Power Platform COMMUNITY CONFERENCE

emPOWER everyone

SEPTEMBER 18–20, 2024 • *Workshops: Sept 16, 17 & 21*

MGM GRAND • *Las Vegas, NV*

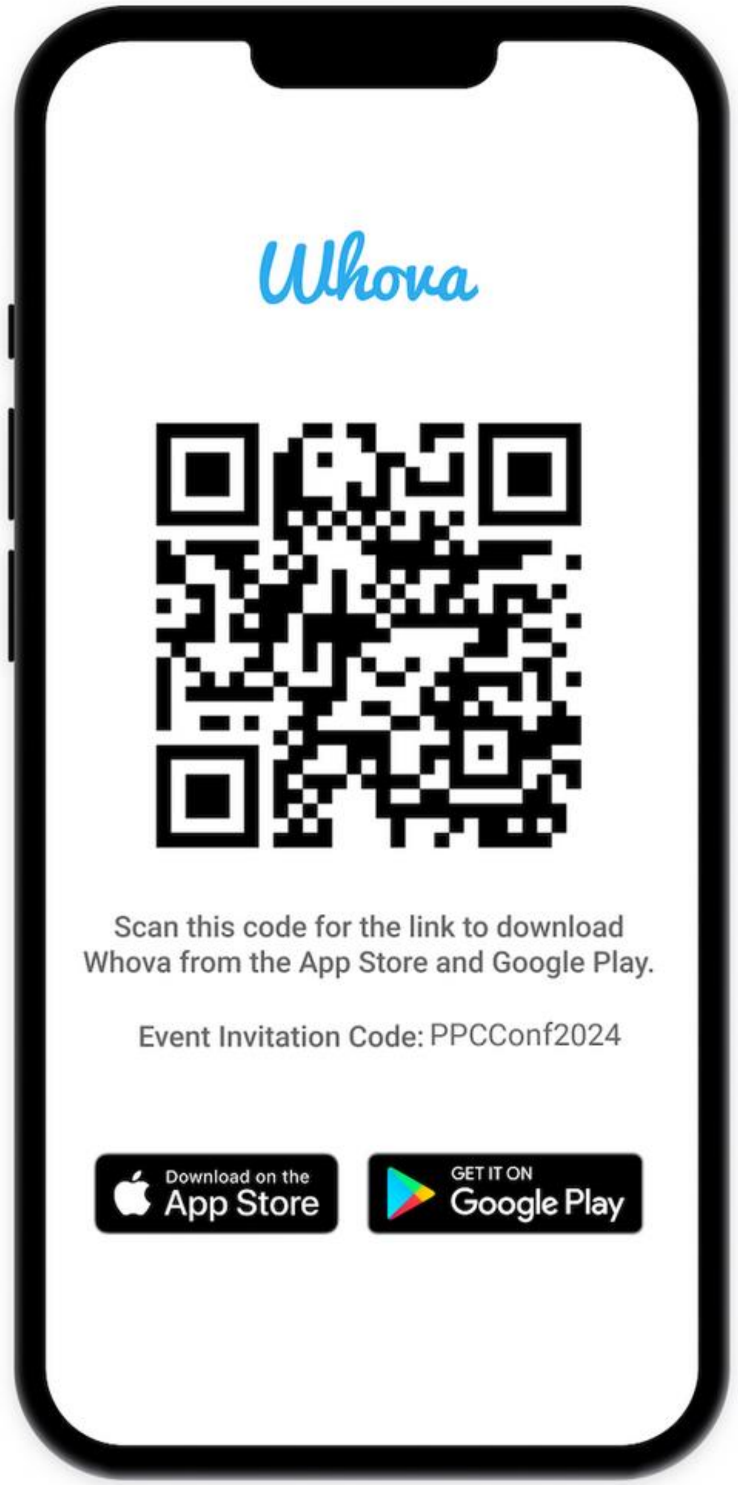
PowerPlatformConf.com #PPCC24 @PowerPlatConf emPOWER everyone

Power Platform
COMMUNITY CONFERENCE

ERROR PROOF YOUR AUTOMATIONS:
A Deep Dive Into Advanced Error Handling and Logging in Power Automate Desktop

Agnius Bartninkas
ab@robovirgin.lt

Whova The official event app for the **Power Platform Community Conference**



Whova

Scan this code for the link to download Whova from the App Store and Google Play.

Event Invitation Code: PPCConf2024

Join the event app to access:

- Event announcements
- Personalized agenda, session details
- Speaker & attendee profiles
- Networking, meet-ups, messages
- Event documents

Event Invitation Code: PPCConf2024

The guy in front of you

Agnius Bartninkas

COO and Co-Founder @ **Definra**

- Microsoft Business Applications MVP (Power Automate).
- The most experienced PAD (formerly Softomotive) user in Lithuania. Most other users were trained by me.
- Used to be the Excel guru of my office, but now am the grumpy guy telling everyone to stop using Excel.
- A “certified” beer expert. Now an aspiring cocktail makAuthor of the PADFramework.er.
- Author of the PADFramework.

in www.linkedin.com/in/agnius-bartninkas

MVP Microsoft® Most Valuable Professional

Agenda

Some “theory”:

- Types of error handling available in PAD
- Options available for logging errors

Demos:

- Different ways to handle errors
- The different options for logging error messages
- Logging runtime non-error messages
- Taking screenshots on error
- Monitoring logs in ELK

