DPN
THE DIGITAL PRESERVATION NETWORK

DPN SCENARIOS V1.1

DPN Technical Group

DRAFT

Table of Contents

Digital Preservation Network (DPN) Scenarios

Version 1.1

This document defines scenarios of use for the Digital Preservation Network (DPN). This is a living document, and as such is not complete. It is a companion document of "DPN Services, Interactions & Processes".

Context of Scenarios

DPN provides resilience by providing redundant copies and preservation services for content stored within its federation of replicating nodes. In addition to basic services such as replicate/copy, fixity, audit, and recovery, DPN also provides an avenue for succession planning by allowing entities that place content into DPN to have confidence that their content will be maintained and preserved in perpetuity. In the event of the dissolution of a content holder/owner, DPN will have succession rights agreements in order to "brighten" the content.

Operationally DPN will support (initially) five replicating nodes, all of which are equal. A node that emplaces content into DPN is called a First Node, and the nodes that receive content are Replicating nodes. All nodes can act as a First Node, or a Replicating Node depending on if they are emplacing content, or replicating content.

A DPN replicating node will act as a First Node and work with DPN members directly to negotiate contracts, determine service levels, and deposit materials into DPN via the First Node. Service levels and contracts will reflect "standard" DPN services; they may also reflect the First Node's unique offerings in terms of access, hosting or other services.

Scenarios Covered

There are four basic scenarios:

1. Place content in DPN for preservation
2. Restoration of content at First Node
3. Succession Event on dissolution of First Node
4. Place encrypted content in DPN using third party key escrow

Scenarios

Example Scenario 1 – SDR places SDR content in DPN

This scenario comes about because of a decision made by a content owner, or the Stanford Digital Repository, regarding content owned/managed by SDR. The decision is to replicate content to DPN.

We Own the Content, but need DPN agreements

In general as part of SDR's process, content is ingested and accessioned into the repository, during this process, accessioning agreements are created and assigned to / associated with content. Once content is flagged for replication by DPN, SDR will review the DPN accessioning/succession rights framework agreements and assign/associate an agreement that allows content to be emplaced in DPN.

