REPSLY WEB API

Web interface for exchanging data with Repsly

Repsly WEB API

Web interface for exchanging data with Repsly

INTRODUCTION

This document is provided as a technical manual for developers and system integrators. The API is most suitable for ERP/CRM integration as it's designed to move clients and products from ERP/CRM to Repsly and clients, products, forms and other documents into ERP/CRM.

With this data exchange there are no more frontiers and disconnected work. Data is maintained in the back-office and propagated to mobile workers, and work results automatically enter ERP/CRM, so two systems act as one.

AUTHENTICATION

Repsly API uses Basic authentication supported by SSL encryption to keep data transfer safe. The client sends the key and passcode as unencrypted base64 encoded text but it is used with HTTPS to prevent data spoofing.

API credentials differ from your login username and password. You can find your API username and password on Repsly settings page under API Settings.

HOW TO GET STARTED

You will need to write your own application which will communicate with both your system and Repsly API in any manner that best suits your needs. For communicating with the API, you will need to create a HTTP GET request to retrieve data or a HTTP POST request to send data to Repsly. How you process the data in your own system is up to you.

A full list of request and response messages is available in both JSON and XML formats on the API reference page located at https://api.repsly.com/v3/help.

ENDPOINTS

API supports both JSON and XML formats for requests and responses. JSON format is set as default.

To send a JSON request, header **Content-Type** should be set to "application/json". To send an XML request, header **Content-Type** should be set to "application/xml".

Also, you can choose between JSON and XML response formats by sending the appropriate **Accept** key in the header request with value "application/json" or "application/xml", respectively.

Endpoints support updating records (POST method) or getting records (GET method).

Import

Import is designed so that you can insert or update entities one by one. Batch import is not supported.

Any import into Repsly works in a way that you POST data, and Repsly decides whether to insert or update it. Key for deciding which records to update or insert is colored red in field definition tables.

Client

Endpoint is used to import existing clients from your system to Repsly.

> *Method: POST*
>
> *Path: https://api.repsly.com/v3/import/client*
>
> *Accept: application/json, application/xml*
>
> *Content-Type: application/json, application/xml*

Request definition		
Name	**Type**	**Description**
Code (*)	String (50)	Unique client code.
Name (*)	String (255)	Client name
Active	Boolean	Active client
Tag	String	Tags associated with this client (separated with commas)
Territory	Each territory in hierarchy: String (80)	Full territory path this client belongs to. The delimiter between territory levels is "**>**" (meaning that a territory in a two-level hierarchy is given as "**ParentTerritory>ChildTerritory**") **Important**: The full path must always be given in import data (and will be returned by the export). If a non-existent territory is assigned to a client in the import data, the existing territory hierarchy will be expanded to accommodate the new territory.
RepresentativeCode	String (20)	Unique representative code (assigned to this client)
RepresentativeName	String (80)	Representative name (assigned to this client)

StreetAddress	String (255)	Client street address
ZIP	String (20)	Client ZIP code
ZIPExt	String (20)	Client ZIP extension
City	String (255)	Client city
State	String (255)	Client state
Country	String (255)	Client country
Email	String (255)	Client email address
Phone	String (128)	Client phone number
Mobile	String (128)	Client mobile phone number
Website	String (255)	Client website
ContactName	String (255)	Contact person name
ContactTitle	String (50)	Contact person title
Note	String (255)	Additional client note
Status	Text	Client status name in sales cycle
CustomFields\Item\Field	String (255)	Custom field name
CustomFields\Item\Value	Text	Custom field value
PriceLists\Name	Text	List of price list names connected to this client. *Note: null and empty list doesn't have the same meaning. In case request has null value on this parameter system will ignore that parameter but in case it has empty list as parameter, system will break all connections between client and price lists.*

(*) – Required

Response field definition:

Response definition		
Name	**Type**	**Description**
Code	Integer	Error identifier (0 = no errors)

Message	String (255)	Response affirmative or error message

For a successful HTTP request (code 200), you will always receive a status response with Code equal to 0 (no error) and Message = "Client import job started asynchronously under ID: {ImportJobID}", where {ImportJobID} is the key you can use to check the current status of the import. To check the status of the import job you can use Import job status endpoint.

