HPE Shadowbase DDL & Schema Transformation (SBDDLUTL & SBCREATP)

Enscribe is an HPE NonStop non-relational file system that powers some of the largest enterprise application databases in the world. 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. SQL/MP is the first release of the HPE NonStop SQL databases.

HPE NonStop users often need to replicate their data into external systems and environments, for example to feed a data warehouse, which is usually accomplished by converting the HPE NonStop Enscribe, SQL/MP, or SQL/MX data into off-platform SQL data structures and schemas.

HPE Shadowbase Data Definition Language Utility (SBDDLUTL) & the HPE Shadowbase SBCREATP Utility simplify this process by providing a powerful interface to convert and map the Enscribe DDL data structures (records, fields) into their SQL equivalents (tables, columns). Both utilities support 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.

SBDDLUTL is a command line interface (CLI) database schema conversion tool that 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 an 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. (Please note that SBCREATP is effectively the same utility as SBDDLUTL, except that it is used in HPE NonStop SQL/MP environments.)

SBDDLUTL and SBCREATP Utilities are available as part of the HPE Shadowbase Essentials bundle.

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.)

SBDDLUTL Example

Show record RECORD-1, flatten RECORD-1, and then generate the CREATE TABLE statement (for an Oracledatabase target).

The original, unflattened Record-1:

```
SBDDLUTL 3> 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.
```

Flattened DDL:

```
SBDDLUTL 4> 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.
```

Table Created in Oracle 12c:

```
SBDDLUTL 5> 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.
```

SBCREATP Example

Show example table OSKEYTAB's SQL/MP CREATE TABLE statement, show OSKEYTAB's table definition, generate a PostgreSQL CREATE TABLE statement, and generate an Oracle CREATE TABLE statement.

1. **Show table OSKEYTAB's SQL/MP CREATE TABLE statement:**

```
CREATE TABLE $djb.DJBDATA.OSKEYTAB   (
    C0IK                          INT NO DEFAULT NOT NULL
  , C1CH                          CHAR(12) DEFAULT "a" NOT NULL
  , C2CHO                         CHAR(13) DEFAULT "aa" NOT NULL
  , C3CH                          CHAR(12) UPSHIFT DEFAULT "B"
  , C4VC                          VARCHAR(24) DEFAULT "c" NOT NULL
  , C5VC                          VARCHAR(22) UPSHIFT DEFAULT "D"
  , C6SI                          SMALLINT DEFAULT 2 NOT NULL
  , C7NMIN                        NUMERIC( 1, 0) DEFAULT 3 NOT NULL
  , C8NMAX                        NUMERIC( 4, 3) DEFAULT 4.123
  , C9SI                          SMALLINT DEFAULT 0
  , C10SI                         SMALLINT UNSIGNED DEFAULT 5 NOT NULL
  , C11NMIN                       NUMERIC( 1, 0) UNSIGNED DEFAULT 6 NOT NULL
  , C12NMAX                       NUMERIC( 4, 3) UNSIGNED DEFAULT 7.123
  , C13I                          INT DEFAULT 8 NOT NULL
  , C14NMIN                       NUMERIC( 5, 0) DEFAULT 9 NOT NULL
  , C15NMAX                       NUMERIC( 9, 8) DEFAULT 2.12345678
  , C16INB                        INT DEFAULT 0
  , C17I                          INT UNSIGNED DEFAULT 3
  , C18NMIN                       NUMERIC( 5, 0) UNSIGNED DEFAULT 198 NOT NULL
  , C19NMAX                       NUMERIC( 9, 8) UNSIGNED DEFAULT 5.12345678
  , C20LI                         LARGEINT DEFAULT 6 NOT NULL
```

```
    , C21NMIN                       NUMERIC( 10, 0) DEFAULT 7 NOT NULL
    , C22NMAX                       NUMERIC( 18, 17) DEFAULT 8.12345678901234567
    , C23LILB                       LARGEINT DEFAULT 0 NOT NULL
    , C24F                          FLOAT(1) DEFAULT 9 NOT NULL
    , C25F                          FLOAT(22) DEFAULT 2 NOT NULL
    , C26R                          FLOAT(22) DEFAULT 3 NOT NULL
    , C27F                          FLOAT(23) DEFAULT 4 NOT NULL
    , C28F                          FLOAT(54) DEFAULT 5
    , C29DP                         FLOAT(54) DEFAULT 6 NOT NULL
    , C54VC                         VARCHAR(24) DEFAULT "prev end"
    PRIMARY KEY (C0IK)
  )
  NO AUDITCOMPRESS
  CATALOG $DSMSCM.SBCAT
  ORGANIZATION KEY SEQUENCED
  SECURE "GGGG"
  AUDIT
  ;
```

