


LIGO Livingston Fellows Program

Wyatt Phillips, April 16th 2026

Overview

- My work was varied
 - My main project was SGN-Cal development
 - I also did lots of miscellaneous (but related) things
 - Each section of this presentation will cover a different project

The background of the slide features a series of stylized, overlapping hills in various shades of green, ranging from light mint to dark forest green. The hills are rendered with soft, rounded edges and a slight gradient, creating a sense of depth and a clean, modern aesthetic. The overall composition is minimalist and serene.

SGN-Cal

The SGN-Cal Project

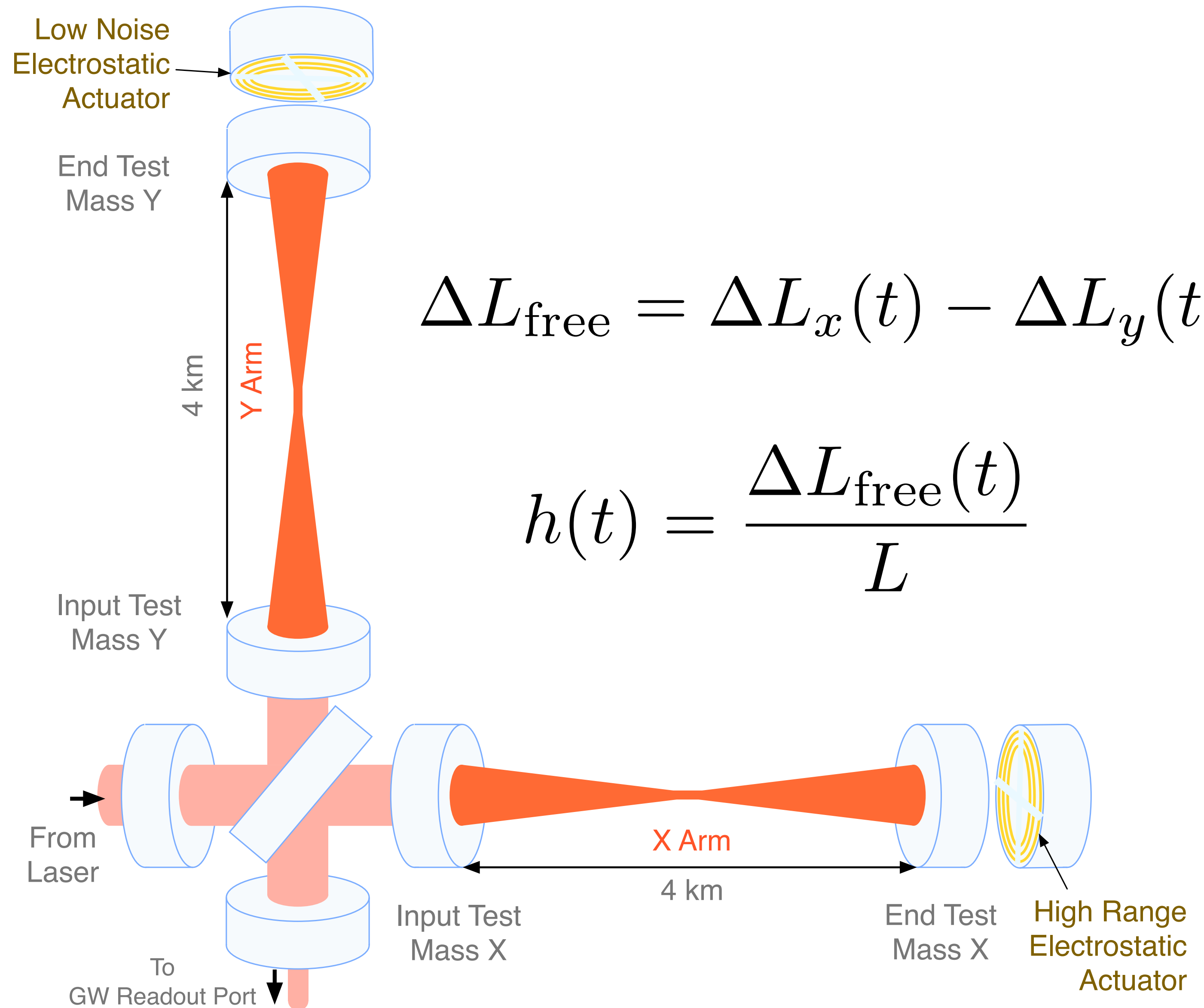
- Building the new low-latency calibration pipeline for LIGO
 - New version of the GstLAL calibration pipeline
 - Produces strain value

SGN-Cal Developers:

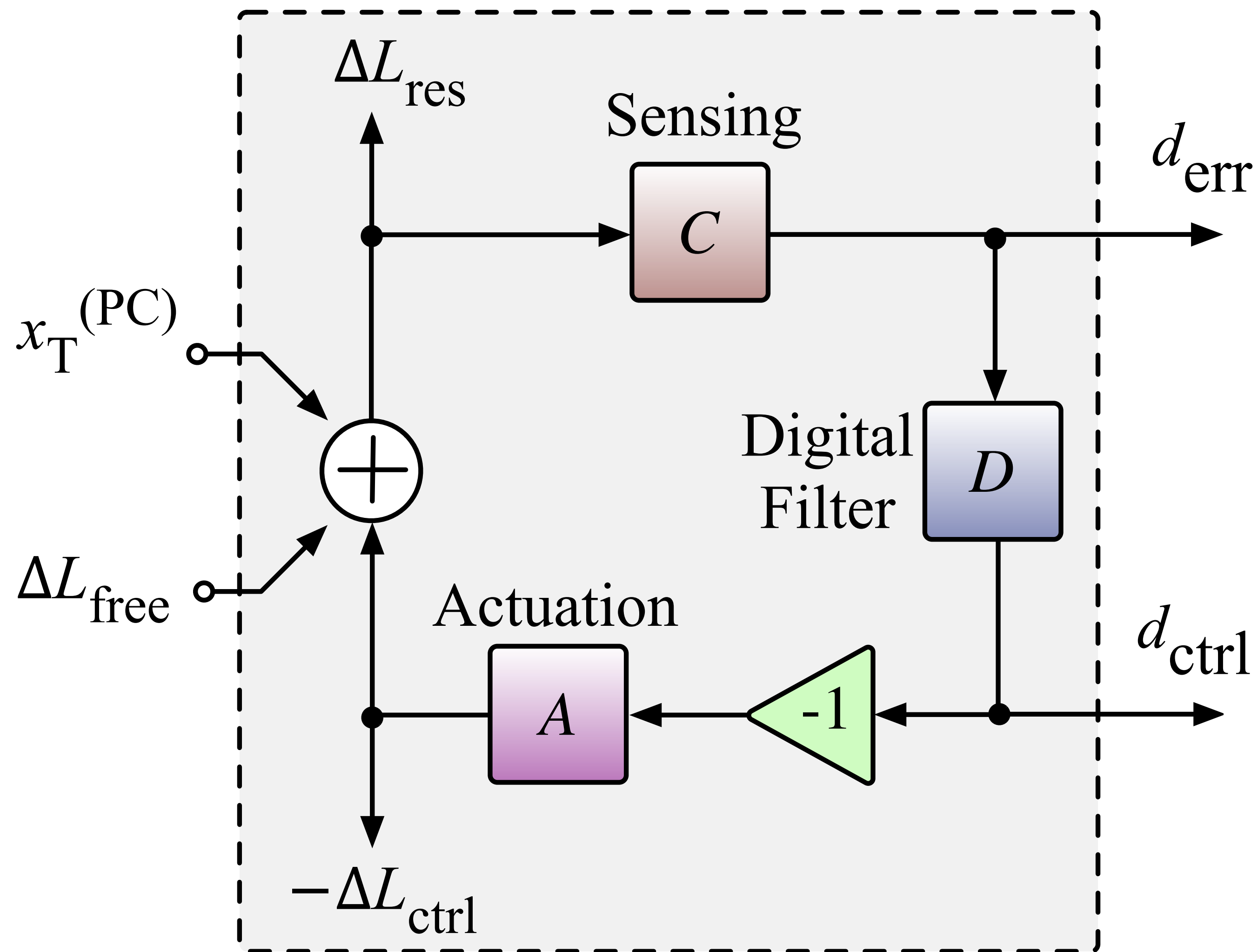
Eve Currens, Jane Glanzer, Olivia Godwin, Chad Hanna, Yun-Jing Huang, Jim Kennington, Max Melching, Avani Patel, Wyatt Phillips, Jameson Rollins, Nathanael Sovitzky, Aaron Viets, Maddie Wade

Differential arm motion

- DARM motion
 - Relative change in lengths of arms
 - Changes the amount of light on photodiode
 - “DARM Error” (d_{err}) signal
- GW affects round-trip phase of light, captured in DARM error
- How do we determine actual DARM motion (and therefore GW strain)? Calibration!

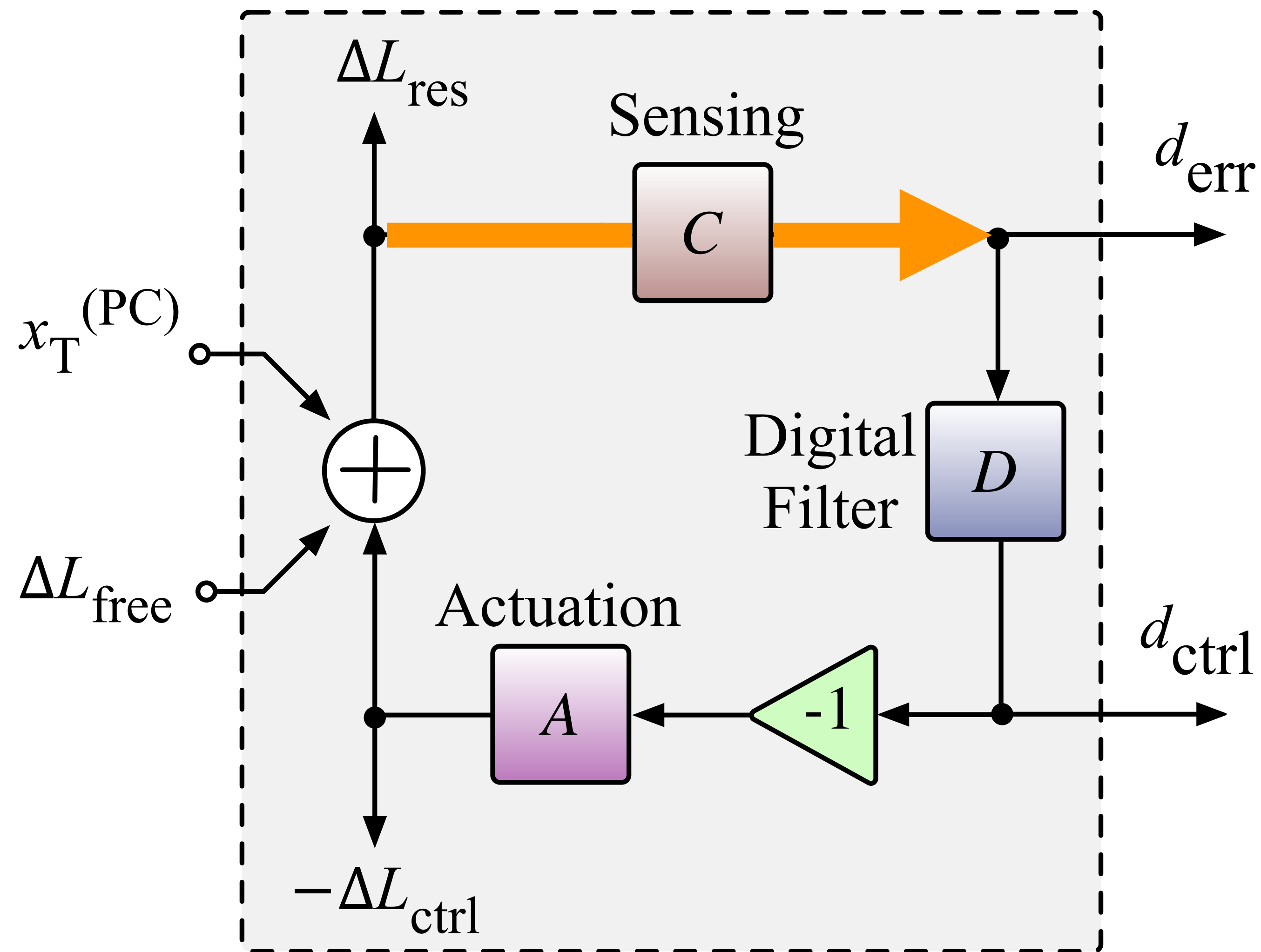


DARM Control Loop



- Purpose is to keep test mass at equilibrium position
 - AKA minimize d_{err}
 - Maintains lock
 - Constantly adjusting mirror position to minimize DARM error signal (photodiode output + offset)

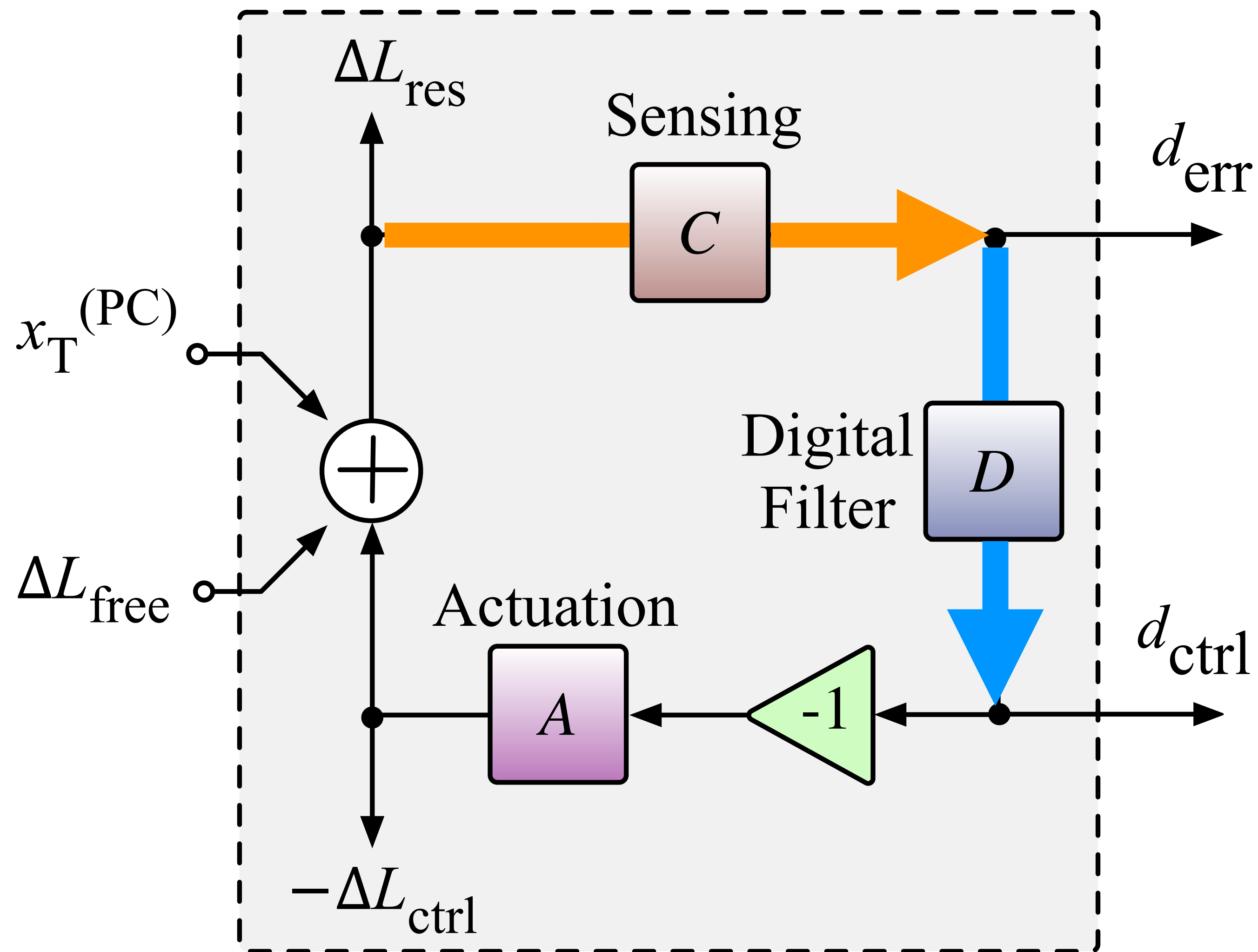
DARM Control Loop



Realtime interferometer control

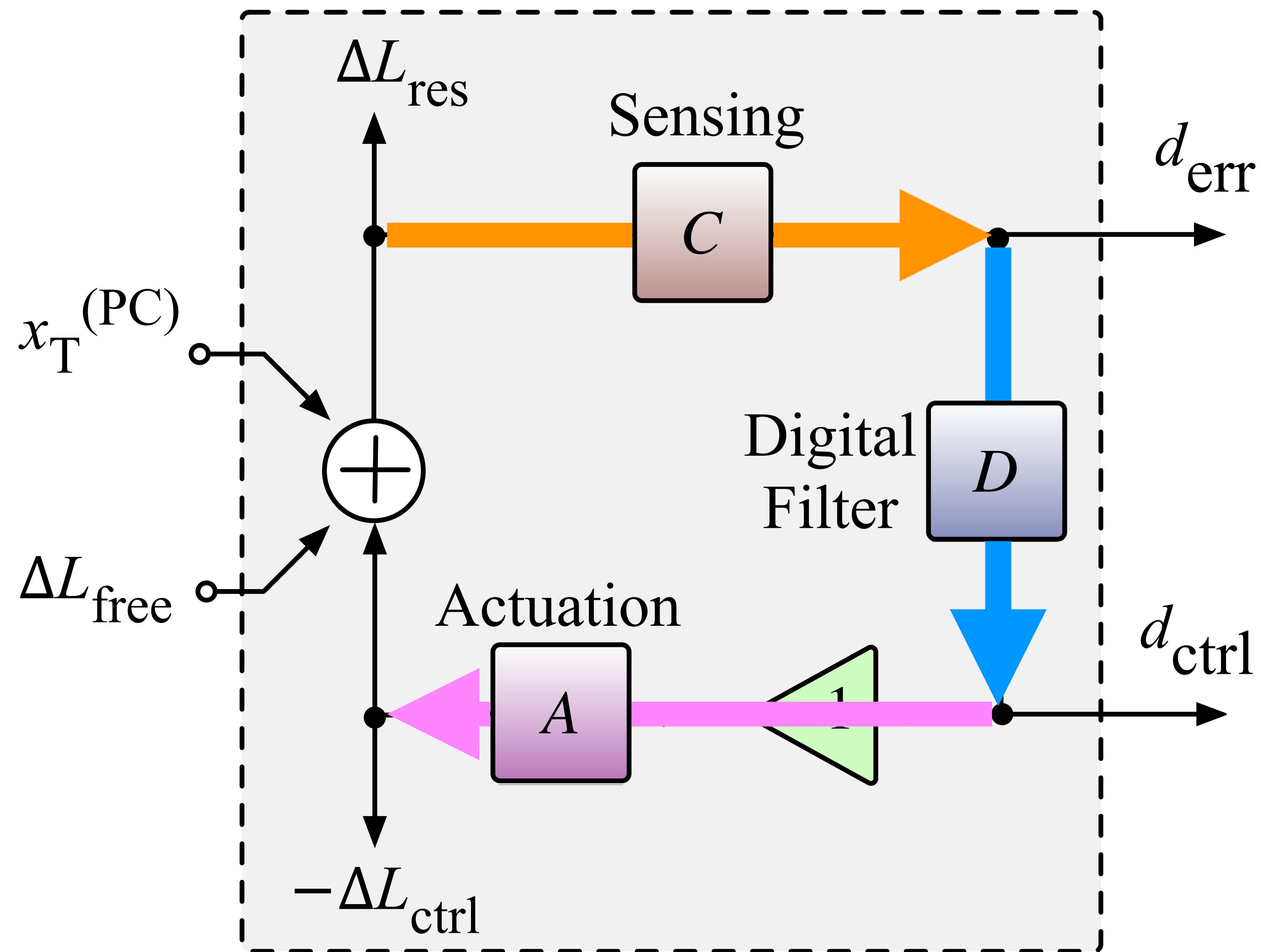
- Purpose is to keep test mass at equilibrium position
 - AKA minimize d_{err}
 - Maintains lock
 - Constantly adjusting mirror position to minimize DARM error signal (photodiode output + offset)

DARM Control Loop



- Purpose is to keep test mass at equilibrium position
 - AKA minimize d_{err}
 - Maintains lock
 - Constantly adjusting mirror position to minimize DARM error signal (photodiode output + offset)

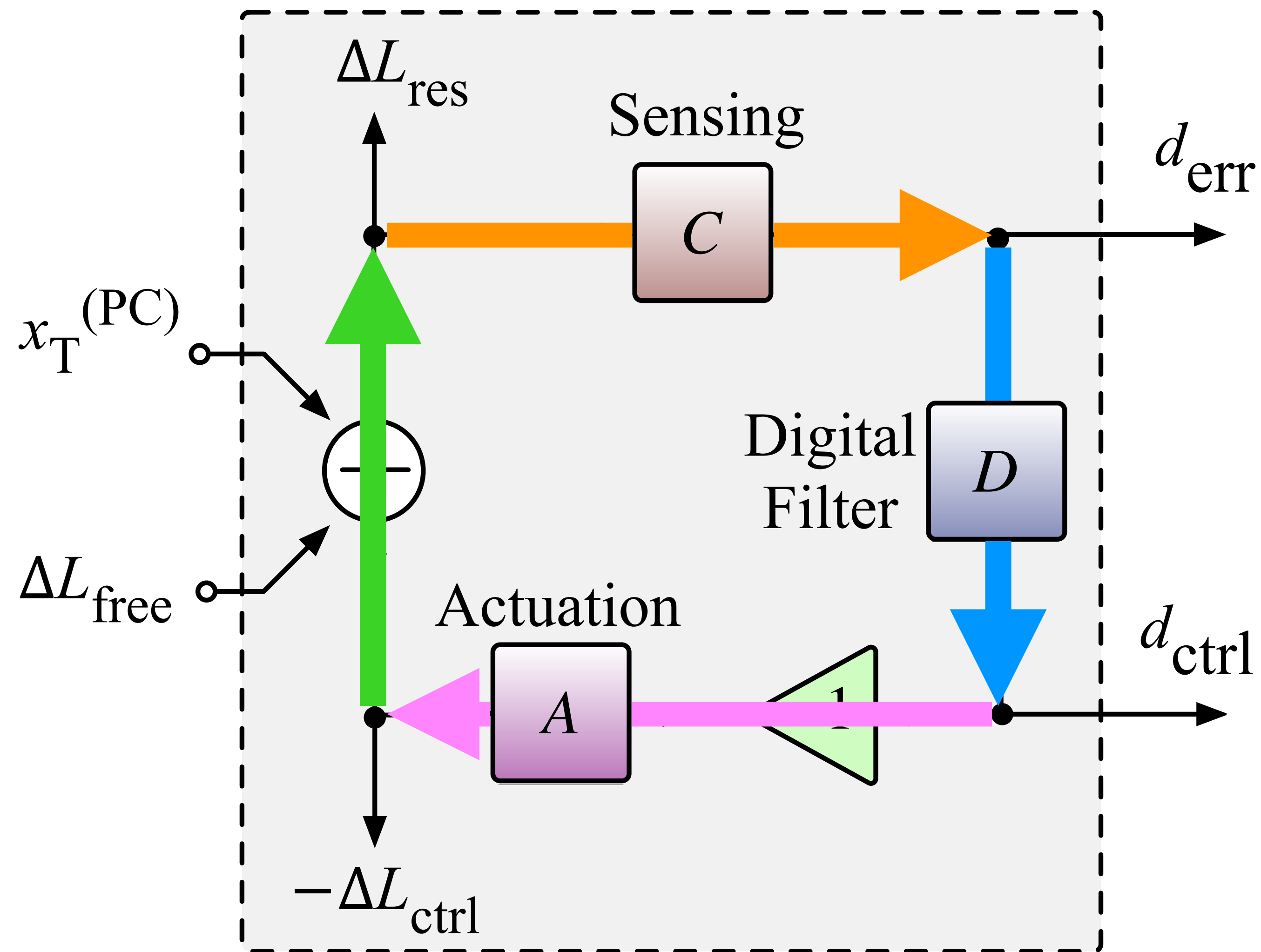
DARM Control Loop



Realtime interferometer control

- Purpose is to keep test mass at equilibrium position
 - AKA minimize d_{err}
 - Maintains lock
 - Constantly adjusting mirror position to minimize DARM error signal (photodiode output + offset)

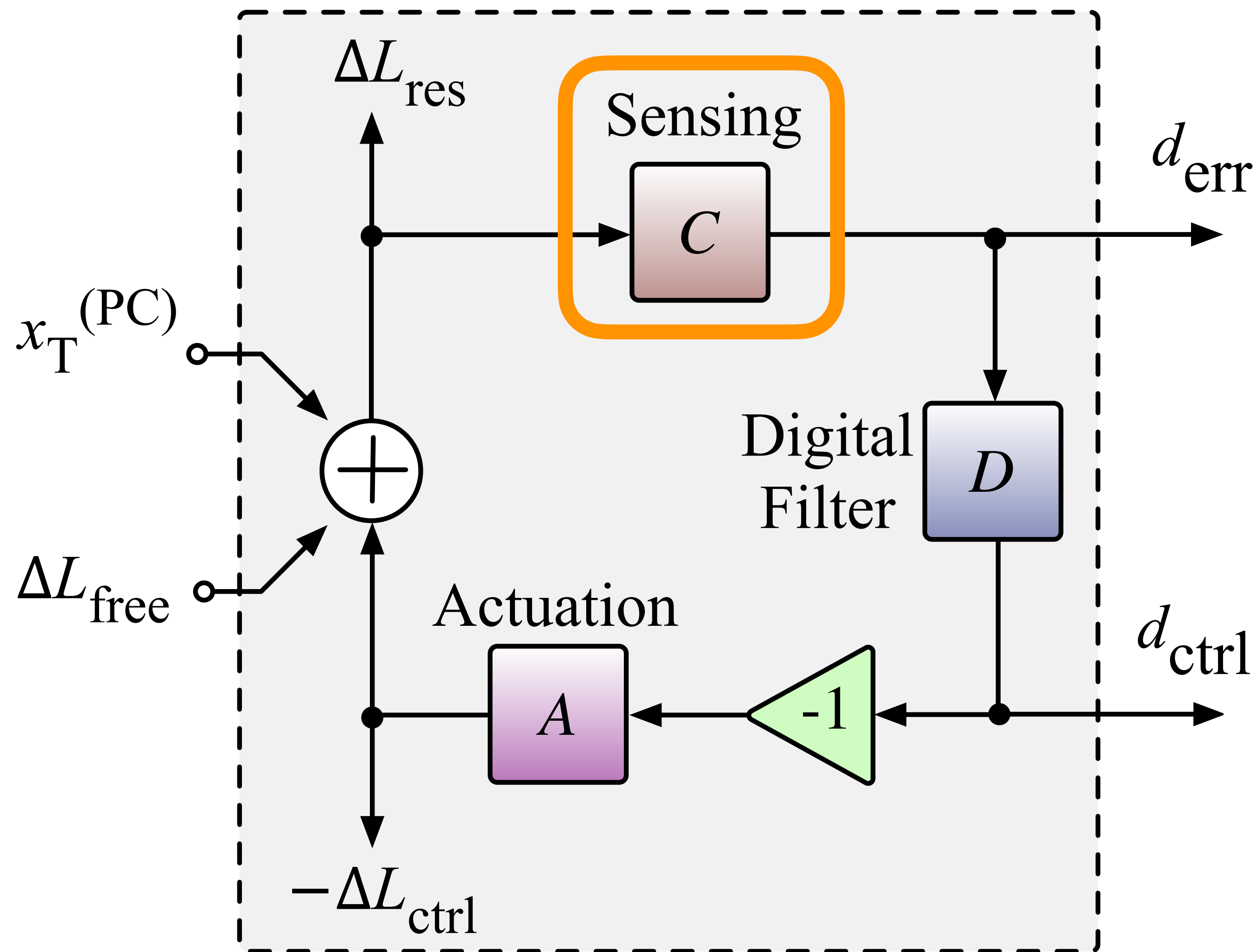
DARM Control Loop



Realtime interferometer control

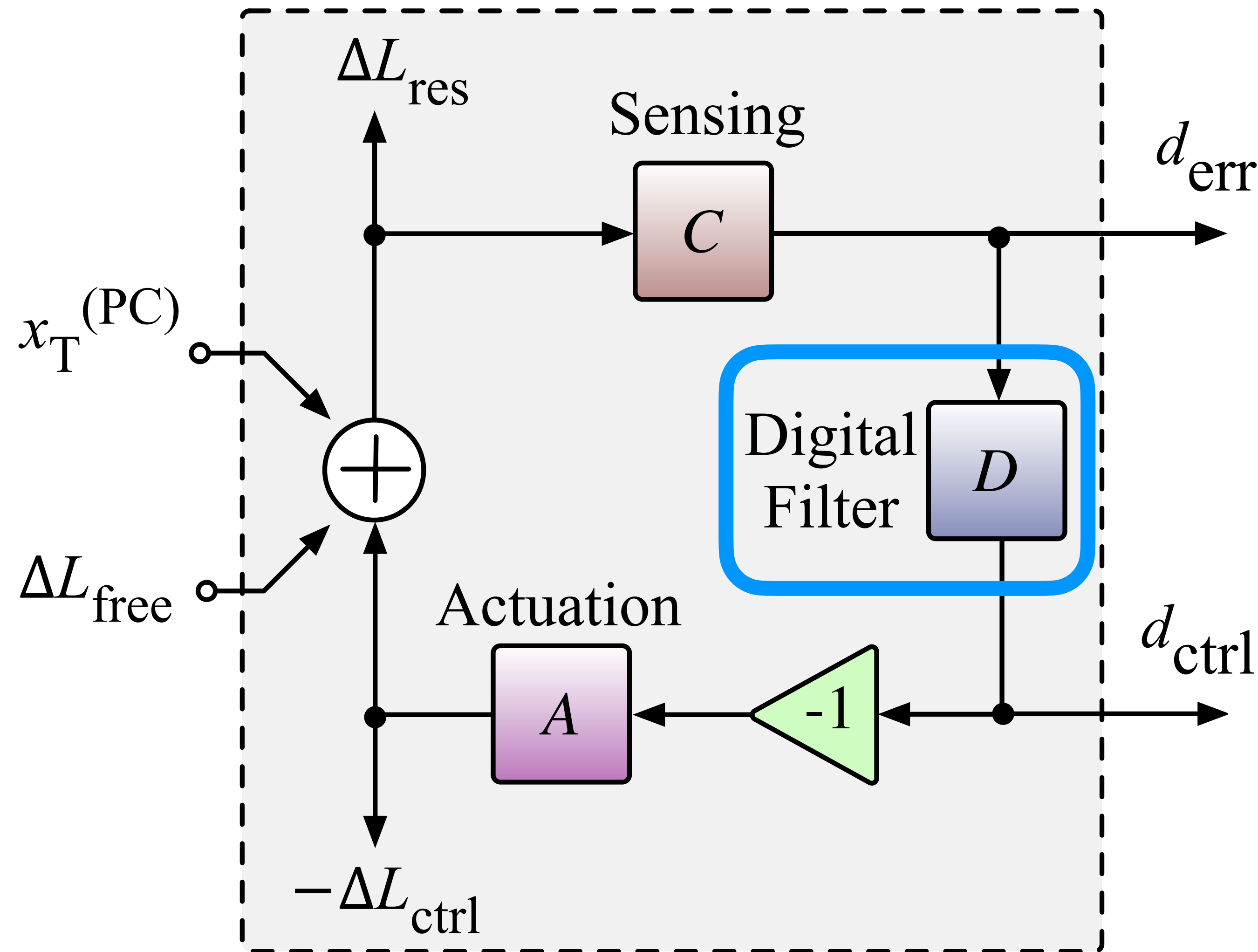
- Purpose is to keep test mass at equilibrium position
 - AKA minimize d_{err}
 - Maintains lock
 - Constantly adjusting mirror position to minimize DARM error signal (photodiode output + offset)

Sensing



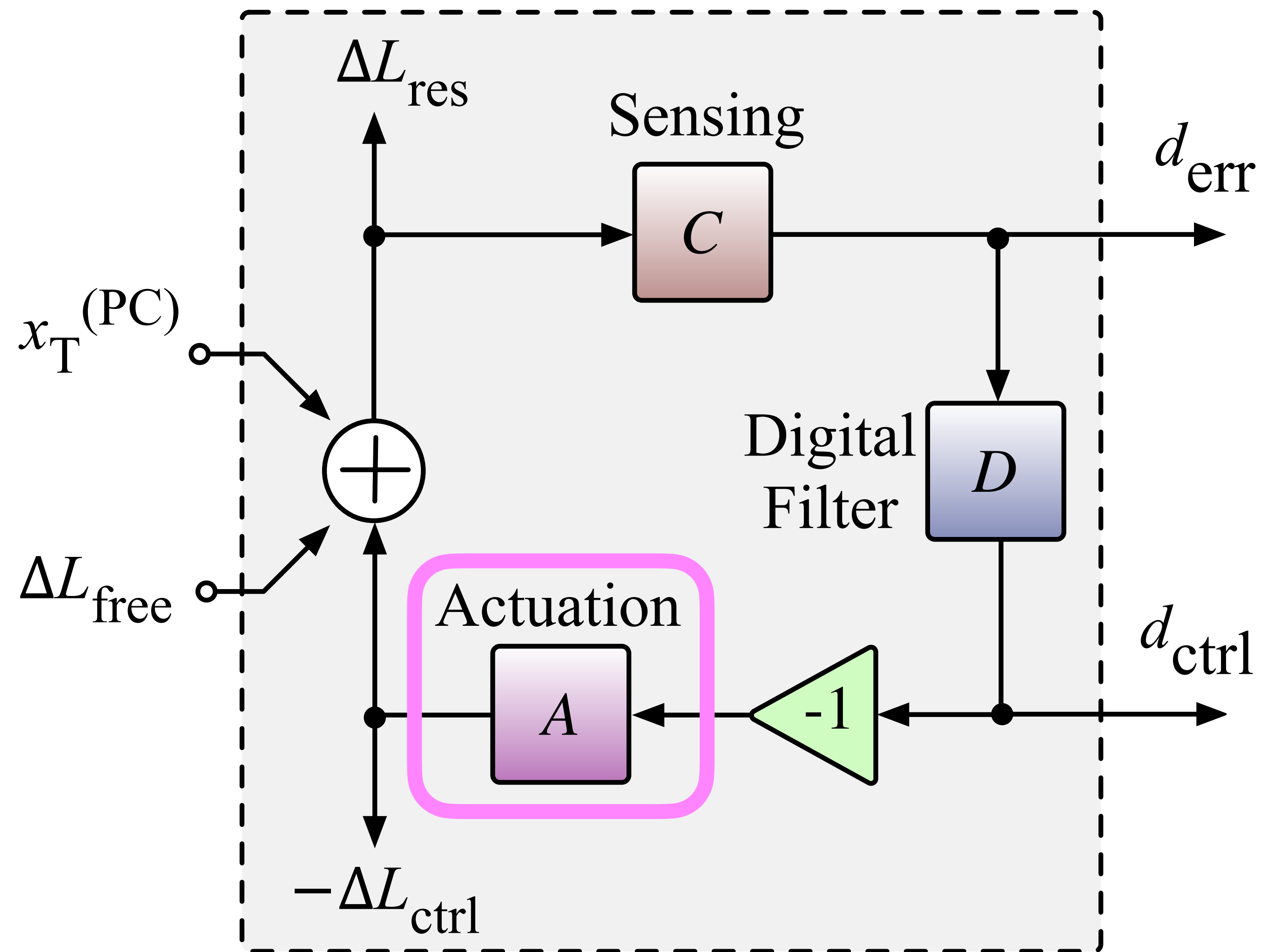
- Sensing function
 - Represents conversion between meters of mirror motion and d_{err}
- DARM error (d_{err})
 - Captures length change from noise sources as well as GWs
- Modeling the sensing function
 - Model for detector signal given some length change in arms
 - Dominated by Fabry-Perot cavity and signal recycling transfer functions

