Quick Integration

Setup Guide

July 2020

coupa

Table of Contents

Introduction

Risk Assess Quick Integration provides users a fast and simple way to automate the import or export of small amounts of data into or out of the system without user intervention.

There are three types of Quick Integration:

- Quick Integration Inbound – used to import information into Risk Assess
- Quick Integration Outbound – used to export information out of Risk Assess
- Quick Integration Roundtrip – used to pull data from various sources in Risk Assess and combine, change or add new data and post the results back into Risk Assess

Assumptions

- The Quick Integration module has been purchased and enabled for your Risk Assess environment. For pricing information, speak with your Coupa Sales Executive.
- The client's IT team has committed to be engaged for this implementation. Setup will be required by the client's IT team.

First Steps

Work with Client Services or Customer Support to ensure the following steps have been completed:

- The Quick Integration module is turned on for the tenant
- The 'QuickIntegration' group is created
- A Quick Integration user is created
- The Quick Integration user is added to the 'QuickIntegration' group

Quick Integration Inbound

Risk Assess Quick Integration Inbound provides the same import functionality exposed via the Risk Assess web site accessible via an HTTP POST. This allows for automated processes to upload files on a recurring schedule or as needed. Supported file formats include all versions of Excel and comma separated value (CSV) files. This section of the document outlines the steps necessary to implement the Quick Integration Inbound functionality for your Risk Assess environment.

NOTE: No data manipulation can be done via this Integration. As described below, the format of the file to be imported into Risk Assess via the Quick Integration process is a standard template. If any data manipulation, validation, error handling, rules and/or procedures based on the integration file are needed, the full Integration solution is required.

Setup

The Quick Integration import allows Risk Assess users to directly upload spreadsheets via the HTTPS protocol. This allows customers to perform inbound imports in an automated manner without the need of a user interface. The HTTP post request allows the sender to submit a file to be loaded by Risk Assess via the web service. This call to the Risk Assess web service can be initiated via several web apps, a browser or embedded into a coding language. Please see Appendix B to view sample code snippets that provide working examples of how this is performed using a variety of popular programming languages.

NOTE: For Future reference, the port number syntax :(Port number), whether for the Test environment or Production environment, should not be included in the QI Inbound Setup.

Schema

The Quick Integration process uses the same rules and functionality as a manual import of the standard data import templates. When Risk Assess Quick Integration receives a file, it will look at the headers of each column to import the data to the appropriate fields. Therefore, the first row of the worksheet must include the field names as the column headers. It is important that the correct syntax for the column headers are used, namely, the field names rather than the display names. It is also important that all required fields are present in the import template.

As with the manual imports, User Defined Fields (UDF) can be added to the files as well in a similar fashion. For example, if using the UDF import procedure for fields on the Relationship object, ensure that the appropriate Relationship columns are present in the file so that the import procedure can add the data to the correct relationship.

All the Import templates schemas and field formats can be found in the Community Section of your Risk Assess environment or by going to https://rportaltest.hiperos.com/QuickIntegration.

HTTP Headers

The following header information is required for all types of imports:

Authentication

Use the Quick Integration user as the login when importing data using QI ensuring that valid credentials are supplied for each Risk Assess Quick Integration request issued. If preferred, a dedicated integration account can be created in Risk Assess similar to the way other user accounts are created within the system.

Authorization Header	Description
username	Valid Risk Assess Username
password	Valid Risk Assess Password

Content Type

The content type header must contain one of the following types listed below. This is used to identify the file format that is being uploaded.

Content Type Header	File Type
application/vnd.openxmlformats-officedocument.spreadsheetml.sheet	Excel 2007 or later versions (xlsx)
application/vnd.ms-excel	Excel 2003 or earlier versions (xls)
text/csv	Comma separated values (csv)

NOTE: If sending data as an octet-stream, in addition to the Content-Type header, you will also need a Content-Disposition header where the filename is required. For example if sending a CSV file with supplier data your header may look like this - Content-Disposition: attachment; filename="supplier.csv".

Response Codes

The following HTTP response status codes are utilized by Quick Integration to communicate import status back to the caller. They are as follows:

Status Code	Result	Description
201	Created	New objects successfully created.
202	Accepted	Data has been queued for import (system-generated e-mail confirmation to follow).
204	No Content	No new or updated data to import was found or the file was not in the correct format.
400	Bad Request	Problem rows and/or columns were found. Review returned validation message in response.
401	Unauthorized	Username & password http headers were not found in the request.
403	Forbidden	Login failed, was not found or not authorized.
500	Server Error	Failed to save import data to database.

All QI files go through the Quick Integration queue, regardless of size. The queue runs on a scheduled basis. Upon successful import of the file, the records will be loaded into Risk Assess and a confirmation email will be sent to the Quick Integration email account alias.

NOTE: The response codes above will not specify problematic records. Furthermore, if there is a failure due to specific records in the file, the entire file will not be imported.

The best way to troubleshoot file import failures is to log into Risk Assess directly and to attempt a manual import of the same file. The Risk Assess interface will show you problematic records which may have caused the import to fail. After correcting those records, you may re-import the file manually, or via the Quick Integration. If you experience further trouble with the file import, even after correcting all the problematic records, contact the Coupa Support team for assistance.

Quick Integration Outbound

Quick Integration Outbound allows you to quickly set up and automate outbound file-based interfaces from Risk Assess to other systems.

File Transfer Options

Quick Integration Outbound offers two methods of exporting data out of Risk Assess:

1. **SFTP transfer of data**

   This method will transfer the file that is generated by reporting to a destination system. This method will use the SFTP configuration options that you set up within the report scheduler to authenticate and connect to the server, change to the destination directory on the server where the file should be transferred, and then initiate the transfer commands to securely transfer the file into the destination system.

The communication between Risk Assess and the SFTP site is encrypted via SSL (in-transit). Once the file is deposited at the SFTP site, the file is clear text, so it is viewable by any individuals with access to the SFTP site.

The file output formats supported for this method are: CSV, PDF, and Microsoft Excel.

