



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



Oracle Payments in Release 12 – Take It To the Bank!

Glen T. Ryen

VP of Technology, Prismo Technologies

glenr@prisiotechologies.com

Phone: (201) 410-2678



103 Carnegie Center, Ste 300

Princeton, NJ 08540



Agenda

- Introduction
- Objectives
- Implementing Payments
- Electronic Payments
- On-site Printed Checks
- Conclusion
- Questions?



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



Introduction – About Me

- Glen T. Ryen
- VP – Technology for Prismo Technologies
- 15+ years with the E-Business Suite
 - Customization, Integration, Implementation
 - Clients large and small, various industries
- 4 years with R12 Payments
- Development background, but also functional and PM



Introduction – About You

- Moving to/considering Oracle Payments?
- Already on...
 - R12?
 - 11.x?
 - 10.x?
- Functional, Technical, Project Management?



Objectives

- Blueprint for bank integration
- High level map, not turn-by-turn directions
- Three areas:
 - Implementing Payments
 - Electronic Payments
 - On-site Printed Checks
- Functional/Project Management vs. Technical
- Overview, then Tips/Best Practices/Lessons Learned



Out of Scope

- Funds Capture functionality
- SRA delivery to Suppliers
- Positive Pay
- Multi-language reporting
- Cash Management payment clearing and bank statements
- Accounts Payable Invoice processing



Implementing Payments

ORACLE Payments

Oracle Payments Setup

Expand All | Collapse All

Focus Task Name	Description	Status	Go To Task
Payments Setup			
Shared Setup			
System Security Options	Set security options for encryption, masking, and credit card control.		
Validations	View validations that can be assigned to formats or disbursement payment methods.		
Formats			
XML Publisher Format Templates	Create and register format templates needed for your payment processing.		
Formats	Define formats, link them to your XML Publisher Templates, and assign validations.		
Payment Systems			
Transmission Configurations	Review transmission protocols and add parameter values to create configurations.		
Payment Systems	Define the external organizations that process your funds capture and disbursement transactions.		
Funds Disbursement Setup			
Payment Methods			
Payment Methods	Define payment methods, rules for their use on documents to be paid, and validations for documents.		
Payment Method Defaulting Rules	Create and maintain rules for how your payment methods should default on documents to be paid.		
Codes			
Bank Instruction Codes	Define codes required by your format specifications for instructions to your bank.		
Delivery Channel Codes	Define codes required by your format specifications for delivery channel instructions.		
Payment Reason Codes	Define codes required by your format specifications for payment reasons.		
Payment Process Profiles	Create profiles for payment processing. The profiles contain all the rules about how payments are created and disbursed.		
Disbursement System Options	Review and set system options used in the disbursement process.		
Funds Capture Setup			
Payment Methods	Review payment methods, and define any needed for bank account transfers.		
Funds Capture Process Profiles	Create profiles for funds capture processing. The profiles contain all the rules for authorization and settlement handling.		
Payees	Define rules for deriving the correct payment system account and funds capture process profile for a transaction.		
Credit Card Brands	Enable credit card brands that you accept.		

Key

- Not Started
- In Progress
- Completed

Task Status

Please note that even if a task status is set to Completed you can always go back to the task and change its status.

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format

* Indicates required field

Code: IBY_PAY_CHK_STANDARD_1_1
* Name: Standard Check Format
Description: Oracle Standard Format Payments
Status: Active
End Date:

Processing Type: Printed
Default Payment Document: Check Stock: A 1448
Payment File: Send to Printer
 Send to File
 Disallow Save/Re-Print
 Automatically Print after Formatting
Default Printer:

Usage Rules | Payment System | **Payment Creation** | Payment Instruction Creation | Payment Instruction Format | Reporting | Additional Information

Usage rules determine when payment process profiles can be assigned for use on documents.

