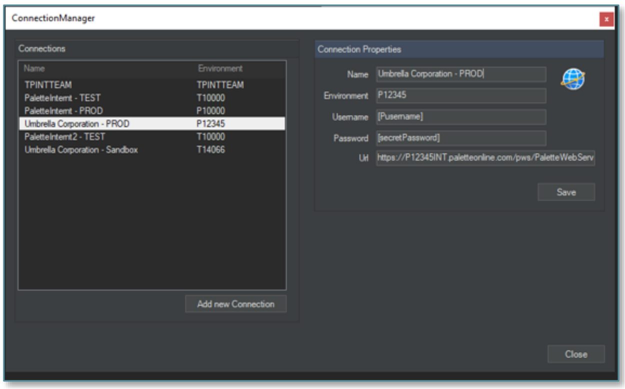
Configuring Connect - RILLION

Step 1 - Connect with Rillion Prime Arena and Connect ConfigWriter (CW)

1. Open ConfigWriter (CW), which is a Windows app
    a. CW is found on the Rillion PO Jump server.
2. Add a "Connection" to the Rillion Prime Arena customer Environment.
3. In the Menu, click "Connect" -> "Manage Connections".



4. "Click "Add new Connection" button and fill in the form on the right.



5. Fill in the Connection info as follows:

Field	Description
Name	Free text field. Enter a descriptive name. It is recommended to add either "Test" or "PROD" as a suffix to indicate if it is a test or production environment.
Environment	Arena environment ID
Username	Username for Arena Web service

Password	Password for the Arena Web service
Url	URL to the web service asmx file.

6. Remember to click "Save" when done.

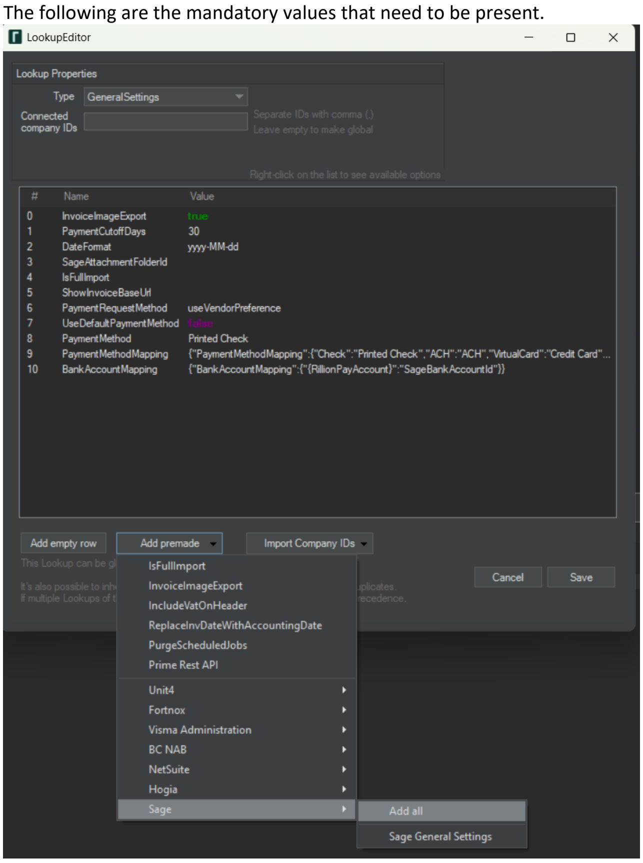
Step 2 – Add lookups in CW incl. General Settings and Credentials

