

| | |
|-------------|---|
| Title | Alternate Keys |
| Product | Integranova User Interface Transformation Engines |
| Last update | 22/06/2011 |

1 Introduction

All the classes of a conceptual model (except the classes defined as *anonymous agent*) have an *Identification Function* composed by constant attributes of the own class, inherited attributes from the parent classes or attributes of related classes (identification dependency).

As it is known, the aim of the class' *Identification Function* is to identify the instances of the class in a unique way, which means that the combination of attributes' values of the *Identification Function* will be unique.

However, sometimes analysts model the Identification Function of a class using attributes which are not very intuitive for the users of the Integranova applications to identify the class' instances.

For this reason and to enhance the usability of the Integranova applications, the *Alternate Keys* feature has been designed in order to display to the user the instances of a class in a more intuitive or friendly way. This new solution will provide the mechanism to model one or more *alternate keys* for a class and select one of them to be used by the user interface.

2 Description

As it has been mentioned, with this new feature analysts are able to model *alternate keys*. Of course, like the primary or main *Identification Function* of a class, these *alternate keys* will identify the class's instances in a unique way, as well.

The *alternate keys* of a class will only be able to be composed by **constant** or **variable attributes** of the class. This means that derived attributes can not be used. Furthermore, the *alternate keys* of a class can not use attributes of related classes. So, the *alternate keys* will only be composed by attributes of the own class. In addition, the attributes of the *alternate keys* will not accept null values.

On the other hand, the *alternate keys* will not be able to be formed by *Autonumeric*, *Text*, *BLOB* or *Password* type attributes. The rest of **attribute's types** can be used without restriction. The attribute's repetition in different *Identification Functions* of a class (primary or alternate keys) is also allowed.

Referring to the **inheritance relationships**, the same rules for the primary *Identification Function* manage the *alternate keys*' behavior. The *alternate keys* will be inherited from parent to child classes.

3 How does this feature work?

Consequently, the elements that are affected by this feature in the Integranova User Interfaces are:

- ◆ **Object-valued inbound arguments:** In the Service Interaction Units (inbound), the object-valued inbound arguments will be composed by the fields or editors that match the attributes of the selected *Identification Function* (primary or alternate keys) of the domain's class.

- ◆ **Object-valued outbound arguments:** In the Service Interaction Units (outbound), the object-valued outbound arguments will be composed by the fields or editors that match the attributes of the selected *Identification Function* (primary or alternate keys) of the domain's class.
- ◆ **Object-valued filter variables:** In the Population Interaction Units where any filter containing an object-valued filter variable is defined, these object-valued filter variables will be formed by the fields or editors that match the attributes of the selected *Identification Function* (primary or alternate keys) of the domain's class.
- ◆ **Instance selector (IIUs):** Finally, the instance selector of the Instance Interaction Units will be composed by the fields or editors that match the attributes of the selected *Identification Function* (primary or alternate keys) of the domain's class.

It is important to mention that the default *Identification Function* of a class will be the primary one. In case that a class belongs to an inheritance network and it does not define its own primary *Identification Function*, although it defines *alternate keys*, the default *Identification Function* of a class will be the inherited primary *Identification Function*. This *default Identification Function* can be changed in the presentation model.