European Commission

EUROPEAN COMMISSION

Directorate-General for Interpretation

Technical Overview

Technical Architecture Document

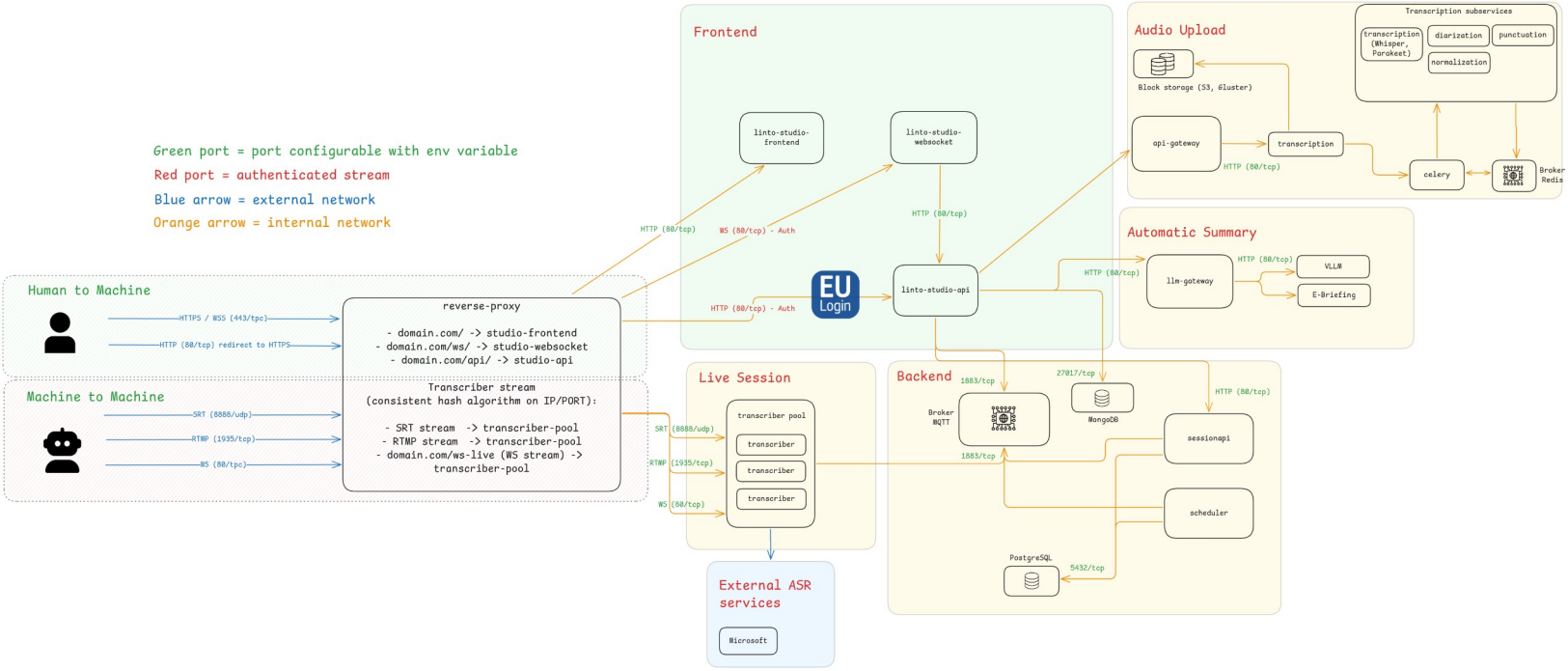
Table of Contents

Introduction to E-Meeting

This document aims to present the various components of the solution and the flows between these components. A user guide is also available in the E-Meeting Toolkit to assist with the onboarding of the solution.

Flow diagram

Diagram



Service Description

reverse-proxy

The reverse proxy receives two types of flows, as we can see on the diagram. Flows from humans interacting via a web browser, and Machine-to-Machine flows, which are audio streams transmitted to the transcribers.

The reverse proxy contains several endpoints (mydomain.com is used as an example):

- / : the root transfers the flows to the LinTO web interface (linto-studio-frontend)
- /ws: transfers the flows to the LinTO websocket service (linto-studio-websocket)
- /api: transfers the flows to the LinTO API service (linto-studio-api)

The endpoints can be adapted as needed. This is just an example.

