



by entwickler.de

Advanced Pull Requests Checks and Policies

Marc Müller
Neno Loje



www.4tecture.ch

4tecture[®]
empower your software solutions

A black and white portrait of Marc Müller, a man with glasses and a mustache, smiling. He is wearing a dark polo shirt under a light-colored jacket.

About me:

Marc Müller
Principal Consultant
@muellermarc



4tecture[©]
empower your software solutions

Our Products:

Multi-Tenant OpenID
Connect Identity Provider



ProAuth

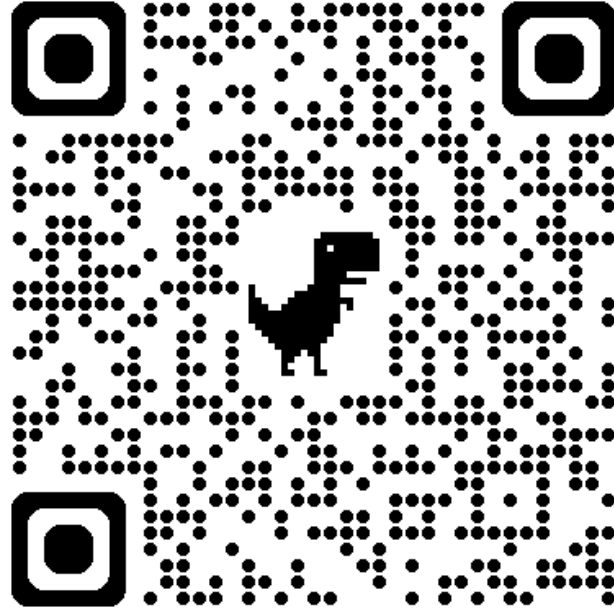
www.proauth.net

Enterprise Application
Framework for .NET



www.reafx.net

Slide Download



<https://www.4tecture.ch/events/basta25-advancedpullrequest>



DEMO

Pull Request

A close-up, low-angle shot of several rowers in a boat, wearing blue and red uniforms, pulling their oars. The oars are black with yellow handles and are connected to a complex mechanical system. The background is a bright, overexposed sky.

Advanced Pull Requests

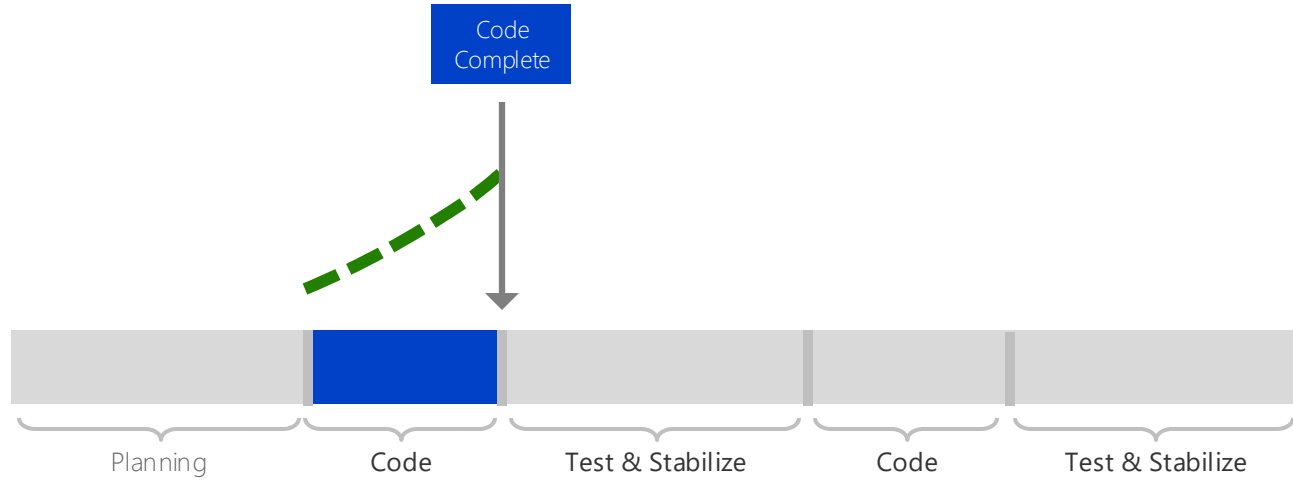
PR Deployments - Why?

