

First project Vending Machine Documentation

Table of Contents

1. Project description
2. Code component descriptions
 - 2.1. DQMH® modules
 - 2.1.1. Preamble
 - 2.1.2. Modules overview
 - 2.1.3. Config.lvlib
 - 2.1.4. Mensajes.lvlib
 - 2.1.5. Monedas.lvlib
 - 2.1.6. Productos.lvlib
 - 2.1.7. UI.lvlib
 - 2.2. Libraries
 - 2.3. Classes
3. VI descriptions
 - 3.1. DQMH® modules
 - 3.1.1. Config.lvlib
 - 3.1.2. Mensajes.lvlib
 - 3.1.3. Monedas.lvlib
 - 3.1.4. Productos.lvlib
 - 3.1.5. UI.lvlib
 - 3.2. Libraries
 - 3.3. Classes
4. Legal Information
 - 4.1. Document creation
 - 4.1.1. Antidoc
 - 4.1.2. AsciiDoc for LabVIEW™
 - 4.1.3. Graph Builder
 - 4.2. Product used in the project
 - 4.2.1. DQMH®

1. Project description

No description found (add content in project description)

2. Code component descriptions

2.1. DQMH® modules

This section describes DQMH® module responsibilities and relationships.

2.1.1. Preamble

A DQMH module is the main component of an architecture based on DQMH® framework. A DQMH module is used to implement a section of the application that has one responsibility.

DQMH® framework defines two different type of DQMH module.

Singleton:

A Singleton DQMH module can have only one instance running at any given time.

Cloneable:

A Cloneable DQMH module can have one or multiple instances running in parallel.

DQMH® framework defines two different ways to carry data throughout the application and with both other DQMH modules and non-DQMH based code.

Request events:

A request is a code that fires an event requesting the DQMH module to do something. Multiple locations in the code can send events to the DQMH module.

Request events are many-to-one.

Requests are usually named using imperative tense.

Broadcast events:

A broadcast is a code that fires an event broadcasting that the DQMH module did something. Multiple Event Structures can register to handle the Broadcast Events.

Broadcast Events are one-to-many.

Broadcasts are usually named using past tense or passive voice.



Refer to the DQMH® framework official [documentation](http://delacor.com/documentation/dqmh-html/) (<http://delacor.com/documentation/dqmh-html/>) to find more details on how the framework works

The following section gives you details on the project architecture relying on this framework. It gives you an overview of the modules' interaction and detailed information on each module.

Graphs used in this section have the following legend:

Components:

```
digraph G268523 {
  rankdir=LR;
  edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
  node[color=black shape=box]
  "DQMH module / Lvlib"[color=black shape=component]
  "Vi"[color=skyblue shape=note]
}
```

Events:

```
digraph G22746 {
  rankdir=LR;
  edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
  node[color=black shape=box]
  " "[color=white shape=box]
  " "[color=white shape=box]
  " "[color=white shape=box]
  " "[color=white shape=box]
  " " -> " " [label="Request to a DQMH module" dir=both color=forestgreen arrowhead=normal arrowtail=none
  style=filled penwidth=1];
  " " -> " " [label="Broadcast from a DQMH module" dir=both color=goldenrod arrowhead=normal arrowtail=none
  style=dashed penwidth=1];
}
```

Start and Stop module callers:

```
digraph G776162 {
  rankdir=LR;
  edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
  node[color=black shape=box]
  "Start Module
  caller"[color=black shape=component]
  "Start Module"[color=yellowgreen shape=note]
  "Start Module" -> "Start Module
  caller" [label="Called by" dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled penwidth=1];
}
```

```
digraph G25320 {
  rankdir=LR;
  edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
  node[color=black shape=box]
  "Stop Module
  caller"[color=black shape=component]
  "Stop Module"[color=tomato shape=note]
  "Stop Module" -> "Stop Module
  caller" [label="Called by" dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
}
```

2.1.2. Modules overview

This project contains the following modules.

Table 1. Modules list

Singleton	Cloneable
Config.lvlib	
Mensajes.lvlib	
Monedas.lvlib	
Productos.lvlib	
UI.lvlib	

This graph represents the links between all DQMH modules.

```
digraph G721463 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"UI"[color=black shape=component]
"Config"[color=black shape=component]
"Mensajes"[color=black shape=component]
"Monedas"[color=black shape=component]
"Productos"[color=black shape=component]
"Config" -> "Config" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Config" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Mensajes" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Monedas" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Monedas" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Config" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Productos" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Productos" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled
penwidth=1];
"Config" -> "Productos" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "UI" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
}
```

2.1.3. Config.lmlib

Type: Singleton

Responsibility: Library that is used to do the config requests

Module Start/Stop calls

```
digraph G761793 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"UI"[color=black shape=component]
"Test Config API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"Config"[color=black shape=component]
"Start Module" -> "UI" [dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Start Module" -> "Test Config API" [dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "Config" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test Config API" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted
penwidth=1];
}
```

Table 2. Start and Stop module callers

Function	Callers

Function	Callers
Config.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Config API.vi
Config.lvlib:Stop Module.vi	Config.lvlib:Handle Exit.vi Test Config API.vi

