

-- Table SEQUENCE_TABLE is an internal table required by DataNucleus.
-- NOTE: Some versions of SchemaTool do not automatically generate this table.

-- See <http://www.datanucleus.org/servlet/jira/browse/NUCRDBMS-416>

```
CREATE TABLE SEQUENCE_TABLE
(
    SEQUENCE_NAME VARCHAR2(255) NOT NULL,
    NEXT_VAL NUMBER NOT NULL
);
```

```
ALTER TABLE SEQUENCE_TABLE ADD CONSTRAINT PART_TABLE_PK PRIMARY KEY
(SEQUENCE_NAME);
```

-- Table NUCLEUS_TABLES is an internal table required by DataNucleus.
-- This table is required if datanucleus.autoStartMechanism=SchemaTable
-- NOTE: Some versions of SchemaTool do not automatically generate this table.

-- See <http://www.datanucleus.org/servlet/jira/browse/NUCRDBMS-416>

```
CREATE TABLE NUCLEUS_TABLES
(
    CLASS_NAME VARCHAR2(128) NOT NULL,
    TABLE_NAME VARCHAR2(128) NOT NULL,
    TYPE VARCHAR2(4) NOT NULL,
    OWNER VARCHAR2(2) NOT NULL,
    VERSION VARCHAR2(20) NOT NULL,
    INTERFACE_NAME VARCHAR2(255) NULL
);
```

```
ALTER TABLE NUCLEUS_TABLES ADD CONSTRAINT NUCLEUS_TABLES_PK PRIMARY KEY
(CLASS_NAME);
```

-- Table PART_COL_PRIVS for classes
[org.apache.hadoop.hive.metastore.model.MPartitionColumnPrivilege]

```
CREATE TABLE PART_COL_PRIVS
(
    PART_COLUMN_GRANT_ID NUMBER NOT NULL,
    "COLUMN_NAME" VARCHAR2(128) NULL,
    CREATE_TIME NUMBER(10) NOT NULL,
    GRANT_OPTION NUMBER(5) NOT NULL,
    GRANTOR VARCHAR2(128) NULL,
    GRANTOR_TYPE VARCHAR2(128) NULL,
    PART_ID NUMBER NULL,
    PRINCIPAL_NAME VARCHAR2(128) NULL,
    PRINCIPAL_TYPE VARCHAR2(128) NULL,
    PART_COL_PRIV VARCHAR2(128) NULL
);
```

```
ALTER TABLE PART_COL_PRIVS ADD CONSTRAINT PART_COL_PRIVS_PK PRIMARY KEY
(PART_COLUMN_GRANT_ID);
```

-- Table CDS.
CREATE TABLE CDS

```
(
    CD_ID NUMBER NOT NULL
);
```

```
ALTER TABLE CDS ADD CONSTRAINT CDS_PK PRIMARY KEY (CD_ID);
```

```

-- Table COLUMNS_V2 for join relationship
CREATE TABLE COLUMNS_V2
(
    CD_ID NUMBER NOT NULL,
    "COMMENT" VARCHAR2(256) NULL,
    "COLUMN_NAME" VARCHAR2(128) NOT NULL,
    TYPE_NAME VARCHAR2(4000) NOT NULL,
    INTEGER_IDX NUMBER(10) NOT NULL
);

ALTER TABLE COLUMNS_V2 ADD CONSTRAINT COLUMNS_V2_PK PRIMARY KEY
(CD_ID, "COLUMN_NAME");

-- Table PARTITION_KEY_VALS for join relationship
CREATE TABLE PARTITION_KEY_VALS
(
    PART_ID NUMBER NOT NULL,
    PART_KEY_VAL VARCHAR2(256) NULL,
    INTEGER_IDX NUMBER(10) NOT NULL
);

ALTER TABLE PARTITION_KEY_VALS ADD CONSTRAINT PARTITION_KEY_VALS_PK
PRIMARY KEY (PART_ID, INTEGER_IDX);

-- Table DBS for classes
[org.apache.hadoop.hive.metastore.model.MDatabase]
CREATE TABLE DBS
(
    DB_ID NUMBER NOT NULL,
    "DESC" VARCHAR2(4000) NULL,
    DB_LOCATION_URI VARCHAR2(4000) NOT NULL,
    "NAME" VARCHAR2(128) NULL,
    OWNER_NAME VARCHAR2(128) NULL,
    OWNER_TYPE VARCHAR2(10) NULL
);

ALTER TABLE DBS ADD CONSTRAINT DBS_PK PRIMARY KEY (DB_ID);

-- Table PARTITION_PARAMS for join relationship
CREATE TABLE PARTITION_PARAMS
(
    PART_ID NUMBER NOT NULL,
    PARAM_KEY VARCHAR2(256) NOT NULL,
    PARAM_VALUE VARCHAR2(4000) NULL
);

ALTER TABLE PARTITION_PARAMS ADD CONSTRAINT PARTITION_PARAMS_PK PRIMARY
KEY (PART_ID, PARAM_KEY);

-- Table SERDES for classes
[org.apache.hadoop.hive.metastore.model.MSerDeInfo]
CREATE TABLE SERDES
(
    SERDE_ID NUMBER NOT NULL,
    "NAME" VARCHAR2(128) NULL,
    SLIB VARCHAR2(4000) NULL
);

