



ODC 20
25

Omnis Developer Conference

Ihre Entwicklung auf die nächste Stufe bringen:

Neue Funktionen in Omnis Studio 11.2, die die Messlatte höher legen

Andreas H. Pfeiffer

JavaScript Remote Forms

Construct Row Variable

Die `$construct-Row-Variable` in einem Remote-Task hat jetzt eine zusätzliche Spalte namens `headers`, die eine Zeile enthält, in der jede Spalte ein Name-Wert-Paar für jeden vom Client empfangenen HTTP-Header enthält.

Der `$construct row-Parameter` in einem Remote-Task hat jetzt eine zusätzliche Spalte namens `cookies`, die eine Zeile mit jeder Spalte enthält, die einem Name-Wert-Paar für jedes Cookie entspricht, z.B. `cookie1=hello` führt zu einer Zeile mit dem Spaltennamen `'cookie'` und `'hello'` als Wert.

Kopieren von Daten in die Zwischenablage

- `$copytoclipboard(vDataToCopy[,cMIMEMType='text/plain',bDatalsBinary=kFalse])`

On evClick

Do `$cinst.$copytoclipboard(iText)`

On evClick

Do `$cinst.$copytoclipboard(iText,"text/html")`

On evClick

Do `$cinst.$copytoclipboard(iImageData,"image/png",kTrue)`

Subformset Panels

Funktioniert ähnlich wie ein Subform Control.

- `$panelcount`
- `$currentpanel`
- `$panelclassname`
- `$closebutton`
- `$esctoclose`
- `$openmax`
- `$panelsubformparams`



Anpassen der Keyboard Shortcuts

Sie können jetzt Tastatureingaben im Remote Form oder Control mithilfe von benutzerdefinierten Ereignishandlern abfangen und ändern.

\$keyboardshortcut (class or field)

pSystemKey, pCharacterKey, pModifiers, pFocusedControl

```
If pCharacterKey="s"&|ControlKey=kTrue
```

```
  Do $cinst.$save()
```

```
End If
```

Lesen und Kopieren von Daten aus der Zwischenablage

\$readclipboard

diese Remote Form Methode ermöglicht es Ihnen, das letzte Datenelement in der Zwischenablage des Endbenutzers zu lesen, was wiederum die Funktion

\$onclipboardread Remote Form Methode

Übergabe von drei Parametern: pSuccess, pPlainText, pExtraData

pExtraData ist eine Row, die alle anderen mit den Daten verbundenen MIME-Typen enthält (z. B.

'text/html', 'image/png').

Return Methoden

Wenn eine vom Server ausgeführte Methode vom Client aus aufgerufen wird, wird ein Versprechen zurückgegeben, es sei denn, es gibt eine passende ..._return-Methode, in diesem Fall wird nach Abschluss die return-Methode aufgerufen.

Der Rückgabewert der Server-Methode wird als Parameter in der Promise-Funktion "then" übergeben. Auf diese Weise können Sie Ihre asynchrone Logik in der Methode verketteten, den Kontext beizubehalten und weiterhin Variablen zu verwenden, die in der äußeren

Methode definiert sind. Zum Beispiel:

```
Do $cinst.$myServerMethod() Returns IPromise
```

```
JavaScript: IPromise.then( (returnVal) => {
```

```
JavaScript: IRetVal = returnVal; // Assign the parameter to a local variable that Omnis code knows about
```

```
Do $cinst.$showmessage(IRetVal,"Server Method Complete")
```

```
JavaScript: });
```

Push Notifications - openpush

Der Client-Befehl 'openpush' (ausgeführt mit `$clientcommand`) hat zwei neue optionale `row()`-Parameter `iMaxTries` und `cRetryCodes`.

- **iMaxTries**

Die maximale Anzahl von Versuchen, die eine Anfrage automatisch durchgeführt wird, wenn sie einen http

Statuscode liefert, der mit `cRetryCodes` übereinstimmt. -1 bedeutet unbegrenzt. Voreinstellung ist 5.

- **cRetryCodes**

HTTP-Statuscodes, die als "temporäre Fehler" zu behandeln sind. Diese werden automatisch wiederholt

automatisch bis zu `iMaxTries` wiederholt, bevor ein Fehler gemeldet wird. Dies ist eine Zeichenkette die durch Kommas getrennte Codes und/oder Codebereiche enthält, z. B. "404,500-599"

bedeutet

alle Antworten mit dem http-Statuscode 404 oder zwischen 500 und 599 werden automatisch erneut versucht werden. Standardwert ist 500-599.

Push Notifications - \$pushclosed

Es gibt eine neue vom Client ausgeführte Methode \$pushclosed, die für Remote Forms implementiert werden kann und (auf allen Forms) aufgerufen wird, wenn eine Push-Verbindung geschlossen wird.

Geben Sie in dieser Methode kTrue zurück, um das Standardverhalten zu verhindern.

\$pushclosed hat zwei Parameter:

- **pErrorCode**

Werte ungleich Null bedeuten, dass sie aufgrund eines Fehlers geschlossen wurde.

0 bedeutet, dass er durch den Aufruf des \$clientcommand 'closepush' geschlossen wurde.

