ROBELLE di Roberto Lano
Via Pallanza 3
10153 Torino (Italy)

Phone: (+39) 3484047935
Web: *www.robelleconsulting.com*
E-mail: info@robelleconsulting.com

BAAN ARCHIVING TOOLKIT

VERSION 1.1 FOR BAAN IV

Table of Contents

ARCHIVING

What is Archiving

The database for an ERP system grows very quickly, and after many years of intensive usage this has a negative impact on system performances and, in general, user productivity.
Archiving allows to maintain better the database.

Why Archiving

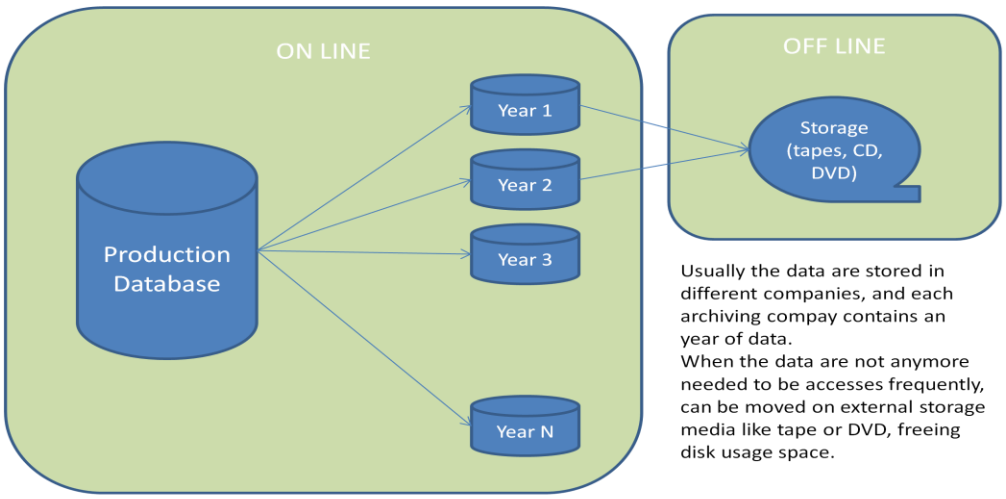
In almost each ERP installation there are a lot of data that have completed their life cycle, and can be moved offline or even deleted.

The archiving procedures allows to:

- Improve system performances
- Free-up online disk space
- Obtain better data management
- Delete obsolete data

When to Archive

First at all, companies have to adopt an "Archiving Strategy", defining what kind of data to move to an archiving company, what kind of data could be defined as not necessary anymore, and when to apply this strategy. So a period of archiving must be defined, usually could be an year (solar or financial). A basic schema could be defined as this:



Basic archiving procedure

In Baan, data is stored using "Companies". So, there is one (or more, in case of multi-company structure) production company, and some "Archiving companies" have to be created.
All those companies are accessible on-line for reporting data, but it is possible to store offline (on tape, CD, DVD or other external media storage) archiving companies containing data that are not necessary to consult online frequently.

Kind of data
Basically, it is possible to define some category of data:

- **Master data** – Those tables are even known as 'static' tables because doesn't change frequently, like Master Items, Customers, Suppliers, Terms of Payment, etc. This kind of data usually is only copied in the archiving company but not removed from the original company.
- **Dynamic data –** Those tables are the core of the system, they are updated constantly, like Sales Orders, Shipments, Ledger Journals, etc. This kind of data usually is moved to the archiving company and removed from the production company.
- **Obsolete data –** Data that isn't anymore necessary to maintain or use (maybe they weren't used at all), so they can be removed from the production company.
- **Texts –** Texts are usually linked to other data, they could be copied in the archiving company or not. Usually the text aren't removed from the original company by means of the archiving procedure, there is a standard Baan session that removes all unused texts.

Standard Baan Archiving
Baan offers a wide range of archiving sessions, but unfortunately they:

- Sometimes are not compliant with the business requirement for your company, by example some data has for your company a different life cycle than the standard (different concepts of "obsolete" data).
- Cannot consider the customizations, new tables could be added and linked to the standard tables and those session doesn't obviously knows the existence of them.
- Doesn't archive completely new, bolt-on tables.

How to solve this problem?
There are basically three solutions for this problem:

- Customize the archiving sessions. This could be difficult or expensive to do due the lack of the sources or the cost of development resources.
- Don't use the archiving sessions, and develop completely new ones, maybe using different tools. This could be accomplished, but, again, the company have to spend a lot of time doing this.
- Use a toolkit that permits to define the archiving procedures in a simple but effective way, without to write code. This is the BAT solution way.

BAAN ARCHIVING TOOLKIT

This tool was written with the intent of override the limitations of the standard archiving sessions, and all without to wrote any kind of code in Baan Tools or in other languages.
Some knowledge of the Baan data dictionary structure and how it works the query language are needed to write the procedures.

