

Archive Import Wizard Implementation Plan

1. Changes to Folder Structure

Update the import-controller.js to use the following structure:

```
{archiveName}/  
  {conversation_uuid}_{ISO8601date}_{title}/  
    conversation.json  # Full original conversation JSON  
    messages/  
      {uuid}/  
        message.json  # Full original message JSON  
    media/  
      # All media files from the conversation
```

2. Media File Handling

Implement a new robust media handling system that:

1. Identifies and processes three types of media:
 - Traditional ChatGPT files (dalle-generations folder, file-*)
 - User uploaded files (file-*)
 - New format files (file_*.dat) with proper type detection
2. Creates a unified media folder in each conversation directory
3. Updates references in the conversation JSON to point to the correct media files

3. File Detection and Processing

Add logic to detect file types by:

1. Reading magic bytes from binary files to determine actual file type
2. Processing files with appropriate extensions
3. Handling special cases like audio files that need specific players

4. Update API Endpoints

Modify server endpoints to:

1. Serve full JSON data instead of markdown
2. Properly serve media files with correct MIME types

3. Support advanced filtering by new metadata elements

5. Frontend Updates

Update the frontend to:

1. Display rich content from the JSON structure
2. Render different message types appropriately
3. Support media playback for different file types

Implementation Steps

Step 1: Modify the Import Controller

Update `import-controller.js` to:

- Use UUIDs for conversation and message folder names
- Store original JSON instead of converting to markdown
- Add ISO8601 date formatting for folder names

Step 2: Enhance Media File Processing

Add new functions to:

- Detect and extract file types from binary data
- Process each media file type appropriately
- Maintain proper file extensions and MIME types

Step 3: Update Schema and APIs

Modify the API schema to:

- Return full JSON data
- Support pagination of complex conversation structures
- Provide proper MIME types for media files

Step 4: Update Frontend Components

Enhance frontend components to:

- Display rich JSON content
- Support rendering of various media types
- Handle complex conversation flows with tools and custom GPTs