> 1 bedeutet, dass pErrorCode der http-Statuscode ist, der den Fehler verursacht hat. Eine Fehler Meldung wird angezeigt, es sei denn, Sie geben kTrue zurück, um das Standardverhalten zu verhindern.

-1 bedeutet einen Transportfehler, d.h. die Anfrage konnte nicht zugestellt werden.

- **pWillRestart**

Wenn true, wird die Push-Verbindung standardmäßig neu gestartet. Geben Sie kTrue zurück, um dies zu verhindern.

Subform Palette - Palette Arrow

Wenn Sie den Client-Befehl "subformpaletteshow" verwenden, um eine Unterformularpalette zu öffnen, können Sie das Flag `kSFSPaletteHideArrow` im Zeilenparameter `iPositionFlags` verwenden, um den normalerweise im Palettenfenster angezeigten Pfeil auszublenden.

Subform Palette - Control Name

Wenn der Name des Steuerelements im Parameter cControl des Client-Befehls "subformpalettshow" nicht gefunden wird (oder Sie den Namen weglassen), wird der Palettendialog im Browserfenster zentriert und der Pfeil wird nicht angezeigt, da es kein Steuerelement gibt, auf das er zeigen könnte.

Subform Palette - Cut out

Es wurde ein Ausschnitt um das Ziel für Palettendialoge hinzugefügt, wenn die Überlagerung angezeigt wird.

Wenn Sie den Parameter `bShowOverlay` als `kTrue` an den Client-Befehl "subformpaletteshow" (`$clientcommand`) übergeben, dann wird, wenn er auf ein Steuerelement (oder ein Unterelement des Steuerelements) zeigt, ein Ausschnitt im Overlay um dieses Element gemacht, um es hervorzuheben.

Datum Parsing

Der Date Parser in Omnis versucht, eine String-Darstellung eines Datums oder eines Datums-/Uhrzeitwerts in ein Omnis-Datums-/Uhrzeitformat zu konvertieren. Das Parsen von Datumsstrings wurde verbessert und greift nun auf kurze Datums-/Uhrzeitformate zurück.

Das kurze Datums-/Zeitformat hängt vom Gebietsschema des Browsers ab, d.h. für GB-Englisch ist es D/M/y und für US-Englisch ist es M/D/y. Daher könnte ein Endbenutzer "3/5" in ein beliebiges Datums- oder Datums-/Uhrzeitfeld eingeben, und es könnte in "3. Mai 2024" oder "5. März 2024" umgewandelt werden.

Überschreiben der Browser History

Neue Form Methoden ermöglichen die Interaktion mit dem Browserverlauf und fügen Navigationsereignisse hinzu, die durch Klicken auf die Schaltflächen Zurück oder Vorwärts im Browser ausgelöst werden.

\$pushhistorystate(wRow, [bReplaceState=kFalse])

\$applynewhistory

erhält die Row "wRow" von \$pushhistorystate als Parameter.

JavaScript Remote Components

JS Entry Field Border Icons

\$bordericonstyle

\$setbordericonstyle

evBorderIconClicked

Field border icons können zu Eingabefeldern (sowie Comboboxen und Dropdown-Listen) hinzugefügt werden, indem die Eigenschaft `$bordericonstyle` verwendet wird, die nur gilt, wenn der Rahmenstil des Feldes in `$effect` auf `kJSborderPlain` eingestellt ist.

Name

Email

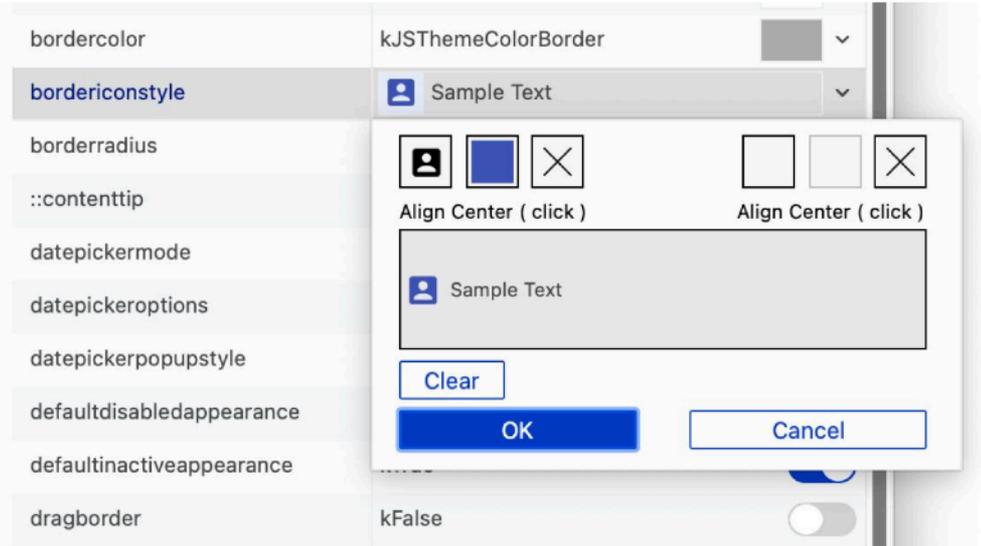
Phone

Address

JS Entry Field Border Icons

Die Property speichert die Einstellungen für die linken und rechten Icons, die Sie im Property Manager festlegen können.

