UNIBAP
SPACE SOLUTIONS

Unibap Loom

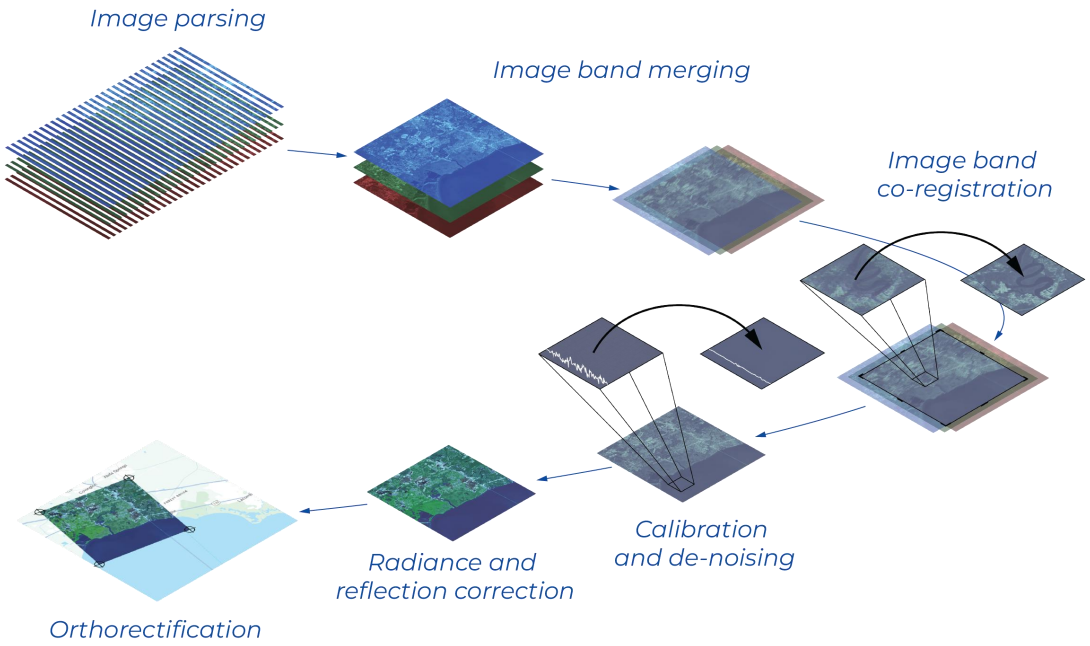
A real-time hyperspectral preprocessing pipeline for in orbit processing

