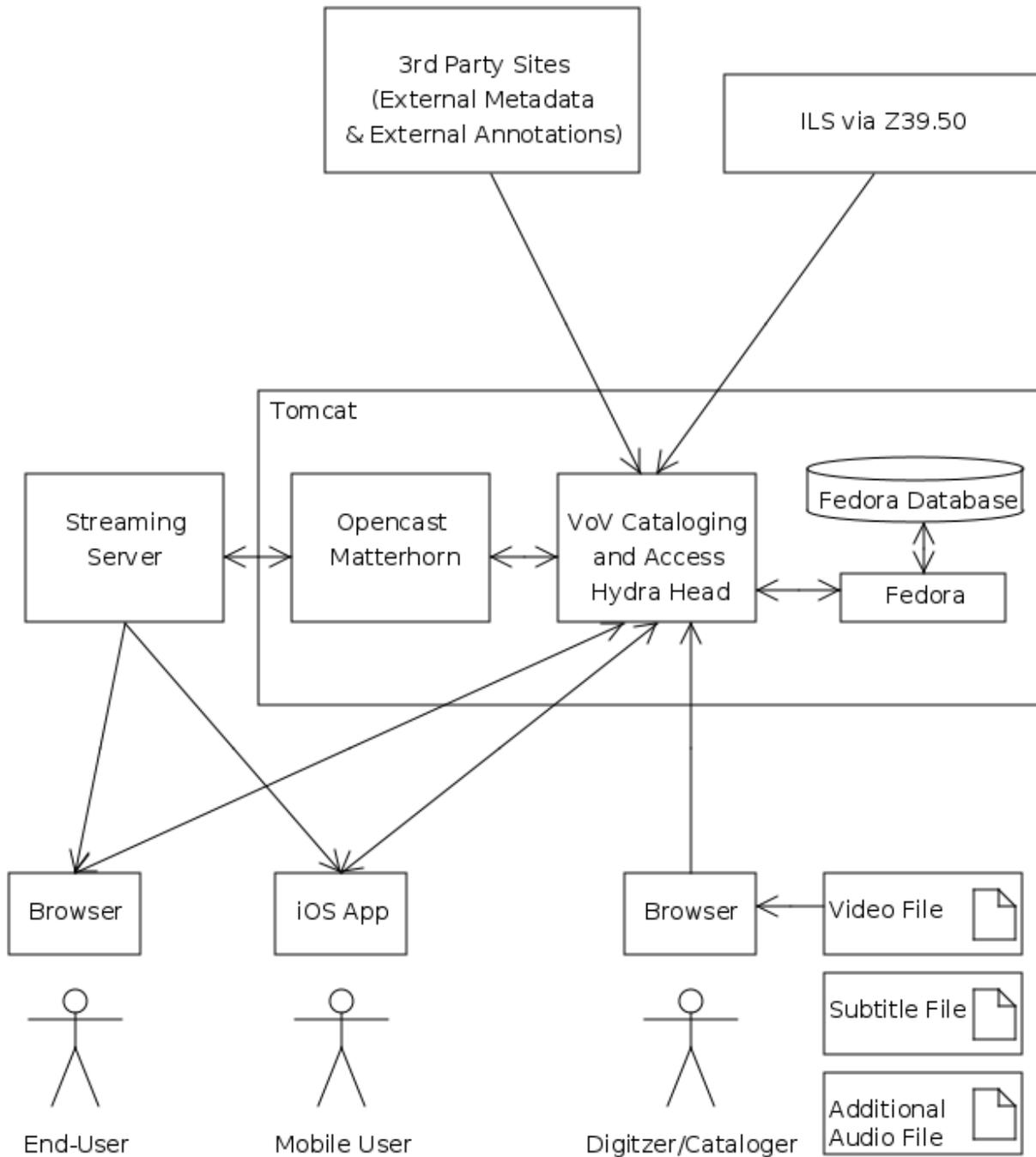


# Architecture (Option 4)

Another option would be to use the emerging Hydra framework. The picture below shows what a Hydra head architecture for Variations on Video might look like.



