



<b>ID</b>	CS001356	<b>Creation date</b>	May 6, 2009
<b>Platform</b>	S60 3rd Edition, FP2; S60 5th Edition	<b>Tested on devices</b>	Nokia 6220 Classic, Nokia 5800 XpressMusic
<b>Category</b>	Symbian C++	<b>Subcategory</b>	Messaging, SMS

**Keywords (APIs, classes, methods, functions):** CMsvSession, CClientMtmRegistry, CSmsClientMtm, CMsvEntry, CMsvEntrySelection, CMsvStore, MMsVSessionObserver, TMsvSelectionOrdering, TMsvId, CRichText, CMsvSession::OpenAsyncL(), CClientMtmRegistry::NewMtmL(), CMsvEntry::ChildrenL(), CSmsClientMtm::SwitchCurrentEntryL(), CMsvSession::GetEntryL(), CMsvEntry::ReadStoreL(), CSmsClientMtm::Body(), CMsvStore::RestoreBodyTextL()

## Overview

This code snippet demonstrates how to list SMS messages in the Inbox.

## MMP file

The following capabilities and libraries are required:

```
CAPABILITY ReadUserData
```

```
LIBRARY msgs.lib
```

## Header file

```
#include <msvapi.h>

// FORWARD DECLARATIONS
class CClientMtmRegistry;
class CSmsClientMtm;

class CAppUi : public CAknAppUi, public MMsVSessionObserver
{
public: // Methods from base classes
    /**
     * From MMsVSessionObserver
     */

```

## CS001356 - Symbian C++: Listing messages in Inbox

```
void HandleSessionEventL(TMsvSessionEvent aEvent, TAny* aArg1,
                        TAny* aArg2, TAny* aArg3);

private: // New methods
    void ListInboxContentsL();

private: // Data
    CMsvSession* iSession;
    CClientMtmRegistry* iMtmRegistry;
    CSmsClientMtm* iSmsMtm;

// Other methods and data omitted for brevity
}
```

## Source file

```
#include <msvapi.h>
#include <msvids.h> // KMsvGlobalInBoxIndexEntryId
#include <mtclreg.h> // CClientMtmRegistry
#include <smsclnt.h> // CSmsClientMtm
#include <smut.h> // KUidMsgTypeSMS
#include <txtrich.h> // CRichText
```

In the constructor, initialise the member variables:

```
// Open the session to message server (asynchronous)
iSession = CMsvSession::OpenAsyncL(*this);

// Create an MTM Registry object
iMtmRegistry = CClientMtmRegistry::NewL(*iSession);

// Create an SMS Client MTM object
iSmsMtm = STATIC_CAST(CSmsClientMtm*, iMtmRegistry->NewMtmL(KUidMsgTypeSMS));
```

In the destructor, free the pointers:

```
delete iSmsMtm;
delete iMtmRegistry;
delete iSession;

void CAppUi::ListInboxContentsL()
{
    // Access the Inbox
    TMsvSelectionOrdering sort;
    CMsvEntry* inboxContext = CMsvEntry::NewL(*iSession,
        KMsvGlobalInBoxIndexEntryId, sort);
    CleanupStack::PushL(inboxContext);

    // Get all entries in the Inbox
    CMsvEntrySelection* entries = inboxContext->ChildrenL();
    CleanupStack::PushL(entries);

    TInt messages = entries->Count();
    for (TInt i = 0; i < messages; i++)
    {
        TMsvId entryID = entries->At(i);
        iSmsMtm->SwitchCurrentEntryL(entryID);
    }
}
```

## CS001356\_-\_Symbian\_C++:\_Listing\_messages\_in\_Inbox

```
CMsvEntry* entry = iSession->GetEntryL((*entries)[i]);
CleanupStack::PushL(entry);

CMsvStore* inboxStore= entry->ReadStoreL();
CleanupStack::PushL(inboxStore);
if (inboxStore->HasBodyTextL())
{
    // Get the sender's details
    TMsvEntry smsEntry = entry->Entry();
    HBufC* aText = smsEntry.iDetails.AllocL();
    CleanupStack::PushL(aText);
    iAppView->LogPrintL(*aText);
    CleanupStack::PopAndDestroy(); // aText

    // Get the SMS contents
    CRichText& richText= iSmsMtm->Body();
    inboxStore->RestoreBodyTextL(richText);
    const TInt length = richText.DocumentLength();

    HBufC* smsContent = richText.Read(0, length).AllocL();
    CleanupStack::PushL(smsContent);
    richText.Reset();

    iAppView->LogPrintL(*smsContent);
    CleanupStack::PopAndDestroy(); // smsContent
}
else
{
    // The SMS contains no text
}

CleanupStack::PopAndDestroy(2, entry);
}

CleanupStack::PopAndDestroy(2, entries);
}

void CAppUi::HandleSessionEventL(TMsvSessionEvent aEvent,
                                TAny* aArg1, TAny* aArg2,
                                TAny* /*aArg3*/)
{
    // Not implemented
}
```

## Postconditions

The application lists the SMS messages in the Inbox on the screen.

## Supplementary material

This code snippet is part of the stub concept, which means that it has been patched on top of a template application in order to be more useful for developers. The version of the Symbian stub application used as a template in this snippet is v1.1.

- The patched, executable application that can be used to test the features described in this snippet is

## CS001356\_-\_Symbian\_C++:\_Listing\_messages\_in\_Inbox

available for download at [Media:SymbianStub\\_w\\_Inbox\\_handling.zip](#).

- You can view all the changes that are required to implement the above-mentioned features. The changes are provided in unified diff and colour-coded diff (HTML) formats in [Media:listing\\_messages\\_in\\_inbox.diff.zip](#).
- For general information on applying the patch, see [Using Diffs](#).
- For unpatched stub applications, see [Example stub](#).