Oskar Flordal, Oliver Petri, Tomas Måhlberg

2025-09-15

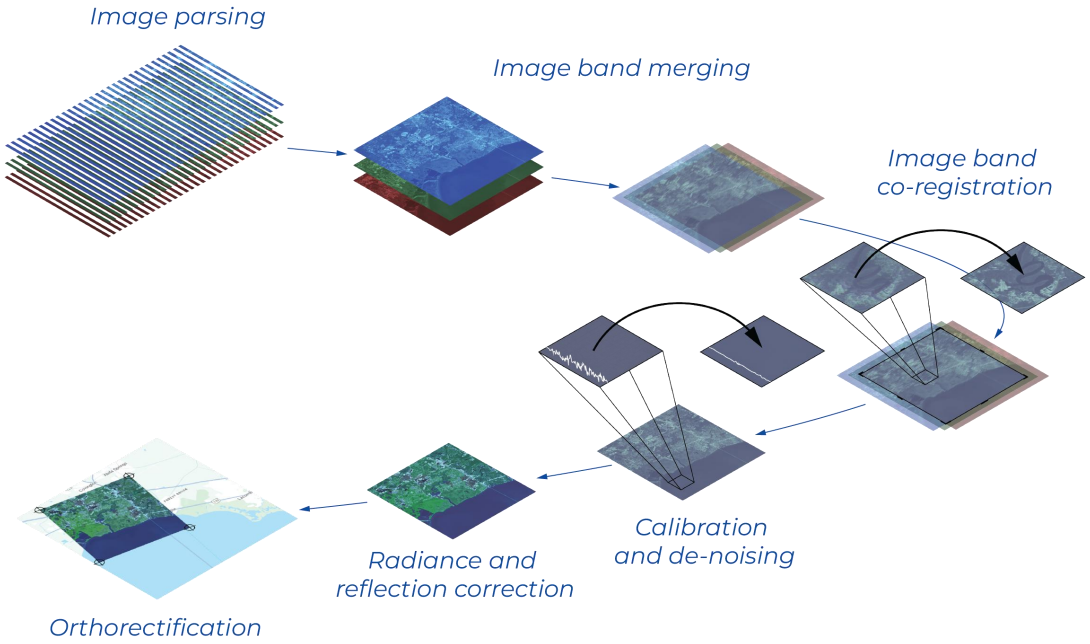
On board processing problem statement

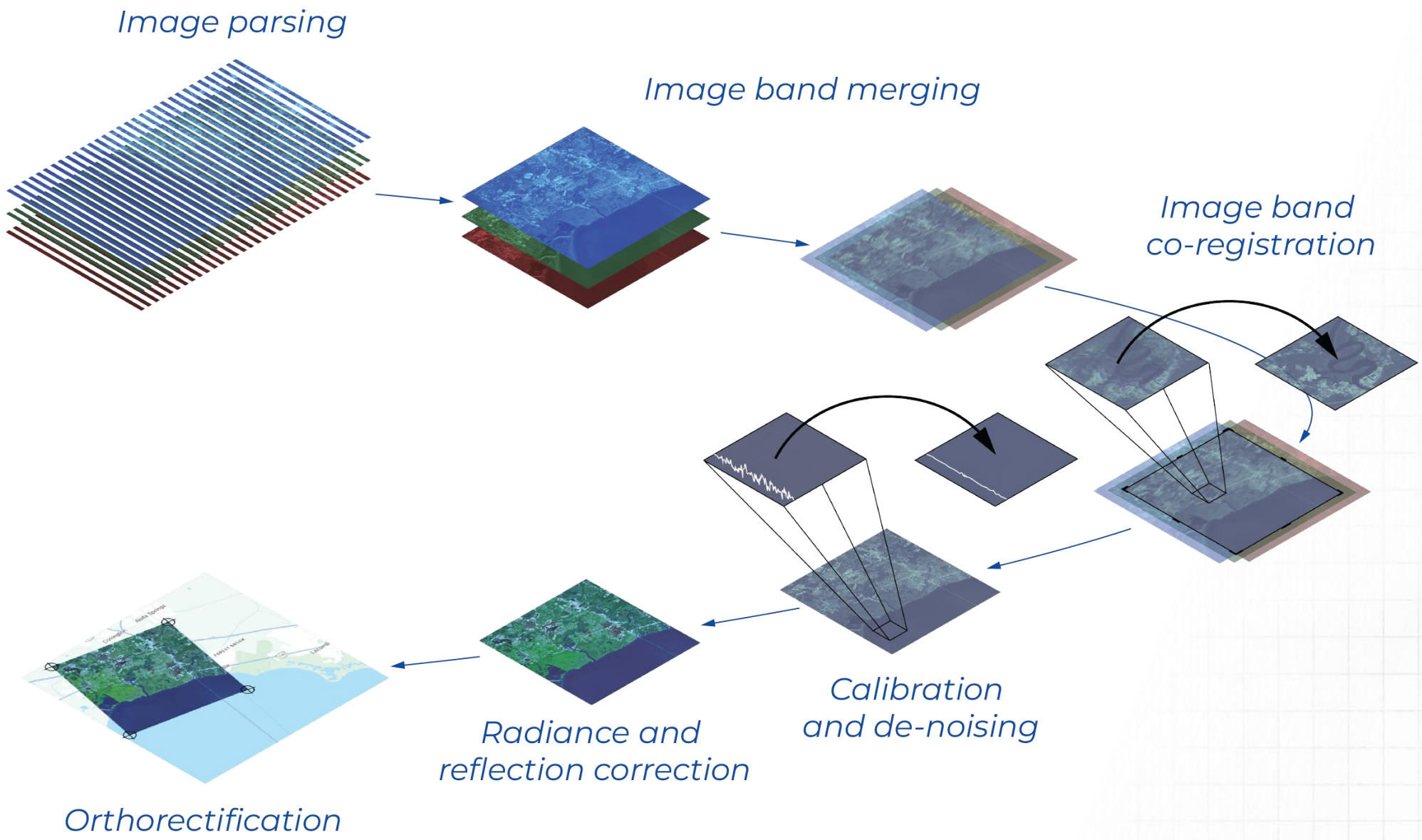
- Increase utilization of on-board resources
  - Lower latency
    - To ground
    - Tip and cue
  - Higher bandwidth from sensor
  - Novel use cases



