IBM TRIRIGA Application Platform
Version 4, Release 3

Post Upgrade Maintenance Specification

Note
Before using this information and the product it supports, read the information in “Notices” on page 19.

This edition applies to version 4, release 3, of IBM® TRIRIGA® Application Platform and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About This Guide

This document provides scripts to accomplish an upgrade that removes unused columns from the `IBS_SPEC` and `IBS_SPEC_ASSIGNMENTS` tables in the IBM® TRIRIGA® database. These tables are the two largest and most used tables in the IBM TRIRIGA system. Removing unused columns from these tables reduces the storage space required.

Intended Audience

This guide supports system administrator personnel who are certified Database Administrators (DBAs).

Prerequisites

This guide assumes the reader has a basic understanding of IBM TRIRIGA Application Platform and the fundamental concepts required to operate the Web-based IBM TRIRIGA system.

Support

IBM Software Support provides assistance with product defects, answering FAQs, and performing rediscovery. View the IBM Software Support site at www.ibm.com/support.

1. Upgrade Process

Depending on the size of your database, this upgrade could take as long as 8 hours to complete. This is the reason IBM TRIRIGA has provided a separate upgrade process. The IBM TRIRIGA system will still function properly without applying this upgrade; however, you need to apply this upgrade. The process described in this document breaks the upgrade into two steps, which allows the work to be done on separate days.

When you perform the upgrade, keep the following requirements in mind:

- The upgrade scripts in this document can only be run after your upgrade to IBM TRIRIGA Application Platform 3.1 or later. After the scripts complete, you can use the database only with IBM TRIRIGA Application Platform 3.1 or later.
- Make sure no IBM TRIRIGA application instances are connected to the database before you start this upgrade. Stop the services and applications at their server.
- This upgrade MUST be performed by a certified DBA.

The overall goal of this upgrade is to remove unused columns from the `IBS_SPEC` and `IBS_SPEC_ASSIGNMENTS` tables. The following table lists the columns that the scripts will remove.

Table	Columns That Will Be Removed
`IBS_SPEC`	▪ `PROPERTY_ID` ▪ `LOCKED_BY` ▪ `COMPANY_ID` ▪ `MIN_REQD_COMPLETE` ▪ `SPECIFIER_ID` ▪ `SPECIFIER_FIRST_NAME` ▪ `SPECIFIER_LAST_NAME` ▪ `TYPICAL_SPEC_ID` ▪ `APPROVAL_HISTORY` ▪ `OBJECT_COLOR` ▪ `WF_TEMPLATE_ID` ▪ `WF_NAME` ▪ `CHANGE_SYSTEM_FLAG` ▪ `HIDE_TAB`
`IBS_SPEC_ASSIGNMENTS`	▪ `PROPERTY_ID` ▪ `UPDATED_DATE`

The upgrade process is the same for each table on each environment (Oracle or MSSQL). This process was selected for speed of upgrade.

Upgrade Process Steps

Step 1 Create a temporary table with the same columns as the table being updated, but without the columns to be removed.

Step 2 Copy all rows from the original table to the temporary table.

Step 3 Drop the original table. This removes any indexes that were on the original table.

Step 4 Rename the temporary table to the original table name.

Step 5 Create new indexes on the new table.

The remaining chapters in this guide provide scripts to perform the upgrade. You can copy the appropriate scripts from this guide:

- Oracle `IBS_SPEC` script
- Oracle `IBS_SPEC_ASSIGNMENTS` script
- Microsoft SQL Server `IBS_SPEC` script
- Microsoft SQL Server `IBS_SPEC_ASSIGNMENTS` script

2. Oracle IBS_SPEC Script

Run the following script on an Oracle system. It removes the unused columns described above. You can copy this script from this document.

Replace the string `$data_data_tblspace$` with your data tablespace name. It may be something like `TRIDATA_DATA`.

Replace the string `$data_index_tblspace$` with your index tablespace name. It may be something like `TRIDATA_INDX`.

