

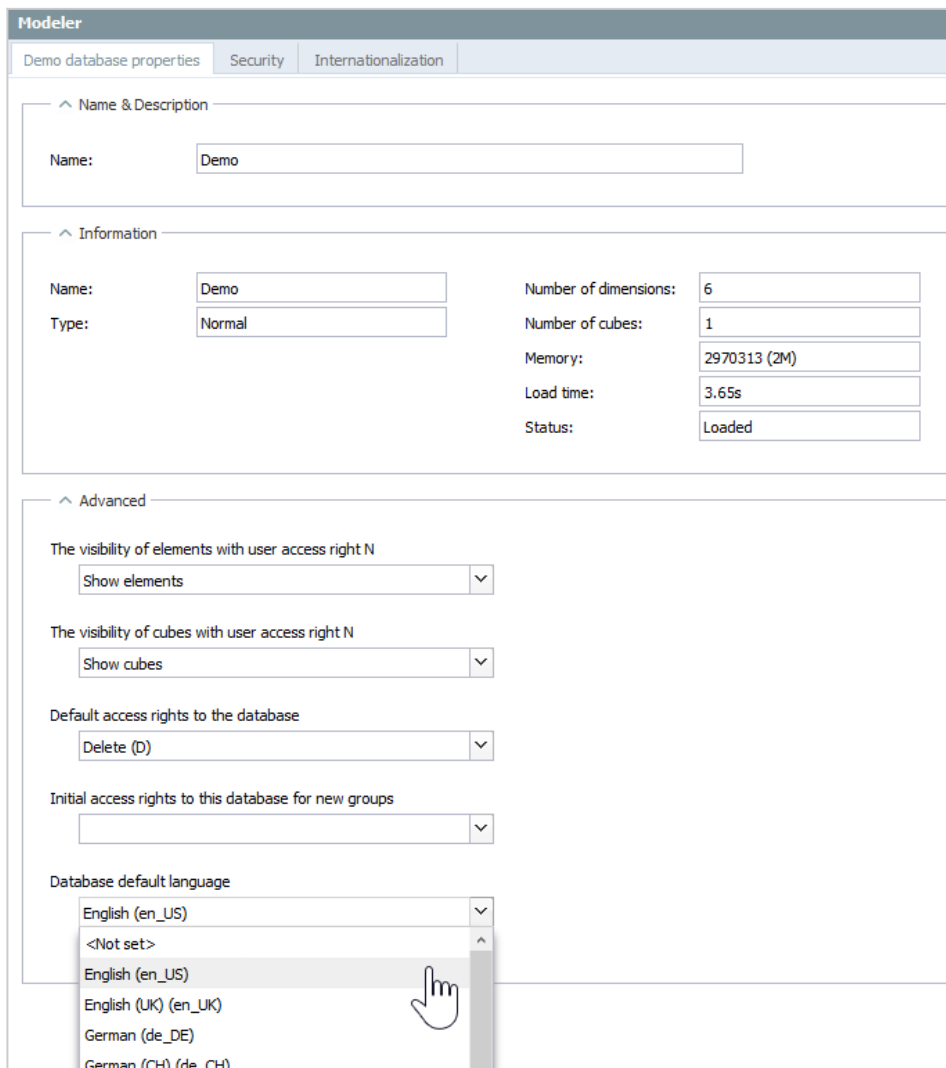
# Internationalization of Databases

In many scenarios, such as in multi-national corporations, the users of an application (or model) built with Jedox have different spoken languages, and the application designer is tasked with storing model contents (names of dimensions, cubes, elements) in several languages.

Jedox offers native support for internationalizing a database. The designer can define a default language for a database (such as English), and then specify several additional languages that the database supports (such as French or German). The database can store translated values for most of the names of database objects. In some scenarios, specific localized object names can be retrieved, but in others (namely when dealing with dimension attribute values), Jedox will implicitly return localized values in the chosen language of the current user.

## Defining the default language of a database

The default language of a database model can be defined in the Properties page of the database in Modeler, in the Advanced section (see screenshot below). All [languages](#) that are generally supported by Jedox Web can be chosen.



