

White Space Television Extraction Logic

Revised as of: 2012.03.26

This paper describes the logic for extracting operating broadcast television station assignments as well as Special Temporary Authorizations (STAs) from the Commission's daily exportation of tables from the Media Bureau's Consolidated Database System (CDBS). These operating assignments must be protected from unlicensed devices that operate on unused frequencies in the broadcast television bands (white space operations).

The extraction logic provided below does not detail all of the code used to perform the several steps in the logical process described. The software code that the Commission uses for the subroutines described in the logic is provided in the subdirectory "/SourceCode" of the compressed file "cnvcdb.zip". This file can be downloaded from www.fcc.gov/oet/info/software/suss/. The subroutines are written in Microsoft Visual Basic version 6. Other implementations of this logic that give identical results are acceptable.

The raw CDBS data files contained in the compressed file "all-cdb-files.zip" can be downloaded from <ftp://ftp.fcc.gov/pub/Bureaus/MB/Databases/cdb/>.

The CnvCdb.exe extraction logic.

Step 1: Create the Microsoft ACCESS database shell "cdbfdb.mdb"

Step 2: Create the following empty tables within "cdbfdb.mdb" using the following subroutines:

"CreateTblApplication" which creates the table "CdbApplication"
"CreateTblAppTracking" which creates the table "CdbAppTracking"
"CreateTblFacility" which creates the table "CdbFacility"
"CreateTblIfSta" which creates the table "CdbIfSta"
"CreateTblTvEngData_Tmp" which creates the table "CdbsvEngData_Tmp"
"CreateTblTvEngData_Sta_x" which creates the table "CdbTvEngData_Sta_x"
"CreateTblTvEngData_Sta" which creates the table "CdbTvEngData_Sta"
"CreateTblTvEngData" which creates the table "CdbTvEngData"

Step 3: Run the following subroutines to process the raw CDBS data tables that have been downloaded from the FCC web site into the empty tables created above. The functions of these subroutines are described below.

CnvCdbApplication
CnvCdbAppTracking
CnvCdbFacility
CnvCdbFacility
CnvCdbIfSta
CnvCdbTvEngData_Tmp
CnvCdbTvEngData_Asr
CnvCdbTvEngData_L_P_1
CnvCdbTvEngData_L_P_2

CnvCdbstvEngData_Sta
 CnvCdbstvEngData_Sta_X
 ExportTvWsData_US_L_P
 ExportTvWsData_US_Sta
 ExportTvWsData_US_Sta_X
 ExportTvWsData_Ca_Mx

It should be noted that some Digital Television (DT) stations have one or more “holes” in their service area, *i.e.*, no signal caused by topographic conditions. The DT license holder can fill-in those unserved areas by using either Distributed Transmission System facilities or Digital Replacement Service facilities.

Distributed Transmission Systems (DTS, see Section 73.626). DTS (Service code = DD) facilities operate on the same channel as DT station, have the same Call Sign and Facility ID number as the DT station, and have one Application number which covers all of the DD facilities. If there is more than one DD facility, additional DD facilities are assigned Site Numbers beginning with number 2.

Digital Replacement Service (See Section 74.787(a)(5)). Digital Television translator or low power (Service code = LD) facilities must operate on a different channel from the parent DT station, have the same Call Sign and Facility ID number as the parent DT station, but have individual Application numbers.

Step 4: The subroutine “CnvCdbstvEngData_Tmp” is used to populate the table “TvEngData_Tmp” with ALL the records contained in the file “tv_eng_data.dat”.

Step 5: The subroutine “CnvCdbstvEngData_Asr” is to retrieve NAD83 latitude and longitude coordinates, ground elevation, structure height and overall height data from the Antenna Structure Registration database if the "TvEngDat_Tmp" record has an Antenna Structure Registration Number (ASRN), and calculate the difference between NAD27 and NAD83 coordinates and append this information to the end "TvEngDat_Tmp" record.

