

Building a Backdoor

Strategies for Bypassing Validation Rules, Flows, and Triggers

What makes this backdoor different than the backdoor used by Sam and Frodo to sneak into Mordor, in order to destroy the one ring, while being led by Gollum, who was secretly leading them into the lair of Shelob the spider, in order for her to kill Sam and Frodo so Gollum could take the one ring for himself, is that my backdoors have less bugs in them.

Alternatively this session is called:

Slipping Past the Black Gate

The tunnel to Cirith Ungol

On LinkedIn I promised LotR references during this talk. I don't know why I did that.

So, here we go.

clip of Gollum

DemandChain SILVER SPONSOR!

Obsessed With Your Salesforce Success!

- ✓ 1,800+ Projects; 400+ Clients
- ✓ 5.0/5 Customer Satisfaction Rating
- ✓ Manufacturing Focused
- ✓ MN based – National Coverage
- ✓ 9x MSPBJ Best Places to Work!

Map of the United States with stars indicating national coverage.

Partnerships: Salesforce Partner Since 2008, IBM Business Partner, Best Places to Work 2023, 9th Consecutive Year.

MIDWEST DREAMIN' 2023

Meet Our Expert Speakers!

Scan to learn more about DemandChain

JOHN GRAF GREAT LAKES A Thursday, August 17 9:20AM - 10:30AM	ADRIENNE CARR LAKE OF THE ISLES Thursday, August 17 2:40PM - 3:20PM
AHMAD HELAL LAKE BEMIDJI Thursday, August 17 1:00PM - 1:40PM	ROBIN MUSICK LAKE SUPERIOR B Thursday, August 17 3:30PM - 4:10PM
JOSH SATERDALE LAKE SUPERIOR A Thursday, August 17 1:50PM - 2:30PM	BEN SOLOPNE GREAT LAKES A Friday, August 18 9:20AM - 10:00AM
	CAITLIN DEGIORGIO GREAT LAKES A Friday, August 18 9:20AM - 10:00AM

Disclaimer

I don't know what I'm talking about.

I believe that we're all just making it up as we go along. no one has learned it all, we all have something to learn from others.

What am I actually talking about

Validation Rules, record-triggered Flows, and Apex Triggers are essential tools for maintaining data cleanliness and automating business processes. While great for our users, these tools can get in the way of data loads, integrations, and record maintenance.

- bypass things per user/profile

- bypass things per record

- bypass automations with automations

Custom Settings Can Sneak Past Anything

Tricksy little Hobbites

Bypassing on a user/profile basis

ask for show of hands on data loads

Custom Setting Definition
Bypasses [Help for this Page](#)

Create the fields for your custom setting. The data in these fields are cached with the application.

Custom Setting Definition Detail		Edit	Delete	Manage	Object Name	Bypasses
Label	Bypasses				Setting Type	Hierarchy
API Name	Bypasses__c				Description	Contains bypasses for validation rules, apex, etc.
Visibility	Public				Created Date	8/17/2023, 11:54 AM
Namespace Prefix					Record Size	150
Last Modified Date	8/17/2023, 11:54 AM					

Action	Field Label	API Name	Data Type
Edit Del	Bypass_Emails	IsBypassEmails__c	Checkbox
Edit Del	Bypass_Invoicing_Processes	IsBypassInvoicing__c	Checkbox
Edit Del	Bypass_That_One_Email	IsBypassOneThing__c	Checkbox
Edit Del	Bypass_Triggers	IsBypassTriggers__c	Checkbox
Edit Del	Bypass_VR	IsBypassVR__c	Checkbox

create custom setting with a type of hierarchy. add checkboxes for each thing you want to bypass. "all account triggers" or "sales opportunity stage automation"

Bypasses Edit

Provide values for the fields you created. This data is cached with the application.

Edit Bypasses [Save](#) [Cancel](#)

Bypasses Information

Location:

Bypass Triggers:

Bypass WF:

Bypass VR:

Bypass Emails:

Bypass Invoicing Processes:

add users or profiles. system admins, data loaders, integration users

Validation Rule Example

✓ **Good**

```
!$Setup.Bypasses__c.IsBypassVR__c && Name = "Frodo"
```

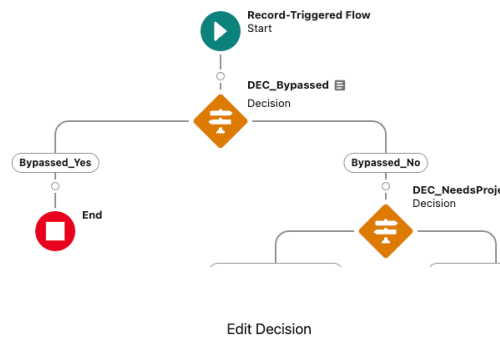
Bad

```
$Profile.id <> '00eo000000KXdC' && SomeField__c = "There and Back Again"
```