Module relationship

```

digraph G519690 {
    rankdir=LR;
    edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
    node[color=black shape=box]
    "Config"[color=slateblue shape=component]
    "UI"[color=black shape=component]
    "Test Config API"[color=skyblue shape=note]
    "Monedas"[color=black shape=component]
    "Productos"[color=black shape=component]
    "UI" -> "Config" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
    "Test Config API" -> "Config" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
    "Config" -> "Config" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
    "Config" -> "Test Config API" [label=" " dir=both color=goldenrod arrowhead=normal arrowtail=none style=dashed penwidth=1];
    "Config" -> "Config" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
    "Config" -> "Monedas" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
    "Config" -> "Productos" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
}

```

Table 3. Requests callers

Request Name	Callers
Config.lvlib:Show Panel.vi	UI.lvlib:Main.vi Test Config API.vi
Config.lvlib:Hide Panel.vi	Test Config API.vi
Config.lvlib:Get Module Execution Status.vi	Config.lvlib:Start Module.vi Config.lvlib:Obtain Broadcast Events for Registration.vi
Config.lvlib:Show Diagram.vi	Test Config API.vi

Table 4. Broadcasts Listeners

Broadcast Name	Listeners
Config.lvlib:Module Did Init.vi	Test Config API.vi
Config.lvlib:Status Updated.vi	Test Config API.vi
Config.lvlib>Error Reported.vi	Test Config API.vi
Config.lvlib:Module Did Stop.vi	Test Config API.vi
Config.lvlib:Update Module Execution Status.vi	Test Config API.vi

Table 5. Used requests

Module	Brodcasts
Config.lvlib	Config.lvlib:Get Module Execution Status.vi
Monedas.lvlib	Monedas.lvlib:Set coin values.vi
Productos.lvlib	Productos.lvlib:Set products.vi

Table 6. Registred broadcast

Module	Brodcasts
—	—

2.1.4. Mensajes.lvlib

Type: Singleton

Responsibility: Library used to display messages in the UI with the publicar mensaje Request

Module Start/Stop calls

```
digraph G16642 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"UI"[color=black shape=component]
"Test Mensajes API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"Mensajes"[color=black shape=component]
"Start Module" -> "UI" [dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Start Module" -> "Test Mensajes API" [dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled penwidth=1];
"Stop Module" -> "Mensajes" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "UI" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test Mensajes API" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
}
```

Table 7. Start and Stop module callers

Function	Callers
Mensajes.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Mensajes API.vi
Mensajes.lvlib:Stop Module.vi	Mensajes.lvlib:Handle Exit.vi UI.lvlib:Main.vi Test Mensajes API.vi

Module relationship

```
digraph G859063 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Mensajes"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Mensajes API"[color=skyblue shape=note]
"Monedas"[color=black shape=component]
"Productos"[color=black shape=component]
"UI" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Mensajes API" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Mensajes" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Monedas" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Mensajes" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Mensajes" -> "Test Mensajes API" [label=" " dir=both color=goldenrod arrowhead=normal arrowtail=none style=dashed penwidth=1];
"Mensajes" -> "Mensajes" [label=" " dir=both color=forestgreen arrowhead=onormal arrowtail=none style=filled penwidth=1];
}
```

Table 8. Requests callers

--

Request Name	Callers
Mensajes.lvlib:Show Panel.vi	Test Mensajes API.vi
Mensajes.lvlib:Hide Panel.vi	Test Mensajes API.vi
Mensajes.lvlib:Get Module Execution Status.vi	Mensajes.lvlib:Start Module.vi Mensajes.lvlib:Obtain Broadcast Events for Registration.vi
Mensajes.lvlib:Show Diagram.vi	Test Mensajes API.vi
Mensajes.lvlib:Publicar Mensaje.vi	Monedas.lvlib:Main.vi Productos.lvlib:Main.vi UI.lvlib:Main.vi Test Mensajes API.vi

Table 9. Broadcasts Listeners

Broadcast Name	Listeners
Mensajes.lvlib:Module Did Init.vi	Test Mensajes API.vi
Mensajes.lvlib:Status Updated.vi	Test Mensajes API.vi
Mensajes.lvlib>Error Reported.vi	Test Mensajes API.vi
Mensajes.lvlib:Module Did Stop.vi	Test Mensajes API.vi
Mensajes.lvlib:Update Module Execution Status.vi	Test Mensajes API.vi

Table 10. Used requests

Module	Brodcasts
Mensajes.lvlib	Mensajes.lvlib:Get Module Execution Status.vi

Table 11. Registred broadcast

--

Module	Broadcasts
—	—

2.1.5. Monedas.lvlib

Type: Singleton

Responsibility: Library used to create the requests from the Monedas Module. Its used to do al the tasks that are needed to do the operation successfully