Step 6: The subroutine “CnvCdbstvEngData_L_P_1” is used to populate the table “TvEngData_L_P” with those records contained in the table “Tv_Eng_Data_Tmp” that meet the following criteria: *i.e.*, EngRecordType = "C" and ' TvDomStatus = "LIC", *etc.*

```
EngRecordType = "C" and TvDomStatus = "LIC"
and
VsdService = "DT" - Digital Television full-service station or
              "DC" - Digital Television -- Class 'A' or
              "CA" - Class 'A' Television -- analog or
              "LD" - Digital television – translator or low power or
              "TX" - TV translator or Low power Television (LPTV) station – analog
or
              "DD" - Distributed Transmission System station
where
KeyTv <> "*-DD-0"
```

Additionally, this subroutine creates an index field “KeyTv” which is added to each record in the tables “TvEngData_Tmp” and “TvEngData” with the following values: (1) For “VsdService” codes equal to “DT”, “DC”, “CA” and “TX”, the key is composed of the “FacilityId” + “VsdService”; (2) For “VsdService” codes equal to “DD”, the key is composed of the “FacilityId” + “VsdService” + “SiteNumber”; and (3) For “VsdService” codes equal to “LD”, the key is composed of “FacilityId” + “VsdService” + “SiteNumber” + “StationChannel”

Step 6: The subroutine “CnvCdbstvEngData_L_P_2” is used to populate the table “TvEngData_L_P” with those records contained in the table “Tv_Eng_Data_Tmp” that meet the following criteria: i.e., EngRecordType = "P" and TvDomStatus = "APP", *etc.*

```
EngRecordType = "P" AndTvDomStatus = "APP"
and
VsdService = "DT" - Digital Television full-service station or
              "DC" - Digital Television -- Class 'A' or
              "CA" - Class 'A' Television – analog or
              "LD" - Digital television – translator or low power or
              "TX" - TV translator or Low power Television (LPTV) station – analog
              or
              "DD" - Distributed Transmission System station
where
KeyTv <> "*-DD-0"
```

If the value of “KeyTv” in tables “TvEngData_Tmp” and “TvEngData” match, then the contents of the “TvEngData_Tmp” record replace the contents of the “TvEngData” record. If there is no match of “KeyTv” between the two tables, then the contents of the “TvEngData_Tmp” record are added to the “TvEngData” table where the above criteria are met.

Step 7: The subroutine “ExportTvWsData_US_Bad” attempts to identify “TX-LD” and “CA-DC” pairings. If the digital record utilizes the same frequency and site location as the analog record, then the digital record replaces the analog record. If the two records do not share the same channel and location, then no action is taken to either record. Once the ‘license application’ has been granted, the problem goes away.

As analog Class “A” (CA) and Low-power TV / TV Translator (TX) facilities are converted to digital Class “A” (DC) and Low-power / Translator (LD) facilities, two “TvEngData” records may appear when a license application is filed for a digital facility, because they represent two different Radio Services and may operate on different channels and at two different locations.

For “TX-LD” pairings, the following SQL statement is used

```
gtSql = "SELECT CdbstvEngData.FacilityId, CdbstvEngData.KeyTv " & _
        "From CdbstvEngData " & _
        "ORDER BY CdbstvEngData.KeyTv"
```

to loop thru the following code

```
gtLD = rsSQL!KeyTv
gtFacilityId = Format(Val(rsSQL!FacilityId), "000000")
```

```

giPosLD = InStr(1, gtLD, "LD", 1)
If (giPosLD > 0) Then
    glRecCnt = glRecCnt + 1
    If (glRecCnt > glNumRecs) Then
        GoTo ExitTvLd
    End If
    rsSQL.MoveNext
    gtTX = Trim(gtFacilityId) & "-TX"
    If (gtTX = rsSQL!KeyTv) Then
        Print #giTvWsFF_US, gtLD & " | " & gtTX
        Print #giTvWsFF_US, ""
    End If
    gtLD = ""
    gtTX = ""
End If
rsSQL.MoveNext

```

For "CA-DC" pairings, the following SQL statement is used

```

gtSql = "SELECT CdbstvEngData.FacilityId, CdbstvEngData.KeyTv " & _
        "From CdbstvEngData " & _
        "ORDER BY CdbstvEngData.KeyTv DESC"

```

to loop thru the following code

```

gtDC = rsSQL!KeyTv
gtFacilityId = Format(Val(rsSQL!FacilityId), "000000")
giPosDC = InStr(1, gtDC, "DC", 1)
If (giPosDC > 0) Then
    glRecCnt = glRecCnt + 1
    If (glRecCnt > glNumRecs) Then
        GoTo ExitTvLd
    End If
    rsSQL.MoveNext
    gtCA = Trim(gtFacilityId) & "-DC"
    If (gtCA = rsSQL!KeyTv) Then
        Print #giTvWsFF_US, gtDC & " | " & gtCA
        Print #giTvWsFF_US, ""
    End If
    gtDC = ""
    gtCA = ""
End If
rsSQL.MoveNext

```

The results of this analysis are exported to the file "tvwsdata_us_a2d.txt"