Parameters

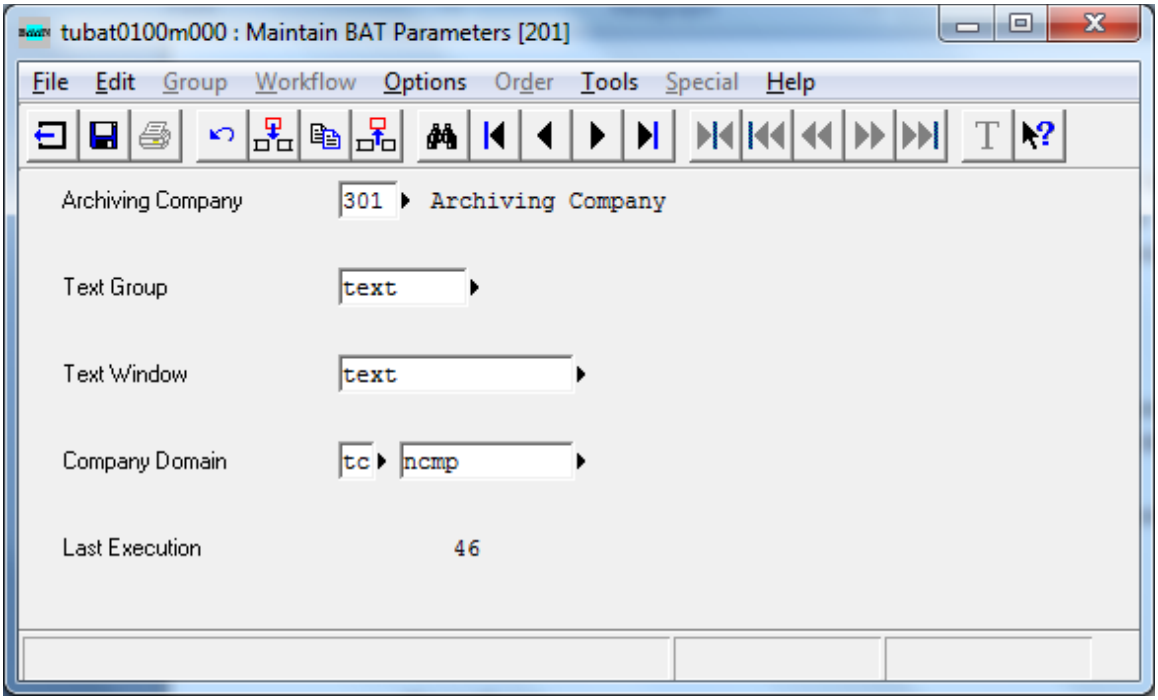
Some parameters are to be set:

The default archiving company (the default for the session of procedure execution)

The default text window and group (used to store text related to the conditions)

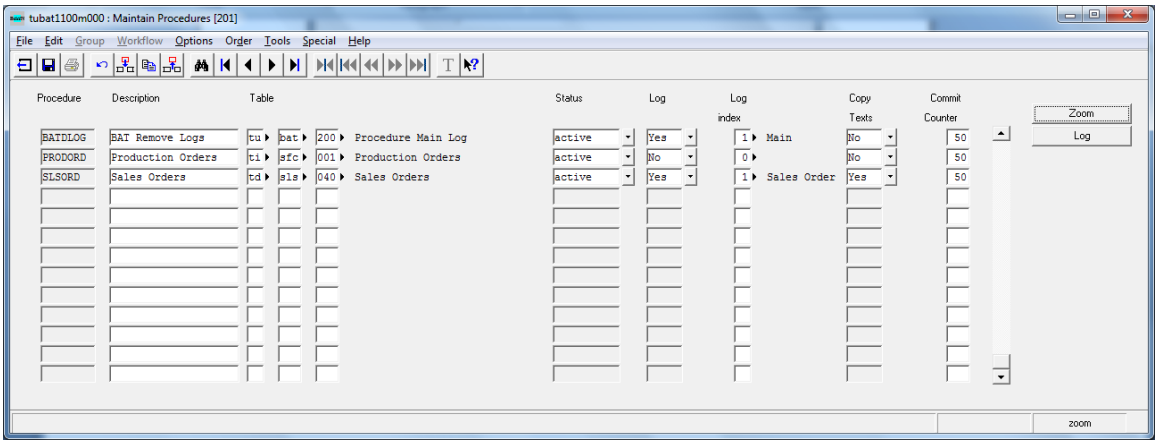
The default domain for company (usually "tcncmp")

In this table is also stored and displayed only the procedure execution counter.

