Extract Metadata

The Extract Metadata menu item can be found in the IBExpert Tools menu, or started using the respective icon in the Tools toolbar. This item is unfortunately not available in the free IBExpert Personal Edition.

The Extract Metadata module can be used to generate a partial or full database metadata script, including table data, privileges and object descriptions if wished. It allows the user to extract metadata to file or clipboard. It is even possible to extract blob data and array fields' data (as blob data into a LOB file).

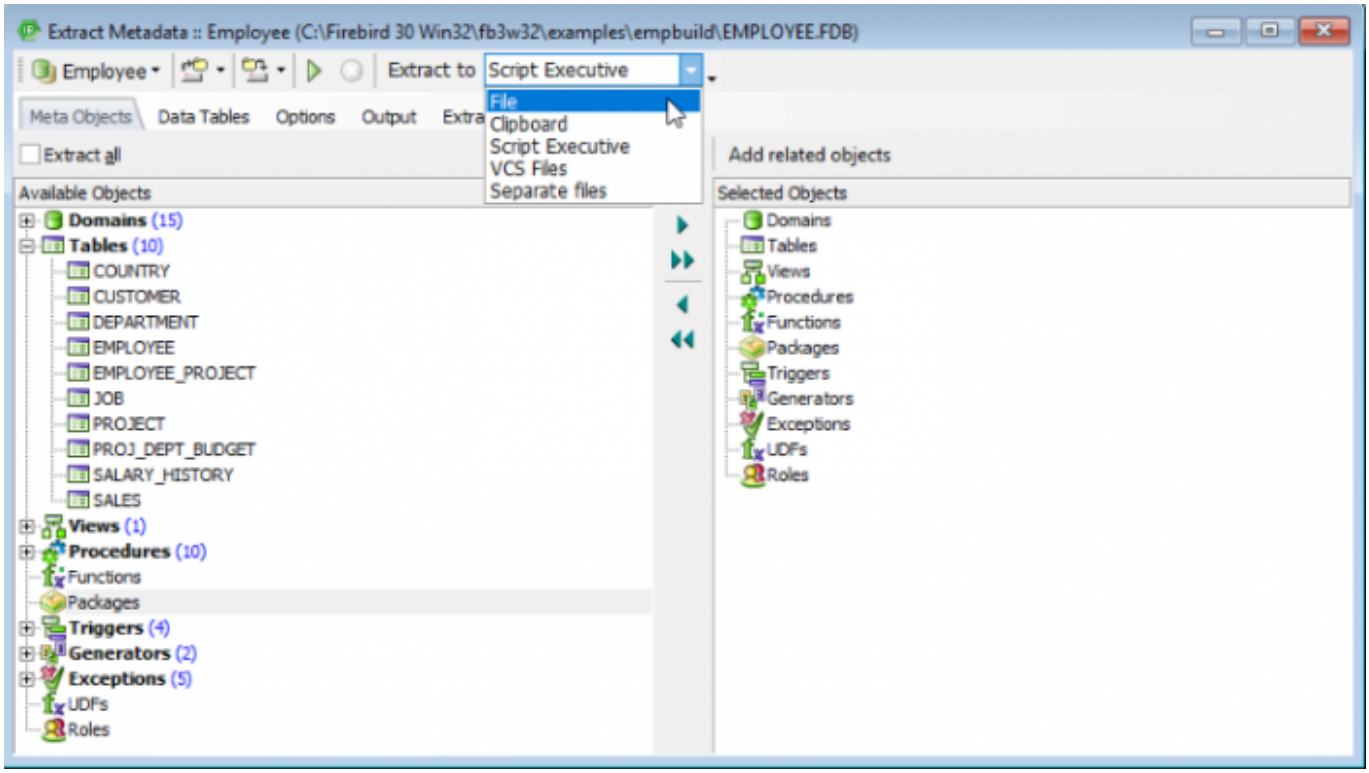


Table data can be extracted into separate files (TABLE_1.sql, TABLE_2.sql, TABLE_3.sql etc.) - the maximum file size can be specified on the *Options* page; once this size is reached, a new file is automatically generated by IBExpert, a particularly useful option when working with extremely large scripts, as problems are often encountered when executing scripts larger than 2 GB.

Support is also offered for secondary database file information; the corresponding ALTER DATABASE statements are included into the result script as comments.

Currently there is no way to select the collations which will be extracted into a script, all user-defined collations will be processed if they exist.

To begin preparing your metadata extraction, first select a database from the toolbar's drop-down list of all registered databases. The toolbar's *Extract* to options include:

- File
- Clipboard
- Script Executive (default)
- VCS Files
- Separate Files

The *Separate* Files mode extracts metadata (and data if specified) into a set of files: two files with metadata (_ibe$start_.sql and _ibe$finish_.sql), files containing table data (one or more files for each database table) and a runme.sql file, that consists of a number of INPUT <file_name> statements in the correct order.

