KAGRA detchar tool development

Chihiro Kozakai (NAOJ)

2019/10/7 L-K Detchar meeting

Overview

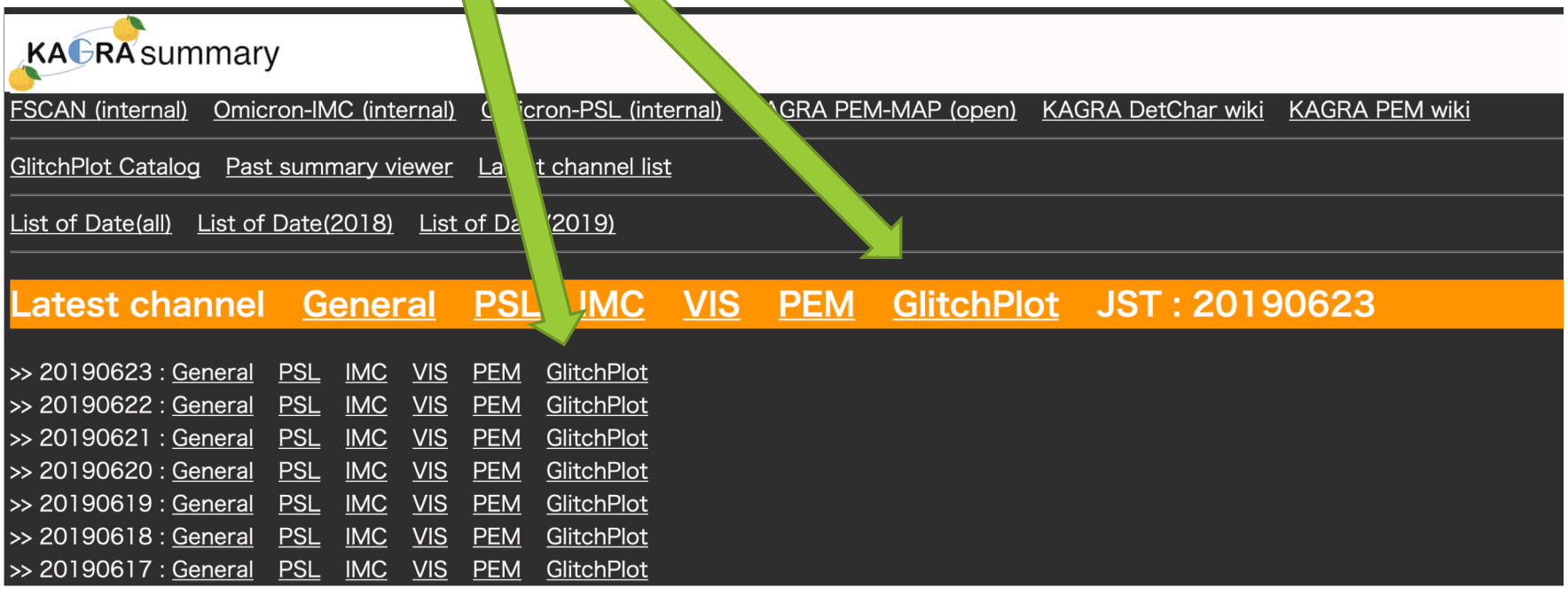
- I developed 2 tools for Detchar activity.
  - GlitchPlot
    - Using trigger information given by external tool, it automatically generates basic plots around the glitch or lock loss for main channel and auxiliary channels.
    - It is used to understand glitch and lock loss by visual inspection.
    - The result is available on web thanks to cooperation by Yuzu-san (ICRR)
  - iKozapy
    - “Kozapy” is sample code collection of GWpy-based plot script.
      - Samples of simple plot code, batch job scripts for KAGRA data system
    - “iKozapy”, interactive Kozapy is GUI tool to make spectrogram or coherencegram.
      - iKozapy is not only interactive but it has real time update function.

GlitchPlot

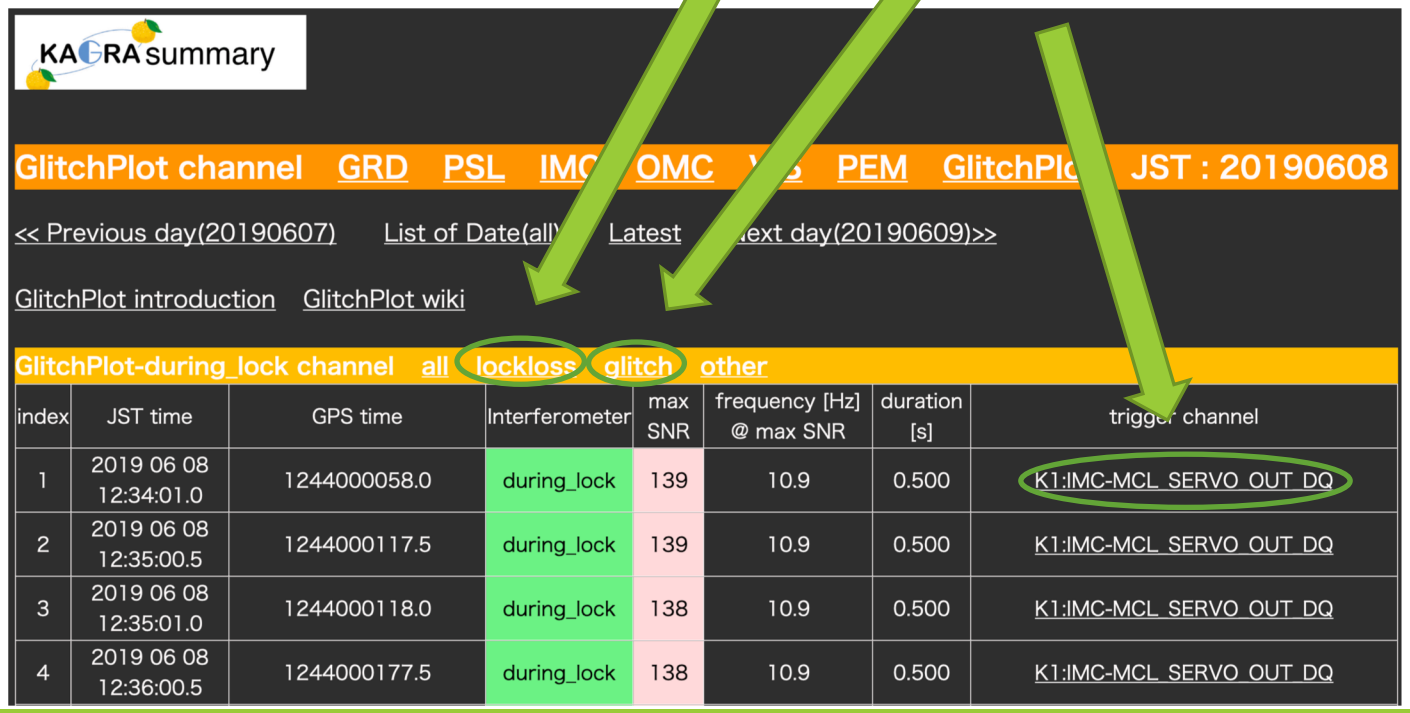
- **Visualizes glitch/lock loss with related channels.** New KAGRA original tool.
- The purposes are
  - **For commissioning**: help identification of the cause of the glitch or lock loss in the most basic and flexible way.
  - **For data analysis**: give convenient tool for detector sanity check.
- For each trigger and related channels, following basic plots are provided. The result can be accessed on webpage.
  - Time series
  - Spectrum before and after trigger
  - Spectrogram
  - Coherencegram with trigger channel
  - Q-transform plot
- The webpage contains comment form for each trigger event.
  - **You can vote for the cause of the glitch.** After many information is collected, we can do statistical analysis with them.
  - Web interface is developed by H. Yuzurihara-san (ICRR).