Module Start/Stop calls

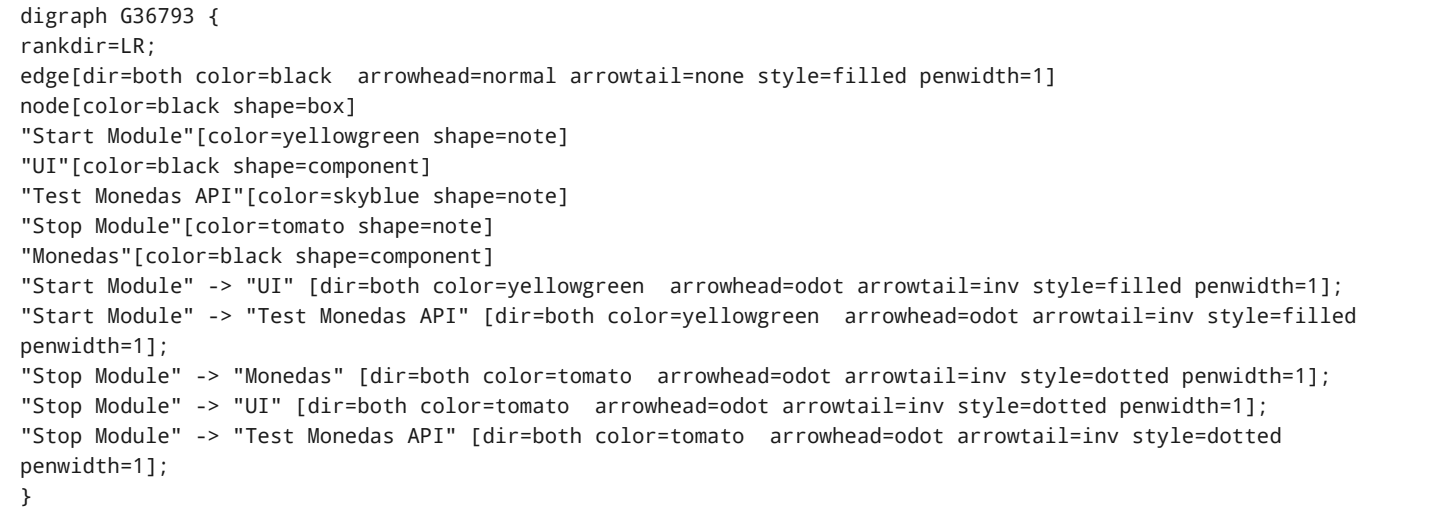


Table 12. Start and Stop module callers

Function	Callers
Monedas.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Monedas API.vi
Monedas.lvlib:Stop Module.vi	Monedas.lvlib:Handle Exit.vi UI.lvlib:Main.vi Test Monedas API.vi

Module relationship

```

digraph G183984 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Monedas"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Monedas API"[color=skyblue shape=note]
"Config"[color=black shape=component]
"Productos"[color=black shape=component]
"Mensajes"[color=black shape=component]
"UI" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Monedas API" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Monedas" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Config" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Monedas" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Monedas" -> "Test Monedas API" [label=" " dir=both color=goldenrod arrowhead=normal arrowtail=none style=dashed penwidth=1];
"Monedas" -> "Mensajes" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Monedas" -> "Monedas" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
}

```

Table 13. Requests callers

Request Name	Callers
Monedas.lvlib:Show Panel.vi	Test Monedas API.vi
Monedas.lvlib:Hide Panel.vi	Test Monedas API.vi
Monedas.lvlib:Get Module Execution Status.vi	Monedas.lvlib:Start Module.vi Monedas.lvlib:Obtain Broadcast Events for Registration.vi
Monedas.lvlib:Show Diagram.vi	Test Monedas API.vi
Monedas.lvlib:Coin pressed.vi	Monedas.lvlib:Main.vi Test Monedas API.vi
Monedas.lvlib:Set coin values.vi	Config.lvlib:Main.vi Test Monedas API.vi
Monedas.lvlib:Reset credit to 0.vi	Productos.lvlib:Main.vi Test Monedas API.vi

Request Name	Callers
Monedas.lvlib:Ask if credit.vi	Productos.lvlib:Main.vi Test Monedas API.vi

Table 14. Broadcasts Listeners

Broadcast Name	Listeners
Monedas.lvlib:Module Did Init.vi	Test Monedas API.vi
Monedas.lvlib:Status Updated.vi	Test Monedas API.vi
Monedas.lvlib:Error Reported.vi	Test Monedas API.vi
Monedas.lvlib:Module Did Stop.vi	Test Monedas API.vi
Monedas.lvlib:Update Module Execution Status.vi	Test Monedas API.vi

Table 15. Used requests

Module	Brodcasts
Mensajes.lvlib	Mensajes.lvlib:Publicar Mensaje.vi
Monedas.lvlib	Monedas.lvlib:Get Module Execution Status.vi Monedas.lvlib:Coin pressed.vi

Table 16. Registred broadcast

Module	Brodcasts
—	—

2.1.6. Productos.lvlib

Type: Singleton

Responsibility: Library used to create all the request to do all the Productos Module tasks and get the Vending machine working

Module Start/Stop calls

```
digraph G954595 {
  rankdir=LR;
  edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
  node[color=black shape=box]
  "Start Module"[color=yellowgreen shape=note]
  "UI"[color=black shape=component]
  "Test Productos API"[color=skyblue shape=note]
  "Stop Module"[color=tomato shape=note]
  "Productos"[color=black shape=component]
  "Start Module" -> "UI" [dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled penwidth=1];
  "Start Module" -> "Test Productos API" [dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled penwidth=1];
  "Stop Module" -> "Productos" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
  "Stop Module" -> "UI" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
  "Stop Module" -> "Test Productos API" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
}
```

