not yet final…

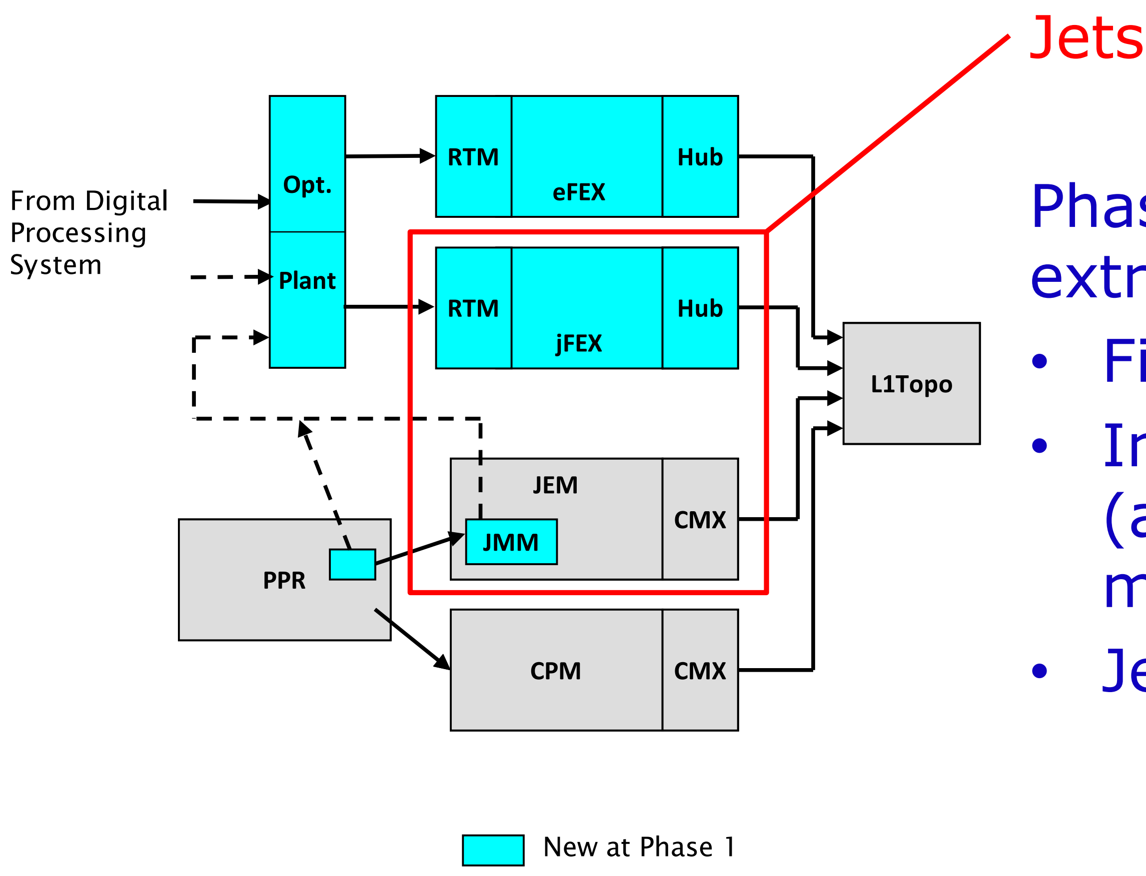
jFEX

Uli

Some questions flagged

Intro / overview / issues

Intro: L1Calo Phase-1 System / Jets



Phase-1: jet feature extractor jFEX

- Finer granularity:
- Improve on jet finding (and MET measurement)
- Jet size…

Baseline jFEX input data

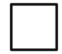
- Fibre bundles via patch panel / fibre re-bundling stage
- Granularity .1×.1 (η×φ)
- One electromagnetic, one hadronic tower per η×φ bin
- Baseline 6.4 Gb/s line rate, 8b/10b encoding,
  → 128 bit per BC

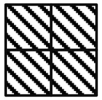
Stick to 6.4 ?

