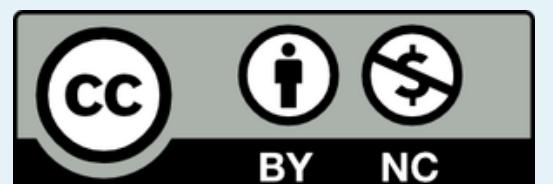


メディアアート・プログラミング2

東京藝術大学 芸術情報センター開設科目 後期金曜4限 第3週

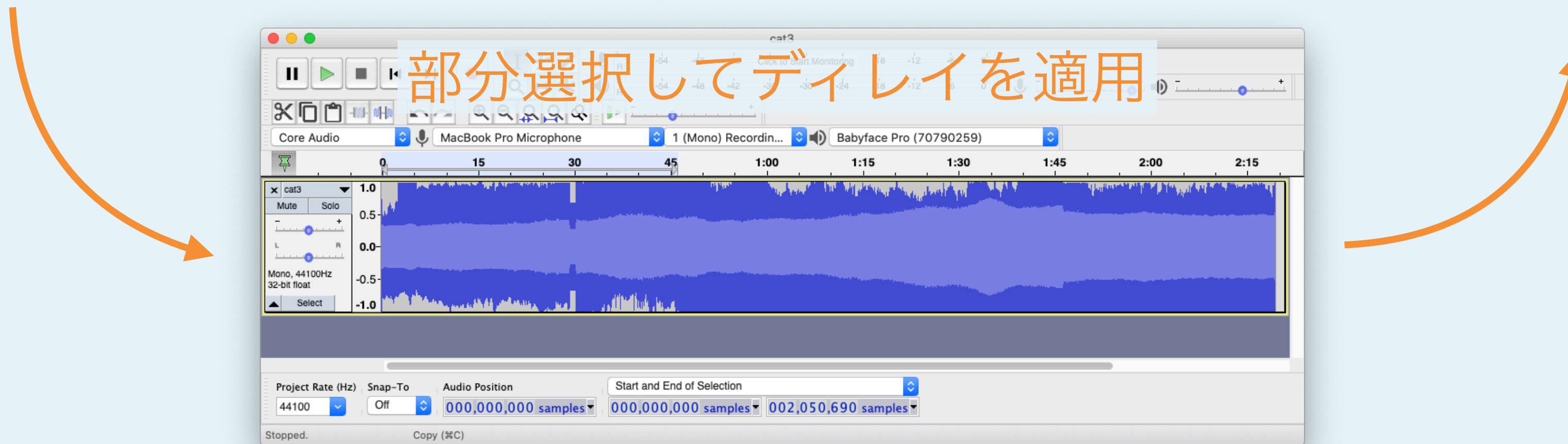
2023.10.20 松浦知也 (matsura.tomoya@noc.geidai.ac.jp teach@matsuuratomo.ya.com)