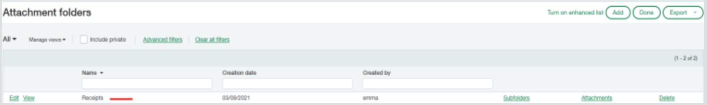
Lookups

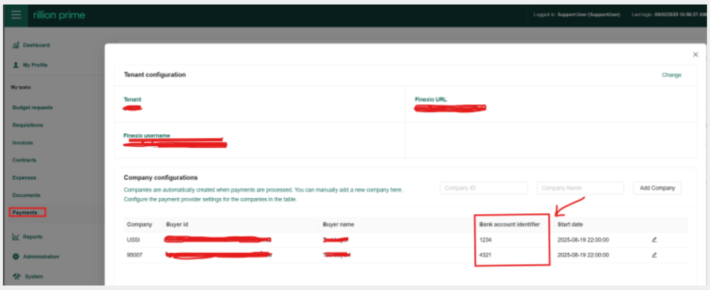
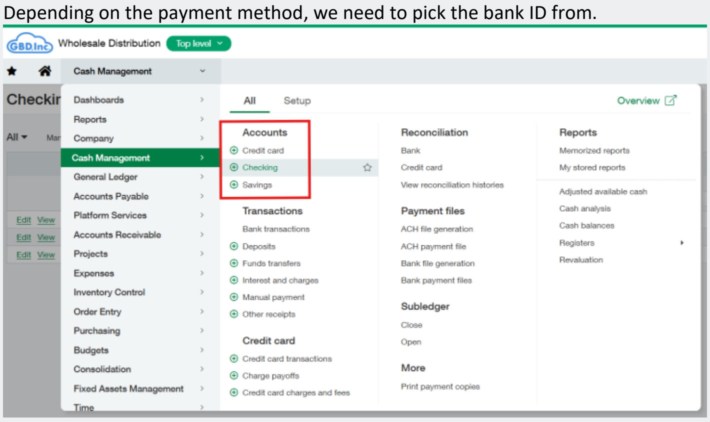
Lookups contain configuration and data needed for Sage jobs. A Sage integration requires "GeneralSettings", "Credentials", and "Objects" to function. To add a new Lookup, right-click in the Lookup list view and click "Add new".

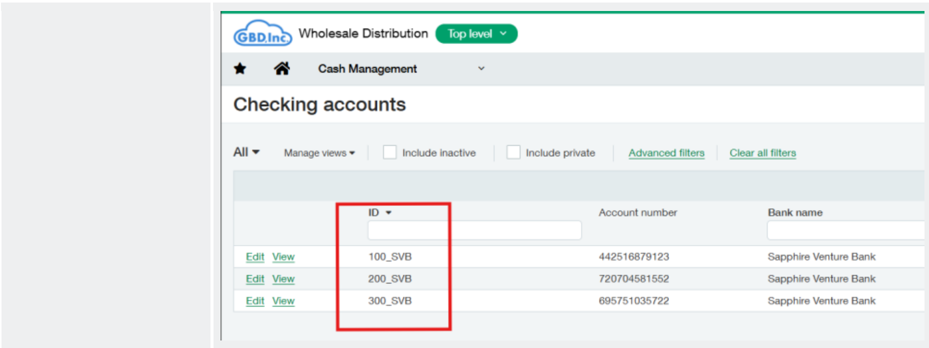
Lookups: General Settings

The following are the mandatory values that need to be present.



