



MANUFACTURING LABEL PRINTING SOLUTION

Abstract

As cloud adoption increases across businesses, one of the key complexities is the interaction between cloud-based systems and non-cloud devices and systems. For example, the connectivity between Oracle Fusion and Zebra printers is challenge. There is a need for third-party software to establish communication between cloud and non-cloud infrastructure to achieve the desired result. For example, to print a label after batch generation during work execution, a middleware layer and a third-party software is required to interpret, translate, and print labels with Zebra printers. When labels include context-sensitive images, the task is further complicated.

The ideal manufacturing label printing solution eliminates the need for third-party software and prints directly from Oracle Fusion Manufacturing Cloud to Zebra printers. This solution can also be leveraged to print labels from any other module to any other printer for which Oracle Cloud does not have drivers.

Table of the content

Abstract	1
Conventional Solution	3
Recommended Solution	3
Pre-requisites for the Recommended Solution	5
Solution Execution.....	6
Technical Details of the Solution.....	7
Benefits of the Solution	10
Glossary	11
Resources.....	12
Authors	12

Conventional Solution

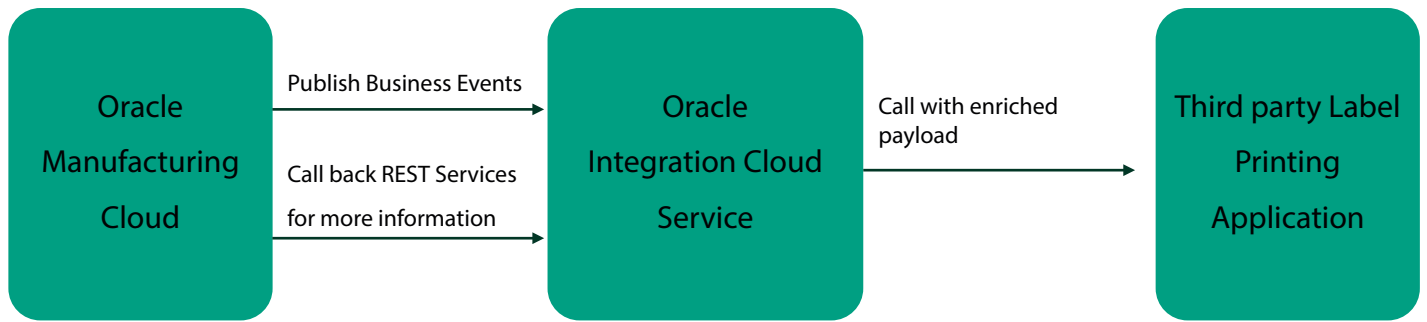


Figure 1 - Conventional solution

The conventional solution for printing manufacturing labels from Oracle Cloud systems to Zebra and other printers requires a third-party software to be bought and maintained. Customers find it challenging to identify compatible software and obtain budget approvals for procurement within project timelines.

This leads to the need for a solution that can print directly from Oracle Cloud onto Zebra printers with some required middleware.

Recommended Solution

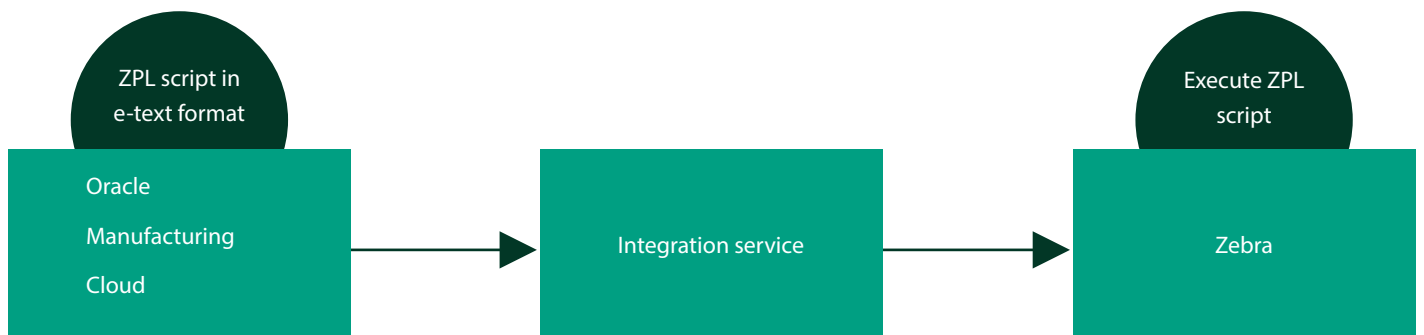


Figure 2 – Recommended solution



Figure 2 depicts the recommended solution. Oracle Cloud needs to generate a Zebra programming language (ZPL) script which is understandable by Zebra printers in e-text format.