バイナリデータを感じてみよう

デジタルデータとしての音って？

- 画像をオーディオとして読み、オーディオを編集してから画像として保存



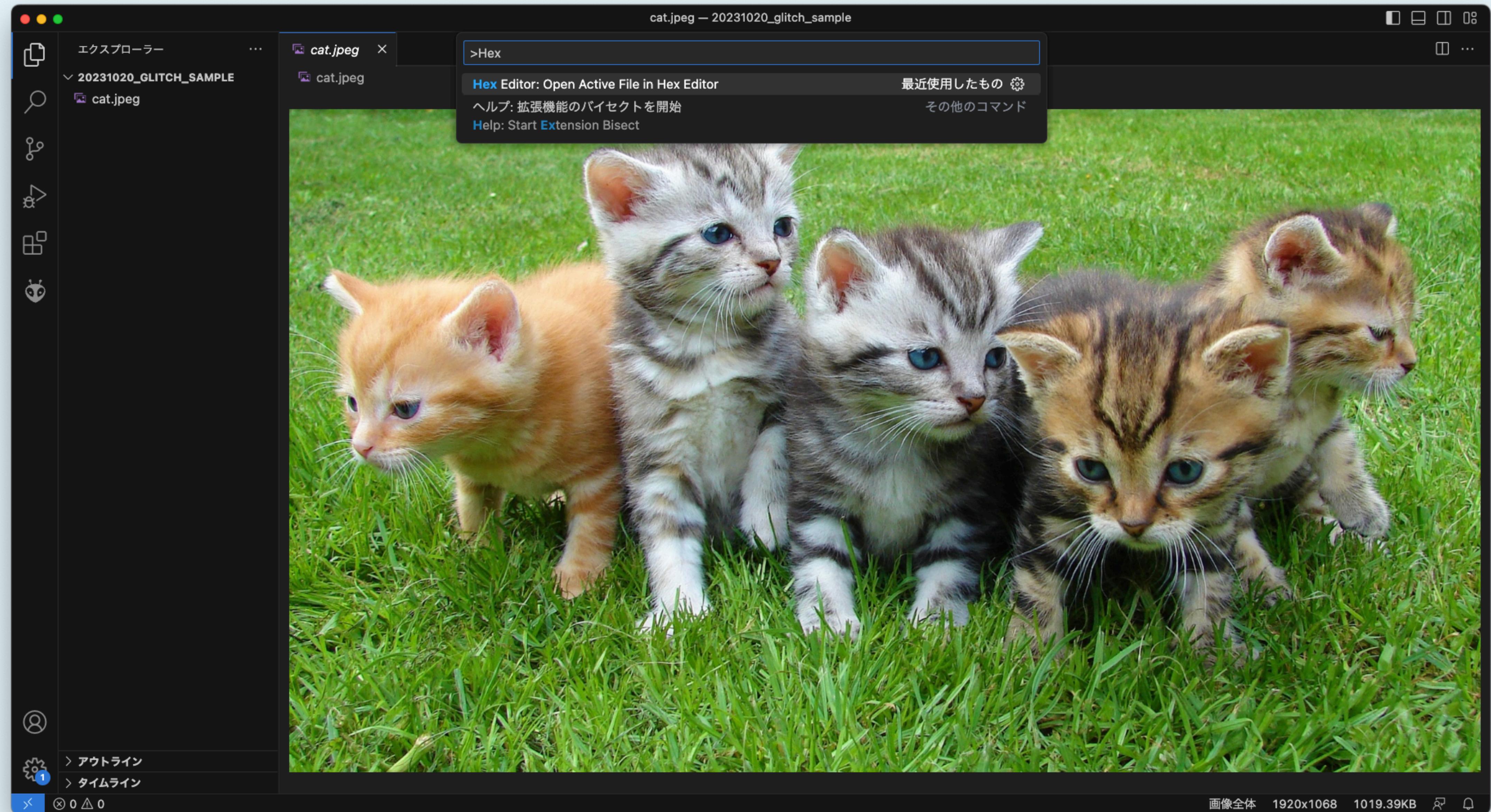
バイナリデータを読んでみよう

<https://marketplace.visualstudio.com/items?itemName=ms-vscode.hexeditor>

- Visual Studio CodeにHex Editor拡張機能をインストール



Visual Studio Codeで適当な画像を入れたフォルダを開き、画像を選択



Cmd+Shift+Pでコマンドパレットを開き、"Hex"で検索
“Open Active File in Hex Editor”を選択

エクスプローラー

20231020_GLITCH_SAMPLE

cat.jpeg

cat.jpeg

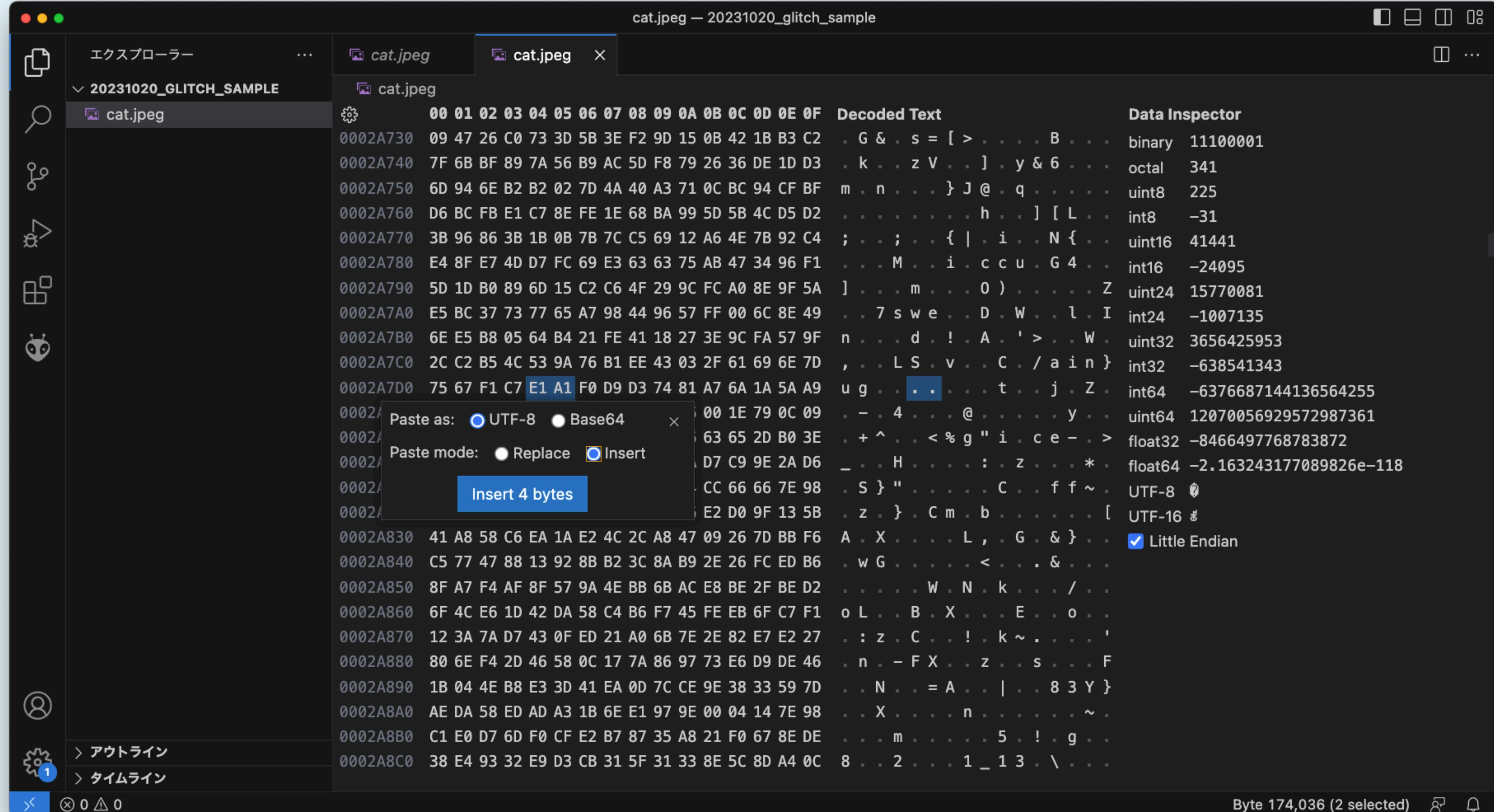
cat.jpeg

Decoded Text

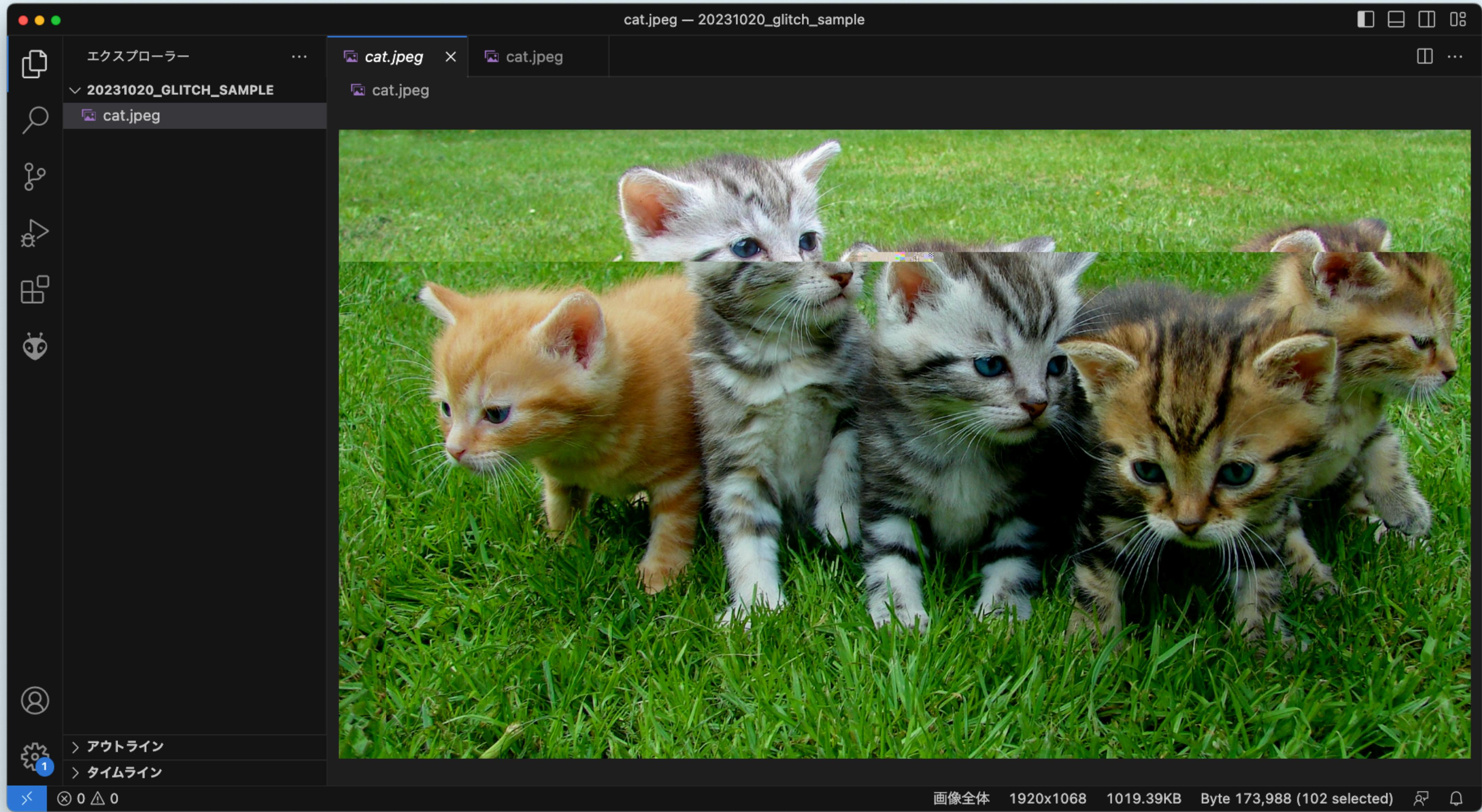
Address	Hex	Text
00000000	FF D8 FF E0 00 10 4A 46 49 46 00 01 01 00 00 01	J F I F
00000010	00 01 00 00 FF FE 00 3B 43 52 45 41 54 4F 52 3A	; C R E A T O R :
00000020	20 67 64 2D 6A 70 65 67 20 76 31 2E 30 20 28 75	g d - j p e g v 1 . 0 (u
00000030	73 69 6E 67 20 49 4A 47 20 4A 50 45 47 20 76 36	s i n g I J G J P E G v 6
00000040	32 29 2C 20 71 75 61 6C 69 74 79 20 3D 20 39 35	2) , q u a l i t y = 9 5
00000050	0A FF DB 00 43 00 02 01 01 01 01 02 01 01 01 C
00000060	02 02 02 02 04 03 02 02 02 02 05 04 04 03 04
00000070	06 05 06 06 06 05 06 06 06 07 09 08 06 07 09 07
00000080	06 06 08 0B 08 09 0A 0A 0A 0A 0A 06 08 0B 0C 0B
00000090	0A 0C 09 0A 0A 0A FF DB 00 43 01 02 02 02 02 02 C
000000A0	02 05 03 03 05 0A 07 06 07 0A 0A 0A 0A 0A 0A 0A
000000B0	0A
000000C0	0A
000000D0	0A FF C0 00 11 08
000000E0	04 2C 07 80 03 01 22 00 02 11 01 03 11 01 FF C4	, "
000000F0	00 1F 00 00 01 05 01 01 01 01 01 00 00 00 00
00000100	00 00 00 00 01 02 03 04 05 06 07 08 09 0A 0B FF
00000110	C4 00 B5 10 00 02 01 03 03 02 04 03 05 05 04 04
00000120	00 00 01 7D 01 02 03 00 04 11 05 12 21 31 41 06	. . . } . . . ! 1 A .
00000130	13 51 61 07 22 71 14 32 81 91 A1 08 23 42 B1 C1	. Q a . " q . 2 . . # B .
00000140	15 52 D1 F0 24 33 62 72 82 09 0A 16 17 18 19 1A	. R . . \$ 3 b r
00000150	25 26 27 28 29 2A 34 35 36 37 38 39 3A 43 44 45	% & ' () * 4 5 6 7 8 9 : C D E
00000160	46 47 48 49 4A 53 54 55 56 57 58 59 5A 63 64 65	F G H I J S T U V W X Y Z c d e
00000170	66 67 68 69 6A 73 74 75 76 77 78 79 7A 83 84 85	f g h i j s t u v w x y z . . .
00000180	86 87 88 89 8A 92 93 94 95 96 97 98 99 9A A2 A3
00000190	A4 A5 A6 A7 A8 A9 AA B2 B3 B4 B5 B6 B7 B8 B9 BA