```

```

ALTER TABLE SERDES ADD CONSTRAINT SERDES_PK PRIMARY KEY (SERDE_ID);

-- Table TYPES for classes [org.apache.hadoop.hive.metastore.model.MType]
CREATE TABLE TYPES
(
    TYPES_ID NUMBER NOT NULL,
    TYPE_NAME VARCHAR2(128) NULL,
    TYPE1 VARCHAR2(767) NULL,
    TYPE2 VARCHAR2(767) NULL
);

ALTER TABLE TYPES ADD CONSTRAINT TYPES_PK PRIMARY KEY (TYPES_ID);

-- Table PARTITION_KEYS for join relationship
CREATE TABLE PARTITION_KEYS
(
    TBL_ID NUMBER NOT NULL,
    PKEY_COMMENT VARCHAR2(4000) NULL,
    PKEY_NAME VARCHAR2(128) NOT NULL,
    PKEY_TYPE VARCHAR2(767) NOT NULL,
    INTEGER_IDX NUMBER(10) NOT NULL
);

ALTER TABLE PARTITION_KEYS ADD CONSTRAINT PARTITION_KEY_PK PRIMARY KEY
(TBL_ID, PKEY_NAME);

-- Table ROLES for classes [org.apache.hadoop.hive.metastore.model.MRole]
CREATE TABLE ROLES
(
    ROLE_ID NUMBER NOT NULL,
    CREATE_TIME NUMBER (10) NOT NULL,
    OWNER_NAME VARCHAR2(128) NULL,
    ROLE_NAME VARCHAR2(128) NULL
);

ALTER TABLE ROLES ADD CONSTRAINT ROLES_PK PRIMARY KEY (ROLE_ID);

-- Table PARTITIONS for classes
[org.apache.hadoop.hive.metastore.model.MPartition]
CREATE TABLE PARTITIONS
(
    PART_ID NUMBER NOT NULL,
    CREATE_TIME NUMBER (10) NOT NULL,
    LAST_ACCESS_TIME NUMBER (10) NOT NULL,
    PART_NAME VARCHAR2(767) NULL,
    SD_ID NUMBER NULL,
    TBL_ID NUMBER NULL
);

ALTER TABLE PARTITIONS ADD CONSTRAINT PARTITIONS_PK PRIMARY KEY
(PART_ID);

-- Table INDEX_PARAMS for join relationship
CREATE TABLE INDEX_PARAMS
(
    INDEX_ID NUMBER NOT NULL,
    PARAM_KEY VARCHAR2(256) NOT NULL,
    PARAM_VALUE VARCHAR2(4000) NULL

```

```
);
```

```
ALTER TABLE INDEX_PARAMS ADD CONSTRAINT INDEX_PARAMS_PK PRIMARY KEY  
(INDEX_ID, PARAM_KEY);
```

```
-- Table TBL_COL_PRIVS for classes  
[org.apache.hadoop.hive.metastore.model.MTableColumnPrivilege]
```

```
CREATE TABLE TBL_COL_PRIVS  
(  
    TBL_COLUMN_GRANT_ID NUMBER NOT NULL,  
    "COLUMN_NAME" VARCHAR2(128) NULL,  
    CREATE_TIME NUMBER (10) NOT NULL,  
    GRANT_OPTION NUMBER (5) NOT NULL,  
    GRANTOR VARCHAR2(128) NULL,  
    GRANTOR_TYPE VARCHAR2(128) NULL,  
    PRINCIPAL_NAME VARCHAR2(128) NULL,  
    PRINCIPAL_TYPE VARCHAR2(128) NULL,  
    TBL_COL_PRIV VARCHAR2(128) NULL,  
    TBL_ID NUMBER NULL  
);
```

```
ALTER TABLE TBL_COL_PRIVS ADD CONSTRAINT TBL_COL_PRIVS_PK PRIMARY KEY  
(TBL_COLUMN_GRANT_ID);
```

```
-- Table IDXS for classes [org.apache.hadoop.hive.metastore.model.MIndex]  
CREATE TABLE IDXS
```

```
(  
    INDEX_ID NUMBER NOT NULL,  
    CREATE_TIME NUMBER (10) NOT NULL,  
    DEFERRED_REBUILD NUMBER(1) NOT NULL CHECK (DEFERRED_REBUILD IN  
(1,0)),  
    INDEX_HANDLER_CLASS VARCHAR2(4000) NULL,  
    INDEX_NAME VARCHAR2(128) NULL,  
    INDEX_TBL_ID NUMBER NULL,  
    LAST_ACCESS_TIME NUMBER (10) NOT NULL,  
    ORIG_TBL_ID NUMBER NULL,  
    SD_ID NUMBER NULL  
);
```

```
ALTER TABLE IDXS ADD CONSTRAINT IDXS_PK PRIMARY KEY (INDEX_ID);
```

```
-- Table BUCKETING_COLS for join relationship
```

```
CREATE TABLE BUCKETING_COLS  
(  
    SD_ID NUMBER NOT NULL,  
    BUCKET_COL_NAME VARCHAR2(256) NULL,  
    INTEGER_IDX NUMBER(10) NOT NULL  
);
```

```
ALTER TABLE BUCKETING_COLS ADD CONSTRAINT BUCKETING_COLS_PK PRIMARY KEY  
(SD_ID, INTEGER_IDX);
```

```
-- Table TYPE_FIELDS for join relationship
```

```
CREATE TABLE TYPE_FIELDS  
(  
    TYPE_NAME NUMBER NOT NULL,  
    "COMMENT" VARCHAR2(256) NULL,  
    FIELD_NAME VARCHAR2(128) NOT NULL,  
);
```

```

        FIELD_TYPE VARCHAR2(767) NOT NULL,
        INTEGER_IDX NUMBER(10) NOT NULL
    );

ALTER TABLE TYPE_FIELDS ADD CONSTRAINT TYPE_FIELDS_PK PRIMARY KEY
(TYPE_NAME, FIELD_NAME);

-- Table SD_PARAMS for join relationship
CREATE TABLE SD_PARAMS
(
    SD_ID NUMBER NOT NULL,
    PARAM_KEY VARCHAR2(256) NOT NULL,
    PARAM_VALUE VARCHAR2(4000) NULL
);

ALTER TABLE SD_PARAMS ADD CONSTRAINT SD_PARAMS_PK PRIMARY KEY
(SD_ID, PARAM_KEY);

-- Table GLOBAL_PRIVS for classes
[org.apache.hadoop.hive.metastore.model.MGlobalPrivilege]
CREATE TABLE GLOBAL_PRIVS
(
    USER_GRANT_ID NUMBER NOT NULL,
    CREATE_TIME NUMBER (10) NOT NULL,
    GRANT_OPTION NUMBER (5) NOT NULL,
    GRANTOR VARCHAR2(128) NULL,
    GRANTOR_TYPE VARCHAR2(128) NULL,
    PRINCIPAL_NAME VARCHAR2(128) NULL,
    PRINCIPAL_TYPE VARCHAR2(128) NULL,
    USER_PRIV VARCHAR2(128) NULL
);

ALTER TABLE GLOBAL_PRIVS ADD CONSTRAINT GLOBAL_PRIVS_PK PRIMARY KEY
(USER_GRANT_ID);

-- Table SDS for classes
[org.apache.hadoop.hive.metastore.model.MStorageDescriptor]
CREATE TABLE SDS
(
    SD_ID NUMBER NOT NULL,
    CD_ID NUMBER NULL,
    INPUT_FORMAT VARCHAR2(4000) NULL,
    IS_COMPRESSED NUMBER(1) NOT NULL CHECK (IS_COMPRESSED IN (1,0)),
    LOCATION VARCHAR2(4000) NULL,
    NUM_BUCKETS NUMBER (10) NOT NULL,
    OUTPUT_FORMAT VARCHAR2(4000) NULL,
    SERDE_ID NUMBER NULL,
    IS_STOREDASSUBDIRECTORIES NUMBER(1) NOT NULL CHECK
(IS_STOREDASSUBDIRECTORIES IN (1,0))
);

ALTER TABLE SDS ADD CONSTRAINT SDS_PK PRIMARY KEY (SD_ID);

-- Table TABLE_PARAMS for join relationship
CREATE TABLE TABLE_PARAMS
(
    TBL_ID NUMBER NOT NULL,
    PARAM_KEY VARCHAR2(256) NOT NULL,