Additionally, the reverse proxy is responsible for redirecting audio streams via the SRT, RTMP, or Websocket protocols. These streams are load balanced between the existing transcribers.

However, note that the flows must be redirected to the transcribers using a consistent hash algorithm based on the IP/PORT pair.

linto-studio-frontend

The linto-studio-frontend service is a statically compiled web interface. Therefore, it is a simple HTML/JS/CSS content server.

linto-studio-websocket

The linto-studio-websocket service manages collaborative editing when multiple users are editing a conversation simultaneously, ensuring conflict resolution.

linto-studio-api

The linto-studio-api service exposes the main REST API, serving as the core platform of the system. It manages resources, users, and authentication, and also acts as an orchestrator for external services such as transcription and automatic summarization. Additionally, it handles authentication through EU-Login. Beyond these responsibilities, it also provides a proxy to the sessionapi service to manage transcription sessions and channels.

transcriber

The transcriber service is the key component responsible for executing live transcriptions. It fulfills three main functions:

- It establishes three mount points to receive streams via SRT, RTMP, or WebSocket.
- Upon receiving a stream, it processes and forwards it to the ASR (Automatic Speech Recognition) system.

- Once the ASR provides the transcription, the transcriber relays it to the broker, making it accessible to other components within the system.

The transcriber is designed to operate within a pool. Each transcriber locally exposes three ports and can handle as many sessions as the underlying system allows. This setup enables an elasticity policy based on monitoring the transcriber's resource usage. For instance, if CPU usage nears 100%, a new transcriber can be launched. Conversely, if a transcriber is no longer in use, it can be terminated.

It is then the responsibility of the load balancer (reverse proxy) to intelligently route streams, ensuring an even distribution of the workload.

sessionapi

The sessionapi service manages the creation of sessions and their channels, as well as transcriber profiles. It also exposes a REST API for managing sessions.

scheduler

The scheduler service is responsible for keeping the live transcription system up to date. It records the emitted final transcriptions in the database and ensures that the state of the transcribers remains consistent in the database.

api-gateway

The api-gateway service is responsible for registering and maintaining a directory of services that communicate with it. It also functions as a proxy, routing requests to the appropriate endpoints of the registered services.

It is primarily used for offline transcription and will support additional services such as NLP (Natural Language Processing) in the future.

transcription

The transcription service is responsible for initiating jobs to fulfill initial requests and for maintaining the status of those jobs. It acts as an orchestrator, ensuring the smooth execution and management of transcription tasks.

It communicates with the transcription subservice through Celery tasks, enabling distributed task management and coordination.

llm-gateway

The llm-gateway service is similar in principle to the api-gateway, but in addition to proxying requests to its services, it also manages the various types of available prompts.

It is used for generating automatic summaries.