**Modeler**

Demo database properties | Security | Internationalization

^ Name & Description

Name: Demo

^ Information

Name: Demo      Number of dimensions: 6

Type: Normal      Number of cubes: 1

Memory: 2970313 (2M)

Load time: 3.65s

Status: Loaded

^ Advanced

The visibility of elements with user access right N  
Show elements

The visibility of cubes with user access right N  
Show cubes

Default access rights to the database  
Delete (D)

Initial access rights to this database for new groups

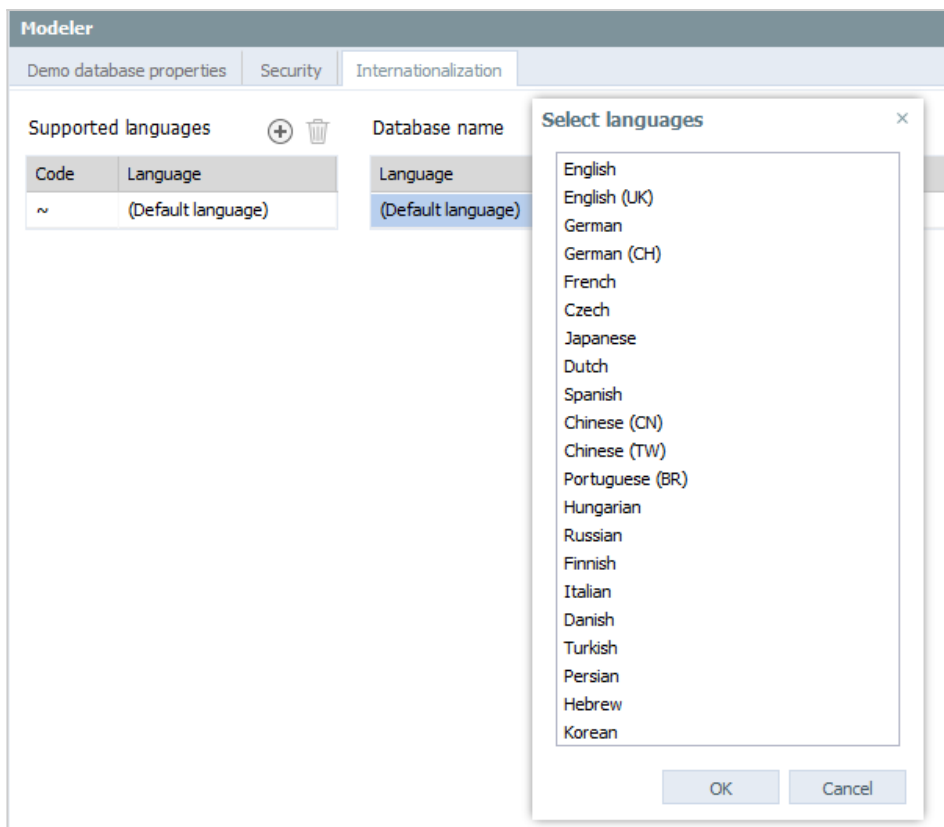
Database default language

- English (en\_US)
- <Not set>
- English (en\_US)
- English (UK) (en\_UK)
- German (de\_DE)
- German (CH) (de\_CH)

The default language of the database is checked in various circumstances, such as when the Time Editor is used on a time dimension. In this case, to generate attribute values in the desired language, the Time Editor has to be aware of the intended default language.

## Defining additional languages

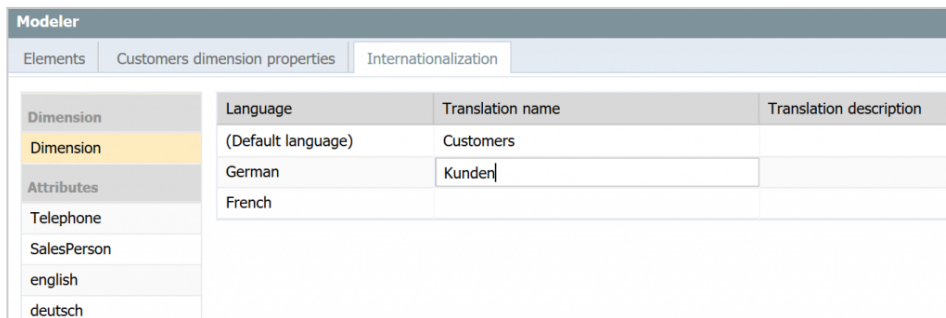
Besides the defined default language of a database, additional languages may be stored as well. These languages can be set in the Modeler under the Internationalization tab (see screenshot below). Again, [all supported languages of Jedox Web](#) are available. For each language that is defined, Jedox In-Memory DB creates storage space for translations in the database.



## Translating database contents

In Modeler, objects such as databases, cubes, and dimensions all have their own internationalization settings. When opening these settings,

the user can translate the objects into the languages that have been added to the database. In the screenshot below, a German translation for the Customer dimension has been entered.



Language	Translation name	Translation description
(Default language)	Customers	
German	Kunden	
French		

For elements, translations are handled slightly differently. Elements themselves don't have translated values; the values of attributes for the element are translated. For example, consider a Products dimension with element names consisting of technical product codes. Product names may be stored in a dimension attribute, e.g. "Product Name". Translations would then be made for the values of this attribute.

Besides showing the Attribute columns for the default language, the element grid in Modeler can also show attributes from one or more localizations. To do this, open the column header menu, and, in the Columns sub-menu, check the attributes/translations you'd like to display.