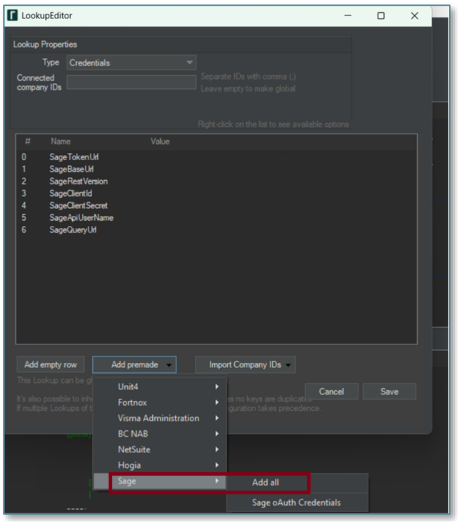
Field	Description
ShowInvoiceBaseURL	URL to customer Rillion Prime.
IsFullImport	Set to true to perform a full registry import. Set to false only to import records changed since the last 7 days. Applies to the object and the supplier.
InvoiceImageExport	Send the invoice attachments.
PaymentCutoffDays	The number of days for fetching the payment date, waiting for the invoice from Prime
DateFormat	Date format used for the payment date endpoint calling.
SageAttachmentFolderId	Get the newly created folder name from the attachment list and paste it under this configuration. 
PaymentRequestMethod	Specifies how payment requests are generated. Default: "useVendorPreference". other values (case sensitive): generateOneRequestPerBill, mergeRequestIntoOnePerVendorPayToContact, mergeRequestsIntoOnePerVendor
UseDefaultPaymentMethod	Ignore the payment method coming from the Rillion pay endpoint and use the CW payment method.
PaymentMethod	The payment method used to send the payment in Sage. The following are the possible values: • Printed Check (Default) • Cash • EFT • Credit Card • ACH
PaymentMethodMapping	The following is the sample and default value for how we can map the Rillion-Pay Payment method to Sage Payment methods. In the following JSON, the Key value should be the RillionPay PM, and the Value property will be the SAGE PM. {"PaymentMethodMapping":{"Check":"Printed Check","ACH":"ACH","VirtualCard":"Credit Card","Other":"EFT"}} All Possible PM in Rillion Pay: • Other • ACH • Check • VirtualCard

	All Possible PM in Sage: • Printed Check (Default) • Cash • EFT • Credit Card • ACH
BankAccountMapping	The following sample shows how we can set up the mapping between the RillionPay BankAccount and Sage BankAccount. {"BankAccountMapping":{"{RillionPayAccount}":"{SageBankAccountId}"}} **RillionPay Bank Account (Key properties):** Navigation Path: Payments ➔ Manage Button ➔ Select “Payment provider” ➔ Bank account identifier  **Sage Bank Accounts (Value properties):** Depending on the payment method, we need to pick the bank ID from.  Select the bank and pick the Bank ID, and paste it in the JSON as a value property.



Lookups: Credentials

It contains 7 mandatory values.



Field	Description
SageTokenUrl	Sage Identity Server URL for getting the authentication URL – Rillion has this URL
SageBaseUrl	URL used to communicate with the Sage REST APIs – Rillion has this URL
SageQueryUrl	URL for querying the objects fetched from Sage – Rillion has this URL

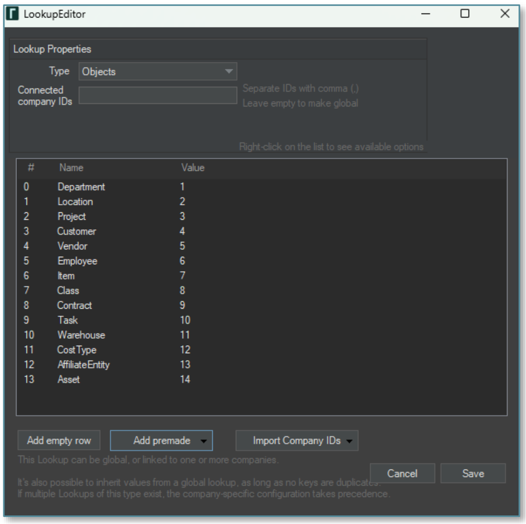
SageRestVersion	Version of REST APIs – Rillion Professional Service resource needs to check with the Rillion Integration team to check the latest available version number of the Sage published REST API. This version number is then expressed in this field as "v.1.X" for example.
SageClientId	Client id for the customer – Provided by customer, see step 6 above This step in CW is in addition to step 6 from previous section done by customer
SageClientSecret	Client Secret for the customer – Provided by customer, see step 6 above This step in CW is in addition to step 6 from previous section done by customer
SageUserName	UserName for the customer **Note: In this field, we need to specify the value in the following format {WebServiceUserID}@{CompanyID}** – Provided by customer, see step 6 above

Commented [MZ1]: @Syed Sarmad Abbas - Sage want us to remove these as they say customer does not give us those, these are id & secrets we make.

