# Data Dictionary — Technology Overview

## Data Dictionary & Technology Overview (Updated)

### Chrome Extension Concepts

#### Manifest V3 (MV3)

The current standard for Chrome extensions. Defines extension metadata, permissions, background service workers, side panels, and offscreen documents.

#### Side Panel UI

A persistent panel attached to the active tab. In this project, it shows live captions in the **desired language** and provides Stop/Start controls.

#### Background Service Worker

A lightweight script that manages lifecycle events. It opens the side panel, tracks which tab is targeted, and ensures the offscreen document exists.

#### Offscreen Document

A hidden HTML page that can use full Web APIs (DOM, WebAudio, etc.). Hosts the **Azure Speech SDK** to stream audio, receive transcriptions, and push results back to the side panel.

#### tabCapture API

Chrome API that captures audio from a tab. Produces a `MediaStream` that can be piped into the Azure Speech SDK's audio input.

---

## Speech & Translation Concepts

#### ASR (Automatic Speech Recognition)

Converts speech audio in a **foreign language** into text in that same language.

**MT (Machine Translation)**

Converts recognized text into a **desired target language**.

**TTS (Text-to-Speech)**

Generates spoken audio output in the desired language. Useful for voice playback in future versions.

**Transcript Chunk**

A partial piece of foreign-language text produced by the SDK during streaming.

**Translation Chunk**

A translated piece of text in the desired language, returned alongside or after the transcript chunk.

**Caption Event**

A structured message from the offscreen document to the side panel containing new transcript/translation content.

---

# Azure Speech SDK Concepts

**Azure Speech SDK**

Microsoft's client library (JavaScript in this project) for real-time speech services. Supports:

- **Speech to Text (ASR)**

- **Speech Translation (ASR + MT)**

- **Speech Synthesis (TTS)**

**AudioConfig**

SDK configuration object that defines audio input. In this project, it uses the **tabCapture MediaStream** as input.

**SpeechConfig**

SDK configuration object holding subscription key, region, and translation settings (e.g., source language, target languages).

**TranslationRecognizer**

SDK class that streams audio to Azure and returns recognized text (foreign language) plus translated text (desired language).

**ResultReason**

An enum provided by the SDK indicating why a result was returned (e.g., `RecognizedSpeech` , `RecognizedTranslation` , `NoMatch` ).

**Synthesizer (SpeechSynthesizer)**

SDK class that turns text into audio (TTS). Can be used for translated voice playback in the extension.

**Continuous Recognition**

Streaming mode where audio is sent in real time and partial results (chunks) are returned until stopped.

## Azure Speech Services Concepts

**Azure Speech Service**

Cloud service that provides ASR, Translation, and TTS. The SDK connects to it via a websocket streaming protocol.

**Speech to Text (STT)**

Service endpoint that receives audio and returns transcription in the same language.

**Speech Translation**

Service endpoint that combines STT + MT. Produces both the original transcription and translations in the desired language(s).

**Text to Speech (Neural TTS)**

Service endpoint that generates human-like speech audio from text.

**Region / Endpoint**

Azure services require specifying the **region** (e.g., `eastus` ) where your subscription is hosted. The SDK automatically uses the correct endpoint when given region + key.

**Subscription Key**

API key issued in Azure Portal, needed to authenticate SDK requests. Stored securely in extension (ideally fetched from a backend, not hardcoded).

## Assumptions

- **Foreign → Desired language** is the core flow, not hardcoded to Spanish/English.

- **Azure Speech SDK (JS)** is embedded in the offscreen document.

- **Azure Speech Translation** is the chosen cloud provider for MVP.

- Extension permissions: `tabCapture` , `offscreen` , `storage` , `tabs` .

- SDK requires valid Azure **Subscription Key** + **Region**.