```

```

    PARAM_VALUE VARCHAR2(4000) NULL
);

ALTER TABLE TABLE_PARAMS ADD CONSTRAINT TABLE_PARAMS_PK PRIMARY KEY
(TBL_ID, PARAM_KEY);

-- Table SORT_COLS for join relationship
CREATE TABLE SORT_COLS
(
    SD_ID NUMBER NOT NULL,
    "COLUMN_NAME" VARCHAR2(128) NULL,
    "ORDER" NUMBER (10) NOT NULL,
    INTEGER_IDX NUMBER(10) NOT NULL
);

ALTER TABLE SORT_COLS ADD CONSTRAINT SORT_COLS_PK PRIMARY KEY
(SD_ID, INTEGER_IDX);

-- Table TBL_PRIVS for classes
[org.apache.hadoop.hive.metastore.model.MTablePrivilege]
CREATE TABLE TBL_PRIVS
(
    TBL_GRANT_ID NUMBER NOT NULL,
    CREATE_TIME NUMBER (10) NOT NULL,
    GRANT_OPTION NUMBER (5) NOT NULL,
    GRANTOR VARCHAR2(128) NULL,
    GRANTOR_TYPE VARCHAR2(128) NULL,
    PRINCIPAL_NAME VARCHAR2(128) NULL,
    PRINCIPAL_TYPE VARCHAR2(128) NULL,
    TBL_PRIV VARCHAR2(128) NULL,
    TBL_ID NUMBER NULL
);

ALTER TABLE TBL_PRIVS ADD CONSTRAINT TBL_PRIVS_PK PRIMARY KEY
(TBL_GRANT_ID);

-- Table DATABASE_PARAMS for join relationship
CREATE TABLE DATABASE_PARAMS
(
    DB_ID NUMBER NOT NULL,
    PARAM_KEY VARCHAR2(180) NOT NULL,
    PARAM_VALUE VARCHAR2(4000) NULL
);

ALTER TABLE DATABASE_PARAMS ADD CONSTRAINT DATABASE_PARAMS_PK PRIMARY KEY
(DB_ID, PARAM_KEY);

-- Table ROLE_MAP for classes
[org.apache.hadoop.hive.metastore.model.MRoleMap]
CREATE TABLE ROLE_MAP
(
    ROLE_GRANT_ID NUMBER NOT NULL,
    ADD_TIME NUMBER (10) NOT NULL,
    GRANT_OPTION NUMBER (5) NOT NULL,
    GRANTOR VARCHAR2(128) NULL,
    GRANTOR_TYPE VARCHAR2(128) NULL,
    PRINCIPAL_NAME VARCHAR2(128) NULL,
    PRINCIPAL_TYPE VARCHAR2(128) NULL,

```

```

        ROLE_ID NUMBER NULL
    );

ALTER TABLE ROLE_MAP ADD CONSTRAINT ROLE_MAP_PK PRIMARY KEY
(ROLE_GRANT_ID);

-- Table SERDE_PARAMS for join relationship
CREATE TABLE SERDE_PARAMS
(
    SERDE_ID NUMBER NOT NULL,
    PARAM_KEY VARCHAR2(256) NOT NULL,
    PARAM_VALUE VARCHAR2(4000) NULL
);

ALTER TABLE SERDE_PARAMS ADD CONSTRAINT SERDE_PARAMS_PK PRIMARY KEY
(SERDE_ID, PARAM_KEY);

-- Table PART_PRIVS for classes
[org.apache.hadoop.hive.metastore.model.MPartitionPrivilege]
CREATE TABLE PART_PRIVS
(
    PART_GRANT_ID NUMBER NOT NULL,
    CREATE_TIME NUMBER (10) NOT NULL,
    GRANT_OPTION NUMBER (5) NOT NULL,
    GRANTOR VARCHAR2(128) NULL,
    GRANTOR_TYPE VARCHAR2(128) NULL,
    PART_ID NUMBER NULL,
    PRINCIPAL_NAME VARCHAR2(128) NULL,
    PRINCIPAL_TYPE VARCHAR2(128) NULL,
    PART_PRIV VARCHAR2(128) NULL
);

ALTER TABLE PART_PRIVS ADD CONSTRAINT PART_PRIVS_PK PRIMARY KEY
(PART_GRANT_ID);

-- Table DB_PRIVS for classes
[org.apache.hadoop.hive.metastore.model.MDBPrivilege]
CREATE TABLE DB_PRIVS
(
    DB_GRANT_ID NUMBER NOT NULL,
    CREATE_TIME NUMBER (10) NOT NULL,
    DB_ID NUMBER NULL,
    GRANT_OPTION NUMBER (5) NOT NULL,
    GRANTOR VARCHAR2(128) NULL,
    GRANTOR_TYPE VARCHAR2(128) NULL,
    PRINCIPAL_NAME VARCHAR2(128) NULL,
    PRINCIPAL_TYPE VARCHAR2(128) NULL,
    DB_PRIV VARCHAR2(128) NULL
);

ALTER TABLE DB_PRIVS ADD CONSTRAINT DB_PRIVS_PK PRIMARY KEY
(DB_GRANT_ID);

-- Table TBLS for classes [org.apache.hadoop.hive.metastore.model.MTable]
CREATE TABLE TBLS
(
    TBL_ID NUMBER NOT NULL,
    CREATE_TIME NUMBER (10) NOT NULL,

