Qbit Label-Request Protocol
XML

Contents

1 -- Overview

The Qbit software will request a label from a web-service if it is configured to do so. The web-service is expected to return the label as a Base64-encoded string in ZPL (Zebra Printer Language) format. This string can represent shipping label, a store label, an error label, or any Zebra-printable label.

The Qbit software can use HTTP POST or an open Socket to submit an XML-formatted packet to a configurable web service URL or IP Address. The response must be received within a (configurable) length of time, else the operation is aborted.

When a valid response is received, it is decoded and stored in a local database for reference as needed at the Print, Verify, or Sort Zones. (Any response received after the package has passed the Print Zone, and before the timeout, will still be decoded and stored. This is done so that the package may be re-inducted later and receive the previously generated label.)

Sample XML Packets for Label-Request

Following are examples of minimal Label Request and Label Response packets in JSON format. (This example assumes the presence of dimensioning and scale equipment, and that the Ship-To information is not required, as the web service will acquire this information another way.)

Label-Request XML example

```
<?xml version="1.0" encoding="utf-8"?>
<Shipment>
 <LicensePlate>CL412382</LicensePlate>
 <Weight>3.00</Weight>
 <WeightUOM>lb</WeightUOM>
 <Length>23.00</Length>
 <Width>3.00</Width>
 <Height>3.00</Height>
 <DimUOM>in</DimUOM>
 <Client>SHIP02</Client>
 <UserName>QMI</UserName>
 <Password>password</Password>
</Shipment>
```

Label-Response XML example for successful request

```
<?xml version="1.0" encoding="utf-8"?>
<Results>
 <Error id="0"/>
 <Ship>
  <Via carrier="36864">UPS Ground USA</Via>
  <Package>
   <LicensePlate>CL412382</LicensePlate>
   <TrackingNumber>1ZV5131X0311111275</TrackingNumber>
   <ShipLabel>DQpODQSEVTVEVSVCAySxOLCIxIE9GIDEiDQpB...</ShipLabel>
   <ReturnLabel>OLCI4IExCUyINCkE2MzAsMjYCw0xOLDEi...</ReturnLabel>
   <DivertLane>1</DivertLane>
   <Published><Total>10.37</Total></Published>
  </Package>
 </Ship>
</Results>
```

Label-Response XML example for failed request

```
<?xml version="1.0" encoding="utf-8"?>
<Results>
 <Error id="-1">Carton already posted</Error>
 <Ship>
  <Package>
   <LicensePlate/>CL412382</LicensePlate>
   <TrackingNumber/>
   <ShipLabel>MyxOLCI4IExCUyINCkE2Mw0xOLCIxIE9GIDEi...</ShipLabel>
   <DivertLane>0</DivertLane>
   <Published/>
  </Package>
 </Ship>
</Results>
```

2 – Terminology

Zones – Each stage of the system that includes scanner hardware is referred to as a Zone.

Induct Zone – This is the first stage of the system, where cartons are first introduced to Cubiscan equipment. This stage will scan the Carton ID, and may include hardware to weigh and/or dimension the carton. This zone also initiates a label request to the host system and stores the response in a local database. If there is no Print Zone, this zone may also initiate printing of labels.

Print Zone – This zone reads the Carton ID and looks in the local database to find a label. If a label is found, it prints and applies it to the carton. If a label is not found, it may print an error label

Verify Zone – This zone reads the Carton ID and the Tracking Number(s) from the carton and compares this against the expected values stored in the local database. If the values do not match, or if a barcode did not scan, the carton may be diverted, or the conveyor line may stop. This zone may also perform a sort function.

Sort Zone – This zone reads the Carton ID, Tracking Number, or other code from the carton and uses a configurable way to determine where to divert the carton.

3 -- Label Request Process

The Qbit software uses HTTP POST to send an XML-formatted label request to the web application URL and waits for a response. If the web application successfully received the request, the HTTP response should always be 200/OK, with the processing status (success or error) indicated in the <Error> tag of the response body.

The structures of the XML request and response packets are described here. Examples of packet content are shown in section 1 – Overview. The Level column indicates the "tag nesting level".

