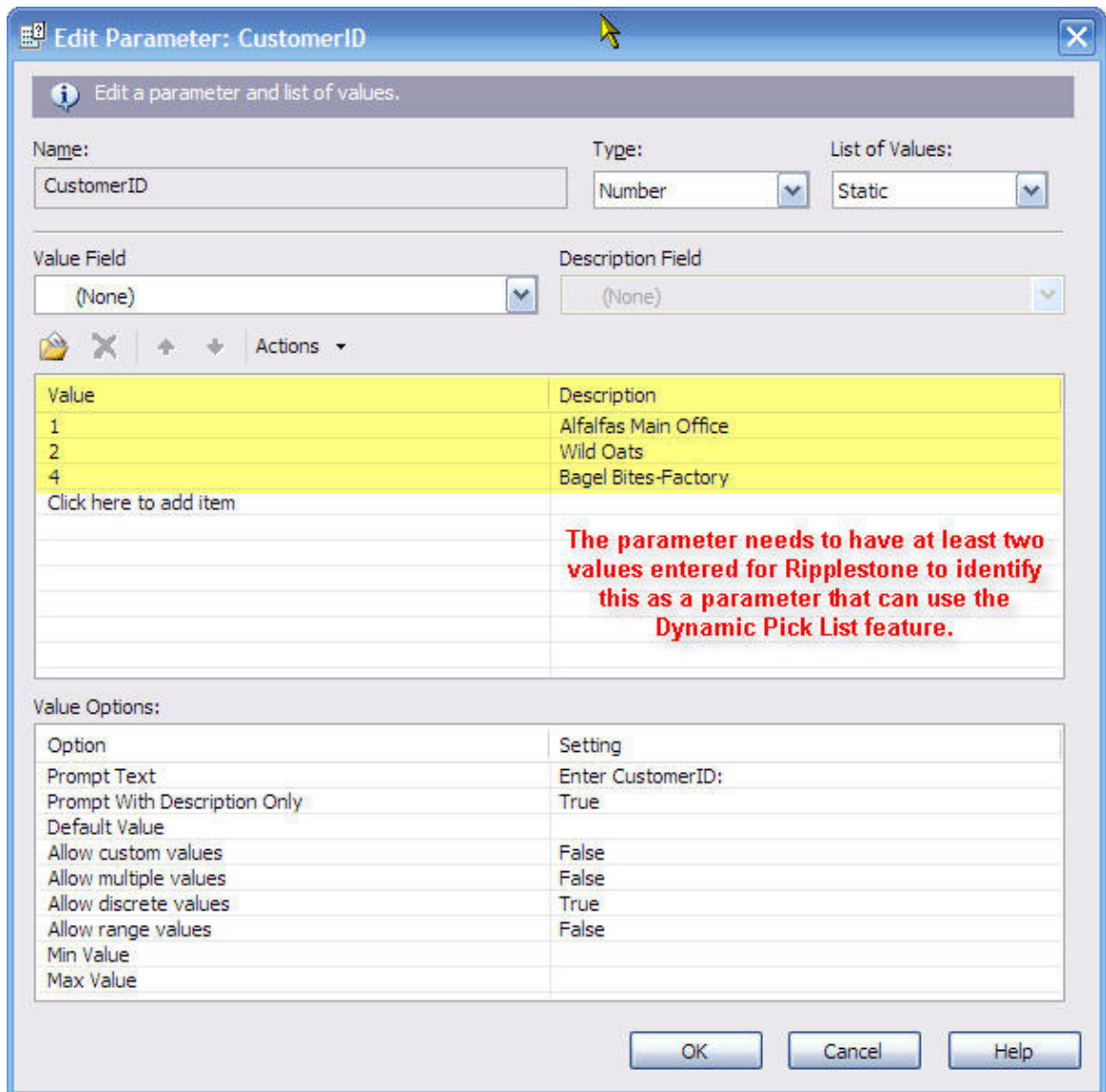


How to use the @UserID token with a Dynamic Pick List

This feature allows a report that is using a Dynamic Pick List to filter the contents of the pick list using the currently logged in user's ID. This allows you to use one report that filters the parameter selection based on the UserID of the logged in user and can protect data from being viewed by the wrong person.

This document will demonstrate the process to create a report that uses the @UserID token.

1. The report that is published needs to have at least one parameter that can be used for a Dynamic Pick List. The image below shows a parameters that can be used since it has at least two static values to select from.



Edit Parameter: CustomerID

Edit a parameter and list of values.

Name: CustomerID Type: Number List of Values: Static

Value Field: (None) Description Field: (None)

| Value | Description |
|------------------------|----------------------|
| 1 | Alfalfas Main Office |
| 2 | Wild Oats |
| 4 | Bagel Bites-Factory |
| Click here to add item | |

The parameter needs to have at least two values entered for Ripplestone to identify this as a parameter that can use the Dynamic Pick List feature.