2. **HTTPS posting of data**

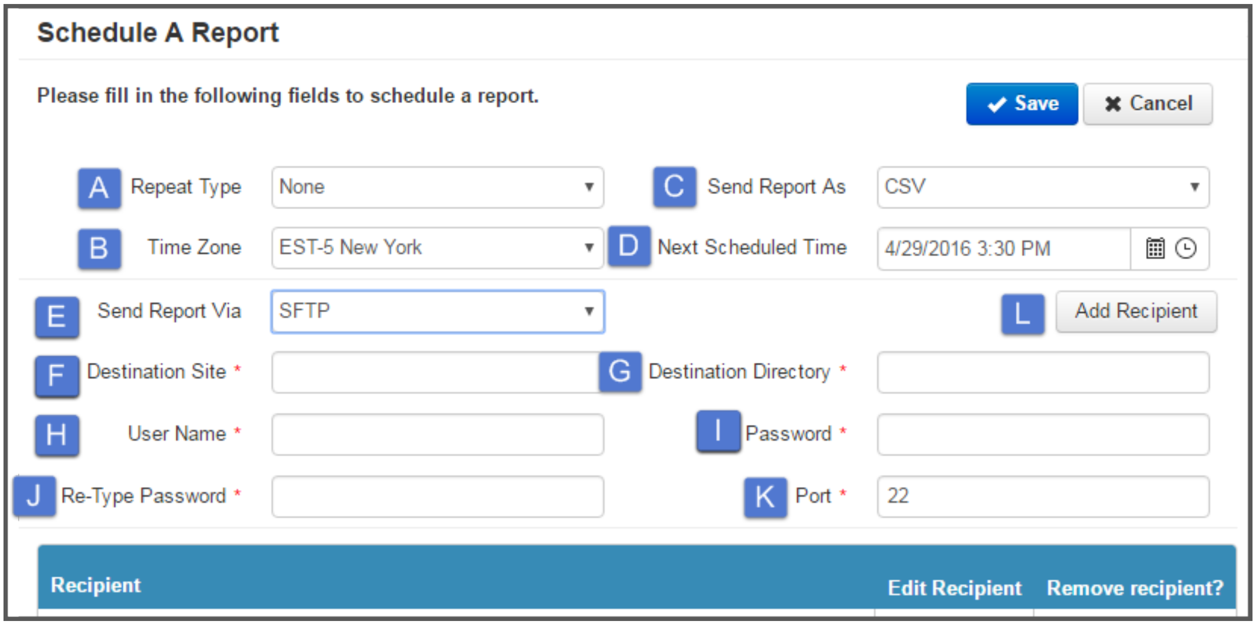
   This method will post a file that is generated by reporting to a destination using an HTTPS post communication. The system will generate the HTTPS posting message based on the configuration you set up in the report scheduler to authenticate, transfer the file, and integrate the file at the destination. HTTP encrypts the file during transport, but there must be an application to process the file on the customer's side.

   The file output formats supported: CSV, PDF, and Microsoft Excel.

SFTP File Transfer

To take advantage of the Quick Integration Outbound SFTP feature:

1. Navigate to the reports screen
2. Identify or build a report that you would like to securely transfer to another system
3. Click on the Scheduler icon of the report



Schedule A Report

Please fill in the following fields to schedule a report.

Repeat Type	None	Send Report As	CSV
Time Zone	EST-5 New York	Next Scheduled Time	4/29/2016 3:30 PM
Send Report Via	SFTP		
Destination Site *		Destination Directory *	
User Name *		Password *	
Re-Type Password *		Port *	22

Recipient	Edit Recipient	Remove recipient?

A. Repeat Type: choose the frequency
B. Time Zone: choose correct time zone
C. Send Report As: select only one of the following output types - CSV, PDF, Microsoft Excel
D. Next Scheduled Time: choose a time at least one hour later the current time

E. Send Report Via: SFTP

F. Destination Site: enter the IP Address of the server that will be accepting the file transferred from Risk Assess via SFTP

G. Destination Directory: enter the name of the directory on the destination server into which the file should be transferred

H. User Name: enter the user name credential that will be used when issuing the SFTP commands to authenticate on the destination server

I. Password: enter the password credential that will be used in conjunction with the user name when issuing the SFTP command to authenticate on the destination server

J. Re-Type Password: enter the same password again. The system will ensure that you entered the same password twice

K. Port: enter the port number through which the SFTP should occur

L. Click the Add Recipient button

-

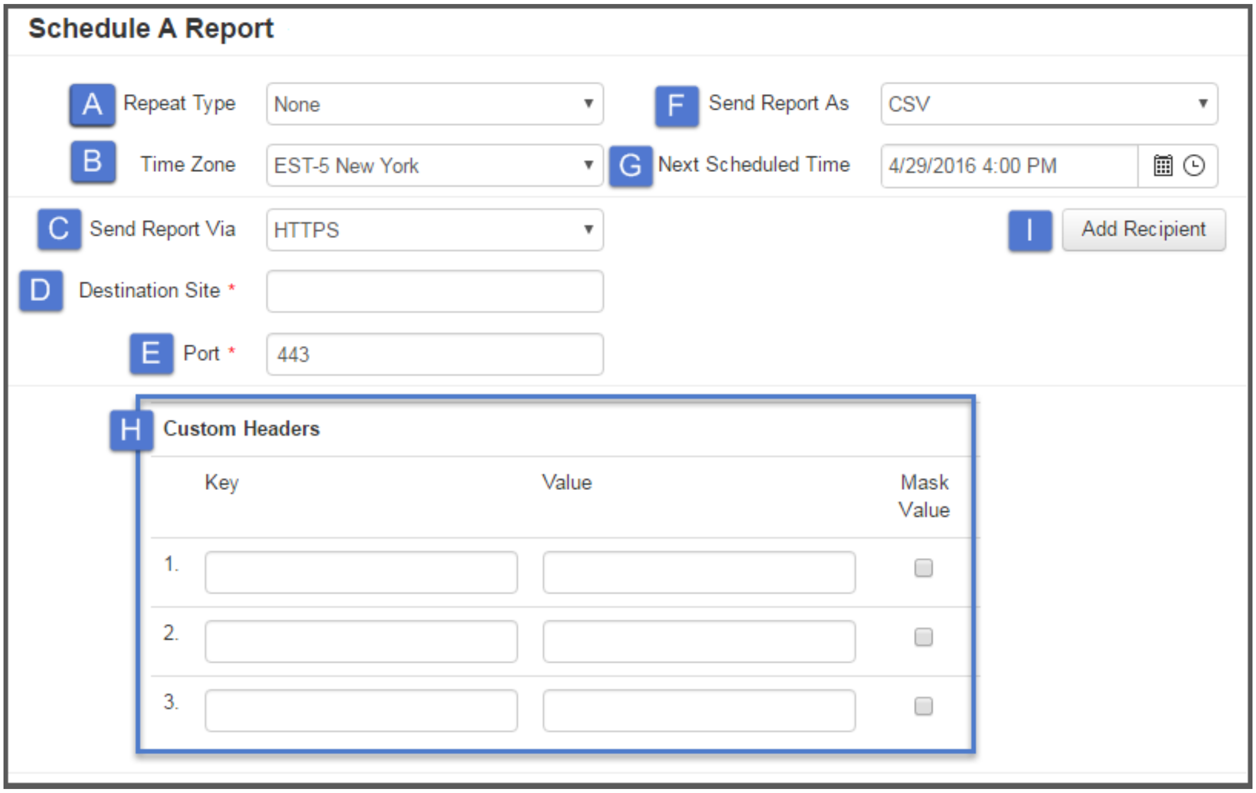
4. The report will run on the schedule you defined and will transfer the result file to the server you configured using the credentials and other technical details you provided during the scheduler configuration.
5. Any issues that may have been encountered when transferring the file will be emailed to the default site administrator of your tenant
6. Whitelist 74.205.24.151 for external access to your SFTP Server.