While creating indexes, Oracle requires a continuous block of free memory. This may require that you extend the size of your index tablespace.

```
CREATE TABLE IBS_SPEC_TMP
(
  PROJECT_ID NUMBER(20)  NOT NULL,
  SPEC_ID NUMBER(20)  NOT NULL,
  REVISION NUMBER(10)  NOT NULL,
  SPEC_TEMPLATE_ID NUMBER(20)  NOT NULL,
  TYPE_NAME VARCHAR2(100)  NULL,
  STATUS VARCHAR2(1)  NOT NULL,
  SPEC_NUMBER VARCHAR2(300)  NULL,
  SPEC_NAME VARCHAR2(100)  NULL,
  DESCRIPTION VARCHAR2(300)  NULL,
  PUBLISHED_DATE DATE  NULL,
  UOM_TYPE_CODE NUMBER(4)  NULL,
  UOM VARCHAR2(100)  NULL,
  SPEC_CLASS_TYPE NUMBER(5) DEFAULT 1 NULL,
  OBJECT_ID NUMBER(20)  NOT NULL,
  ROOT_FLG NUMBER(1) DEFAULT 0 NULL,
  CREATED_BY NUMBER(10)  NULL,
  CREATED_DATE DATE DEFAULT sysdate NULL,
  UPDATED_BY NUMBER(10)  NULL,
  UPDATED_DATE TIMESTAMP(6)  NULL,
  OBJECT_STATE VARCHAR2(100)  NULL,
  LOCATION_ID NUMBER(20) DEFAULT 0 NULL,
  ORGANIZATION_ID NUMBER(20) DEFAULT 0 NULL,
  GEOGRAPHY_ID NUMBER(20) DEFAULT 0 NULL,
  QTY NUMBER(16,4) DEFAULT 0.00 NULL,
  ROLLUP_FIELDS_FLAG NUMBER(1) DEFAULT 0 NULL,
  SCHEDULABLE NUMBER(1) DEFAULT 0 NULL,
  DELETED_FLAG NUMBER(1) DEFAULT 0 NULL,
  OBJECT_PATH VARCHAR2(500)  NULL,
  AUDIT_FLAG NUMBER(1) DEFAULT 0 NULL,
  AUDIT_LEVEL VARCHAR2(20)  NULL,
  AUDIT_COMMENT_FLAG NUMBER(1) DEFAULT 0 NULL,
  AUDIT_SIGN_FLAG NUMBER(1) DEFAULT 0 NULL,
  IMAGE_MAP VARCHAR2(400)  NULL,
  CONTROL_NUMBER VARCHAR2(200)  NULL,
  PRICE NUMBER(16,4) DEFAULT 0.00 NULL,
  TEMPLATE_VERSION NUMBER(10) DEFAULT 0 NOT NULL,
  SYSTEM_FLAG NUMBER(1) DEFAULT 0 NULL,
  SHOW_TAB_ID NUMBER(10) DEFAULT 0 NULL,
```

```
  RESERVABLE_FLAG NUMBER(1) DEFAULT 0 NULL,
  GUI_ID NUMBER(20) DEFAULT 0 NULL,
  CONVERSION_GROUP VARCHAR2(100)  NULL,
  EXCHANGE_DATE DATE  NULL,
  SYSTEM_STATE NUMBER(4)  NULL
)

TABLESPACE $data_data_tblspace$
    PCTUSED    0
    PCTFREE    10
    INITRANS   10
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
LOGGING
NOCACHE
NOPARALLEL;

  INSERT INTO IBS_SPEC_TMP
  (
            PROJECT_ID,
            SPEC_ID,
            REVISION,
            SPEC_TEMPLATE_ID,
            TYPE_NAME,
            STATUS,
            SPEC_NUMBER,
            SPEC_NAME,
            DESCRIPTION,
            PUBLISHED_DATE,
            UOM_TYPE_CODE,
            UOM,
            SPEC_CLASS_TYPE,
            OBJECT_ID,
            ROOT_FLG,
            CREATED_BY,
            CREATED_DATE,
            UPDATED_BY,
            UPDATED_DATE,
            OBJECT_STATE,
            LOCATION_ID,
            ORGANIZATION_ID,
            GEOGRAPHY_ID,
            QTY,
            ROLLUP_FIELDS_FLAG,
            SCHEDULABLE,
            DELETED_FLAG,
            OBJECT_PATH,
            AUDIT_FLAG,
            AUDIT_LEVEL,
            AUDIT_COMMENT_FLAG,
            AUDIT_SIGN_FLAG,
```

```
            IMAGE_MAP,
            CONTROL_NUMBER,
            PRICE,
            TEMPLATE_VERSION,
            SYSTEM_FLAG,
            SHOW_TAB_ID,
            RESERVABLE_FLAG,
            GUI_ID,
            CONVERSION_GROUP,
            EXCHANGE_DATE,
          SYSTEM_STATE
)
     (SELECT
            PROJECT_ID,
            SPEC_ID,
            REVISION,
            SPEC_TEMPLATE_ID,
            TYPE_NAME,
            STATUS,
            SPEC_NUMBER,
            SPEC_NAME,
            DESCRIPTION,
            PUBLISHED_DATE,
            UOM_TYPE_CODE,
            UOM,
            SPEC_CLASS_TYPE,
            OBJECT_ID,