In case of a HTTP bad request (code 400), you will also receive a status response with Code not equal to 0 and a descriptive message. For incorrect format or other unhandled errors you will receive an HTML Request error page.

Product

Endpoint is used to import existing products from your system to Repsly.

> *Method: POST*
>
> *Path: https://api.repsly.com/v3/import/product*
>
> *Accept: application/json, application/xml*
>
> *Content-Type: application/json, application/xml*

Request definition		
Name	**Type**	**Description**
Code (*)	String (20)	Unique product code
Name (*)	String (80)	Product name
ProductGroupCode	String (20)	Unique product group code
ProductGroupName	String (80)	Product group name
Active	Boolean	Marks a product active or inactive
Tag	Text	Tags associated with this product (separated with commas)
UnitPrice (*)	Decimal (18,4)	Product price
EAN	String (20)	International Article Number (Barcode)
Note	String (1000)	Additional product note

ImageUrl	String	Input image url from which product image will be downloaded. **Must** be publicly accessible and **must** be direct link to image.
MasterProduct	String	Used for grouping purposes
Packaging\Codes\Value	string	List of package code values that match already existing packages inside the system. No new packages can be added through product import

(*) – Required

Response definition		
Name	**Type**	**Description**
Code	Integer	Error identifier (0 = no errors)
Message	String (255)	Response affirmative or error message

For a successful HTTP request (code 200), you will always receive a status response with Code equal to 0 (no error) and Message = "Client import job started asynchronously under ID: {ImportJobID}", where {ImportJobID} is the key you can use to check the current status of the import. To check the status of the import job you can use Import job status endpoint.

In case of a HTTP bad request (code 400), you will also receive a status response with Code not equal to 0 and an error message.

Package

Endpoint is used to import packages from your system into Repsly.

> *Method: POST*
> *Path: https://api.repsly.com/v3/import/package*
> *Accept: application/json, application/xml*
> *Content-Type: application/json, application/xml*

Request definition		
Name	**Type**	**Description**
Code (*)	String	Package code
Name (*)	String	Package name
Size (*)	Double	Package size
Active	Boolean	If package is active or inactive (Default is true)

(*) – Required

Response definition		
Name	**Type**	**Description**
Code	Integer	Error identifier (0 = no errors)
Message	String (255)	Response affirmative or error message

For a successful HTTP request (code 200), you will always receive a status response with Code equal to 0 (no error) and Message = "Package import job started asynchronously under ID: {ImportJobID}", where {ImportJobID} is the key you can use to check the current status of the import. To check the status of the import job you can use Import job status endpoint.

 In case of a HTTP bad request (code 400), you will also receive a status response with Code not equal to 0 and an error message.

Pricelist

Endpoint is used to import items from your system into pricelists that already exist in Repsly.

Request definition		
Name	**Type**	**Description**
PricelistName (*)	String	Pricelist name
ProductCode (*)	String	Product code
Price (*)	Double	Product price on pricelist
Active	Boolean	If pricelist item is active or inactive
DateAvailableFrom	Nullable<DateAvailableFrom>	Date after which the product is available on the pricelist. If null, the lower date limit is considered open.
DateAvailableTo	Nullable<DateAvailableTo>	Date until which the product is available on the pricelist. If null the upper date limit is considered open.
ClientID	String	Unique client ID
ManufactureID	String	Unique manufacture ID

(*) – Required

Response definition		
Name	**Type**	**Description**
Code	Integer	Error identifier (0 = no errors)
Message	String (255)	Response affirmative or error message

For a successful HTTP request (code 200), you will always receive a status response with Code equal to 0 (no error) and Message = "Client import job started asynchronously under ID: {ImportJobID}", where {ImportJobID} is the key you can use to check the current status of the import. To check the status of the import job you can use Import job status endpoint.