Digital Filter



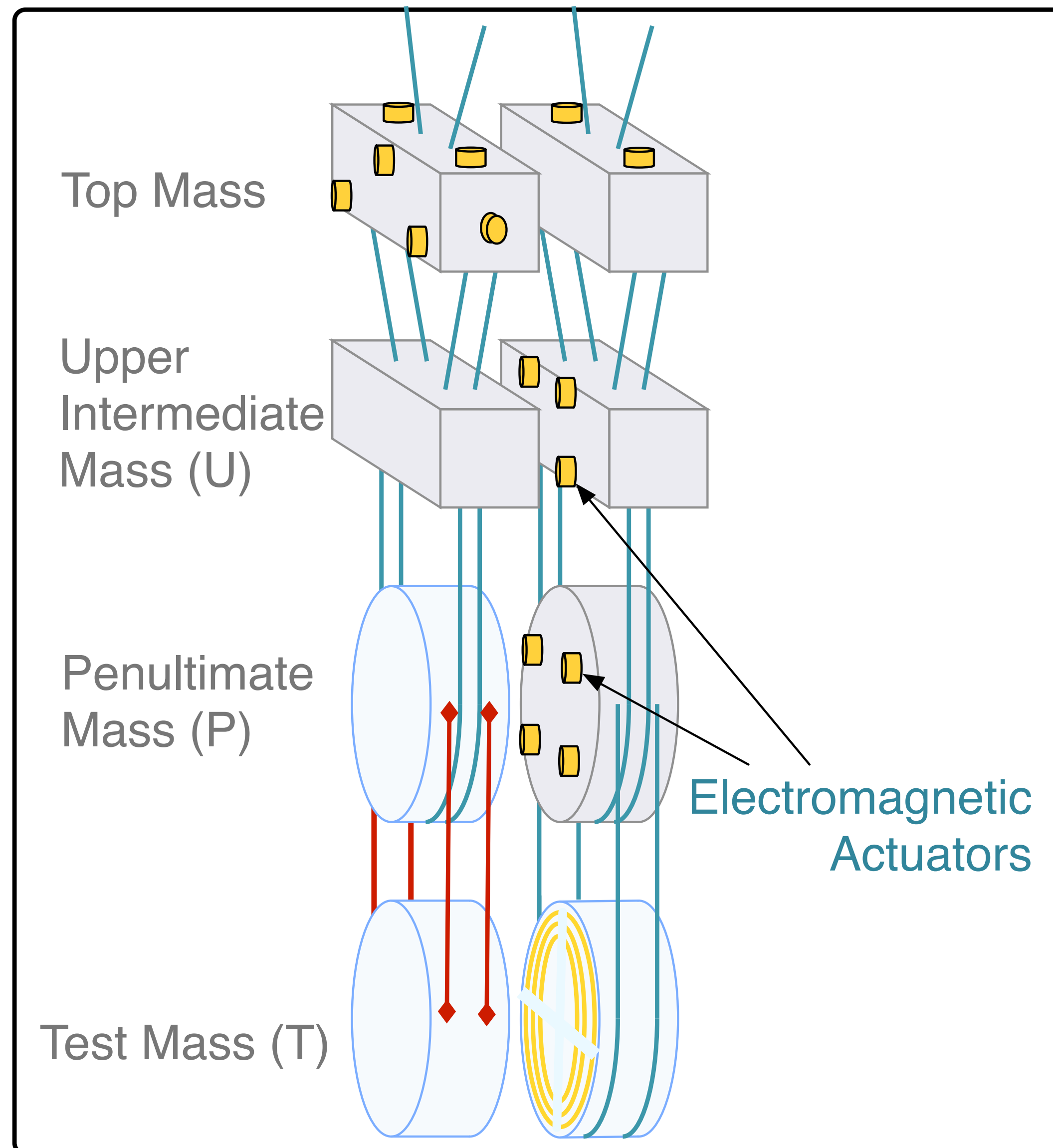
Realtime interferometer control

- Digital filter
 - Converts error signal (d_{err}) to control signal (d_{ctrl})
 - d_{ctrl} contains the changes we want to make to mirror position
 - Shapes the control loop so it is a stable control system
- Don't need to model, known exactly



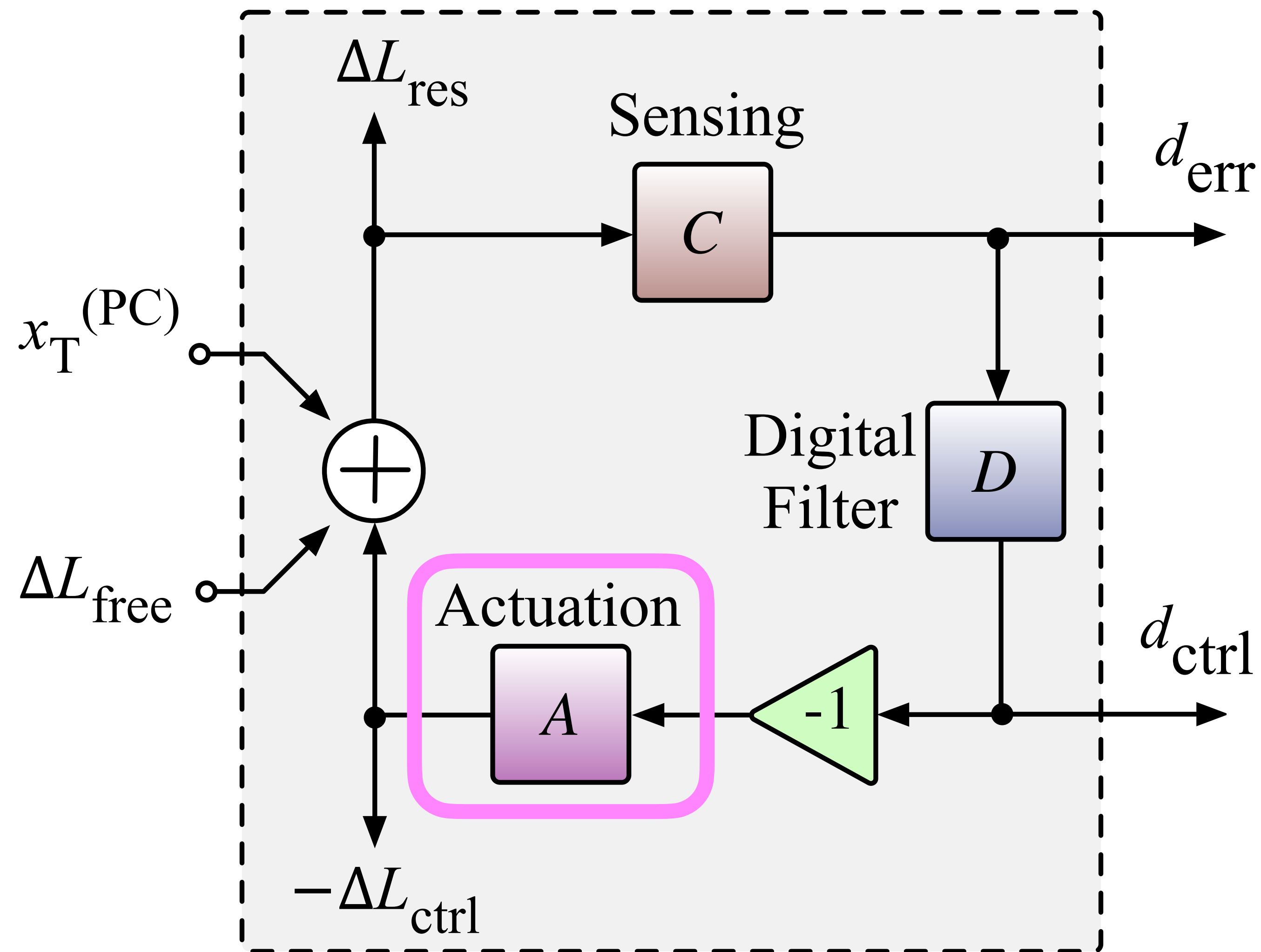
Actuation

- Actuators
 - Actuators part of the double quad pendulum, push masses
- Actuation function
 - Relates applied force to motion on masses
- Actuation function model
 - Quadruple pendulum, FR



Actuation

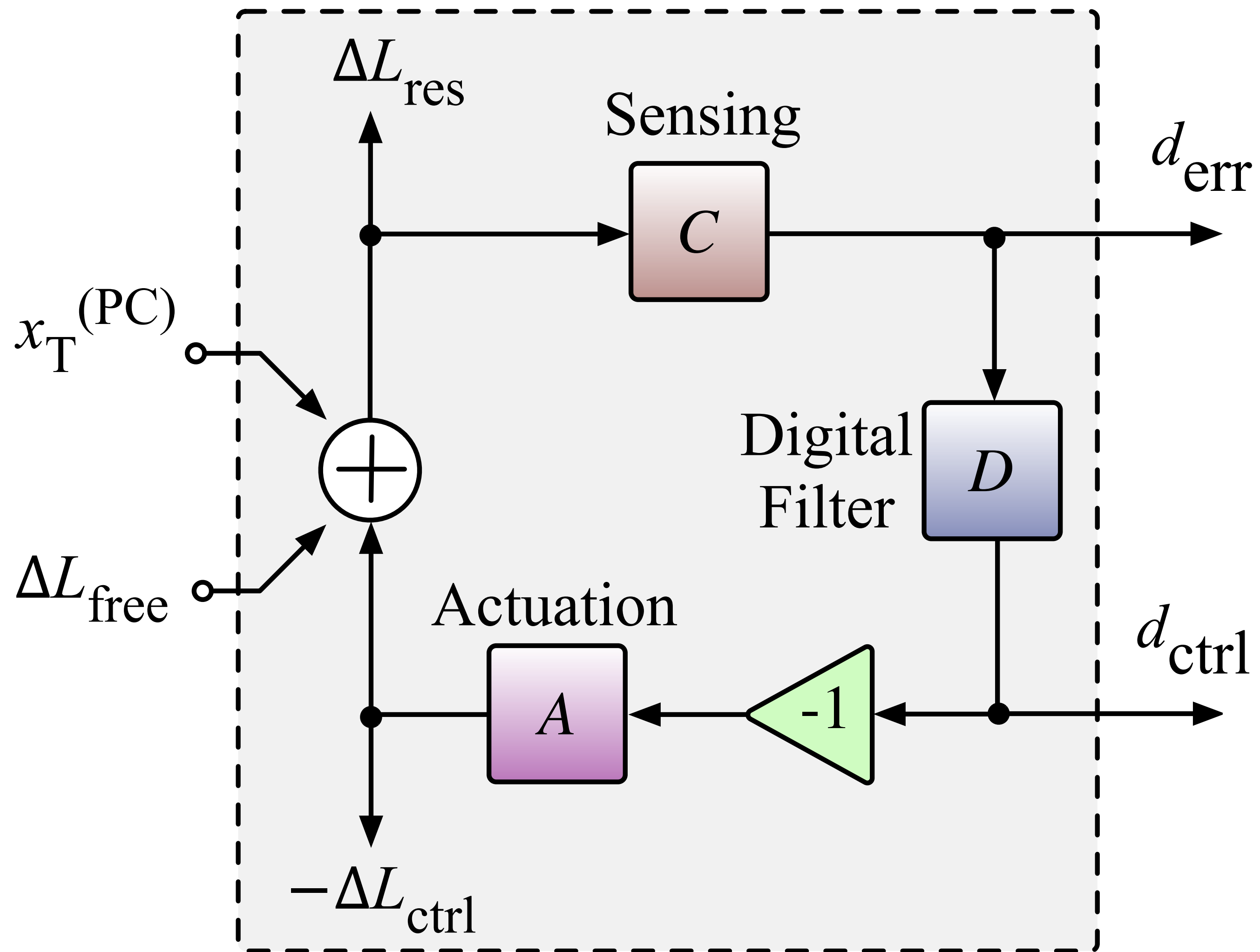
- Actuators
 - Actuators part of the double quad pendulum, push masses
- Actuation function
 - Relates applied force to motion on masses
- Actuation function model
 - Quadruple pendulum, FR



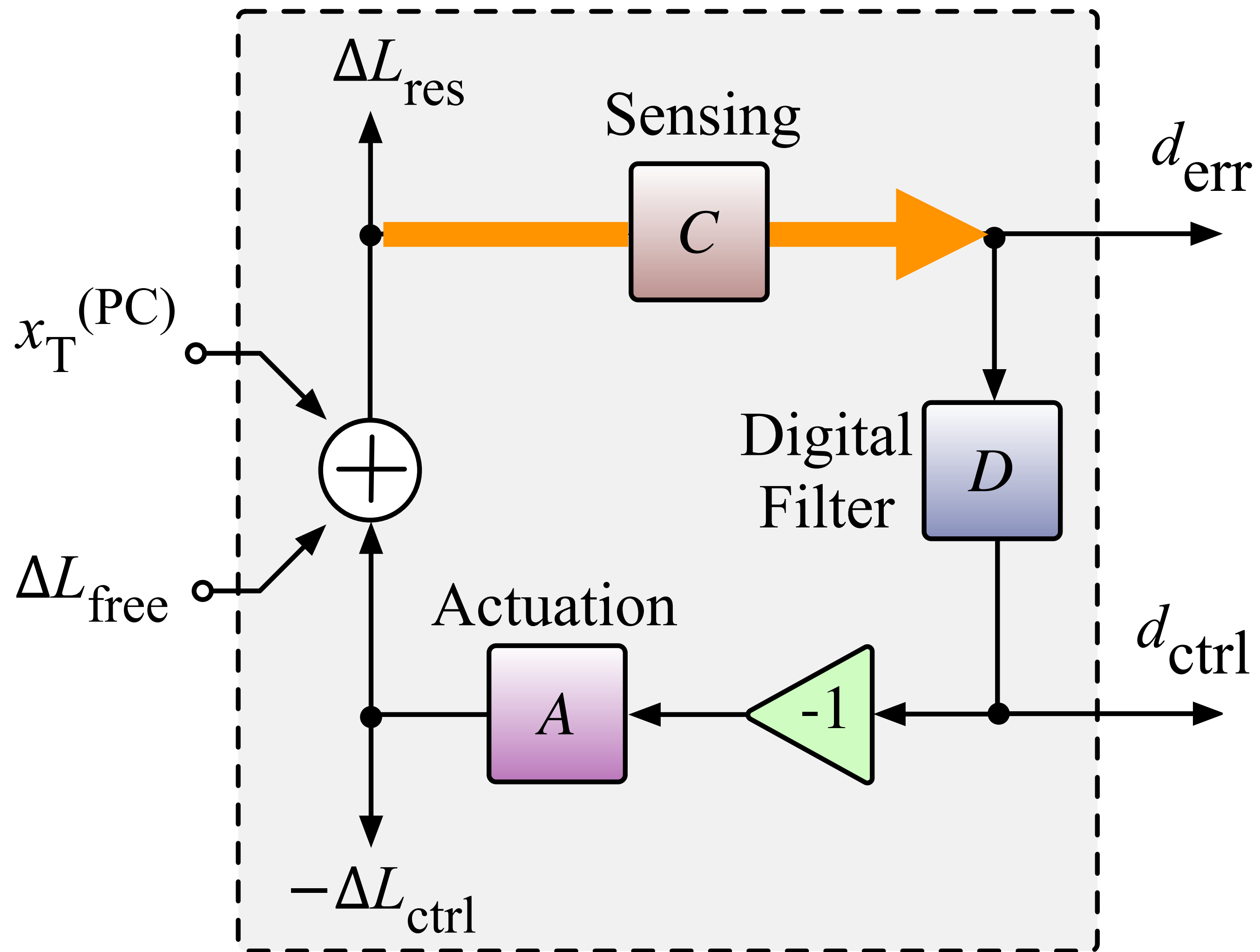
Realtime interferometer control

Actuation

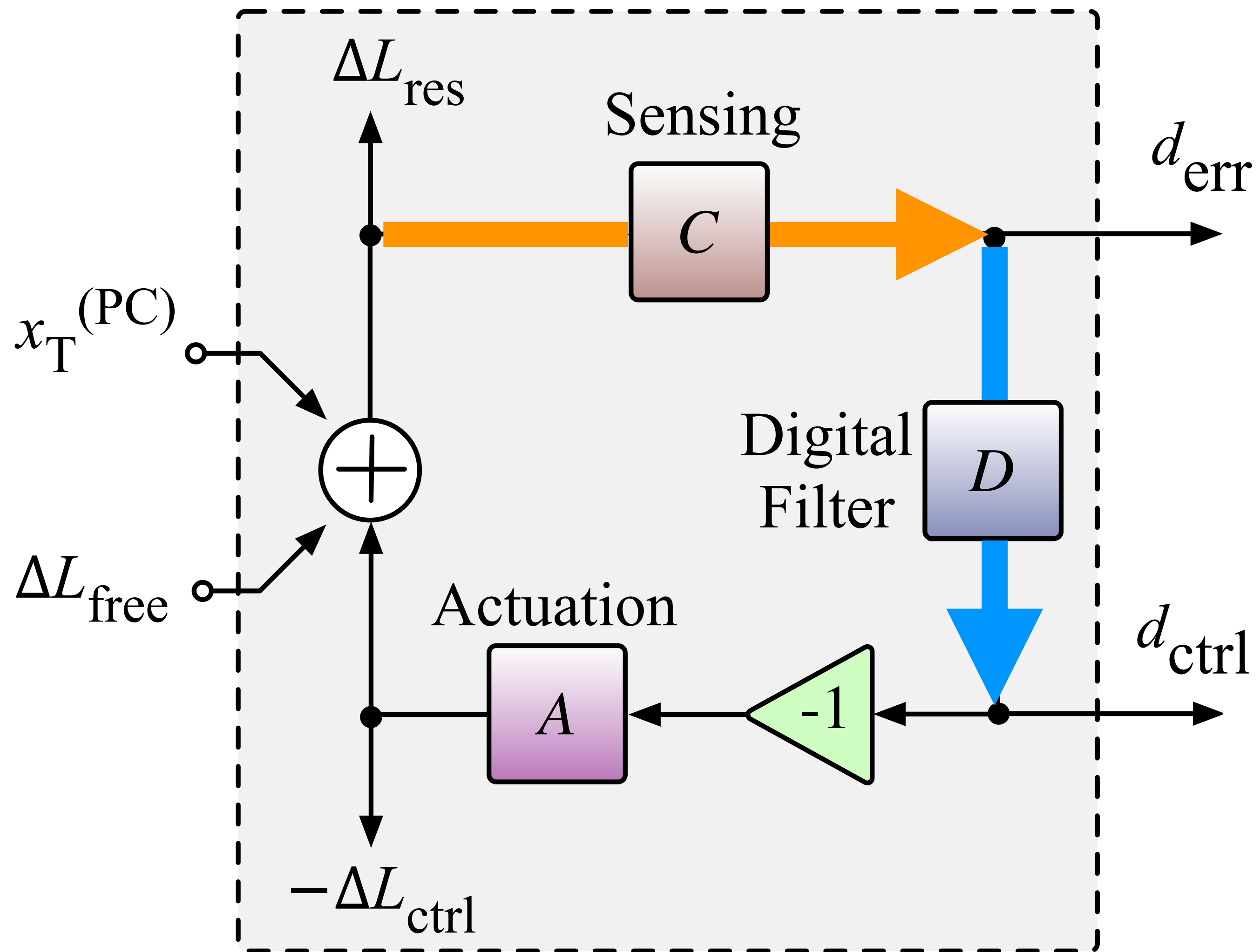
- Actuators
 - Actuators part of the double quad pendulum, push masses
- Actuation function
 - Relates applied force to motion on masses
- Actuation function model
 - Quadruple pendulum, FR



