

# DSpace Docker and Cloud Deployment Goals

- [Problem Summary](#)
  - [Assumption](#)
- [Goal 1: Simplify and standardize the process for creating a development environment for all supported versions of DSpace](#)
  - [Docker provides a consistent, predictable environment](#)
  - [Docker allows a user to manage multiple \(and incompatible\) environments](#)
- [Goal 2: Publish a standard set of AIP files \(archival ingest packages\) to facilitate system testing](#)
  - [What test resources exist?](#)
  - [What test resources should exist?](#)
  - [How should these test resources be distributed?](#)
- [Goal 3: Create a simple workflow to deploy a DSpace branch or snapshot to the cloud](#)
- [Goal 4: Manage hosted instances of DSpace for each supported branch of the system](#)
- [Goal 5: Publish instructions for a replicable development environment for on-boarding, tutorials, and troubleshooting](#)
- [Implementation Costs](#)

## Problem Summary

Configuring and managing DSpace test and development environments requires significant effort.

When developing and testing against multiple versions of DSpace, the management of system software becomes significantly more complex especially when there is a need to downgrade versions of software.

DSpace 4, 5, and 6 require Postgres and Tomcat as prerequisite software.

A full DSpace 7 instance will require Postgres, Tomcat, Node (angular), and Solr as pre-requisite software. A developer may wish to focus on either the full set of components or a subset of these components.

## Assumption

The complexity of managing these development and test environments has likely prevent some institutions from contributing to the platform.

The project might be able to engage a larger audience of developers if the prerequisites for creating a development environment could be simplified.

Many DSpace stakeholders are not developers. If it was easier to deploy a snapshot instance of DSpace to the cloud, these stakeholders could play a more active role in system testing.

## Goal 1: Simplify and standardize the process for creating a development environment for all supported versions of DSpace

### Docker provides a consistent, predictable environment

Docker provides users with a consistent and predictable runtime environment. With such an environment, new DSpace developers will be able to more easily isolate local environment inconsistencies when seeking help from the DSpace community.

### Docker allows a user to manage multiple (and incompatible) environments

Docker containers allow a developer to manage and run multiple system configurations from a single desktop or test server.

Significant work has already been done in this area. <https://github.com/DSpace-Labs/DSpace-Docker-Images>

Each major branch of DSpace and each recent release of DSpace has been published as a Docker image.

## Goal 2: Publish a standard set of AIP files (archival ingest packages) to facilitate system testing

In order to expedite and simplify DSpace testing, it would be useful to provide developers with content suitable for testing a majority of DSpace use cases.

Fortunately, the DSpace AIP ingest process provides a mechanism for constructing a DSpace repository that is DSpace version agnostic.

### What test resources exist?

We have generated one simple set of re-usable AIP files posted on GitHub. <https://github.com/DSpace-Labs/AIP-Files>

These files are too simplistic for testing real DSpace use cases.

Demo.dspace.org is populated with a set of AIP files that are stored on Amazon S3. These files are approximately 1.5G in size.

A regular GitHub repository is not appropriate for sharing large files. See <https://help.github.com/articles/working-with-large-files/>.

## What test resources should exist?

- A community/collection hierarchy containing items of various sample document types (PDF, image) that illustrate DSpace features including some access-restricted material.
  - These examples should be created from contrived examples or from open-access resources.
- A community/collection hierarchy containing multilingual metadata
- A community/collection hierarchy containing some digital collection material

## How should these test resources be distributed?

The collections of material that are likely to be shared will exceed the space and bandwidth limitations of a GitHub code repository, so there are likely to be some storage and bandwidth costs associated with sharing these resources.

A test collection would need to be downloadable as a single zip file or as a collection of individual zip files.

- Explore GitHub LFS (large file storage) as an option
- Share resources through a cloud service such as AWS S3.
- Share resources using a collaboration platform such as Box or Google Drive
- Upload these assets to a data repository and share them from that platform

## Goal 3: Create a simple workflow to deploy a DSpace branch or snapshot to the cloud

Each of the major cloud providers provides a mechanism for running Docker containers.

If we could streamline the process for deploying published Docker images to the cloud, we could support a unique test environment for each major branch of DSpace. This would allow users to compare and contrast functionality on each version of DSpace. It would also allow an institution to evaluate the functionality on a specific DSpace branch.

[Here](#) is a write up of my experimentation with Docker Images running on specific cloud providers.

## Goal 4: Manage hosted instances of DSpace for each supported branch of the system

If the prior goals are achieved, it would be possible to manage multiple hosted instances of DSpace for each major branch of the system in addition to the production reference version at [demo.dspace.org](http://demo.dspace.org).

## Goal 5: Publish instructions for a replicable development environment for on-boarding, tutorials, and troubleshooting

Frequent contributors to DSpace are likely to manage and maintain a robust development environment that includes best of class development tools such as IDE's.

There is also a need to assist new contributors and potential contributors with the creation of a simple and basic development environment for DSpace.

While Docker can standardize the runtime and test environments, it may be possible to document a replicable development environment to help with on-boarding.

See [Using Published Images on AWS Cloud9](#) as a possible example.

## Implementation Costs

- Hosting AIP resources (storage and bandwidth)
- Hosting multiple DSpace reference instances (dspace-4x, dspace-5x, dspace-6x, dspace-7x) in addition to [demo.dspace.org](http://demo.dspace.org) (compute, storage, bandwidth)