Key takeaways

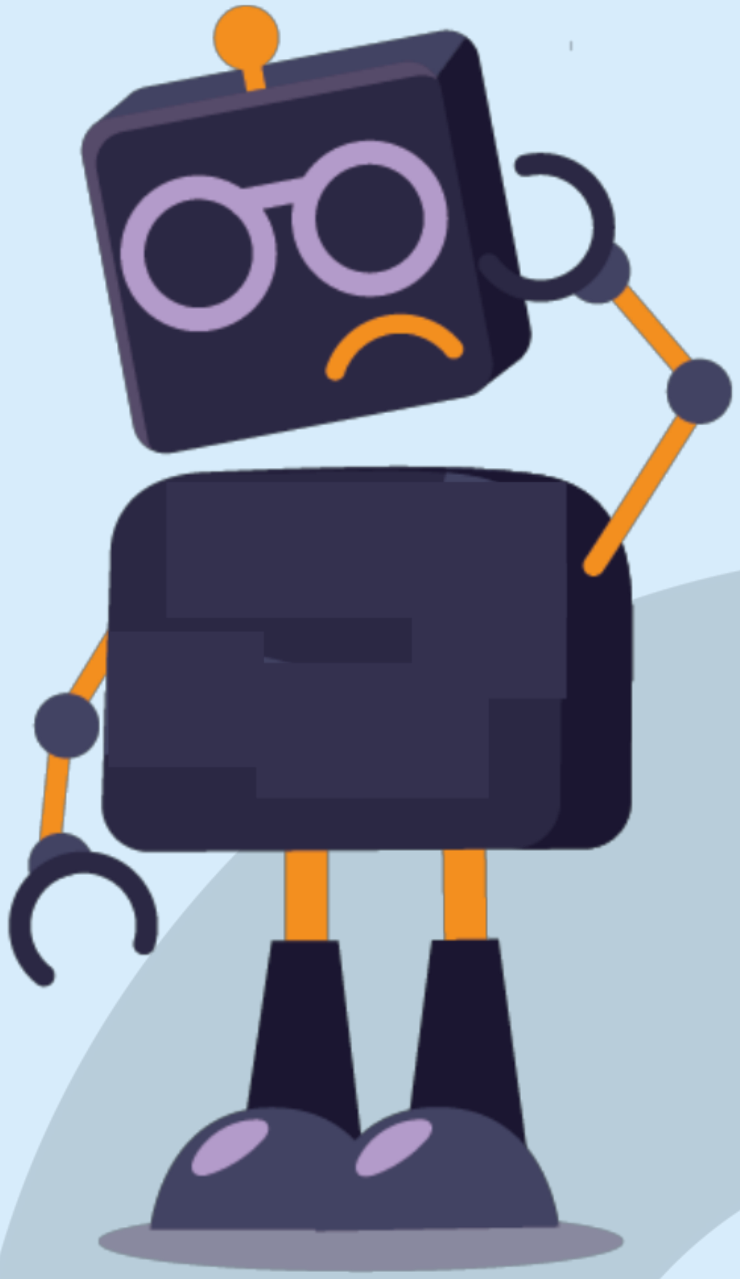
Q&A

Why Error Handling?

Errors can and **will** occur in your flows.

There‘s no official template for handling it in PAD.

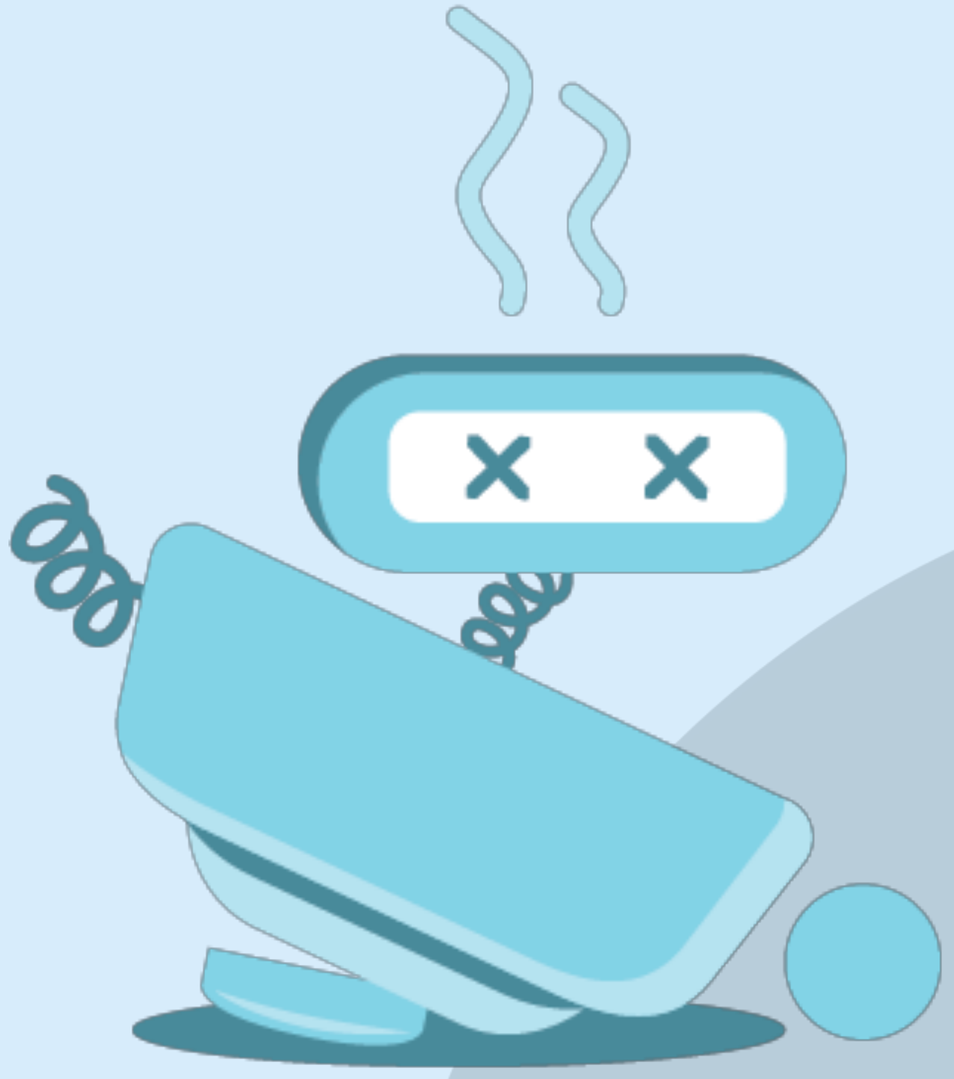
Open-sourced PADFramework available here:
https://github.com/AgniusBartninkas/power-automate-desktop-framework



Types of Error Handling

The main options on how to handle errors in PAD are:

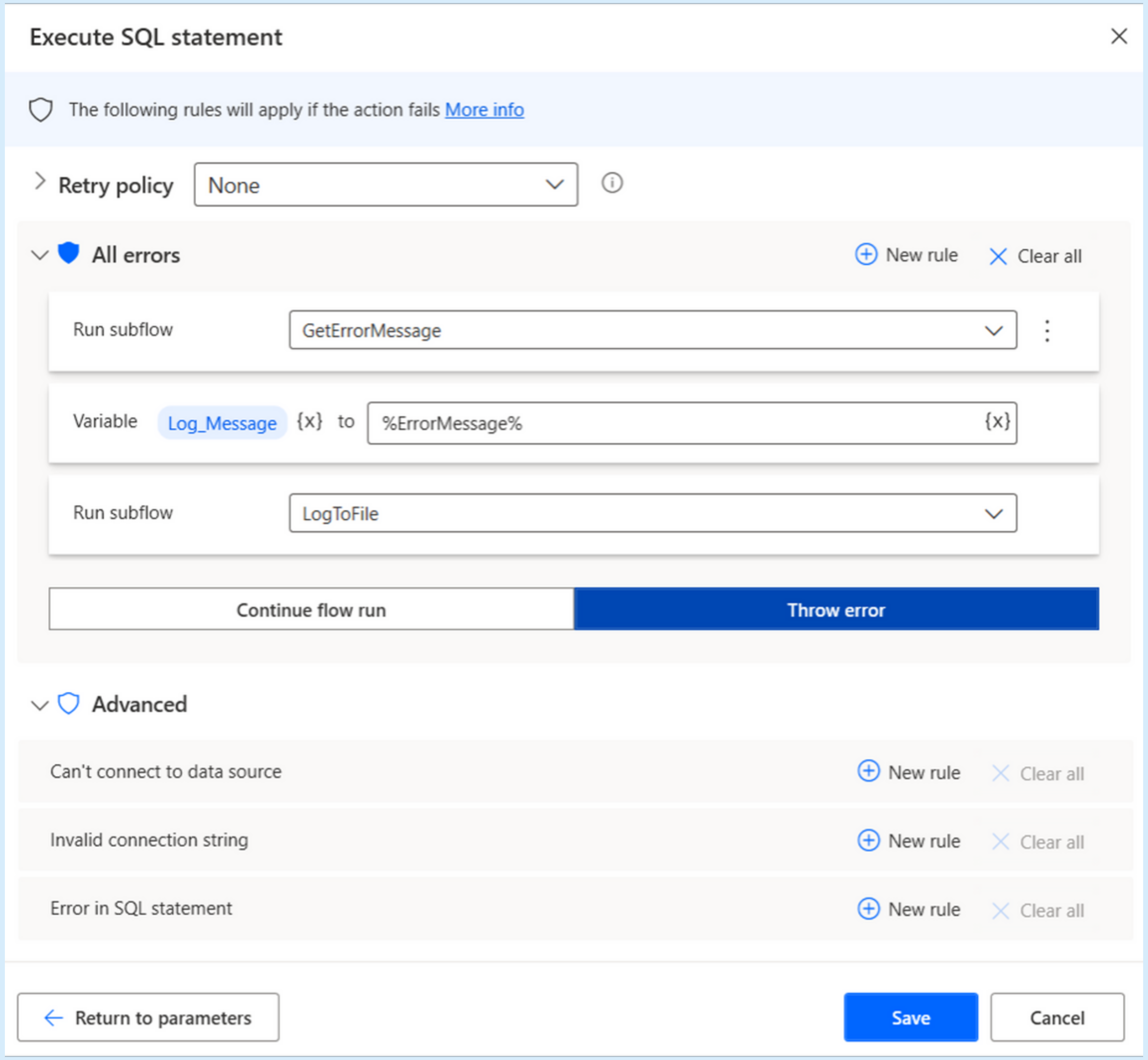
- **None** – the flow fails on any error (not recommended)
- **Action level** – special rules set on each action (not recommended, except for special cases)
- **Error blocks** – setting rules for blocks of actions or entire sub-flows (recommended)



Action Level Error Handling

+ Very powerful, lots of rules

- Time-consuming to build and maintain
- Does not apply to all actions
- Cannot handle unexpected logic errors



