

“Walk the Line” between Shared and Divisional AP

Bev Fess

Kathy Skilton

Jean West

Fran Brady

The Sherwin-Williams Company

Chris Tyner

Moreland Partners, Inc.

Abstract

Sherwin-Williams had the challenge of transforming a divisional AP environment onto an Oracle platform that would allow a shared service approach. Rather than force all divisions to immediately begin living in a single operating unit together, the implementation team rolled out some techniques to ease the transition. This paper will cover how Custom Responsibilities, Folders, Forms Personalizations and other enhancements were used to support unique Divisional requirements while at the same time moving the company into a new common AP instance.

Introduction

Founded in 1866, The Sherwin-Williams Company is a world leader in the manufacture, development, distribution and sale of coatings and related products to professional, industrial, commercial and retail customers. The Company manufactures and sells products under well-known brand names such as Sherwin-Williams, Dutch Boy, Krylon, Minwax, Thompson’s Water Seal, Duron, M.A.B. and many more. Sherwin-Williams branded products are sold exclusively through a chain of 3,226 company-operated stores, while the company’s other brands are sold through leading mass merchandisers, home centers, independent paint dealers, hardware stores, automotive retailers and industrial distributors throughout North America. The Sherwin-Williams Global Group distributes a wide range of products in more than 30 countries around the world.

History of the Project

Before the FACTS (Financial Analysis, Control & Tracking System) project that included the implementation of Oracle AP, the Sherwin-Williams organization was highly divisionalized. Very little AP functionality was performed in a common system, or with a common process. About the only shared resource was a specialized printing group that could process payment files from multiple sources and print, stuff & mail paper checks. However, separate checks for the same legal entity still existed for the two internal systems.

Within the original segregated systems, there existed 50+ “pay entities” used to track unique combinations of divisions, legal entities, bank accounts, and check stocks. Each pay entity could have its own payment schedule. The various divisions used multiple AP Liability accounts which led to increased effort in total AP liability tracking in the GL.

There was a history of issues with the multiple systems. Any effort to sum up AP activity across the entire organization was lengthy and imprecise. For example, it would require many man hours to produce reports like a “Top 100 Vendors List” by spend dollars. Year-end 1099 reporting was a time consuming manual effort to consolidate data from the two systems. The same vendor could be paid with different terms, on a different payment schedule (daily verses weekly) and by check in one system, ACH in the other. There was no consistency.

Daily ledger extracts in the system were performed 3 times a month in one system, 4 times a month in the other. The multiple systems each maintained their own closing schedules.

Sherwin-Williams embarked on a multi-phased implementation of the Oracle e-Business Suite 11.5.10 Financials modules. The initial phase saw the entire organization go live on the GL module. The subsequent phase included the AP module, and that is the topic of this paper.

Oracle Financial implementation teams in the past have struggled with the important organization definition decisions at the beginning of a project. Should an enterprise with multiple divisions set up separate operating units to support them all? Is the data segregation in a multi-org system a benefit? Are there any other ways to preserve divisional independence while everyone shares a single operating unit? These concepts were considered as a part of the Sherwin-Williams implementation planning. A decision was made to create one operating unit per set of books, while at the same time supporting certain divisional requirements.

Specific Benefits of the Project

The Sherwin-Williams organization chose to move to a single operating unit in Oracle AP for a variety of benefits. The primary benefits were summarized by a list of “Ones” that clearly illustrated the major differences of a shared services approach.

The “Ones” (Original List)

1. One Vendor Master
2. One Set of Payment Terms
3. One Operating Unit (One for US and One for Canada)
4. One Check Run by Legal Entity
5. One Closing Schedule

Additional “Ones”

1. One Vendor Master Maintenance Team
2. One Check Processing Team
3. One consolidated Cash forecast
4. Centralized phone number on all checks.
5. One PO Box for returned checks.

One shared operating unit in the Oracle database will more easily allow for:

- Vendor spend analysis
- Consolidation of vendor's invoices within a legal entity into one payment
- Reduced debit balances
- Consolidated bank accounts (now 6 accounts instead of 9)
- Reduced cost of check stock, envelopes, postage. The ability to group mailing by weight for reduced postage rates
- Consolidated bank reconciliation (now 1 person instead of 3)
- Simplified Minority Reporting
- Better control over duplicate payments

Given that there had been multiple AP systems with segregated databases, a “duplicate invoice check” between divisions was required by Internal Audit. This required reports run from two different systems to be combined on one report based on multiple criteria/scenarios. The need for any sort of a duplicate invoice check would be reduced in a common system, with duplicates occurring with different sites being used. Duplicates are now much easier to uncover.

