Version of this post: 0.3

The first document tries to give a representation of the Flex Family meta-model.
the second one gives an exemple of implemation of a specific Data Model.
It’s basic at this stage, and I’ll document more in depth, especially for the mapping between meta-model and a Custom model to be implemented.
A clear understanding of Flex Family is important in data Modeling, especially, the metamodel is useful to give ideas on relationship cardinality between entities
The document can be used as a base for a skills transfert to a partner, for instance, or in the context of a workshop with a customer.

A Flex Family allows implementing a functional domain. We can say that, given a Conceptual model to be implemented, we’ll use the Flex Family Maker which allows  as many entities as needed through the concept of *Asset* and *Parent Asset.*

Physically, we must be aware that numerous relationnal tables can be generated during this process in the underlying database.

## Rules that can be deduced from the model

- A flex family can have one or more *Parent Definition Asset Type*.
- A *Definition Asset Type* is a container for *Parent Asset Type* and *Asset Type*
- The different *Parent dDefinitions* are visible in the scope of a *Parent definition Type*
- A *Parent Asset Type* is a Container for *Asset Type*
- We can define multiple *Asset type* inside a *parent Asset Type*

- An *asset type* is in the scope of one and only one *Parent Asset Type*
- It's possible to create multiple *Definition Asset Type* within a *Parent definition Asset Type*
- A *Definition Asset* (Created with start menu) is attached to one and only one *Definition Asset Type*
- *An Asset Type is in the scope of one and only one Parent Asset Type*

Other observations

- A *Definition Asset* (created with a start menu) acts as a subtype for an *Asset Type*
- The scope of a *Definition Asset* is the *Parent Definition Type.* That means that a given *Definition Asset* will not be usable for an *Asset Type* defined in an other *Parent Definition Type* of the same *Flex Family*.
- A *Parent Asset Type* acts as scope of valid values for a parent during contribution
- A *Parent asset* instance is created in contribution mode.
- An *Asset* instance is created in contribution mode. If some parents are defined in the choosed *Definition Asset*, then the values proposed (in a drop Box for instance) will be in the scope of the *Parent Asset Type* owner of the given *Asset Type* . It can be a way to implement a filter for allowed values of parents. (Note: *a sample will be given later*)
- Each new object (*Parent Definition Asset Type, Definition Asset Type, Asset Type, Parent Asset Type, Definition Asset Type*) results in the creation of 9 to 11 tables

# Author


- 

  Patrick

  GPM Factory