



OpenDCL Migration Guide

Mit diesem Dokument erhalten Sie ein paar Tipps zur Migration Ihrer alten Projektdateien aus OpenDCL 4 bzw. ObjectDCL 3.0 (oder früher) in die neueste Version von OpenDCL 5.

Werden Fehlermeldungen der OpenDCL Laufzeitumgebung innerhalb von AutoCAD ausgegeben oder verhält sich die OpenDCL Laufzeitumgebung oder das OpenDCL Studio unerwartet, empfehlen wir Ihnen **dringend**, zunächst die kontextsensitive Hilfe des Dialogs bzw. Steuerelements zu konsultieren, ob sich die Syntax von Funktionen oder allgemein das Verhalten des Steuerelements, seiner Eigenschaften, Methoden und Ereignissen geändert hat, um den Quellcode Ihrer Applikation anzupassen.

Problem:	Keine der "ODCL"-Funktionen lässt sich mehr aufrufen. <code>(odcl_loadproject "myProject")</code> <code>; error: no function definition: ODCL_LOADPROJECT</code>
Lösung:	Durchsuchen Sie den gesamten Quellcode nach "odcl_" und ersetzen Sie den Text durch "dcl_"
Grund:	Zur Vermeidung von Namenskonflikten zwischen OpenDCL und dem Vorgänger ObjectDCL3.0
Geändert in:	V4.0

Problem:	Für <code>odcl_grid_setcurrow</code> gibt es keine direkte Entsprechung. <code>(dcl_grid_setcurrow myProject_myForm_myGrid 0)</code> <code>; error: no function definition: DCL_GRID_SETCURROW</code>
Lösung:	Verwenden Sie stattdessen die Funktion <code>dcl_Grid_SetCurCell</code> und lassen Sie das Argument für <code>inColumn</code> weg oder weisen Sie dem Argument den Wert -1 zu.
Grund:	Zur Vermeidung von redundanten Funktionen.
Geändert in:	V4.0

Problem:	OpenDCL Funktionen akzeptieren nicht länger leere Argumente. <code>(dcl_Form_IsActive MyProj_MyForm)</code> <code>; error: ADS request error</code>
Lösung:	Rufen Sie <code>(dcl_project_load "myProject")</code> ohne die vorherige Prüfung mit der Methode "IsActive" auf. Ist der Dialog bereits geladen, bleibt die Funktion ohne Effekt.
Grund:	Das war eine notwendige Änderung zugunsten einer besseren Fehlerdokumentation.

Geändert in:	V4.0
--------------	------

Problem:	Fehler: Falscher Datentyp für das Argument Funktion: <code>dcl_ListBox_GetCurSel</code> Argument: 0 Kann die Funktion(<code>dcl_ListBox_GetCurSel</code>) nicht auf eine Optionsliste anwenden.
Lösung:	Verwenden Sie stattdessen (<code>dcl_OptionList_GetCurSel</code>).
Grund:	N/A
Geändert in:	V5.0

Problem:	Fehler: Zu viele Argumente Funktion: <code>dcl_messagebox</code> Argument: 4
Lösung:	Das 4. Argument {ShowHelpButton } muss jetzt ein Boolean sein, keine Ganzzahl.
Grund:	N/A
Geändert in:	V5.0

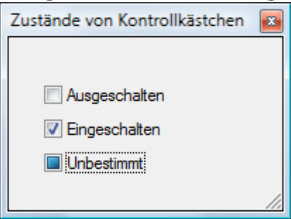
Problem:	<code>(dcl_ListView_Clear)</code> löscht nicht mehr die Spalten.
Lösung:	Verwenden Sie stattdessen (<code>dcl_ListView_DeleteColumns</code>).
Grund:	N/A
Geändert in:	V5.0

Problem:	Die Argumente für <code>(dcl_ListView_FillList)</code> haben sich geändert.
Lösung:	<code>(dcl_ListView_FillList <CONTROL> als Liste von Listen in der Form ((strItemLabel [als String] {intItemImageIndex [als Integer]} {strColumn1Label [als String]} {intColumn1ImageIndex [als Integer]} ...))</code>
Grund:	N/A
Geändert in:	V5.0

