

Neon.js | Editing 0400_corr...
ddmal.music.mcgill.ca/neondemo/editor/0400_corr.mei

gburlet (Greg Burlet) MSc Thesis TD Canada Trust Gregory Burlet | McGill MW
Other Bookmarks
Neon.js File - Developers | DOMAL

Editing 0400_corr.mei

BACKGROUND
Image Opacity:

MODE

EDIT

Information
Selected: Clivis
Pitch(es): A3, G3

276 Sunday at Compline.

From 1st Vespers of the Feast of the Blessed Trinity to None on Saturday before the 1st Sunday of Advent.

Ant.
1.
S Al-ve, * Re-gí- na, máter mi-se-ricórdi- ae :

Ví- ta, dulcé- do, et spes nóstra, sál-ve. Ad te

clamá-mus, éxsu-les, fí-li- i Hévae. Ad te suspi-rá-

NEON.JS

An Application of JavaScript

OUTLINE

JavaScript Overview

variables, arrays, functions, objects

JavaScript Libraries

jQuery, audiolib.js, Fabric.js, Kinetic.js

Aesthetics

Twitter Bootstrap, jQuery UI

Neon.js

purpose, design, implementation

ABOUT ME

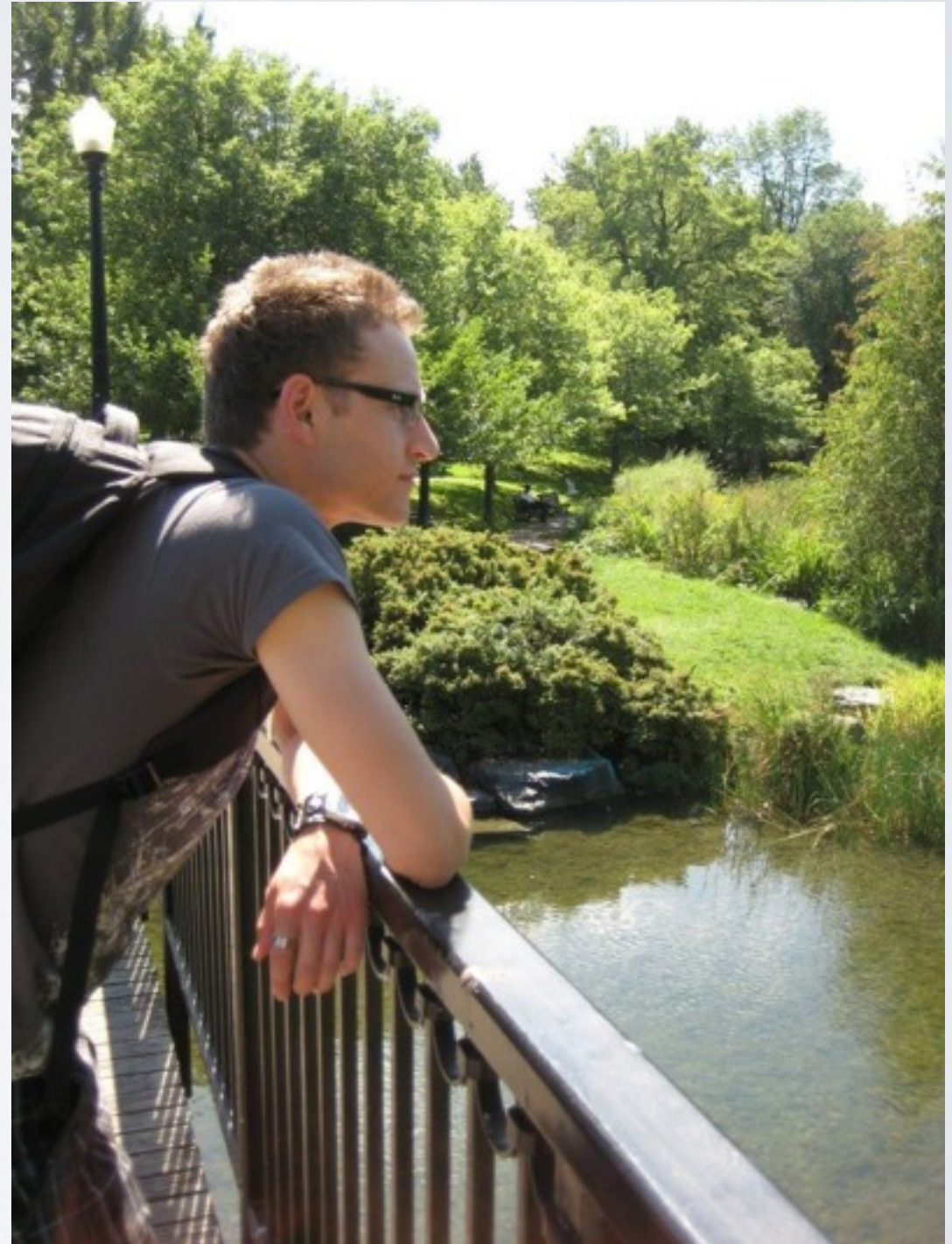
- B.Sc. Specialization in
Computing Science
University of Alberta
- M.A. Candidate in Music
Technology
McGill University

