



# **Hands-On with DART Workshop**

---

DPC Americas Member Forum

April 25, 2025

Nathan Tallman, APTrust

# Workshop Goals

- Learn DART basics
- Create & validate bags
- Set up reusable workflows
- Perform batch processing via CSV

## Download Workshop Materials



<https://bit.ly/42zmuwZ>

# Quick Intro to DART

- Free, open-source, cross-platform
- GUI and command-line utility
- Used for digital preservation & transfer
- Non-APTrust use cases:
  - internal transfers
  - donor submissions
  - deposit to preservation repository
- If you haven't already done so, install [DART 2](#) and [FileZilla Client](#) on your laptop.

# Hands-On Task 1: Configure SFTP Storage

- Navigate: **Settings** → **Storage Services** → **New**
- Enter values for:
  - Name: **Nathan's Home Server (SFPT)**
  - Description: **Personal FTP server of instructor Nathan Tallman.**
- Enter connection details provided (see Participant Guide)
  - Protocol:
  - Host:
  - Port:
  - Login:
  - Password:
- Save the connection (DART 3 will let you test it.)

# Hands-On Task 2: Create & Validate a Bag

## Create a Bag

- **Jobs → New**
- Drag and drop files. Click **Next**.
- Select Packaging options
  - Package Format: **BagIt**
  - BagIt Profile: **Empty Profile**
  - Serialization: **.tar**
  - Package Name: **[enter an appropriate package name]**
  - Make **note of the output path** (this can be changed in settings).
- Add basic metadata
  - Click on **Show All Tags** to reveal fields automatically populated by DART
  - Click **Next**.
- Choose an upload target
  - **Nathan's Home Server (SFTP)**
  - Click **Next**
- Run job to create, validate, and upload the bag.

## Validate a Bag

- **Jobs → Validate a Bag**
- Validation Settings
  - BagIt Profile: Empty Profile
  - Bag is a tar file
  - Use the file picker to select the bag (was in the output path on previous screens).
- Click **Validate**

# Hands-On Task 3: Set Up & Run Workflow

- **Workflows → New**
- Workflow Settings
  - Name: **Nathan's SFTP Server**
  - Description: **Send bags to Nathan's SFTP server.**
  - Package Format: **BagIt**
  - BagIt Profile ID: **APTrust** (Sometimes there's a bug where you have to save the Workflow with the BagIt packaging format and then edit it again to choose the BagIt Profile.)
  - Upload To: **Nathan's Home Server (SFTP)**
  - Click **Save**.
- Run workflow with new files
  - **Workflows → Run Nathan's SFTP Server**
  - Add files.
  - Name the package and click **Next**.
  - Add basic metadata and click **Next**.
  - Click **Run Job**.
- Confirm transfer via SFTP viewer (e.g., FileZilla)

**WARNING:** DART2 has a bug because it can only transfer files, not directories. Because of this, you have to configure the Workflow to use a BagIt profile that requires serialization (e.g., converting the bag to a TAR file). You can use the APTrust profile, but know it will add more required metadata fields and create an extra tag file (aptrust-info.txt).

**WORKAROUND:** If you need to create and transfer unserialized bags, you can create them with DART and upload them with an FTP client such as FileZilla.

# Hands-On Task 4: Batch Workflows via CSV

- Batches let you feed DART a CSV of bag metadata and paths, then loops each bag through a workflow.
- Review and update the provided CSV sample
  - Update the Root-Directory field for the full path on your system. Do not enclose paths in single or double quotes, even if there are spaces.
- **Workflows → Run Batch**
  - Choose the **workflow**
  - Use the file picker to browse the the **CSV**.
  - Click **Run**.
- Confirm bags created & transfer via SFTP viewer (e.g., FileZilla)

# Troubleshooting Quick Tips

- Common issues:
  - Incorrect storage service details
  - Bag validation errors (missing metadata, wrong serialization)
  - Upload errors
  - CSV formatting errors
- Refer to provided troubleshooting checklist

# Additional Resources

- [DART Documentation](#)
  - [DART 2](#) has a command-line implementation, but it cannot be run in a headless environment.
  - [DART Runner](#) is the headless command-line companion, Jobs and Workflows can be configured in DART, copied as JSON, and passed on to DART Runner for execution. This is useful when creating automated pipelines.
- [DART User Group](#)
  - Join mailing list for community support and feature discussions.
- [DART 3](#) is the next-version of DART, rewritten from the ground up in Golang. It will have better support for large, unattended file transfers.
  - DART3 is available as an alpha preview in a web app implementation. It will likely be changed back to a desktop app in a beta release sometime this fall.
  - The alpha is a complicated install, but you're welcome to try it out.

# Questions & Wrap-Up

- Open Q&A
- Feedback and further assistance