Problem:	Die Argumente für <code>(dcl_ListView_AddColumns)</code> haben sich geändert.
Lösung:	<code>(dcl_ListView_AddColumns <CONTROL> IstNewColumns [als Liste von Listen in der Form ((strCaption [als String] {intJustification [als Integer]} {intColumnWidth [als Long]} {intColumnImageIndex [als Integer]}) ...))</code>
Grund:	N/A
Geändert in:	V5.0

Problem:	Die Argumente für <code>(dcl_ListView_AddItem)</code> haben sich geändert.
Lösung:	<code>(dcl_ListView_AddItem <CONTROL> {intImageIndex [als Integer]} strItemLabel [als String] {strColumn1Label [als String]} strColumn2Label [als String] ...)</code>

Grund:	N/A
Geändert in:	V5.0

Problem:	Das Textargument in (<code>dcl_ListView_SetItemText</code>) wurde verschoben.
Lösung:	<code>(dcl_ListView_SetItemText <CONTROL> intItem [als Long] {intColumn [als Long]} strLabel [als String])</code>
Grund:	N/A
Geändert in:	V5.0
Problem:	Kontrollkästchen und Optionsfelder geben den aktuellen Zustand nicht mehr als Boolean zurück. <code>(if (dcl_Control_GetValue MyProj_Form1_CheckBox1) (MachWas))</code>
Lösung:	Ändern Sie den Quellcode in dieser Art und Weise: <code>(if (= 1 (dcl_Control_GetValue MyProj_Form1_CheckBox1)) (MachWas))</code> oder <code>(if (> 0 (dcl_Control_GetValue MyProj_Form1_CheckBox1)) (MachWas))</code>
Grund:	Kontrollkästchen unterstützen nun drei verschiedene Zustände als Integer: (0 = Ausgeschaltet, 1 = Eingeschalten, 2 = Unbestimmt). 
Geändert in:	V5.0

Problem:	Die Funktion (<code>dcl_Control_SetValue</code>) erwartet eine Ganzzahl
Lösung:	Ändern Sie (<code>dcl_Control_SetValue MyProj_Form1_CheckBox1 T</code>) in (<code>dcl_Control_SetValue MyProj_Form1_CheckBox1 1</code>)
Grund:	Kontrollkästchen unterstützen nun drei verschiedene Zustände als Integer: (0 = Ausgeschaltet, 1 = Eingeschalten, 2 = Unbestimmt).
Geändert in:	V5.0

Empfohlene Änderungen.

Problem:	Die Funktion (<code>dcl_loadproject</code>) ist nunmehr obsolet.
Lösung:	Durchsuchen Sie Ihren Quellcode nach " <code>dcl_loadproject</code> " und ersetzen Sie es durch " <code>dcl_project_load</code> "
Grund:	Zugunsten weiterer Funktionalitäten
Geändert in:	V4.0

Problem:	Die Eigenschaften (VarName) bzw. "Lisp Symbol name" werden lediglich aufrechterhalten, um die Funktionsfähigkeit des existierenden Quellcodes zu gewährleisten.
Lösung:	OpenDCL generiert automatisch den Namen der Variable für das Steuerelement zur Laufzeit aus der Eigenschaft (Name), z.B. MeinProjekt_Dialog1_Steuerelement1. Sie können durchaus den alten Wert in (VarName) entfernen, indem Sie aus dem Menü Extras den Befehl AutoLISP-Symbolnamen löschen aufrufen. Bitte beachten Sie, dass dies nachhaltige Auswirkungen auf Ihren Quellcode hat und Sie ihn daraufhin aktualisieren müssen, um die vorgegebenen Namen der Steuerelemente zu verwenden.
Grund:	Zur Vermeidung von Irritationen aufgrund zweier "Namen"-Eigenschaften.
Geändert in:	V4.0

Bitte teilen Sie uns umgehend mit, wenn Sie weitere Änderungen gefunden haben, die Sie für erwähnenswert halten. Senden Sie einfach eine e-Mail an: f.tomke@opendcl.de