The script must contain all required data such as information to be printed on the label, placement of each field on the label, size of the label, name of the image to be printed (stored at item master level), logo information if the label is meant for the external world, and bar code information. The ZPL script must also contain

the name of the printer on which this label is to be printed. This information is sent in e-text format from Oracle to the Zebra printer. When a user completes a transaction in Oracle, Zebra prints the information at specified locations on the label and uses the image name to retrieve the image from its storage and print it on the label.

This solution avoids the use of third-party software and its maintenance and reduces the cost of implementation.

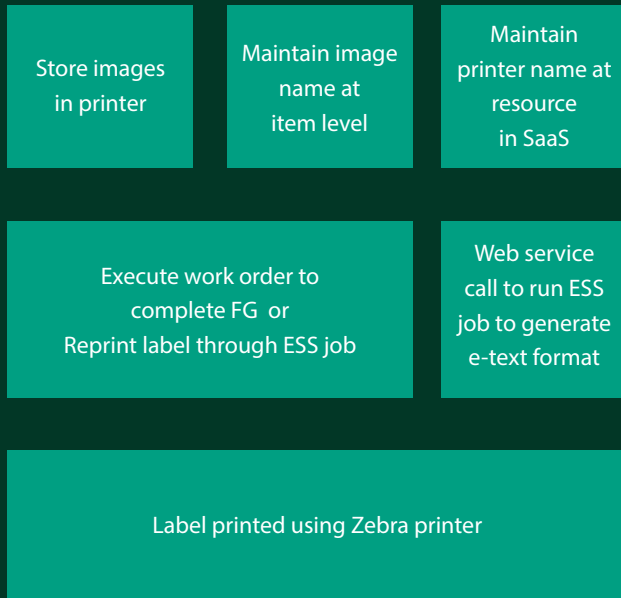


Figure 2.2 - Recommended solution



Pre-requisites for the Recommended Solution

For the recommended solution to work smoothly, there are some pre-requisites:

- Images need to be stored on all printers that are planned to be used
- The name with which the image is stored on the Zebra printer must be maintained at master item level in SaaS (Oracle, in this case) as an attachment

