SCExAO as a prototype platform available to the GSMT extreme-AO community

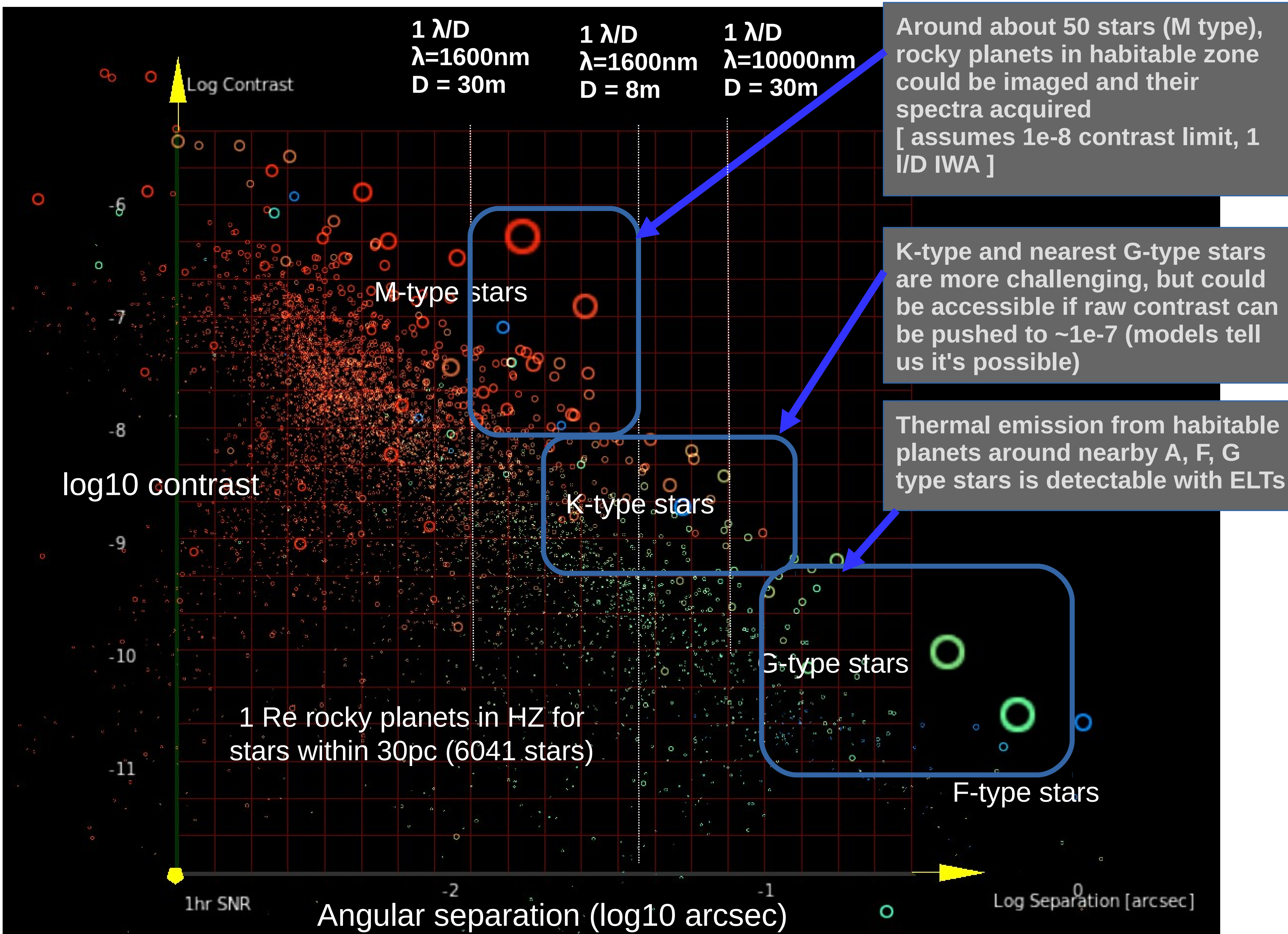
Olivier Guyon

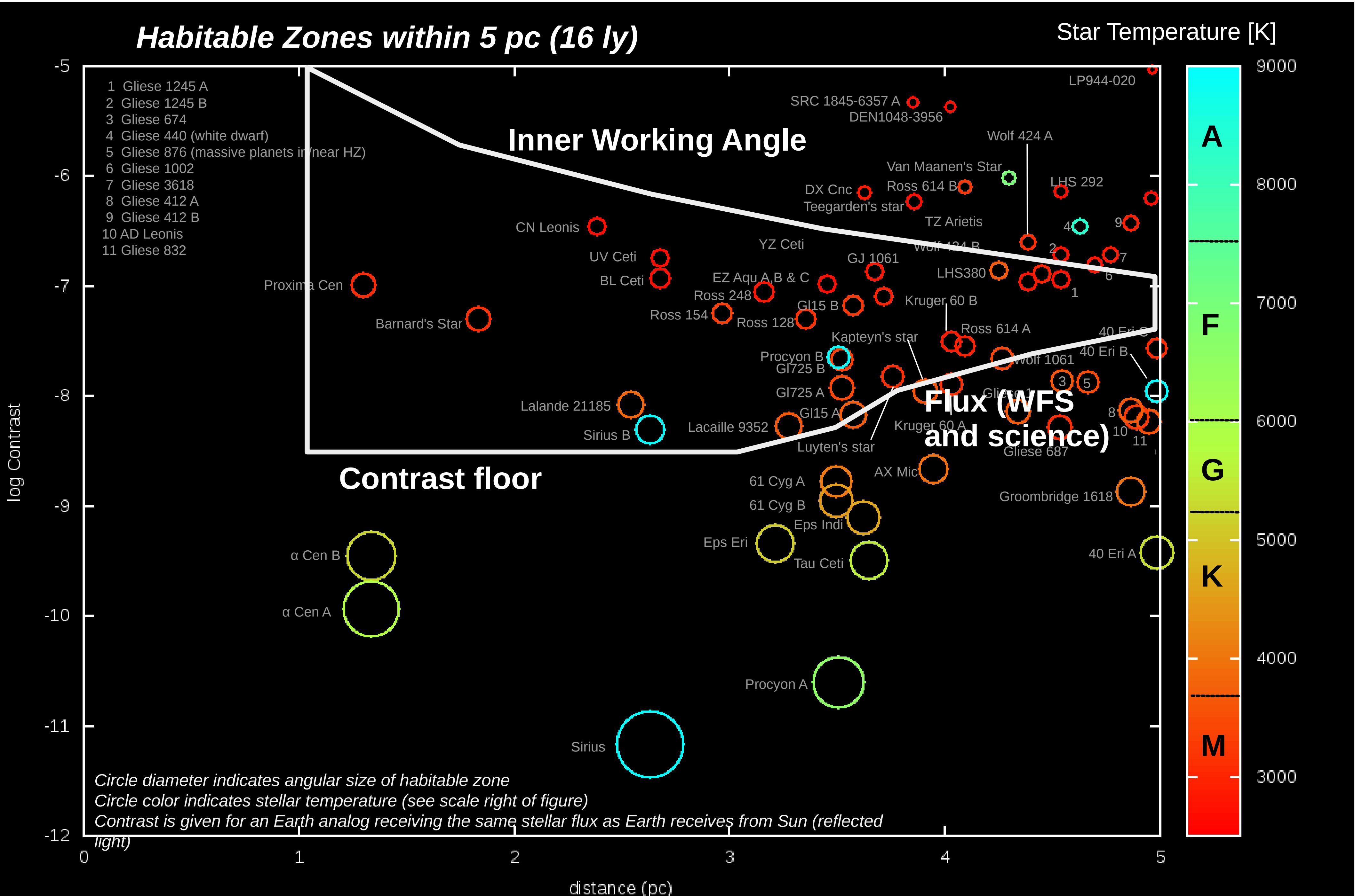
Japanese Astrobiology Center, National Institutes for Natural Sciences (NINS)
Subaru Telescope, National Astronomical Observatory of Japan (NINS)
University of Arizona
Breakthrough Watch committee chair

Julien Lozi, Nemanja Jovanovic, SCExAO team & collaborators

June 27, 2017

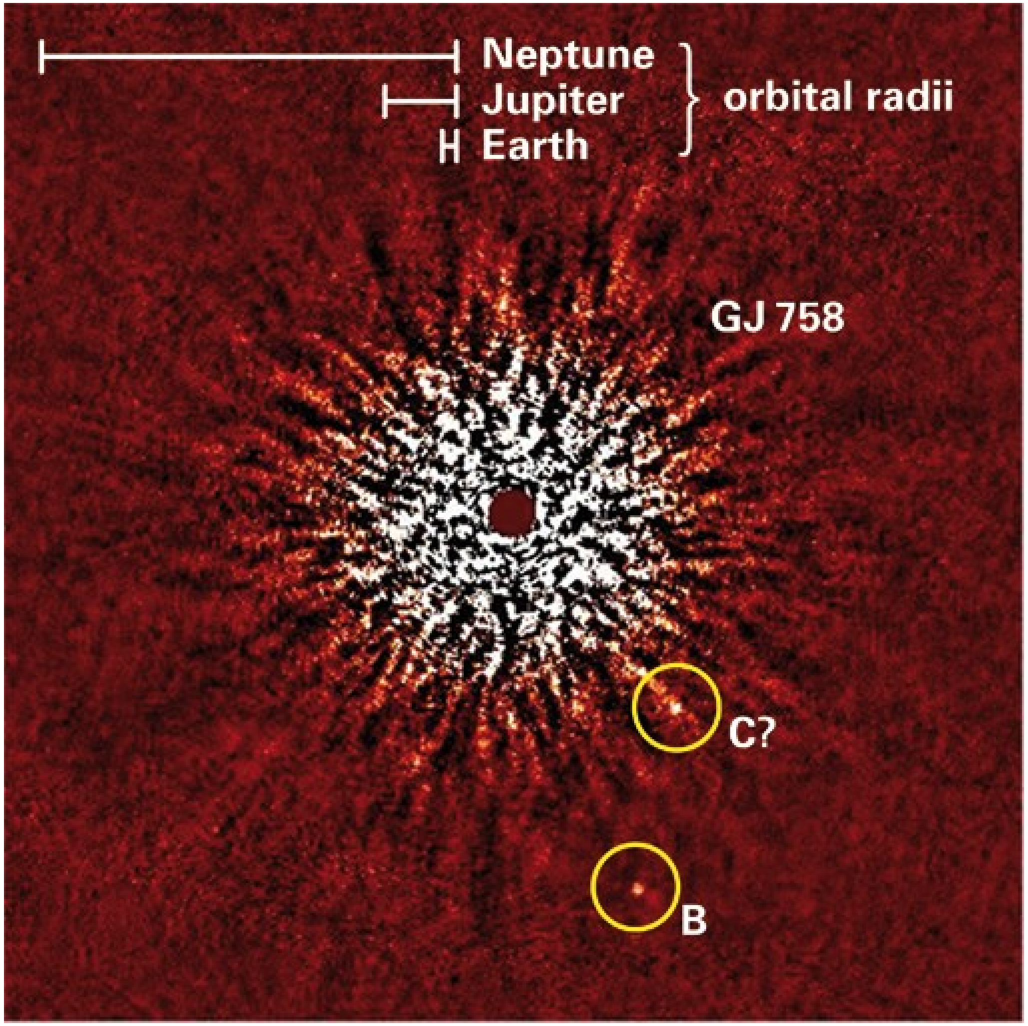
Contrast and Angular separation



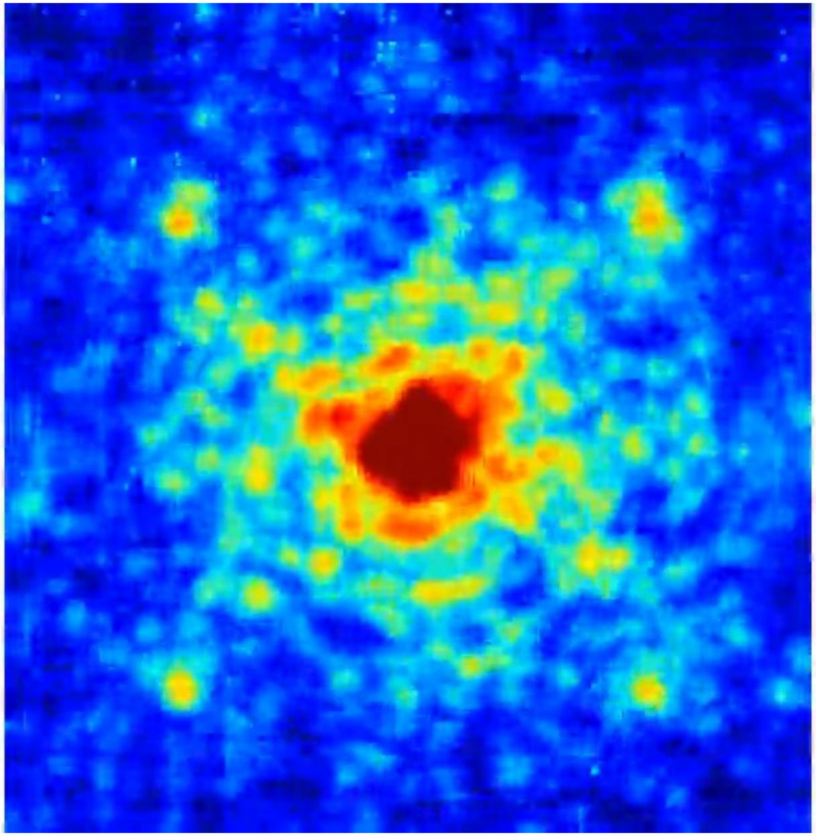


Habitable Zones within 5 pc (16 ly)
Star Temperature [K]
Inner Working Angle
Contrast floor
Flux (WFS and science)
1 Gliese 1245 A
2 Gliese 1245 B
3 Gliese 674
4 Gliese 440 (white dwarf)
5 Gliese 876 (massive planets in/near HZ)
6 Gliese 1002
7 Gliese 3618
8 Gliese 412 A
9 Gliese 412 B
10 AD Leonis
11 Gliese 832
log Contrast
distance (pc)
A
F
G
K
M
Proxima Cen
Barnard's Star
CN Leonis
UV Ceti
BL Ceti
Lalande 21185
Sirius B
α Cen B
α Cen A
Sirius
Procyon A
Eps Eri
Tau Ceti
Eps Indi
61 Cyg A
61 Cyg B
AX Mic
Groombridge 1618
40 Eri A
40 Eri B
40 Eri C
Lacaille 9352
Luyten's star
Kruger 60 A
Kruger 60 B
Gliese 687
Wolf 1061
Ross 614 A
Ross 614 B
Kapteyn's star
Procyon B
Gl725 B
Gl725 A
Gl15 A
Gl15 B
Ross 128
Ross 154
Ross 248
EZ Aqu A,B & C
YZ Ceti
GJ 1061
LHS380
Wolf 424 A
Wolf 424 B
TZ Arietis
LHS 292
Van Maanen's Star
DX Cnc
Teegarden's star
SRC 1845-6357 A
DEN1048-3956
LP944-020
Circle diameter indicates angular size of habitable zone
Circle color indicates stellar temperature (see scale right of figure)
Contrast is given for an Earth analog receiving the same stellar flux as Earth receives from Sun (reflected light)

The REAL challenge: Wavefront error (speckles)

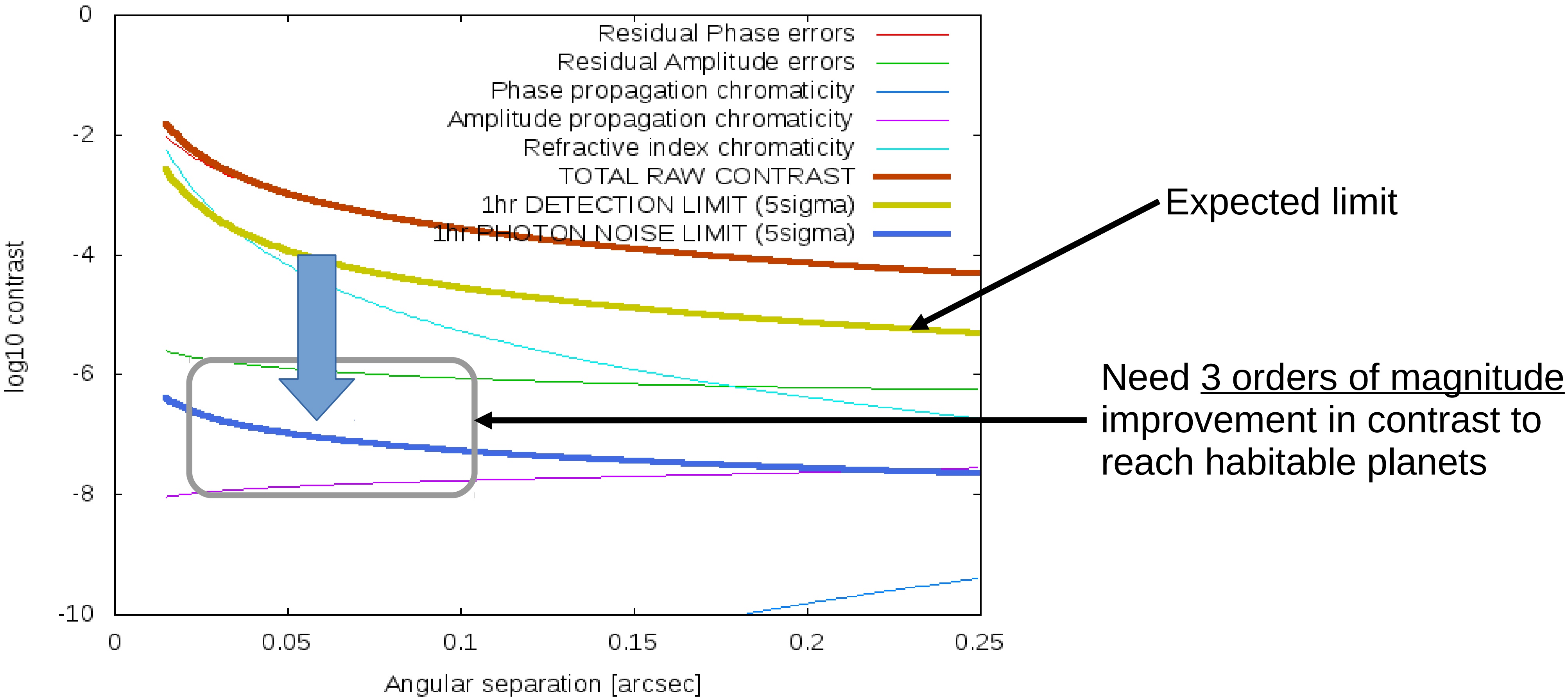


H-band fast frame imaging (1.6 kHz)



PREVIOUS technologies

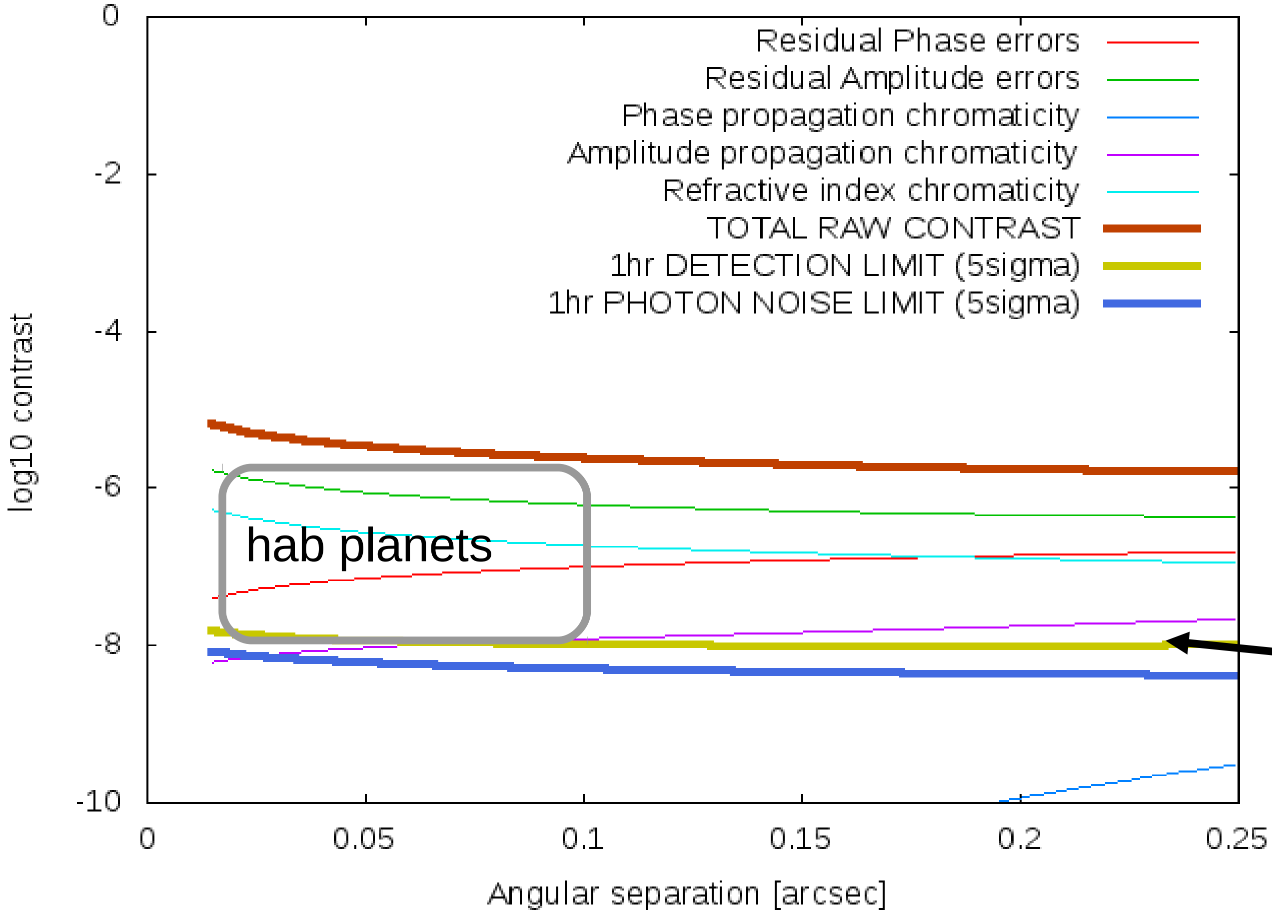
30m: SH-based system, 15cm subapertures



Limited by residual OPD errors: time lag + WFS noise
kHz loop (no benefit from running faster) – same speed as 8m telescope
>10kph per WFS required

Detection limit ~1e-3 at IWA, **POOR AVERAGING due to crossing time**

CURRENT/NEW technologies



Expected limit

Assumes:
- high-sensitivity WFS
- speckle control

Does not take into account:
HR spectral calibration
Predictive control

300Hz speckle control loop (~1kHz frame rate) is optimal

Residual speckle at ~1e-6 contrast and fast → good averaging to detection limit at ~1e-8

Wavefront Control: challenges & solutions

WFS efficiency
M stars are not very bright for ExAO → need high efficiency WFS
For low-order modes (TT), seeing-limited (SHWFS) requires (D/r0)^2 times more light than diffraction-limited WFS
This is a **40,000x gain for 30m telescope** (assuming r0=15cm) → 11.5 mag gain

Low latency WFC
System lag is extremely problematic → creates "ghost" slow speckles that last crossing time
Need ~200us latency (10 kHz system, or slower system + lag compensation), or multiple loops

WF chromaticity
Wavefront chromaticity is a serious concern when working at ~1e-8 contrast
Visible light (~0.6 – 0.8 um) photon carry most of the WF information, but science is in near-IR

Non-common path errors
It doesn't take much to create a 1e-8 speckle !