Background: Why Pre-Processing is Critical

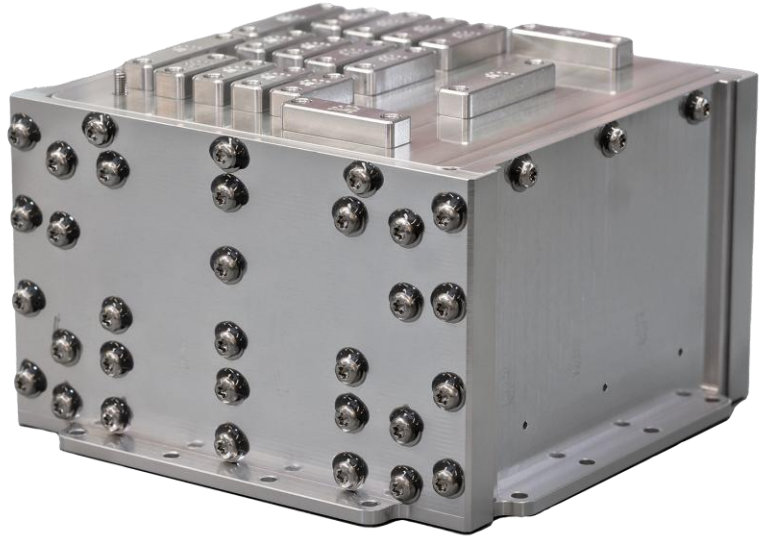
- Pre-processing for Hyperspectral/Multispectral
- Initial version built for iX5/Simera imagers
- Loom goals
  - High speed – imager bandwidth limited
  - Simplify neural network deployments



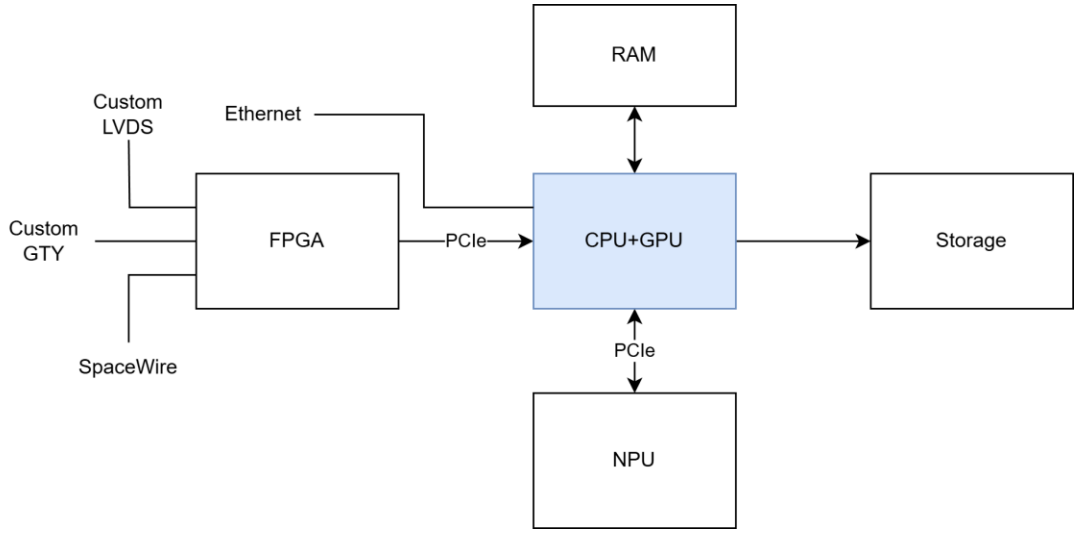


Platform targets

	iX5	iX10
APU RAM	2 GB	24 GB
GPU FLOPS32	42 GFLOPS	~1TFLOPS
MEM BW	5.7 GB/s	30+ GB/s
GPU API	OpenCL	OpenCL/HIP



Unibap HW crash course



- Ingest via FPGA to APU
- Shared memory CPU/GPU
- Process, then write to disk or NPU