Sie können das Ereignis `evBorderIconClicked` aktivieren, um die Rand Icon einzuschalten. Das Ereignis hat den Parameter `pLeftBorderIcon`. Wenn `true`, wurde das linke Rand Icon angeklickt, bei `false` wurde das rechte Rand Icon angeklickt.



JS Data Grid - \$filteronkeypress

Wenn **\$filteronkeypress** auf true gesetzt ist, wird das Grid bei jedem Tastendruck gefiltert (die Voreinstellung, um das Verhalten von früheren Versionen beizubehalten).

Wenn false, wird gefiltert, wenn die Enter Taste gedrückt wird oder wenn der Fokus das Feld verlässt.

JS Data Grid - Grid Cell Tooltips

Sie können nun Tooltips für einzelne Zellen in einer Data Grid erstellen.

Die vom Client ausgeführte Methode **\$gettooltip()** wird aufgerufen, wenn der Mauszeiger über eine Zelle bewegt wird. Der zurückgegebene String wird als Tooltip für die Zelle angezeigt. Die Methode hat die folgenden Parameter pHorzCell, pVertCell, pDataColumnName, um zu identifizieren, welche Zelle mit dem Mauszeiger überfahren wurde.

JS Data Grid - Drop events

Die Events `evDrop`, `evCanDrop` und `evWillDrop` haben einen neuen Parameter **pColNum**, um die Spaltennummer für die verworfenen Daten zu erhalten.

Das Data Grid Control hat einen neuen Parameter, **pColNum**, der die Spaltennummer der Zelle angibt, in der die Daten gelöscht werden sollen. Der Parameter ist verfügbar für die `evDrop`, `evCanDrop` und `evWillDrop` Ereignisse.

JS Data Grid - Pick Lists

Die Eigenschaft `$picklistrowheight` wurde dem Data Grid Control hinzugefügt, damit Sie die Zeilenhöhe in Pick Lists festlegen können.

Sie können diese Eigenschaft auf denselben Wert wie die Zeilenhöhe für das Data Grid setzen, z. B. 44, was die Standardzeilenhöhe des Grids ist.

Die Zeilenhöhe der Pick List wird nicht kleiner als die Schrifthöhe sein.

Datagrid - \$enablecell()

Die Methode **\$enablecell** wurde dem Data Grid hinzugefügt, um zu verhindern, dass bestimmte Zellen bearbeitbar sind.

Die Methode `$enablecell(pRow,pCol,pDataColumnName)` wird aufgerufen, bevor eine Zelle bearbeitbar gemacht wird, und gibt daher `kFalse` zurück, um die Bearbeitung für die Zelle zu deaktivieren.

Diese Methode kann sowohl vom Client als auch vom Server ausgeführt werden.

JS Data Grid - Date Picker

Wenn eine der Data Grid Eigenschaften `$editdatetext` oder `$columneditdatetext` wahr ist, wird die Datumsauswahl-Schaltfläche nicht mehr als separater Tabstopp behandelt. Sie können die Datumsauswahl mit der Tastenkombination Alt + Pfeil nach unten oder durch Anklicken der Schaltfläche öffnen.

Wenn Sie das alte Verhalten wünschen, bei dem die Datumsauswahl ein separater Tabstopp ist, können Sie die JavaScript-Variable 'alwaysTabToDatePicker' in der Datentabelle auf true setzen. Zum Beispiel:

```
Calculate IDataGrid as $cinst.$objs.datagrid
```

```
JavaScript:IDataGrid.alwaysTabToDatePicker = true;
```

Native List - \$extrascrollheight

Die Eigenschaft **\$extrascrollheight** gibt die zusätzliche Scrollhöhe an, die am Ende einer Native List hinzugefügt wird.

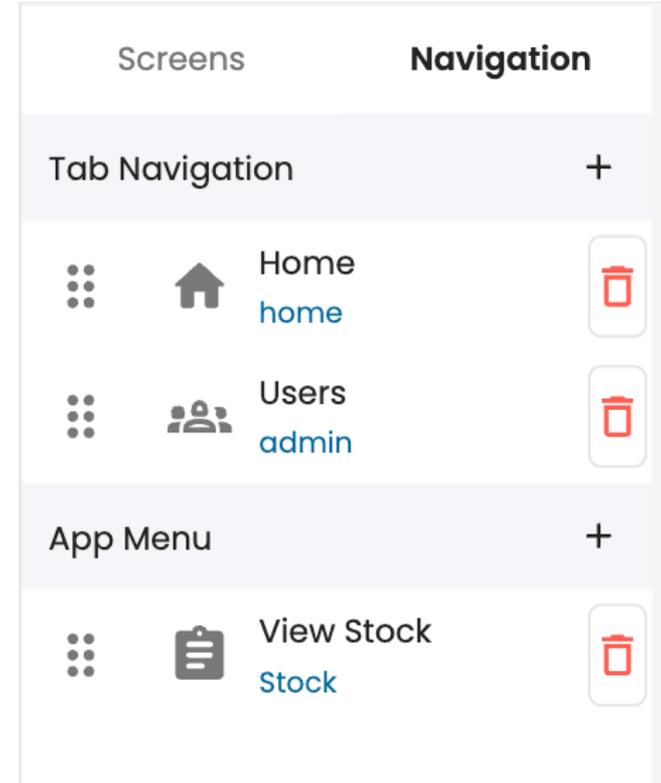
Dies kann z. B. nützlich sein, um Platz für eine schwebenden Button zu schaffen, der über einer nativen Liste angezeigt wird.

