silwood
DISCOVER, SCOPE, DELIVER

Trying to answer the question “Where’s the data?” in the context of Information Management projects which involve packaged applications can be frustrating and time-consuming. This white paper offers insight into why the traditional methods are not effective and an alternative software based approach to solving the problem.

“Where’s the Data?”

Finding, understanding and using packaged application metadata.

Roland Bullivant,
Silwood Technology Limited

Introduction

Ever since applications have been used as a source of data for Information Management projects, one question has been a constant challenge and source of frustration to data professionals seeking to use their data for their projects.

That question is: "Where's the data?"

By that I do not mean "how much have we sold, of a specific product, in a specific region, over a specific period and is that trend going up or down?" What I mean is how to locate where that data resides in order that the data professional can answer that question.

This simple question is at the heart of many data critical projects including

- Data Warehouse and Analytics
- Extract, Transform and Load (ETL) and Extract, Load and Transform (ELT)
- Data Integration
- Data Take On and Migration
- Master Data
- Data Governance

… in fact any project which requires knowledge of the location of data in order to succeed.

What is needed is a map, or data model which represents the metadata (i.e. information about logical and well as physical table and attribute or field names, descriptions and relationships etc.) which underpins these systems of record.

In some instances, for example internally developed applications or relatively small packaged systems, it may be easy to find and make use of their metadata because of their relatively small size or because the developers are still around.

For many however, particularly those packages from SAP, Oracle and increasingly SaaS Cloud based vendors such as Salesforce, the task of discovery has remained stuck in the past with virtually no tools available for data professionals, rather than technical specialists, to use.

So given that this is a critical requirement, how is the question "Where's the data?" usually answered?

*"The team was originally informed that **no data model was available** for the SAP application or for SAP BW."*

Scott Delaney
BI Team Leader
Hydro Tasmania

How to answer the question: "Where's the data?"

With no really useful or accessible tools from major package or Information Management software vendors, the data professional is limited in what they can do for themselves.

The traditional approaches include:

Manual effort

This typically involves activities such as scouring the tables and fields in the relational database (RDBMS) System Catalogue for any information which might provide clues as to what data the tables contain, what attributes and fields they include and crucially the relationships between tables.

Another approach might be to try to discern some of this from the data itself through profiling.

The challenge in trying to do this manually is that virtually every packaged application that stores its data in an RDBMS does not store any 'useful' descriptive information about tables and fields, relationships or anything else in that database. Instead, each package has a proprietary data dictionary storing the required 'logical' view of the data. This does not exist in the System Catalogue.

> For example in SAP, amongst the 90,000+ tables which come with the standard system there is a table called MARA. Unless you happen to be an SAP specialist, looking at this table in the System Catalogue would not be very helpful in terms of understanding what the table contains and what other tables it is related to.
>
> As it happens MARA is the 'General Materials Master' table and in Silwood's development SAP system it has 1,823 'child' related tables and 61 'parent' related tables.

Also because these packages are designed to meet the needs of a number of business processes, the data model tends to be large. The table below gives an indication of the number of database tables each application comprises (before customisations) and illustrates just how impractical it is to try to manually identify the tables needed for a specific task.

SAP	95,000
Oracle EBusiness Suite	20,000
PeopleSoft	20,000
Siebel	4000
JD Edwards	4000
Salesforce	300

"Frankly we simply could not have done what we did without some way to extract that metadata automatically (without Safyr).

To discover it and hand enter it manually would have taken thousands of hours."

Lorin Yeaton
Boeing

Finally performing this task manually increases the risk of inaccuracy which can mean rework being required or the wrong data being introduced into the organisation’s information ecosystem.

Documentation

The first question to ask when referring to documentation relating to metadata is: “Does it exist?” and if it does exist can it be accessed easily and does it reflect any changes that have been made to the data model during implementation?

Some packages do provide a selection of data models in diagrammatic fashion. One example of this is JD Edwards which delivers a small number of data models in document form.

Delivering data models in this static way means that they would have to be updated to remain in line with any customisations made to the overall data model and of course any useful information cannot be shared easily with other tools.

Even if documentation does exist and is up to date it is difficult to see how it would be practical to find say, the table MARA (in SAP) and all its related tables in that way. The data models which underpin these systems are too large and complex for documentation to be of any real value.

Asking application or technical specialists

Our experience is that asking specialists from your own company is probably the most common method for getting hold of the metadata needed for Information Management projects.

They have the most familiarity with the application and its underlying data model, although even that can be patchy if for example they specialise in a particular module that is not relevant in the context of the requirement. They are also most likely to have access to any tools which are provided by the vendor which can identify the information required.

The challenge with this approach is that delays may occur as requests for information about tables could be added to a long list of tasks as staff in these roles are often busy.

In addition it is possible that the technical specialist does not quite grasp the business context of the request and therefore provides the wrong information – leading to rework being required and more delays.

Hire consultants

Another popular approach to solving the problem of finding the source tables which contain the data you need is to hire specialists or utilise those who are already engaged on the project.

Aside from the cost which can be considerable and ongoing there are potential delays if the consultants have to get to know the data model with all its customisations before they can start to give accurate answers. Even in situations where a systems integrator has implemented the package finding small subsets of the whole data model can be very consuming and expensive.

A further drawback to this approach is lack of control. Employing external consultants to do this means that an organisation’s own staff are not able to perform this work when they need to and may even have to bring consultants back after they have left to provide further data models to meet additional requirements.

Internet search

In recent times we have noticed an increase in the number of customers who come to Silwood after trying to locate data models from their chosen application package via an internet search.

The problem with trying to search the internet for a data model relating to a specific business term for a particular ERP is that the results may not be as accurate as necessary, perhaps because the versions are different and obviously anything found will not reflect the same customisations.

