Seven Strategic Mistakes Clients Make Implementing SRM

(and How to Avoid Them)

By Jim Stewart, Equity Technology Group

Editor's Note: *For our first piece on Supplier Relationship Management, we asked SRM expert Jim Stewart to move beyond the SRM buzz and write about actual SRM project mistakes and how to avoid them. The author of The SAP® SRM Advanced EBP Cookbook, Jim has seen SRM in action on multiple projects and lived to tell. As promised, he put together an entertaining and instructive guide that should help both new and mature SRM projects. In this piece, Jim divides SRM project miscues into three different categories: strategic mistakes, HR and staffing mistakes, and implementation-related missteps. More importantly, he explains how to avoid these problems. One key point we took away from this piece: implementing SRM is not going to work if you use "SOS," or "same old sourcing" methods. From e-procurement to live auctions, SRM is definitely not "sourcing as usual." The good news: the change in business processes required by SRM has a direct impact on the bottom line. Jim tells us that he has seen a measurable return on investment on all of his SRM projects. He's glad to have the chance to share the secrets of SRM success with SAPtips readers.*

Introduction

Supplier Relationship Management (SRM) is a relatively new product offering from SAP. Out of an installed base of some twenty thousand customers worldwide for the R/3 Enterprise product, I have read that to date about forty or fifty companies have installed SRM. Due to its user-friendly interface and massive potential to save companies money, I think this product is poised for rapid growth worldwide. I also think that it is important to get "beyond the brochure" and get a better understanding of how the product should be used.

This paper will focus on my experiences on very large Supplier Relationship Management implementations and the various strategies that organizations have employed in the installation and use of the software. The title of the paper may sound a bit negative, especially given that all of my SRM implementations were considered great successes organizationally. But clearly, every project has its lessons learned, and I hope that by sharing these examples, SRM projects will be planned and executed more efficiently as a whole. I wrote this paper for SAPtips readers as a way to draw out some of the key points from my recently-released book *The SAP SRM Advanced EBP Cookbook*. Readers looking for more detail on SRM may want to give my book a closer look – it is available for sale at http://www.srmcookbook.com/.

As of this writing, the SRM product is currently on release 5.0. There has been some confusion about the SRM release numbering, since SRM and EBP were on different version numbers, but with SRM server 5.0, I believe SAP will keep the version numbers the same. This being said, the SRM product is really a suite of related products, anchored by the EBP (Enterprise Buyer Professional) e-procurement product. But while EBP may be the most mature of the SRM products, the deeper ROI benefits will come from some of the more sophisticated functionality built into advanced features like the sourcing cockpit, live auction and bidding, and supplier enablement.

The sourcing cockpit will allow a group of buyers to perform advanced sourcing activities all from one location, sending requirements out for bid, or directly to purchase order. Live auction and bidding implements a rich set of features that corresponds to two very common and forward looking procurement activities. First, you can conduct live reverse auction events where bidders compete for your business in an eBay-style auction environment. Second, you can conduct private, blind-bid invitations to receive price quotes from bidding vendors. Another important area in SRM is the supplier enablement component, which is commonly called SUS. SUS extends a single point of interaction to your vendors, who can in turn manage their own data and submit bid responses, advance shipment notifications, and service invoices.

It’s easy to underestimate the value of a good strategy on a project, and somehow in our task-oriented workplaces, setting aside time for strategic planning often gets overlooked. As an implementation consultant, I’m very task oriented. I realize that it is hard to take a step back and think, “Exactly why are we implementing SRM?” But I have to say that the success of an SRM implementation depends largely upon the quality of the strategic thinking that goes into the planning of the implementation. If the strategic thinking is done properly, SRM can be used to achieve savings across all functional areas of the business, continually improving your procurement activities. It is this constantly improving, cross-functional cost savings aspect of the SRM product that makes it especially well-suited to act as a tool to help businesses make decisions that span all of a company’s products and all of a company's business units.

In this paper, I’ve divided the most common SRM mistakes into three categories: –

- First, “Strategy,” or problems with the underlying assumptions about the software and how it should function in the organization. This section should help you formulate better deployment strategies and help fit SRM into the “big picture” of your company and your procurement strategy in general.
- Second, the “HR and Staffing” section describes common problems associated with resource requirements and the unique human resource elements of SRM projects. The HR and staffing section will help you understand more about the common mistakes associated with staffing SRM projects, and help you avoid them.
- Finally, there is a class of mistakes that fall under “Implementation” issues such as which components should be installed, when should they be considered, and general mistakes that can be easily avoided during an SRM implementation. These implementation tips and tricks will help you get your SRM implementation off the ground with some key lessons learned.