PSF calibration
What is a speckle, what is a planet ?

Diffraction-limited pupil-plane WFS
Low or no modulation PyWFS is diffraction-limited
This is a **40,000x gain for 30m telescope** (assuming r0=15cm) → 11.5 mag gain

Fast WFC loop
Fast hardware (Cameras, GPUs) can now run loop at ~5 kHz on ELT
Example: SCExAO runs 2000 actuators, 14,400 sensors at 3.5kHz using ~10% of available RTS computing power

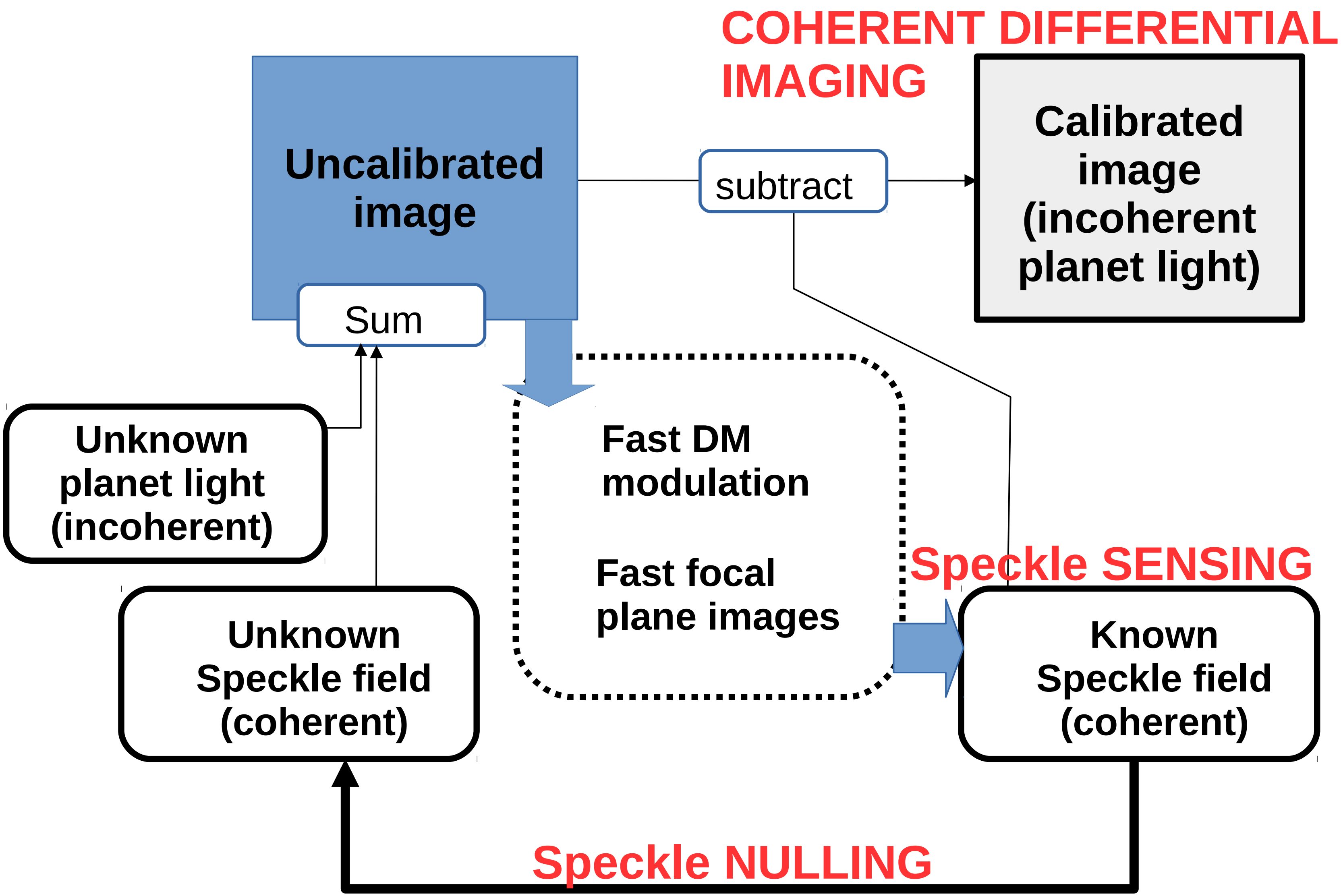
Predictive Control
Eliminates time lag, improves sensitivity

Fast speckle control, enabled by new detector technologies
Addresses simultaneously non-common path errors, (most of) lag error, chromaticity, and calibration

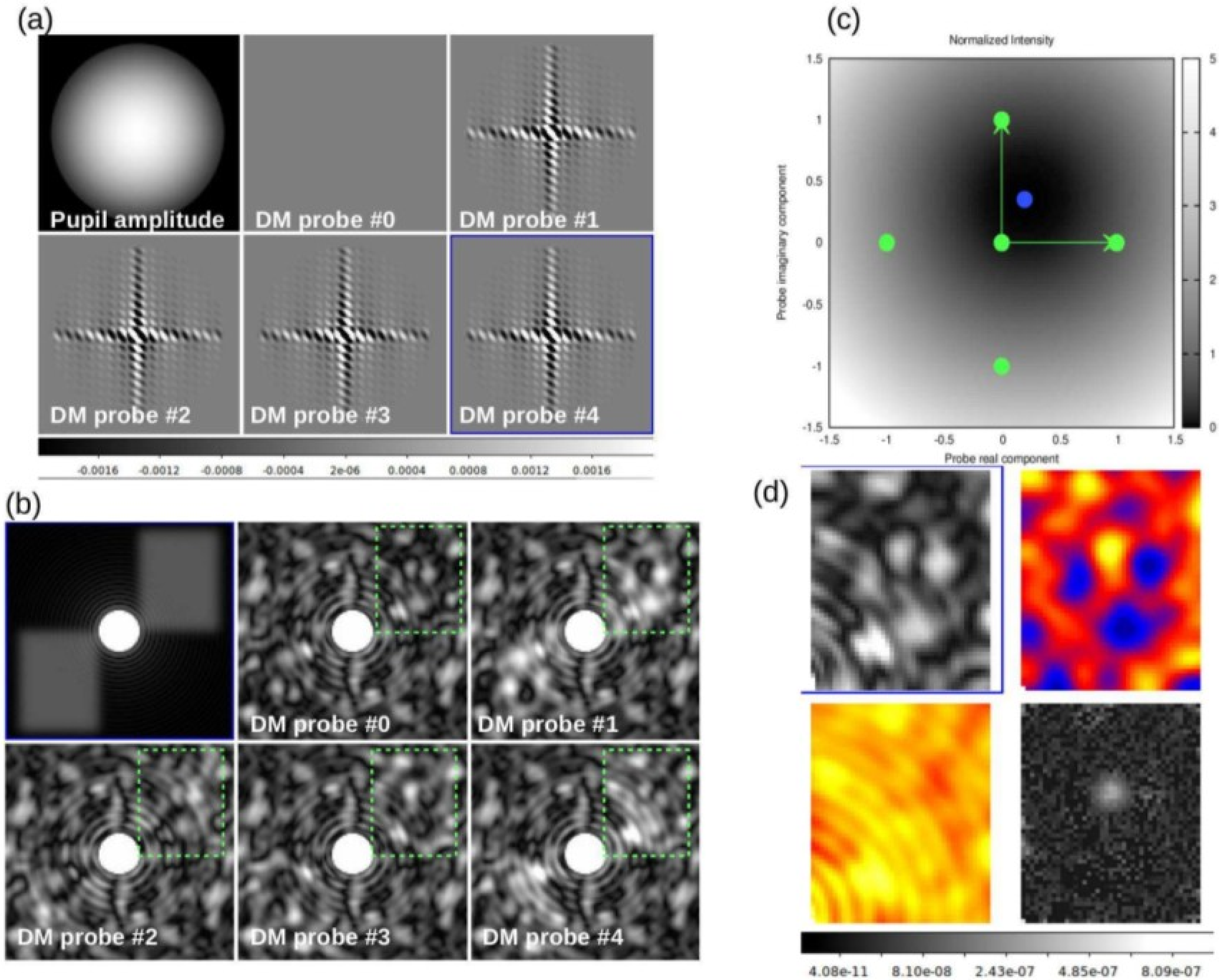
Real-time telemetry → PSF calibration
WFS telemetry tells us where speckles are → significant gain using telemetry into post-processing

Spectral discrimination (HR)
Especially powerful at high spectral resolution

High Speed Speckle Control & Calibration



Coherent Speckle Differential Imaging

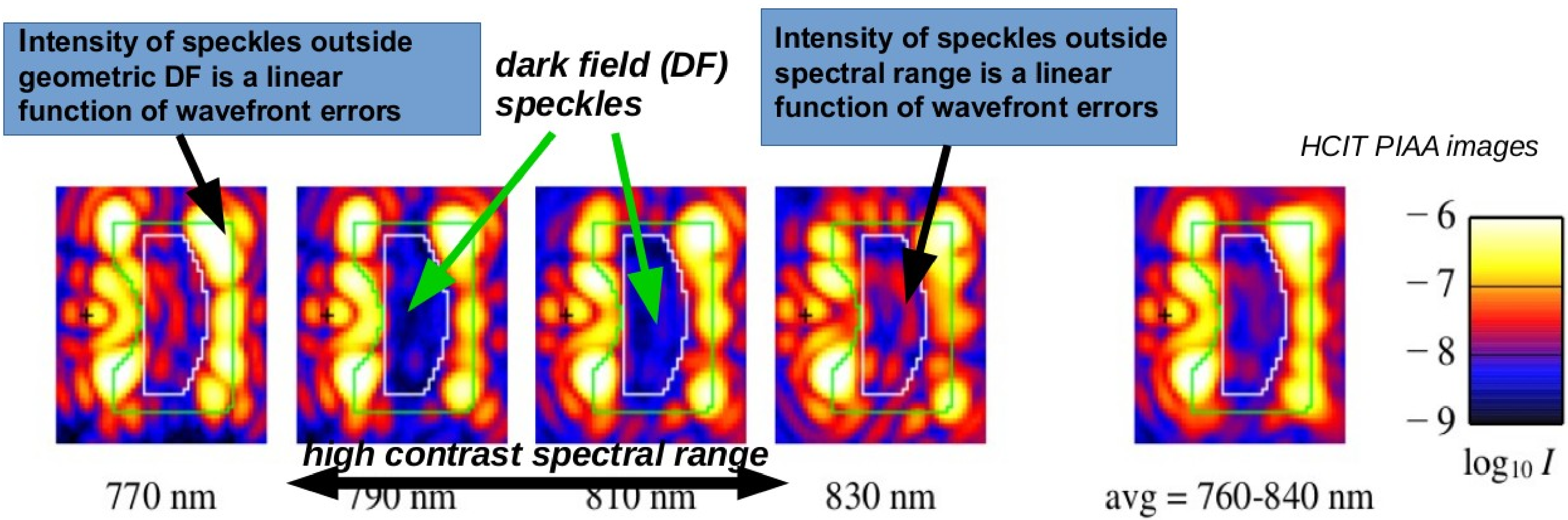


Linear Dark Field Control (LDFC)

See also: Miller et al. 2017, Guyon et al. 2017 (astro-ph)

Speckle intensity in the DF are a non-linear function of wavefront errors
→ current wavefront control technique uses several images (each obtained with a different DM shape) and a non-linear reconstruction algorithm (for example, Electric Field Conjugation – EFC)

Speckle intensity in the BF are linearly coupled to wavefront errors → we have developed a new control scheme using BF light to freeze the wavefront and therefore prevent light from appearing inside the DF



Predictive control & sensor fusion → 100x contrast gain ?

See also: Males & Guyon 2017 (astro-ph)

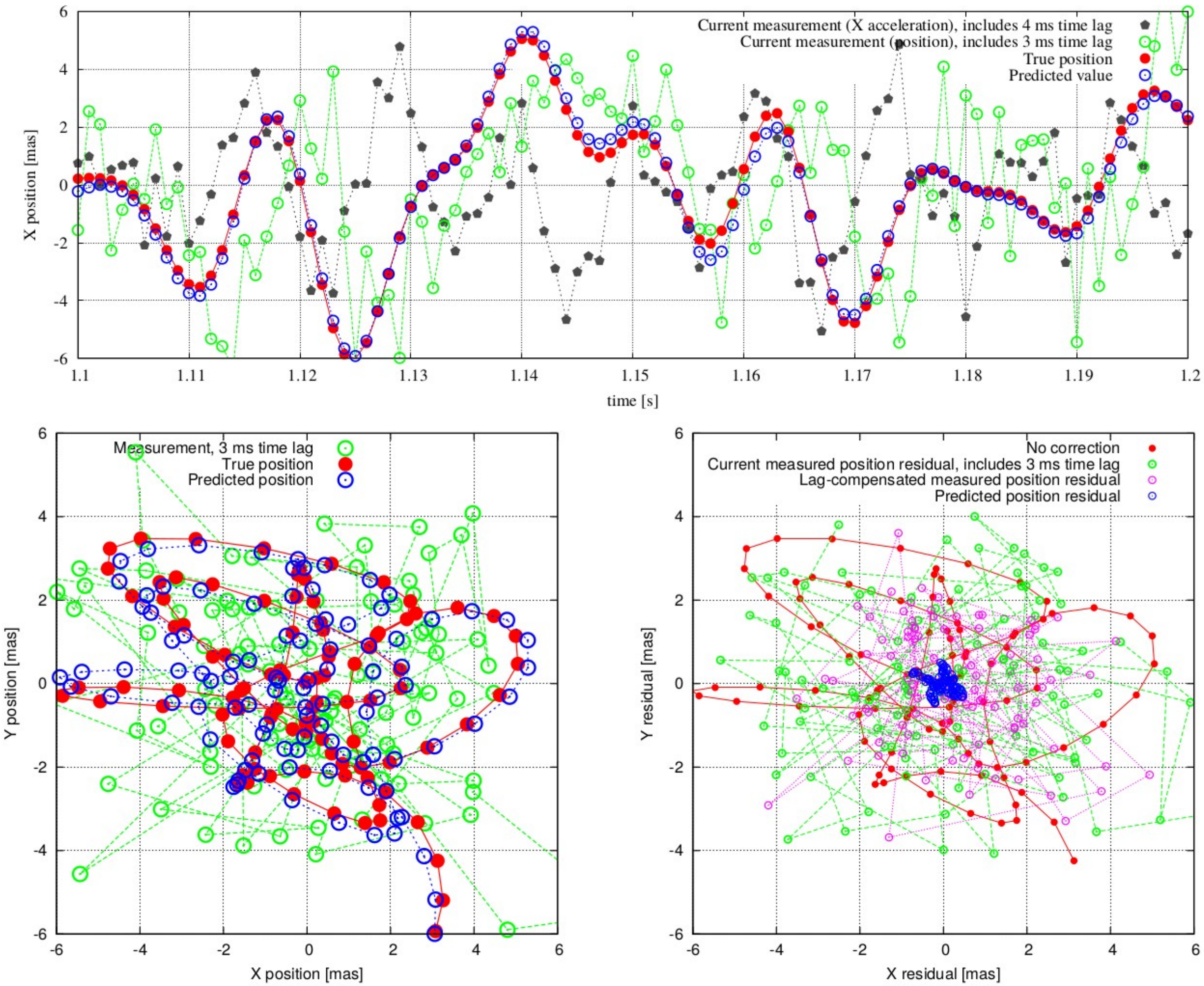
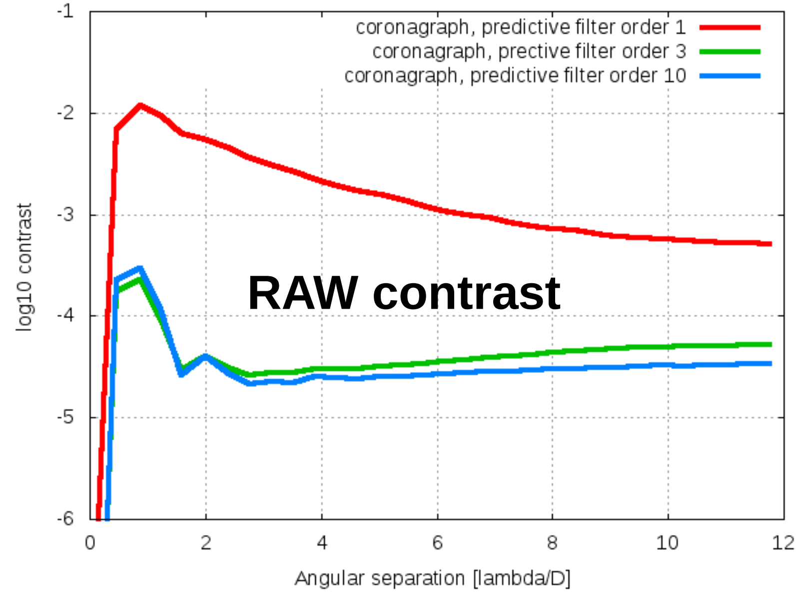
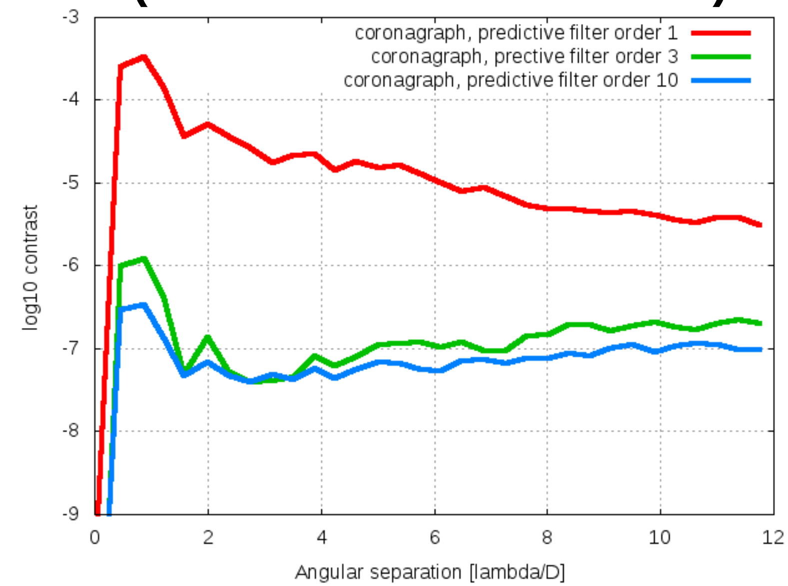


