

2016-04-20 What is NDS Labs?

These are notes from a whiteboard discussion about the NDS Labs "big picture"

| | NDS Labs "Test Drive" | NDS Labs "Developer" | NDS Share "Staging" | NDS Share |
|-----------------------|--|---|---|---|
| What is it used for? | <ul style="list-style-type: none"> • Exploration • Education/training • Comparative evaluation • Development (Explore what capabilities we can add to help developers addressing interoperability!) | <- Merge? Development Prototyping | <- Merge? Alpha/beta testing | <ul style="list-style-type: none"> • Production releases <ul style="list-style-type: none"> ◦ From Labs as well as external resources (e.g. DataONE, NEON, Libraray archives, ...) |
| When is it used? | <ul style="list-style-type: none"> • Proposal stage • Demos/trials/test drives • Tutorials • Application development | Post-proposal development and integration | Alpha/beta testing | Production deployments |
| What does it include? | <ul style="list-style-type: none"> • Functional services • Sample data | Customized services Realistic but limited resources Fully-configured common services (e.g., iRODS federation, Shibboleth IdP, Swift storage) | Stable releases of integrated services Real users Real resources Real users Stable release | Actual scale resources SLA Security/Hardening |
| Analogy | <ul style="list-style-type: none"> • Trial installations, Test installations, Quick test instances | Development VMs | Staging system | Production system |

- Workbench/Test Drive: This is the current workbench service – supporting the ability to "test drive", demo, or trials services. This would be limited to services and configurations that are easily deployable via Kubernetes
- Developer: When users graduate from the Workbench, they are given access to real resources for development and integration. This might include services not easily available via Workbench such as a functioning iRODS preservation federation, Swift Object store, Shibboleth IdP, etc. Services wouldn't be started and stopped quickly – configuration options would be specified and services configured based on project requirements.
- Staging: When projects are ready for alpha/beta/rc testing, they would deploy their services to a staging environment. This would be limited-access, but on real resources (i.e., actual production configuration, but maybe not scale)
- Share: Once a project is ready for production deployment, it is moved to the "Share" environment. This could be hosted at AWS, GCE, TACC /SDSC/PSC/NCSA.

A note about Nebula

We spoke with the Nebula team about deploying "production" systems (services that need SLA/uptime guarantees). Nebula is currently not ready – for example, one compute node is failing, there's no UPS, the backend isn't parallel. Production services are currently running on VSphere/VMWare