In case of a HTTP bad request (code 400), you will also receive a status response with Code not equal to 0 and an error message.

Schedule

Endpoint is used to import schedules from your system into Repsly.

> **i** *Method: POST*
> *Path: https://api.repsly.com/v3/import/schedule*
> *Accept: application/json, application/xml*
> *Content-Type: application/json, application/xml*

Request definition		
Name	**Type**	**Description**
PlaceID (*)	String	Place (client) code that will be assigned to this schedule
RepresentativeID (*)	String	Representative code that will be assigned to this schedule
DateBegin (*)	Date	Date when schedule begins
RepeatEveryWeeks (*)	Integer	Interval for weekly schedule recurrence. 0 for non recurring schedule.
VisitTime	Time	Time for scheduled visit, empty for any time visit
VisitDuration	Integer	Expected duration of scheduled visit in minutes
Tasks\TaskType	Array	List of scheduled tasks, for definition check table below.

(*) – Required

Tasks array definition		
Name	**Type**	**Description**
TaskType (*)	String	Type of task (Photo, Note, Form, Audit, Sales document)
TaskDescription (*)	String	Custom description for Photo or Note. Form, Audit and Sales document require their exact title as description.

(*) – Required

Response definition		
Name	**Type**	**Description**
Code	Integer	Error identifier (0 = no errors)
Message	String (255)	Response affirmative or error message

For a successful HTTP request (code 200), you will always receive a status response with Code equal to 0 (no error) and Message = "Package import job started asynchronously under ID: {ImportJobID}", where {ImportJobID} is the key you can use to check the current status of the import. To check the status of the import job you can use Import job status endpoint.

 In case of a HTTP bad request (code 400), you will also receive a status response with Code not equal to 0 and an error message.

Export

Export of entities returns a maximum of 50 records.

Every response contains metadata with additional info about fetched records. Metadata records are gray-colored rows in field definition tables.

Clients

Endpoint is used to export your clients from Repsly. System limit is 50 rows.

In every response you will receive meta variable LastTimeStamp of the last changed record in the list. Save it for future requests so you can use it as a parameter {lastTimeStamp} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of clients, start your request with parameter {lastTimeStamp} = 0.

> *Method: GET*
>
> *Path: https://api.repsly.com/v3/export/clients/{lastTimestamp}*
>
> *Accept: application/json, application/xml*
>
> *Content-Type: application/json, application/xml*

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of clients
MetaCollectionResult\LastTimeStamp	Long	Timestamp of the latest client in the list
ClientID	Integer	Unique client ID
TimeStamp	Long	Client timestamp
Code	String (50)	Unique client code
Name	String (255)	Client name
Active	Boolean	Marks a client active or inactive
Tag	String	Tags associated with this client (separated with commas)

Territory	Each territory in hierarchy: String (80)	The full territory hierarchy that this client is assigned to. The delimiter between territory levels is "**>**" (meaning that a territory in a two-level hierarchy is given as "**ParentTerritory>ChildTerritory**")
RepresentativeCode	String (20)	Unique representative code (assigned to this client)
RepresentativeName	String (80)	Representative name (assigned to this client)
StreetAddress	String (255)	Client street address
ZIP	String (20)	Client ZIP code
ZIPExt	String (20)	Client ZIP extension
City	String (255)	Client city
State	String (255)	Client state
Country	String (255)	Client country
Email	String (255)	Client email address
Phone	String (128)	Client phone number
Mobile	String (128)	Client mobile phone number
Website	String (255)	Client website
ContactName	String (255)	Contact person name
ContactTitle	String (50)	Contact person title
Note	String (255)	Additional client note
Status	Text	Client status name in sales cycle
CustomFields\Item\Field	String (255)	Custom field name
CustomFields\Item\Value	Text	Custom field value
PriceLists\Name	Text	List of price list names connected to this client
GPSLongitude	Text	Gps longitude (0 to +/-180)
GPSLatitude	Text	Gps latitude (0 to +/-90)

Client notes

Endpoint is used to export your client notes from Repsly. System limit is 50 rows.

