Ήχος στο Διαδίκτυο

Web Audio API

Χρυσούλα Αλεξανδράκη
Τμήμα Μουσικής Τεχνολογίας και Ακουστικής
Ελληνικό Μεσογειακό Πανεπιστήμιο

Previous

- HTML DOM API (e.g. HTMLAudioElement)
  - Can do a lot
- Web Audio API
  - Can do much more, e.g.
  - Chrome Music Lab
    - **How were these built?**
    - All our experiments are all built with freely accessible web technology such as Web Audio API, WebMIDI, Tone.js, and more. These tools make it easier for coders to build new interactive music experiences. You can get the open-source code to lots of these experiments here on Github.

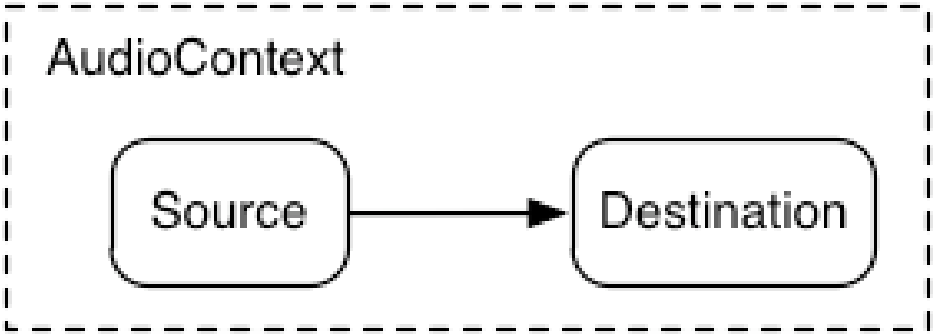
Web Audio API: Cool Audio Stuff

- Audio Signal Processing
- Synthesizing Audio
- Mixing Processing, Filtering
- Audio Visualization
- Game Audio and 3D Audio
- etc.

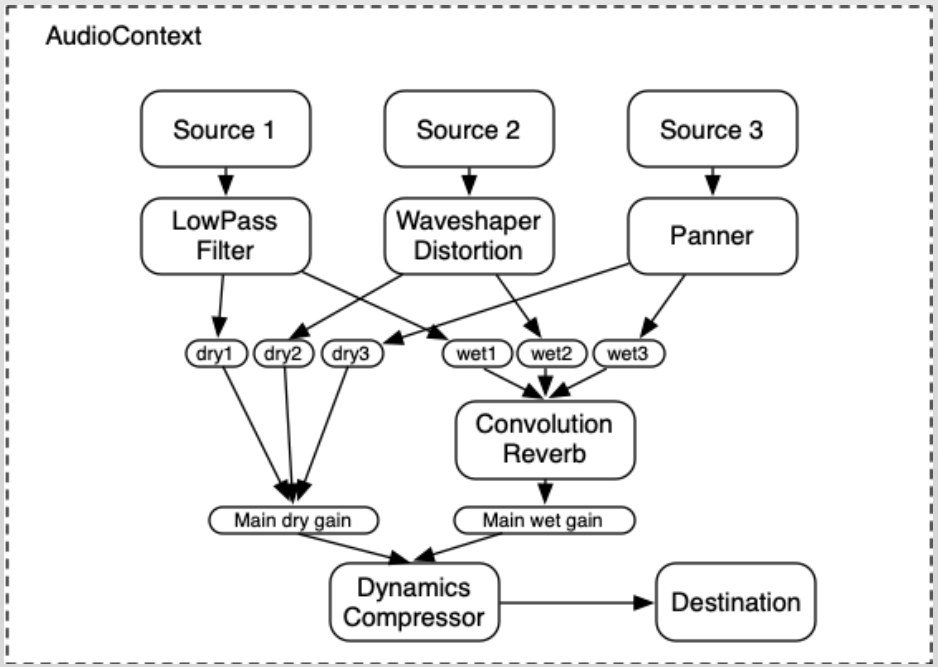
Target Web Audio Use Cases

1. Video Chat Application
2. 3D game with music and convincing sound effects
3. Online music production tool
4. Online radio broadcast
5. Music Creation Environment with Sampled Instruments
6. Connected DJ booth
7. Playful sonification of user interfaces
8. Podcast on a flight
9. Short film with director's commentary and audio description
10. Web-based guitar practice service
11. User Control of Audio

Modular Routing



- Modular routing allows arbitrary connections between different AudioNode objects.
- Each node can have *inputs* and/or *outputs*.
  - A *source node* has no inputs and a single output.
  - A *destination node* has one input and no outputs.
  - Other nodes such as filters can be placed between the source and destination nodes.
- AudioNodes are connected to form simple or complex audio contexts



The way it works

- Σύνολο μεθόδων και κλάσεων με τις οποίες μπορούμε να διαχειριστούμε ήχο μέσα από τον Internet browser
  - https://developer.mozilla.org/en-US/docs/Web/API/Web_Audio_API