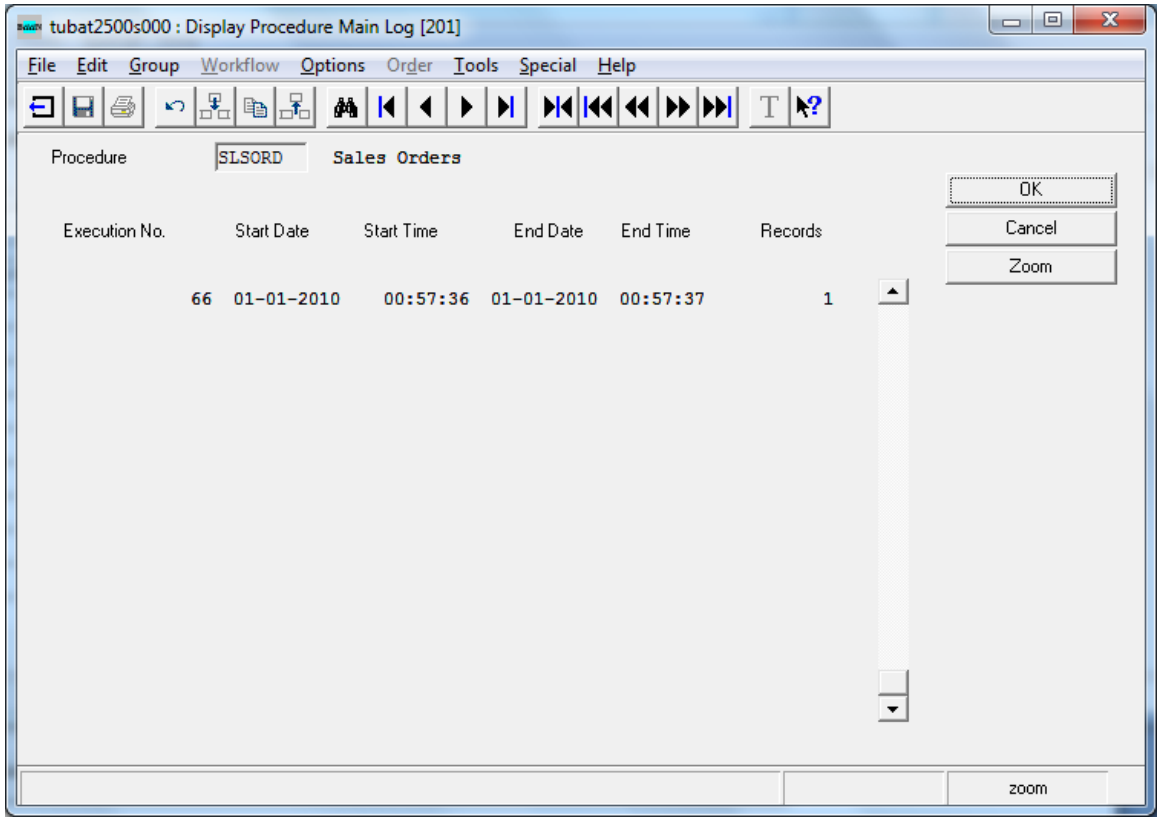


Procedures

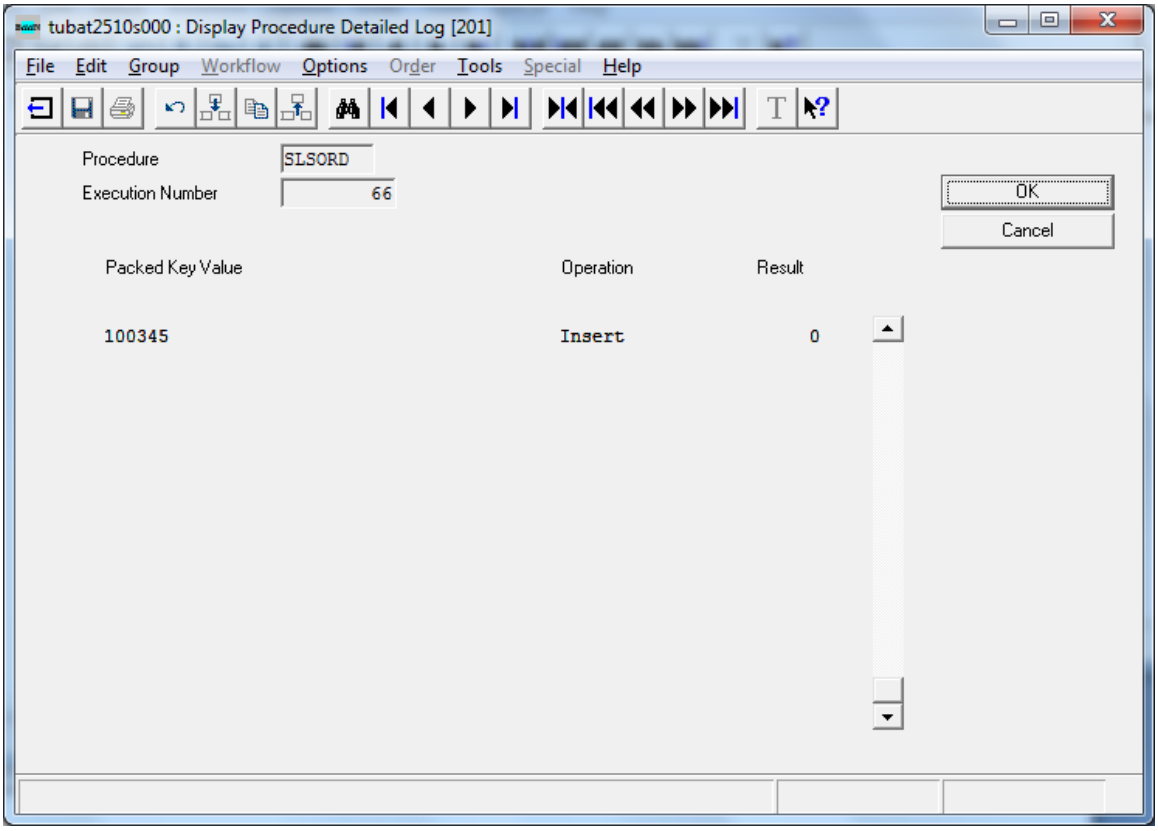
The procedure defines a main table (the major table involved in this part of archiving), if it is necessary to keep a detailed log of the operations and in this case which key of the main table is to be tracked, if the linked text are to be copied in the archiving company, and how many records are to be committed to the database. The procedure have 3 statuses: Simulation, Active, Expired. Expired procedures cannot be run anymore (This is to avoid the use of obsolete procedures). Simulation procedures cannot be run, only compiled, just to test correctness of the procedure. Active procedures can be compiled and run.