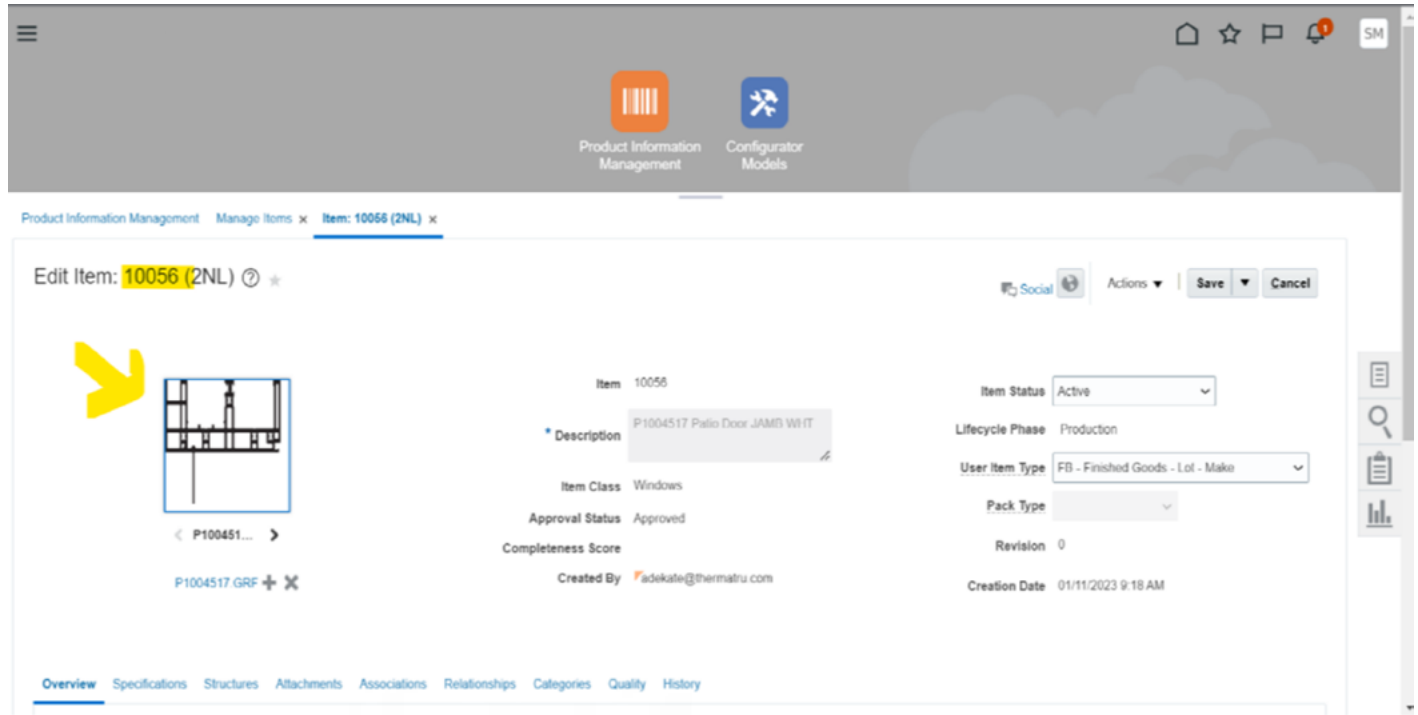


Figure 3 – Product Information Management – Edit item

Further, the following steps need to be taken:

Maintain printer name at resource level



At the SaaS level, maintain the printer name in DFF for equipment resources

Printer registration in SaaS



The printer must be registered in Oracle using the IPP server of the Zebra printer

Lookup to maintain item classes



Maintain a lookup in SaaS to identify the number of labels to be printed for a class of item inventory organization-wise for which a transaction is being executed. The use case in this paper is work order transaction, but lookup can be used in other transaction types such as shipping and receiving where a business event is available. The lookup shown in Figure 4 also consists of label layout information for a specific class of item

▲ FIB_MFG_PRINTER: Lookup Codes

Actions ▾ View ▾ Format ▾ + ✕ 📄 🌐 🗨️ 📄 Detach ↩️ Wrap

Lookup Code	Display Sequence ▲ ▾	Enabled	Start Date	End Date	Meaning
▶ Decking_1NL	10	<input checked="" type="checkbox"/>	mm/dd/yyyy 📅	mm/dd/yyyy 📅	10
▶ Cladding_1NL	20	<input checked="" type="checkbox"/>	mm/dd/yyyy 📅	mm/dd/yyyy 📅	20
▶ DeckingWolf_1NL	30	<input checked="" type="checkbox"/>	mm/dd/yyyy 📅	mm/dd/yyyy 📅	30
▶ Decking_2NL	40	<input checked="" type="checkbox"/>	mm/dd/yyyy 📅	mm/dd/yyyy 📅	70

