**Boxnote**

Voir pour s’inspirer <https://github.com/niklaushirt/ibm-aiops-deployer>

**Customisation pour chaque produit**

ACE

* Un flux pour les taxis, qui génère un Slack message (Toolkit) - Saad
* Génération de BAR (aspect IS versus BAR) - Plusieurs BAR files dans un IS - Saad

APIC

* Topology is already done
* Create Mail Server - Arno
* Vérifier que EEM est enabled (pour Event Endpoint Gateway) - Saad ou Arno
* Une organisation org1 - Arno
* Création de 2 catalogues avec leur portails (Sandbox/QA//Prod) - Arno
* Ping API - Arno (scripts/14a-apic-create-apis-draft.sh) - Producer: publish Ping API to prod catalog
* Peststore API (Swagger.io) en OAS 2 et 3
* Producer: scripts/14b-apic-config-catalogs-publish-apis.sh
* Create consumer-organisation + user (scripts/14c-apic-new-consumer-org.sh)
* Consumer side: scripts/14d-apic-create-apps-subscription.sh, create app/subscribe to API
* Un user (owner) org1owner - Arno
* Un example OAuth - Arno à relier avec PetStore
* Vérifier que l’on a les async API (Les 2) (<https://ibm.ent.box.com/notes/953089231688>) Taxi Locator et Taxi Finding - Arno
* L’API async pour le lab taxi - Arno

AR (Automation Foundation Assets)

* On va mettre un asset (bar file) - Saad

AsperaHSTS

* Ajouter un envoi du résultat dans un fichier - Saad ou Laurent

ES

* Topic pour Taxi - Arno ou Saad
* Kafka User - Arno ou Saad

MQ

* QM et queues pour Taxi s’inspirer du repo d’Emir Garza (HA Native avec Sécurité) - Saad

DataPower

* MPGW + Skip-backside - Arno

LDAP

* La configuration LDAP Arno ou Saad
* Quelques users dans LDAP (LDIF) - Arno

**Autres**

* Monitoring - Saad
* Containers NodeRed pour Taxi - Arno
* Génération de certificats et surtout les certificats exposés dans les browsers (aider, par exemple extraire le root et le mettre dans un fichier temporaire) - Arno
* Et les certificats end to end (multi-produits)
* (S’inspirer de ce qu’a fait Joel: <https://github.ibm.com/joel-gomez/cp4i-demo>)
* Ajouter les times stamps : début opération - fin opération (pour les souscriptions et les capabilities).
* License service/ ibm-license-service-reporter-instance
* Subscriptions : [asset-repo, api-connect, event-streams, mq, app-connect, aspera-hsts], eem, ea-flink, event-processing

**Questions pending**

* Pipelines and/or CI/CD and/or gitops?
* Async API avec MQ?
* For the mail Server mailtrap?
* ELK? Instana?
* Cluster and user access security? Better segregation (remove admin user)?
* Event Automation?

**Design**

Structure:

customisation/<ACRONYM produit>/scripts pour les scripts

customisation/<ACRONYM produit>/scripts/config pour les fichiers de config (nottament JSON, YAM, WSDL, ...)

Principe: idempotence, coder le unconfigure pour chaque composant.

Aujourd’hui on a le choix d’installer un composant (produit)

On veut garder le choix de customisation au niveau de chaque composant (produit) ou global: On a décidé que ce soit pour chaque produit