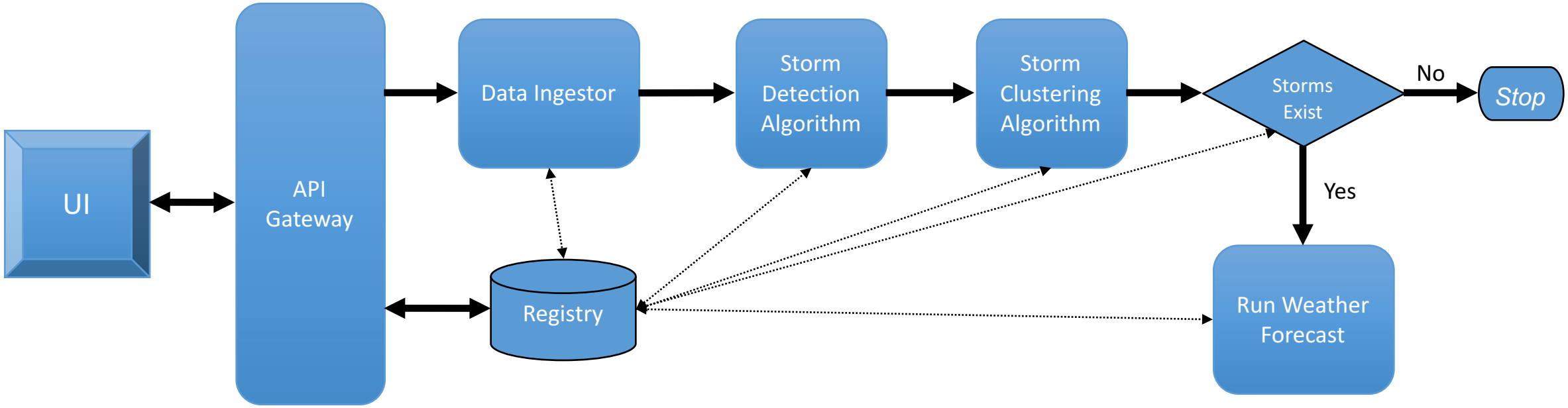
Practical Introduction to Continuous Integration and Delivery

Needed to complete Project Milestone 1

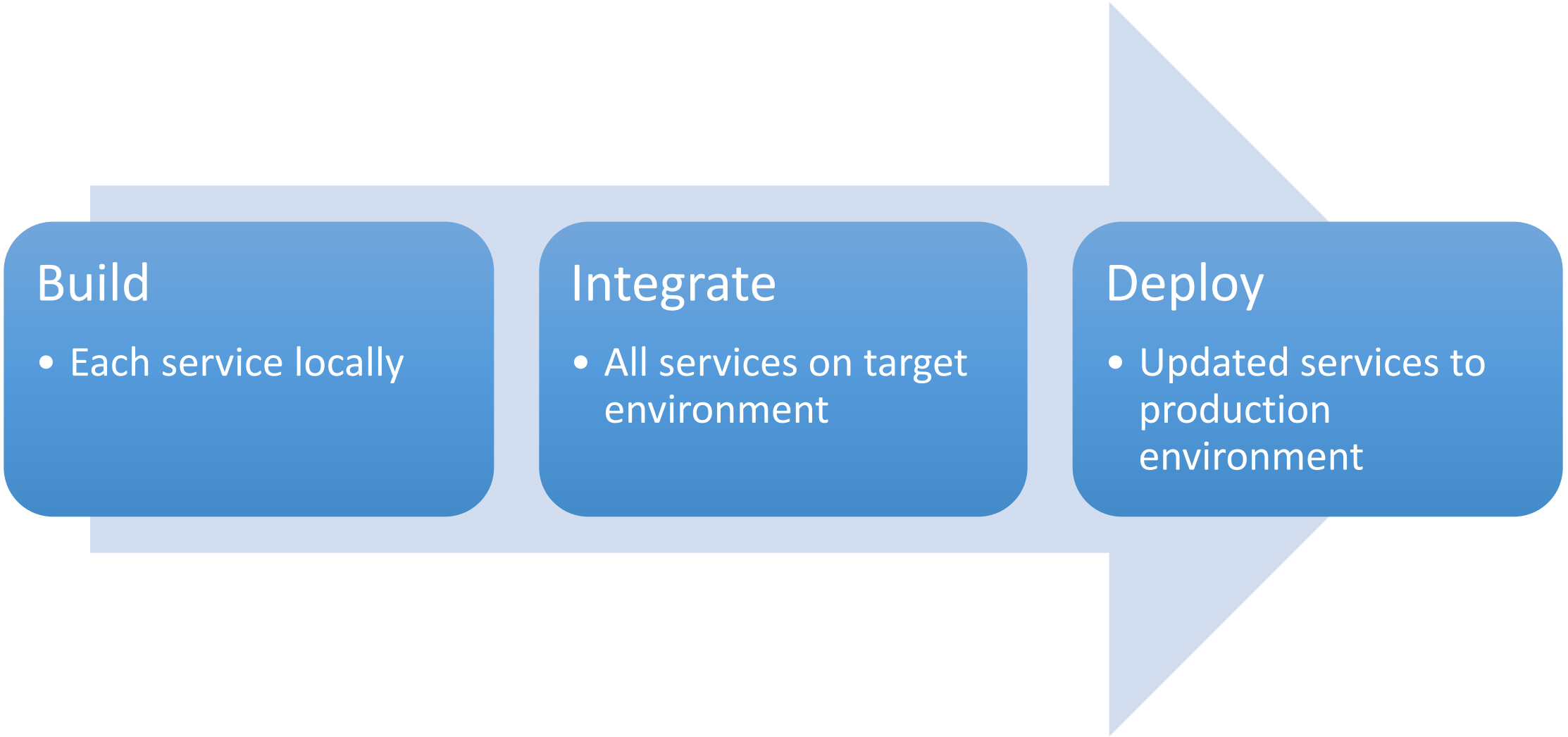
Project Milestone 1 Microservices



Components need to be implemented in multiple programming languages and deployed into operations.

Software	Software as a Service
Release software artifacts (zips, tars, dmgs, rpms, etc).	Continuously update services that are composed of multiple internal microservices.
Monolithic build system	Multiple build systems for each microservice.
Many layers of testing of entire monolith	Limited testing of each service; full system testing more important
One release	Each service gets released, or goes into operation, independently.

Continuous Integration and Delivery Pipeline



What are some examples of build systems?

You will need more than one

Record your discussions and conclusions using GitHub Issues

Continuous Integration and Deployment

- For Project Milestone 1, use Travis-CI and Amazon Cloud Deploy
- https://github.com/airavata-courses/TeamApex/wiki/Milestone-5-Guide-to-Setting-Up-Amazon's-CodeDeploy-Travis-Integration
- https://github.com/airavata-courses/TeamApex/wiki/Milestone-5-Testing-Continuous-Integration-and-Deployment
- One of the course AI's will help you get set up.

In Summary

- You need build systems for each service.
- Connect your GitHub projects to Travis-CI to trigger builds.
  - Only do this for the Project Milestone branches.
  - Use .travis.yml to specify what you want Travis to do
- Connect Travis-CI to Amazon Code Deploy
  - This copies your build artifacts from Travis-CI to Amazon
  - Use appspec.yml file to tell Amazon what you want to do with these build artifacts.
- Look at Spring 2016 teams' Git repos for examples.