Additional benefits are:

- End-of-year process is simplified
- Reduced IT cost of processing, upgrading and enhancing two systems
- When an eAM (Plant Maintenance Scheduling, Purchasing & Payable) project goes live for the Paint & Coatings Division, the purchasing group processes for vendor maintenance will also be centralized

Plan for Implementation

Some divisional requirements had to be maintained, even though the Oracle system would have just one operating unit per set of books. All U.S. data would be combined in one operating unit, but Sherwin-Williams still had to support some divisional concepts.

One requirement was that certain tax legal entities would have to be supported by separate bank accounts and check stock. These entities had mainly been created as a result of past acquisitions and historic divisional operations.

Organizational Techniques

A centralized Vendor Master Maintenance team was essential for keeping control over the new, shared vendor data. The VMM team, as they are known, was set up with the ground rules that only they would be allowed to maintain & add records. This will include the purchasing information for the PO module that is soon to be implemented. Another helpful technique was to set up a common VMM Team email address at Sherwin-Williams to handle requests, along with an ACH reject mailbox.

Despite the fact a common AP system was implemented, there remained a host of legacy systems across the divisions that would need to feed invoice information to Oracle. There are 27 source system feeds that create daily invoice batches, and these would need to indicate sending system, source division, and other important info. The AP feeds and manually entered invoices that define invoice batches would include a divisional prefix in the batch naming standard. These batch names would then follow the invoices and are printed on all payment remittances. For AP support purposes, one phone number would be published on the check for the vendors, and the calls would be routed to the appropriate Division based on the Invoice Batch Prefix.

A standardized weekly pay schedule was implemented. Each pay group was used to ensure the correct check stock was used for a legal entity, but the cycles were the same across all groups. Standard payments were made once a week and certain vendors were flagged for "special handling" which meant that their payment cycles were run on a nightly basis. For these vendors, the day their invoices came due, they were selected in the nightly payment cycle. One more Oracle feature used in Check processing was Pay Priorities. Pay priorities could flag special handling pay groups. Checks in these pay groups are returned to the divisions for a manual processing (such as adding additional remittance documentation and enclosures). In the end, there were still many combinations of pay criteria, but all of the payment batches could be setup once in a common system and easily monitored from then on.

The biggest external impact is that vendors that do business within one Sherwin-Williams legal entity, but across Sherwin-Williams divisions, will receive a single payment/check. There are only a couple of legal entities that must remain on separate check stock/runs. In any event, there is the capability to easily scan across the entire US enterprise and see all AP activity.

System Techniques

The implementation team came up with several system enhancements and extensions for the Oracle AP module to support the divisional use of a system with a single operating unit.

- 1.

One technique for the separation of division data was through the use of distinct liability accounts on the Invoice Batches. Invoices batches were loaded through both manual entry and interfaced "feeder" systems. These feeder systems contained fully matched invoices, ready to be validated and paid.

Batch Name	Date	Liability Account	Pay Group	Invoice Type	Payment Terms
AUMAN20071207LBRF	07-DEC-2007	C101-1200005-43RF99-0000-0000	AUTO	Standard	
AUMAN20071207LBRF1	07-DEC-2007	C101-1200005-43RF99-0000-0000	AUTO	Standard	
CNMAN20071207DJN-1	07-DEC-2007	B100-1200005-CAPA99-0000-0000	SW	Standard	
CNMAN20071207DJN-2	07-DEC-2007	B100-1200005-CAPA99-0000-0000	BRUD	Standard	
CNMAN20071207DJN-3	07-DEC-2007	B150-1200005-CAT000-0000-0000	CTS	Standard	
CNMAN20071207DSB-1	07-DEC-2007	B100-1200005-CA0400-0000-0000	SW CONS SPEC HAN	Standard	NET IMMEDIATE

The invoice batches had a naming standard with the first two characters indicating the division. But as batch names are free-form fields, the division segment on the Liability account could more accurately reveal which division owned the batch and invoices. In addition, each division will use slightly different values in the natural account segment for the AP Trade liability account.