Byte 2,420 (6 selected)

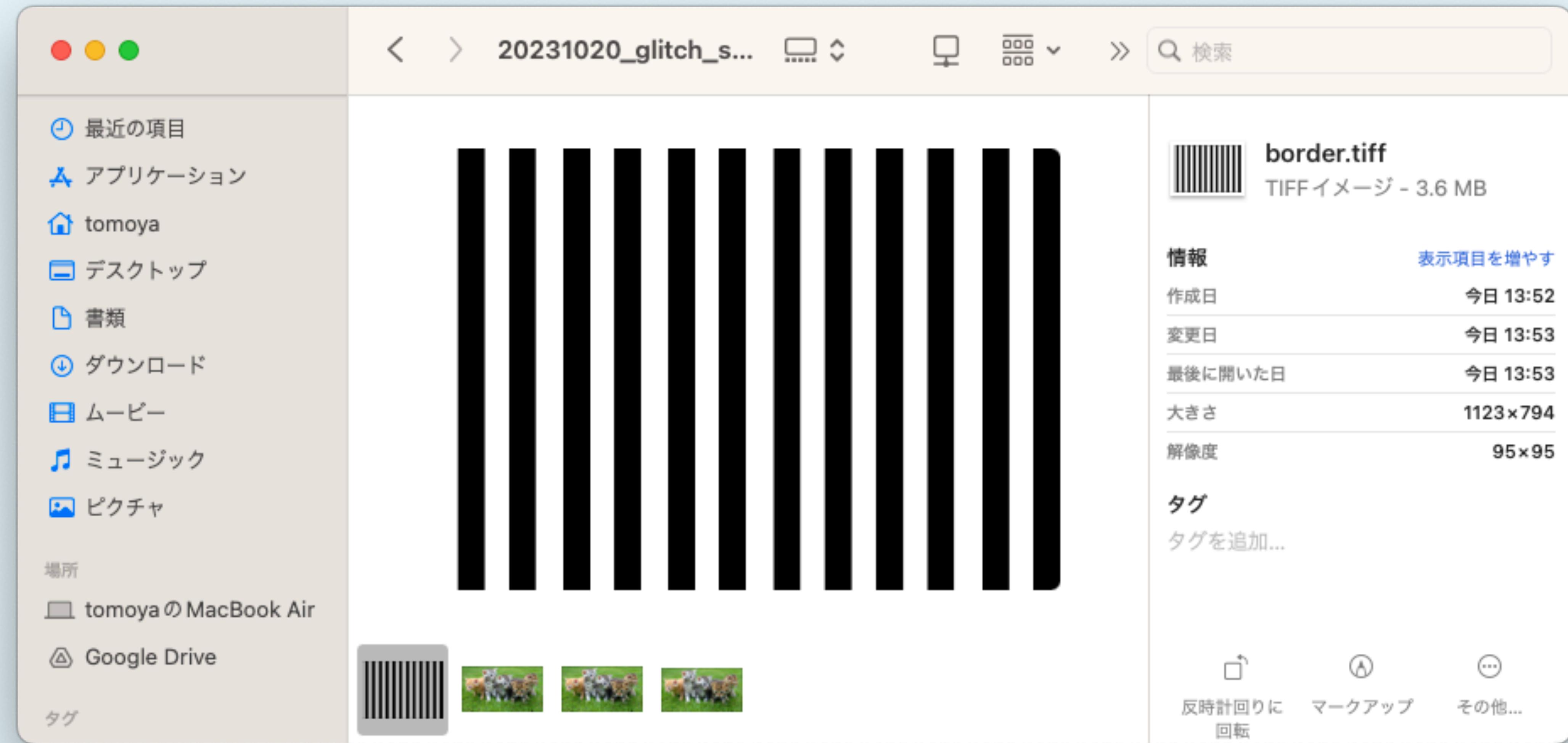
開かれた様子



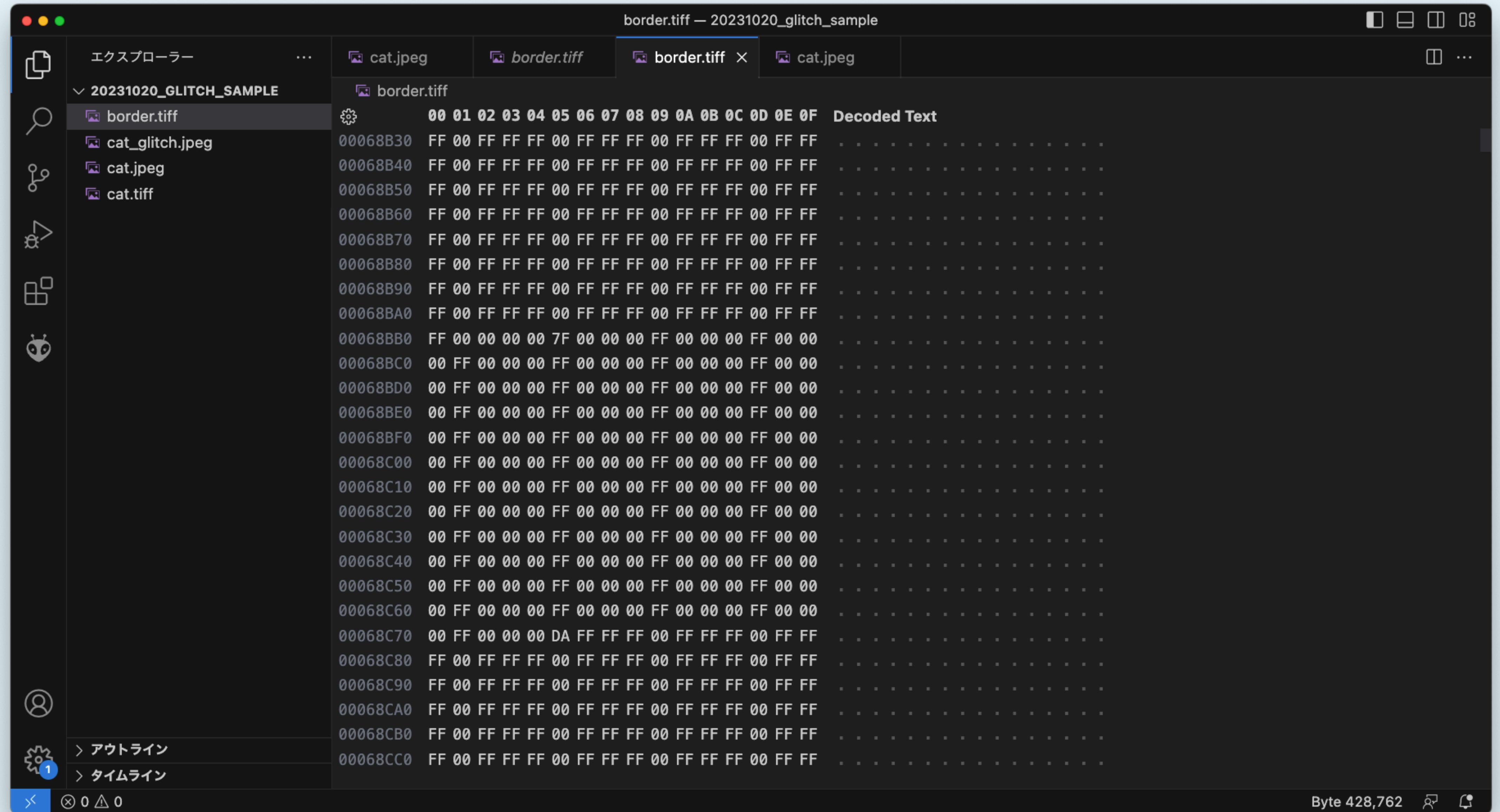
真ん中あたりまでスクロールして、適当な範囲を選択、Cmd+Cでコピー、Cmd+Vで貼り付けると
画面のようなメニューが出る。Insertを選択



