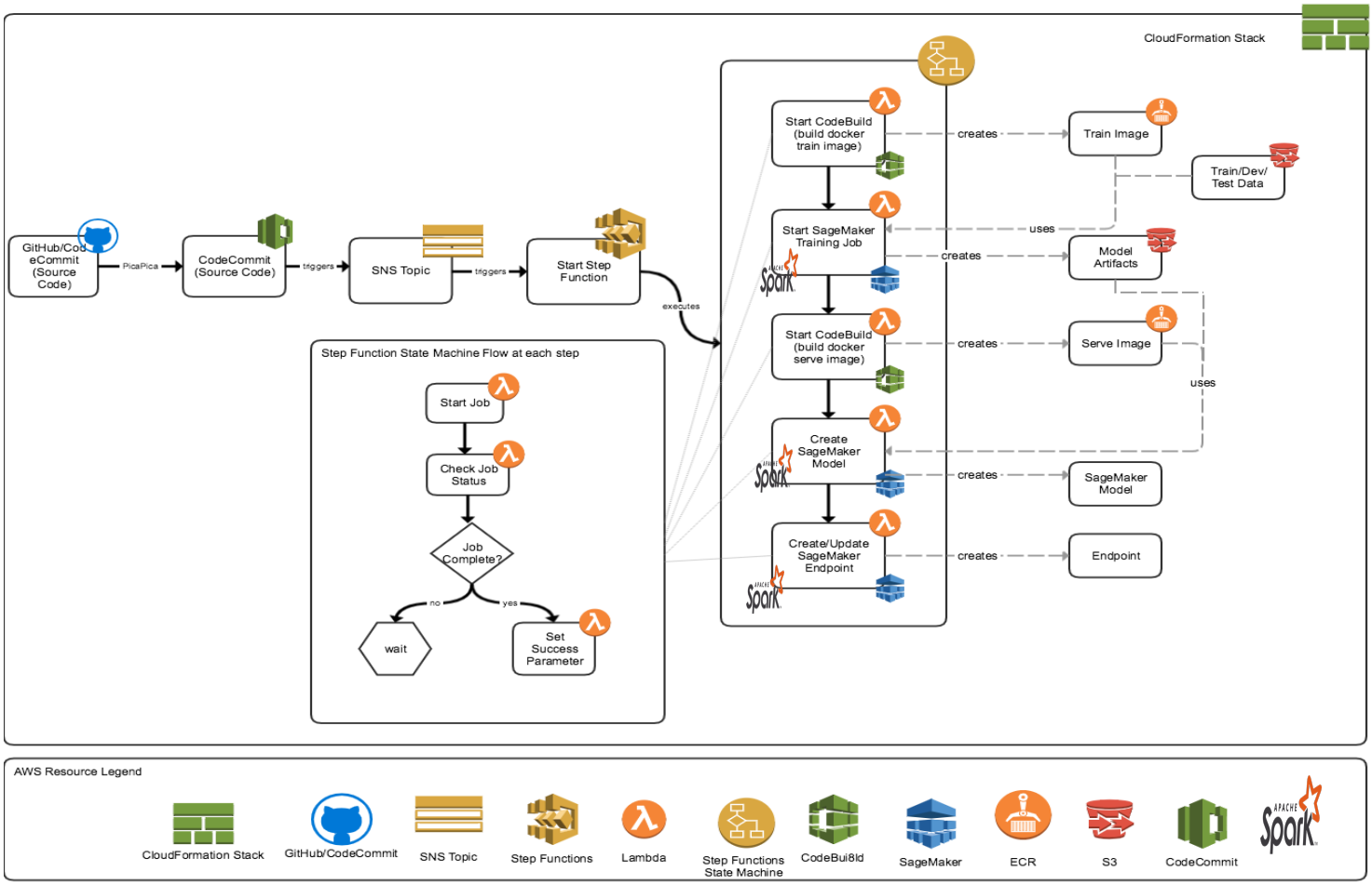
AWS Pipeline for Submission Ratio ML Model:



This pipeline covers from pushing code to Github till deployment.

Training — When Sagemaker creates the training job, it launches the ML compute instance, runs the train docker image which creates the docker container in the ML compute instance, injects the training data from an S3 location into the container and uses the training code and training dataset to train the model. It saves the resulting model artifacts and other output in the S3 bucket you specified for that purpose. The training pipeline we might use spark for training, create docker image which can be consumed by Sagemaker.

Deployment — For model deployment, Sagemaker first creates the model resource using the S3 path where the model artifacts are stored and the Docker registry path for the image that contains the inference code. It then creates an HTTPS endpoint using the endpoint configuration which specifies the production model variant and the ML compute instances to deploy to. Here Inference pipeline of spark model can be converted to docker image which can be consumed by Sagemaker for inference.

AWS Step Functions: Used for workflow orchestration. Using Step Functions here are used for orchestration of workflow.

AWS CodeBuild: For generating docker images to push to ECR

S3 Buckers: For storing model training data and trained model artifacts.

AWS Lambda: It can be used to trigger model training or check training status etc.

Apache Spark: Apache spark is used for enabling distributed computing of model training and inference.

Weak points: To scale the system to handle million requests a minute we should probably go for streaming solution instead of simple API calls.