Internationalization

4 5 6 7 8

Parent element	Weight	Name	Name (German)	City	Latitude
All Customers	+1	All Customers	Alle Kunden		
All Customers	+1	Not applicable	Nicht anwendbar		
All Customers	+1	Unassigned	Nicht zugeordnet		
All Customers	+1	East	Ost		
East	+1	Austria	Österreich		48.200000000000
AT	+1	Another Sporting Good...	Another Sporting Good...		48.0500900
AT	+1	Basic Sports Equipment	Basic Sports Equipment		48.2035300
AT	+1	Central Discount Store	Central Discount Store		47.0707140
AT	+1	City Cycling	City Cycling		
AT	+1	Commendable Bikes	Commendable Bikes	Linz	
AT	+1	Elite Bikes	Elite Bikes	Wien	
AT	+1	Exclusive Bicycle Mart	Exclusive Bicycle Mart	Innsbruck	
AT	+1	Fun Toys and Bikes	Fun Toys and Bikes	Wörgl	
AT	+1	Future Bikes	Future Bikes	Klagenfurt am Wörth...	
AT	+1	Locks Company	Locks Company	Wels	
AT	+1	Number 1 Supply	Number 1 Supply	Stockerau	
AT	+1	Road-Way Mart	Road-Way Mart	Wien	
AT	+1	Small Cycle Store	Small Cycle Store	Villach	
AT	+1	The Bicycle Accessories	The Bicycle Accessories	Fieberbrunn	
AT	+1	Third Bike Store	Third Bike Store	Telfs	
AT	+1	Unique Bikes	Unique Bikes	Enns	

Displaying localized attribute values

## Returning localized values

The translations of database names, cube names, and attribute names (or their descriptions) can be retrieved by creating a PALO.DATA formula for the appropriate storage.

- To retrieve a specific translation of a database name, retrieve the value from cube #\_CONFIGURATION. For example, in a spreadsheet cell, for the German translation:

```
=PALO.DATAC("localhost/Biker", "#_CONFIGURATION", "DatabaseName", "de_DE")
```

- To retrieve a specific translation of a dimension name,

retrieve the value from cube #\_#\_DIMENSION\_.

For example, in a spreadsheet cell, for the French translation:

```
=PALO.DATAC("localhost/Biker", "#_#_DIMENSION_", "Name", "Customers", "fr_FR")
```

- To retrieve a specific translation of a cube name, retrieve the value from cube #\_#\_CUBE\_.

For example, in a spreadsheet cell, for the Spanish translation:

```
=PALO.DATAC("localhost/Biker", "#_#_CUBE_", "Name", "Orders", "es_ES")
```

- To retrieve a translation of an attribute name, description, or another attribute, retrieve the value from the new cube #\_<dimension-name>\_METAATTRIBUTES.

For example, in a spreadsheet cell, for the Spanish translation:

```
=PALO.DATAC("localhost/Biker", "#_Customers_METAATTRIBUTES", "Name", "City", "es_ES")
```

You can retrieve localized values for element attributes in the same way, by making an explicit request to the dimension's attribute cube. However, for attributes, Jedox also does implicit localization.

If you are retrieving an attribute as part of a subset in an ESELECT formula or to use in a View, Jedox will automatically try to retrieve a localized value; for this, it will check for localizations in the language defined by the current user. If there is no translated value in the user's

language, or if that language hasn't been added to the database at all, Jedox will automatically revert to the default language to ensure that a value is returned in any case.

Consider a Customers dimension, where element names are customer companies. The dimension has an attribute called "City". In the default language, English city names are stored on this attribute for each customer. If German was added as a language to the database, then German city names will be stored the German locale of the City attribute.

The spreadsheet formula

=PALO.ESELECT("localhost/Biker","Customers","Elite Bikes",0,,"City",0) will return the value of the City attribute for the element Elite Bikes. If the current user has enabled German as a language option in a database whose default language is English, the database will try to retrieve and display the German value for "City". However, if no German value exists, it will try to retrieve and display the City value in the default language (English).

**Note:** the function PALO.ENAME does **not** support implicit attribute translation.

A similar mechanism is used if a PALO.SUBSET formula specifies an alias to return. In Subset, additional handling was added for the AFILTER() formula of a subset. The user can control whether the filter expressions of an AFilter should be applied to the default language or a user's current language. The latter can be useful, for example, if user

input is passed to AFilter, such as when a dimension should be filtered by City attribute. Users would be able to enter the filter term in their native language (provided that translations for the attribute are maintained by the designer). Again, if the localized value of the given attribute is empty, the filter will fall back on the default language, and check the filtering expression against that value. If the localized value is filled, the fallback filter is not applied.

---