Execute SQL statement

The following rules will apply if the action fails More info

Retry policy: None

All errors

Run subflow: GetErrorMessage

Variable Log_Message to %ErrorMessage%

Run subflow: LogToFile

Advanced

Can't connect to data source

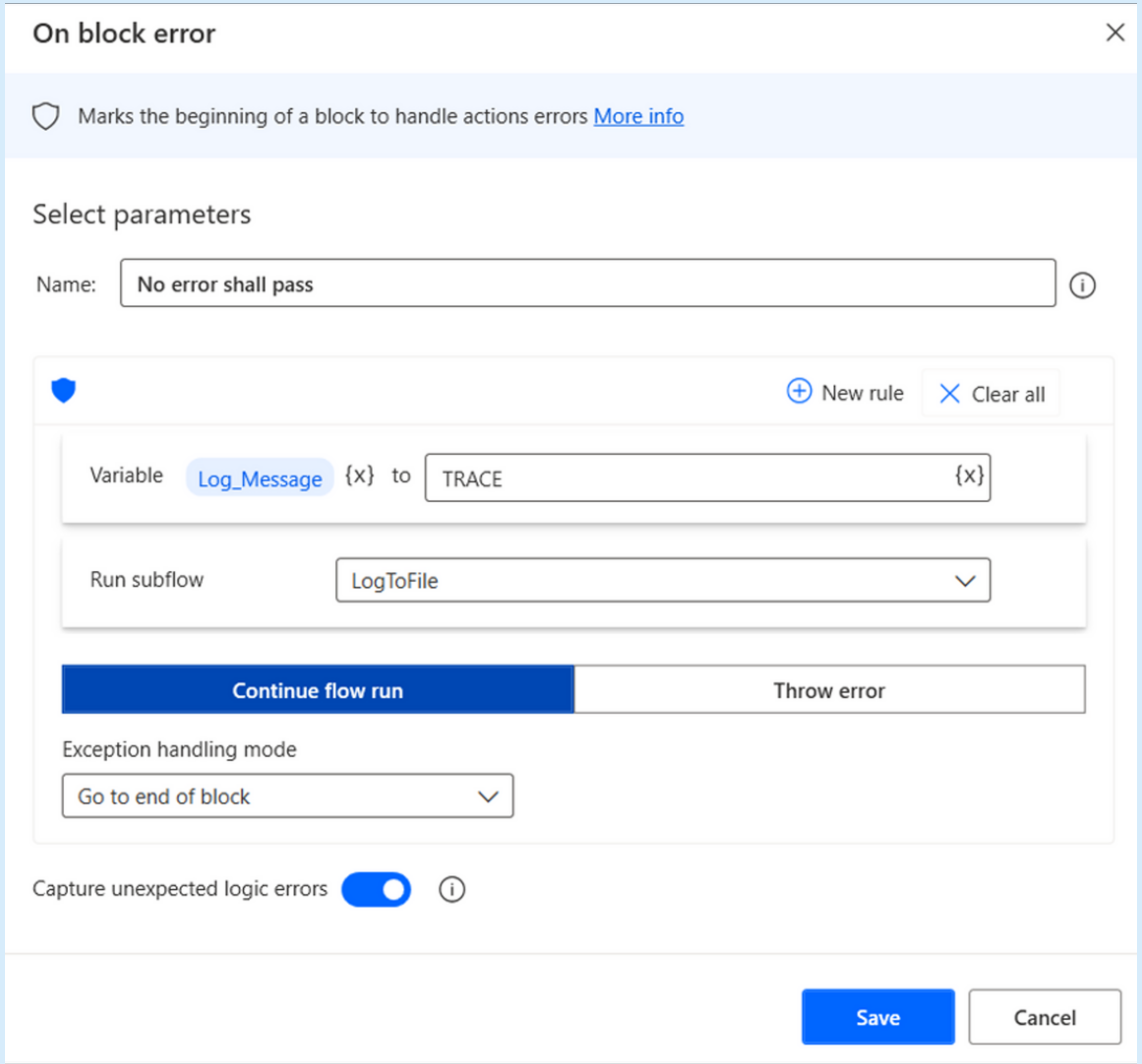
Invalid connection string

Error in SQL statement

Error Blocks

+ Very powerful
+ Easy to set up and maintain
+ Can handle all exceptions

- Less options
- Same rules apply to all actions



On block error

Marks the beginning of a block to handle actions errors More info

Select parameters

Name: No error shall pass

Variable Log_Message to TRACE

Run subflow: LogToFile

[✓ Continue flow run] Throw error

Exception handling mode: Go to end of block

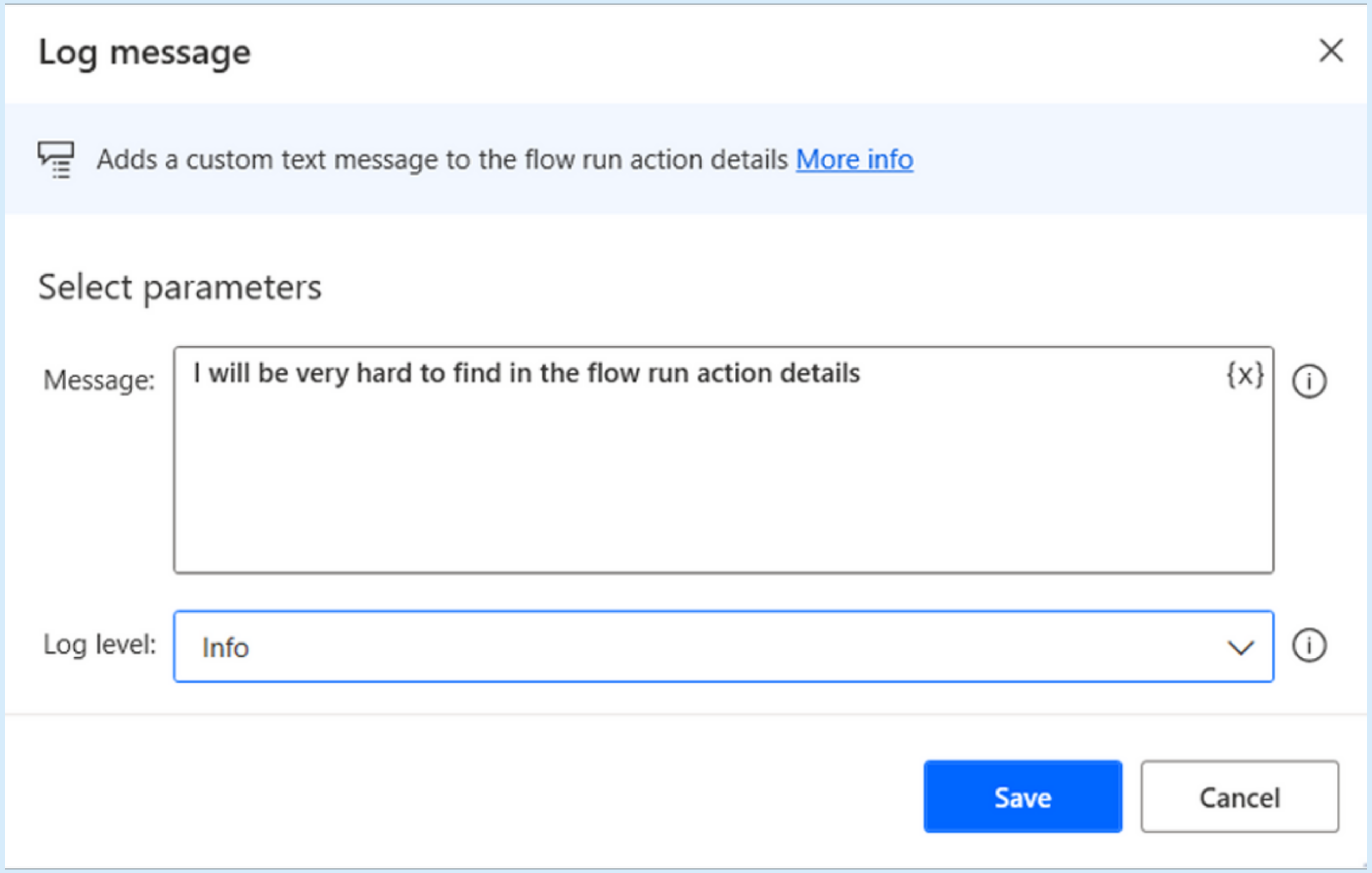
Capture unexpected logic errors: On

Logging Errors

Handling errors is cool, but without logs, it will usually be quite hard to find the issue.

Currently, PAD supports the **Log message** action.

While **Log message** is better than no logging, any custom logging alternative is better than using **Log message**.



Log message

Adds a custom text message to the flow run action details More info

Select parameters

Message: I will be very hard to find in the flow run action details

Log level: Info

Better Ways to Log Errors

Some options that I’ve seen:

- Logging to **SQL databases** – great for techies, might be harder to use for non-tech people.
- Logging to **SharePoint lists** or **Dataverse tables** – better for non-tech people
- Logging to **CSV files** – easy to implement, but a bit limited
- Logging to **plain text files** – very easy to implement and easy to use

