Archived:Speech Encoder Config Symbian API

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*}}.

Note: :This API is not part of the public SDK. It can be found in the SDK API Plug-in.

The Speech Encoder Config API acts as an interface for configuring speech encoders, in particular AMR-NB.

Using the API, it is possible to retrieve supported bit rates for a particular audio codec as well as set the encoding bit rate. Previously, the only public way to configure the AMR encoder was *Audio Proxy Server* (APS uses this interface internally). From S60 3rd Edition, Feature Pack 2 onwards, the speech encoder API can be used directly. A reference to either a successfully initialized `CMMFDevSound` or `CMdaAudioInputStream` instance must be passed to the `CSpeechEncoderConfig::NewL()` constructor.

The following code snippet shows how to set the AMR encoding mode using the Speech Encoder Config API:

Header file

```
speechencoderconfig.h
```

Library

```
SpeechEncoderConfig.lib
```

Code example

```
CSpeechEncoderConfig* custInt =
    CSpeechEncoderConfig::NewL( iDevSound );
// iDevSound is initialized with AMN-NB FourCC code
if (custInt)
  {
  TInt iBitRate = 4750;  // AMR-NB 4.75 kbps (mode 0)
  // set the encoding mode(0) to the HW device
  TInt err = custInt->SetBitrate( iBitRate );
  if( err != KErrNone )
    {
    CEikonEnv::InfoWinL(_L("Error setting AMR encoding mode"),_L(""));
    }
  else
    {
    CEikonEnv::InfoWinL(_L("Setting AMR encoding mode successful"),_L(""));
    }
```

Using the Speech Encoder Config API, it is possible to set the VAD (Voice Activity Detection) mode as well as to get the status of VAD.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.