Lookups: Objects

Add the registries and objects / dimensions used by the customer. Sage supports more than 10 dimensions, but Prime only supports up to 8 dimensions, so we need to select the most frequently used by the customer.

Note: *Department and Location are mandatory in Sage.*

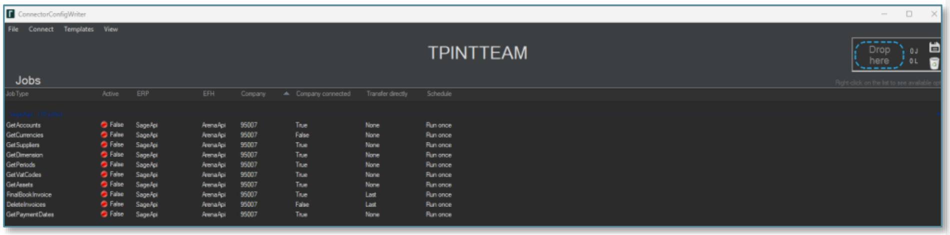


Step 3 – Add jobs in CW

Jobs

To perform any import/export, you will need to set up a job for that task. To add a new job, right-click in the Jobs list view and click "Add new Job…".

Field	Description
JobType	Select appropriate value. "GetDimension" will import all defined dimensions.
Erp	Select SageApi
CompanyID	Enter Rillion Prime Arena companyID
Cron	Click Edit to set the run interval. There are predefined values to choose from. Refer to ConnectConfigWriter documentation for more information.
TFQ	Refer to ConnectConfigWriter documentation for more information.
Active	Check (true)
Connect registers to Company	Check (true)
Transfer directly	Run the transfer from the queue directly. None, All, Last.



Transfer from queue tasks is created and run in Rillion Prime Master for better performance.
Jobs for PrelBook invoice, FinalBook invoice, Delete invoice is set to transfer directly=last.
Meaning transfer from the queue is run on the current job.

***Note:** When it comes to canceling the invoice, we are only able to cancel the invoices that are in the state of "Posted, Draft, Declined, NoValue". If the invoice is in a state other than this, the process will fail to cancel that invoice.*

Tips & Tricks/Troubleshooting & FAQ

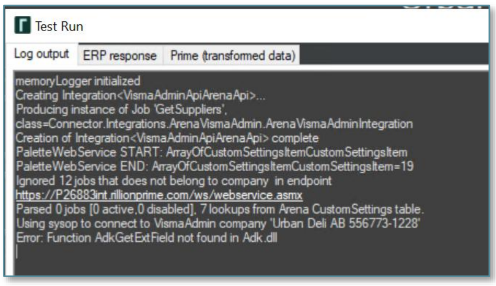
When there are issues with the integration/in the log, first thing to do...

When there are issues with the integration or seeing issues in the Connect log files, the very first thing to do is check the Connect installation and look through the Connect folder for the file named ***version.json*** to find the exact version of Connect that is running.

There have been many issue fixes in the Sage integration over time, so it is important to ensure they are running the latest version of Connect. (Can be downloaded from RCP)

Integration "Ignores" jobs for a specific company

When running or testing the integration from ConfigWriter, you begin to see error messages that the jobs are ignored. Similar to an error message as shown below:



Troubleshooting steps:

1. Need to check all the steps mentioned in "Prep of Sage"
2. In the ConfigWriter all the mandatory properties are filled out
3. The "**SageApiUserName**" property in the **Credential** lookup is correctly configured

Job spacing (timing)

Always space jobs at least 8 minutes apart when there is a multi-company setup in CW.

Payment Workflow

When a bill is created in Sage, if the bill's state is “Posted, Selected, PartiallyPaid” then you can make a payment against that bill. Some scenarios need to be discussed as follows:

Scenarios 1

You can only pay against the bills that are in the state of “Posted, Selected, PartiallyPaid”.