4tecture
empower your software solutions

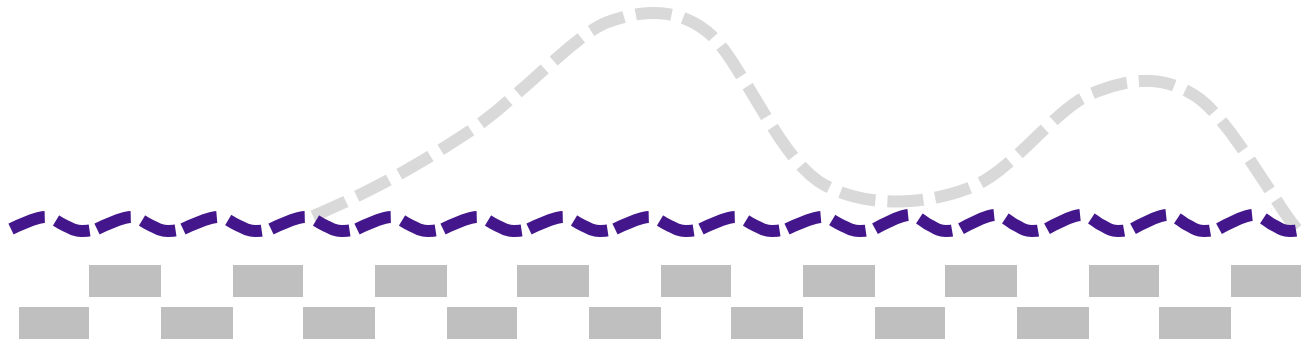


Fail fast!

Before



After



Shift left

Move the testing process to the left

- Integrate testing into the sprint / pull request
- Fast detection and fix
- Testers are part of the team

Continuous Testing

- Effective and continuous integration
- No Bulks of tests / bug-fixing

Without shift left context-switching is expensive / lowers throughput drastically

Shift Left Benefits

- Reduced costs involved in dev/test
- Early bug detection – better quality
- Effective resolution of bugs
- Massive time and effort saved





If it
hurts, do
it more
often!

Test Automation

- Reduce test time
- Have regression tests
- Focus on test design and management, rather than manual repetitive tasks

There is no better
place than
production!

Production-like environments

- Integration is important
- Real scenarios with real tests
- Test how it will be used in production
- Real flows, delays, latency, retries, etc.
- Reduction of no-repro bugs