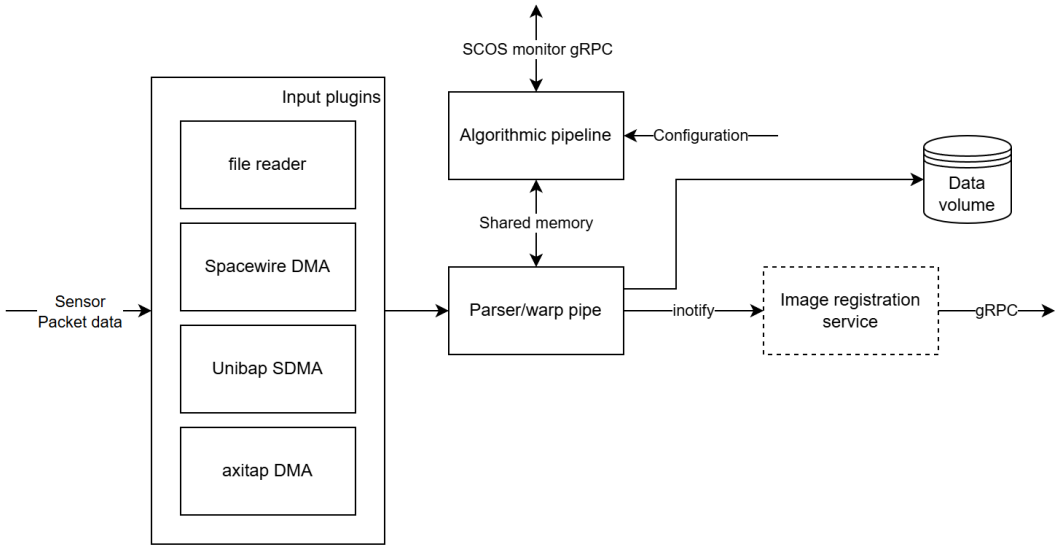
Performance targets

- iX5 1+ Gbit/s
- iX10 6+ Gbit/s

- Sub pixel interpolation errors
- Filters can be optimized for speed
- Positional errors on perfect telemetry <1 px avg