Native List - Buttons

Sie können jetzt Buttons zu den Gruppenüberschriften in nativen Listen hinzufügen. Außerdem können Schaltflächen-Accessoires jetzt eine Symbol-URL (oder Bild-URL) als Inhalt annehmen.

Sie können **\$groupaccessorytypecol** auf `kJSNativeListAccessoryTypeButton` setzen, um der Gruppenüberschrift einen Button hinzuzufügen (ein anderer Zubehörtyp wird derzeit nicht unterstützt).

\$groupaccessorycontentcol kann auf Text oder eine Icon/Bild-URL gesetzt werden, wobei das Bild an der URL mit der Endung `.png/.jpg/.jpeg/.svg` erkannt wird.



Native List - Buttons

Wenn eine Gruppenüberschrift angeklickt wird, wird `evClick` ausgelöst.

Der Parameter `pWhat` ist `kJSNativeListPartGroupAccessory`, und `pGroup` ist die Gruppennummer.

Scroll Shadows für Listen

Randschatten heben jetzt scrollbare Inhalte in Listen, Grids, Containern und anderen Steuerelementen hervor.

\$scrollshadowwidth

\$scrollshadowcolor

wList

wTest

wTest2

wTestFTP

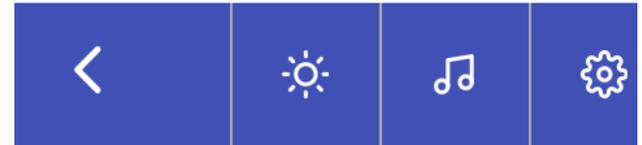
Toolbar Control Back Button

Sie können jetzt eine "Zurück"-Schaltfläche auf der linken Seite eines JS-Toolbar-Steuererelements hinzufügen, indem Sie **\$backbutton** auf `kTrue` setzt. In diesem Fall, wenn das Seitenmenü sichtbar ist (`$sidemenu` ist `kTrue`), wird das Hamburger-Menü durch den Zurück-Button ersetzt. Wenn der Zurück-Button sichtbar ist und angeklickt wird, wird das neue Ereignis `evBackClick` ausgelöst.

\$backbutton = `kTrue`

\$event

On `evBackClick`



JS Tab Bar Control - \$currenttab

Die **\$currenttab** Eigenschaft des Tab Bar-Steuerelements kann jetzt auf 0 gesetzt werden.

Das hat den Effekt, dass alle Registerkarten abgewählt werden.

Html Link Control - Link URL

Wenn die Eigenschaften \$linkurl, \$linkurlname oder \$linkedobject leer sind, ist die URL für das Html-Link-Steuerelement standardmäßig der Text aus \$text oder \$dataname, vorausgesetzt der Text hat ein gültiges URL-Format hat (http://... oder https://...).

Events

Das Standard-Ereignisverhalten für evClick im Html-Link-Control kann jetzt in der \$event-Methode mit dem Befehl Quit event handler (Discard event) überschrieben werden.

Date Pickers

Datumsauswahl-Popups verwenden jetzt den Standard des Client-Gebietsschemas für den ersten Tag der Woche.

Datumsauswahlfelder für Eingabefelder und Datengitter (nicht das Steuerelement für die Datumsauswahl) verwenden jetzt den standardmäßigen ersten Wochentag aus dem Gebietsschema des Client-Browsers, wenn sie sich in der Kalenderansicht befinden.

In früheren Versionen war der erste Tag der Woche standardmäßig der Sonntag.

Navigation Menu Object

Zwei neue Eigenschaften **\$::iconcolor** und **\$cascadeiconcolor** wurden zu dem Navigationsmenü Control hinzugefügt, mit denen Sie die Farbe der SVG-Symbole im Steuerelement festlegen können. Diese Verbesserung gilt für die JavaScript (Remote form) und Window class Nav Menu Controls.

