

The logo for the Omnis Developer Conference 2026. It features a stylized 'O' with an orange dot inside, followed by 'DC' in blue, and '2026' in orange. The entire logo is centered within a white circle.

ODC 2026

Omnis Developer Conference

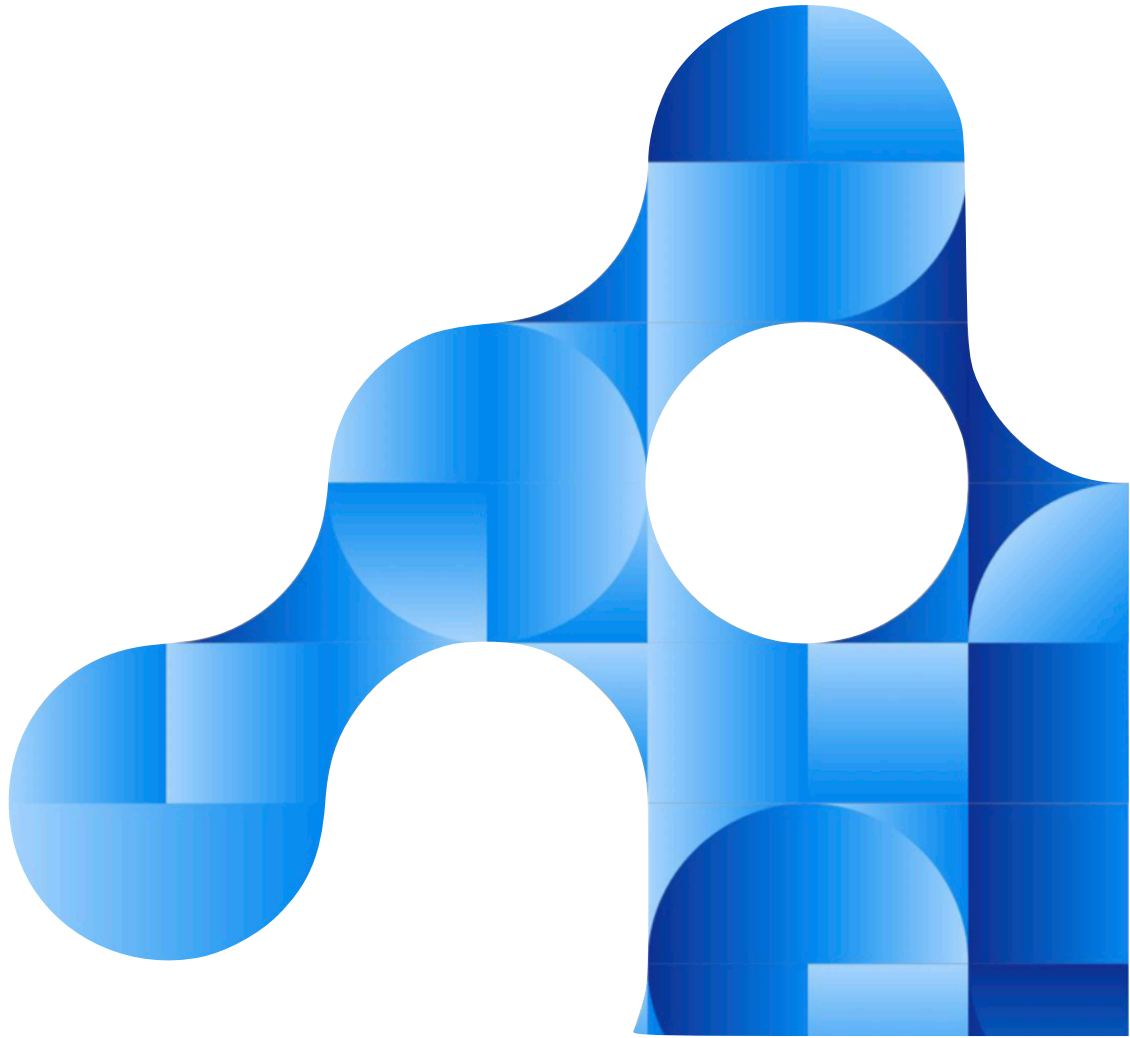
# Omnis Studio 12

## What's New. What Matters.

Andreas H. Pfeiffer  
Sen. Consultant

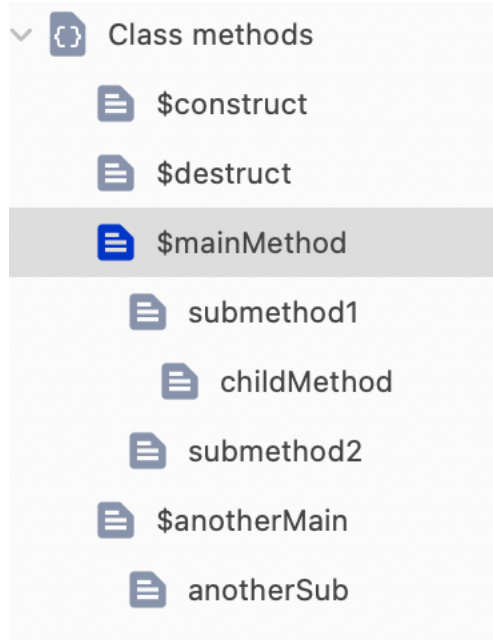


# Developer Productivity Boosts



# Method Indentation

Improving readability and consistency

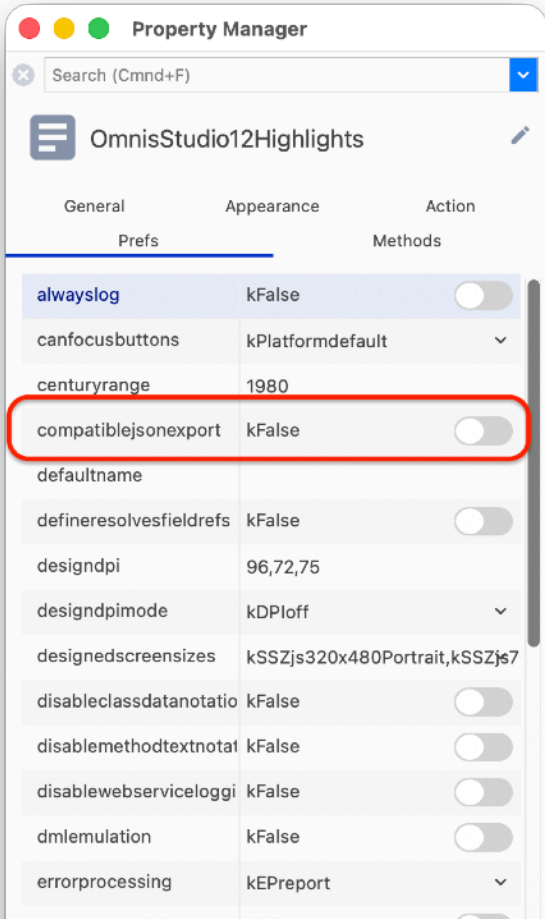


- Visually group related methods
- Organise large classes more clearly
- JSON export preserves indentation
- Ctrl/Cmnd [ or ] to indent left or right or right-click selected methods

# JSON Export Improvements

Great advantage for teams using version control

- Optional removal of method/variable order metadata
- Reduced Git conflicts and Cleaner diffs



# Touch-Friendly Desktop Apps with Gesture Events

Handle pinch, swipe, and  
more with ease