Logging to the Plain Text Files

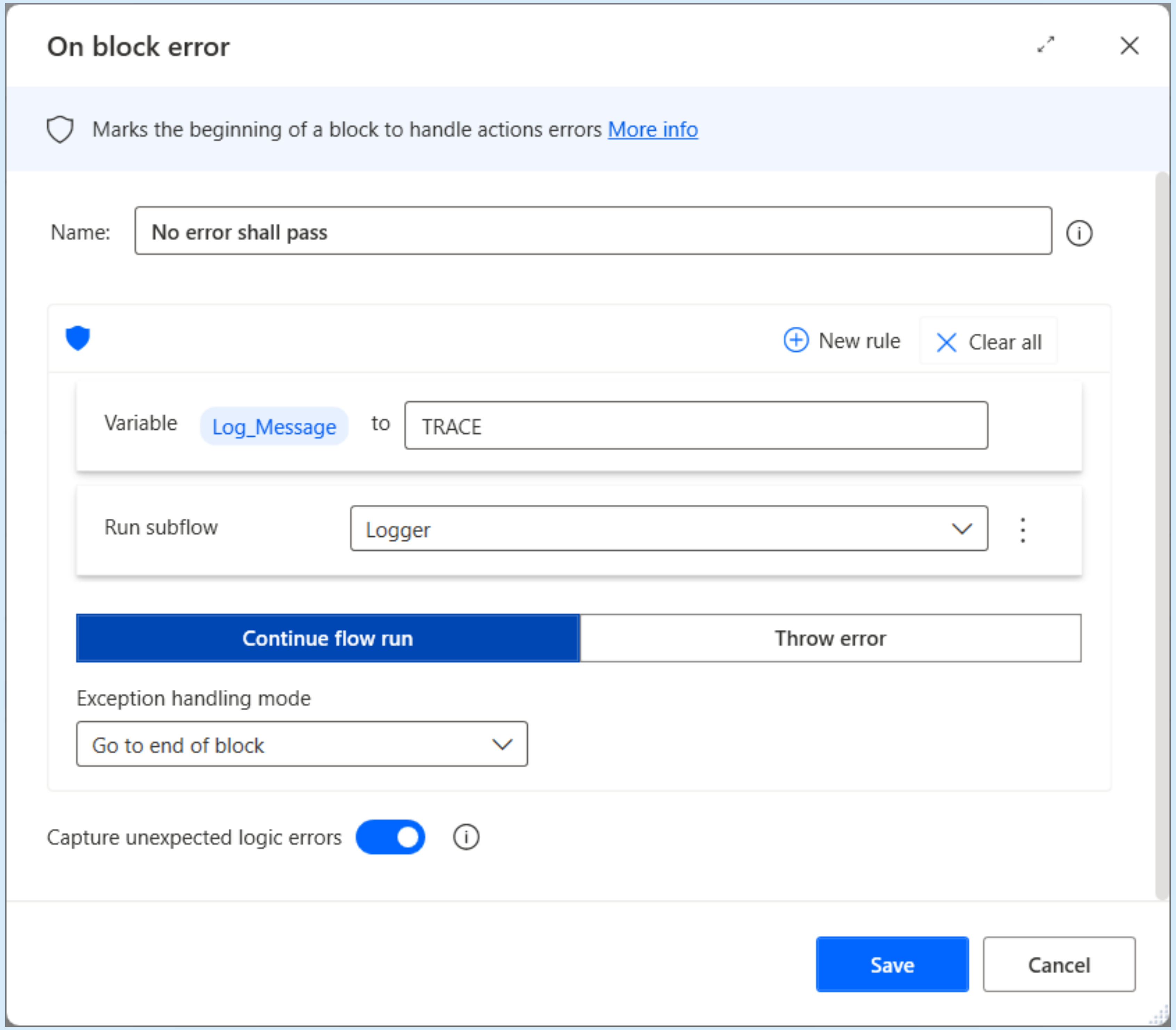
The reason I like plain text files:

- They’re very easy to set up in PAD
- **Write text to file** is much less likely to fail due to application errors.

The one limitation of logging to local files over SQL databases and SharePoint lists are the fact that they’re local to the machine (unless we push them to some network location).

But Enough Talking...

