# ROSSI & ROVETTI FLOWERS

## Basic API

The structure is simple: a push or pull request is initiated from a Partner Organization, or PO, for the purposes of a single record transfer of an order to be fulfilled.

A JSON API is used for a simple push and pull for each order.  Webhooks, while supported, are not covered in this documentation.  When this isn't possible, data polling is supported with CSV from a given URL on intervals no more than 5 minute polling increments.  Discuss with your API team leader whether it's better to use *zappier or mashape* when neither option is available.

An order is defined as:
- a one-to-one sender to receiver for [flowers, bouquets, etc]
- when more than one receiver is required by the sender for multiple receivers, multiple orders *must* be created.  As of the date of this document, there are no plans to support a one-to-many relationship scenario for the orders.

For example, the minimum information required for an order is as follows:

**Partner organization [PO]:** *United States House of Representatives*
**Sender:**          **Nancy Pelosi**
**Sender Address:**     **233 Cannon H.O.B.**
**Sender City:**       **Washington**
**Sender State:**      **DC**
**Sender Zip:**       **20515**

**Recipient:**        **Barry Bonds**
**Recipient Address:**   **301 3rd Street**
**Recipient City**     **San Francisco**
**Recipient State:**    **CA**
**Recipient Zip:**     **94107**

**Delivery Date:**     **1/12/2015**
**Card Message:**      **Congratulations on your retirement! Nancy**

**Product:**         **Designers Choice**
**Description:**       **Celebratory bright colors no vase**
**Price:**          **75.00**

These orders will show up in your API team's portal alongside our sandbox data.

## Sample JSON push:

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "api-key" : "12AB712C09911A18B165210CC002127756E"
    "product": "Designers Choice",
    "description": "Celebratory bright colors no vase",
    "price": "75",
    "type": "object",
    "properties": {
        "name": {
            "description": "Nancy Pelosi",
            "type": "string"
        },
            "address": {
            "description": "233 Cannon H.O.B.",
            "type": "string"
        },
            "city": {
            "description": "Washington",
            "type": "string"
        },
            "locState": {
            "description": "DC",
            "type": "string"
        },
            "zip": {
            "description": "20515",
            "type": "integer"
        },
            "shippingname": {
            "description": "Barry Bonds",
            "type": "string"
        },
            "shippingaddress": {
            "description": "301 3rd Street",
            "type": "string"
        },
            "shippingcity": {
            "description": "San Francisco",
            "type": "string"
        },
            "shippingsLoctate": {
            "description": "CA",
            "type": "string"
        },
            "shippingzip": {
            "description": "94107",
            "type": "integer"
        },
        "deliverydate": {
            "description": "01/12/2015",
            "type": "string"
        },
    "cardmessage": {
            "description": "Congratulations on your retirement! Nancy",
            "type": "string"