            ROOT_FLG,
            CREATED_BY,
            CREATED_DATE,
            UPDATED_BY,
            UPDATED_DATE,
            OBJECT_STATE,
            LOCATION_ID,
            ORGANIZATION_ID,
            GEOGRAPHY_ID,
            QTY,
            ROLLUP_FIELDS_FLAG,
            SCHEDULABLE,
            DELETED_FLAG,
            OBJECT_PATH,
            AUDIT_FLAG,
            AUDIT_LEVEL,
            AUDIT_COMMENT_FLAG,
            AUDIT_SIGN_FLAG,
            IMAGE_MAP,
            CONTROL_NUMBER,
            PRICE,
            TEMPLATE_VERSION,
            SYSTEM_FLAG,
            SHOW_TAB_ID,
            RESERVABLE_FLAG,
            GUI_ID,
            CONVERSION_GROUP,
            EXCHANGE_DATE,
          SYSTEM_STATE
     FROM IBS_SPEC);
```

```
  DROP TABLE IBS_SPEC;

  ALTER TABLE IBS_SPEC_TMP RENAME TO IBS_SPEC;

  ALTER TABLE IBS_SPEC ADD (
  CONSTRAINT PK_IBS_SPEC PRIMARY KEY (SPEC_ID, REVISION)
      USING INDEX
      TABLESPACE $data_index_tblspace$
          PCTFREE    10
          INITRANS   2
          MAXTRANS   255
          STORAGE    (
                          INITIAL          200M
                          MINEXTENTS       1
                          MAXEXTENTS       2147483645
                          PCTINCREASE      0
                         ));

CREATE  INDEX IDX04_IBS_SPEC ON IBS_SPEC
(SPEC_TEMPLATE_ID, SPEC_NAME, OBJECT_ID, SPEC_ID)
LOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   2
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;

CREATE  INDEX IDX06_IBS_SPEC ON IBS_SPEC
(SPEC_CLASS_TYPE, ROOT_FLG, SPEC_ID)
NOLOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   2
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;

CREATE  INDEX IDX07_IBS_SPEC ON IBS_SPEC
(OBJECT_PATH)
NOLOGGING
```

```
TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   2
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;

CREATE  INDEX SPEC_TMPL_STATE_GEO_ORG ON IBS_SPEC
(SPEC_TEMPLATE_ID, OBJECT_STATE, GEOGRAPHY_ID, ORGANIZATION_ID)
LOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   2
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;

CREATE  INDEX PERF01_IBS_SPEC ON IBS_SPEC
(SPEC_TEMPLATE_ID, SPEC_ID, PROJECT_ID)
LOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   2
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;

CREATE UNIQUE INDEX PERF02_IBS_SPEC ON IBS_SPEC
(SPEC_ID, OBJECT_ID, SPEC_NAME)
LOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   2
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
```

```
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;

CREATE UNIQUE INDEX PERF03_IBS_SPEC ON IBS_SPEC
(SPEC_ID, SYSTEM_STATE)
LOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   2
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;

CREATE  INDEX PERF04_IBS_SPEC ON IBS_SPEC
(SPEC_NAME, TYPE_NAME, SPEC_CLASS_TYPE)
LOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   10
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;
CREATE  INDEX PERF05_IBS_SPEC ON IBS_SPEC
(SPEC_ID, SPEC_TEMPLATE_ID)
LOGGING

TABLESPACE $data_index_tblspace$
    PCTFREE    10
    INITRANS   10
    MAXTRANS   255
    STORAGE    (
                    INITIAL          200M
                    MINEXTENTS       1
                    MAXEXTENTS       2147483645
                    PCTINCREASE      0
                    BUFFER_POOL      DEFAULT
                   )
NOPARALLEL;
COMMIT;
```