Table 17. Start and Stop module callers

Function	Callers
Productos.lvlib:Start Module.vi	UI.lvlib:Main.vi Test Productos API.vi
Productos.lvlib:Stop Module.vi	Productos.lvlib:Handle Exit.vi UI.lvlib:Main.vi Test Productos API.vi

Module relationship

```
digraph G427538 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Productos"[color=slateblue shape=component]
"UI"[color=black shape=component]
"Test Productos API"[color=skyblue shape=note]
"Config"[color=black shape=component]
"Mensajes"[color=black shape=component]
"Monedas"[color=black shape=component]
"UI" -> "Productos" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Test Productos API" -> "Productos" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Productos" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Config" -> "Productos" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Test Productos API" [label=" " dir=both color=goldenrod arrowhead=normal arrowtail=none style=dashed penwidth=1];
"Productos" -> "Mensajes" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Monedas" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"Productos" -> "Productos" [label=" " dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
}
```

Table 18. Requests callers

Request Name	Callers
Productos.lvlib:Show Panel.vi	Test Productos API.vi
Productos.lvlib:Hide Panel.vi	Test Productos API.vi
Productos.lvlib:Get Module Execution Status.vi	Productos.lvlib:Start Module.vi Productos.lvlib:Obtain Broadcast Events for Registration.vi
Productos.lvlib:Show Diagram.vi	Test Productos API.vi
Productos.lvlib:Set products.vi	Config.lvlib:Main.vi Productos.lvlib:Main.vi Test Productos API.vi
Productos.lvlib:Product Selection.vi	Test Productos API.vi
Productos.lvlib:Update Product Aspect.vi	Test Productos API.vi

Table 19. Broadcasts Listeners

Broadcast Name	Listeners
Productos.lvlib:Module Did Init.vi	Test Productos API.vi
Productos.lvlib:Status Updated.vi	Test Productos API.vi
Productos.lvlib:Error Reported.vi	Test Productos API.vi
Productos.lvlib:Module Did Stop.vi	Test Productos API.vi
Productos.lvlib:Update Module Execution Status.vi	Test Productos API.vi

Table 20. Used requests

Module	Brodcasts
Mensajes.lvlib	Mensajes.lvlib:Publicar Mensaje.vi
Monedas.lvlib	Monedas.lvlib:Reset credit to 0.vi Monedas.lvlib:Ask if credit.vi
Productos.lvlib	Productos.lvlib:Get Module Execution Status.vi Productos.lvlib:Set products.vi

Table 21. Registred broadcast

Module	Brodcasts
—	—

2.1.7. UI.lvlib

Type: Singleton

Responsibility: Library used to display all the modules that are used in the DQMH Architecture to make the vending machine working properly

Module Start/Stop calls

```
digraph G62743 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"Start Module"[color=yellowgreen shape=note]
"Test UI API"[color=skyblue shape=note]
"Stop Module"[color=tomato shape=note]
"UI"[color=black shape=component]
"Start Module" -> "Test UI API" [dir=both color=yellowgreen arrowhead=odot arrowtail=inv style=filled
penwidth=1];
"Stop Module" -> "UI" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
"Stop Module" -> "Test UI API" [dir=both color=tomato arrowhead=odot arrowtail=inv style=dotted penwidth=1];
}
```

Table 22. Start and Stop module callers

Function	Callers
UI.lvlib:Start Module.vi	Test UI API.vi
UI.lvlib:Stop Module.vi	UI.lvlib:Handle Exit.vi Test UI API.vi

Module relationship

```
digraph G688635 {
rankdir=LR;
edge[dir=both color=black arrowhead=normal arrowtail=none style=filled penwidth=1]
node[color=black shape=box]
"UI"[color=slateblue shape=component]
"Test UI API"[color=skyblue shape=note]
"Config"[color=black shape=component]
"Mensajes"[color=black shape=component]
"Test UI API" -> "UI" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "UI" [dir=both color=forestgreen arrowhead=normal arrowtail=none style=filled penwidth=1];
"UI" -> "Test UI API" [label=" " dir=both color=goldenrod arrowhead=normal arrowtail=none style=dashed
penwidth=1];
"UI" -> "Config" [label=" " dir=both color=forestgreen arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"UI" -> "Mensajes" [label=" " dir=both color=forestgreen arrowhead=onormal arrowtail=none style=filled
penwidth=1];
"UI" -> "UI" [label=" " dir=both color=forestgreen arrowhead=onormal arrowtail=none style=filled penwidth=1];
}
```

Table 23. Requests callers

Request Name	Callers
UI.lvlib:Show Panel.vi	Test UI API.vi
UI.lvlib:Hide Panel.vi	Test UI API.vi

Request Name	Callers
UI.lvlib:Get Module Execution Status.vi	UI.lvlib:Start Module.vi UI.lvlib:Obtain Broadcast Events for Registration.vi
UI.lvlib:Show Diagram.vi	Test UI API.vi

Table 24. Broadcasts Listeners

Broadcast Name	Listeners
UI.lvlib:Module Did Init.vi	Test UI API.vi
UI.lvlib:Status Updated.vi	Test UI API.vi
UI.lvlib>Error Reported.vi	Test UI API.vi
UI.lvlib:Module Did Stop.vi	Test UI API.vi
UI.lvlib:Update Module Execution Status.vi	Test UI API.vi

Table 25. Used requests

Module	Brodcasts
Config.lvlib	Config.lvlib:Show Panel.vi
Mensajes.lvlib	Mensajes.lvlib:Publicar Mensaje.vi
UI.lvlib	UI.lvlib:Get Module Execution Status.vi

Table 26. Registred broadcast

Module	Brodcasts
—	—

2.2. Libraries

This section describes the libraries contained in the project.

