Valid as per: July 2000

LSA Developer Pack

LSA-DesignTracker

A component of the Lotus Solution Architecure



Design Tracker

Lotus Solution Architecture

LSA Developer Pack that is composed of various libraries in order to create a single source, reuse environment.

The Design Tracker is integral part of the LSA development environment which helps to get an overview of the inheriance structure of the design of all databases, that are exisiting on given servers, on the local disk or in created projectfolders. Furthermore it offers a set of functions refreshing the database design properties.

The Designer task and the inheriance conflict

To use a consistent design for multiple databases, database designers can associate databases or elements within databases with a master template. Designers can manually synchronize databases with a master template, but more often they rely on the Designer task to do this. When a master template design changes, the Designer task updates all databases that inherit their designs from the master template. Designer runs daily by default at 1 am. The Updall task, which runs by default at 2 am, updates the view indexes of databases changed by Designer. For a server's Designer task to update databases, you must create a replica of the master template on each server that stores databases that inherit from the master template.

After updating database designs, the Designer task also reloads the LDAP schema on a Domino server that runs the LDAP service.

You can't run the Designer task against a specific database or folder. It runs only against all databases on a server.

Problem and motivation:

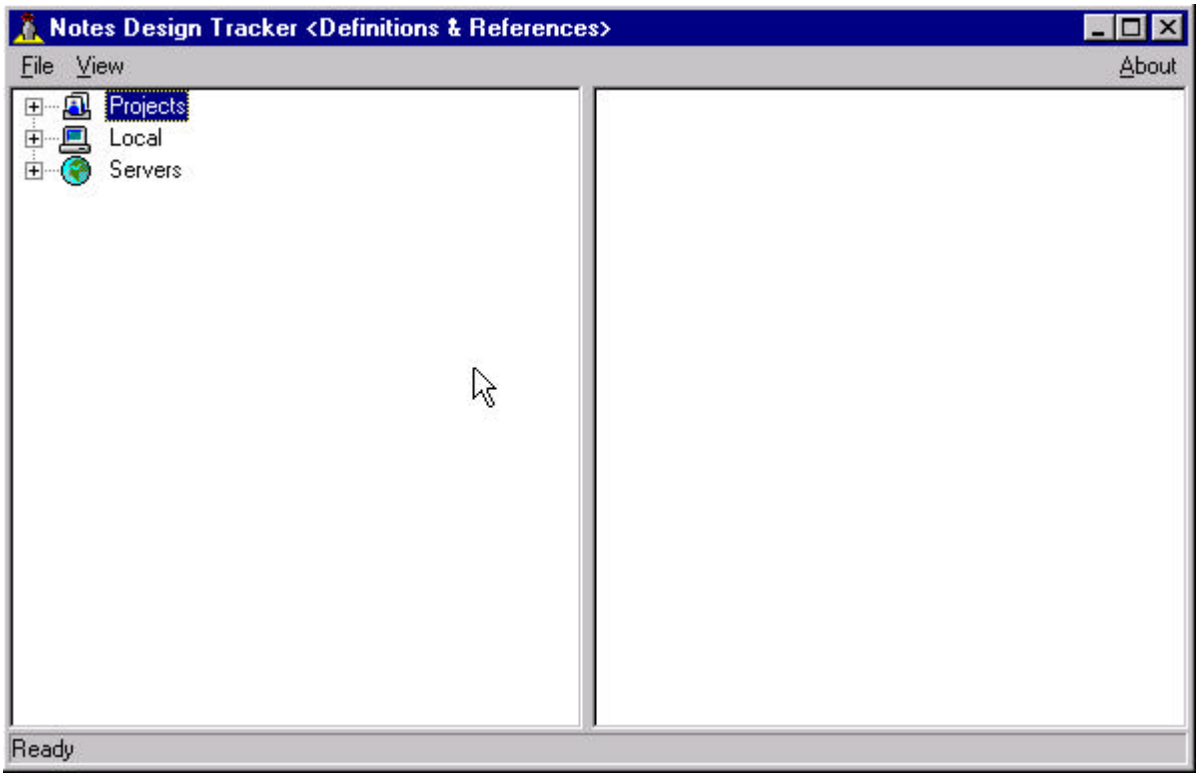
In Lotus Domino Designer, it is not only possible to create a database inheriting designelements of only one database template, you could also build the design repertoire with single design element of other database templates. Thus the design of one database could consist of the combination of several designelements with diverse origins.

Synchronizing this created database with the diverse templates creates a conflict in receiving a correct update. The reason lies in the randomized manner of synchronized process by the designer task. If there exist a tree like inheriance structure, it isn’t guaranteed that the designer task will start at the root position. To refresh the database in a strict order so that the origin database will be refreshed first, the design tracker creates a datastructure to accomplish perfectly this synchronizing task.