3. Oracle IBS_SPEC_ASSIGNMENTS Script

Run the following script on an Oracle system. It removes the unused columns described above. You can copy this script from this document.

Replace the string `$data_data_tblspace$` with your data tablespace name. It may be something like `TRIDATA_DATA`.

Replace the string `$data_index_tblspace$` with your index tablespace name. It may be something like `TRIDATA_INDX`.

While creating indexes, Oracle requires a continuous block of free memory. This may require that you extend the size of your index tablespace.

```
CREATE TABLE IBS_SPEC_ASSIGNMENTS_TMP
   (  SPEC_ID NUMBER(20,0),
      SPEC_CLASS_TYPE NUMBER(5,0) DEFAULT 1,
      ASS_SPEC_ID NUMBER(20,0),
      ASS_SPEC_CLASS_TYPE NUMBER(5,0) DEFAULT 1,
      ASS_TYPE VARCHAR2(100 BYTE),
      SPEC_TEMPLATE_ID NUMBER(20,0),
      ASS_SPEC_TEMPLATE_ID NUMBER(20,0),
      DEPENDENT_FLAG NUMBER(1,0) DEFAULT 0
   )
TABLESPACE $data_data_tblspace$
PCTFREE    10
INITRANS   10
MAXTRANS   255
STORAGE    (
            INITIAL          2000M
            NEXT             200M
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      10
            FREELISTS        2
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           )
LOGGING;

INSERT INTO IBS_SPEC_ASSIGNMENTS_TMP
(
      SPEC_ID,
      SPEC_CLASS_TYPE,
      ASS_SPEC_ID,
      ASS_SPEC_CLASS_TYPE,
      ASS_TYPE,
      SPEC_TEMPLATE_ID,
      ASS_SPEC_TEMPLATE_ID,
      DEPENDENT_FLAG
)
(SELECT SPEC_ID,
      SPEC_CLASS_TYPE,
      ASS_SPEC_ID,
```

```
      ASS_SPEC_CLASS_TYPE,
      ASS_TYPE,
      SPEC_TEMPLATE_ID,
      ASS_SPEC_TEMPLATE_ID,
      DEPENDENT_FLAG FROM IBS_SPEC_ASSIGNMENTS);

DROP TABLE IBS_SPEC_ASSIGNMENTS;

ALTER TABLE IBS_SPEC_ASSIGNMENTS_TMP RENAME TO IBS_SPEC_ASSIGNMENTS;

ALTER TABLE IBS_SPEC_ASSIGNMENTS ADD (
   CONSTRAINT PK_IBS_SPEC_ASSIGNMENTS PRIMARY KEY (SPEC_ID, ASS_SPEC_ID,
ASS_TYPE)
   USING INDEX
   TABLESPACE $data_index_tblspace$
   PCTFREE    10
   INITRANS   10
   MAXTRANS   255
   COMPUTE STATISTICS
   STORAGE    (
            INITIAL          1000M
            NEXT             100M
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      10
            FREELISTS        2
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           ));

CREATE INDEX ASS_CTYP_TMPL_SPID ON IBS_SPEC_ASSIGNMENTS (ASS_SPEC_ID,
ASS_TYPE, SPEC_CLASS_TYPE, SPEC_TEMPLATE_ID, SPEC_ID)
LOGGING
TABLESPACE $data_index_tblspace$
PCTFREE    10
INITRANS   10
MAXTRANS   255
COMPUTE STATISTICS
STORAGE    (
            INITIAL          1000M
            NEXT             100M
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      10
            FREELISTS        2
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           );

CREATE INDEX IDX03_IBS_SPEC_ASSIGN ON IBS_SPEC_ASSIGNMENTS (SPEC_ID,
ASS_TYPE, ASS_SPEC_CLASS_TYPE, ASS_SPEC_TEMPLATE_ID)
LOGGING
TABLESPACE $data_index_tblspace$
PCTFREE    10
INITRANS   10
MAXTRANS   255
COMPUTE STATISTICS
```

```
STORAGE    (
            INITIAL          1000M
            NEXT             100M
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      10
            FREELISTS        2
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           );

CREATE INDEX PERF01_IBS_SPEC_ASSIGNMENTS ON IBS_SPEC_ASSIGNMENTS
(ASS_SPEC_ID, SPEC_ID)
LOGGING
TABLESPACE $data_index_tblspace$
PCTFREE    10
INITRANS   10
MAXTRANS   255
COMPUTE STATISTICS
STORAGE    (
            INITIAL          1000M
            NEXT             100M
            MINEXTENTS       1
            MAXEXTENTS       2147483645
            PCTINCREASE      10
            FREELISTS        2
            FREELIST GROUPS  1
            BUFFER_POOL      DEFAULT
           );

COMMIT;
```