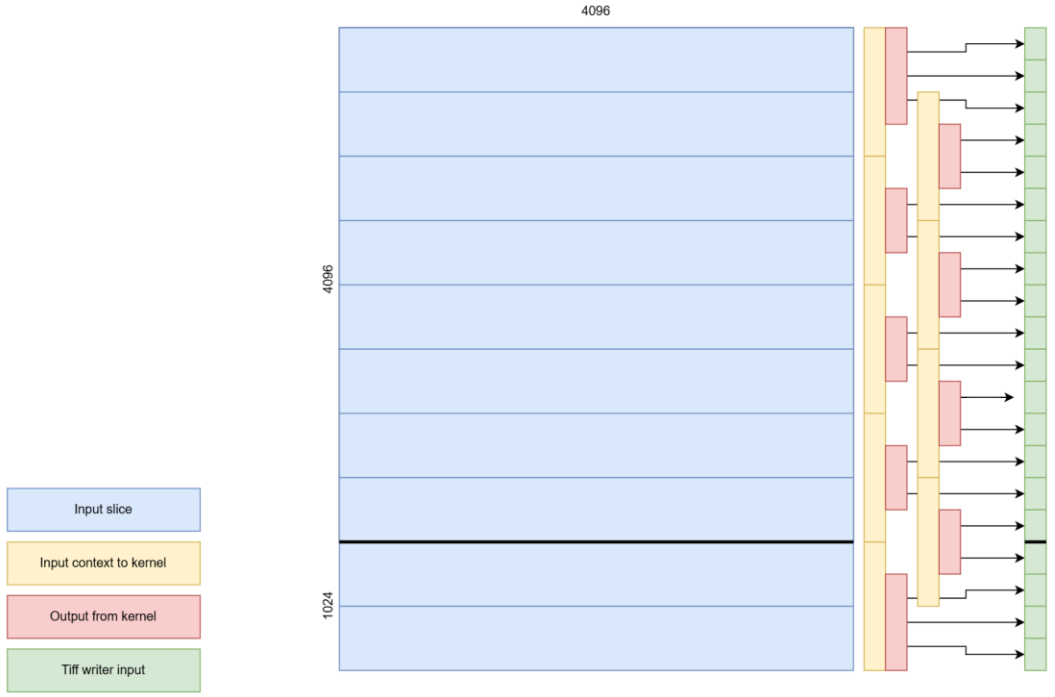
Overall architecture

- 1 pass operation
- Optimized for CPU/GPU architectures and handing data to neural network accelerators
- C++/HIP/OpenCL for anything that touches pixels
  - Python for support
- Main threads
  - Data ingest
  - Parsing
  - Coefficient calculation
  - Warper
  - Image output



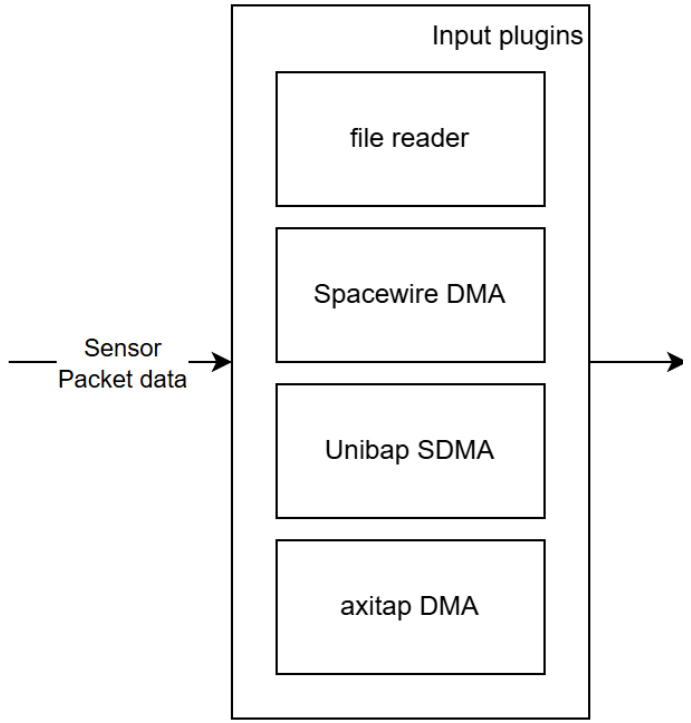
Slots and data passing

- Pipeline is run as a set of threads exchanging indices across queues
- Each interchange has a set of pre-allocated slots
- Context window kept small to minimize memory usage
  - Target < 256 MB on iX5