In every response you will receive meta variables FirstID and LastID. Save LastID for future requests so you can use it as a parameter {lastClientNoteID} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of forms, start your request with parameter {lastClientNoteID} = 0.

> *Method: GET*
>
> *Path: https://api.repsly.com/v3/export/clientnotes/{lastClientNoteID}*
>
> *Accept: application/json, application/xml*
>
> *Content-Type: application/json, application/xml*

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of client notes
MetaCollectionResult\FirstID	Integer	Unique ID of the first item in the list
MetaCollectionResult\LastID	Integer	Unique ID of the last item in the list
ClientNoteID	Integer	Unique client note ID
TimeStamp	Long	Client note timestamp
DateAndTime	Datetime	Date and time of note entry
RepresentativeCode	String (20)	Unique representative code (note author)
RepresentativeName	String (80)	Representative name (note author)
ClientCode	String (50)	Unique client code
ClientName	String (255)	Client name
StreetAddress	String (255)	Client street address
ZIP	String (20)	Client ZIP code
ZIPExt	String (20)	Client ZIP extension
City	String (255)	Client city

State	String (255)	Client state
Country	String (255)	Client country
Email	String (255)	Client email address
Phone	String (128)	Client phone number
Mobile	String (128)	Client mobile phone number
Territory	String (80)	Territory name this client belongs to
Longitude	Long	Longitude
Latitude	Long	Latitude
Note	Text	Client note
VisitID	Integer	Unique Visit ID

Visits

Endpoint is used to export your visits from Repsly. System limit is 50 rows.

In every response you will receive meta variable LastTimeStamp of the last changed record in the list. Save it for future requests so you can use it as a parameter {lastTimeStamp} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of visits, start your request with parameter {lastTimeStamp} = 0.

Method: GET

Path: https://api.repsly.com/v3/export/visits/{lastTimestamp}

Accept: application/json, application/xml

Content-Type: application/json, application/xml

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of client visits
MetaCollectionResult\LastTimeStamp	Long	Timestamp of the latest visit in the list

VisitID	Integer	Unique visit ID
TimeStamp	Long	Visit timestamp
Date	Date	Date of visit
RepresentativeCode	String (20)	Unique representative code (visit author)
RepresentativeName	String (80)	Representative name (visit author)
ExplicitCheckIn	Boolean	Marks an explicit check-in at client (Representative has clicked a button in the mobile app)
DateAndTimeStart	Datetime	Date and time of visit start
DateAndTimeEnd	Datetime	Date and time of visit end
ClientCode	String (50)	Unique client code
ClientName	String (255)	Client name
StreetAddress	String (255)	Client street address
ZIP	String (20)	Client ZIP code
ZIPExt	String (20)	Client ZIP extension
City	String (255)	Client city
State	String (255)	Client state
Country	String (255)	Client country
Territory	String (80)	Territory name this client belongs to
LatitudeStart	Long	Latitude at visit start
LongitudeStart	Long	Longitude at visit start
LatitudeEnd	Long	Latitude at visit end
LongitudeEnd	Long	Longitude at visit end
PrecisionStart	Integer	Precision at visit start
PrecisionEnd	Integer	Precision at visit end
VisitStatusBySchedule	Integer	Status of current visit 1 - Scheduled

		2 - Unscheduled

Retail audits

Endpoint is used to export your retail audits from Repsly. System limit is 50 rows.

In every response you will receive meta variables FirstID and LastID. Save LastID for future requests so you can use it as a parameter {lastRetailAuditID} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of retail audits, start your request with parameter {lastRetailAuditID} = 0.