4. MSSQL IBS_SPEC Script

Run the following script on an MSSQL system. It removes the unused columns described above. You can copy this script from this document. Run this script from Microsoft SQL Server Management Studio.

```
CREATE TABLE [IBS_SPEC_TMP] (
    [PROJECT_ID] [numeric](20)  NOT NULL,
    [SPEC_ID] [numeric](20)  NOT NULL,
    [REVISION] [numeric](10)  NOT NULL,
    [SPEC_TEMPLATE_ID] [numeric](20)  NOT NULL,
    [TYPE_NAME] [varchar] (100) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [STATUS] [varchar] (1) COLLATE SQL_Latin1_General_CP1_CS_AS  NOT NULL,
    [SPEC_NUMBER] [varchar] (300) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [SPEC_NAME] [varchar] (100) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [DESCRIPTION] [varchar] (300) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [PUBLISHED_DATE] [datetime]  NULL,
    [UOM_TYPE_CODE] [numeric](4)  NULL,
    [UOM] [varchar] (100) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [SPEC_CLASS_TYPE] [numeric](5) DEFAULT (1) NULL,
    [OBJECT_ID] [numeric](20)  NOT NULL,
    [ROOT_FLG] [numeric](1) DEFAULT (0) NULL,
    [CREATED_BY] [numeric](10)  NULL,
    [CREATED_DATE] [datetime] DEFAULT (getdate()) NULL,
    [UPDATED_BY] [numeric](10)  NULL,
    [UPDATED_DATE] [datetime]  NULL,
    [OBJECT_STATE] [varchar] (100) COLLATE SQL_Latin1_General_CP1_CS_AS
NULL,
    [LOCATION_ID] [numeric](20) DEFAULT (0) NULL,
    [ORGANIZATION_ID] [numeric](20) DEFAULT (0) NULL,
    [GEOGRAPHY_ID] [numeric](20) DEFAULT (0) NULL,
    [QTY] [numeric](16, 4) DEFAULT (0.00) NULL,
    [ROLLUP_FIELDS_FLAG] [numeric](1) DEFAULT (0) NULL,
    [SCHEDULABLE] [numeric](1) DEFAULT (0) NULL,
    [DELETED_FLAG] [numeric](1) DEFAULT (0) NULL,
    [OBJECT_PATH] [varchar] (500) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [AUDIT_FLAG] [numeric](1) DEFAULT (0) NULL,
    [AUDIT_LEVEL] [varchar] (20) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [AUDIT_COMMENT_FLAG] [numeric](1) DEFAULT (0) NULL,
    [AUDIT_SIGN_FLAG] [numeric](1) DEFAULT (0) NULL,
    [IMAGE_MAP] [varchar] (400) COLLATE SQL_Latin1_General_CP1_CS_AS  NULL,
    [CONTROL_NUMBER] [varchar] (200) COLLATE SQL_Latin1_General_CP1_CS_AS
NULL,
    [PRICE] [numeric](16, 4) DEFAULT (0.00) NULL,
    [TEMPLATE_VERSION] [numeric](10) DEFAULT (0) NOT NULL,
    [SYSTEM_FLAG] [numeric](1) DEFAULT (0) NULL,
    [SHOW_TAB_ID] [numeric](10) DEFAULT (0) NULL,
    [RESERVABLE_FLAG] [numeric](1) DEFAULT (0) NULL,
    [GUI_ID] [numeric](20) DEFAULT (0) NULL,
    [CONVERSION_GROUP] [varchar] (100) COLLATE SQL_Latin1_General_CP1_CS_AS
NULL,
    [EXCHANGE_DATE] [datetime]  NULL,
    [SYSTEM_STATE] [numeric](4)  NULL
)
GO
```



```
INSERT INTO IBS_SPEC_TMP
(
          PROJECT_ID,
          SPEC_ID,
          REVISION,
          SPEC_TEMPLATE_ID,
          TYPE_NAME,
          STATUS,
          SPEC_NUMBER,
          SPEC_NAME,
          DESCRIPTION,
          PUBLISHED_DATE,
