1 Interview transfer service

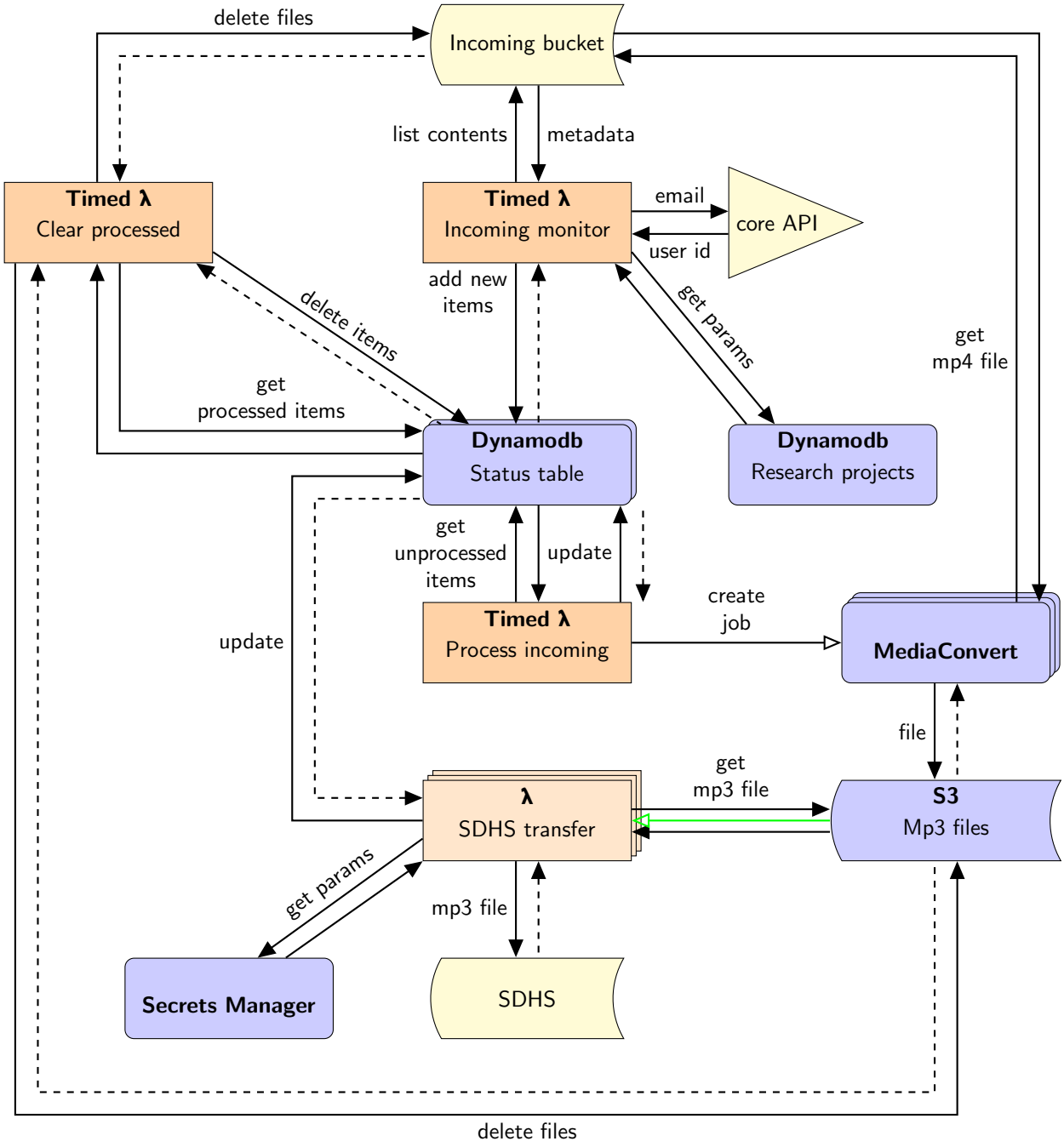

Figure 1: Thiscovery interview transfer service architecture.