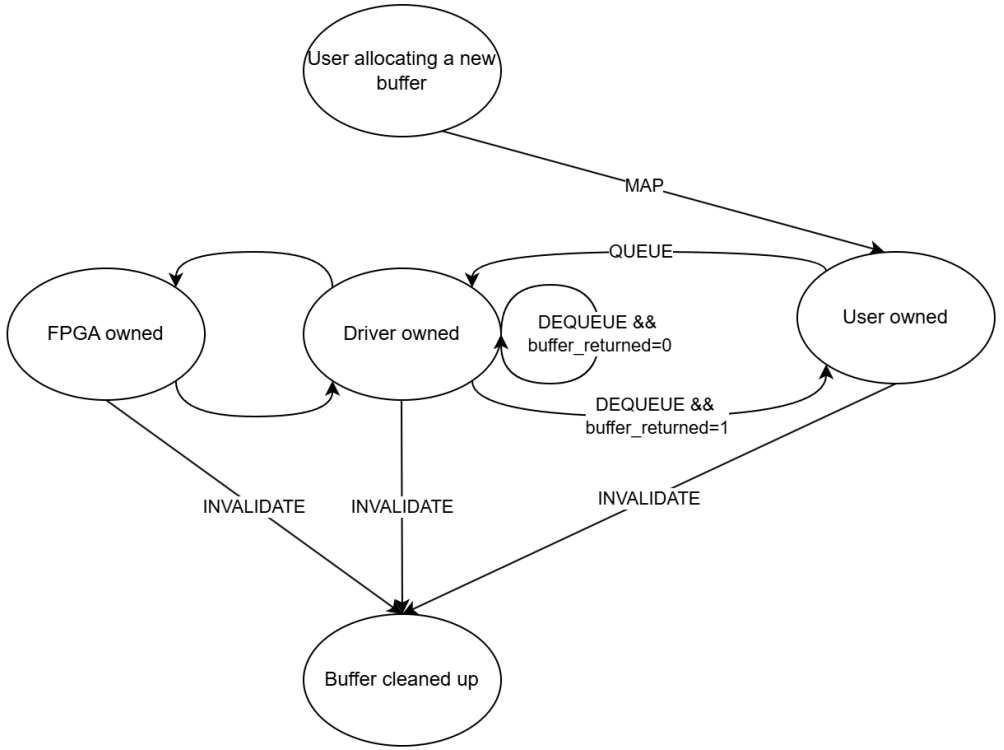
Data ingest

- Get data from various sources to GPU owned memory
- User owned buffers

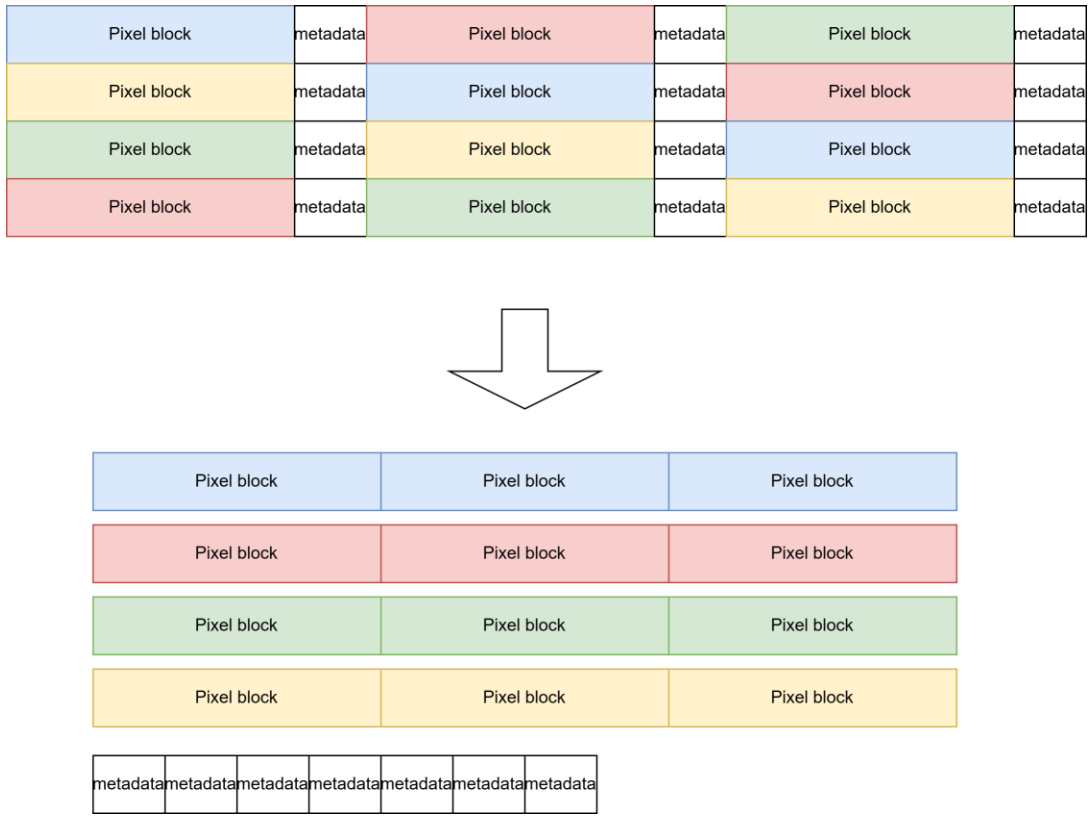


SDMA

- Primary way is SDMA
- Shared DMA across ~all interfaces on the FPGA
- Minimize system overhead, all to user owned buffers