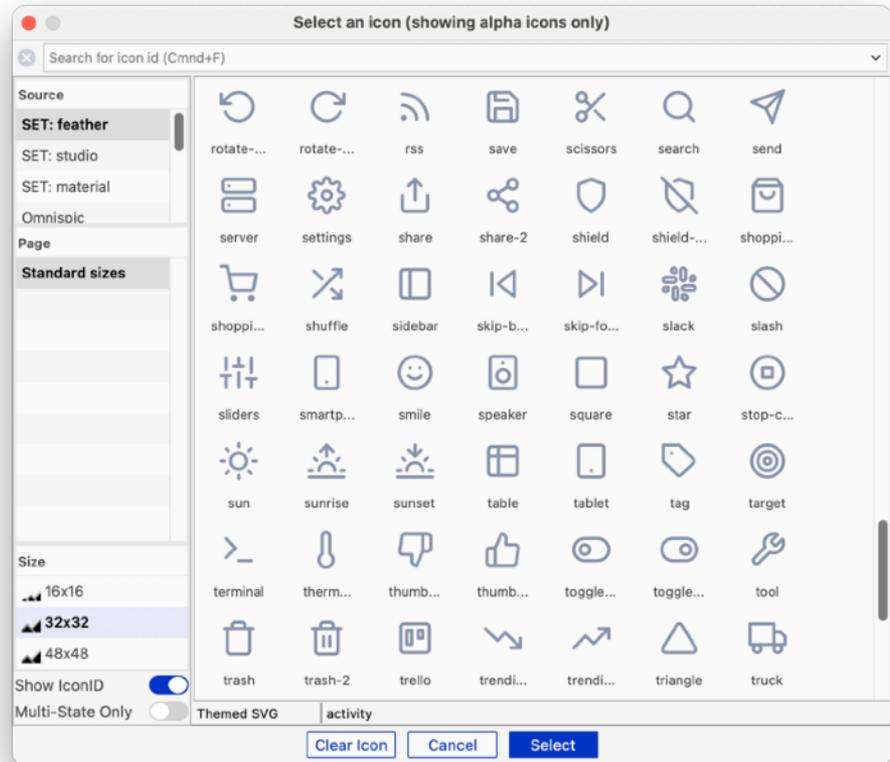
Die Eigenschaft **\$::iconcolor** legt die Farbe des SVG-Symbols auf der obersten Menüebene fest angegeben in \$verticalcascadeiconid.

Die Eigenschaft **\$cascadeiconcolor** legt die Farbe der SVG-Symbole in den kaskadierten Menüs fest, die in \$horizontalcascadeiconid, \$closeboxiconid und \$hotcloseboxiconid angegeben ist.

In der JS-Client-Komponente entspricht kColorDefault den Standardregeln für Textfarben, während in der Window-Klassensteuerung kColorDefault bewirkt, dass die Farben der Symbole der Textfarbe folgen.

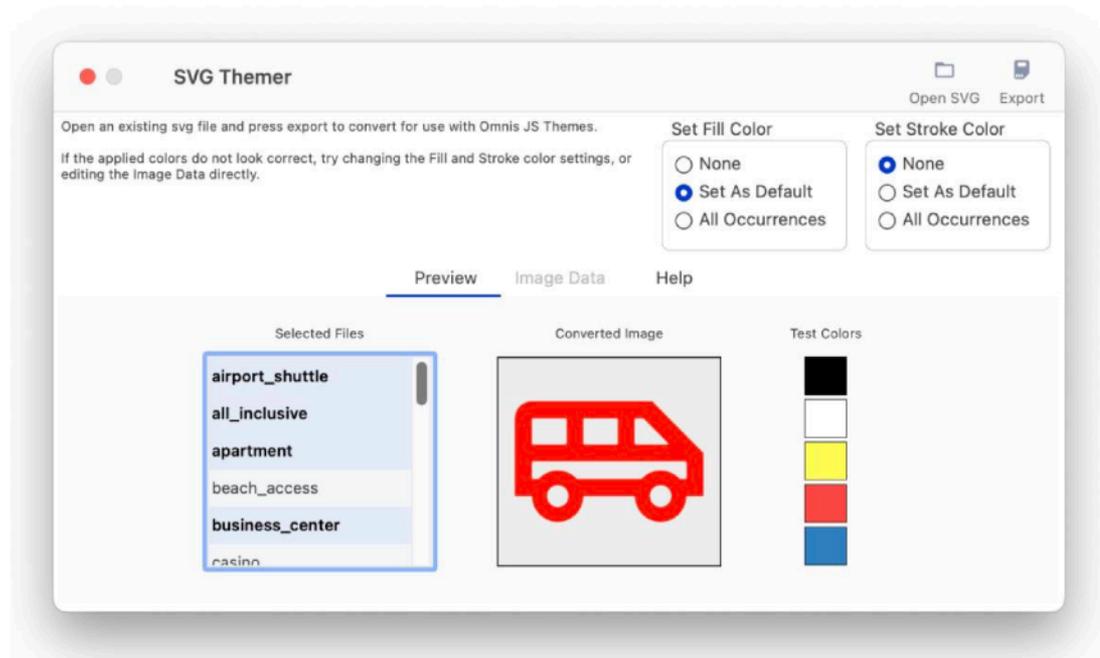
Component Icons new Feather Icon Set

Fügen Sie "feather" zu der Eigenschaft **\$iconset** Ihrer Library hinzu.



SVG Themer Tool Stapelverarbeitung

Mit dem Werkzeug SVG Themer können Sie mehrere SVG-Symbole "themen" und exportieren.



**Bereitstellen Ihrer
Apps**

Server Konfiguration - discardRequestsContaining

Das Element **discardRequestsContaining** wurde dem Abschnitt "server" der Omnis-Konfigurationsdatei hinzugefügt.

discardRequestsContaining ist ein Array von Zeichenketten, wobei jede Zeichenkette mit einer angeforderten URL vom Omnis-Server abgeglichen wird und bei Übereinstimmung die Anfrage verworfen wird. Standardmäßig enthält das Array die Zeichenkette:

```
"discardRequestsContaining": [  
  "../"  
]
```

Dies bedeutet, dass alle vom Omnis Server empfangenen Anfragen, die irgendwo im URL-Pfad ../ enthalten, verworfen werden und ein Fehler an den Client zurückgegeben wird.