The Zoom button allows to access to the Procedure Steps session, the Log button to the display main log session:



If there is the "detailed log" activated, through the Zoom button is possible to see even the detailed log, like this:



In this example, the procedure "SLSORD" was run at 01/01/2010 from 00:56:36 to 00:56:37, and the record with index1 (tdsls040.orno) equal to 100345 (and all the other steps) were inserted in the archiving company without errors (result = 0).

Procedure Steps

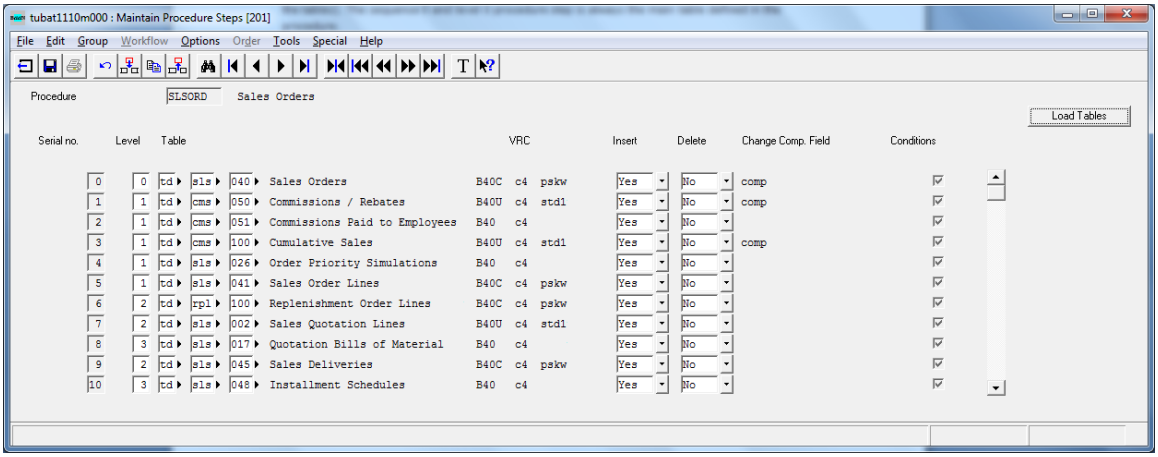
The procedure steps have a sequence (order on which are executed) and a level (nesting level of the tables). The sequence 0 and level 0 procedure step is always the main table defined in the procedure.

It is possible to define the way the data must be changed by setting the "insert" and the "delete" flags.

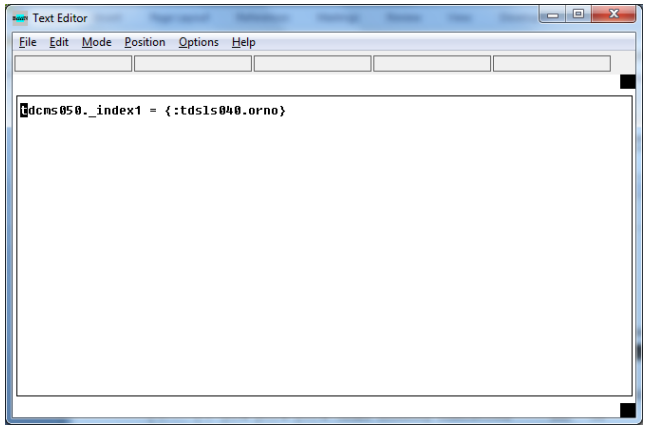
The possibilities are 4:

Insert / Delete	FALSE	TRUE
FALSE	No action is done for this table	The data is deleted from the actual company
TRUE	The data is copied in the archiving company ***NOTE: This is the preferred option to 'test' the procedure before to risk to delete the actual data.***	The data is moved from the actual company to the archiving company

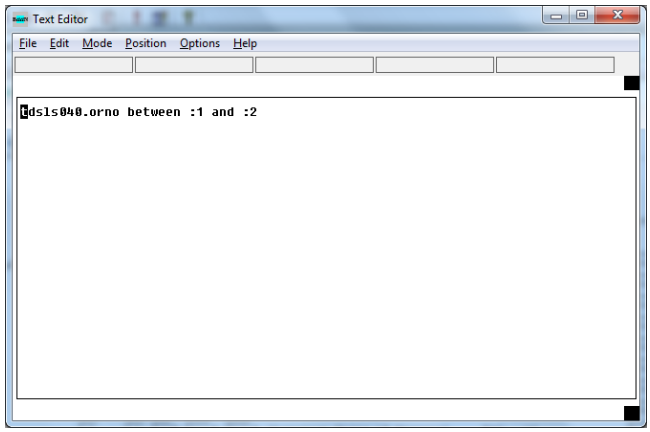
In some cases, there is in the table a field that contains the company number, for the multi-company structure purposes. This field must be not simply copied in the archiving company but needs to be re-codified before the copy. The definition of which field is automatically taken from the parameter that defines the "company" domain.



There is also the need of a 'condition' that links this table from the parent table, or, in case of the main table, the definition of how the selection of data is to be done. For the main table conditions it is possible to define and use parameters. The condition have the same syntax as for the "where" clause of a standard Baan query. All the possible clauses are possible, even the "exists" with sub-queries.



In this example the way the tdcmcs050 is linked to the previous level table (the tdsls040).



