



5 WAYS OF MAKING A FIELD REQUIRED IN MAXIMO

Presented by Stephen Hume, MaxTECH Chair

Agenda

- 1 **Background**
- 2 **Using Database Configuration**
- 3 **Using Application Designer**
- 4 **Using Conditional Expression**
- 5 **Using Data Restrictions**
- 6 **Using Automation Scripts**



Stephen Hume

Stephen has been working with IBM Maximo for over ten years in a variety of industries (Oil and Gas, Utilities). He has taught Maximo courses to end users for both Technical and Functional audiences and chairs the MaxTECH User Group.

Background

Part of the configuration work required for an application in Maximo is to determine which fields are to be made required, and when to make them required. As with many other things in Maximo, there are more than one way to achieve the same result. Each method has conditions and will determine when it is best to use that solution.

Here are the 5 ways to make a field required in Maximo.


Method	Results	Comments
Database Configuration	Attribute always required	Will cause failure of record creation in some cases.
Application Designer	Set attribute to required	Field will always be required in application, could cause an issue if the field is read only.
Conditional Expressions	Use conditional UI in application designer	Sets the field as required depending on other conditions. May also be an issue if the field is read only. Does not work with list tables.
Data Restrictions	Field can be made required based on conditions	Works well with list tables
Automation Scripts	Fields can be made required based on logic, user access, or other complex requirements	Only valid for specific fields (attributes), and can be difficult to validate/test.

1: Database Configuration

Setting an attribute (field) to required in the database configuration is as simple as clicking the REQUIRED check box and then running the database configuration.

Attribute:

Description:

Type: 

Length:

Scale:

Required?

1. You need to make sure that all of the records in the database have data in that attribute before you apply the changes. This may require you to run an SQL update statement to put some value into the field if it is NULL.
2. Doing this will make it impossible to save a new record unless there is data in that field. This means that if it is possible to add a new record through a select action menu in another application, that you have added a dialog to allow the input of the required field.

2: Application Designer

Within application designer it is possible to set a field to required, by opening the properties for that attribute, and setting the input mode to Required.

1. This will mean that if any changes are made to the record it cannot be saved until the required field has a value entered or selected.
2. This can be an issue if the field is hidden or set to read only based on other conditions of the record (status, business requirements, etc).

The screenshot shows a 'Textbox Properties' dialog box with two tabs: 'General' and 'Advanced'. The 'General' tab is active. The following fields are visible:

- Control ID: main_grid3a_col1_1
- Default Label: Summary
- Label: (empty text box)
- Hide Label?:
- Turn Smart Fill Off?:
- Attribute: DESCRIPTION
- Data Source ID: (empty text box)
- Lookup: LONGDESC
- Go To Applications: (empty text box)
- Input Mode: Required (dropdown menu)
- Menu Type: (empty text box)
- Size: 96
- Icon Menu ID: (empty text box)

3: Conditional Expressions

Using a signature option, and conditional expression it is possible to set a field in application designer as “required” depending on specific conditions.

1. This requires that the signature option is authorized for a specific security group (we always use the EVERYONE security group for this).
2. The only issue with this choice is that if there are several conditions that can make the field required, it can be complex to set up.

The screenshot displays the 'Configure Conditional Properties' interface, which is organized into three main sections:

- Signature Option:** Shows 'CNRRELOTH' and a 'Conditional UI: Show release material other' field.
- Security Groups:** A table with columns 'Name', 'Description', and 'Sequence'. It lists 'EVERYONE' with the description 'Role - All Maximo Users (All)' and a sequence of '10'. Buttons for 'Select Groups' and 'New Row' are present.
- Conditions for Security Group EVERYONE:** A table with columns 'Name', 'Description', 'Reevaluate?', and 'Sequence'. It lists 'CNRINCRELOTH' with the description 'Release Material Type = Other' (checked) and a sequence of '10'. Buttons for 'Select Conditions' and 'New Row' are present.
- Property Values for Condition CNRINCRELOTH:** A section with two tabs: 'Condition is true' (selected) and 'Condition is false'. Below the tabs is a table with columns 'Property' and 'Value'. It lists 'inputmode' with the value 'required'. An 'Enter Property' button is at the bottom right.

4: Data Restrictions

In the security groups application it is possible to set an attribute in a specific application as required depending on certain conditions.

It is also possible to set the same attribute as required for all applications where it appears depending on conditions.

This option works well with list tables.