DDMAL

DISTRIBUTED DIGITAL MUSIC
ARCHIVES & LIBRARIES LAB



Centre for Interdisciplinary
Research in Music Media
and Technology



JAVASCRIPT

variables

- Symbol with a known or unknown value
- Loosely typed language (no need to specify data type)

Example

```
var x;                // declared but undefined
var y = 5.4;
var foo = "bar";
var consumedCoffee = false;
var greeting = "hello " + "world"
```

JAVASCRIPT

variables



Example

```
// note  
  
var type = "eighth";  
var pitchName = "C";  
var octave = 4;  
...
```

JAVASCRIPT

arrays

- Container that can hold multiple values
- Can contain different types of variables

Array Creation

```
var crazyCatLadyCats = new Array();  
  
crazyCatLadyCats[0] = "Sir Ferocious"  
crazyCatLadyCats[1] = "Mister Whiskerson"  
crazyCatLadyCats[2] = "Winklestiltskin"
```

JAVASCRIPT

arrays

- Refer to an element in an array by the index number
- Starts from 0!

Array Indexing

```
var cat1 = crazyCatLadyCats[0]; // Sir Ferocious  
var cat2 = crazyCatLadyCats[1]; // Mister Whiskerson  
var cat3 = crazyCatLadyCats[2]; // Wink1lestiltskin
```

JAVASCRIPT

arrays

Array Object Properties

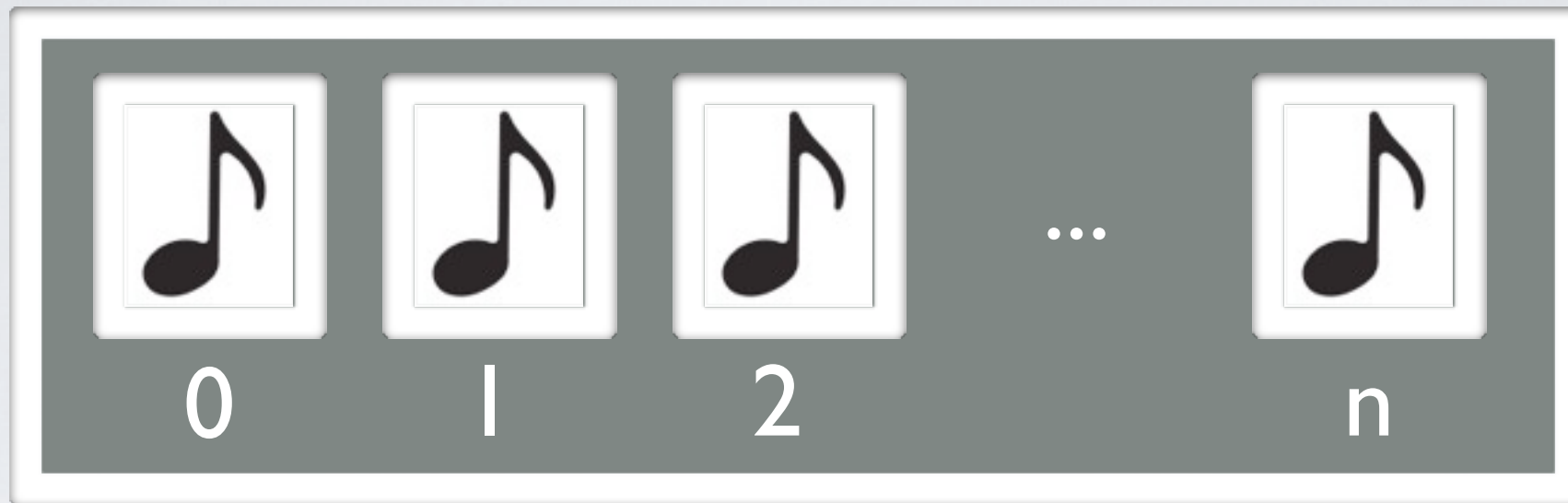
Property	Description
<u>constructor</u>	Returns the function that created the Array object's prototype
<u>length</u>	Sets or returns the number of elements in an array
<u>prototype</u>	Allows you to add properties and methods to an Array object