```

```

    DB_ID NUMBER NULL,
    LAST_ACCESS_TIME NUMBER (10) NOT NULL,
    OWNER VARCHAR2(767) NULL,
    RETENTION NUMBER (10) NOT NULL,
    SD_ID NUMBER NULL,
    TBL_NAME VARCHAR2(128) NULL,
    TBL_TYPE VARCHAR2(128) NULL,
    VIEW_EXPANDED_TEXT CLOB NULL,
    VIEW_ORIGINAL_TEXT CLOB NULL
);

ALTER TABLE TBLS ADD CONSTRAINT TBLS_PK PRIMARY KEY (TBL_ID);

-- Table PARTITION_EVENTS for classes
[org.apache.hadoop.hive.metastore.model.MPartitionEvent]
CREATE TABLE PARTITION_EVENTS
(
    PART_NAME_ID NUMBER NOT NULL,
    DB_NAME VARCHAR2(128) NULL,
    EVENT_TIME NUMBER NOT NULL,
    EVENT_TYPE NUMBER (10) NOT NULL,
    PARTITION_NAME VARCHAR2(767) NULL,
    TBL_NAME VARCHAR2(128) NULL
);

ALTER TABLE PARTITION_EVENTS ADD CONSTRAINT PARTITION_EVENTS_PK PRIMARY
KEY (PART_NAME_ID);

-- Table SKEWED_STRING_LIST for classes
[org.apache.hadoop.hive.metastore.model.MStringList]
CREATE TABLE SKEWED_STRING_LIST
(
    STRING_LIST_ID NUMBER NOT NULL
);

ALTER TABLE SKEWED_STRING_LIST ADD CONSTRAINT SKEWED_STRING_LIST_PK
PRIMARY KEY (STRING_LIST_ID);

CREATE TABLE SKEWED_STRING_LIST_VALUES
(
    STRING_LIST_ID NUMBER NOT NULL,
    "STRING_LIST_VALUE" VARCHAR2(256) NULL,
    INTEGER_IDX NUMBER(10) NOT NULL
);

ALTER TABLE SKEWED_STRING_LIST_VALUES ADD CONSTRAINT
SKEWED_STRING_LIST_VALUES_PK PRIMARY KEY (STRING_LIST_ID, INTEGER_IDX);

ALTER TABLE SKEWED_STRING_LIST_VALUES ADD CONSTRAINT
SKEWED_STRING_LIST_VALUES_FK1 FOREIGN KEY (STRING_LIST_ID) REFERENCES
SKEWED_STRING_LIST (STRING_LIST_ID) INITIALLY DEFERRED ;

CREATE TABLE SKEWED_COL_NAMES
(
    SD_ID NUMBER NOT NULL,
    "SKEWED_COL_NAME" VARCHAR2(256) NULL,
    INTEGER_IDX NUMBER(10) NOT NULL
);

```



```

ALTER TABLE SKEWED_COL_NAMES ADD CONSTRAINT SKEWED_COL_NAMES_PK PRIMARY
KEY (SD_ID, INTEGER_IDX);

ALTER TABLE SKEWED_COL_NAMES ADD CONSTRAINT SKEWED_COL_NAMES_FK1 FOREIGN
KEY (SD_ID) REFERENCES SDS (SD_ID) INITIALLY DEFERRED ;

CREATE TABLE SKEWED_COL_VALUE_LOC_MAP
(
    SD_ID NUMBER NOT NULL,
    STRING_LIST_ID_KID NUMBER NOT NULL,
    "LOCATION" VARCHAR2(4000) NULL
);

CREATE TABLE MASTER_KEYS
(
    KEY_ID NUMBER (10) NOT NULL,
    MASTER_KEY VARCHAR2(767) NULL
);

CREATE TABLE DELEGATION_TOKENS
(
    TOKEN_IDENT VARCHAR2(767) NOT NULL,
    TOKEN VARCHAR2(767) NULL
);

ALTER TABLE SKEWED_COL_VALUE_LOC_MAP ADD CONSTRAINT
SKEWED_COL_VALUE_LOC_MAP_PK PRIMARY KEY (SD_ID, STRING_LIST_ID_KID);

ALTER TABLE SKEWED_COL_VALUE_LOC_MAP ADD CONSTRAINT
SKEWED_COL_VALUE_LOC_MAP_FK1 FOREIGN KEY (STRING_LIST_ID_KID) REFERENCES
SKEWED_STRING_LIST (STRING_LIST_ID) INITIALLY DEFERRED ;

ALTER TABLE SKEWED_COL_VALUE_LOC_MAP ADD CONSTRAINT
SKEWED_COL_VALUE_LOC_MAP_FK2 FOREIGN KEY (SD_ID) REFERENCES SDS (SD_ID)
INITIALLY DEFERRED ;

CREATE TABLE SKEWED_VALUES
(
    SD_ID_OID NUMBER NOT NULL,
    STRING_LIST_ID_EID NUMBER NOT NULL,
    INTEGER_IDX NUMBER(10) NOT NULL
);

ALTER TABLE SKEWED_VALUES ADD CONSTRAINT SKEWED_VALUES_PK PRIMARY KEY
(SD_ID_OID, INTEGER_IDX);

ALTER TABLE SKEWED_VALUES ADD CONSTRAINT SKEWED_VALUES_FK1 FOREIGN KEY
(STRING_LIST_ID_EID) REFERENCES SKEWED_STRING_LIST (STRING_LIST_ID)
INITIALLY DEFERRED ;

ALTER TABLE SKEWED_VALUES ADD CONSTRAINT SKEWED_VALUES_FK2 FOREIGN KEY
(SD_ID_OID) REFERENCES SDS (SD_ID) INITIALLY DEFERRED ;

-- column statistics

CREATE TABLE TAB_COL_STATS (
    CS_ID NUMBER NOT NULL,