Scenarios 2

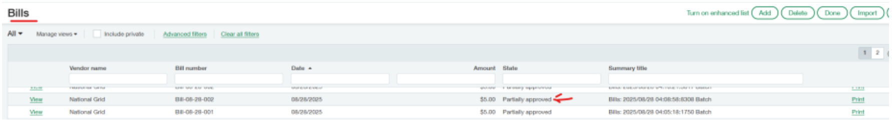
The Date of the payment such be equal or greater then the bill creation date otherwise following error message will be generated by SAGE

```
"details": [
                {
                    "errorId": "SL-0417",
                    "code": "informationMissing",
                    "message": "Select a payment date that occurs after the bill 
creation date, then try again.",
                    "additionalInfo": {
                        "messageId": 
"IA.SELECT_A_PAYMENT_DATE_THAT_OCCURS_AFTER_THE_CRE",
                        "placeholders": {
                            "THIS_GETTRANSACTIONTYPE": "bill"
                        },
                        "propertySet": {}
                    }
                }
            ]
```

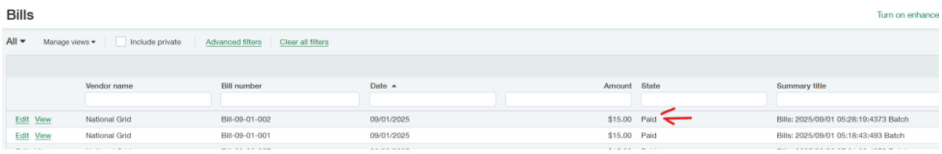
Scenarios 3

The amount of the payment will be less than or equal to the total amount of the bill.

If the bill payment is less than the bill amount, the bill's status will become “Partially approved”.



Next, if you pay against the same bill, the bill amount should be equal to the remaining amount of the bill. Once paid fully, the state of the bill becomes “Paid”.



Scenarios 4

If the of the payment is greater than total amount of the bill, then will encounter the following error


```
"details": [
                {
                    "code": "BL01001973",
                    "message": "The amount received is greater than the amount to
apply. Reduce the amount received, reduce the credit amount to apply, or add an
overpayment. Then try again.",
                    "additionalInfo": {
                        "placeholders": {},
                        "propertySet": {}
                    }
                }
            ]
```