Array Object Methods

Method	Description
<u>concat()</u>	Joins two or more arrays, and returns a copy of the joined arrays
<u>indexOf()</u>	Search the array for an element and returns it's position
<u>join()</u>	Joins all elements of an array into a string
<u>lastIndexOf()</u>	Search the array for an element, starting at the end, and returns it's position
<u>pop()</u>	Removes the last element of an array, and returns that element
<u>push()</u>	Adds new elements to the end of an array, and returns the new length
<u>reverse()</u>	Reverses the order of the elements in an array
<u>shift()</u>	Removes the first element of an array, and returns that element
<u>slice()</u>	Selects a part of an array, and returns the new array
<u>sort()</u>	Sorts the elements of an array
<u>splice()</u>	Adds/Removes elements from an array
<u>toString()</u>	Converts an array to a string, and returns the result
<u>unshift()</u>	Adds new elements to the beginning of an array, and returns the new length
<u>valueOf()</u>	Returns the primitive value of an array

JAVASCRIPT

arrays



Code

```
// array of note types  
  
var n = 5;  
var notes = new Array(n);  
for (var i = 0; i < n; i++) {  
    notes[i] = "eighth";  
}
```

JAVASCRIPT

functions

- A set of statements that perform a specific task
- Arguments (inputs), return value (output)
- Executed when it is called

Example

```
function square(x) {  
    var theSquare = x*x;  
    return theSquare;  
}  
  
var x = 2;  
var xSquared = square(x);
```

JAVASCRIPT

functions

mtof {



MIDI 69

Example

```
// MIDI number to frequency function
```

```
function mtof(p) {  
    var freq = Math.pow(2, (p-69)/12) * 440;  
    return freq;  
}
```

```
var midiNumber = 69;           // A4  
var freq = mtof(midiNumber);   // 440 Hz
```

}

JAVASCRIPT

objects

- JavaScript is an interpreted and object-oriented language
 - An object can represent a physical object, idea, or entity
-

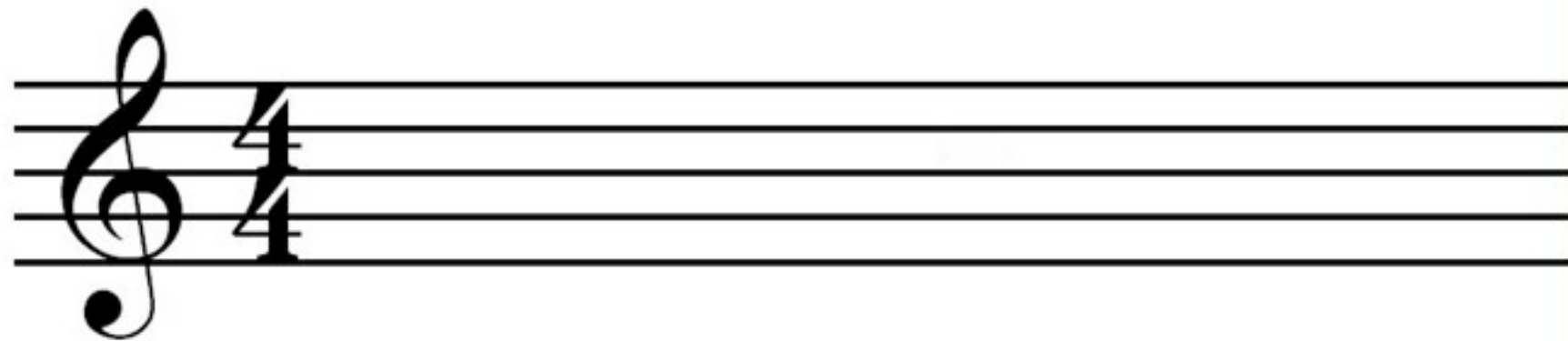
Objects can have:

- **Properties or attributes:** values associated with the object
- **Methods:** actions to perform on the object