In general, I think it is very instructive to look at business cases and “lessons learned” before embarking on a new implementation. These lessons learned act as analogies from which decisions are made. All of the mistakes in this paper are based on projects that I have been directly involved with, or projects that I have talked with SAP or other consulting organizations about. I know these lessons learned form a large part of my decision making process, and I think they can serve to solidify an understanding of the underlying business processes associated with making and saving our businesses money. And to a certain extent, these lessons are the value that a top-notch consultant can bring to your organization. Lessons like “use external catalogs if you can,” or “try to go with the Classic mode,” or “be careful with your org plan” often get ignored. It's also possible that a consultant will be the one who makes the case that “this is not SOX compliant.”

To me, the key to effective consulting is simple: it's all about seeing something that has worked well before and making a recommendation that will help a client deploy a stable piece of software that will work for them for years to come. As we begin, it's good to keep in mind that anything that’s bad for a business is bad for an SRM implementation project. Don’t forget that SAP is not only a tool, but it also implements best business practices.

Over the years, I have had experience working on several high-profile SRM projects. All of these clients went live with the solution and consider the product a success; but for each client, the “mileage” each gained from their implementation varied widely. Hopefully, in reading about the most common mistakes, this paper will be a “lesson learned” for many SAPtips readers.

I. Strategic Mistakes

Mistake #1 – Using SRM to Implement SOS ("Same Old Sourcing")

The number one problem, and hence the #1 position in my list of seven mistakes, is called "SOS" or "same old sourcing." Using your company's same old sourcing strategy will ruin your SRM implementation and possibly your business. One of the great things about SRM is that SAP has delivered a set of best business practices and the software to enforce their use within your organization. This being said, I realize that it's very difficult to change large organizations, but there are many rewards for people who break the mold and rethink their business processes. .

Here are some common quotes from "SOS" projects –

> "We can't do that, it's just not how we do business!"
>
> "We'll consider this for phase two…"
>
> "You can't do this; it's not SOX compliant."
>
> "Can we build a custom front end for the shopping cart that looks like our current purchasing system?"

How to avoid using the same old sourcing strategy:

First, your SRM project should have been initiated by your corporate procurement department, or have them involved as a key stakeholder – they're the folks who should be responsible for formulating supply strategy. If you're working on an IT initiative, try to form relationships with a procurement manager who wants a promotion and a big raise. Have your SRM team leader set up a prototype system ASAP, and show your procurement friends all about the benefits of deploying SRM, namely: self service procurement (requisitioners complete their own reqs online and go for subsequent electronic approval), vendor enablement (vendors enter and maintain their own data), live auction and bidding (automate your RFP process), and even demo a vendor catalog.

The key to saving big by implementing SAP SRM is using the software to change your organization and to enforce best business practices. In other words, implementing SRM can help your organization make the shift from tactical purchasing to strategic purchasing.

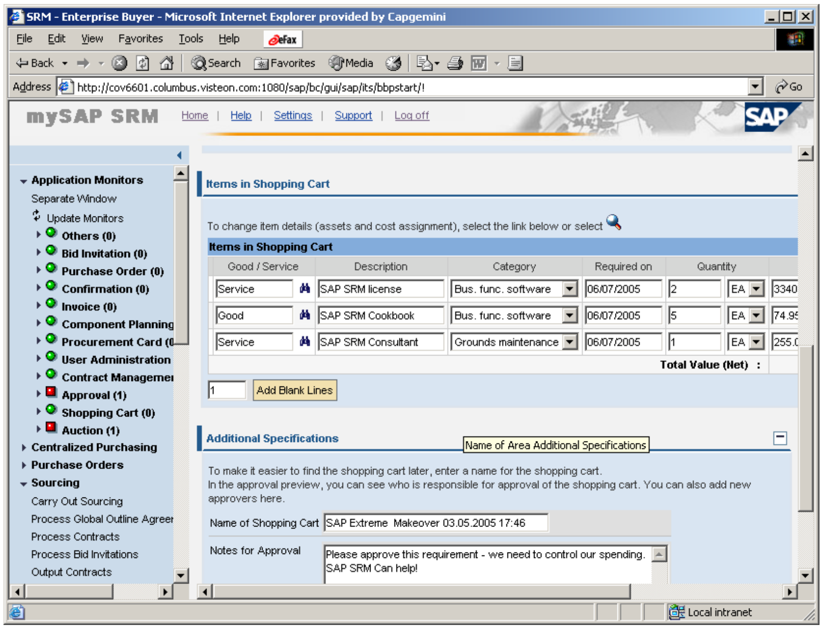
Benefits of shifting to a strategic approach to purchasing include:

- reducing your overhead associated with purchasing approvals,
- using a competitive or blind bidding strategy,
- negotiating contracts
- developing relationships with your vendors that allow you to improve quality and pricing and allow them to maintain their own catalog data

These are all activities that will help you move from tactical, "one off" savings to strategic, enforceable, and repeatable efficiencies that will pay back your investment and perhaps even save your business.

Mistake #2 – Thinking that SRM Is Just Another SAP Module

I've seen companies run into trouble because they didn't realize early enough in their implementation planning that "SRM is different." Most implementation scenarios require heavy integration into your back-end R/3 system and will require an easy familiarity with several new technologies. This mistake is ignoring the differences and assuming that the project will get along fine without help from SAP or another experienced organization. Many important SRM transactions rely on ITS services/business HTML, which are new for most customers. See Figure 1 for an example of an SRM application screen and how it differs from R/3. You can see in Figure 1 that in SRM, free-form entry for shopping cart items is allowed, along with drop-down lists and Web search helps for standard units and currency. And the entire transaction is Web enabled.



http://cov6601.columbus.visteon.com:1080/sap/bc/gui/sap/its/bbpstart/!

Items in Shopping Cart

To change item details (assets and cost assignment), select the link below or select

Items in Shopping Cart

Good / Service	Description	Category	Required on	Quantity		
Service	SAP SRM license	Bus. func. software	06/07/2005	2	EA	3340
Good	SAP SRM Cookbook	Bus. func. software	06/07/2005	5	EA	74.95
Service	SAP SRM Consultant	Grounds maintenance	06/07/2005	1	EA	255.0
				Total Value (Net) :		

1

Additional Specifications

To make it easier to find the shopping cart later, enter a name for the shopping cart.
In the approval preview, you can see who is responsible for approval of the shopping cart. You can also add new approvers here.

Name of Shopping Cart: SAP Extreme Makeover 03.05.2005 17:46

Notes for Approval: Please approve this requirement - we need to control our spending. SAP SRM Can help!

Figure 1: Create Shopping Cart – Not the R3 You're Used To

How to avoid this mistake:

If you need any exposure to the underlying SRM objects, make sure you have someone on hand who has a sound understanding of the Web and surrounding technologies like the operations of a Web server. I've been on an SRM project that brought 3000 users live globally in five months for direct and indirect spending BW, approvals, and portals. This included blueprinting, construction, testing, and training. It was possible because SRM is easier to use and is getting easier to implement quickly. This is especially true for customers with an existing installed base of MM or FI/CO.

Mistake #3 – Not Considering Live Auction Cockpit and Bidding Engine as a Part of the Implementation Plan

The fact of the matter is that most of my SRM clients have not deployed, and are currently not considering deploying, one of the most valuable aspects of the SRM system. The mistake probably stems from using the "SOS" plan.

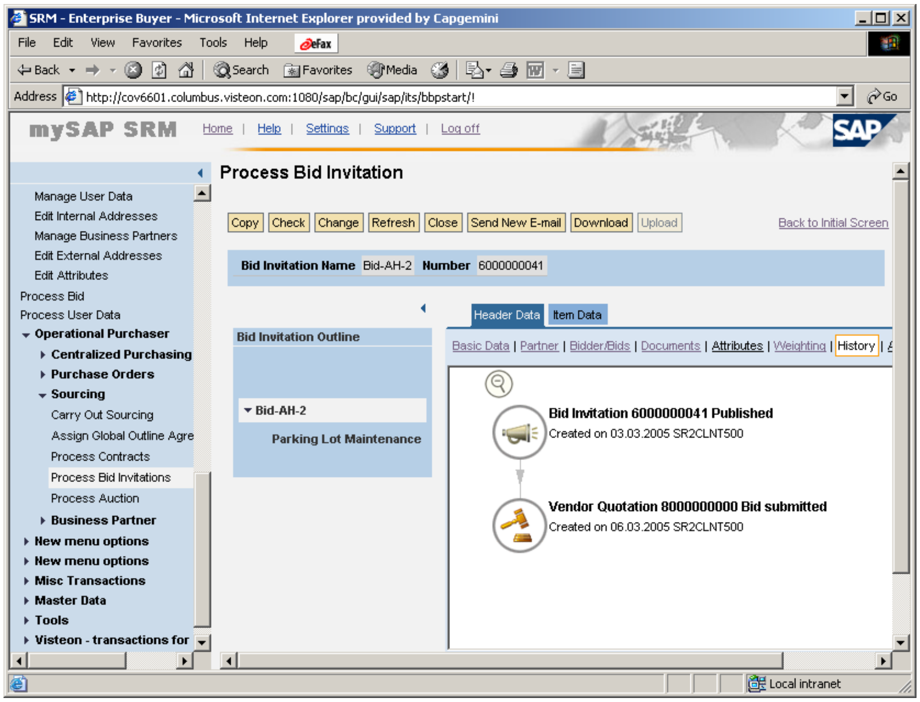
How to avoid this mistake:

Customers are always surprised at how easy Live Auction and Bidding is to implement as a pilot project, and they are also equally surprised when they understand the vast savings that can be earned by putting requirements out for competitive bid.

Here's all you need to know: The delta between the SRM and basic auction and bidding configuration is quite small, especially if you want to set up a small pilot to demo to procurement. Review the following checklist – if you have the following implementation parameters, rolling out bidding will be a snap:

- Classic or Standalone mode
- Limited number of items put out for bid (1 per month)
- Limited number of bidders/vendors for each auction (12 per auction)

See Figure 2 for a sample screen from the Live Auction bidding process:



Process Bid Invitation

Bid Invitation Name Bid-AH-2 Number 6000000041

Bid Invitation 6000000041 Published
Created on 03.03.2005 SR2CLNT500

Vendor Quotation 8000000000 Bid submitted
Created on 06.03.2005 SR2CLNT500

Figure 2: Bidding Engine – Graphical Document History of a Vendor Bid

Even if the above don't apply to your implementation, piloting Live Auction and Bidding shouldn't be too much work. That is to say, if you can get around your Basis team's SOX requirements review – which I'm sure is much more comprehensive than the configuration required to enable the SRM Live Auction and Bidding Engine.

II. HR and Staffing Mistakes

Mistake #4 -- Underestimating Security and Basis Resource Requirements

Underestimating security and Basis resource requirements is a crippling mistake that can be made on even the most well-planned SRM implementation project. A common mistake is for an R/3 security team to be assigned to an SRM project in some remote location. The security team takes a look at the standard SRM profiles and roles and immediately throws them out, assigning transactions that they think look like what an administrator, manager, and employee might need. What happens is that nothing at all works how it should, and the configuration team is stuck because they don't have access to create a shopping cart or replicate material groups from the backend.

How to avoid this mistake:

SRM is very much a "role-driven" module; that is to say, that what a user sees, and many important behaviors of the SRM system, are driven by the user roles, org plan attributes, and security profiles. It is important to understand that SAP delivers standard roles for SRM that should be adhered to as closely as possible. If your security team absolutely has to stir the pot and insists upon custom "Z roles", then you need to insist that they are present in the war room, or that they have a full-time resource dedicated to the project, and that they are there to fix everything they break. This is because you can count on problems with any deviation from the standard delivered SAP SRM roles.

The lack of proper security support comes in many forms. First, if you can find security people who have experience using the SRM product, they will understand the importance of sticking to the standard roles, and your entire implementation will run more smoothly. The second form in which a lack of security pops up is that your implementation partner may decide that it's enough to have a security team offsite. Offsite support can work for security support of SRM in a stable productive system, but if you are working under deadlines in the construction phase of a project, you will have a terrible time. You will need security and Basis support onsite if you decide to go with custom roles.

In short, SAP SRM requires cooperation between your security/Basis team and implementation consultants. They should all be in the same physical location or have very open lines of communication and cooperation. With the owner of the org structure and organizational management tools on the main implementation team, and a security or Basis administrator halfway across the country, you could be headed for disaster. Using the standard delivered SAP roles could help you avoid this situation, as well as collocating your team and encouraging them to work together. In summary, SRM is very much a role-based product. This makes developing new roles and integrating SRM roles with the existing security plan a challenge for remote security or understaffed basis teams. SRM will require your Basis and security team to learn some new transactions, and this will take some time; and if they insist upon developing custom roles, they should be in the same location as your main implementation team. Don't leave them out of the plan, or you won't find out until it's too late.