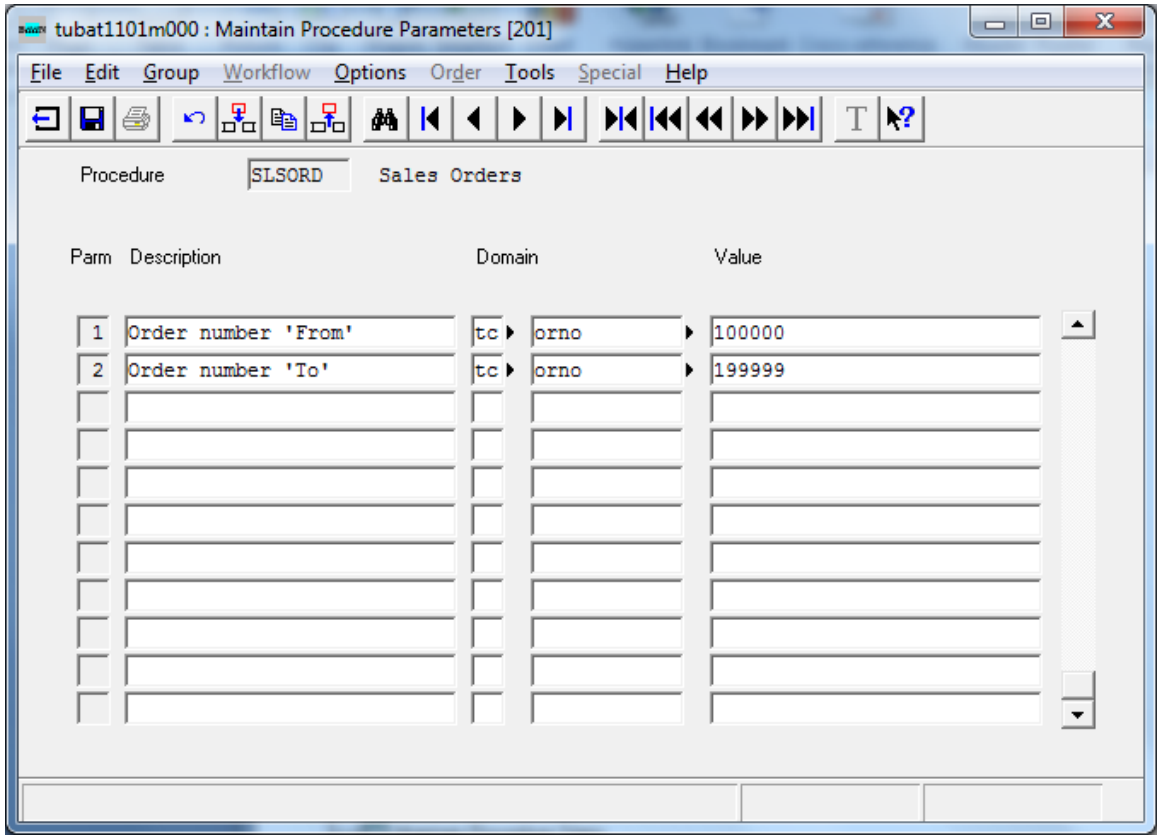
In this example how the main table is selected, using the parameters 1 and 2 (see Procedure Parameters).

Note that all those conditions (with the except of the main table condition, which is to be written manually), are automatically written by the "Load Tables" button, that deletes all the conditions but the level 0 condition, and starting from the main table loads all the steps linked, nesting them according to the data dictionary definition.
In this example, all the steps and the conditions were obtained automatically by pressing the Load Tables button.
However, it is always possible to change, delete, add steps and to modify the conditions.

Procedure Parameters

The procedure parameters could be used to define the condition of the selection related to the main table of a procedure.

They are identified by a number from 1 to 9, a description, a domain and a value. The value could be changed before to execute the procedure.