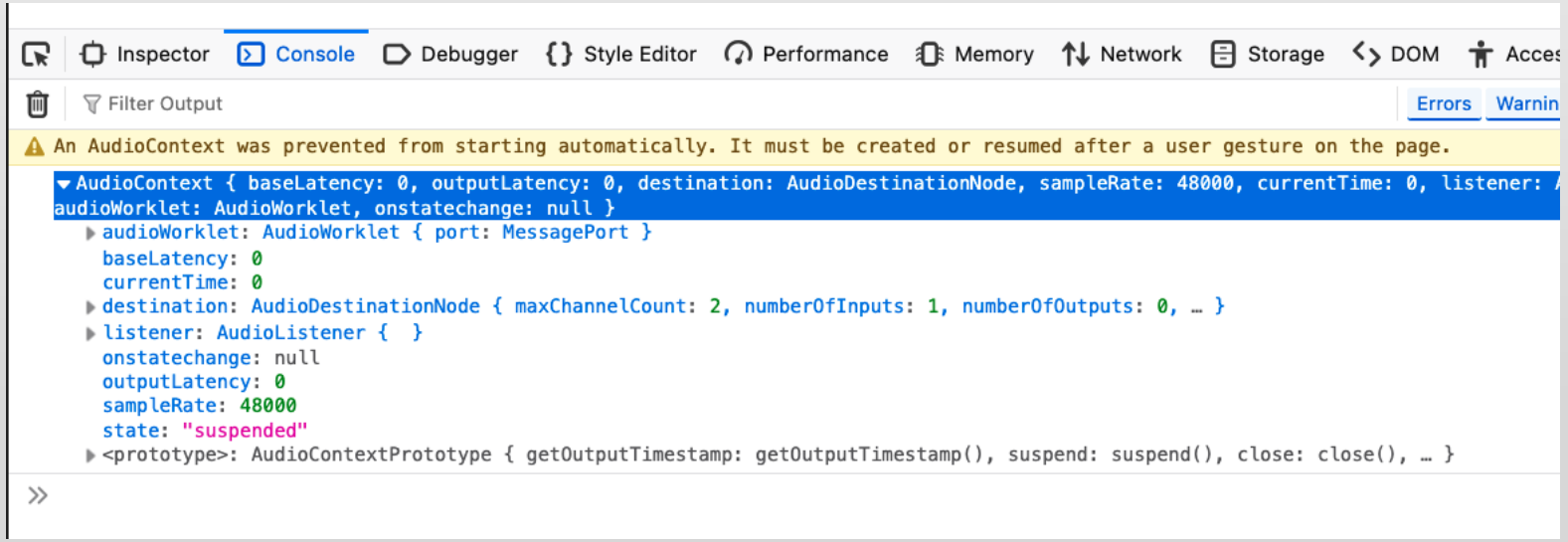
AudioContext

- **AudioContext** is a **JavaScript interface**
  - It represents the **audio-processing graph**.
  - All audio nodes (GainNode, OscillatorNode, BiquadFilterNode, etc.) are created **through an AudioContext**.
  - It controls **timing, audio routing, and processing**.

AudioContext

- Normally one instance per document

```
let ctx = new AudioContext();
console.log(ctx);
```



Example

- If run as a file, the following code will result in a CORS error

```
let ctx = new AudioContext();
const audio = new Audio('../audio/disco0.mp3');
const source = ctx.createMediaElementSource(audio);
source.connect(ctx.destination);
```

- Για λόγους ασφαλείας το **Web Audio API** (η createMediaElementSource()) δεν επιτρέπει επεξεργασία και δίνει το μήνυμα:
  - *"The HTMLMediaElement passed to createMediaElementSource has a cross-origin resource, the node will output silence."*
- Οι browsers εφαρμόζουν **CORS policy**.
  - Τα αρχεία που φορτώνονται από file:// δεν έχουν HTTP headers, οπότε θεωρούνται μη ασφαλή για επεξεργασία από το Web Audio API.

Cross Origin Resource Sharing - CORS

- It is a browser security mechanism that controls whether a webpage is allowed to access resources from another origin (domain).
  - If it didn't exist
- An **origin** is the combination of:
  - **protocol** (http / https), **domain**, **port** e.g. https://example.com:8080
- CORS does not apply to the files I was loading up to now..
  - Rule of thumb: It applies when **JavaScript tries to access or analyse the contents of a resource from another origin**, not just load it.

What to do

- Run a local web server, examples:
  1. Using python:
     - python -m http.server 8000
  2. Using the LiveServer module of VS Code
     - Install, Then right click on index.html
  3. Using Node.js
     - npm install http-server
     - npx http-server
- In options (2), (3), beware of the ***Document Root*** of the server:
  - By default, it is the working directory of the terminal session when you run the command to start the server

Adding an AudioNode between source-dest

- E.g. a gain node

```
const source = ctx.createMediaElementSource(audio);
const gainNode = ctx.createGain();
source.connect(gainNode).connect(ctx.destination);
```

- How do you control the gain node using a volume slider?

What else can I insert between source-dest?

- You could try one of the following:
  - BiquadFilterNode
  - StereoPannerNode
  - DelayNode
  - … and many more

Next

- Playlist
- Mixing
- Synthesizing
  - Noise, osclator
- Visualization
  - Display waveform
- Mic Signal and getUserMedia()