How to use GlitchPlot

- Go to Yuzu summary page KAGRA summary
  - https://www.icrr.u-tokyo.ac.jp/~yuzu/bKAGRA_summary/html/
- Click GlitchPlot.



- You see list of glitches/lock loss events and information of it.
  - If you have special interest in lock loss or glitch, events can be filtered.
  - Interferometer state and SNR is colored.
- Click the last column of the interested event.



KAGRA summary

GlitchPlot channel GRD PSL IMC OMC VIS PEM GlitchPlot JST : 20190608

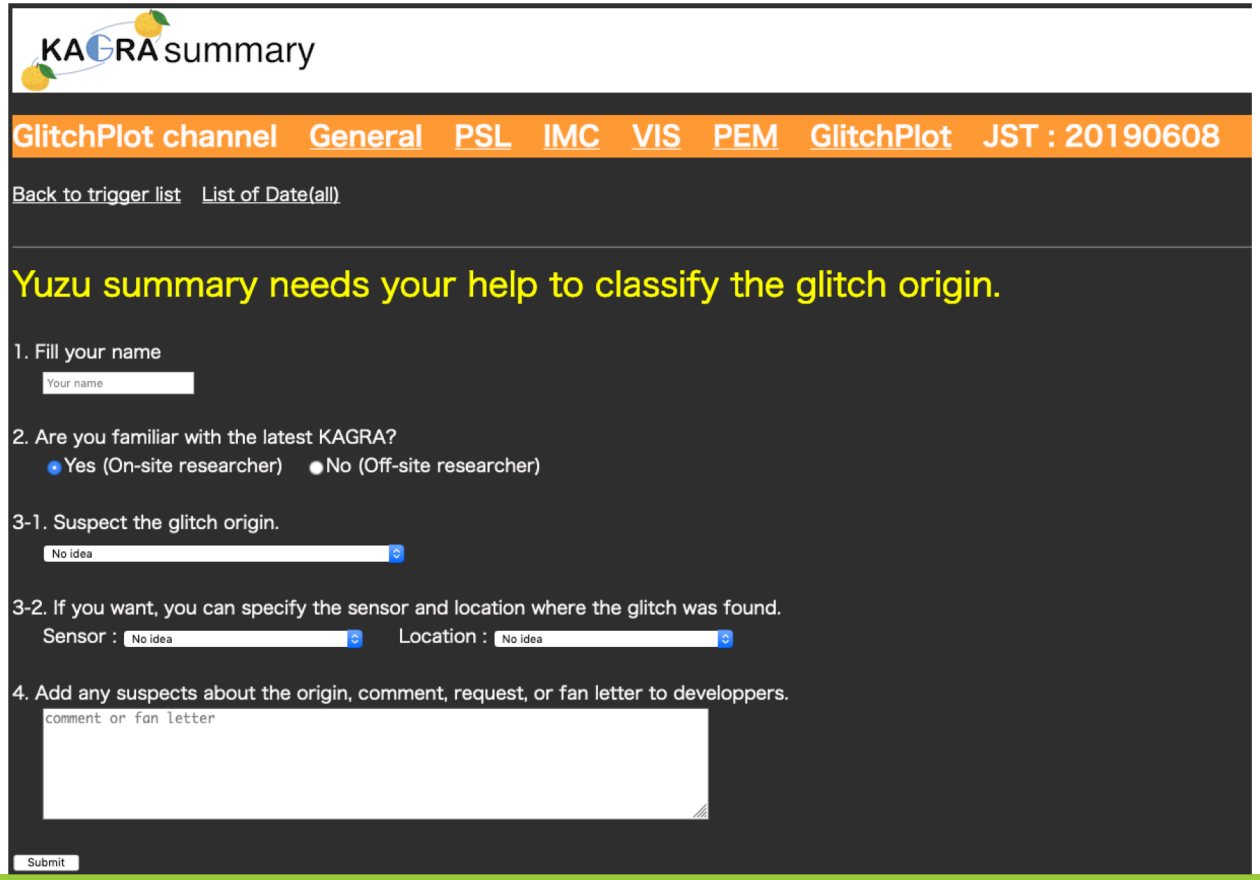
<< Previous day(20190607) List of Date(all) Latest Next day(20190609)>>

GlitchPlot introduction GlitchPlot wiki

GlitchPlot-during_lock channel all lockloss glitch other

index	JST time	GPS time	Interferometer	max SNR	frequency [Hz] @ max SNR	duration [s]	trigger channel
1	2019 06 08 12:34:01.0	1244000058.0	during_lock	139	10.9	0.500	K1:IMC-MCL_SERVO_OUT_DQ
2	2019 06 08 12:35:00.5	1244000117.5	during_lock	139	10.9	0.500	K1:IMC-MCL_SERVO_OUT_DQ
3	2019 06 08 12:35:01.0	1244000118.0	during_lock	138	10.9	0.500	K1:IMC-MCL_SERVO_OUT_DQ
4	2019 06 08 12:36:00.5	1244000177.5	during_lock	138	10.9	0.500	K1:IMC-MCL_SERVO_OUT_DQ