If either the *File*, *VCS Files* or *Separate Files* options are selected, it is of course necessary to specify a file path and name (*.sql or *Metadata Extract Configuration* *.mec).

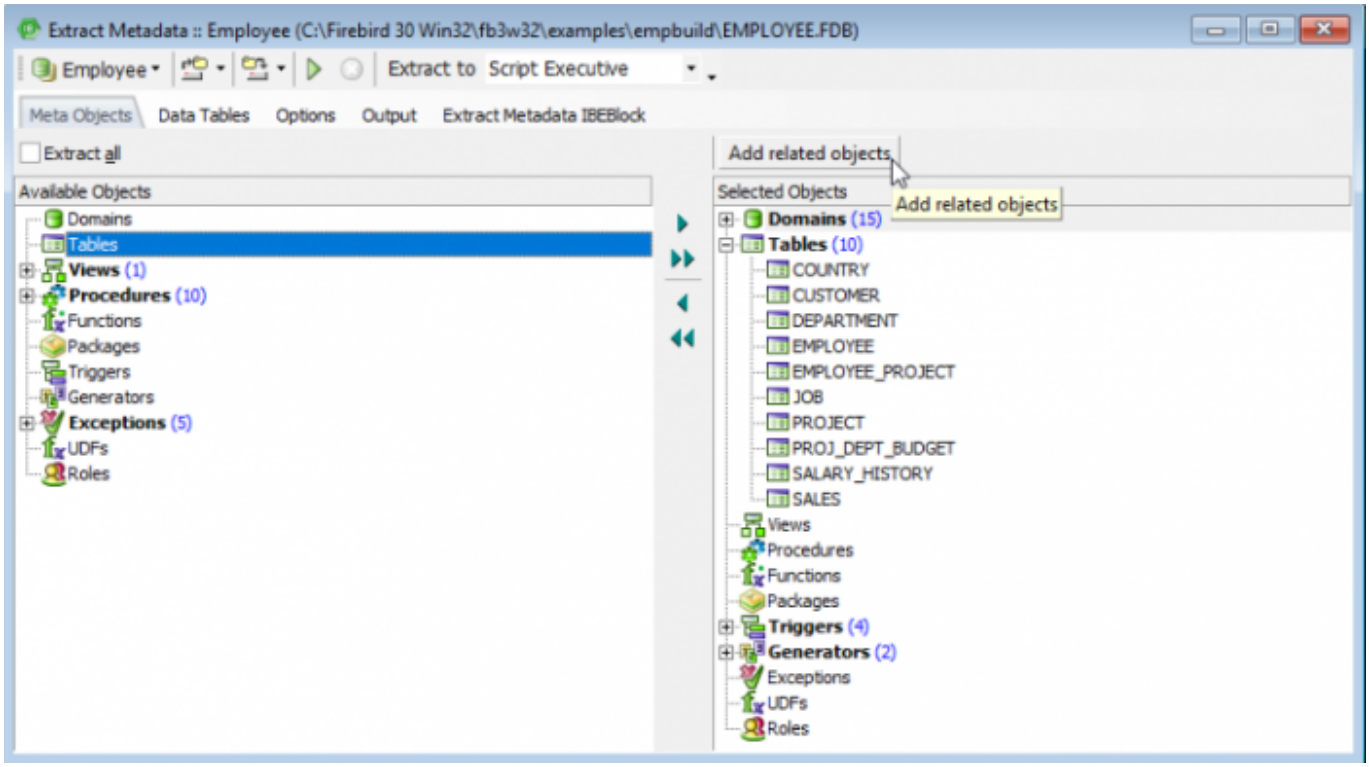
back to top of page

Extracting metadata in IBExpert

Meta Objects Page

The first dialog page, *Meta Objects*, displays the *Select Objects* tree. This *Objects Tree* window can be found in the following editors:

- Extract Metadata Editor on the first page, *Meta Objects*,
- Generate HTML Documentation Editor, also on the Objects page,
- Print Metadata dialog.



The *Select Objects Tree* feature offers the user the choice whether to extract all database objects (check option), or specify individual objects, (using the < or > buttons, drag 'n' dropping the object names or double-clicking on them), or object groups (using the « or » buttons, drag 'n' dropping the object headings or double-clicking on them).