不要なデータが挿入されたため、画像がズレる



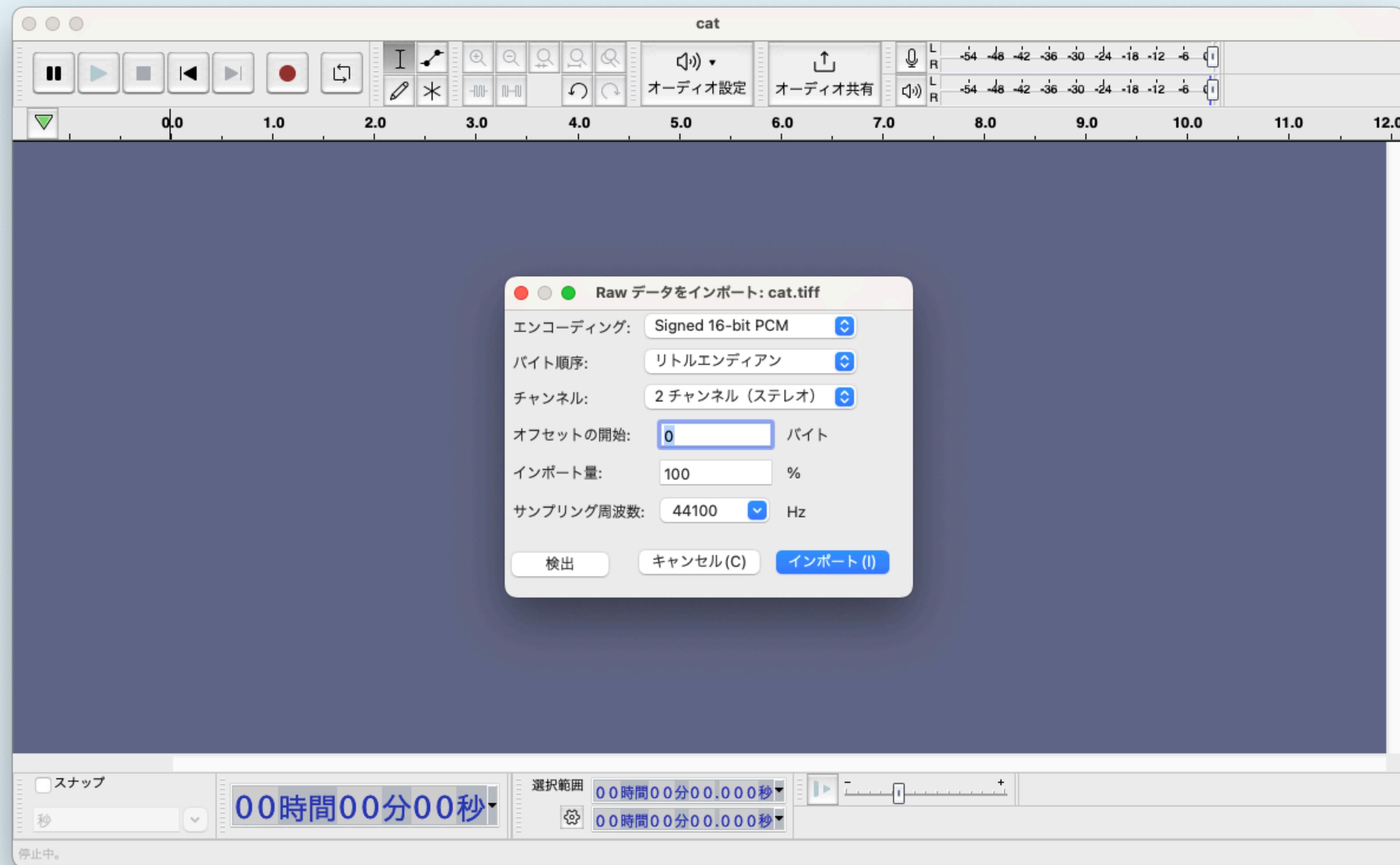
JPEGは圧縮されていたが、TIFF (WindowsならBMP) のような非圧縮フォーマットを使うと、ピクセルがひとつづつデータとして並んでいることが確認できる

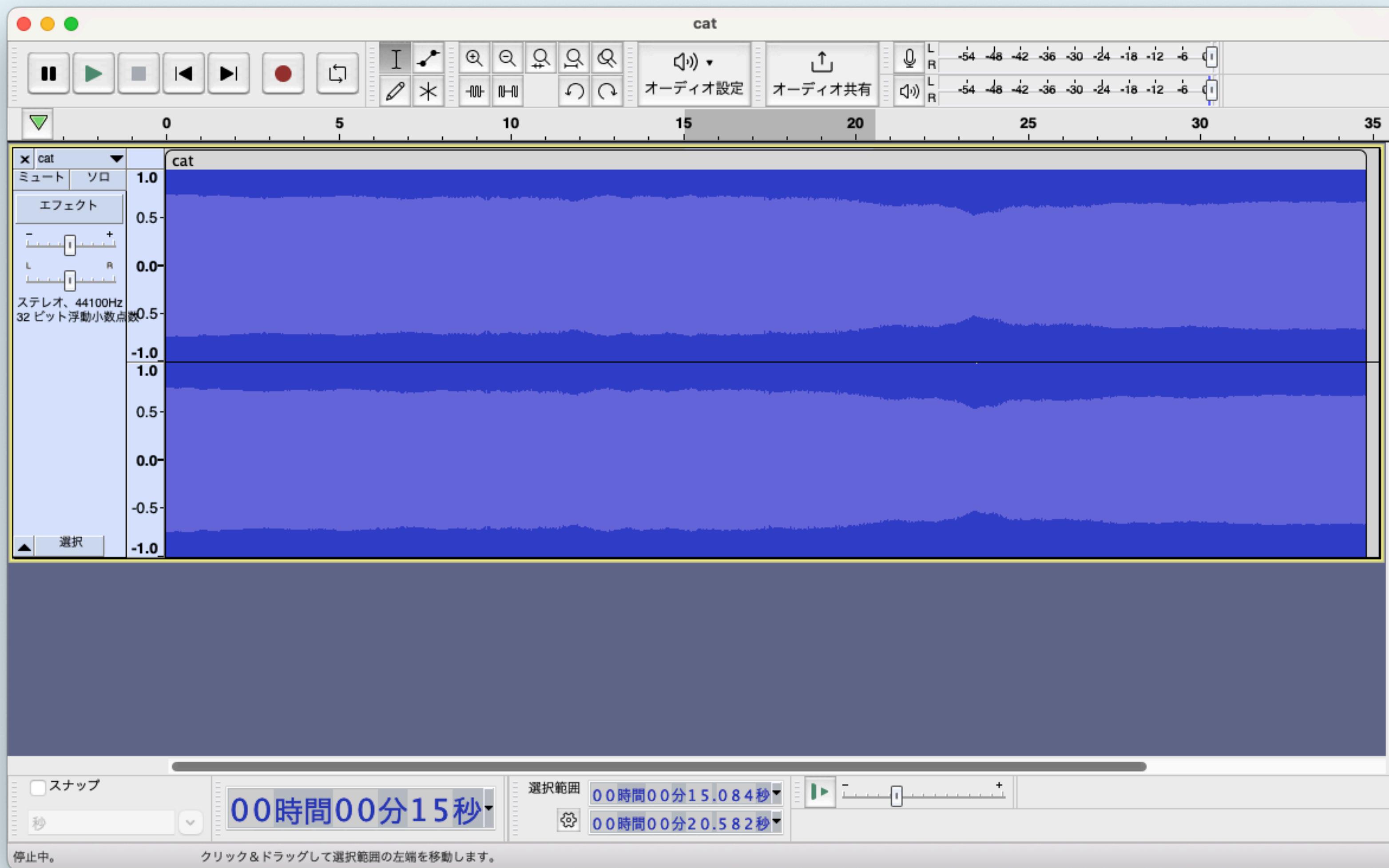


Border.tiffの中身を見ると、00とFF(255)が規則的な間隔で並んで現れている
(イラレで作ったので、アンチエイリアスで間の値がたまに現れている)

