Sync Work Items between multiple Team Projects:
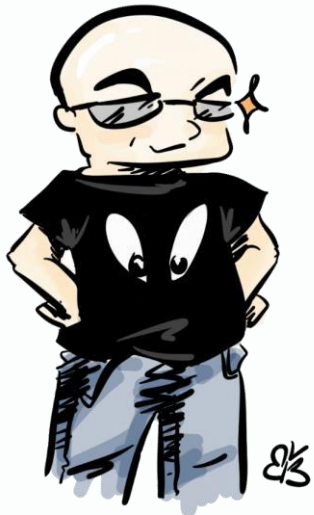## the good, the bad, the ugly…

lorenzo.barbieri@microsoft.com
linkedin.com/in/geniodelmale
@_geniodelmale

What is Azure DevOps

Azure Boards

Deliver value to your users faster using proven agile tools to plan, track, and discuss work across your teams.

Azure Pipelines

Build, test, and deploy with CI/CD that works with any language, platform, and cloud. Connect to GitHub or any other Git provider and deploy continuously.
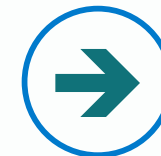
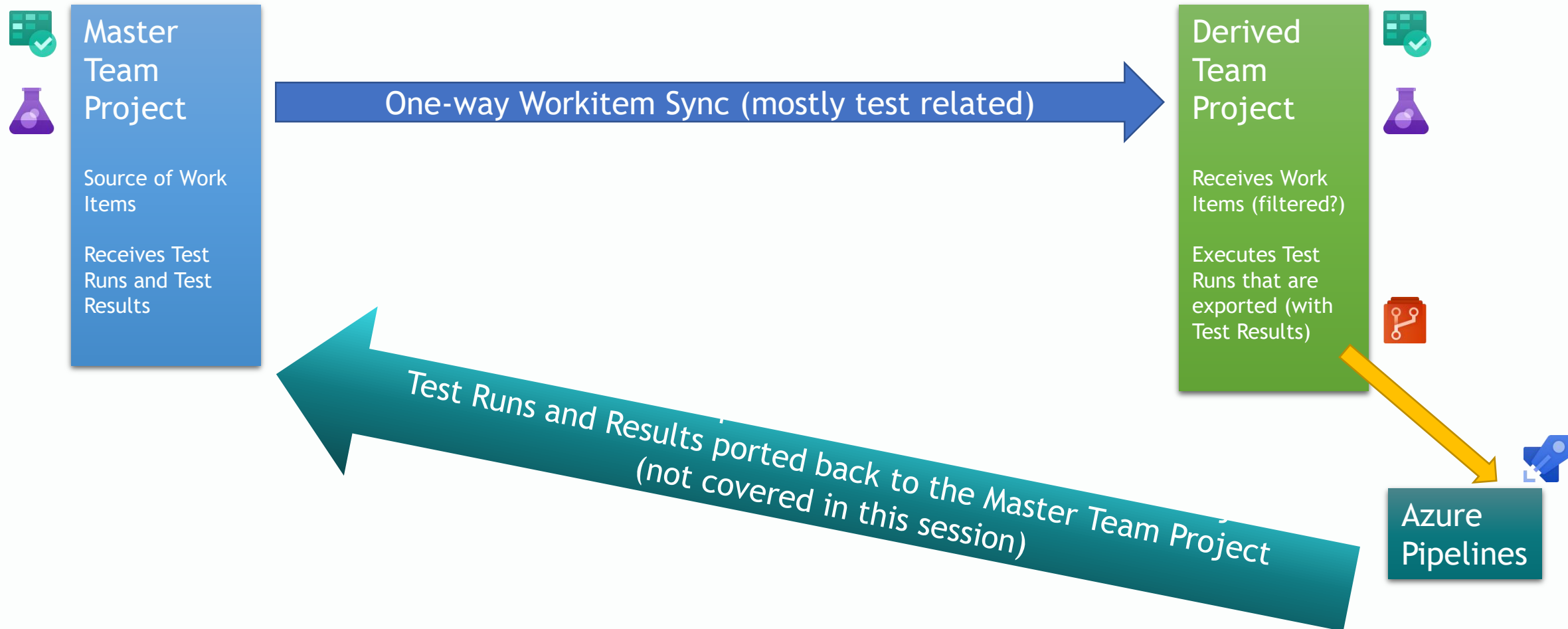## Azure Repos

Get unlimited, cloud-hosted private Git repos and collaborate to build better code with pull requests and advanced file management.

