Push Service (Triggered Email Delivery)

#### Design Specifications

# Purpose:

The purpose of this document is to provide a comprehensive documentation of the Push Web Services (Triggered Email Delivery) that will be implemented. Key end to end features will be discussed to include java classes, technologies used, configurations, files and file utilities, constraints as well as key components and how these parts interact with each other and how the application functions. This document will serve as an agreement as to what will be delivered to the [internal] client/customer.

# Scope:

The scope of this document is restricted to describing how the Push Web Services (Triggered Email Delivery) is designed and how it is able to achieve its intended goal.

# Audience:

The intended audience of this document are the customer, stake holders as well as any technical resources involved in current or future development of this application / web service.

# Requirements:

Design and implement the following web services:

This will be background job.

Atlas will provide the data via an xml file.

The technology to write the application in is java.

It will be a scheduled chrontab /hour\_glass job.

It will send RESTful service calls to the Push REST API.

There will be one push request for each order.

10/01/2017 – New requirement was added for processing user track details changes i.e. changes to user [email notification] opt-in preferences.

The Push reference source that is used to make this API is:

[https://www.push.com/documentation/REST\_API/#sending-messages-via-api-triggered-delivery](https://www.appboy.com/documentation/REST_API/#sending-messages-via-api-triggered-delivery)

[https://www.push.com/documentation/REST\_API/#user-track-request](https://www.appboy.com/documentation/REST_API/#user-track-request)

# Design:

## Key Components:

There is an implicit and obvious service level agreement (SLA) that requires online ecommerce traffic and interactions to be secure. Security will be a key design time concern. Resilience was built into the application; so there is error recovery and error mitigation was a design time concern. Invalid xml file and other data issues are proactively detected and handled. Since there is the potential for future processing of various Push related file feeds, the API was designed to for easy maintained and for extensibility.

#### Industry Standards:

The job design will include many industry standard components and design features. The technology stack includes Apache Maven, Java 1.8, Stream API, Lambda expressions, Functional Interfaces and other more recent tools available.

Queue ADT will be used to organize multiple pending files for processing.

Performance was a key concern in terms of throughput and resource (cpu|memory) management.

As a result the following were carefully considered:

1. Lazy instantiation of objects paradigm of was be used as needed.
2. Special care was made to ensure that looping and other expensive operations are limited to a maximum O(log n) performance.
3. Stream API was used to perform bulk operations on various Collections objects.
4. Stream API was preferred/used instead of any looping control mechanisms.
5. Http request/response was done asynchronously along with multi-threading for processing multiple data inputs in parallel.

Capacity to process large files restricting how much resources it is allowed to use and how it process large files.

Security being an implicit and obvious design time concern was built in from day one. Various OWASP suggested code features were implemented.

These include: input XML file validated via regular expression to remove potentially malicious characters (XML)

SAX Parse validation is used for verify well-formed and other XML errors.

There will be an industry recommended configurable artificial delay between http method operations so as to avoid the consumer (Push) mistaking us for a denial of service (DDoS) attack.

Apache Maven was used to streamline and manage jar file versions as well as automate the various JUnit tests that were integrated into the compile and packaging process.

Many key components are configurable (via configuration file) with reduces the need for code changes/recompile/deploy/testing.

Designed specifically to satisfy implicit SLA of being maintainable, extensible and scalable.

The performance (memory, time, through-put) is also a design time concern. Object instantiation is delayed until actually needed. Nested loops are always avoided unless needed or necessary. Looping in general is optimized with goal a minimum O(n log n ) performance.

The general design philosophy is to make components loosely coupled but highly cohesive.

The design is intended to be data-driven and various execution paths occur as a result of the contents of the input xml file.

The source code will be documented as expected, and will be Javadoc compliant.

There will be extensive use of software design patterns such as Command, Factory, Value Objects, Service Locator, Abstract Factory, Null Object, Singleton and Façade.

Processing of pending xml input files is done via single queue which submits file processing to a configurable cached thread pool as the reusable threads become available.

The application can be configured to send http requests asynchronously with future response object processing so that the threads don't get blocked waiting for the response from the Push server.

The application can be configured to force the communication to be done via secure socket (SSL).

Exceptions will not be allowed to propagate to the consumer of this service.

The API can be modified easily for new Feed Types by having all the common feed functionality separated from the specific needs of the particular feed, only the specific file processing code and needs to be changed.

#### Trade-offs:

The initial intention was to design the service to be completely data agnostic with the emphasis on not hard coding anything inside the source code but the development was becoming overly complicated so a compromise was made to hard code action, campaign, broadcast. The trigger properties DOM node is completely data agnostic and effectively is a pass-through function to the Push REST endpoint.

For simplicity, if the input file that is being processed has a fail scenario, then it will be placed in the failed folder. The alternative of logging the specific record within a file, or the entire data file would have been too problematic.

#### Other Considerations:

Since the performance of the service relies heavily on multi-threading, the log4j Asynchronous appender was used in a rolling log file configuration for archiving the log file when a predefined threshold is reached. Logging will be extensive with production support in mind in case of any production issues. Log levels are also carefully considered so as to not clutter the log file with too much debug information.

HTTP Connection will be purposely left open for the duration of the processing so as to eliminate the overhead associated with open/close connection for every Push request.

The job has an optional command argument where the location of the properties configuration file to use is located.