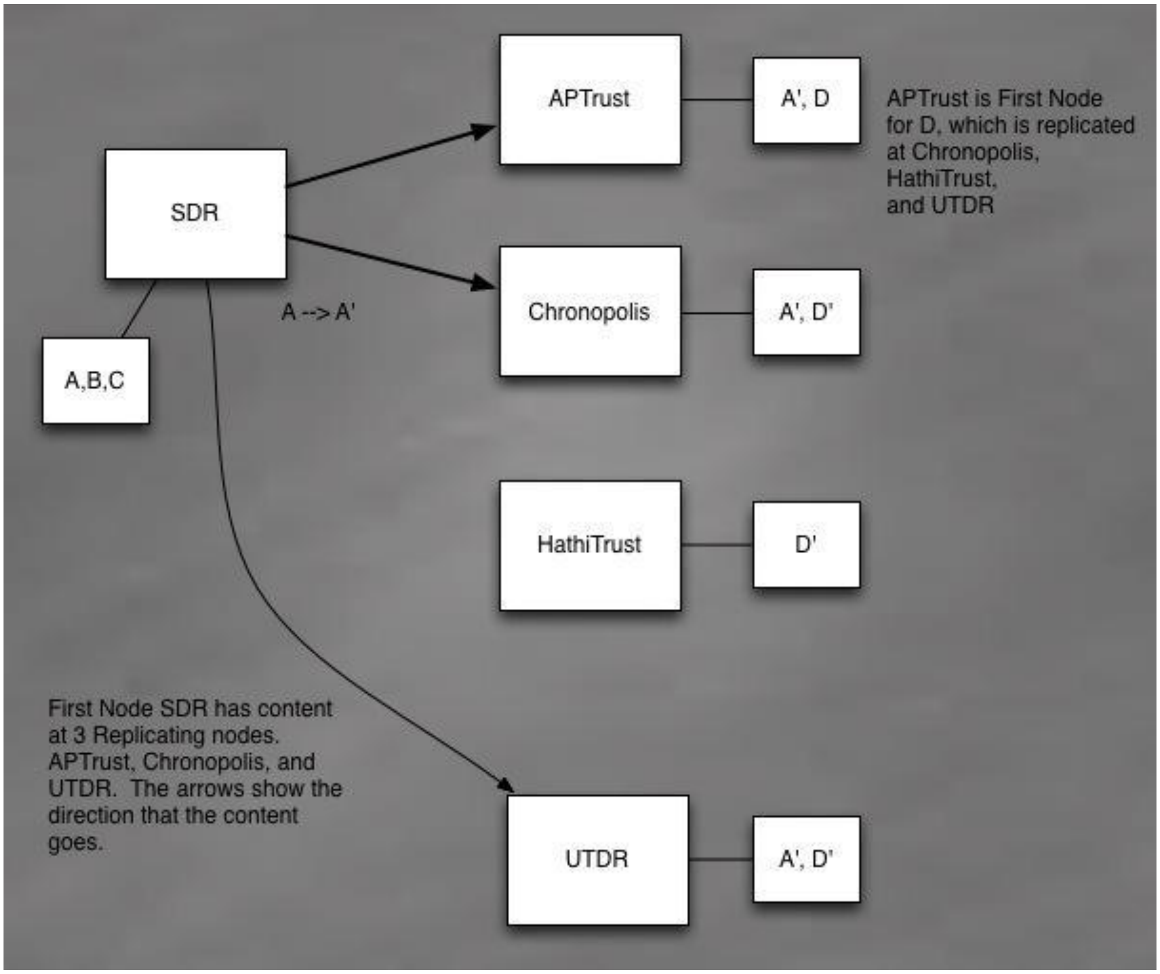

APTrust
A', D
APTrust is First Node for D, which is replicated at Chronopolis, HathiTrust, and UTDR
SDR
A --> A'
Chronopolis
A', D'
A,B,C
HathiTrust
D'
First Node SDR has content at 3 Replicating nodes. APTrust, Chronopolis, and UTDR. The arrows show the direction that the content goes.
UTDR
A', D'

Figure 1 - After content is replicated from SDR, showing direction of content flow.

Packaging Content and Replicating via DPN Replicating Nodes

Once SDR has created a legal framework with proper agreements, the next step is to replicate the content within the DPN framework. Figure 1. illustrates the end state for simple replication, i.e., where content represented by ***A*** is delivered to three Replicating nodes and shown as ***A'.*** The arrows show the directionality of content.

Technical flow for DPN ingestion of content

SDR

1. SDR identifies content, or ingests content that will be placed into DPN.
2. SDR creates agreements for content that allow DPN to archive/maintain content over time. This also includes a legal framework for succession, brightening and deletion of content.
3. SDR wraps SDR AIP(s) using DPN Profiled Bagit bag, to include:
    a. DPN GUID and SDR UUID
    b. DPN Bagit includes DPN specific tag files
    c. DPN Bag includes common check summing
    d. DPN checksum used for audit of DPN objects
    e. Descriptive metadata; Formats? Provenance? Rights?
4. SDR places DPN objects in transfer area
5. SDR notifies transfer/syncing application that there are DPN objects to upload
    a. Notification is parameterized with transfer location
    b. Notification handshake contains DPN UUID orr SDR UUID
    c. Protocol tracks and logs transfer process

DPN Transfer/Syncing application (SDR)

1. Confirms transfer area exists and has valid DO(s) for transfer
    a. Checks validity of DPN Bag and validates checksum(s)
    b. DPN Bag version, format, etc.
    c. Common checksum format (as specified by Bagit)
2. DPN Replicating Node pulls content from transfer area, acknowledge receipt
3. First Node updates local DPN Registry on acknowledgement of validated transfer
4. Transfer application notifies DPN nodes (replicating nodes)
    a. Update Registry
    b. Update Registry of non-copy nodes
5. Registry updated by each DPN Node
    a. Transfer complete, checksum validated
    b. DPN node(s) holding content
    c. DPN non-copy nodes have updated Registry
6. Clean up of transfer area

Example Scenario 2 – Restoration of content at First Node

The background for this scenario is this: A First Node discovers that it is missing content (some or all) and for some reason their normal recovery processes are unable to restore the missing content to the repository. The First Node, using DPN services, restores its content and confirms that it is valid.

Although this scenario is unlikely, it might come about due to some natural disaster that affects a large swath of territory, such as earthquake or hurricane, where the natural disaster encompasses not only the primary repository site, but also its backup sites. This scenario might also come about if a site is compromised and the repository administrators have little confidence that they hold valid content.

Once content is deemed unrecoverable, the First node must send out a request for DPN content to the DPN replicating nodes. This can follow three possible cases:

1 The First Node has none of its DPN content, and requests restoration of its registry of content, and the content itself.
    a The replicating nodes return content to the First Node and the First Node restores its repository. In this case, the First Node will also have to restore its Registry of DPN content.