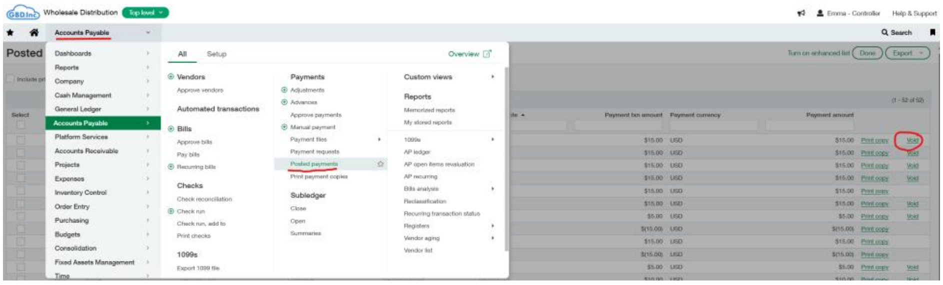

Scenarios 5

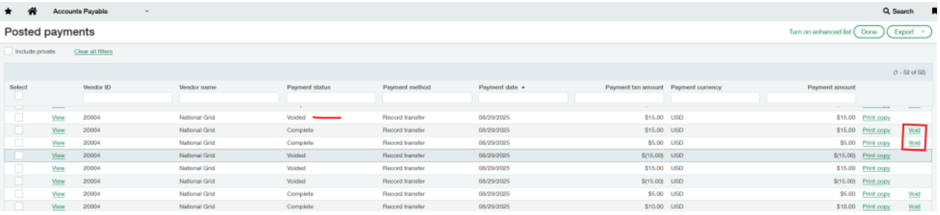
When trying to create a payment for a payment method and against that payment method, the Bank Account is not configured or a wrong account is used against that Payment Method in Sage, then you will encounter the following error message


```
"details": [
                {
                    "errorId": "AP-0038",
                    "code": "informationMissing",
                    "message": "Select a valid credit card account, then try
again.",
                    "additionalInfo": {
                        "messageId": "IA.INVALID_CHARGE_CARD_ACCOUNT_SELECTED",
                        "placeholders": {},
                        "propertySet": {}
                    }
                }
            ]
```

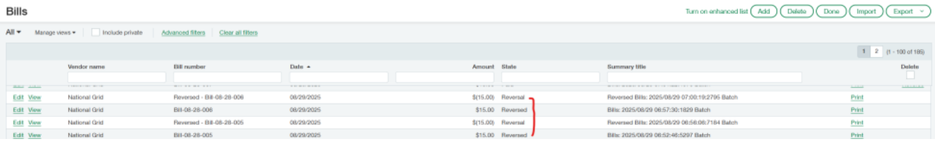
Scenarios 6

When the payment needs to be voided, you need to come to the SAGE UI and navigates to AP → Posted payments and select the payment to void and click on the void and follow the instructions.





NOTE: *If the payment is voided against the bill then the original bill will be reversed and need to create a new bill.*



Governance – RILLION Only

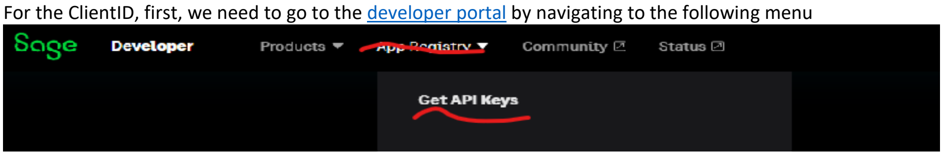
Creating a Sage Intacct integration app ClientId and Secret keys

The following steps are **ONLY** to be used by Rillion Integration Team development members. These are the steps we follow to create a new version of the Sage Intacct integration app which requires us to create new ClientID's and Secret keys.

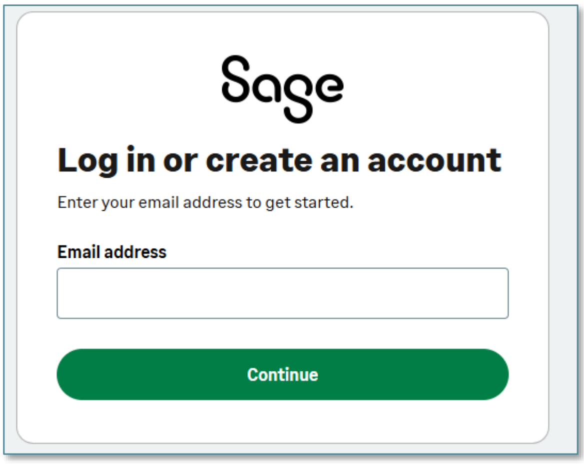
The app for Production client ID is created only ONCE by Rillion and that same client ID is used across all Sage Intacct customers using this Rillion Connect integration.

Step 1 – Login to the Developer portal

To generate the ClientID, follow the following steps
For the ClientID, first, we need to go to the developer portal by navigating to the following menu