The job can process multiple files concurrently as mentioned before, and also multiple push request within the same file.

The xml input file only has a few required fields:

|  |  |
| --- | --- |
| **Table 1. Required Fields (XML) (ORDER)** | |
| Field | Valid values | | Comments |
| “CPY\_CODE” | 081 | 091 | [blank] | | This specifies the type of device e.g. 081-apple, 091-android or other. |
| “EMAIL\_TYPE” | “DELIVER\_CONFIRM”, ”OUT\_FOR\_DELIVERY”,  ”ORDER\_PREPARE”,  ”ORDER\_CONFIRM” | | Specifies the type of campaign to be used in the push request. |
|  |  | |  |

The application will have the capability to anticipate a situation where the job scheduler has collisions and or process overlap issues. The design strategy is to use a Queue abstract data type so as to manage the processing of multiple files if needed. The current file that is being processed will be locked via a file rename to “[filename].lck” and upon completion, the file will be moved to a processed directory and then unlocked.

A timestamp will be appended to the file name to avoid file naming collision issues if the same file/filename is processed more than once.

No database will be used.

Logging mechanism will be log4J which will be used extensively.

Data will be obtained from an xml file that is output from the Atlas system.

There will be an order level field for the total items in the order with the label “NUMITEMS”.

The Push reference source that is used to make this API is:

[https://www.push.com/documentation/REST\_API/#sending-messages-via-api-triggered-delivery](https://www.appboy.com/documentation/REST_API/#sending-messages-via-api-triggered-delivery)

# Design:

|  |  |
| --- | --- |
| **Table 2. Directory Structure** | |
| Directory | Purpose | | Comments |
| /pending | Output files from Atlas will be placed here | |  |
| /processed | Processed files will be placed here | | A timestamp will be appended to the filename. |
| /failed | File validation and other data issues will result file placed here. | | If the file has any type of processing problem then it will be placed in the failed folder. |
| /logs | Log4j log files placed here | |  |
| /app | Location of the application | |  |
| /dependency-jars | Jar files | |  |
| /resources | Various resource files such as config.properties and log4j.properties | |  |

# Data Field Naming Convention **(ORDER)**:

Please consider variables X and Y as indexes which always start from 1, and are incremented as needed as the number of recipients and products respectively increases.

X - is the index that keeps track of the number of recipients.

Y - is the index that tracks the number of products associated with the corresponding recipient.

P.S. A recipient in this context is the person(s) who is receiving the acme/gift/candy etc. and not the Push user/customer. The term can be confusing so I just wanted to clarify.

|  |  |
| --- | --- |
| **Table 3. Field Naming Convention (ORDER)** | |
| **SHORT** | **COMMENTS** | |
| REC\_[X]\_[FIELD\_NAME] | Recipient level fields i.e. recipient name, address etc. e.g. REC\_1\_REC\_ADDR1 | |
| REC\_[X]\_ITEM\_[Y]\_[FIELD\_NAME] | Recipient product level fields e.g. REC\_1\_ITEM\_1\_PRODUCT\_NAME | |

The naming convention therefore looks like the following:

REC\_[X]\_[FIELD\_NAME] - for recipient level fields such as recipient name, recipient address etc. e.g. REC\_1\_REC\_CITY

REC\_[X]\_ITEM\_[Y]\_[FIELD\_NAME] - for product level fields such as product name, product code etc. e.g. REC\_1\_ITEM\_1\_PRODUCT\_NAME

Please see below for a few use cases:

|  |
| --- |
| **Use Case 1: (1 recipient with 1 product) (ORDER)** |
| REC\_1\_REC\_FNAME  REC\_1\_REC\_LNAME  REC\_1\_REC\_ADD1  REC\_1\_REC\_CITY    REC\_1\_ITEM\_1\_PRODUCT\_NAME  REC\_1\_ITEM\_1\_PRODUCT\_CODE  REC\_1\_ITEM\_1\_QUANTITY  REC\_1\_ITEM\_1\_PRICE |

|  |
| --- |
| **Use Case 2: (1 recipient with 3 products) (ORDER)** |
| REC\_1\_REC\_FNAME  REC\_1\_REC\_LNAME  REC\_1\_REC\_ADD1  REC\_1\_REC\_CITY    REC\_1\_ITEM\_1\_PRODUCT\_NAME  REC\_1\_ITEM\_1\_PRODUCT\_CODE  REC\_1\_ITEM\_1\_QUANTITY  REC\_1\_ITEM\_1\_PRICE    REC\_1\_ITEM\_2\_PRODUCT\_NAME  REC\_1\_ITEM\_2\_PRODUCT\_CODE  REC\_1\_ITEM\_2\_QUANTITY  REC\_1\_ITEM\_2\_PRICE    REC\_1\_ITEM\_3\_PRODUCT\_NAME  REC\_1\_ITEM\_3\_PRODUCT\_CODE  REC\_1\_ITEM\_3\_QUANTITY  REC\_1\_ITEM\_3\_PRICE |