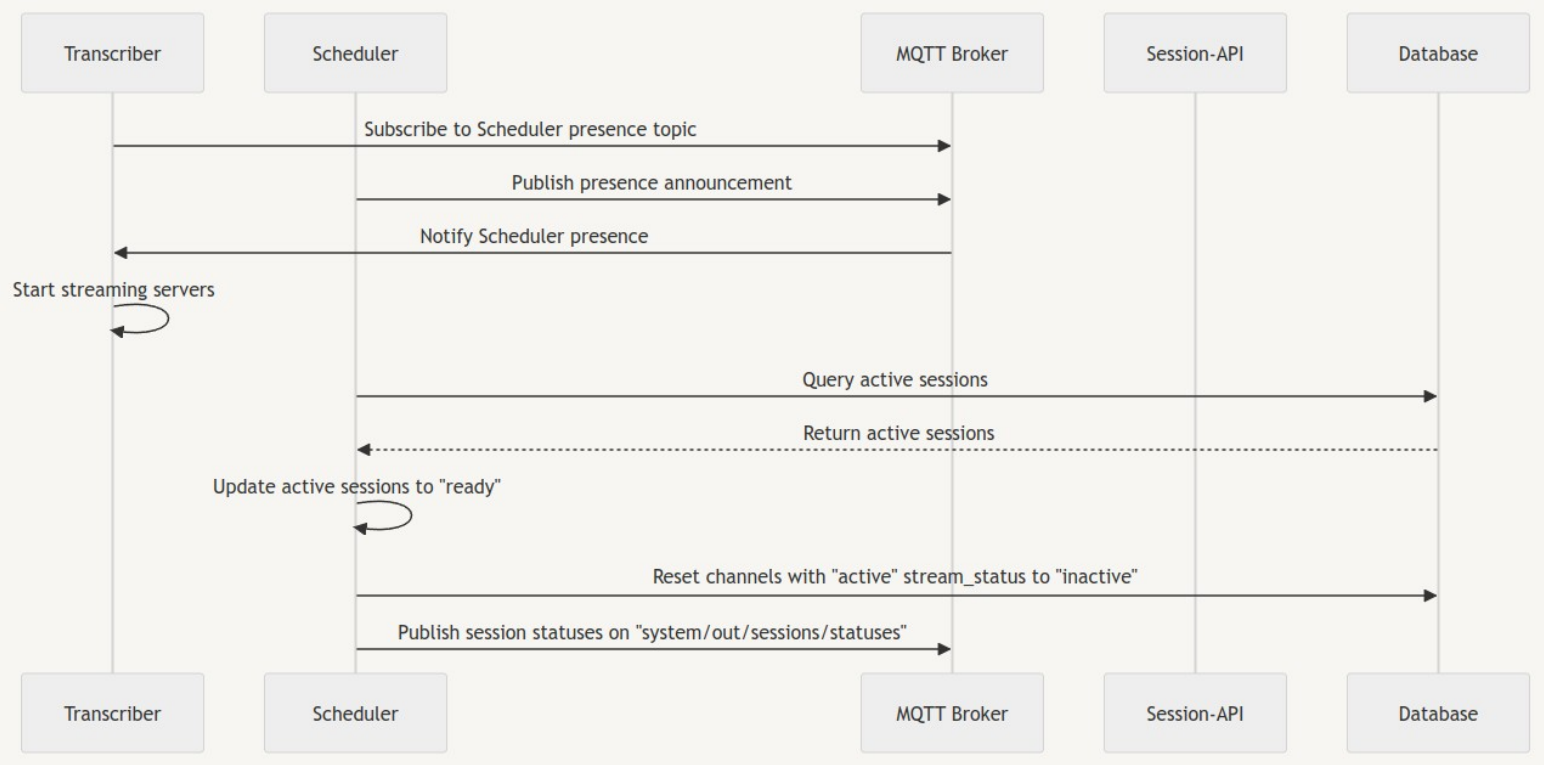
vllm / E-Briefing

These services enable the generation of automatic summaries. The E-Briefing service is a service provided by the European Commission.

