

# Using Microsoft Flow for IT Professionals



Jaap Brasser  
@jaap\_brasser



[jaapbrasser.com/about](http://jaapbrasser.com/about)

- Technical Marketing Engineer at Rubrik
- PowerShell Conference EU/Asia
- Dutch PowerShell User Group
- Speaker / Blogger / Tech Enthusiast
- PowerShell Gallery
- TechNet / MSTechCommunity



 @jaap\_brasser

# Agenda

---

## Why PowerShell?

---

## What is Microsoft Flow

---

## PowerShell & Flow

---

## Demos

---

## Questions



# Why PowerShell???

P O W E R S H E L L



What is  
Microsoft  
Flow



**Microsoft Flow**

# Connectors



Office 365 Outl...



OneDrive for B...



Office 365 Users



SharePoint



Twitter



Notifications



RSS



Outlook.com

## Recently added connectors



AtBot Admin



Marketing Cont...



Pitney Bowes T...



Excel Online (B...




Excel Online (O...




Azure SQL Data...

# Flow Templates



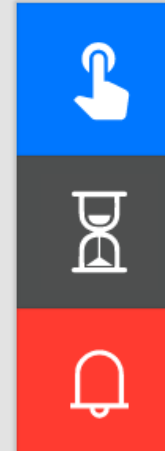
Save Office 365 email attachments to OneDrive for Business

By Microsoft  
Used 235650 times




Get a push notification when you receive an email from your...

By Microsoft  
Used 136809 times




Send myself a reminder in 10 minutes

By Microsoft  
Used 95634 times




Get today's weather forecast for my current location

By Microsoft  
Used 83902 times



Save Outlook.com email attachments to your OneDrive

By Microsoft Flow Community  
Used 64359 times



Get a push notification with updates from the Flow blog

By Microsoft  
Used 47714 times

# Why Microsoft Flow?

---

Use connectors to get data

---

Create triggers based on this data

---

Build workflows

---

Integrate and combine different services





Demo

Create your first Flow





# Demo Summary

- [Flow.Microsoft.com](https://flow.microsoft.com)
- Create and modify from template
- Create a custom flow





Demo

Import and Export Flows





# Demo Summary

- Export flow
- Zipped json files in package
- Import flow
- Select connectors upon Import



# Flow packages

---

Packaged  
as zip

Manifest files

---

APIs

---

Connections

---

Definition

---



# What are webhooks

---

A webhook transfers data over HTTP

---

Create triggers based on this data

---

Build workflows

---

Integrate and combine different services

---

Methods: **Get & Post**

The background of the slide is a dark blue/black field filled with a dense, vertical stream of white and light blue binary digits (0s and 1s), reminiscent of the 'Matrix' digital rain effect. The digits appear to be falling or scrolling from top to bottom, with some digits being sharper and more prominent than others, creating a sense of depth and motion.

# Demo

## Webhooks and Flow

Encryption!



**Bill Sempf**

@sempf

Follow



Base64 is not encryption. Base64 is not encryption. Base64 is not encryption. Base64 is not encryption. Base64 is not encryption. Base64 is not encryption. Base64 is not encryption. Base64 is not encryption. Base64 is not encryption. Base64 is not encryption.

11:10 PM - 23 Apr 2018

363 Retweets 1,173 Likes







# Demo Summary

- Created a Flow with webhook
- Write payload to disk
- Viewed what was delivered
- Transferred binary content
- Created binary file from content

# Request Flow

The screenshot displays an Azure Logic App workflow. The first step is a trigger named "When a HTTP request is received". Below the trigger, the "HTTP POST URL" is shown as `https://prod-37.westeurope.logic.azure.com:443/workflows/f59a4e50...`. The "Request Body JSON Schema" is defined as:

```
{
  "type": "object",
  "properties": {
    "Trigger": {
      "type": "string"
    }
  }
}
```

Below the schema, there are two links: "Use sample payload to generate schema" and "Show advanced options". A connector arrow points down to the second step, "Create file". This action is configured with the following settings:

- \* Folder Path: `/1FlowDemo`
- \* File Name: `ReceivedRequest` *fx* `utcNow()` *x* `.json`
- \* File Content: *fx* `Body` *x*



## Request Limits

# Request limits

These are limits for a single outgoing request.

## Timeout

Name	Limit
Request Timeout	120 Seconds

## Message size

Name	Limit	Notes
Message size	100 MB	Not all APIs support the full 100MB.
Expression evaluation limit	131,072 characters	<code>@concat()</code> , <code>@base64()</code> , <code>string</code> can't exceed this limit.



## Runtime Limits

# Run duration and retention

These are the limits for a single flow run.