2 The First Node has a partial failure and requires some of its DPN content.
    a If the First Nodes Registry of content is intact, the First Node can request content directly from its Replicating nodes.
    b If the First Nodes Registry of content is not intact, then it will have to recover its Registry first, then request content from Replicating nodes.

Technical example of content recovery

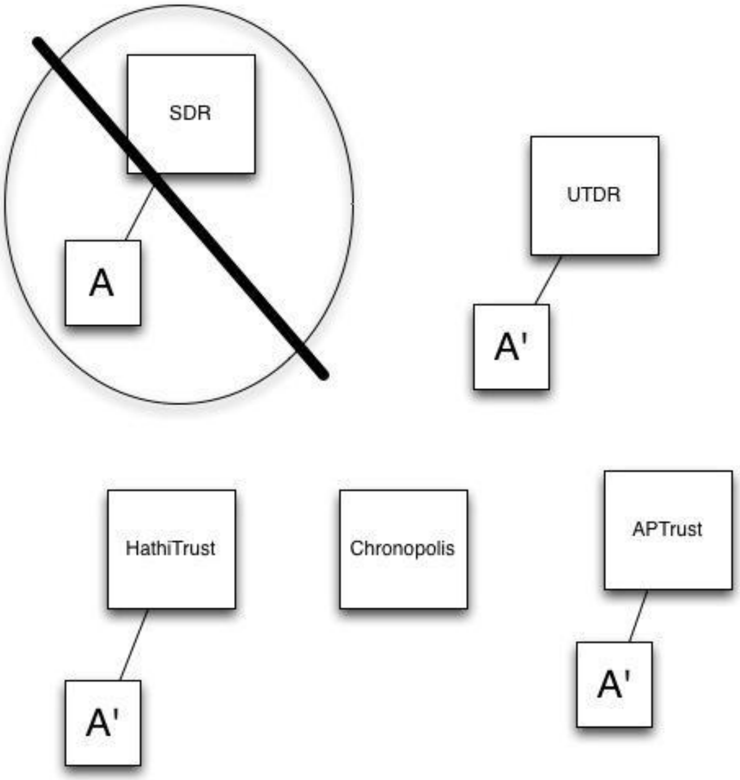
SDR recovers content from DPN Nodes

1 SDR notifies DPN nodes that it needs Registry recovery (as DPN First node for content)
2 SDR identifies DPN objects from Registry
3 SDR notifies through transfer/syncing application that there are DPN objects to recover
    a Notification is parameterized with transfer location
    b Notification handshake contains DPN UUID and SDR UUID from Registry, possibly digital signature
4 SDR receives content, validates DPN content
5 SDR moves from transfer area to SDR preservation area
6 SDR validates Registry
10) SDR logs recovery event completion

Example Scenario 3 - DPN Succession Event / Brightening Scenario

This scenario represents the core value of DPN, i.e. a repository dissolution occurs and preserved digital content is _not_ at risk for being lost ***forever***.

Using SDR as an example:



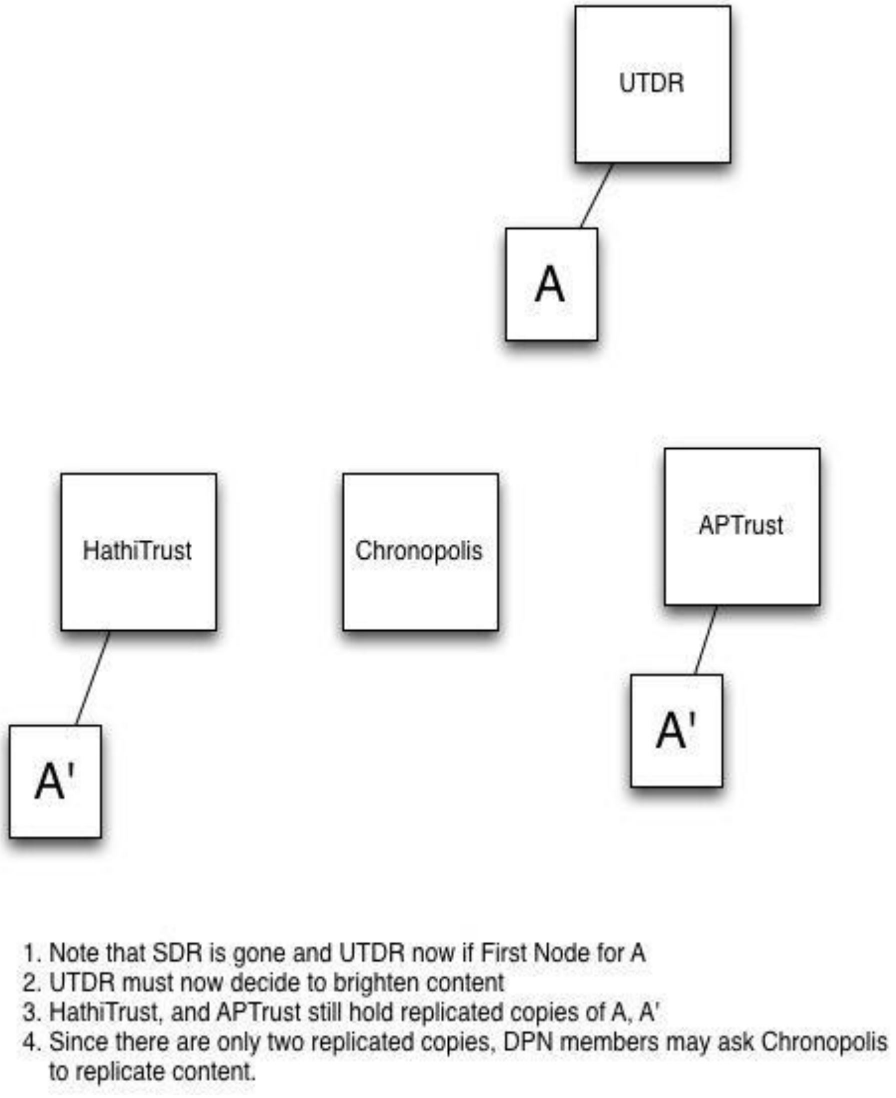
1. First Node dissolution trigging succession and brightening
2. SDR was First Node for content A
3. UTDR, HathiTrust, and APTrust hold replicated copies of A, A'

Figure 2 - SDR as First Node fails, content replicated.

In this scenario SDR has dissolved as a repository, this triggers a succession event. The example illustrated in Figure 2 shows that SDR has failed and that content held by SDR is also held at UTDR, HathiTrust, and APTrust. Once the succession event is triggered DPN succession agreements start a process to transfer First Node rights to one of the existing DPN replicating nodes. This is not an automated process. For example, if UTDR agrees (negotiates), it may take over the primary rights and control of SDR's content. Illustrated in Figure 3.

As part of the DPN agreement, content may be brightened for its community of users.

Of primary concern is what it means to brighten content. Part of DPN's charter is to provide information that supplies descriptive, technical, institutional, legal, and operational regarding content that needs to be brightened.



1. Note that SDR is gone and UTDR now if First Node for A
2. UTDR must now decide to brighten content
3. HathiTrust, and APTrust still hold replicated copies of A, A'
4. Since there are only two replicated copies, DPN members may ask Chronopolis to replicate content.

Figure 3 - UTDR assumes responsibility for SDR content A.

The end result of this scenario is that a new First Node has taken over responsibility for content from a failed repository. In the best case, the new First Node also fully brightens content for its community of users, thus providing access.

Other options might include take over of basic responsibility for content, but not access. This might be the case where a community of interest is no longer present. However, it is useful to note that the content itself will be brightened and maintained.

Example Scenario 4 – DPN Stores Encrypted Content using Key Escrow

Scenario: SDR ingests content that has agreements (with the content originator) that require the content to be encrypted. The content creator along with SDR, or SDR alone, decide that the content is sufficiently valuable for long term preservation protection via DPN.

SDR must encrypt content and then using DPN standard packaging, package the content for inclusion in DPN. Along with preservation brightening information, SDR must also include a reference to the key, or keys so that the content can be decrypted.

In this scenario, DPN has decided to use a third party escrow model to hold brightening information pertaining to encryption. The brightening information itself need only reference the escrow holder. It is important to note that the day to day activities require that this brightening information be kept up to date. This may include managing encryption strength over time and escrowed keys and third party escrow companies and legal contracts and agreements.

Reasoning for third party escrow

To protect content, from non First Nodes, the standard approach is to use a third party escrow for holding keys. The non-first nodes do not have legal access to the keys via the escrow, so the content is protected. In the event of a succession event, the new First node will inherit the legal rights (and agreements) to access the key(s) from the third party escrow holder.