2. Show OSKEYTAB's table definition:

```
  >>INVOKE DJBDATA.OSKEYTAB;
  -- Definition of table \VIV1.$QA.DJBDATA.OSKEYTAB
  -- Definition current at 09:50:52 - 05/04/22

    (
      C0IK                          INT NO DEFAULT NOT NULL
    , C1CH                          CHAR(12) DEFAULT "a" NOT NULL
    , C2CHO                         CHAR(13) DEFAULT "aa" NOT NULL
    , C3CH                          CHAR(12) UPSHIFT DEFAULT "B"
    , C4VC                          VARCHAR(24) DEFAULT "c" NOT NULL
    , C5VC                          VARCHAR(22) UPSHIFT DEFAULT "D"
    , C6SI                          SMALLINT DEFAULT 2 NOT NULL
    , C7NMIN                        NUMERIC( 1, 0) DEFAULT 3 NOT NULL
    , C8NMAX                        NUMERIC( 4, 3) DEFAULT 4.123
    , C9SI                          SMALLINT DEFAULT 0
    , C10SI                         SMALLINT UNSIGNED DEFAULT 5 NOT NULL
    , C11NMIN                       NUMERIC( 1, 0) UNSIGNED DEFAULT 6 NOT NULL
    , C12NMAX                       NUMERIC( 4, 3) UNSIGNED DEFAULT 7.123
    , C13I                          INT DEFAULT 8 NOT NULL
    , C14NMIN                       NUMERIC( 5, 0) DEFAULT 9 NOT NULL
    , C15NMAX                       NUMERIC( 9, 8) DEFAULT 2.12345678
    , C16INB                        INT DEFAULT 0
    , C17I                          INT UNSIGNED DEFAULT 3
    , C18NMIN                       NUMERIC( 5, 0) UNSIGNED DEFAULT 198 NOT NULL
    , C19NMAX                       NUMERIC( 9, 8) UNSIGNED DEFAULT 5.12345678
    , C20LI                         LARGEINT DEFAULT 6 NOT NULL
    , C21NMIN                       NUMERIC( 10, 0) DEFAULT 7 NOT NULL
    , C22NMAX                       NUMERIC( 18, 17) DEFAULT 8.12345678901234567
    , C23LILB                       LARGEINT DEFAULT 0 NOT NULL
    , C24F                          FLOAT(1) DEFAULT 9 NOT NULL
    , C25F                          FLOAT(22) DEFAULT 2 NOT NULL
    , C26R                          FLOAT(22) DEFAULT 3 NOT NULL
    , C27F                          FLOAT(23) DEFAULT 4 NOT NULL
    , C28F                          FLOAT(54) DEFAULT 5
    , C29DP                         FLOAT(54) DEFAULT 6 NOT NULL
    , C54VC                         VARCHAR(24) DEFAULT "prev end"
    )
  >>
```