The Batch liability account would be restricted and validated through a combination of Forms Personalization and a PL/SQL stored procedure that can be called to enforce the correct Batch Name, division segment, and liability natural account. This ensured the correct divisional liability, across US and CAN sets of books, was controlled in the operating unit.

Batch Name	Date	Liability Account	Pay Group	Invoice Type	Payment Terms
AUMAN20071207LBRF	07-DEC-2007	C101-1200005-43RF99-0000-0000	AUTO	Standard	
AUMAN20071207LBRF1	07-DEC-2007	C101-1200005-43RF99-0000-0000	AUTO	Standard	
CNMAN20071207DJN-1	07-DEC-2007	B100-1200005-CAPA99-0000-0000	SW	Standard	
CNMAN20071207DJN-2	07-DEC-2007	B100-1200005-CAPA99-0000-0000	BRUD	Standard	
CNMAN20071207DJN-3	07-DEC-2007	B150-1200005-CAT000-0000-0000	CTS	Standard	
CNMAN20071207ABC	20-FEB-2008	C101-1200005-43RF99-0000-0000	CTS	Standard	

Error

The Pay Group is not valid for the selected Liability Account Division.

OK

The Forms Personalization used several elements to ensure data integrity when users are creating Invoice batches online through the forms. Every type of action that a user may take on a new or existing batch must be processed through the Forms Personalization logic. All rules are applied at a site level.

Form Personalizations (Invoice Batches)

Function Name: **AP_APXINWKB_BATCHES** Form Name: **APXINWKB** Debug Mode: **Off**

Seq	Description	Level	Enabled
20	SWC Batch: Ensure Liability Account is populated if required	Function	<input checked="" type="checkbox"/>
25	SWC Batch: Ensure Pay Group is populated if required	Function	<input checked="" type="checkbox"/>
30	SWC Batch: Err when saving 'Changed' records if Liability Account has no valid Pay Group	Function	<input checked="" type="checkbox"/>
35	SWC Batch: Err when saving 'Changed' records if Pay Group is invalid for Liability Account	Function	<input checked="" type="checkbox"/>

Condition **Actions**

Trigger Event: **WHEN_VALIDATE_RECORD** (You can enter additional event names.)

Trigger Object: **BAT_SUM_FOLDER**

Condition: `(:SYSTEM.FORM_STATUS = 'CHANGED') AND (SWC_AP_FORM_PER_UTL.pay_group_is_valid (:BAT_SUM_FOLDER.PAY_GROUP_LOOKUP_CODE, :BAT_SUM_FOLDER.LIABILITY_ACCOUNT) = 'N')`

Processing Mode: **Not in Enter-Query Mode**

Context

Level	Value
Site	

Insert 'Get' Expression... Insert Item Value... Validate Apply Now

The Condition section of the rule calls a custom PL/SQL package to provide the flexible rules check needed. If the result of the SWC_AP_FORM_PER_UTL.pay_group_is_valid function returns an 'N', then the error message is raised in the online form and a user cannot save the record.

The same logic in the Forms Personalizations is applied to the custom processing program that serves as the front end to populate the Invoice Open Interface table. In this way, feeder systems must follow the same validation rules.

2.

The vendors had a new custom KFF, loaded into a single DFF, implemented at the site level. This would continue to indicate which sites were associated with which divisions. Additional DFFs were defined to store the legacy vendor numbers (now moved to the site level) in case any cross-reference or investigation was needed.

The screenshot shows the Oracle Supplier Sites form for site CHICAGO001. The SWC Vendor Site Usages dialog box is open, showing the following settings:

- Consumer Usage: No
- Corporate Usage: No
- Auto Usage: Yes
- CTS Usage: No
- Stores Usage: Yes

Other fields in the dialog include Stores Supplier Number (0000929433) and Site Usages (N.N.Y.N.Y).

A custom Vendor Inquiry form was used to ease the transition. The legacy feeder systems were scheduled to slowly switch over to the new Oracle Site numbers. The site numbers are displayed in a DFF, and are visible in a custom Inquiry form.