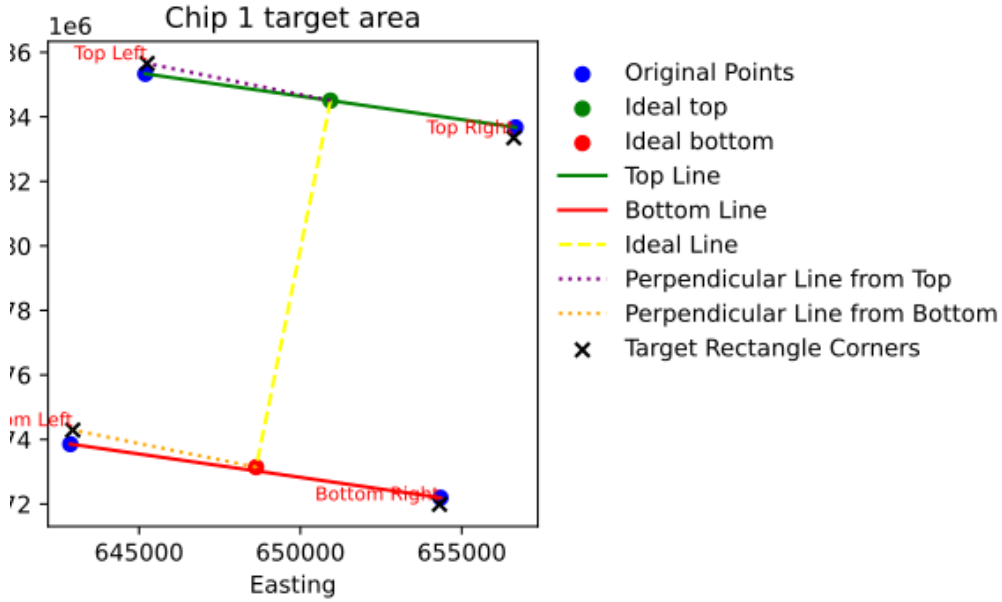
Parser



- Go from vendor representation to something that is easier to work with for upcoming steps
  - Separated pixel buffers
  - Timestamps
- Can ideally be moved to FPGA (or imager(!))

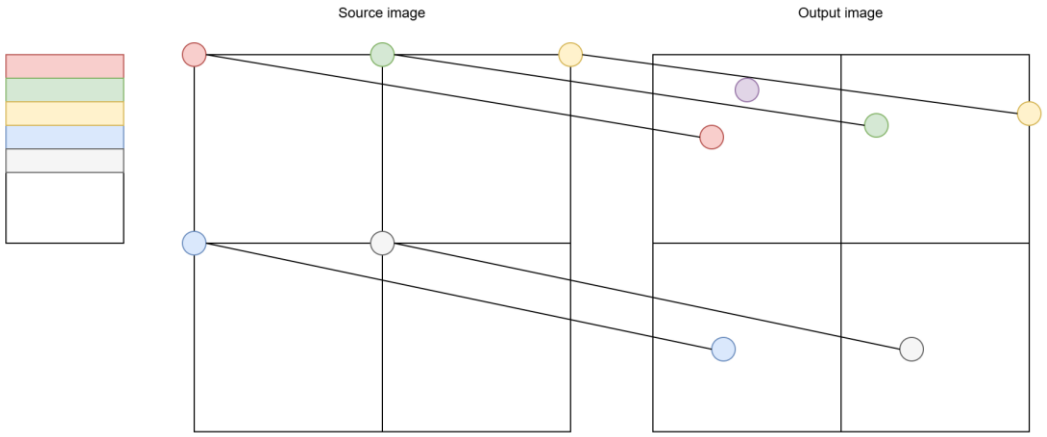
Coefficient pipe

- Initial version in python, communicating with shared memory
- Decide target
  - Decide individual warp pairs
  - RPC
- Set rad to reflectance weight and level adjustment
- Can be improved



