TUTORIAL

Deduping in Salesforce: The complete how-to

Don't be duped by Salesforce.com duplicate records. Deduping seems like one of those routine system maintenance issues, but it's actually very tricky and there's no undo.

By David Taber

CIO |
AUG 31, 2017 3:00 AM PT

Table of Contents   ⌄

Page 3 of 7

Nipping dupes in the bud

We all make mistakes, and with certain extract, transform, load (ETL) and import cycles, big fat errors are just a mouse-click away. The problem is, with triggers and workflows, any SFDC insert or update may have tons of repercussions on other records. You can't just delete the mistake, you have to look for all the places where it propagated and reverse that, too. Obviously, every situation is different, but here's the basic strategy to undo the thing you shouldn't have done.

- Send an email out to users telling them to ignore records created today not do updates to the system's records until further notice. Yes, it's very painful, but if you let them continue to update things, you may never unscramble the data.

- Figure out every table that was directly affected and do a backup of every field and every record in those tables.

- Go to the backup tapes – you DID do a backup before you started, right? And you DO have backup files, at least on a weekly basis, right? Get the most recent files for comparison/analysis.

- Don't start modifying anything until you understand the failure mode of what caused it. If you don't appreciate this properly, you will make erroneous deletes and be "unwinding" the wrong things, which is even more irritating and visible for the users.

- Do a bunch of tests to find out what your main dupe pattern is. Typically, they will all be created by the same user (you) at (almost) the same time. Look to see if there are any updates to any of those records since you created them — if you're lucky, there aren't any (so there's no information value there, and you don't have to look for relevant updates). Look to see if there is a "last activity date" of today — if there isn't, that's a good thing.

- Do an extract of the IDs of every record that could be related (as a parent or child) to the object where there are dupes. For Accounts, this is, at minimum, contacts, tasks, opportunities/deals and notes. All you need of those tables is the record ID and the pointer back to the account. What you're looking for is records that may have been mistakenly attached to the dupes because if your newly-created dupe is deleted, the cascading deletes could wipe out valuable information.

- Do a vlookup to find out if any of the related tables currently point to the dupe records. Hopefully not. If they do, you need to move the related records' pointer to the original (non-cloned) record. This is fun in and of itself.

- If there is any information of value to be fished out of the dupes, do that now. This is almost entirely a manual process.

- Once the original (non-cloned) records are the "high water mark" for information value, you're ready to remove the dupes.

- During this dupe-deletion cycle, turn off all related triggers and workflows. Don't forget to turn them back on when you are done.

- Create a tally report (using a DIFFERENT TOOL than what you use for data manipulation, as a cross-check) of the record IDs to be deleted. Double-check that you've got the right list, and that the theory you developed in the first bullet proves out no matter how you look at the data.

- Remove a few of the dupes and examine their related records. You are looking for any error messages or weird updates that might occur. If nothing surprising happens, remove a portion of your dupes (say, 25 percent). Do it in phases – never remove all the dupes at once.

- Delete those records, but make sure you keep a copy of the original UOID (so that if you have to, you can go find that data in your dupe backup you did at the beginning).

- Keep all the files (including success/failure output files) and notes for this process in a single ZIP file, and put it in your data archive. Permanently.

💡 **Benchmark your IT organization in our 2018 State of the CIO Survey**

YOU MIGHT LIKE ⠿