Requirements/Design Specification

**Request fields in XML from InfoPro**

**GH 461 – CR 13938**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision** | **Description** | **Author** |
| 5/28/2015 | 1.0 | Initial Version with Requirements | Roger Behm |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

The information furnished herein by Republic Services Inc. is proprietary and confidential to Republic Services Inc. personnel and is not to be duplicated, published, or disclosed to any third party in whole or in part without permission from Republic Services, Inc.

© Copyright 2012, Republic Services Inc. - All rights reserved.

Table of contents

1 Business Requirements/Owner – Brittany Smith 6

2 Assumptions 6

3 Design 6

4 InfoPro/TIBCO/BI Impacts 6

5 Report Changes 6

6 Appendix 6

# Business Requirements/Owner – Brittany Smith

InfoPro has requested several fields that are not coming through the XML that are needed for our July release

# Assumptions

NA

# Design

**Fields Needed**

Scout Route

Enclosure  
Lock (lock type)  
Casters  
requested pick up days  
Owner (Container) RSG  
Driving Distance 0.0  
Customer Site Time (mins)   
Round trip drive time (mins)   
Disposal time (mins)  
Total Time (mins)  
Customer Site Time Override   
Round trip drive time Override(mins)   
Disposal time override   
Additional Paperwork  
Disposal Ticket Signature

These are some of the fields are in the below XML field (I call it a BLOB)   
~~lock~lockAndHasp  
casters~false~  
rolloutFeet~  
~scoutRoute~true~  
isEnclosure~true~

**Design / Development** - From Oracle:  
I would suggest that you create a new line attribute and set it in a dedicated commerce library function on the advanced modify of the action that either submits the order itself (if possible) or one of the earlier actions in the workflow. In this new commerce library, you would simply split the \_config\_attributes system attribute (I would actually suggest the \_config\_attr\_info system attribute instead, since it has more delimiters and is more conducive to this kind of operation), loop over its elements, and write out an XML form of it however you like. Something like this:

ret = “”;  
configArr = split(\_config\_attr\_info, “|^|”);  
for each in confArr {  
eachArr = split(each, “~”);  
name = eachArr[1];  
value = eachArr[2];  
ret = ret + “<” + name + “>” + value + “</” + name “>”;  
}

Obviously be sure to check the position of the name and value pairs in the \_config\_attr\_info string.

Also, I would suggest wrapping the entire return string, which will represent the XML you’re looking for, in CDATA, so that the transaction XML validates.

Also, your configurations are currently small and do not have many attributes, so the psuedocode I suggested will work just fine and quickly. However, when writing very large strings, performance is vastly improved by doing something like this instead:

ret = “”;  
retArr = String[];  
configArr = split(\_config\_attr\_info, “|^|”);  
for each in confArr {  
eachArr = split(each, “~”);  
name = eachArr[1];  
value = eachArr[2];  
append(retArr, “<” + name + “>” + value + “</” + name “>”);  
}  
ret = join(retArr,””);

The above avoids constantly referencing an ever-increasing string and appending to it.

# InfoPro/TIBCO/BI Impacts

New varrible names to be added here:

# Report Changes

NA

# Appendix