2.3. Classes

This section describes the classes contained in the project.

3. VI descriptions

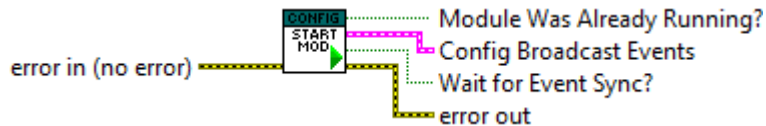
3.1. DQMH® modules

This section describes DQMH® modules events.

3.1.1. Config.lvlib

Config.lvlib:Start Module.vi

Event type: Not a DQMH Event

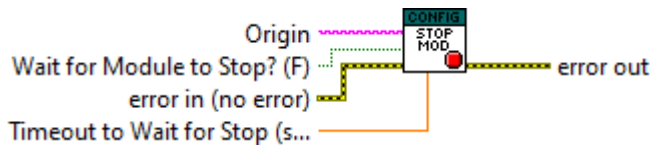


Description:

Launches the Module Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.lvlib:Stop Module.vi

Event type: Not a DQMH Event

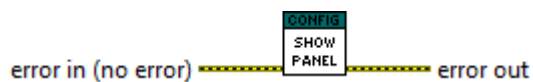


Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.lvlib:Show Panel.vi

Event type: Request

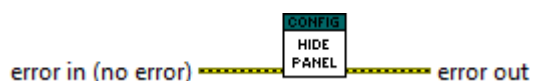


Description:

Send the Show Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.lvlib:Hide Panel.vi

Event type: Request

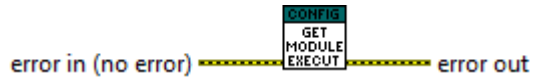


Description:

Send the Hide Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.IVlib:Get Module Execution Status.vi

Event type: Request

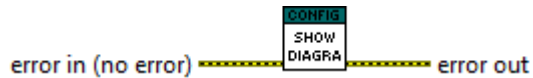


Description:

Fire the Get Module Execution Status request. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.IVlib:Show Diagram.vi

Event type: Request

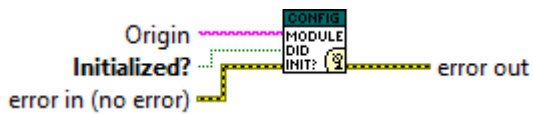


Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.IVlib:Module Did Init.vi

Event type: Broadcast

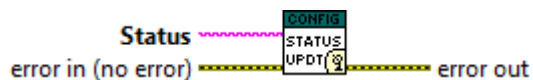


Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.IVlib:Status Updated.vi

Event type: Broadcast

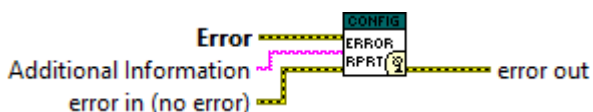


Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.IVlib>Error Reported.vi

Event type: Broadcast



Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.Ilib:Module Did Stop.vi

Event type: Broadcast

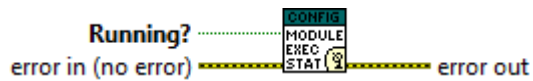


Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

Config.Ilib:Update Module Execution Status.vi

Event type: Broadcast



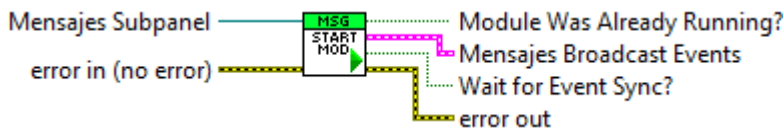
Description:

Broadcast event to specify whether or not the module is running. ____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.2. Mensajes.Ilib

Mensajes.Ilib:Start Module.vi

Event type: Not a DQMH Event

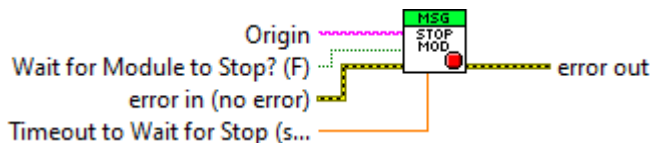


Description:

Launches the Module Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.Ilib:Stop Module.vi

Event type: Not a DQMH Event

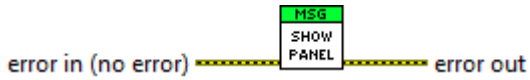


Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. ____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.Ilib:Show Panel.vi

Event type: Request



Description:

Send the Show Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Mensajes.Mlib:Hide Panel.vi](#)

Event type: Request

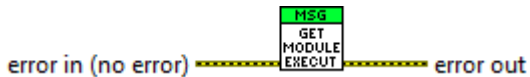


Description:

Send the Hide Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Mensajes.Mlib:Get Module Execution Status.vi](#)

Event type: Request



Description:

Fire the Get Module Execution Status request. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Mensajes.Mlib:Show Diagram.vi](#)

Event type: Request

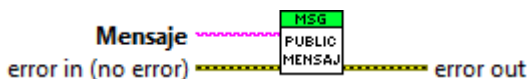


Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). ____ Based on Delacor QMH Project Template 5.0.0.82.