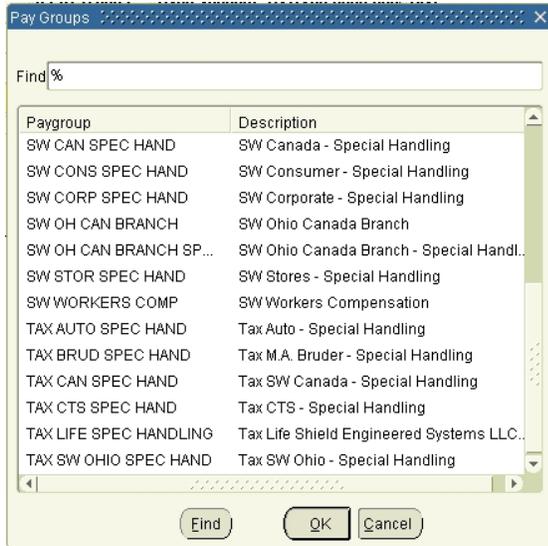
The screenshot shows the Oracle Supplier Sites form with a table of site data. The table has the following columns: Supplier Site Name, Supplier Site ID, Legacy Number, Stores Number, Site Inactive Date, Supplier Name, Supplier Number, and Supplier ID.

Supplier Site Name	Supplier Site ID	Legacy Number	Stores Number	Site Inactive Date	Supplier Name	Supplier Number	Supplier ID
CHICAGO001	29154		0000929433		3M	67797	68796
CHICAGO002	18442	683510			3M	67797	68796
CHICAGO003	19353	929433			3M	67797	68796

For certain vendors, several divisions all use the same site. During the initial conversion of vendor data, the only thing stopping the collapsing of these shared sites to the minimum number was the possibility of different terms across divisions. Many of these terms were contractual and Sherwin-Williams was unable to renegotiate these until after a contract has ended.

3.

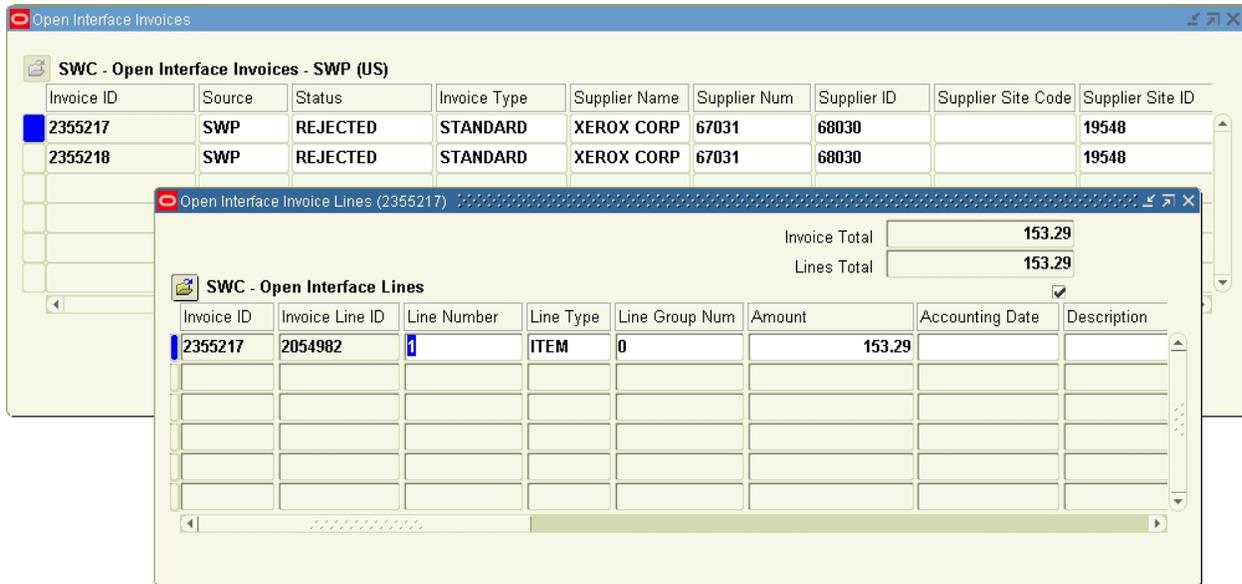
Pay Groups were another technique used to manage AP data that was associated not just in different divisions, but also in different legal entities. Given the Tax Identification numbers of various business entities under the Corporate umbrella, the AP data would have to be tracked separately in order to generate different payment documents.