> **i** *Method: GET*
>
> *Path: https://api.repsly.com/v3/export/retailaudits/{lastRetailAuditID}*
>
> *Accept: application/json, application/xml*
>
> *Content-Type: application/json, application/xml*

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of retail audits
MetaCollectionResult\FirstID	Integer	Unique ID of the first item in the list
MetaCollectionResult\LastID	Integer	Unique ID of the last item in the list
RetailAuditID	Integer	Unique retail audit ID
RetailAuditName	String (50)	Name of the retail audit
Cancelled	Boolean	Retail audit is cancelled (Representative has cancelled it via mobile app)
ClientCode	String (50)	Unique client code
ClientName	String (255)	Client name
DateAndTime	Datetime	Date and time of creation of this retail audit
RepresentativeCode	String (20)	Unique representative code (retail audit author)
RepresentativeName	String (80)	Representative name (retail audit author)

Item\ProductGroupCode	String (20)	Item unique product group code
Item\ProductGroupName	String (80)	Item product group name
Item\ProductCode	String (20)	Item unique product code
Item\ProductName	String (80)	Item product name
Item\Present	Boolean	Item is present
Item\Price	Decimal (18, 4)	Item price
Item\Promotion	Boolean	Item is promoted
Item\ShelfShare	Decimal (18, 4)	Item shelf share
Item\ShelfSharePercent	Decimal (18, 4)	Item shelf share percent
Item\SoldOut	Boolean	Item is sold out
Item\Stock	Integer	Item stock
CustomFields\ExtraInfoItem\Field	String (255)	Custom field name
CustomFields\ExtraInfoItem\Value	String (255)	Custom field value
Note	String (255)	Note on the retail audit
VisitID	Integer	Unique Visit ID

Purchase orders

Endpoint is used to export your purchase orders from Repsly. System limit is 50 rows.

In every response you will receive meta variables FirstID and LastID. Save LastID for future requests so you can use it as a parameter {lastDocumentID} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of purchase orders, start your request with parameter {lastDocumentID} = 0.

Method: GET

Path: https://api.repsly.com/v3/export/purchaseorders/{lastDocumentID}

Accept: application/json, application/xml

Content-Type: application/json, application/xml

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of purchase orders
MetaCollectionResult\FirstID	Integer	Unique ID of the first item in the list
MetaCollectionResult\LastID	Integer	Unique ID of the last item in the list
PurchaseOrderID	Integer	Unique purchase order ID
TransactionType	String (50)	Transaction type (*PurchaseOrder*)
DocumentTypeID	Integer	Document type id
DocumentTypeName	String(100)	Document type name by custom definition
DocumentStatus	String(100)	Document status name by custom definition
DocumentStatusID	Nullable<Integer>	Document status ID, null means status is not set
DocumentItemAttributeCaption	String	Describes caption name of document type attributes (group name)
DateAndTime	Datetime	Date and time of creation of this purchase order
DocumentNo	String (50)	Document number
ClientCode	String (50)	Unique client code
ClientName	String (255)	Client name
DocumentDate	Date	Date of this purchase order
DueDate	Date	Due date of this purchase order
RepresentativeCode	String (20)	Unique representative code (author of this purchase order)
RepresentativeName	String (80)	Representative name (author of this purchase order)
Item\LineNo	Integer	Position of an item
Item\ProductCode	String (20)	Unique product code
Item\ProductName	String (80)	Product name

Item\UnitAmount	Decimal (18, 4)	Item unit amount
Item\UnitPrice	Decimal (18, 4)	Item unit price
Item\PackageTypeCode	String(45)	Packaging type code
Item\PackageTypeName	String(40)	Packaging type
Item\PackageTypeConversion	Integer	Packaging type conversion
Item\Quantity	Integer	Item quantity
Item\Amount	Decimal (18, 4)	Item amount
Item\DiscountAmount	Decimal (18, 4)	Discount amount
Item\DiscountPercent	Decimal (18, 4)	Discount percent
Item\TaxAmount	Decimal (18, 4)	Tax amount
Item\TaxPercent	Decimal (18, 4)	Tax percent
Item\TotalAmount	Decimal (18, 4)	Total amount
Item\Note	String (255)	Item note
Item\DocumentItemAttributeName	String(100)	Item custom attribute name - used for tagging items with specific custom defined attributes (eg. reason of return)
Item\DocumentItemAttributeID	Nullable<Integer>	Item custom attribute ID, null means no attribute is set to document.
SignatureURL	String (512)	Signature image URL
Note	String (255)	Note on the purchase order
Taxable	Boolean	Marks if this purchase order is taxable
VisitID	Integer	Unique Visit ID
StreetAddress	String (255)	Client street address
ZIP	String (20)	Client ZIP code
ZIPExt	String (20)	Client ZIP extension
City	String (255)	Client city
State	String (255)	Client state