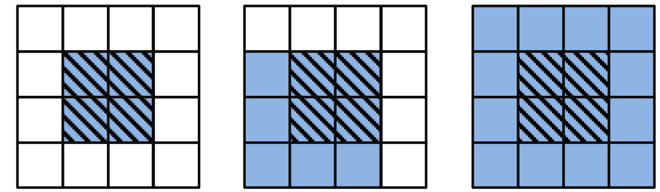
- Up to 16bit energy per tower → 8 towers per fibre

- LAr : pre-summed data from DPS (no BCmux)
- Tile … options…

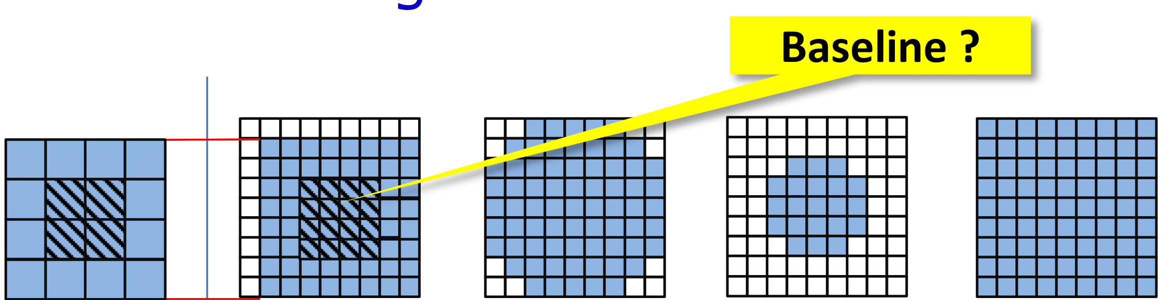
Sliding Window Algorithms

Jet elements (towers) 

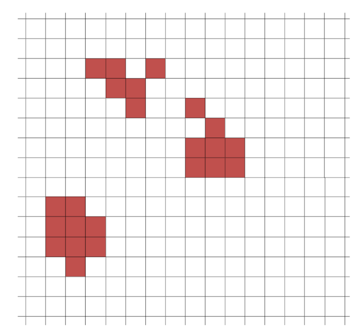
ROIs  environment 



- Increase dynamic range
- Improve granularity by factor of four, to 0.1×0.1 (η×φ)
- Slightly increase environment (0.9 × 0.9 baseline)
- Allow for flexibility in jet definition
- Fat jets to be calculated from high granularity small jets ?
- Optionally increase jet environment
- Assume fat tau algorithm resides in same FPGAs



Phase 0 Phase 1

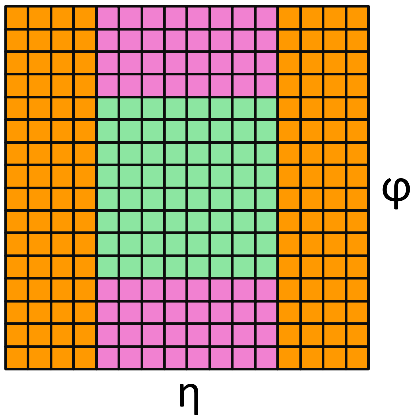


Data replication

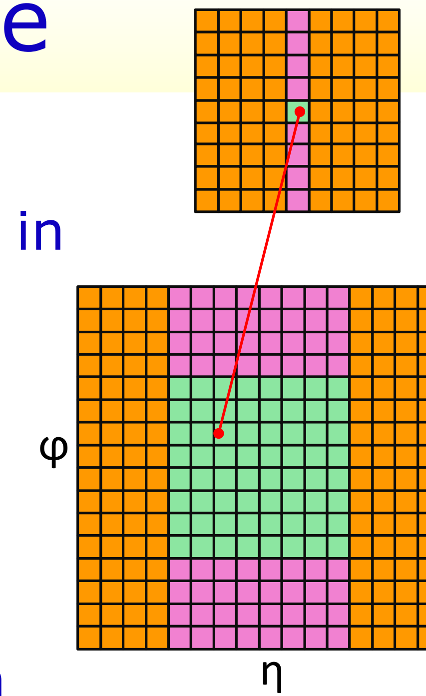
Sliding window algorithm requiring large scale replication of data

- Forward duplication only (fan-out), no re-transmission
- Baseline: no replication of any source into more than two sinks
- Eta-strip / phi quadrant organisation:
- **Fan-out in phi** handled at source only (DPS)
  - Transmit **"core"** and **"environment"** data
  - Duplication at the parallel end (on-FPGA), using additional Multi-Gigabit Transceivers
  - Allowing for differently composed streams
  - Minimizing latency
- **Fan-out in eta** handled at destination only
  - Baseline "far end PMA loopback"
  - Looking into details and alternatives (passive electrical splitting)

?



jFEX module partitioning baseline



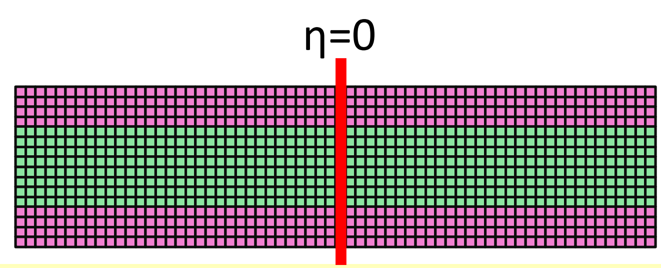
Algorithm requiring environment of 0.9×0.9 around each tower to be processed → +/- 4 neighbours in eta and phi

Enough?