JAVASCRIPT

objects

Example: Music Staff

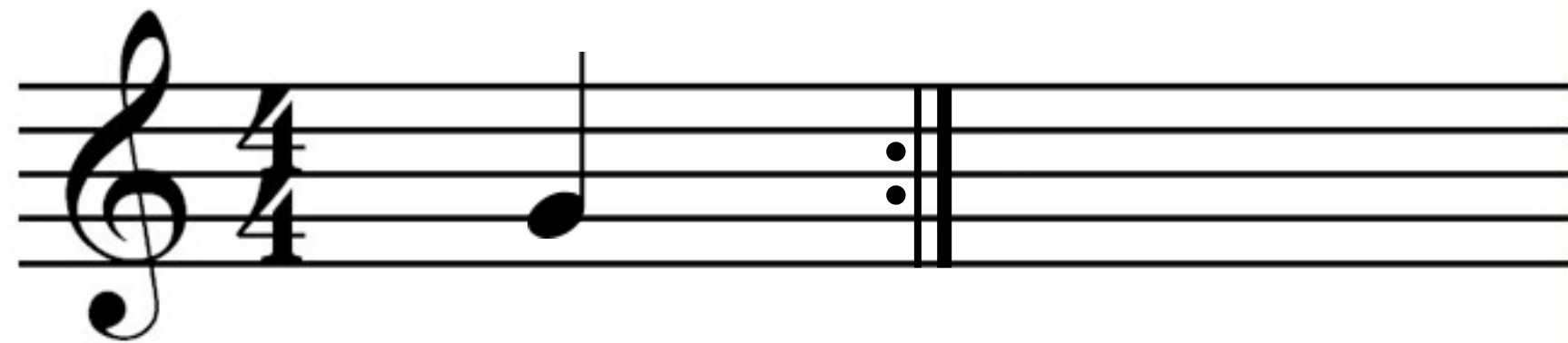


```
// create a staff object
var staff = new Object();
// add properties
staff.numLines = 5;
staff.clef = "treble";
staff.elements = new Array();
// add methods
staff.setNumLines = function(nLines) {
    this.numLines = nLines;
}
```

JAVASCRIPT

objects

Example: Music Staff - adding elements

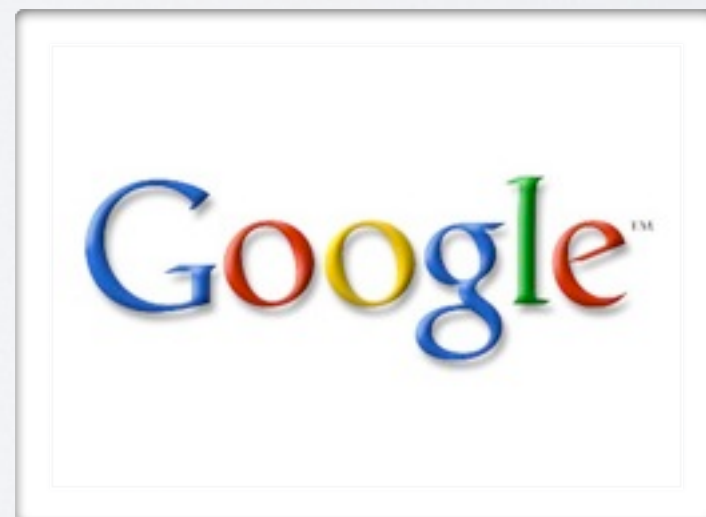


```
var note = new Object();  
note.pitch = "G";  
note.octave = 4;  
  
var barline = new Object();  
barline.type = "repeat";  
  
staff.elements.push(note);  
staff.elements.push(barline);
```

PRACTICE

The best way to learn a language is practice.

and



LIBRARIES



the write less, do more, JavaScript library

LIBRARIES

jQuery

GETTING STARTED
[Main Page](#)
[Downloading jQuery](#)
[How jQuery Works](#)
[FAQ](#)
[Tutorials](#)
[Using jQuery with Other Libraries](#)
[Variable Types](#)

Tutorials:How jQuery Works

(Redirected from How jQuery Works)

A basic introduction to jQuery and the concepts that you need to know to use it.