Figure 4 – FIB_MFG_PRINTER: Lookup Codes

Solution Execution

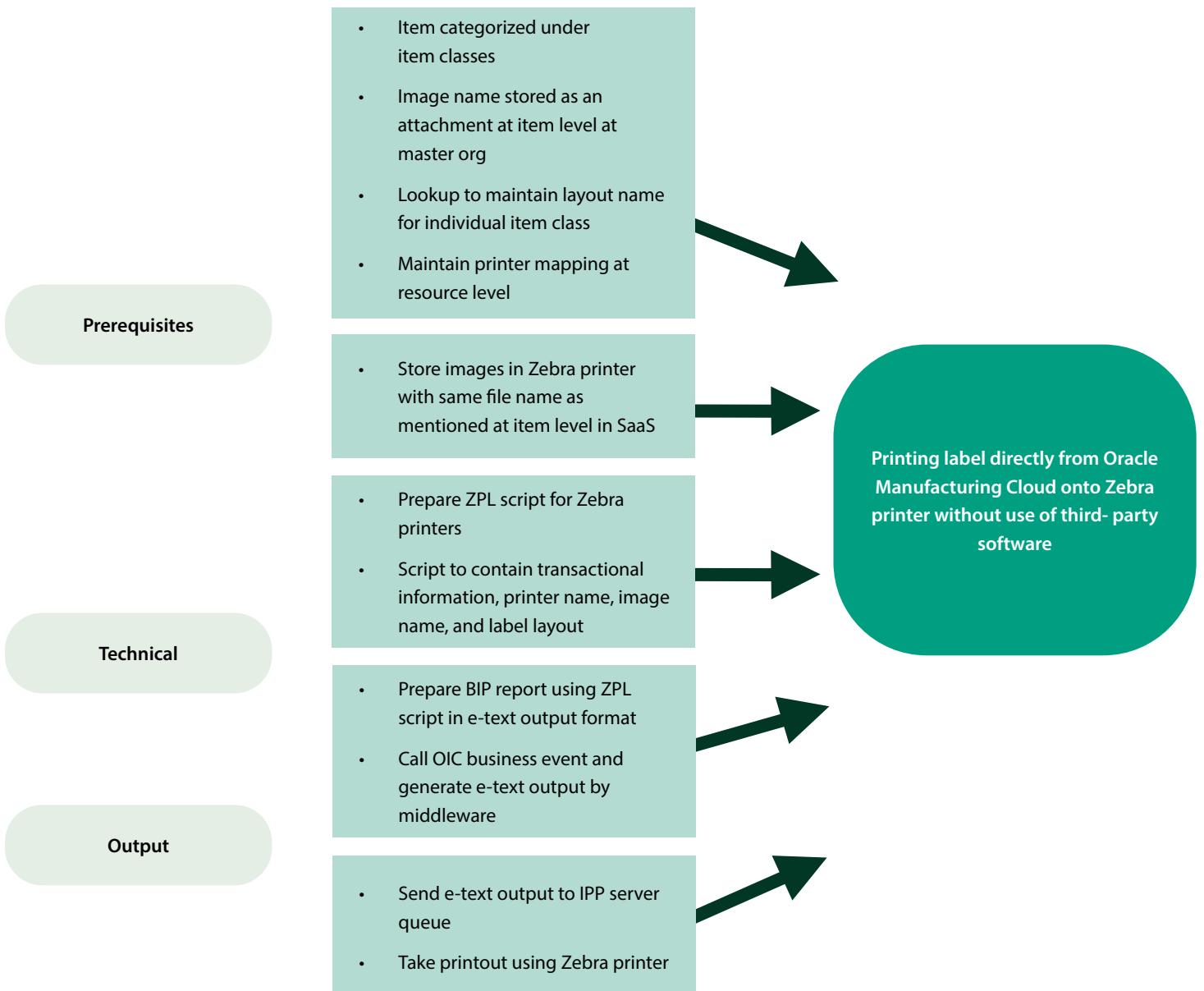


Figure 4 – Detailed solution

If all the pre-requisites are met, when a user executes a transaction (in this case, work order completion), a business event is generated and sent to OIC. The middleware calls the required application programming interface (API) to fetch information about the item, class of item, inventory organization, label layout information from the lookup and the number of labels to be printed, the image name from the item level, and the printer name from the resource level to generate an e-text. The e-text contains all this information and the script sends additional information about placement of transactional information and the font to be used to the Zebra printer. Zebra reads the e-text format script, identifies the printer from which this label needs to be printed, retrieves image information, and prints the required number of labels. Table 1 shows a sample e-text format ZPL script template.