          UOM_TYPE_CODE,
          UOM,
          SPEC_CLASS_TYPE,
          OBJECT_ID,
          ROOT_FLG,
          CREATED_BY,
          CREATED_DATE,
          UPDATED_BY,
          UPDATED_DATE,
          OBJECT_STATE,
          LOCATION_ID,
          ORGANIZATION_ID,
          GEOGRAPHY_ID,
          QTY,
          ROLLUP_FIELDS_FLAG,
          SCHEDULABLE,
          DELETED_FLAG,
          OBJECT_PATH,
          AUDIT_FLAG,
          AUDIT_LEVEL,
          AUDIT_COMMENT_FLAG,
          AUDIT_SIGN_FLAG,
          IMAGE_MAP,
          CONTROL_NUMBER,
          PRICE,
          TEMPLATE_VERSION,
          SYSTEM_FLAG,
          SHOW_TAB_ID,
          RESERVABLE_FLAG,
          GUI_ID,
          CONVERSION_GROUP,
          EXCHANGE_DATE,
        SYSTEM_STATE
)
    (SELECT
          PROJECT_ID,
          SPEC_ID,
          REVISION,
          SPEC_TEMPLATE_ID,
          TYPE_NAME,
          STATUS,
          SPEC_NUMBER,
          SPEC_NAME,
          DESCRIPTION,
```

```
            PUBLISHED_DATE,
            UOM_TYPE_CODE,
            UOM,
            SPEC_CLASS_TYPE,
            OBJECT_ID,
            ROOT_FLG,
            CREATED_BY,
            CREATED_DATE,
            UPDATED_BY,
            UPDATED_DATE,
            OBJECT_STATE,
            LOCATION_ID,
            ORGANIZATION_ID,
            GEOGRAPHY_ID,
            QTY,
            ROLLUP_FIELDS_FLAG,
            SCHEDULABLE,
            DELETED_FLAG,
            OBJECT_PATH,
            AUDIT_FLAG,
            AUDIT_LEVEL,
            AUDIT_COMMENT_FLAG,
            AUDIT_SIGN_FLAG,
            IMAGE_MAP,
            CONTROL_NUMBER,
            PRICE,
            TEMPLATE_VERSION,
            SYSTEM_FLAG,
            SHOW_TAB_ID,
            RESERVABLE_FLAG,
            GUI_ID,
            CONVERSION_GROUP,
            EXCHANGE_DATE,
        SYSTEM_STATE
    FROM IBS_SPEC)
    GO

  DROP TABLE IBS_SPEC
  GO

  EXEC sp_rename 'IBS_SPEC_TMP', 'IBS_SPEC'
  GO

ALTER TABLE [IBS_SPEC] ADD
    CONSTRAINT [PK_IBS_SPEC] PRIMARY KEY  CLUSTERED  ([SPEC_ID], [REVISION])
GO

CREATE INDEX [IDX04_IBS_SPEC] ON [IBS_SPEC] ([SPEC_TEMPLATE_ID], [SPEC_NAME],
[OBJECT_ID], [SPEC_ID])
GO

CREATE INDEX [IDX06_IBS_SPEC] ON [IBS_SPEC] ([SPEC_CLASS_TYPE], [ROOT_FLG],
[SPEC_ID])
GO

CREATE INDEX [IDX07_IBS_SPEC] ON [IBS_SPEC] ([OBJECT_PATH])
GO
```

```
CREATE INDEX [PERF01_IBS_SPEC] ON [IBS_SPEC] ([SPEC_TEMPLATE_ID], [SPEC_ID],
[PROJECT_ID])
GO

CREATE INDEX [PERF02_IBS_SPEC] ON [IBS_SPEC] ([SPEC_ID], [OBJECT_ID],
[SPEC_NAME])
GO

CREATE INDEX [PERF03_IBS_SPEC] ON [IBS_SPEC] ([SPEC_ID], [SYSTEM_STATE])
GO

CREATE INDEX [PERF04_IBS_SPEC] ON [IBS_SPEC] ([SPEC_NAME], [TYPE_NAME],
[SPEC_CLASS_TYPE])
GO

CREATE INDEX [PERF05_IBS_SPEC] ON [IBS_SPEC] ([SPEC_ID], [SPEC_TEMPLATE_ID])
GO

CREATE INDEX [SPEC_TMPL_STATE_GEO_ORG] ON [IBS_SPEC] ([SPEC_TEMPLATE_ID],
[OBJECT_STATE], [GEOGRAPHY_ID], [ORGANIZATION_ID])
GO
```