Value Options:

| Option | Setting |
|------------------------------|-------------------|
| Prompt Text | Enter CustomerID: |
| Prompt With Description Only | True |
| Default Value | |
| Allow custom values | False |
| Allow multiple values | False |
| Allow discrete values | True |
| Allow range values | False |
| Min Value | |
| Max Value | |

OK Cancel Help

Figure 1 – Crystal Reports Edit Parameter Dialog

2. After you have published the report, open the Edit Published Document and click on the Dynamic Pick Lists tab as shown in figure 2

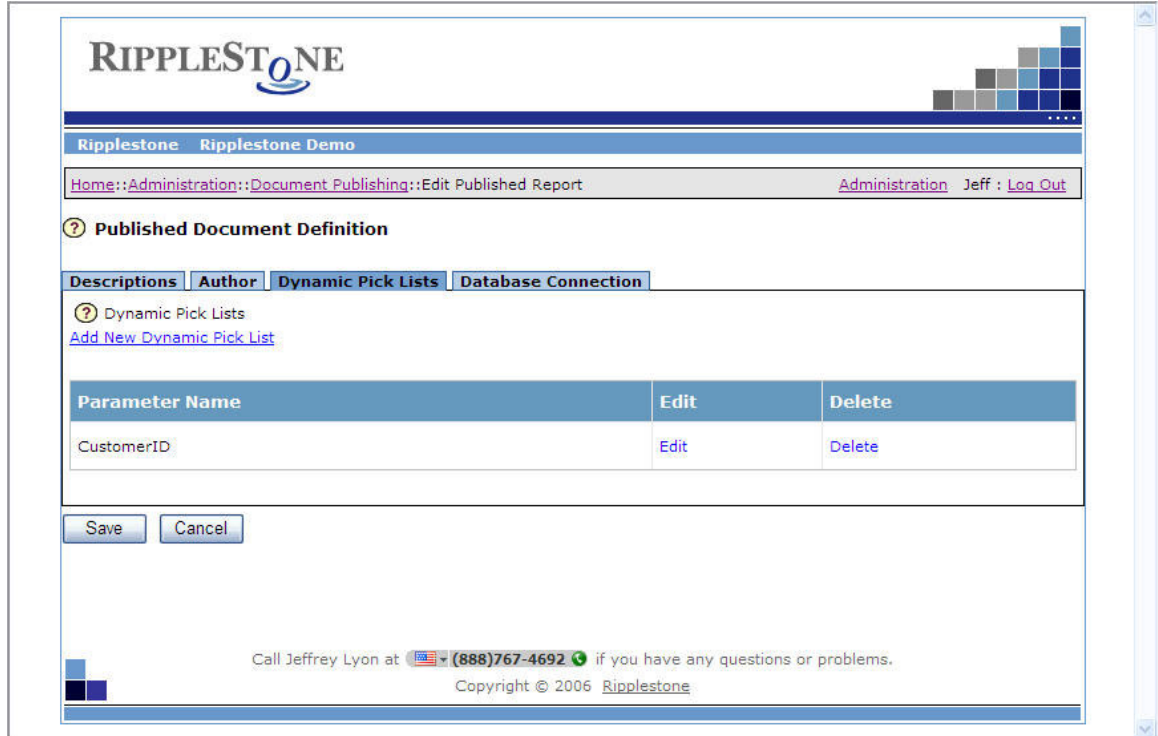


Figure 2 – Edit Published Document Page

- Click on Add for a new Dynamic Pick List or Edit if you have already added one. This will open the Edit Dynamic Pick List page where you can add the @UserID token to the Pick List SQL. The page will display the parameters that are identified within the report that can be used for Dynamic Pick Lists. The Database Connection dropdown will display the Dynamic Pick List connections that have been created in the Database Connection page. You will need to create a Database Connection to the data source so Ripplestone can connect to the data source to run the SQL.

You can then test the SQL by entering a UserID into the Test UserID textbox.

Click OK to save the Dynamic Pick List once you have the SQL entered correctly.

The screenshot shows the 'Edit Dynamic Pick List' page. The fields are as follows:

- Folder: Pulse
- Report: CustomerDetail.rpt
- Parameter Name: CustomerID
- Database Connection: Pulse_Production
- Pick List SQL:


```
SELECT Customers.CustomerID AS ValueColumn, Customers.CustomerName AS
DescriptionColumn
FROM Customers INNER JOIN Users ON Customers.SalesManager = Users.UserID
WHERE Users.FirstName = @UserID ORDER BY Customers.CustomerName
```
- Test Pick List: Bleu Sun Market & Caterers
- Test UserID: Jeff

Red arrows point to the @UserID token in the SQL and the Test UserID field. The text next to the arrows reads: "The @UserID token is added to filter the Pick List using the UserID of the currently logged in user." and "The SQL can be tested by entering a UserID into the Test UserID textbox and then clicking the Test Pick List link."

Figure 3 – Edit Dynamic Pick List Page