

## Sending SMS Messages to email address

### ◆ ITAPI\_SendSMS()

- Note: Max length of email address is device dependent
- May experience buffer overrun and device reset

```
ITAPI_SendSMS (pITapi,          // interface pointer
               "fjr@fredraab.com", //email
               "Hi There",      // message text
               0,                // Remote app set to 0
               MySMSCallback,    // callback
               pMe);             // my own data
```

## IWeb Interfaces

### ◆ IWeb

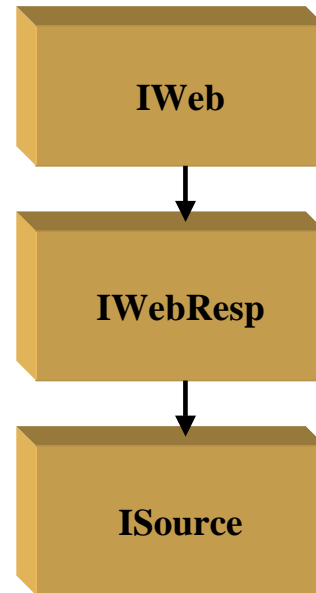
- Sets HTTP header options
- Gets / Posts to URL
- Returns pointer to IWebResponse

### ◆ IWebResponse

- Checks for errors
- Returns pointer to ISource for incoming data

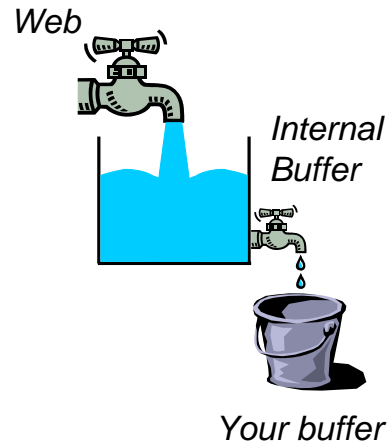
### ◆ ISource

- Reads response from server

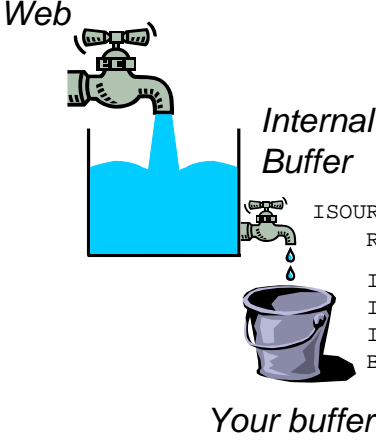


## ISource Methods

- ◆ **ISOURCE\_Read( )**  
Reads data from internal buffer
- ◆ **ISOURCE\_Readable( )**  
Register CB when internal buffer is empty
- ◆ **ISOURCE\_Exhausted( )**  
TRUE when “Web is empty”



## ISOURCE\_Read



*Web*

*Internal Buffer*

*Your buffer*

```
ISOURCE_Read(Size of Your Buffer)
```

Returns:

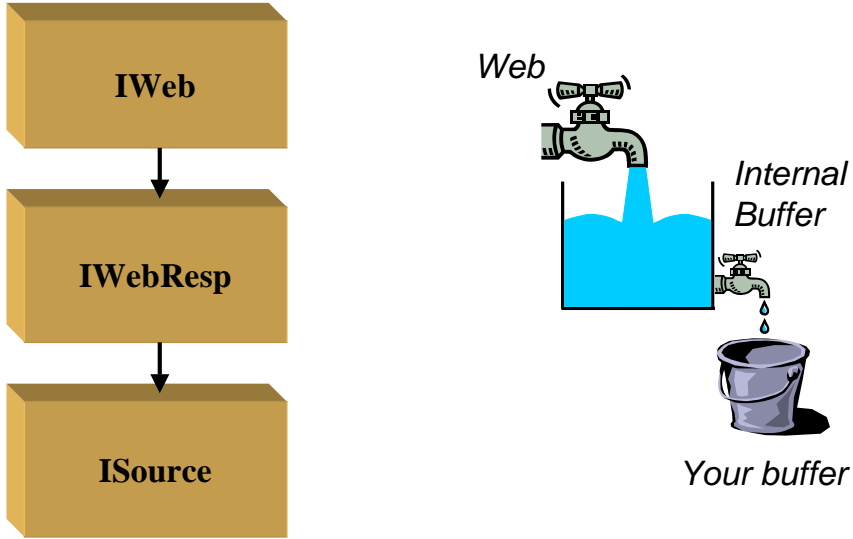
- ISOURCE\_WAIT - Internal Buffer Empty
- ISOURCE\_ERROR - Error Occurred Somewhere
- ISOURCE\_END - Got everything from server
- ByteCount - number of bytes returned

QUALCOMM PROPRIETARY 5

## ISOURCE\_Read Usage

```
static void ReadFromWebCB(MyApp *pMe) {
    ByteCount = ISOURCE_Read(pMe->pISource, (char*)buf, sizeof(buf));
    switch (ByteCount) {
        case ISOURCE_WAIT:
            ISOURCE_Readable(pMe->pISource, &pMe->WebCBStruct);
            return;
        case ISOURCE_ERROR:
            DisplayError(pMe, "Connection error...");
            return;
        case ISOURCE_END:
            DisplayResults(pMe);
            return;
        default:
            // process data in buffer (copy or write to file)
            MEMCPY(finalDest, buf, ByteCount);
            finalDest = finalDest + ByteCount;
            ISOURCE_Readable(pMe->pISource, &pMe->WebCBStruct);
            return;
    } // end switch
}
```

## Review IWeb Sample Code Handout



The diagram illustrates the data flow and a water tap analogy. On the left, three brown boxes are stacked vertically: **IWeb** at the top, **IWebResp** in the middle, and **ISource** at the bottom. Arrows point downwards from IWeb to IWebResp, and from IWebResp to ISource. On the right, a water tap labeled **Web** is shown pouring blue water into a container labeled **Internal Buffer**. A second tap on the right side of the container is dripping a single drop into a bucket labeled **Your buffer**.

QUALCOMM PROPRIETARY 7