

# Create a STOP-sign on a Sales Order

Summary	
<b>Module</b>	B1 Validation System + Universal functions
<b>Level</b>	Advanced
<b>Requirements</b>	B1UP 3.0.0.0 or higher

## Scenario

In this sample we wish to have a visual Indication on a Sales Order if the Customer tied to the sales Order is on hold. We therefor make a "STOP-sign" to visually indicate the status.

The screenshot shows a 'Sales Order' window. At the top, the 'Customer' field is highlighted with a red arrow pointing to the word 'STOP' in bold red letters. The customer details include: Customer No. C20000, Name Norm Thompson, Contact Person Norm Thompson, Customer Ref. No. 1234, and Local Currency. The 'Contents' tab is selected, showing a table with one item: Consulting, G/L Account 41100000-01-0 Sales Revenues - Don, Price after Disc. USD 1000.00, Tax Code PA, and Total (LC) USD 1000.00. The 'Accounting' tab is also visible. The bottom section contains fields for Sales Employee (Brad Thompson), Owner (Thompson, Brad), and a summary of totals: Total Before Discount USD 1000.00, Discount %, Freight, Rounding, Tax USD 95.00, and Total USD 1095.00. There are buttons for OK, Cancel, Copy From, and Copy To.

## Needed Configuration

We will need 3 universal functions and a bunch of B1 Validation System configuration to make the functionality perfect.

## Universal Functions

First thing we need to do is design a "Stop" sign... For this sample we use a label in bold red letters next to the Customer number.

**Sales Order**

Customer **STOP** → C20000

Name Norm Thompson

Contact Person Norm Thompson

Customer Ref. No. 1234

Local Currency

To do this we make the following UF: Content Creator:

**Universal Function Configuration**

Code UF-002

Name **STOP SIGN**

Type Content Creator

Remarks

Active	Item Uid	Type	Description	Left	Top	Width	Height	From Pane	T
<input checked="" type="checkbox"/>	UF_01	Label	STOP	75	4	35		0	0

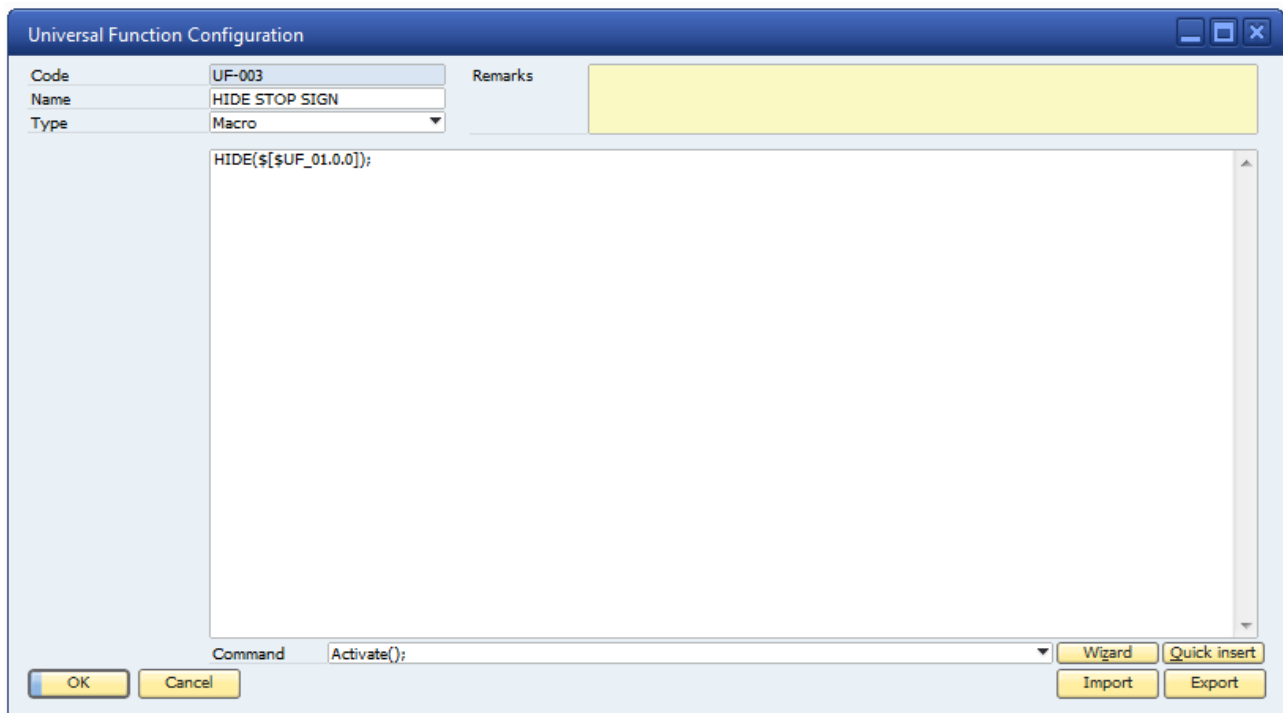
Buttons: Add, Remove, OK, Cancel, Import, Export