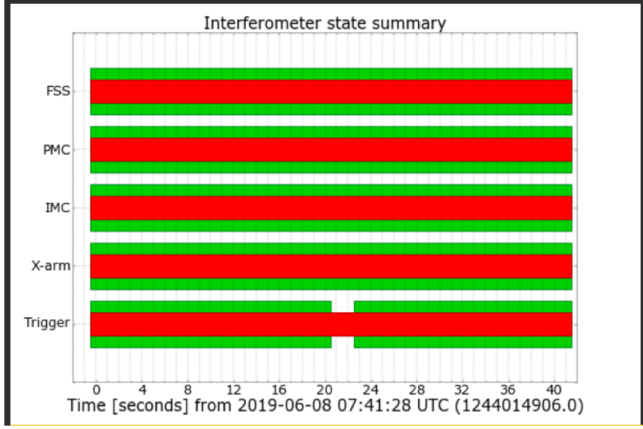
- Top of the each page is form to report your opinion about the glitch origin.



- Glitch information is given.
- Interferometer state around glitch time is given. The bottom red region is triggered time.

Trigger information

index	JST time	GPS time	Interferometer	max SNR	frequency [Hz] @ max SNR	duration [s]	trigger channel
420	2019 06 08 16:41:49.5	1244014926.5	during_lock	18	12.8	2.000	K1:LSC-CARM_SERVO_MIXER_DAQ_OUT_DQ



- The result starts from triggered channel. The center of the plot is trigger time. Parameters are selected according to the trigger information.

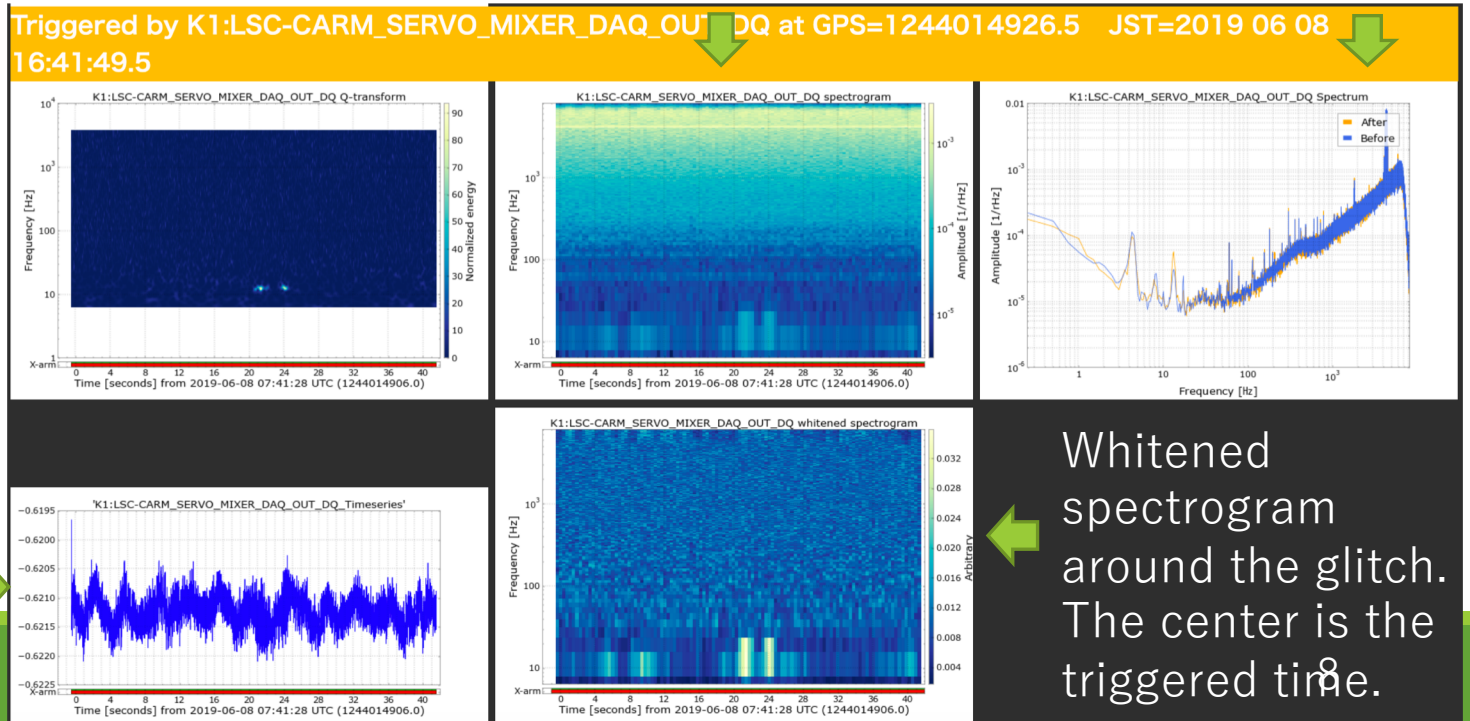
Spectrogram around the glitch. The center is the triggered time.

The spectrum
Before trigger (-30 ~ -2 sec)
After trigger (+2 ~ +30 sec)

Q-transform

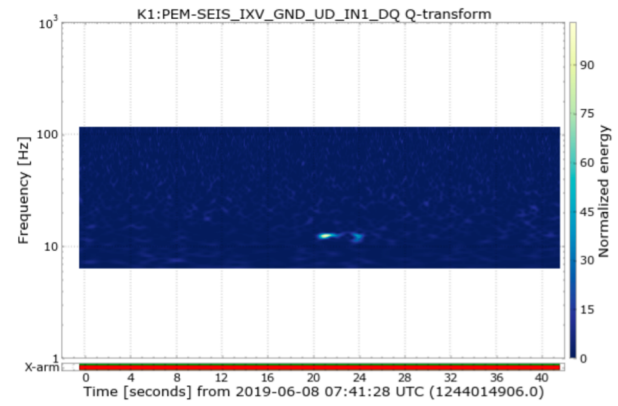
Time series around the glitch. The center is the triggered time.