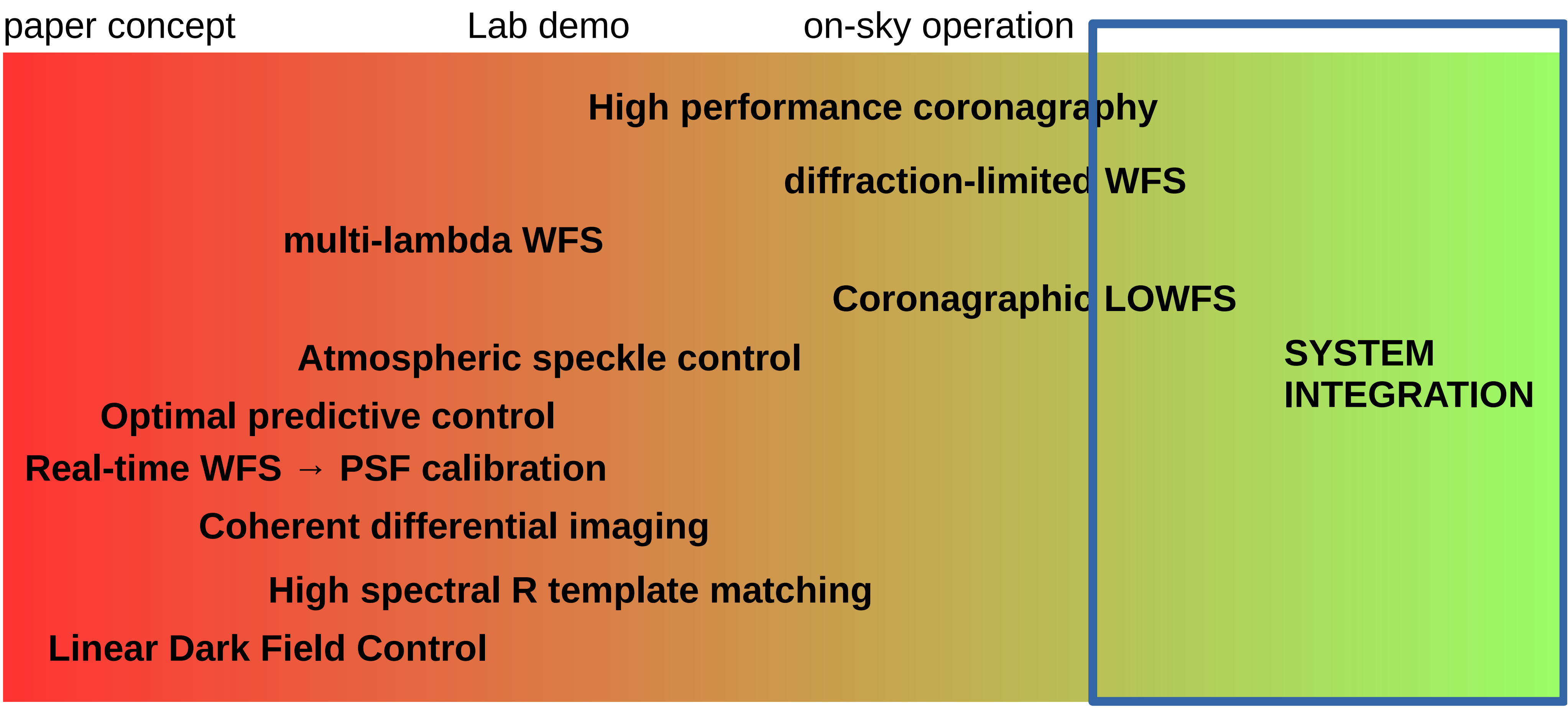
FIG. 3.— Top left: 2D-tracks for true pointing (red), predicted pointing (blue) and last measured position (green). Top right: Residual pointing error. Bottom: Single axis (x) values.



1hr detection limit (PSF subtraction residual)



Key technologies need rapid maturation from paper concepts to system integration



Subaru Coronagraphic Extreme Adaptive Optics

- **Flexible** high contrast imaging platform (Nas port)
- Meant to **evolve to TMT instrument** and validate key technologies required for direct imaging and spectroscopy of habitable exoplanets

Telescope time available to US community (Keck & Gemini time exchange) and non-US through collaborations with team

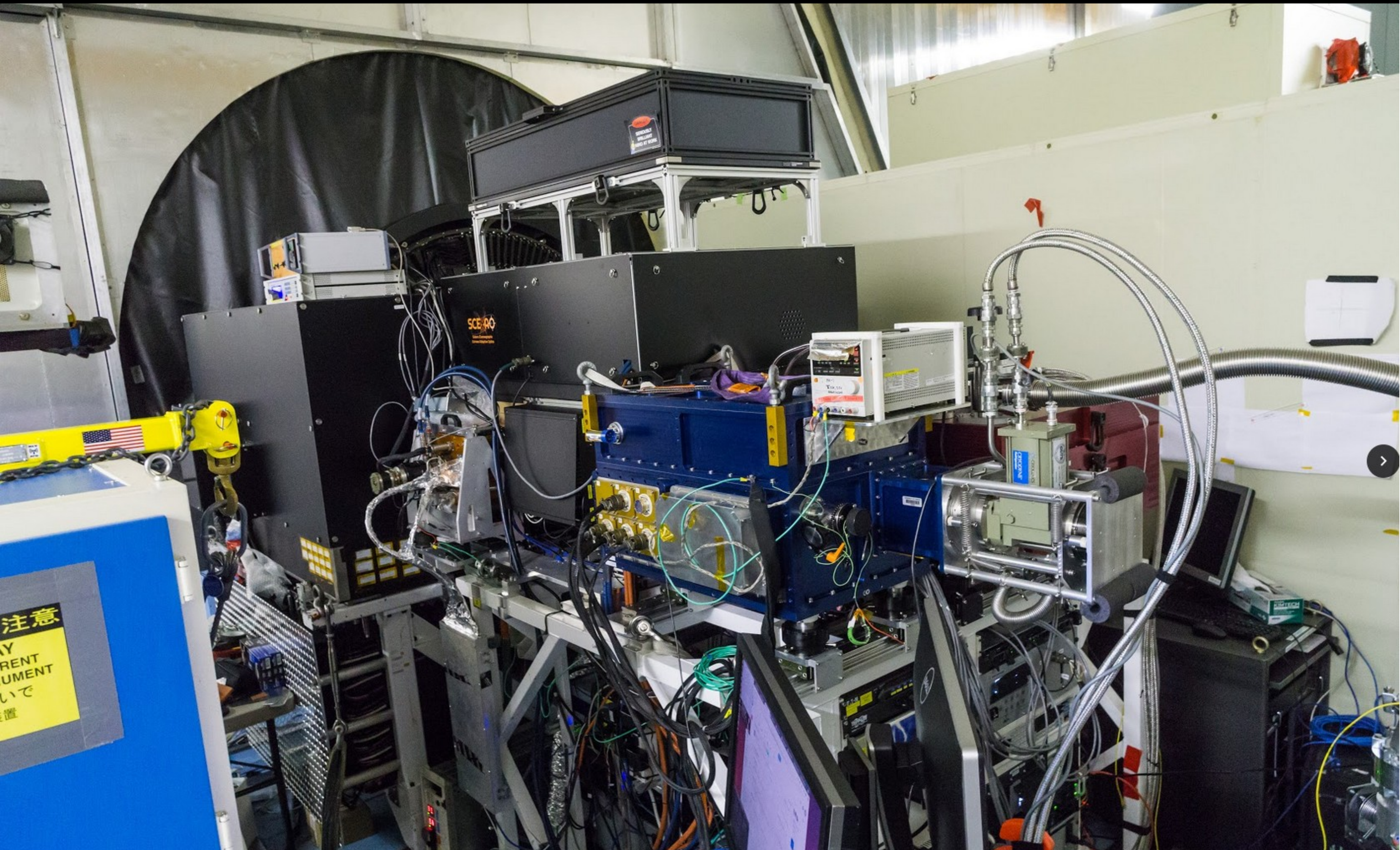
Modules/instruments funded by Japan + international partners:

- CHARIS IFU built by Princeton Univ (Japan-funded)
- MKIDs built by UCSB (Japan-funded)
- SAPHIRA camera provided by UH
- VAMPIRES instrument funded and built by Australia
- FIRST instrument funded and built by Europe
- RHEA IFU provided by Australian team

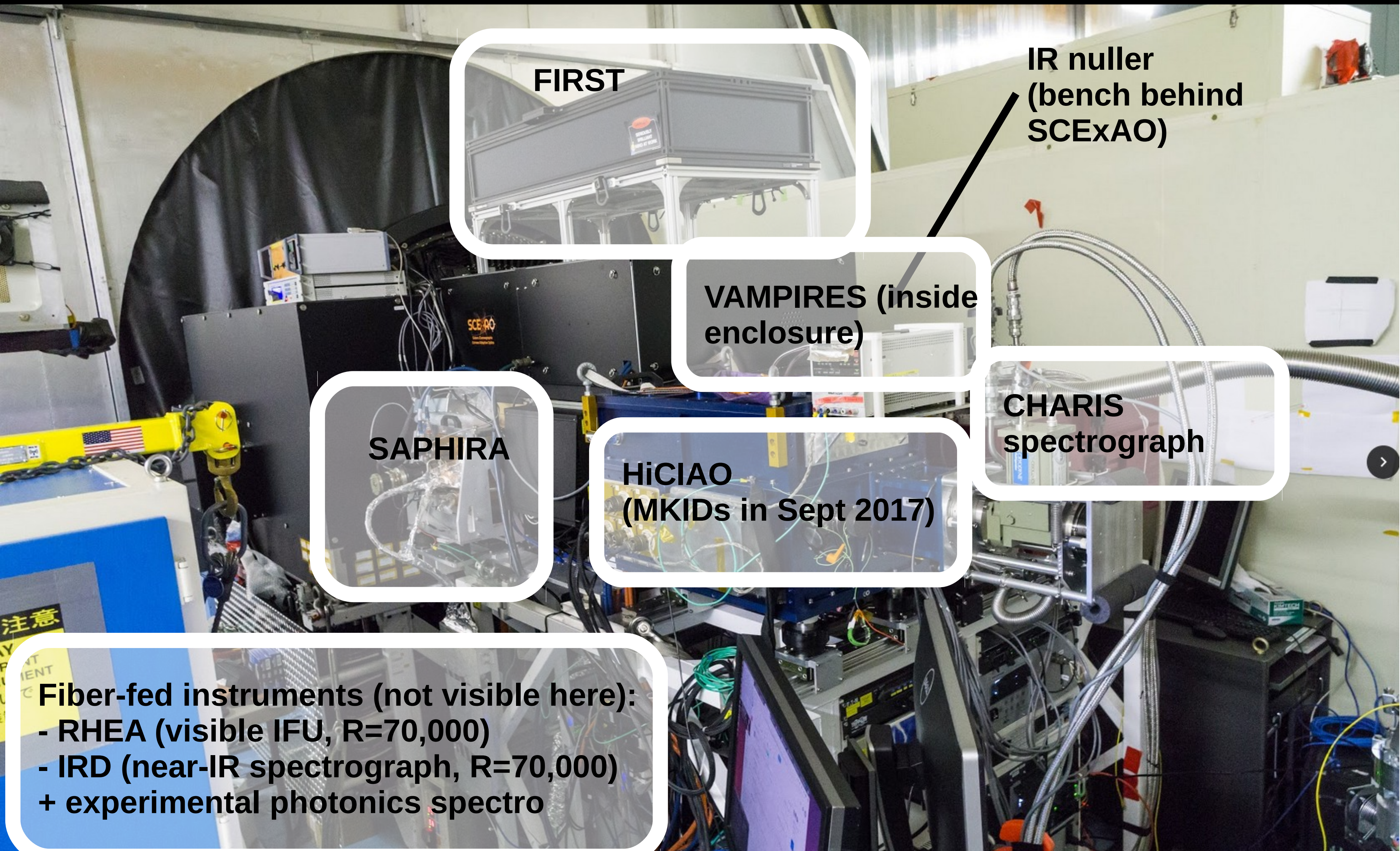
Strong research collaborations with multiple groups:

- Univ. of Arizona / MagAO(-X) (shared dev., wavefront control, coronagraphy)
- Kernel group @ Observatoire de la Cote d'Azur (wavefront control)
- Leiden Univ, JPL (coronagraphy)
- Northwestern Univ (detector dev)
- Univ. of Sydney (Photonics techs, nulling interferometry)
- Keck (near-IR WFS)

Subaru Coronagraphic Extreme Adaptive Optics

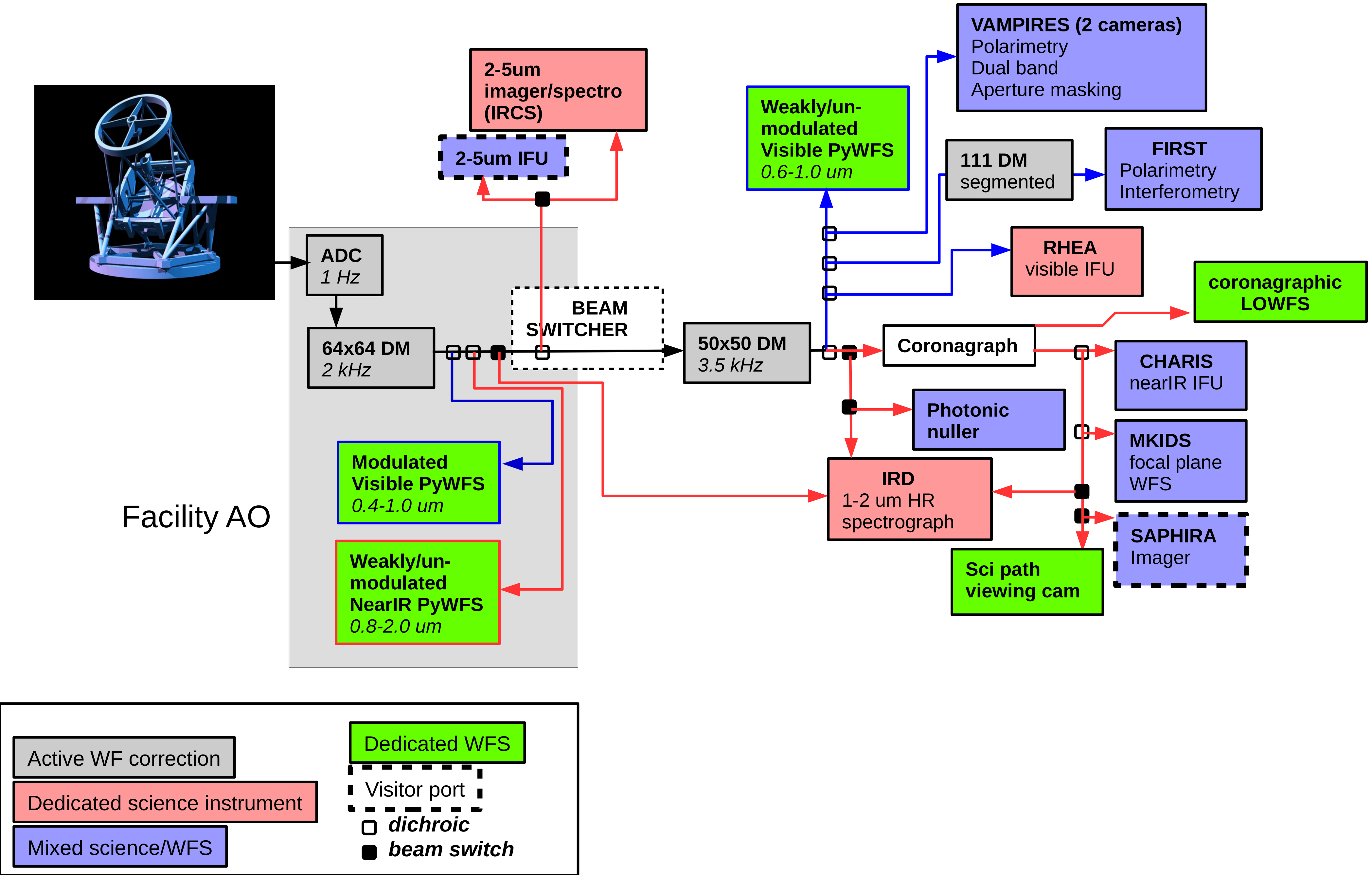


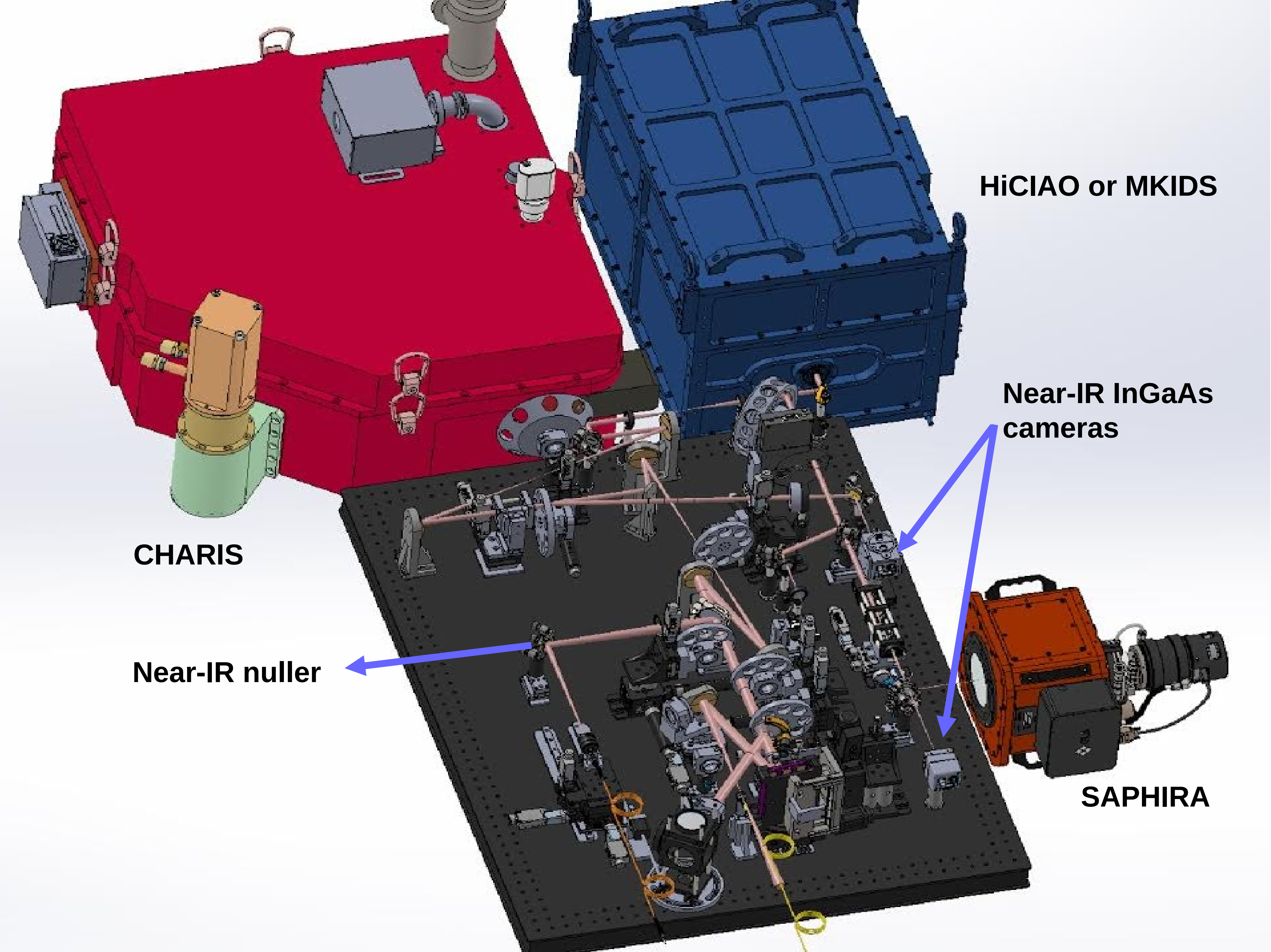
Subaru Coronagraphic Extreme Adaptive Optics