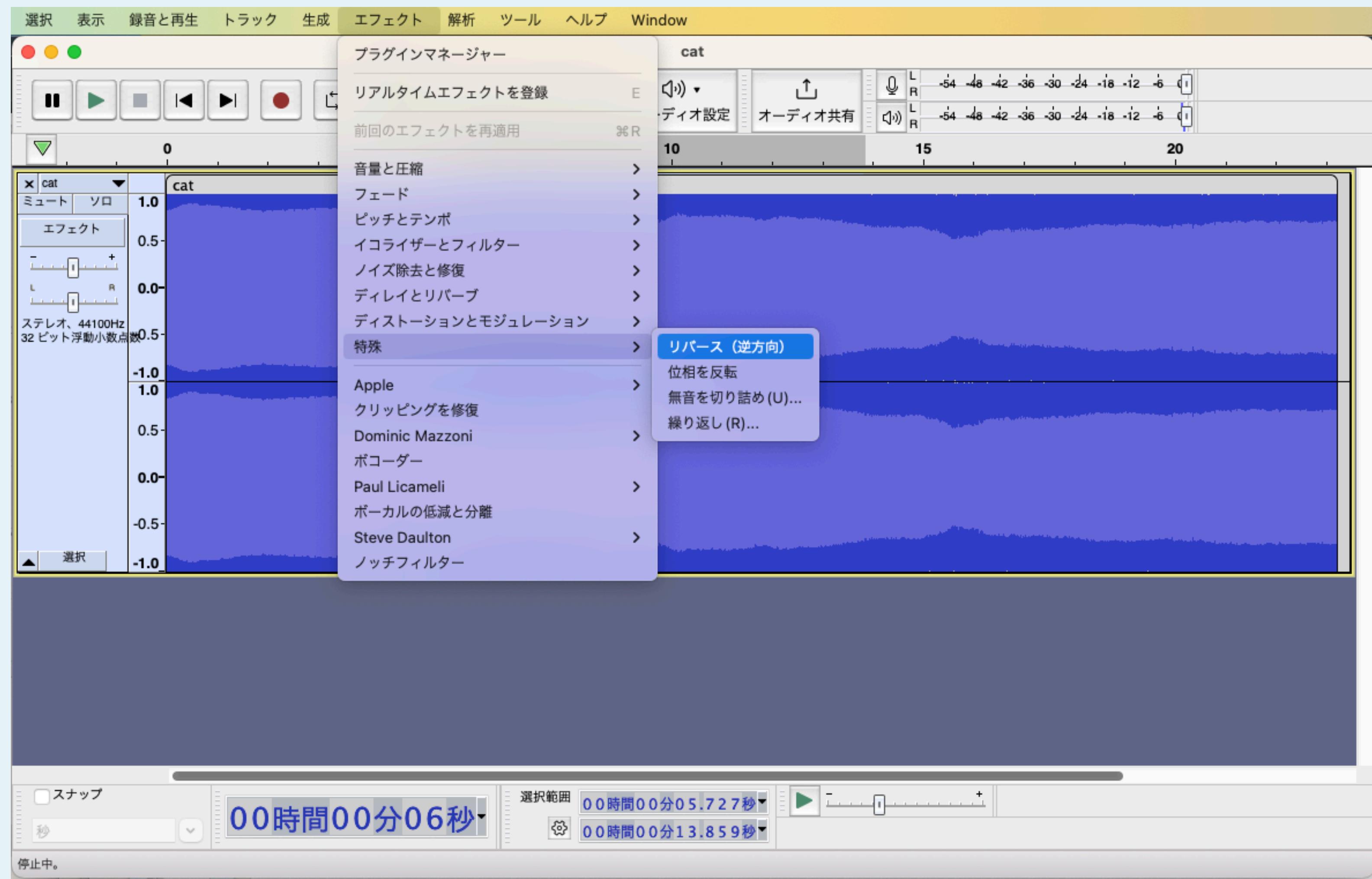
Audacityでグリッヂ

- ファイル→インポート→rawデータをインポートでファイル選択
- 右のようなオプションでTIFFファイルを読み込む
- 8bitか16bit、ステレオがおすすめ（24とかだとメタデータが崩れるかも）
- 真ん中ら辺を選択し、適当にエフェクトをかける
- ExportでもRaw Dataで、読み込みと同じフォーマットで書き出す