Multiple objects can be selected using the [Ctrl] or [Shift] keys. There is even the option to *Add*

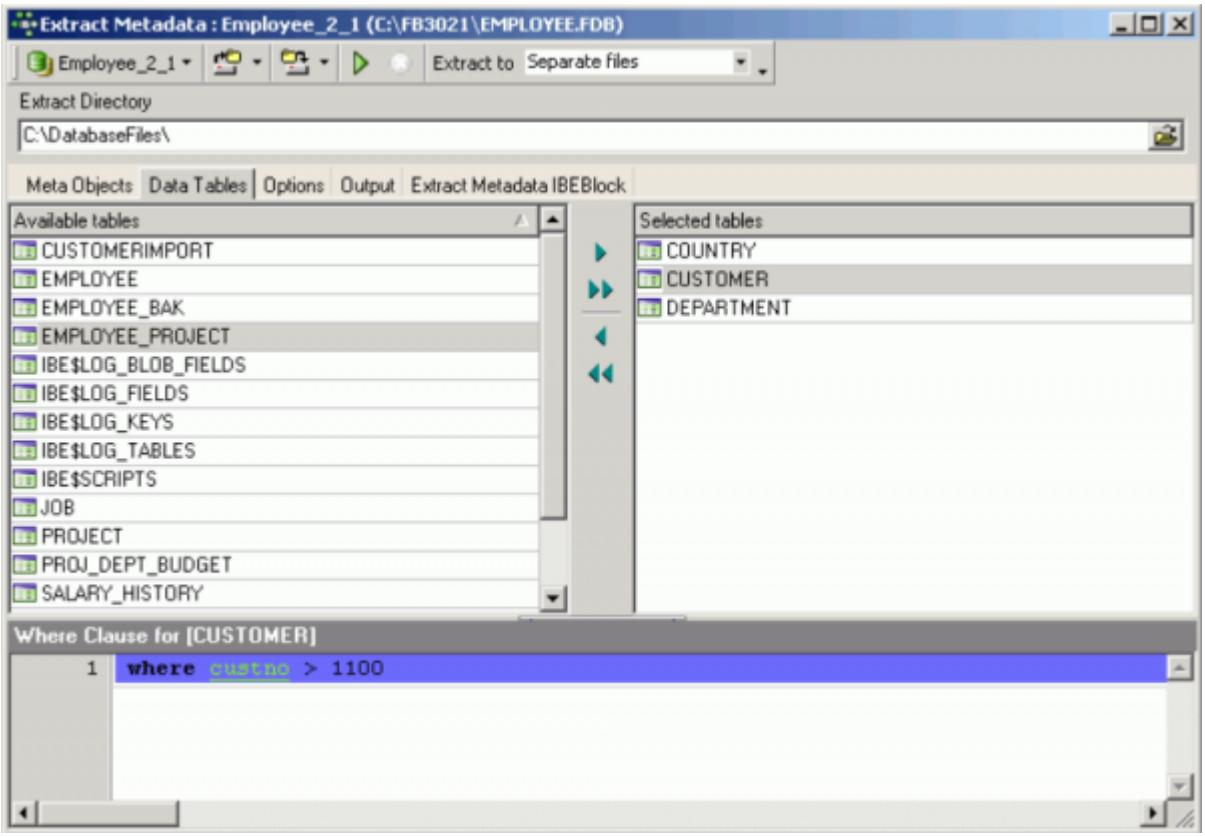
Related Objects by simply clicking the respective button above the *Selected Objects* window.

Objects can be dragged from the object dependencies trees (found on the *Dependencies* page in the object editors) and the field dependencies list (found in the *Field Dependencies* window at the bottom of the *Fields* page in the table/view editors) into the *Selected Objects* tree.

back to top of page

Data Tables Page

The *Data Tables* page can be used to specify whether data should also be extracted. This allows both user-defined and system tables to be selected - either all or individually:



again using the < , », > or » buttons, drag 'n' dropping or double-clicking.

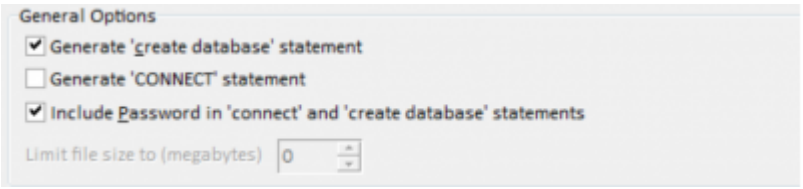
By selecting one of the tables in the *Selected Tables* list on the right-hand side, it is possible to add a WHERE clause, if wished.

