ServiceNow

Technical Best Practices

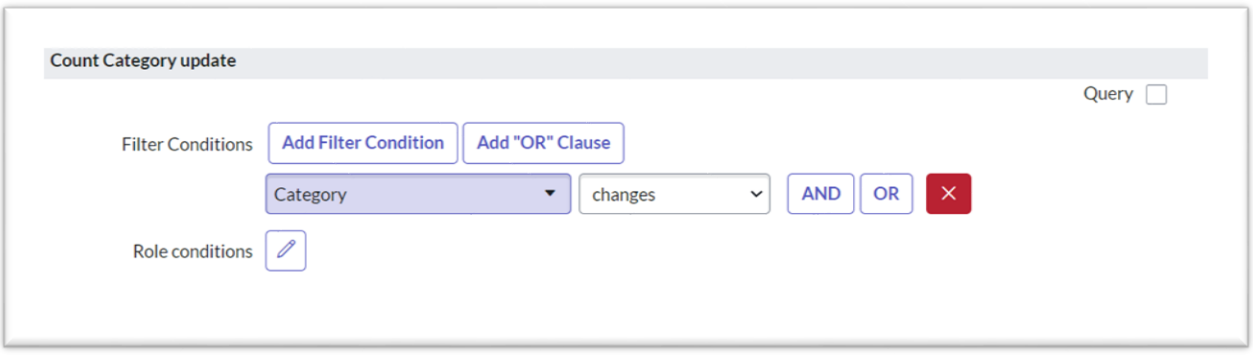
Business Rule Technical Best Practices

Know when to run the Business Rule

Value	Use Case
display	Use to provide client-side scripts access to server-side data.
before	Use to update information on the current object. For example, a Business Rule containing current.state=3; would set the State field on the current record to the state with a value of 3.
after	Use to update information on related objects that need to be displayed immediately, such as GlideRecord queries.
async	Use to update information on related objects that do not need to be displayed immediately, such as calculating metrics and SLAs.

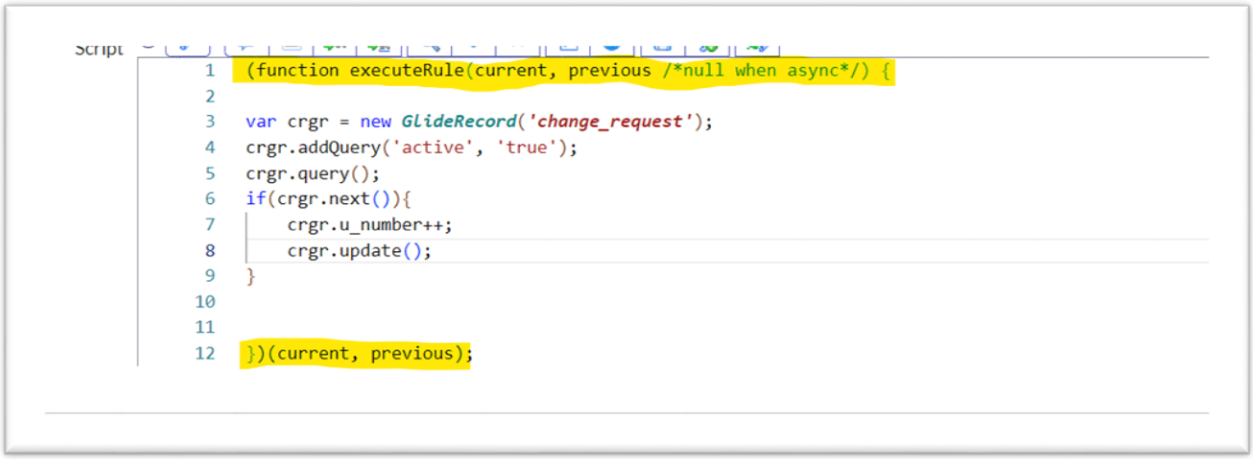
Use Conditions in Business Rule

- Since Business Rules are evaluated whenever an insert, update, delete or query action is made to a record, it is important to ensure you are using conditions.
- Conditions are evaluated before the rule is executed, if the condition is met, the script is evaluated and executed. If there is no condition, the system assumes that the Business Rule should be evaluated and executed for every action to a record on the specified table of the Business Rule.
- It is easier to debug Business Rules when you can see which one meet a particular condition and which do not.



Keep Code in Functions

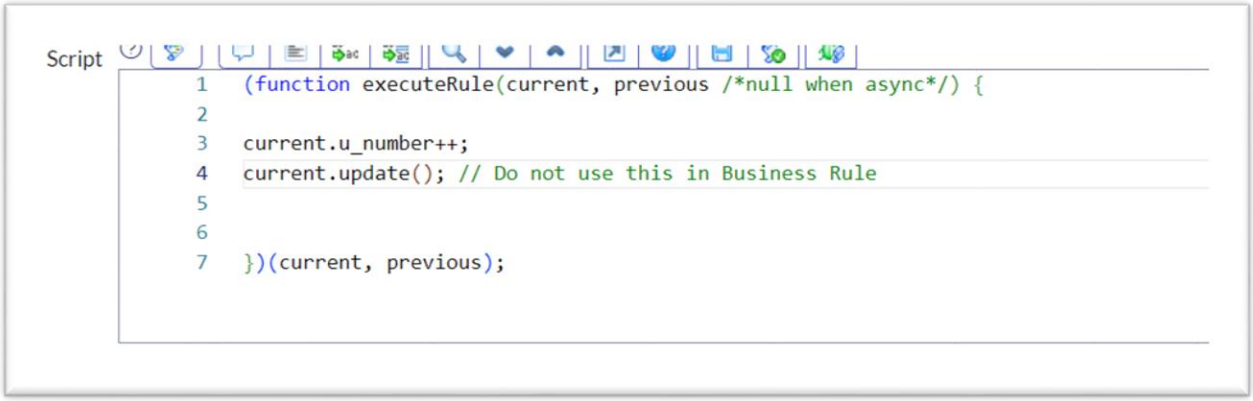
- By default, an advanced Business Rule will wrap your code in a function, and it is important that this guideline is followed.
- When code is not enclosed in a function, variables and other objects are available to all other server-side scripts.
- This availability can lead to unexpected consequences that are difficult to troubleshoot.



Prevent Recursive Business Rule

- Do not use current.update() in a Business Rule script. The update() method triggers Business Rules to run on the same table for insert and update operations, potentially leading to a Business Rule calling itself over and over.
- Changes made in before Business Rules are automatically saved when all before Business Rules are complete, and after Business Rules are best used for updating related, not current, objects.
- When a recursive Business Rule is detected, ServiceNow stops it and logs the error in the system log. However, this behavior may cause system performance issues and is never necessary.
- You can prevent recursive Business Rules by using the setWorkflow() method with the false parameter, current.setWorkFlow(false).
- This will stop Business Rules and other related functions from running on this database access.

- The combination of the update() and setWorkflow() methods is only recommended in special circumstances where the normal before and after guidelines mentioned above do not meet your requirements.



Use Script Include instead of Global Business Rule

- A global Business Rule is any Business Rule where the selected Table is Global. Any other script can call global Business Rules.
- Global Business Rules have no condition or table restrictions and load on every page in the system. Most functions defined in global Business Rules are fairly specific, such as an advanced reference qualifier on one field of one form.
- There is no benefit to loading this kind of script on every page.
- Script includes only load when called. If you have already written a global Business Rule, move the function definition to a Script Include.