Archived:Hiding unnecessary access points from user display on Symbian

Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

Description

On some S60 3rd Edition, Feature Pack 2 devices there is a hidden access point called "IPDC" which is used only by the mobile TV engine and should not be presented to end users.

Depending on the method used for access point handling, this AP may be listed as one of the connection points. Typically, the problem occurs when using the deprecated Access Point Engine (APEngine) API.

Solution

These applications should deal with access points in S60 3rd Edition, Feature Pack 2 by using the Connection Method Manager (CMManager) component. If CMManager is used, the IPDC access point will remain hidden.

APEngine is deprecated from S60 3rd Edition, FP2 onwards and using it is discouraged. However, if its still required for compatibility reasons, note the following things:

The communications database in, for example, the Nokia N96 is constructed as follows:

```
Access Point Name IAP ID WAP ID
        Easy WLAN 1 1
             IPDC 2         -
         internet 3 2
              mms 4 3
              wap 5 4
```

APEngine returns the WAP ID to application. If the application needs to have an IAP ID, then `CApUtils::IapIdFromWapIdL()` can be used for conversion between WAP ID and IAP ID. When accessing the communications database directly (not using CMManager or APEngine), the same can be achieved by reading the WAP ID from the `WAP_ACCESS_POINT` table and then mapping that ID to the correct IAP ID:

```
// commDb is a pointer to CCommsDatabase
CCommsDbTableView* wapTable = commDb->OpenViewMatchingUintLC(
                                TPtrC(WAP_ACCESS_POINT),
                                TPtrC(COMMDB_ID), wapID );

User::LeaveIfError( wapTable->GotoFirstRecord() );

TBuf<64> wapBearer;
wapTable->ReadTextL(TPtrC(WAP_CURRENT_BEARER), wapBearer);

if ( wapBearer != TPtrC(WAP_IP_BEARER) )
  {
  User::Leave( KErrInvalidBearerType );
  }

CCommsDbTableView* bearerTable = commDb->OpenViewMatchingUintLC(
                                   TPtrC(wapBearer),
                                   TPtrC(WAP_ACCESS_POINT_ID),
```


(C) Copyright Nokia 2014. All rights reserved.

```
                                    wapID );

User::LeaveIfError( bearerTable->GotoFirstRecord() );
bearerTable->ReadUintL(TPtrC(WAP_IAP), iapID );
CleanupStack::PopAndDestroy(2); // bearerTable, wapTable
```

See also

S60 Platform: Access Points Example with Roaming Support - Example application for access point handling on S60 3rd and 5th Edition devices, including application-level roaming using service network access points (SNAPs), supported from S60 3rd Edition, FP2 onwards.

 (C) Copyright Nokia 2014. All rights reserved.