1.1 General

If several files are added to Incoming bucket at once, they will be processed simultaneously by MediaConvert, which will then trigger simultaneous connections and transfer attempts to the SFTP servers. If this proves problematic, adjust Process incoming to only create one or a few MediaConvert jobs per timed invocation.

1.2 Specific resources

1.2.1 Incoming bucket

This bucket is not part of the stack. The stack also includes a mock incoming bucket (not represented), which is part of the stack.

1.2.2 Incoming monitor function

Calculates and stores in Dynamodb the name that will be used to rename each incoming file at the SFTP server. At present, metadata provided by MyInterview does not contain any project identifier, so the name of the file can only reflect the

relationship with the thiscovery user.

Raises error if:

1. core API does not return user id

Status table

The stack also includes a second table with the same data fields as this one (not represented), which will keep historical transactions for audit purposes. That is, old items will be deleted from the Status table, but a copy of those old items will be stored in this second table.

Each item in Status table relates to a file uploaded to the Incoming bucket by MyInterview. Files created by function Extract audio will not be entered as a separate item in this table.

Each item should include:

1. **processing status**: new, audio extracted, processed
2. **extraction attempts**: a counter to keep track of failed attempts to run Extract audio on this item
3. **transfer attempts**: a counter to keep track of failed attempts to transfer this item to SDHS
4. **target basename**: calculated basename to use when renaming the original files

2 Participant info transfer service

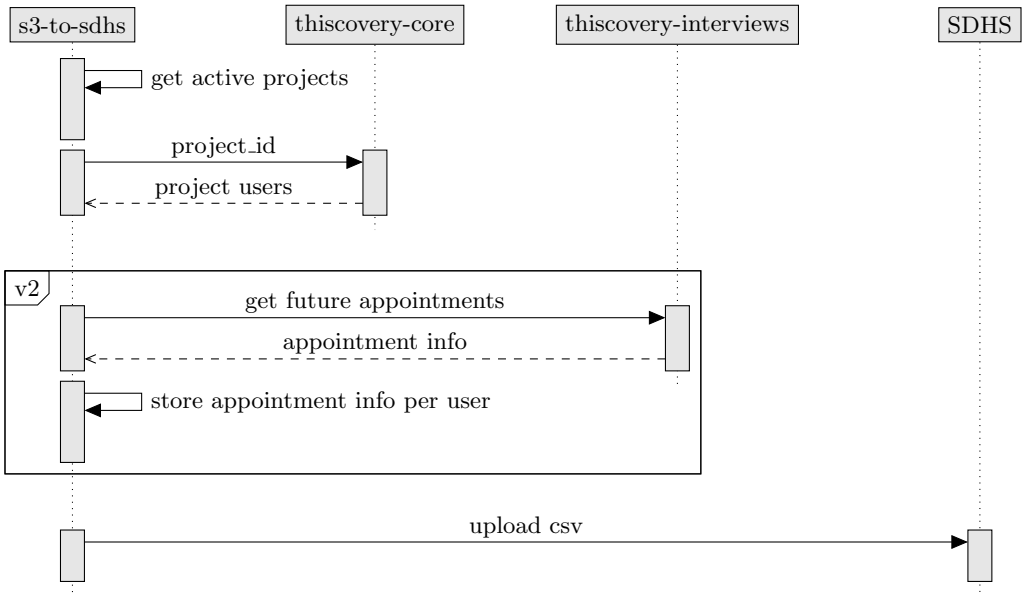

Figure 2: Sequence diagram of the thiscovery participant info transfer service.

2.1 General

Version 2 of the participant info transfer service output should include details of the scheduled interview:

- interview datetime
- appointment type (usually a proxy for participant group)
- Acuity calendar name (proxy for interviewer first name)

Version 2 will require:

1. a new API endpoint in thiscovery-interviews to retrieve future appoitments
2. a new Dynamodb table in s3-to-sdhs to store appointment info indexed by project_id (hash key), participant id (sort key)