Conclusion

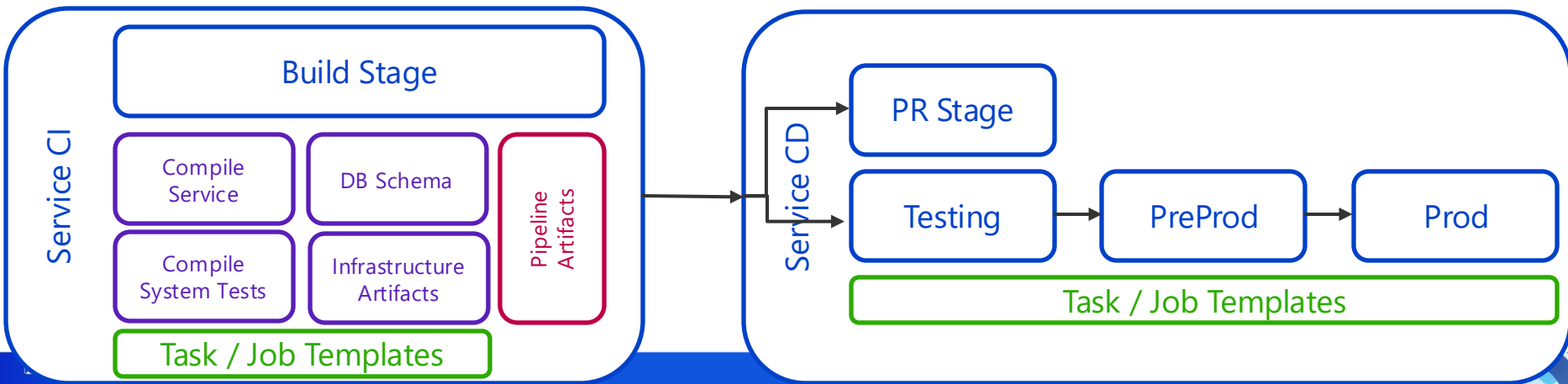
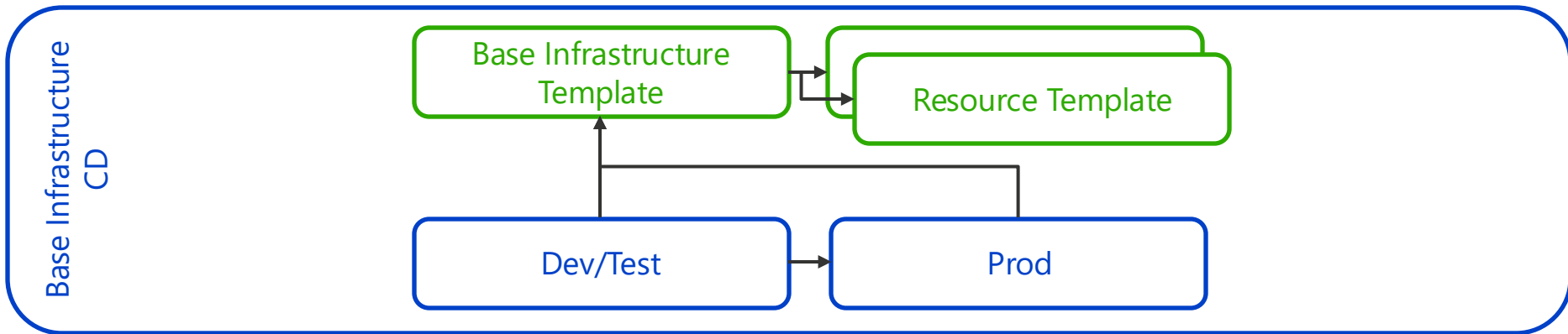
- No “shift left” only strategy
Combination of shift left and system testing on target system recommended
- Reducing or avoiding long circle times is crucial
- “There is no place like production”

A background image showing a rowing team in a boat. The rowers are wearing blue and red uniforms and are pulling oars with yellow handles. The boat is on water, and the background is slightly blurred.

Advanced Pull Requests

Azure Pipelines Structure and Setup

Best Practices



Is PR Validation 100% safe?

- It depends...
- Integrate multiple parallel PRs
 - Sequential
 - Parallel
- Depends on PR integration frequency / pipelines runtime

Edit build policy ×

Enabled
☒ On

Build pipeline *

Path filter (optional)