## Assumptions

- We don't want to put video into the Java client; new user interface work should be web or mobile-app based.
- The Hydra head would also include custom web services and the user profile database.

## Components

## Opencast Matterhorn

Matterhorn would be used for transcoding and potentially additional analysis or transformation necessary to prepare a video for delivery. Matterhorn would probably need to be configured to run externally or on a different machine in a production setting.

See [Technical Investigation - Opencast Matterhorn](#) for some more discussion.

## Streaming Server

The streaming server would serve the derivative files stored in the Video Data Store. This piece will probably be the hardest to choose a solution for. Maybe different systems could be used by different implementors? See [Streaming Servers](#) and [Video Formats](#) for some options.

## Fedora

An existing Fedora instance would be used to store the video metadata and potentially provide other available actions on the video content. See [Technical Investigation - Fedora Commons](#) for some more discussion.

## VoV Cataloging and Access Hydra Head

This Hydra head would accept video input (optionally with additional subtitle, audio, and structural input) for ingest. The video and audio content would be passed along to the transcoder process and metadata would be stored in Fedora. Additionally, some level of cataloging could be done for the newly ingested item, potentially with a review workflow. The access tools would be a part of the Hydra / Blacklight search catalog that runs under Ruby on Rails. Ingest could be passed off to Fedora to handle. The user profile database would be embedded into this component or could be split out to an external database. This component would require the most new work (designing and implementing).

## Z39.50

Z39.50 could be used to pull in bibliographic data and associate an asset to a record in an institution's OPAC. New work would need to be done to figure out the best way of extracting the necessary data from a MARC record for a video.

## 3rd Party Sites

3rd party sites could be used for additional metadata or for annotations either during the ingest/cataloging process or during an end-user's session with the web interface.

## iOS App

An iOS application could be developed to reach iPhones and iPads. A browser-based option could potentially replace this option.

## Open Questions

1. How tightly integrated should Matterhorn be? Should we allow switching it out for another media processing pipeline like Kaltura or a local solution?
2. Do we need a layer of abstraction between the Hydra head and Matterhorn?
3. Is the Hydra head the best place to put additional web services needed for the player and iOS app?
  - It might be easiest to put these additional web services and the Matterhorn/media processing pipeline abstraction layer into a separate webapp.
4. Does it reduce system complexity or increase it by running Matterhorn and Hydra under Tomcat?
5. Matterhorn creates Dublin Core and MPEG-7 metadata that could be distributed to Fedora. Do we want to use MPEG-7 or METS for structural metadata?

## Use Cases

When a digitizer wants to add a container to their collection, they will first prepare the video (or audio) file(s) and any ancillary materials associated with the container. Using their browser, they will log into the VoV Hydra head and use its ingest forms to add the files and associated descriptive, structural, and technical metadata as necessary. The metadata will be stored in a Fedora item and the video (or audio) files will be sent off to Matterhorn for encoding and analysis via its workflow service. When Matterhorn finishes its workflow, it will distribute the derivative files (flavors) to the streaming server(s) and to Fedora along with any metadata derived from analysis. Optionally, descriptive metadata can be sent to the institution's discovery system. After ingest, adding metadata and editing existing metadata can be performed in the Hydra head. If a video (or audio) file is corrupt or low quality, the original file can be uploaded again or it can be sent through Matterhorn's encoding and analysis services again.

When a user wants to find a container, they would use the institution's discovery system if available and Hydra's Blacklight discovery system otherwise. Once found and passing authorization checks, the user can play the container and create bookmarks that are stored on the server. Additionally, video and audio from VoV can be made available to other tools via an HTML embeddable player (see [unAPI and oEmbed](#)). The embeddable player will have a way for the user to authenticate and will perform the necessary authorization checks.