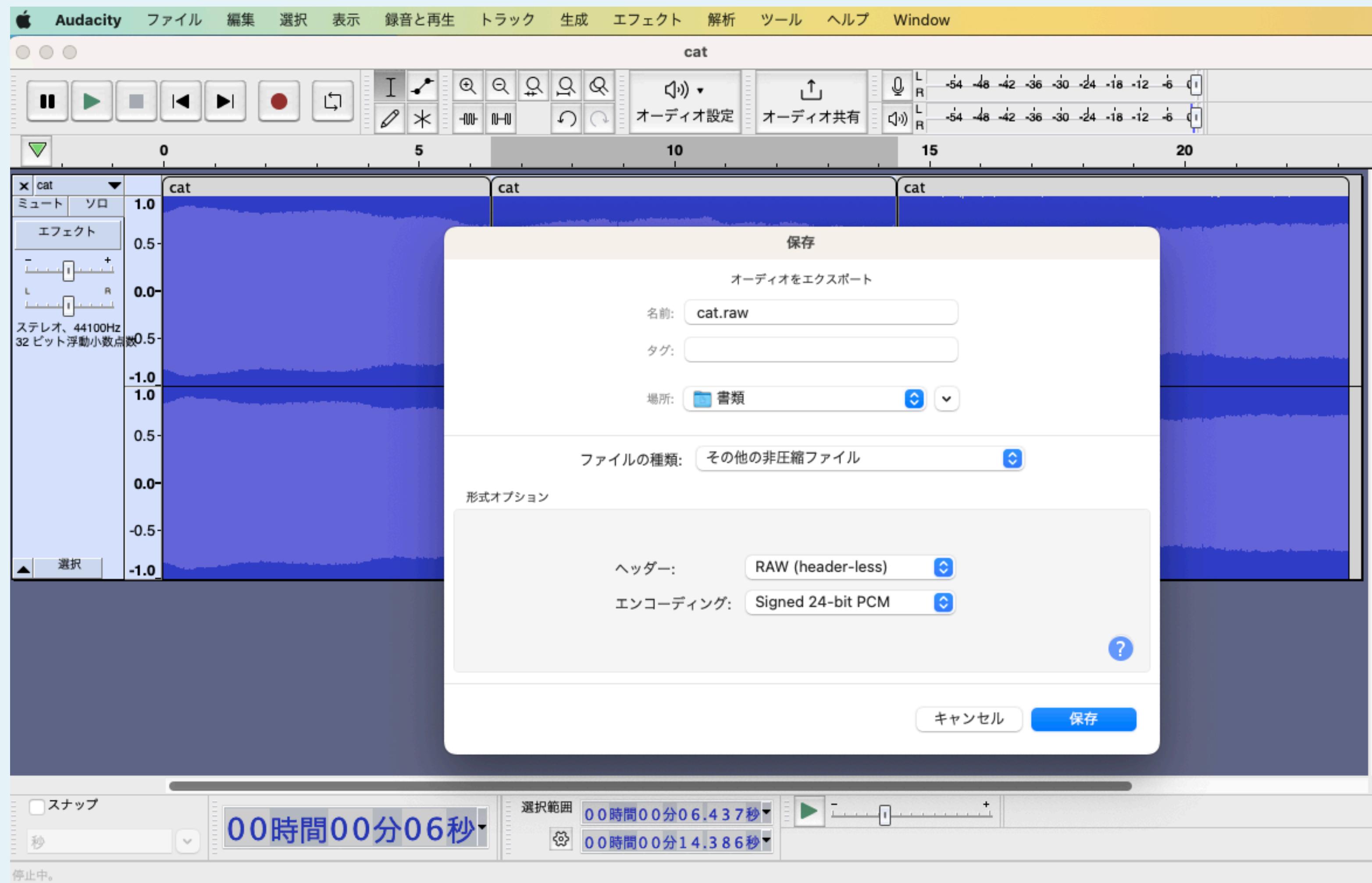




Cat.tifをsingled-16bit,stereoとして読んだところ

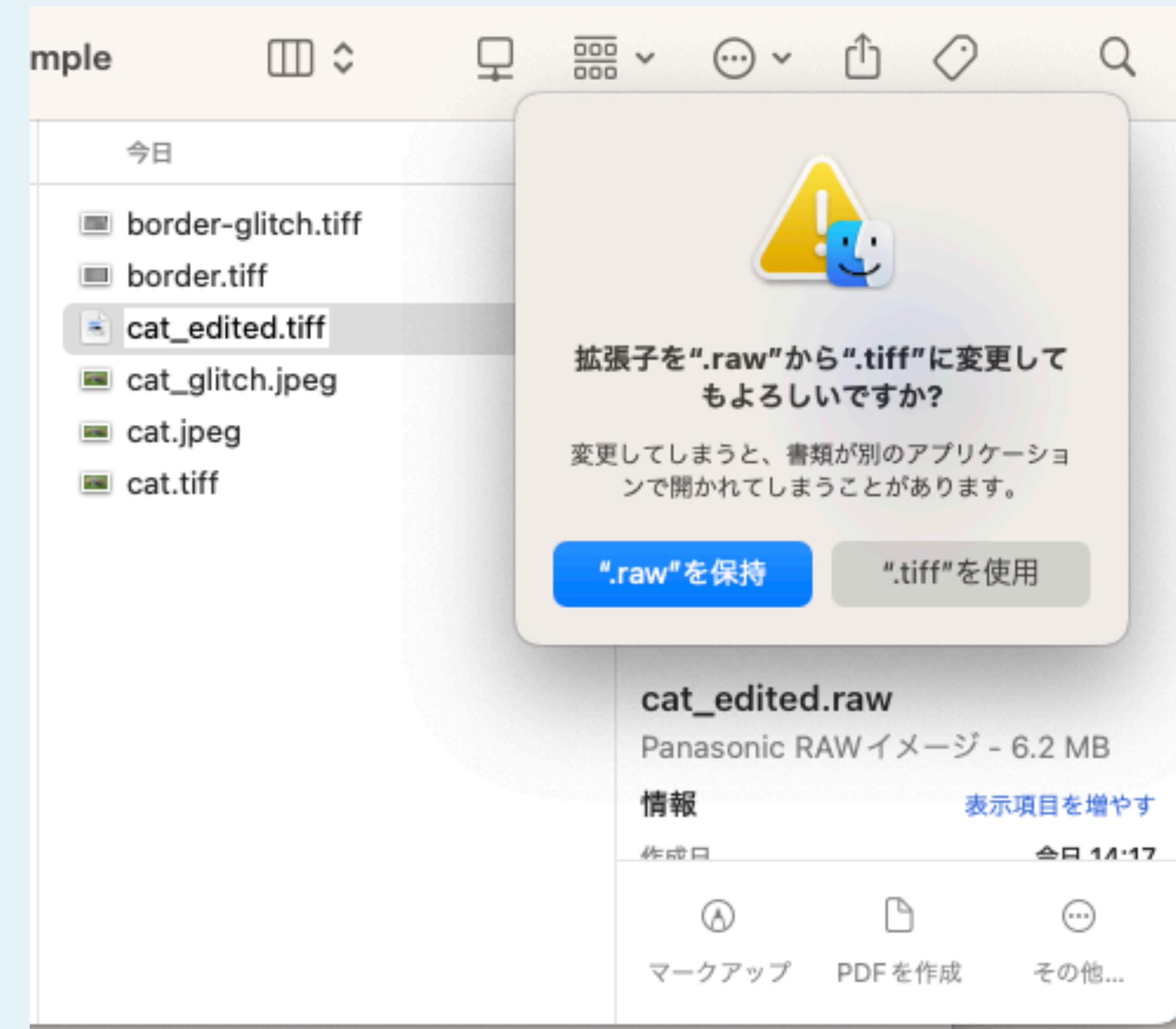


適当な範囲を選択して、リバース（逆方向）



ファイル→オーディオをエクスポート

“その他の非圧縮ファイル”でヘッダーは”RAW(header-less)”、エンコーディングは読み込み時と同じもの
保存の後にメタデータの編集ダイアログが出るが無視



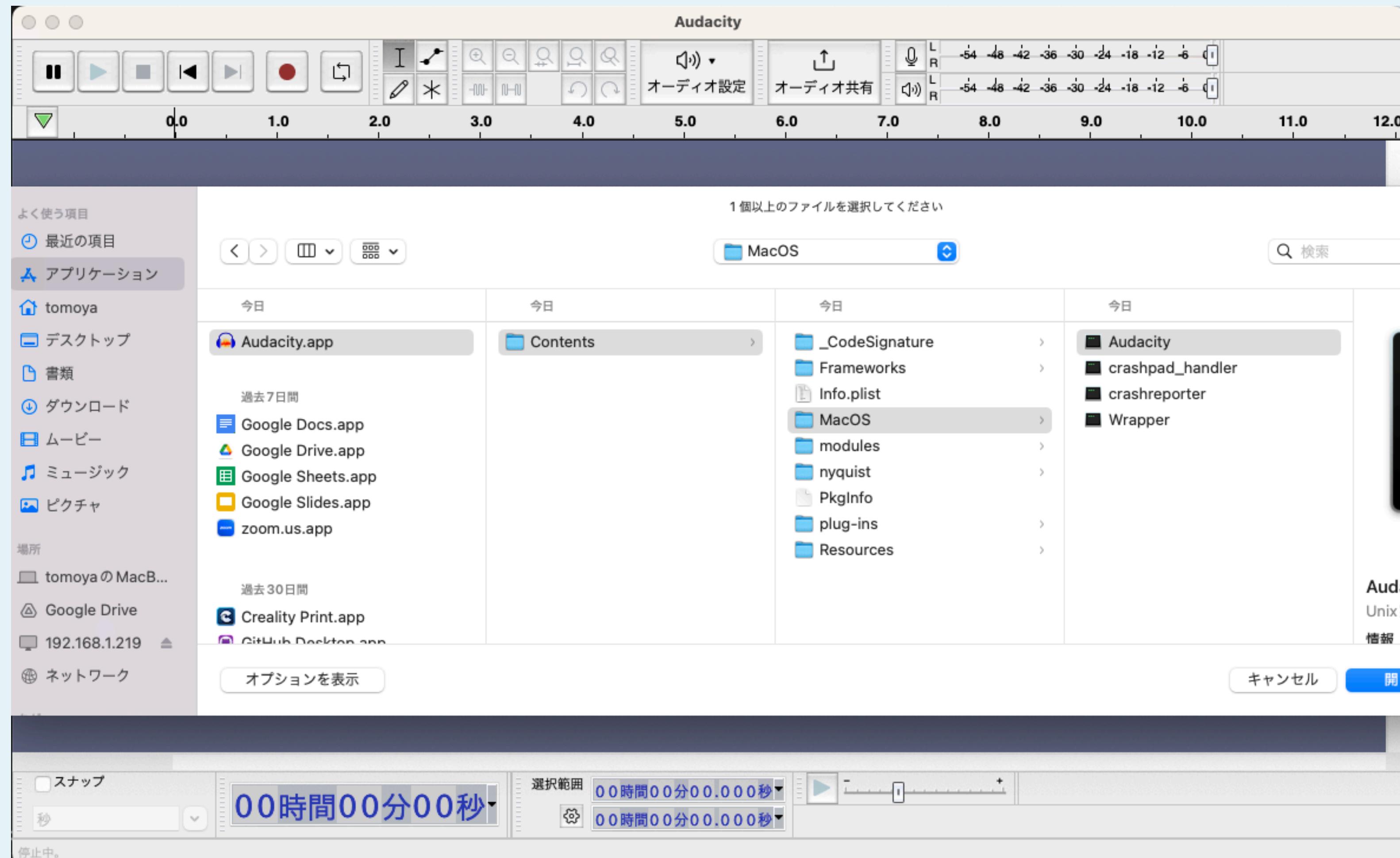
Finderで拡張子を元のtiffに変更(うまくいってたらやらなくてもOK)