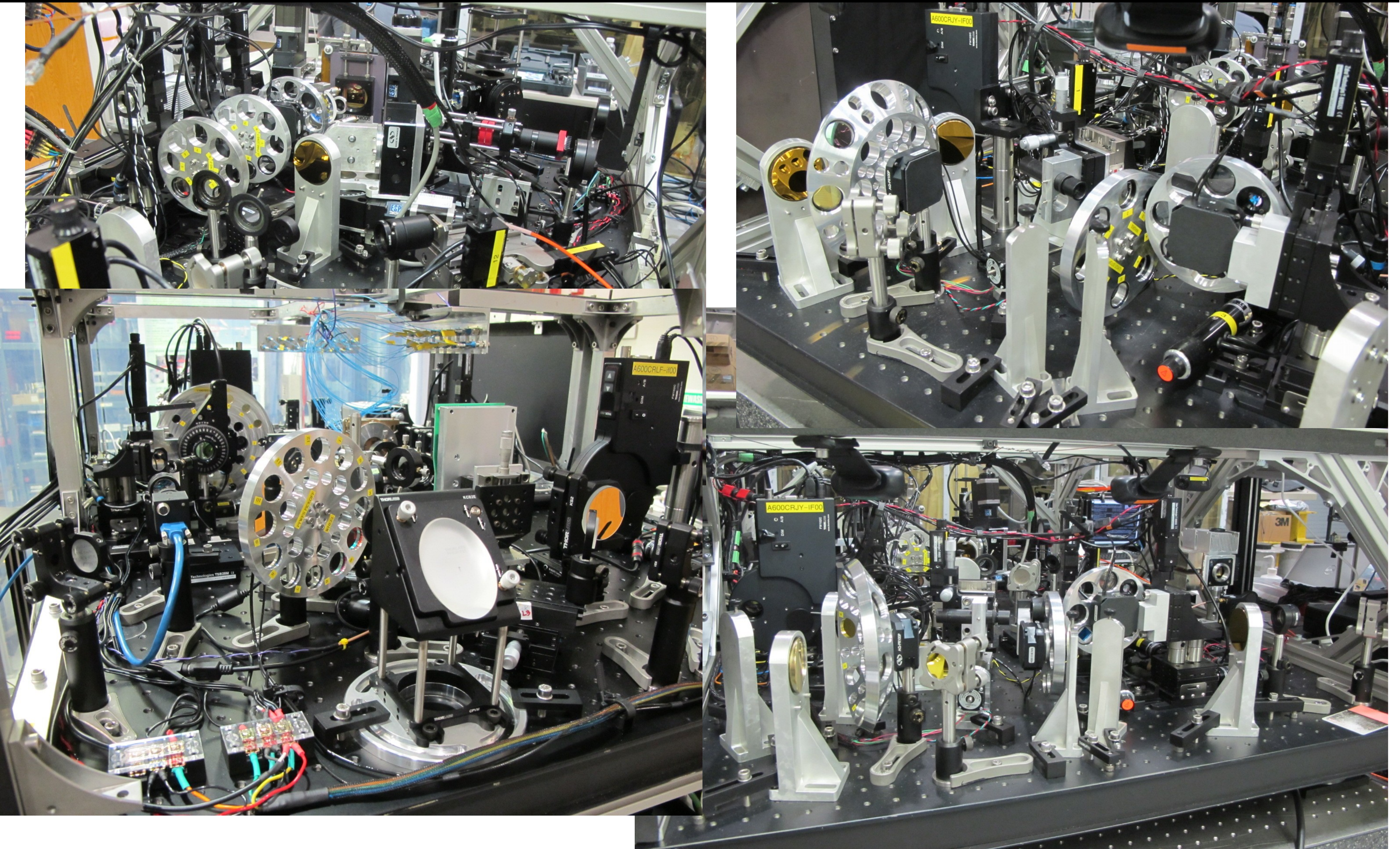
Subaru Coronagraphic Extreme Adaptive Optics

SCExAO Light path

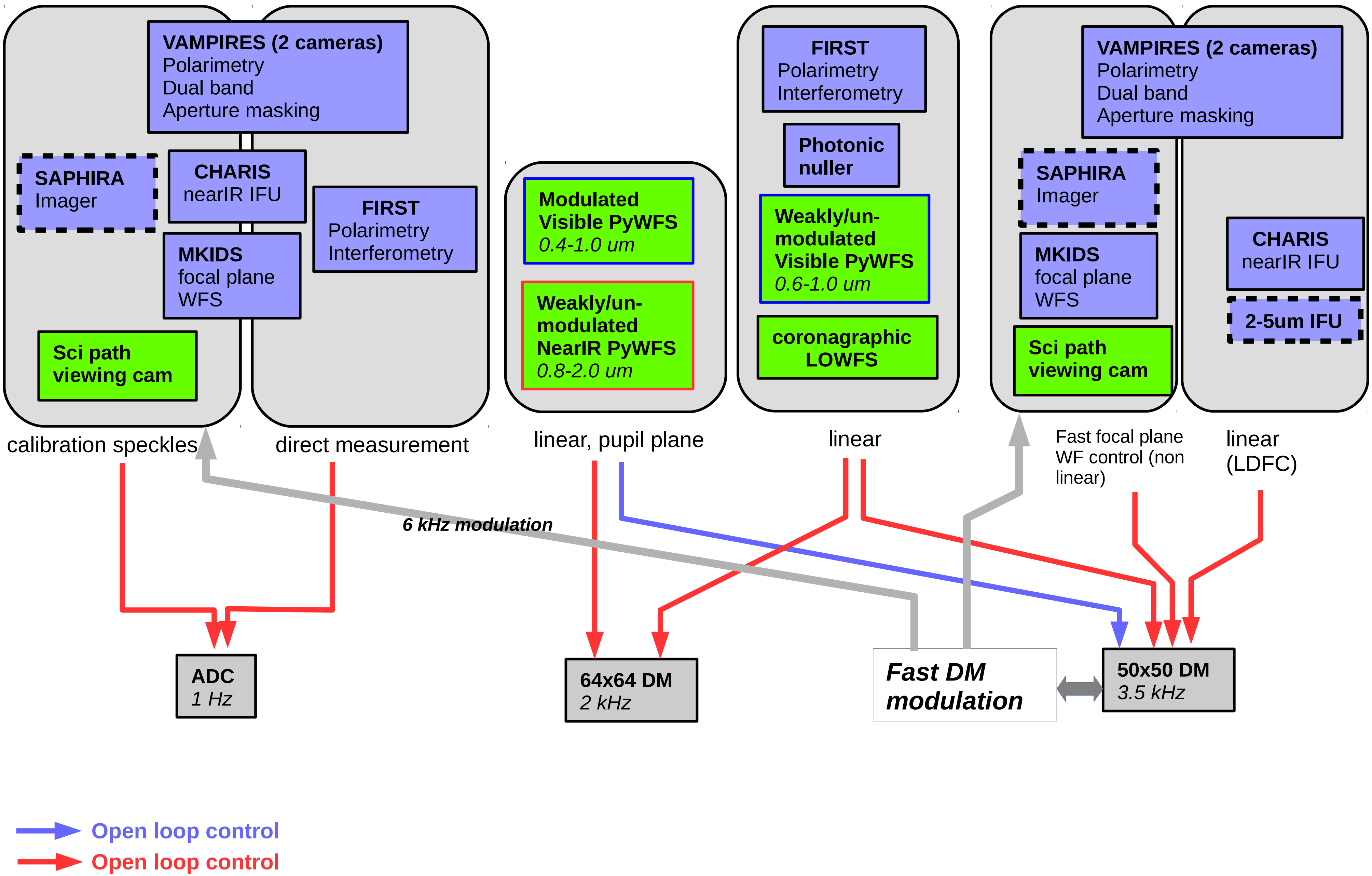



HiCIAO or MKIDS
Near-IR InGaAs
cameras
CHARIS
Near-IR nuller
SAPHIRA

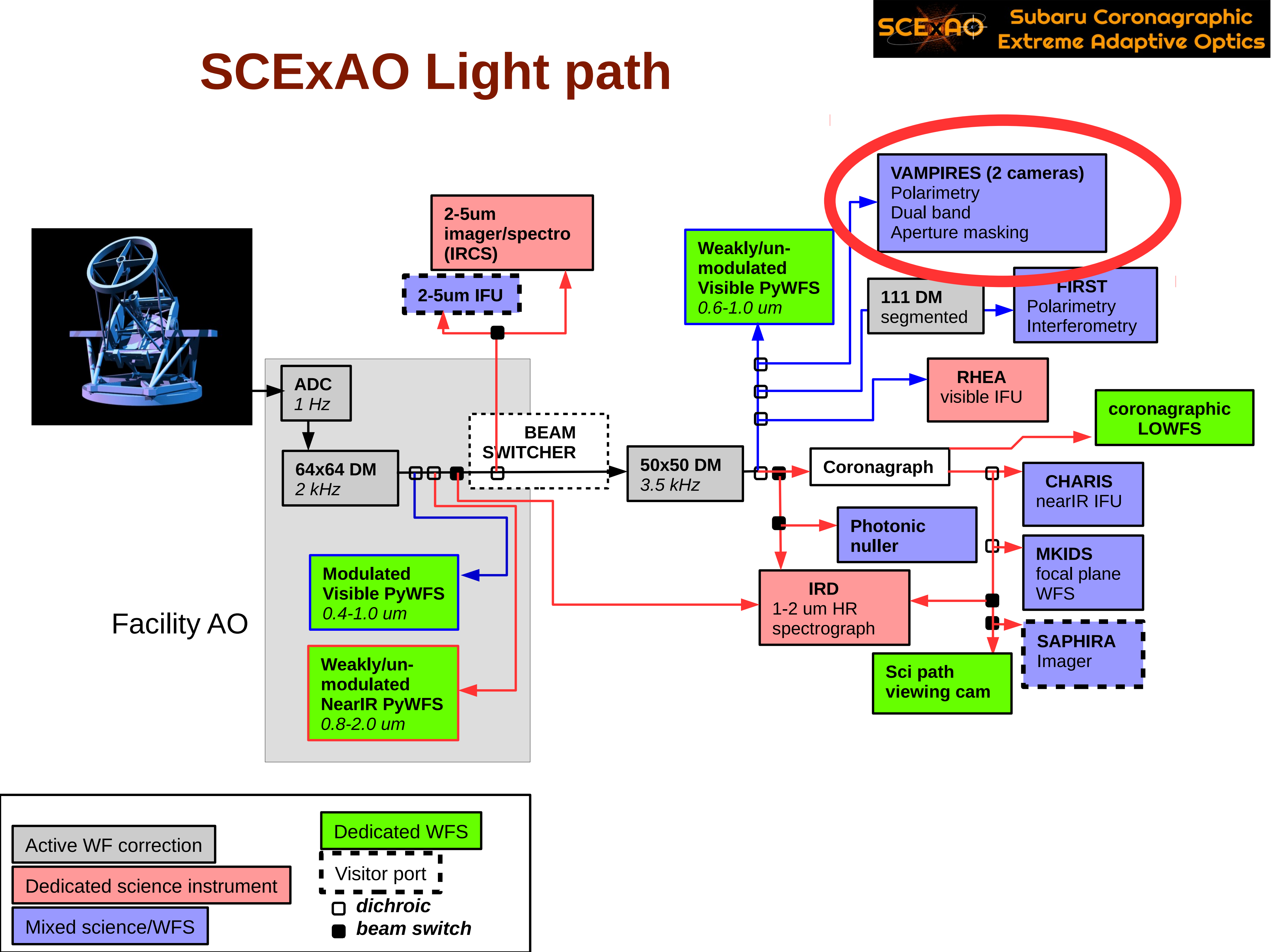
Subaru Coronagraphic Extreme Adaptive Optics



Control loops



SCExAO Light path

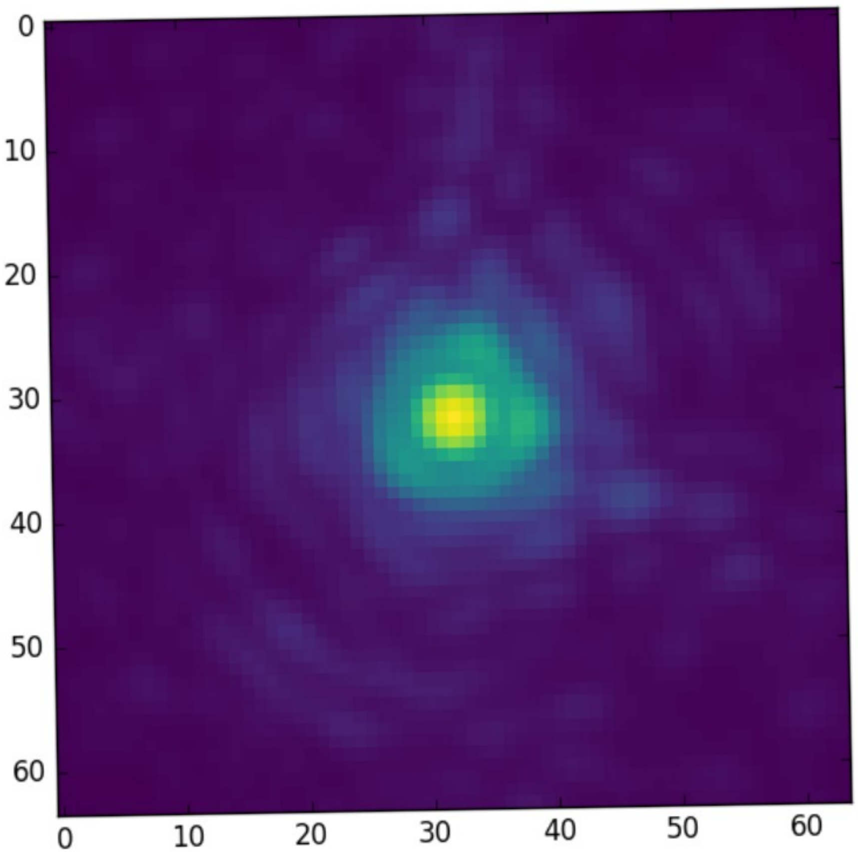


Preliminary VAMPIRES science

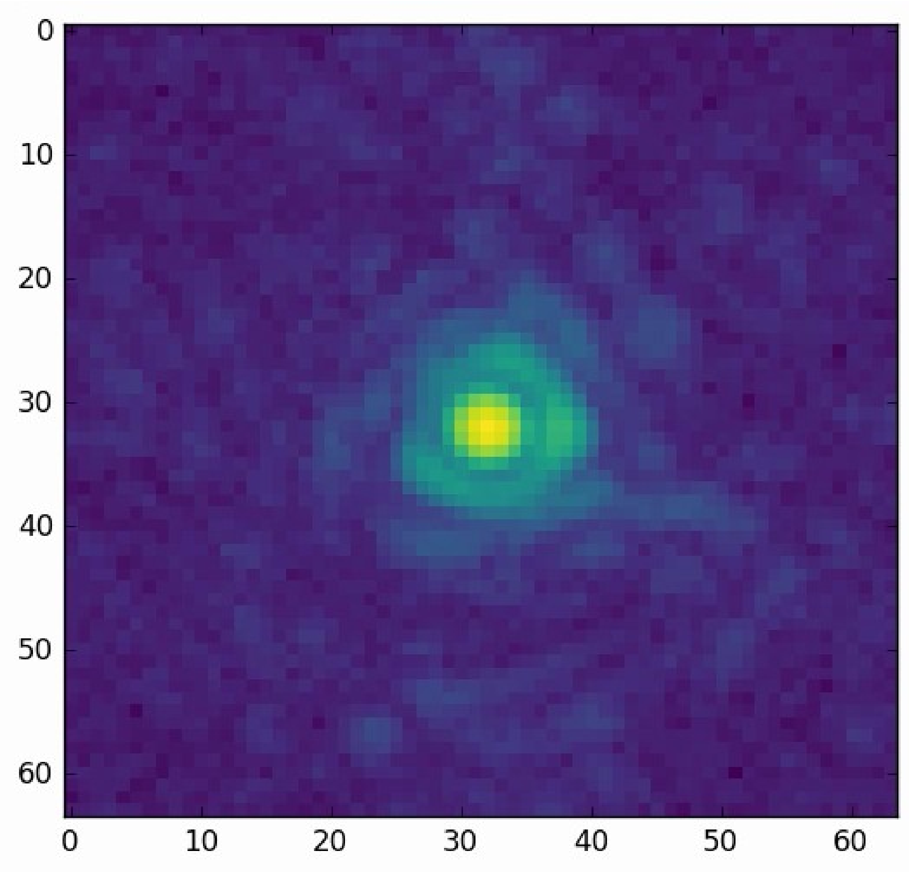
Diffraction-limited imaging in visible light

750nm, 1kHz imaging
log scale

Summed image

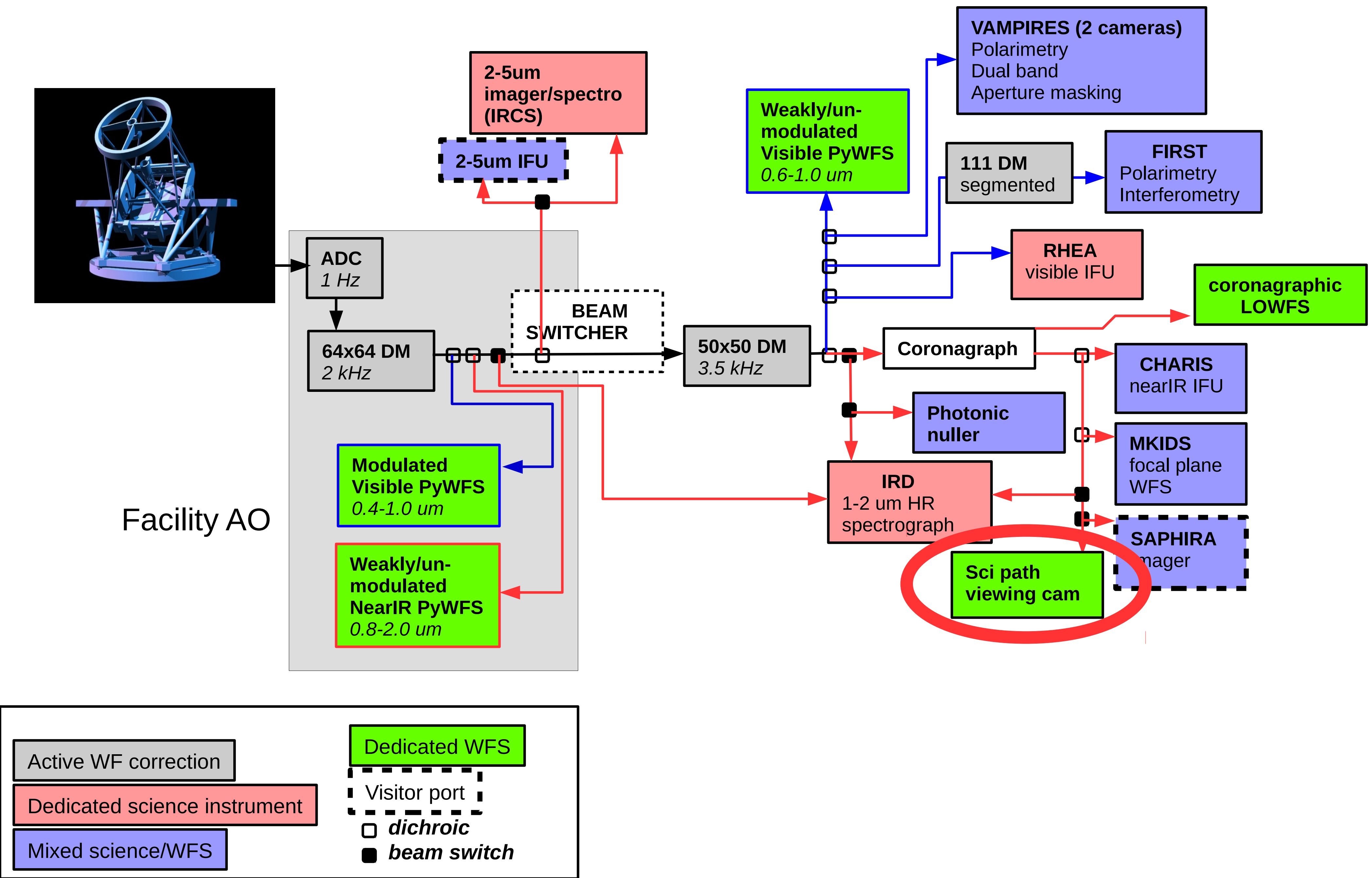


Video



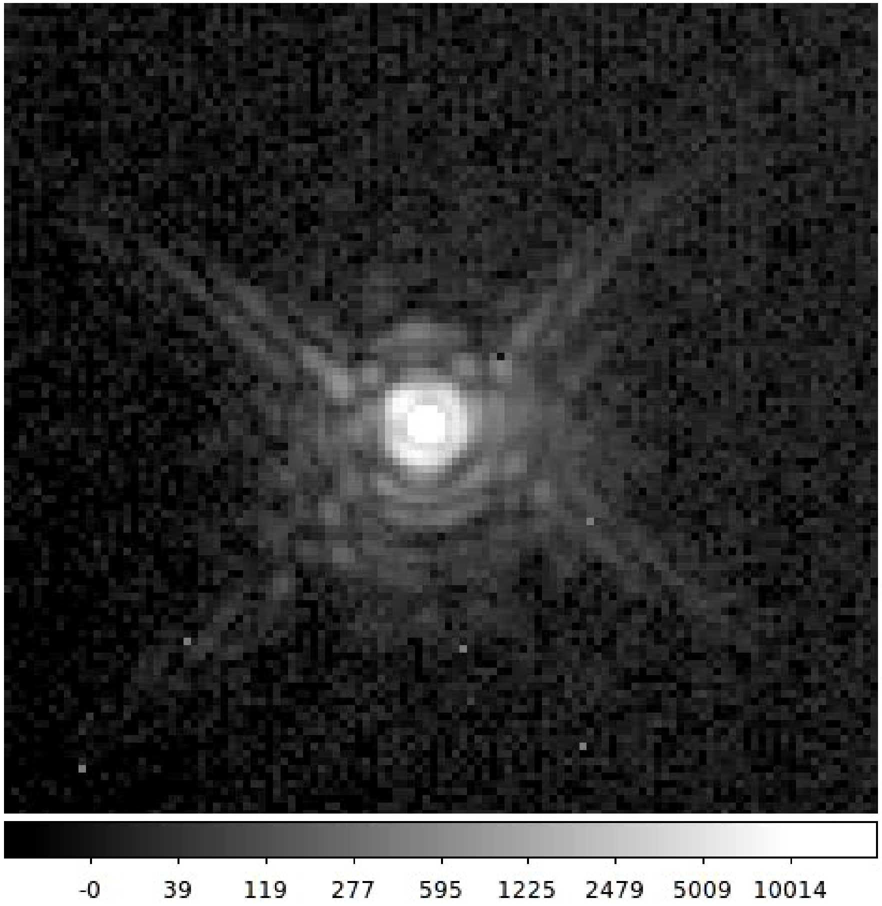
Subaru Coronagraphic Extreme Adaptive Optics