Original: http://docs.jquery.com/Tutorials:How_jQuery_Works 

Author: [John Resig.](#) 

Similar Tutorials: [jQuery Core](#), [Selectors](#), [CSS](#), [Traversing](#), [Manipulation](#), [Events](#), [Effects](#)

HTML document traversal & manipulation

Animation

Event handling

AJAX interactions

.addClass()

.slideUp()

.click()

.ajax()

LIBRARIES

jQuery



- Send data to, and retrieve data from, a server asynchronously without reloading the web page.
- jQuery provides easy functions to make GET and POST requests.
- For example, querying server to retrieve artist information

LIBRARIES

audiolib.js

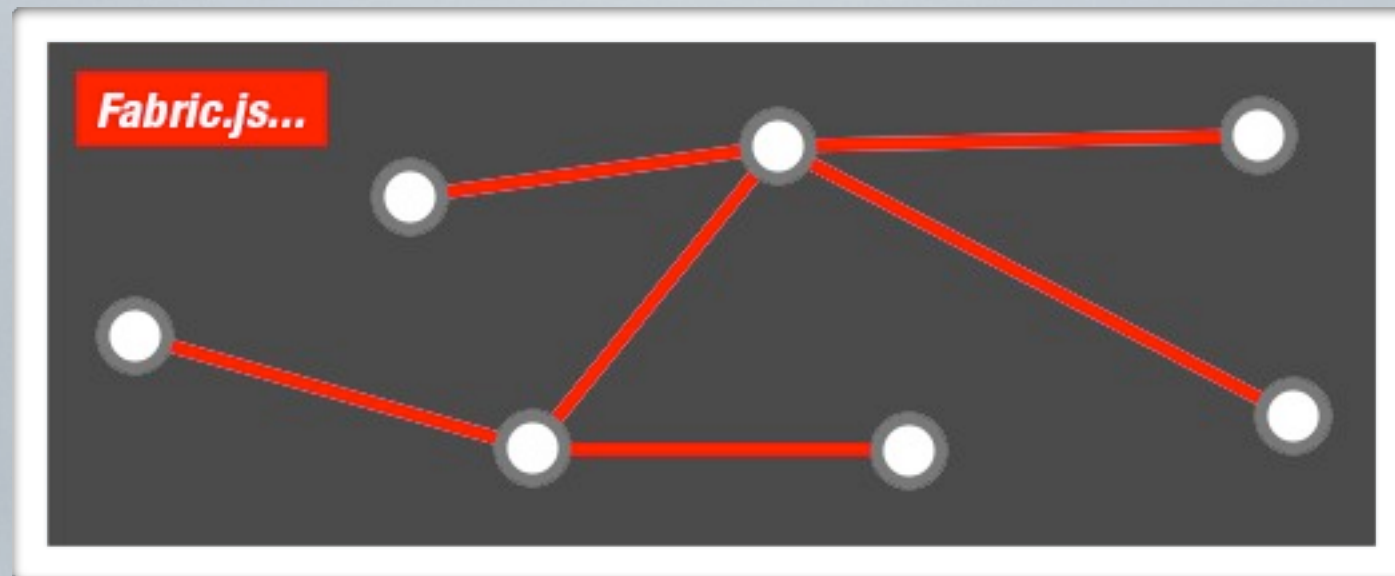


JavaScript toolkit for audio