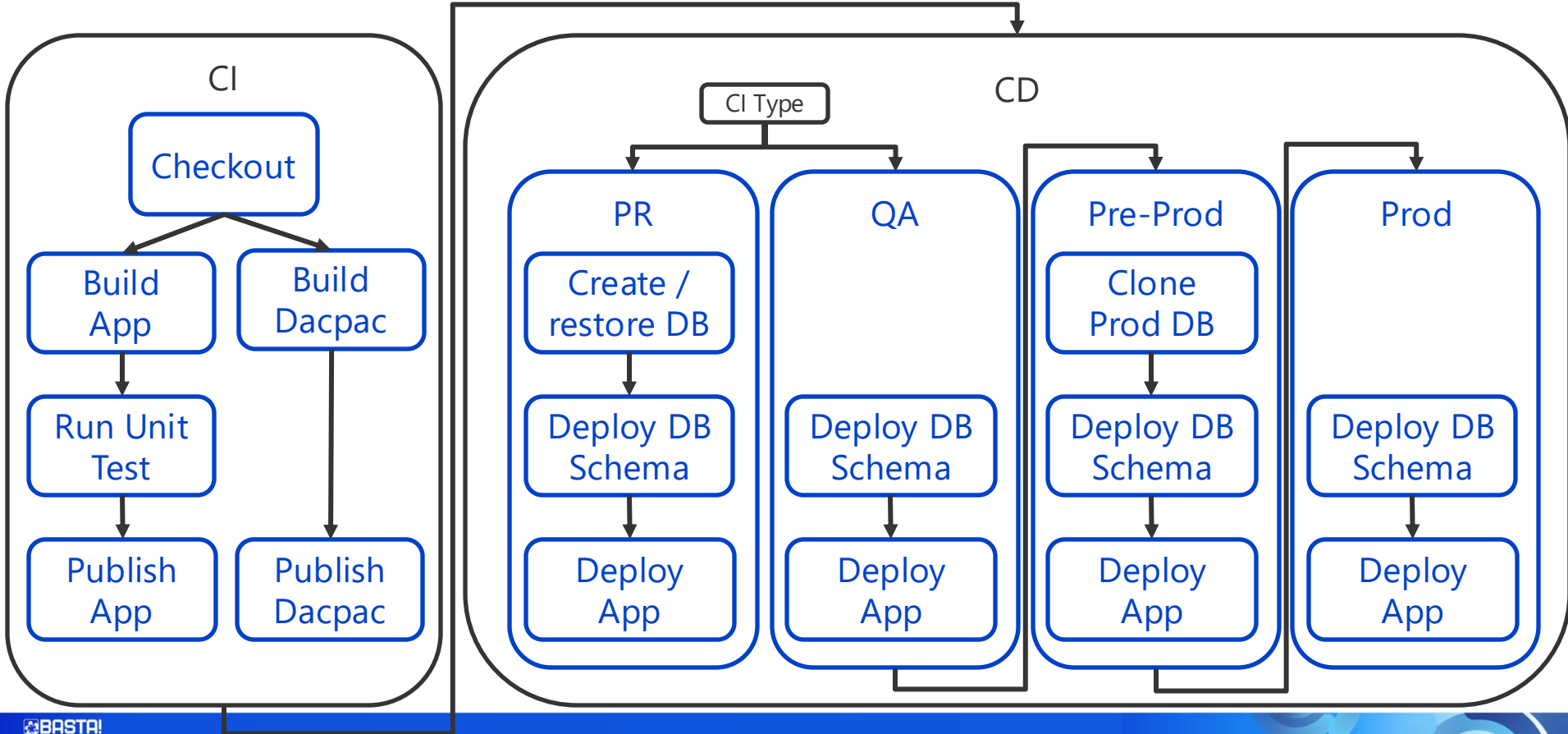
Trigger
☒ Automatic (whenever the source branch is updated)
☐ Manual

Policy requirement
☒ Required
Build must succeed in order to complete pull requests.
☐ Optional
Build failure will not block completion of pull requests.

Build expiration
☐ Immediately when \S^9 main is updated
☒ After hours if \S^9 main has been updated
☐ Never

Display name

CI / CD Pipeline



A background image showing a rowing team in a boat. The rowers are wearing blue and red uniforms. The focus is on the oars and the rowing mechanism, which are yellow and blue. The water is visible in the background.