SCExAO Light path



Current PSF stability @ SCExAO

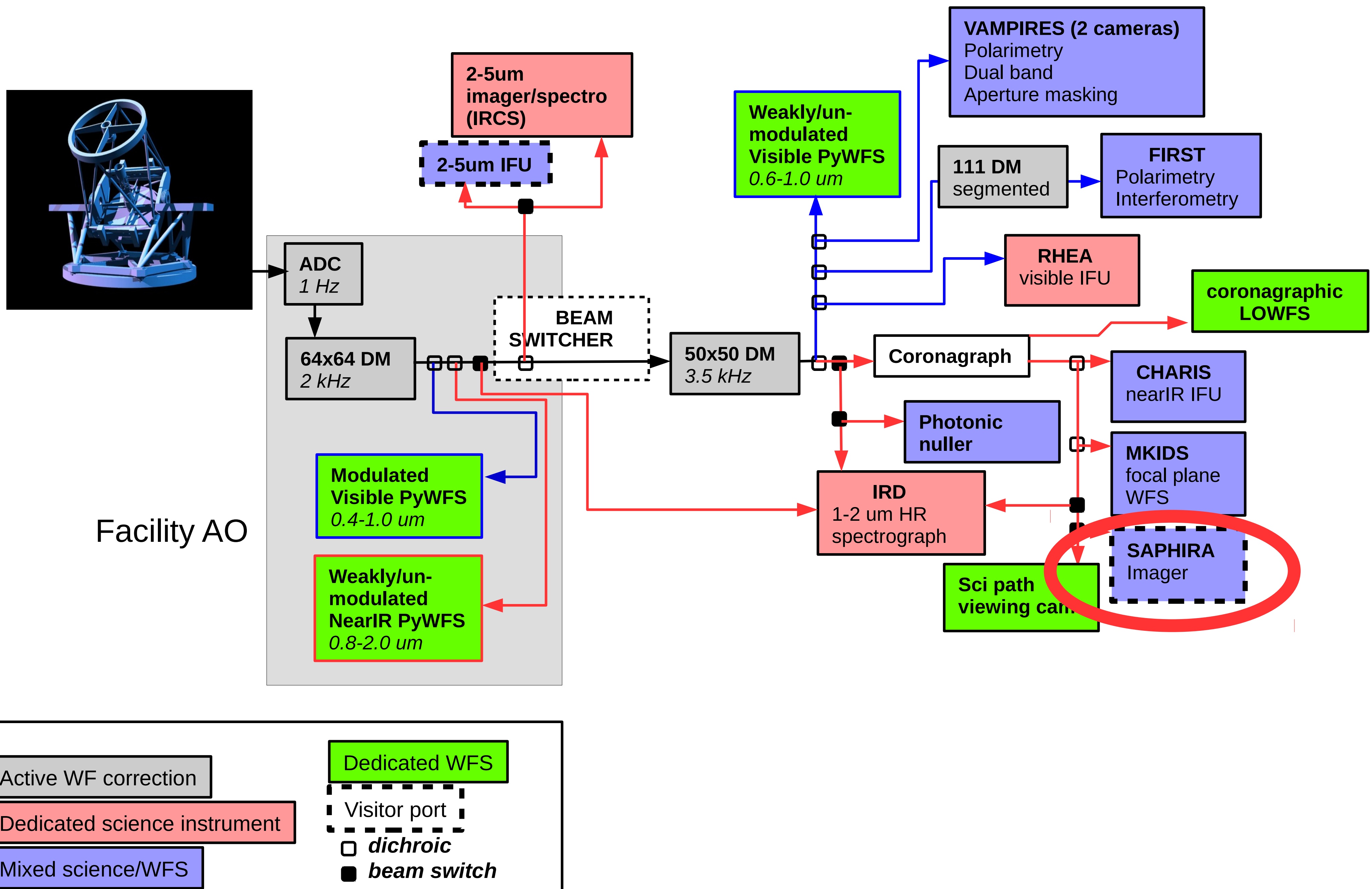
Stable PSF for coronagraphy
SCExAO provides sensing and correction at 500 Hz - 3.5 kHz
14,400 pixel WFS → 2000 actuators



1630nm (SCExAO internal camera)
3 Hz sampling

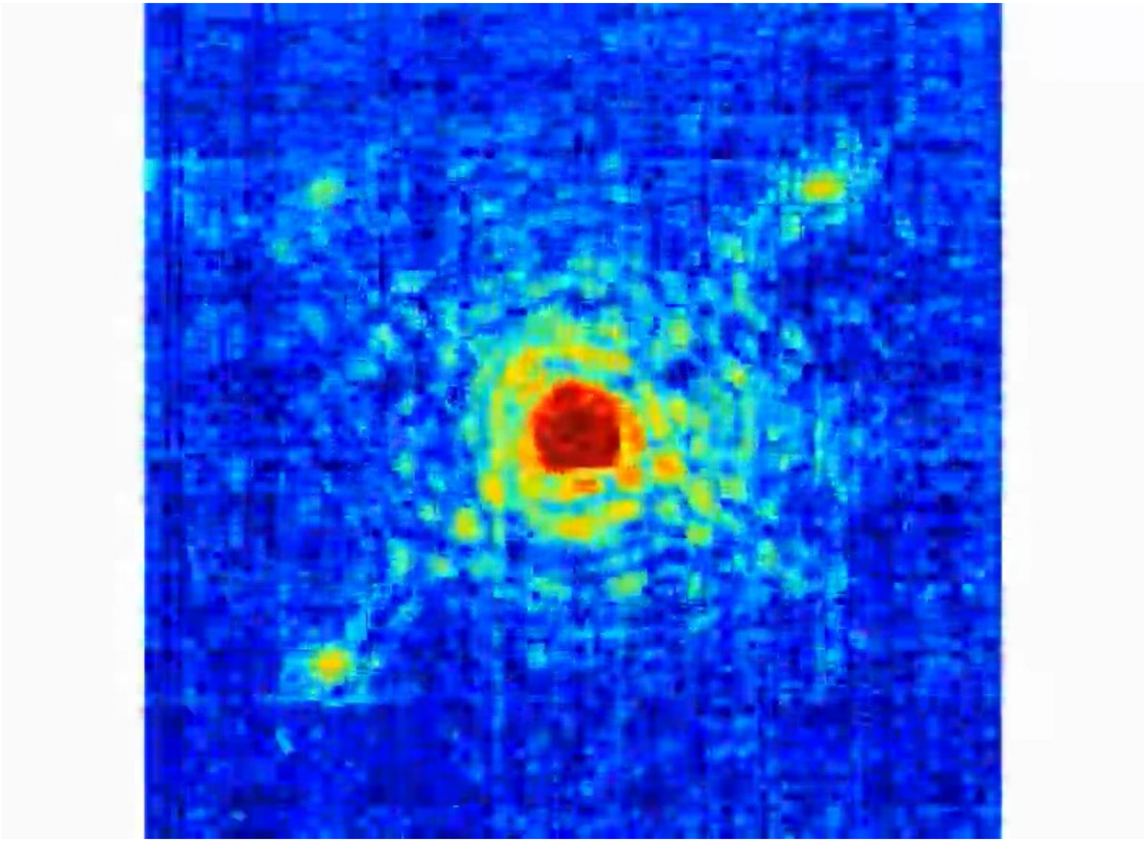
Subaru Coronagraphic Extreme Adaptive Optics