Country	String (255)	Client country
CountryCode	String (20)	Client country code
DocumentCustomAttributes\CustomAttributeInfoID	GUID	Custom attribute unique identifier
DocumentCustomAttributes\Title	String (max)	Custom attribute question title
DocumentCustomAttributes\Type	String (50)	Type can be one of the following: Text, textarea, select, decimal, date, barcode, photo
DocumentCustomAttributes\Value	String (max)	Custom attribute question answer. For type photo this value contains GUID for photo filename on S3. By presigning this value you can access it via publicly available link. repsly.s3.amazonaws.com + value + .jpg

Document types

Endpoint is used to export your basic document types' data from Repsly. This serves with the document status update capability as a document's status can only be updated with a valid status (from the corresponding document type)

> **Method:** GET
> Path: https://api.repsly.com/v3/export/documentTypes/
> Accept: application/json, application/xml
> Content-Type: application/json, application/xml

Response definition		
Name	**Type**	**Description**
DocumentTypeID	Integer	Unique ID of the document type
DocumentTypeName	String	Document type name
Statuses	List<DocumentStatus>	List of document type statuses (described below)
Pricelists	List<DocumentTypePricelist>	List of pricelists available for this document type

DocumentStatus		
Name	**Type**	**Description**
DocumentStatusID	Integer	Unique ID of the document status
DocumentStatusName	String	Document status name

DocumentTypePricelist		
Name	**Type**	**Description**
PricelistID	Integer	Unique ID of the pricelist
PricelistName	String	Pricelist name

Pricelists

Endpoint is used to export basic pricelist data from Repsly.

Response definition		
Name	**Type**	**Description**
ID	Integer	Unique ID of the pricelist
Name	String	Pricelist name
IsDefault	Boolean	Determines if pricelist is default or not
Active	Boolean	Is pricelist active or inactive
UsePrices	Boolean	Does pricelist display product prices or not

Pricelists Items

Endpoint is used to export pricelist items' data from Repsly which belong to a specific pricelist.

Response definition		
Name	**Type**	**Description**
ID	Integer	Unique pricelist item ID
ProductID	Integer	Unique product ID
ProductCode	String	Product code
Price	Double	Product price
Active	Boolean	If product item is active or inactive
ClientID	String	Unique client ID
ManufactureID	String	Unique manufacture ID
DateAvailableFrom	Nullable<DateTime>	Date after which the product is available on this pricelist. If null, the lower date limit is considered open.
DateAvailableTo	Nullable<DateTime>	Date up to which the product is available on this pricelist. If null, the upper date limit is considered open.

Forms

Endpoint is used to export your forms from Repsly. System limit is 50 rows.

In every response you will receive meta variables FirstID and LastID. Save LastID for future requests so you can use it as a parameter {lastFormID} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of forms, start your request with parameter {lastFormID} = 0.