|  |
| --- |
| **Use Case 3: (2 recipients with 2 and 3 products respectively) (ORDER)** |
| REC\_1\_REC\_FNAME  REC\_1\_REC\_LNAME  REC\_1\_REC\_ADD1  REC\_1\_REC\_CITY    REC\_1\_ITEM\_1\_PRODUCT\_NAME  REC\_1\_ITEM\_1\_PRODUCT\_CODE  REC\_1\_ITEM\_1\_QUANTITY  REC\_1\_ITEM\_1\_PRICE    REC\_1\_ITEM\_2\_PRODUCT\_NAME  REC\_1\_ITEM\_2\_PRODUCT\_CODE  REC\_1\_ITEM\_2\_QUANTITY  REC\_1\_ITEM\_2\_PRICE    REC\_2\_REC\_FNAME  REC\_2\_REC\_LNAME  REC\_2\_REC\_ADD1  REC\_2\_REC\_CITY    REC\_2\_ITEM\_1\_PRODUCT\_NAME  REC\_2\_ITEM\_1\_PRODUCT\_CODE  REC\_2\_ITEM\_1\_QUANTITY  REC\_2\_ITEM\_1\_PRICE    REC\_2\_ITEM\_2\_PRODUCT\_NAME  REC\_2\_ITEM\_2\_PRODUCT\_CODE  REC\_2\_ITEM\_2\_QUANTITY  REC\_2\_ITEM\_2\_PRICE    REC\_2\_ITEM\_3\_PRODUCT\_NAME  REC\_2\_ITEM\_3\_PRODUCT\_CODE  REC\_2\_ITEM\_3\_QUANTITY  REC\_2\_ITEM\_3\_PRICE |

|  |  |  |
| --- | --- | --- |
| **Table 4. Push Service Response Codes** | | |
| **CODE** | **SHORT** | **COMMENTS** | |
| **400** | Bad Request | Bad syntax. | |
| **400** | No Recipients | There are no external IDs or segment IDs or no push tokens in the request | |
| **400** | Invalid Campaign ID | No Messaging API Campaign was found for the campaign ID you provided | |
| **400** | Message Variant Unspecified | You provide a campaign ID but no message variation ID | |
| **400** | Invalid Message Variant | You provided a valid campaign ID, but the message variation ID doesn’t match any of that campaign’s messages | |
| **400** | Mismatched Message Type | You provided a message variation of the wrong message type for at least one of your messages | |
| **400** | Invalid Extra Push Payload | You provide the “extra” key for either “apple\_push” or “android\_push” but it is not a dictionary | |
| **400** | Max input length exceeded | Caused by: More than 50 external ids | |
|  |  |  | |
| **400** | No message to send | No payload is specified for the message | |
| **400** | Slideup Message Length Exceeded | Slideup message > 140 characters | |
| **400** | Apple Push Length Exceeded | JSON payload > 1912 bytes | |
| **400** | Android Push Length Exceeded - | JSON payload > 4000 bytes | |
| **400** | Bad Request | Cannot parse send\_at datetime | |
| **400** | Bad Request | In\_local\_time is true but time has passed in your company’s time zone | |
| **401** | Unauthorized | Unknown or missing app group id | |
| **403** | Forbidden | Rate plan doesn’t support or ccount is otherwise inactivated | |
| **404** | Not Found | Unknown App Group ID | |
| **429** | Rate limited | Over rate limit | |
| **5XX** | Internal Server Error | Internal server error, you should retry your request with exponential backoff | |

