

# Widgets

Blog by Jyoti Singh

The purpose of this blog is to inform you about some changes in the way widgets are documented and described.

Widgets are the elements that enable the user to interact with a Uniface application. In Uniface, we recognize two types of widgets:

- Physical widgets are the implementations that provide the functionality to the user. They vary depending on the underlying GUI technology.
- Logical widgets are named configurations of the physical widgets. Developers can configure physical widgets to use specific properties/styles to create their own logical widgets. Consequently, one physical widget can be referenced by multiple logical widgets. The available widget properties are dictated by the physical widget, and it is the physical widget that knows how to interpret properties. Logical widget definitions are defined in the [webwidgets] and [widgets] sections from the initialization file.

The logical widget definitions are available to the Development Environment (IDE), where they are shown in the **Widget Type** property for entities and fields, allowing developers to select one for use. By default, a Uniface installation provides a default set of logical widgets, but this set can be customized by the developer.

The logical widgets used in the application must also be made available at runtime via the [webwidgets] and [widgets] sections from the initialization file in the runtime environment. Forms, Reports, and Dynamic Server Pages (DSPs) use this mechanism.

With the introduction of the physical widget HTMLINPUT for DSPs, which can be used to specify very different input elements, the emphasis is moving more and more towards the capabilities of the physical widget rather than the logical widget. To reflect this shift in emphasis, the widget documentation for Uniface 10 has been restructured. It now documents the functionality, triggers, and properties of physical widgets, and describes logical widgets in terms of their purpose, since the functionality and properties can be changed by the developer.

To illustrate with an example:

By default, physical widget HTMLINPUT is mapped to EDITBOX, but HTMLINPUT can also be customized to get a custom Input search field which can set or return the value of the placeholder attribute.

Ini file:

```
[webwidget]
```

```
Editbox = htmlinput(html:type=text;detail=dblclick)
```

```
Searchbox = htmlinput(html:type=text;html:size=20;html:placeholder="Type to search")
```

Above two configurations of same physical widget would result into two logical widgets with different capabilities:



If you are interested to know more about the widgets in Uniface, please visit the online documentation <https://u.uniface.info/docs/1000/uniface/componentConstruction/widgets.htm?Highlight=widget>