The net result is that content can be encrypted and stored in DPN *and* be protected from prying eyes of replicating nodes. A succession event would trigger the transference of legal agreements, including escrow accounts, to the new First node.

Figure 4 shows the relationship to content and key location. Note that the replicating node(s) only hold a reference to the escrow and not the key(s) themselves.

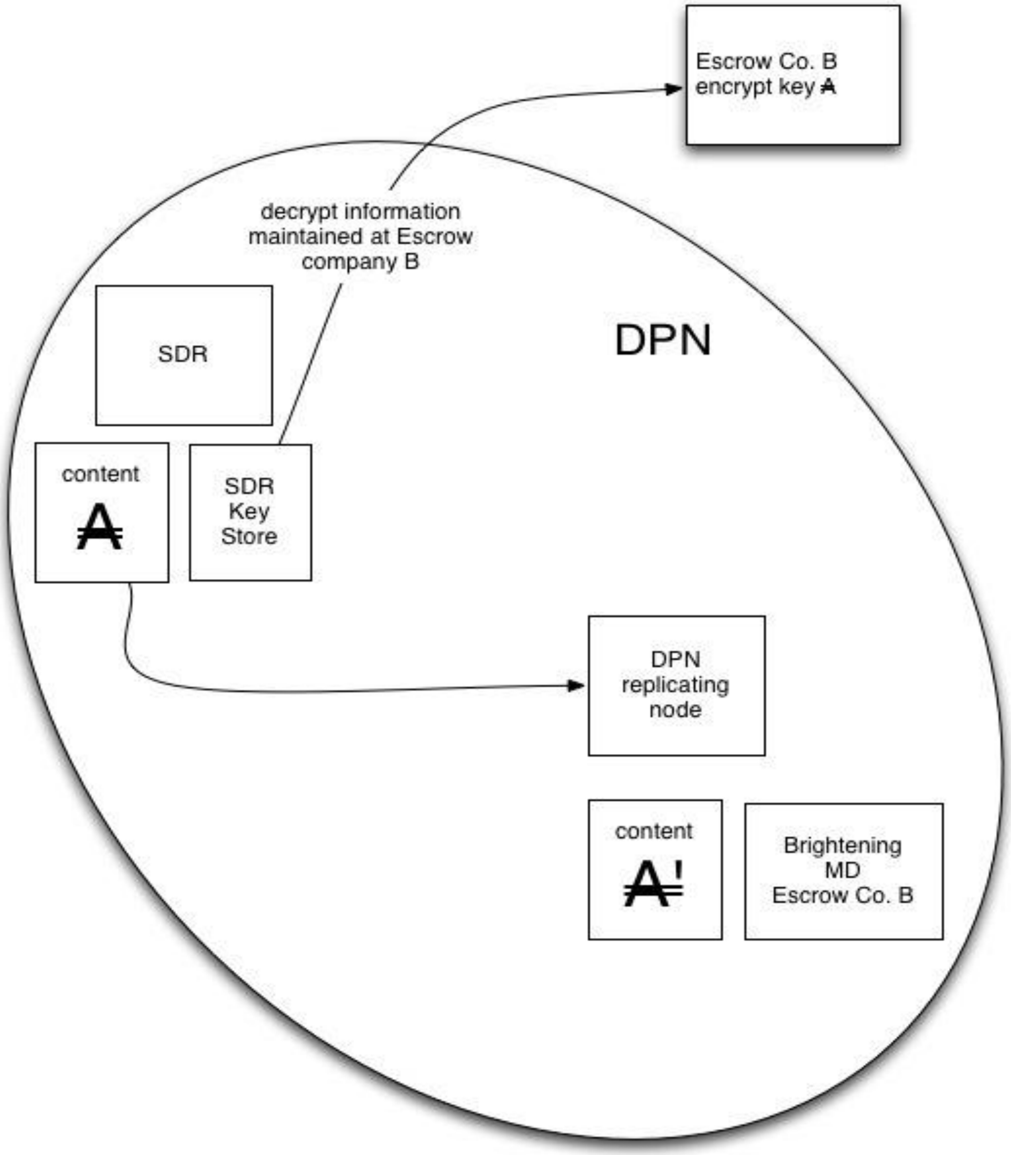


Figure 4 - SDR placing encrypted content into DPN with third party escrow of encryption information.

Further Scenario List

In addition to the scenarios elucidated in this document, below is a list of other potential scenarios for DPN:

1. Another node is added to DPN.
2. There is a challenge to disseminate or purge content in DPN from a 3rd party.
3. *More to be defined*