Operating with the design tracker:

Starting the design tracker tool a window is opened, which is splitted into two parts. The **left window** show three collapsed folders

1. Projects
2. Local
3. Servers

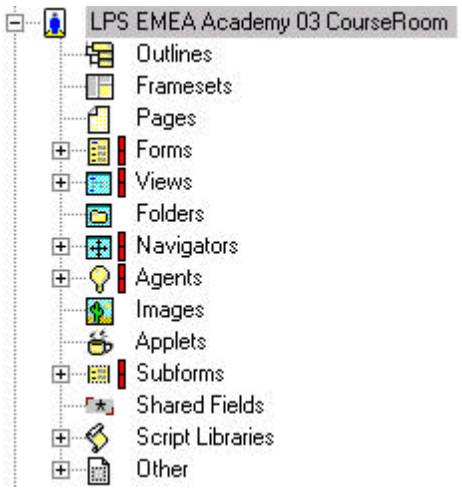


In this window all existing databases are respresented, which are on your local disk, **Local**, and on connected servers, **Servers**.
In the folder **Projects** you can build your own projects to fill them with proper databases choosing from the **Local**.

Context menues of the left window.
While standing on a subfolder of Local you can use a context menu, that offers to add the selected subfolder to the Projects folder. On the other side exist in context for removing a selected project out of the Projects folder.
Using the Refresh Button creates an update of the datastructure of all existing databases, so any changes done in the background are fetch up.



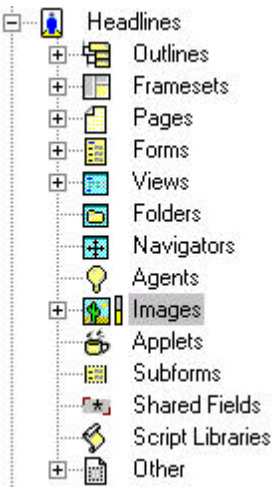
Expanding a database you see the list of all possible design elements (see at the picture on the right). There are only cross twistie for those design elements, which have instances.



Watching at the listed design elements there are some **red** bars.
They are symbolizing *errors*, which could have one of these reasons:

	red bars - possible errors
1.	There has happened a change in the design element, either in the template database or in the derivated database
2.	Inheritance isn't possible, because you are working on a local replica

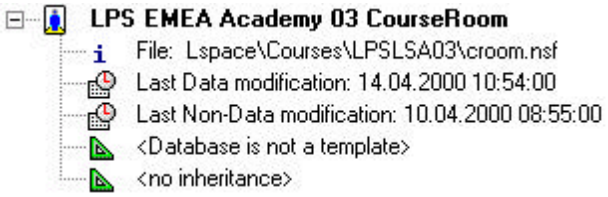
There could also exist some **yellow** bars symbolizing *warnings*, making aware of some validity conflicts in the actually state of this kind of design element.



The reasons for these visible warnings could be:

	yellow bars - possible warnings
1.	The database use itself as a template.
2	If the database design isn't allow to refresh or modify.
3	If this single design element inherits from another template as the database do.

Positioning the mouse on a database or on an instance of a design element you get more detailed information about it in the **info window**, the right one. The example print shows it for an database.

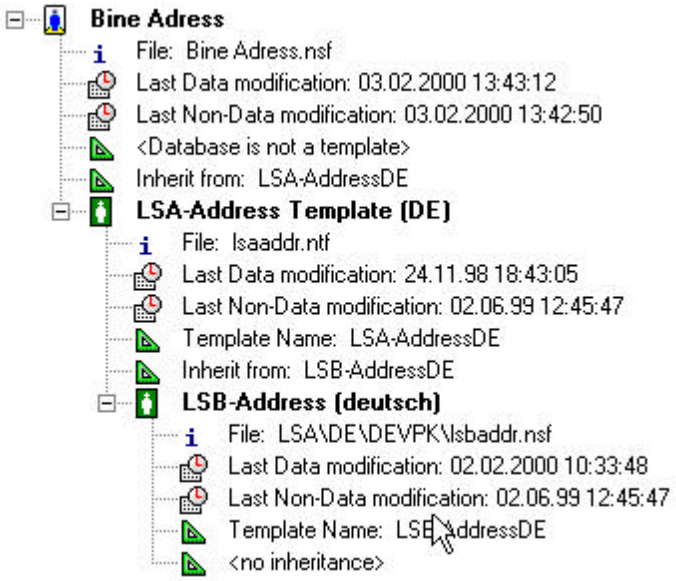


By the way you can choose between two kinds of view setting in the menu bar, either **References & Definitions** or **Definitions & References**. The context menues in this info window offers functionalities to refresh your database design in a different manner. There are two possibilies:

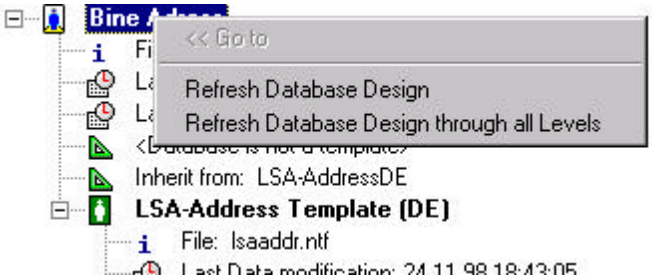
	Context menu
References & Definitions	Refresh database design or refresh through all levels respectively
Definitions & References	Push database design down one level or down through all levels respectively

Depending of the view context you only have a selected set of functions.