# Appendix

## Key Files

|  |
| --- |
| **Appendix A. Configuration - config.properties** |
| # Push Push to REST Service API configuration file.  # author: Casmon Gordon  # date: August 8, 2017  # www.acme.com  #  # Push REST configuration details  #  #configuration settings for the push service (Send Triggered Email Delivery)  trigger.send.endpoint=https://api.push.com/campaigns/trigger/send  trigger.send.method=POST  trigger.send.contenttype=application/json  #configuration settings for the Push service (Message Create Triggered Email Delivery)  message.schedule.create.endpoint=https://api.push.com/messages/schedule/create  message.schedule.create.method=POST  message.schedule.create.contenttype=application/json  #configuration settings for the Push service (Schedule Create Triggered Email Delivery)  trigger.schedule.create.endpoint=https://api.push.com/campaigns/trigger/schedule/create  trigger.schedule.create.method=POST  trigger.schedule.create.contenttype=application/json  #configuration settings for the Push service (Schedule Update Triggered Email Delivery)  trigger.schedule.update.endpoint=https://api.push.com/campaigns/trigger/schedule/update  trigger.schedule.update.method=POST  trigger.schedule.update.contenttype=application/json  #configuration settings for the Push service (Schedule Delete Triggered Email Delivery)  trigger.schedule.delete.endpoint=https://api.push.com/campaigns/trigger/schedule/delete  trigger.schedule.delete.method=POST  trigger.schedule.delete.contenttype=application/json  #configuration settings for the Push service (User Track Request)  user.track.endpoint=https://api.push.com/users/track  user.track.method=POST  user.track.contenttype=application/json  #  #general properties  #  #folder location where input xml files are to be placed  pending.dir=/opt/jobuser/push/indata/pending/  #folder location where processed output xml files are renamed and placed  processed.dir=/opt/jobuser/push/indata/processed/  #limit on how many files any output folder (failed | processed) is allowed to have  service.dir.max.files=100  #folder location where failed output xml files are renamed and placed  failed.dir=/opt/jobuser/push/indata/failed/  #log4j folder settings  log.dir=/opt/jobuser/push/logs  log.config=/opt/jobuser/push/resources/log4j.xml  #industry recommended artificial delay between push requests  thread.delay=0  # message send delay (minutes)  message.send.delay=10  # message expire (minutes) for apple iOS device push  message.expire=150  #device code  application.supported.devices.definition=apple,android,web  # maximum payload size (OWASP recommended)  payload.threshhold.size=8192  #Push dashboard config identifiers  application.group.apple=ce0d1785-845d-4140-b727-de4847e2a928  application.group.android=7a55d6ad-76f4-4d29-977f-1f17b8bf992e  application.group.web=5a55d6wd-76f4-4r29-477f-1f17b8bf992e  #company identifier  company.identifier.apple=3374f94b-617f-499f-bef9-b79da6f034a7  company.identifier.android=51d50875-afc4-4fc2-ad6f-9b355588df4f  company.identifier.web=91d50t75-aff4-3fc2-adrf-9i395587df4f  #declare list of campaigns reference keys  trigger.campaign.list.definition=ORDER\_CONFIRM,OUT\_FOR\_DELIVERY,ORDER\_PREPARE,DELIVERY\_CONFIRM  #sms send enable default opt-in value setting  trigger.sms.default.optin=Y  #  #declare campaign details  #  #declare default campaign no sms campaign details  campaigns.definition.nosms.campaign.default=no-sms-default-c6f69158-98a0-407c-843f-b3de5c2658b3  #declare order\_confirm campaign details  campaigns.definition.ORDER\_CONFIRM.apple.id=c6f69158-98a0-407c-843f-b3de5c2658b3  campaigns.definition.ORDER\_CONFIRM.apple.msg.variation.id=iosPush-3461  campaigns.definition.ORDER\_CONFIRM.android.id=c6f69158-98a0-407c-843f-3de5c2658b3  campaigns.definition.ORDER\_CONFIRM.android.msg.variation.id=androidPush-2304  campaigns.definition.ORDER\_CONFIRM.web.id=h6f69158-98a0-406c-843f-3de5c2658b3  campaigns.definition.ORDER\_CONFIRM.web.msg.variation.id=webPush-2290  #declare shipping\_status campaign details  campaigns.definition.OUT\_FOR\_DELIVERY.apple.id=cde7fccf-2443-4eeb-ab12-3cd2781335aa  campaigns.definition.OUT\_FOR\_DELIVERY.apple.msg.variation.id=iosPush-3812  campaigns.definition.OUT\_FOR\_DELIVERY.android.id=cde7fccf-2443-4eeb-ab12-3cd2781335aa  campaigns.definition.OUT\_FOR\_DELIVERY.android.msg.variation.id=androidPush-2709  campaigns.definition.OUT\_FOR\_DELIVERY.web.id=cde7fccf-2443-4eeb-ab12-3cd2781335aa  campaigns.definition.OUT\_FOR\_DELIVERY.web.msg.variation.id=webPush-2291  #declare shipping\_status campaign details  campaigns.definition.ORDER\_PREPARE.apple.id=000e93f9-0593-430e-bea3-26683e20e998  campaigns.definition.ORDER\_PREPARE.apple.msg.variation.id=iosPush-3812  campaigns.definition.ORDER\_PREPARE.android.id=000e93f9-0593-430e-bea3-26683e20e998  campaigns.definition.ORDER\_PREPARE.android.msg.variation.id=androidPush-2709  campaigns.definition.ORDER\_PREPARE.web.id=23e93f9-0593-430r-bea3-266e4e20e998  campaigns.definition.ORDER\_PREPARE.web.msg.variation.id=webPush-2292  #declare delivery\_confirm campaign details  campaigns.definition.DELIVERY\_CONFIRM.apple.id=705ca620-17f4-4fd5-bd7e-9d93048c3f4d  campaigns.definition.DELIVERY\_CONFIRM.apple.msg.variation.id=iosPush-4131  campaigns.definition.DELIVERY\_CONFIRM.android.id=705ca620-17f4-4fd5-bd7e-9d93048c3f4d  campaigns.definition.DELIVERY\_CONFIRM.android.msg.variation.id=androidPush-3111  campaigns.definition.DELIVERY\_CONFIRM.web.id=305cws20-17f4-4fd5-rd7e-9d93048c3fwe  campaigns.definition.DELIVERY\_CONFIRM.web.msg.variation.id=webPush-2293  #declare SMTP email config details  service.email.SMTPHost=mailerny.acme.int  service.email.From=do-not-reply@acme.com  service.email.To=cgordon@acme.com  service.email.Subject=PushAPI Web Services Error log  service.email.BufferSize=5  service.email.Port=25  service.email.EvaluatorClass=TriggerLogEvent  service.email.layout=org.apache.log4j.PatternLayout  service.email.layout.ConversionPattern=%m  service.email.level=error  #toggle email capability  service.email.enabled=false  #Mapping configuration for standard Atlas file  data.mapping.reference.ROOT=ORDER  data.mapping.reference.RECIPIENT=RECIPIENT  data.mapping.reference.PRODUCT\_ITEM=ITEM  data.mapping.reference.BROADCAST=  data.mapping.reference.SMSOPT=SMSOPTIN  data.mapping.reference.NUMITEMS=NUMITEMS  data.mapping.reference.STORE\_BRAND=BRAND  data.mapping.reference.ORDER\_DATE=ORDER\_DATE  data.mapping.required.ORDER\_NUMBER=ORDER\_NUM  data.mapping.required.EXTERNAL\_USER\_ID=EXTERNAL\_USER\_ID  data.mapping.required.USER\_ALIAS=CUST\_EMAIL  data.mapping.required.DEVICE\_CODE=CPY\_CODE  data.mapping.required.NOTIFICATION\_TYPE=EMAIL\_TYPE  data.mapping.reference.notification.key.apple=1  data.mapping.reference.notification.key.android=2  data.mapping.reference.notification.key.web=3  data.transform.reference.ORDER=TRIGGER\_SEND  data.transform.reference.TRACK=USER\_TRACK  data.transform.reference.device.apple=081  data.transform.reference.device.android=091  data.transform.reference.device.web=000  data.transform.replace.character.regex=\t  #  # multi threading configuartion settings  #  service.threads.maximum=10  service.threads.initial=5  service.threads.pool.delay=3  service.threads.keepalive=20  service.threads.pool.sleep=3000  service.threads.monitor.sleep=2000  service.threads.monitor.interval=1000  # http async client settings  service.http.async.max.clients=10  service.http.async.future.max.wait=3000  # default store brand if missing from data file  service.store.brand.default=1001  # default company code (device) if missing from data file  service.reference.device.default=web  # boolean flag show external\_user\_id  data.show.external\_user\_id.flag=false  # define mapping key for non order file feeds  data.feed.type.reference.list=ORDER,TRACK  data.feed.type.reference.TRACK.id=TRACK  data.feed.type.reference.ORDER.id=ORDER  # specific data settings for USER TRACK Service  data.feed.type.reference.TRACK.fields.list=app\_group\_id  data.feed.type.reference.TRACK.field.app\_group\_id.value=qo0wei23ru2-3ir3-r234otr234i-ebverb  data.feed.type.reference.TRACK.fields.data.root=USER  data.feed.type.reference.TRACK.fields.map.source.id=SOURCE  data.feed.type.reference.TRACK.fields.map.SOURCE.code=T::FLWS,B::BOTH,M::MKTG | |