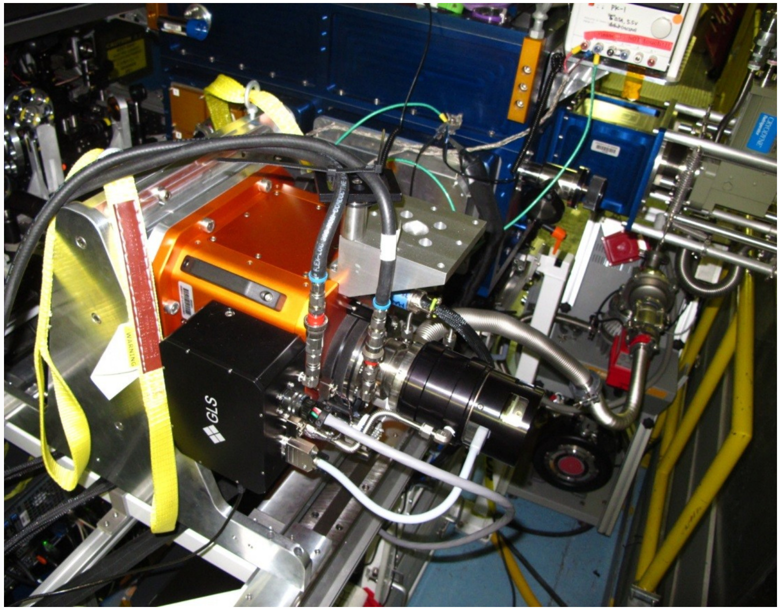
SCExAO Light path



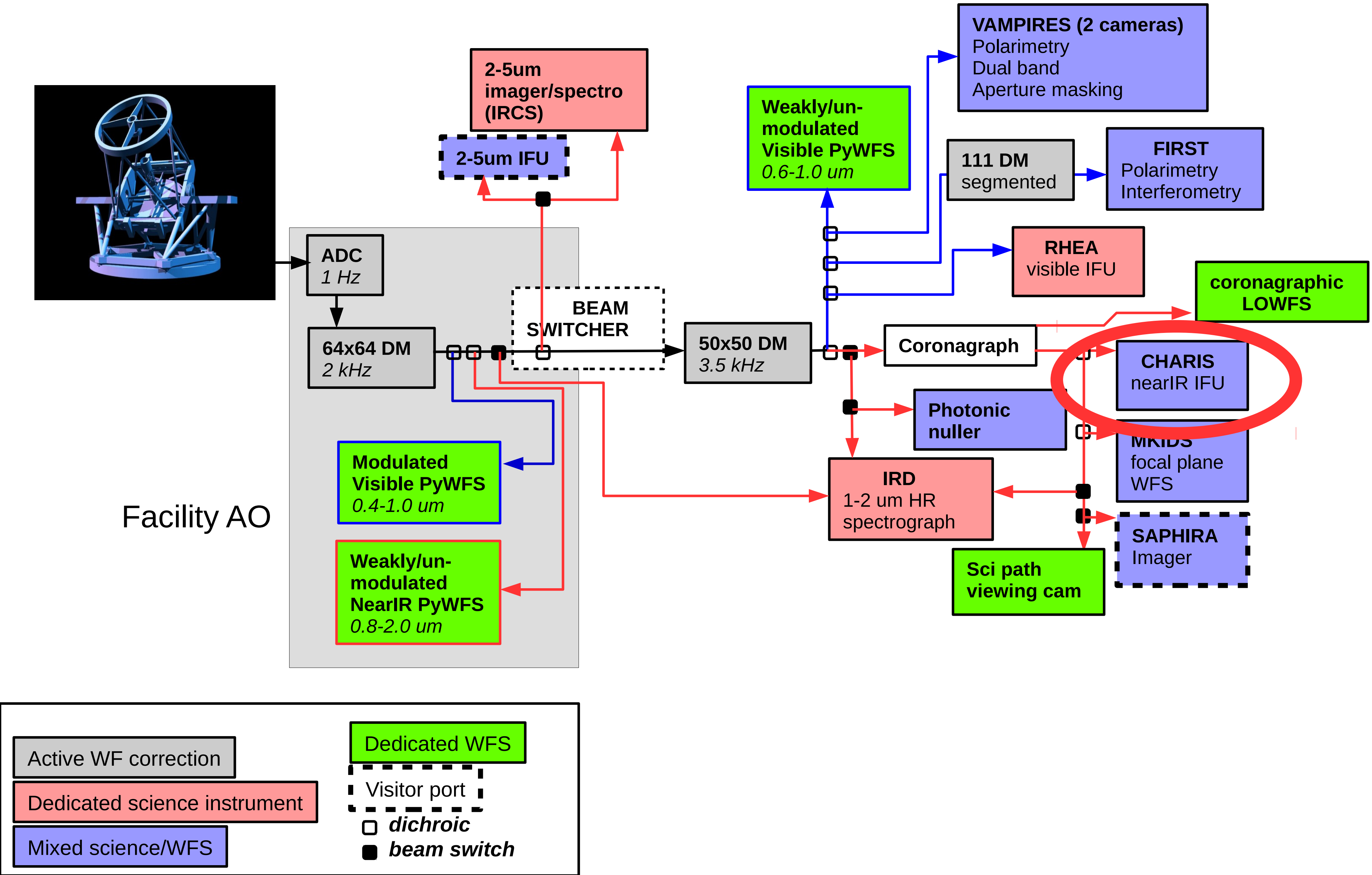
SAPHIRA camera

1.68 kHz frame rate, H-band
(played at 90 Hz)
SCExAO PyWFS ON → OFF

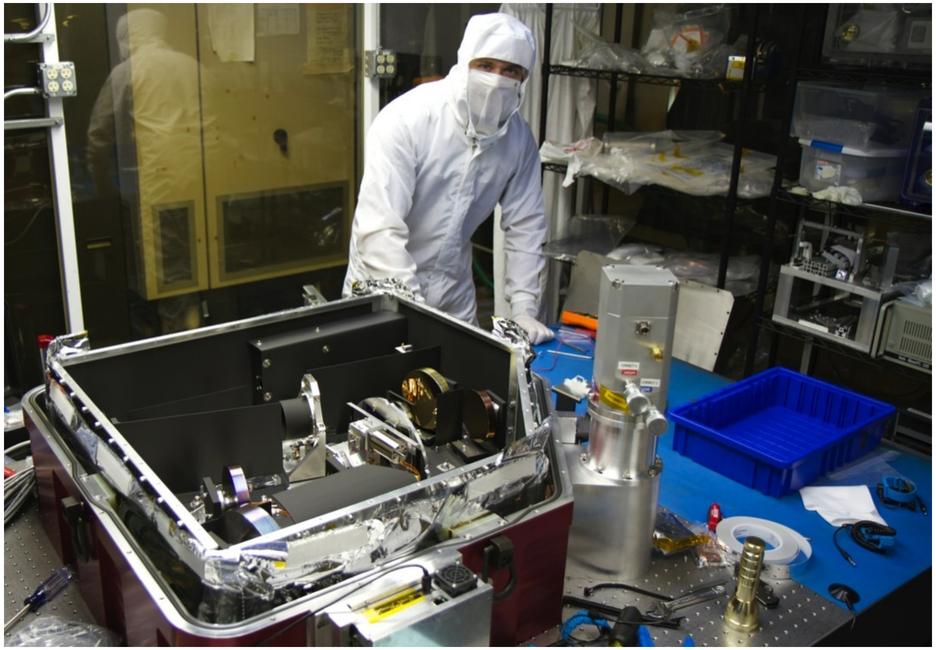


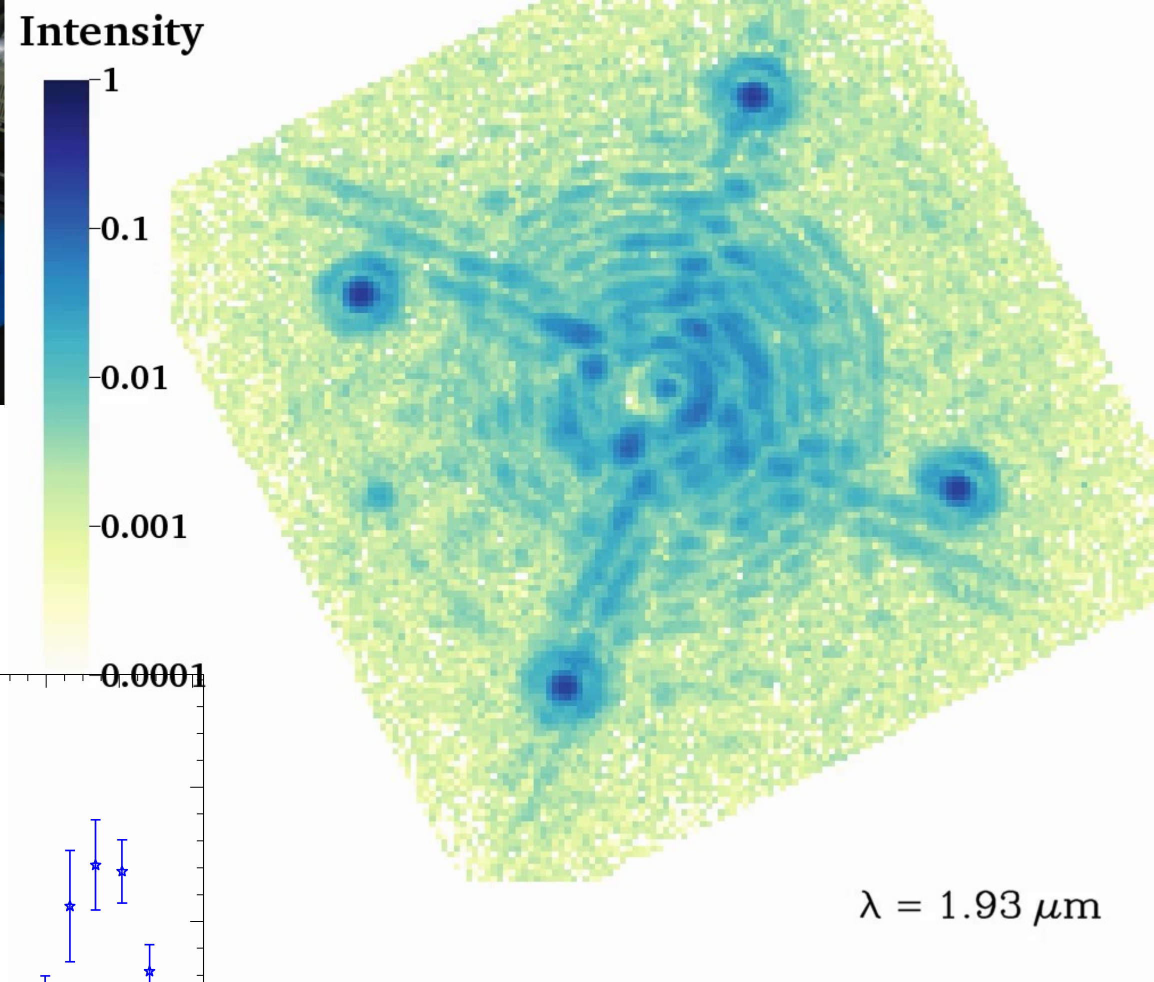


SCExAO Light path

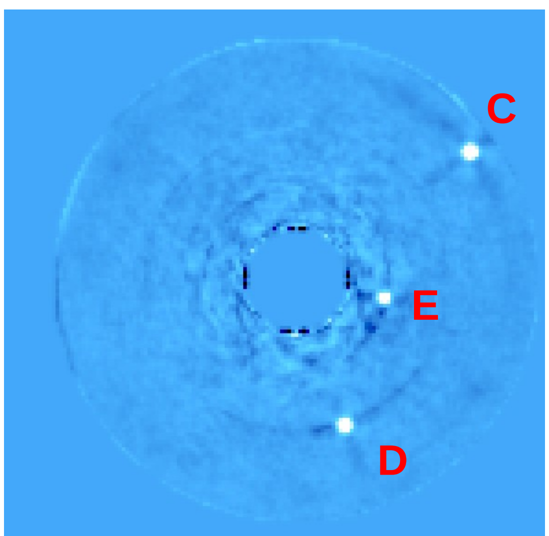


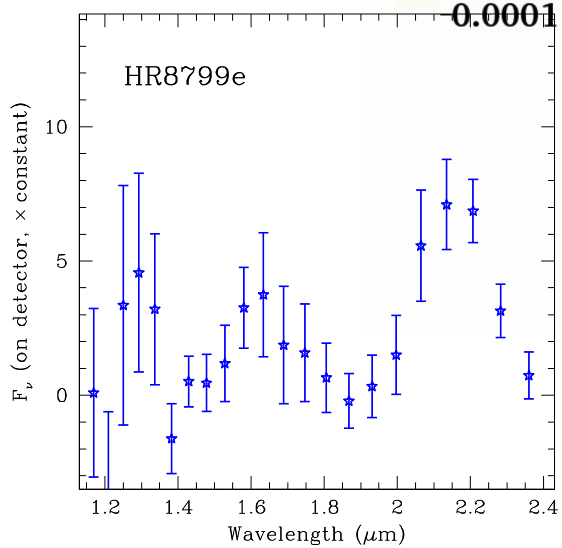
CHARIS IFS





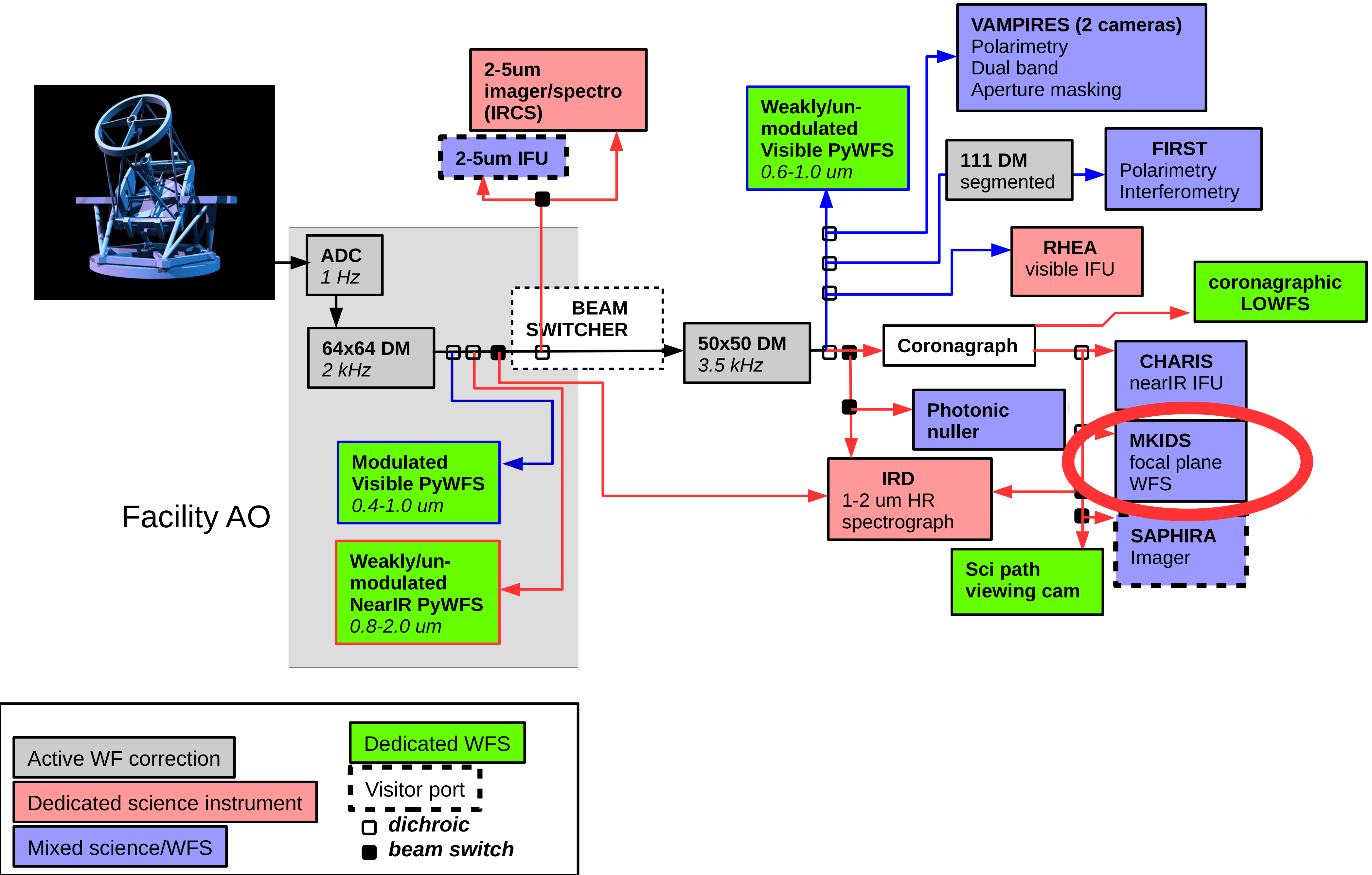
HR8799





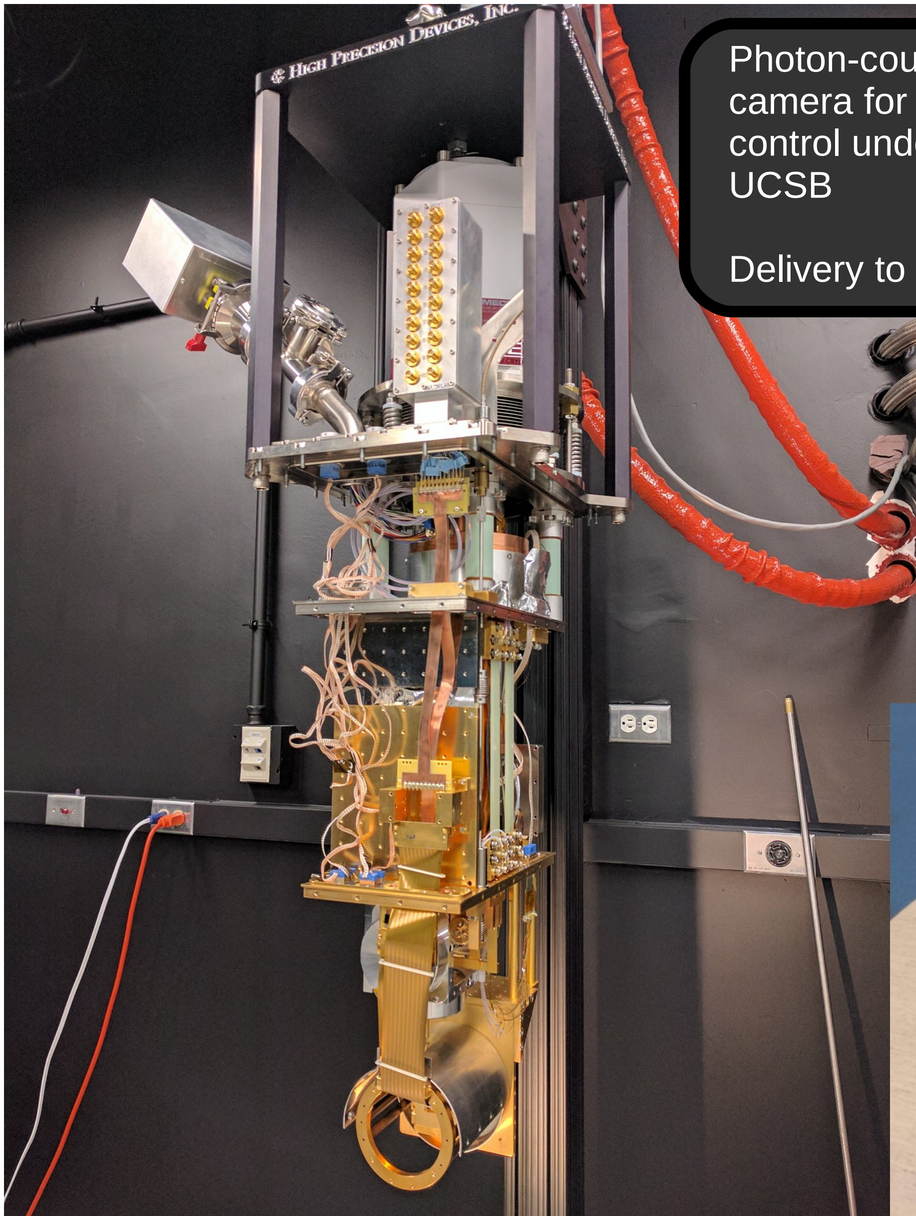
HR8799 Observations by J. Chilcote & T. Groff preliminary data processing by T. Brandt

SCExAO Light path



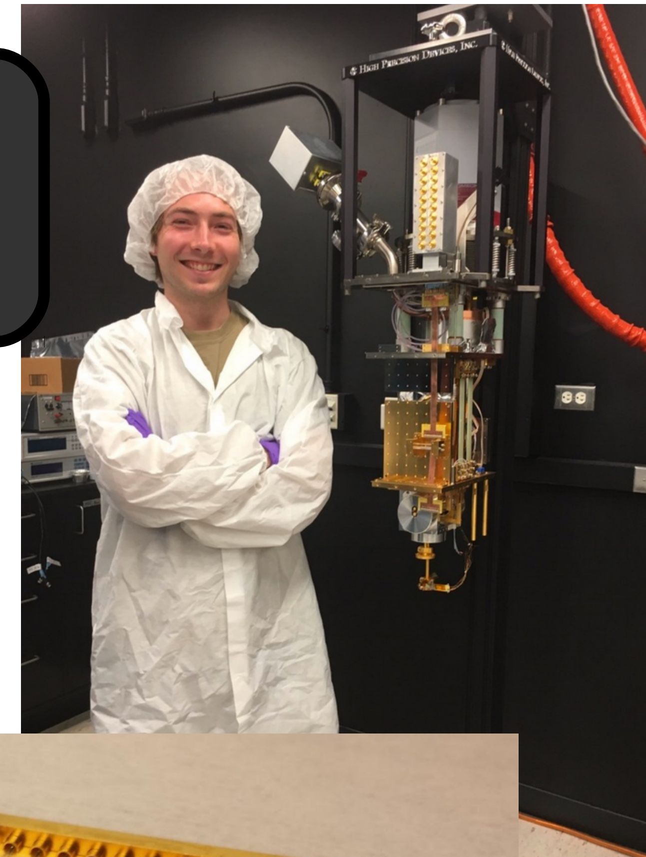
MKIDS camera (built by UCSB for SCExAO)

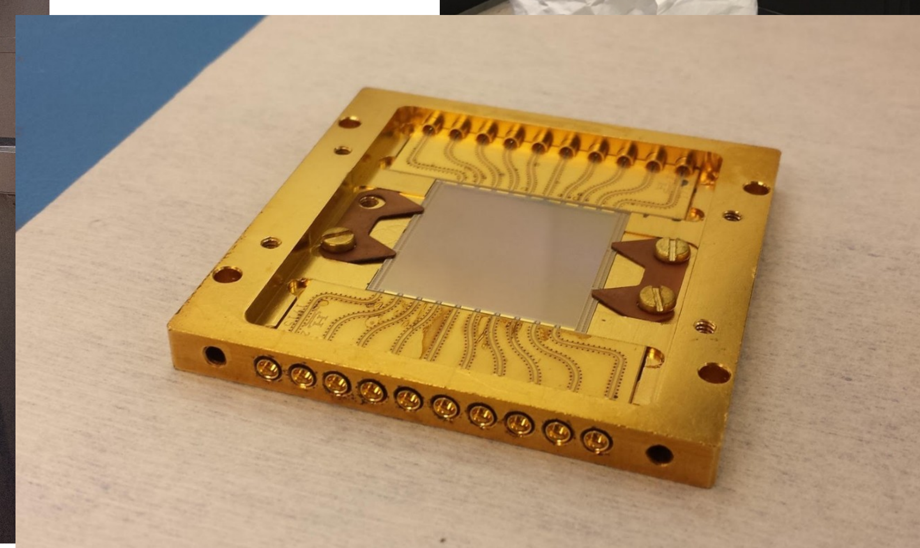
Photon-counting, wavelength resolving 140x140 pixel camera

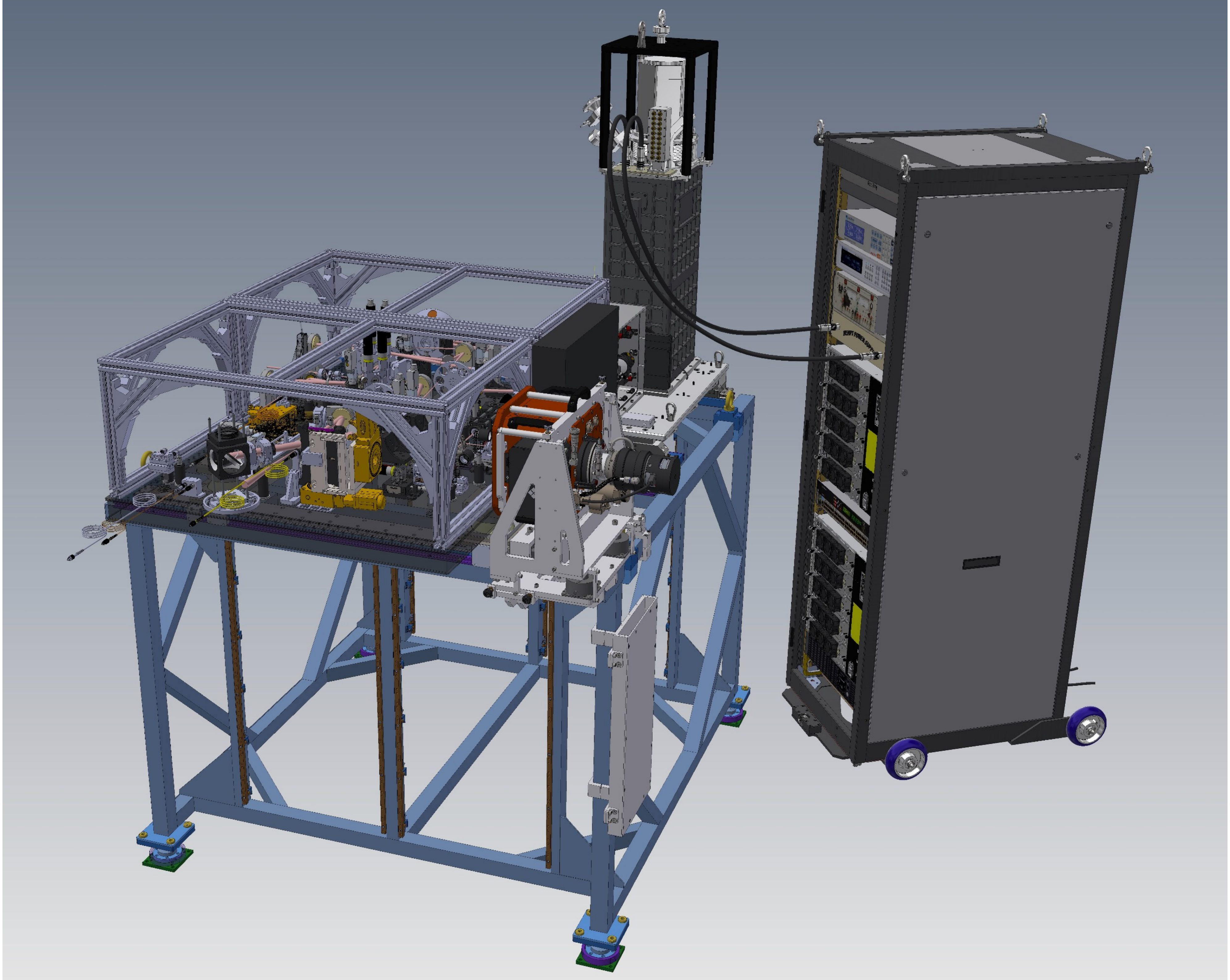


Photon-counting near-IR MKIDs camera for kHz speed speckle control under construction at UCSB

Delivery to SCExAO in sept 2017







Building community RTC / Software Ecosystem

Provide low-latency to run control loops

→ Use mixed CPU & GPU resources, configured to RTC computer system

On SCExAO, control matrix is 14,000 x 2000. Matrix-vector computed in 100us using 15% of RTC resources @ 3kHz

Portable, open source, modular, COTS hardware

→ No closed-source driver

→ std Linux install (no need for real-time OS)

→ using NVIDIA GPUs, also working on FPGA use

→ All code on github: **https://github.com/oguyon/AdaptiveOpticsControl**

Easy for collaborators to improve/add processes

→ Hooks to data streams in Python or C

→ Template code, easy to adapt and implement new algorithms

→ Provide abstraction of link between loops

→ Toolkit includes viewers, data logger, low-latency TCP transfer of streams

RTC code used at Keck, MagAO-X, OCA ...

→ *community support and development*

Data Stream Format

Uses file-mapped POSIX shared memory → multiple processes have access to data

Supports low latency IPC through semaphores → us-level latency

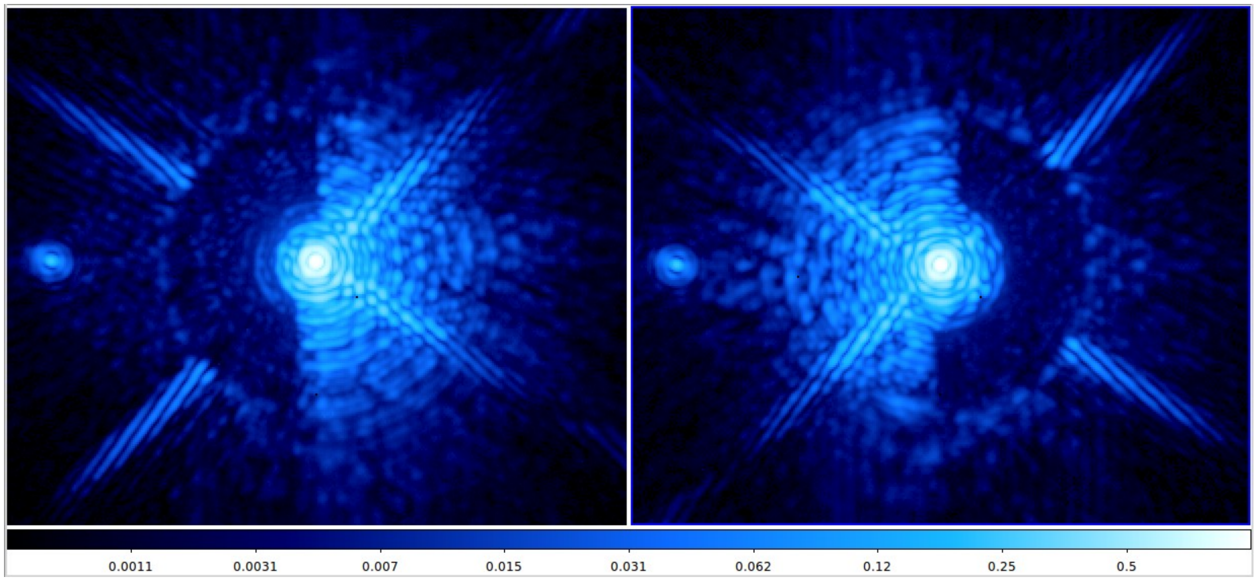
```c
typedef struct                /**< structure used to store data arrays */
{
    int used;
    int shmfd;                /**< if shared memory, file descriptor */
    size_t memsize;           /**< total size in memory if shared */

    IMAGE_METADATA *md;

    union
    {
        char *C;
        int *I;
        long *L;
        float *F;
        double *D;
        complex_float *CF;
        complex_double *CD;
        unsigned short int *U;
    } array;                  /**< pointer to data array */

    IMAGE_KEYWORD *kw;

    int sem;                  /**< number of semaphores in use     */
    sem_t **semptr;           /**< semaphore array */

    sem_t *semlog;            /**< semaphore for logging */


    char name[80];            /**< local name (can be different from name in shared memory) */

} IMAGE;
```

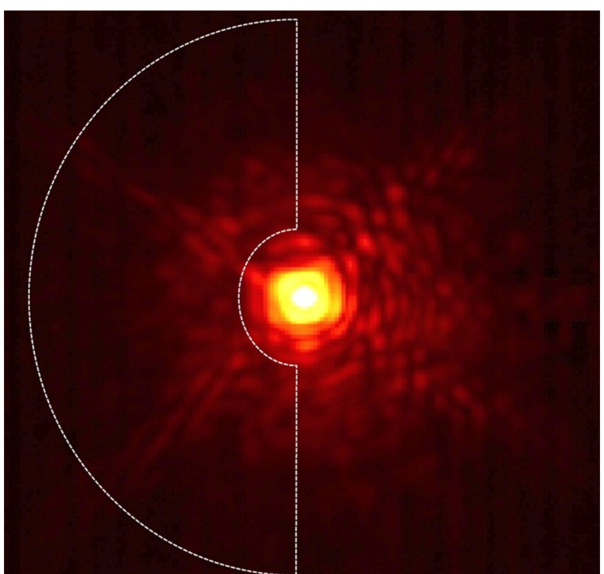
```c
typedef struct
{
    char name[80];              /**< image name */

    long naxis;                 /**< number of axis */
    long size[3];               /**< image size */
    long nelement;              /**< number of elements in image */
    int atype;                  /**< data type code */

    double creation_time;       /**< creation time (since program start) */
    double last_access;         /**< last time the image was accessed  (since program start) */
    struct timespec wtime;

    int shared;                 /**< 1 if in shared memory */

    int write;                  /**< 1 if image is being written */
    int status;                 /**< 1 to log image (default); 0 : do not log: 2 : stop log (then goes back to 2) */
    long cnt0;                  /**< counter (incremented if image is updated) */
    long cnt1;                  /**< in 3D rolling buffer image, this is the last slice written */

    long NBkw;                  /**< number of keywords */

} IMAGE_METADATA;
```

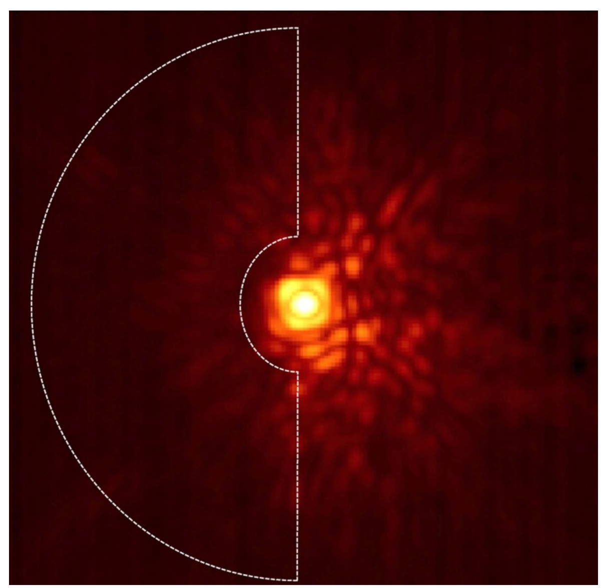
```c
typedef struct
{
    char name[16];          /**< keyword name */
    char type;              /**< N: unused, L: long, D: double, S: 16-char string */

    union {
        long numl;
        double numf;
        char valstr[16];
    } value;

    char comment[80];

} IMAGE_KEYWORD;
```

Drivers written for:
OCAM2k, BMC DM, SAPHIRA camera, InGaAs cameras

Collaboration with OCA: speckle







Speckle nulling, in the lab and on-sky (no XAO).