HTTPS File Transfer

To take advantage of the Quick Integration Outbound HTTPS feature:

1. Navigate to the reports screen
2. Identify or build a report that you would like to securely transfer to another system
3. Click on the Scheduler icon of the report

-



Schedule A Report

A Repeat Type: None

B Time Zone: EST-5 New York

C Send Report Via: HTTPS

D Destination Site *:

E Port *: 443

F Send Report As: CSV

G Next Scheduled Time: 4/29/2016 4:00 PM

I

H Custom Headers

	Key	Value	Mask Value
1.			☐
2.			☐
3.			☐

A. Repeat Type: choose the frequency

B. Time Zone: choose correct time zone

C. Send Report Via: HTTPS

D. Destination Site: enter the URL destination that will be accepting the file via HTTPS that Risk Assess transfers

E. Port: enter the port through which the communication will occur, likely port 443

F. Send Report As: select the appropriate output type - CSV, PDF, and Microsoft Excel

G. Next Scheduled Time: Choose a time at least one hour later the current time

H. Custom Headers: enter all key value pairs that you will need to successfully initiate the posting request to the destination. Values will likely include user name and password but may include another value pair as well.

I. Click the Add Recipient button

4. The report will run on the schedule you defined and will post the result file to the destination you configured using the credentials and other technical details you provided during the scheduler configuration.
5. Any issues that may have been encountered when transferring the file will be emailed to the default site administrator of your tenant

Quick Integration Round Trip

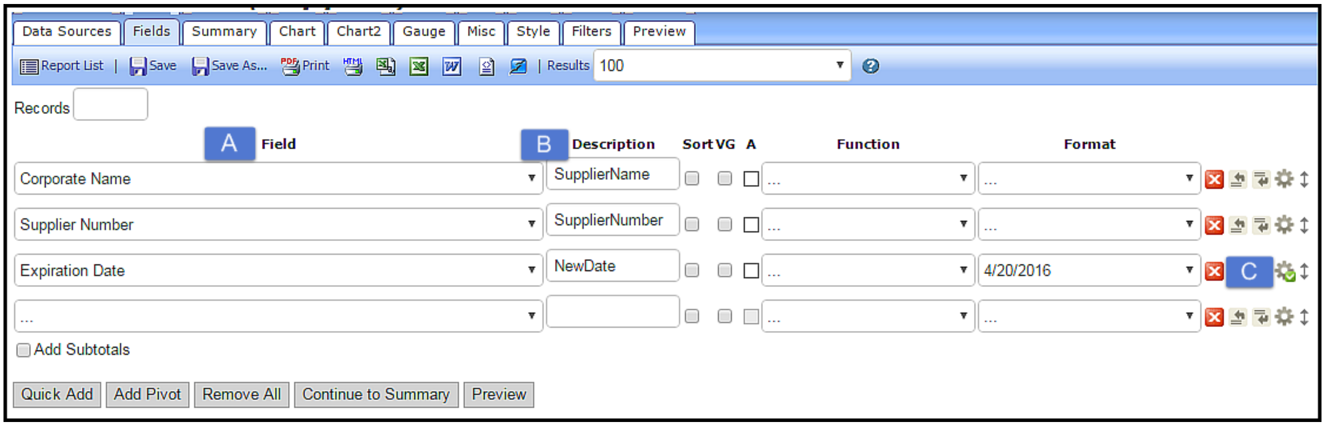
It is possible, using reports, to pull data from various sources in Risk Assess, combine, change or add new data and post the results back into the system. The combined/new data is written to the designated field(s) on import. This functionality can be used in a variety of different ways. For example:

- Information residing on a Relationship record could be used to populate data on an associated Supplier record, or vice versa
- Computations done on a report, could be imported to a Supplier or Relationship record
- Data could populated on a Parent Supplier by pulling it from a Subordinate Supplier

Setup

Create the Report

1. Navigate to the reports screen
2. Identify or build a report – Keep in mind that the report format and headers must match one of the standard data import templates. See the Schema section for additional information.



Field	Description	Sort	VG	A	Function	Format
Corporate Name	SupplierName	□	□	□	...	...
Supplier Number	SupplierNumber	□	□	□	...	...
Expiration Date	NewDate	□	□	□	...	4/20/2016
...		□	□	□	...	...

A. **Field**: Choose from available options in the database
B. **Description**: The column header on the report which must be a valid field name and not a display name.
    a. For instance in the screenshot above, Corporate Name is the display name and SupplierName is the actual field name. This is important because the import would fail if the column header is not a valid field name.
C. **OPTIONAL - Advanced Field Settings** ⚙: Click on the Advanced Field Settings icon *if* further data manipulation is required.

a. Clicking the Advanced Field Settings icon brings up a modal that allows for data manipulation and calculations.

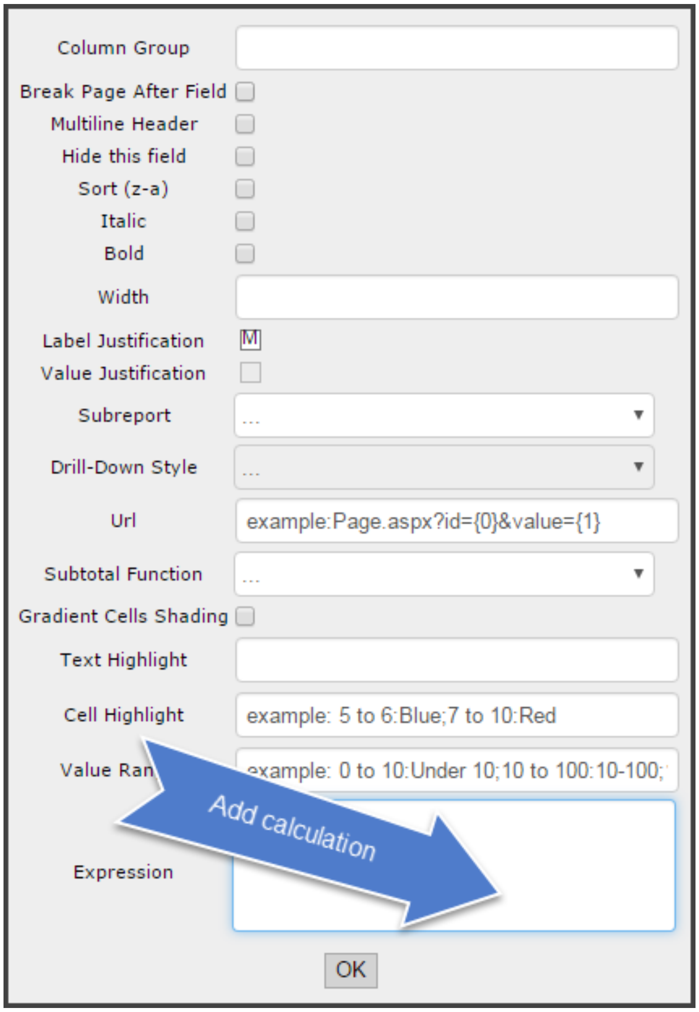
b. In the example above, the data in the 'Expiration Date' field could be used to calculate a new date which would be written to the 'NewDate' field.

   i. Click on the Advanced Field Settings icon ⚙ in the 'Expiration Date' row

   ii. Calculate the date you wish to be written to the 'NewDate' field by using the Expression box to add a calculation.

   iii. After clicking OK, if you have updated the settings the ⚙✓ Advanced Field Settings icon *may* have green check mark.