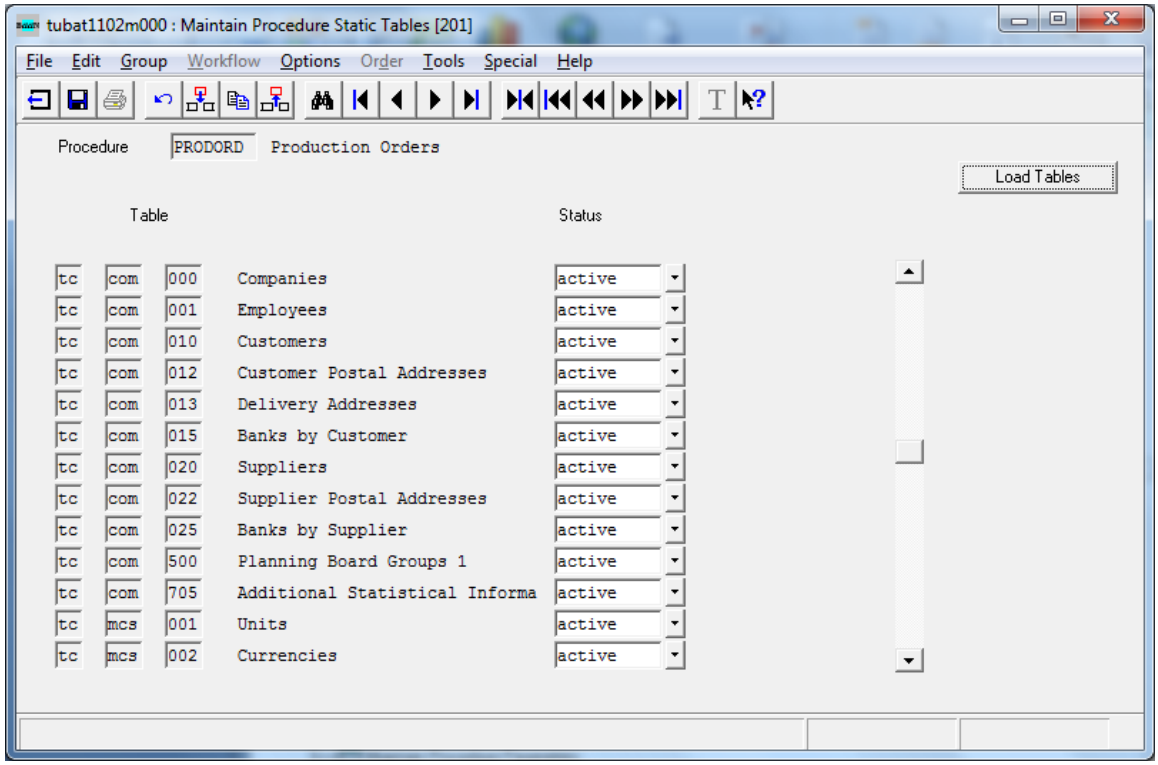


In this case, the value 100000 and 199999 will be bind to the variables :1 and :2 in the main table selection of the tdsls040.

Procedure Static Tables

The static tables are all the tables that are referred by the procedure step tables. The static tables are only 'refreshed' on the archiving company, never deleted from the actual company.

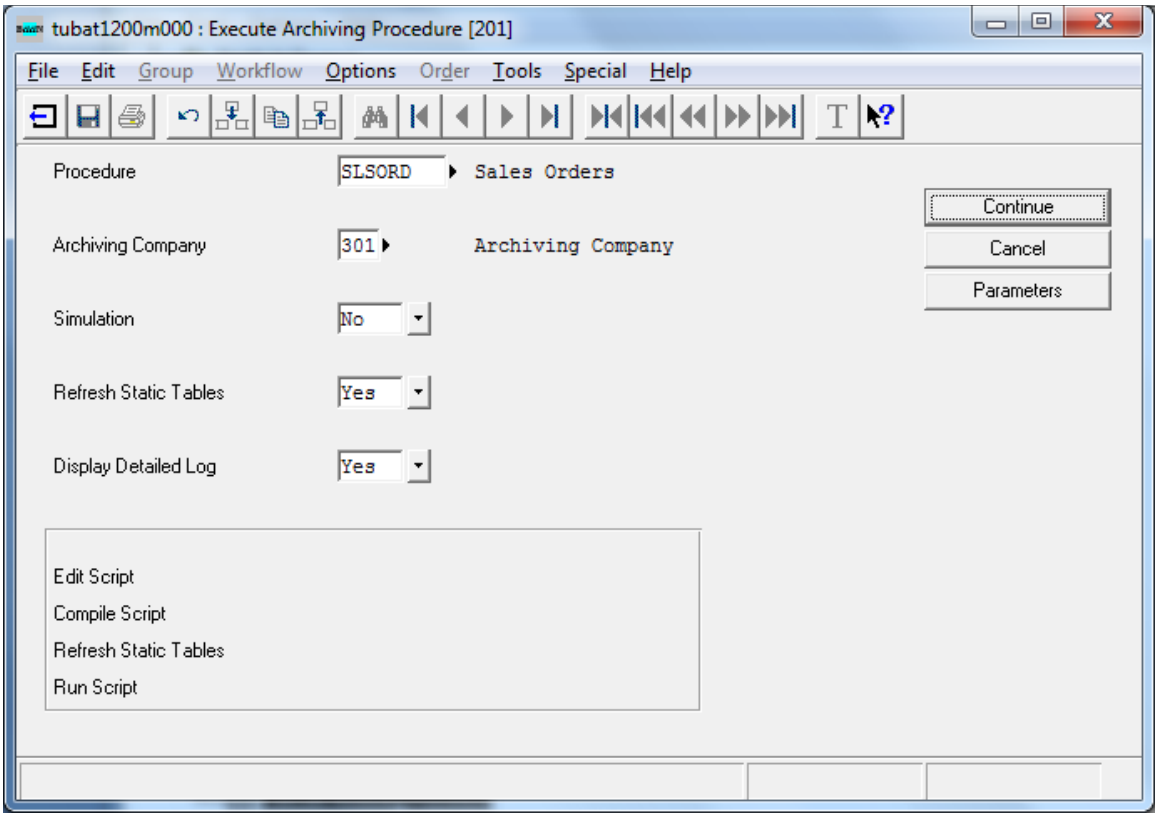
It is possible to load automatically the static tables by pressing the Load Tables button once all the procedures steps are defined.



Static tables can have status as active or expired. Expired tables never be refreshed on the archive company.