# Next-Generation Desktop Interaction

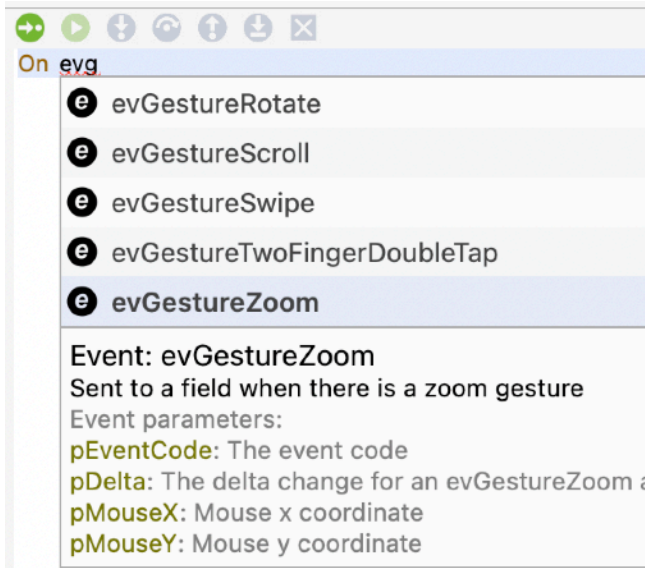
Designed for touchscreens and modern trackpads



- Scroll, Swipe, Rotate
- Pinch-to-zoom
- Two-finger double-tap

# Using Gestures in Desktop Apps

Build apps that feel native on modern hardware



- Enabled via new \$touchgestures root, library or field property
- With full event parameters: deltas, anchor points, direction, magnification.
- Great for image viewers, dashboards, maps, and custom UI
- No custom platform code required

# JavaScript Remote Forms Major Enhancements

More interactive, modern  
web-style interfaces  
More control over layout,  
behaviour, and improved UX



# Subform Palettes

.. can now target specific rows or cells

2	Borussia Dortmu...	Gelb/Schwarz	Niko Kovač
16	Borussia Mönch...	Schwarz/Weiß	Gerardo Seoane
4	Eintracht Frankfurt	Schwarz/Rot	Dino Toppmöller
15	FC Augsburg	Rot/Weiß	Jess Thorup
1	FC Bayern Münc...	Rot	Vincent Kompany
17	FC St. Pauli	Braun/Weiß	Alexander Bless
18	Hamburger SV	Blau/Weiß/Schw...	Steffen Baumgart
9	RB Leipzig	Rot/Weiß	Ole Werner

**Selected Team**

grid:Verein=FC St. Pauli

# Promises for subforms

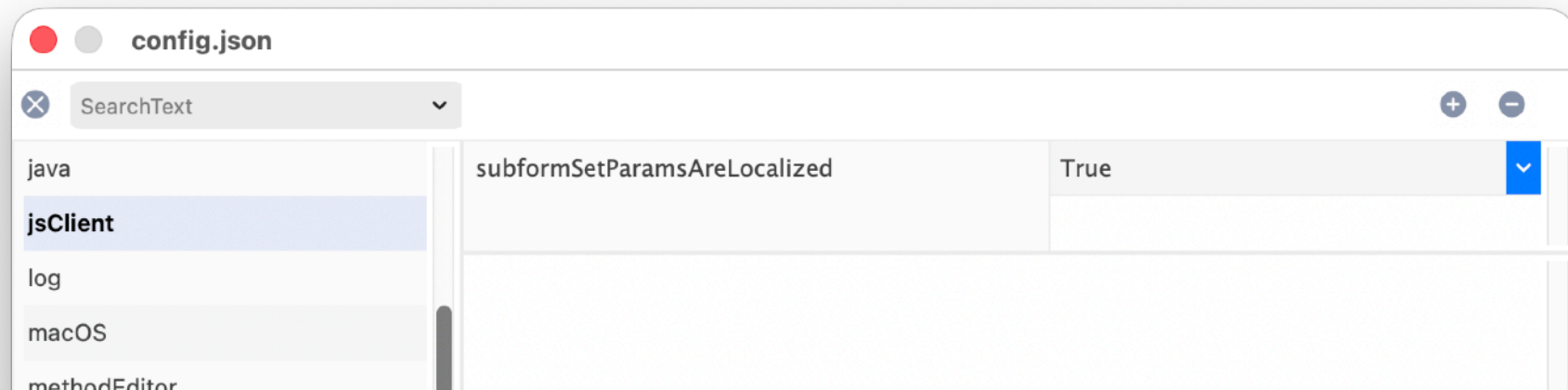
The promise returns from the subformdialog and paletteshow client commands gives now structured objects (form & errorText)

```
Do $cinst.$clientcommand("subformdialogshow",row(...)) Returns lPromise
```

```
JavaScript:lPromise.then( (result) => {  
JavaScript:    lForm = result.form;  
Do lForm.$myMethod()  
JavaScript:});
```

# Subform Sets

.. can now use the server's localised function and decimal separators



# Data Grid

.. supports cell-level drag-and-drop

The image shows a configuration panel for the 'celldraganddrop' property. The panel includes a dropdown menu with the selected value '[kJSDDataDragCells,kJSDData]' and two toggle switches for 'kJSDDataDragCells' and 'kJSDDataDropCells', both of which are currently turned on. To the right, a list of event parameters is displayed, including 'pColNum', 'pDragDataType', 'pDragField', 'pDragType', 'pDragValue', 'pDropColName', 'pDropId', 'pDropRow', and 'pEventCode'.

5	On evDrop	Task	pColNum	3
		Class	pDragDataType	21
		Instance	pDragField	grid
		Local	pDragType	1
		Parameters	pDragValue	Bayer 04 Leverkus
		Event param	pDropColName	Trikotfarbe
			pDropId	2
			pDropRow	(0 bytes)
			pEventCode	18

# Tree List

✓  Car Parts

✓  Brakes

 Discs

 Pads

 Shoes

.. gets better spacing and icon control

`$::iconcolor`

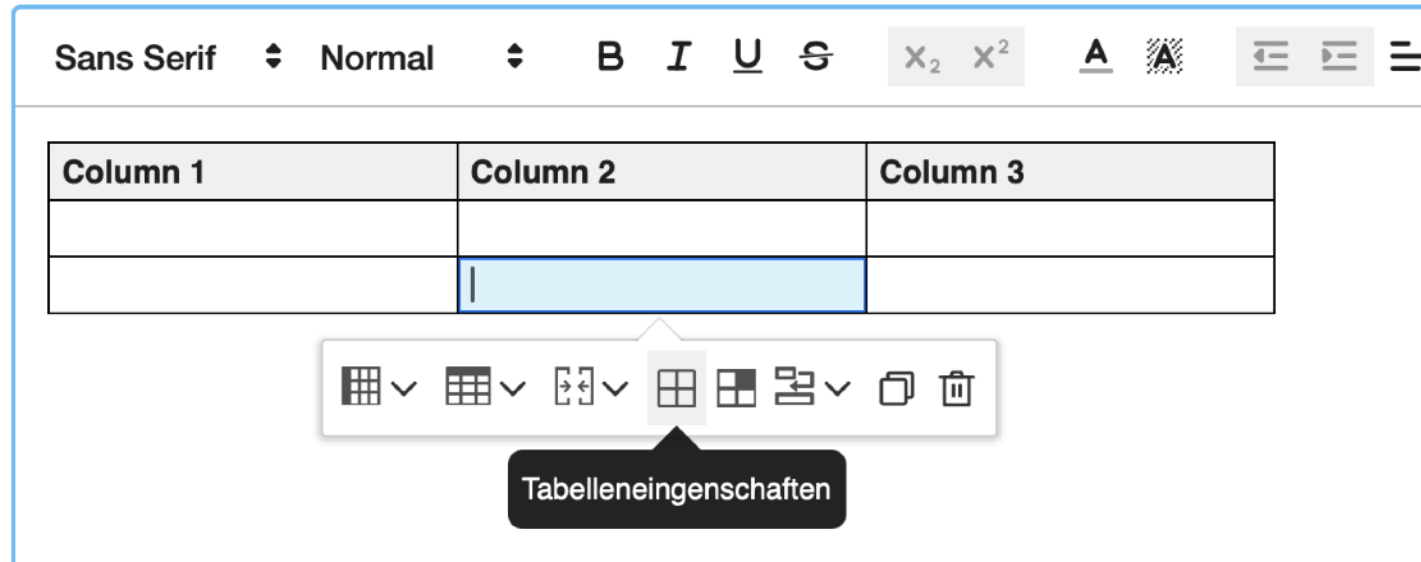
Allows to set individual colors to SVG icons. 0 uses text color.

`$icontextspacing`

Controls the distance between the icon and the node text, default is 4 pixels.

# Rich Text Editor

.. adds table support



The screenshot displays a Rich Text Editor interface. At the top, there is a toolbar with various formatting options: font family (Sans Serif), font style (Normal), bold (B), italic (I), underline (U), strikethrough (ABC), subscript (x<sub>2</sub>), superscript (x<sup>2</sup>), text color (A), background color (A with dots), and alignment (left, right, center). Below the toolbar is a table with three columns labeled "Column 1", "Column 2", and "Column 3". The table has three rows. The second row is empty, and the third row has a light blue highlight on the "Column 2" cell. A context menu is open below the table, showing icons for table operations: grid, insert table, table border, table properties, table merge, table split, copy, and delete. The "Table Properties" icon is highlighted, and a callout box points to it with the text "Tabelleneigenschaften".

Column 1	Column 2	Column 3

Tabelleneigenschaften

# Markdown Control

.. gains scroll-to-position and scroll shadows

## Markdown Heading

[You can change the colour of links](#)

---


Some `<p>inline code</p>` with a horizontal rule above and a table below.

Table Col 1	Table Col 2
Table Data	The table border color and width properties can be

### MarkdownParser

Markdown Control

Search

scrollshadowcolor RGBA(168,168,168,5)  ▾

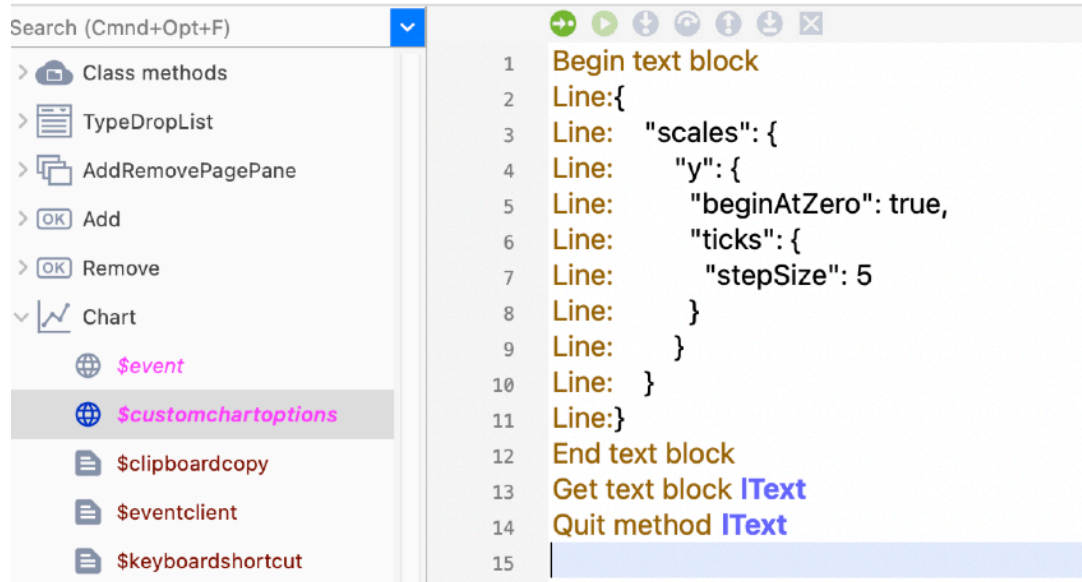
scrollshadowwidth 0

scrolltoposition kJSMardownMaintainScroll ▾

- kJSMardownMaintainScroll**
- kJSMardownScrollToTop
- kJSMardownScrollToBottom

# Chart Control

.. accepts custom ChartJS options

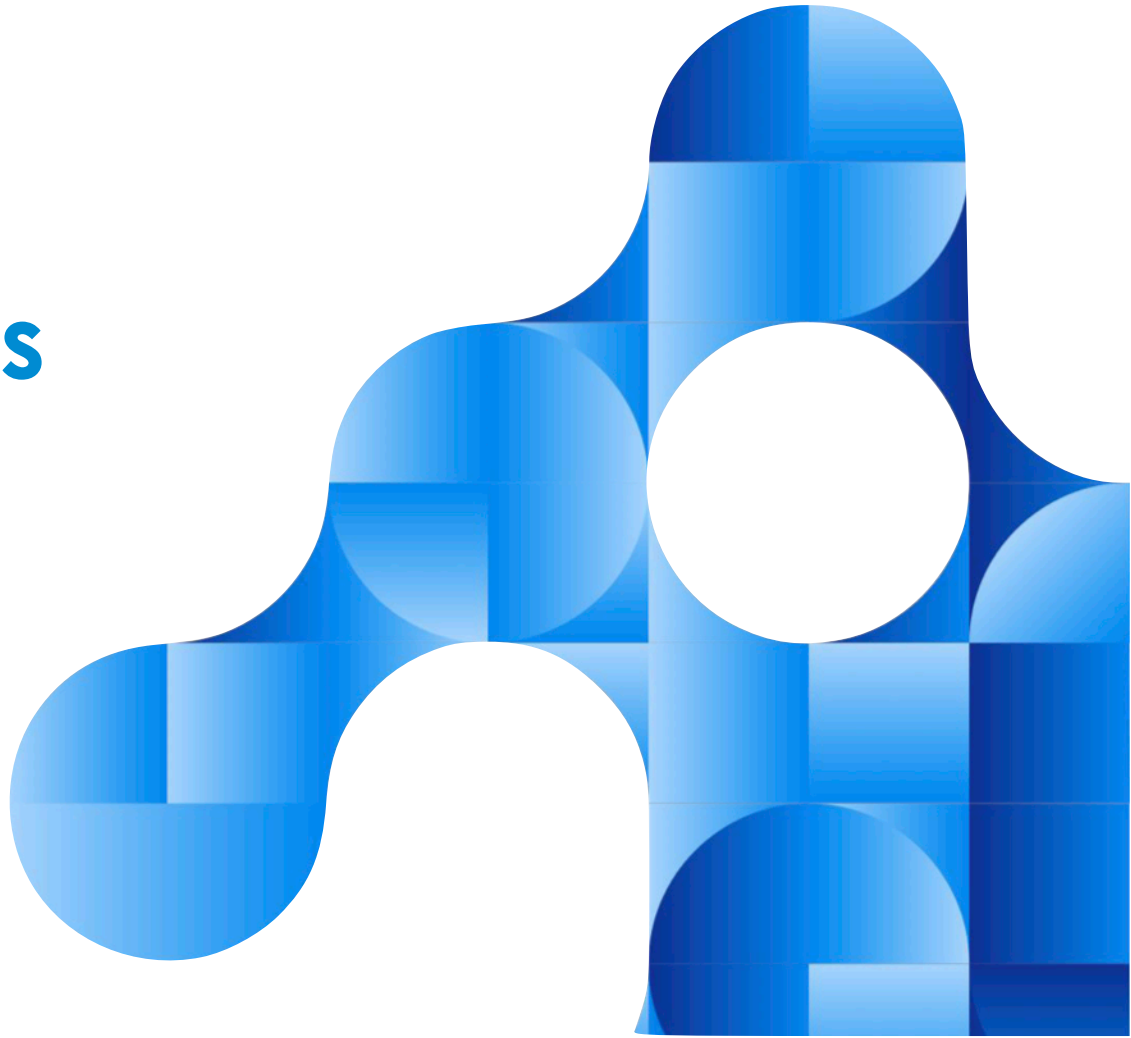


```
Search (Cmnd+Opt+F)
> Class methods
> TypeDropList
> AddRemovePagePane
> Add
> Remove
> Chart
  $event
  $customchartoptions
  $clipboardcopy
  $eventclient
  $keyboardshortcut

1 Begin text block
2 Line: {
3 Line: "scales": {
4 Line: "y": {
5 Line: "beginAtZero": true,
6 Line: "ticks": {
7 Line: "stepSize": 5
8 Line: }
9 Line: }
10 Line: }
11 Line: }
12 End text block
13 Get text block IText
14 Quit method IText
15
```

# Powerful OW3 Worker Enhancements

New capabilities in the  
OW3 Workers providing  
modern API integration





# HTTP streaming

Stream content rather than waiting for the whole response

- Avoids waiting for the whole response
- Supports SSE events and NDJSON
- **`$streamresponse=kTrue`**  
Data will be sent to `$ondata` immediately.
- **`$ondata`**  
Receives a row as its single parameter, which varies based on the Content-Type of the response.
- **`$minstreamdelay`**  
the minimum number of milliseconds to wait between calls to `$ondata`

# OpenID Connect

## Support in OAuth2 Worker

- Providing standardised, secure, and efficient authentication

- **\$openidtoken**

If your authorization endpoint supports OpenID Connect, and you include "openid" in the scope, the \$openidtoken property will be populated with the OpenID token after calling \$authorize()



# SMTP Worker

Advanced SMTP capabilities for modern email

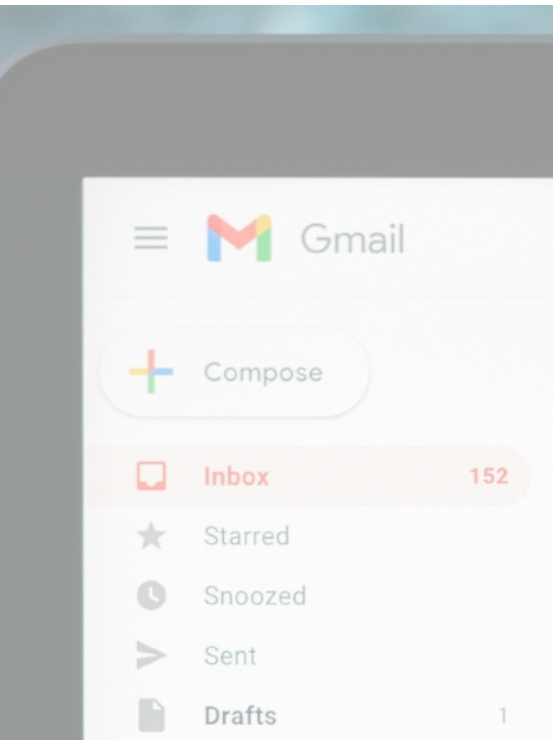


- Now supports body-part headers, providing support for rich, multimedia email communication
- The MIMEList sent as vContent in the \$init() method for the OW3 SMTP Worker now supports a tenth column bodypartheaders of type Row, representing the Body Part Headers.

# New raw MIME message creation

Advanced email composition for third-party integrations

- Integrate with third-party APIs such as provided by Gmail
- `OW3.$createmimemessage()`  
Allows to create a raw MIME email message to then send.



# macOS Sequoia support and Native AI Tools

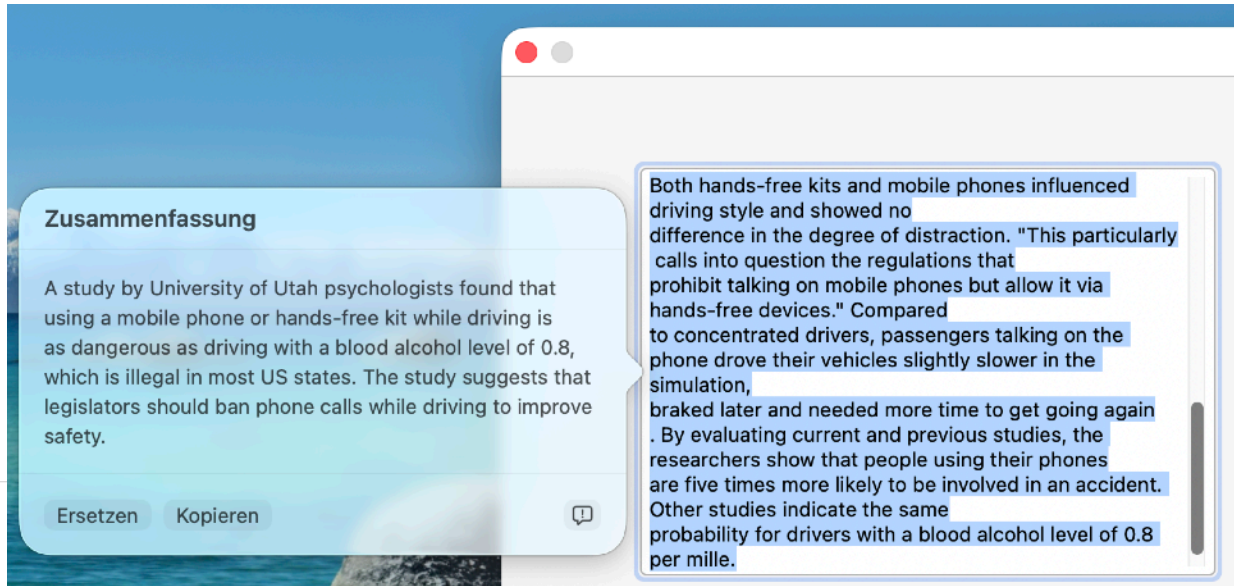
Develop and deploy on  
Apple's latest macOS and  
use AI in development and  
end user apps.



# System-Level AI Writing APIs

Built-in AI text capabilities available in every edit field

- System-level AI writing tools integrated into edit fields, with rewrite, summarise etc



# System-Level AI Writing APIs

Bring Apple's built-in AI tools to your application workflows

Enter Prompt

motor sport bike making wheelie through  
a fire wall

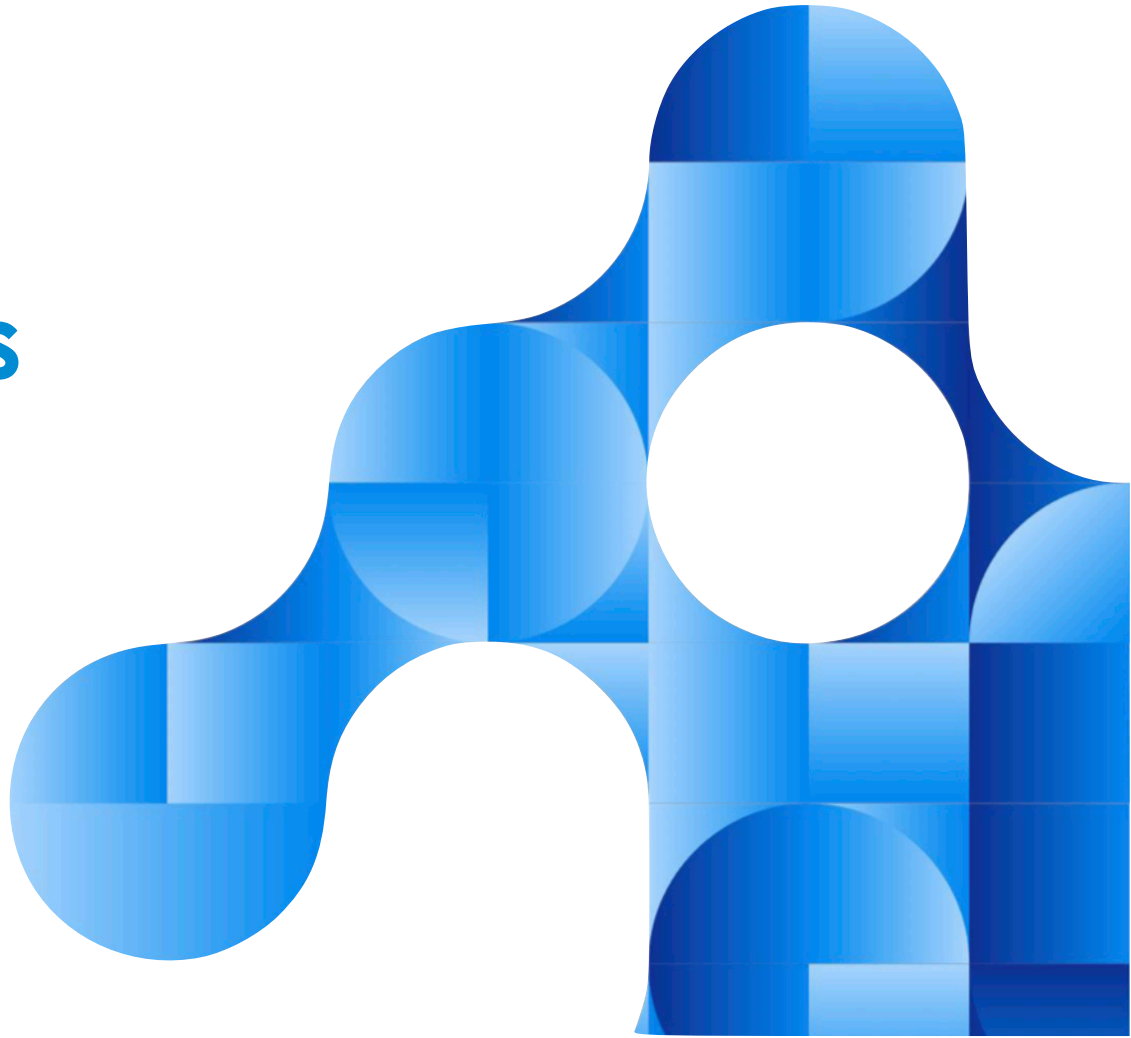
generate image



- Image Playground support for AI-generated images

# Desktop Component Improvements

Improved behaviour and  
modernized functionality



# OBrowser

Cursor tracking and component layering control

You can set the **\$enablecursortracking** property to kTrue to allow embedded web content to control cursor updates via system cursor tracking. For example, to allow CSS managed cursor state in the browser.

The **\$disablecomponentorder** property has been added to the OBrowser window component to allow other window controls to be placed over it by preventing the OBrowser component coming to the front when it gets the focus.

```
/* general cursors */
.auto      { cursor: auto; }
.default   { cursor: default; }
.none      { cursor: none; }

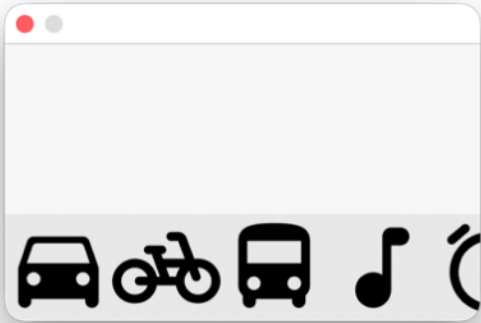
/* link and status cursors */
context-menu { cursor: context-menu; }
help         { cursor: help; }
pointer      { cursor: pointer; }
progress     { cursor: progress; }
wait        { cursor: wait; }

selection cursors */
all          { cursor: cell; }
crosshair    { cursor: crosshair; }
text         { cursor: text; }
vertical-text { cursor: vertical-text; }

drag-and-drop cursors */
alias        { cursor: alias; }
copy         { cursor: copy; }
move         { cursor: move; }
no-drop      { cursor: no-drop; }
not-allowed  { cursor: not-allowed; }
grab         { cursor: grab; }
grabbing     { cursor: grabbing; }
over         { cursor: over; }
wait-cursor  { cursor: wait; }
```

# FishEye

Improved scrolling and clipping



- **\$cliptosize** property ensures that content is shown only within the dimensions of the window containing the control.
- **\$usewheeldelta** increases granularity of mouse wheel scrolling
- **\$resetonresize** set to false will preserve the position of scrolled content when the field is resized.

# Keyboard shortcuts

.. for window cycling

- F11 and Shift F11 on Windows
- Cmd. ` on macOS



# User Components Library

A great improvement for larger teams enabling greater component development and re-use.

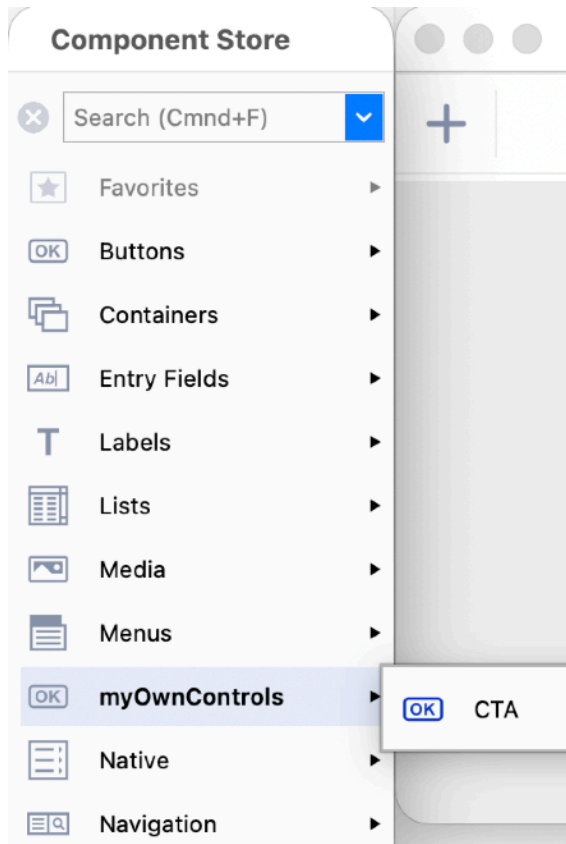


# usercomps.lbs

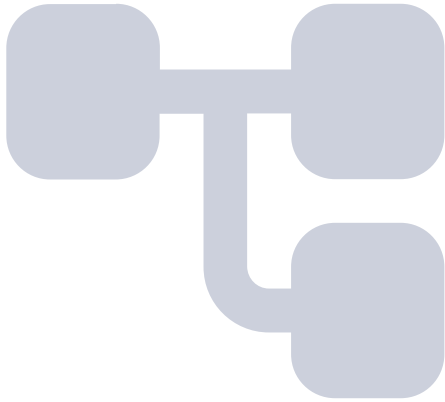
A new dedicated library

for storing:

- Custom components
- Wizards and Templates
- Reusable UI elements



# Scalable Component Collaboration



Empowering teams to build, share, and reuse components efficiently

- Keeps custom components separate from built-in components
- Enables team-wide collaboration for Studio "add-ons"
- Version-control friendly, via built-in VCS or Git with JSON export/import

# New Powerful Functions

More power for image  
processing, file paths, and  
lists



# Image Cropping

Crop images programmatically with ease

**OIMAGE.\$crop()**

for server-side image manipulation



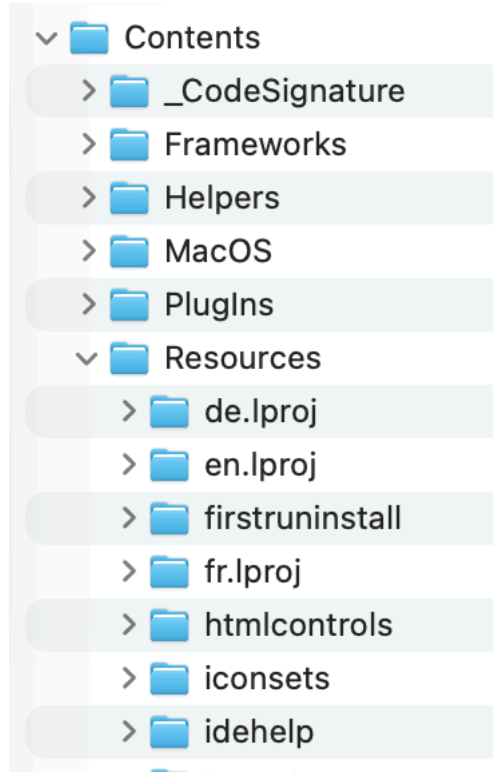
# sys(295)

New sys() function

## **sys(295)**

Returns the location of the program Resources folder e.g.

/Applications/Omnis Studio 12 42929.app/  
Contents/Resources/



# join()

enhanced support

**join()** now accepts column names

The screenshot shows a code editor with the following content:

```
On evClick ## Event Parameters - pRow( Itemreference  
Do join(iCountryList,kCr,'capital') Returns IString
```

Below the code, a table displays the output of the join() function. The table has four columns: name, capital, and population. The first two rows are visible:

	name	capital	population
1	Afghanistan	Kabul	43844000
2	Albania	Tirana	2363314

To the right of the table, a list of capital names is displayed under the heading "Variable":

- Kabul
- Tirana
- Algiers
- Pago Pago
- Andorra la Vella
- Luanda
- The Valley

The logo for the Omnis Developer Conference 2026. It features a stylized 'O' with an orange dot inside, followed by 'DC' in blue, and '2026' in orange. The entire logo is centered within a white circle.

ODC 2026

Omnis Developer Conference