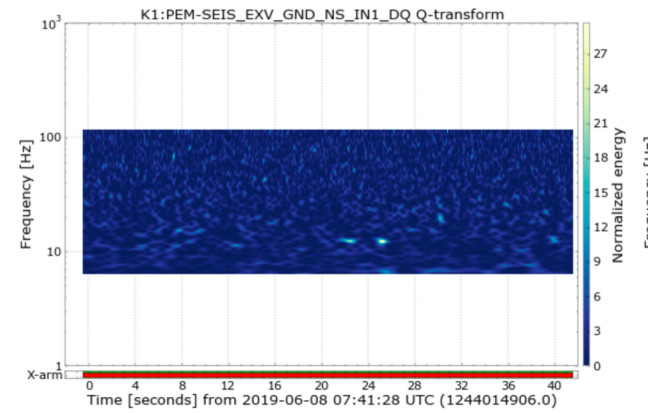
Whitened spectrogram around the glitch. The center is the triggered time.

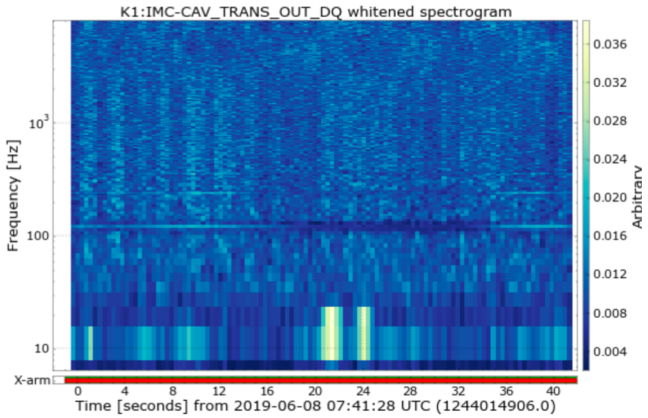


- Further below, you can see more detail in each related channels.
  - Unsafe channels
  - Important upstream channels
  - Relevant VIS channels
  - PEM channels

Not trigger channel (K1:LSC-CARM_SERVO_MIXER_DAQ_OUT_DQ) at GPS=1244014926.5 JST=2019 06 08 16:41:49.5



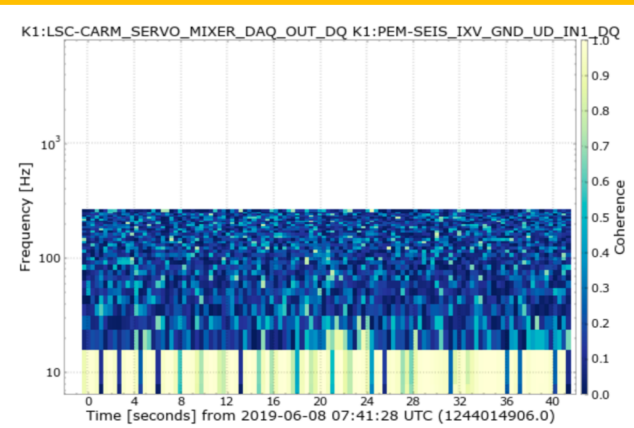


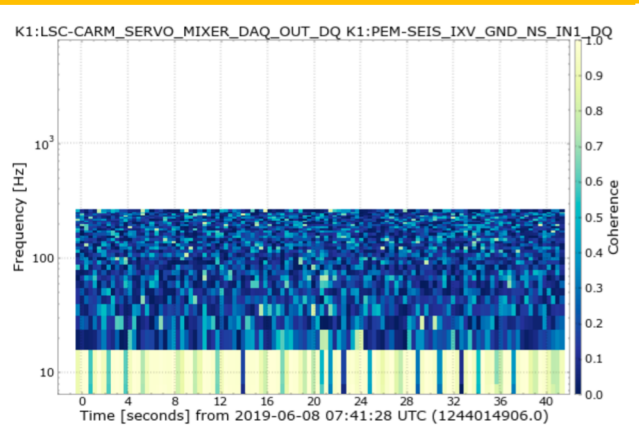


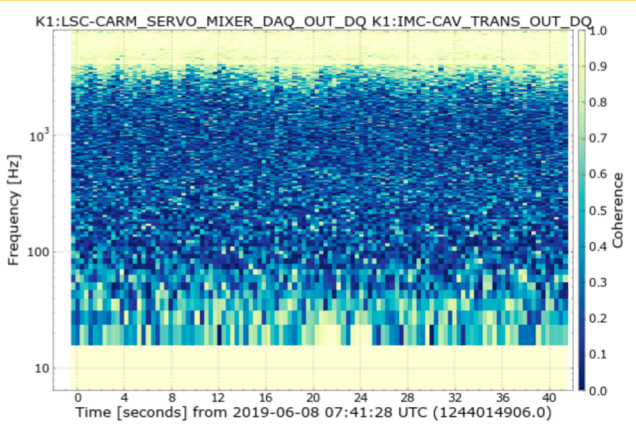
... and more.

- Coherencegrams with other channels are shown.
  - The title is $triggerchannel $relatedchannel

Coherence of K1:LSC-CARM_SERVO_MIXER_DAQ_OUT_DQ at GPS=1244014926.5 JST=2019 06 08 16:41:49.5

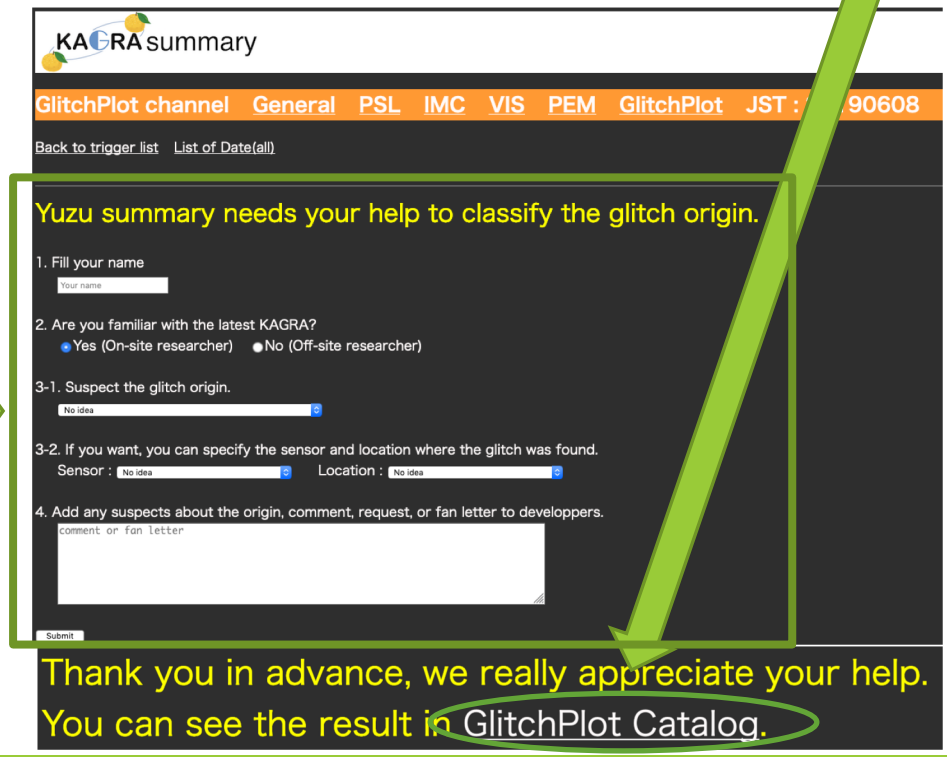






- After checking plots, let's go back to the top of the page.
- Collabolators can vote their opinion for each glitch.
- Collected information is accessible from here.