back to top of page

Extract Metadata Options Page

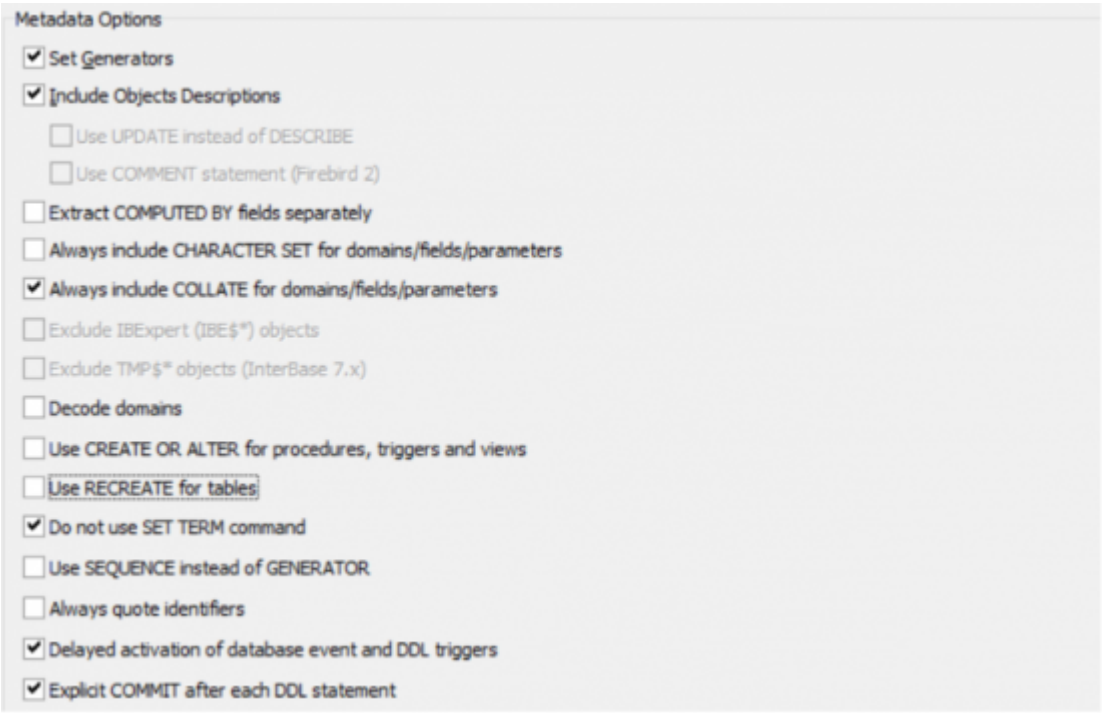
The *Extract Metadata Options* page offers a wide range of further check options:

General Options



- **Generate 'CREATE DATABASE' statement:** this determines whether a CREATE DATABASE statement should be included at the beginning of the generated script. If this option is unchecked, the CONNECT statement will be included instead.
- **Generate 'CONNECT' statement:** specifies the CONNECT statement.
- **Include password into 'CONNECT' and 'CREATE DATABASE' statements:** this determines whether the password should be included into the CREATE DATABASE or the CONNECT statement in the resulting SQL script.
- **Limit File Size:** This defines the maximum file size of the resulting script(s). When this option is specified and the maximum file size is reached, IBExpert automatically creates the next file with suffixes 0001, 0002 etc. Don't forget to specify the *Separate Files* option from the Extract to drop-down list at the top of the *Extract Metadata* window, when working with this option.

Metadata Options



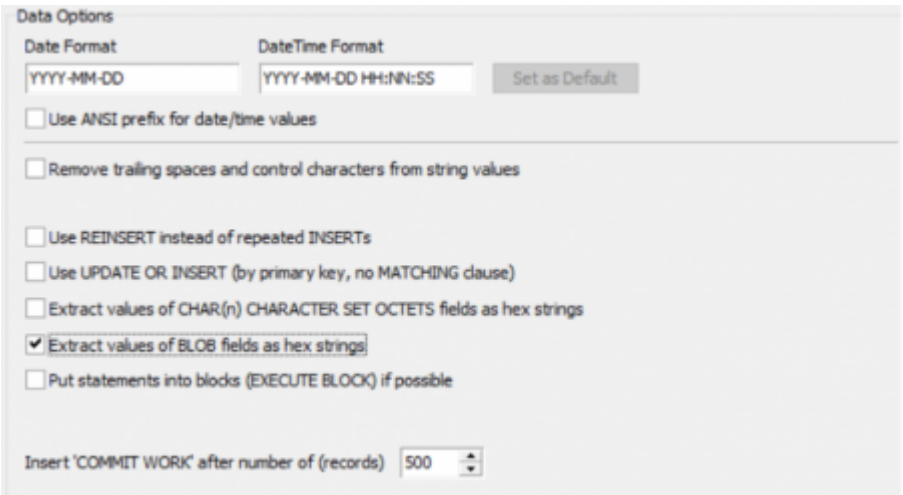
- **Set Generators:** if this option is checked, the SET GENERATOR statement for each generator will be included into the resulting script.
- **Include object descriptions:** this determines whether database objects descriptions should be included into the generated script. See How does IBExpert extract objects descriptions? for more details.
    - **Use UPDATE instead of DESCRIBE:** if this option is enabled, IBExpert will generate an UPDATE RDB$xxx SET RDB$DESCRPTION ... statement instead of DESCRIBE while

