

How-to Guide

# Mobile Client Technology

## How To... Code Choice Autocomplete

Version 1.00 - August 2007

Applicable Release: CRM Mobile Technology 4.0 and 5.0

© Copyright 2007 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary. These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

1	Introduction.....	1
2	Prerequisites .....	1
3	Mechanism.....	1
4	Handling of Empty Control Values .....	2
5	Implementation.....	3

# 1 Introduction

The *Auto Complete* feature allows a user to fill in a few characters in a choice field and the actual value gets displayed in the control on tabbing out.

## 2 Prerequisites

To provide the *Auto Complete* feature the UI Framework makes the following assumptions on the modeling information for a choice control and its choice tileset:

1. At least one choice tileset has been defined for the choice control.
2. The main anchor of the choice tileset is the *BusinessQuery* object used for finding the result of this choice control.
3. The result anchor of the choice tileset is holding the Business collection which is the query result of the main anchor.
4. The *InputEditable* attribute of the choice control is set to *true*.
  - This can be done at design time as a property of the choice control in the MAS or
  - At runtime by setting the *inputEditable* attribute of the choice control in the code (for example, in the *onLoad* event handler of the tile).

## 3 Mechanism

The choice control first raises an event *BeforeShowPopup* where the application developer has the choice to change the default modeled choice tileset.

```
Public Event BeforeShowPopup(ByVal defaultPopupTSEnumVal As Integer, ByRef newPopupTSEnumVal As Integer)
```

### Note:

If this event is not listened to, then the default choice tileset is used for further processing.

Depending on the choice tileset, the *DisplayAttribute*, *AlternateKeyAttribute* and the *QueryObjectAttribute* are picked up from the modeled entries:

- *DisplayAttribute*: name of the attribute (from the *BusinessObject* of the result anchor of the choice tileset) which should be displayed (can be blank).
- *AlternateKeyAttribute*: name of an alternative attribute name (from the *BusinessObject* of the result anchor of the choice tileset) which will be used instead of the *DisplayAttribute*.
- *QueryObjectAttribute*: name of the attribute (from the *QueryObject* of the main anchor of the choice tileset) which is used for querying in the popup.

The underlying query object of the main anchor is then created and the value already added in the choice field is then set to the query object for the attribute identified by *QueryObjectAttribute*, after adding a '\*' at the end shown as below:

```
bq.SetAttribute(mQOAttribute, value.toString() + "*")
```

The choice control raises an event *OnQuery*, which can be used to initialize the query parameters of the *BOL BusinessQuery* object. This query object will be used by the UI to trigger the query in order to try the auto completion.

### Note:

The parameter *run* can be used to cancel the *Auto Complete* functionality by application developers and has to be explicitly set to *true* to make it to work (default value is *false*).

```
Public Event onQuery(ByVal query As SFABOFLIB.BusinessQuery,  
ByRef run As Boolean)
```

UI Framework then fires the query. After firing the query the UI distinguishes four different situations:

1. Query did crash - This might happen sometimes. In this case we will raise a SAP exception, which will be displayed in the status bar.
2. Query did return no result - The choice window opens displaying these query parameters and the empty list.
3. Query did return exactly one item – depending on the button pressed in the choice window the corresponding values will be changed.
4. Query did return more than one item - The choice window will open displaying these query parameters and the list of the returned items.

If the choice window is opened by the *Auto Complete* feature and the user closes the choice window without selecting an item, the choice control becomes invalid. Again tabbing out of the choice control does not lead to a re-run of the *Auto Complete*, because the UI Framework knows that this value does not lead to a valid result.

This is also true in the following scenario: In the database there are two employees: *Mueller, Marco* and *Mueller, Peter*. In a choice control for employees the user types *Muell* and leaves the field. This will open the Choice window displaying both employees in the search result. The user closes the window without selecting an item. The control displaying *Muell* becomes invalid. When removing again the focus from the control, the *Auto Complete* would not be newly triggered, because the user did not want to select an item from the displayed and already known list of matching items. Instead of that he can press the choice button to see the list again and select an item out of it.

If the bound attribute of this choice control has changed, this change will be treated as valid if the BOL tells that it is a valid change of the attribute. This means a change of the control value via the business logic will not lead to triggering the *Auto Complete* when leaving the field.

## 4 Handling of Empty Control Values

If the content of the choice control has been changed by the user to an empty string,

```
Check: len(cstr(ctrl.value))==0
```

then the UI will call the internal blank function of the choice control, which will set the related target attribute of the choice control to nothing or empty depending on the basic type of the target attribute object or key.

## 5 Implementation

To provide auto completion for one choice control the application developer has to define an event handler for the choice control event *onQuery*. Inside this event handler only the query parameters have to be set along with setting *run* to *true*.

```
Private Sub ctrlChoice_onQuery(ByVal query As
SFABOFLIB.BusinessQuery, ByRef run As Boolean)
    With query
        .SetAttribute("Mandt", gBOLSettings.getValue("Mandt"))
        .SetAttribute("Langu", "EN")
    End With
    run = true
End Sub
```

**Note:**

*onQuery* event is also raised on clicking the *Choice* button. Here, the event parameter *run* of the event *onQuery* has no effect. Setting this parameter in the event handler will be ignored by the UI.