Voting form



- Collected information is summarized in a google spread sheet.
- The data can be downloaded as CSV format.

GlitchPlot Catalog

ファイル 編集 表示 挿入 表示形式 データ ツール アドオン ヘルプ

共有

100% 閲覧のみ

fx # submit date

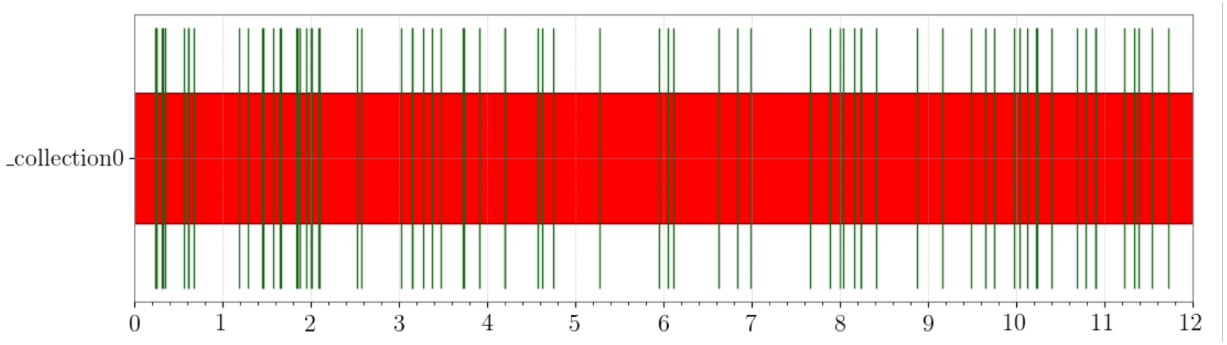
	A	B	C	D	E	F	G	H
1	# submit date	# submitter name	# onsite or offsite	# trigger channel name	# glitch class	# comment	# GPS time	# JST time of glit
2			0 : submit from on-site 1 : submit from off-site		0 : Unknown 20 : PSL 40 : IMC 60 : Seismic 80 : Acoustic 100 : Trigger channel 120 : other reason 777 : GW signal			
3	2019/06/21 16:58:29	yuzurihara	1	K1:PEM_ACC_MCF_TABLE_REFL_Z_OUT_DQ	777	check check check	1245008418	2019 06 20 4
4	2019/06/21 17:24:39	Tomotada Akutsu	0	K1:PEM_ACC_MCF_TABLE_REFL_Z_OUT_DQ	120	It does not seem the MCo oplev is properly wind-shielded.	1245008418	2019 06 20 4
5	2019/06/22 0:29:49	KNagano	1	K1:PSL_PMC_TRANS_DC_OUT_DQ	80	Not so sure. Not so sure. Might the PMC lockloss affect the other PSL instruments the other way around, e.g. via power supply? Similar glitch happened at 1244976288.0.	1245008474	2019 06 20 4
6	2019/06/24 16:32:39	Chihiro Kozakai	1	K1:PSL-PMC_TRANS_DC_OUT_DQ	60	PEM-ACC_PSL_TABLE_PSL3 is suspicious.	1245134540	2019 06 21 15
7								

Method overview

- Event trigger is given by external tool.
  - **Omicron** (Q-transform based general trigger)
    - Used for daily data taking.
    - Currently, result of night data taking is automatically uploaded to the webpage in ~1 day delay.
  - **Burst search trigger (cWB), CBC pipeline**
    - Work ongoing. Test version is available for ER2.
- Plot parameter is determined according to the trigger information.
- Plots are summarized in Yuzu summary page.
  - Daily update

Event counting method (Omicron)

- Omicron searches for large SNR time/band.
  - https://tds.virgo-gw.eu/?content=3&r=11553
  - https://tds.virgo-gw.eu/?content=3&r=14693
- Apply pre-cut by SNR and lock flag for the omicron trigger events.
- Make a segment list by duration and start time of the trigger.



- 1 segment is counted as 1 event.
  - If 2 or more trigger times are overlapped or continuous, they are merged to 1 event.
  - Frequency difference is ignored.

Parameter setting for plots

- Time series
  - Full data
  - Time span is roughly larger one of
    - 4 sec.
    - Trigger duration * 10
    - (Adjusted to fit the spectrogram requirement)
  - Starting time of the glitch segment is the center of the plot.
  - If time span is too long, down sample is applied.

Parameter setting for plots

- Spectrum
  - Spectrum of before and after trigger time is given.
    - To check if the channel state is affected by anything.
  - 2 sec. margin is taken around the trigger.
  - According to the band width of the trigger, enough fine frequency resolution is required.
    - FFT length ~ 1 / Band width (Select the nearest larger 2^n sec.)
  - To get high resolution in amplitude, ~30 sec long data is used
    - Actual length is adjusted to be the multiple of the FFT length.

Parameter setting for plots

- Spectrogram, coherencegram
  - Time bin stride is determined to have enough time resolution to see the glitch.