Experience limited by detector readout noise and speed.

KERNEL project: C-RED-ONE camera.

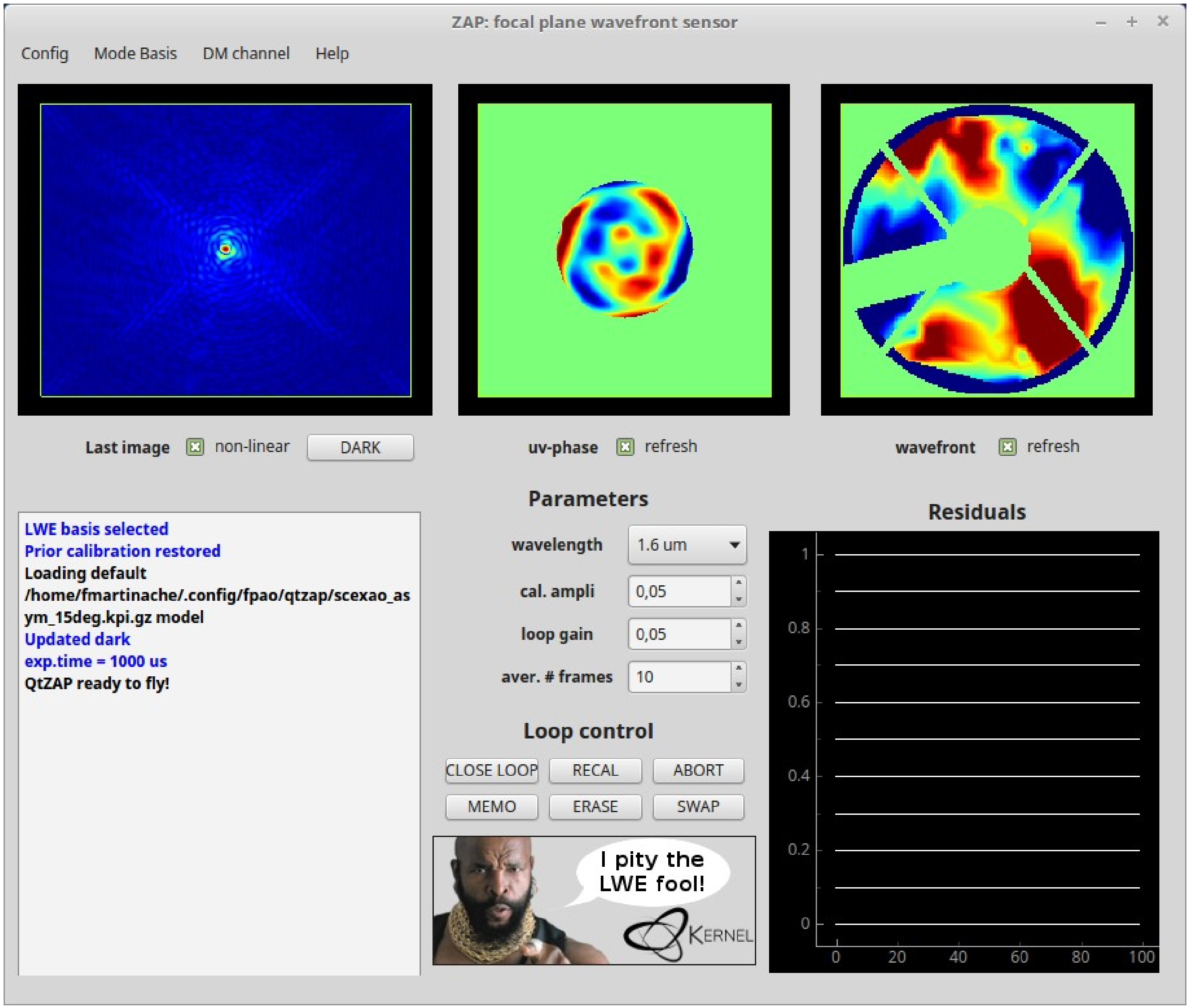
From:
- 114 e- RON
- 170 Hz frame rate

To:
- 0.8 e- RON
- 3500 Hz frame rate

Expect some updates

Observatoire de la CÔTE d'AZUR

OCA/KERNEL – developed software



- Address NCPA
- Asymmetric mask (pupil)
- On-sky closed-loop control
- Focal plane based WFS Low-order (Zernike and LWE) modes.
- mode compatible with coronagraphy in development

KERNEL

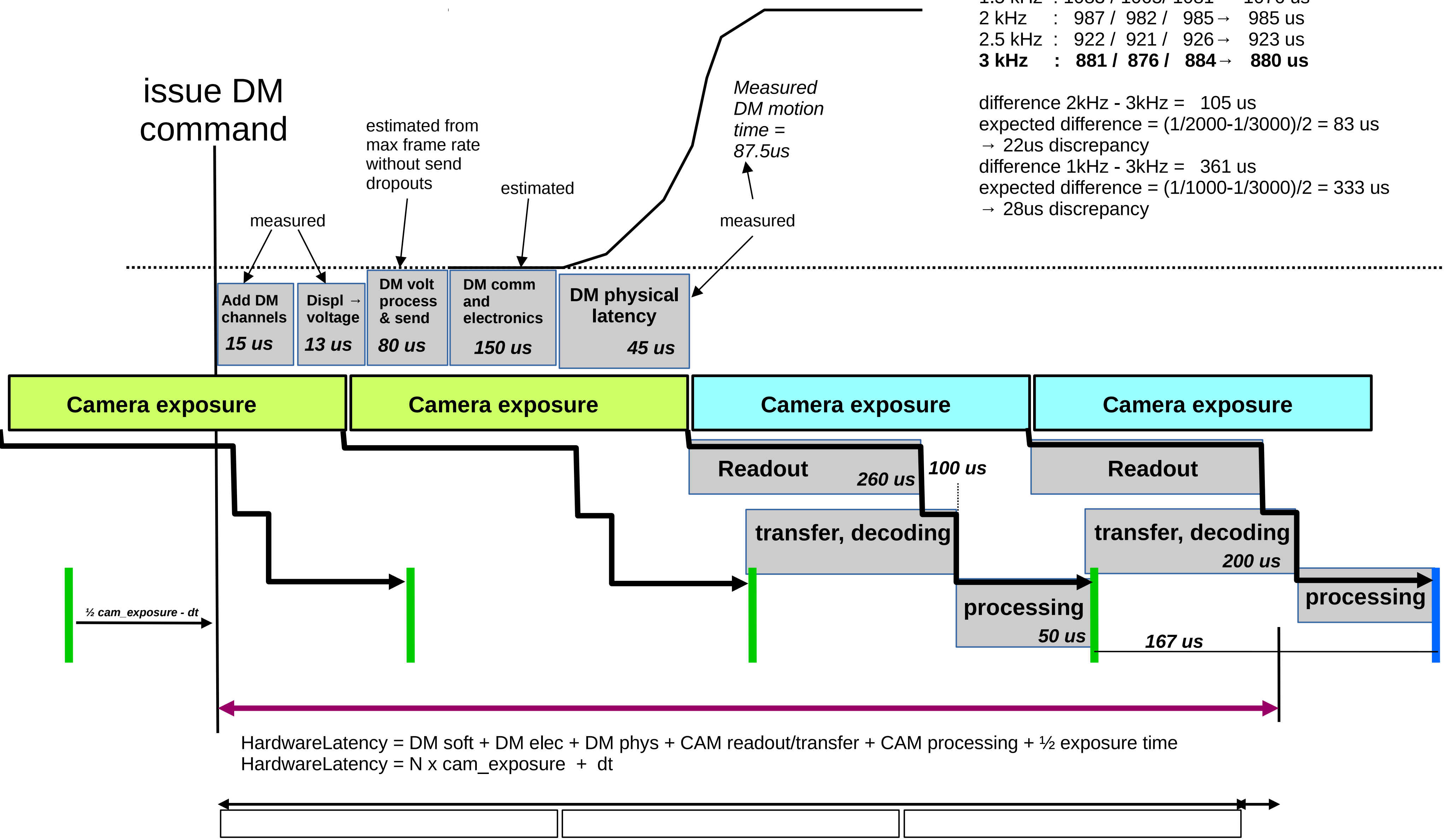
Hardware Latency measured on SCExAO

Definition:
Time offset between ***DM command issued****, and* ***mid-point between 2 consecutive WFS frames with largest difference***

SCExAO measured hardware latencies:

1 kHz : 1253 / 1260/ 1269 → 1261 us
1.5 kHz : 1083 / 1065/ 1081 → 1076 us
2 kHz : 987 / 982 / 985 → 985 us
2.5 kHz : 922 / 921 / 926 → 923 us
3 kHz : 881 / 876 / 884 → 880 us

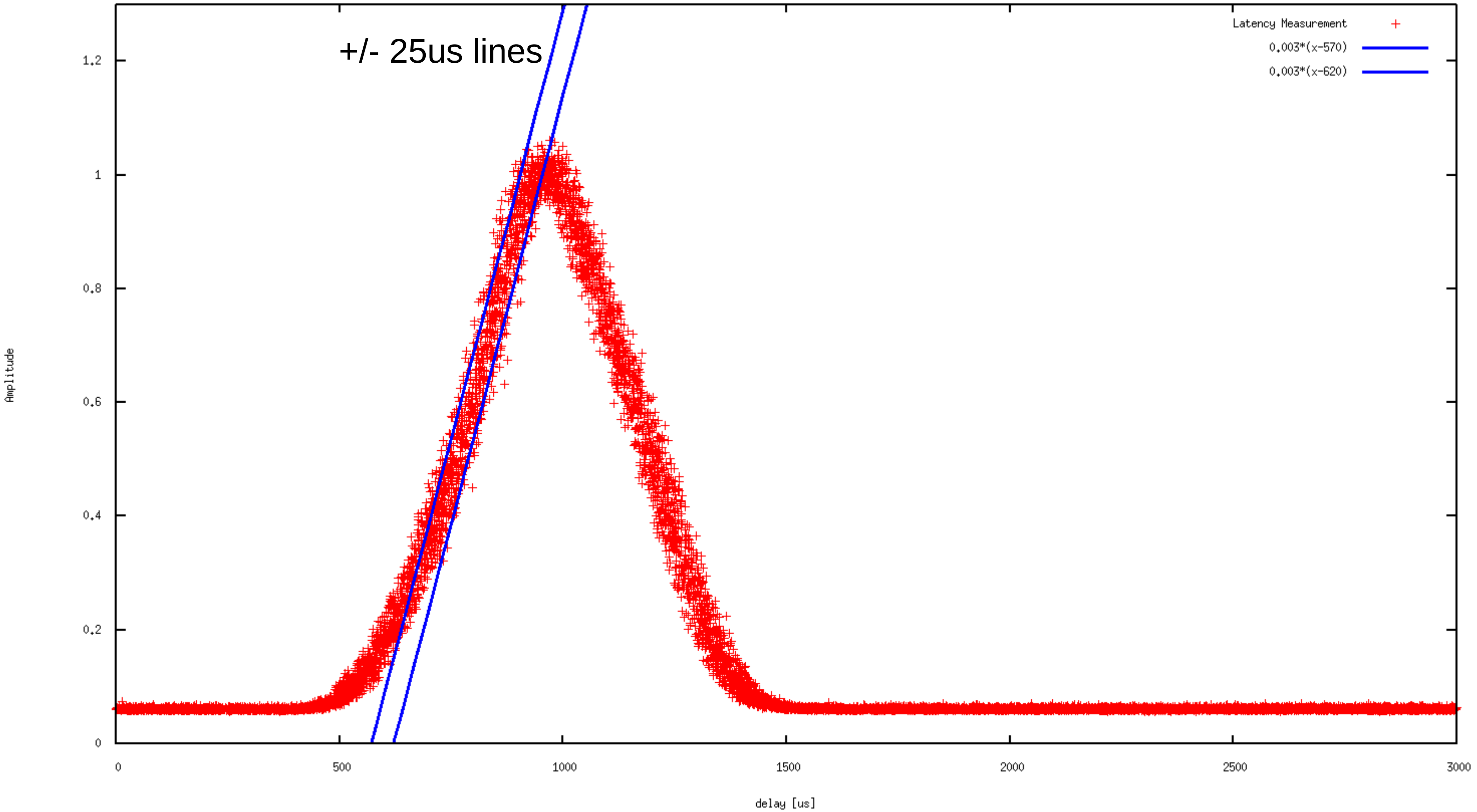
difference 2kHz - 3kHz = 105 us
expected difference = (1/2000-1/3000)/2 = 83 us
→ 22us discrepancy
difference 1kHz - 3kHz = 361 us
expected difference = (1/1000-1/3000)/2 = 333 us
→ 28us discrepancy



HardwareLatency = DM soft + DM elec + DM phys + CAM readout/transfer + CAM processing + ½ exposure time
HardwareLatency = N x cam_exposure + dt

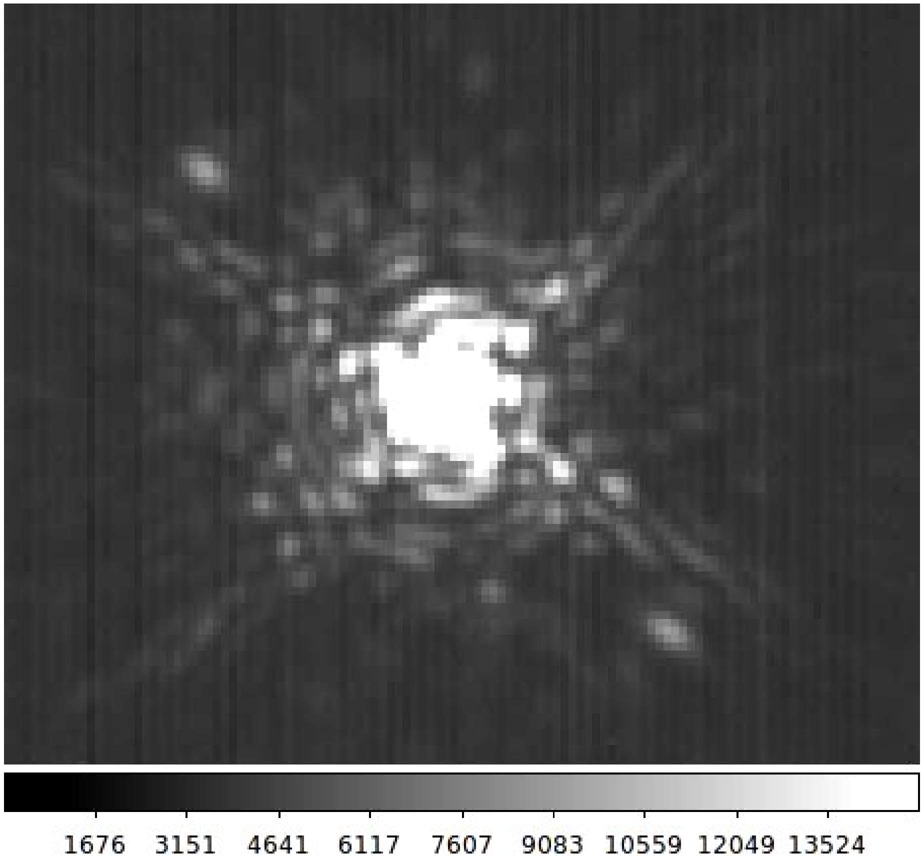
Hardware Latency measured on SCExAO

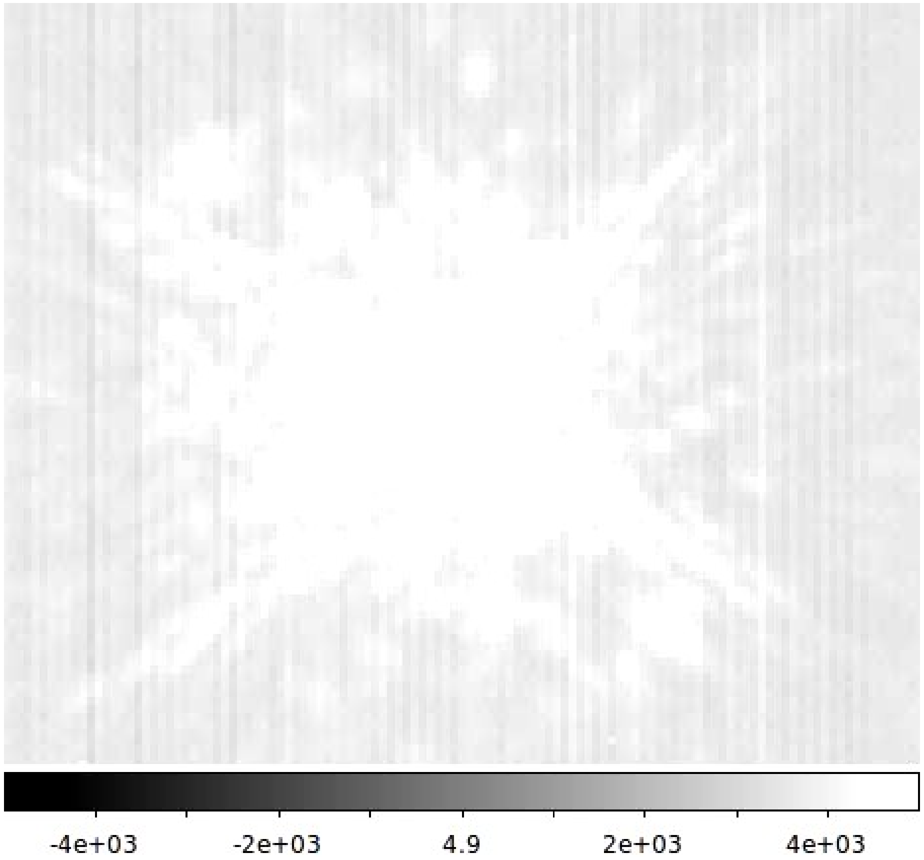
Total jitter <20us RMS = 6% of loop iteration @ 3kHz
(Camera readout + TCP transfer + processing + DM electronics)
Max jitter <40us



Synchronizing camera stream to DM (170 Hz)