Step 8: The subroutine "CnvCdbstvEngData_Sta" is used to populate the table "TvEngData_Sta" with all the records contained in "TvEngData_Tmp" using the following code:

```

gtSql = "SELECT CdbstvEngData.FacChannel, CdbstvEngData.FacCallsign, CdbstvEngData_Tmp.FacilityId, " & _
        "CdbstvEngData_Tmp.EngRecordType, CdbstvEngData_Tmp.VsdService, " & _

```

```

“CdbstvEngData_Tmp.ApplicationId, CdbsvApplication.FacCallsign, CdbsvApplication.FacCallsign, ” & _
“CdbsvApplication.AppType, CdbsvApplication.StationChannel, CdbsvApplication.AppArn, ” & _
“CdbsvAppTracking.CpExpDate ” & _
“FROM CdbsvFacility RIGHT JOIN (CdbstvEngData_Tmp LEFT JOIN (CdbsvApplication LEFT JOIN ” & _
“CdbsvAppTracking ON CdbsvApplication.ApplicationId = CdbsvAppTracking.ApplicationId) ON ” & _
“CdbstvEngData_Tmp.ApplicationId = CdbsvApplication.ApplicationId) ON CdbsvFacility.FacilityId = ” & _
“CdbstvEngData_Tmp.FacilityId ” & _
“Where (((CdbstvEngData_Tmp.FacilityId) > 1) And ((CdbsvApplication.AppType) = ” & Chr(34) &
“STA” & Chr(34) & “) And ” & _
“((CdbsvAppTracking.CpExpDate) = ” & Chr(34) & gtTodaysDate & Chr(34) & “ Or ” & _
“(CdbsvAppTracking.CpExpDate) > ” & Chr(34) & gtTodaysDate & Chr(34) & “))” & _
“ORDER BY CdbstvEngData_Tmp.FacilityId, CdbstvEngData_Tmp.ApplicationId”

```

Do While Not rsSQL.EOF

```

rsCdbstvEngData_Tmp.Index = “idxApplicationId”
rsCdbstvEngData_Tmp.Seek “=”, rsSQL!ApplicationId
If (Not rsCdbstvEngData_Tmp.NoMatch) Then
    “Code to used to populate the table TvEngData_Sta”
End If
rsSQL.MoveNext

```

Loop

Step 9: If a station renews an existing STA which is contained in the “CdbstvEngData_Sta”, a new entry in the “CdbstvEngData_Sta” table is NOT created. Instead, an entry is made in the “CdbsvIfSta” table which cross references the original STA entry in the “CdbstvEngData_Sta” table. The subroutine “CnvCdbstvEngData_Sta_X” is then used to update the table “TvEngData_Sta_X” with all the records contained in “TvEngData_Tmp” using the following code:

```

gtSql = “SELECT CdbsvFacility.FacCallsign, CdbsvFacility.FacilityId, CdbsvFacility.FacStatus, ” & _
“CdbsvFacility.FacChannel, ” & _
“CdbsvApplication.ApplicationId, CdbsvApplication.AppArn, CdbsvApplication.AppService, ” & _
“CdbsvApplication.AppType, CdbsvAppTracking.CpExpDate, CdbsvIfSta.RefAppArn, ” & _
“CdbsvIfSta.RefFilePrefix ” & _
“FROM (CdbsvFacility INNER JOIN (CdbsvApplication INNER JOIN CdbsvAppTracking ON ” & _
“CdbsvApplication.ApplicationId = CdbsvAppTracking.ApplicationId) ON CdbsvFacility.FacilityId = ” & _
“CdbsvApplication.FacilityId) INNER JOIN CdbsvIfSta ON CdbsvAppTracking.ApplicationId = ” & _
“CdbsvIfSta.ApplicationId ” & _
“Where (((CdbsvFacility.FacStatus) = ” & Chr(34) & “LICEN” & Chr(34) & “) And ” & _
“((CdbsvApplication.AppService) = ” & Chr(34) & “CA” & Chr(34) & “ OR ” & _
“(CdbsvApplication.AppService) = ” & Chr(34) & “DC” & Chr(34) & “ OR ” & _
“(CdbsvApplication.AppService) = ” & Chr(34) & “DD” & Chr(34) & “ OR ” & _
“(CdbsvApplication.AppService) = ” & Chr(34) & “DS” & Chr(34) & “ Or ” & _
“(CdbsvApplication.AppService) = ” & Chr(34) & “DT” & Chr(34) & “ Or ” & _
“(CdbsvApplication.AppService) = ” & Chr(34) & “DX” & Chr(34) & “ Or ” & _
“(CdbsvApplication.AppService) = ” & Chr(34) & “LD” & Chr(34) & “ Or ” & _
“(CdbsvApplication.AppService) = ” & Chr(34) & “TX” & Chr(34) & “) And ” & _
“((CdbsvApplication.AppType) = ” & Chr(34) & “STAX” & Chr(34) & “) And ” & _
“((CdbsvAppTracking.CpExpDate) = ” & Chr(34) & gtTodaysDate & Chr(34) & “ Or ” & _
“(CdbsvAppTracking.CpExpDate) > ” & Chr(34) & gtTodaysDate & Chr(34) & “))” & _
“ORDER BY CdbsvFacility.FacilityId, CdbsvAppTracking.CpExpDate”