```

```

DB_NAME VARCHAR2(128) NOT NULL,
TABLE_NAME VARCHAR2(128) NOT NULL,
COLUMN_NAME VARCHAR2(128) NOT NULL,
COLUMN_TYPE VARCHAR2(128) NOT NULL,
TBL_ID NUMBER NOT NULL,
LONG_LOW_VALUE NUMBER,
LONG_HIGH_VALUE NUMBER,
DOUBLE_LOW_VALUE NUMBER,
DOUBLE_HIGH_VALUE NUMBER,
BIG_DECIMAL_LOW_VALUE VARCHAR2(4000),
BIG_DECIMAL_HIGH_VALUE VARCHAR2(4000),
NUM_NULLS NUMBER NOT NULL,
NUM_DISTINCTS NUMBER,
AVG_COL_LEN NUMBER,
MAX_COL_LEN NUMBER,
NUM_TRUES NUMBER,
NUM_FALSES NUMBER,
LAST_ANALYZED NUMBER NOT NULL
);

```

```

CREATE TABLE VERSION (
  VER_ID NUMBER NOT NULL,
  SCHEMA_VERSION VARCHAR(127) NOT NULL,
  VERSION_COMMENT VARCHAR(255)
);

```

```

ALTER TABLE VERSION ADD CONSTRAINT VERSION_PK PRIMARY KEY (VER_ID);

```

```

ALTER TABLE TAB_COL_STATS ADD CONSTRAINT TAB_COL_STATS_PKEY PRIMARY KEY
(CS_ID);

```

```

ALTER TABLE TAB_COL_STATS ADD CONSTRAINT TAB_COL_STATS_FK FOREIGN KEY
(TBL_ID) REFERENCES TBLS (TBL_ID) INITIALLY DEFERRED ;

```

```

CREATE INDEX TAB_COL_STATS_N49 ON TAB_COL_STATS(TBL_ID);

```

```

CREATE TABLE PART_COL_STATS (
  CS_ID NUMBER NOT NULL,
  DB_NAME VARCHAR2(128) NOT NULL,
  TABLE_NAME VARCHAR2(128) NOT NULL,
  PARTITION_NAME VARCHAR2(767) NOT NULL,
  COLUMN_NAME VARCHAR2(128) NOT NULL,
  COLUMN_TYPE VARCHAR2(128) NOT NULL,
  PART_ID NUMBER NOT NULL,
  LONG_LOW_VALUE NUMBER,
  LONG_HIGH_VALUE NUMBER,
  DOUBLE_LOW_VALUE NUMBER,
  DOUBLE_HIGH_VALUE NUMBER,
  BIG_DECIMAL_LOW_VALUE VARCHAR2(4000),
  BIG_DECIMAL_HIGH_VALUE VARCHAR2(4000),
  NUM_NULLS NUMBER NOT NULL,
  NUM_DISTINCTS NUMBER,
  AVG_COL_LEN NUMBER,
  MAX_COL_LEN NUMBER,
  NUM_TRUES NUMBER,
  NUM_FALSES NUMBER,
  LAST_ANALYZED NUMBER NOT NULL
);