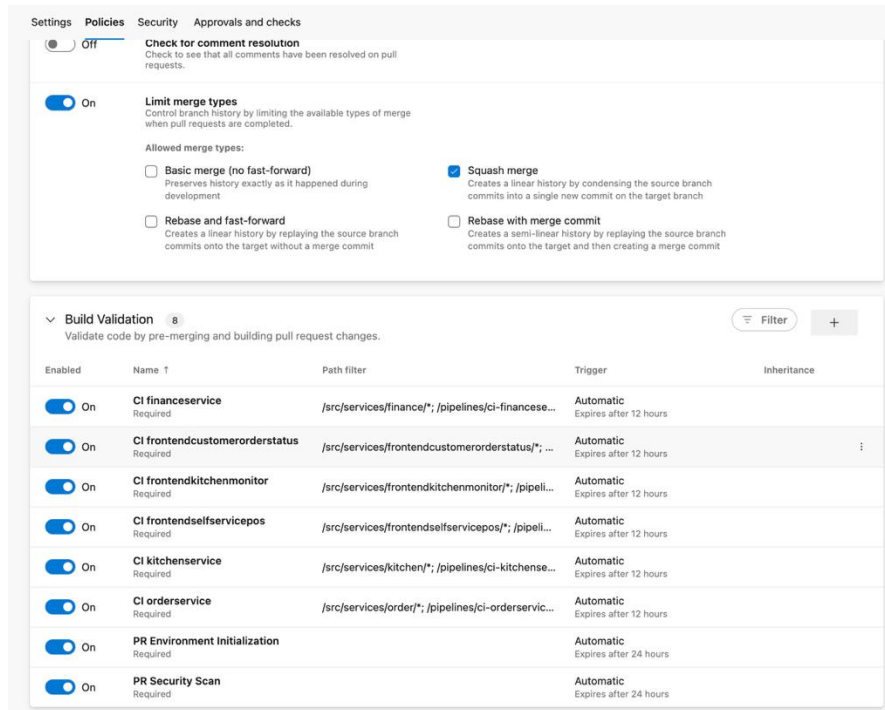
Advanced Pull Requests

Branch Policies

4tecture
empower your software solutions

Branch Policies

- Forced policies to be able to integrate into target branch
- General Policies
- Build Validation Policies
- Status Checks
- Automatic Reviewers