        }
    },
}
```

## Additional Fields

Of course, this is the simplest example of an order, pushed into our system for delivery, on 01/12/2015.  Attributes such as 'earliestTime' and 'latestTime' are also available, in addition to other options of the push [such as desiredResultURI].

The JSON is structure is purposefully <u>flat.</u>  No child to parent relationships exist in the current architecture, nor are there development plans to do so.  This is because our system must, in real time, calculate ETAs, schedule drivers and designers, and decide on product spacing for each Point-of-Presence [POP].

Because of the nature of this real-time calculus, during the holiday seasons, race conditions exist.  In other words, our just-in-time inventory system and our route calculator is always [and forever] working on optimizing itself, thus even if the header of the JSON is received [identifying the product, the price, etc] will at least make it into the queue if a simultaneous request is made by the delivery API [internal to Rossi & Rovetti] is discerning a delivery delay.  This allows our design teams within each POP to prepare the product specified and to – offline – discern which delivery means necessary to deliver the product.

It's important to stress that date and time fields are all of type <u>string</u>.  Our system integrates worldwide data from different data points [with different date formats, etc] thus we normalize the date and time where necessary within our systems quite capably; whether it's a 24-hour notation, 12-hour notation, a hex notation, a binary notation, etc.  Since much of the duty of our service engine is to ensure the most optimized send-to-deliver experience,  it's important to ensure your data is being integrated correctly using your API Team's portal.

Your API Team Relationship Manager will explain the portal, how to check for data validity, etc.  It's not the current scope of this document to cover the portal, since we do provide varying UIs to each PO within our network, depending on the needs and the requirements of the PO.


## Push versus Pull

It's the same!  To pull the data that you are looking for does not require a record id; rather, some verbs that would make the order 'distinct' is all that's required.  Usually the delivery address, last name and delivery date.  However, as a best practice, use the orderID that our API system provides post-push to query the database for ETA, delivery signature URIs, etc.

## Time Travel

In early 2014, we introduced the concept of 'time travel,' which is extending pushing an order to test for an ETA without actually placing the order.

The other use of Time Travel is to push 'tenative' placeholders, or orders that should be put into the routing system but not fully placed into our billing system.

The last use of Time Travel [verb: `timetravel`] is to insert an order when the time has past, but to ensure that it's in the system for the route calculator to build it into the delivery and/or designer mechanism for the specific POP it's arriving at.

Whether or not your organization requires the use of this verb is dependent on the needs of your organization.  However, we restrict the time travel authority to only those who require it, so please consult with your relationship manager should you require this access.

We're not meaning to be restrictive - but we have hundreds of data insert points into our system, thus it's necessary to ensure that the data quality and/or routing is the best experience for each of our POs.  While we haven't had any abuses of this system to date – *so be kind when you rewind*. ☺


## Teams

To manage workflow more efficiently, we separate out each API segment into teams, or sender PO regions.  As of the writing of this document:

| | |
|---|---|
| *Luigi* | *California* |
| *Mario* | *Pacific Northwest* |
| *Antonio* | *Southwest* |
| *Gincarlo* | *Midwest* |
| *Giorgio* | *Plain States* |
| *Giovanni* | *New England* |
| *Placido* | *Mid-Atlantic* |
| *Tomasso* | *Southern Atlantic* |
| *Bruno* | *South-Atlantic+Gulf* |
| *Alfonso* | *South-Pacific* |
| *Cecilia* | *Hawaii* |
| *Bella* | *EMEA* |
| *Carmine* | *Canada & Mexico* |
| *Carlotta* | *South America* |

## Data Dictionary

| verb | Definition | Use Required? |
|---|---|---|
| idOrder | Specific identity/key numerical to each order inserted into the system. | Best practice is to use this when constructing a JSON query to request information. |
| idCust | Each sender receives this id. Sender-related preferences will require this in the future "Allergic to Lilies." | |
| orderDate | This must be the timestamp from your server at the time of push. | Yes, push only. |
| orderDateInt | Alternative timestamp when pulling. | |
| randomKey | This is for security purposes. | |
| rewardsUsed | If a PO decides on designating sender rewards, this is a key to a separate database to populate the amount of rewards that the sender has left. | |
| subTotal | Applicable only if taxes are to be collected by us and not the PO. | Yes, push only, only if applicable. |
| taxTotal | Applicable only if taxes are to be collected by us and not the PO. | Yes, push only, only if applicable. |
| shipmentTotal | Applicable if the PO decides on a shipment fee in addition to our billing. | |
| Total | Total of the arrangement | Yes, push only. |
| shipmentMethod | Requested shipmentMethod. Valid examples include a driver's | |

| | | | |
|---|---|---|---|
| | name or service, 'Bunny' or 'Jane' in a specific POP. This simply differs priority to that method for a PUSH. | | |
| shipmentStore | The name of the POP that it's being fulfilled by; e.g. '644 Brannan Street San Francisco' | | |
| name | The first name of the sender. May be generic if the PO decides to cloak this information. | Yes, push only | |
| lastName | The last name of the sender. May be generic if the PO decides to cloak this information. [Last name may be omitted if submitted through name] | Yes, push only | |
| customerCompany | Optional information if a sub-company is being used or applied by the PO | | |
| phone | Phone number of the sender | | |
| email | eMail address of the sender | | |
| address | Address of the sender | Yes, push only | |
| city | City of the sender | Yes, push only | |
| locState | State of the sender | Yes, push only | |
| locCountry | Country of the sender | | |
| zip | Zipcode of the sender | Yes, push only | |