```

```
ALTER TABLE PART_COL_STATS ADD CONSTRAINT PART_COL_STATS_PKEY PRIMARY KEY
(CS_ID);
```

```
ALTER TABLE PART_COL_STATS ADD CONSTRAINT PART_COL_STATS_FK FOREIGN KEY
(PART_ID) REFERENCES PARTITIONS (PART_ID) INITIALLY DEFERRED;
```

```
CREATE INDEX PART_COL_STATS_N49 ON PART_COL_STATS (PART_ID);
```

```
CREATE INDEX PCS_STATS_IDX ON PART_COL_STATS
(DB_NAME, TABLE_NAME, COLUMN_NAME, PARTITION_NAME);
```

```
CREATE TABLE FUNCS (
  FUNC_ID NUMBER NOT NULL,
  CLASS_NAME VARCHAR2(4000),
  CREATE_TIME NUMBER(10) NOT NULL,
  DB_ID NUMBER,
  FUNC_NAME VARCHAR2(128),
  FUNC_TYPE NUMBER(10) NOT NULL,
  OWNER_NAME VARCHAR2(128),
  OWNER_TYPE VARCHAR2(10)
);
```

```
ALTER TABLE FUNCS ADD CONSTRAINT FUNCS_PK PRIMARY KEY (FUNC_ID);
```

```
CREATE TABLE FUNC_RU (
  FUNC_ID NUMBER NOT NULL,
  RESOURCE_TYPE NUMBER(10) NOT NULL,
  RESOURCE_URI VARCHAR2(4000),
  INTEGER_IDX NUMBER(10) NOT NULL
);
```

```
ALTER TABLE FUNC_RU ADD CONSTRAINT FUNC_RU_PK PRIMARY KEY (FUNC_ID,
INTEGER_IDX);
```

```
-- Constraints for table PART_COL_PRIVS for class(es)
[org.apache.hadoop.hive.metastore.model.MPartitionColumnPrivilege]
ALTER TABLE PART_COL_PRIVS ADD CONSTRAINT PART_COL_PRIVS_FK1 FOREIGN KEY
(PART_ID) REFERENCES PARTITIONS (PART_ID) INITIALLY DEFERRED ;
```

```
CREATE INDEX PART_COL_PRIVS_N49 ON PART_COL_PRIVS (PART_ID);
```

```
CREATE INDEX PARTITIONCOLUMNPRIVILEGEINDEX ON PART_COL_PRIVS
(PART_ID, "COLUMN_NAME", PRINCIPAL_NAME, PRINCIPAL_TYPE, PART_COL_PRIV, GRANTOR,
GRANTOR_TYPE);
```

```
-- Constraints for table COLUMNS_V2
ALTER TABLE COLUMNS_V2 ADD CONSTRAINT COLUMNS_V2_FK1 FOREIGN KEY (CD_ID)
REFERENCES CDS (CD_ID) INITIALLY DEFERRED ;
```

```
CREATE INDEX COLUMNS_V2_N49 ON COLUMNS_V2 (CD_ID);
```

```
-- Constraints for table PARTITION_KEY_VALS
ALTER TABLE PARTITION_KEY_VALS ADD CONSTRAINT PARTITION_KEY_VALS_FK1
FOREIGN KEY (PART_ID) REFERENCES PARTITIONS (PART_ID) INITIALLY DEFERRED
;
```

```

CREATE INDEX PARTITION_KEY_VALS_N49 ON PARTITION_KEY_VALS (PART_ID);

-- Constraints for table DBS for class(es)
[org.apache.hadoop.hive.metastore.model.MDatabase]
CREATE UNIQUE INDEX UNIQUE_DATABASE ON DBS ("NAME");

-- Constraints for table PARTITION_PARAMS
ALTER TABLE PARTITION_PARAMS ADD CONSTRAINT PARTITION_PARAMS_FK1 FOREIGN
KEY (PART_ID) REFERENCES PARTITIONS (PART_ID) INITIALLY DEFERRED ;

CREATE INDEX PARTITION_PARAMS_N49 ON PARTITION_PARAMS (PART_ID);

-- Constraints for table SERDES for class(es)
[org.apache.hadoop.hive.metastore.model.MSerDeInfo]

-- Constraints for table TYPES for class(es)
[org.apache.hadoop.hive.metastore.model.MType]
CREATE UNIQUE INDEX UNIQUE_TYPE ON TYPES (TYPE_NAME);

-- Constraints for table PARTITION_KEYS
ALTER TABLE PARTITION_KEYS ADD CONSTRAINT PARTITION_KEYS_FK1 FOREIGN KEY
(TBL_ID) REFERENCES TBLs (TBL_ID) INITIALLY DEFERRED ;

CREATE INDEX PARTITION_KEYS_N49 ON PARTITION_KEYS (TBL_ID);

-- Constraints for table ROLES for class(es)
[org.apache.hadoop.hive.metastore.model.MRole]
CREATE UNIQUE INDEX ROLEENTITYINDEX ON ROLES (ROLE_NAME);

-- Constraints for table PARTITIONS for class(es)
[org.apache.hadoop.hive.metastore.model.MPartition]
ALTER TABLE PARTITIONS ADD CONSTRAINT PARTITIONS_FK1 FOREIGN KEY (TBL_ID)
REFERENCES TBLs (TBL_ID) INITIALLY DEFERRED ;

ALTER TABLE PARTITIONS ADD CONSTRAINT PARTITIONS_FK2 FOREIGN KEY (SD_ID)
REFERENCES SDS (SD_ID) INITIALLY DEFERRED ;

CREATE INDEX PARTITIONS_N49 ON PARTITIONS (SD_ID);

CREATE INDEX PARTITIONS_N50 ON PARTITIONS (TBL_ID);

CREATE UNIQUE INDEX UNIQUEPARTITION ON PARTITIONS (PART_NAME,TBL_ID);

-- Constraints for table INDEX_PARAMS
ALTER TABLE INDEX_PARAMS ADD CONSTRAINT INDEX_PARAMS_FK1 FOREIGN KEY
(INDEX_ID) REFERENCES IDXS (INDEX_ID) INITIALLY DEFERRED ;

CREATE INDEX INDEX_PARAMS_N49 ON INDEX_PARAMS (INDEX_ID);

```

```

-- Constraints for table TBL_COL_PRIVS for class(es)
[org.apache.hadoop.hive.metastore.model.MTableColumnPrivilege]
ALTER TABLE TBL_COL_PRIVS ADD CONSTRAINT TBL_COL_PRIVS_FK1 FOREIGN KEY
(TBL_ID) REFERENCES TBLS (TBL_ID) INITIALLY DEFERRED ;

CREATE INDEX TABLECOLUMNPRIVILEGEINDEX ON TBL_COL_PRIVS
(TBL_ID,"COLUMN_NAME",PRINCIPAL_NAME,PRINCIPAL_TYPE,TBL_COL_PRIV,GRANTOR,
GRANTOR_TYPE);

CREATE INDEX TBL_COL_PRIVS_N49 ON TBL_COL_PRIVS (TBL_ID);

-- Constraints for table IDXS for class(es)
[org.apache.hadoop.hive.metastore.model.MIndex]
ALTER TABLE IDXS ADD CONSTRAINT IDXS_FK2 FOREIGN KEY (SD_ID) REFERENCES
SDS (SD_ID) INITIALLY DEFERRED ;

ALTER TABLE IDXS ADD CONSTRAINT IDXS_FK1 FOREIGN KEY (ORIG_TBL_ID)
REFERENCES TBLS (TBL_ID) INITIALLY DEFERRED ;

ALTER TABLE IDXS ADD CONSTRAINT IDXS_FK3 FOREIGN KEY (INDEX_TBL_ID)
REFERENCES TBLS (TBL_ID) INITIALLY DEFERRED ;

CREATE UNIQUE INDEX UNIQUEINDEX ON IDXS (INDEX_NAME,ORIG_TBL_ID);

CREATE INDEX IDXS_N50 ON IDXS (INDEX_TBL_ID);

CREATE INDEX IDXS_N51 ON IDXS (SD_ID);

CREATE INDEX IDXS_N49 ON IDXS (ORIG_TBL_ID);

-- Constraints for table BUCKETING_COLS
ALTER TABLE BUCKETING_COLS ADD CONSTRAINT BUCKETING_COLS_FK1 FOREIGN KEY
(SD_ID) REFERENCES SDS (SD_ID) INITIALLY DEFERRED ;

CREATE INDEX BUCKETING_COLS_N49 ON BUCKETING_COLS (SD_ID);

-- Constraints for table TYPE_FIELDS
ALTER TABLE TYPE_FIELDS ADD CONSTRAINT TYPE_FIELDS_FK1 FOREIGN KEY
(TYPE_NAME) REFERENCES TYPES (TYPES_ID) INITIALLY DEFERRED ;

CREATE INDEX TYPE_FIELDS_N49 ON TYPE_FIELDS (TYPE_NAME);

-- Constraints for table SD_PARAMS
ALTER TABLE SD_PARAMS ADD CONSTRAINT SD_PARAMS_FK1 FOREIGN KEY (SD_ID)
REFERENCES SDS (SD_ID) INITIALLY DEFERRED ;

CREATE INDEX SD_PARAMS_N49 ON SD_PARAMS (SD_ID);

-- Constraints for table GLOBAL_PRIVS for class(es)
[org.apache.hadoop.hive.metastore.model.MGlobalPrivilege]
CREATE UNIQUE INDEX GLOBALPRIVILEGEINDEX ON GLOBAL_PRIVS
(PRINCIPAL_NAME,PRINCIPAL_TYPE,USER_PRIV,GRANTOR,GRANTOR_TYPE);