Name	Limit	Notes
Run duration	30 days	Includes workflows with pending steps like approvals. After 30 days, any pending steps time-out. Timed-out approvals are removed from the approvals center. If someone attempts to approve a timed-out request, they'll receive an error message.
Storage retention	30 days	This is from the run start time.
Min recurrence interval	1 minute	
Max recurrence interval	500 days	



Demo

Microsoft Flow & Forms





# Demo Summary

- Created a form in Forms
- Created a flow with a trigger
- Added advanced condition
- Take action based on condition

## Advanced conditions

The screenshot shows a 'Condition' editor interface. At the top, there is a header bar with a tree icon, the word 'Condition', and a three-dot menu. Below this is a text input field containing the condition: `@or(equals(item()?['status'], 'unnecessary'), equals(item()?['status'], 'completed'))`. Below the input field are two buttons: 'Edit in basic mode' and 'Collapse condition'. Four callout boxes provide explanations: 1. 'Use the equals function to compare two values.' points to the first `equals` function. 2. '"status" is the name of the column in the spreadsheet. This is the 1st value used by the equals function. The equals function determines if the value of the status column is set to "unnecessary".' points to the `['status']` property access. 3. 'The "or" function name.' points to the `@or` function name. 4. '"unnecessary" is the 2nd value used by the equals function.' points to the string `'unnecessary'`.

Condition

`@or(equals(item()?['status'], 'unnecessary'), equals(item()?['status'], 'completed'))`

Edit in basic mode Collapse condition

Use the equals function to compare two values.

"status" is the name of the column in the spreadsheet. This is the 1st value used by the equals function. The equals function determines if the value of the status column is set to "unnecessary".

The "or" function name.

"unnecessary" is the 2nd value used by the equals function.



The background of the image is a dark blue gradient with a pattern of falling binary digits (0s and 1s) in a lighter blue color, reminiscent of the 'Matrix' digital rain effect. The digits are arranged in vertical columns that appear to be moving downwards.

# Demo

## Run PowerShell with Flow





# Demo Summary

- Install Data gateway
- Create connection online
- Run a watcher locally
- Create button enabled flow
- Monitor and restart flow agent



# Data Gateway sources

---

SQL Server

---

SharePoint

---

Oracle

---

Informix

---

Filesystem

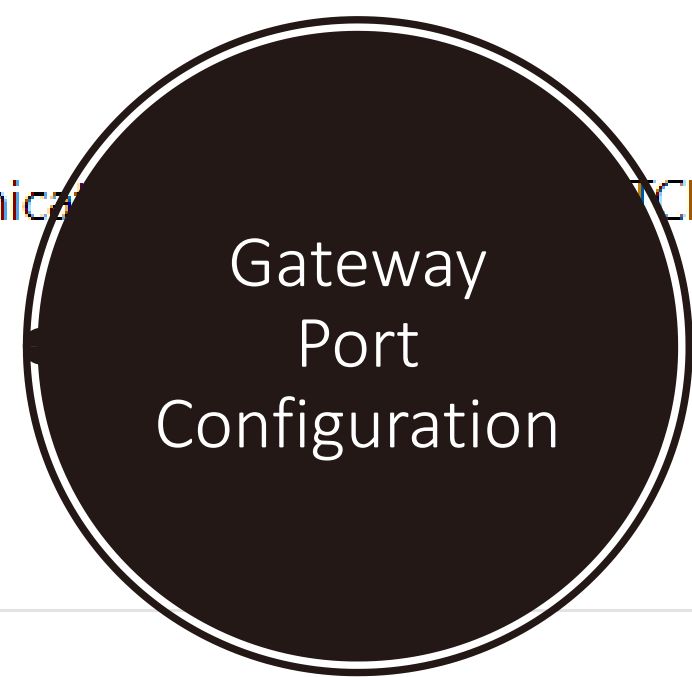
---

DB2

# Configure ports

The gateway creates an outbound connection to Azure Service Bus. It communicates on ports 443, 5671-5672, 9350 thru 9354. The gateway doesn't require inbound ports.

Learn more about [hybrid solutions](#).



Domain names	Outbound ports	Description
*.analysis.windows.net	443	HTTPS
*.login.windows.net	443	HTTPS
*.servicebus.windows.net	5671-5672	Advanced Message Queuing Protocol (AMQP)
*.servicebus.windows.net	443, 9350-9354	Listeners on Service Bus Relay over TCP (requires 443 for Access Co
*.frontend.clouddatahub.net	443	HTTPS

# Summary

---

Use cases of Microsoft Flow

---

Microsoft Flow templates

---

Took a look at connectors

---

Created workflows

---

Ran PowerShell triggered by Flow





Questions?

# References

- <https://docs.microsoft.com/en-us/flow/gateway-reference>
- <https://docs.microsoft.com/en-us/flow/guided-learning/>
- <https://docs.microsoft.com/en-us/flow/use-expressions-in-conditions>
- <https://github.com/jaapbrasser/events>
- <https://www.jaapbrasser.com>