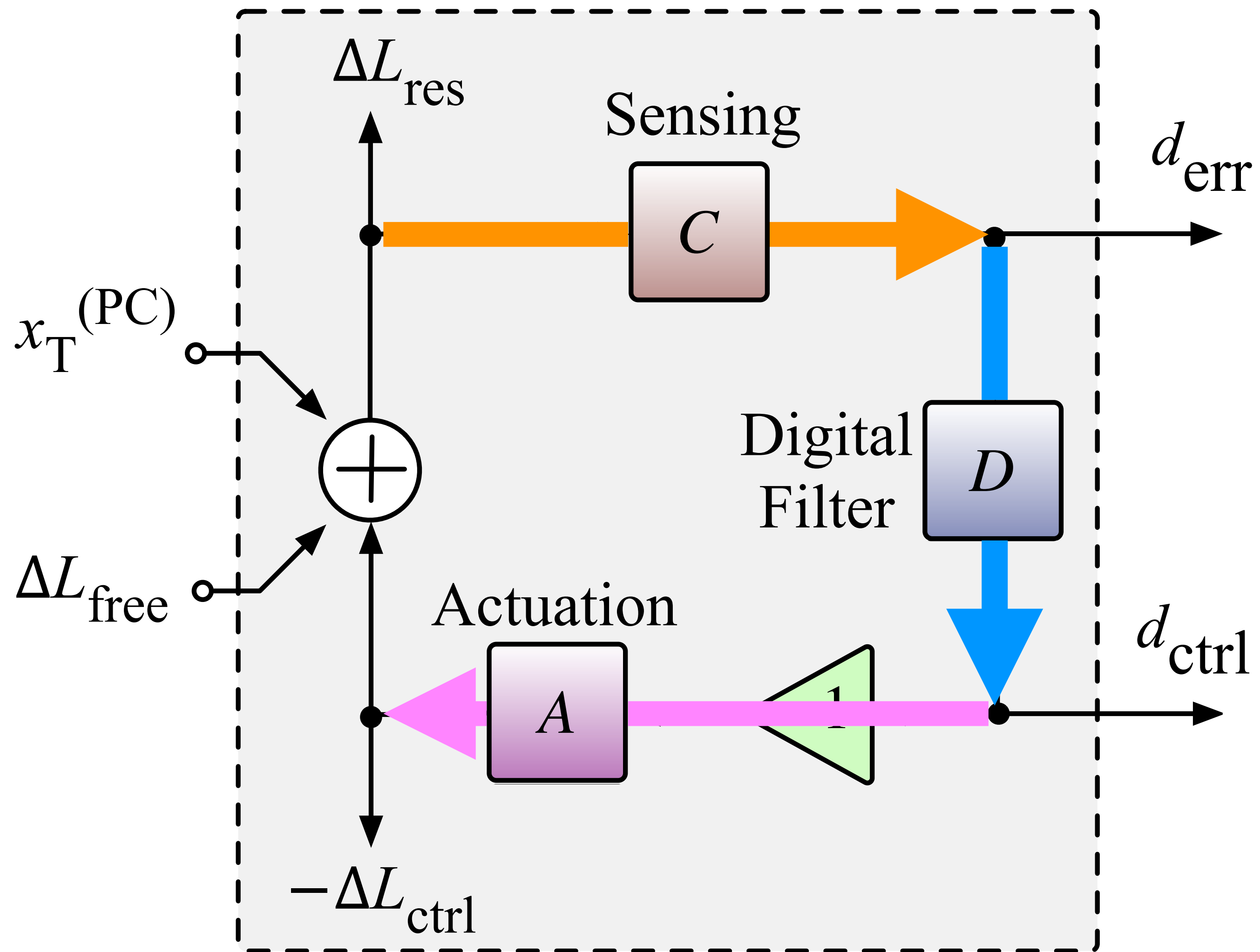
Realtime interferometer control



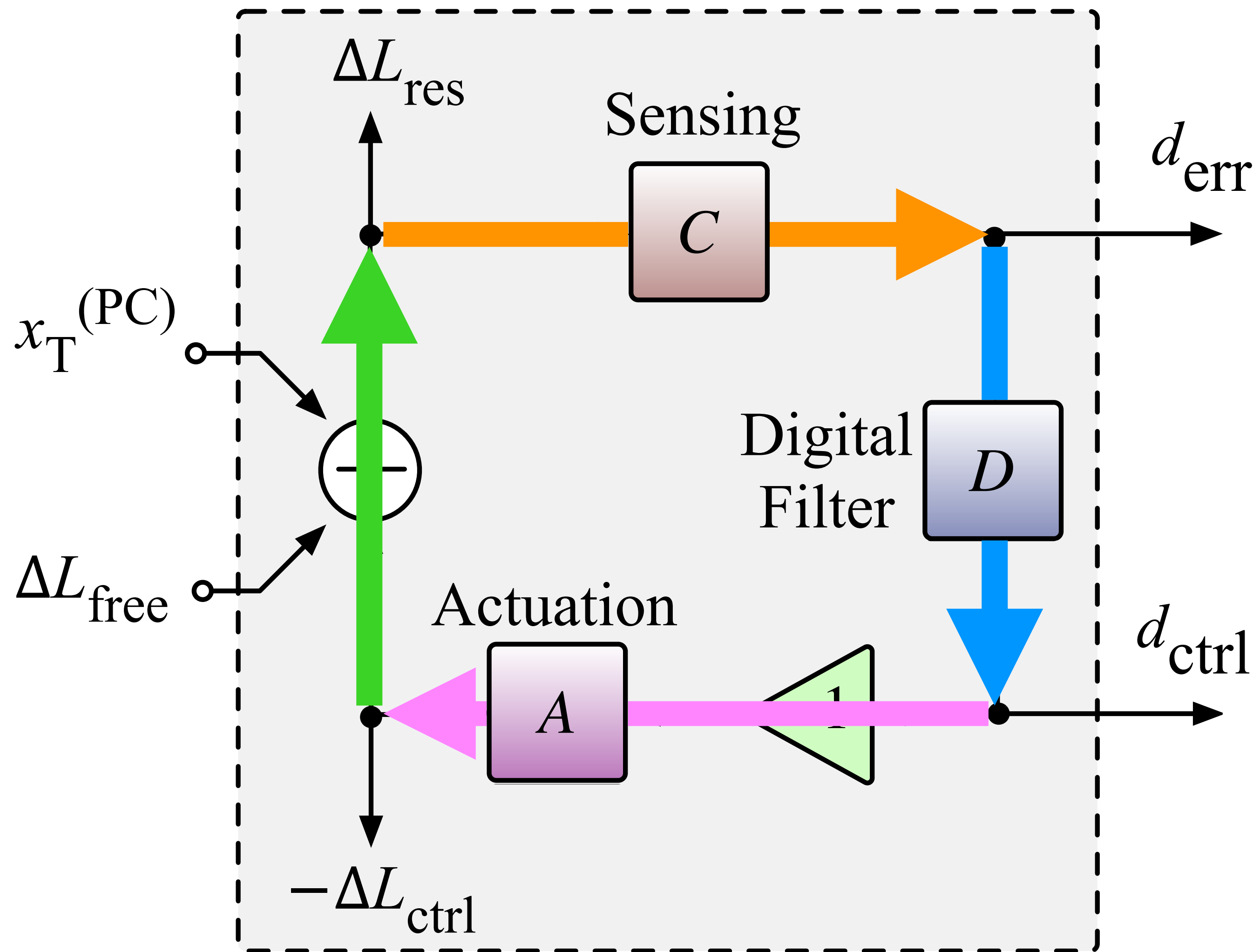
Realtime interferometer control



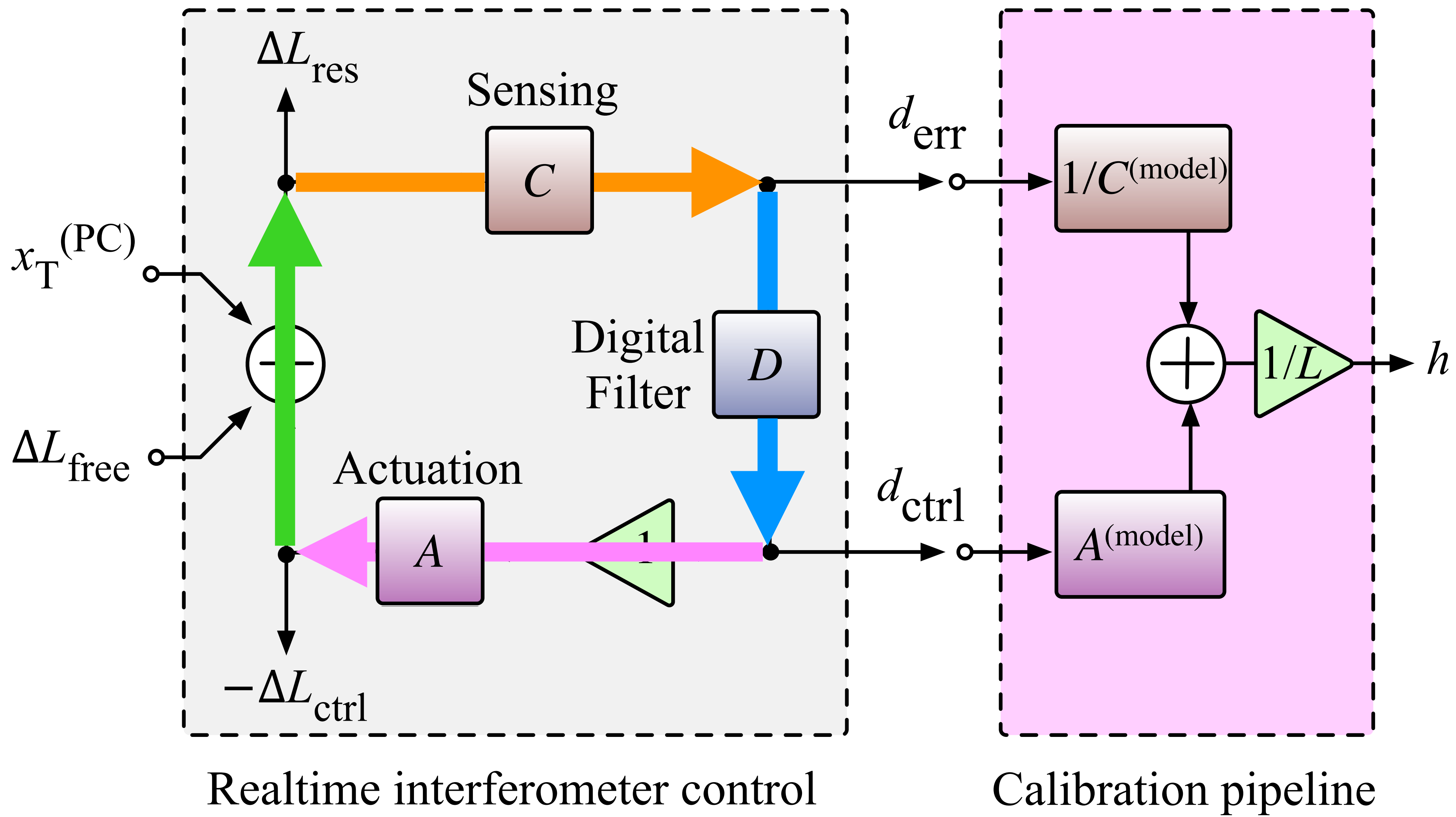
Realtime interferometer control



Realtime interferometer control

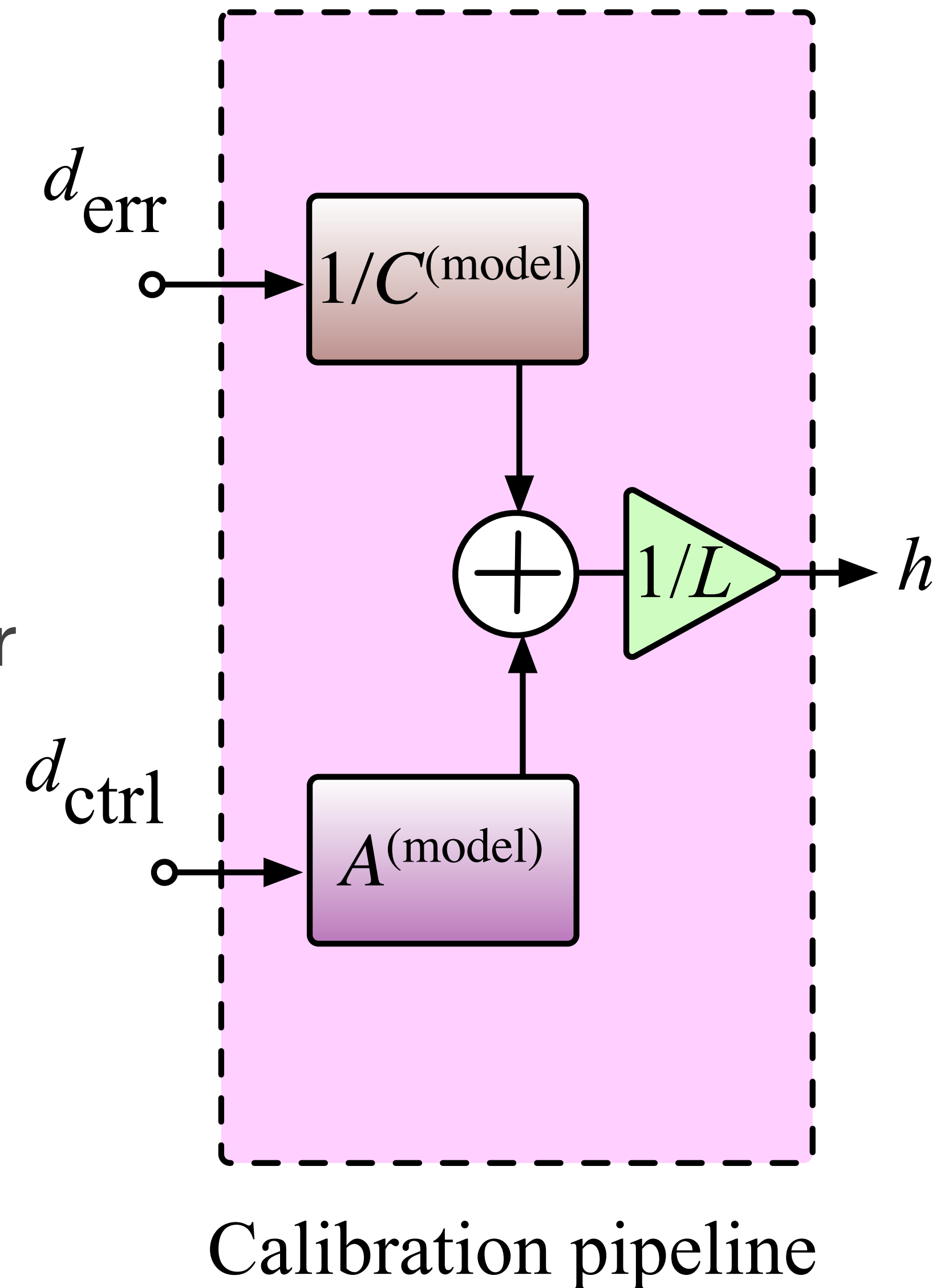


Realtime interferometer control



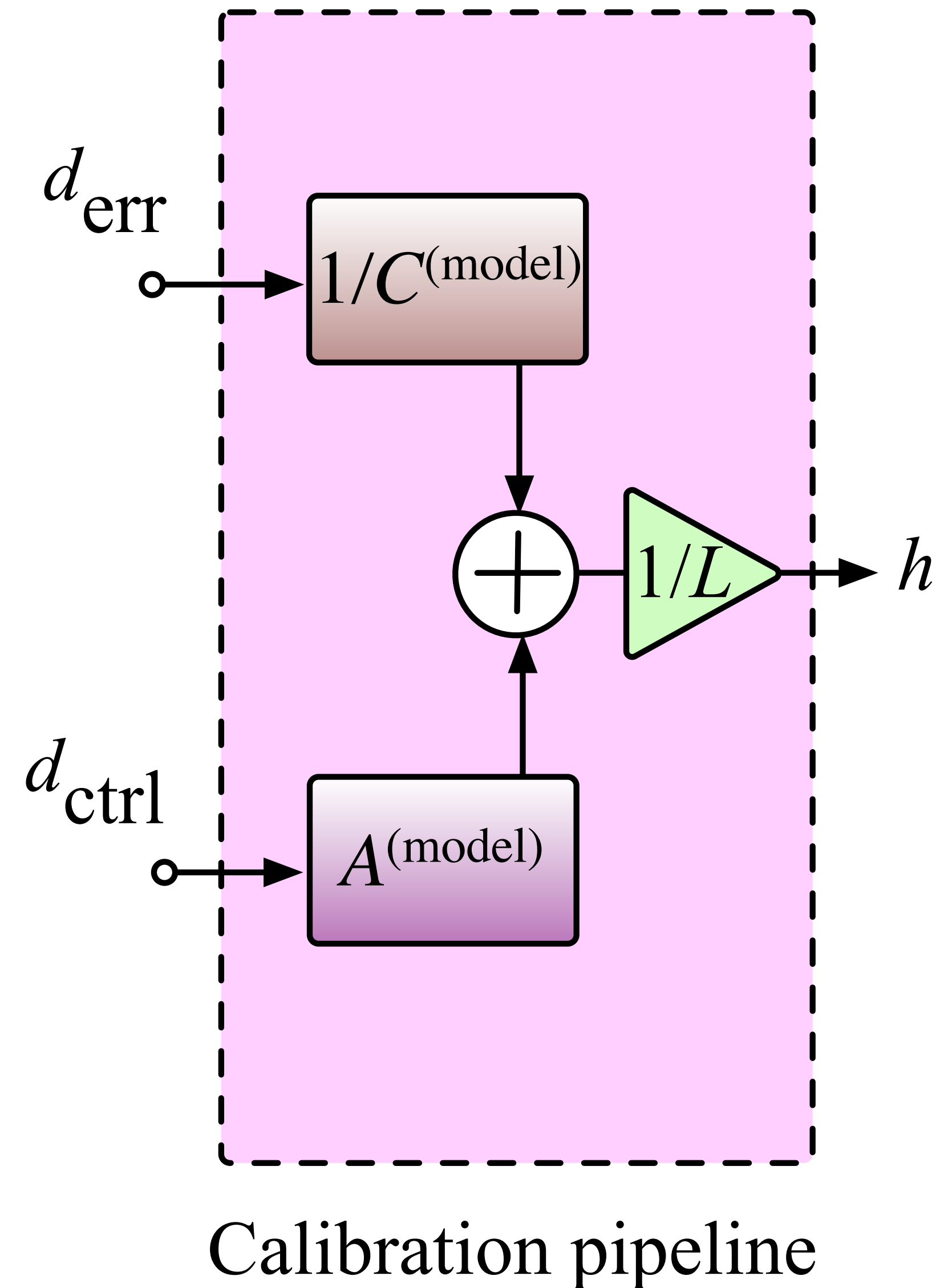
The Calibration Pipeline

- Pipeline that performs these calculations
 - Takes in raw detector readings (d_{err} and d_{ctrl})
 - Reconstructs the strain of the detector using models
 - Inverts the sensing model, sends error signal through
 - Sends control signal into Actuation model



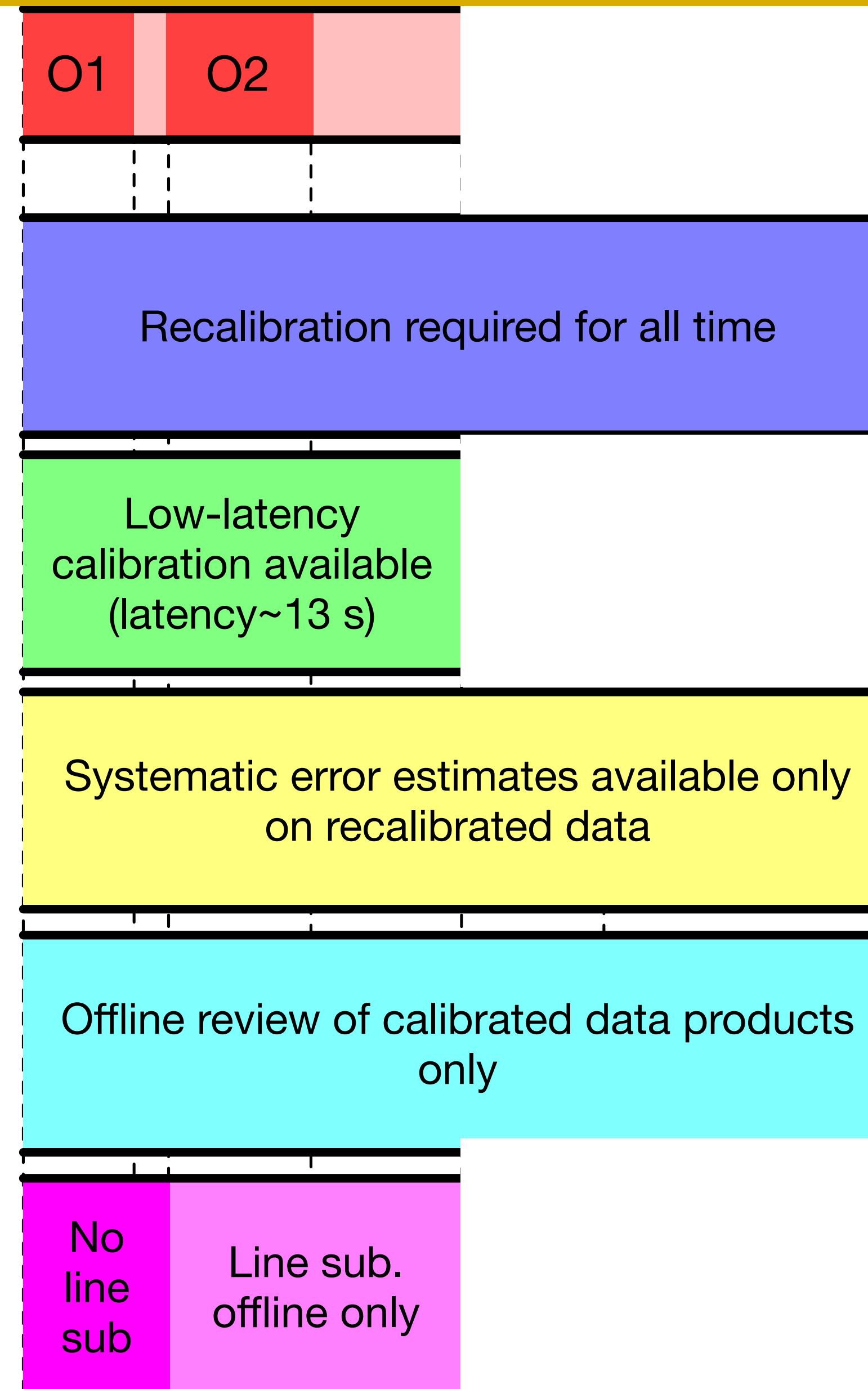
The Calibration Pipeline: Other Features

- Time-Dependent Correction Factors (TDCFs)
 - Models are static by default
 - Use live detector monitoring (from photon calibrators and suspension actuator lines) to make corrections to our models of detector behavior
- State Vector
 - Communicate information about the state of calibration and the detector
 - Used downstream to determine data usability at each time



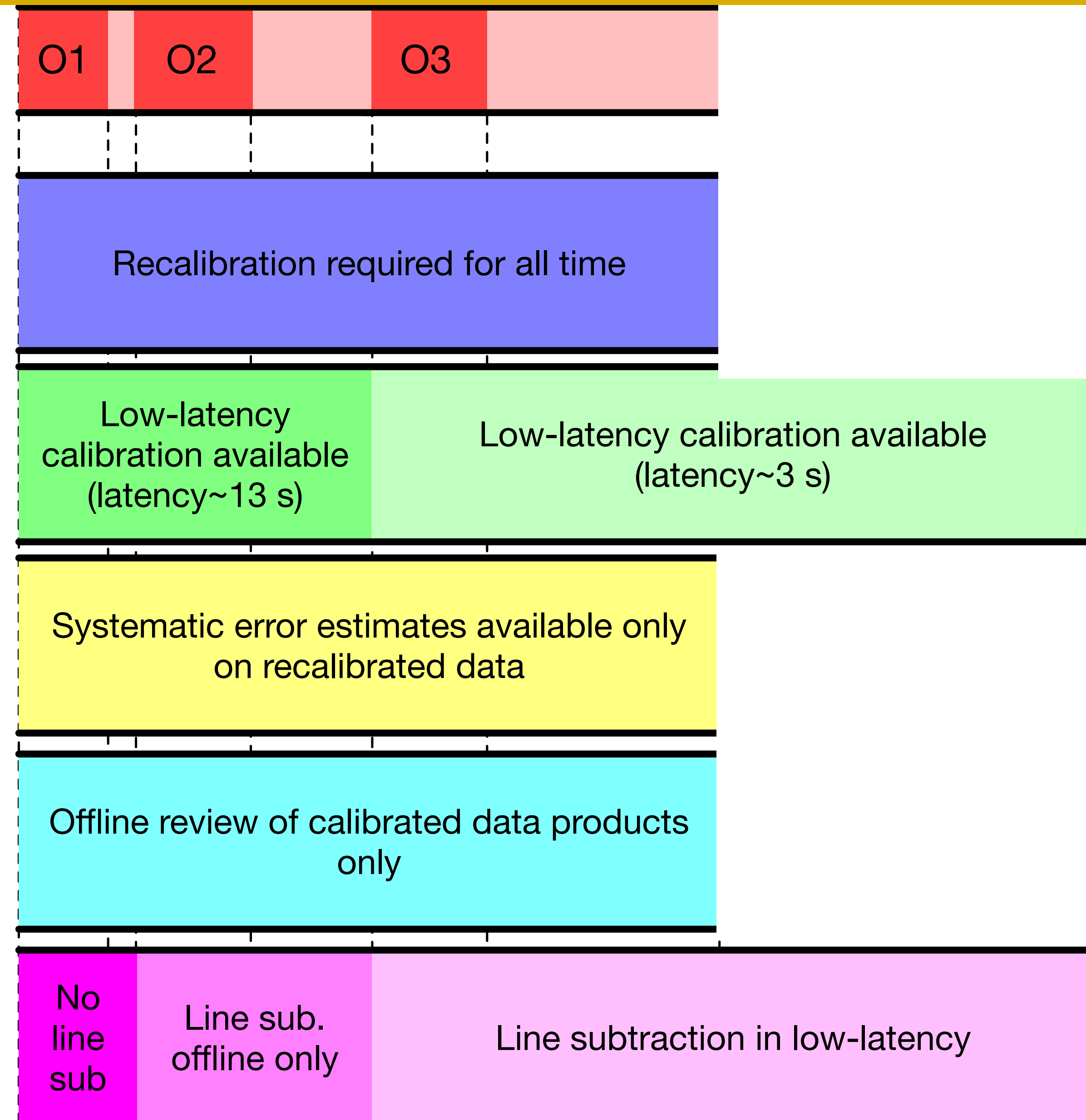
Calibration History

- O1-O3, final calibration was done after-the-fact
 - Low-latency calibration operated as well, but resulting strain was not officially released
- In O4, final calibration was done in low-latency!
 - Produced systematic error estimate
 - Real-time monitoring
 - Only occasional recalibration needed



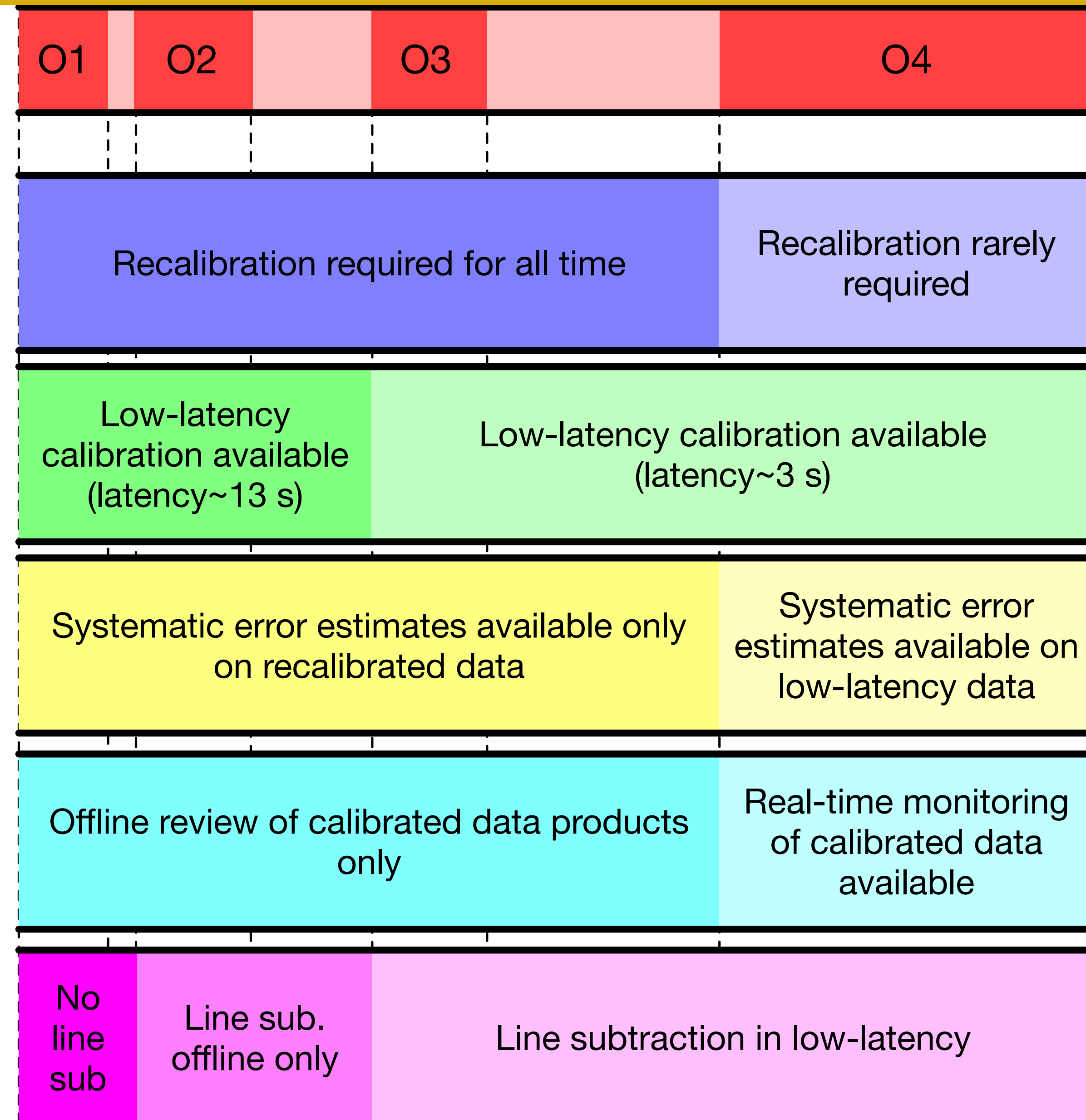
Calibration History

- O1-O3, final calibration was done after-the-fact
 - Low-latency calibration operated as well, but resulting strain was not officially released
- In O4, final calibration was done in low-latency!
 - Produced systematic error estimate
 - Real-time monitoring
 - Only occasional recalibration needed



Calibration History

- O1-O3, final calibration was done after-the-fact
 - Low-latency calibration operated as well, but resulting strain was not officially released
- In O4, final calibration was done in low-latency!
 - Produced systematic error estimate
 - Real-time monitoring
 - Only occasional recalibration needed



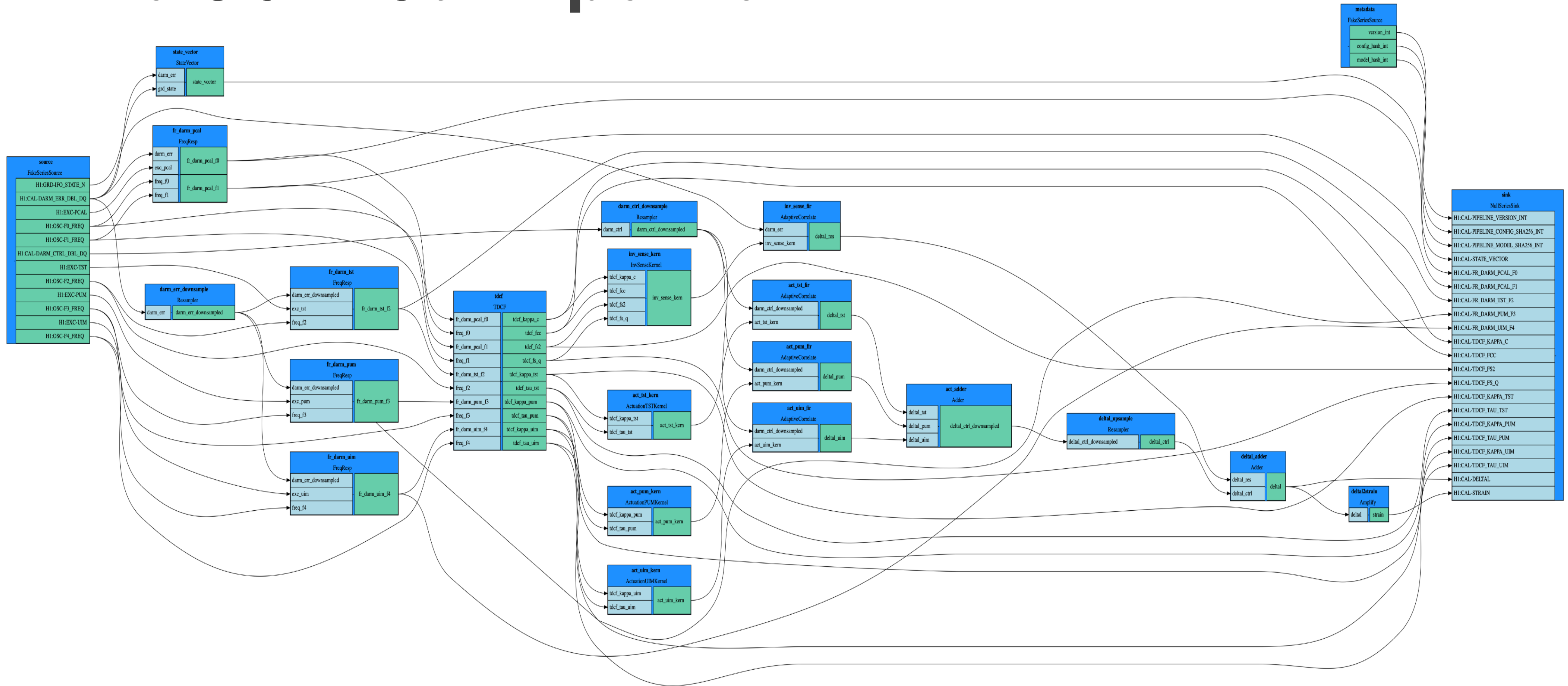
The GstLAL Calibration Pipeline

- The low-latency and offline calibration pipeline
- Used since O1, improved over time
- Room for improvement
 - Maintaining has become more difficult as complexity has increased
 - Certain model changes can be difficult to make
 - GstLAL is receiving less support
 - High barrier to entry for new developers
- Simplified layout