Payment Methods

Payment Method Name
Check

First Party Organizations

Values: All Specify

Currencies

Currency Code	Currency Name
USD	US dollar

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format



ORACLE Payments

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format

* Indicates required field

Code: IBY_PAY_CHK_STANDARD_1_1
* Name: Standard Check Format
Description: Oracle Standard Format Payments
Status: Active
End Date:

Processing Type: Printed
Default Payment Document: Check Stock: A 1448
Payment File: Send to Printer
 Send to File
 Disallow Save/Re-Print
 Automatically Print after Formatting
Default Printer:

Usage Rules | Payment System | **Payment Creation** | Payment Instruction Creation | Payment Instruction Format | Reporting | Additional Information

Usage rules determine when payment process profiles can be assigned for use on documents.

Payment Methods

Payment Method Name
Check

First Party Organizations

Values: All Specify

Currencies

Currency Code	Currency Name
USD	US dollar

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format

Oracle Payments Setup > Payment Process Profiles >
Update Payment Process Profile: Standard Check Format



Implementing Payments

- Key functionality, getting started, and flow
- Project Manager/Functional Implementer
 - Overview
 - Tips and Lessons Learned
- Technical Developer
 - Overview
 - Tips and Lessons Learned



Implementing – Key functionality

- Defaulting
- Validation
- Aggregation
- Formatting
- Transmission



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



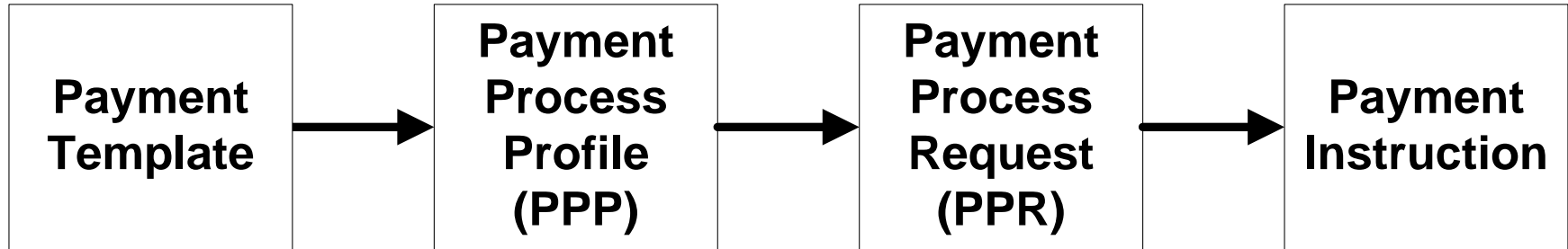
Implementing – Getting Started

- MOS Note #1391460.1 – “R12: Understanding What the Oracle EBS "Payments" (IBY) module Is, and What It Does”
- Payments Implementation Guide
 - Pg. 1-1 – “Overview”
 - Pg. 3.26 – “Understanding Funds Disbursement”
- MOS Note #733573.1 for upgraders
 - New tables for transactions (IBY), banks (CE)
 - Suppliers to the TCA tables



Implementing – Flow

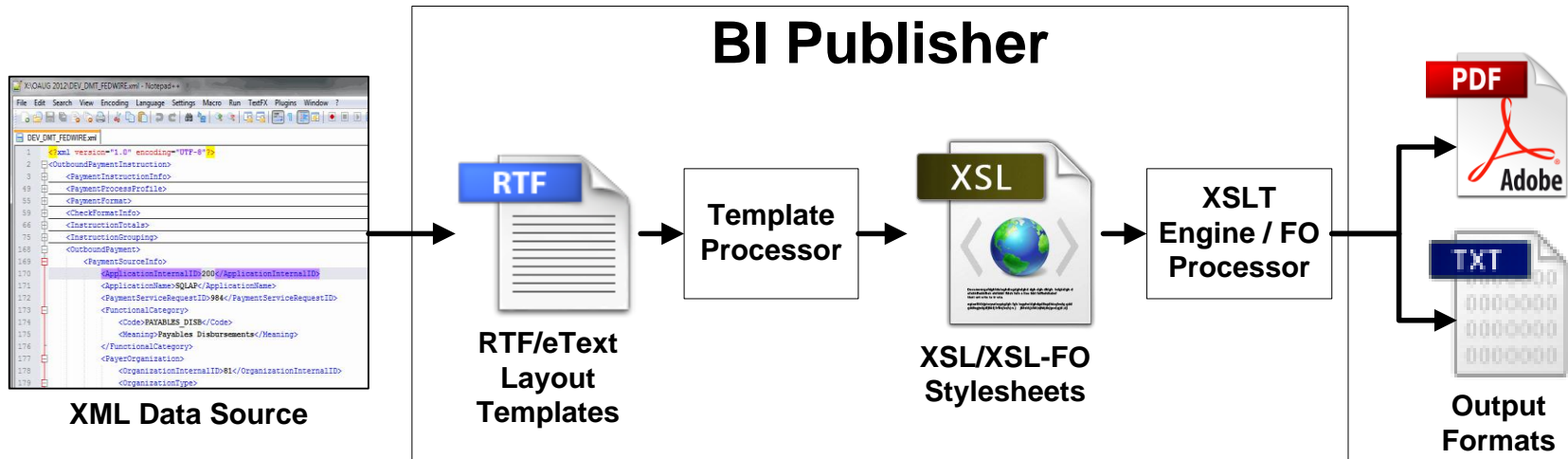
- Know the flow from:





Implementing – Flow (cont.)

- Know how BI Publisher fits in the picture:





PM/Functional – Overview

- What are your security requirements?
 - Electronic, procedural, data, fraud prevention, physical
- What formats do you need?
- What seeded reports must you modify?
- Do you have the right skillsets?



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



PM/Functional – Tips/Lessons

- Analysis/Design
 - Identify the interfaces and templates, nail down the scope
 - Engage your bank, IT, and hosting provider early
 - Get buy-in on the “to-be” state
 - Keep it simple
- Testing
 - As much converted and user-entered data as possible
 - Regression testing of patches
 - Performance testing



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



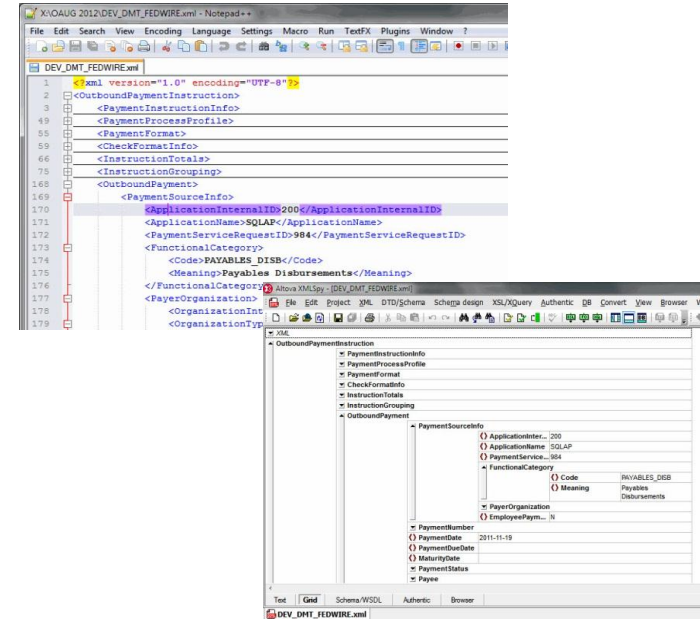
PM/Functional – Tips/Lessons

- User Training
 - Know key concepts:
 - Flow of defaults
 - What Payment Method to use when
 - How Payment Templates relate to Invoices/Suppliers
 - Get them really involved in the testing
- Contingency planning – Expect the best, prepare for the worst
 - Manual checks
 - Bank website for manual wires/ACH's



Technical – Overview

- Get familiar with:
 - XML – Elements, attributes, tags
 - XPath expressions – tree navigation
 - XSLT for eText, XSL-FO for RTF
- Get BI Publisher Desktop
- Get an XML editor
- Know how to code the transmission





Technical – Tips/Lessons Learned

- Get the seeded XML output from your instance
- Extend it if you need to (IBY_FD_EXTRACT_EXT_PUB)
- Code your eText or RTF Layout Template
 - Think “Where am I?” for your XPath expressions
 - XML is case-sensitive
 - Know how to get (and read) the debug information
- Migration with FNDLOAD and XDOLoader



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



Electronic Payments





COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



PM/Functional – Overview

- What are your security requirements?
 - Consensus among bank, IT, and hosting provider
 - Start the discussions early, allow time for:
 - Getting/exchanging keys, certificates, new software
 - Opening firewall ports, establishing a VPN
 - Procuring/adding a server to a DMZ
- Development/test instances – Yours and the banks
- Separate Remittance Advices?