[Mensajes.Mlib:Publicar Mensaje.vi](#)

Event type: Request

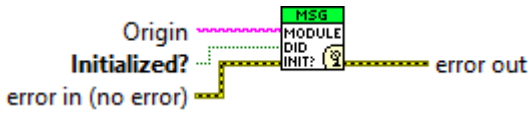


Description:

Publica un mensaje en el modulo de DQMH Mensajes ____ Created using Delacor QMH Event Scriptor 5.0.0.112.

[Mensajes.Mlib:Module Did Init.vi](#)

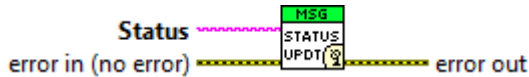
Event type: Broadcast

**Description:**

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.Mlib:Status Updated.vi

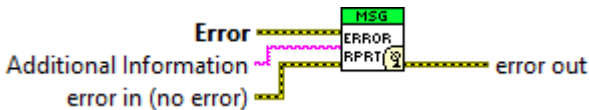
Event type: Broadcast

**Description:**

Send the Status Updated event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.Mlib>Error Reported.vi

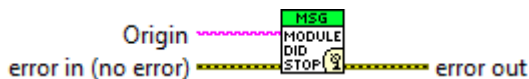
Event type: Broadcast

**Description:**

Send the Error Reported event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.Mlib:Module Did Stop.vi

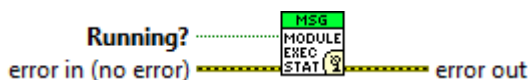
Event type: Broadcast

**Description:**

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

Mensajes.Mlib:Update Module Execution Status.vi

Event type: Broadcast

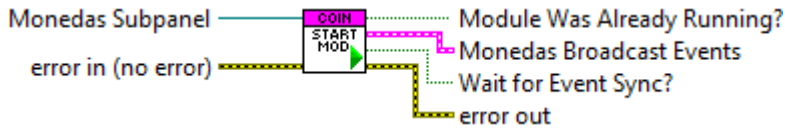
**Description:**

Broadcast event to specify whether or not the module is running. ____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.3. Monedas.Mlib

Monedas.lvlib:Start Module.vi

Event type: Not a DQMH Event

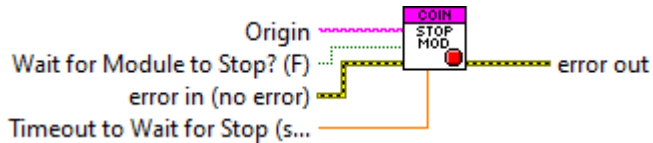


Description:

Launches the Module Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Stop Module.vi

Event type: Not a DQMH Event

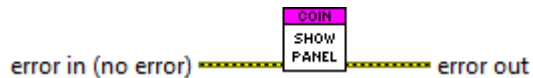


Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Show Panel.vi

Event type: Request

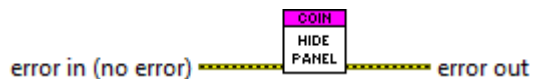


Description:

Send the Show Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Hide Panel.vi

Event type: Request

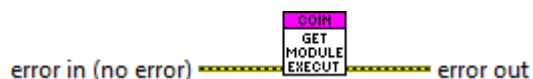


Description:

Send the Hide Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Get Module Execution Status.vi

Event type: Request

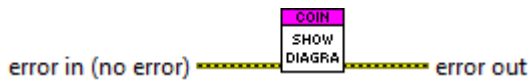


Description:

Fire the Get Module Execution Status request. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Monedas.lvlib:Show Diagram.vi](#)

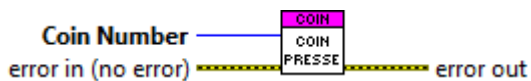
Event type: Request

**Description:**

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). ____ Based on Delacor QMH Project Template 5.0.0.82.

[Monedas.lvlib:Coin pressed.vi](#)

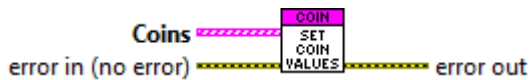
Event type: Request

**Description:**

Event that occurs when a coin is pressed ____ Created using Delacor QMH Event Scriptor 5.0.0.112.

[Monedas.lvlib:Set coin values.vi](#)

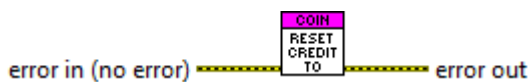
Event type: Request

**Description:**

Set the coin values and saves them in the xml config file ____ Created using Delacor QMH Event Scriptor 5.0.0.112.

[Monedas.lvlib:Reset credit to 0.vi](#)

Event type: Request

**Description:**

Resets the credit to 0 ____ Created using Delacor QMH Event Scriptor 5.0.0.112.

[Monedas.lvlib:Ask if credit.vi](#)

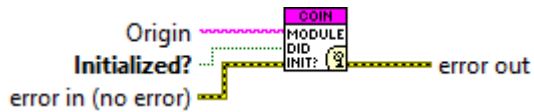
Event type: Request And Wait For Reply

**Description:**

Asks if theres credit ____ Created using Delacor QMH Event Scripter 5.0.0.112.