Mistake #5 – Not Getting Change Management Under Control

I've said this before, but SRM is a different sort of beast than R/3. Here's one concrete reason why. SRM is different because there are dozens of critical configuration and development items that are client and system specific, and as such are *non-transportable*. I've seen large enterprises struggle to keep any sort of change management systems in place with the SRM product, and they have suffered in maintenance. The mistake here is not to validate that all configuration and development artifacts are kept consistent throughout your system landscape for each and every change. It's a mistake to keep different org plans in DEV, QA, and PROD, and it's a mistake to have to test your code in production after you've gone live.

How to avoid this mistake:

The change management strategy should account for one very important difference in all SRM implementations – the client specific configuration that needs to be completed for each system. It is a mistake to set your consultants off configuring, and not keep tabs on DEV, QA, and PRODUCTION. SRM can require a fair amount of client-specific configuration and master data maintenance. You should devise a plan and make an employee responsible for keeping each system and client in synch. Pay careful attention to your org plan. It makes sense to put one person in charge of the org plan, and also make that person responsible for keeping and changing each applicable system.

Making something a priority in your implementation means that your resources have enough time to do things the right way. Make sure there is enough time built into your plan to come up with and enforce a proper change management process.

III. Implementation Mistakes

Mistake #6 -- Not Leveraging External Catalogs.

One of the most forward-thinking capabilities of the SAP SRM system is also one of the most overlooked. Allowing your vendors to maintain a special view of their catalog for use by you and only you can save your company mountains of work in keeping vendor catalogs coherent with current pricing. Not letting your vendors maintain their own data is a mistake.

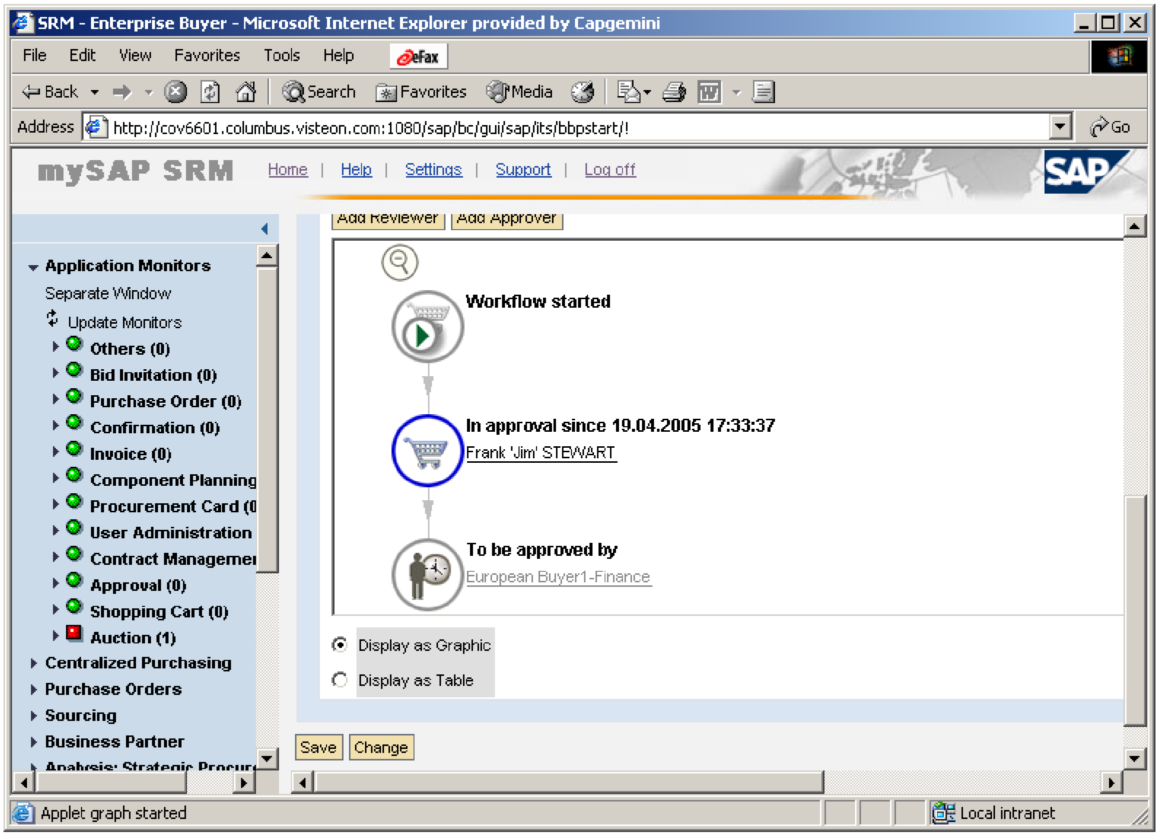
How to avoid this mistake:

Start early and decide on several key vendors to form preferred relationships with. You will negotiate a contract with each vendor. One particularly successful SRM implementation offloaded the quite considerable master data maintenance task associated with maintaining an internal catalog to about a dozen key external vendors who agreed to maintain external catalogs.

External, or so-called "punch out" catalogs, are catalogs maintained outside of your control. "How do we maintain control?" you might ask? . You can keep control by implementing a business process change – you will now negotiate a contract with each vendor, something to the tune of, "You will offer us 20% off of your best published Web rate," or some similar contract.

Mistake #7 – Not Using the Standard SAP Approval Processes for Purchasing Document Approvals.

As a management consultant, I see these procurement document approval processes as an opportunity to change your corporate finance policy. Too often, companies get tied up in the “this is what we do today” game, and think that they need to replicate their manual or outdated approval processes in the SRM system. This is a bad idea – the idea behind implementing an integrated electronic procurement system is to accelerate your business and make your administrative tasks easier and less time consuming, so that your time is freed up to focus on strategic purchasing initiatives, rather than chasing down a level-three asset manager for a rubber stamp. See Figure 3 for an example of how SRM automates the approval process.


SRM - Enterprise Buyer - Microsoft Internet Explorer provided by Capgemini

Address http://cov6601.columbus.visteon.com:1080/sap/bc/gui/sap/its/bbpstart/!

Workflow started

In approval since 19.04.2005 17:33:37
Frank 'Jim' STEWART

To be approved by
European Buyer1-Finance

● Display as Graphic
○ Display as Table

Applet graph started

Figure 3: Graphical Approval Preview, a Popular Standard Feature of SRM.

I’ve had the opportunity to lead several SRM projects, and usually end up focusing a very large amount of my attention in the area of helping accelerate the purchasing document approval processes. SAP SRM implements standard approval workflows that are very easy to implement, use, and maintain. SAP has really outdone itself with their standard approval workflow in SRM 5.0, and my hat is off to their entire development and consulting team that supports these approval processes. You can capture an extremely rich set of approvals for a very matrix-like organization without writing a single line of code.

How to avoid this mistake:

SAP has provided standard approval workflows for shopping carts, purchase orders, invoices, and bid approval (to name a few). Consider using these before launching into expensive custom development of custom approval workflows.

Conclusion

In summary, SRM extends a set of tools that, if used properly, can create a platform for building information-rich decision making tools. If you consider how most companies make decisions – trial and error and intuition – I think you'll agree that it is worthwhile to invest time in formulating an SRM strategy for your business. If you can avoid repeating these "seven mistakes," you will be on the road to successfully deploying one of the richest and easiest to use tools that SAP has delivered to date.

Jim Stewart, Equity Technology Group (Los Angeles, CA) *has over 10 years of experience as an SAP systems professional, during which time he has served as a functional team lead, technical team lead, and ABAP programmer. He is the author of the popular SAP SRM Advanced EBP Cookbook available at www.srmcookbook.com Mr. Stewart has implemented SAP SRM for several major public and private organizations and is an expert in EBP and the associated approval and workflow processes. Mr. Stewart is the founder of Equity Technology Group, an SAP consulting partner, and continues to practice as a consultant in the area of SAP SRM/EBP, Workflow, and Web programming. He can be reached at fjstewart@equitytechgroup.com*

NO WARRANTY: This documentation is delivered as is, and Klee Associates, Inc. makes no warranty as to its accuracy or use. Any use of this documentation is at the risk of the user. Although we make every good faith effort to ensure accuracy, this document may include technical or other inaccuracies or typographical errors. Klee Associates, Inc. reserves the right to make changes without prior notice.

NO AFFILIATION: Klee Associates, Inc. and this publication are not affiliated with or endorsed by SAP AG. SAP AG software referenced on this site is furnished under license agreements between SAP AG and its customers and can be used only within the terms of such agreements. SAP AG and mySAP are registered trademarks of SAP AG.