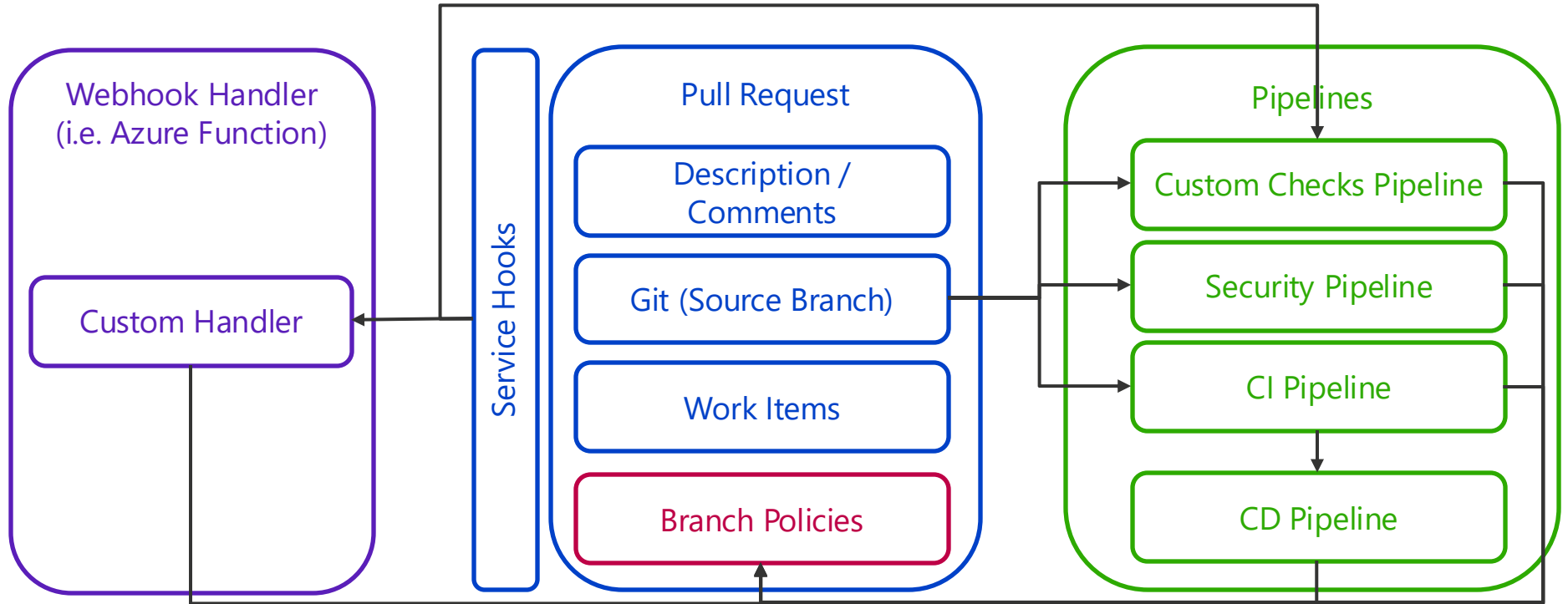
The screenshot displays the GitHub Branch Policies configuration interface. It is divided into several tabs: Settings, Policies, Security, and Approvals and checks. The 'Policies' tab is active, showing a toggle for 'Check for comment resolution' set to 'Off'. Below this, the 'Limit merge types' section is turned 'On', allowing users to control branch history by limiting merge types. It lists four options: 'Basic merge (no fast-forward)' (unchecked), 'Squash merge' (checked), 'Rebase and fast-forward' (unchecked), and 'Rebase with merge commit' (unchecked). The 'Build Validation' section is expanded, showing a table of policies that validate code by pre-merging and building pull request changes. The table includes columns for 'Enabled', 'Name', 'Path filter', 'Trigger', and 'Inheritance'. Eight policies are listed, all with 'Automatic' triggers and 'Expires after 12 hours' (except for 'PR Environment Initialization' which expires after 24 hours and 'PR Security Scan' which expires after 24 hours).

Enabled	Name	Path filter	Trigger	Inheritance
On	CI financeservice Required	/src/services/finance*/; /pipelines/ci-financeservice...	Automatic Expires after 12 hours	
On	CI frontendcustomerorderstatus Required	/src/services/frontendcustomerorderstatus*/; ...	Automatic Expires after 12 hours	
On	CI frontendkitchenmonitor Required	/src/services/frontendkitchenmonitor*/; /pipeli...	Automatic Expires after 12 hours	
On	CI frontendselfservicepos Required	/src/services/frontendselfservicepos*/; /pipeli...	Automatic Expires after 12 hours	
On	CI kitchenservice Required	/src/services/kitchen*/; /pipelines/ci-kitchense...	Automatic Expires after 12 hours	
On	CI orderservice Required	/src/services/order*/; /pipelines/ci-orderservic...	Automatic Expires after 12 hours	
On	PR Environment Initialization Required		Automatic Expires after 24 hours	
On	PR Security Scan Required		Automatic Expires after 24 hours	

Pull Request Status and Comments

- Status checks perform automated verification
- PR deployment should be added as PR status (as required policy)
 - Only deployable apps / services should be integrated
 - System integration tests should be part of the PR deployment
- Work Item Validation, PR Content Validation, External System Integration
- Automated comments guide the reviewers and product owners

Custom Checks and Policies



Custom Checks

- Security Checks
- Work Items → content, type, status, links
- PR Deployments → tests, monitoring, ...
- PR Description → Check boxes checked?

Additional Automation

- Pull Request Tags
- AI description and work item text verification
- ...

