# Esta-Tekton AzureAd-, APIM-, Kafka-Registration Prototyp - Technologieauswahl

[ESTA-5242](https://flow.sbb.ch/browse/ESTA-5242) - Esta-Tekton AzureAd-, APIM-, Kafka-Registration Prototyp Closed Nach dem Workshop mit [Spirig Lukas (IT-PTR-CEN2-SL3)](file:///C:\display\~u228304)  [Jeanneret Julien (IT-PTR-CEN1-BDE3)](file:///C:\display\~u231154)  [Brüderli Thomas (IT-PTR-CEN2-SL2)](file:///C:\display\~u233658) [Masen Igor (IT-PTR-CEN2-SL11)](file:///C:\display\~u210691) [Barahona José Luis (IT-PTR-CEN2-BDE16)](file:///C:\display\~u233586) [Wallrapp Manuel (IT-PTR-EXT-EXT2 - Extern)](file:///C:\display\~ue64007)

haben wir entscheiden auf Openshift Operators als Technologie für AzureAd, APIM und Kafka Registrations zu setzen. Wir werden eine Beispielimplementierung anhand AzureAd Registration für einen Workshop mit den jeweiligen Teams vorbereiten.

Uns stehen zur Zeit 3 Technologien zur Auswahl:

* Operator<https://flow.sbb.ch/browse/ESTA-5242>
* Tekton Taskrun
* Openshift Job.

Jede dieser Technologien bringt vor on und Nachteile. Dabei wollen wir evaluieren welchen technologischen Weg wir gehen wollen.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Operator** | **Tekton  Taskrun** | **Openshift Job** |
| **Entwicklungsstart ohne CON** | :cross: | :tick: | :tick: |
| **Rollout über alle Cluster ohne CON** | :tick: | :tick: | :tick: |
| **Namespace spezifisches Rollout** | :tick: | :tick: | :tick: |
| **Technologische Expertise ESTA** | (:cross:) | :tick: | :tick: |
| **Expertise SBB / Azure AD, Kafka, APIM Team** | :cross: | :cross: | :tick: |
| **Releases von Helm & Docker in Artifactory** | :question: | :tick: | :tick: |
| **Existierender Esta-Standard Build&Deployment Prozess** | :cross: | :tick: | :tick: |
| **Gitops über Argocd** | :tick: | :tick: | :tick: |
| **Gitops über Argocd & CRD** | :tick: | :cross: | :cross: |
| **Endprojekt kann/muss verwendete Version selber managen  (z.B. per Angabe der Helmchartversion 2.2.^)** | :cross: | :tick: | :tick: |
| **Können Breaking Changes bei Logik / Format der Konfiguration gemanaged werden?**  **Beim Operator müssten im Sourcecode Weichen eingebaut werden,  um Abwärtskompatibilität zu gwährleisten.** | (:cross:):tick: | :tick: | :tick: |
| **Kein erhöhtes Risiko für System Wide Failure**  **(Bsp Tekton Namespace Operator ->  Secrets Voiding in allen Tekton Namespaces) Ein Bug im Operator kann alle Instanzen auf einmal zerstören.  Bei einem Job oder Pipelinerun release, betrifft es nicht alle auf einmal.** | :tick: | :tick: | :tick: |
| **Fehlerhandling des Job-/Operators (Logs) wenn z.B.  eine appRegistration nicht applied werden konnte  (Im Operatorfall als Annotation auf ConfigMap oder Status in der CRD)** | :tick: | :tick: | :tick: |
| **Vorgesehen für Funktionalität um Modifikationen ausserhalb des Clusters durchzuführen** | :tick: | :tick: | :tick: |