Often the models found using this method are part of documentation which means that they are static and there is no way to make use of that information in other software tools without rekeying information.

Finally it is often necessary to ask a technical specialist to interpret the model and augment it with relevant information, perhaps about table joins.

Best guess and hypothesis testing

Finally, when faced with this problem many companies use guesswork or hypothesis testing methods to try to find tables and related tables they need. They rely on data observation, insight and on trying to find an appropriate start point from which to launch a search.

This can be frustrating and time consuming.

“(By using Safyr) RS Components are succeeding in achieving a level of understanding of data in SAP that we previously thought impossible.”

Jean Ashby
RS Components

What about data modelling tools?

Data modelling tools would on the face of it appear to offer a solution based on their ability to reverse engineer RDBMS’ and create a data model from the tables, fields and relationships they find there.

There are three main reasons why this does not work for packaged applications. Firstly, as mentioned above their RDBMS’ do not contain any Primary or Secondary key constraints which means that the modelling tool will have nothing with which to create joins between tables.

Secondly, the RDBMS does not contain any logical names for tables and fields and so even if the relationships were available the physical names would need to be ‘translated’ in order to make sense to a data professional.

Thirdly, even if there was sufficient information for the modelling tool to create a meaningful model, the sheer size of the model produced would render the tool useless even if it could cope with thousands of tables involved. Imagine trying to navigate a single data model containing 95,000 entities and relationships.

What about the vendors?

There are two different categories of vendors who might be expected to provide a comprehensive metadata discovery and analysis tool and customers are often surprised when it becomes apparent that this is not the case.

The first is the vendors of the packaged application themselves. They have not provided any functionality which makes it easy and fast for data professionals to be able to access, navigate and find relevant metadata in the context of the project they are working on.

You may ask why not? There could be many reasons. Perhaps the package was never built with the outside world in mind so integration and master data was deemed to be an irrelevance? Perhaps they expected customers to only use the reporting and analytics provided by their system and that their data would never be combined with other sources to deliver greater insight.

Or maybe the vendors just thought it was not necessary because the only people who would need access to this type of information were technical specialists or consultants.

The second type of vendor who might be considered as needing to provide this functionality are those in the general group of Information Management or Enterprise Management software makers. These are the companies who provide the tools which in broad terms access, move, transform, integrate, master and govern data.

At best, these tools offer a few templates for common topics in the form of data model schemas in specific packages as a start point in the effort to gain enough knowledge of the customer’s system so that their products will hopefully meet their requirements.

Other solutions involve looking through lists of tables, which requires the customer to know what they are looking for before they start, which is not always the case or mere connectors. Connectors do just that - they connect – however they offer little in the way of help to navigate the package’s data model.

“The bottom line is that we are surprised that the major vendors are not doing more to introduce the sort of automation discussed.

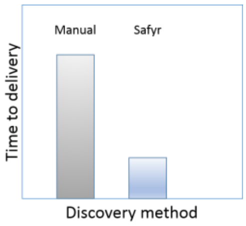
This is particularly true as Silwood Technology, a company we have been tracking for some time, does this sort of thing.”

Philip Howard
Bloor Research, 2014

Safyr – fast and easy metadata discovery for application packages

Safyr is a software product designed to give control back to the data professionals so that they can find what they need, when they need it – without having to resort to asking technical specialists or consultants. They do not need to guess, search the internet or rely on documentation or 3rd party tools that only provide a partial solution.

Safyr saves time and money and accelerates time to delivery. Safyr also reduces risk because data professionals are working with the metadata as implemented – with all the customisations that have been made to the data model of their system so they can be more confident that the data they are working with is correct.



What does Safyr do?

Safyr’s functionality can be broken down into three distinct phases.

Discover

This covers the important task of extracting the metadata from the target system. Most applications store their metadata in a series of Data Dictionary tables.

Safyr connects to these tables and reverse engineers their contents including any customisations that have made to the system. Safyr also collects the number of rows contained in each table.

Next the metadata is loaded into a Safyr repository and the connection with the source system is terminated.

© 2016-2020 Silwood Technology Limited

Safyr then creates the relationships between tables and in the case of most packages an application hierarchy which offers an alternative way of searching the application for what is required.

Scope

This is where the user begins the process of reducing the number of tables down to those that are relevant for the task at hand.

Depending on the requirement and potentially the amount of background knowledge a user has, there are different methods for beginning the search and analysis process.

Each of the mechanisms provides the user with access to the rich set of functionality with which they can quickly and easily isolate the tables and related tables they need. No prior knowledge of the application is required.

Once the tables are located they are grouped into Subject Areas for further analysis or utilisation.

Deliver

The final stage is to make use of the resulting Subject Areas. They can be visualised to aid communication and collaboration using Safyr’s ER Diagrammer or exported.

Subject Areas are used to make 3rd party tools more effective when working with metadata from large complex packages. They can be exported to the most popular data modelling tools (CA ERwin, Embarcadero ER/Studio, SAP PowerDesigner and Unicom System Architect). The contents can also be exported as XML or CSV files. These export capabilities mean that the metadata can be utilised and exploited with products from major Enterprise Information Management vendors.

> *“After doing a quick prototype metadata extract from SAP, the response has been very positive!*
>
> *I’m really grieving for the lost years without access to this tool (Safyr). It has met and exceeded my lofty expectations.”*
>
> *Brian Farish*
> *AMD*

There are also functions to allow comparison between partial or complete data models of two applications of the same type. For example two instances of SAP or two instances of Salesforce.

Finally, in Safyr, there is a product that allows data professionals working with some of the world’s largest and most complex packages to answer the question “Where’s the data?”

Want to learn more?

You can learn more about Safyr here.