Firstruninstall

Es wurde eine versteckte Option hinzugefügt, die das Kopieren des firstruninstall-Ordners verhindert, wenn der Datenordner unter Windows oder macOS bereits existiert.

Omnis prüft nun auf eine Datei namens "**disable_fri.txt**" im Stammverzeichnis des Datenordners. Wenn diese Datei vorhanden ist, kopiert Omnis keine Dateien aus dem Ordner "firstruninstall". Omnis liest nicht den Inhalt der Datei, sondern prüft nur, ob "**disable_fri.txt**" existiert.

Window Komponenten

Entry Fields und Combo Boxen

Die Eigenschaften **\$labelcolor**, **\$labelfontsize** und **\$labelhottextcolor** wurden zu Eingabefeldern und Combo Boxen hinzugefügt, um die Textfarbe und -größe der Floating Content Tips zu steuern.

Standardmäßig sind diese einer Schriftgröße von 0 und kColorDefault zugeordnet, um das Verhalten früherer Versionen beizubehalten, bei denen die Farbe des Beschriftungstextes dieselbe ist wie der Feldtext und die Schriftgröße 80 % des Bearbeitungsfeldes beträgt.

Masked Entry Feld - \$passwordchar

Die Eigenschaft **\$passwordchar** wurde zur Masked Entry Field hinzugefügt.

Sie können '*' in \$passwordchar in einem Masked Entry Field verwenden, um Zeichen durch Sternchen zu ersetzen und den Feldinhalt zu verbergen.

Pushbuttons

Der Stil von Push Buttons, die auf `kSystemButton` in `$buttonstyle` gesetzt sind, hat sich unter macOS geändert.

Für Überschriftenlisten und Tree Listen ist `$buttonstyle` der Stil des Headers, entweder `kSystemButton` oder `kUserButton`. Unter macOS verwendet der `kSystemButton`-Schaltflächenstil nun den Stil eines systemeigenen Buttons mit einer flexiblen Höhe. Wenn die Höhe die eines normalen Buttons überschreitet, behält die Schaltfläche den gleichen Stil.

In früheren Versionen wurde der quadratische Systemstil verwendet, wenn die Höhe größer war als die einer Schaltfläche mit normaler Größe. Um das alte Verhalten zu aktivieren, setzen Sie die neue Root Eigenschaft `$squaresystembuttons` auf `kTrue`.

Headed List - \$showcolumnlinesifempty

Eine neue Eigenschaft \$showcolumnlinesifempty wurde zu Headed Lists hinzugefügt, um Spaltenzeilen anzuzeigen, wenn die Liste leer ist.

Dies gilt nur, wenn \$showcolumnlines ebenfalls kTrue ist.

oBrowser - \$allowsrollzoom

Mit der Eigenschaft \$allowsrollzoom kann die Scroll-Zoom-Funktionalität innerhalb des OBrowser Controls umgeschaltet werden.

Wenn sie auf kTrue (Standard) gesetzt ist, kann der Endbenutzer in den Browserinhalt hinein- und herauszoomen, indem er die Strg-Taste gedrückt hält, während er mit dem Mausrad scrollt.

Wenn \$allowsrollzoom auf false gesetzt ist, ist diese Zoom-Funktionalität deaktiviert.

Border Icons

Border Icons reagieren jetzt auf Klicks. Dies gilt für Steuerelemente, die \$bordericonstyle unterstützen, einschließlich Bearbeitungsfelder, Token-Editierfelder, Combo Boxen und Drop Lists.

```
On evBorderIconClicked
```

```
  If pLeftBorderIcon
```

```
    Popup menu MyOptionsMenu
```

```
  Else
```

```
    Send to trace log Right Icon Clicked
```

```
  End If
```

Methoden Editor

Methoden Lesezeichen

Fügen Sie Methoden oder Zeilen Lesezeichen hinzu, um wichtigen Code einfach zu markieren und zu finden.

The screenshot shows an IDE interface with a code editor on the left and a 'Bookmarks' dialog box on the right. The code editor displays a list of class methods on the left sidebar, including 'Sconstruct', 'Sdestruct', 'Scompleted', 'Scancelled', 'Sstart', 'Scancelreque', 'updateKeySize', 'updateIVSize', 'Sgetkeyoriv', 'Sbeginanimal', 'Sbringtofront', 'Scancelclose', 'Sclose', 'Scommitanin', 'Scontrol', 'Sevent', 'Smaximize', 'Sminimize', and 'Spreviewurcl'. The main code editor shows a list of lines of code, with line 29 highlighted in blue. The 'Bookmarks' dialog box is open, showing a table of bookmarks. The table has three columns: 'Bookmark', 'Project Library', and 'Location'. The first row is 'start method' with project 'crypto_ow3.lbs' and location 'wCrypto/\$start'. The second row is 'error' with project 'crypto_ow3.lbs' and location 'wCrypto/\$start Line 29'. Below the table is an 'Open Bookmark' button, a search field, and a 'Description' field. There are also 'Delete', 'Update', and 'Close' buttons. At the bottom of the dialog are several colored squares representing different bookmark colors.

