



[WWW.TECHNODIVINE.COM](http://WWW.TECHNODIVINE.COM)

## IMPROVE PERFORMANCE – WEB APPLICATION

**For Automation Geeks** | By Bharat Kakkar

## Improve Performance – Web Application?

**Question:** How can we increase performance of our scripts for a web application with deep paths (objects within the objects for an example webtablewithin several webtables)?

**Description:** We are using Descriptive programing in our project. From the performance point of view our scripts are very slow even if a single text box has to be populated with a specific text, it takes several minutes to perform the Action. It seems too many similar objects are the reason.

**Question by: Sachin Ratra**

**Answer:** Sachin, the issue was obviously because of several object of similar kind but it wasn't the only issue, there was something very basic which was triggering the issue.

Let's go step by step and see what all may cause delays in QTP scripts and how can we overcome them:

### Step 1 Defining the Objects:

```
Set oBrowser=Description.Create  
oBrowser("title").value="Create Assignment"  
oBrowser("index").value="0"
```

Most of us describe the objects like this only, either we don't know what's wrong with this or we never consider performance aspects of the code we write:

Every object created (using Description.Create method) in QTP has a **RegularExpression** Property, which specifies whether the value of a Property object in the Properties collection is a regular expression or not. And **By default, value of all Property objects added to a Properties collection are treated as regular expressions.**

Which clearly means that if every value would be considered as a regular expression, matching a regular expression would take more time as compare to matching a constant with several values in the available object pool. Although generally we do not experience such issues, but if your object pool grows significantly large you will surely see a significant difference (as in case of your application).

So, it's always advisable to set the **RegularExpression** property to false. Now the question is how would you do it in practical scenario, because it doesn't seem practical to set **RegularExpression** property to false for every single definition. If you will do it for every value you will land up with increasing several

lines of code and code will look like as follows:

```
Set oBrowser = Description.Create  
oBrowser("title").value="Create Assignment"  
oBrowser("title").RegularExpression=false  
oBrowser("index").value="0"  
oBrowser("index").RegularExpression=false
```

Hence, to avoid this you have to do two things:

1. Identify what all objects are eating up most of the time while identification
2. Set **RegularExpression** property to false for these objects, using a custom Function/Procedure, you may write a function like the following one to set all defined TO properties

```
Sub SetRegularExpressionOFF(oObject)
  For i=0 to oObject.Count-1
    sPropertyname=oObject.Item(i).name
    oObject(sPropertyname).RegularExpression =False
  Next
End Sub
```

Above procedure can be called after define the object (only for those objects wherein identification take too long)

```
'Creating an Object for a Browser
Set oBrowser=Description.Create
oBrowser("title").value="Create Assignment"
oBrowser("index").value="0"
SetRegularExpressionOFF oBrowser
```

### Step 2 Optimizing the staements:

Another thing that you guys seem to be missing was the way script was written; every time we call an object for a specific operation we write complete path to access the object. For an example:

```
Line 1: Browser(Create_Assignment).Page(Create_Assignment).WebEdit(oTxtEditBox).Set "Bharat"
Line
2: Browser(Create_Assignment).Page(Create_Assignment).WebEdit(oEmailEditBox).Set "Bharatkakkar@t
echnodivine.com"
```

There are two things which can be Optimize:

1. For access the object on the same path all the parent objects are Identified again example Browser class had been called twice and also been identified twice by QTP. Same has happened with Page class, to overcome this we can use **With** statement.

```
Line 1: With Browser(Create_Assignment).Page(Create_Assignment)
Line 2:   WebEdit(oTxtEditBox).Set "Bharat"
Line 3:   WebEdit(oEmailEditBox).Set "Bharatkakkar@technodivine.com"
Line 4: End With
```

Now Browser and page class will be hit only once (at the beginning of **With** statement)

2. Also, in case of a Web Application it is advisable to keep the Root Class node (Browser) and the first child (Page) in QTP's Object repository, as they are constant in most of the cases (even if not you may update the definition at run time but it's for sure that you see a significant difference in terms of performance.

I hope this Article would have helped you up to your expectation.

**Further, if you need more performance related tips you may refer to "Tweaks for better Performance" Article**



**Regards,**  
**Bharat Kakkar**