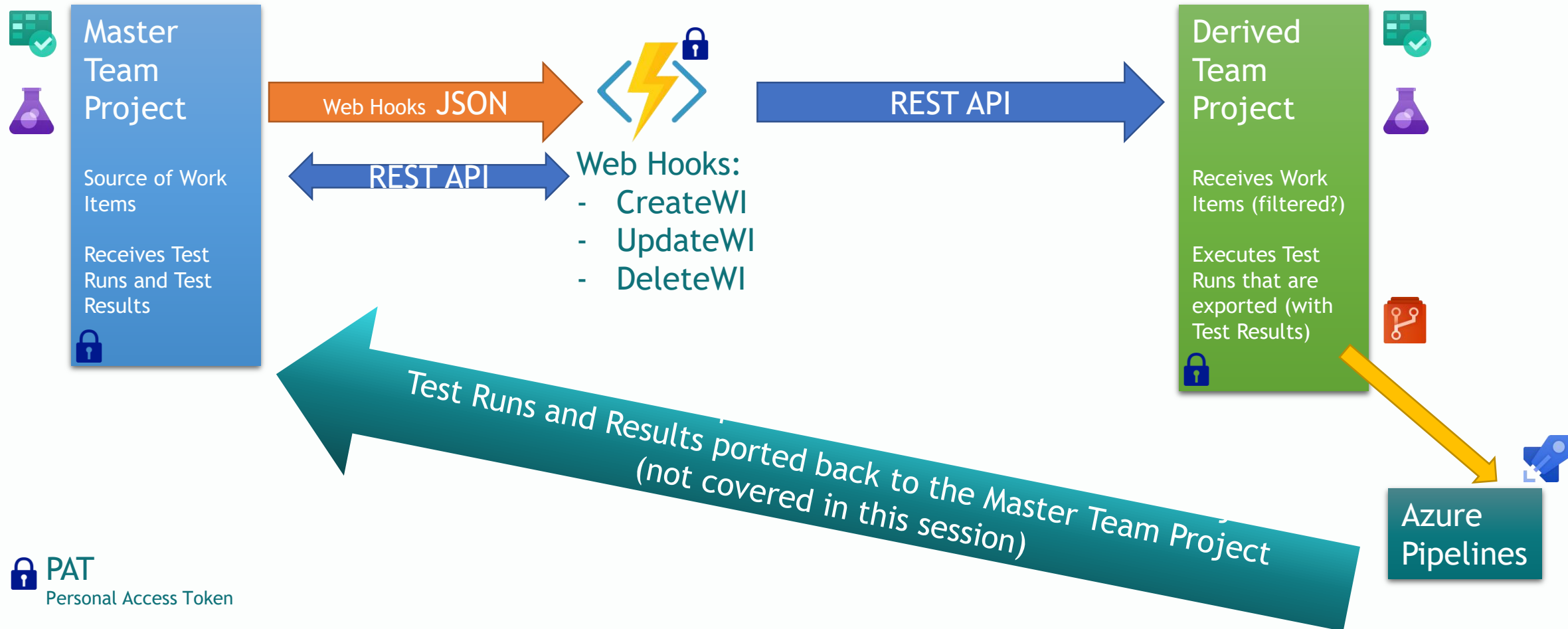## Azure Test Plans

Test and ship with confidence using manual and exploratory testing tools.

Azure Artifacts

Create, host, and share packages with your team, and add artifacts to your CI/CD pipelines with a single click.

https://azure.com/devops

Our scenario

Master Team Project

Source of Work Items

Receives Test Runs and Test Results

One-way Workitem Sync (mostly test related) →

Derived Team Project

Receives Work Items (filtered?)

Executes Test Runs that are exported (with Test Results)

Azure Pipelines

Test Runs and Results ported back to the Master Team Project (not covered in this session)

Our scenario

Master Team Project

Source of Work Items

Receives Test Runs and Test Results

Web Hooks **JSON** →

← REST API

Web Hooks:
- CreateWI
- UpdateWI
- DeleteWI

REST API →

Derived Team Project

Receives Work Items (filtered?)

Executes Test Runs that are exported (with Test Results)

Test Runs and Results ported back to the Master Team Project (not covered in this session)

Azure Pipelines

🔒 **PAT**
Personal Access Token

What is NOT part of THIS solution

Code sync

- Could be «easily» done with GIT, could be harder with TFVC

Users, groups, iterations, areas, artifacts, etc…

- This solution «works» but everything else should be set-up manually
- Links to code, wiki, etc… are not converted, not required in this scenario

Process templates MUST be equal.

- No on-the-fly transformation of different templates, not required in this scenario.

How do we «track» the Original WI?

- We've added a custom Tag to the Work Items

- PRO:
    - No need to create a custom process

- CON:
    - Could be «modified» by the users if not blocked in some ways
    - Difficult to read in the Work Items list, easier in the Kanban Board

DEMO

And beware...
This code is a Proof of Concept...
It's very far from being perfect!