```

Do While Not rsSQL.EOF

```

rsCdbsvApplication.Index = “idxAppArn”
rsCdbsvApplication.Seek “=”, rsSQL!RefAppArn
If (Not rsCdbsvApplication.NoMatch) Then
    gtApplicationId = rsCdbsvApplication!ApplicationId
    rsCdbstvEngData_Tmp.Index = “idxApplicationId”

```

```

rsCdbstvEngData_Tmp.Seek "=", gtApplicationId
If (Not rsCdbstvEngData_Tmp.NoMatch) Then
    "Code to used to populate the table TvEngData_Sta"
End If
rsSQL.MoveNext
End If
Loop

```

Step 10: United States stations – The subroutine “ExportTvWsData_US_L_P” is used to export the contents of “TvEngData_L_P” as the file “tvwsdata_us_l_p.txt”

Step 11: United States stations – The subroutine “ExportTvWsData_US_Sta” is used to export the contents of “TvEngData_Sta” as the file “tvwsdata_us-sta.txt”

Step 12: United States stations – The subroutine “ExportTvWsData_US_Sta_X” is used to export the contents of “TvEngData_Sta_X” as the file “tvwsdata_us_sta_x.txt”

Step 13: Mexican stations – The subroutine “ExportTvWsData_Ca_Mx” is used to export Mexican assignments from the table “TvEngData_Tmp” which must be protected by White Space devices. The outputs are contained in the files “tvwsdata_mx.txt”.

The following tables (CdbstvEngData_Tmp, CdbstvFacility) are joined by an SQL statement to handle this process, as follows:

```

gtSql = "SELECT CdbstvFacility.FacCountry, CdbstvFacility.CommState, CdbstvFacility.CommCity, " & _
        "CdbstvEngData_Tmp.StationChannel, CdbstvFacility.FacilityId, CdbstvEngData_Tmp.ApplicationId, " & _
        "CdbstvEngData_Tmp.VsdService, CdbstvEngData_Tmp.EngRecordType, " & _
        "CdbstvEngData_Tmp.TvDomStatus FROM CdbstvEngData_Tmp LEFT JOIN CdbstvFacility ON " & _
        "CdbstvEngData_Tmp.FacilityId = CdbstvFacility.FacilityId WHERE " & _
        "(((CdbstvFacility.FacCountry)="MX") AND " & _
        "((CdbstvEngData_Tmp.StationChannel)>=1 And (CdbstvEngData_Tmp.StationChannel)<52) AND " & _
        "((CdbstvFacility.FacilityId)>0) AND ((CdbstvEngData_Tmp.EngRecordType)="C") AND " & _
        "((CdbstvEngData_Tmp.TvDomStatus)="GRANT")) ORDER BY CdbstvFacility.FacCountry, " & _
        "CdbstvFacility.CommState, CdbstvFacility.CommCity, CdbstvEngData_Tmp.StationChannel, " & _
        "CdbstvFacility.FacilityId, CdbstvEngData_Tmp.ApplicationId "

```

NOTES:

- 1 The Mexican records contained in the “CDBS” database are based upon information supplied by Mexican authorities to the Commission's International Bureau.
- 2 Live Canadian data is now available for download from www.fcc.gov/encyclopedia/whitespace-database-administration
- 2 It should be noted that geographical coordinates for United States assignments are expressed in the NAD 27 datum, while the geographical coordinates for Canadian and Mexican assignments are expressed in the NAD 83 datum.
- 3 If you find any other questionable “TvEngData” records, please advise us of them.

If you have questions, please contact:
Donald Draper Campbell
Senior Engineer
Office of Engineering and Technology
202-418-2405
donald.campbell@fcc.gov