Render mode: XML

Text Style	Fore Color
Bold	255

<< Additional properties for the label (255 being the code for Red)

Next we need to Universal Functions to either hide or show the STOP-sign depending on the data. These are quite easily done using macros:

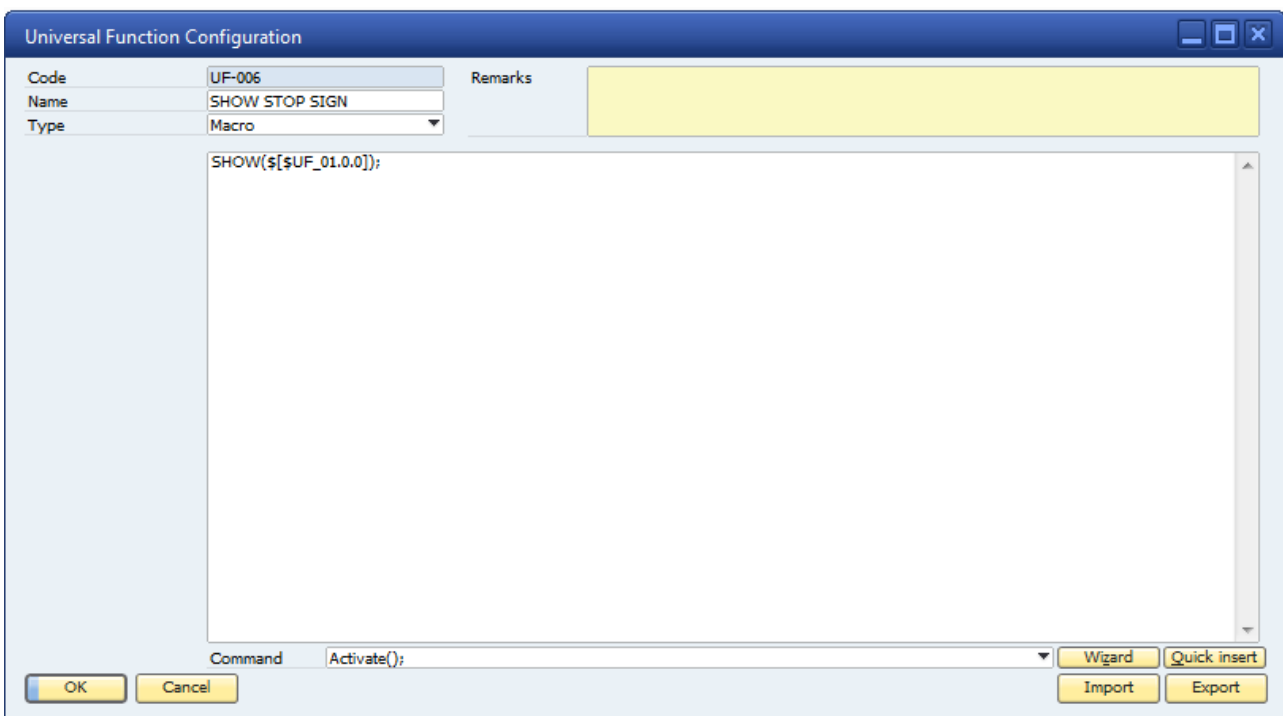


The image shows the 'Universal Function Configuration' dialog box for a macro named 'HIDE STOP SIGN'. The 'Code' field contains 'UF-003', the 'Name' field contains 'HIDE STOP SIGN', and the 'Type' is set to 'Macro'. The 'Remarks' field is empty. The main text area contains the macro code: `HIDE($[UF_01.0.0]);`. At the bottom, the 'Command' dropdown is set to 'Activate()'. There are 'OK', 'Cancel', 'Wizard', 'Quick insert', 'Import', and 'Export' buttons.

Code	UF-003	Remarks
Name	HIDE STOP SIGN	
Type	Macro	

```
HIDE($[UF_01.0.0]);
```

Command: Activate();



The image shows the 'Universal Function Configuration' dialog box for a macro named 'SHOW STOP SIGN'. The 'Code' field contains 'UF-006', the 'Name' field contains 'SHOW STOP SIGN', and the 'Type' is set to 'Macro'. The 'Remarks' field is empty. The main text area contains the macro code: `SHOW($[UF_01.0.0]);`. At the bottom, the 'Command' dropdown is set to 'Activate()'. There are 'OK', 'Cancel', 'Wizard', 'Quick insert', 'Import', and 'Export' buttons.

Code	UF-006	Remarks
Name	SHOW STOP SIGN	
Type	Macro	

```
SHOW($[UF_01.0.0]);
```

Command: Activate();

## B1 Validation System Configurations

Now that we have the functionality we will need some B1 Validation Systems.. We need 3 primary validations

- One to add the STOP Sign to the window (in this example a sales order)
- One to initially hide the STOP-sign (since a window starts in Add-mode and it would not make sense to see the stop-sign without data)