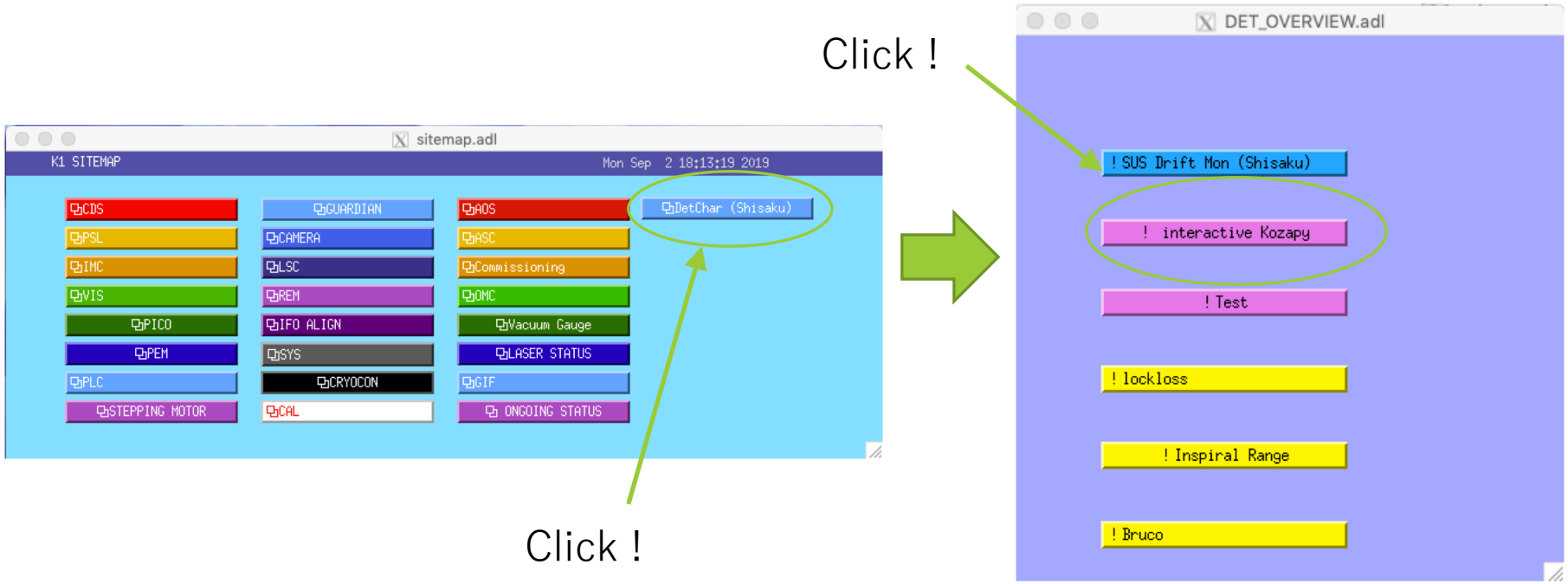
    stride < trigger length
    - Trade-off to frequency resolution.
    - If frequency resolution is higher than trigger frequency,

    stride = 1/trigger frequency

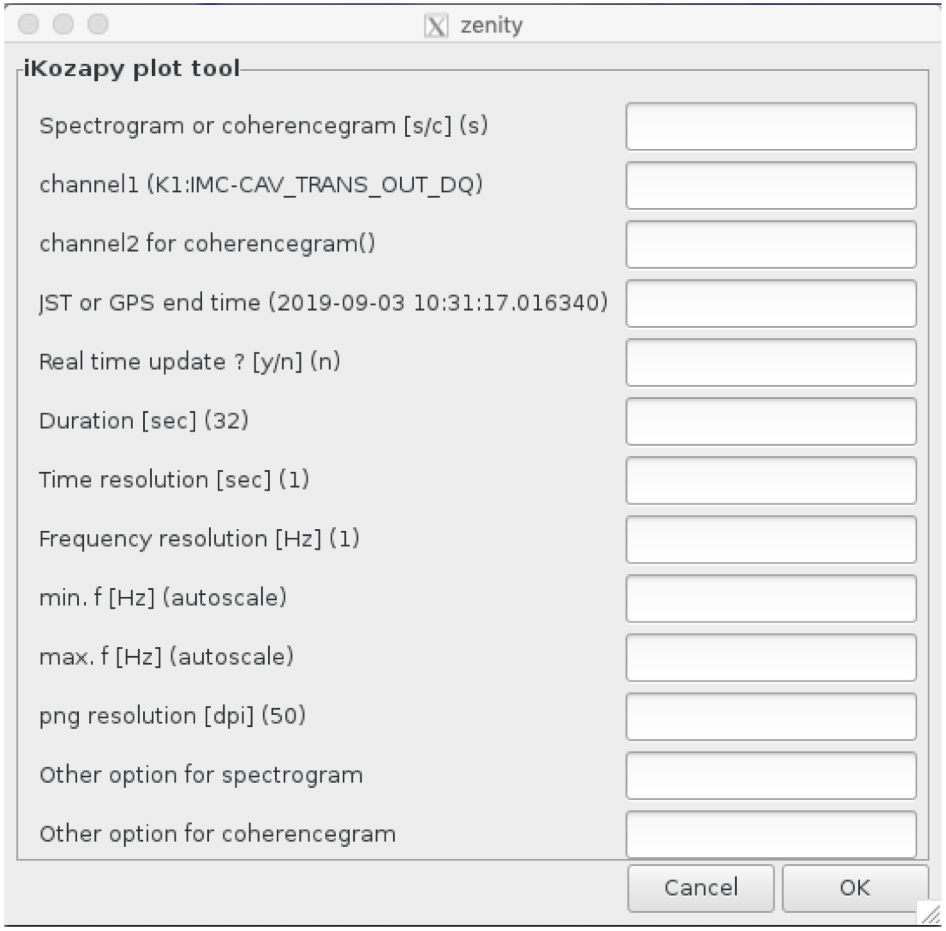
  - FFT length is determined to be less than stride.
    - stride = FFT length * n
  - Total data length is adjusted to be n * stride around the time span given for time series.
  - For spectrogram, whitened one is also given.

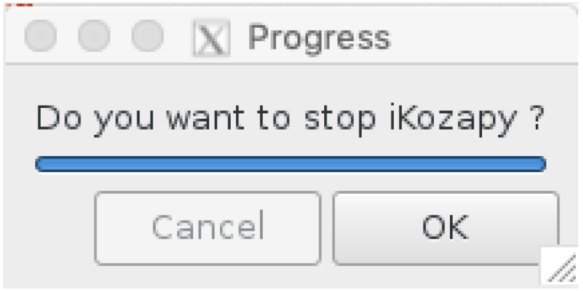
iKozapy (interactive Kozapy)

- GUI tool to make spectrogram or coherencegram. It is available on Kamioka PCs.



- You will get 2 windows. You can start from bigger window.





