

**Note!**

This API is not part of the public SDK. It can be found in the [SDK API Plug-in](#).

Purpose

The services offered by DOS(Domestic OS) server can be accessed through DOS Client API. The services offered by DOS Client API are ? Registering for listening and receiving DOS events, accessory related services, bluetooth audio related services etc.

Use cases

1. We can get notification when an accessory like wired headset, Bluetooth headset is connected or disconnected. We can listen to accessory mode change events using CDosEventListenerBase.
2. We can also listen to wired headset button events with CDosEventListenerBase

Example code

Header

```
#include <dossvrservices.h> // To access dos services
```

Link against

```
LIBRARY dsclient.lib
```

1. Derive from CDosEventListenerBase.
2. Connect to dos server session

```
RDosServer session;
session.Connect();
```

Usecase1: Get notification when an accessory is connected or disconnected

3. Start listening to accessory mode changed events.

```
StartListeningL(KAccessoryModeChanged, sizeof(TDosAccessoryMode), EOnlyLast);
```

4. Callback function AccessoryModeChangedL will be called whenever it detects an accessory.

```
void CDosServExampleAppUi::AccessoryModeChangedL(TDosAccessoryMode aAccessoryState)
```

Purpose

KAccessoryModeChanged_API

```
{
    switch(aAccessoryState)
    {
    case EDosAccNotConnected:
    {
        ::Infof(CIKonf, "Accessory not connected"), _L("");
    }
    break;
    case EDosAccUnsupportedConnected:
    {
        ::Infof(CIKonf, "Unsupported accessory"), _L("");
    }
    break;
    case EDosAccModeHeadset:
    {
        ::Infof(CIKonf, "Wired Headset detected"), _L("");
    }
    break;
    };
}
```

Usecase2 :To listen to wired headset button events

3. Start listening to wired headset button change events

```
StartListeningL(KHeadsetButtonChanged, sizeof(EPSButtonState), EOnlyLast);
```

4. Callback function HeadsetButtonChangedL will be called whenever there is headset buttonpress.

```
void CDosServExampleAppUi::HeadsetButtonChangedL(EPSButtonState aState)
{
    switch(aState)
    {
    case EPSButtonUp:
    {
        CEiknf5WynL(_L("Button up event"), _L(""));
    }
    break;
    case EPSButtonDown:
    {
        CEiknf5WynL(_L("Button down event"), _L(""));
    }
    break;
    case EPSButtonDownLongPress:
    {
        CEiknf5WynL(_L("Long press "), _L(""));
    }
    break;
    };
}
```

Example project

[File:DosServExam.zip](#)

Known Issues

Note!

While the API is include in the S60 3rd Edition FP1 version of the SDK API Plug-in it is a known issue that it will not work with S60 3rd Edition FP1 phones.