Technical Details of the Solution

<TEMPLATE TYPE>	DELIMITER_BASED
<OUTPUT CHARACTER SET>	iso-8859-1
<NEW RECORD CHARACTER>	Carriage Return

<LEVEL>	G_1		
<MAXIMUM LENGTH>	<FORMAT>	<DATA>	<COMMENTS>
<NEW RECORD>	Label		
<DISPLAY CONDITION>	ITEM_CLASS IN('Decking','PVC Railing','AL Railing','Cladding','Windows')		
20	Alpha	'^XA'	
1	Alpha	chr (10)	
200	Alpha	'^BY4,3.0,114^FT702,126'	
1	Alpha	chr (10)	
200	Alpha	'^BCI,,N,N,,'	
1	Alpha	chr (10)	
200	Alpha	'^FD' LOT_NUMBER '^FS'	
1	Alpha	chr (10)	
100	Alpha	'^BY4,3.0,240^FT262,1402'	
1	Alpha	chr (10)	
200	Alpha	'^BCB,,N,N,,'	
1	Alpha	chr (10)	
200	Alpha	'^FD' LOT_NUMBER '^FS'	
1	Alpha	chr (10)	
200	Alpha	'^BY3.0,3.0,78'	
1	Alpha	chr (10)	
100	Alpha	'^FT426,1404'	
1	Alpha	chr (10)	
100	Alpha	'^BCB,,N,N,,'	
1	Alpha	chr (10)	
100	Alpha	'^FD' ITEM_NUMBER '^FS'	
1	Alpha	chr (10)	
100	Alpha	'^BY3.0,3.0,78'	
1	Alpha	chr (10)	
100	Alpha	'^FT656,1394^BCB,,N,N,,'	

1	Alpha	chr (10)	
100	Alpha	'^AFD' TRANSACTION_QUANTITY '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT473,585^GB178,4,4^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT477,951^GB228,4,4^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT276,262^GB430,4,4^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT526,941^A0B,36,^FH\^FD' USERNAME '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT611,941^A0B,38,^FH\^FD' RESOURCE_NAME '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT644,536^A0B,26,^FH\^FD' TIME '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT162,709^A0B,90,^FH\^FD' LOT_NUMBER '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT426,900^A0B,130,^FH\^FD' ITEM_NUMBER '^FS'	
1	Alpha	chr (10)	
100	Alpha	'^FT556,938^A0B,20,^FH\^FD' USER_NAME '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT800,1394^A0B,28,^FH\^FD' PRODUCT_DESCRIPTION '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT554,552^A0B,22,^FH\^FD' MFG_DATE '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT668,42^A0I,76,^FH\^FD' LOT_NUMBER '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT565,1353^A0B,66,^FH\^FD' TRANSACTION_QUANTITY '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT644,941^A0B,20,^FH\^FD' RESOURCE_ID '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT34,609^A0B,22,^FH\^FD' SERIAL # '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT706,117^A0B,20,^FH\^FD' SERIAL # '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT695,936^A0B,36,100^FH\^FD' FIBERON '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT465,1379^A0B,26,^FH\^FD' PRODUCT # '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT693,1337^A0B,24,^FH\^FD' QUANTITY # '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT613,544^A0B,40,^FH\^FD' TIME '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT524,550^A0B,40,^FH\^FD' MFGDATE '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT761,1392^A0B,39,^FH\^FD' DESCRIPTION '^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT276,1420^GB4,1161,4^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT647,948^GB4,690,4^FS'	

1	Alpha	chr (10)	
400	Alpha	'^FT563,951^GB4,690,4^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT701,1420^GB4,1161,4^FS'	
1	Alpha	chr (10)	
400	Alpha	'^FT473,1418^GB4,1159,4^FS'	
1	Alpha	chr (10)	
200	Alpha	'^PQ1'	
1	Alpha	Chr (10)	
100	Alpha	'^XZ'	
<END LEVEL>		G_1	