Monedas.lvlib:Module Did Init.vi

Event type: Broadcast

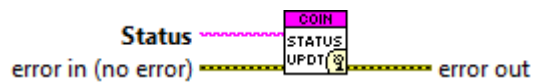


Description:

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Status Updated.vi

Event type: Broadcast

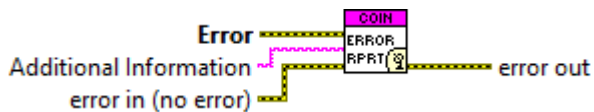


Description:

Send the Status Updated event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Error Reported.vi

Event type: Broadcast

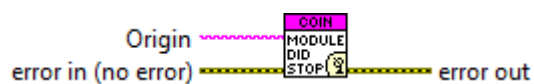


Description:

Send the Error Reported event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Module Did Stop.vi

Event type: Broadcast

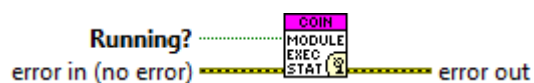


Description:

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

Monedas.lvlib:Update Module Execution Status.vi

Event type: Broadcast



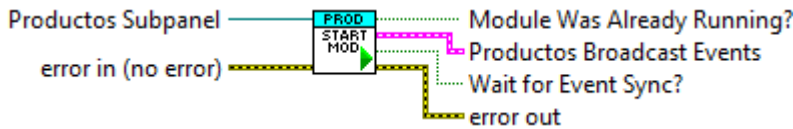
Description:

Broadcast event to specify whether or not the module is running. ____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.4. Productos.lvlib

Productos.lvlib:Start Module.vi

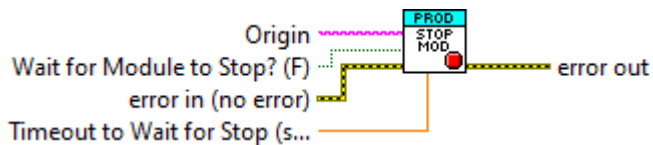
Event type: Not a DQMH Event

**Description:**

Launches the Module Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Productos.lvlib:Stop Module.vi

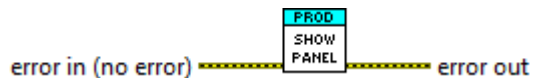
Event type: Not a DQMH Event

**Description:**

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. ____ Based on Delacor QMH Project Template 5.0.0.82.

Productos.lvlib:Show Panel.vi

Event type: Request

**Description:**

Send the Show Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Productos.lvlib:Hide Panel.vi

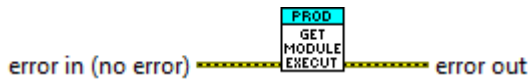
Event type: Request

**Description:**

Send the Hide Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

Productos.lvlib:Get Module Execution Status.vi

Event type: Request

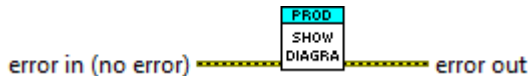


Description:

Fire the Get Module Execution Status request. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Productos.lvlb:Show Diagram.vi](#)

Event type: Request

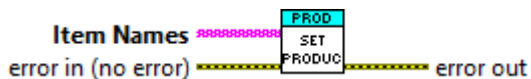


Description:

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). ____ Based on Delacor QMH Project Template 5.0.0.82.

[Productos.lvlb:Set products.vi](#)

Event type: Request

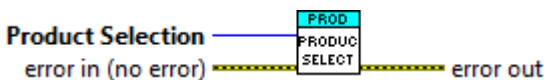


Description:

Set the products in the multi column display ____ Created using Delacor QMH Event Scripiter 5.0.0.112.

[Productos.lvlb:Product Selection.vi](#)

Event type: Request

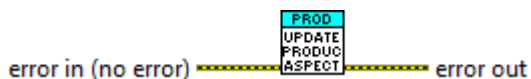


Description:

Provides the information of which product is selected by the user ____ Created using Delacor QMH Event Scripiter 5.0.0.112.

[Productos.lvlb:Update Product Aspect.vi](#)

Event type: Request

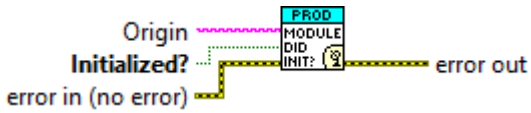


Description:

Updates the information of the config menu to the product display ____ Created using Delacor QMH Event Scripiter 5.0.0.112.

[Productos.lvlb:Module Did Init.vi](#)

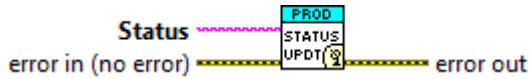
Event type: Broadcast

**Description:**

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Productos.lvlib:Status Updated.vi](#)

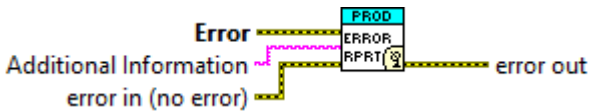
Event type: Broadcast

**Description:**

Send the Status Updated event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Productos.lvlib>Error Reported.vi](#)

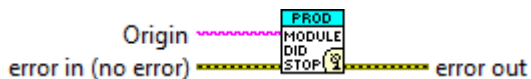
Event type: Broadcast

**Description:**

Send the Error Reported event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Productos.lvlib:Module Did Stop.vi](#)