- One to take care of navigating to a new sales order and dynamically show or hide the STOP-sign

## Add StopSign

First is a normal display of a content creator using the Form Load Event

**B1 Validation Configuration**

Name:  Sort:  Number:

Users (Empty = All):  ☒ Include users ☐ Exclude users

**Trigger**

FormType:   
 Item:   
 Column:   
 Row:   
 Menu Uid:   
 FormMode:   
 Event:   
 BeforeAction:   
 ActionSuccess:   
 ItemChanged:   
 InnerEvent:   
 CharPressed:  Modifier:

**Condition**

☒ No condition, just execute function   
☐ Simple condition  
 When item | (column) | Comparison type:     
 Value | Type | Execute function:     
☐ SQL Condition

**Comments**

☒ Active

Condition result	Function to execute

It simply call the UF-002 which is the Content creator

## Initially hide

Next we also at Form-load need to make the initial hide.. Please note that we use the Sort field to make sure the content creator is called before the hide (If it is not created it cannot be hidden). The way the sort field works is that it simply takes similar events in the sort-order (lowest first)

**B1 Validation Configuration**

Name: STOP SIGN INITIAL HIDE      Sort: 101      Number: 69  
 Users (Empty = All):      ☒ Include users      ☐ Exclude users

**Trigger**

FormType: 139  
 Item:      Column:      Row:      Menu Uid:      FormMode: All  
 Event: Form Load  
 BeforeAction: True  
 ActionSuccess: Ignore  
 ItemChanged: Ignore  
 InnerEvent: Ignore  
 CharPressed:      Modifier: None

**Condition**

☒ No condition, just execute function      UF-003  
☐ Simple condition  
 When item | (column) | Comparison type      Equals  
 Value | Type | Execute function      String (0)  
☐ SQL Condition      Common SQL structures