|  |
| --- |
| **Appendix B. JSON REST Output (Sample) (ORDER)** |
| {  "app\_group\_id": "5a55d6wd-76f4-4r29-477f-1f17b8bf992e",  "campaign\_id": "305cws20-17f4-4fd5-rd7e-9d93048c3fwe",  "trigger\_properties": {  "sms\_optin": "Y",  "store\_brand": "1001",  "user\_email": "casmong@yahoo.com",  "order\_number": "W00995422264837",  "broadcast": "false",  "device": "web",  "order\_date": "08-24-2017",  "numitems": "2"  },  "broadcast": false,  "audience": {  "custom\_attribute": {  "custom\_attribute\_name": "",  "comparison": "",  "value": ""  }  },  "recipients": [  {  "external\_user\_id": "",  "user\_alias": {  "alias\_name": "casmong@yahoo.com",  "alias\_label": "email"  },  "trigger\_properties": {  "REC\_1\_ITEM\_1\_CARD\_MESSAGE": "No Card Message",  "REC\_1\_ITEM\_1\_PRODUCT\_IMAGE\_CAT": "http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg",  "REC\_1\_ITEM\_2\_BASE\_CODE": "141014",  "REC\_1\_ITEM\_2\_SHIPPING\_CHARGE": "49.99",  "REC\_1\_REC\_LNAME": "Test",  "REC\_1\_REC\_COUNTRY": "USA",  "REC\_1\_REC\_ATTN\_TEXT": "",  "REC\_1\_ITEM\_1\_ATTMPT\_DATE": "",  "REC\_1\_ITEM\_2\_ALIAS\_NAME": "Meat and Cheese Hamper-CONGRT",  "REC\_1\_ITEM\_2\_FLEX\_DATE": "",  "REC\_1\_ITEM\_2\_PRODUCT\_NAME": "Congrats Sausage & Cheese Basket Orchard Congrats",  "REC\_1\_ITEM\_1\_PRODUCT\_TYPE": "GPT",  "REC\_1\_ITEM\_2\_QUANTITY": "1",  "REC\_1\_ITEM\_1\_PRODUCT\_IMAGE\_TMN": "http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg",  "REC\_1\_REC\_CITY": "Carle Place",  "REC\_1\_ITEM\_1\_PRODUCT\_CODE": "141014",  "REC\_1\_ITEM\_1\_DELIVERY\_DATE": "04/05/2017",  "REC\_1\_REC\_ZIP": "11514",  "REC\_1\_ITEM\_1\_SHIPPING\_CHARGE": "49.99",  "REC\_1\_ITEM\_2\_TAX": "",  "REC\_1\_REC\_STATE": "NY",  "REC\_1\_ITEM\_1\_FLEX\_TEXT": "",  "REC\_1\_ITEM\_2\_ITEM\_BRAND": "1009",  "REC\_1\_ITEM\_1\_PRODUCT\_NAME": "Congrats Sausage & Cheese Basket Orchard Congrats",  "REC\_1\_ITEM\_1\_TRK\_NUM": "",  "REC\_1\_ITEM\_2\_PRICE": "49.99",  "REC\_1\_ITEM\_1\_PRICE": "49.99",  "REC\_1\_ITEM\_2\_SHIPPER\_TYPE": "",  "REC\_1\_ITEM\_2\_PRODUCT\_IMAGE\_CAT": "http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg",  "REC\_1\_ITEM\_1\_TAX": "",  "REC\_1\_ITEM\_1\_ALIAS\_NAME": "Meat and Cheese Hamper-CONGRT",  "REC\_1\_REC\_ADD2": "Suite 250",  "REC\_1\_REC\_ADD1": "175 Westbury Ave",  "REC\_1\_ITEM\_1\_FLEX\_DATE": "",  "REC\_1\_ITEM\_2\_CARD\_MESSAGE": "No Card Message",  "REC\_1\_REC\_FNAME": "Test",  "REC\_1\_ITEM\_2\_PRODUCT\_IMAGE\_TMN": "http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg",  "REC\_1\_ITEM\_1\_DEL\_ON": "",  "REC\_1\_ITEM\_2\_FLEX\_TEXT": "",  "REC\_1\_ITEM\_2\_ATTMPT\_DATE": "",  "REC\_1\_ITEM\_1\_SHIPPER\_TYPE": "",  "REC\_1\_ITEM\_1\_ITEM\_BRAND": "1009",  "REC\_1\_ITEM\_2\_DEL\_ON": "",  "REC\_1\_ITEM\_2\_PRODUCT\_TYPE": "GPT",  "REC\_1\_ITEM\_2\_DELIVERY\_DATE": "04/05/2017",  "REC\_1\_ITEM\_2\_TRK\_NUM": "",  "REC\_1\_ITEM\_2\_PRODUCT\_CODE": "141014",  "REC\_1\_ITEM\_1\_QUANTITY": "1"  }  }  ]  } | |