Processor modules

- Process strips along eta
- Receive fully duplicated data in phi from DPS
- 8 modules covering half of eta, phi quadrant each
- Most cells duplicated in a regular way at both source and sink
- "Irregular" duplication at $\boldsymbol{\eta=0}$
- (Current) detector cabling has lower latency around $\eta=0$ due to cable **path**
- Benefit from this latency reserve and use for additional optical fan-out (re-transmission)

Rely on that ?



jFEX baseline partitioning

Processor modules

- Half eta × phi quadrant per module ~3.2 × 1.6
- Total of ~32 × 16 bins × 2 × 2 (upstream duplication, e/h)
- ~256 incoming fibres @ 6.4Gb/s baseline
- ~22 × 12-way "MicroPOD" high density receivers
- Four 72-way fibre connectors ("MPO/MTP")
- Note: further connectivity required for duplication at η=0: either front (re-transmission) or back (separate active optical fanout station)

288 fibres enough ?

Processor FPGAs

- Core of up to 1.2×0.8 (η×φ), plus environment 0.8×0.8
- 20 × 16 bins → 80 high speed links
- 6 large FPGAs per module (3 × 2)
- Eta coverage nominally (3 × 1.2)+0.4 = 4

FCAL granularity overlap region

Baseline and options

Phi quadrant baseline driven by need for a rock solid design:

- Line rate limited to 6.4 Gb/s only
- Avoid extremely dense module
- Sufficient $\eta \times \varphi$ coverage at module level to allow for increase of environment » baseline of 0.9×0.9
- Orientation along eta might better match Tilecal in phase 2 ?

Backup

- In case of problems with $\eta=0$ - duplication go for full eta, phi octant scheme

Options

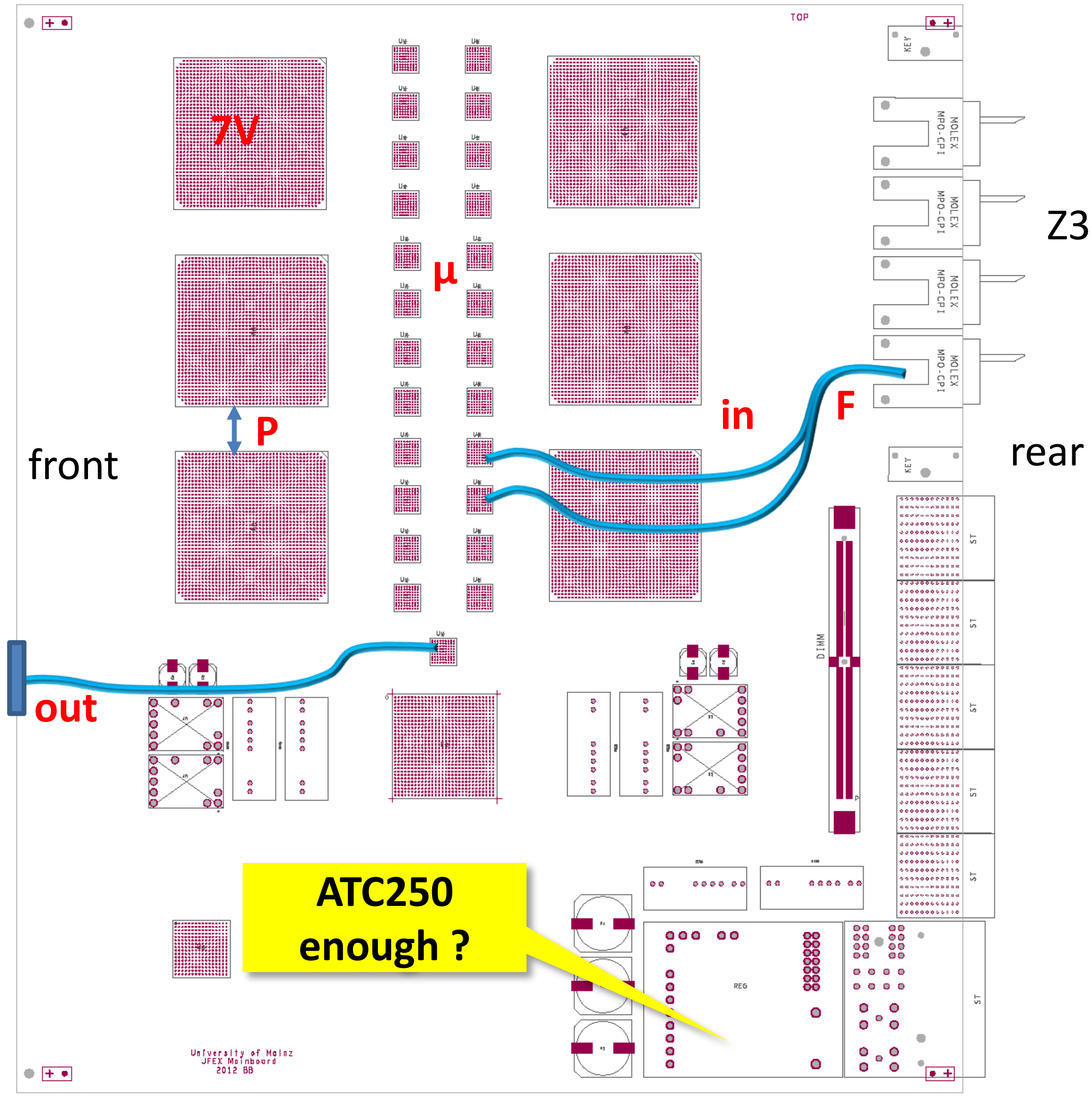
- Continue work on higher link speed
- Continue work on duplication schemes
- Consider slightly larger/more FPGAs