extracting metadata. You first need to check the option, Use *UPDATE* instead of *DESCRIBE* when extracting object descriptions, found in the IBExpert Database menu item, Register Database or Database Registration Info under Additional / Extract Metadata. By default it corresponds to the value specified in the Database Registration Info.
    - **Use COMMENT statement (Firebird 2):** this forces object descriptions to be extracted as a set of COMMENT statements. Refer to the Firebird 2.0.4 Release Notes chapter, COMMENT statement implemented for further information.
- **Extract COMPUTED BY fields separately:** this option can be used to specify whether computed fields should be extracted separately (useful if there are bugs in the database; realistically however this option is seldom used). When a computed field is used in a view, the definition of that field will appear in the result script just before the view definition.
- **Always include the CHARACTER SET for domains/fields/parameters.**
- **Always include COLLATE for domains/fields/parameters.**
- **Exclude IBExpert (IBE$*) objects:** check option.
- **Exclude TMP$* objects (InterBase 7.x):** check option.
- **Decode domains.** if this is activated, the domain types will be inserted as comments just after domain names. For example:

```
CREATE TABLE Z (
     B BOOL /* INTEGER DEFAULT 0 CHECK (VALUE IN(0,1)) */
);
```

- **Use CREATE OR ALTER for procedures and triggers:** compliant to Firebird 2.x.
- **Use RECREATE for tables:** If this option is ON IBExpert will use the RECREATE TABLE statement instead of CREATE TABLE.
- **Do not use SET TERM command:** SET TERM is not necessary for scripts executed by IBExpert/IBEScript but may be necessary when working with other tools.
- **Use SEQUENCE instead of GENERATOR:** compliant to Firebird 2.x.
- **Always quote identifiers:**
- **Delayed activation of database events and DDL triggers:** If this option is ON, DB event and DDL triggers will be first created as inactive, and activated at the end of the result script.
- **Explicit COMMIT after each DDL statement:** implemented in IBExpert version 2020.09.13. This option adds the COMMIT statement after each DDL statement in the result script.

Data Options



- **Date Format:** this can be used to specify the date format and datetime format, with options to

use an ANSI prefix for date/time values and to set the specified format as default.
- **Remove trailing spaces and control characters from string values**
- (*pre-IBExpert 2017.04.24*): **Extract Blobs:** IBExpert cannot "read" blobs; it therefore uses a detour to make a reference to a separate database file containing such blobs. Only IBExpert has been able to do this so far. Other products only extract the definition of the blobs, and not the contents themselves. This option is only offered when you export data/metadata into a file. It is invisible when the *Script Executive* and *Clipboard* options have been selected.
- **Use REINSERT instead of repeated INSERTs:** uses the IBExpert REINSERT command, to insert multiple data records.
- **Use UPDATE OR INSERT (by primary key, no MATCHING clause)**
- **Extract values of CHAR(n) CHARACTER SET OCTETS fields as hex strings:**
- **Put BLOB values into script as hex strings** If this option is enabled the values of blob fields will be written directly into the result script in hexadecimal notation, i.e. UPDATE MYTABLE SET BLOB_FIELD = x'4E657276656E';

Hexadecimal notation for binary values is available in Firebird 2.5 and Firebird 3. The length of the binary value which may be encoded in hexadecimal notation is limited to 32765 bytes in Firebird 2.5 and to 65535 bytes in Firebird 3. If the blob value exceeds the limit it will be written in the lob-file as is.

Please note: if you extract data into the Script Executive or clipboard you may get a warning about skipped blobs if their length exceeds the limitation of the hexadecimal notation.

- **Put DML statements into Firebird blocks (EXECUTE BLOCK) if possible:** implemented in IBExpert version 2021.02.09. The Use *REINSERT* option is ignored when the Put statements in blocks option is selected.
- **Insert 'COMMIT WORK' after number of (records):** this option defines the number of records before inserting the COMMIT statement into the script. The default value is 500, i.e. 500 INSERT commands are performed and then committed.