Label-Request Fields		
Level	Tag Name	Tag Content
1	Shipment	Top-Level Node for Ship Request
2	Sequence	Sequence number of this request, a unique number that generally increments by one for each request, and wraps to 0 when it exceeds 5 digits. This is intended for systems that rely on a sequence number to differentiate requests. Format is **99999**.
2	LicensePlate	Carton Identifier scanned at the Induct Zone
2	Weight	Weight of carton measured at Induct Zone. Format is **9.99**. The value may be blank if the Induct Zone does not include a scale.
2	WeightUOM	Weight Unit-of-Measure. Configurable in Qbit-IMS. Content is one of **oz**,

		lb, **gr**, or **kg**.
2	Length	Dimensions of carton, measured at the Induct Zone. Format is **9.99**. The value may be blank if the Induct Zone does not include dimensioning equipment.
2	Width	
2	Height	
2	DimUOM	Dimension Unit-of-Measure. Configurable in Qbit-IMS. Content is one of **in**, **mm**, or **cm**.
2	Company	Ship-To information. These tags are given non-blank values only if the Qbit software is configured to read this information from the local ordering system.
2	Contact	
2	Address1	
2	Address2	
2	Address3	
2	City	
2	State	
2	Zip	
2	Phone	
2	Transaction	
2	ShipVia	
2	ShipOn	
2	ShipmentId	
2	FreightPayment	
2	ThirdPartyAccountNumber	
2	ThirdPartyCompany	
2	ThirdPartyAddress1	
2	ThirdPartyAddress2	
2	ThirdPartyCity	
2	ThirdPartyState	
2	ThirdPartyZip	
2	ThirdPartyCountry	
2	ThirdPartyPhone	
2	PONumber	
2	ShipmentInstruction	

2	PackageInstruction	
2	ShipFromCompany	
2	ShipFromContact	
2	ShipFromAddress1	
2	ShipFromAddress2	
2	ShipFromAddress3	
2	ShipFromCity	
2	ShipFromState	
2	ShipFromZip	
2	ShipFromCountry	
2	ShipFromPhone	
2	Email	
2	EmailNotificationEvents	
2	Client	Client Identifier for authentication purposes. Configurable.
2	UserName	User name for authentication purposes. Configurable.
2	Password	Password for authentication purposes. Configurable.

Label-Response Fields		
Level	Tag Name	Tag Content
1	Results	Top-Level Node for Label Response.
2	Error	The content of this tag is a descriptive message about the status of the request, meant to be a human-readable version of the "id" attribute of this tag. The "id" attribute specifies the *error status* of the label request. "0" means request was processed successfully, and that a label and tracking number were generated and returned. "-1" means there was an error processing the request, and that the ShipLabel tag may contain a printable *error label*.
2	Ship	Parent Node for shipping details.
3	Via	The content of this tag is the host-specific service level description. Tag attribute [carrier] is a host-specific carrier code, which can be used by the Qbit software for sorting purposes.
3	Package	Parent Node for package information
4	LicensePlate	Carton Identifier from the ship request
4	TrackingNumber	The content of this tag is the tracking number generated for the carton (if the label request was successful). This is stored as a string in the Qbit database, and is used at the Verify Zone (if enabled) to determine whether the package received the correct label.
4	ShipLabel	If the [id] attribute of the Error tag is "0", the content of this tag is a label. If the [id] attribute is "-1", the content is optional, and might represent a printable *error label* instead of a shipping label. The label is a complete ZPL string, Base64-encoded.
4	ReturnLabel	If the [id] attribute of the Error tag is "0", the content of this tag is a return label. The label is a complete ZPL string, Base64-encoded. (Optional)
4	DivertLane	The content of this tag is a divert lane, for package sorting purposes. Valid values are 0 thru 99. (Optional)
4	Published	Enclosing tag for Total. (Optional)
5	Total	The content of this tag is the total charge for the shipment. This is stored as a string in the Qbit database and log file. (Optional)

5 -- Appendix

5.1 Revision History

Date	Revision	Comments
2016-02-12	1.0	initial release
2016-03-02	1.1	added option for simpler XML structure
2017-05-25	1.2	added Sequence number to request payload; removed simpler XML structure
2017-09-10	1.3	changed “QMI software” references to “Qbit software”
2018-03-07	1.4	added ReturnLabel to XML response
2018-08-09	1.5	Changed terminology throughout from “ship request” to “label request”
2020-04-03	1.6	Reorganized content to match JSON version.
2021-10-15	n/a	Changed to revision tracking by date.