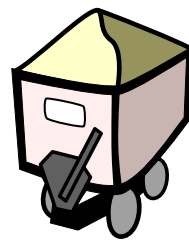


Database model and data entities

Juan J. García de Soria



ORE
Open Rating Environment



Chapter 1

Database model concepts

The database model for ORE will have to deal with the following kinds of data:

- User/Customer Accounts and related Subscriptions.
- Product/Service/Catalog definitions.
- Parameter Definitions and Values.
- Event Types and Rated Event Records.
- Invoices.
- Business procedures:
- Event guidance procedures.
- Event rating procedures.
- Subscription modifier procedures.
- Billing modifier procedures.

1.1 User and Customer Accounts

An Account will be the entity to represent a single customer or user in the service platform. Actually it represents a single billable entity, although it is not a monetary-only concept, since it could refer to several currencies at the same time, and will usually contain associated contact data.

That said, a single account will typically contain data for the following kinds of informations:

- Default, secondary currencies.
- Currency balances.

- Billing period and date information.
- Account status (active/blocked/deleted...).
- Commercial model information determining the Products available for the Account.
- Contact data, like:
- Personal data (Name, surname...).
- Address.
- Phone number(s).
- E-mail address(es).
- Additional parameter values as needed for the specific deployment.
- Product and associated Service Subscriptions, representing each service that the Account has subscribed to with the operator.

Each independent field or parameter value may be optional depending on specific deployment needs.

1.2 Product, Service and Catalog information

A Service is the definition of a kind of service that may be provided to a customer or user.

A Service definition may define specific Parameters that determine specific characteristics of the Service. For instance, for a basic phone Service, the phone number for the subscriber line would be a Parameter to be defined.

Services may be subclassed. Each Service may be a root Service, so that it has no parent Service, or may inherit from exactly one parent Service. When a Service inherits from a parent Service, it inherits all the Parameter defined for its parent, as well as any Parameter inherited by its parent Service.

Each root Service defines an Event Type (a definition for the data that defines each Service usage) that is inherited and cannot be modified by child Services.

In the same fashion, each root Service defines an event guidance procedure that will be used in order to find to which Account and Service Subscription every incoming event of the matching Event Type will be routed to.

Each Service may assign Values for the Parameters that itself defines, or for any Parameter inherited from its parent Service, if any. If a Service doesn't define a Value for a Parameter, it inherits any Value that may be defined or inherited by its parent Service, if any.

Products are combinations of Services that may be sold as a unit to a customer. Services themselves may not be sold. Each Product will group one or more Services that will be sold together under a commercial name.

Products may define Parameters and give Values to them too, in the same way as Services do.

Products may subclass other products. In this case Parameter and Value inheritance applies in the same way as it does for Services. Additionally, each Product that subclasses another Product inherits its parent Services, and may add new Services to the group of Services inherited. However, a child Product definition cannot remove any one of the Services inherited from its parent Product.

A Catalog is a group of Products that are available to a given class of customers. In each one of these groups, each individual Product may be mandatory or optional. Mandatory Products are automatically subscribed whenever an Account is assigned the containing Catalog. Optional Products may be subscribed or terminated by the customer afterwards in any free combination.