On Block Errors at Sub-flow Level + Get Last Error



On block error

Marks the beginning of a block to handle actions errors More info

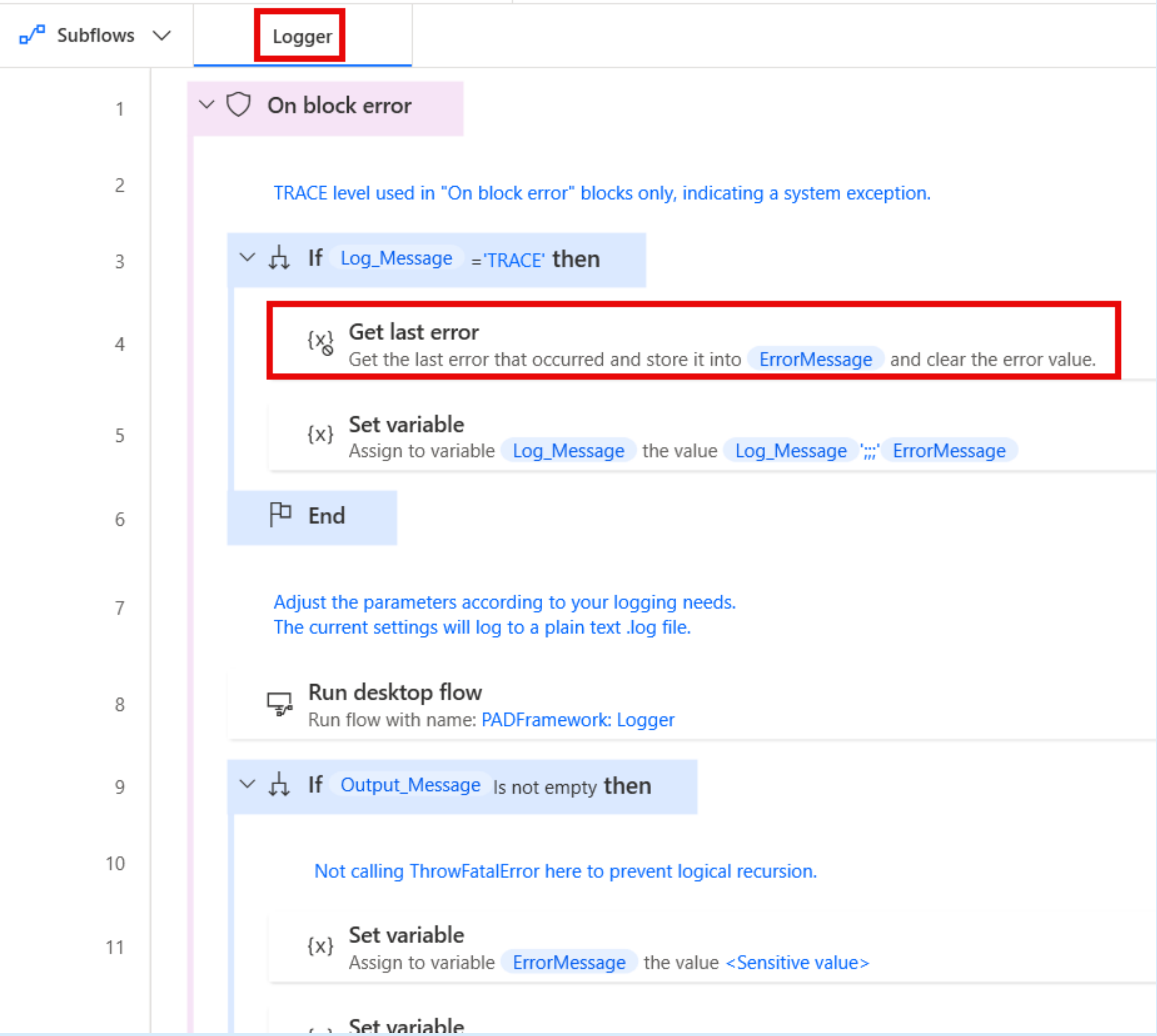
Name: No error shall pass

Variable Log_Message to TRACE

Run subflow: Logger

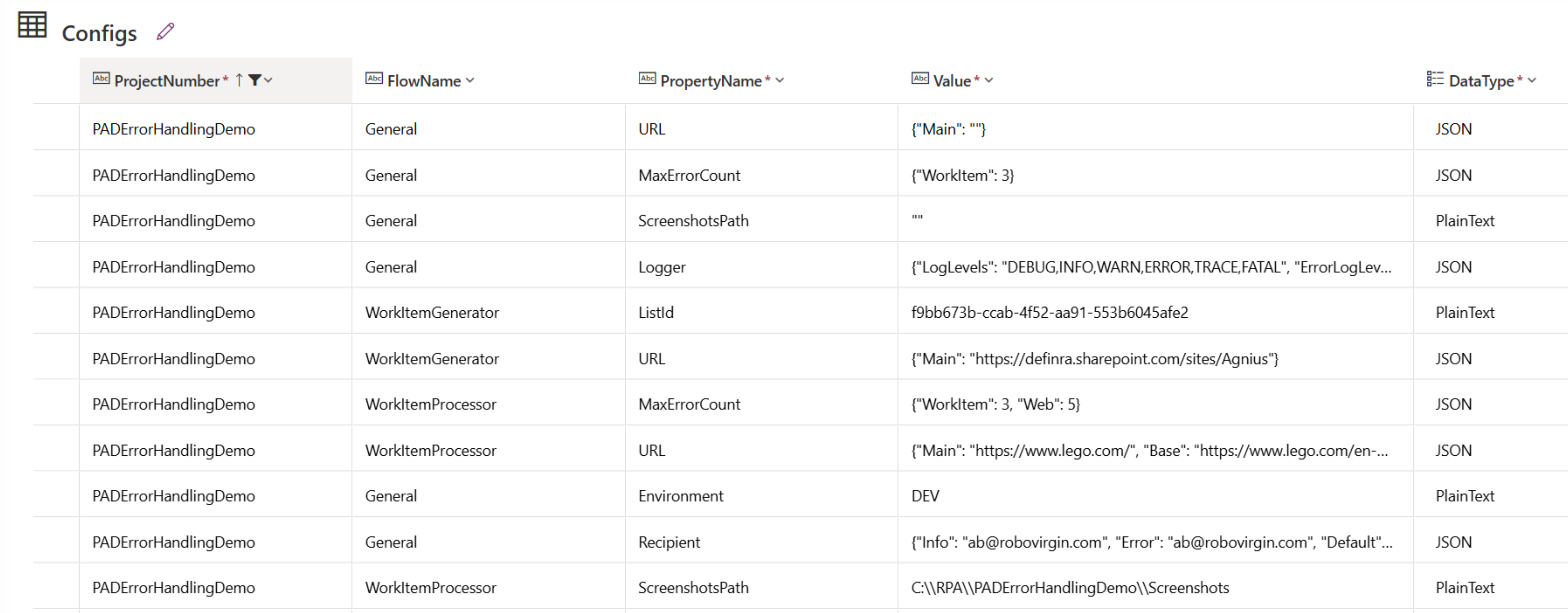
Exception handling mode: Go to end of block

Capture unexpected logic errors: on



```
1   On block error
2       TRACE level used in "On block error" blocks only, indicating a system exception.
3       If Log_Message ='TRACE' then
4           Get last error
            Get the last error that occurred and store it into ErrorMessage and clear the error value.
5           Set variable
            Assign to variable Log_Message the value Log_Message ';;;' ErrorMessage
6       End
7       Adjust the parameters according to your logging needs.
        The current settings will log to a plain text .log file.
8       Run desktop flow
        Run flow with name: PADFramework: Logger
9       If Output_Message Is not empty then
10          Not calling ThrowFatalError here to prevent logical recursion.
11          Set variable
            Assign to variable ErrorMessage the value <Sensitive value>
            Set variable
```

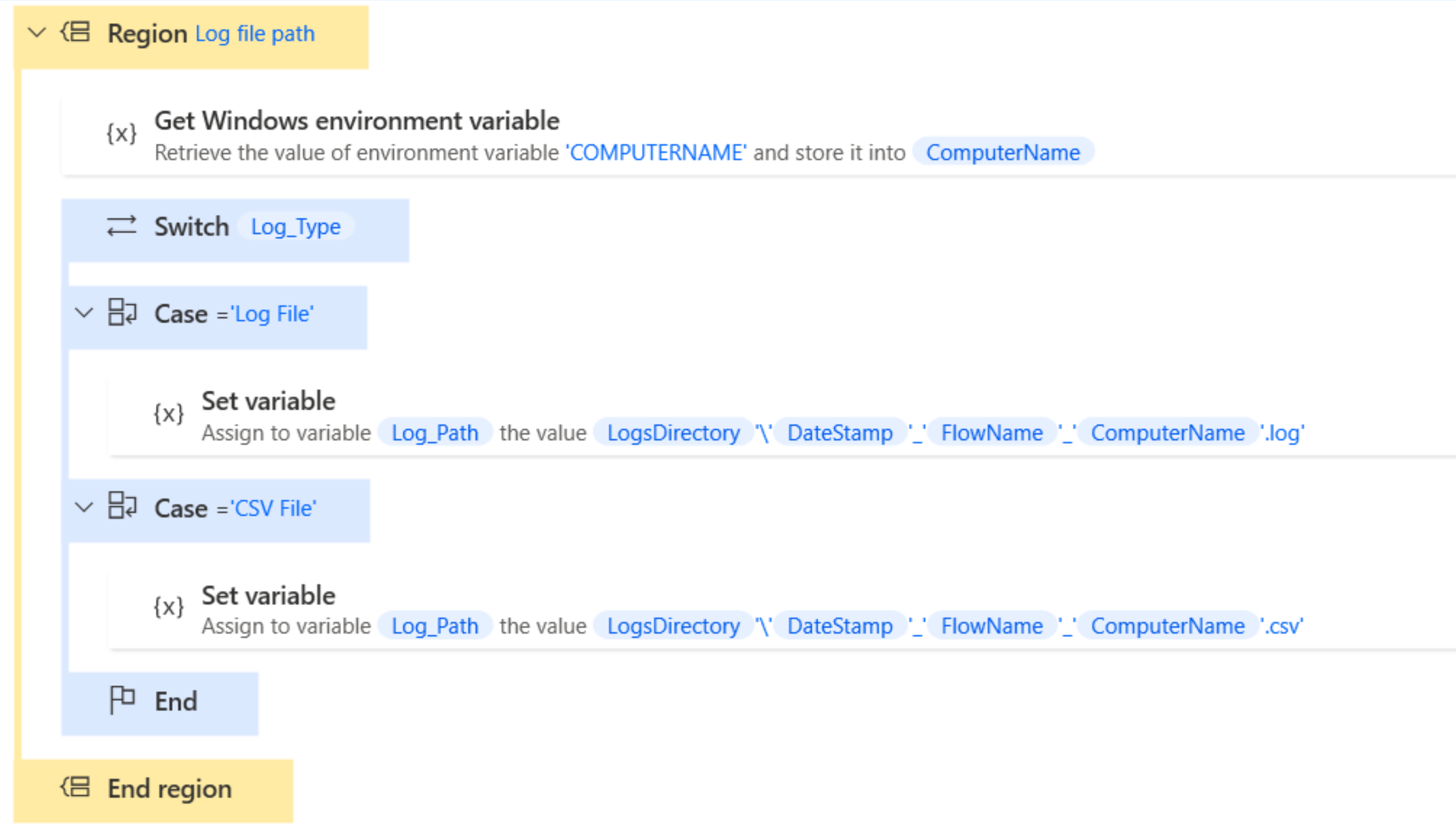
Logger Object + Screenshots Path in External Config