Grants

- **Extract privileges:** for all or only for selected objects.
- **Extract DDL privileges (Firebird 3 and higher):**

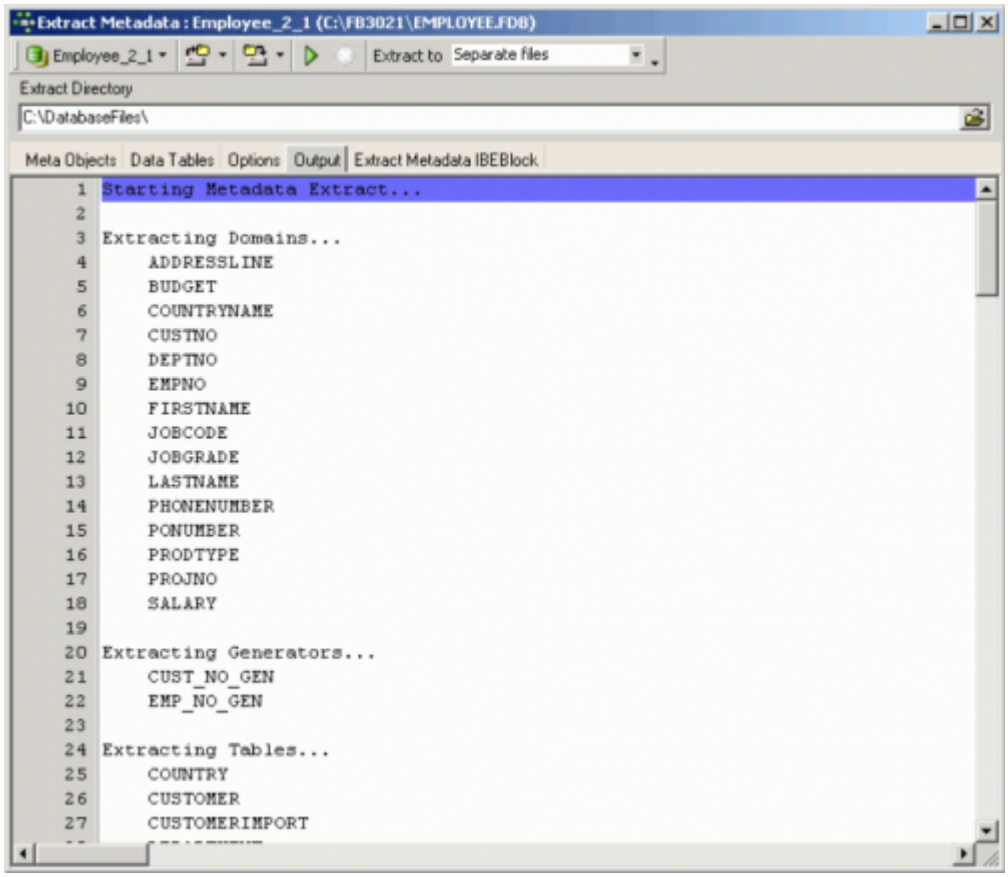
Finally, if wished, use the toolbar icon *Save Configuration to File* or the key combination [Ctrl + S] to save this configuration as a template for future use. The next time round, the template can be quickly and easily loaded using the *Load Configuration* icon (or [Ctrl + L]); the template specifications amended if necessary, and the extract started.

Once all objects and data have been selected, and all options specified, the extract can be started using the green > button or [F9].