-
- Note: The checkmark does not always appear if you are working with expressions or label/value justification.



Column Group

Break Page After Field □

Multiline Header □

Hide this field □

Sort (z-a) □

Italic □

Bold □

Width

Label Justification M

Value Justification □

Subreport ...

Drill-Down Style ...

Url example:Page.aspx?id={0}&value={1}

Subtotal Function ...

Gradient Cells Shading □

Text Highlight

Cell Highlight example: 5 to 6:Blue;7 to 10:Red

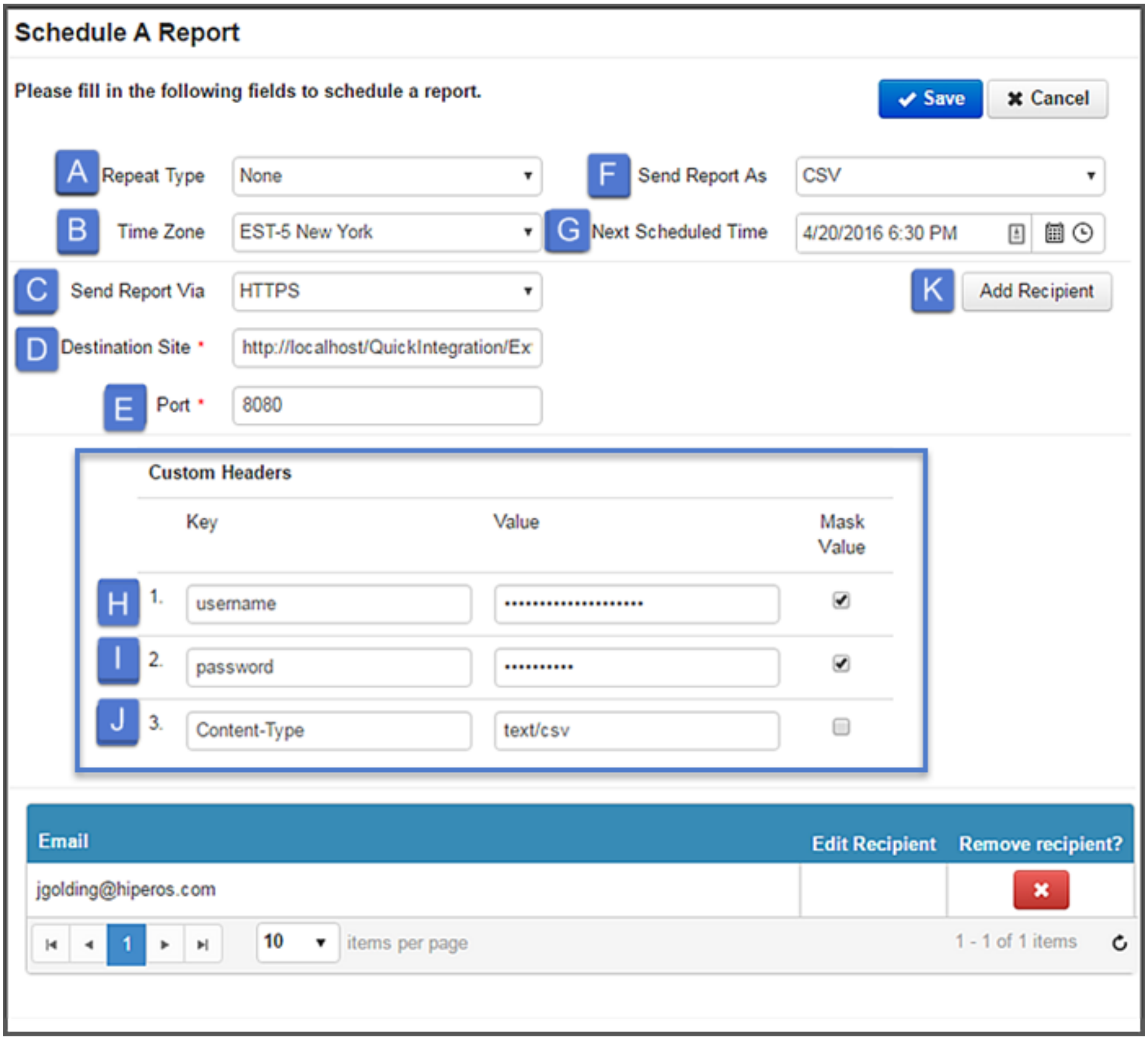
Value Ran example: 0 to 10:Under 10;10 to 100:10-100;

Expression

OK

Schedule the Report

1. Click on the Scheduler icon of the report



Schedule A Report

Please fill in the following fields to schedule a report.

Repeat Type	None ▾	Send Report As	CSV ▾
Time Zone	EST-5 New York ▾	Next Scheduled Time	4/20/2016 6:30 PM
Send Report Via	HTTPS ▾		
Destination Site *	http://localhost/QuickIntegration/Ex		
Port *	8080		

Custom Headers

	Key	Value	Mask Value
1.	username	••••••••••••••••••••	☑
2.	password	••••••••••	☑
3.	Content-Type	text/csv	☐

Email	Edit Recipient	Remove recipient?
jgolding@hiperos.com		✖

1 | 10 ▾ items per page | 1 - 1 of 1 items

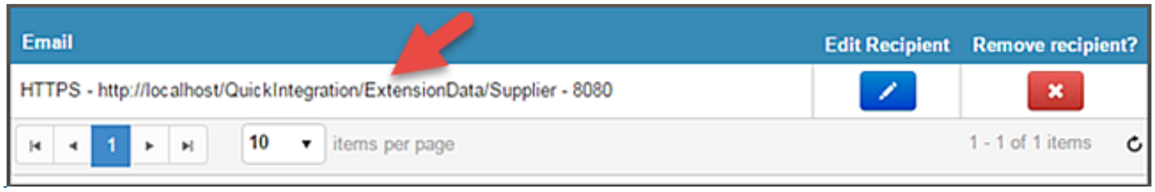
2. Using the following options, set the Report Scheduler to create the automatic export/import of data
   A. Repeat Type: choose the frequency
   B. Time Zone: choose correct time zone
   C. Send Report Via: HTTPS
   D. Destination Site: http://localhost/QuickIntegration/{ObjectType} – For a full list of object types available for import, please see Appendix A.
   E. Port:

        i. Test = 8080
        ii. Production = 8095

    F. Send Report As: choose either CSV or Excel (CSV and Excel files are the only formats that Risk Assess will recognize when importing data)
    G. Next Scheduled Time: Choose a time at least one hour later the current time
    H. Custom Header 1
        iii. **Key**: username
        iv. **Value**: <Risk Assess user in QuickIntegration group>
    I. Custom Header 2
        v. **Key**: password
        vi. **Value**: <user password>
    J. Custom Header 3
        vii. **Key**: Content-Type
        viii. **Value**:
            1. (for CSV file): text/csv
            2. (for Excel file): application/vnd.ms-excel

- **Please note**: Make sure to include all three custom headers as noted above (H-J).