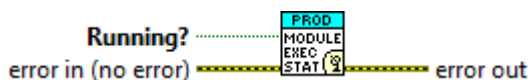
Event type: Broadcast

**Description:**

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

[Productos.lvlib:Update Module Execution Status.vi](#)

Event type: Broadcast

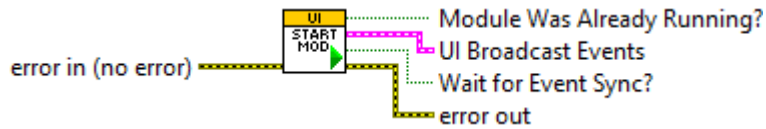
**Description:**

Broadcast event to specify whether or not the module is running. ____ Based on Delacor QMH Project Template 5.0.0.82.

3.1.5. UI.lvlib

UI.lvlib:Start Module.vi

Event type: Not a DQMH Event

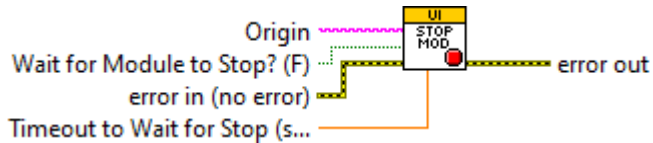


Description:

Launches the Module Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Stop Module.vi

Event type: Not a DQMH Event

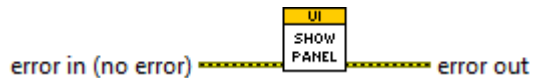


Description:

Send the Stop request to the Module's Main.vi. If **Wait for Module to Stop?** is TRUE, this VI will wait until the module main VI stops, and will timeout at the **Timeout to Wait for Stop** value. This value defaults to "-1", which means the VI will not timeout, and will always wait until the module main VI stops before completing execution. Note: The **Timeout to Wait for Stop** value is ignored if 'Wait for Module to Stop?' is set to FALSE. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Show Panel.vi

Event type: Request



Description:

Send the Show Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Hide Panel.vi

Event type: Request



Description:

Send the Hide Panel request to the Module's Main.vi. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Get Module Execution Status.vi

Event type: Request



Description:

Fire the Get Module Execution Status request. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.Lib:Show Diagram.vi

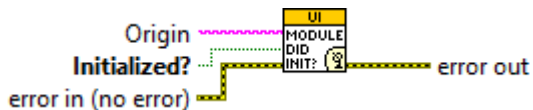
Event type: Request

**Description:**

This VI tells the Module to show its block diagram to facilitate troubleshooting (add probes, breakpoints, highlight execution, etc). ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.Lib:Module Did Init.vi

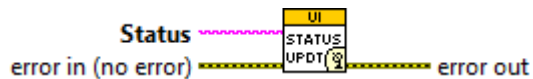
Event type: Broadcast

**Description:**

Send the Module Did Init event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.Lib:Status Updated.vi

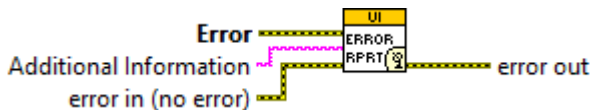
Event type: Broadcast

**Description:**

Send the Status Updated event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.Lib:Error Reported.vi

Event type: Broadcast

**Description:**

Send the Error Reported event to any VI registered to listen to events from the owning module. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.Lib:Module Did Stop.vi

Event type: Broadcast

**Description:**

Send the Module Did Stop event to any VI registered to listen to this module's broadcast events. ____ Based on Delacor QMH Project Template 5.0.0.82.

UI.lvlib:Update Module Execution Status.vi

Event type: Broadcast

**Description:**

Broadcast event to specify whether or not the module is running. ____ Based on Delacor QMH Project Template 5.0.0.82.

3.2. Libraries

This section describes libraries public VIs.

3.3. Classes

This section describes classes public VIs.

4. Legal Information

4.1. Document creation

This document has been generated using the following tools.

4.1.1. Antidoc

Project website: [Antidoc](https://wovalab.gitlab.io/open-source/labview-doc-generator/) (https://wovalab.gitlab.io/open-source/labview-doc-generator/)

Maintainer website: [Wovalab](https://wovalab.com) (https://wovalab.com)

BSD 3-Clause License

Copyright © 2019, Wovalab, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.1.2. AsciiDoc for LabVIEW™

Project website: [AsciiDoc toolkit](https://wovalab.gitlab.io/open-source/asciidoc-toolkit/) (https://wovalab.gitlab.io/open-source/asciidoc-toolkit/)

Maintainer website: [Wovalab](https://wovalab.com) (https://wovalab.com)

BSD 3-Clause License

Copyright © 2019, Wovalab, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.1.3. Graph Builder

Project website: [Graph Builder](https://gitlab.com/cgambini/graph-builder) (<https://gitlab.com/cgambini/graph-builder>)

BSD 3-Clause License

Copyright (c) 2020, Cyril GAMBINI All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4.2. Product used in the project

The documented project has been developed with the following products.

4.2.1. DQMH®

Copyright © 2015-2020 by Delacor, LLC. All Rights Reserved.

Find more details on [Delacor](https://delacor.com/products/dqmh/) (<https://delacor.com/products/dqmh/>) website