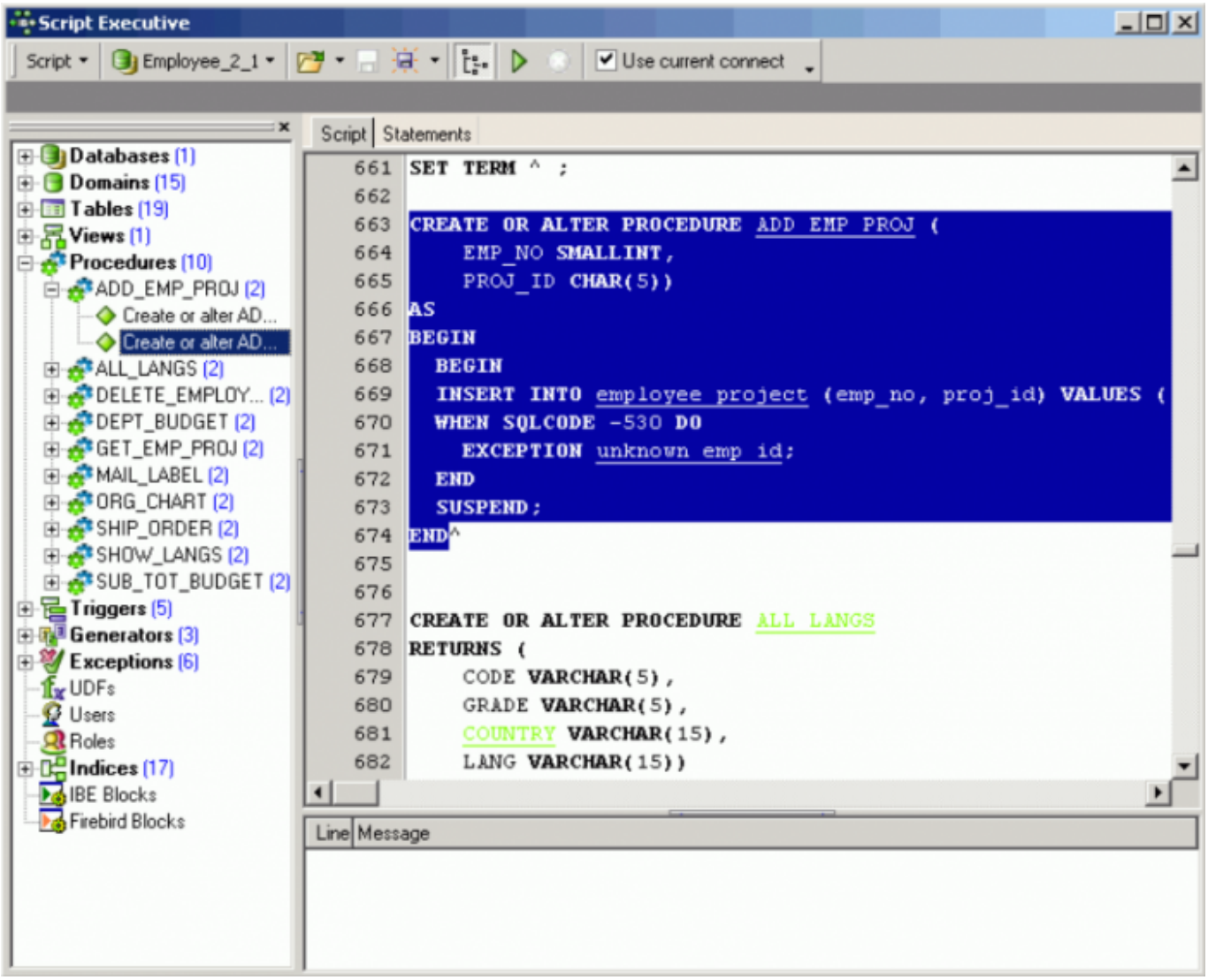
back to top of page

Output Page

The *Output* page displays the IBExpert log during the extraction. Following completion, if a file was specified, IBExpert asks whether the file should be loaded into the script editor.



If the Script Executive has been specified as the output option, the Script Executive is automatically loaded. The object tree on the left-hand side can be opened to display the individual statements relating to an object. By clicking on any of these statements, IBExpert springs to that part of SQL code, which is displayed on the right:

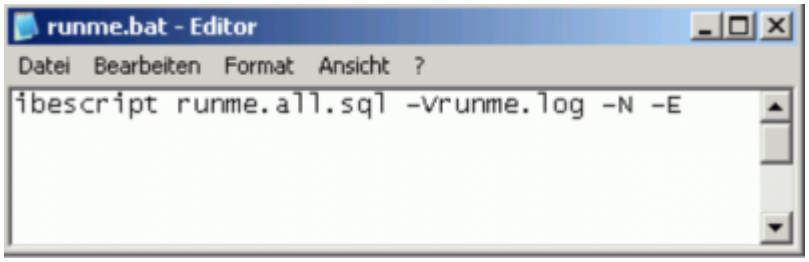


The statements display what IBExpert is doing and in which order. The script displays the creation of all objects, and then the subsequent insertion of the content data, using the ALTER command.

Extract Metadata is a great tool, and can be useful in a variety of situations. For example, it can be used to perform an incremental backup, should it be necessary for example, to back up just one single table every evening.

Any number of configurations may be saved in various formats:

- **Metadata extract configuration (*.mec):** this allows you to quickly and simply load a specified configuration in the *Extract Metadata* dialog.
- **IBEBlock (*.ibeblock):** this enables you to save the current settings as an EXECUTE STATEMENT statement. IBExpert creates a valid IBEBlock with the ibec_ExtractMetadata function, which may be used later in scripts.
- **All files (*.*).**



back to top of page

Extract Metadata IBEBlock

Simply specify the directory and file name you wish to extract to, and then customize the Extract Metadata block on the *IBEBlock* page as required and save to file. This function offers a quick and simple solution for a number of otherwise cumbersome tasks, such as generating foreign language versions of your database, subsequent alteration of the character set, alternative backup and restore or incremental backups. Please refer to ibec_ExtractMetadata for further information.

How does IBExpert extract objects descriptions?

IBExpert uses a special extension of script language that enables it to extract objects' descriptions into script and then execute one using the Script Executive.

back to top of page

How does IBExpert extract blobs?

IBExpert uses an original mechanism to extract values of blob fields into a script. This allows you to store an entire database (metadata and data) in script files and execute these scripts with IBExpert. The following small example illustrates our method to extract blob values.

