CIO

Print Article Close Window

From: www.cio.com

How To Support Cloud Applications

– David Taber, CIO

September 30, 2011

Cloud computing gives user organizations and IT alike the freedom of maneuvering and long-run flexibility that was never possible with traditional enterprise applications. However, that freedom makes for new levels of unpredictability when it comes to big deployments, and some new requirements for support.

As I've written endlessly, CRM applications are different from other enterprise applications, largely due to the needs and habits of its users. Generally speaking, sales and marketing folks put a much higher priority on success (both the appearance and the reality) than they do on process discipline or little details like data quality. Couple that with the shifting sands of marketplace competition and evolving sales models, and the result is a system that must frequently accommodate significant degrees of change.

So the combination of cloud and CRM can make for some stressful nights in the internal support organization. When there are sales territory splits or redefined partner overlays, there will urgent pushes to change lead routing, account ownership, and opportunity validation rules. These changes may seem to slide right in, but can sow the seeds of bugs in commissions or referral fees. Almost inevitably, these problems won't show up until you've incorrectly processed hundreds of orders.

The traditional help desk infrastructure (and, I have to say, attitude) really won't cut it for these high-profile problems. All too often, there are political overtones that lead to blame-games that you'll want to avoid. So let's look at some best practices that are the cornerstones of cloud system support:

• **It all starts with architecture and configuration control.** The truly scary problems in cloud CRM don't occur within the narrow purview of the CRM system. The issues surface at the points of integration with accounting, order processing, e-commerce, portal, or other customer-facing applications. Even though it's easy to make changes — literally in hours — in your CRM system, that doesn't mean that you should do so...or that your CRM changes will correctly propagate to those other systems. There needs to be a configuration control process that looks at the consequences of any change made to the code, the data model, or even the semantics of picklist values. Really, this one can kill you, because there's no administrative "center" of a large cloud deployment. Each application has its own control panel and dashboard, but the consequences of changes across the interfaces and mashups can only be assessed by a panel of SMEs.

• **Your team will need instrumentation.** Under the best of conditions, troubleshooting across applications is tricky precisely because of the loose coupling that provides the advantages of the cloud. Early in your development cycle, make sure to write error handling and logging classes that write diagnostic records during production use. As discussed here, the error log should itself be stored in a "centralized" cloud service, to supplement the logging available in each application.

• **Get really good at deployment.** As the world of cloud computing is relatively new, industrial-strength infrastructure is still on the way. Deployment is a combination of hand-crafted scripts and checklists, with details that seem to be always be evolving. Whether it's a bug patch or an Agile release cycle, the process needs to be repeatable, reliable, and as streamlined as possible.

Slideshow: What is Cloud Computing?
Slideshow: 10 Cloud Management Companies to Watch

• **There must be a publicly-available information repository about cloud design, implementation, and deployment issues.** No single cCloud vendor will provide this, so it's a DIY item. As I mentioned a few weeks ago, the "what's what" must be documented both in the individual cloud applications and in a communal area. In addition to descriptive materials, the support team will need standard test cases, expected results, and work-arounds for known problem areas. Whether you use a Wiki, a set of Google docs, or shared folder — *everyone* should be able to read these docs, and everybody on the IT team should be able to update them in real time. Transparency matters — it's a cornerstone of collaborative problem solving. In case it isn't obvious, forget paper for these docs.

• **Support needs to be the consummate Agile function.** Since things can go awry so quickly when there's a bug in your CRM configuration or data, the support team needs to work in fast scrums, with daily triage and prioritization meetings. It's a smart idea to have your sharpest QA person in on these meetings, as they can help troubleshoot and they'll have to plan for deploying the fix. A well-run 10-minute daily "stand up" meeting is a hard requirement, particularly if your internal support team is operating in more than one country.

• **Prioritization is everything.** Since the worst support problems always seem to occur in clusters near business deadlines, the first order of business is to work on the few problems that really matter — and avoid the noise. In addition to the traditional priority and severity categories for bugs, use a pick-list for organizations impacted, a picklist for cloud systems affected, and a metric for number (or dollar volume) of orders that would require rework.

While some of these may represent new practices, they can typically be overlaid on an existing support organization. The trick is to become more responsive on the things that really matter — even when they're spread across several systems — without driving up costs.

David Taber is the author of the new Prentice Hall book, "Salesforce.com Secrets of Success" and is the CEO of SalesLogistix, a certified Salesforce.com consultancy focused on business process improvement through use of CRM systems. SalesLogistix clients are in North America, Europe, Israel, and India, and David has over 25 years experience in high tech, including 10 years at the VP level or above.

Follow everything from CIO.com on Twitter @CIOonline.