> **i** *Method: GET*
>
> *Path: https://api.repsly.com/v3/export/forms/{lastFormID}*
>
> *Accept: application/json, application/xml*
>
> *Content-Type: application/json, application/xml*

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of forms
MetaCollectionResult\FirstID	Integer	Unique ID of the first item in the list
MetaCollectionResult\LastID	Integer	Unique ID of the last item in the list
FormID	Integer	Unique form ID
FormName	String (255)	Name of the form
ClientCode	String (50)	Unique client code
ClientName	String (255)	Client name
DateAndTime	Datetime	Date and time of creation of this form
RepresentativeCode	String (20)	Unique representative code (author of this form)
RepresentativeName	String (80)	Representative name (author of this form)
StreetAddress	String (255)	Client street address

ZIP	String (20)	Client ZIP code
ZIPExt	String (20)	Client ZIP extension
City	String (255)	Client city
State	String (255)	Client state
Country	String (255)	Client country
Email	String (255)	Client email address
Phone	String (128)	Client phone number
Mobile	String (128)	Client mobile phone number
Territory	String (80)	Territory name this client belongs to
Longitude	Long	Longitude
Latitude	Long	Latitude
SignatureURL	String (512)	Signature image URL
VisitStart	Datetime	Date and time of visit start
VisitEnd	Datetime	Date and time of visit end
Item\Field	String (255)	Custom field name
Item\Value	Text	Custom field value
VisitID	Integer	Unique Visit ID

Photos

Endpoint is used to export your photos from Repsly. System limit is 50 rows.

In every response you will receive meta variables FirstID and LastID. Save LastID for future requests so you can use it as a parameter {lastPhotoID} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of photos, start your request with parameter {lastPhotoID} = 0.

Method: GET

Path: https://api.repsly.com/v3/export/photos/{lastPhotoID}

Accept: application/json, application/xml

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of photos
MetaCollectionResult\FirstID	Integer	Unique ID of the first item in the list
MetaCollectionResult\LastID	Integer	Unique ID of the last item in the list
PhotoID	Integer	Unique photo ID
ClientCode	String (50)	Unique client code
ClientName	String (255)	Client name
Note	String (1000)	Photo note
DateAndTime	Datetime	Date and time of the photo
PhotoURL	String (512)	Photo URL
RepresentativeCode	String (20)	Unique representative code (author of this photo)
RepresentativeName	String (80)	Representative name (author of this photo)
VisitID	Integer	Unique Visit ID
Tag	String	Tags associated with this client (separated with commas)

Daily working time

Endpoint is used to export your daily working time from Repsly. System limit is 50 rows.

In every response you will receive meta variables FirstID and LastID. Save LastID for future requests so you can use it as a parameter {lastDailyWorkingTimeID} for every subsequent request until the response meta variable TotalCount is equal to 0, which means you have received all updates.

To get the complete list of daily working times, start your request with parameter {lastDailyWorkingTimeID} = 0.

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of daily working times
MetaCollectionResult\FirstID	Integer	Unique ID of the first item in the list
MetaCollectionResult\LastID	Integer	Unique ID of the last item in the list
DailyWorkingTimeID	Integer	Unique daily working time ID
Date	Date	Date of entry
DateAndTimeStart	Datetime	Start of workday
DateAndTimeEnd	Datetime	End of workday
Length	Integer	Length of workday in minutes
MileageStart	Integer	Start mileage
MileageEnd	Integer	End mileage
MileageTotal	Integer	Total miles
LatitudeStart	Long	Latitude at day start
LongitudeStart	Long	Longitude at day start
LatitudeEnd	Long	Latitude at day end
LongitudeEnd	Long	Longitude at day end
RepresentativeCode	String (20)	Unique representative code (owner of this workday)
RepresentativeName	String (80)	Representative name (owner of this workday)
Note	String (255)	Workday note

Tag	Text	Tags associated with this working day (separated with commas)
NoOfVisits	Integer	Total number of visits in this workday
MinOfVisits	Datetime	Earliest date and time of a visit in this workday
MaxOfVisits	Datetime	Latest date and time of a visit in this workday
MinMaxVisitsTime	Integer	Length in minutes between the earliest and the latest visit
TimeAtClient	Integer	Length in minutes spent at client locations
TimeAtTravel	Integer	Length in minutes spent in travelling

Visit Schedules

Endpoint is used to export your visit schedules from Repsly.

Define start and end date to receive response with visit schedules occurred in wanted period.