3. Click the Add Recipient button (item K above). Ensure that you can see the HTTPS recipient at the bottom of the scheduler screen before saving the schedule.



Email	Edit Recipient	Remove recipient?
HTTPS - http://localhost/QuickIntegration/ExtensionData/Supplier - 8080		

10 items per page — 1 - 1 of 1 items

4. The report will run on the schedule you defined and will post the result file back to Risk Assess.
5. Any issues that may have been encountered will be emailed to the default site administrator of your tenant.

Quick Integration Troubleshooting Tips

Exporting Data from Risk Assess

If the data does not reach the 'receiving' system, the following steps may help narrow down the cause of the issue.

1. After configuring the scheduler to export the report using HTTPS or SFTP, also configure the report to be sent via email to a user account as an attachment.
    a. If no email is received after more than 30 minutes from the scheduled time, there may be a misconfiguration in the scheduler time. Check the time zone settings.
    b. If the email is received, but HTTPS/SFTP scheduled task appears to have failed, check the credentials and configurations for HTTPS/SFTP and validate they are correct.
2. Try to manually connect to the remote server using Filezilla or similar tool to see if you can access the directory and manually load a file.
3. Check to ensure that the security setting at the receiving end will allow access. Many networks will support whitelisted IP access only.
    a. Production IP Addresses:
        - 74.205.24.151
    b. Test IP Addresses:
        - 74.205.24.145
    c. SMTP Whitelisting
        - Mail.hiperos.com – 74.205.24.154
        - Mailtest.hiperos.com - 74.205.24.145

    Please note: The above IP Addresses have been provided by the support team. Please check with them for any updates or changes to this list.

4. If none of the above options work, submit a support ticket to the Coupa Support team.

Importing Data into Risk Assess

If the data is not importing correctly into Risk Assess, the following steps may help narrow down the cause of the issue.

1. Manually import the file using the standard import procedures via the 'Admin' tab to validate the headers are correct and all the required fields contain data. If there are issues importing the file, cross check the format against the acceptable formats by going to: https://rportaltest.hiperos.com/QuickIntegration.
2. In the report scheduler, add an email recipient to the list for the same file being integrated back into Risk Assess (for troubleshooting round trip functionality).
3. Test importing the file back into Risk Assess from a browser
    a. Open a tab in Chrome Browser to

    chrome-extension://aejoelaoggembcahagimdiliamlcdmfm/dhc.html (Note: you may need to add the DHC by Restlet app to by going to the chrome web store)

A. Select HTTPS

B. Add the correct URL – if testing Quick Integration Round Trip, the URL would be QuickIntegration/{ObjectType}

C. Select POST

D. To add Headers, choose form from the drop down list

E. Click the plus sign to add the following headers

   i. username <Risk Assess user in QuickIntegration group>

   ii. password <user password>

   iii. Content-Type

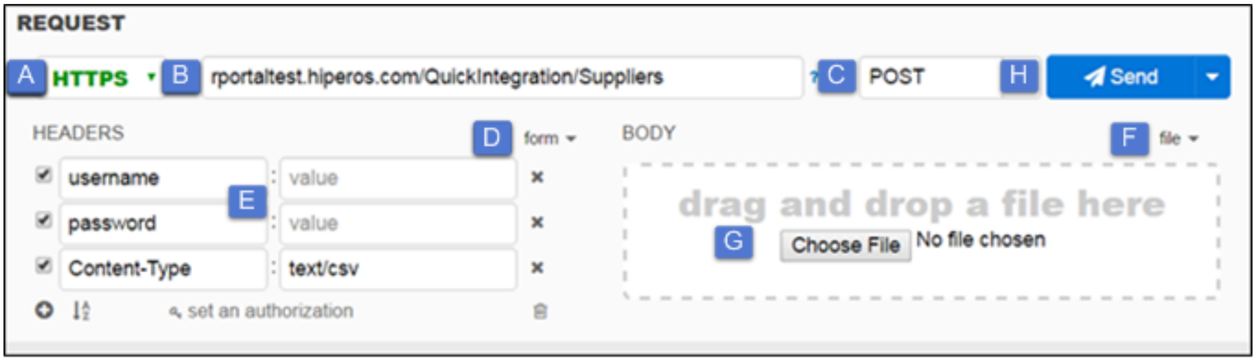
      1. Value (for CSV file): text/csv

      2. Value (for Excel file): application/vnd.ms-excel

F. Choose file from the drop down list

G. Add file in the 'BODY'

H. Click on the 'Send' button



rportaltest.hiperos.com/QuickIntegration/Suppliers

POST

username	value
password	value
Content-Type	text/csv

Messages will appear below the setup area

4. Data can also be posted using a command line URL

CORRECT:

HTTPSTATUS=$(curl -v -X POST -w "%{http_code}" -H 'username:myusername@hiperos.com' -H 'password:<password>' -d @filename.csv -i https://rportaltest.hiperos.com/QuickIntegration/ExtendedSuppliers -o responsefile.txt)

INCORRECT:

HTTPSTATUS=$(curl -v -X POST -w "%{http_code}" -u username@hiperos.com:password -d @filename.csv -i https://rportaltest.hiperos.com/QuickIntegration/ExtendedSuppliers -o responsefile.txt)

5. If the file is valid and if the file posts, then the problem is with one of the configurations:

   a. Is the user a member of the QuickIntegration group?

   b. Is QuickIntegration enabled for the tenant?

   c. Is the username, password and content type correct in the report scheduler?

   d. Is the URL and port correct for the HTTP recipient?

   e. Is the time zone correct in the scheduler?

Frequently Asked Questions (FAQ)

How many records can be sent using QI?

It is the overall size of the file rather than the number of records that impacts the process. A file that is too large may not load due to a connection timeout. While the estimated max number of records is somewhere in the 5,000 – 10,000 range, each client should test with actual volumes and files to ensure that imports will not timeout after go-live.

Can I send a file using SFTP?