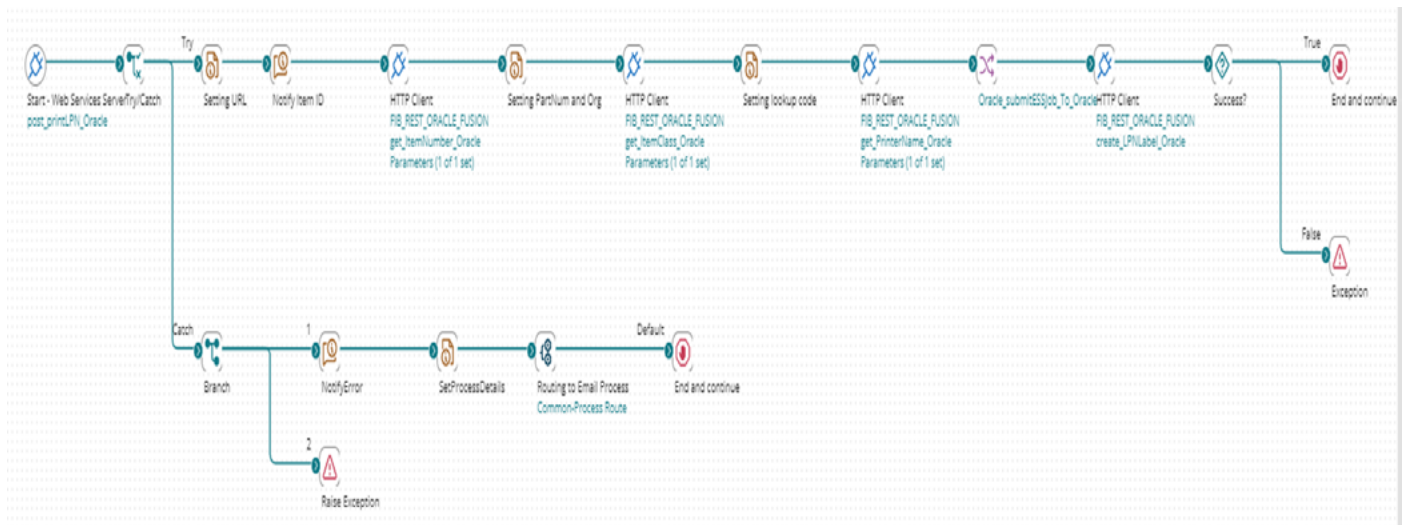
JavaScript at the transaction screen

```

<html>
<button type="submit" onclick="UserAction()">Print</button>
<script type="text/javascript" language="javascript">
function UserAction() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("response").innerHTML = this.responseText;
        }
    };
    xhttp.open("POST", https://fbosboomi.thermatru.com:8444/ws/rest/orcl/test/, true);
    xhttp.setRequestHeader("Content-type", "application/json");
    xhttp.send ();
xhttp.send ();
}
</script >
</html>

