



Introduction

If you don't understand what your software engineers are talking about, perhaps it's because they are using a vocabulary they invented for the problem they are solving. Engineers invent a vocabulary and data structure for each system they build and each problem they solve, and only the engineers who built the system understand this structure and vocabulary. Even other engineers must learn it in order to make the data usable. In most enterprises today, we have as many different ways to ask questions of our data as we have systems to store it. We have as many different vocabularies and data structures as we have systems.

Integraf is a middleware platform for integrating data across data sources and for building enterprise data services upon a canonical business vocabulary. It is based upon semantic technology--which stores meaning with the data and removes the need to define structure or data models. There is no pre-defined structure to which the data must be bound. This provides tremendous flexibility in what information can be stored, and it enables information to be combined and used both rapidly and in ways that are not possible with relational or traditional XML technology.

Perhaps most importantly, integraf has low cost of ownership. Engineers have traditionally been the stewards of systems and data. Integraf lowers the cost of your data by lowering the amount of engineering needed to make data accessible. In fact, integraf will enable enterprises to make better use of the business analysts and other people who understand the business and what the business needs.

The Cost of Data

One negative consequence of traditional data modeling is that if you have a variety of uses for your data, it must be stored in a variety of different structures and must also be constantly translated, copied, and kept in sync. One user's need is almost always at least slightly different from another user's need, and need. The intended use directly affects structure.

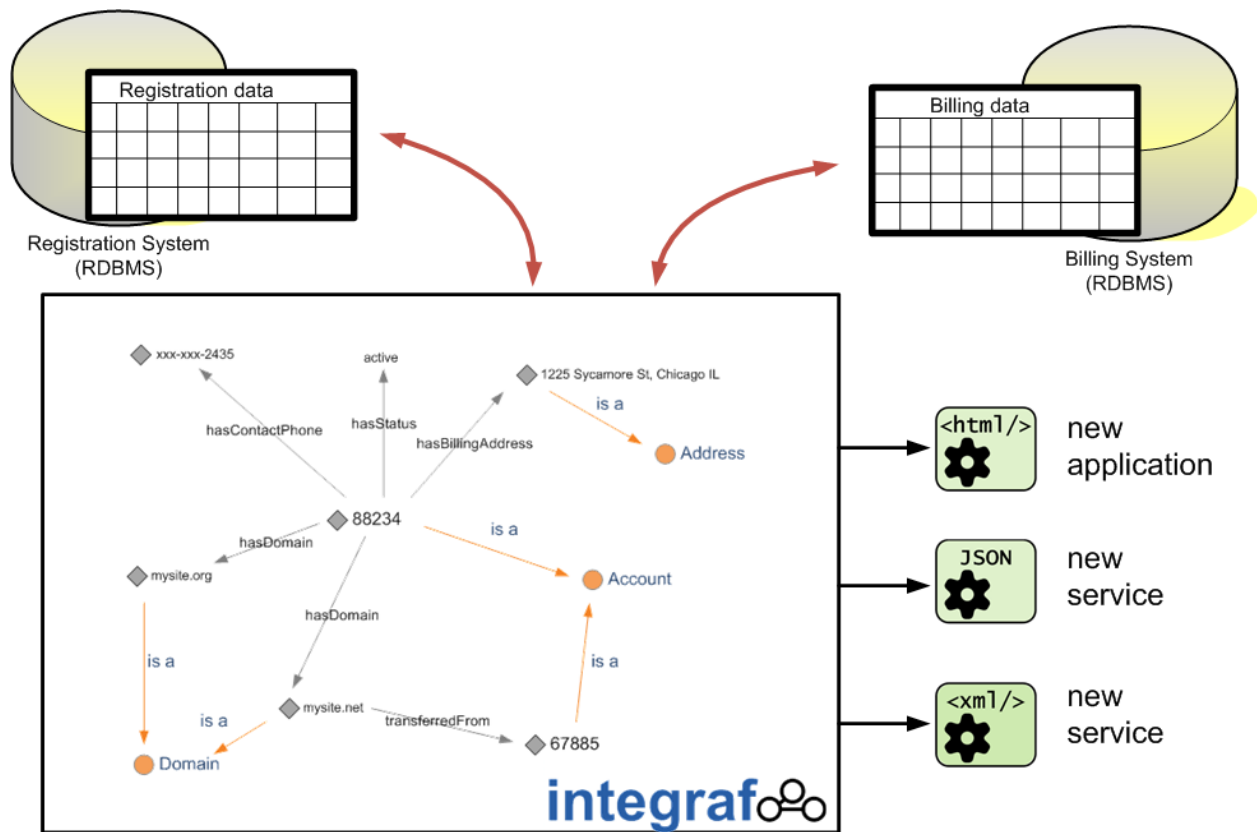
Data and data modeling incur very high engineering costs. Most of the cost of systems today and most of IT budgets come not from modernization or development of new capability but from maintaining data and the applications that use it. It comes from maintaining all the

different copies of data that are created for each use. It comes from maintaining all the code that copies data from one system to another. It comes from maintaining a staff whose purpose is simply to understand what the structures mean and how data is shuttled between systems. And it comes from the need to change the structures to accommodate new uses.