```

```

-- Constraints for table SDS for class(es)
[org.apache.hadoop.hive.metastore.model.MStorageDescriptor]
ALTER TABLE SDS ADD CONSTRAINT SDS_FK1 FOREIGN KEY (SERDE_ID) REFERENCES
SERDES (SERDE_ID) INITIALLY DEFERRED ;
ALTER TABLE SDS ADD CONSTRAINT SDS_FK2 FOREIGN KEY (CD_ID) REFERENCES CDS
(CD_ID) INITIALLY DEFERRED ;

CREATE INDEX SDS_N49 ON SDS (SERDE_ID);
CREATE INDEX SDS_N50 ON SDS (CD_ID);

-- Constraints for table TABLE_PARAMS
ALTER TABLE TABLE_PARAMS ADD CONSTRAINT TABLE_PARAMS_FK1 FOREIGN KEY
(TBL_ID) REFERENCES TBLs (TBL_ID) INITIALLY DEFERRED ;

CREATE INDEX TABLE_PARAMS_N49 ON TABLE_PARAMS (TBL_ID);

-- Constraints for table SORT_COLS
ALTER TABLE SORT_COLS ADD CONSTRAINT SORT_COLS_FK1 FOREIGN KEY (SD_ID)
REFERENCES SDS (SD_ID) INITIALLY DEFERRED ;

CREATE INDEX SORT_COLS_N49 ON SORT_COLS (SD_ID);

-- Constraints for table TBL_PRIVS for class(es)
[org.apache.hadoop.hive.metastore.model.MTablePrivilege]
ALTER TABLE TBL_PRIVS ADD CONSTRAINT TBL_PRIVS_FK1 FOREIGN KEY (TBL_ID)
REFERENCES TBLs (TBL_ID) INITIALLY DEFERRED ;

CREATE INDEX TBL_PRIVS_N49 ON TBL_PRIVS (TBL_ID);

CREATE INDEX TABLEPRIVILEGEINDEX ON TBL_PRIVS
(TBL_ID, PRINCIPAL_NAME, PRINCIPAL_TYPE, TBL_PRIV, GRANTOR, GRANTOR_TYPE);

-- Constraints for table DATABASE_PARAMS
ALTER TABLE DATABASE_PARAMS ADD CONSTRAINT DATABASE_PARAMS_FK1 FOREIGN
KEY (DB_ID) REFERENCES DBS (DB_ID) INITIALLY DEFERRED ;

CREATE INDEX DATABASE_PARAMS_N49 ON DATABASE_PARAMS (DB_ID);

-- Constraints for table ROLE_MAP for class(es)
[org.apache.hadoop.hive.metastore.model.MRoleMap]
ALTER TABLE ROLE_MAP ADD CONSTRAINT ROLE_MAP_FK1 FOREIGN KEY (ROLE_ID)
REFERENCES ROLES (ROLE_ID) INITIALLY DEFERRED ;

CREATE INDEX ROLE_MAP_N49 ON ROLE_MAP (ROLE_ID);

CREATE UNIQUE INDEX USERROLEMAPINDEX ON ROLE_MAP
(PRINCIPAL_NAME, ROLE_ID, GRANTOR, GRANTOR_TYPE);

-- Constraints for table SERDE_PARAMS
ALTER TABLE SERDE_PARAMS ADD CONSTRAINT SERDE_PARAMS_FK1 FOREIGN KEY
(SERDE_ID) REFERENCES SERDES (SERDE_ID) INITIALLY DEFERRED ;