5. MSSQL IBS_SPEC_ASSIGNMENTS Script

Run the following script on an MSSQL system. It removes the unused columns described above. You can copy this script from this document. Run this script from Microsoft SQL Server Management Studio.

```
CREATE TABLE [IBS_SPEC_ASSIGNMENTS_TMP] (
    [SPEC_ID] [numeric](20)  NOT NULL,
    [SPEC_CLASS_TYPE] [numeric](5) DEFAULT (1) NULL,
    [ASS_SPEC_ID] [numeric](20)  NOT NULL,
    [ASS_SPEC_CLASS_TYPE] [numeric](5) DEFAULT (1) NULL,
    [ASS_TYPE] [varchar] (100) COLLATE SQL_Latin1_General_CP1_CS_AS  NOT
NULL,
    [SPEC_TEMPLATE_ID] [numeric](20)  NULL,
    [ASS_SPEC_TEMPLATE_ID] [numeric](20)  NULL,
    [DEPENDENT_FLAG] [numeric](1) DEFAULT (0) NULL
)
GO

INSERT
INTO IBS_SPEC_ASSIGNMENTS_TMP
  (
    SPEC_ID,
    SPEC_CLASS_TYPE,
    ASS_SPEC_ID,
    ASS_SPEC_CLASS_TYPE,
    ASS_TYPE,
    SPEC_TEMPLATE_ID,
    ASS_SPEC_TEMPLATE_ID,
    DEPENDENT_FLAG
  )
  (SELECT SPEC_ID,
      SPEC_CLASS_TYPE,
      ASS_SPEC_ID,
      ASS_SPEC_CLASS_TYPE,
      ASS_TYPE,
      SPEC_TEMPLATE_ID,
      ASS_SPEC_TEMPLATE_ID,
      DEPENDENT_FLAG
    FROM IBS_SPEC_ASSIGNMENTS
  )
GO

DROP TABLE IBS_SPEC_ASSIGNMENTS
GO

EXEC sp_rename 'IBS_SPEC_ASSIGNMENTS_TMP', 'IBS_SPEC_ASSIGNMENTS'
GO

ALTER TABLE [IBS_SPEC_ASSIGNMENTS] ADD
    CONSTRAINT [PK_IBS_SPEC_ASSIGNMENTS] PRIMARY KEY  CLUSTERED  ([SPEC_ID],
[ASS_SPEC_ID], [ASS_TYPE])
GO
```

```
CREATE  INDEX [ASS_CTYP_TMPL_SPID] ON [IBS_SPEC_ASSIGNMENTS] ([ASS_SPEC_ID],
[ASS_TYPE], [SPEC_CLASS_TYPE], [SPEC_TEMPLATE_ID], [SPEC_ID])
GO

CREATE  INDEX [IDX03_IBS_SPEC_ASSIGN] ON [IBS_SPEC_ASSIGNMENTS] ([SPEC_ID],
[ASS_TYPE], [ASS_SPEC_CLASS_TYPE], [ASS_SPEC_TEMPLATE_ID])
GO

CREATE  INDEX [PERF01_IBS_SPEC_ASSIGNMENTS] ON [IBS_SPEC_ASSIGNMENTS]
([ASS_SPEC_ID], [SPEC_ID])
GO
```

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Trademarks

IBM, the IBM logo, ibm.com, and TRIRIGA are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.