Reverb, Filters, Delays, Oscillators, FFT, Sequencers, Samplers

LIBRARIES

fabric.js



Interactive object model on top of the HTML5 canvas

Draw shapes & images and animate

LIBRARIES

kinetic.js



Interactive object model on top of the HTML5 canvas

Draw shapes & images and animate

AESTHETICS



Framework for graphical user interface development

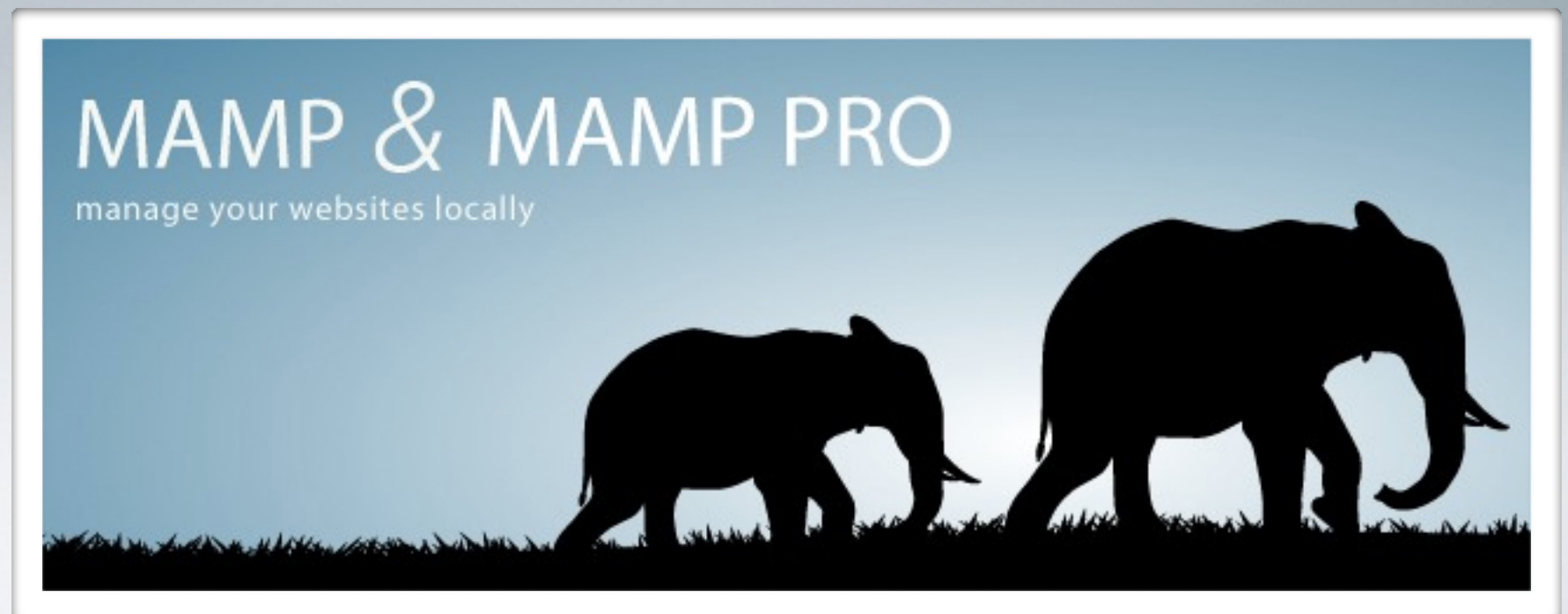
AESTHETICS



User interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript Library

AESTHETICS

Local Testing



NEON.js

Neume Editor Online

SIMSSA

Single Interface for Music Score Searching and Analysis



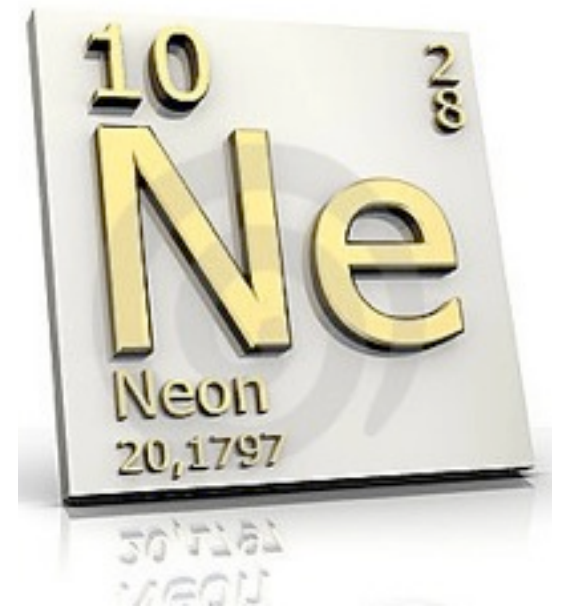
RODAN

