



From: www.cio.com

Cloud-based CRM: Beware Bad Data and Silly Object Models

– David Taber, CIO

February 08, 2011

Over the course of too many years, I've said that you can't make a user interface that's too easy for users. But cloud-based software vendors have essentially done just that. System setup is now deceptively simple, and too many sales pitches tell stories of how users can set up a cloud-based system themselves.

The problem is, users (and junior consultants) will feel empowered to set up cloud-based systems up without a plan or even a model for proper system execution. The cloud system's GUI will let them do so. A few months later, the system will be mired in bad data (that was too sloppily imported), a silly object model (that was too easily created), and a cacophony of business rules (that were never vetted).

Here are some important steps you can take in order to avoid pain later on.

Agile != Ready, Fire, Aim

Too many sins of omission have been committed in the name of Agile. Agile projects may be light on documentation, but they're supposed to be heavy on domain expertise, user feedback, and — most critically — technical excellence. So don't let anyone make the excuse that because the project is Agile there isn't time to think through classic IT issues.

In cloud CRM systems, some of the messiest novice errors come from a misunderstanding of the object model. Leads are confused with opportunities. Assets are confused with products. Entitlements are confused with contracts. By using these terms interchangeably, undisciplined users will end up asking for extra fields to be added in the wrong places.

Consequently, imports and ongoing data feeds will be scrambled. Because the data is in the wrong place, the users will soon ask for triggers to denormalize fields or replicate entire objects. Later, they will ask for functions and triggers that are essentially reinventing the wheel of standard CRM functionality.

This isn't just cloud CRM systems: nearly any cloud DBMS can be easily misconfigured with data models and entity relationships that won't serve the business need.

Not Just Semantics

The next trap for novice users of cloud CRM systems stems from imprecise definitions of the state and meaning of data. The UI will not challenge blurred or even contradictory definitions. While this issue is starkly obvious with pick lists (in the choice and number of values), it also comes up in the definition of record types and the setup of account hierarchies. Classic examples of blurred meanings include:

- What does it mean to be a lead? How many lead statuses are there (e.g., open, contact attempted, contacted, qualified, unqualified, dead)?
- What does it mean to convert a lead? What are the qualification criteria required before conversion?
- When is an opportunity created? What does it mean to have an opportunity of \$0 value?
- What are the sales stages for an opportunity? What are the entry and exit criteria for each of the stages, and how long is each stage expected to last?
- What's the difference between a no-action opportunity (there isn't a deal here for any competitor), an unqualified one (there's a deal, but not one that matches us), and a lost opportunity (where we were in full competitive status, but were not selected)?
- What are the criteria for a Severity-1 vs a Priority-1 case? What are the SLA thresholds and

ramifications for each of the severity and priority levels?

- What's the difference between a "strategic customer" and a "mid-market customer"? If a strategic account has only a small departmental deal, does that make it a mid-market opportunity?

Some of the ambiguity and imprecision surrounding these status and criteria come from the political consequences of their definitions. If it's too clear what's in and what's out of the sales pipeline, this can make sales or marketing look bad. The problem, of course, is that a CRM system is going to enforce logic and consistency — and the ambiguity around meaning can cause ridiculous misrepresentations in reports, alerts, and dashboards. The temptation among novices is to try to blur the clear logic of the CRM system, rather than tighten up the definitions of their own terms.

Shoot the Messenger

In both marketing and sales, there's a real aversion to lost deals and weak leads. The temptation is to simply delete all the losers from the system, so that all that's left are winners and hopefuls.

Unfortunately, as Bill Gates wrote, "success is a lousy teacher." By erasing data — even clearly bogus or corrupted records — we're erasing the possibility of learning anything. Instead, leaving the "bad" and "loser" data in the system leaves open the possibility of troubleshooting and rectifying the business process that created them. Too many bad leads? Maybe that marketing vendor we use isn't careful enough. Conversion rates too low? Maybe we should stop buying leads from a list broker.

This last issue can be often solved by simply removing the Delete privileges from most users. But the general problem remains: just because clouds make it look easy doesn't mean that users can forget about IT and system design issues.

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](#).

© 2010 CXO Media Inc.