3. Generate a PostgreSQL CREATE TABLE statement:

```
$QA QA6700 23> SBCREATP DJBDATA.OSKEYTAB POSTGRESQL
SBCREATP - T1122 - V6700L06 - (15APR22), COPYRIGHT GRAVIC, INC.
1995-2022. PORTIONS COPYRIGHTED BY AND LICENSED FROM THIRD PARTIES. SEE
README FILE. USAGE SUBJECT TO THE TERMS OF A WRITTEN LICENSE AGREEMENT.
PATENTS: SEE WWW.GRAVIC.COM/GRAVICLABS/PATENTS/PRODUCTS.
/*
   POSTGRESQL create script generated May 04 2022 10:02:16
   Source  \VIV1.$QA.DJBDATA.OSKEYTAB
   Table   key-sequenced
   Catalog \VIV1.$QA.QACAT
*/
CREATE TABLE OSKEYTAB(
 C0IK int not null
,C1CH char(12) default 'a' not null
,C2CHO char(13) default 'aa' not null
,C3CH char(12) default 'B' null
,C4VC varchar(24) default 'c' not null
,C5VC varchar(22) default 'D' null
,C6SI smallint default 2 not null
,C7NMIN smallint default 3 not null
,C8NMAX numeric(4,3) default 4.123 null
,C9SI smallint default 0 null
,C10SI int default 5 not null
,C11NMIN int default 6 not null
,C12NMAX numeric(4,3) default 7.123 null
,C13I int default 8 not null
,C14NMIN int default 9 not null
,C15NMAX numeric(9,8) default 2.12345678 null
,C16INB int default 0 null
,C17I bigint default 3 null
,C18NMIN bigint default 198 not null
,C19NMAX numeric(9,8) default 5.12345678 null
,C20LI bigint default 6 not null
,C21NMIN bigint default 7 not null
,C22NMAX numeric(18,17) default 8.12345678901234567 null
,C23LILB bigint default 0 not null
,C24F real default 9 not null
,C25F real default 2 not null
,C26R real default 3 not null
,C27F double precision default 4 not null
,C28F double precision default 5 null
,C29DP double precision default 6 not null
,C54VC varchar(24) default 'prev end' null
, primary key (
 C0IK
));
```

4. Generate an Oracle CREATE TABLE statement:

```
$QA QA6700 24> SBCREATP DJBDATA.OSKEYTAB ORACLE
SBCREATP - T1122 - V6700L06 - (15APR22), COPYRIGHT GRAVIC, INC.
1995-2022. PORTIONS COPYRIGHTED BY AND LICENSED FROM THIRD PARTIES. SEE
README FILE. USAGE SUBJECT TO THE TERMS OF A WRITTEN LICENSE AGREEMENT.
PATENTS: SEE WWW.GRAVIC.COM/GRAVICLABS/PATENTS/PRODUCTS.
/*
   ORACLE create script generated May 04 2022 10:02:34
   Source  \VIV1.$QA.DJBDATA.OSKEYTAB
   Table   key-sequenced
   Catalog \VIV1.$QA.QACAT
*/
CREATE TABLE "OSKEYTAB"(
 "C0IK" number not null
,"C1CH" char(12) default 'a' not null
,"C2CHO" char(13) default 'aa' not null
,"C3CH" char(12) default 'B' null
,"C4VC" varchar2(24) default 'c' not null
,"C5VC" varchar2(22) default 'D' null
,"C6SI" number default 2 not null
```

```
       ,"C7NMIN" number default 3 not null
       ,"C8NMAX" number(4,3) default 4.123 null
       ,"C9SI" number default 0 null
       ,"C10SI" number default 5 not null
       ,"C11NMIN" number default 6 not null
       ,"C12NMAX" number(4,3) default 7.123 null
       ,"C13I" number default 8 not null
       ,"C14NMIN" number default 9 not null
       ,"C15NMAX" number(9,8) default 2.12345678 null
       ,"C16INB" number default 0 null
       ,"C17I" number default 3 null
       ,"C18NMIN" number default 198 not null
       ,"C19NMAX" number(9,8) default 5.12345678 null
       ,"C20LI" number default 6 not null
       ,"C21NMIN" number default 7 not null
       ,"C22NMAX" number(18,17) default 8.12345678901234567 null
       ,"C23LILB" number default 0 not null
       ,"C24F" binary_float default 9 not null
       ,"C25F" binary_float default 2 not null
       ,"C26R" binary_float default 3 not null
       ,"C27F" binary_double default 4 not null
       ,"C28F" binary_double default 5 null
       ,"C29DP" binary_double default 6 not null
       ,"C54VC" varchar2(24) default 'prev end' null
       , primary key (
        "C0IK"
       ))
```

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. SBCREATP performs a similar translation from a source SQL/MP schema into a user-selected target SQL variant.

Learn more:
shadowbasesoftware.com
hpe.com

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
Email Support: sbsupport@gravic.com

Please follow:

Copyright © 2018, 2022 by Gravic, Inc. Gravic, Shadowbase and Total Replication Solutions are registered trademarks of Gravic, Inc. All other brand and product names are the trademarks or registered trademarks of their respective owners. Specifications subject to change without notice.