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



PM/Functional – Tips/Lessons

- Set up your seeded User-defined Validations:
 - By Payment Method, on individual fields
 - ABA Number must be nine digits, SWIFT BIC at least eight characters, Currency must be “EUR”, etc.
- (No. of Accts) x (No. of Payment Types) x (No. of Currencies)
= A lot of scenarios to test
- Testing covers your process, the bank’s process, and your data



Technical – Overview

- eText Layout Templates -> XSL
 - Install the Template Viewer application
 - Get the XML, modify as needed
 - Don't start from scratch
 - Unit test locally
- Transmission logic
 - Payment Servlet (Java, Perl)
 - Server-side script (UNIX shell)

XDD file name: IBYDE_NIUS_en.rtf
(APXNACHA.rtf) Date: 3/7/2006

**Mapping of Payment Format
US NACHA CCD Format**

Format Setup:

Hint: Define formatting options...

<TEMPLATE TYPE>	FIXED_POSITION_BASED
<OUTPUT CHARACTER SET>	iso-8859-1
<CASE CONVERSION>	Upper
<NEW RECORD CHARACTER>	Carriage Return

Sequences:

Hint: Define sequence generators...

<DEFINE SEQUENCE>	PaymentSeq
<START AT LEVEL>	OutboundPaymentInstructionLoc
<INCREMENT BASIS>	LEVEL

Format Data Records:

*Hint: This is the body of the format. Define your format records here.
Create one table for each record or group of records that are at the same level.*

<BEGIN FILLER BLOCK>	AllRecordsBlock
<FILLER CHARACTER>	9
<BLOCK SIZE>	10

<LEVEL>	<POSITION>	<LENGTH>	OutboundPaymentInstruction	<COMMENTS>
<NEW RECORD>	<FORMAL>	<PAD>	<DATA>	
			FILE_HEADER	
1	1	Number	L, '0'	Record Type Code
2	2	Number	L, '0'	Priority Code
4	1	Alpha	R, ' '	Immediate Destination: the first position is a blank
5	9	Number	S, '0'	Immediate Destination: the second to the tenth position of this field is the ABA routing number of the receiving bank of this payment file. The ABA routing number is a nine digit number composed of three parts. The first four digits are the Federal Reserve Routing Symbol, the next four digits are the ABA Institution Identifier, and the last one digit is the check digit.
14	1	Number	L, '0'	Immediate Origin: mutually defined - ANSI one-digit IOC
15	9	Alpha	L, '0'	REPLACE [InstructionGrouping/Payer/LegalEntityRegistrationNumber, ' ']
24	6	Date	SYSDATE	File Creation Date
30	4	Date	SYSDATE	File Creation Time (Military Time)
24	1	Alpha	R, ' '	DDCODE [SEQUENCE NUMBER (F1056)] 1=42, 2=08, 3=26, 4=30, 5=38



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY

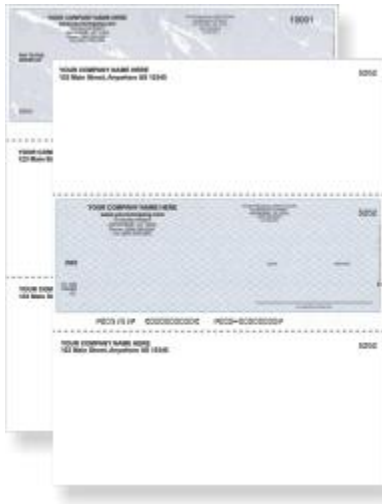


Technical – Tips/Lessons Learned

- Start with a seeded template (e.g., NACHA)
- Mock up the data and preview locally in the Template Viewer
- Know your commands, expressions, and operators:
 - <TEMPLATE TYPE>
 - <DEFINE LEVEL>
 - <DEFINE SEQUENCE>
 - <DISPLAY CONDITION>
 - <BEGIN FILLER BLOCK>
 - <DEFINE CONCATENATION>
 - IF-THEN-ELSIF-END IF
 - COUNT
 - SUM
 - DECODE
 - SUBSTR
 - Etc.