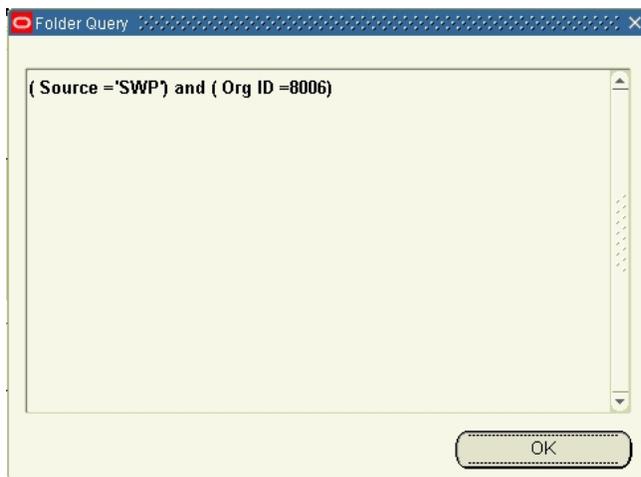
The task of generating over 5,000 payment documents per week necessitated an efficient, shared printing group. This had already existed before the Oracle implementation. There was a need for a variety of check stocks to be used in the common print

4.

Folders were used to limit the data that a particular user has access to view and correct. One important place that a divisional focus is needed is in the processing of “feeder” data into AP through the Invoice Open Interface. If some data does error out, then the people correcting the data want to only focus on their division and feeder system. The users that manage the errors resulting from the import of feeder files have specific folders that limit the data displayed on the Open Interface form.



When these folders are defined, they have the “Include Query” box checked on to lock in the base query information for the folder. In this way, additional queries only add to the restrictions when viewing the data on the form. If a user were to select the View Query option from the Folders menu for the Invoices folder, they would see in (for the example above):



This user/responsibility level does not have the ability to alter the folder or choose another folder. This is accomplished by the setting the Behavior Mode to “Restrict fields and folder functions” on the Administer Folders form when attaching the folder to the user or Responsibility.

5.

Discounts were accounted for differently across the divisions. Unfortunately, the common AP instance only let Sherwin-Williams choose a single Discount setting in the Payment tab of the Payables Options. Virtual Trader, a 3rd party system, has been employed to perform account transformations, such as allocating discounts to one natural account per division. Also, one division wanted the discounts to be recorded on the original cost center from the AP transaction, while another division wanted to use a common cost center.

The AP Discount transactions were updated using the Virtual Trader tool, which allows users to define complex rules for manipulating accounting transactions. The tool will intercept data as it passes through the GL Interface table, apply the custom Sherwin-Williams rules, and then send the data through for Journal Import into the GL module.

Lessons Learned

A successful implementation could only be achieved when the FACTS team included Divisional personnel (from all levels) early on. This provides for more positive buy in & acceptance of change. Also, all of the divisional nuances could be identified and understood quickly by the common implementation team. When the FACTS AP team was identified and assembled, a Supervisor/Super User from each division was selected so all division needs were addressed. The team members had to have intimate knowledge of their respective divisions systems, operations, and business model.

It is important to have strong, experienced AP Functional team members and AP business analysts that supported the prior AP systems, along with individuals that were not afraid of change. This combination, along with management support and a “short” deadline, allowed us to achieve significant change during the implementation.

No one division determined the team’s decision. All decisions were discussed and the team came to a consensus. In the case of multiple requests and needs, divisions were asked what they could and could not live without.

Conclusion

Sherwin-Williams has a defined divisional structure that runs along its lines of business. The AP business area, however, had an opportunity for significant improvements through a move to a shared services environment. The ability to make enterprise business decisions, and support common functions had been a problem with the past systems environment. The ability to keep users focused on their divisional business, through a variety of techniques and tools, was the final piece of the implementation puzzle.

Since the August 2007 go live, we have spent the last six months enhancing our divisional needs, refining our reports, and writing Discoverer reports for end users. The Divisions have learned to work within the Oracle system. Each of the team members has found new roles within the corporation. Some went back to their respective divisions, others moved into enhanced roles and some into new divisions. The “view was definitely worth the climb.”

The “SHARED” concept of the FACTS Payables system is a change to the current Sherwin-Williams paradigm. All previously separate divisions will need to work together to achieve continued success on the implementation.

Contributors to this implementation, and the concepts included in this document included:

Bev Fess, Fran Brady, Kathy Skilton, Mary Travagliante, Jean West, Joe Igusky, Rick Calich of The Sherwin-Williams Company

Chris Tyner of Moreland Partners, Inc.

Jonathan Osso of IBM