Integraf separates how data is used from how it is stored. It enables services and applications to be built upon the meaning of data, not the structure used to store it. There is no need to keep the information in a myriad of structures in order to satisfy a myriad of intended uses. It can be stored in one place to meet any number of intended uses.

Building upon what things mean

Integraf stores data about things in an enterprise. Each person, place, or thing is represented by a URL and each URL is a web page within integraf. These pages contain *semantic* data about the thing, and they contain references to additional data that integraf can use query to enrich what it knows about a thing. So when you build an application or service upon integraf, you are building it upon what the data means and the relationships between things in the data.

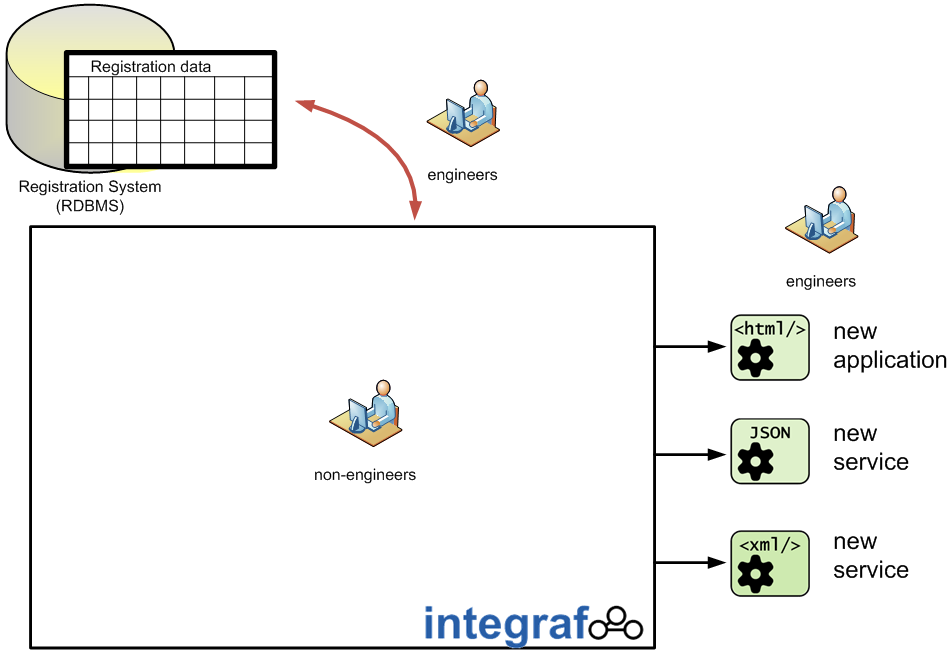


In the example above, integraf might store the basic information about account 88234 internally. But it also stores information that enables it to reach into the Registration and Billing systems for data that is specific to the need of an application or web service. The advantage integraf offers to is that applications built upon it are not coded for the other two systems. They are coded using a vocabulary that is common to the two of them. Not only does this make the application simpler (one model rather than two), it means source systems such as the Registration and Billing systems can be replaced without affecting the design of systems downstream. It lowers the cost of data all around.

The role of the engineer

Engineers often don't realize how much they get in the way. Enterprise systems built in the traditional sense, with XML, and databases, and Java class structures, and etc, are something that only they can do. But translating the world of business requirements into something an engineer writes is expensive and time consuming. Integraf is cheaper and easier largely because it reduces what an engineer must do and involves non-engineers directly in the creation of new applications and web services.

It does this in two principal ways. First, the semantic model at the heart of integraf and upon which applications and services are built is created and maintained by non-technical business experts. Engineers are responsible for the SQL and Java code that brings data to integraf. Non-engineers are responsible for almost



everything else. Data within integraf is stored in simple English sentences in text files that can be edited using any tool. Rules for data are also composed in English as are the descriptions of services that integraf actually uses to query data from source systems.

Why integraf

Systems within most enterprises today carry a heavy engineering cost and rely almost entirely upon engineers for design and maintenance. The technology they are based upon is inherently complex and difficult to integrate. This makes modernization and the realization of new capability elusive. Integraf offers a solution to these problems. It makes information less expensive. It enables better use of information. It facilitates new business capability and system modernization.