Service Hooks Subscription – Web Hooks

EDIT SERVICE HOOKS SUBSCRIPTION

Trigger

Select an event to trigger on and configure any filters.

Trigger on this type of event

Work item created

ⓘ Remember that selected events are visible to users of the target service, even if they don't have permission to view the related artifact.

FILTERS

Area path ⓘ   optional

[Any]

Work item type ⓘ   optional

[Any]

☐ Links are added or removed ⓘ

Tag ⓘ   optional

EDIT SERVICE HOOKS SUBSCRIPTION

Perform this action

Post via HTTP

This action posts a JSON object representation of the event to the specified URL. Secure, HTTPS endpoints are recommended due to the potential for private data in the event payload. Learn More

SETTINGS

URL ⓘ   required

https://⬛⬛⬛⬛/api/WorkItemCreated?code=⬛⬛ ✓

☐ Accept untrusted SSL certificates ⓘ

Basic authentication username ⓘ   optional

Basic authentication password ⓘ   optional

HTTP headers ⓘ   optional

Resource details to send ⓘ   optional

All

Messages to send ⓘ   optional

All

Detailed messages to send ⓘ   optional

All

RESOURCE VERSION

1.0

Previous   Next   Test   Finish

TEST NOTIFICATION

Web Hooks (Post via HTTP)

Summary   Request   Response   Event

❌ **Failed**

Sent at: Tuesday, October 16, 2018 4:46:06 PM

Message
Bug #5 (Some great new idea!) created by Jamal Hartnett.

Error Message
Internal Server Error (500)

It uses a «standard» message. My code doesn't handle it and returns a HTTP 500

8

Service Hooks

Service Hooks

Integrate with your favorite services by notifying them when events happen in your project.

Consumer ↑	Event ↑	Action		Owner	7 Day Status	State
Web Hooks	··· Work item created	Any work item	Post via HTTP	To host ███████azurewebsi... ██████████	**1** succeeded	Enabled
Web Hooks	Work item deleted	Any work item	Post via HTTP	To host ███████azurewebsi... ██████████	**1** succeeded	Enabled
Web Hooks	Work item updated	Any work item	Post via HTTP	To host ██████.azurewebsi... ████████	**1** succeeded	Enabled

- You've to «monitor» this tab. If the functions fail for too many calls the corresponding Service Hook will be suspended, and should be enabled again manually.

- You can automate this: https://docs.microsoft.com/en-us/rest/api/azure/devops/hooks/?view=azure-devops-rest-5.0

9

Web Hooks – History and «debugging»

Realtime Workitem Sync...

The Good

It works

It solved partner and customer problem

We discovered a lot of things, and learned a lot of the «internals» of Work Items and especially Test items and APIs

Azure Functions offer 1M free executions per month, with 400.000Gb*s total memory usage... well enough also for «crowded» Team Projects...

Did I already say that «It works!»? ☺

The Bad

APIs are not «wery vell» documented. Perhaps we should step in and propose PRs for docs.microsoft.com... (if we could find 48hrs days ☺).

Most of the time we made some trade-offs to simplify the code, especially in the Test sync code.

- It worked on our scenarios,a full-fledged solution could be much more complex.
- Some cases (i.e., Work Item Restore, Work Item Delete from Recycle Bin) were ignored.
- Certain things are not, yet, aligned: i.e., Test Plans are visible as Work Items in the Derived TP

We're still on Azure Functions v1, with manual HTTP calls because of issues with Azure DevOps SDK on Functions (at least when we started the project).

Areas and Iterations should be updated manually, there aren't Service Hooks on those ones (we have thought about some workarounds but didn't implement any).

The Ugly

My code! ☺
Error handling, monitoring, etc… is very basic and very limited.

We need to figure out how to «restart» a working environment after a failure. Now it's a manual process.

- It's not a big problem for this project, considering the low volume of new/updated Work Items.

PAT should be updated manually… every year or so.

- A «service account» user should be used, not an actual user like in this demo.
- We're using a single PAT for both Team Projects, also if they're on different hosts.

What about a «full» two way sync

Realtime Sync

One could extend this approach for a «full» two way sync

- You need to remove «fixed» paths inside the code

- Workitem IDs should be tracked on both sides

•- Beware of «circular loops», check WI revisions

Scheduled Sync

An easier option could be using a «traditional sync» and scheduling it

- You should handle the scheduler, on-premise or in the cloud (timed functions?)

- External tool could be difficult to set-up

Where's the code?

https://github.com/geniodelmale/azdevops-sync

Thank you!!!

Join at Slido.com with #devops2019

lorenzo.barbieri@microsoft.com
linkedin.com/in/geniodelmale
@_geniodelmale