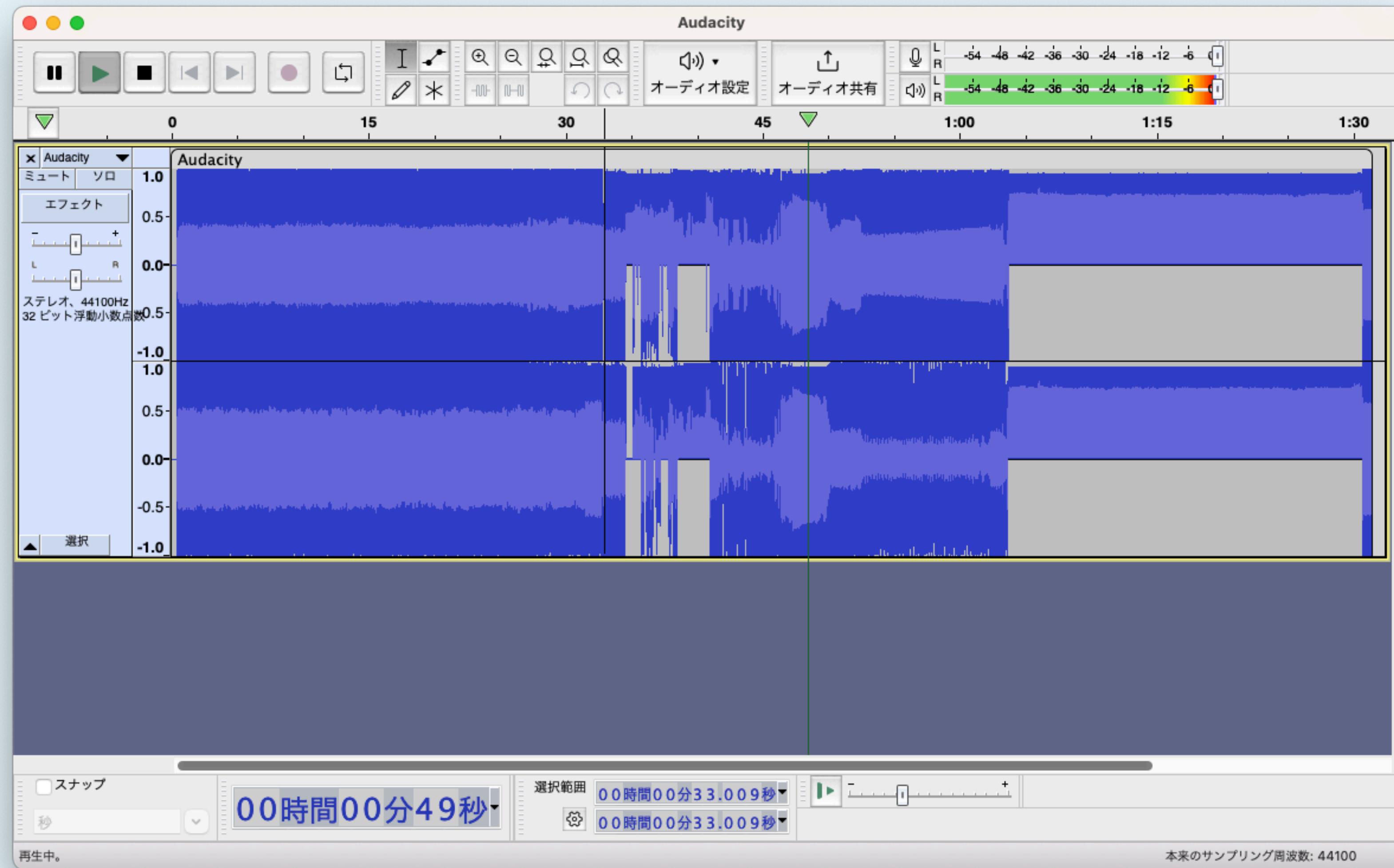
他にもいろんなファイルを聞いてみよう

- 例えば・・・Zipファイルやdocx、xlsx(内部的にzipを使っている)は、圧縮でエントロピーが高まっているためホワイトノイズに近い音になる
- 特にPDFファイルやアプリケーションのバイナリなどは結構複雑な音になりやすい
- アプリケーションは直接開けないので、Finderで、xxx.appを右クリックして、”パッケージの内容を表示”→Contents/MacOSの中にあるファイルを、インポートのダイアログにドラッグ&ドロップ

AudacityでAudacityを聴く



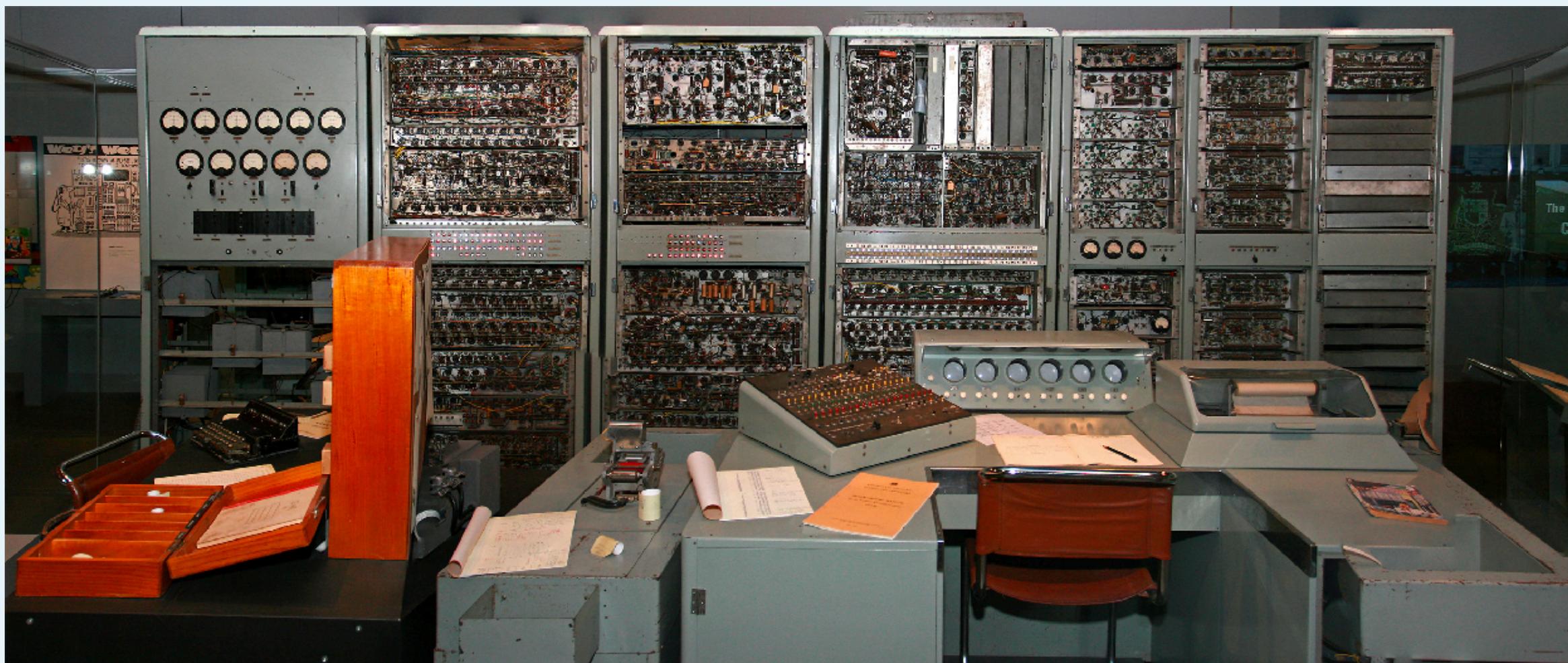
Audacity.app/Contents/MacOS/Audacityがアプリの本体



Signed 24bit、Stereoで聞くとこんな感じ

歴史：1950年代

- 世界初の電子計算機ENIAC(1946)
- デバッグ機能のスピーカーなどを使って音楽を鳴らした例：BINAC(1949),CSIRAC(1950~51)
- 初めての音楽プログラミング言語(非リアルタイム) MUSIC I(1957)
 - 余談：世界初の汎用プログラミング言語FORTRAN(1957)



CSIRAC[1]