```

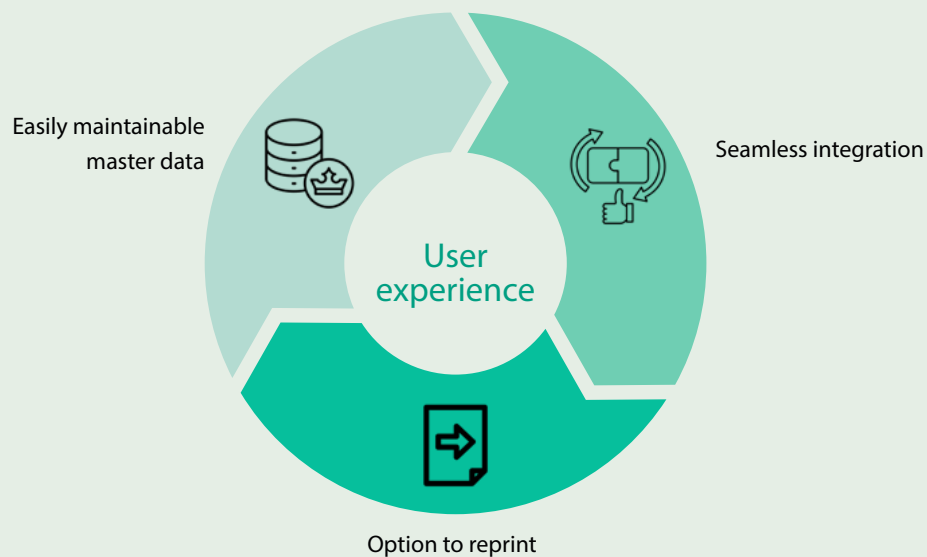
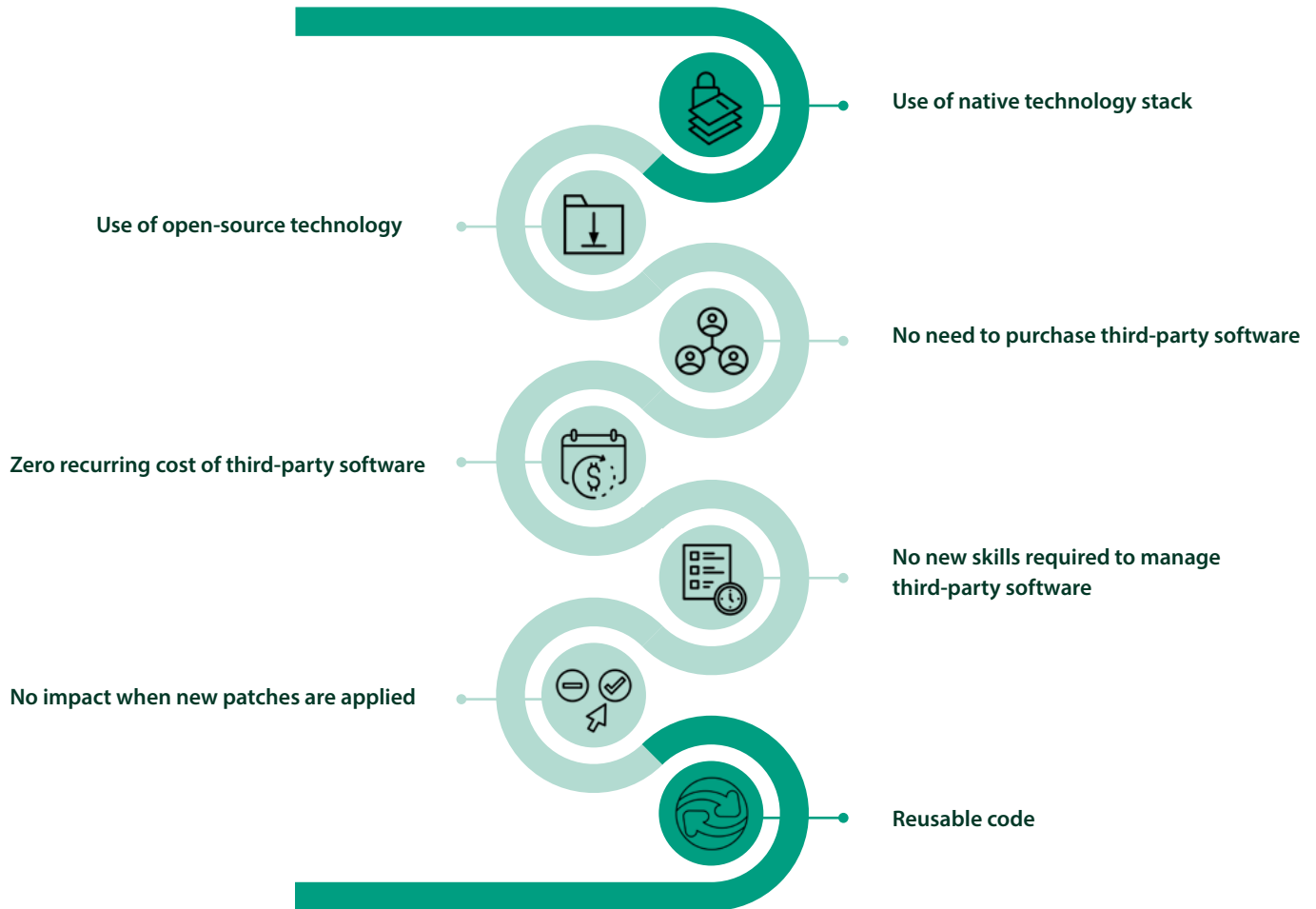
Middleware process