Group: EVERYONE Role - All Maximo Users (All)

Object Restrictions Attribute Restrictions Collection Restrictions

Attributes Filter > 1 - 10 of 51

Object	Attribute	Application	Type	Condition	
			required		>>
▶ CNRSAFETY	DESCRIPTION		REQUIRED	CNRACTREC	>> 🗑️
▶ INCIDENT	LOCATION	CNRINCIDENT	REQUIRED	CNRINCVER	>> 🗑️
▶ INVUSELINE	WONUM	INVUSAGEM	REQUIRED	CNRWOREC	>> 🗑️
▶ INVUSELINE	WONUM	INVUSAGE	REQUIRED	CNRWOREC	>> 🗑️
▶ ITEM	COMMODITYGROUP	ITEM	REQUIRED	CNRDRILITM	>> 🗑️
▶ ITEM	DESCRIPTION	ITEM	REQUIRED		>> 🗑️
▶ MATRECTR/	CNRHOLDLOCATION	RECEIPTS	REQUIRED	CNRONHOL	>> 🗑️
▶ MATUSETR/	WONUM	CNRRETURNS	REQUIRED	CNRWOREC	>> 🗑️
▶ MRLINE	CNRFAO	CREATEDR	REQUIRED	CNRLINETYI	>> 🗑️
▶ MRLINE	CNRSHIPFROM	CREATEDR	REQUIRED	CNRLINETYI	>> 🗑️

New Row

4: Data Restrictions

One example of this is in the Incident application. Set the target completion date to 'REQUIRED' if the actual finish date is NULL. The re-evaluate checkbox means that this condition will be checked each time the record is saved.

The screenshot shows a configuration form titled "Details" with the following fields and values:

- Object: PLUSGACT (Action Tracking)
- Attribute: TARGCOMPDATE (Target Completion)
- Application: CNRINCIDENT (Incidents (CNR))
- Type: REQUIRED
- Reevaluate?
- Condition: CNRFINISH >> Actual Finish
- Condition Class: (empty)
- Expression: :actfinish is null

5: Automation Script

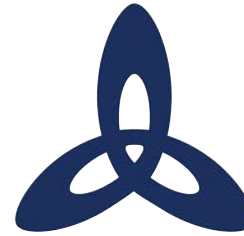
In one project, there were a long list of questions and if the response was “Y” Yes to the question, then three fields needed to be set as required and an error message presented to the user. This was achieved with an automation script (for performance reasons).

This script was used to do what was called “application conditioning” - setting fields to read only, required, depending on the user accessing the record and record status and stage.

```
if not onadd and not mbq.sigopGranted("CNRMOC", "CNRNOCOND") and not mbq.isFlagSet(MboConstants.READONLY):  
  
    if mboname == "CNRMOCANSPHACHK":  
        # When response is Y: CONSEQUENCE, SAFEGUARDS, MITIGATION required  
        if mbq.getString("RESPONSE") == "Y":  
            if mbq.isNull(service):  
                elif 0 <= phaScore <= 1:  
                    # Read only for score 0-1  
                    mbq.setFieldFlag(["PHAMETHOD", "RESIDUALRISK"], MboConstants.READONLY, True) # Reasoning is optional  
                    mbq.setFieldFlag(["PHAMETHOD"], MboConstants.REQUIRED, False)  
            else:  
                # Required for score 2+  
                mbq.setFieldFlag(["DESCRIPTION", "PHAMETHOD"], MboConstants.READONLY, False)  
                mbq.setFieldFlag(["PHAMETHOD"], MboConstants.REQUIRED, True)  
            if mbq.isNull(service):  
            if mbq.isNull(service):  
            if mbq.isNull(service):
```

THANK YOU!

Success! You should have learned 5 ways of making a Field Required in Maximo.



Ask us a technical question:

maxtech@bpdzenith.com

www.bpdzenith.com/maxtech

MaxTECH is the first ever dedicated Maximo Technical User Group aimed at Maximo Administrators, Developers and Technical Support staff.

It is a great place for users to ask and answer technical questions, learn from each other, collaborate and help improve Maximo in your organization.

MaxTECH was founded in 2017 by BPD Zenith and is chaired by Maximo Consultant Stephen Hume. We host several events every year (Calgary, St. Louis, MaximoWorld, Houston, MUWG, Northern California MUG, Maximo UK & Ireland User Group) including digital events.

MaxTEACH is a free online user group designed to go in depth into a Maximo topic.