ポスト・デジタルとグリッヂ

90s~

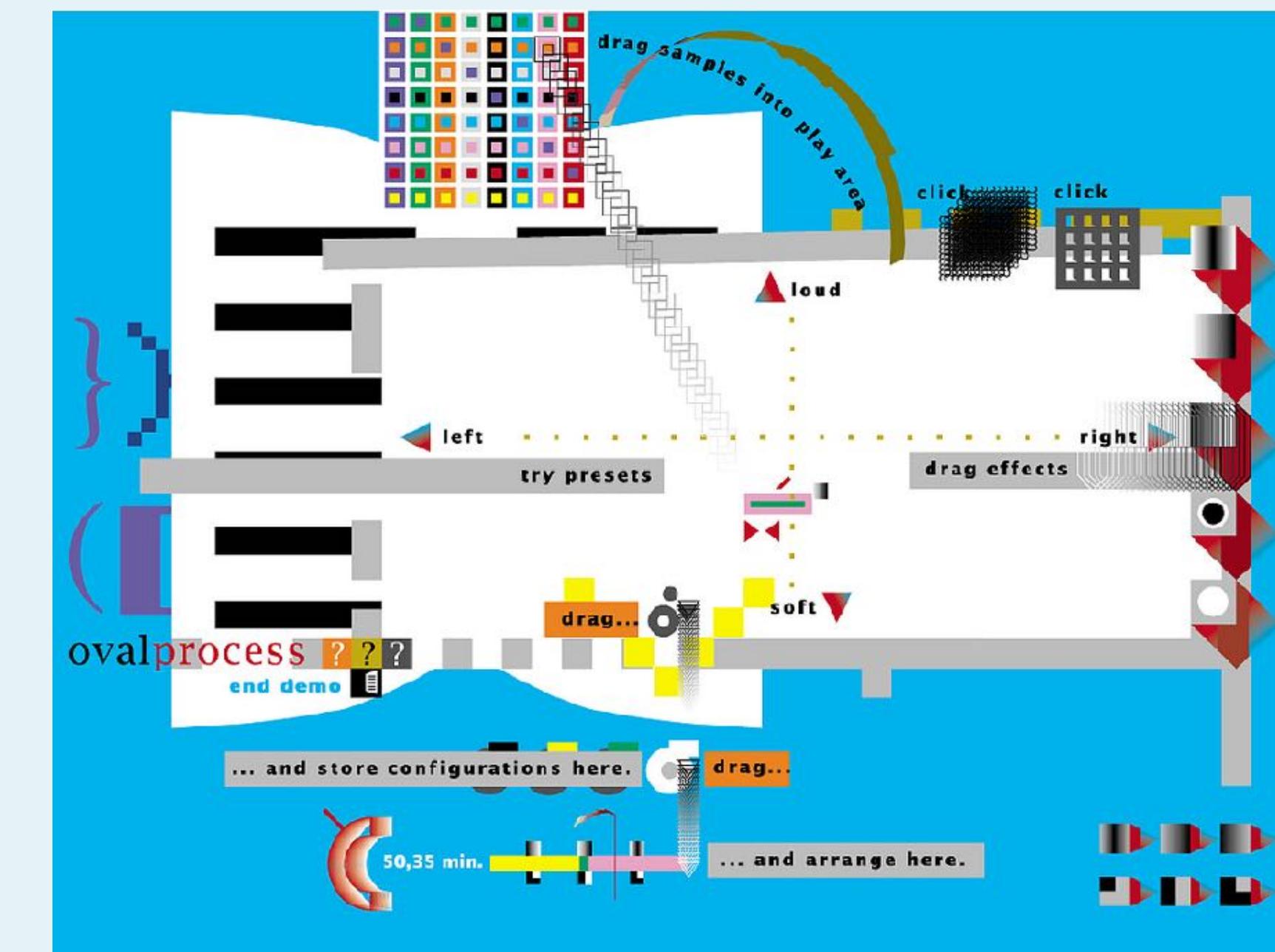
技術の誤用



[2]

サーキットベンディング
グリッヂ

創作環境自体のDIY



[3]

Ovalprocess

Max, Puredataを始めとする
プログラミング環境

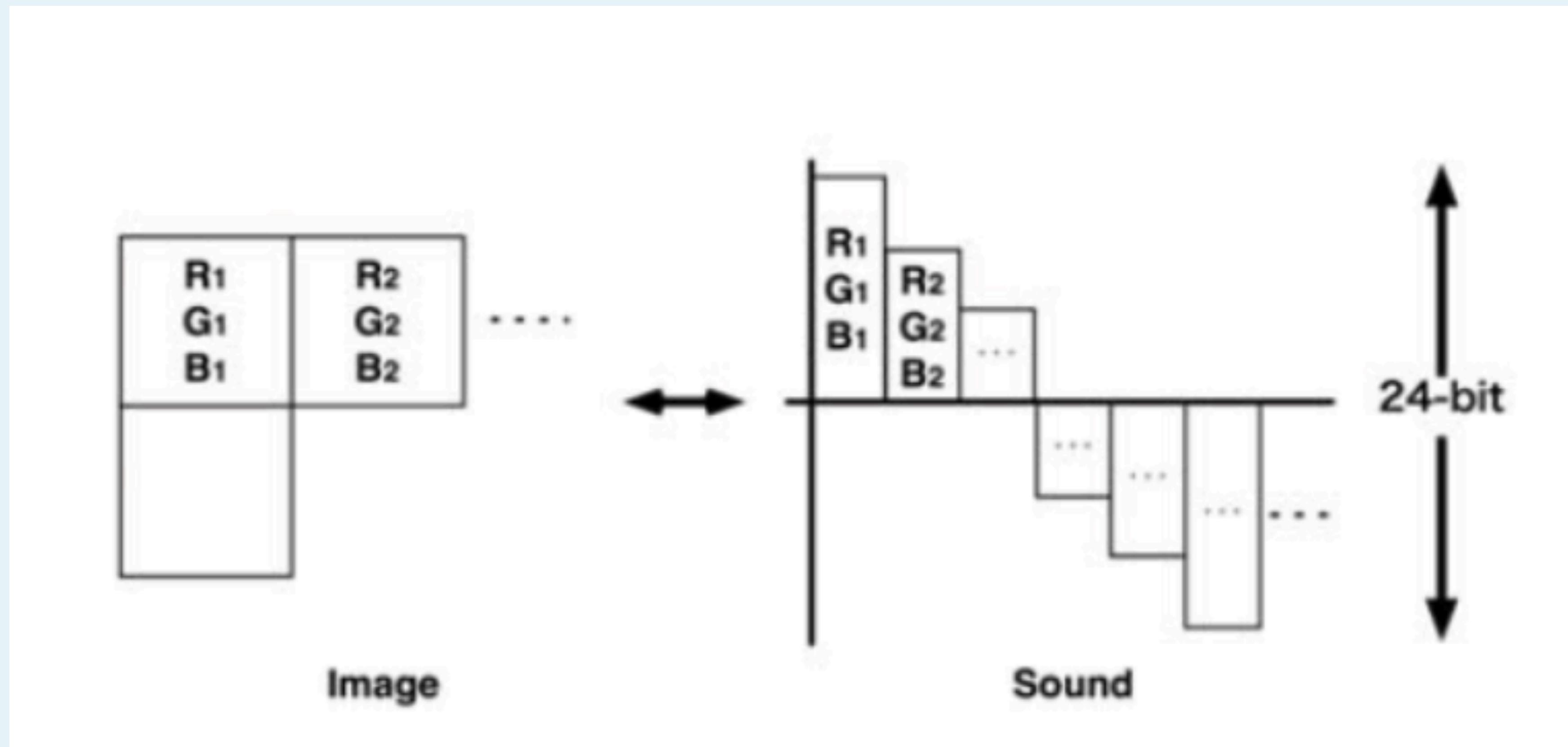
Monalisa(Jo and Nagano, 2005)

“Monalisa: shadow of your sound” [2005]

NAGANO Norihisa + JO Kazuhiro



<https://www.ntticc.or.jp/en/archive/works/monalisa-shadow-of-your-sound/>



Jo and Nagano (2008) より

<https://doi.org/10.5281/zenodo.1179569>

Image Credits

- [1]By jjron - Own work, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=4613186>
- [2]By Loz Pycock from London, UK - Circuit Bent Speak & Spell, The Science Museum, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=32968059>
- [3]Sam Inglis, Markus Popp: Music As Software Oval - Sound on Sound, <https://www.soundonsound.com/people/oval-markus-popp>