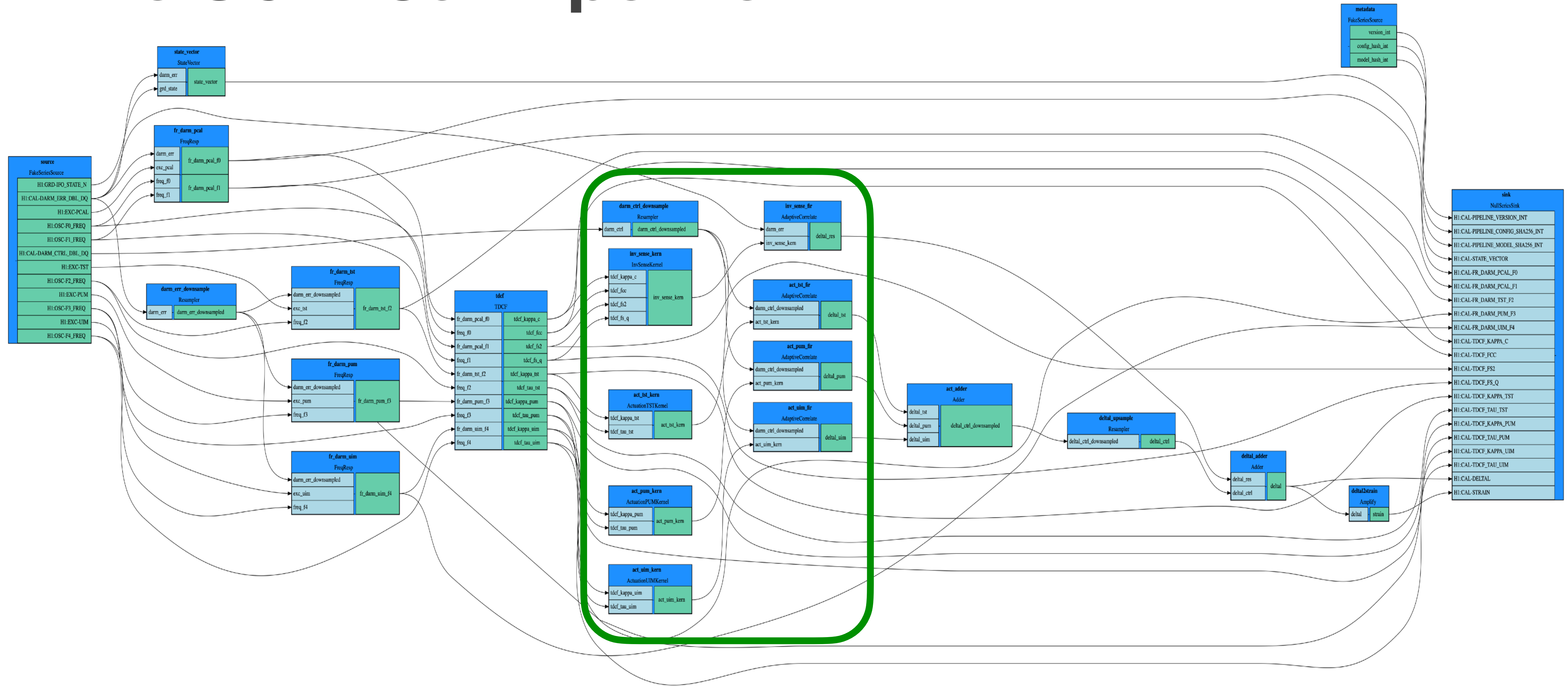
The GstLAL Calibration Pipeline



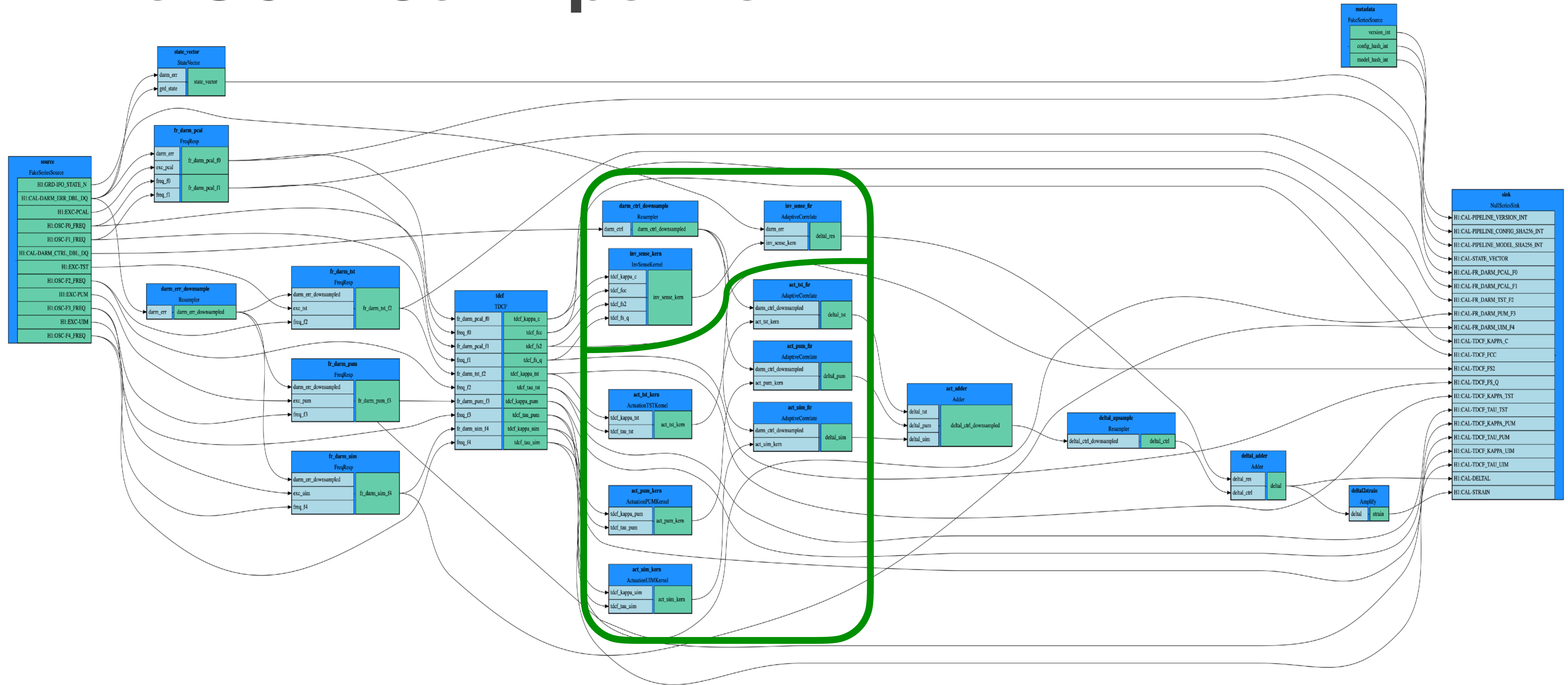
The SGN-Cal Pipeline



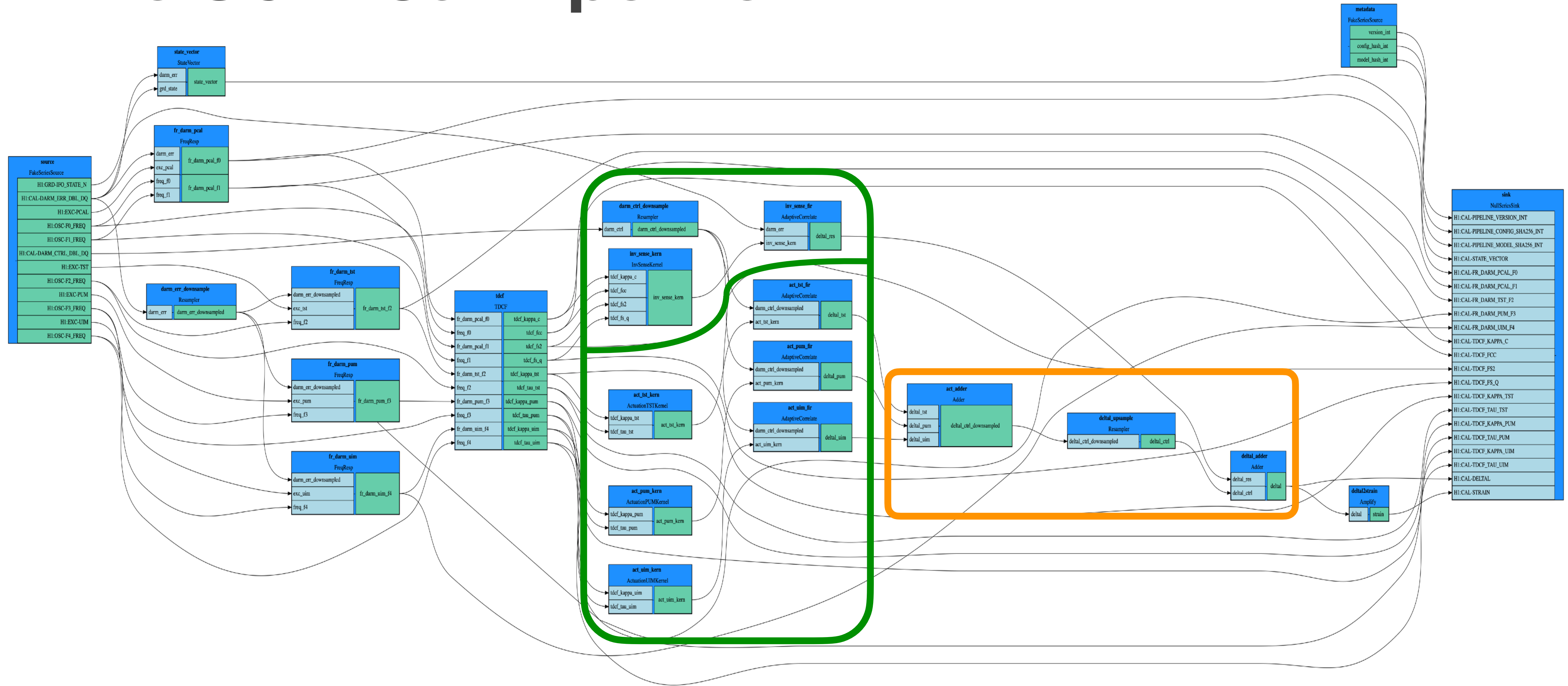
The SGN-Cal Pipeline



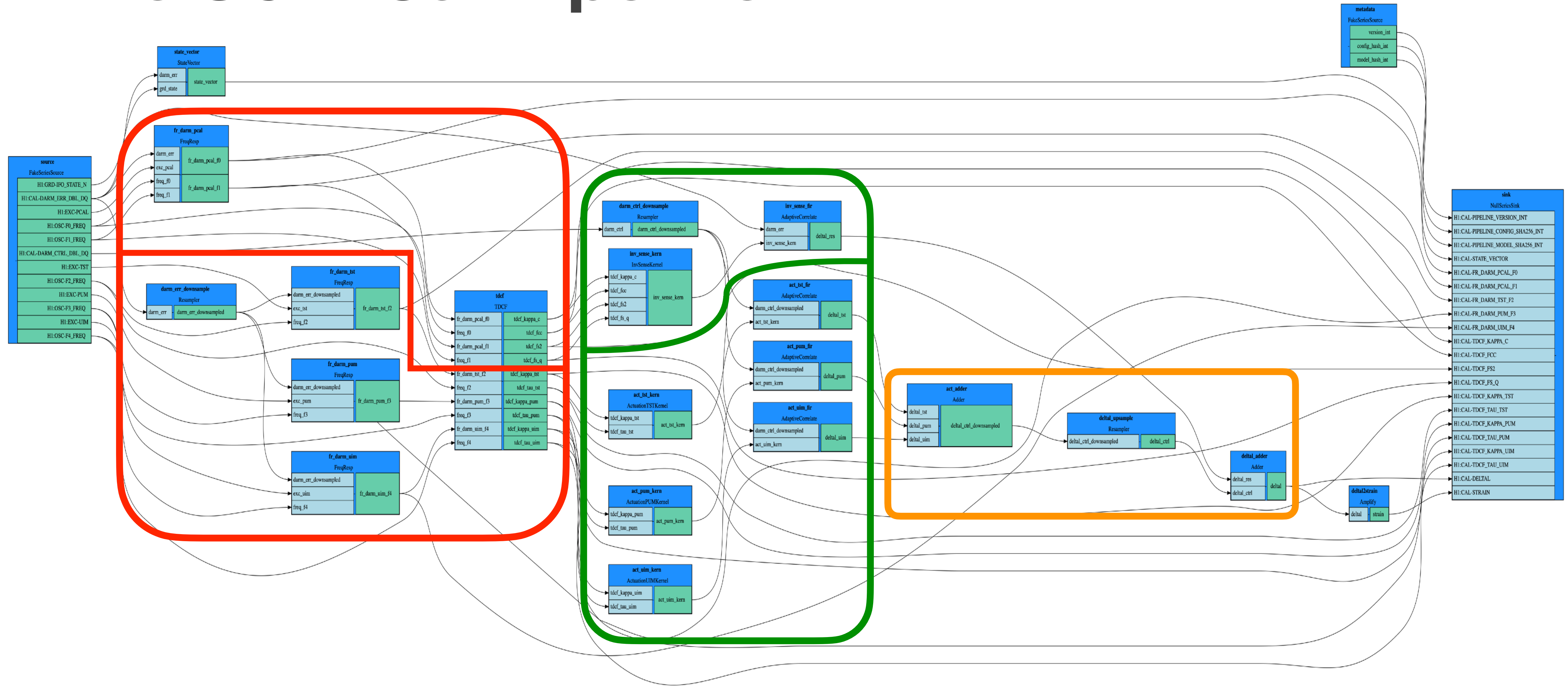
The SGN-Cal Pipeline



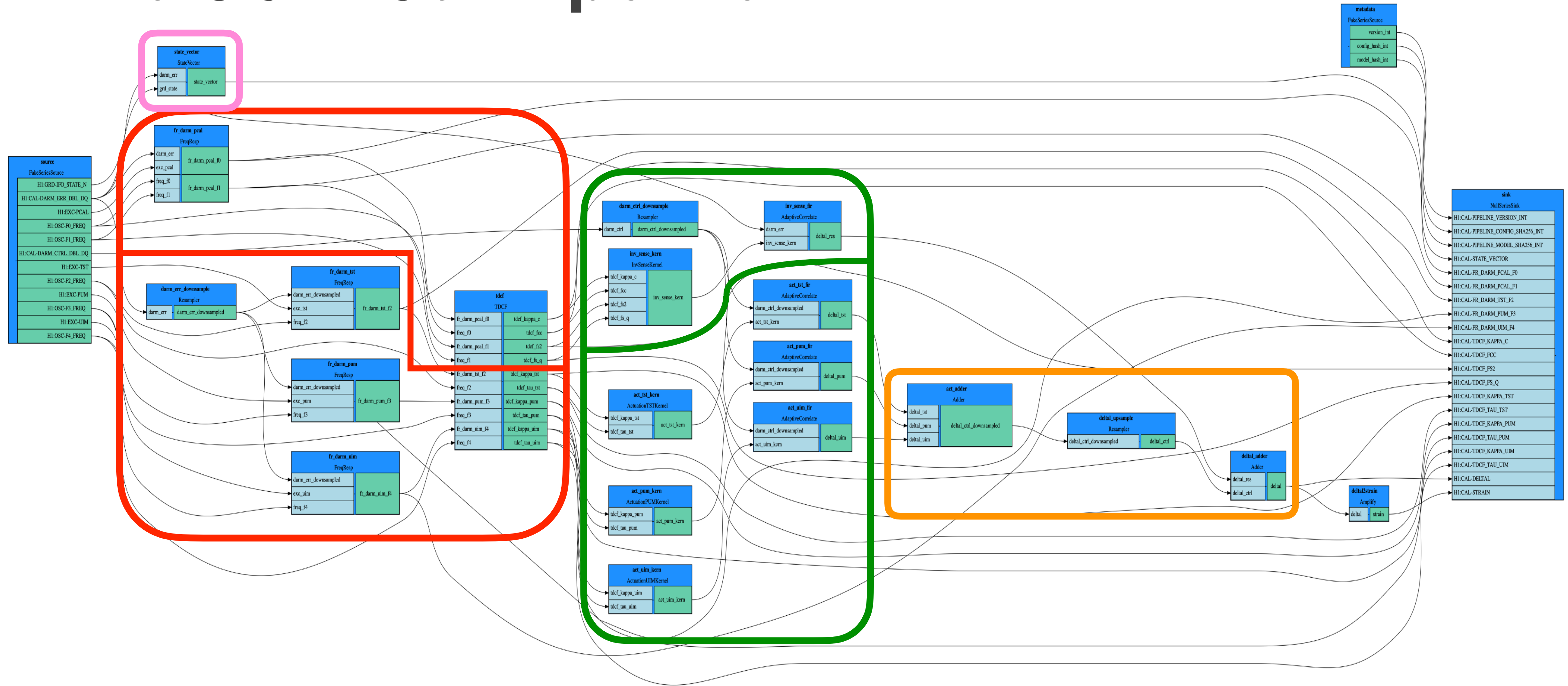
The SGN-Cal Pipeline



The SGN-Cal Pipeline



The SGN-Cal Pipeline

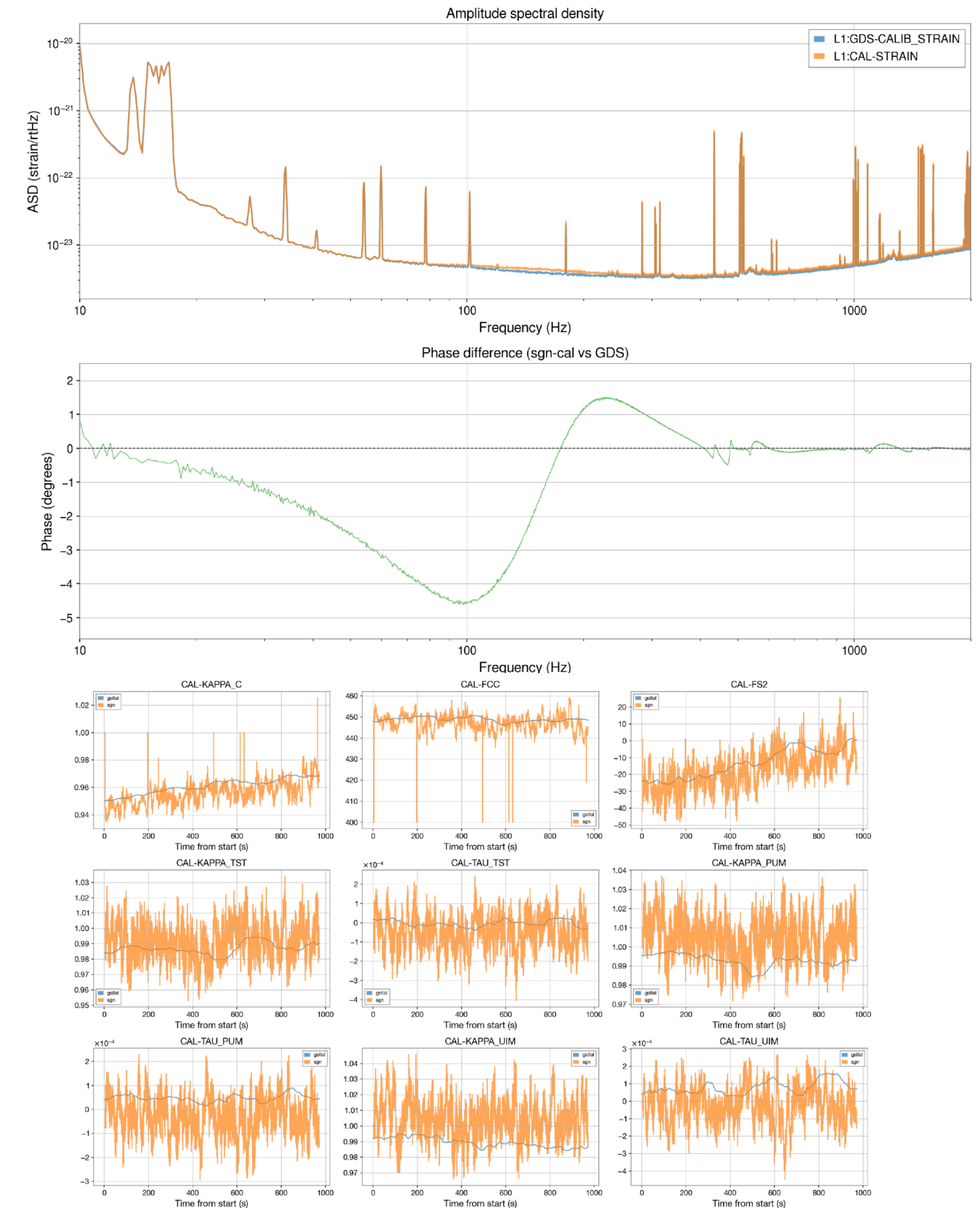


Benefits of SGN-Cal

- Continues to deliver high-quality low-latency calibration
- Simpler layout
- Modular (easier to swap models)
- Lower barrier to contribution (python)
- Better integrated with monitoring of calibration validity
 - Working on this now

Status and Future Work

- Status:
 - Pipeline is assembled and generally functional!
 - Comparing SGN-Cal output to GstLAL pipeline to verify matching results
- Future Work:
 - Debugging discrepancies
 - Testing on long-time data
 - Plan to be ready for use in IR1



Calibration State Vector Progress

- 16 Bits
 - 5 Assorted
 - 4 Hardware Injection
 - 7 TDCF

bit	Short descriptor	Long descriptor
0	HOFT_OK	$h(t)$ was successfully computed
1	OBS_INTENT	interferometer is in “observation intent” mode
2	OBS_READY	interferometer is in “observation ready” mode
3	FILTERS_OK	calibration filters settled in
4	NO_GAP	The input data was present (not a gap)
5	NO_STOCH_HW_INJ	No stochastic hardware injections present
6	NO_CBC_HW_INJ	No compact binary coalescence hardware injections present
7	NO_BURST_HW_INJ	No burst hardware injections present
8	NO_DETCHAR_HW_INJ	No detector characterization hardware injections present
9	UNDISTURBED_OK	interferometer is in “undisturbed” mode
10	KAPPA_TST_SMOOTH_OK	κ_T smoothing is settled in and values are in range
11	KAPPA_PUM_SMOOTH_OK	$\kappa_{PUM/PU}$ smoothing is settled in and values are in range
12	KAPPA_UIM_SMOOTH_OK	κ_{UIM} smoothing is settled in and values are in range
13	KAPPA_C_SMOOTH_OK	κ_C smoothing is settled in and values are in range
14	F_CC_SMOOTH_OK	f_{cc} smoothing is settled in and values are in range
15	F_S_SQUARED_SMOOTH_OK	f_s^2 smoothing is settled in and values are in range
16	F_S_OVER_Q_SMOOTH_OK	f_s/Q smoothing is settled in and values are in range
17	LOWFREQ_LINE_SUBTR	Low-frequency line subtraction okay
18	MIDFREQ_LINE_SUBTR	Mid-frequency line subtraction okay
19	LINE_SUBTR_GATE	Line subtraction gate okay
20	NONSENS_SUBTR	NonSENS noise subtraction okay

https://dcc.ligo.org/DocDB/0158/T1900007/005/T1900007_CALIB_STATE_VECTOR.pdf

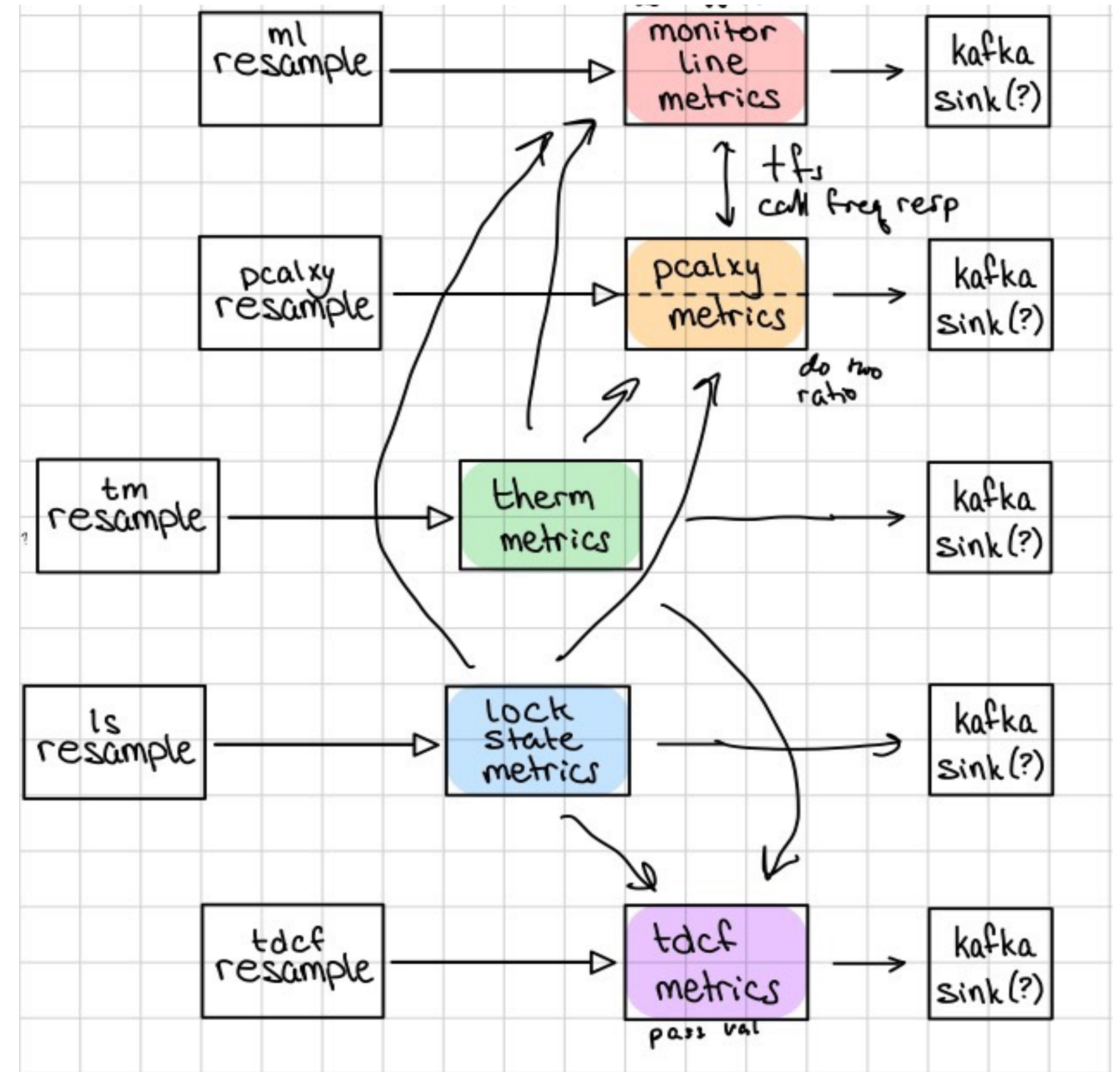
The background of the slide features a series of overlapping, rounded hills in various shades of green, ranging from light mint to dark forest green. The hills are layered to create a sense of depth and perspective, with the foreground hills being the most prominent and darkest, and the background hills being lighter and more faded.

SGN-CalMon

The New Monitoring Pipeline

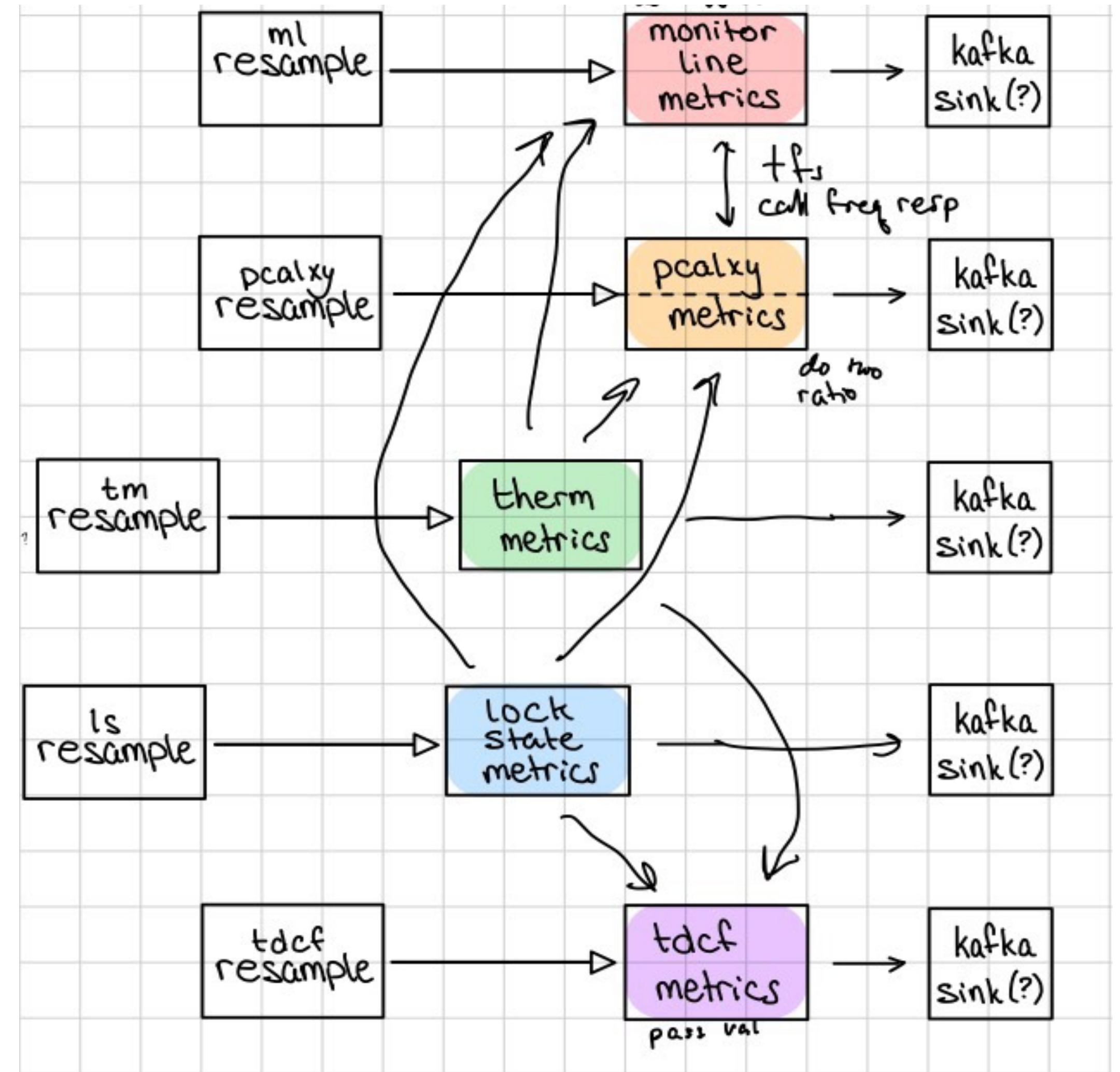
The Monitoring Pipeline

- Real-time monitoring system for calibrated data
- Computes metrics that communicate state of detector
- Tell us about quality of calibration
- Important because
 - Gives systematic error on low-latency data (so low-latency calibration can be final product)
 - Catch calibration errors quickly



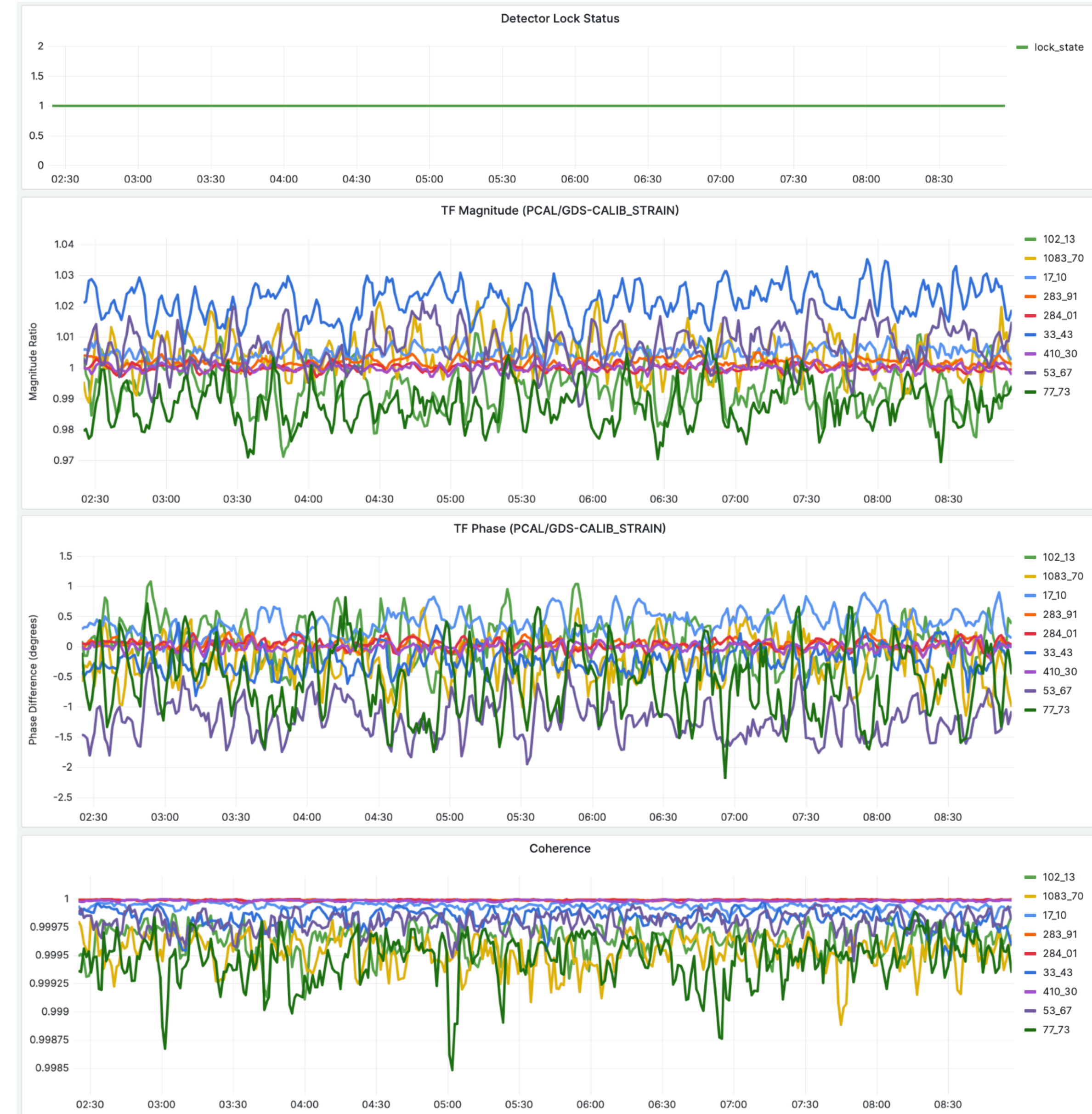
Monitoring Components

- Monitor line metrics
 - Mag, phase, coherence
- PcalXY
 - pcalx, pcaly, x/y ratio
- Therm metrics
- Lock state metrics
- TDCF metrics
 - Pulled from calibration pipeline



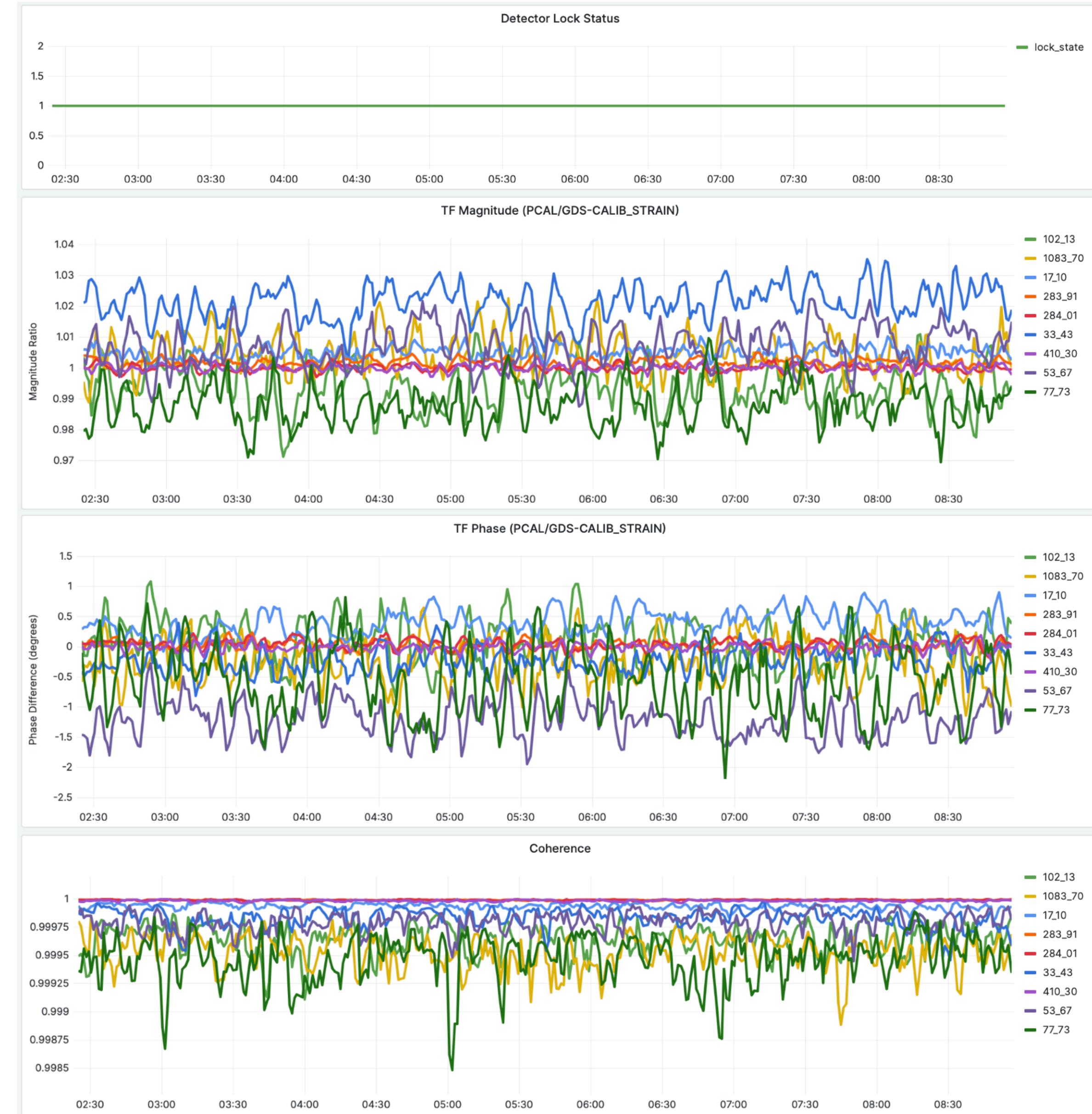
Monitoring Data

- Many metrics computed
- Let's look at Pcal/Strain Ratio



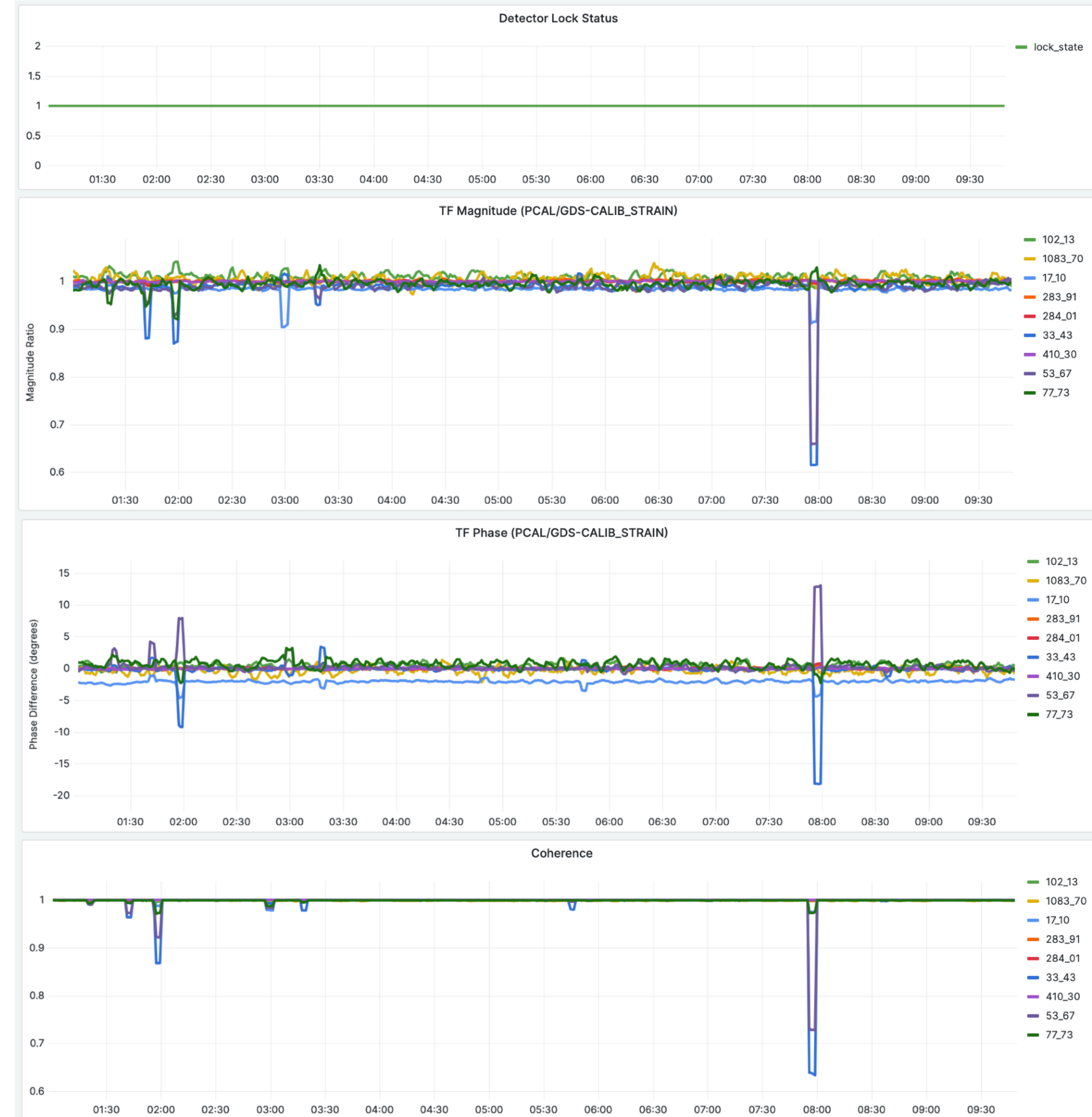
Monitoring Data

- Many metrics computed
- Let's look at Pcal/Strain Ratio
 - Sometimes good



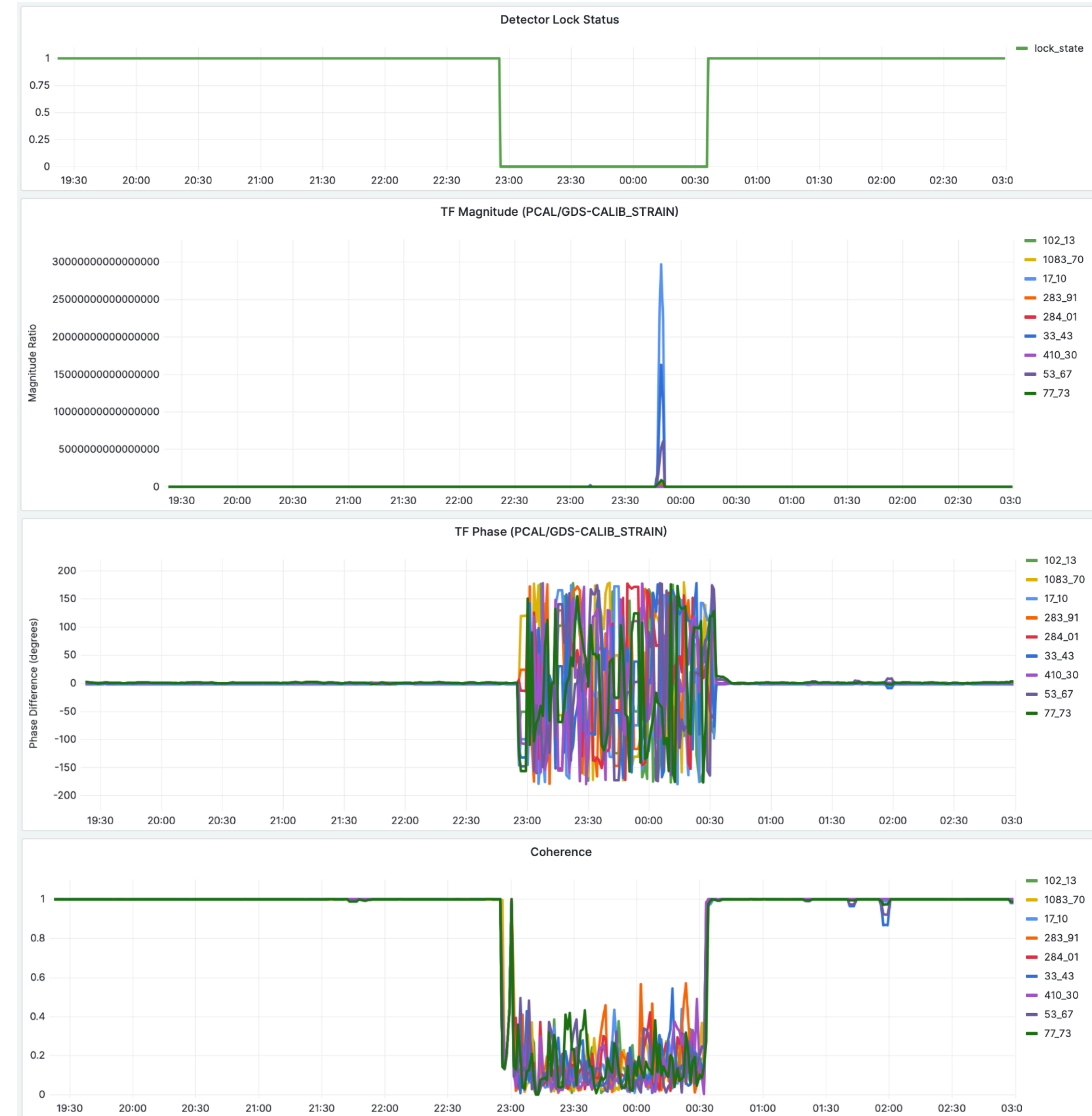
Monitoring Data

- Many metrics computed
- Let's look at Pcal/Strain Ratio
 - Sometimes good
 - Sometimes bad