Execute the procedure

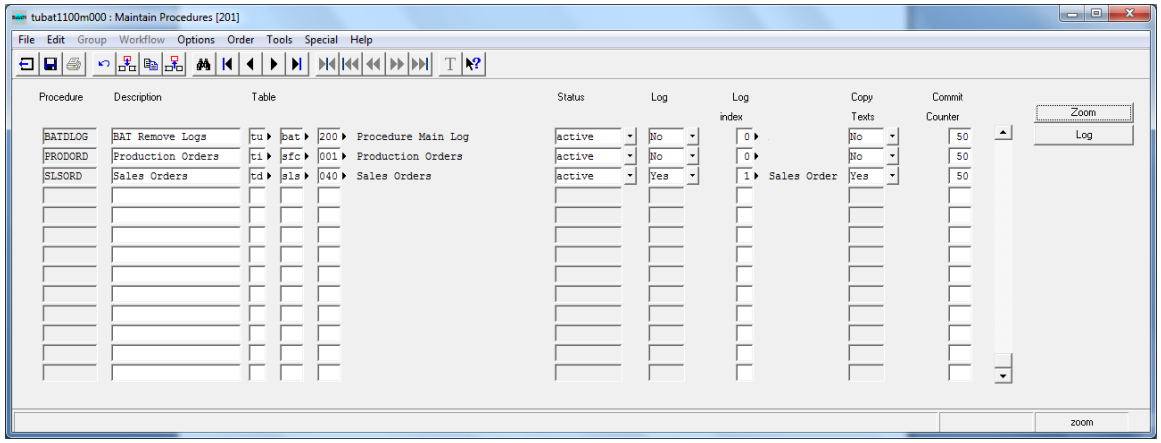
To execute the procedure, the session "Execute Archiving Procedure" must be run.
Choose the archiving procedure, the archiving company, if is a simulation (means that the resulting program is only compiled but not run), if the static tables linked to the procedure have to be refreshed on the archiving company and if the detailed log of transaction have to be displayed.
The phases are shown, and if a syntax error happens, the output is clearly displayed to the screen.



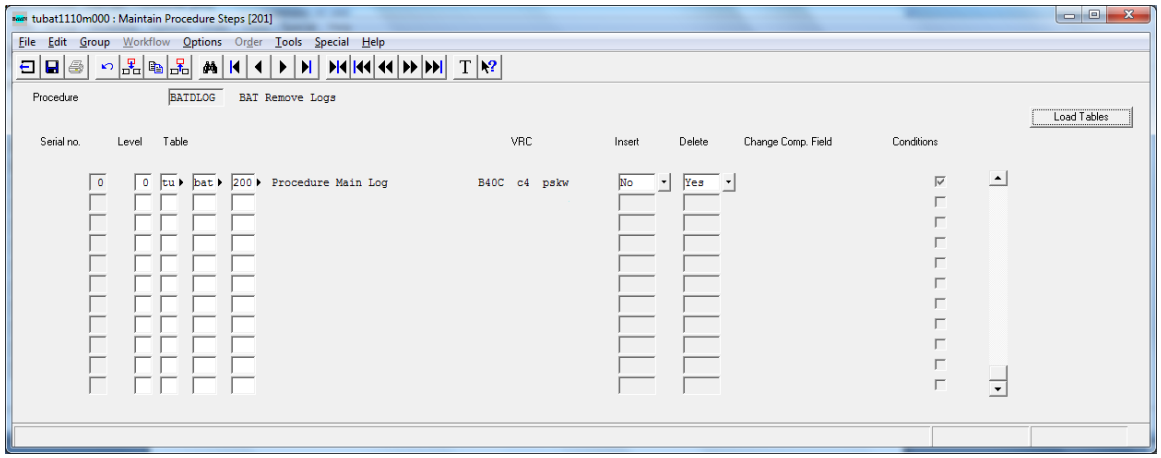
Example

This is a little example how the software works.
The BAT procedure fills two tables for tracking the log of executions. There is nothing in the software that provides to 'clean' those tables. But, the software was written also to solve this kind of problem, it is possible to archive data but even delete them only, so why don't use it to clean those tables? It could sound strange, but…

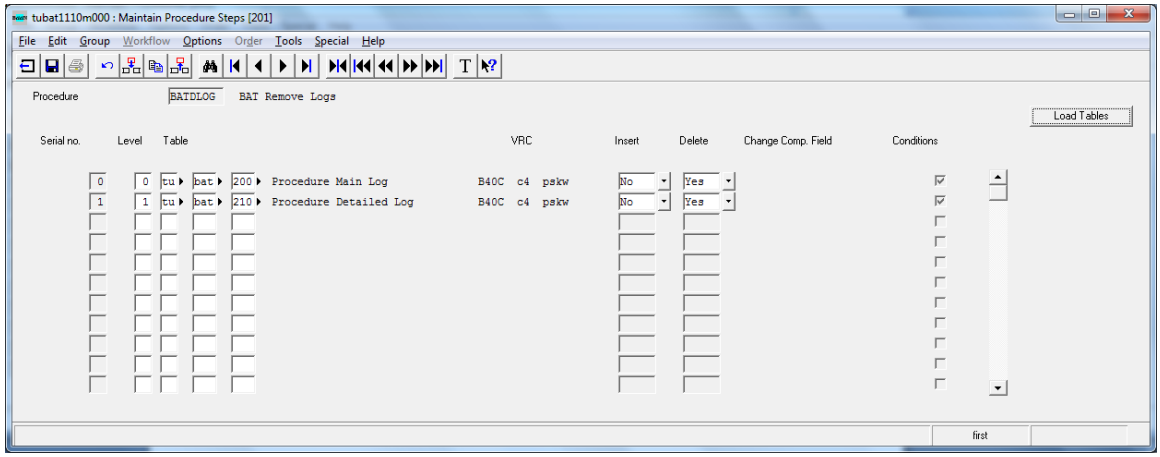
First, define the procedure BATDLOG in this way:



Then zoom to the procedure steps:

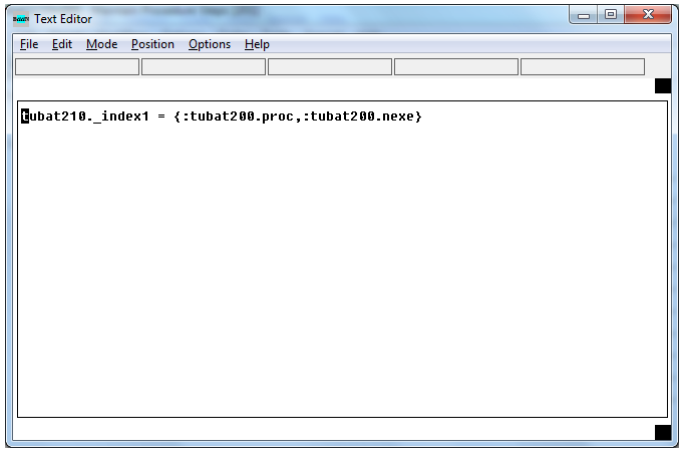


Note that the sequence 0 (with level 0) is automatically added to the steps.
Then press the "Load Tables" button. The result is:

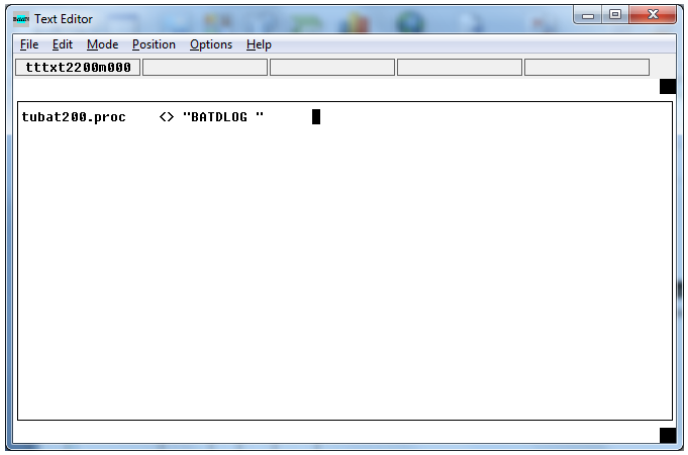


The sequence 1 and level 1 is added, because the tubat210 refers to the tubat200 with tubat210.cmba field.

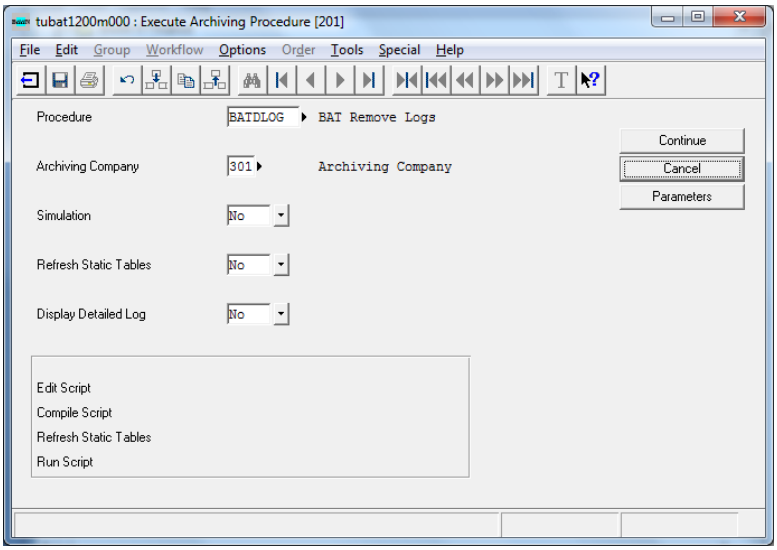
The condition is automatically added:



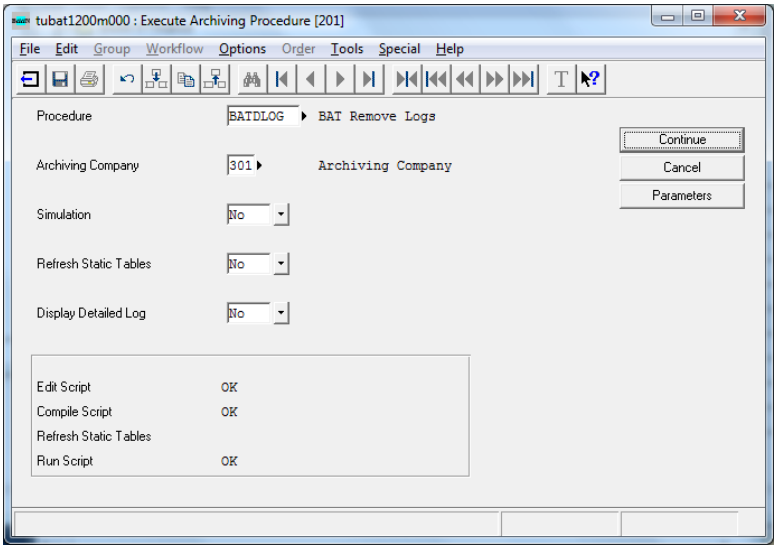
The only thing to do manually is to add the selection condition for the main table. In this case, for brevity, no procedure parameter is chosen, but a simple condition:



So, the procedure is ready to be executed:



And, after pressing the "Continue" button:



The procedure is run without problems, deleting all the logs that are not equal to the procedure "BATDLOG". And all the procedure was written in a couple of minutes.