```

```

CREATE INDEX SERDE_PARAMS_N49 ON SERDE_PARAMS (SERDE_ID);

-- Constraints for table PART_PRIVS for class(es)
[org.apache.hadoop.hive.metastore.model.MPartitionPrivilege]
ALTER TABLE PART_PRIVS ADD CONSTRAINT PART_PRIVS_FK1 FOREIGN KEY
(PART_ID) REFERENCES PARTITIONS (PART_ID) INITIALLY DEFERRED ;

CREATE INDEX PARTPRIVILEGEINDEX ON PART_PRIVS
(PART_ID, PRINCIPAL_NAME, PRINCIPAL_TYPE, PART_PRIV, GRANTOR, GRANTOR_TYPE);

CREATE INDEX PART_PRIVS_N49 ON PART_PRIVS (PART_ID);

-- Constraints for table DB_PRIVS for class(es)
[org.apache.hadoop.hive.metastore.model.MDBPrivilege]
ALTER TABLE DB_PRIVS ADD CONSTRAINT DB_PRIVS_FK1 FOREIGN KEY (DB_ID)
REFERENCES DBS (DB_ID) INITIALLY DEFERRED ;

CREATE UNIQUE INDEX DBPRIVILEGEINDEX ON DB_PRIVS
(DB_ID, PRINCIPAL_NAME, PRINCIPAL_TYPE, DB_PRIV, GRANTOR, GRANTOR_TYPE);

CREATE INDEX DB_PRIVS_N49 ON DB_PRIVS (DB_ID);

-- Constraints for table TBLs for class(es)
[org.apache.hadoop.hive.metastore.model.MTable]
ALTER TABLE TBLs ADD CONSTRAINT TBLs_FK2 FOREIGN KEY (DB_ID) REFERENCES
DBS (DB_ID) INITIALLY DEFERRED ;

ALTER TABLE TBLs ADD CONSTRAINT TBLs_FK1 FOREIGN KEY (SD_ID) REFERENCES
SDS (SD_ID) INITIALLY DEFERRED ;

CREATE INDEX TBLs_N49 ON TBLs (DB_ID);

CREATE UNIQUE INDEX UNIQUETABLE ON TBLs (TBL_NAME, DB_ID);

CREATE INDEX TBLs_N50 ON TBLs (SD_ID);

-- Constraints for table PARTITION_EVENTS for class(es)
[org.apache.hadoop.hive.metastore.model.MPartitionEvent]
CREATE INDEX PARTITIONEVENTINDEX ON PARTITION_EVENTS (PARTITION_NAME);

-- Constraints for table FUNCS for class(es)
[org.apache.hadoop.hive.metastore.model.MFunctions]
ALTER TABLE FUNCS ADD CONSTRAINT FUNCS_FK1 FOREIGN KEY (DB_ID) REFERENCES
DBS (DB_ID) INITIALLY DEFERRED;

CREATE UNIQUE INDEX UNIQUEFUNCTION ON FUNCS (FUNC_NAME, DB_ID);

CREATE INDEX FUNCS_N49 ON FUNCS (DB_ID);

-- Constraints for table FUNC_RU for class(es)
[org.apache.hadoop.hive.metastore.model.MFunctions]

```

```
ALTER TABLE FUNC_RU ADD CONSTRAINT FUNC_RU_FK1 FOREIGN KEY (FUNC_ID)
REFERENCES FUNCS (FUNC_ID) INITIALLY DEFERRED;
```

```
CREATE INDEX FUNC_RU_N49 ON FUNC_RU (FUNC_ID);
```

```
-----
-- Transaction and lock tables
-- These are not part of package jdo, so if you are going to regenerate
this file you need to manually add the following section back to the
file.
-----
```

```
CREATE TABLE TXNS (
  TXN_ID NUMBER(19) PRIMARY KEY,
  TXN_STATE char(1) NOT NULL,
  TXN_STARTED NUMBER(19) NOT NULL,
  TXN_LAST_HEARTBEAT NUMBER(19) NOT NULL,
  TXN_USER varchar(128) NOT NULL,
  TXN_HOST varchar(128) NOT NULL
);
```

```
CREATE TABLE TXN_COMPONENTS (
  TC_TXNID NUMBER(19) REFERENCES TXNS (TXN_ID),
  TC_DATABASE VARCHAR2(128) NOT NULL,
  TC_TABLE VARCHAR2(128),
  TC_PARTITION VARCHAR2(767) NULL
);
```

```
CREATE TABLE COMPLETED_TXN_COMPONENTS (
  CTC_TXNID NUMBER(19),
  CTC_DATABASE varchar(128) NOT NULL,
  CTC_TABLE varchar(128),
  CTC_PARTITION varchar(767)
);
```

```
CREATE TABLE NEXT_TXN_ID (
  NTXN_NEXT NUMBER(19) NOT NULL
);
INSERT INTO NEXT_TXN_ID VALUES(1);
```

```
CREATE TABLE HIVE_LOCKS (
  HL_LOCK_EXT_ID NUMBER(19) NOT NULL,
  HL_LOCK_INT_ID NUMBER(19) NOT NULL,
  HL_TXNID NUMBER(19),
  HL_DB VARCHAR2(128) NOT NULL,
  HL_TABLE VARCHAR2(128),
  HL_PARTITION VARCHAR2(767),
  HL_LOCK_STATE CHAR(1) NOT NULL,
  HL_LOCK_TYPE CHAR(1) NOT NULL,
  HL_LAST_HEARTBEAT NUMBER(19) NOT NULL,
  HL_ACQUIRED_AT NUMBER(19),
  HL_USER varchar(128) NOT NULL,
  HL_HOST varchar(128) NOT NULL,
  PRIMARY KEY(HL_LOCK_EXT_ID, HL_LOCK_INT_ID)
);
```

```
CREATE INDEX HL_TXNID_INDEX ON HIVE_LOCKS (HL_TXNID);
```

```
CREATE TABLE NEXT_LOCK_ID (
```



```
NL_NEXT NUMBER(19) NOT NULL
);
INSERT INTO NEXT_LOCK_ID VALUES(1);
```

```
CREATE TABLE COMPACTION_QUEUE (
  CQ_ID NUMBER(19) PRIMARY KEY,
  CQ_DATABASE varchar(128) NOT NULL,
  CQ_TABLE varchar(128) NOT NULL,
  CQ_PARTITION varchar(767),
  CQ_STATE char(1) NOT NULL,
  CQ_TYPE char(1) NOT NULL,
  CQ_WORKER_ID varchar(128),
  CQ_START NUMBER(19),
  CQ_RUN_AS varchar(128)
);
```

```
CREATE TABLE NEXT_COMPACTION_QUEUE_ID (
  NCQ_NEXT NUMBER(19) NOT NULL
);
INSERT INTO NEXT_COMPACTION_QUEUE_ID VALUES(1);
```

```
-- -----
-- Record schema version. Should be the last step in the init script
-- -----
INSERT INTO VERSION (VER_ID, SCHEMA_VERSION, VERSION_COMMENT) VALUES (1,
'0.14.0', 'Hive release version 0.14.0');
```