If you are using Quick Integration Outbound, you can send a file via SFTP. For Quick Integration Inbound or Roundtrip, you must send files via HTTPS, as Risk Assess does not have an SFTP server.

If I have a bad record in my QI data file, will the entire import fail?

Yes. Unlike a manual import where the bad record is ignored, QI will fail the entire file on import.

I doubled checked and everything is set up correctly, but I am still getting a 302 error?

A 302 error means something failed on the server and the message is not always propagated up to the application. Please check the following:

- Is the user internal?
- Is the user a member of the QuickIntegration group?
- Is Quick Integration enabled for the tenant?

My data is in an excel format, can I send it?

Yes, but you need to specify the content type. Add another parameter to the post for 'Content-Type' and in the next box, set the type.

Use the listing below to determine what type you may need:

.csv	text/csv
.xls	application/vnd.ms-excel
.xlt	application/vnd.ms-excel
.xla	application/vnd.ms-excel
.xlsx	application/vnd.openxmlformats-officedocument.spreadsheetml.sheet

Appendix A – Import Options

Risk Assess Quick Integration makes the same import functionality exposed via the Risk Assess web site accessible via an HTTP POST. This allows for automated processes to upload files either ad hoc or on a reoccurring schedule without user intervention. Supported file formats include all versions of Excel and comma separated value (CSV) flat files. The list below provides the various import options. For additional information please visit https://rportaltest.hiperos.com/QuickIntegration.

- **QuickIntegration/Users**

  Imports user data into the Risk Assess system

- **QuickIntegration/Suppliers**

  Imports supplier data into the Risk Assess system

- **QuickIntegration/Relationships**

  Imports relationship data into the Risk Assess system

- **QuickIntegration/ExtendedSuppliers**

  Imports extended supplier data into the Risk Assess system

- **QuickIntegration/ExtensionData/{ObjectType}**

  Imports extension data into the Risk Assess system. The following object types are supported:

  - supplier
  - address
  - user
  - relationship
  - engagement
  - product
  - organizationunit
  - supplierproduct

- **QuickIntegration/UserDefinedObjects/{ParentObjectType}/{UDOType}**

  Imports user-defined objects into the Risk Assess system. The following parent object types are supported:

  - supplier
  - relationship
  - supplierproduct

- **QuickIntegration/Engagements**

  Imports engagements into the Risk Assess system

- **QuickIntegration/Organizations**

  Imports organizations into the Risk Assess system

- **QuickIntegration/Products**

  Imports products via HTTPS into the Risk Assess system

- **QuickIntegration/ProgramEnrollments/{ProgramType}**

  Imports program enrollments into the Risk Assess system. The following program types are supported:

  - compliance
  - performance
  - risk
  - info management
  - robo

  Note: This import also requires the program name to be supplied in the HTTP request header

- **QuickIntegration/SetupData**

  Imports setup data into the Risk Assess system

- **QuickIntegration/SupplierDiversities**

  Imports supplier diversities into the Risk Assess system

- **QuickIntegration/SupplierLocations**

  Imports supplier locations into the Risk Assess system

- **QuickIntegration/SupplierMarketIntelligence**

  Imports supplier market intelligence into the Risk Assess system

- **QuickIntegration/SupplierPrincipals**

  Imports supplier principals into the Risk Assess system

- **QuickIntegration/SupplierProducts**

  Imports supplier products into the Risk Assess system

- **QuickIntegration/UserGroups**

Imports user groups into the Risk Assess system

Appendix B - Code Samples

Supplier Import

The following code allows for import of Supplier information. Using this method, new suppliers can be imported and existing suppliers in Risk Assess can be updated. NOTE: this supplier import procedure is equivalent to using the Supplier Import functionality in the Risk Assess Admin Interface. As such, it is important to use the appropriate file format, as described below.

Java

```
package httpPostExample;

import java.io.File;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.HttpMultipartMode;
import org.apache.http.entity.mime.MultipartEntityBuilder;
import org.apache.http.entity.mime.content.FileBody;
import org.apache.http.impl.client.HttpClientBuilder;
import org.apache.http.protocol.BasicHttpContext;
import org.apache.http.protocol.HttpContext;
import org.apache.http.util.EntityUtils;

public class uploadFile
{
    public static void main(String[] args)
    {
        try
        {
            String url = "https://rportal.hiperos.com/QuickIntegration/Suppliers";
            String filePath = "c:\\hiperos\\outbox\\suppliers_import.xlsx";

            HttpClient httpClient = HttpClientBuilder.create().build();
            HttpContext httpContext = new BasicHttpContext();

            HttpPost httpPost = new HttpPost(url);
            httpPost.setHeader("username", "integration@client.com ");
            httpPost.setHeader("password", "xxxyyy");

            MultipartEntityBuilder builder = MultipartEntityBuilder.create();
            builder.setMode(HttpMultipartMode.BROWSER_COMPATIBLE);

            final File file = new File(filePath);
            FileBody fb = new FileBody(file);

            builder.addPart("file", fb);

            final HttpEntity entity = builder.build();
            httpPost.setEntity(entity);

            HttpResponse postResponse = httpClient.execute(httpPost, httpContext);

            HttpEntity responseEntity = postResponse.getEntity();
            System.out.println(EntityUtils.toString(responseEntity));
        }
        catch(Exception ex)
        {
            ex.printStackTrace();
        }
    }
}
```

C#

```
using System;
using System.Net;

namespace QuickIntegration_Sample
{
    class Program
    {
        static void Main()
        {
            const string uriString = "https://rportal.hiperos.com/QuickIntegration/Suppliers"
;
            const string fileName = @"c:\\hiperos\\outbox\\suppliers_import.xlsx";

            // Create a new WebClient instance.
            WebClient client = new WebClient();

            // Add authentication headers to request.
            client.Headers.Add("username", "integration@client.com");
            client.Headers.Add("password", "xxxyyy");

            // Upload the file to the URI.
            // The 'UploadFile(uriString,fileName)' method implicitly uses HTTP POST method.
            byte[] responseArray = client.UploadFile(uriString, fileName);

            string response = System.Text.Encoding.ASCII.GetString(responseArray);

            // Decode and display the response.
            Console.WriteLine("\nResponse Received.  The contents of the file uploaded are:\n
{0}", response);
```

```
                }
        }
}
```

Extended Supplier Import

The following code allows for import of Supplier information with User Defined Fields (UDF) that reside on the supplier record.