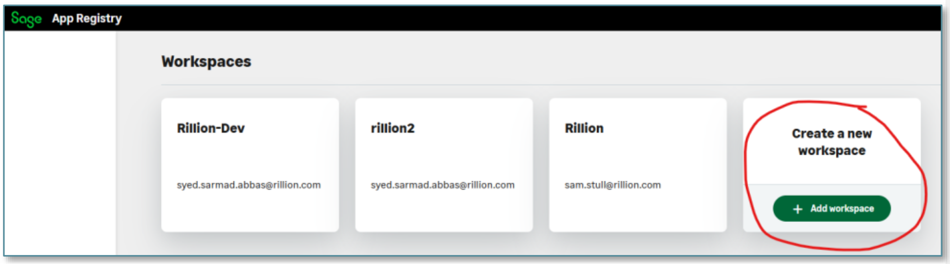


And then, create the account with the Sage portal owner email, and login

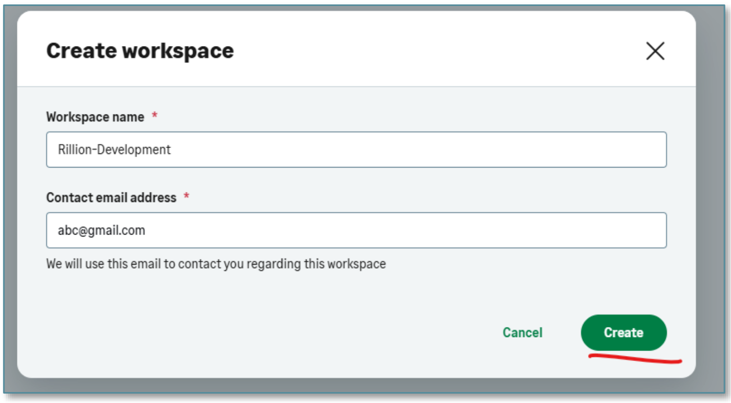


Step 2 – Adding the Workspace

Click on the Add workspace button

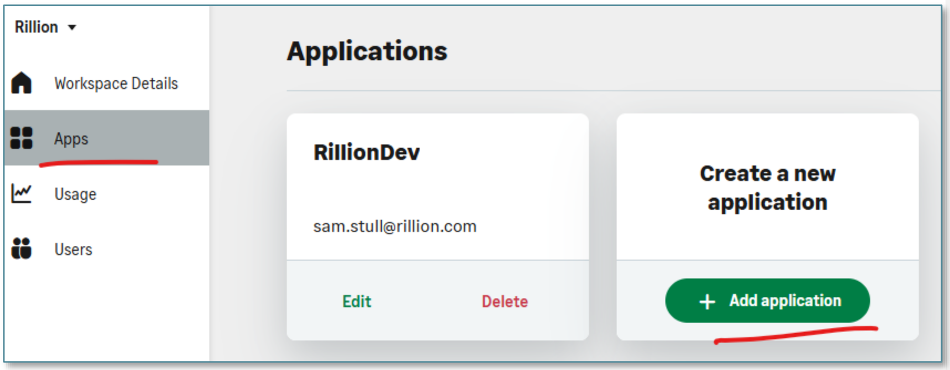


Enter the required information, enter the email address, the same as the owner's address of the Sage portal