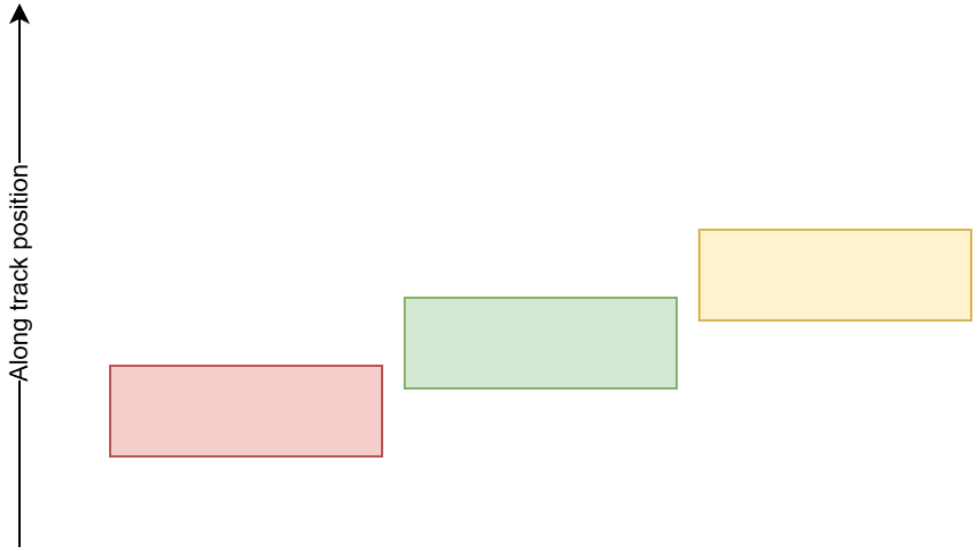
Warp pipe

- Sample source from well defined output pixels
- OpenCL for iX5 support
  - Going to HIP for future versions
- Inefficient in first version due to copy



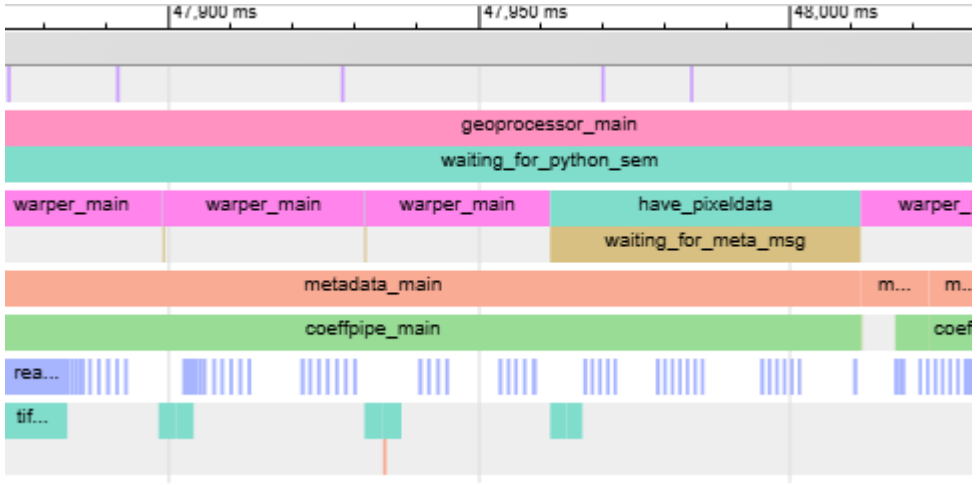
Output

- Custom Cloud optimized GeoTIFF writer
- Writes a slot right to memory in continuous chunks



Performance

- Home built tracing for first version
- Make sure all threads have work
- Performance caveats
  - Slower disks on development kits
  - Python!



info@unibap.com

Unibap Space Solutions AB (publ).
Västra Ågatan 16, 5 tr
SE-753 09 Uppsala Sweden

unibap.com