On-site Printed Checks



Sample Company 1215
100 Main St.
Bridgeport, CT 06604

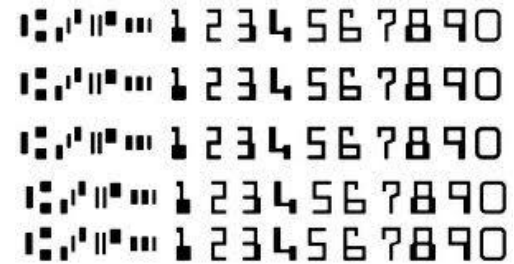
DATE 04/25/yyyy

PAY TO THE ORDER OF ABC Company \$ 53.50
Fifty Three ⁵⁰/₁₀₀ DOLLARS

MEMO Sample

001215 221142156 064 70025291 0000005350

Auxiliary On-Us Field Transit Number Field Account Number or On-Us Field Amount Field





COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



PM/Functional – Overview

- What are your security requirements?
 - Blank stock or Pre-MICR'ed?
 - Printed signatures or manual? Image security?
 - Are your auditors okay with the Payables Manager responsibility running PPR's start to finish?
- Physical security of the printer and stock?



PM/Functional – Tips/Lessons

- Allow time to procure and setup the check printer
- Plan for more than one round of MICR acceptance testing
 - Lots of variables can affect placement
 - Very tight tolerances



Technical – Overview

- RTF template -> XSL-FO
- Basic steps
- Features
 - Custom Fonts
 - Fixed Row Enumeration
 - Conditional Formatting

V_top FE_Pmt(Sort) V_inmr_group FE_inmr_group IF_room_on_page V_First_rec

Vision Operations CHECK NUMBER: 999999
CHECK DATE: Payment Date

INVOICE NUMBER	DATE	DESCRIPTION	GROSS AMOUNT	DEDUCTIONS	AMOUNT PAID
FEInv Num filer_chk	Doc Date	Document Description	Gross Amt	Discount Amt	Pmt AmtEFE End_filer
PAGE TOTAL			Total Gross Amt	Total Disc	Total Pmt Amt

+ Cboose2W_First_page

Your Bank, N.A.
New York, NY

12-345
678

CHECK NO. 999999

Vision Operations
AddressLine1
City, State Zip

DATE
Pmt Date

AMOUNT
*** Check Amt

PAY
Check Amount in Words

TO THE ORDER OF:

PAYEE NAME
ADDRESS LINE 1
ADDRESS LINE 2
CITY, STATE ZIP
If_Non_US COUNTRYNAME Elf_Non_US

IF_over_25k Underline Elf_over_25k

VOID
Two Signatures Required for Amounts over \$25,000.00

EW_First_page
Otherwise2

⑈999999⑈ ⑆123456789⑆ 123456789012⑈



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



Technical – Tips/Lessons Learned

- Pre-Printed Stock
 - Less flexible and secure, but easier than printing MICR
 - Fixed-height single row table for stub, repeating table inside
- Testing
 - Bank will specify minimum for MICR acceptance
 - All your scenarios – Number of address lines, field lengths, conditional logic threshold amounts
 - Varying number of Invoices (and overflow documents)



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



Conclusion

- Lots of functionality, lots of capabilities, lots to learn
- Start with the blueprint, follow the pointers
 - References – Blogs, sites, MOS notes, Oracle docs
 - Useful scripts
- The good news? Oracle Fusion Payments will look very familiar:
 - PPP's, PPR's, Payment Methods, Validations
 - Layout Templates tied to Payment Formats
 - Same XML extract structure, same package for extending it



COLLABORATE 12

TECHNOLOGY AND APPLICATIONS FORUM
FOR THE ORACLE COMMUNITY



Questions?

