

The importance of understanding Salesforce data models

Published March 2016

Salesforce.com has extended beyond its origins as a CRM (customer relationship management) vendor and has added complementary applications in both the services (for example, call centres) and marketing arenas. In addition, it offers App Cloud, where the company provides tools for either extending existing Salesforce applications or for building entirely new ones. This has meant that Salesforce is increasingly viewed, and used, as a strategic platform by large enterprises. For example, Barclays Bank has implemented a community application for external mortgage brokers, based on the Salesforce platform (see <https://www.salesforce.com/uk/customers/stories/barclays.jsp>).

As an environment for supporting enterprise-class applications there are considerations that did not apply when Salesforce was used only for tactical purposes. In particular, there is a greater need for data and information management rigour to be applied to these systems. Information and data governance policies and procedures need to apply to Salesforce just as much as they do to on-premises systems. Unfortunately, recognising that this is a requirement and doing something about it, are two different things. As a tactical application CRM can happily live within the confines of the relevant business department. However, once a platform becomes strategic then its governance falls under the aegis of the IT department. This can cause problems when Salesforce applications become very large and complex or because there are multiple Salesforce applications in use across the organisation. In particular, Salesforce provides very little in the way of tools for data professionals to understand the data models being used, whether from the point of view of navigation and analysis or where you want to create, save or share sub-models.

There are a number of particular issues that are worth discussing:

- Many companies are either heavily customising their applications or building entirely new applications using App Cloud (and especially Force.com). Many of the latter (for example, Financial Force, Veeva and RootStock) are effectively ERP (enterprise resource planning) applications. In either case, this can mean much larger data footprints than has traditionally been the case: the standard for Salesforce is about 200 tables but we know of some companies with more than 3,000.
- App Cloud supports agile development methodologies. In agile environments this tends to mean that data models change very frequently – often, with every sprint stage – and it is important to be able to understand those changes and to be able to compare the latest build with previous builds.
- Migrating to a Salesforce environment means that you need to know where your current legacy data should go. Clearly, the larger and more complex the Salesforce environment you are migrating to is, the more you would like to have tools to help you understand the data model you are migrating to. We should add that the same applies in the event that you want to migrate away from Salesforce.
- Large organisations will often have multiple Salesforce instances for different geographies or business units, or as a result of mergers and acquisitions. Typically, these will have been developed and customised independently. However, once you start to tackle these silos from a strategic governance perspective you will likely want to either converge on a corporate standard or, at least, you will want to be able to report across these instances. Achieving this successfully, will be a major issue without an understanding of the respective data models. For example, we know of one company with ten different Salesforce implementations across Europe that had to employ full-time staff for reporting purposes because there was no easy way to understand the different customisations that had been implemented across the various Salesforce instances.

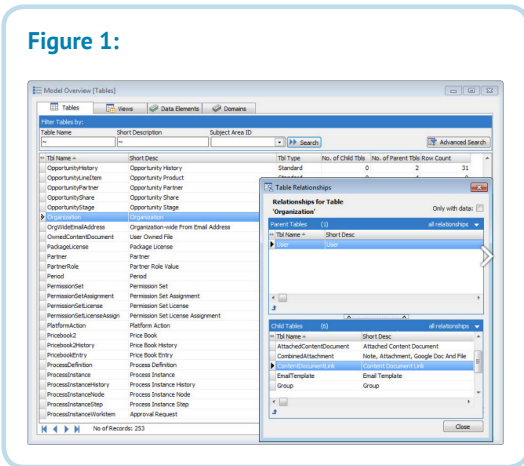
It should be clear from the foregoing that understanding Salesforce data models – whether for the company's own applications or for those built on the App Cloud – is an increasingly important issue. But, as we have mentioned, there are no tools available from Salesforce to enable an understanding of these increasingly complex data models. Fortunately, exactly that capability is provided by Silwood Technology's Safyr® for Salesforce.

Silwood has historically provided the ability to understand and analyse data models for large ERP environments from SAP and Oracle, including all the ►

customisations that users might have incorporated into their installations. Now that capability is also provided for Salesforce: both the applications provided directly by Salesforce and applications built using the App Cloud and Force.com.

Specifically, Safyr is a metadata discovery tool for applications. In the case of Salesforce, it can list all tables used within the application, regardless of whether these are tables shipped by Salesforce (or by some other vendor using the App Cloud platform) or whether these are tables that you have personally created or customised. You can then drill into any table by right clicking, to see not just the details about that table – such as foreign keys – but also that table's relationship to other tables and, in particular, parent-child relationships (see **Figure 1**).

Figure 1:

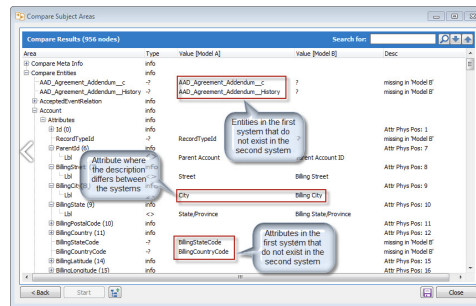


There is also a comparison facility that allows you to compare instances of Salesforce (see **Figure 2**), which will be particularly useful both in agile development environments, when you want to compare data models across sprints, and when you are consolidating multiple instances of Salesforce. The results of such comparisons can be exported in CSV format so that you can do further analysis of the comparison in, for example, Microsoft Excel.

The data model that you have discovered using Safyr can be visualised as an entity-relationship (ER) diagram within the tool itself. You can also export these details to popular data modelling tools such as ERwin, IDERA's ER/Studio or SAP's PowerDesigner or for use with other Information Management solutions. In addition, you can also create and visualise sub-models – for example, by subject area – that can be treated in the same way. This is most likely to be helpful where ERP style applications have been deployed based on App Cloud developments and where, for example, you want to create a sub-model just for sales details or HR or some other sub-category of the environment.

The bottom line is that data modelling is a growing issue within Salesforce environments as that company expands its platform and ecosystem. Increasing numbers of larger and more complex applications are being deployed, and the need to explore and understand the underlying model that underpins these applications is becoming more

Figure 2:



urgent. Indeed, the fact that Silwood has extended its capabilities from traditional ERP environments into the Salesforce realm is testament to a growing demand for its expertise. As far as we know, Safyr is the only product on the market that has the sorts of capabilities we have described, so if you have an issue with Salesforce it would be worth your while to investigate what Silwood can offer.

Philip Howard
Research Director