CIO

Print Article Close Window

From: www.cio.com

10 Cloud Administration Must-Haves You Can't Have Yet

– David Taber, CIO

December 06, 2012

As I wrote in Cloud UI Design Mistakes to Avoid, it *is* possible to make a user interface too easy. Because by making the UI a snap to navigate, using cutesy names in place of standard terms and filling the control panel with bright, shiny objects, you make users *think* they know what they are doing, even when they have no clue about the consequences of their actions. This goes double for system administrators who may be setting up disaster scenarios at the touch of a button.

Don't set off the curmudgeon alarms yet. I'm not going to rant about the need for a command line for cloud applications. However, I am going to make the case for a disciplined user interface, particularly for the system administrator or super user whose actions often have serious consequences.

Cloud Administration UI Can't Follow the Tracks of Your Tears

Cloud applications and platforms have made tremendous strides in interactivity and usability of their administrative control systems. Considering that these administrator tools have to run inside a browser, some of the drag-and-drop capabilities are nothing short of amazing. No complaint from anyone there.

Commentary: Cloud and UI: CIOs Have a Big Opportunity on Their Hands

Given the amount of work involved in getting those control panels to work, it shouldn't be a surprise that there isn't much in the way of administrative cues or context-sensitive help. Most of the time, "Are you sure?" is all you can expect. Sometimes, though, "Do you understand that 108,347 leads will be affected by this?" would be a better idea. Of course, admins may blithely click Continue anyway, but at least they know they have themselves to blame.

The real problem starts when an administrator UI doesn't have a whiff of an undo. Inevitably, some actions will cause data changes that can't be undone. At the very least, you could offer to revert the administrative setting to whatever it was before the "oops."

When coding, word processing or using spreadsheets, we all expect to have several levels of undo. In the admin's browser UI, though, I can't think of an example of even a two-level administrative undo. In many cloud apps, the only way to back-track is to take a screenshot before every administrative change. What a pain.

If you're lucky enough to use an IDE to modify XML control files, you've got a fighting chance of restoring things to before-the-oops status. The Salesforce.com Eclipse-based IDE is a great example of this recovery strategy, but many admins don't know the dozen less-than-obvious best practices required to make it work in the real world.

This could be OK if the system has an administrative change log. This isn't an audit trail on the data, though: It's the change log on the system metadata. This feature is easily presented in the browser, and it's fairly easy for administrators to use to reconstruct the crime (particularly if the log can be downloaded as a CSV file).

One non-obvious best practice: Back up this change log and the user login history on at least a quarterly basis, as these metadata logs will not get backed up as part of the main database. These log backups can save a lot of pain and literally tens of thousands of dollars, so don't neglect them.

The Same Old Song, But With a Different Meaning

In the client/server Dark Ages, we all went through a lot of these administrative and control issues. The outcome then was *not* a replay of either PC or mainframe administrative systems. This time, we won't just get a replay of the management and control systems of the 1990s, either.

There are three reasons why. First, the administration problem is a bit harder when systems are (and must stay) loosely coupled. Further, the cloud is opaque: It's not like you can go look at the underlying database transaction logs. Finally, there's the IT organizational reality. It's much less likely in today's world that administrators will be centralized, so cross-department coordination of systems administrator becomes a deeper challenge.

Tips: How to Document Cloud Design Decisions

There won't ever be a single, unified platform and master control panel for cloud apps—that is the stuff of absurd dreams and sales presentations. What we will see—and, as customers, need to support when we see vendors make serious progress—is the development of administrative tools that can work in a multi-cloud environment.

Don't Let the Cloud Catch You Crying

Without further ado, here are the 10 core administration elements we need in cloud applications:

- Cloud administration UIs designed around a more highly trained target user. Lighten up on the cutesy stuff and provide clear, unambiguous indications of what is about to occur when buttons are clicked.
- Long-text "comment" fields on *every page an administrator uses*, to allow for annotations of the before-state, after-state and rationales for actions.
- Mechanisms to do global searches within metadata, for where-used analysis and troubleshooting.
- System audit trails for data changes, including the running user and changing mechanism for each change.
- Logs of every administrative, metadata and security change, even when applied via API.
- Logs of every time a report is changed and run, including the running user's ID.
- Records of at least the last time any workflow or trigger is run, including the running user and the error message, if any.
- An API that provides at least read access to administrative state, metadata (including all related comments and annotations), security settings, and metadata logs so external tools can be used to identify and troubleshoot changes. (And you thought Regex was dead!)
- A way to export all metadata logs as either CSV or XML files.
- A standard service for centrally logging all error messages, as I discussed in a previous tip.

I know of no cloud application or platform that supplies all of the above today, though Salesforce.com is admittedly close. Several other popular cloud apps out there that have almost none of the bullets covered.

That said, be patient. This is hard work, and vendors can't make money at it. Even though this won't be solved soon, you need to be firm with your cloud vendors every time they pitch you some bright, shiny object with only sketchy manageability.

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. Taber has more than 25 years of experience in high tech, including 10 years at the VP level or above.

Follow everything from CIO.com on Twitter @CIOonline, on Facebook, and on Google +.