A person is seen from the side, sitting at a desk in a dark room. They are looking at two computer monitors. The left monitor displays a web application with a sidebar and a main content area. The right monitor displays a code editor with syntax-highlighted code. A red Coca-Cola can is on the desk between the monitors. The person's hands are on a keyboard. The overall lighting is blue and dim, with the primary light source being the monitors.

DEMO

Branch Policies

A background image showing a rowing team in a boat. The rowers are wearing blue and red uniforms, and their oars are visible, extending from the boat into the water. The image is slightly blurred, emphasizing the motion of the rowing.

Advanced Pull Requests

Automatic Code Reviewers

4tecture
empower your software solutions



DEMO

Automatic Code Reviewers

A background image showing a rowing team in a boat. The rowers are wearing blue and red uniforms and are pulling oars with yellow handles. The boat is on water, and the background is slightly blurred.

Advanced Pull Requests

PR Templates

4tecture
empower your software solutions

PR Templates - About

■ Definition

- A PR template is a .md or .txt file whose contents are injected into the pull-request description when the PR is first created

■ Built-in template types:

- **Default template**
Applied to *every* new PR unless a branch-specific file overrides it
- **Branch-specific template**
Automatically used when the PR's *target* branch matches the file name, e.g., dev.md for all dev/* PRs
- **Additional/optional templates**
Selectable from **Add a template** drop-down so authors can append extra guidance

PR Templates – Content, Benefits

■ Typical content

- check-lists (unit tests, docs updated, work-item links)
- acceptance criteria
- “Definition of Done” reminders
- security or performance gates, etc.

■ Benefits

- Standardizes hand-overs
- reduces missing info
- accelerates reviews
- enforces internal conventions & compliance and pairs well with branch policies for true “shift-left” quality



DEMO

PR Templates

A background image showing a rowing team in a boat. The rowers are wearing blue and red uniforms and are pulling oars with yellow handles. The boat is on water, and the background is slightly blurred.

Advanced Pull Requests

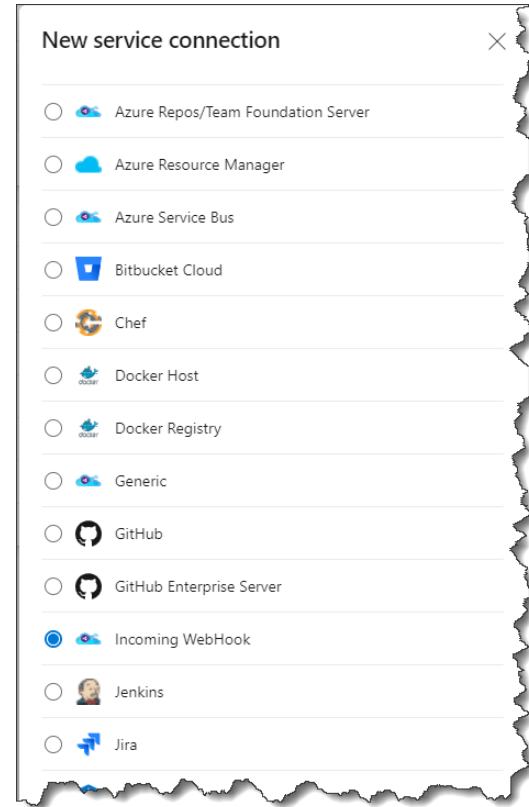
Azure Pipelines Incoming Webhook

Trigger

- Main triggers are repository / code related
- Many automation scenarios trigger from other events
 - Work Item Update
 - Pull Request Update

Incoming Webhook Service Connection

- Incoming webhook can be defined as service connection
- Generic webhook trigger in pipelines
- Payload can be filtered