| shippingName | Name of the recipient | Yes, push only | |
| shippingLastName | Lastname of the recipient ; may be omitted if inserted into name. | Yes, push only | |
| shippingCompany | Company name of recipient. | Yes for push but a NULL value or non-inserted will assume a residential address. | |
| shippingAddress | Address of recipient. Separate more than one line by slash '/' | Yes, push only | |
| shippingCity | City name of the recipient | Yes, push only | |

| | | |
|---|---|---|
| shippingLocState | State of the recipient | Yes, push only |
| shippingLocCountry | Country of the recipient | Assumes USA on push. |
| shippingZip | Postal code of the recipient | |
| paymentType | If it's to be collected by Rossi & Rovetti, valid forms would be 'Credit Card,' 'Paypal,' etc. | |
| cardType | Valid types would be 'Visa,' 'Discover,' 'JCB,' etc. | |
| cardNumber | Credit card number MD5 HASH using an onsite pre-exchanged PKI | |
| deliverydate_comment | Comment on delivery date, useful for 'May deliver on Saturday if she's not home,' or 'Call first to ensure he's at work,' etc. | |
| cardExpMonth | Month of credit card number | |
| cardExpYear | Year of credit card number, last 2 digits only [assumes 20 as leading digits] | |
| cardVerify | CVV or EVC code from the credit card. | |
| cardName | Name on the credit card. | |
| generalComments | General Comments on the order visible to the sender and usually entered by the sender. "She really hates purple, and don't use roses." | |
| orderStatus | Valid Pull information examples are 'Processed,' 'En-Route,' 'Delivered,' 'Rejected' [rare] etc. | |
| auditInfo | The biological test information of the flower arrangement including alkalinity, biological contaminant PPM, phosphorisity, etc. | |
| storeComments | Comments directed to the POP by the PO. For | |

| | | |
|---|---|---|
| | example, "VIP Client to us", etc. | |
| storeCommentsPriv | Comments directed to Rossi & Rovetti HQ <u>NOT</u> viewable by the POP.  For example, "Designer Yolanda should not be used for this arrangement for XYZ reason." | |
| adjustAmount | Adjustment amount if necessary. | |
| adjustReason | Adjustment reason if necessary. | |
| taxExempt | If system is required to maintain taxExempt status, example would be 1 or 0, or 'Y' or 'N' [this gets normalized] | |
| discCode | Discount code | |
| discPerc | Discount percentage | |
| discTotal | Discount Total | |
| shippingPhone | Recipient's Phone number | Yes, push only |
| handlingFeeTotal | HandlingFee if required by PO | |
| idAffiliate | id of Affiliate ; not applicable to POs. | |
| commPerc | Not usually applicable to POs. | |
| searchRefererDomain | Not usually applicable to POs. | |
| searchPhrase | Not usually applicable to POs. | |
| Referer | Not usually applicable to POs. | |
| smsNumber | Cellphone number of sender should they'd need an automated text message upon order status, delivery. | |
| smsGateway | Desired MMS/SMS gateway.  If null or not provided, our gateway is used. | |
| orderType | Contextual | |
| orderSource | Contextual | |

| | | |
|---|---|---|
| lastpage | Contextual | |
| ccArchived | Not usually applicable by POs. | |
| heardaboutus | Captures the information if it was a recapture or if was self-inserted by the PO. | |
| dnc | Means Do-not-call.  This is a do-not-disturb sender or recipient under any circumstances.  The POP will relay information to the PO if this is set to '1' or 'Y' [normalized]. | |
| freeshirt | Code verb for a re-delivery. If selected 'Y' or '1' [normalized,] this designates to our system to require a different designer and driver to complete the re-delivery than the previous attempt. | |
| cardmessage | Card message to be put onto the order | Yes, push only |
| driver | Driver's name, for pull only | |
| deliverydate | Delivery date in many formations [normalized] | Yes, push only. 'Adjustable' dates are supported by the comment field only. |
| designer | Designer's name, for pull only | |
| imgExists | For pull, it returns either a NULL or the URL of the final image | |
| flag | Flags set to 'Y' or '1' will require the designer at the POP to contact the PO prior to processing the order. | |
| priority | Values are from 1-10, and simply designating the priority on the order.  This assists our system with routing, so we suggest either leaving this blank | |

| | | |
|---|---|---|
| | [which defaults to '3'] or setting it to '2' under time-driven circumstances, and '1' only for an absolute required urgent design and delivery. | |
| trackerCount | Returns the number of times the record has been pulled | |
| isMobile | If the recipient is mobile or homeless, designate this as a '1' with the POP | |
| postcardUrl | Inserting an IMG URL here will print that image on a one-sided postcard at the POP and will be provided with the card message. | For push only, optional |
| signatureImg | URL of the signature image from the driver | For pull, image captured on delivery. |
| signatureName | Text-based representation of the delivery, for example, 'Front Desk' | For pull |
| Timetravel | See above definition, use cautiously. | For push only/restricted |
| ETA | Driver ETA | For pull |
| DriverWhereabouts | Lat and Long of current driver whereabouts | Useful for realtime mapping of where the driver is relative to the route |
| RouteSequence | The number of deliveries left before this gets delivered. '5' would mean 5 deliveries for that chosen driver before the specific delivery is completed. | |
| ohShit | Use sparingly, this is a 'jump-ahead' scenario and bypasses all routing and forces the POP to manually insert it into the delivery queue. | For push only/restricted. |

## We heart you!

While this document is borderline negligent in providing full examples of every scenario, it's purposeful.  Each PO has it's own requirements, restrictions, intranet integrations, etc.

 Your Relationship Manager will designate two full time developers at your disposal on 12 hour intervals [out of Appleton, Wisconsin and out of Banglore, India] for integration, testing and technical support.

Thank you so much for being a part of the Rossi & Rovetti Flowers PO team!  We assure you the highest in quality service with the fastest delivery times and always a zero delivery cost.  We've worked hard on this API over the years and as a result, we're always learning new ways to improve it.  Please share anything you could identify as a need to your Relationship Manager.