> *Method: GET*
> *Path: https://api.repsly.com/v3/export/visitschedules/{beginDateTime}/{endDateTime}*
> *Accept: application/json, application/xml*
> *Content-Type: application/json, application/xml*

Response definition		
Name	**Type**	**Description**
MetaCollectionResult\TotalCount	Integer	Total count of visit schedules
ScheduleDateAndTime	Datetime	Date and time of visit
RepresentativeCode	String (20)	Unique representative code (visitor in this visit)
RepresentativeName	String (80)	Representative name (visitor in this visit)
ClientCode	String (50)	Unique client code
ClientName	String (255)	Client name
StreetAddress	String (255)	Client street address
ZIP	String (20)	Client ZIP code

ZIPExt	String (20)	Client ZIP extension
City	String (255)	Client city
State	String (255)	Client state
Country	String (255)	Client country
Territory	String (80)	Territory name this client belongs to
VisitNote	Text	Note on the visit

Representatives

Endpoint is used to export your representatives from Repsly.

This endpoint always returns full list of all representatives.

> **i** *Method: GET*
> *Path: https://api.repsly.com/v3/export/representatives*
> *Accept: application/json, application/xml*
> *Content-Type: application/json, application/xml*

Response definition		
Name	**Type**	**Description**
Code	String (20)	Unique representative code
Name	String (80)	Representative name
Note	String (255)	Additional note about representative
UserID	String (50)	Mobile credentials - username
Password	String (50)	Mobile credentials - password (plain text)
Email	String (256)	Representative's email address
Phone	String (128)	Representative's phone number
Mobile	String (128)	Representative's mobile phone number
Territories	Array(String)	The array of strings of territory paths that representative is associated with.

		Territory path is representation of hierarchical territory position as territory names separated by "/".
		Ex. "Territory 1/Territory 1.1/Territory 1.1.1"
Active	Boolean	Active representative

Import job status

Endpoint is used to export the status of any imports done via Repsly API. To use this endpoint you need to provide the importJobID, which is received when you import data through import endpoints. When data is imported into Repsly, following scenarios can occur:

New data

If the record being inserted doesn't exist in Repsly, then the record will be created. All newly created records are counted in the RowsInserted field.

Existing data

If the record being inserted already exists in Repsly, then the record will be updated. All updated records are counted in the RowsUpdated field.

Data with warnings

If the record being inserted has an error, but this error doesn't prevent insertion, details about this record and the adjoining error will be shown in the Warnings part of the response. All records with warnings will still be counted in the RowsInserted field (in the case of new data) or in the RowsUpdated field (in the case of existing data).

Data with errors

If the record being inserted has an error, and this error prevents insertion, details about this record and the adjoining error will be shown in the Errors part of the response. All records with errors are counted in the RowsInvalid field.

> **i** *Method: GET*
> *Path: https://api.repsly.com/v3/export/importStatus/{importJobID}*
> *Accept: application/json, application/xml*
> *Content-Type: application/json, application/xml*

Response definition		
Name	**Type**	**Description**
ImportStatus	String	Description about the status of the Import. The status can be:

		● Completed - import is completed ● Failed - import has failed ● Pending - import is in progress
RowsInserted	Integer	Total numbers of new rows inserted with the import
RowsUpdated	Integer	Total numbers of rows updated with the import
RowsInvalid	Integer	Total numbers of rows which were not imported due to some data issues
RowsTotal	Integer	The sum of inserted and updated rows
Warnings\ItemID	String	ID of the record which has data warning
Warnings\ItemName	String	Name of the field in the record which has data warning
Warnings\ItemStatus	String	Description of the import warning for the field
Errors\ItemID	String	ID of the record which has data error
Errors\ItemName	String	Name of the field in the record which has data error
Errors\ItemStatus	String	Description of the import error for the field

ERROR REFERENCE

Error reference table	
200 - OK	Everything went fine, you receive a status message.
400 - Bad Request	Invalid format or other error, you receive an error message or an error page.
401 - Unauthorized	Your credentials didn't work.