Benefits of the Solution

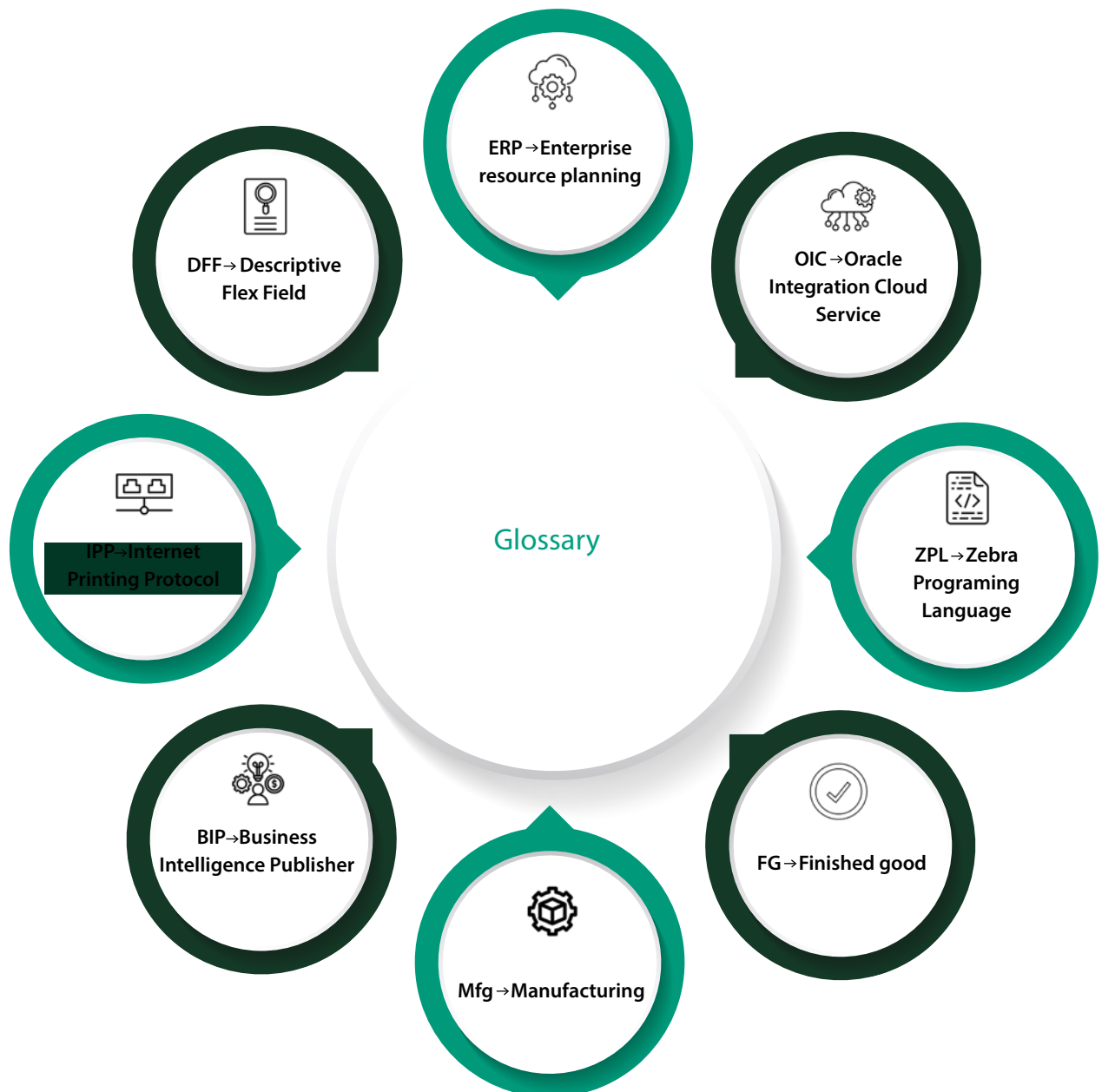
The manufacturing label printing solution presents several benefits over the conventional method followed currently. The solution has the potential to work with printers other than Zebra if Oracle can generate printer-understandable language. Some of the key benefits are:

Technology



Scalability

- Scalable solution
- Can be used in other modules such as shipping and warehouse
- Easy to:
 - o Add new item classes
 - o Change layouts
 - o Change images
 - o Change the number of labels for each item class
 - o Change the printer name as required



Resources

- SR 3-30032840931
 - FA:SCM:MFG: White paper on Label Printing using Oracle Manufacturing Cloud (Doc ID 2471760.1)
- ZPL Command Information and Details

About the Author



Viswanadham Sighakolli,

Principal Consultant

Viswanadham is an Enterprise Solution Architect with expertise in SCM including several Oracle SCM modules. He has extensive experience in designing systems for change management in IT project operations and custom designing solutions (process and product). Viswanadham has in-depth knowledge of business process analysis and design, application-based process reengineering, process optimization, and revenue maximization. He is skilled in performing user acceptance testing (UAT) execution as well as tracking test cases to ensure that applications developed are compliant with pre-set technical specifications.

For more information, contact askus@infosys.com



© 2023 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.