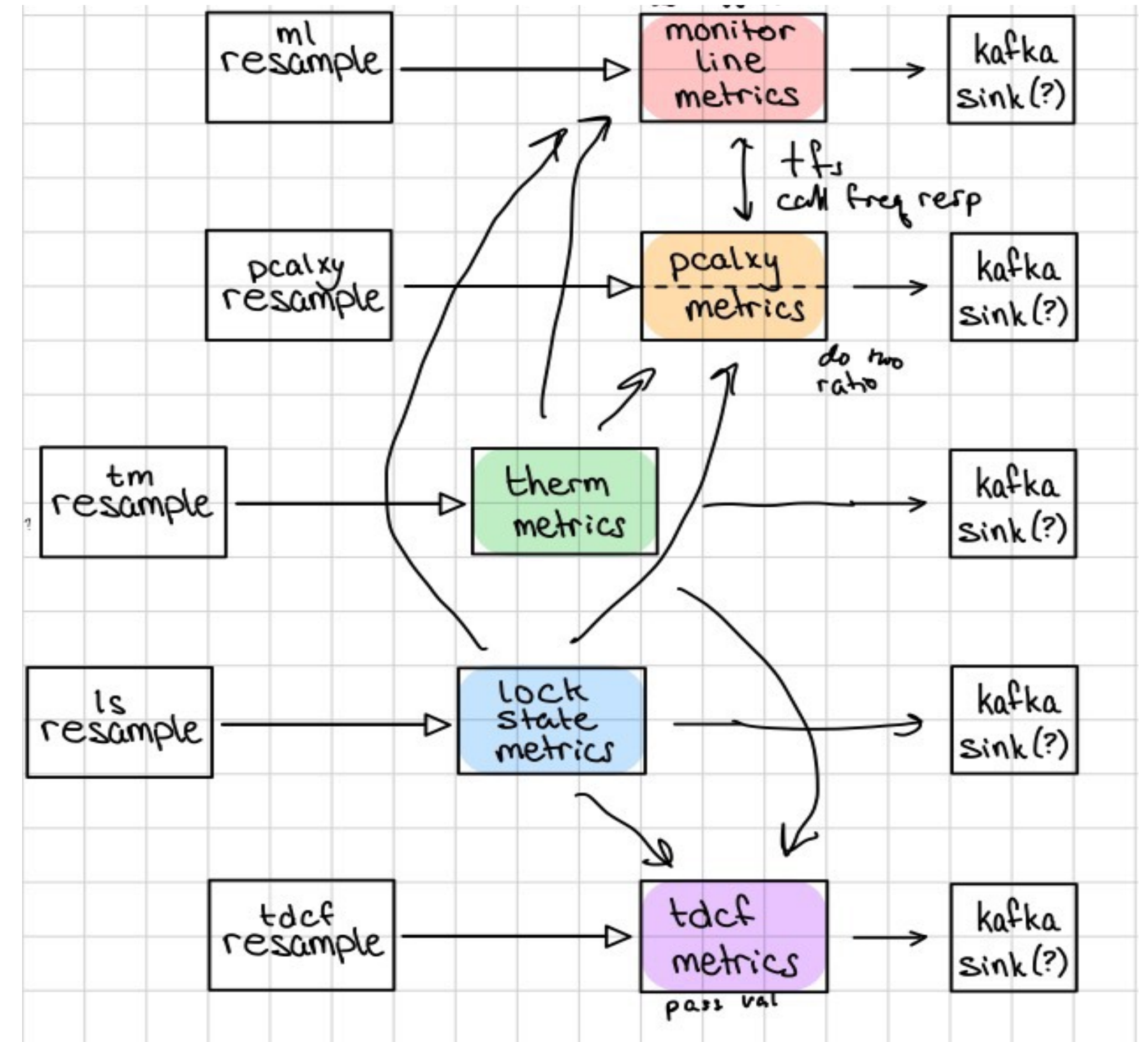
Monitoring Data

- Many metrics computed
- Let's look at Pcal/Strain Ratio
 - Sometimes good
 - Sometimes bad
 - Sometimes the detector isn't locked



The job

- Help create the new monitoring pipeline
 - Create elements that perform GstLAL-equivalent calculations
 - Assemble into working pipeline
 - Integrate with SGN-Cal pipeline
- Primarily done by Eve Currens
- I have assisted in creating elements
- The elements are now largely completed
- On track for IR1



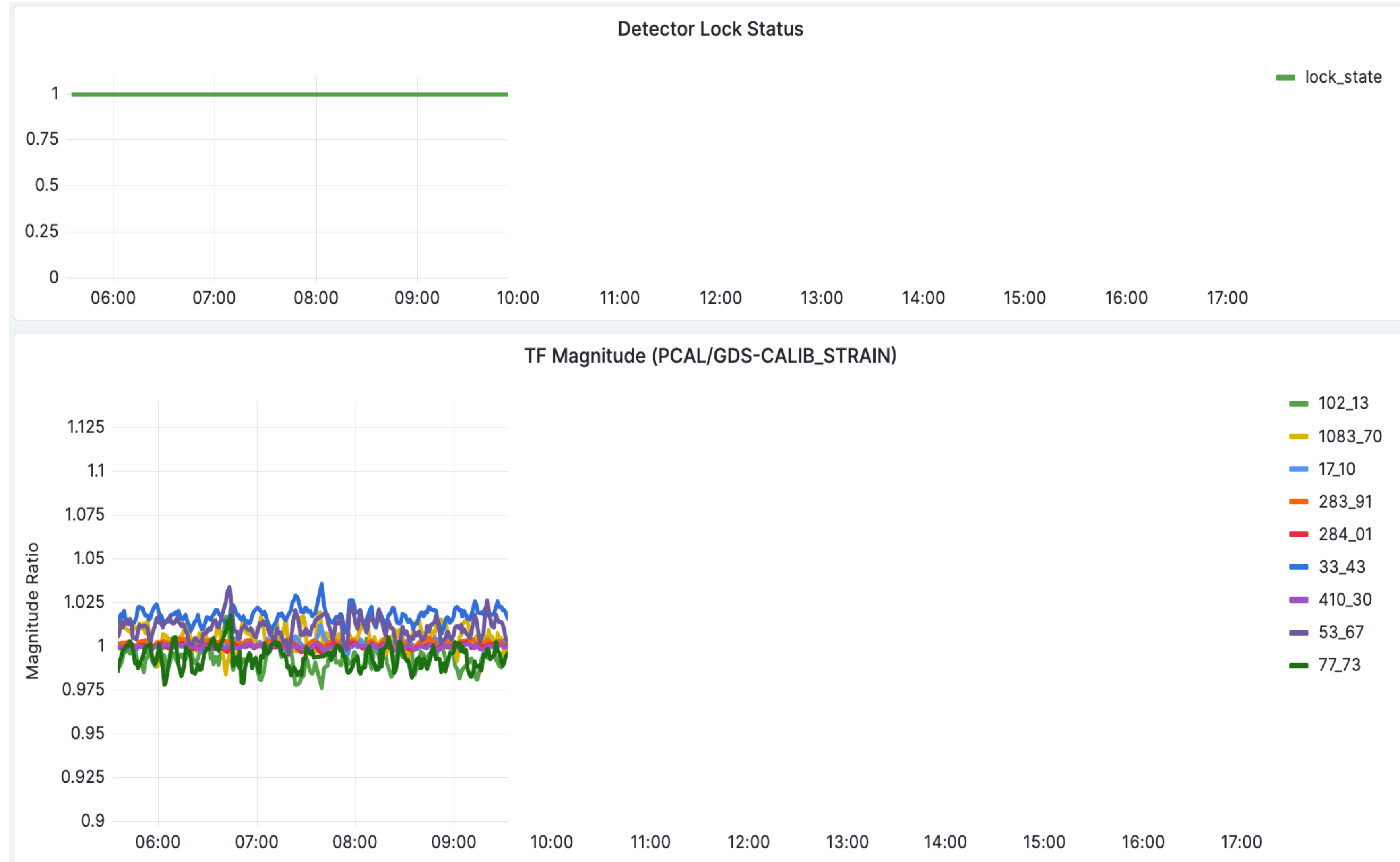


Control Room

Adding Monitoring Alerts

Backstory

- One fateful day...
- Someone changes the calibration
- Same someone checks Grafana
- The calibration is wrong!
- Noticing this quickly saved lots of work
- We want to set up automatic alerts



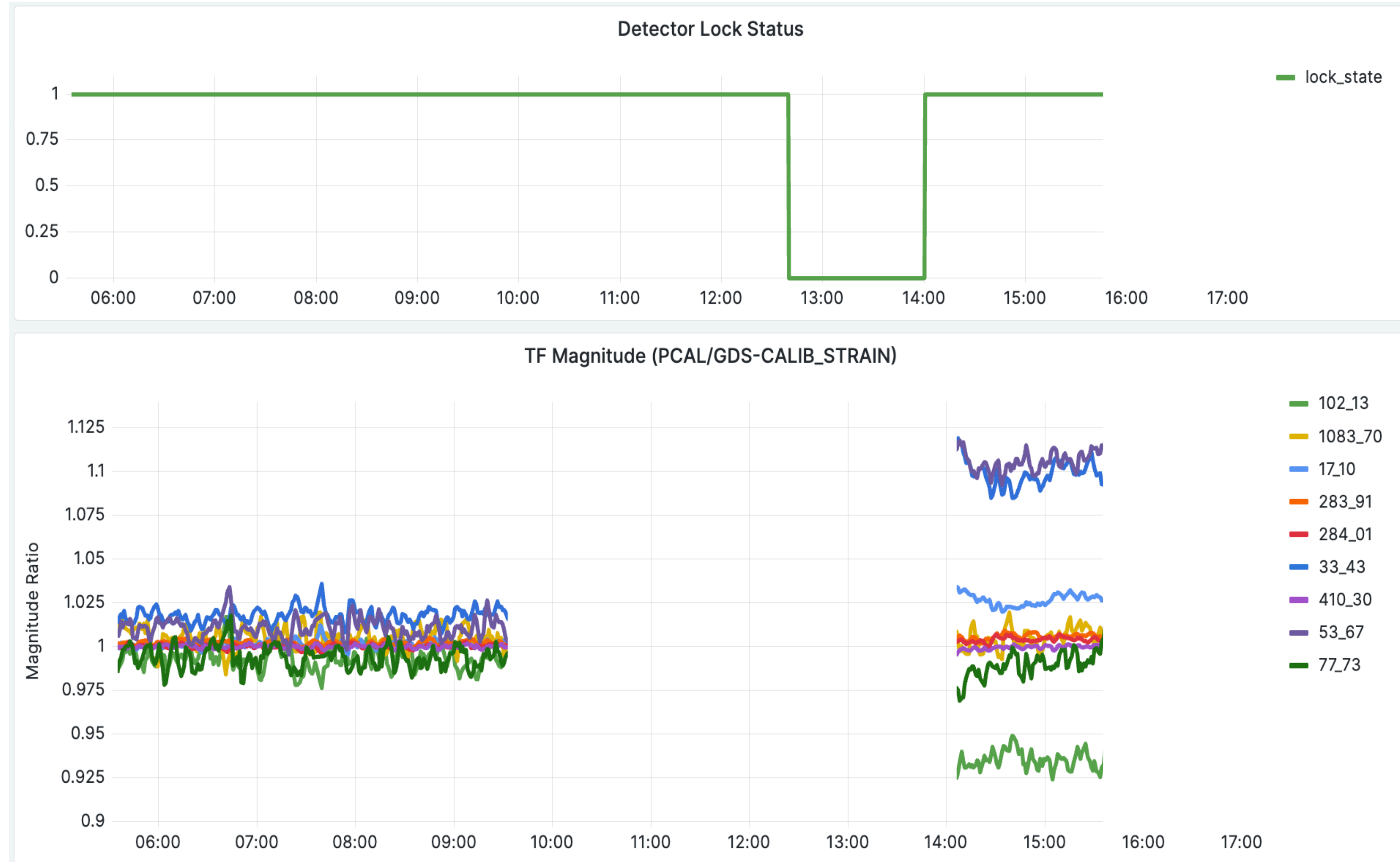
Backstory

- One fateful day...
- Someone changes the calibration
- Same someone checks Grafana
- The calibration is wrong!
- Noticing this quickly saved lots of work
- We want to set up automatic alerts



Backstory

- One fateful day...
- Someone changes the calibration
- Same someone checks Grafana
- The calibration is wrong!
- Noticing this quickly saved lots of work
- We want to set up automatic alerts



Backstory

- One fateful day...
- Someone changes the calibration
- Same someone checks Grafana
- The calibration is wrong!
- Noticing this quickly saved lots of work
- We want to set up automatic alerts



My job



My job

Influxdb

Control
Room
Screen

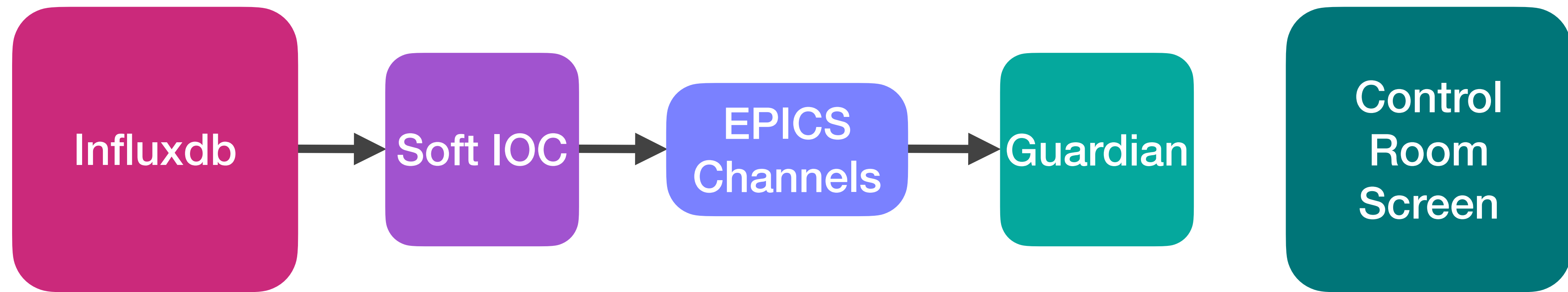
My job



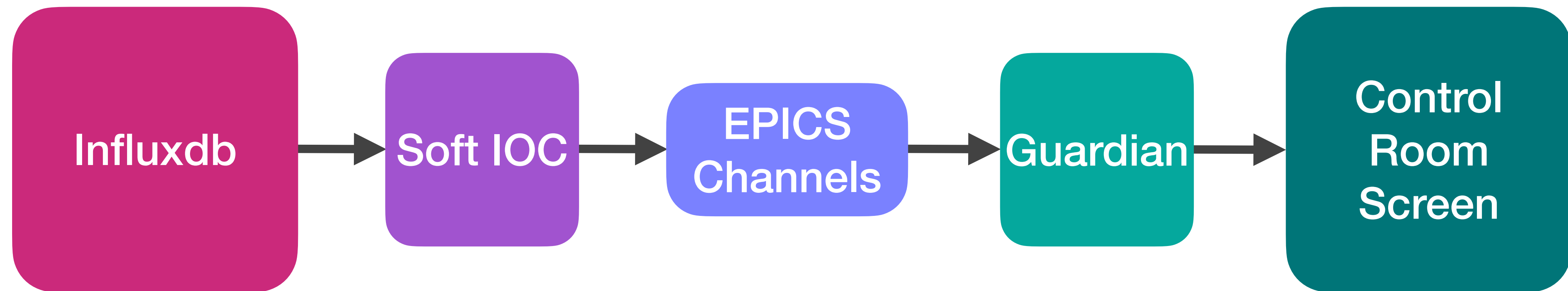
My job



My job



My job



Soft IOC

- Software input/output controller
- Scrapes monitoring values from influxdb
- Publishes them to epics channels

EPICS Channels

- Created 26 channels
 - Two meta channels

L1:CAL-MONITOR_LINE1_MAG
L1:CAL-MONITOR_LINE2_MAG
L1:CAL-MONITOR_LINE3_MAG
L1:CAL-MONITOR_LINE4_MAG
L1:CAL-MONITOR_LINE5_MAG
L1:CAL-MONITOR_LINE6_MAG
L1:CAL-MONITOR_LINE7_MAG
L1:CAL-MONITOR_LINE8_MAG

L1:CAL-MONITOR_LINE1_PHA
L1:CAL-MONITOR_LINE2_PHA
L1:CAL-MONITOR_LINE3_PHA
L1:CAL-MONITOR_LINE4_PHA
L1:CAL-MONITOR_LINE5_PHA
L1:CAL-MONITOR_LINE6_PHA
L1:CAL-MONITOR_LINE7_PHA
L1:CAL-MONITOR_LINE8_PHA

L1:CAL-MONITOR_LINE1_COH
L1:CAL-MONITOR_LINE2_COH
L1:CAL-MONITOR_LINE3_COH
L1:CAL-MONITOR_LINE4_COH
L1:CAL-MONITOR_LINE5_COH
L1:CAL-MONITOR_LINE6_COH
L1:CAL-MONITOR_LINE7_COH
L1:CAL-MONITOR_LINE8_COH

L1:CAL-MONITOR_DATA_TIME
L1:CAL-MONITOR_FAIL_COUNTER

Guardian

- We added some guardian code that:
 - Reads in the data from our channels
 - Checks thresholds over duration
 - Puts alerts on control room screen if conditions aren't met

Ongoing Plans

- Better triggering
 - Value thresholds aren't very informative
 - Want to trigger on derivatives, other things
 - Plan to add new metrics to monitoring pipeline
 - Currently building the pipeline, will add this later
- Port to Hanford
 - Interface with Hanford people, modify to match their setup

The background of the slide features a series of overlapping, rounded hills in various shades of green, ranging from light mint to dark forest green. The hills are layered to create a sense of depth, with the most vibrant green in the foreground and lighter, more translucent greens in the background. The overall aesthetic is clean, modern, and nature-inspired.

SGN-TS

Time Series Tools

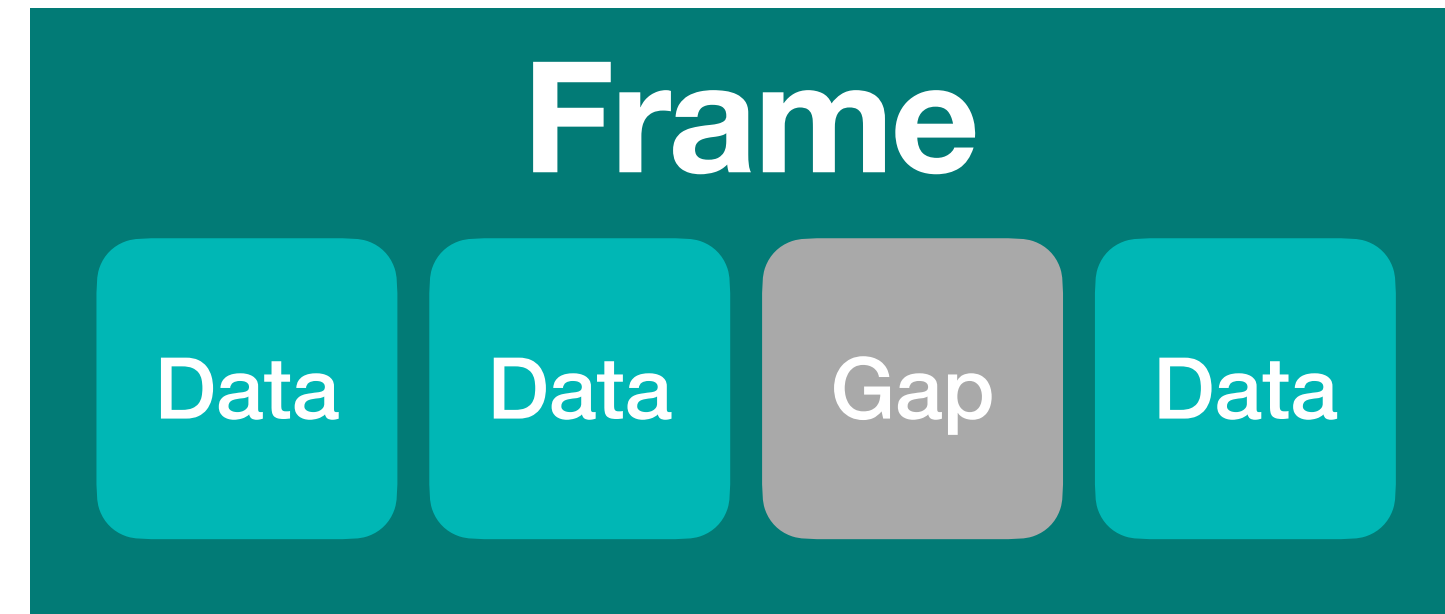
Added Frame Methods

Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask

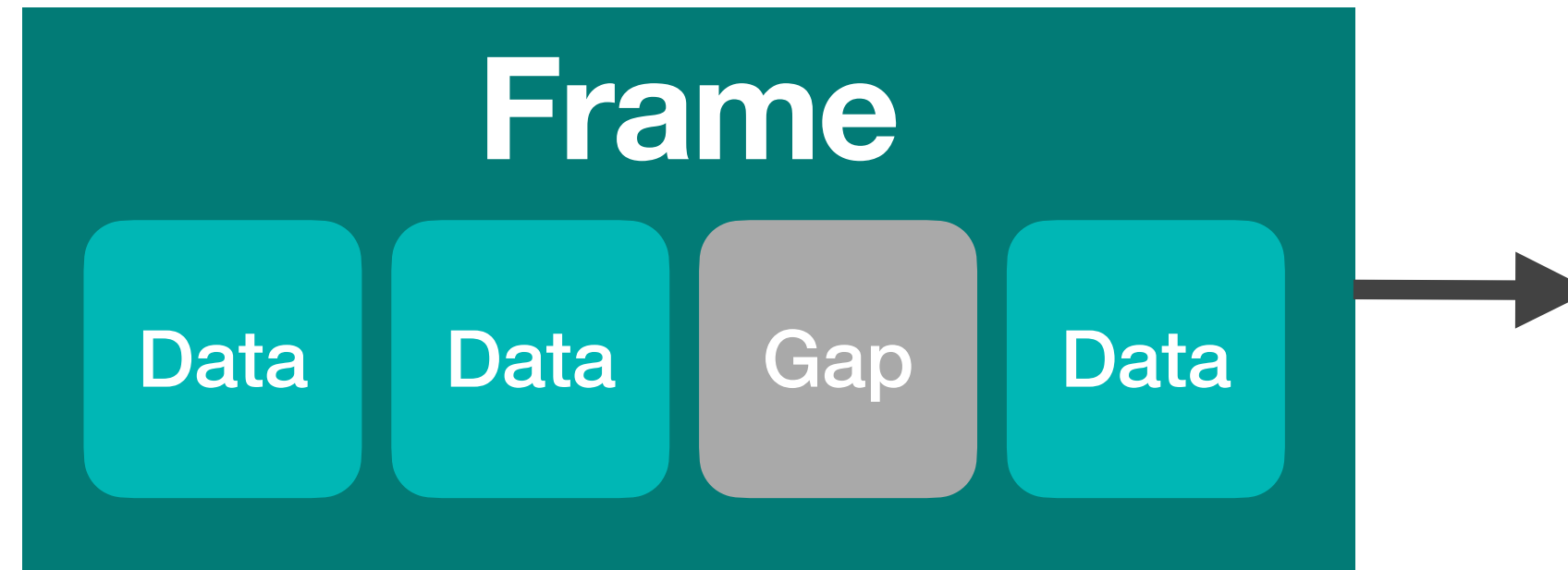
Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask



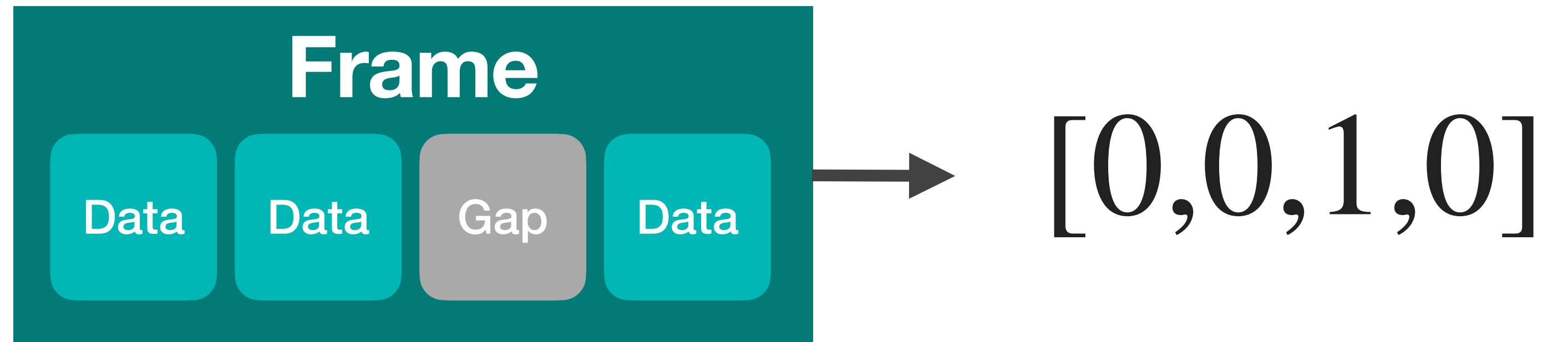
Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask



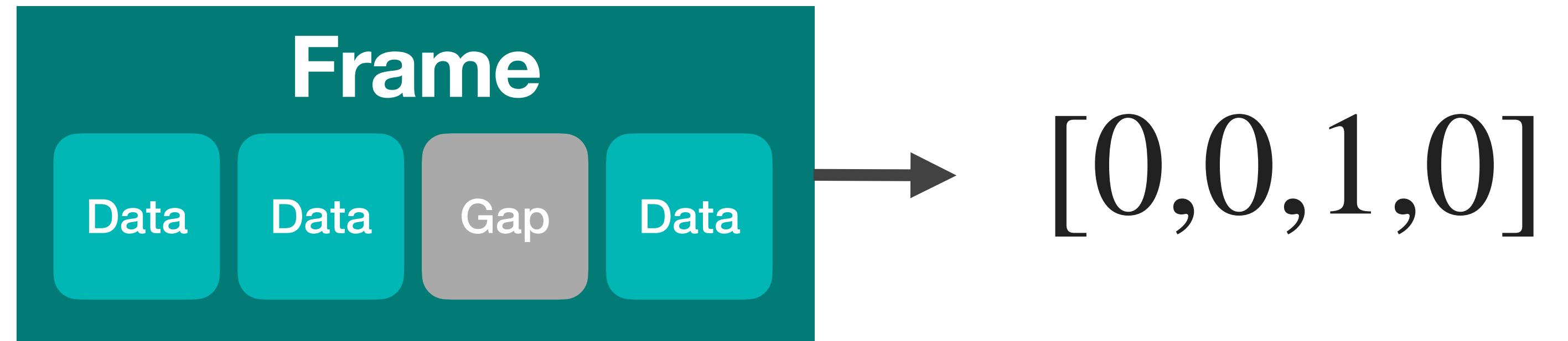
Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask



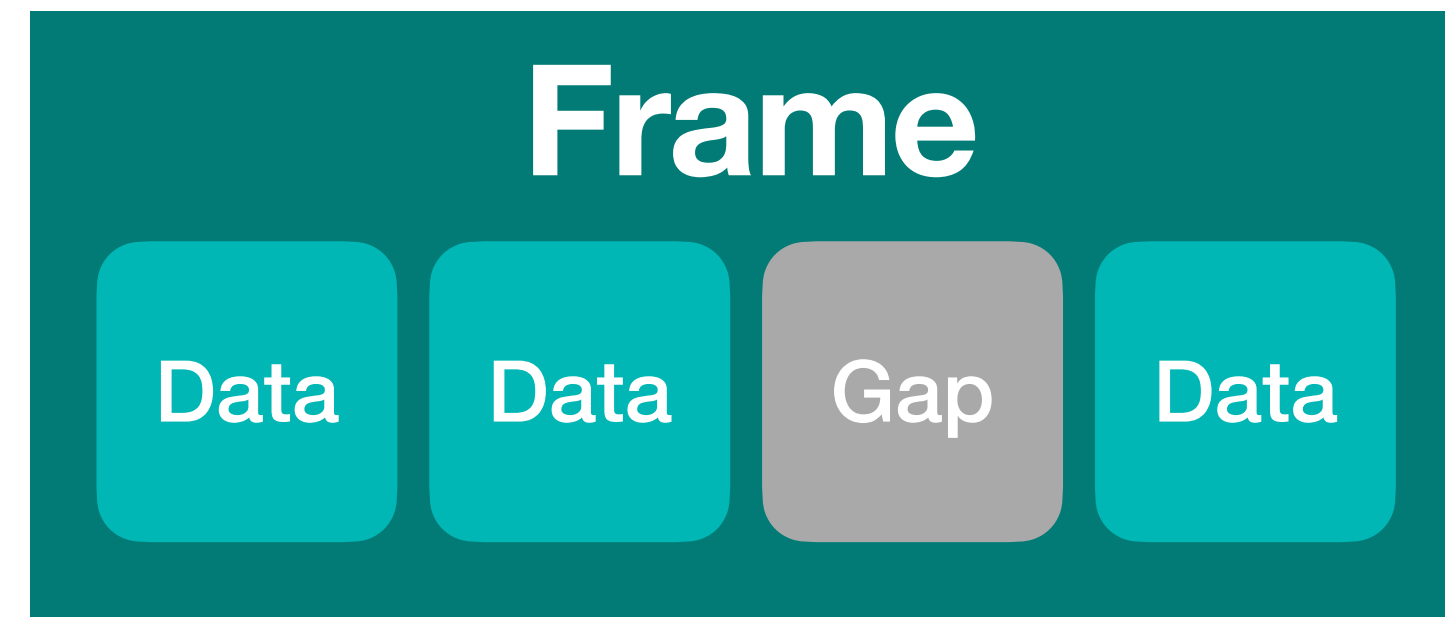
Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask
- maskeddata
 - Convert frame to masked array

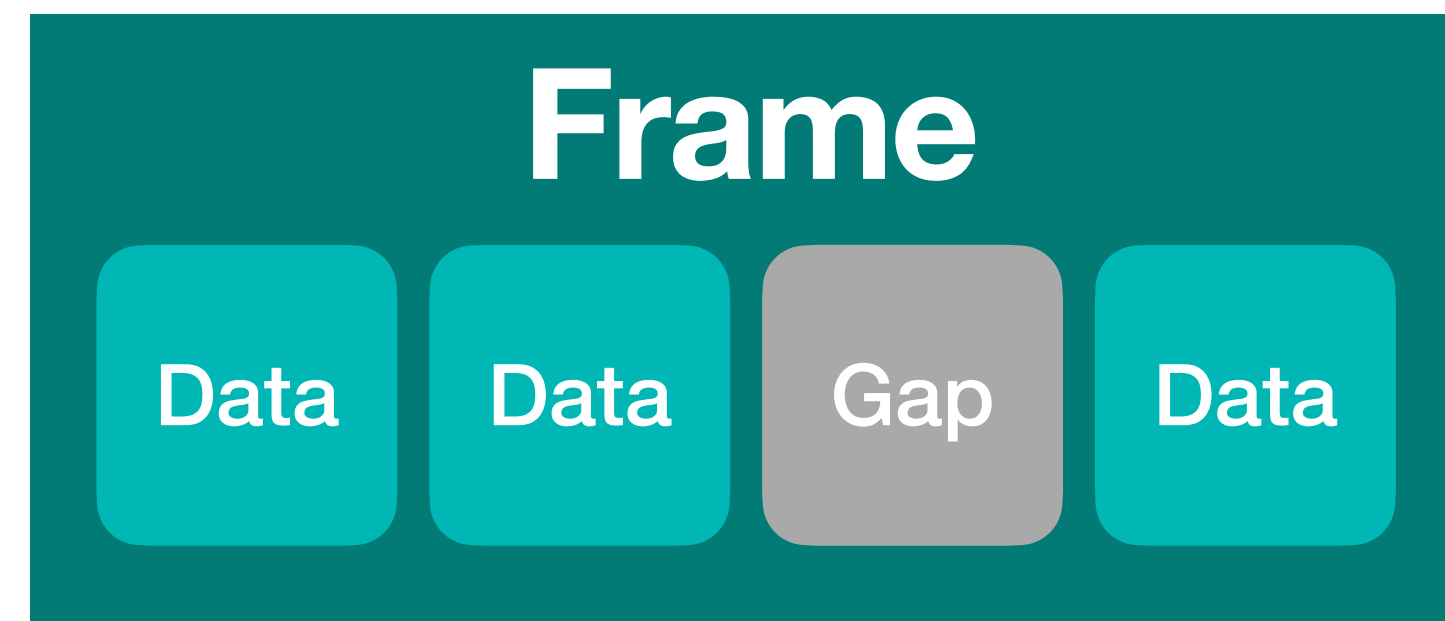


Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask
- maskeddata
 - Convert frame to masked array