Pipeline Resource

- Azure DevOps creates webhook endpoint based on alias
- Pipeline resource triggers pipeline
- Filters can be applied to trigger

```
1 trigger: none
2
3 pool:
4   vmImage: ubuntu-latest
5
6 resources:
7   # https://dev.azure.com/4tecture-demo/_apis/public/distributedtask/webhooks/prupdated?api-version=6.0-preview
8   webhooks:
9     - webhook: PRUpdated .....### Webhook alias
10      connection: PREventsConnection .....### Incoming webhook service connection
11      filters:
12        - path: eventType
13          value: git.pullrequest.updated
14        - path: publisherId
15          value: tfs
16        - path: resource.repository.name
17          value: DevFun
18
19 variables:
20   View template
21   - template: templates/common_variables.yml
22   View template
23   - template: templates/common_variables-od
```

Service Hook & Incoming Webhook

- Service hooks can trigger webhooks for many Azure DevOps Events

- Build completed
- Code pushed
- Elastic agent pool resized
- Pull request commented on
- Pull request created
- Pull request merge attempted
- Pull request updated
- Release abandoned
- Release created
- Release deployment approval completed
- Release deployment approval pending
- Release deployment completed
- Release deployment started
- Run stage approval completed
- Run stage state changed
- Run stage waiting for approval
- Run state changed
- Work item commented on
- Work item created
- Work item deleted
- Work item restored
- Work item updated

The screenshot displays the 'EDIT SERVICE HOOKS SUBSCRIPTION' dialog in Azure DevOps, which is used to configure webhooks triggered by service hooks. The dialog is split into two main sections: 'Filters' and 'Action/Settings'.

Filters Section:

- Trigger on this type of event:** A dropdown menu currently set to 'Pull request updated'.
- Remember that selected events are visible to users of the target service, even if they don't have permission to view the related artifact.** (Informational note)
- FILTERS:**
 - Repository:** DevFun
 - Target branch:** [Any]
 - Change:** Status changed
 - Requested by a member of group:** [Any]
 - Reviewer includes group:** [Any]
- Previous** button

Action Section:

- Action:** Select and configure the action to perform.
- Perform this action:** Post via HTTP
- This action posts a JSON object representation of the event to the specified URL.**
- It's recommended that you only use HTTPS endpoints due to the potential for private data including any authentication headers in the event payload. [Learn more about Webhooks](#)**
- SETTINGS:**
 - URL:** (required, with a green checkmark)
 - ☐ Accept untrusted SSL certificates (optional)
 - Basic authentication username:** (optional)
 - Basic authentication password:** (optional)
- Previous**, **Next**, **Test**, **Finish**, and **Cancel** buttons

A person is seen from the side, sitting at a desk in a dark room. They are looking at two computer monitors. The left monitor displays a web application with a sidebar and a main content area. The right monitor displays a code editor with syntax-highlighted code. A red Coca-Cola can is on the desk between the monitors. The person's hands are on a keyboard. The overall lighting is blue and dim, with the primary light source being the screens.

DEMO

Incoming Webhook

A background image showing a rowing team in blue uniforms and red accents, pulling oars with yellow handles. The oars are in a synchronized motion, creating a sense of teamwork and effort.

Advanced Pull Requests

Q & A

4tecture
empower your software solutions

Recap

- Pull Request is single point of interaction / status for developers, testers and product owners
- Fail fast – learn fast & fix fast
- Only an integrated change provides clarity if it runs successfully in production
- Use events and change triggers to verify the status of the PR and report it through PR Status / Checks

Thank you for your attention!

If you have any questions do not hesitate to contact us:

4tecture GmbH
Industriestrasse 25
CH-8604 Volketswil

+41 44 508 37 00
info@4tecture.ch
www.4tecture.ch

Marc Müller
Principal Consultant

www.powerofdevops.com



A background image showing several hands of different skin tones reaching towards the center, where they are assembling four interlocking wooden puzzle pieces. The pieces are colored light brown, white, red, and green. The overall scene is softly blurred, focusing attention on the hands and the puzzle.

4tecture[©]
empower your software solutions