Configs

ProjectNumber * ↑	FlowName	PropertyName *	Value *	DataType *
PADErrorHandlingDemo	General	URL	{"Main": ""}	JSON
PADErrorHandlingDemo	General	MaxErrorCount	{"WorkItem": 3}	JSON
PADErrorHandlingDemo	General	ScreenshotsPath	""	PlainText
PADErrorHandlingDemo	General	Logger	{"LogLevels": "DEBUG,INFO,WARN,ERROR,TRACE,FATAL", "ErrorLogLev...	JSON
PADErrorHandlingDemo	WorkItemGenerator	ListId	f9bb673b-ccab-4f52-aa91-553b6045afe2	PlainText
PADErrorHandlingDemo	WorkItemGenerator	URL	{"Main": "https://definra.sharepoint.com/sites/Agnius"}	JSON
PADErrorHandlingDemo	WorkItemProcessor	MaxErrorCount	{"WorkItem": 3, "Web": 5}	JSON
PADErrorHandlingDemo	WorkItemProcessor	URL	{"Main": "https://www.lego.com/", "Base": "https://www.lego.com/en-...	JSON
PADErrorHandlingDemo	General	Environment	DEV	PlainText
PADErrorHandlingDemo	General	Recipient	{"Info": "ab@robovirgin.com", "Error": "ab@robovirgin.com", "Default"...	JSON
PADErrorHandlingDemo	WorkItemProcessor	ScreenshotsPath	C:\\RPA\\PADErrorHandlingDemo\\Screenshots	PlainText

Log File Per Day Per Flow
(+ Per Machine Optionally)



Region Log file path

{x} Get Windows environment variable
Retrieve the value of environment variable 'COMPUTERNAME' and store it into ComputerName

Switch Log_Type

Case ='Log File'

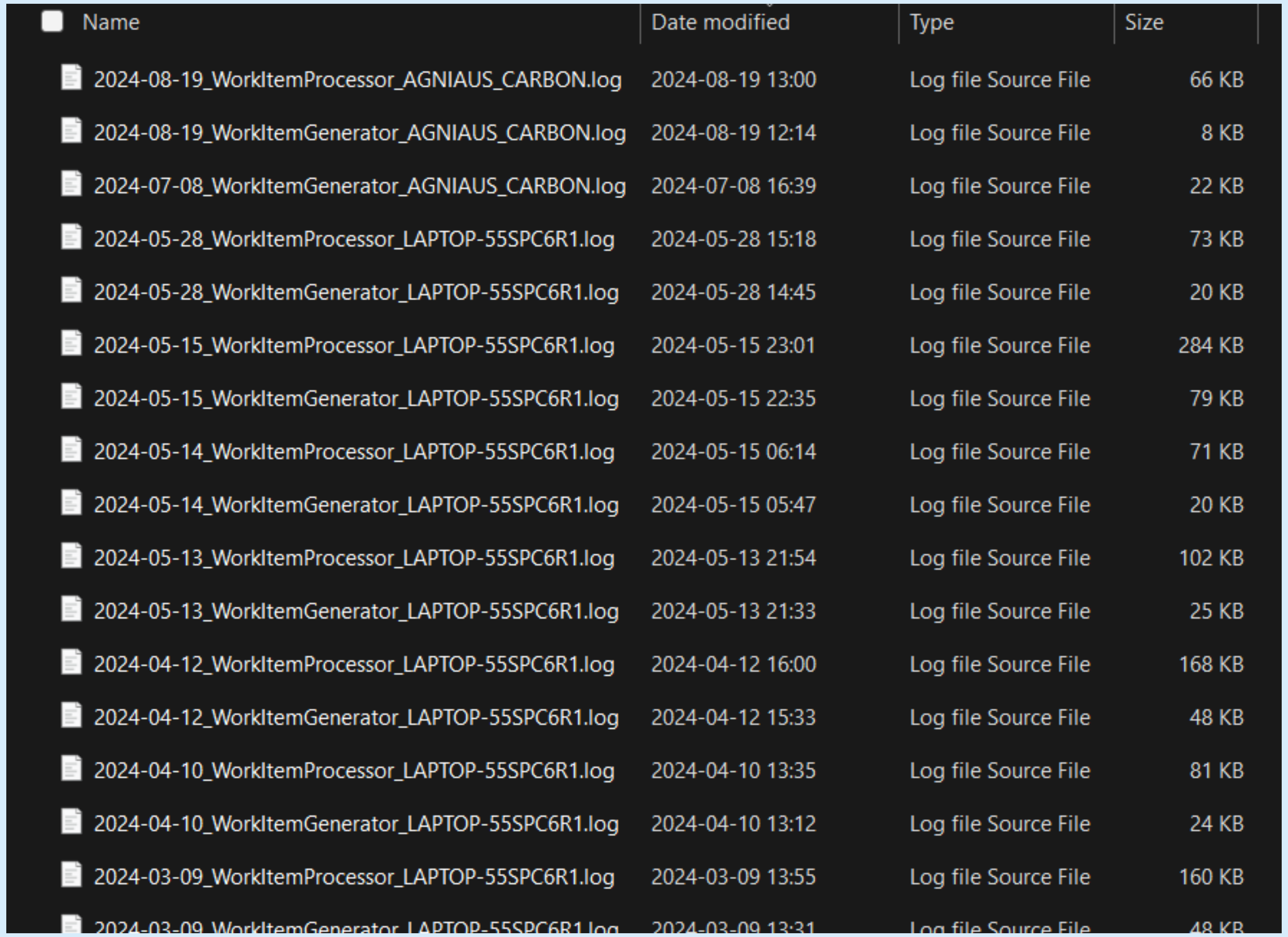
{x} Set variable
Assign to variable Log_Path the value LogsDirectory '\' DateStamp '_' FlowName '_' ComputerName '.log'

Case ='CSV File'

{x} Set variable
Assign to variable Log_Path the value LogsDirectory '\' DateStamp '_' FlowName '_' ComputerName '.csv'