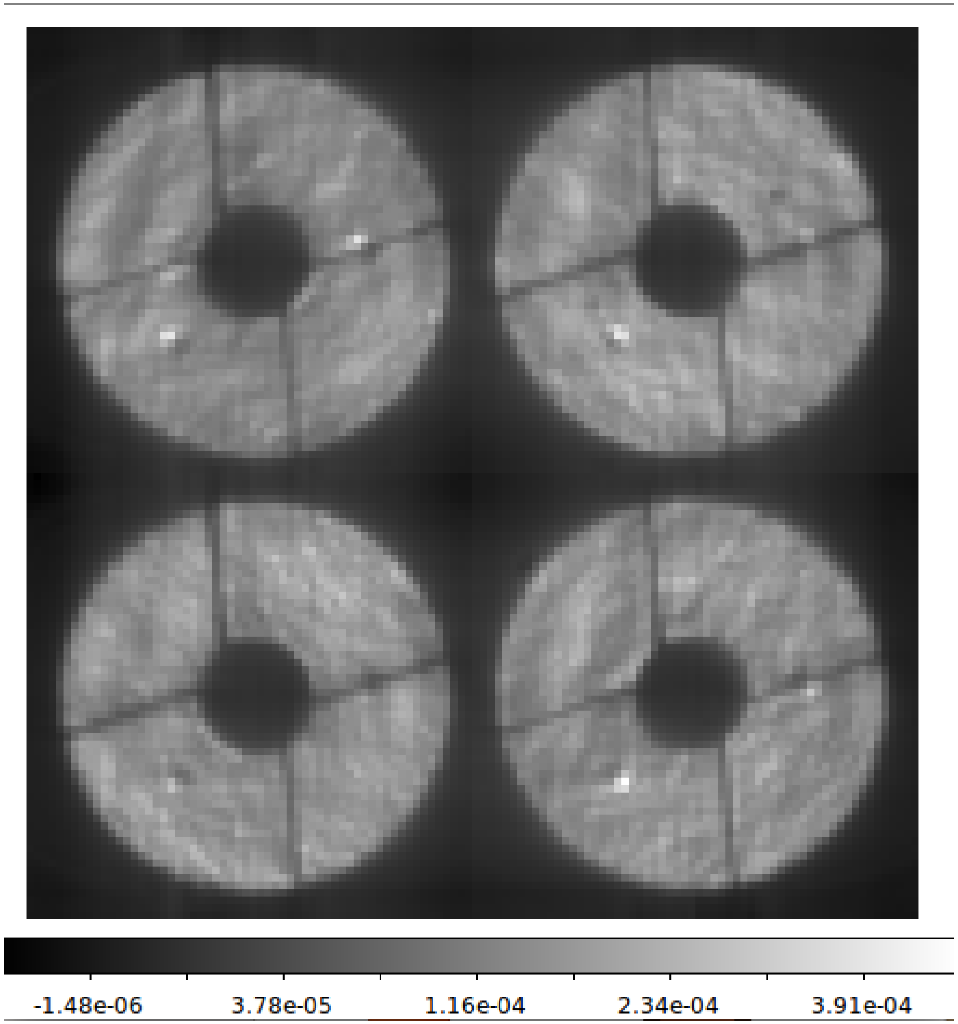
6kHz DM modulation swaps between 2 diag patterns

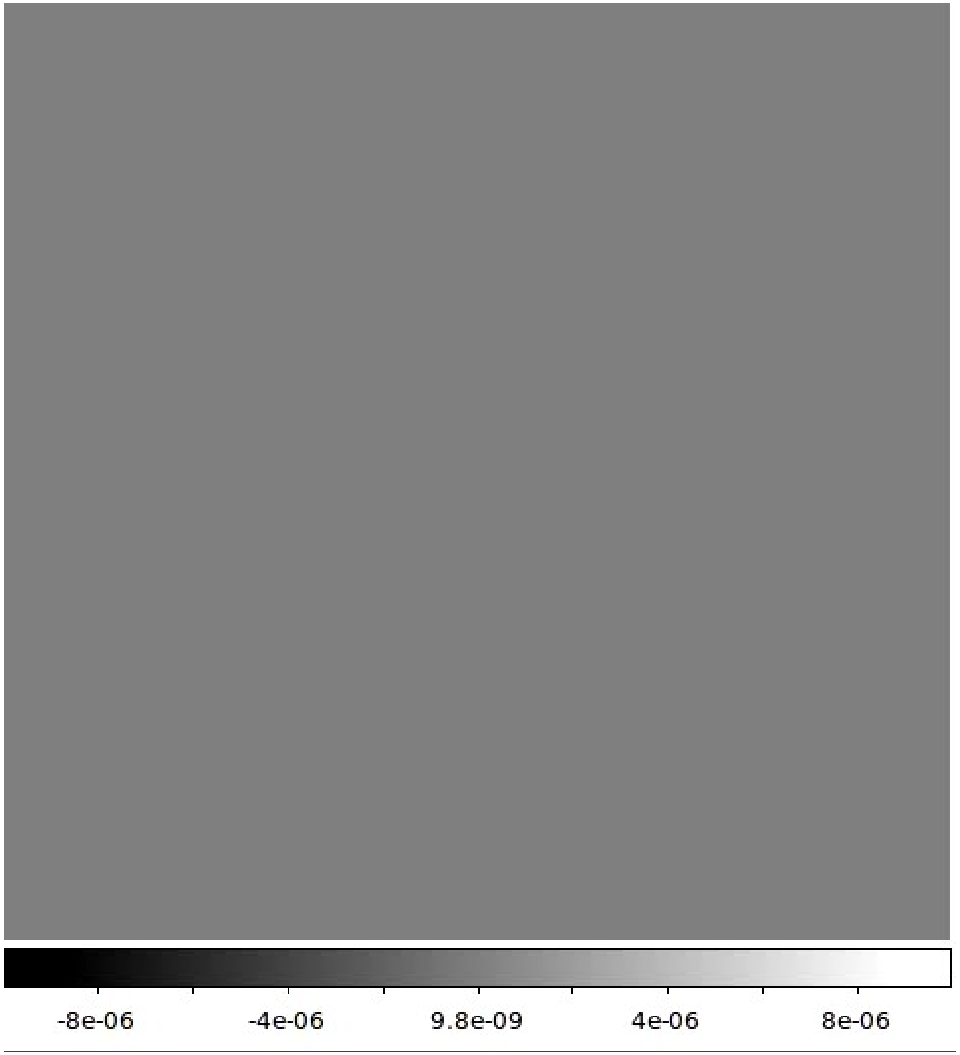




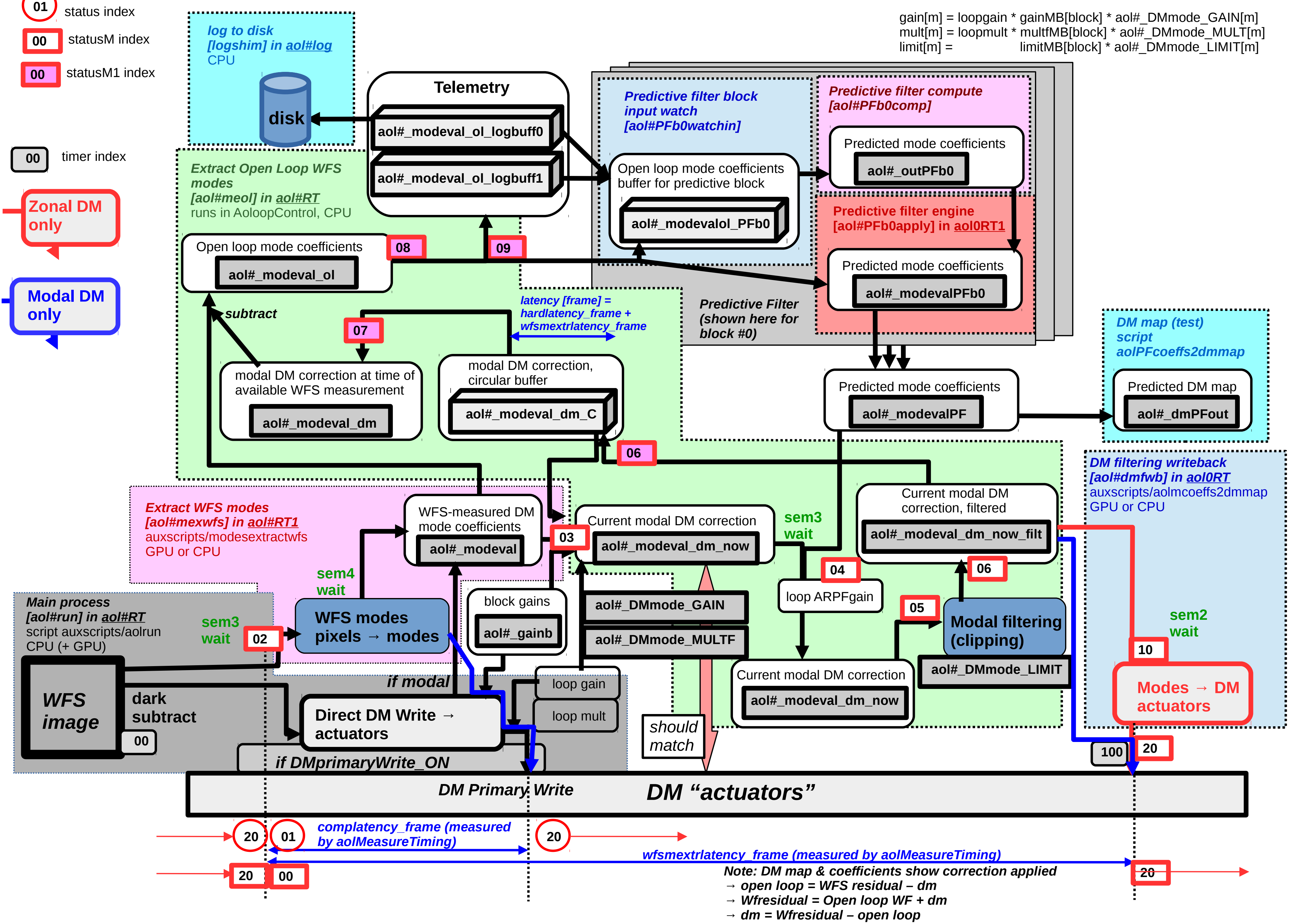
Measuring system response matrix at 3kHz

Full speed DM modulation to measure response matrix
DM motion occurs during EMCCD frame transfer
2000 modes measured in 1.33 sec @ 3kHz, 2sec @ 2kHz. Multiple cycles averaged to build up SNR

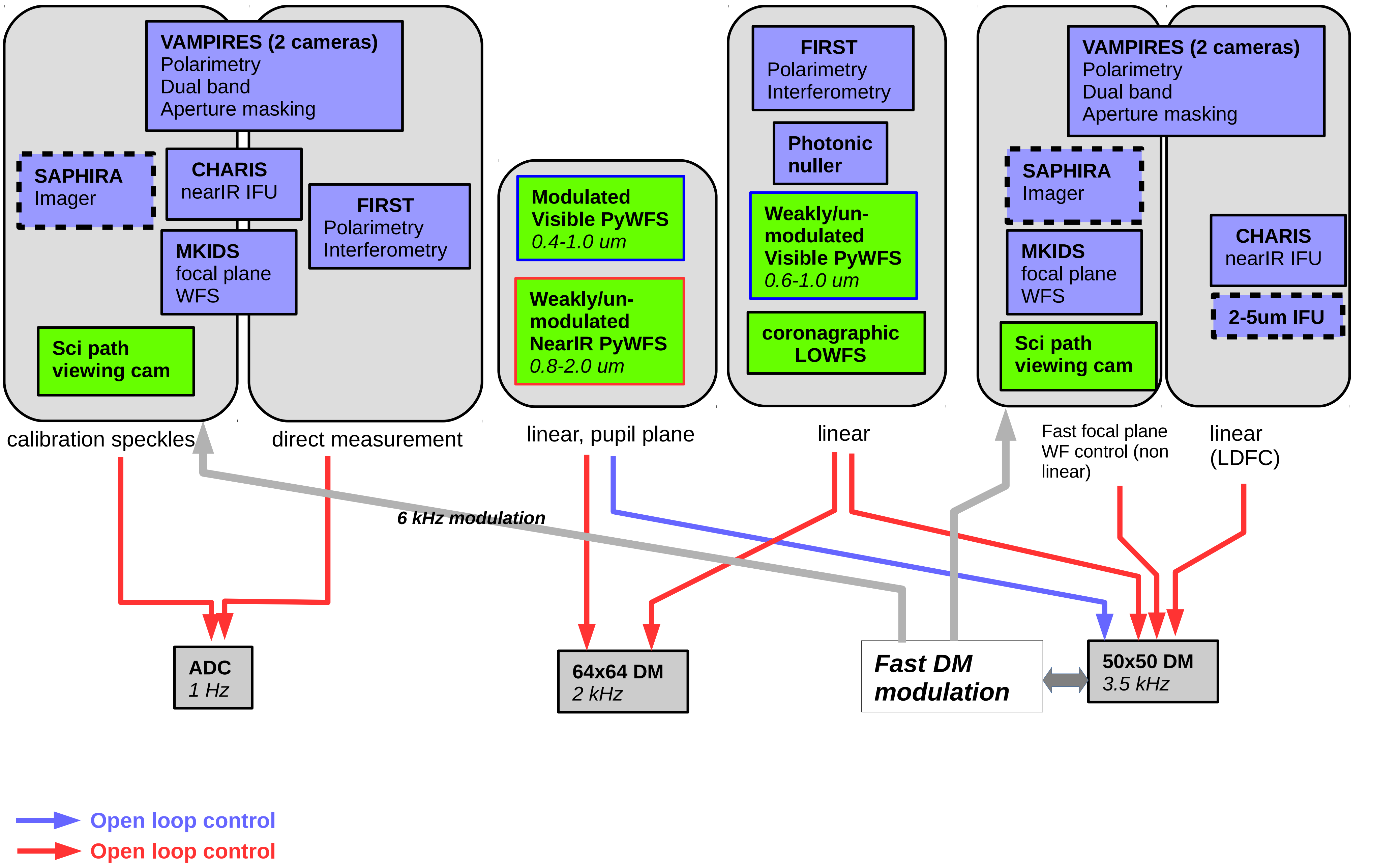




Data flow from WFS to DM

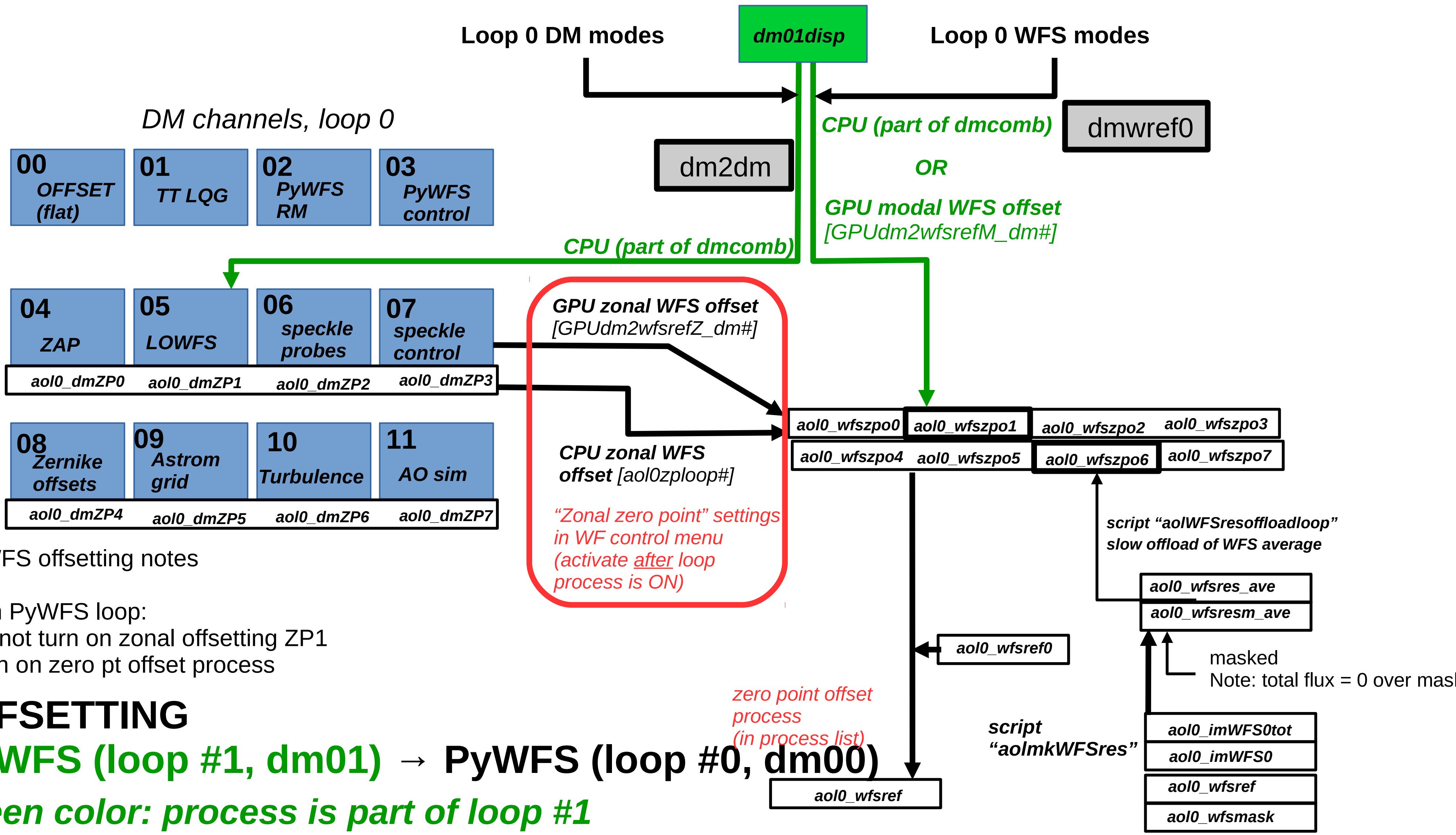


Control loops



Linking multiple control loops (zero point offsetting)

A control loop can offset the convergence point of another loop @> kHz (GPU or CPU)
Example: speckle control, LOWFS need to offset pyramid control loop
THIS IS DONE TRANSPARENTLY FOR USER → <u>don't pay attention to the diagram below !</u>



LOWFS offsetting notes

From PyWFS loop:
- Do not turn on zonal offsetting ZP1
- Turn on zero pt offset process

OFFSETTING

LOWFS (loop #1, dm01) → PyWFS (loop #0, dm00)

Green color: process is part of loop #1

Using SCExAO instrument

← slack channel to coordinate instrument use over multiple continents

December 19th, 2016

bnorris 1:34 AM
Crap! Yes sounds like it's best to leave it off. Well if you do go up be sure to look out for snow kangaroos. They're deadly.

olivier 1:36 AM
it's low probability that a significant leak would develop... but I prefer to play it safe

I'll go up in the day with DC if I can and then we can both test with the DM

I'm in simulator software mode, so you can use the superK as you wish

bnorris 1:38 AM
Ok cool, thx. I'll turn it off when I'm done.

olivier 1:38 AM
OK

olivier 1:45 AM
I'll try (later tonight) to run the DM at reduced voltage - this should be safe and should allow enough stroke for the dmflat

bnorris 1:49 AM
ok cool

olivier 3:20 AM
B: let me know if you need flat - I'm set it up so that is uses 80V instead of 120V. The flat may not be perfect but it's close.

bnorris 3:21 AM
It's ok for now, I don

't need it flat at the moment.

olivier 3:21 AM
OK - let me know when you need it

December 20th, 2016

bnorris 6:13 AM
Can I turn on the superk?

olivier 7:54 AM
yes... I see you just did

do you need a DM flat (I keep the DM off ... haven't been able to go to summit check things out)

bnorris 7:56 AM
Yeah that would be good. I'm going to stop in < 1 hr. Also was wondering about turb simulator.

olivier 7:56 AM
mmm... I prefer to keep the DM off for now. Can you wait another 12hr ?

bnorris 7:57 AM
Yep sure.

olivier 7:57 AM
OK - feel free to use bench as you need (without DM) for now. I am working on a "simulated" SCExAO for now.

just turn off superK when done

bnorris 7:58 AM
Ok. I like the sound of the simulated scexao - let's just use that all the time, instead of the real one.

olivier 9:15 PM
B: DM can be used safely

nem 9:15 PM
great, what was the humidity? like 2% right

olivier 9:21 PM
14%

a bit high, but safe

nem 9:22 PM
hmm, thats high for the vac pump being on

olivier 9:24 PM
B: let me know when you need DM flat

olivier 10:17 PM
OK - I'm keeping full control of DM until someone else screams

Conclusions

GSMTs can image and characterize habitable planets around nearby M-type stars
Significant progress is being made in high contrast imaging techniques…
→ testing on current large telescopes is critical to be ready & efficient for GSMTs era

SCExAO is a powerful platform for testing and deploying new techniques (hardware, algorithms).
Daytime testing with internal source → nighttime on-sky validation

Coordinated development with MagAO-X (→GMT), Keck (→TMT), SPHERE upgrades (→ELT)
Major ongoing effort to develop software ecosystem to facilitate algorithm development and test across observatories/instruments/labs.

Multiple opportunities to get involved:
Test algorithms, reduce data, new hardware, looking for exoplanets, cool project for postdoc fellowship ?
→ talk to us