- If not filled default value is applied (value in ()).

s for spectrogram, c for coherencegram

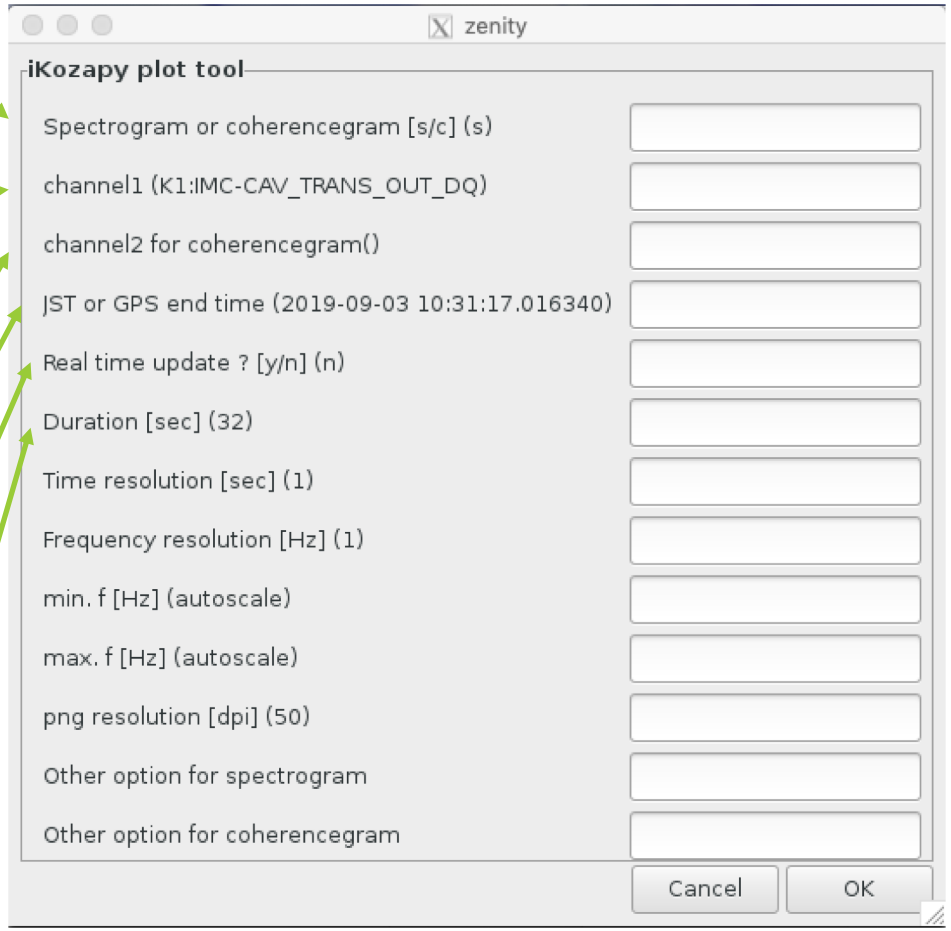
Channel name

2nd Channel name for coherencegram

END time (GPS or JST)

See p7

Data length.



- If not filled, default value is applied (value in ()).

Time resolution is time bin width

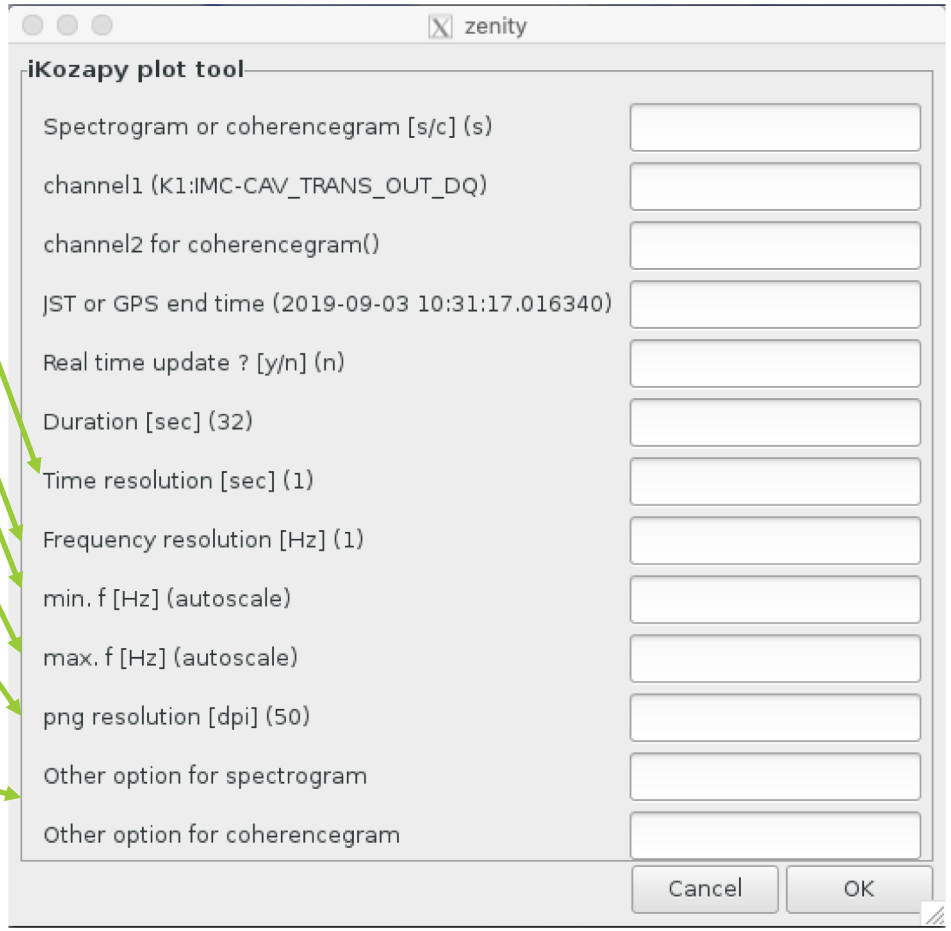
Frequency resolution
(* dt > 1/df is required)

Minimum frequency of the plot

Minimum frequency of the plot

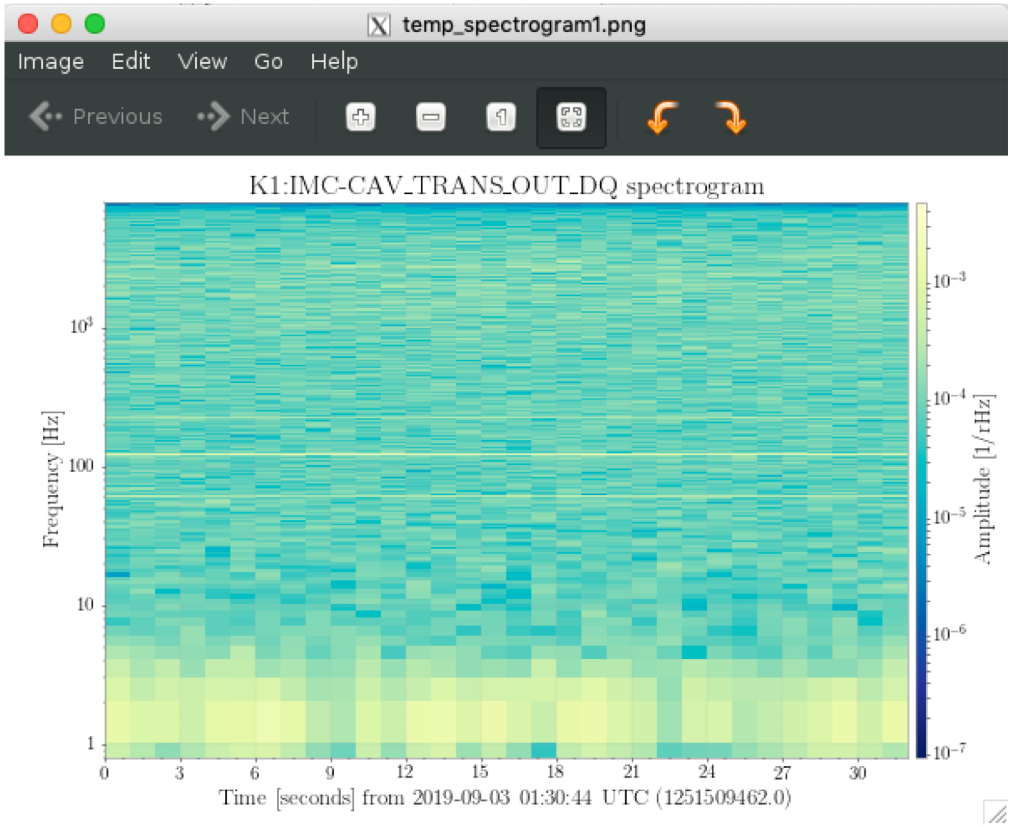
PNG resolution, dot per inch

You can add any option available in kozapy code.
e.g.) -w will apply whitening.



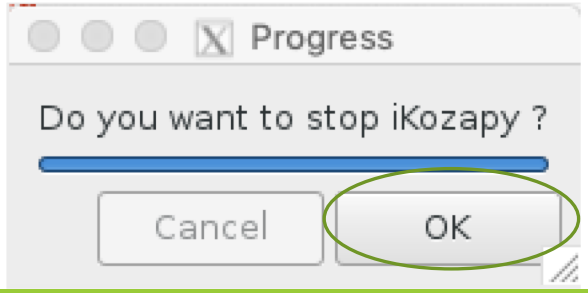
- Click “OK” and you will get the plot in ~10 sec (depends on parameter)

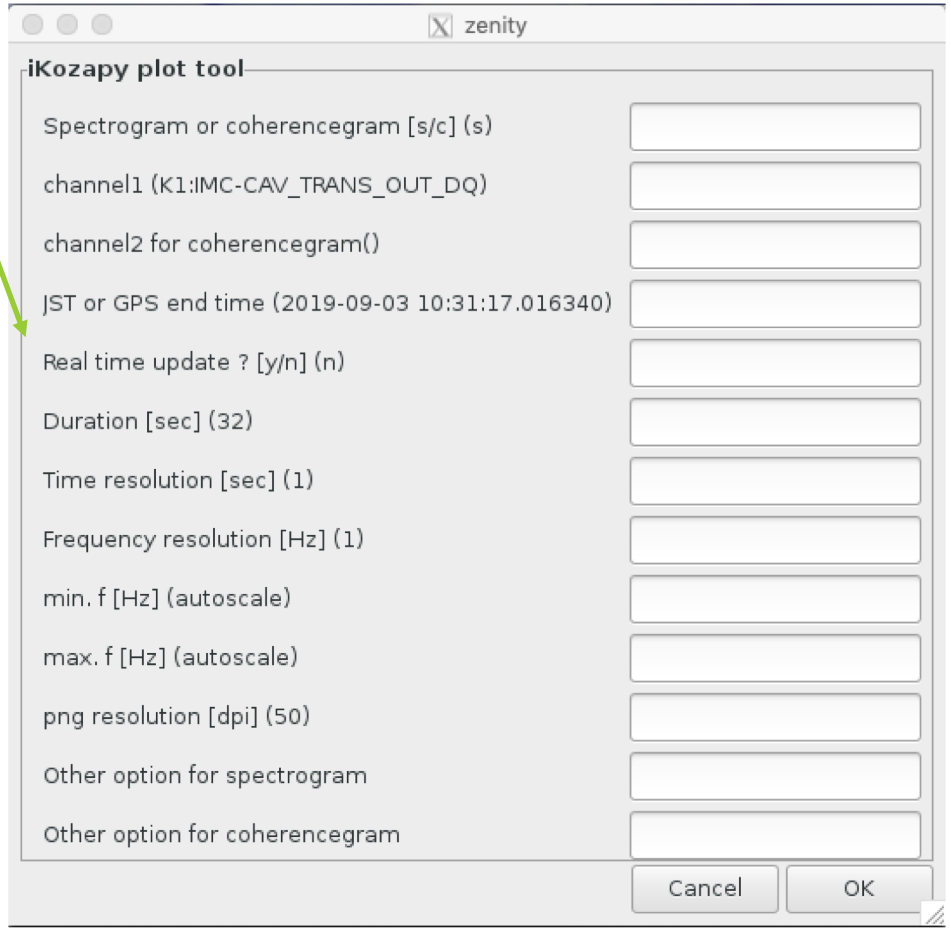
If you need to save the picture, please do it from “Image” tab. Or it will vanish later.



Real time update

- If you set y, the spectrogram will be updated automatically using the most recent data with interval of ~10 sec (depends on parameter).
- When you want to stop it, please press the “OK” in the small window to break infinity loop.





Conclusion

- New KAGRA detchar tools are developed.
  - GlitchPlot
    - Visual inspection tool for glitch and lock loss.
  - iKozapy
    - GUI spectrogram and coherencegram plot tool.