Bookmark	Project Library	Location
start method	crypto_ow3.lbs	wCrypto/\$start
error	crypto_ow3.lbs	wCrypto/\$start Line 29

Libraries und Klassen

Exportieren von Libraries nach JSON

Die folgenden VCS-bezogenen Eigenschaften werden vom JSON-Export ausgeschlossen, wenn die `fullexportimport` Option (Teil der Stammeinstellung `$exportimportjsonoptions`) `kFalse` ist:

- Library Properties:

`$vcsbuilddate`, `$vcsbuildersname`, `$vcsbuildnotes`

- Klassen Properties:

`$vcsrevision`, `$showascheckedout`

File Klassen

Der File Klassen Editor enthält jetzt für jedes Feld in der Dateiklasse eine Benutzerinformation, die zum Speichern von Daten beliebigen Typs für das Feld verwendet werden kann.

Dies entspricht `fileclass.field.$userinfo`.

Description	<input type="text" value="my primary key"/>
UserInfo	<input type="text" value="use this for any additional information"/>

The background is a solid blue color with several diagonal stripes of varying shades of blue running from the top-left to the bottom-right. A single, thin white line is positioned diagonally in the upper-left quadrant, parallel to the stripes.

Reports

Enterable Report Fields

Setzen Sie die Eigenschaft **\$enterable** eines Feldes auf `kTrue`, damit Benutzer Text in dem PDF-Report eingeben und speichern können.

Dies funktioniert nur mit PDFs, die mit dem Omnis PDF Device erzeugt wurden.

\$setdocinfo

Mit der Methode **\$setdocinfo** können Sie nun den Author und Ersteller der ausgegebenen PDF-Datei festzulegen.

Dies funktioniert nur bei PDFs, die mit dem Omnis PDF Device erzeugt wurden.

```
$setdocinfo(cAuthor[,cTitle="",cSubject="",cKeywords="",cLanguage='en-GB',cProducer='Omnis Studio PDF Device',cCreator='Omnis Studio'])
```

cProducer

is the software that generated the PDF, the default is Omnis Studio PDF Device.

cCreator

is the software that created the content of the PDF, the default is Omnis Studio.

\$embedfile() und \$embeddata()

Omnis PDF Device wurde um zwei neue Funktionen erweitert, mit denen Sie Dateien und Daten in PDF-Dateien einbetten können.

Dies funktioniert nur mit PDFs, die mit dem Omnis PDF Device erzeugt wurden.

Omnis VCS API

VCS API - \$getapiobject()

Die Art und Weise, wie Sie auf die VCS-API zugreifen, hat sich geändert. Sie sollten nun die Methode **\$getapiobject** verwenden, um auf die Methoden der VCS-API zuzugreifen.

Um die VCS-API-Methoden aufzurufen, müssen Sie zunächst die Methode **\$getapiobject()** verwenden, um eine Objektreferenz auf die VCS-API zurückzugeben, zum Beispiel:

```
Do $root.$modes.$getapiobject('VCS') Gibt iAPIObjRef zurück
```

Sobald die Objektreferenz erzeugt wurde, können alle Aufrufe der API mit dieser Referenz erfolgen, was eine einfachere Syntax für Methodenaufrufe ermöglicht.

Um beispielsweise die Methode \$logon aufzurufen, können Sie verwenden:

```
Do iAPIObjRef.$logon(cUSERNAME, cPASSWORD, nTokenTime, cToken, cErrors,  
rSessionOrSessionPoolRef, cDBName) Returns bStatus
```

VCS API - \$buildProject

Die VCS-API-Methode \$buildProject erlaubt es nun, Klassen zu sperren oder vom Build auszuschließen.

Die Methode \$buildProject hat zwei neue Listenparameter, **excludeClassList** und **lockClassesList**.

Beide werden mit **cClassName** und **cClassType** (z.B. wTest, kWindow) definiert und enthalten eine Liste von Klassen, die entweder vom Build ausgeschlossen oder gesperrt werden sollen.

OW3 Worker Objekte

LDAP Worker Objekt

Die \$init-Methode im LDAP-Worker kann nun die Parameter cUser und cPassword entweder als Character oder Binary akzeptieren; diese Verbesserung dient der Anpassung an Namen mit Umlauten.

Wenn Character verwendet wird, sendet Omnis cUser und cPassword als UTF-8. Bei Verwendung von Binary sendet Omnis cUser und cPassword in der vom Server geforderten Kodierung, z. B. CP1252, die von einigen Servern verlangt wird.

HTTP Worker - Einsatz von KI in Omnis

Mit dem Omnis HTTP-Worker können Sie auf KI-Modelle über deren APIs zugreifen.

Eine Beispielanwendung namens HTTP AI wurde dem Abschnitt Samples des Hubs im Studio Browser hinzugefügt, um zu demonstrieren, wie Sie auf verschiedene AI-Modelle zugreifen können.

HASH Worker - \$initsignature

Die Methode \$initsignature hat einen neuen optionalen Parameter iSignatureType, mit dem Sie den RSA- oder ECDSA-Signaturtyp auswählen können.