Also Bad

```
$Profile.Name <> 'System Admin' && SomeField__c = "There and Back Again"
```

Flow Example



Start flows with a check for this custom setting

ssed) [?]

s a flow bypass enabled. If so, end the flow.

v can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

OUTCOME DETAILS

* Label * Outcome API Name

Condition Requirements to Execute Outcome

All Conditions Are Met (AND)

Resource	Operator	Value
<input type="text" value="\$Setup.Bypasses__c.IsBypassFlows__c"/>	<input type="text" value="Equals"/>	<input type="text" value="True"/>

+ Add Condition

When to Execute Outcome

If the condition requirements are met

Only if the record that triggered the flow to run is updated to meet the condition requirements

Apex Example

```
String uId = UserInfo.getUserId();
Bypasses__c bypassSetting = Bypasses__c.getValues(uId);

if (!bypassSetting.IsBypassTrigger__c) {
    System.debug('something!');
}
```

I have a recording of someone using a bypass

clip of Bilbo

actual footage of Bilbo Baggins bypassing your triggers

Custom Permission

Use those fancy Permission Sets

can also use custom perms for this. reference in VR

Per Record Validation Bypass

We are Gandalf the White

Where we get into ignoring validation rules on-demand.

time stamp on records. check within VR. add to automation. add to integration
clip of Gandalf and Theoden

We'll say that Theoden is the validation rule in this scenario, telling users they have no power.

```
AutomationBypassDateTime__c
```

- *date/time field*

```
IsAutomationBypassed__c
```

- *formula, checkbox*

```
AutomationBypassDateTime__c > NOW() - 0.00005787037 /*Five Seconds*/
```

lots of variations of this floating around. credit Jen Lee for this specific setup

Flow Example

New Assignment

* Label: Bypass VR * API Name: BypassVR

Description:

Set Variable Values

Each variable is modified by the operator and value combination.

Variable	Operator	Value
AutomationBypassDateTime__c	Equals	Current DateTime

+ Add Assignment

Cancel Done

jetstream Developer josh@waterdata.com

2 Objects selected

Type	Label	Name	Help Text	Description
Date Time	AutomationBypassDate	AutomationBypassDate		
Formula	IsAutomationBypassed	IsAutomationBypassed		

AutomationBypassDate: Formula Type: Blanks As: Checkboxes

IsAutomationBypassed: Formula: AutomationBypassDate > NOW() - 0.00005787037 /*Five Seconds*/

pitch jetstream because it's amazing

Quick detour through the Mines

Naming Conventions

naming things well is useful when building these patterns. being consistent helps you and whoever comes after you

Naming Validation Rules

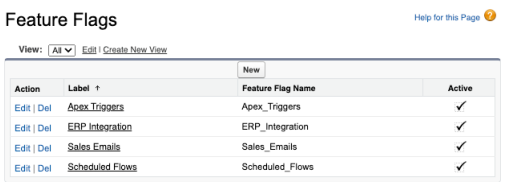
Rule name: `AcctVR01_LockAccountName`

Error

Sales Reps are not allowed to change an account's name [AcctVR01]

Use codes like this to keep track of your rules when you, users, and automations get VR errors

Feature Flags



Action	Label ↑	Feature Flag Name	Active
Edit Del	Apex Triggers	Apex_Triggers	✓
Edit Del	ERP Integration	ERP_Integration	✓
Edit Del	Sales Emails	Sales_Emails	✓
Edit Del	Scheduled Flows	Scheduled_Flows	✓

Feature flag custom metadata can be used to enable/disable specific business processes system-wide. Could be used as a toggle for all Apex triggers, all flows, etc. Disable all automated Sales emails temporarily.

Implement with custom settings, like above, to give a new feature you're working on to only certain users or profiles.

Triggers on Triggers

What's triggers, precious?

bypassing other automations within trigger/flow transactions is hard. weird workarounds. could use the timestamp switch and check for that

Trigger Frameworks

***Come, Mr. Frodo. I can't carry it for you, but I can carry you!* – Sam Gamgee**

analogies are really breaking apart here. don't reinvent the wheel. let others carry the load

Trigger Actions Framework

One that I really like is the trigger actions framework, by Mitch Spano.