Java

```
package httpPostExample;

import java.io.File;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.HttpMultipartMode;
import org.apache.http.entity.mime.MultipartEntityBuilder;
import org.apache.http.entity.mime.content.FileBody;
import org.apache.http.impl.client.HttpClientBuilder;
import org.apache.http.protocol.BasicHttpContext;
import org.apache.http.protocol.HttpContext;
import org.apache.http.util.EntityUtils;

public class uploadFile
{
    public static void main(String[] args)
    {
        try
        {
            String url = "https://rportal.hiperos.com/QuickIntegration/ExtendedSuppliers";
            String filePath = "c:\\hiperos\\outbox\\ExtendedSuppliers_import.xlsx";

            HttpClient httpClient = HttpClientBuilder.create().build();
            HttpContext httpContext = new BasicHttpContext();

            HttpPost httpPost = new HttpPost(url);
            httpPost.setHeader("username", "integration@client.com ");
            httpPost.setHeader("password", "xxxyyy");

            MultipartEntityBuilder builder = MultipartEntityBuilder.create();
            builder.setMode(HttpMultipartMode.BROWSER_COMPATIBLE);

            final File file = new File(filePath);
            FileBody fb = new FileBody(file);

            builder.addPart("file", fb);

            final HttpEntity entity = builder.build();
            httpPost.setEntity(entity);
            HttpResponse postResponse = httpClient.execute(httpPost, httpContext);

            HttpEntity responseEntity = postResponse.getEntity();
            System.out.println(EntityUtils.toString(responseEntity));
        }
        catch(Exception ex)
        {
            ex.printStackTrace();
        }
    }
}
```

C#

```
using System;
using System.Net;

namespace QuickIntegration_Sample
{
    class Program
    {
        static void Main()
        {
            const string uriString = "https://rportal.hiperos.com/QuickIntegration/ExtendedSu
ppliers";
            const string fileName = @"c:\\hiperos\\outbox\\ExtendedSuppliers_import.xlsx";

            // Create a new WebClient instance.
            WebClient client = new WebClient();

            // Add authentication headers to request.
            client.Headers.Add("username", "integration@client.com");
            client.Headers.Add("password", "xxxyyy");

            // Upload the file to the URI.
            // The 'UploadFile(uriString,fileName)' method implicitly uses HTTP POST method.
            byte[] responseArray = client.UploadFile(uriString, fileName);

            string response = System.Text.Encoding.ASCII.GetString(responseArray);

            // Decode and display the response.
            Console.WriteLine("\nResponse Received.  The contents of the file uploaded are:\n
{0}", response);
        }
    }
}
```

User Import

Java

```java
package httpPostExample;

import java.io.File;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.HttpMultipartMode;
import org.apache.http.entity.mime.MultipartEntityBuilder;
import org.apache.http.entity.mime.content.FileBody;
import org.apache.http.impl.client.HttpClientBuilder;
import org.apache.http.protocol.BasicHttpContext;
import org.apache.http.protocol.HttpContext;
import org.apache.http.util.EntityUtils;

public class uploadFile
{
    public static void main(String[] args)
    {
        try
        {
            String url = "https://rportal.hiperos.com/QuickIntegration/users";
            String filePath = "c:\\hiperos\\outbox\\users_import.xlsx";

            HttpClient httpClient = HttpClientBuilder.create().build();
            HttpContext httpContext = new BasicHttpContext();

            HttpPost httpPost = new HttpPost(url);
            httpPost.setHeader("username", "integration@client.com ");
            httpPost.setHeader("password", "xxxyyy");

            MultipartEntityBuilder builder = MultipartEntityBuilder.create();
            builder.setMode(HttpMultipartMode.BROWSER_COMPATIBLE);

            final File file = new File(filePath);
            FileBody fb = new FileBody(file);

            builder.addPart("file", fb);

            final HttpEntity entity = builder.build();
            httpPost.setEntity(entity);

            HttpResponse postResponse = httpClient.execute(httpPost, httpContext);

            HttpEntity responseEntity = postResponse.getEntity();
            System.out.println(EntityUtils.toString(responseEntity));
        }
        catch(Exception ex)
        {
            ex.printStackTrace();
        }
    }
}
```

C#

```
using System;
using System.Net;

namespace QuickIntegration_Sample
{
    class Program
    {
        static void Main()
        {
            const string uriString = "https://rportal.hiperos.com/QuickIntegration/users";
            const string fileName = @"c:\\hiperos\\outbox\\users_import.xlsx";

            // Create a new WebClient instance.
            WebClient client = new WebClient();

            // Add authentication headers to request.
           client.Headers.Add("username", "integration@client.com");
           client.Headers.Add("password", "xxxyyy");

            // Upload the file to the URI.
            // The 'UploadFile(uriString,fileName)' method implicitly uses HTTP POST method.
            byte[] responseArray = client.UploadFile(uriString, fileName);

            string response = System.Text.Encoding.ASCII.GetString(responseArray);

            // Decode and display the response.
            Console.WriteLine("\nResponse Received.  The contents of the file uploaded are:\n{0}", response);
        }
    }
}
```

Relationship Import

Java

```
package httpPostExample;

import java.io.File;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.HttpMultipartMode;
import org.apache.http.entity.mime.MultipartEntityBuilder;
import org.apache.http.entity.mime.content.FileBody;
import org.apache.http.impl.client.HttpClientBuilder;
import org.apache.http.protocol.BasicHttpContext;
import org.apache.http.protocol.HttpContext;
import org.apache.http.util.EntityUtils;

public class uploadFile
{
    public static void main(String[] args)
    {
        try
        {
            String url = "https://rportal.hiperos.com/QuickIntegration/Relationship
s";
            String filePath = "c:\\hiperos\\outbox\\Relationships_import.xlsx";

            HttpClient httpClient = HttpClientBuilder.create().build();
            HttpContext httpContext = new BasicHttpContext();

            HttpPost httpPost = new HttpPost(url);
            httpPost.setHeader("username", "integration@client.com ");
            httpPost.setHeader("password", "xxxyyy");

            MultipartEntityBuilder builder = MultipartEntityBuilder.create();
            builder.setMode(HttpMultipartMode.BROWSER_COMPATIBLE);

            final File file = new File(filePath);
            FileBody fb = new FileBody(file);

            builder.addPart("file", fb);

            final HttpEntity entity = builder.build();
            httpPost.setEntity(entity);
            HttpResponse postResponse = httpClient.execute(httpPost, httpContext);

            HttpEntity responseEntity = postResponse.getEntity();
            System.out.println(EntityUtils.toString(responseEntity));
        }
        catch(Exception ex)
        {
            ex.printStackTrace();
        }
    }
}
```

C#

```
using System;
using System.Net;

namespace QuickIntegration_Sample
{
    class Program
    {
        static void Main()
        {
            const string uriString = "https://rportal.hiperos.com/QuickIntegration/
Relationships";
            const string fileName = @"c:\\hiperos\\outbox\\Relationships_import.xls
x";

            // Create a new WebClient instance.
            WebClient client = new WebClient();

            // Add authentication headers to request.
            client.Headers.Add("username", "integration@client.com");
            client.Headers.Add("password", "xxxyyy");

            // Upload the file to the URI.
            // The 'UploadFile(uriString,fileName)' method implicitly uses HTTP POS
T method.
            byte[] responseArray = client.UploadFile(uriString, fileName);

            string response = System.Text.Encoding.ASCII.GetString(responseArray);

            // Decode and display the response.
            Console.WriteLine("\nResponse Received.  The contents of the file uploa
ded are:\n{0}", response);
        }
        }
}
```

User Defined Field (UDF) Import

User Defined Fields can be imported onto the various objects:

- supplier
- address
- user
- relationship
- engagement
- product
- organizationunit
- supplierproduct

Depending on the Object that data is being imported to, ensure the proper unique identifier columns are included on the template. For more information, see:
https://rportaltest.hiperos.com/QuickIntegration/ExtensionData
In addition, you must ensure that the appropriate object is referenced to in the code below, (highlighted in yellow).