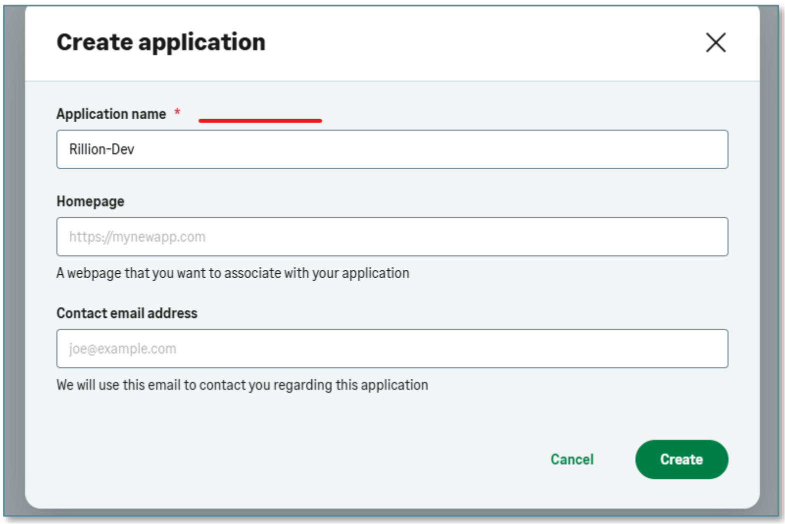


Step 3 – Creating the new application

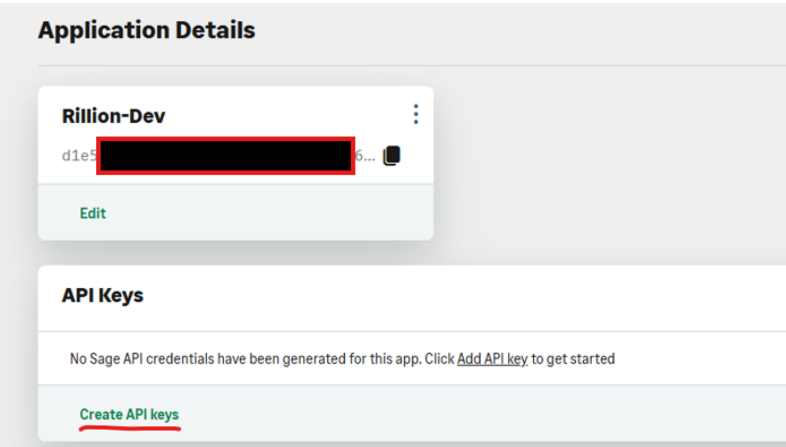
Go to the Apps tab and add a new application.



Name the app and hit the create button.

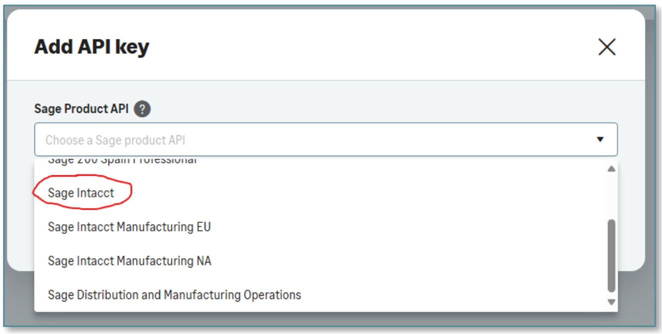


After creating the new application, click on it. There you will see the details of the application.



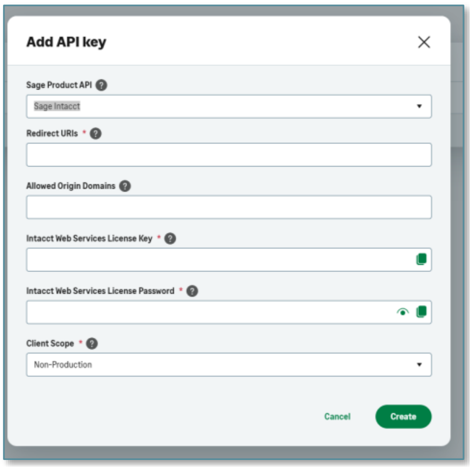
Step 4 – Creating the API key

Now click on the Create API keys and select "Sage Intacct" from the dropdown



Fill the mandatory fields:

- **Redirect URLs** = https://www.sage.com
- **Intacct Web Services License Key =** will be obtained from the email received from the Sage image shown below.
    - **Note:** If the Client does not activate the API access activation link within 7 days, the customer must generate a Ticket in the Sage portal to regenerate the link.
- **Intacct Web Services License Password:** will be obtained from the email from Sage.
- **Client Scope:** depends on the environment
    - **Note:** Once you generate the Production Scope API key, it starts billing as soon as you start using it in any deployed environment (Sandbox/Prod). So, if we are doing testing or UAT, use the Non-Prod scoped keys.



Once the fields are filled out and all information is good, then ClientID and Secret will be generated.