

# Shadowbase Data Definition Language Utility



Enscribe is an HPE NonStop non-relational file system that powers some of the largest enterprise application databases. The Enscribe Data Definition Language (DDL) is a standard tool supplied with HPE NonStop systems for defining data file and record metadata relationships in Enscribe file systems.

Many Enscribe users need to replicate their Enscribe data into external systems and environments, for example to feed a data warehouse, which is usually accomplished by converting the Enscribe information into standard SQL data structures and schemas. The HPE Shadowbase Data Definition Language Utility (SBDDLUTL) simplifies this process by providing a powerful interface to convert and map the Enscribe DDL data structures (records, fields) into their SQL equivalents (tables, columns). SBDDLUTL supports a variety of target SQL databases, including NonStop SQL/MP, NonStop SQL/MX, Oracle, Microsoft SQL Server, IBM Db2®, Oracle MySQL, SAP HANA, and SAP Sybase. The SBDDLUTL feature is available as part of the HPE Shadowbase Data Management Utilities product bundle.



In its simplest form, SBDDLUTL is a command line interface (CLI) database schema conversion tool, which converts an Enscribe record definition (which is naturally hierarchical), into a SQL schema definition (which is naturally a 'flattened' structure). SBDDLUTL performs various data normalization tasks during this conversion. The output of the conversion is a flattened version of the original Enscribe DDL definition, along with a SQL CREATE TABLE DDL definition for the selected target SQL database environment. Both of these definitions can then be utilized in an HPE Shadowbase data replication environment to replicate the source Enscribe data into the target SQL table structure.

## Key SBDDLUTL Features

- Creates a flattened Enscribe DDL definition from the original hierarchical Enscribe DDL definition
- Creates a SQL CREATE TABLE DDL statement from the flattened Enscribe DDL definition
- Maps Enscribe DDL fields/data types into equivalent SQL columns/data types
- Converts Enscribe DDL REDEFINES structures and ARRAYS/OCCURS clauses into uniquely-named and normalized SQL target columns
- Alerts if the target SQL environment does not support any of the source Enscribe DDL definitions, allowing for user input to either correct or disregard the offending data structures
- Supports a powerful command language to manage the metadata conversion environment (e.g., for target column naming conventions or for ARRAY processing rules)
- Can be run on any NonStop system, even if no NonStop SQL/MP or SQL/MX license exists

## Easily Convert Enscribe File Definitions to and/or from SQL Databases

Converting Enscribe data into its SQL equivalent is essential for building a data warehouse and/or business intelligence environment, offloading querying/reporting from the operational host environment (which often consumes costly computing power), and modernizing the application by converting the underlying Enscribe file system into a SQL relational database format.

Enscribe data files often have nested DDL definitions, i.e., definitions with sub-definitions that eventually deconstruct into individual fields. SQL does not support nested data schemas; therefore, the data must be *flattened*, causing the sub-definition's fields to become individual and uniquely renamed columns. SBDDLUTL automatically renames fields when flattening to ensure that there are no target column name collisions and the Enscribe record is properly converted to an SQL table.

The user has full control over the flattened Enscribe DDL definition field names and target table column-naming conventions. If the user specifies settings and program options that cause the resulting field or column names to be invalid, SBDDLUTL will warn the user about the invalid or duplicate field/column names. The flattened DDL or converted SQL will still be output, allowing the user to see and correct the issue and rerun the conversion. If required, the user can edit the resulting SQL DDL scripts to add additional platform/database specific features, such as table storage commands or

partitioning details. Note that SBDDLUTL does not modify the source Enscribe file's DDL; it creates a copy of the record definitions found in the specified DDL dictionary and displays it in the terminal window, which optionally can be copied or directly sent to an output file. Since all transformations occur on copies of the record definition, the user can freely apply settings to convert the Enscribe DDL without concern for changes propagating back to the original source Enscribe file's DDL. When compared to manually completing this error-prone process, SBDDLUTL dramatically reduces the total time involved by instantly flattening the Enscribe DDL definition and creating the SQL table schema.

SBDDLUTL allows the user to view Enscribe DDL structures, and includes a variety of useful formatting commands to read, store, and manipulate Enscribe DDL file definitions. Read in existing DDL dictionaries, set options, perform conversions on the stored records, and output the results to various destinations including the terminal, an edit file in ASCII, or to a binary file such as a serialized Google Protocol Buffer. (Please reference the manual for more information.)

## Summary

Many HPE NonStop customers use Enscribe files in their application's database. SBDDLUTL can convert the Enscribe DDL file definitions into standard SQL table schema definitions and data types by using a rich set of formatting commands. Typically, this manual process takes many hours to complete depending on the size of the Enscribe DDL definition. Instead, SBDDLUTL instantly flattens the definition, eliminating this error-prone effort.

**Example:** Show record RECORD-1, flatten RECORD-1, and then generate the CREATE TABLE statement (for an Oracle database target).

<u>The original, unflattened Record-1:</u>	<u>Flattened DDL:</u>	<u>Table Created in Oracle 12c:</u>
<pre>SBDDLUTL 3&gt; SHOW RECORD-1 RECORD RECORD-1. FILE IS "RECORD1" Key-sequenced. 02 KEY-1      PIC X(3). 02 GROUP-1.  03 FIELD-1   PIC 9(2). 02 GROUP-2.  03 FIELD-1   PIC X(2).  03 FIELD-2   PIC 9. 02 GROUP-3.  03 GROUP-A.  04 FIELD-A   PIC 9(4) COMP. 02 GROUP-4   OCCURS 2 TIMES.  03 FIELD-A   PIC X.  Key is KEY-1.  End  DDL Printed.</pre>	<pre>SBDDLUTL 4&gt; CONVERT RECORD-1 RECORD RECORD-1. FILE IS "RECORD1" Key-sequenced. 02 KEY-1      PIC X(3). 02 FIELD-1-D1 PIC 9(2). 02 FIELD-1-D2 PIC X(2). 02 FIELD-2    PIC 9. 02 FIELD-A-D1 PIC 9(4) COMP. 02 FIELD-A-D2 PIC X. 02 FIELD-A-D3 PIC X.  Key is KEY-1.  End  DDL Printed.</pre>	<pre>SBDDLUTL 5&gt; CONVERT RECORD-1 TARGET ORACLE ----  CREATE TABLE "RECORD1" ( "KEY_1"          CHAR(3)          NOT NULL, "FIELD_1_D1"     NUMBER(2, 0)      , "FIELD_1_D2"     CHAR(2)          , "FIELD_2"        NUMBER(1, 0)      , "FIELD_A_D1"     NUMBER(5, 0)      , "FIELD_A_D2"     CHAR(1)          , "FIELD_A_D3"     CHAR(1)          , PRIMARY KEY("KEY_1") ); ----  Create table statements generated.</pre>

### Learn more:

[shadowbasesoftware.com](http://shadowbasesoftware.com)  
[hpecom](http://hpecom)

### Contact us:

Gravic, Inc.  
 17 General Warren Blvd  
 Malvern, PA 19355-1245 USA  
 Tel: +1.610.647.6250  
 Fax: +1.610.647.7958  
 Email Sales: [shadowbase@gravic.com](mailto:shadowbase@gravic.com)  
 Email Support: [sbsupport@gravic.com](mailto:sbsupport@gravic.com)

### Please follow:

