

Using the Glossary with a Complex Grid

Due to the behavior of the Complex Grid it is not possible to directly attach to the field inside a Complex Grid. If you do, some functionality like exceptions ceases to function correctly. However, it is possible to attach the glossary object to the Complex Grid instead. The glossary will then intercept key presses for all the grids children once you call \$start().

It has come to our attention that you cannot call \$start() and \$stop() on evBefore and evAfter events for child controls of the Complex Grid. It causes redraw problems and Omnis may crash (Windows platform only).

If you require the glossary active for only some of the grid's fields, call \$start() and \$stop() in the \$construct and \$destruct methods of the window. You can add a property to your glossary object to enable and disable the glossary at the appropriate times.

Adding \$enabled to the glossary object

```
;; new instance variable of glossary object
```

No.	Instance Variable	Type	Subtype	Initial value
1	ivEnabled	Boolean		kTrue

```
;; new methods for reading and writing the property
```

```
##### Method $enabled #####
```

```
No. Method text
```

```
1 Quit method ivEnabled
```

```
##### Method $enabled.$assign #####
```

No.	Parameter	Type	Subtype
1	pEnabled	Boolean	

```
1 pEnabled Boolean
```

```
No. Method text
```

```
1 Calculate ivEnabled as pEnabled
```

```
2 Quit method kTrue
```

```
;; modify the following method
```

```
##### Method $word #####
```

```
No. Method text
```

```
If not(ivEnabled)
```

```
Calculate $cinst.$text as #NULL
```

```
End If
```

Code in your window class:

```
;; instance variables of window class
```

No.	Instance Variable	Type	Subtype
1	ivGlossary	Object	Example.oGlossary

```
;; $construct method of window
```

```
Do ivGlossary.$setfield($cinst.$objs.ComplexGrid)  
Do ivGlossary.$enabled.$assign(kFalse)  
Do ivGlossary.$start()
```

```
;; $destruct method of window
```

```
Do ivGlossary.$stop()
```

```
;; $event method of field inside the grid that requires the glossary
```

```
On evBefore
```

```
  Do ivGlossary.$enabled.$assign(kTrue)  
  Quit event handler (Pass to next handler)
```

```
On evAfter
```

```
  Do ivGlossary.$enabled.$assign(kFalse)  
  Quit event handler (Pass to next handler)
```