Darüber hinaus ist der Parameter bBlind jetzt standardmäßig auf kTrue gesetzt.

OAuth2 Worker - \$pkceentropy

Die Eigenschaft **\$pkceentropy** wurde dem OAuth2 Worker hinzugefügt, um die Länge des Code Verifiers für PKCE zu kontrollieren.

Die Eigenschaft **\$pkceentropy** steuert die Länge der Codeüberprüfung für PKCE und nimmt einen ganzzahligen Wert zwischen 43 (Standard) und 128 an.

Funktionen

Searching Lists - bSetCurLineWhenNotSelecting

```
$search(calculation [,bFromStart=kTrue, bOnlySelected=kFalse,  
bSelectMatches=kTrue, bDeselectNonMatches=kTrue,  
bSetCurLineWhenNotSelecting=kTrue])
```

Wenn **bSetCurLineWhenNotSelecting** als false übergeben wird (der Standardwert ist kTrue) und die Optionen select/deselect false sind, wird die Suche abgeschlossen, die gefundene Zeilennummer wird zurückgegeben, aber die gefundene Zeile wird nicht ausgewählt.

FileOps

\$copy() and **\$move()**

- funktioniert für Dateien und Ordner.
- neue Parameter zum Erstellen fehlender Pfade, zum Ersetzen vorhandener Dateien und zum Zusammenführen von Inhalten enthalten.

\$joinpath(cSegment, ...)

- Verbindet alle angegebenen Segmente unter Verwendung des plattformspezifischen Trennzeichens.

FileOps.\$writefile - bCreateParentDirectories

Ein neuer Parameter **bCreateParentDirectories** wurde der Funktion FileOps.\$writefile hinzugefügt, um die Verzeichnisse im Pfad zu erstellen.

```
FileOps.$writefile(cFilePath,vVariable [,iEnc=kUniTypeUTF8, bBOM=kTrue,  
bReplace=kTrue, bCreateParentDirectories=kFalse])
```

bool()

Es gibt eine neue Funktion **bool()**, die einen Wert in einen Boolean umwandelt.

Do bool(#NULL) ## würde kFalse zurückgeben

If kTrue | bool(#NULL) ## Würde sich zu wahr auflösen

...

End If

If bool(lObjRef) ## Würde zu true aufgelöst, wenn lObjRef auf eine Instanz gesetzt wurde, und false, wenn nicht gesetzt

...

End If

idletime()

Es gibt eine neue Funktion **idletime()**, die die verstrichene Zeit in Millisekunden zurückgibt, seit die Tastatur oder Maus benutzt wurde.

coalesce() und coalesceempty()

coalesce()

- Gibt den ersten Nicht-NULL-Parameter aus einer Liste von Werten zurück.

\$coalesceempty()

- Gibt den ersten nicht-NULL und nicht-leeren Parameter aus einer Liste von Werten zurück.

OREGEX

OREGEX.\$replace(cSource, cTarget, cReplace [, iFlags, &cErrorText]))

- Sucht nach einer Übereinstimmung in cSource mit dem Muster in cTarget und ersetzt die übereinstimmende Teilzeichenkette durch cReplace.

OREGEX.\$replaceall(cSource, cTarget, cReplace [, iFlags, &cErrorText]))

- Gibt den ersten nicht-NULL und nicht-leeren Parameter aus einer Liste von Werten zurück.

Die Flags können addiert werden, z.B.

kORegExCaseInsensitive+kORegExMatchNotNull.

idletime()

Gibt die Anzahl der Millisekunden zurück, die seit der letzten Benutzung der Maus oder der Tastatur verstrichen sind.

JSON Objekt Methoden

Einige der JSON-Objektmethoden haben einen neuen Parameter, um sicherzustellen, dass fehlende Objekte automatisch erstellt werden.

`$setstring()`, `$setbool()`, `$setinteger()`, `$setfloat()`, und `$setobject()`.

syntax for `$setstring()` ist jetzt:

- `$setstring(cMember,cString[,bAddMissingMembers=kFalse])`

Setzt das angegebene Mitglied auf eine JSON-Zeichenkette mit dem Wert `cString`. Wenn `bAddMissingMembers` als `kTrue` übergeben wird, werden die fehlenden Mitglieder (nur Objekte), Standardwert ist `kFalse`. Gibt bei Erfolg `true` zurück.

rcredit

Neue statische Methoden in der Komponente **rcredit** (nur Windows) geben verschiedene Dateiinformationen zurück.

- **\$getapplicationmanifest**(cFileName)
Gibt das Manifest der Exe/Dll in cFileName zurück.
- **\$getfileversion**(cFileName)
Gibt die Dateiversion der Exe/Dll in cFileName zurück.
- **\$getproductversion**(cFileName)
Gibt die Produktversion aus der Exe/Dll in cFileName zurück.
- **\$getresourcestring**(cFileName, iKey)
Liefert den Resource-String mit dem Schlüssel iKey aus der Exe/Dll in cFileName.
- **\$getversionstring**(cFileName, cKey)
Gibt die Versionszeichenfolge mit dem Schlüssel cKey aus der Exe/Dll in cFileName zurück.

**Es gibt viele weitere
kleine Verbesserungen ..**