For example, a database has a table named COMMENTS:

```
CREATE TABLE COMMENTS (   COMMENT_ID INTEGER NOT NULL PRIMARY KEY,
COMMENT_TEXT BLOB SUBTYPE TEXT);
```

This table has three records:

COMMENT_ID	COMMENT_TEXT
1	First comment
2	NULL
3	Another comment

If the *Extract BLOBs* option is unchecked you will get the following script:

```
CREATE TABLE COMMENTS (   COMMENT_ID INTEGER NOT NULL PRIMARY KEY,
COMMENT_TEXT BLOB SUBTYPE TEXT);

INSERT INTO COMMENTS (COMMENT_ID) VALUES (1);
INSERT INTO COMMENTS (COMMENT_ID) VALUES (2);
INSERT INTO COMMENTS (COMMENT_ID) VALUES (3);
```

... and, of course, you will lose your comments if you restore your database from this script.

But if the *Extract BLOBs* option is checked, IBExpert will generate a somewhat different script:

```
SET BLOBFILE 'C:\MY_SCRIPTS\RESULT.LOB';

CREATE TABLE COMMENTS (
   COMMENT_ID INTEGER NOT NULL PRIMARY KEY,
   COMMENT_TEXT BLOB SUBTYPE TEXT);

INSERT INTO COMMENTS (COMMENT_ID, COMMENT_TEXT) VALUES (1,
h0000000_0000000D);
INSERT INTO COMMENTS (COMMENT_ID, COMMENT_TEXT) VALUES (2, NULL);
INSERT INTO COMMENTS (COMMENT_ID, COMMENT_TEXT) VALUES (3,
h000000D_0000000F);
```

IBExpert also generates a special file with the extension LOB, where blob values are stored. In the current example result.lob will be 28 bytes long and its contents will be First commentAnother comment.

SET BLOBFILE is a special extension of script language that allows the IBExpert Script Executive to execute scripts containing references to blob field values.

back to top of page

Obtain current generator values

There are two methods to obtain the current generator values in a database. The first is using the IBExpert menu item Tools / Extract Metadata, where there is an option to set generators on the Options page.

In Firebird this can also be done using a stored procedure:

```
CREATE PROCEDURE GET_GENERATORS
RETURNS (
    GENERATOR_NAME CHAR(31),
    CURR_VAL BIGINT)
AS
```

```
declare variable sql varchar(100);
BEGIN
  FOR
    select r.rdb$generator_name generator_name, cast(0 as bigint) curr_val
from rdb$generators r
    where r.rdb$generator_name not containing '$'
    INTO :GENERATOR_NAME,
         :CURR_VAL
  DO
  BEGIN
    sql='Select gen_id('||GENERATOR_NAME||',0) from rdb$database';
    execute statement :sql into :curr_val;
    SUSPEND;
  END
END
```

back to top of page

Database repair using Extract Metadata

The Firebird core package has no dump tool. So it's important to analyze your metadata scripts to trace what started to go wrong, where and when.

If your backups are failing regularly on the same table(s) due to irreparable data damage, and you've not been able to solve the problem using GFIX, this is an alternative way to save at least all remaining healthy data and the database itself.

First attempt to restrict the problem to as few data sets as possible, using the SELECT command on the table ID field.

1. Then use the IBExpert Tools menu item, Extract Metadata. Connect to your database and select all tables for metadata and data.
2. Extract to - select *separate files* from the drop-down list.
3. Extract all objects and data from all tables.
4. If any error occurs on specific data, add a WHERE condition for the table concerned. For example, click on the table name in the right-hand column of *Selected Objects* and add your WHERE clause to exclude the range of damaged data, e.g. WHERE ID>1000 AND ID<1100.
5. Generate your script (green arrow icon or [F9]).
6. Delete the original database file.
7. If required, add the missing data as far as possible from an older extract file or backup copy of the database.
8. Execute runme.all.bat (don't forget to add the path to IBEScript.exe. This starts IBExpert's IBEScript, runme.all.sql, which loads the files from IBE$Start, then the data files and finally IBE$Finish.

This will create a new database with all objects and data, even including blob data.

IBE$Start runs the operations such as creating the database and metadata. Tables are generated, without any primary keys, foreign keys, constraints, triggers, etc. This is followed by a series of insert commands, using the IBEBlock function, REINSERT. IBE$Finish then inserts all primary keys, foreign

keys etc.

You can, of course carry all this out at script level, using ibec_ExtractMetadata.

This method can also be used if you wish to make an alteration to an existing database, for example, update from SQL dialect 1 to 3, or specify a character set if no default character set was specified at the time of database creation. For example, to alter the default character set from NONE to ISO8859_1, simply open IBE$Start, search CHARACTER SET NONE and replace with CHARACTER SET ISO8859_1, and then run the runme.all.sql script, as mentioned above.

From:
http://ibexpert.com/docu/ - **IBExpert**

Permanent link:
http://ibexpert.com/docu/doku.php?id=02-ibexpert:02-08-ibexpert-tools-menu:extract-metadata

Last update: **2023/10/06 18:43**