**Comments**

☒ Active

OK      Cancel      Import      Export

All we do is to call the hide-macro

## Navigate

Finally we catch the navigation (Data-Load Event).. We here use an advance SQL Condition to determine if the currently displayed BP is on hold or not (On hold = Show stop-sign Not on Hold = Hide Stop Sign)

**B1 Validation Configuration**

Name: STOP SIGN HIDE/SHOW      Sort: 102      Number: 70  
 Users (Empty = All):      ☒ Include users      ☐ Exclude users

**Trigger**

FormType: 139  
 Item:      Column:      Row:      Menu Uid:      FormMode: All  
 Event: Data Load  
 BeforeAction: Ignore  
 ActionSuccess: True  
 ItemChanged: Ignore  
 InnerEvent: Ignore  
 CharPressed:      Modifier: None

**Condition**

☐ No condition, just execute function        
☐ Simple condition  
 When item | (column) | Comparison type      Equals  
 Value | Type | Execute function      String (0)  
☒ SQL Condition      Common SQL structures

```
IF(SELECT FrozenFor FROM OCRD WHERE CardCode = ${4.0.0}) = 'Y'
BEGIN SELECT 'SHOW' FOR BROWSE
END
ELSE
BEGIN SELECT 'HIDE' FOR BROWSE END
```

Condition result	Function to execute
SHOW	UF-006
HIDE	UF-003

**Comments**

☒ Active

OK      Cancel      Import      Export

## SQL for copy/paste

```
IF(SELECT FrozenFor FROM OCRD WHERE CardCode = ${$4.0.0}) = 'Y'
BEGIN
  SELECT 'SHOW' FOR BROWSE
END
ELSE
BEGIN
  SELECT 'HIDE' FOR BROWSE
END
```

## Making the functionality complete

We now have the basic functionality but there are additional scenarios that we need to handle in order to get a fully professional solution.

- When we switch to Add or Find mode the STOP sign should be hidden (since no data)
- When we choose a customer the STOP-sign should be displayed or not

## Switch to add/find mode

This requires the following configs calling the Hide Macro

**B1 Validation Configuration**

Name:  Sort:  Number:

Users (Empty = All):  ☒ Include users ☐ Exclude users

**Trigger**

FormType:   
 Item:   
 Column:   
 Row:   
 Menu Uid:   
 FormMode:   
 Event:   
 BeforeAction:   
 ActionSuccess:   
 ItemChanged:   
 InnerEvent:   
 CharPressed:  Modifier:

**Condition**

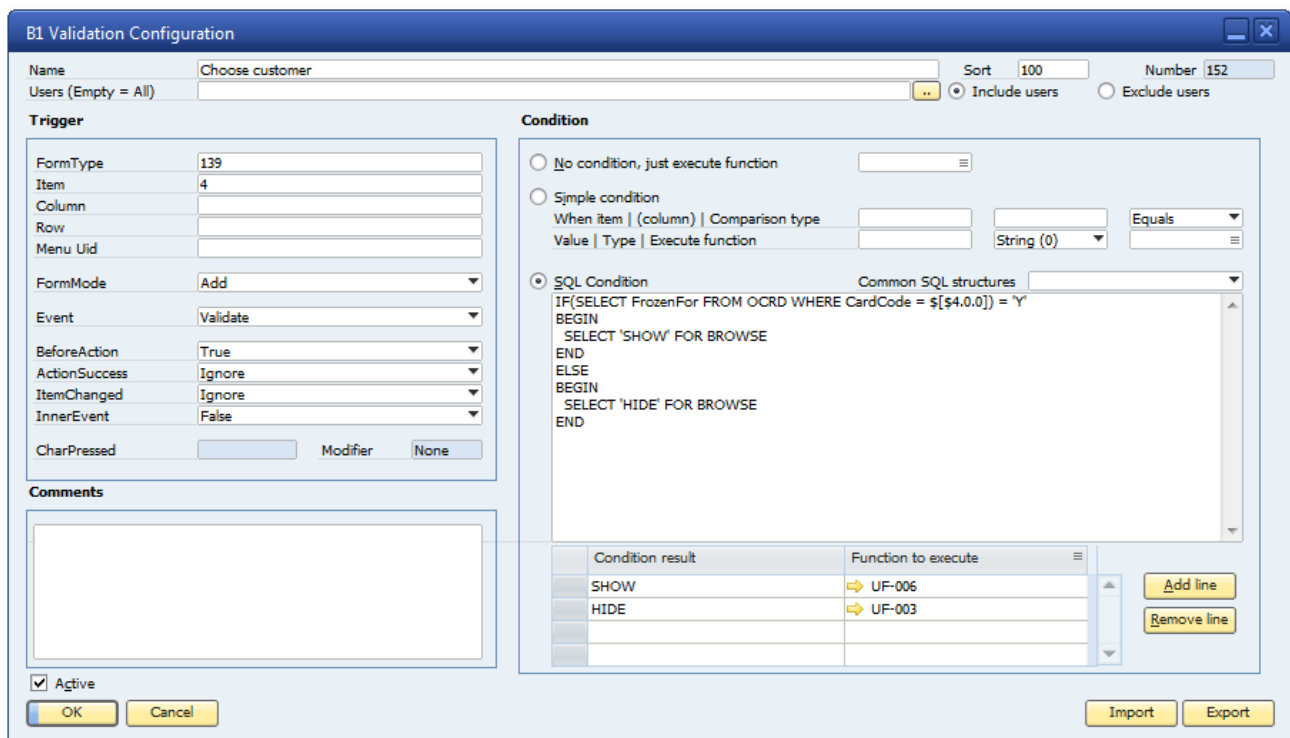
☒ No condition, just execute function   
☐ Simple condition  
 When item | (column) | Comparison type:     
 Value | Type | Execute function:     
☐ SQL Condition

Condition result	Function to execute

☒ Active

## Choose customer

Finally when we leave the customer code field we wish to validate in the same manner as in navigation.



The image shows the 'B1 Validation Configuration' dialog box. It has a title bar with a close button. The main area is divided into several sections:

- Name:** 'Choose customer' (text field)
- Sort:** '100' (text field)
- Number:** '152' (text field)
- Users (Empty = All):** (text field with a dropdown arrow)
- Include users:** (radio button, selected)
- Exclude users:** (radio button)
- Trigger:**
  - FormType: '139' (text field)
  - Item: '4' (text field)
  - Column: (text field)
  - Row: (text field)
  - Menu Uid: (text field)
  - FormMode: 'Add' (dropdown)
  - Event: 'Validate' (dropdown)
  - BeforeAction: 'True' (dropdown)
  - ActionSuccess: 'Ignore' (dropdown)
  - ItemChanged: 'Ignore' (dropdown)
  - InnerEvent: 'False' (dropdown)
  - CharPressed: (text field)
  - Modifier: 'None' (text field)
- Condition:**
  - ☐ No condition, just execute function
  - ☐ Simple condition
    - When item | (column) | Comparison type: (text field) | (text field) | 'Equals' (dropdown)
    - Value | Type | Execute function: (text field) | 'String (0)' (dropdown) | (text field)
  - ☒ SQL Condition
    - Common SQL structures: (dropdown)
    - IF(SELECT FrozenFor FROM OCRD WHERE CardCode = \${4.0.0}) = 'Y'  
BEGIN  
  SELECT 'SHOW' FOR BROWSE  
END  
ELSE  
BEGIN  
  SELECT 'HIDE' FOR BROWSE  
END
- Comments:** (text area)
- Condition result / Function to execute table:**

Condition result	Function to execute
SHOW	UF-006
HIDE	UF-003

Buttons: Add line, Remove line
- Active:** (checkbox, checked)
- Buttons:** OK, Cancel, Import, Export

## Use of functionality

You should now be able to navigate through existing sales-orders and create new ones and always have a visual indication of the Customer Status.