|  |
| --- |
| **Appendix C. XML Request (Sample) (ORDER)** |
| <ORDER>  <ORDER\_NUM>W00995422264837</ORDER\_NUM>  <ORDER\_DATE>08-30-2017</ORDER\_DATE>  <CPY\_CODE>081</CPY\_CODE><!--DEVICE TYPE -->  <EXTERNAL\_USER\_ID>awjkhbvfawlvhsldvhavldv</EXTERNAL\_USER\_ID>  <CUST\_EMAIL>casmong@yahoo.com</CUST\_EMAIL>  <SMSOPTIN>Y</SMSOPTIN>  <EMAIL\_TYPE>DELIVERY\_CONFIRM</EMAIL\_TYPE><!--ORDER\_CONFIRM ,SHIPPING\_STATUS, DELIVERY\_CONFIRM -->  <RECIPIENT>  <REC\_FNAME>Test</REC\_FNAME>  <REC\_LNAME>Test</REC\_LNAME>  <REC\_ADD1>175 Westbury Ave</REC\_ADD1>  <REC\_ADD2>Suite 250</REC\_ADD2>  <REC\_CITY>Carle Place</REC\_CITY>  <REC\_STATE>NY</REC\_STATE>  <REC\_ZIP>11514</REC\_ZIP>  <REC\_COUNTRY>USA</REC\_COUNTRY>  <REC\_ATTN\_TEXT/>  <ITEM>  <PRODUCT\_NAME>Congrats Sausage &amp; Cheese Basket Orchard Congratulations</PRODUCT\_NAME>  <PRODUCT\_CODE>141014</PRODUCT\_CODE>  <PRODUCT\_TYPE>GPT</PRODUCT\_TYPE>  <ALIAS\_NAME>Meat and Cheese Hamper-CONGRTULATE</ALIAS\_NAME>  <DELIVERY\_DATE>04/05/2017</DELIVERY\_DATE>  <FLEX\_DATE/>  <FLEX\_TEXT/>  <QUANTITY>1</QUANTITY>  <PRICE>49.99</PRICE>  <SHIPPING\_CHARGE>49.99</SHIPPING\_CHARGE>  <TAX/>  <CARD\_MESSAGE>No Card Message</CARD\_MESSAGE>  <ITEM\_BRAND>1009</ITEM\_BRAND>  <ATTMPT\_DATE/>  <DEL\_ON/>  <SHIPPER\_TYPE/>  <TRK\_NUM/>  <PRODUCT\_IMAGE\_CAT>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg</PRODUCT\_IMAGE\_CAT>  <PRODUCT\_IMAGE\_TMN>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg</PRODUCT\_IMAGE\_TMN>  </ITEM>  <ITEM>  <PRODUCT\_NAME>Congrats Sausage &amp; Cheese Basket Orchard Congraaaaats</PRODUCT\_NAME>  <PRODUCT\_CODE>141014</PRODUCT\_CODE>  <PRODUCT\_TYPE>GPT</PRODUCT\_TYPE>  <ALIAS\_NAME>Meat and Cheese Hamper-CONGRAAAT</ALIAS\_NAME>  <DELIVERY\_DATE>04/05/2017</DELIVERY\_DATE>  <FLEX\_DATE/>  <FLEX\_TEXT/>  <QUANTITY>1</QUANTITY>  <PRICE>49.99</PRICE>  <SHIPPING\_CHARGE>49.99</SHIPPING\_CHARGE>  <TAX/>  <CARD\_MESSAGE>No Card Message</CARD\_MESSAGE>  <BASE\_CODE>141014</BASE\_CODE>  <ITEM\_BRAND>1009</ITEM\_BRAND>  <ATTMPT\_DATE/>  <DEL\_ON/>  <SHIPPER\_TYPE/>  <TRK\_NUM/>  <PRODUCT\_IMAGE\_CAT>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg</PRODUCT\_IMAGE\_CAT>  <PRODUCT\_IMAGE\_TMN>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg</PRODUCT\_IMAGE\_TMN>  </ITEM>  <ITEM>  <PRODUCT\_NAME>Congrats Sausage &amp; Cheese Basket Orchard Congratszzzzz</PRODUCT\_NAME>  <PRODUCT\_CODE>141014</PRODUCT\_CODE>  <PRODUCT\_TYPE>GPT</PRODUCT\_TYPE>  <ALIAS\_NAME>Meat and Cheese Hamper-CONGRTZZZZZ</ALIAS\_NAME>  <DELIVERY\_DATE>04/05/2017</DELIVERY\_DATE>  <FLEX\_DATE/>  <FLEX\_TEXT/>  <QUANTITY>1</QUANTITY>  <PRICE>49.99</PRICE>  <SHIPPING\_CHARGE>49.99</SHIPPING\_CHARGE>  <TAX/>  <CARD\_MESSAGE>No Card Message</CARD\_MESSAGE>  <ITEM\_BRAND>1009</ITEM\_BRAND>  <ATTMPT\_DATE/>  <DEL\_ON/>  <SHIPPER\_TYPE/>  <TRK\_NUM/>  <PRODUCT\_IMAGE\_CAT>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg</PRODUCT\_IMAGE\_CAT>  <PRODUCT\_IMAGE\_TMN>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg</PRODUCT\_IMAGE\_TMN>  </ITEM>  </RECIPIENT>  <RECIPIENT>  <REC\_FNAME>Test Recepti</REC\_FNAME>  <REC\_LNAME>Test Recepti</REC\_LNAME>  <REC\_ADD1>175 Westbury Ave</REC\_ADD1>  <REC\_ADD2>Suite 250</REC\_ADD2>  <REC\_CITY>Carle Place</REC\_CITY>  <REC\_STATE>NY</REC\_STATE>  <REC\_ZIP>11514</REC\_ZIP>  <REC\_COUNTRY>USA</REC\_COUNTRY>  <REC\_ATTN\_TEXT/>  <ITEM>  <PRODUCT\_NAME>Congrats Sausage &amp; Cheese Basket Orchard Congratulations ZZ</PRODUCT\_NAME>  <PRODUCT\_CODE>141014</PRODUCT\_CODE>  <PRODUCT\_TYPE>GPT</PRODUCT\_TYPE>  <ALIAS\_NAME>Meat and Cheese Hamper-CONGRTULATE ZZ</ALIAS\_NAME>  <DELIVERY\_DATE>04/05/2017</DELIVERY\_DATE>  <FLEX\_DATE/>  <FLEX\_TEXT/>  <QUANTITY>1</QUANTITY>  <PRICE>49.99</PRICE>  <SHIPPING\_CHARGE>49.99</SHIPPING\_CHARGE>  <TAX/>  <CARD\_MESSAGE>No Card Message</CARD\_MESSAGE>  <ITEM\_BRAND>1009</ITEM\_BRAND>  <ATTMPT\_DATE/>  <DEL\_ON/>  <SHIPPER\_TYPE/>  <TRK\_NUM/>  <PRODUCT\_IMAGE\_CAT>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg</PRODUCT\_IMAGE\_CAT>  <PRODUCT\_IMAGE\_TMN>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg</PRODUCT\_IMAGE\_TMN>  </ITEM>  <ITEM>  <PRODUCT\_NAME>Congrats Sausage &amp; Cheese Basket Orchard Congraaaaats YY</PRODUCT\_NAME>  <PRODUCT\_CODE>141014</PRODUCT\_CODE>  <PRODUCT\_TYPE>GPT</PRODUCT\_TYPE>  <ALIAS\_NAME>Meat and Cheese Hamper-CONGRAAAT YY</ALIAS\_NAME>  <DELIVERY\_DATE>04/05/2017</DELIVERY\_DATE>  <FLEX\_DATE/>  <FLEX\_TEXT/>  <QUANTITY>1</QUANTITY>  <PRICE>49.99</PRICE>  <SHIPPING\_CHARGE>49.99</SHIPPING\_CHARGE>  <TAX/>  <CARD\_MESSAGE>No Card Message</CARD\_MESSAGE>  <BASE\_CODE>141014</BASE\_CODE>  <ITEM\_BRAND>1009</ITEM\_BRAND>  <ATTMPT\_DATE/>  <DEL\_ON/>  <SHIPPER\_TYPE/>  <TRK\_NUM/>  <PRODUCT\_IMAGE\_CAT>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg</PRODUCT\_IMAGE\_CAT>  <PRODUCT\_IMAGE\_TMN>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg</PRODUCT\_IMAGE\_TMN>  </ITEM>  <ITEM>  <PRODUCT\_NAME>Congrats Sausage &amp; Cheese Basket Orchard Congratszzzzz PP</PRODUCT\_NAME>  <PRODUCT\_CODE>141014</PRODUCT\_CODE>  <PRODUCT\_TYPE>GPT</PRODUCT\_TYPE>  <ALIAS\_NAME>Meat and Cheese Hamper-CONGRTZZZZZ nPP</ALIAS\_NAME>  <DELIVERY\_DATE>04/05/2017</DELIVERY\_DATE>  <FLEX\_DATE/>  <FLEX\_TEXT/>  <QUANTITY>1</QUANTITY>  <PRICE>49.99</PRICE>  <SHIPPING\_CHARGE>49.99</SHIPPING\_CHARGE>  <TAX/>  <CARD\_MESSAGE>No Card Message</CARD\_MESSAGE>  <ITEM\_BRAND>1009</ITEM\_BRAND>  <ATTMPT\_DATE/>  <DEL\_ON/>  <SHIPPER\_TYPE/>  <TRK\_NUM/>  <PRODUCT\_IMAGE\_CAT>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014c.jpg</PRODUCT\_IMAGE\_CAT>  <PRODUCT\_IMAGE\_TMN>http://cdn1.baskets-uat.net/wcsstore/Baskets/images/catalog/141014t.jpg</PRODUCT\_IMAGE\_TMN>  </ITEM>  </RECIPIENT>  </ORDER> | |