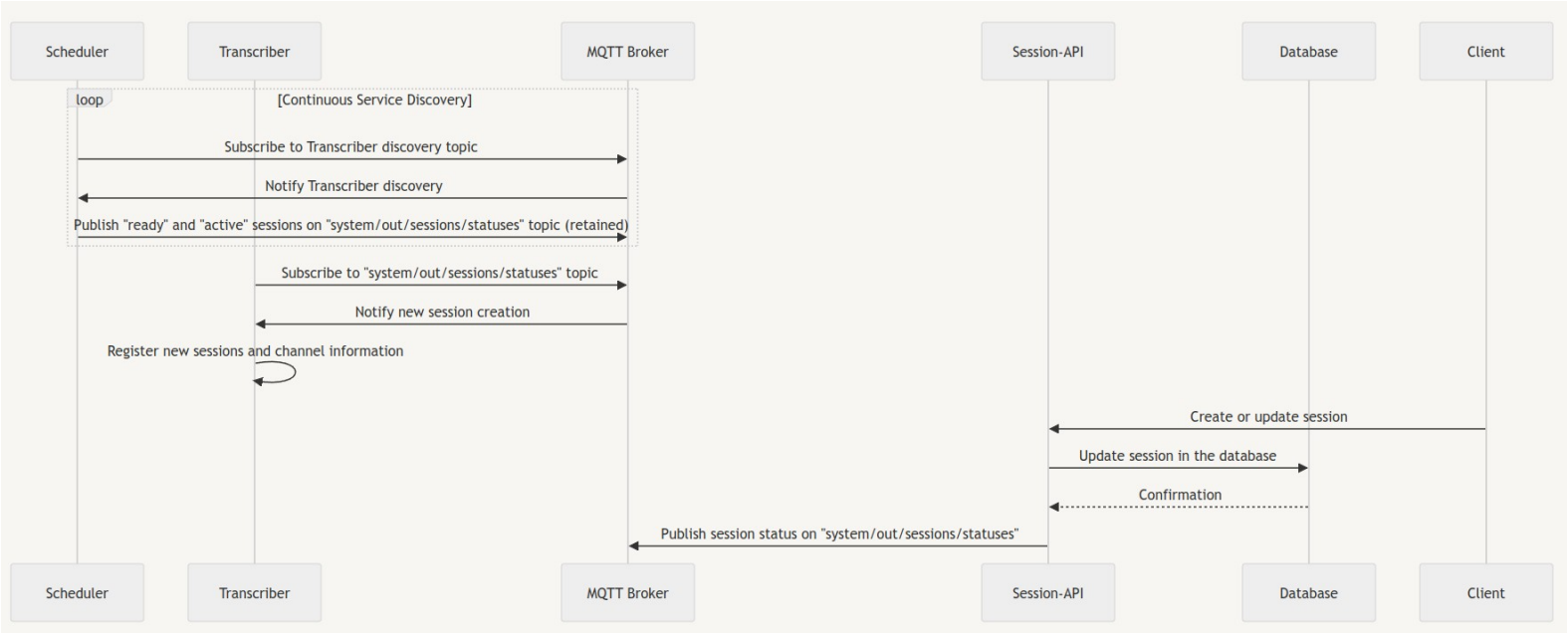
Internal workings of the transcriber

Below is a macro view of the flow sequences with the multiplexed transcriber to provide a clear understanding of how it operates.