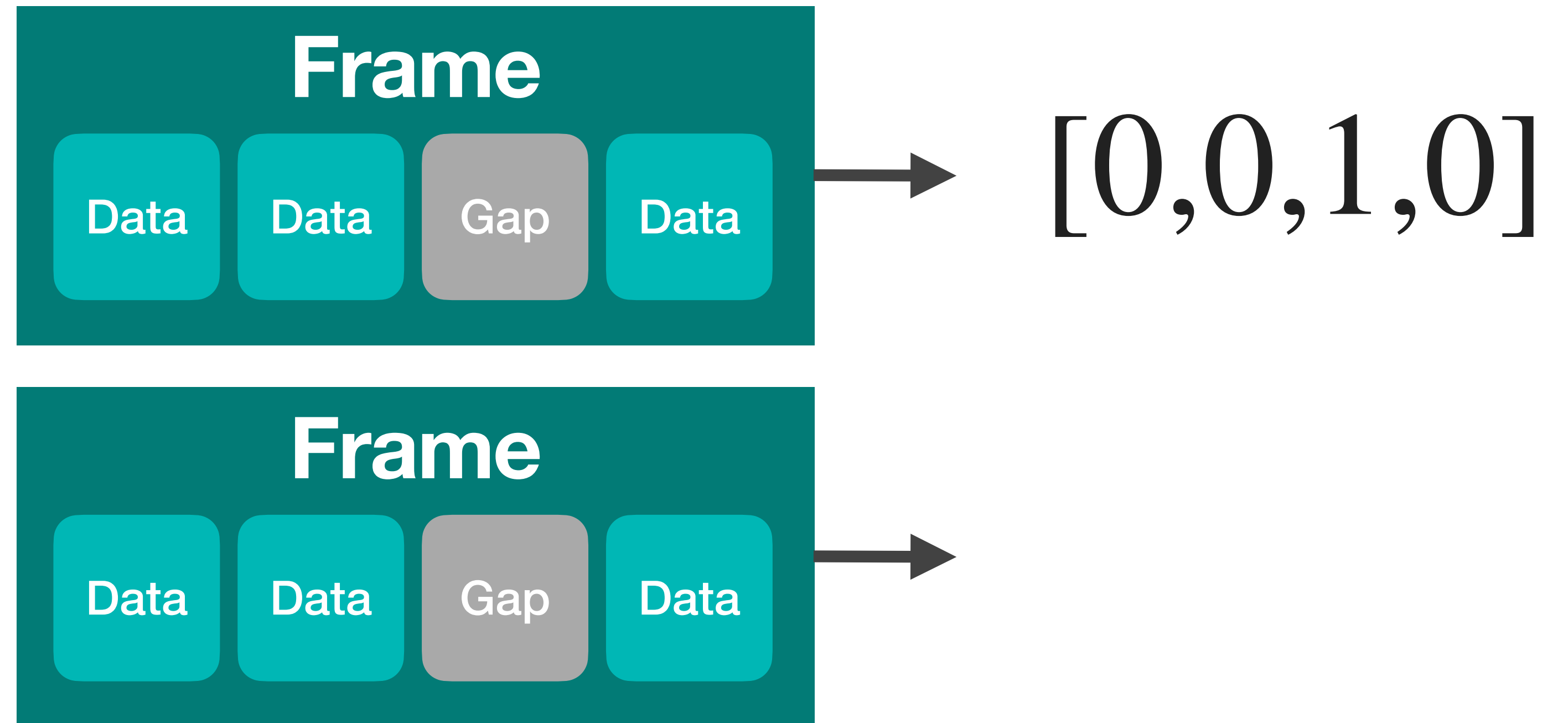


[0,0,1,0]



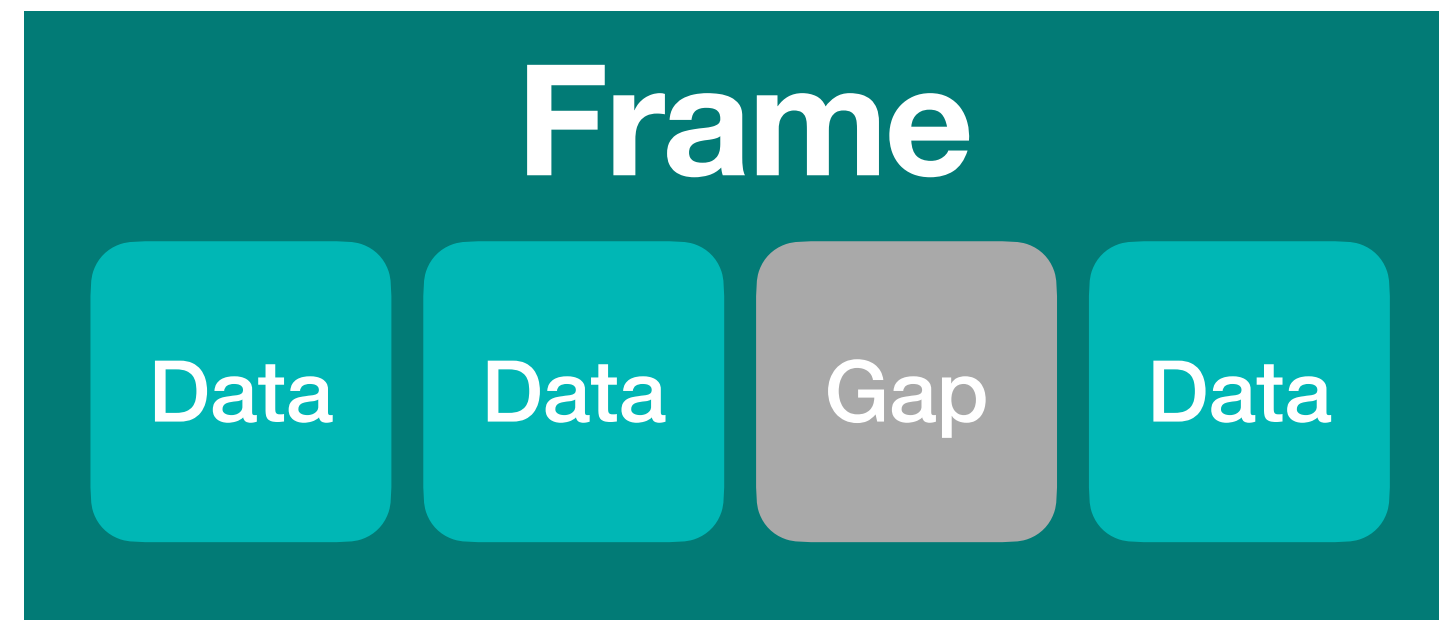
Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask
- maskeddata
 - Convert frame to masked array

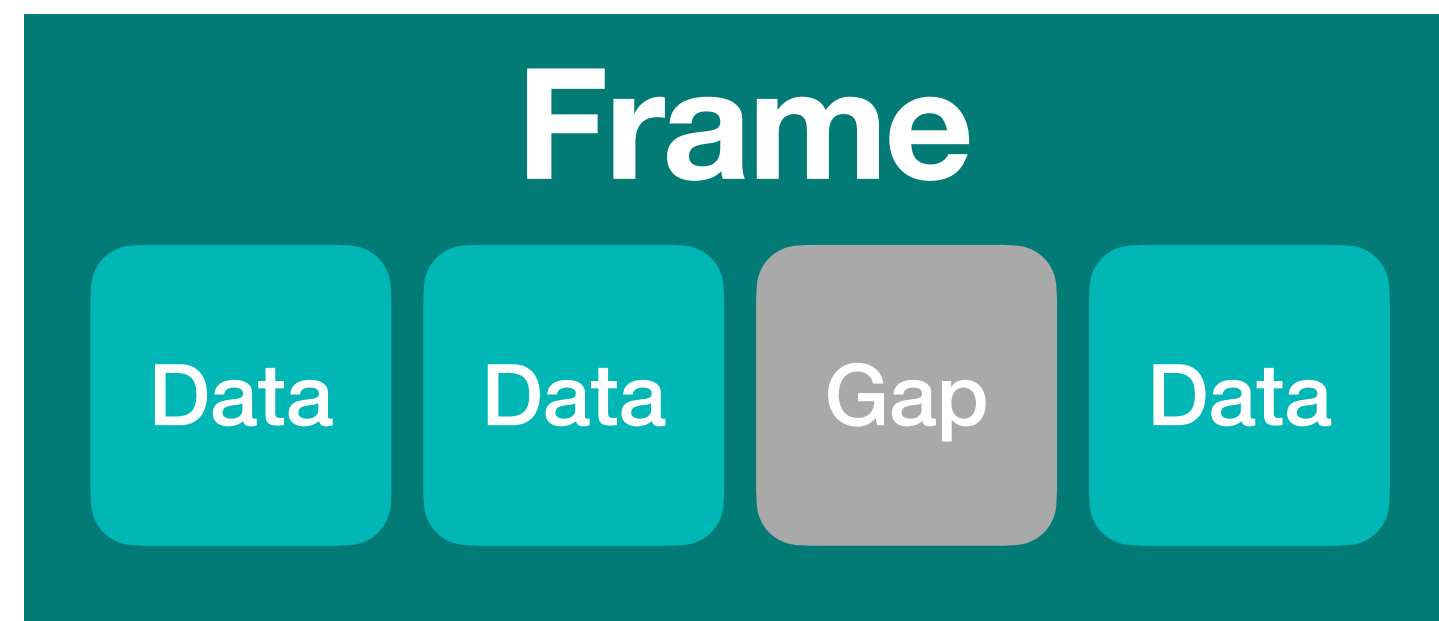


Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask
- maskeddata
 - Convert frame to masked array



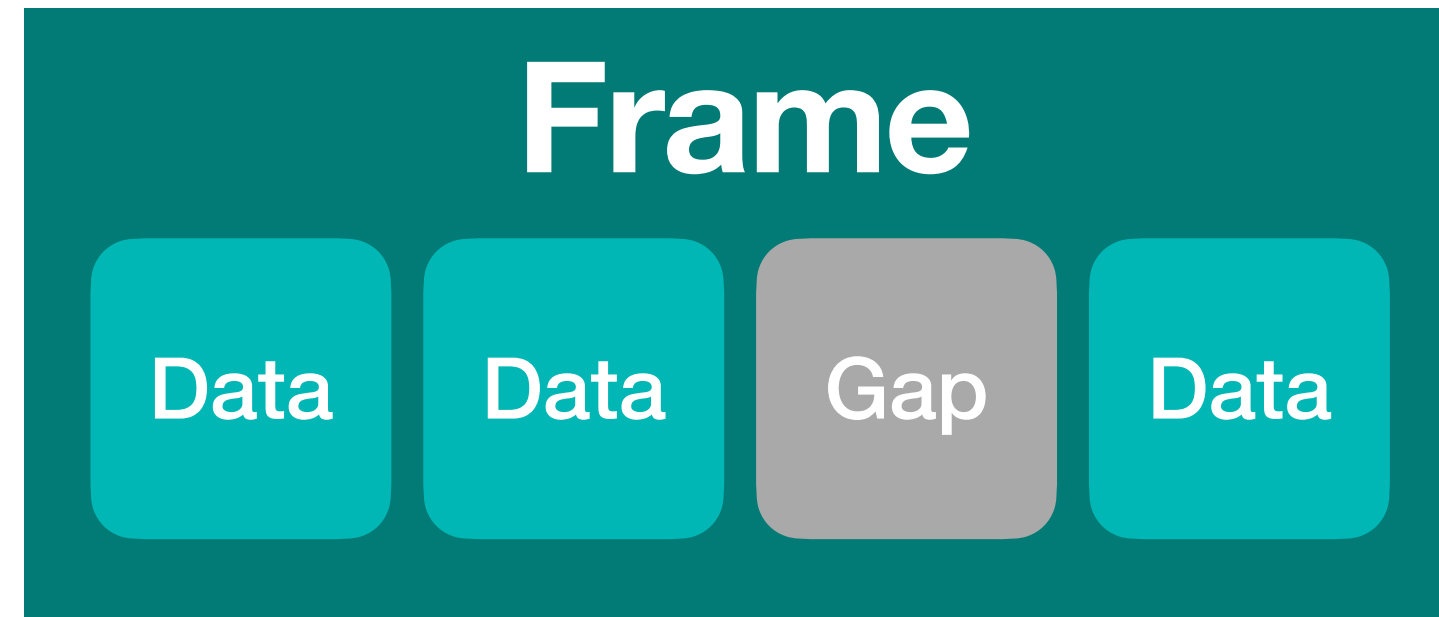
$[0,0,1,0]$



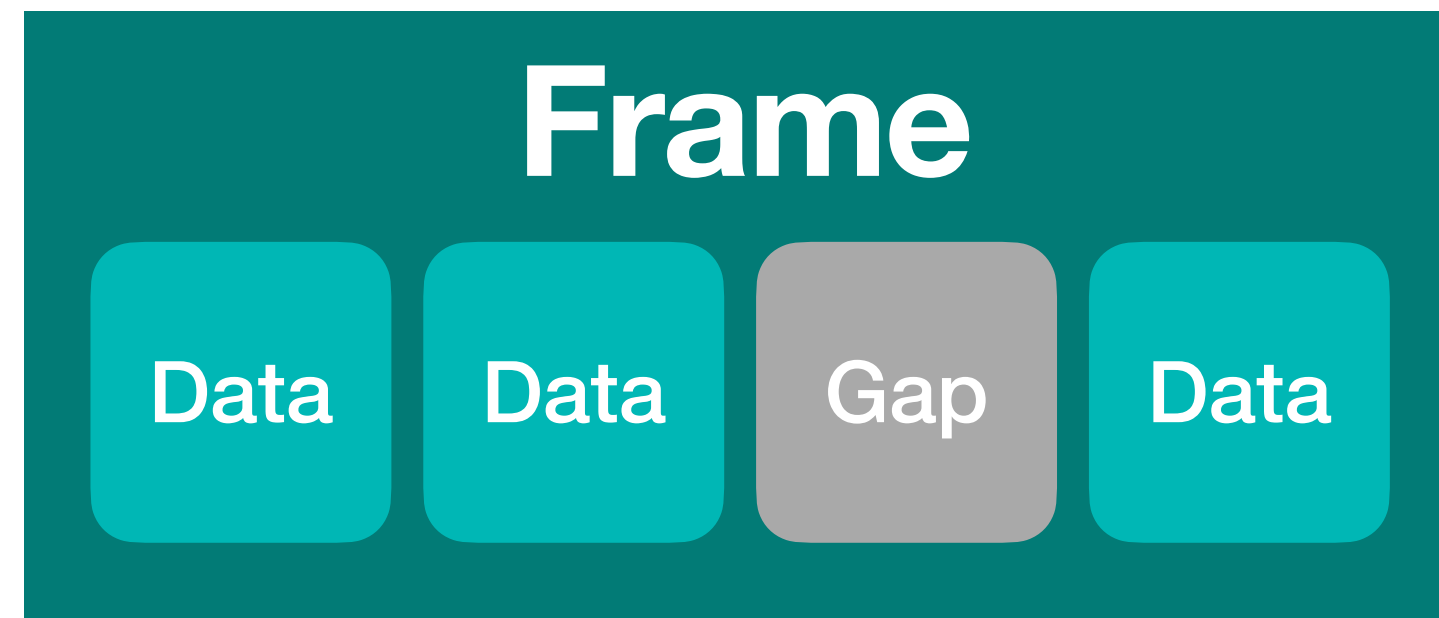
$[d, d, -, d]$

Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask
- maskeddata
 - Convert frame to masked array
- frame_from_ma
 - Convert masked array to frame



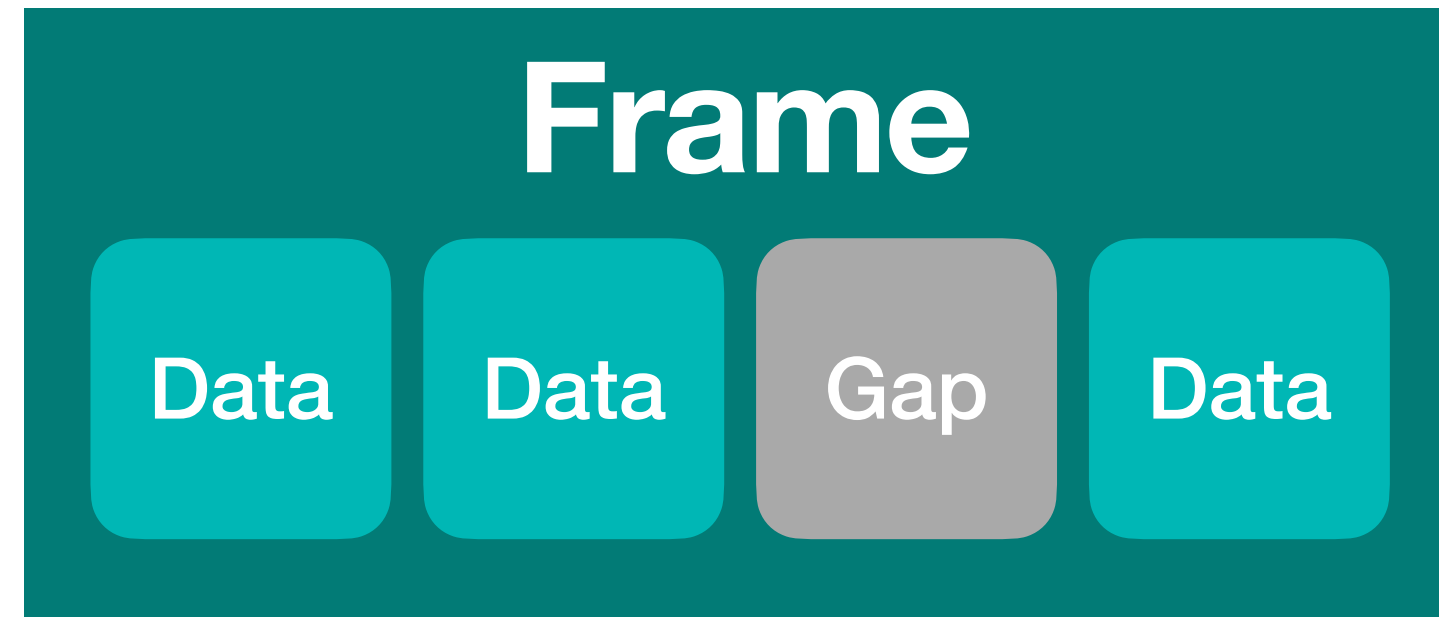
$[0,0,1,0]$



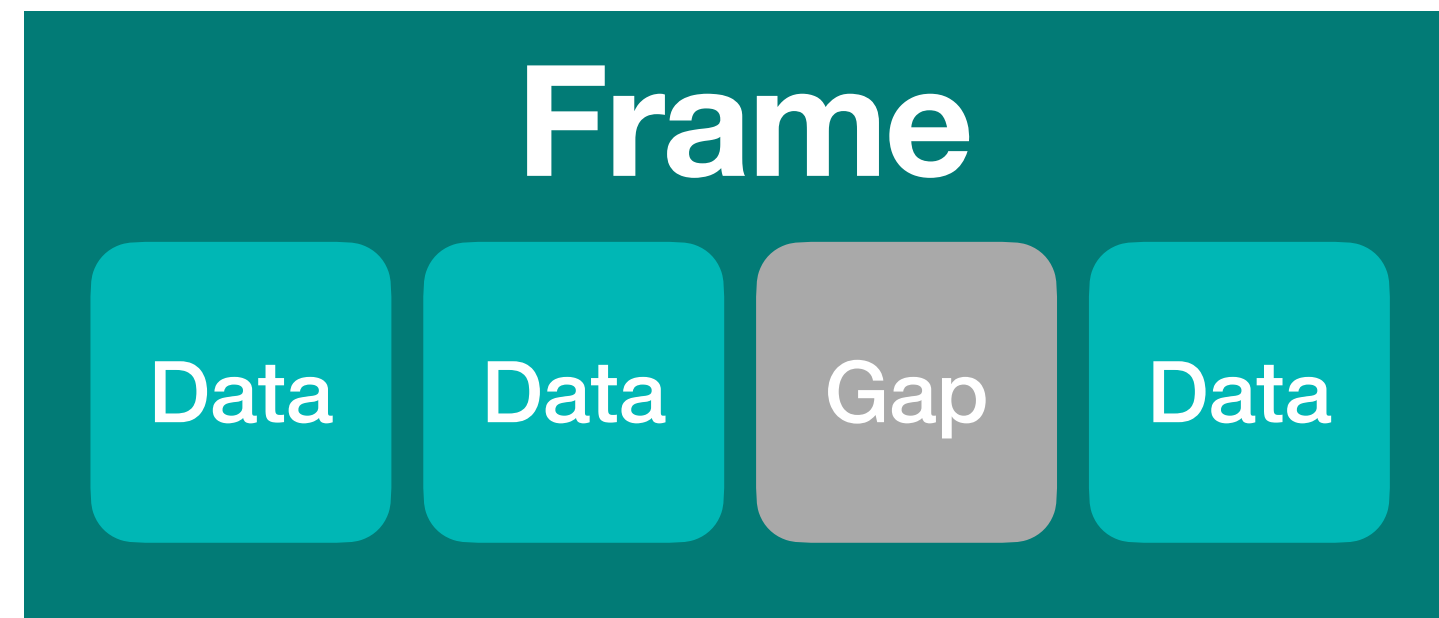
$[d, d, -, d]$

Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask
- maskeddata
 - Convert frame to masked array
- frame_from_ma
 - Convert masked array to frame



$[0,0,1,0]$



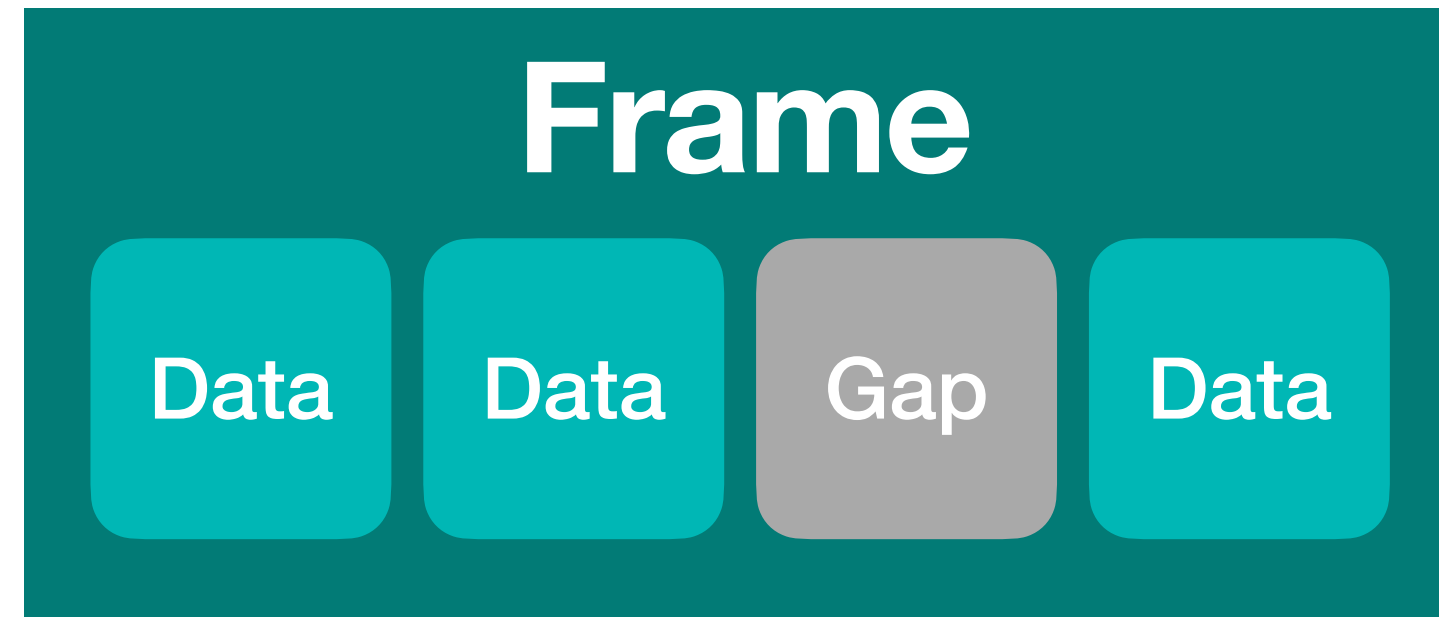
$[d, d, -, d]$

$[d, d, -, d]$

Added Frame Methods

- gap_mask

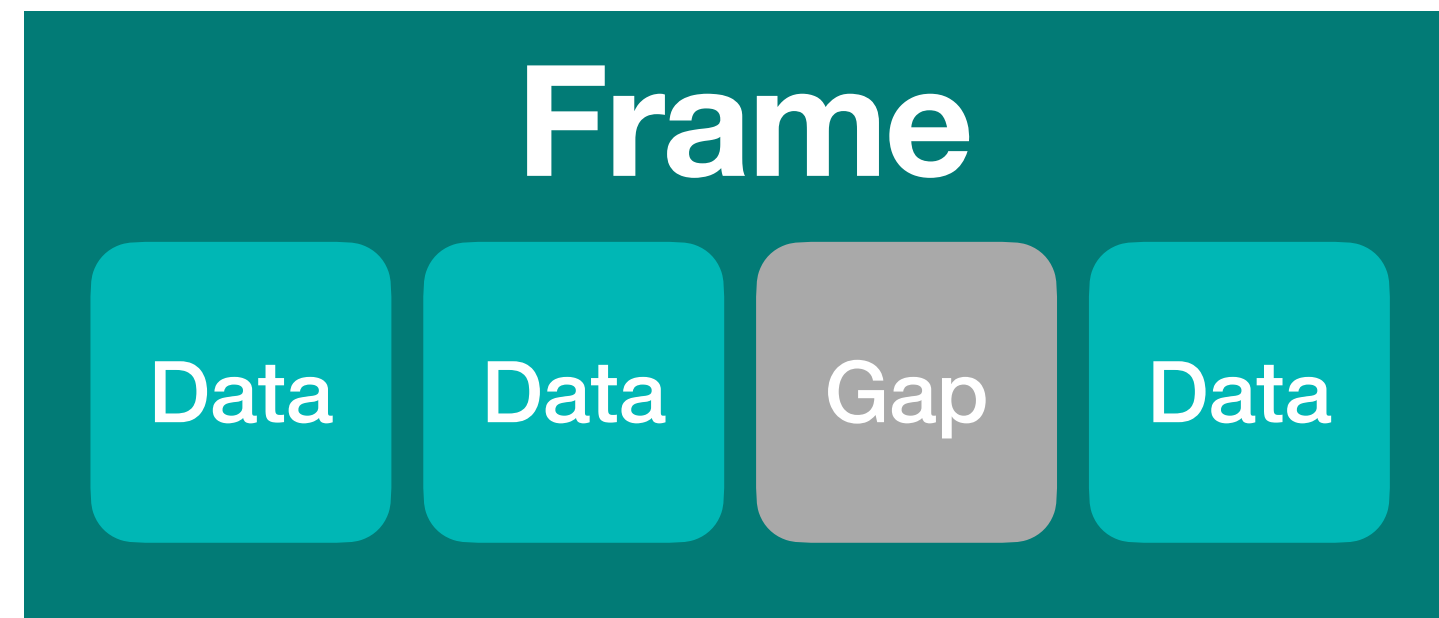
- Send in frame, get its gap mask



$[0,0,1,0]$

- maskeddata

- Convert frame to masked array



$[d, d, -, d]$

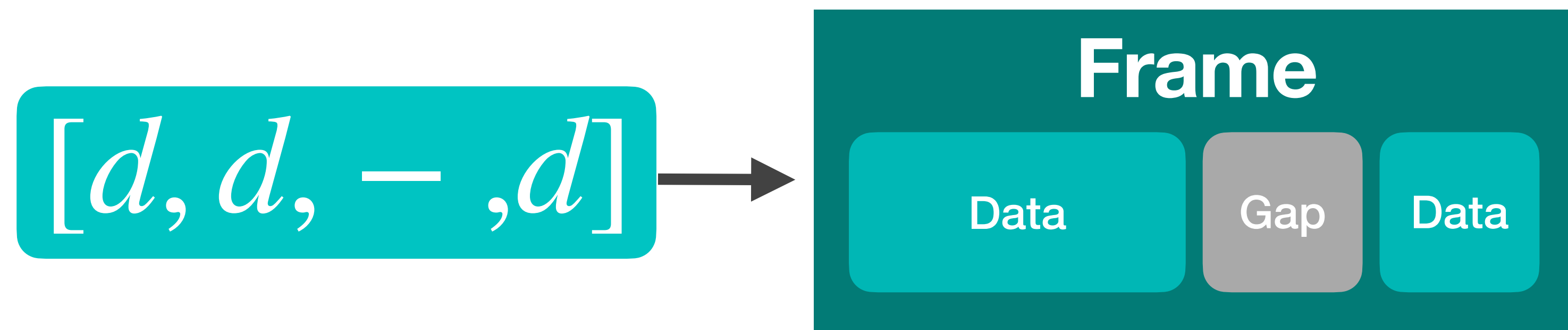
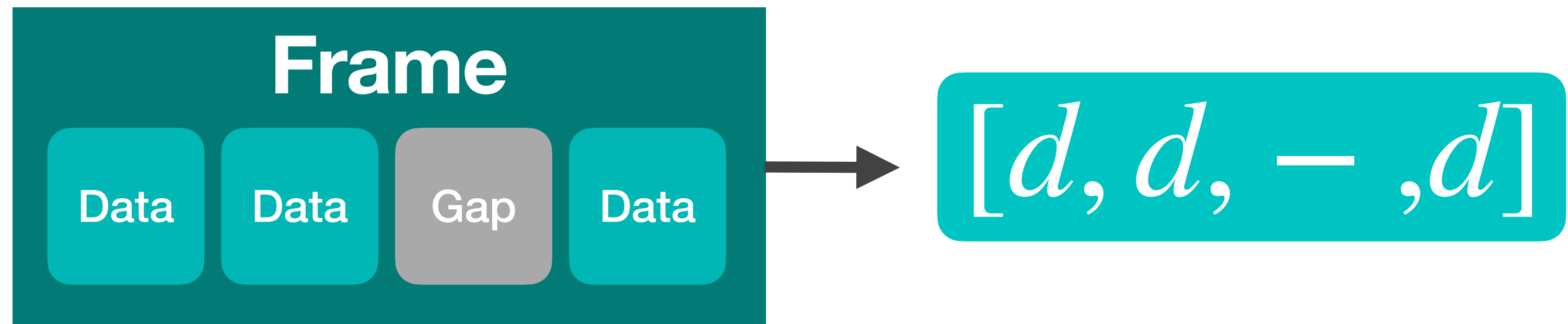
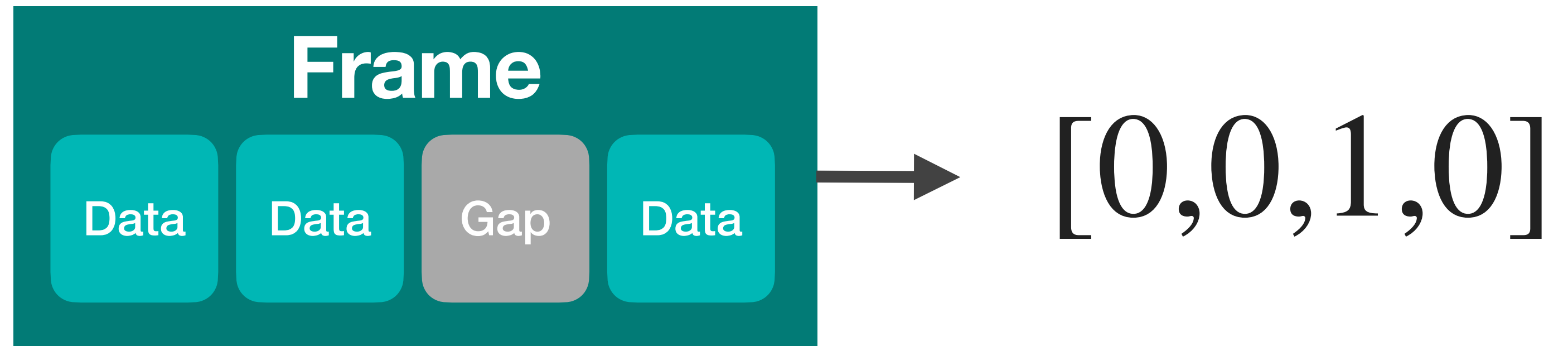
- frame_from_ma

- Convert masked array to frame

$[d, d, -, d]$

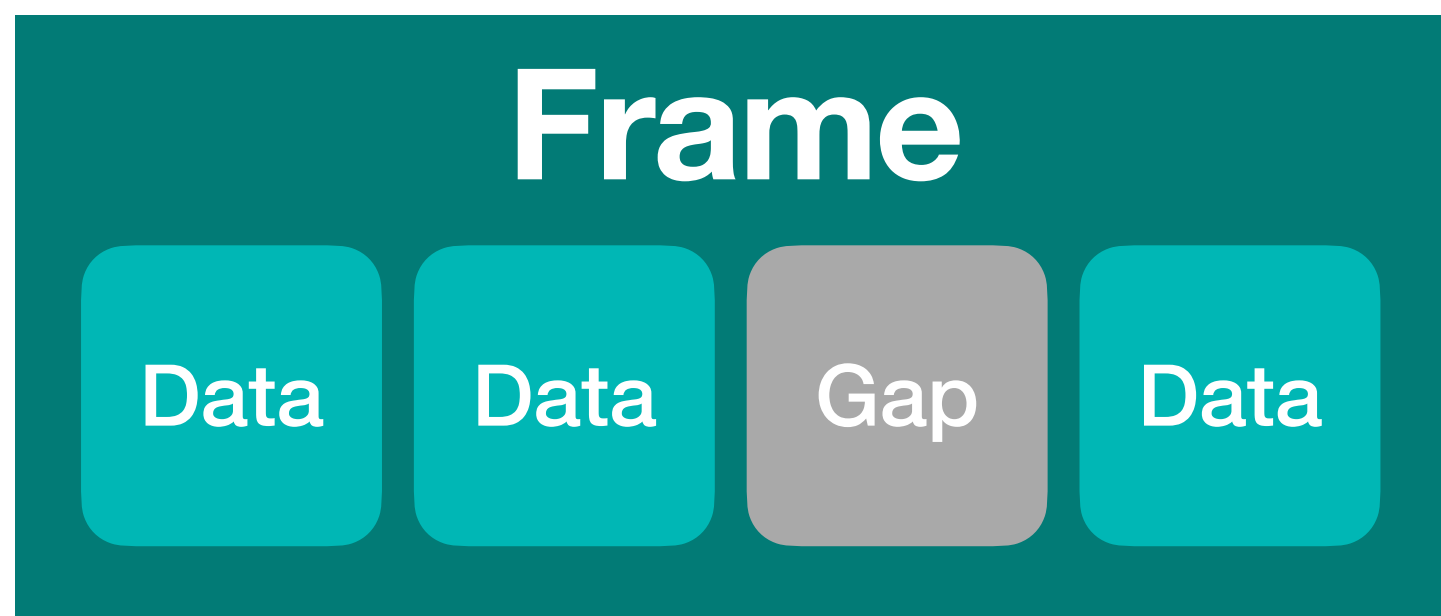
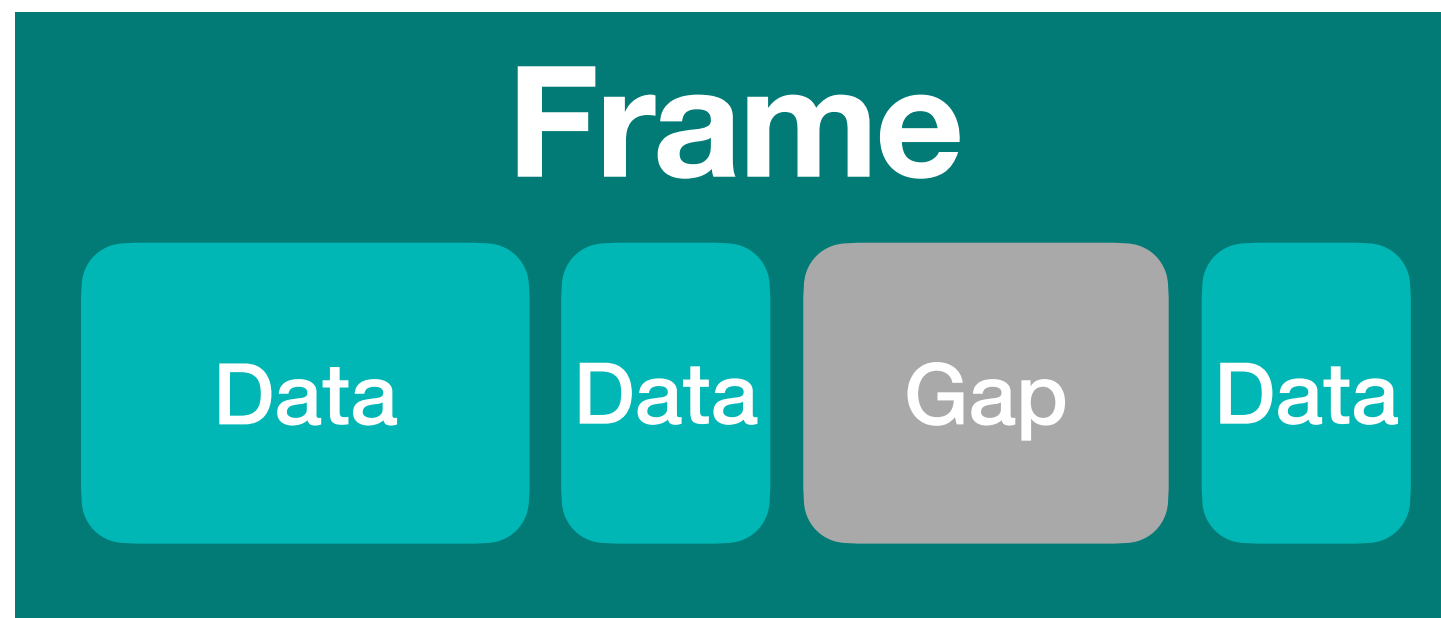
Added Frame Methods

- gap_mask
 - Send in frame, get its gap mask
- maskeddata
 - Convert frame to masked array
- frame_from_ma
 - Convert masked array to frame



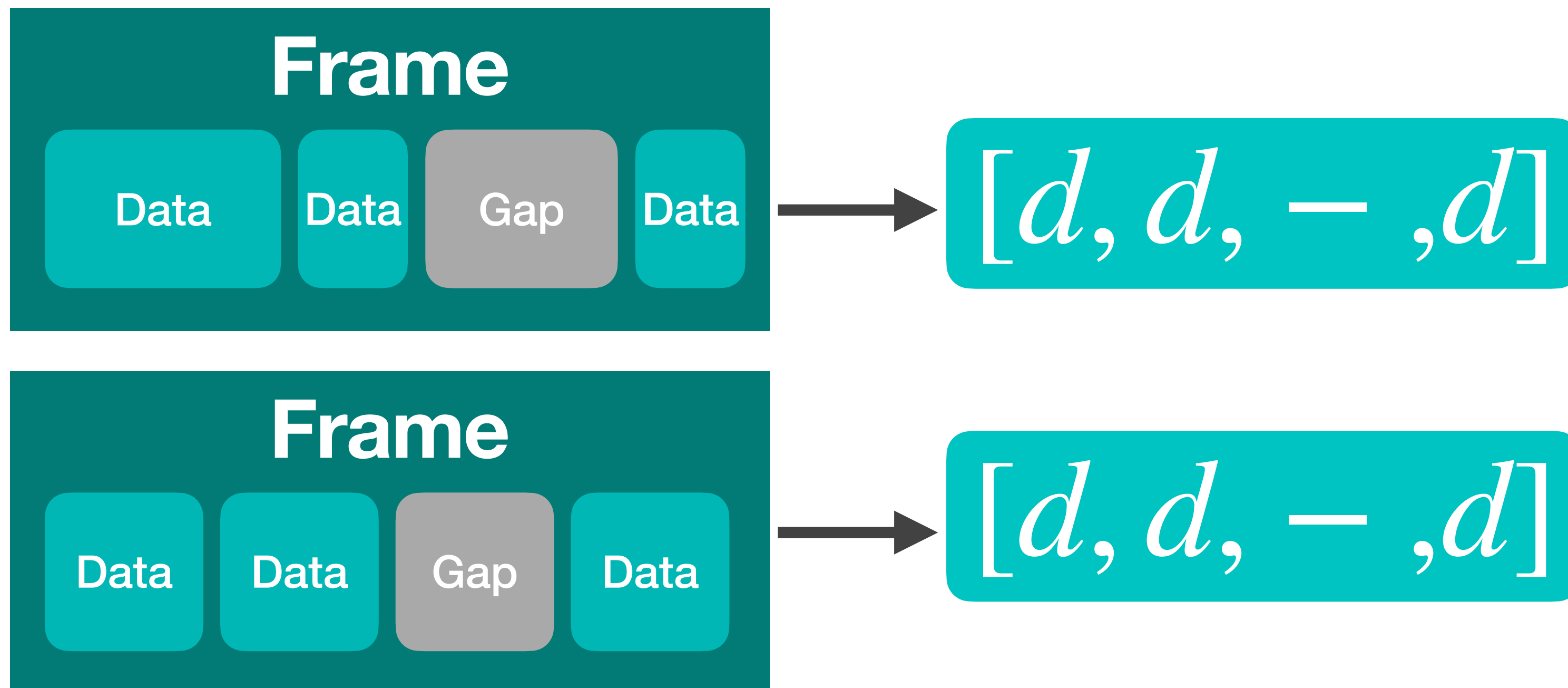
Added Frame Methods

- These methods are helpful for processing on frame level, rather than buffer level



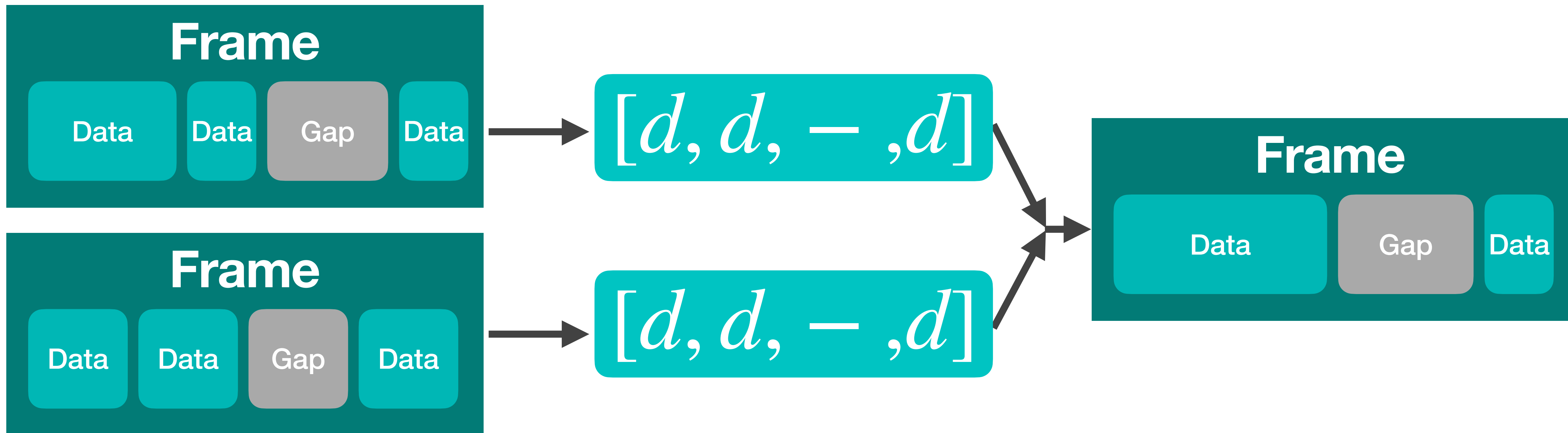
Added Frame Methods

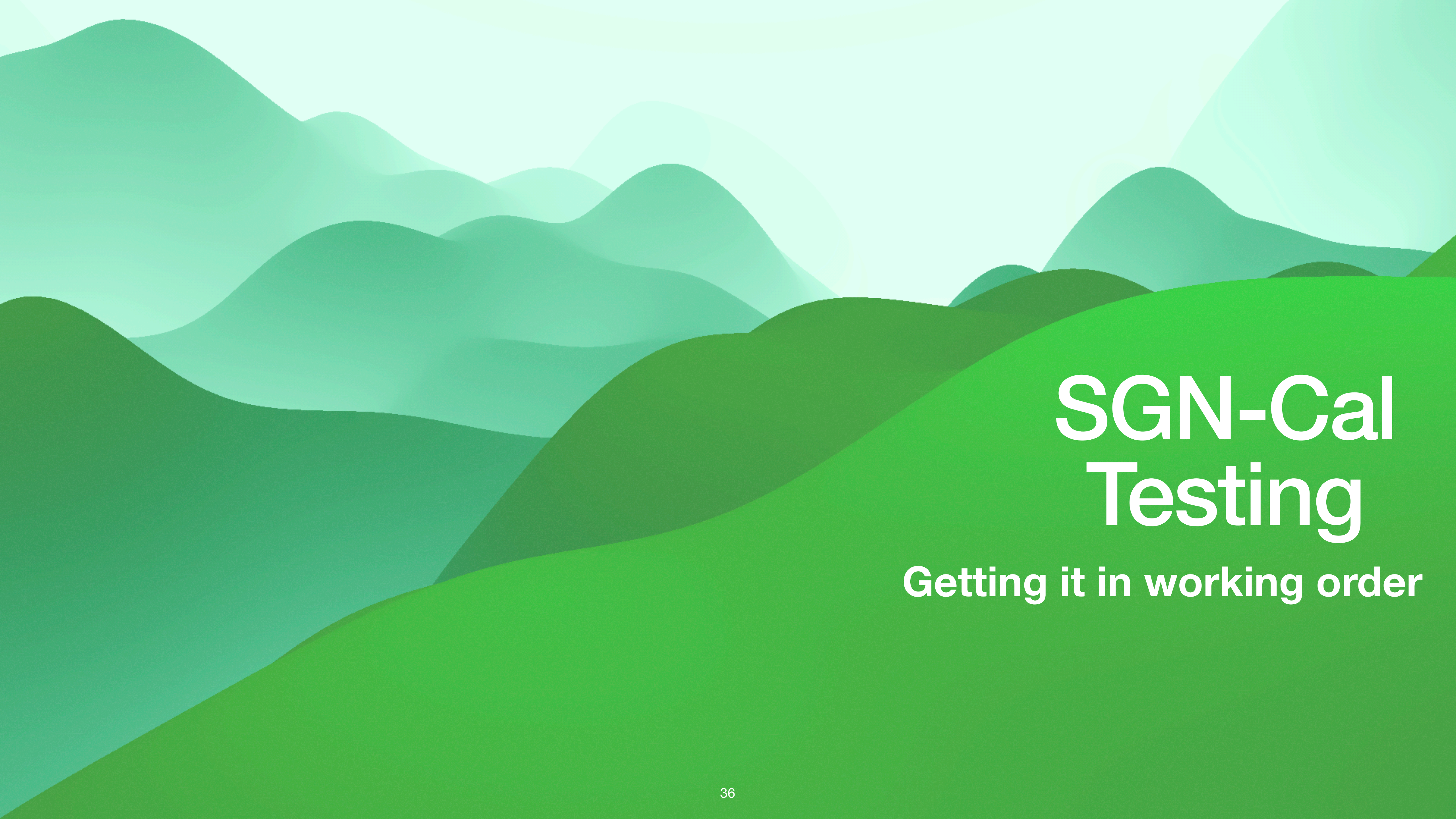
- These methods are helpful for processing on frame level, rather than buffer level



Added Frame Methods

- These methods are helpful for processing on frame level, rather than buffer level



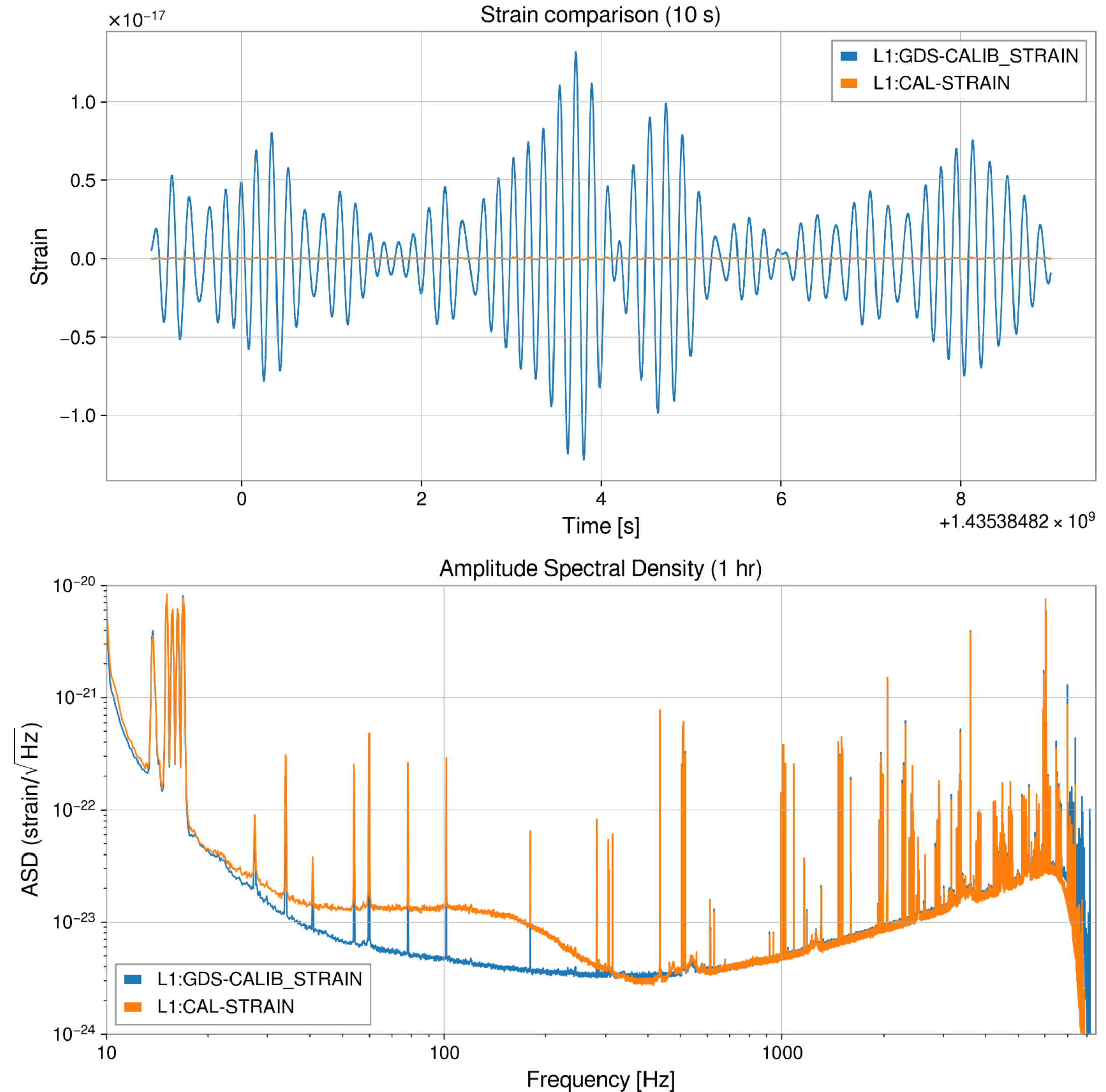
The background of the slide features a series of overlapping, rounded hills in various shades of green, ranging from light mint to dark forest green. The hills are layered to create a sense of depth and perspective, with the foreground hills being darker and more prominent, while the background hills are lighter and more ethereal. The overall aesthetic is clean, modern, and nature-inspired.

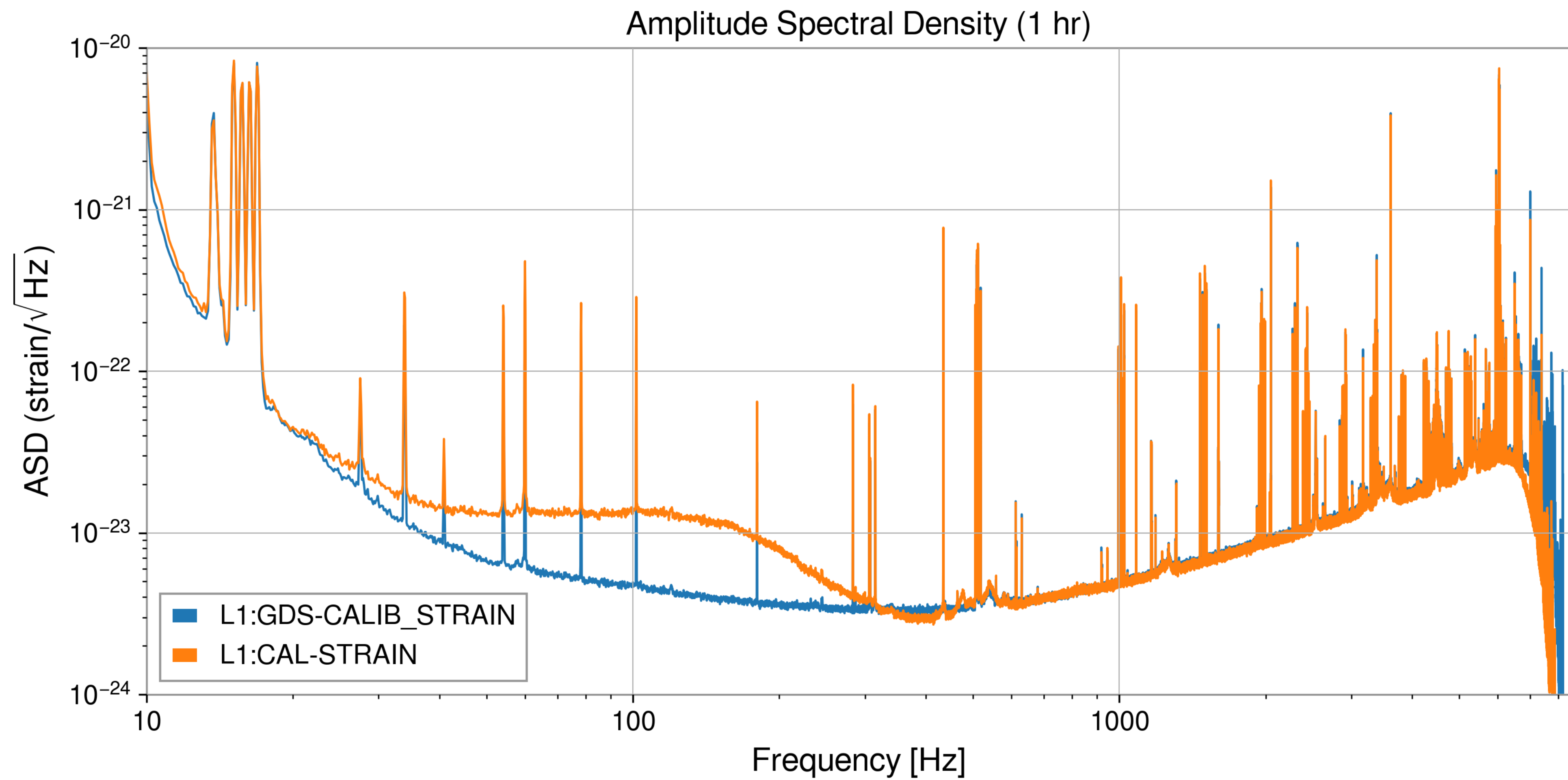
SGN-Cal Testing

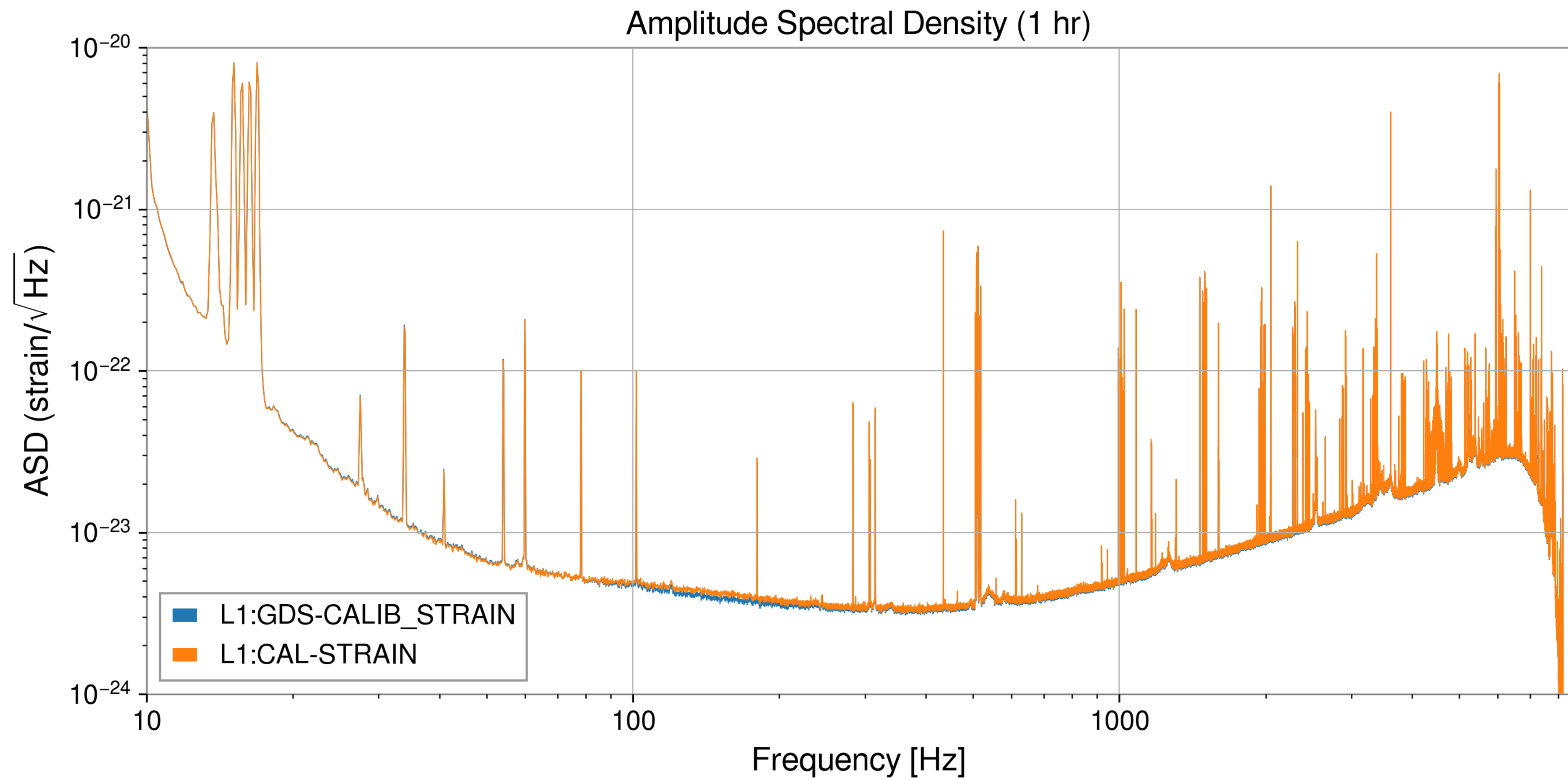
Getting it in working order

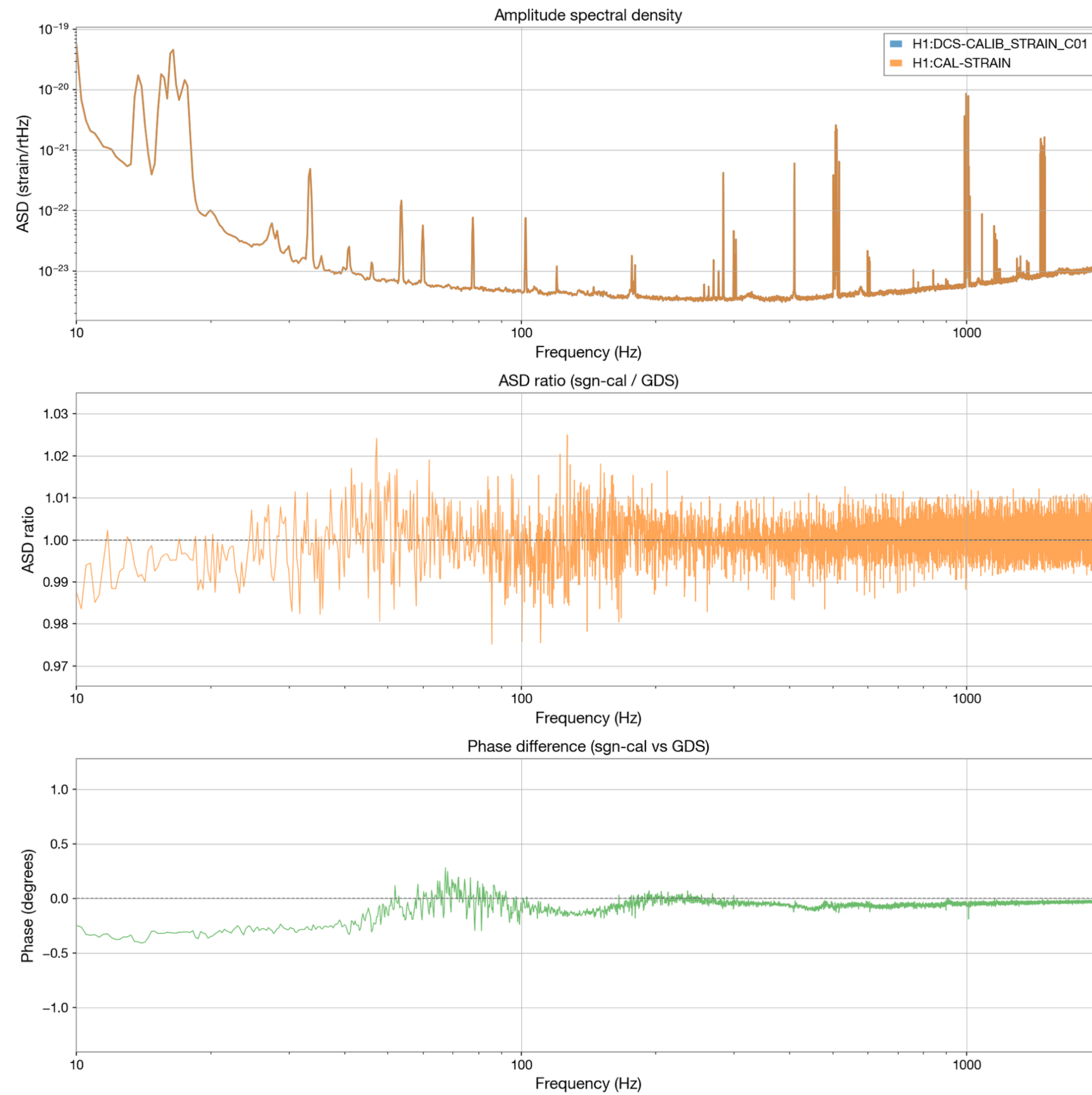
The pipeline is running

- While I was here we (Jamie and Maddie) got the SGN-Cal pipeline assembled, and running on real data! 🎉🎉🎉🎉🎉
- But... it still needed some tweaking
 - Substantial differences with gstlal
- Maddie and I tested the pipeline to track down sources of difference
 - Some bugs
 - Some differences in calibration config











Script for Writing Frames

Testing SGN-Cal

- Archived raw frames have X channels
- We only need Y channels for calibration pipeline
- It would be nice to have smaller frames for testing!

```
L1:OAF-CLEANING_NOISE_EST_DBL_DQ 16384
L1:CAL-INJ_STATUS_OUT_DQ 16384
L1:CAL-CS_TDEP_SUS_LINE1_UNCERTAINTY 16
L1:CAL-CS_TDEP_PCAL_LINE3_UNCERTAINTY 16
L1:CAL-DELTA_REF_PCAL_DQ 16384
L1:SUS-ETMX_L1_CAL_LINE_OUT_DQ 512
L1:SUS-ETMX_L2_CAL_LINE_OUT_DQ 512
L1:SUS-ETMX_L3_CAL_LINE_OUT_DQ 512
L1:SUS-ETMY_L3_CAL_LINE_OUT_DQ 512
L1:SUS-ETMY_L3_CAL_LINE_OUT_DQ 512
L1:SUS-ETMY_L3_CAL_LINE_OUT_DQ 512
L1:SUS-ETMY_L3_CAL_LINE_OUT_DQ 512
L1:CAL-PCALY_RX_PD_OUT_DQ 16384
L1:CAL-PCALX_RX_PD_OUT_DQ 16384
L1:CAL-PCALX_PCALOSC1_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC2_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC3_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC4_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC5_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC6_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC7_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC8_OSC_FREQ 16
L1:CAL-PCALX_PCALOSC9_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC1_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC2_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC3_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC4_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC5_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC6_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC7_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC8_OSC_FREQ 16
L1:CAL-PCALY_PCALOSC9_OSC_FREQ 16
```

Making Frame Files

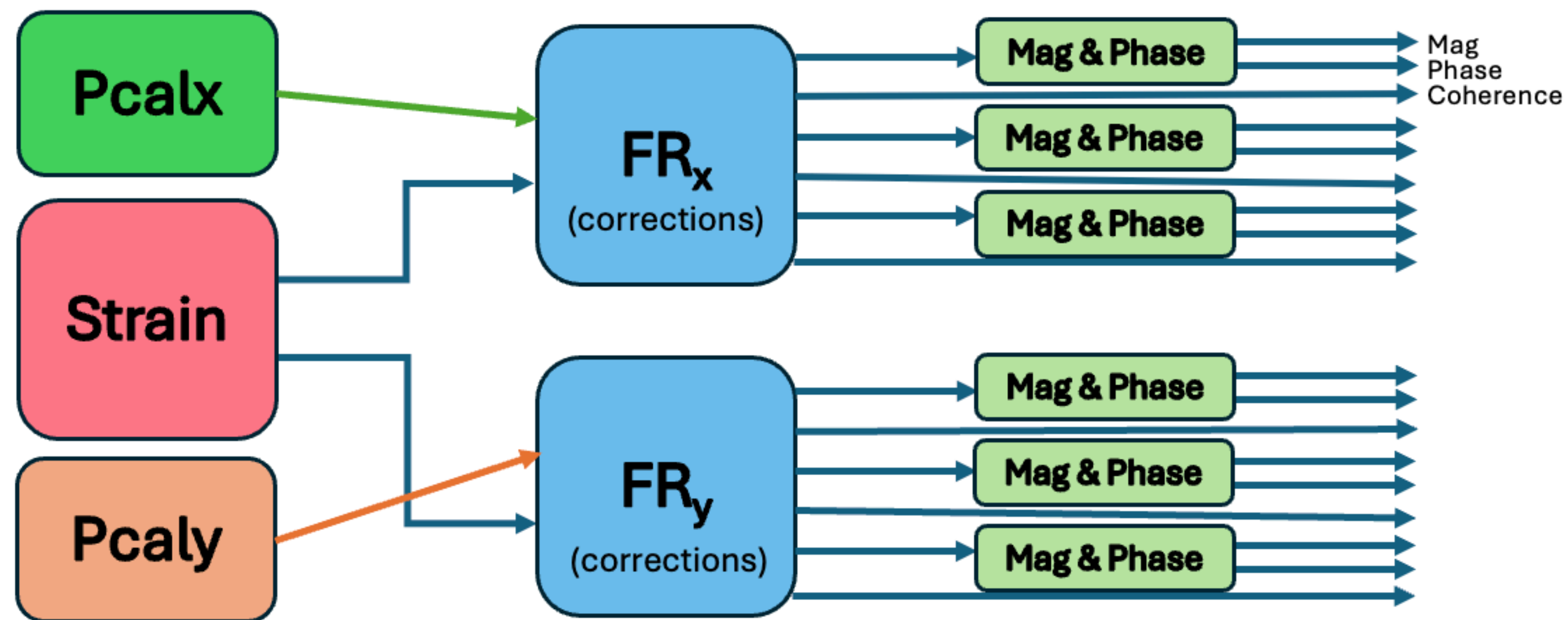
- I made a script that makes custom raw frames files!
 - Uses gwpy to retrieve data and store as frames
 - Only contains selected channels
 - Customizable
 - Length
 - Duration
 - File type

The background of the slide features a series of overlapping, rounded hills in various shades of green, ranging from light mint to dark forest green. The hills are layered to create a sense of depth and perspective, with the foreground hills being the most prominent and darkest, and subsequent hills receding into the distance, becoming progressively lighter and more transparent. The overall aesthetic is clean, modern, and nature-inspired.