Remote Online Document Analysis Network



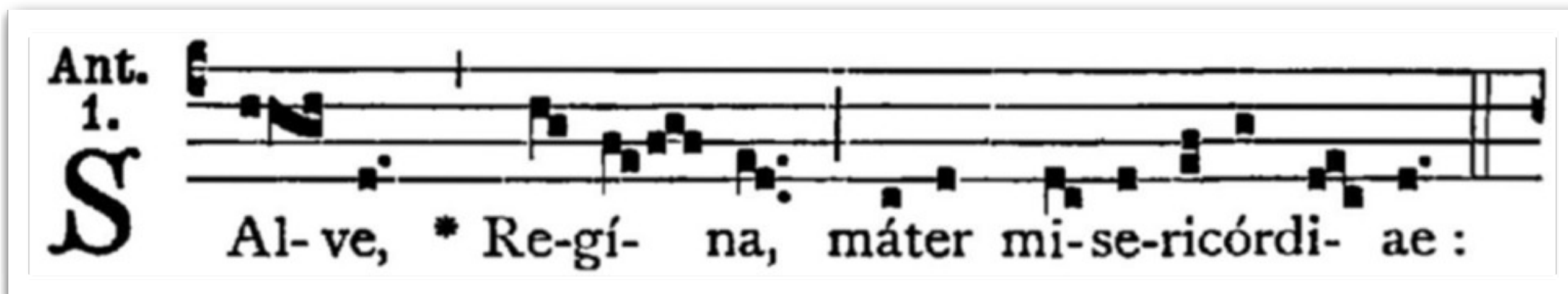
NEON.js

Neume Editor ONline

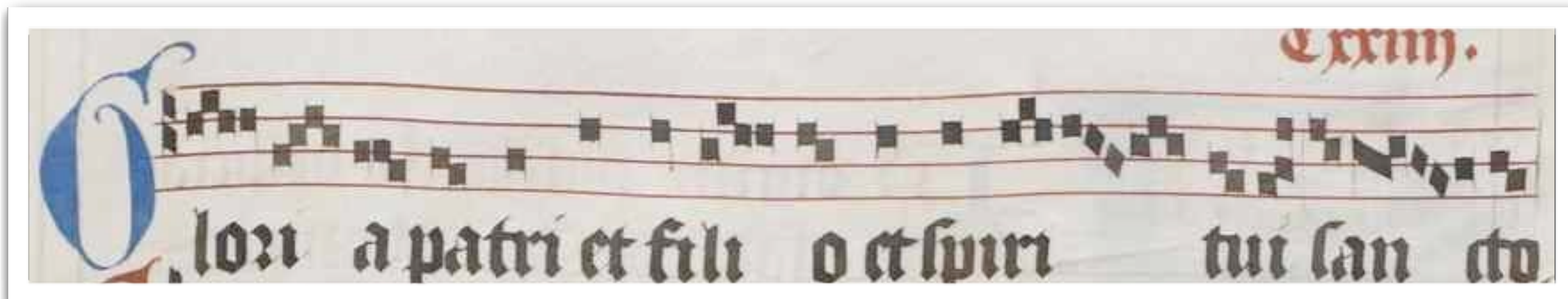


Neume Notation Editor

A web-based music notation editor for square-note (neume) notation



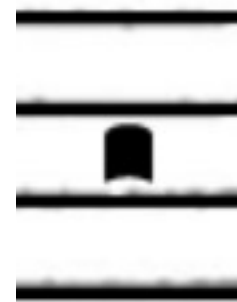
Liber Usualis



Salzinnes Antiphonal

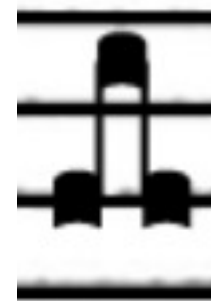
Square-note Notation

This is a neume:

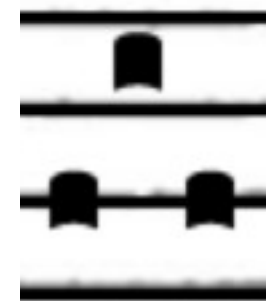


punctum

This is also a neume:



torculus

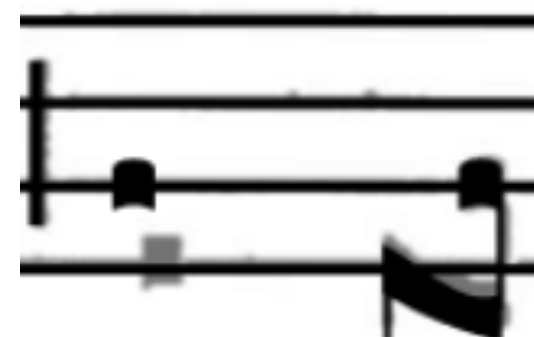


underlying
notes

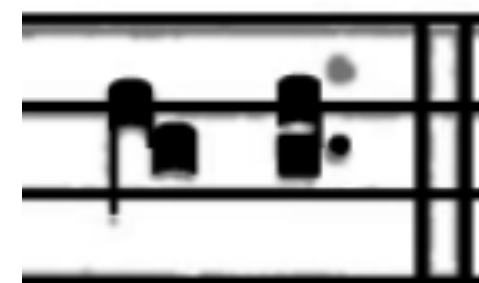
Project Aim

Allow the user to fix Optical Music Recognition (OMR) errors.