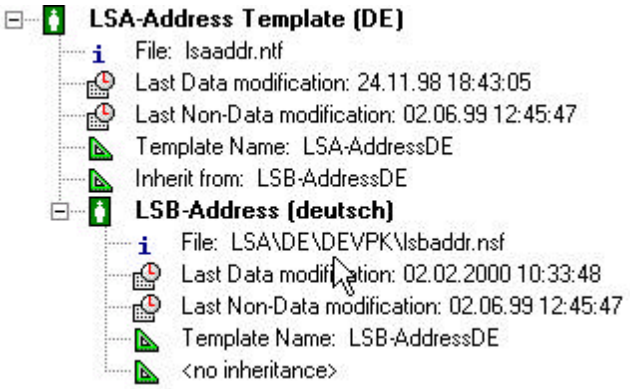
If you are in the view of ***References & Definitions*** and set the mouse cursor on a database, you see detailed informations about this database, about the originally template and its inheriance structure, if it's exist.



In this view you get a context menu to refresh database design. Opening the context menu while positioning the mouse cursor on the template (e.g. LSA-Adress Template) the **go to** is highlighted. This command switchs to the selected item, either the template or the database.

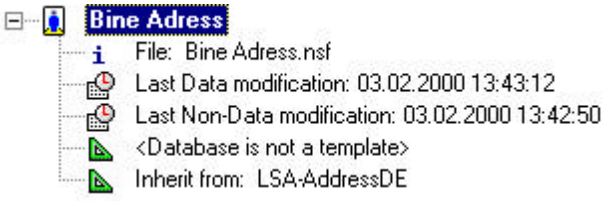


If you choose a template in the right window, in this view you get informations about the properties of this template and about ist inheriance structure, if it's exist.

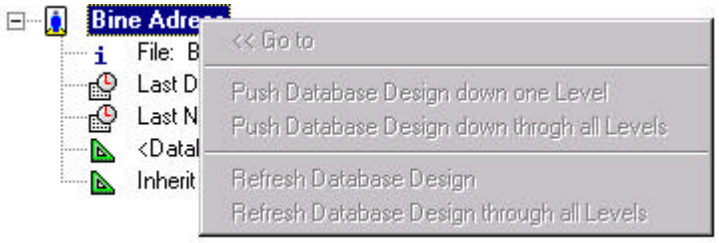


The context menues are the same like viewing informations for a database, like above.

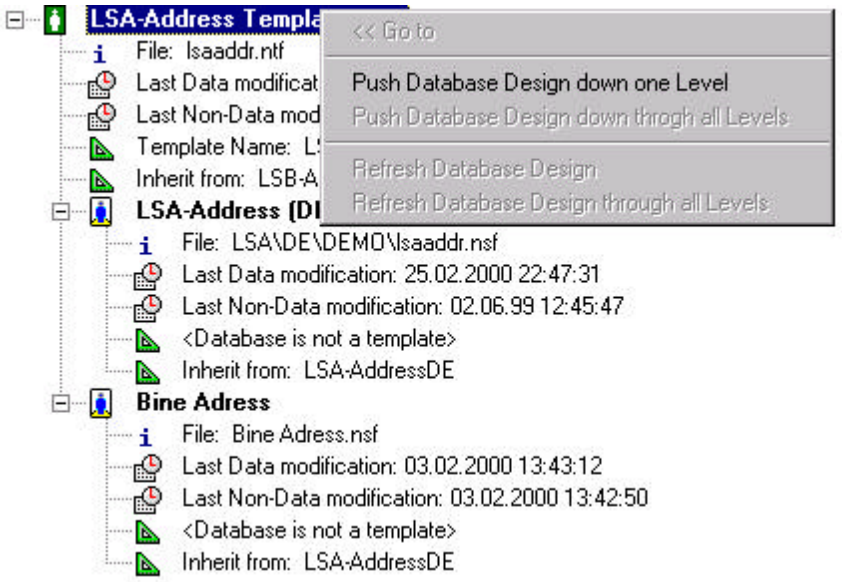
If you see the info window in the view of ***Definitions & References*** while positioning the mouse cursor of an database, you only get informations about the properties database, like the filename or if it is a template.



In this case all points of the context menu are grey, either sense nor association is built to offer these commands.



In the context of an template you see all derivated databases. On template level you are able to use the push function, passing all design changes on the database. On database level you are able to refresh the database design, fetching all design changes from template.



The difference betw een one level and all levels means that design of a database pushed dow n one level is passed only to the next derivated “generation”. Pushing it dow n through all level means, that it is pushed dow n to every derivated database. Nevertheless the picture is only an example for the one level, because it show s, that this template has only one generation.

Opening the context menu for the database, Refesh Database Design is highlighted.