Calibration Comparison Tool

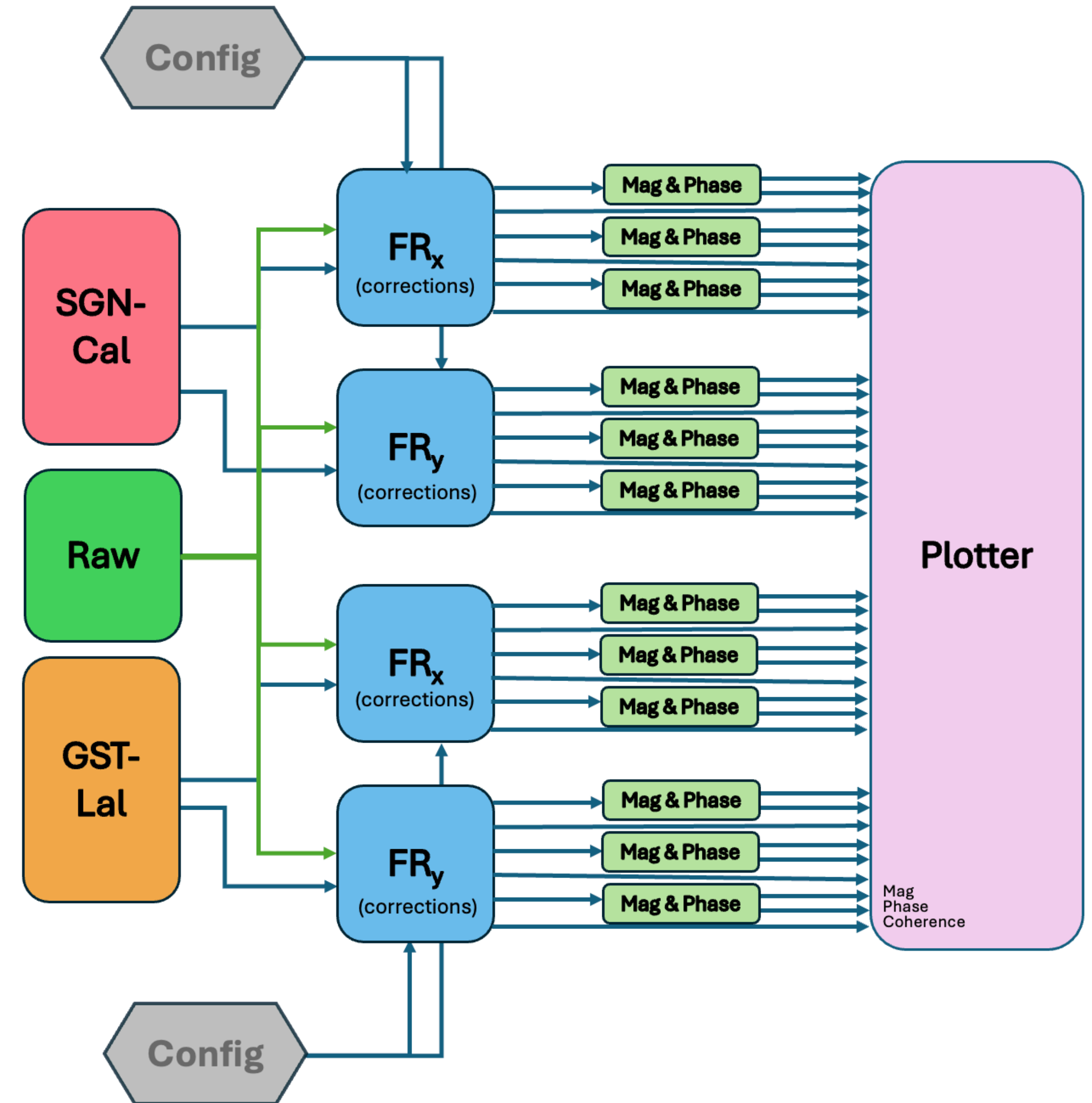
Checking Calibration

- Monitoring calculations
- We want to see magnitude, phase, and coherence of pcal/strain for each line

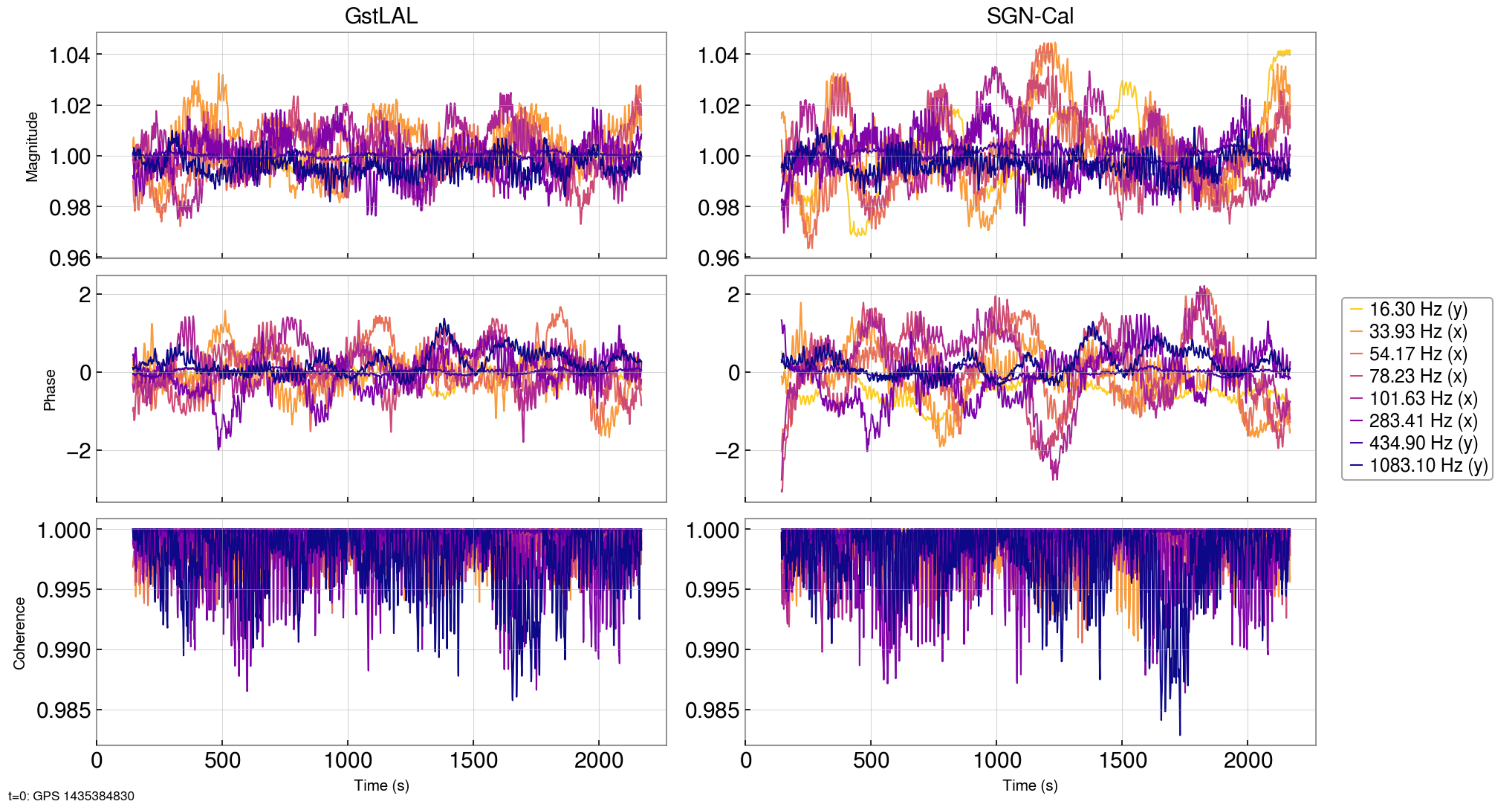


Comparing Calibration

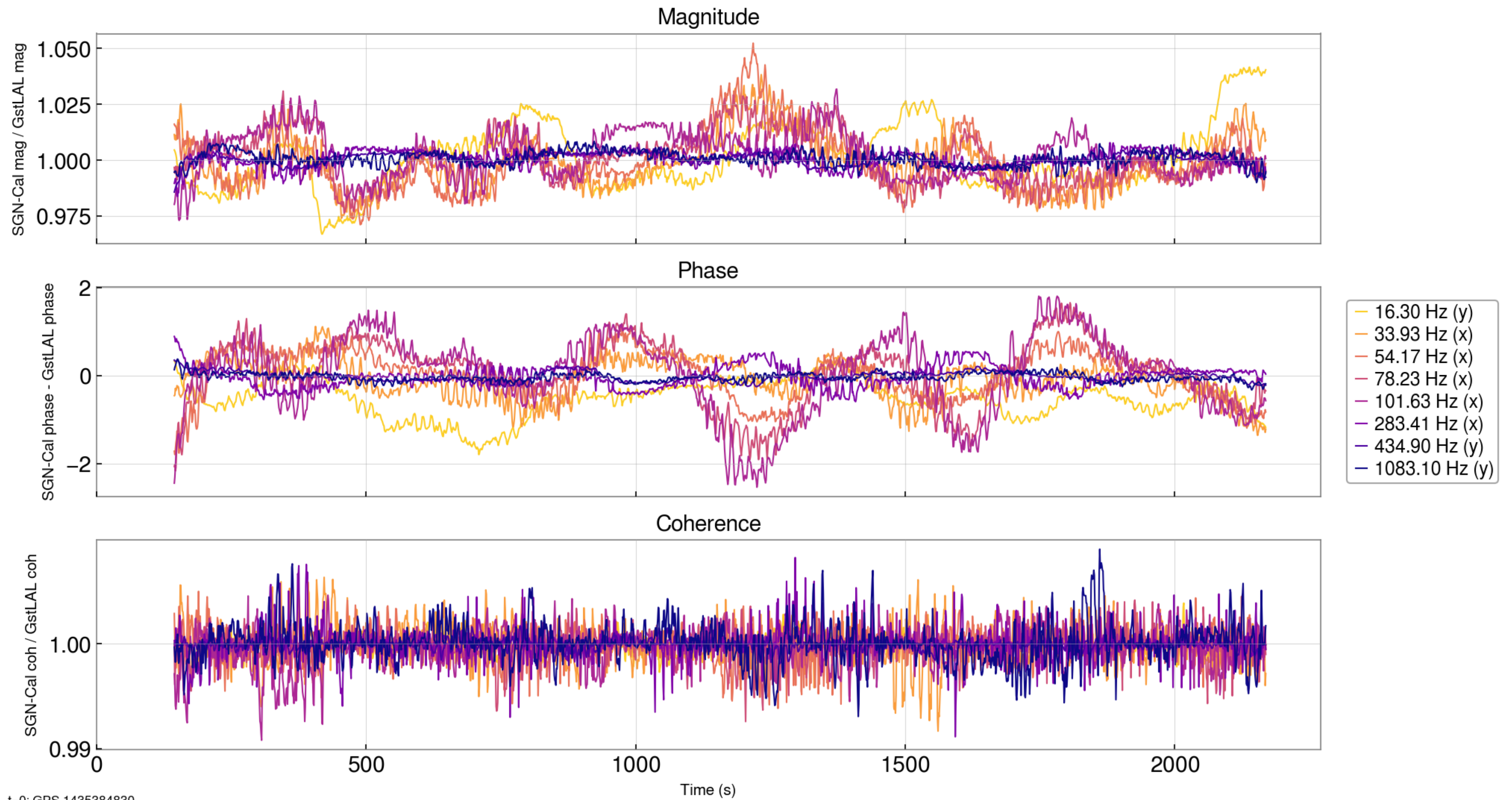
- Comparing calibration
 - We want to see whether the sgn-cal results are worse, the same as, or better than gstlal results
 - Absolute marker
- Comparison tool
 - Takes in frames from both GstLAL and SGN-Cal
 - Produce values for both
 - Compare quality of calibration



Calibration Comparison



SGN-Cal GstLAL comparison

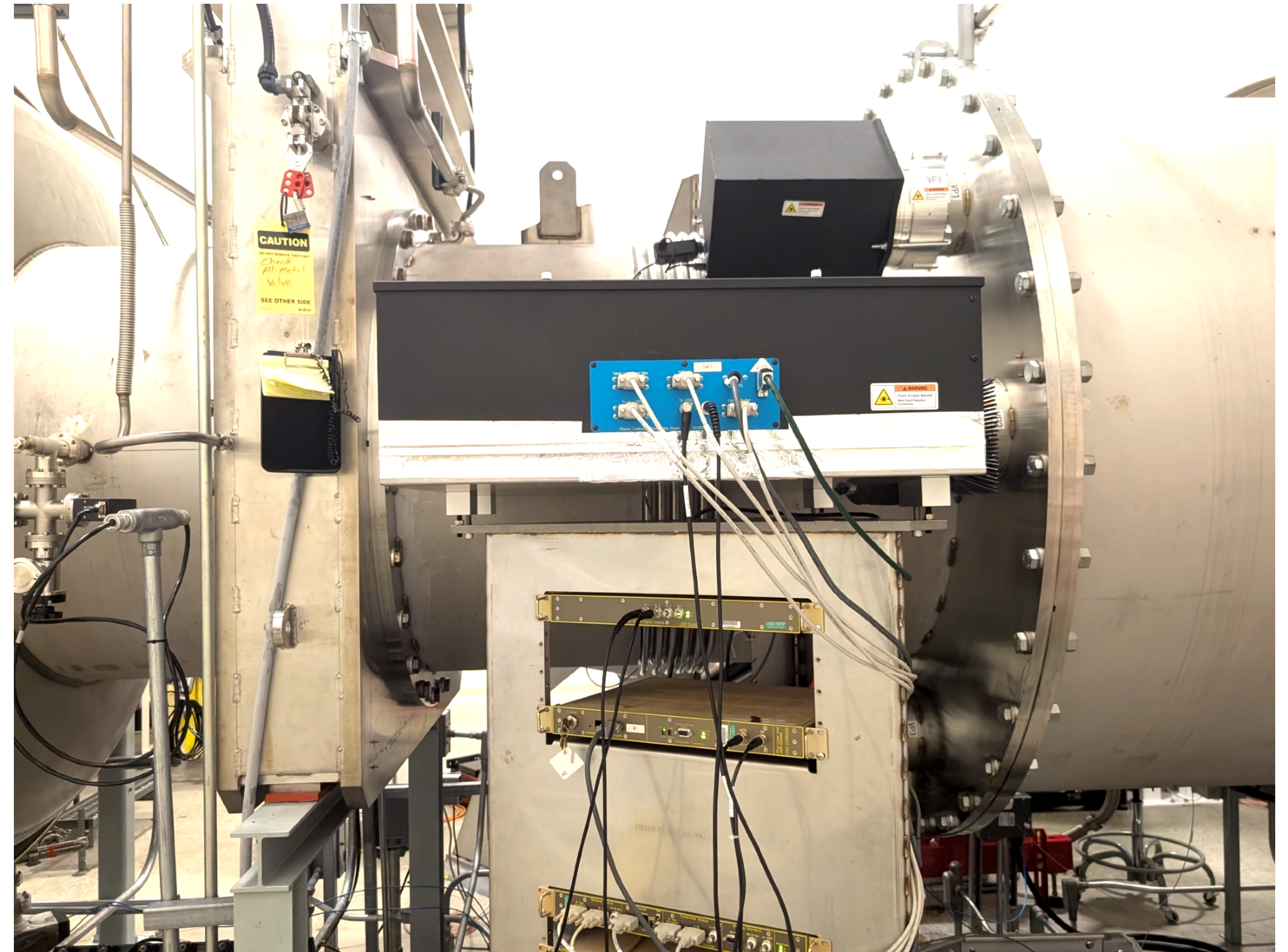


The background of the slide features a series of overlapping, rounded hills in various shades of green, ranging from light mint to dark forest green. The hills are layered to create a sense of depth and perspective, with the foreground hills being the most prominent and darkest, and the background hills being lighter and more faded. The overall aesthetic is clean, modern, and nature-inspired.

Pcal Laser Replacement

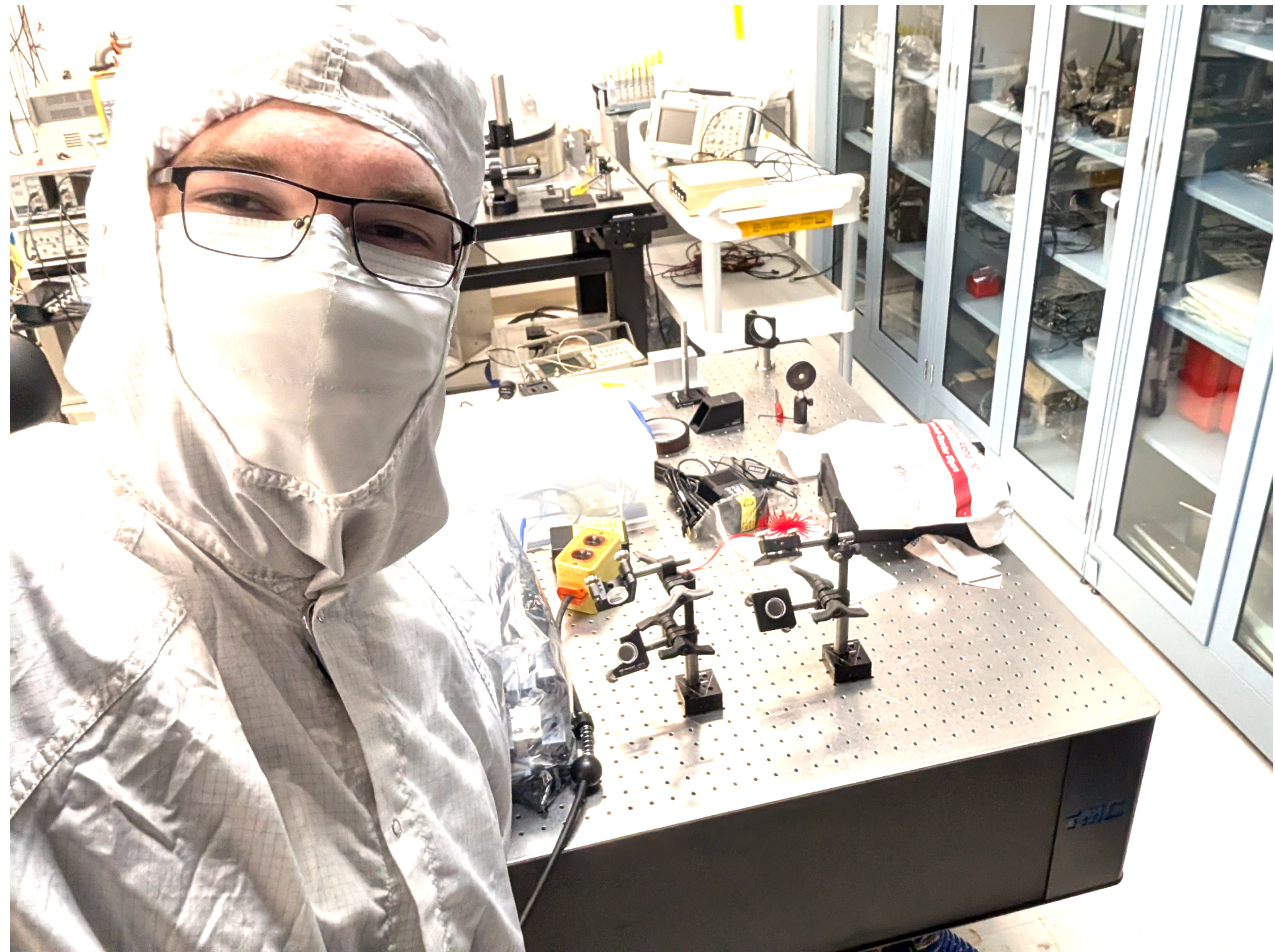
Pcal Laser Replacement

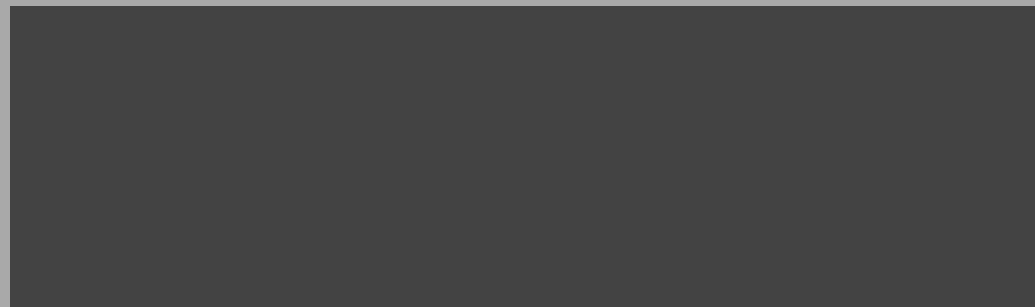
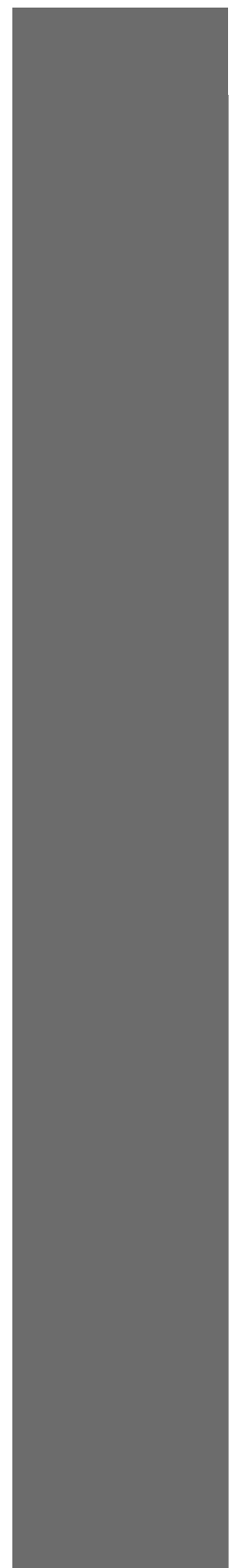
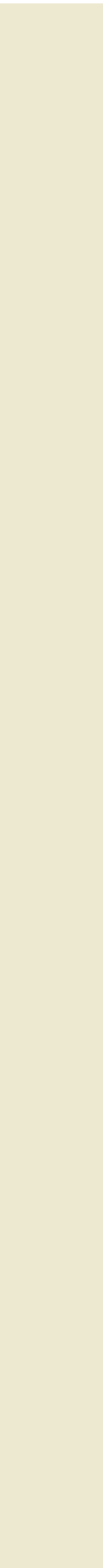
- Power of one of the Pcal lasers was low
- Joe wanted to swap it with an extra laser
- The plan:
 - Mark current laser position (very well)
 - Remove laser, install new one
 - Align with marked position

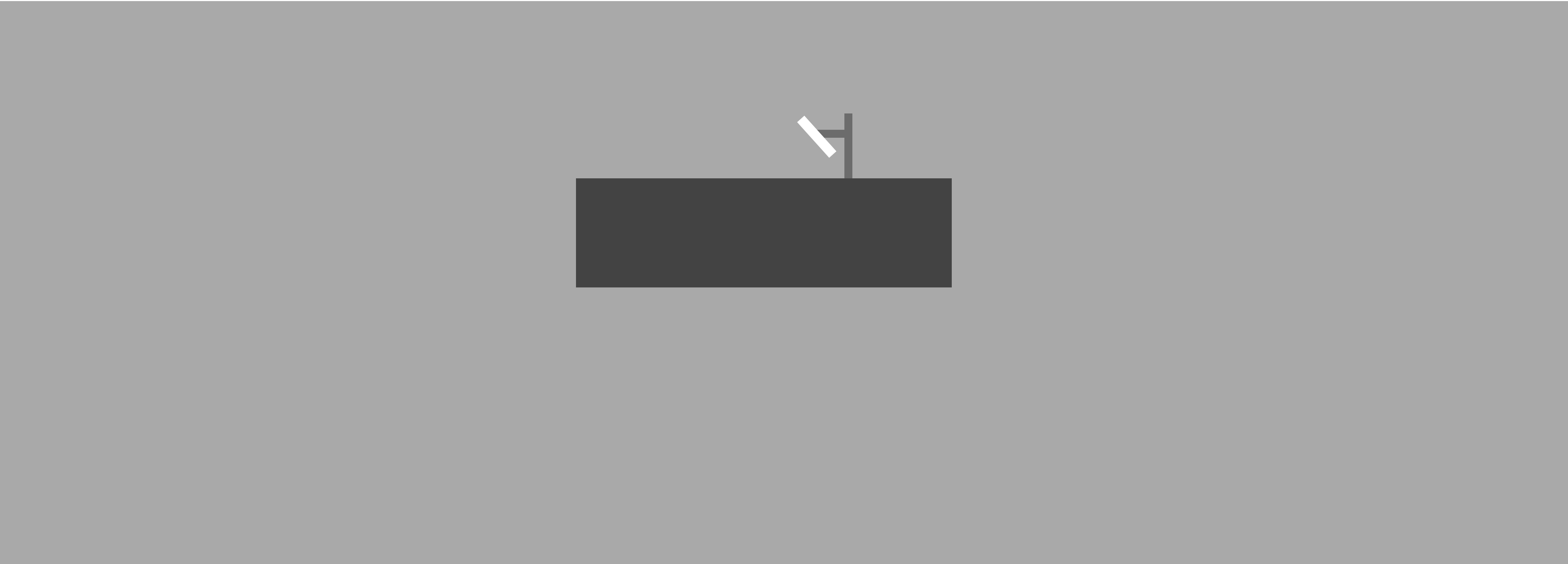
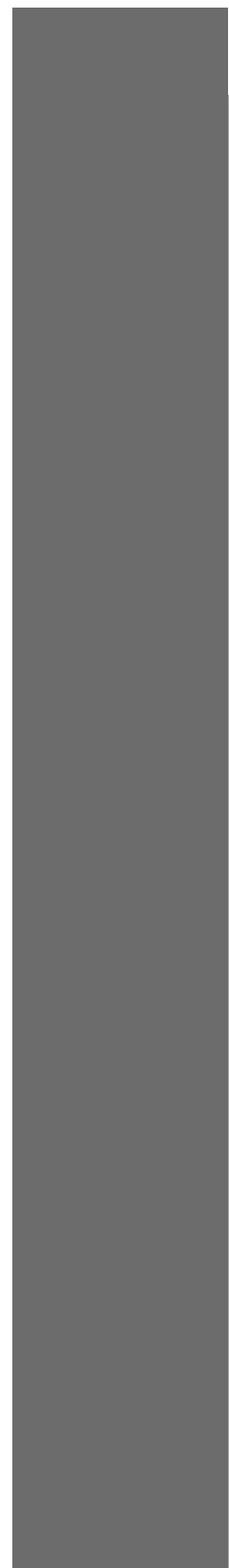
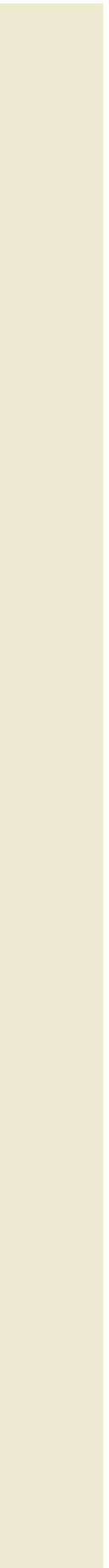


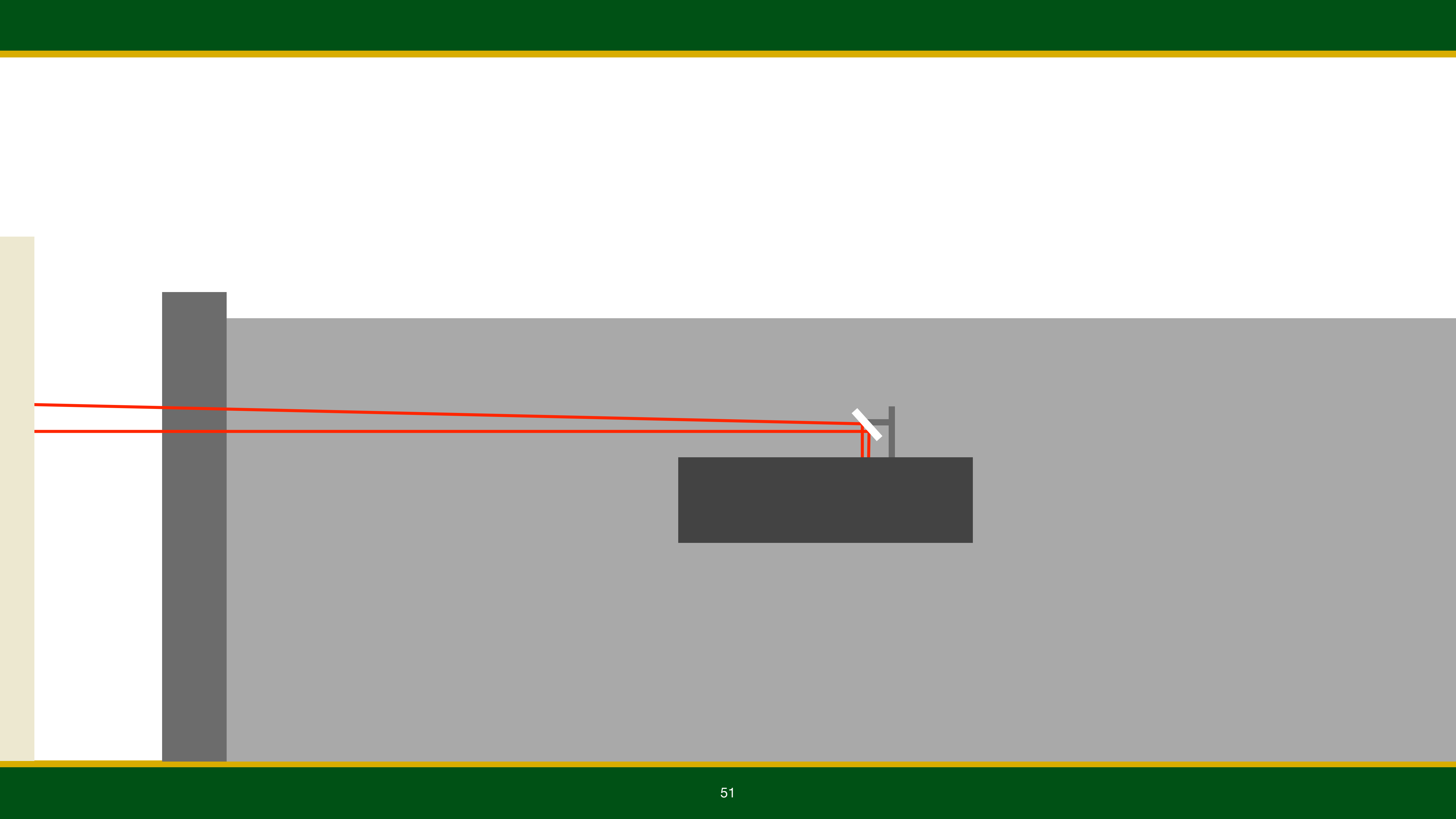
Periscopes

- Needed to point lasers at fixed point, for reference when aligning new laser
- Made two periscopes to get beams out of pcal box
- There were many tribulations









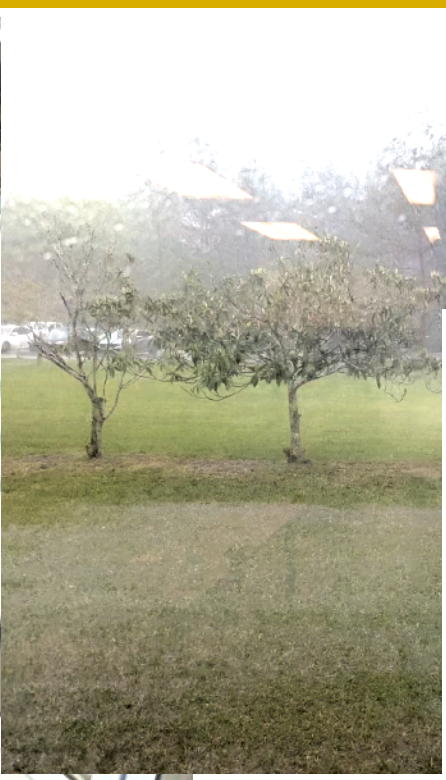
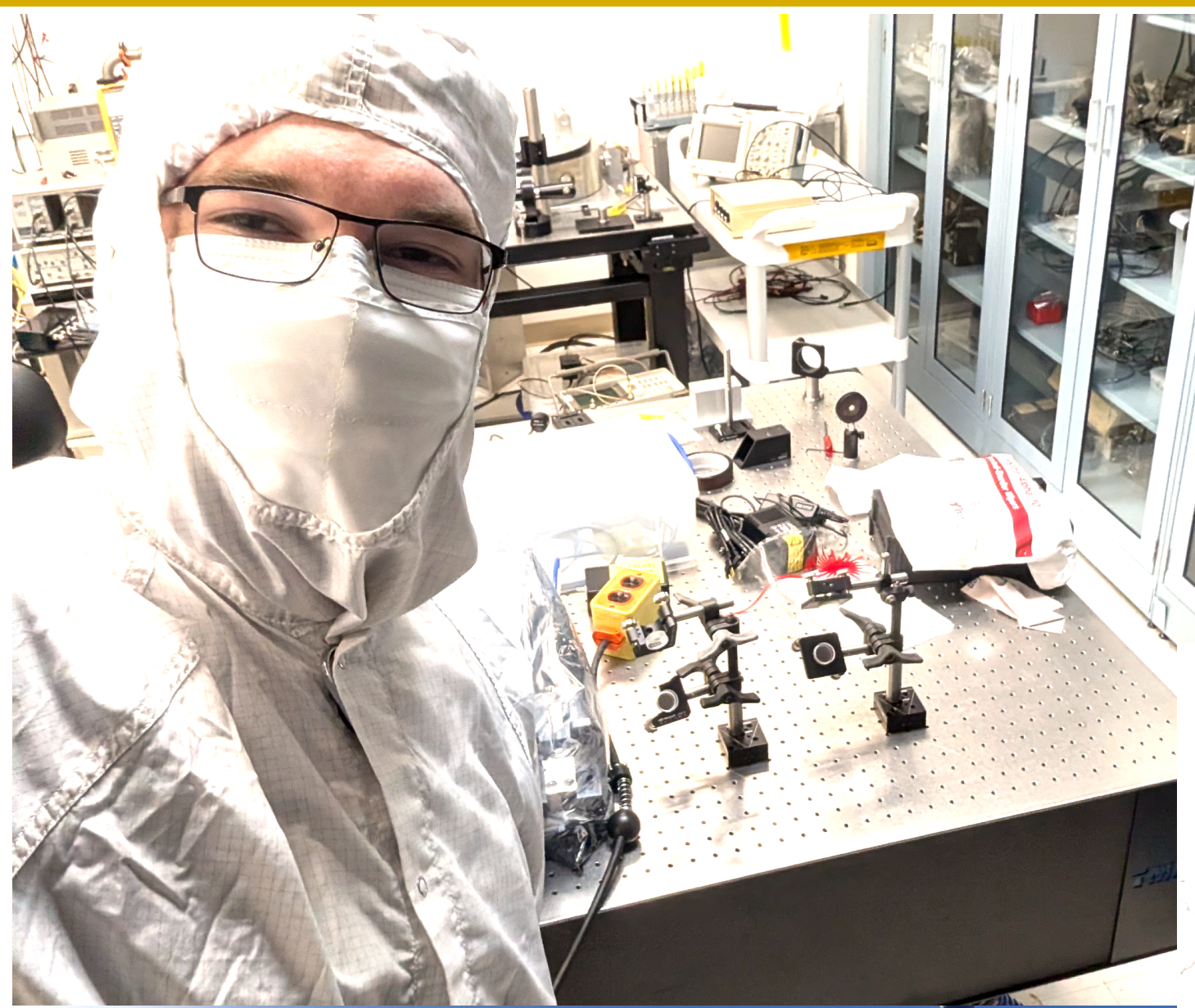
Summary

- SGN-Cal
 - Developing elements
 - Testing pipeline
 - Building tools
- SGN-CalMon
 - Developing elements
 - Building Tools
- SGN-TS
 - Created masked array methods
- Monitoring Alerts
 - Set up monitoring alert in control room
- Pcal
 - Helped with laser swap



Questions?

Bonus Slides



[Not Pictured:
me in vertex] (I
forgot to take
pictures)