End

End region



Name	Date modified	Type	Size
2024-08-19_WorkItemProcessor_AGNIAUS_CARBON.log	2024-08-19 13:00	Log file Source File	66 KB
2024-08-19_WorkItemGenerator_AGNIAUS_CARBON.log	2024-08-19 12:14	Log file Source File	8 KB
2024-07-08_WorkItemGenerator_AGNIAUS_CARBON.log	2024-07-08 16:39	Log file Source File	22 KB
2024-05-28_WorkItemProcessor_LAPTOP-55SPC6R1.log	2024-05-28 15:18	Log file Source File	73 KB
2024-05-28_WorkItemGenerator_LAPTOP-55SPC6R1.log	2024-05-28 14:45	Log file Source File	20 KB
2024-05-15_WorkItemProcessor_LAPTOP-55SPC6R1.log	2024-05-15 23:01	Log file Source File	284 KB
2024-05-15_WorkItemGenerator_LAPTOP-55SPC6R1.log	2024-05-15 22:35	Log file Source File	79 KB
2024-05-14_WorkItemProcessor_LAPTOP-55SPC6R1.log	2024-05-15 06:14	Log file Source File	71 KB
2024-05-14_WorkItemGenerator_LAPTOP-55SPC6R1.log	2024-05-15 05:47	Log file Source File	20 KB
2024-05-13_WorkItemProcessor_LAPTOP-55SPC6R1.log	2024-05-13 21:54	Log file Source File	102 KB
2024-05-13_WorkItemGenerator_LAPTOP-55SPC6R1.log	2024-05-13 21:33	Log file Source File	25 KB
2024-04-12_WorkItemProcessor_LAPTOP-55SPC6R1.log	2024-04-12 16:00	Log file Source File	168 KB
2024-04-12_WorkItemGenerator_LAPTOP-55SPC6R1.log	2024-04-12 15:33	Log file Source File	48 KB
2024-04-10_WorkItemProcessor_LAPTOP-55SPC6R1.log	2024-04-10 13:35	Log file Source File	81 KB
2024-04-10_WorkItemGenerator_LAPTOP-55SPC6R1.log	2024-04-10 13:12	Log file Source File	24 KB
2024-03-09_WorkItemProcessor_LAPTOP-55SPC6R1.log	2024-03-09 13:55	Log file Source File	160 KB
2024-03-09_WorkItemGenerator_LAPTOP-55SPC6R1.log	2024-03-09 13:31	Log file Source File	48 KB

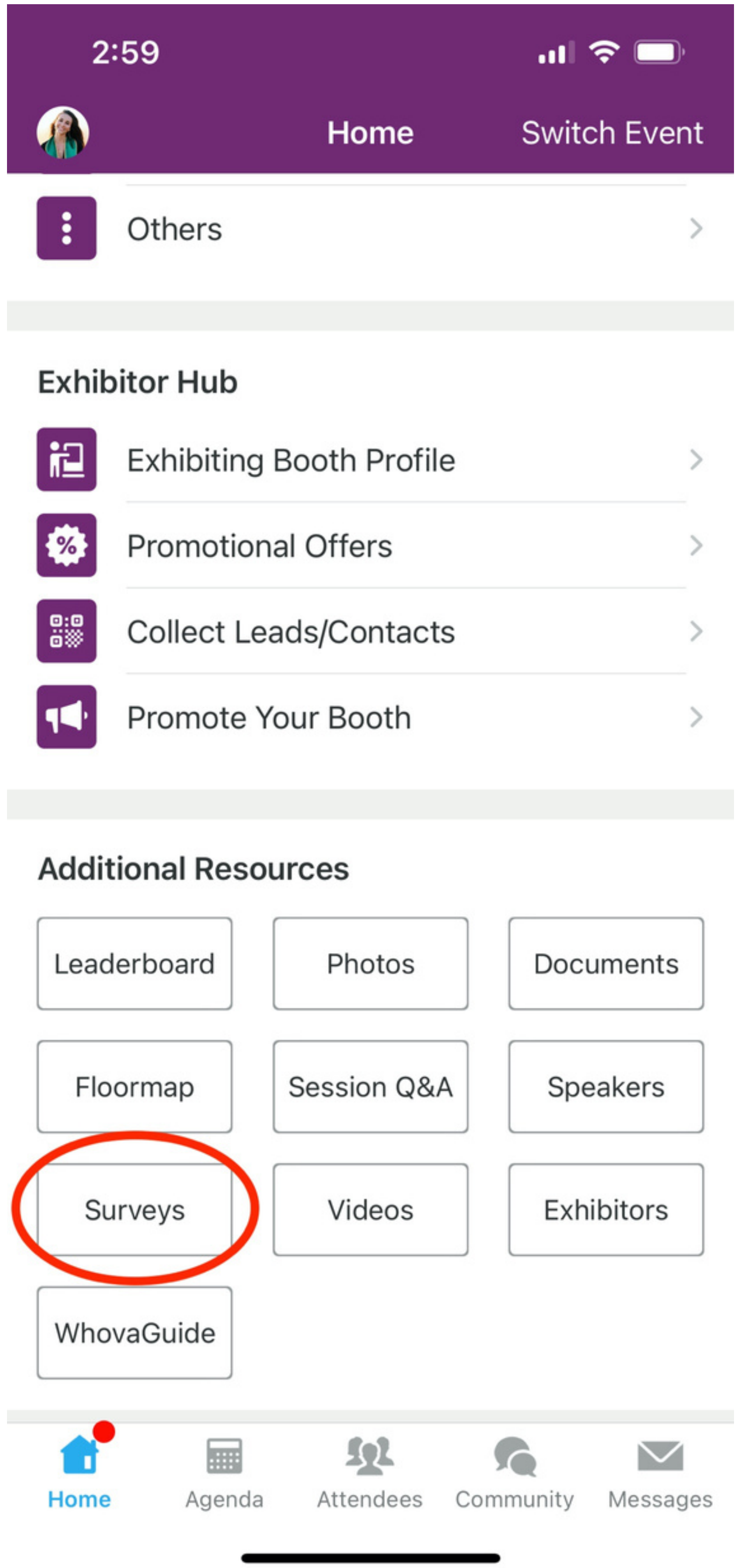
Thank you!

Connect with me:

Check out my
Youtube channel:

Agnius Bartninkas
ab@robovirgin.com
+370 654 81 936
www.robovirgin.com

Session Feedback Surveys



We really want to hear from YOU!
In the pursuit of making next year's Power Platform Community Conference even better, we want to hear your feedback about this session.

Here's How –

- *Simply go to the Whova App on your smartphone*
- *Scroll down on the Power Platform Community Conference Homepage to 'Additional Resources' to click "Surveys'.*
- *Click Session Feedback.*
- *Scroll down to find this session title.*
- *Complete the session feedback survey.*
- *Finally, click 'Submit'*

It's just that easy!