1. Pitch and position errors



2. Ornamentation errors



3. Neume recognition errors



Project Aim

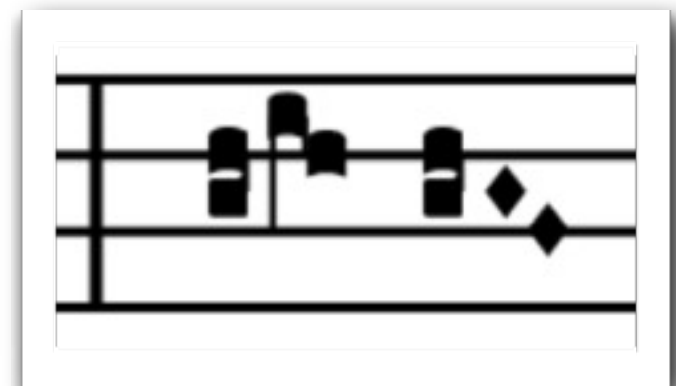
1. Editor can be accessed through a web browser
2. Interface that requires minimal musical knowledge to fix OMR errors
3. Edit properties of individual notes within complex neumes
4. Modify underlying symbolic music file: MEI

Software Design

- Pluggable drawing libraries for different manuscripts



Salzinnes Antiphonal

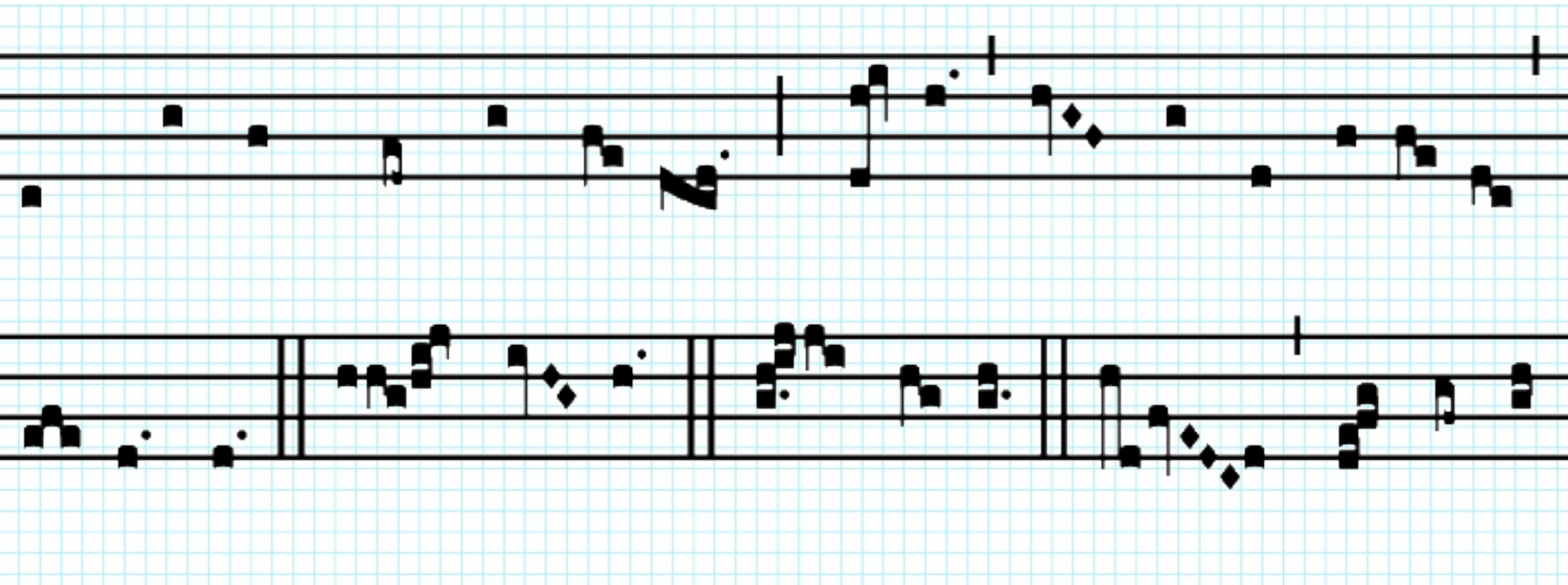


Liber Usualis

- Client-server architecture

Technical Details

- Drawing in the web browser
MEI  HTML canvas (Fabric.js)



Data Structures

staff



`staff.elements = [clef, neume, neume, neume, division, ...]`

clef

```
var staffPos = 0;  
var clefShape = "c";  
var boundingBox = [x1,y1,x2,y2];
```

neume

```
var notes = [n1,n2,n3];  
var name = "torculus";  
var boundingBox = [x1,y1,x2,y2];
```

division

```
var type = "final";  
var boundingBox = [x1,y1,x2,y2];
```

Neon.js
File
Developers | DOMAL

Editing 0400_corr.mei

BACKGROUND
Image Opacity:

MODE

+ Insert

Edit

EDIT

Delete

Neumify

Ungroup

Information

Selected: Porrectus
Pitch(es): A3, G3, A3

Neon.js Editor

Thank you