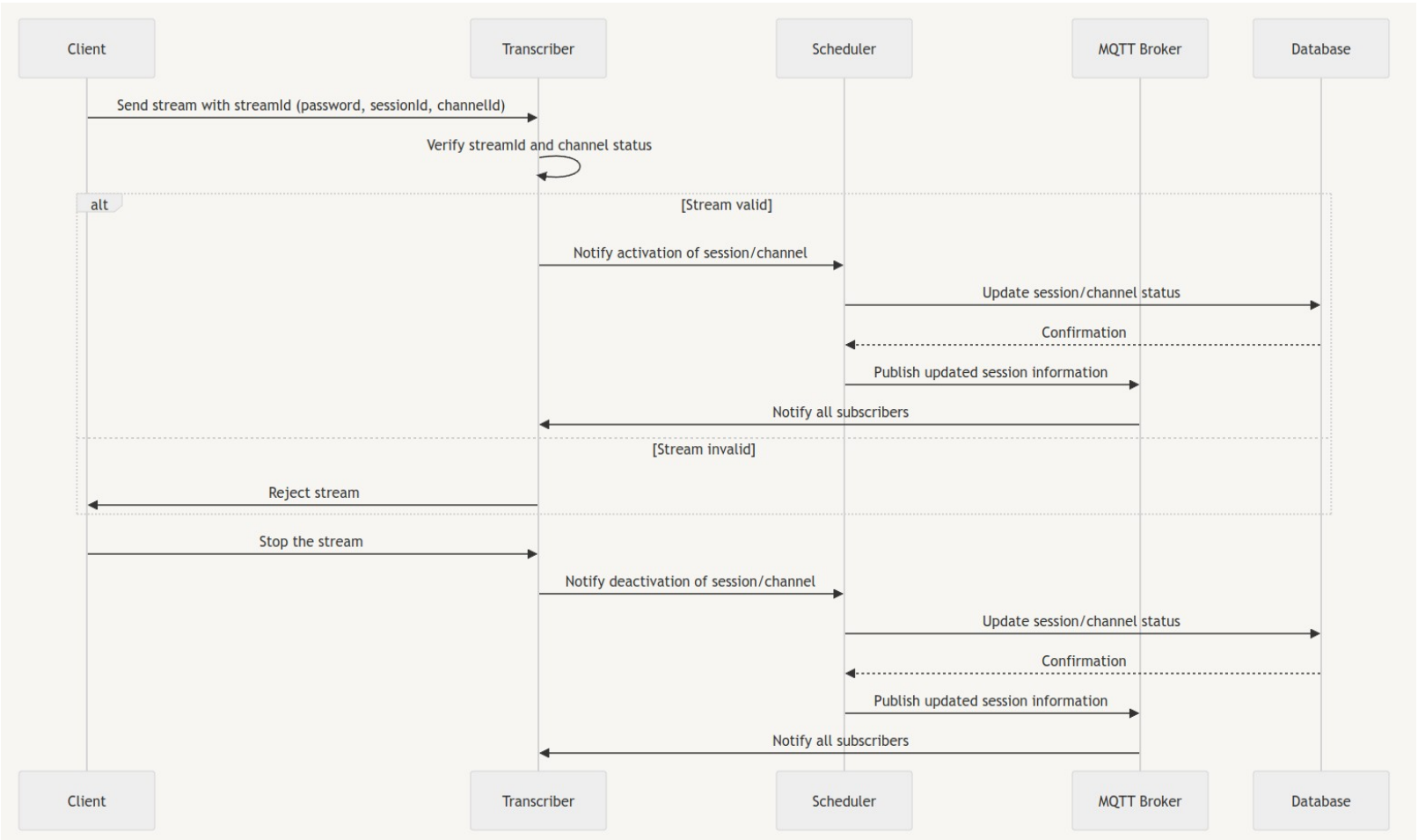
System Boot


Transcriber
Scheduler
MQTT Broker
Session-API
Database
Subscribe to Scheduler presence topic
Publish presence announcement
Notify Scheduler presence
Start streaming servers
Query active sessions
Return active sessions
Update active sessions to "ready"
Reset channels with "active" stream_status to "inactive"
Publish session statuses on "system/out/sessions/statuses"
Transcriber
Scheduler
MQTT Broker
Session-API
Database

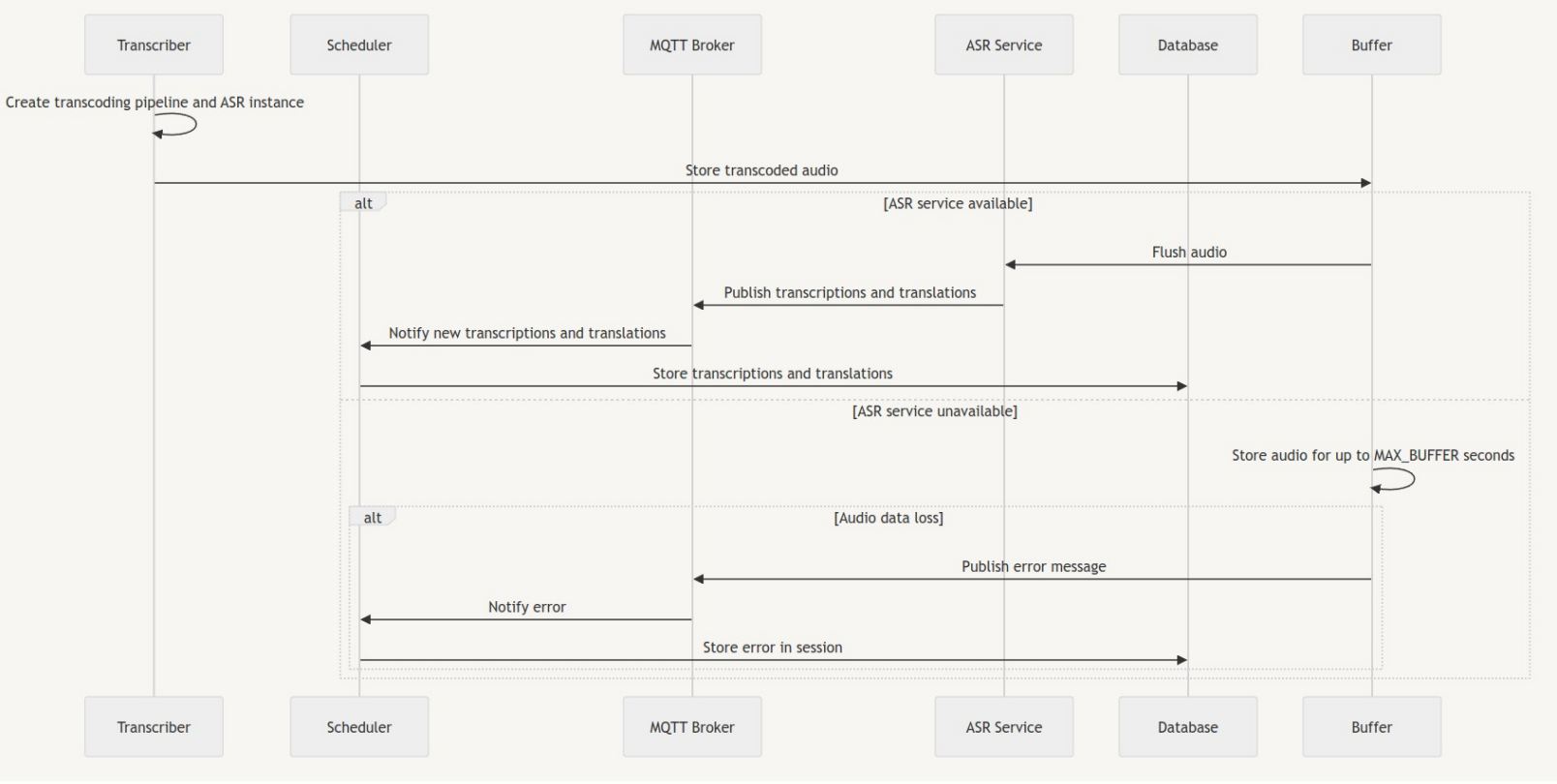
System status management



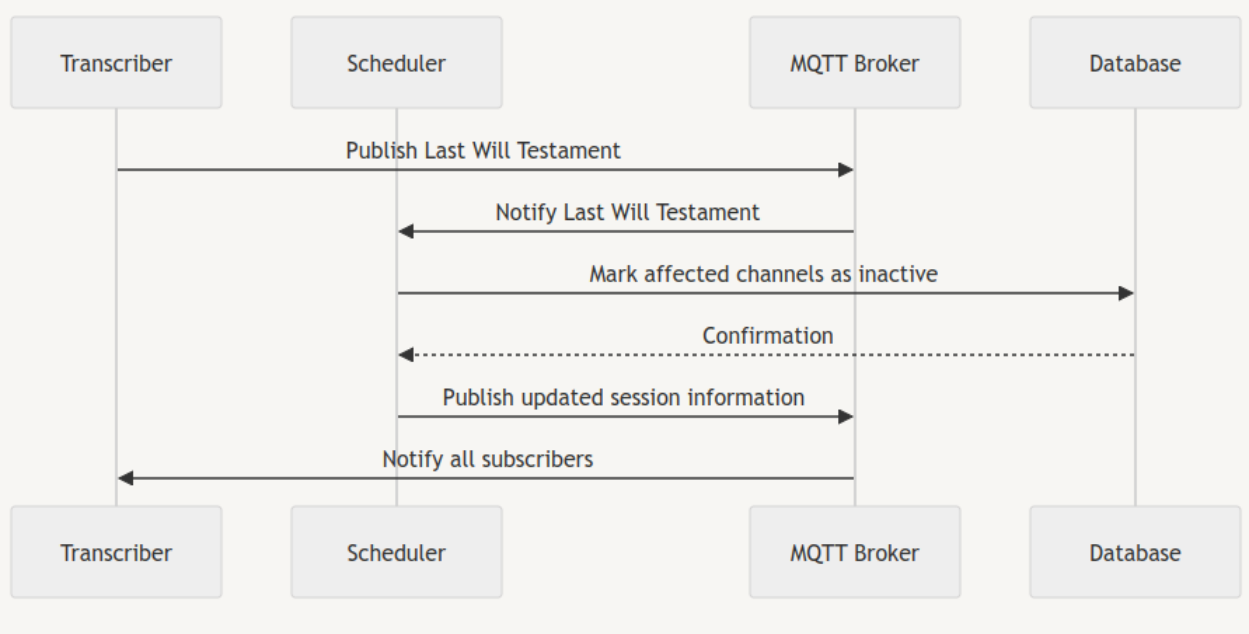
Stream received



Transcription



Transcriber crash



New transcriber spins up after a crash