Java

```
package httpPostExample;

import java.io.File;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.HttpMultipartMode;
import org.apache.http.entity.mime.MultipartEntityBuilder;
import org.apache.http.entity.mime.content.FileBody;
import org.apache.http.impl.client.HttpClientBuilder;
import org.apache.http.protocol.BasicHttpContext;
import org.apache.http.protocol.HttpContext;
import org.apache.http.util.EntityUtils;

public class uploadFile
{
    public static void main(String[] args)
    {
        try
        {
            String url = "https://rportal.hiperos.com/QuickIntegration/ExtensionDat
a/supplier";
                  // replace /supplier with the appropriate object.
            String filePath = "c:\\hiperos\\outbox\\ExtensionData\\ import.xlsx";

            HttpClient httpClient = HttpClientBuilder.create().build();
            HttpContext httpContext = new BasicHttpContext();

            HttpPost httpPost = new HttpPost(url);
            httpPost.setHeader("username", "integration@client.com ");
            httpPost.setHeader("password", "xxxyyy");

            MultipartEntityBuilder builder = MultipartEntityBuilder.create();
            builder.setMode(HttpMultipartMode.BROWSER_COMPATIBLE);

            final File file = new File(filePath);
            FileBody fb = new FileBody(file);

            builder.addPart("file", fb);

            final HttpEntity entity = builder.build();
            httpPost.setEntity(entity);

            HttpResponse postResponse = httpClient.execute(httpPost, httpContext);

            HttpEntity responseEntity = postResponse.getEntity();
            System.out.println(EntityUtils.toString(responseEntity));
        }
        catch(Exception ex)
        {
            ex.printStackTrace();
        }
    }
}
```

C#

```
using System;
using System.Net;

namespace QuickIntegration_Sample
{
    class Program
    {
        static void Main()
        {
            const string uriString = "https://rportal.hiperos.com/QuickIntegration/
ExtensionData/supplier";
            const string fileName = @"c:\\hiperos\\outbox\\ExtensionData\\import.xl
sx";

            // Create a new WebClient instance.
            WebClient client = new WebClient();

            // Add authentication headers to request.
            client.Headers.Add("username", "integration@client.com");
            client.Headers.Add("password", "xxxyyy");

            // Upload the file to the URI.
            // The 'UploadFile(uriString,fileName)' method implicitly uses HTTP POS
T method.
            byte[] responseArray = client.UploadFile(uriString, fileName);

            string response = System.Text.Encoding.ASCII.GetString(responseArray);

            // Decode and display the response.
            Console.WriteLine("\nResponse Received.  The contents of the file uploa
ded are:\n{0}", response);
        }
    }
}
```

User Defined Object (UDO) Import

User Defined Object fields can be imported onto the following objects:

- supplier
- relationship
- supplierproduct

Depending on the Object that data is being imported to, ensure the proper unique identifier columns are included on the template. For more information, see:
https://rportaltest.hiperos.com/QuickIntegration/ExtensionData
In addition, you must ensure that the appropriate object is referenced to in the code below (highlighted in yellow).

Java

```
package httpPostExample;

import java.io.File;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.HttpMultipartMode;
import org.apache.http.entity.mime.MultipartEntityBuilder;
import org.apache.http.entity.mime.content.FileBody;
import org.apache.http.impl.client.HttpClientBuilder;
import org.apache.http.protocol.BasicHttpContext;
import org.apache.http.protocol.HttpContext;
import org.apache.http.util.EntityUtils;

public class uploadFile
{
    public static void main(String[] args)
    {
        try
        {
            String url = "https://rportal.hiperos.com/QuickIntegration/UserDefinedO
bjects/supplier/myudo";
            String filePath = "c:\\hiperos\\outbox\\UserDefinedObjects\\supplier\\m
yudo_import.xlsx";

            HttpClient httpClient = HttpClientBuilder.create().build();
            HttpContext httpContext = new BasicHttpContext();

            HttpPost httpPost = new HttpPost(url);
            httpPost.setHeader("username", "integration@client.com ");
            httpPost.setHeader("password", "xxxyyy");

            MultipartEntityBuilder builder = MultipartEntityBuilder.create();
            builder.setMode(HttpMultipartMode.BROWSER_COMPATIBLE);

            final File file = new File(filePath);
            FileBody fb = new FileBody(file);

            builder.addPart("file", fb);

            final HttpEntity entity = builder.build();
            httpPost.setEntity(entity);

            HttpResponse postResponse = httpClient.execute(httpPost, httpContext);

            HttpEntity responseEntity = postResponse.getEntity();
            System.out.println(EntityUtils.toString(responseEntity));
        }
        catch(Exception ex)
        {
            ex.printStackTrace();
        }
    }
}
```

C#

```
using System;
using System.Net;

namespace QuickIntegration_Sample
{
    class Program
    {
        static void Main()
        {
            const string uriString = "https://rportal.hiperos.com/QuickIntegration/
UserDefinedObjects/supplier/myudo";
            const string fileName = @"c:\\hiperos\\outbox\\UserDefinedObjects\\supp
lier\\myudo_import.xlsx";

            // Create a new WebClient instance.
            WebClient client = new WebClient();

            // Add authentication headers to request.
            client.Headers.Add("username", "integration@client.com");
            client.Headers.Add("password", "xxxyyy");

            // Upload the file to the URI.
            // The 'UploadFile(uriString,fileName)' method implicitly uses HTTP POS
T method.
            byte[] responseArray = client.UploadFile(uriString, fileName);

            string response = System.Text.Encoding.ASCII.GetString(responseArray);

            // Decode and display the response.
            Console.WriteLine("\nResponse Received.  The contents of the file uploa
ded are:\n{0}", response);
        }
    }
}
```

Additional Import Instructions and Code Samples

For additional import procedure instructions and Code samples, see the following site and choose the object you wish to integrate with:

https://rportaltest.hiperos.com/QuickIntegration

Additional Import procedures include the following objects:

- Engagements
- Organizations
- Products
- Program Enrollments
- Setup Data
- Supplier Diversity
- Supplier Location
- Supplier Market Intelligence
- Supplier Principals
- Supplier Products
- User Groups