Offline Configuration

4.02.02 Properties

There are a number of properties associated with the Master Record and the object hierarchy. These properties are as follows:

- **ApplicationObject:** The Application object contains the Master Record for a given application. It also contains the Addendums, the Addendum index if there is more than a single Addendum and the properties associated with the given Master Record and Addendum objects. An example of an application object is shown below. In this example, the application object will represent the work order application in Offline.

  ```
  g_WorkOrderAppDesc = { };  // Work Order Application
  g_WorkOrderApp = SR3ApplicationFramework.createApplication(g_Wor
  kOrderAppDesc);
  ```

- **ApplicationObject.MasterRecord:** The Master Record for a given application. There can be only one Master Record for any application. In our example, the Master Record is represented as follows:

  ```
  g_WorkOrderAppDesc.MasterRecord = { };
  ```

- **ApplicationObject.Addendum[nIndex]:** The index for the Addendums to a given Master Record. Since there are usually multiple Addendums associated with a given Master Record, the Addendums are stored in an array. The array is represented as follows:

  ```
  g_WorkOrderAppDesc.Addendum = [{ },{ },{ }];
  ```

  The individual Addendums themselves are represented as follows:

  ```
  g_WorkOrderAppDesc.Addendum = [0];
  g_WorkOrderAppDesc.Addendum = [1];
  g_WorkOrderAppDesc.Addendum = [2];
  ```

- **ApplicationObject.Name:** Represents the name of the application as in the following example:

  ```
  g_WorkOrderAppDesc.Name = "Work Order";
  ```

- **ApplicationObject.Pattern:** The Pattern property represents the pattern structure of the Application. For the Master-Addendum data structure, the following code would be used:

  ```
  g_WorkOrderAppDesc.Pattern = APPLICATION_PATTERN_MASTER_ADDENDUM
  ;
  ```

- **ApplicationObject.MasterRecord.Screen:** The Screen property represents the screen caption for either the Master Record or the Addendum. It is written as follows:

(c) 2026 Liquid UI | Synactive | GuiXT <dev@guixt.com> | 2026-11-03 01:12

URL: https://www.liquid-ui.com/knowledge_base/content/130/1054/en/40202-properties.html

```
g_WorkOrderAppDesc.MasterRecord.Screen = "Process Order | Change
 Order | Configuration";
```

- **ApplicationObject.MasterRecord.TCode:** This property represents the transaction code (Tcode) of the Master Record. For a Master Record, the code would be as follows:

```
g_WorkOrderAppDesc.MasterRecord.TCode = "wkdr";
```

- **ApplicationObject.MasterRecord.Database:** Represents the application database and contains the screen elements that are displayed on the relevant screens. In our example, we are using three elements, separated by commas. We are also using the Table style pattern, so the elements will be displayed in a table. The code is as follows:

```
g_WorkOrderAppDesc.
MasterRecord.Database = [{
label:"Order",fieldname:"g_wo_no", displaywidth:8,fieldlength:10
,showintable:1,sharedfield:1},{
label:"Activity",fieldname:"g_wo_activity",displaywidth:8,fieldl
ength:10,showintable:1,share
dfield:2},{
label:"Location",fieldname:"g_wo_location",displaywidth:8,fieldl
ength:10,showintable:1}];
```

- **ApplicationObject.MasterRecord.Config:** This property represents the Configuration Screen and typically will contain the default values for the Master data download filter. The code for our example Configuration Screen is shown below

```
g_WorkOrderAppDesc.Ma
sterRecord.Config = [{
label:"P
lant",fieldname:
"g_cf_plant",defaultvalue:"1000"},{



},{
lab
el:"Or
der Type",fi
eldname:"g_cf_otype",def
aultvalue:"PM01"},{
la
bel:
"Downloa
d Rows",fieldnam
e:"g_cf_dlrows",defaultvalue:"5"},{
label:"From Date",fieldname:"g_cf_frdate",defaultvalue:"03.21.20
```


(c) 2026 Liquid UI | Synactive | GuiXT <dev@guixt.com> | 2026-11-03 01:12
URL: https://www.liquid-ui.com/knowledge_base/content/130/1054/en/40202-properties.html

```
11"}];
```

- **ApplicationObject.Addendum[nIndex].Name:** The Addendum Name property is used in the same manner as the Master Record name property. it identifies the name of an Addendum as shown below:

  ```
  g_WorkOrderAppDesc.Addendum[0].Name = "Parts";
  ```

- **ApplicationObject.Addendum[nIndex].Screen:** Like the Name property, the Screen property is used exactly the same as the Master Record Screen property described earlier. The code is as follows:

  ```
  g_WorkOrderAppDesc.Addendum[0].Screen = "@17@Add Parts";
  ```

- **ApplicationObject.Addendum[nIndex].TCode:** Represents the transaction code for an Addendum. See the following example:

  ```
  g_WorkOrderAppDesc.MasterRecord.TCode = "iw41";
  ```

- **ApplicationObject.Addendum[nIndex].Pattern:** The Addendum pattern differentiates between the Addendums and can have two possible styles. For a screen containing edit fields, lookup tables, checkboxes and radiobuttons, the following code would be used:

  ```
  g_WorkOrderAppDesc.Addendum[0].Pattern = ADDENDUM_PATTERN_NORMAL
  _STYLE;
  ```

  For a screen consisting of elements represented in table format, the following code would be used:

  ```
  g_WorkOrderAppDesc.Addendum[0].Pattern = ADDENDUM_PATTERN_TABLE_
  STYLE;
  ```

- **ApplicationObject.Addendum[nIndex].Database:** Like the Master Record, the Addendums also have a database property that will contain the screen elements used in the Addendum screen or screens.
- **Additional Config and Database Attributes:** There are several other attributes that can be used with the Database and Config properties. These attributes are as follows:
  - **Fieldname:** The field name for a given screen element.
  - **Fieldtype:** The data type (string, integer, etc) for a given screen element.
  - **Fieldlength:** The length of the field in a table.
  - **Keytype:** The database column key type.
  - **Columntype:** The column type. Based on the screen element type.
  - **Usesameline:** Displays the screen element in the same row as the previous screen element.
  - **Sharedfield:** Specifies that the field is shared across more than one screen.

(c) 2026 Liquid UI | Synactive | GuiXT <dev@guixt.com> | 2026-11-03 01:12
URL: https://www.liquid-ui.com/knowledge_base/content/130/1054/en/40202-properties.html

- **Hidden:** Specifies that the field is hidden in the screen.
- **Showintable:** Displays the element in a screen table.
- **Displaywidth:** Specifies the width for edit control and table column.
- **Readonly:** Makes the field read-only if set to '1'.
- **Screenonly:** Specifies that a particular field used onscreen only and is not stored in the database.

Unique solution ID: #2057
Author: sarvani.kusuri@guixt.com
Last update: 2021-06-03 17:10

(c) 2026 Liquid UI | Synactive | GuiXT <dev@guixt.com> | 2026-11-03 01:12
URL: https://www.liquid-ui.com/knowledge_base/content/130/1054/en/40202-properties.html