Trigger Action

[Help for this Page](#)

Trigger Action Detail		Edit	Delete	Clone
Label	TA_Opportunity_StageInsertRules_BI	Protected Component <input type="checkbox"/>		
Trigger Action Name	TA_Opportunity_StageInsertRules_BI	Namespace Prefix		
▼ Trigger Action Information				
Apex Class Name	TA_Opportunity_StageInsertRules	Required Permission		
Order	1,000	Bypass Permission		
Flow Name		Bypass Execution <input type="checkbox"/>		
Allow Flow Recursion?	<input type="checkbox"/>			
▼ Context (Choose One)				
Before Insert	OnOpportunity	After Insert		
Before Update		After Update		
Before Delete		After Delete		
		After Undelete		
▼ Description				
Description	Prevents an opportunity from being inserted when the StageName is not "Prospecting".			
Created By	User\User, 3/19/2023, 6:50 PM	Last Modified By User\User, 3/19/2023, 7:12 PM		
Edit Delete Clone				

Bypass per Transaction

bypass all actions on an sObject as well as specific Apex or Flow actions for the remainder of the transaction using Apex or Flow.

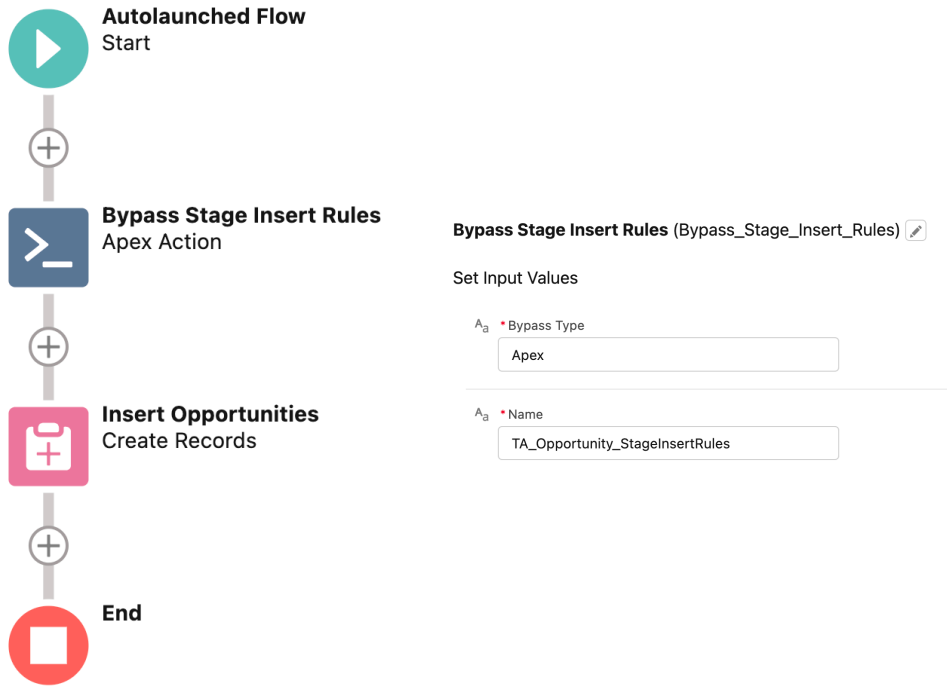
Bypass from Apex

Use the `static bypass(String name)` method in the `TriggerBase`, `MetadataTriggerHandler`, OR `TriggerActionFlow` classes.

```
public void insertOpportunitiesNoRules(List<Opportunity> opportunitiesToInsert) {  
    MetadataTriggerHandler.bypass('TA_Opportunity_StageInsertRules');  
    insert opportunitiesToInsert;  
    MetadataTriggerHandler.clearBypass('TA_Opportunity_StageInsertRules');  
}
```

Bypass from Flow

Use the `TriggerActionFlowBypass.bypass` invocable method. You can set the `Bypass Type` to `Apex`, `Object`, OR `Flow`, then pass the API name of the sObject, class, or flow you would like to bypass into the `Name` field.



...and more!

bypass globally, bypass per user, bypass based on custom permissions. pretty robust tool when you have an advanced use case

Resources

- [Jennifer Lee](#)
- [Jetstream](#)
- [SFXD Wiki](#)
- [Trigger Actions Framework](#)

Josh Saterdalen

Salesforce Consultant

Minneapolis, MN

6x Certified Salesforce Consultant with 5 years of Sales Cloud experience.

- [Blog](#)
- [LinkedIn](#)
- [Threads](#)
- [Mastodon](#)
- [Twitter](#)
- [Github](#)

👉 Free RAM!

saterdalen.me/mwd