|  |
| --- |
| **Appendix D. XML Request (Sample) (TRACK)** |
| <TRACK>  <USER>  <EMAIL>cgordon@mail.com</EMAIL>  <SMSOPTIN>Y</SMSOPTIN>  <PHONE>908-213-1234</PHONE>  <SOURCE>T</SOURCE>  <FIRSTNAME>Nanny</FIRSTNAME>  <LASTNAME>Browser</LASTNAME>  </USER>  <USER>  <EMAIL>cgordon1@mail.com</EMAIL>  <SMSOPTIN>Y</SMSOPTIN>  <PHONE>908-213-1231</PHONE>  <SOURCE>M</SOURCE>  <FIRSTNAME>Kenny</FIRSTNAME>  <LASTNAME>Edmonds</LASTNAME>  </USER>  <USER>  <EMAIL>cgordon2@mail.com</EMAIL>  <SMSOPTIN>Y</SMSOPTIN>  <PHONE>908-213-1232</PHONE>  <SOURCE>M</SOURCE>  <FIRSTNAME>Sarah</FIRSTNAME>  <LASTNAME>Parker</LASTNAME>  </USER>  </TRACK> | |

|  |
| --- |
| **Appendix E. JSON REST Output (Sample) (TRACK)** |
| {  "app\_group\_id": "qo0wei23ru2-3ir3-r234otr234i-ebverb",  "attributes": [  {  "PHONE": "908-213-1234",  "FIRSTNAME": "Fran”,  "LASTNAME": "Dreschner",  "EMAIL": "cgordon@mail.com",  "SMSOPTIN": "Y",  "SOURCE": "FLWS"  },  {  "PHONE": "908-213-1231",  "FIRSTNAME": "Arnold”,  "LASTNAME": "Swartzenegger",  "EMAIL": "cgordon1@mail.com",  "SMSOPTIN": "N",  "SOURCE": "MKTG"  },  {  "PHONE": "908-213-1232",  "FIRSTNAME": "Vin”,  "LASTNAME": "Diesel",  "EMAIL": "cgordon2@mail.com",  "SMSOPTIN": "Y",  "SOURCE": "FLWS"  }  ]  } | |

|  |
| --- |
| **Appendix F. Log Configuration - log4J.xml** |
| <!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">  <log4j:configuration debug="true" xmlns:log4j='http://jakarta.apache.org/log4j/'>  <!--  # Push Push to REST Service API log4J XML configuration file.  # author: Casmon Gordon  # date: August 8, 2017  # www.acme.com  -->  <!-- appenders configuration -->  <!-- {console} appenders for standard output -->  <appender name="console" class="org.apache.log4j.ConsoleAppender">  <param name="Target" value="System.out"/>  <layout class="org.apache.log4j.PatternLayout">  <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n" />  </layout>  </appender>  <!-- {file} appenders for file output -->  <appender name="fileAppender" class="org.apache.log4j.RollingFileAppender">  <param name="File" value="${log.dir}/app-logging.log"/>  <param name="MaxFileSize" value="10240KB" />  <param name="Threshold" value="DEBUG" />  <param name="MaxBackupIndex" value="10" />  <layout class="org.apache.log4j.PatternLayout">  <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n" />  </layout>  </appender>    <!-- {async} appenders for asynchronous file logging -->  <appender name="async" class="org.apache.log4j.AsyncAppender">  <param name="BufferSize" value="500"/>  <appender-ref ref="fileAppender"/>  </appender>  <root>  <priority value ="debug"></priority>  <appender-ref ref="console"></appender-ref>  <appender-ref ref="async"></appender-ref>  </root>  </log4j:configuration> | |

# Change Log

|  |  |  |
| --- | --- | --- |
| Changed By | Date | Comments |
| Casmon Gordon | 08-04-2017 | Initial Revision |
| Casmon Gordon | 08-11-2017 | Various updates due to clarifications, log4j configured details |
| Casmon Gordon | 08-23-2017 | Modify for New input file xml format (Atlas Standard). SMS capable. Configurable multithreading. Asynchronous request response with future response object. |
| Casmon Gordon | 08-25-2017 | Updates based on discussions with Jessica. Miscellaneous changes. |
| Casmon Gordon | 08-30-2017 | Changes for 1 push per order, add numitems JSON field, explain naming convention. |
| Casmon Gordon | 09-11-2017 | Added discussion describing the field naming convention |
| Casmon Gordon | 10-19-2017 | Added discussion describing user track feature addition and the general design for extensibility. |
| Casmon Gordon | 10-24-2017 | Added changes for source mapping functionality. |