→ Increase environment beyond 0.9×0.9

Do we need the details for higher rate option now ?

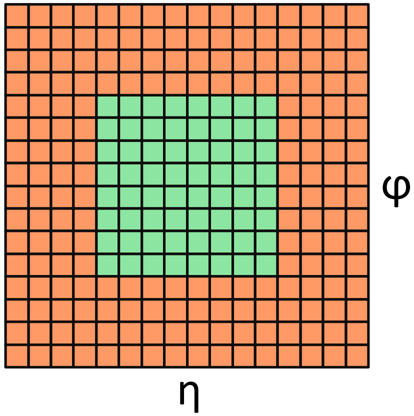
How to fit on a module ?

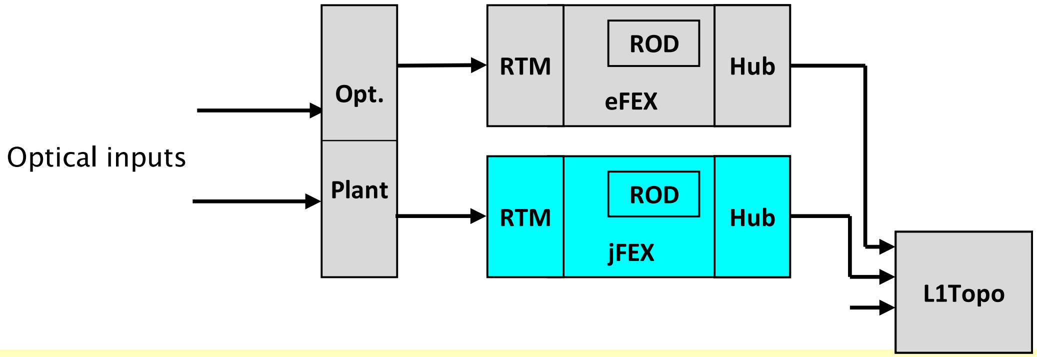
- AdvancedTCA format
- 6 processors XC**7V**X690T
- 4 microPOD sockets each **µ**
- Opto connectors in Zone 3
- Fibre bundles from rear **F**
- fan-out via "far end PMA loopback" **P**
- Output to front panel
- Readout via backplane
- Maximise module payload: small-footprint ATCA power brick, tiny IPMC mini-DIMM



jFEX system (baseline)

- Need to handle both fine granularity and large jet environment (minimum 0.9×0.9)
- **Require** high density / high bandwidth to keep input replication factor at acceptable level (~3/4 of all FPGA inputs are duplicates)
- Fit in 8 modules (difficult to spread out)
- Single ATCA shelf
- Sharing infrastructure with eFEX
  - Handling / splitting of fibre bundles
  - Some communalities in ROD design
  - Hub design
  - RTM





Some remarks on baseline

- FPGA density reduced wrt previously presented phi ring scheme
- Still very dense design, high power dissipation
- FPGA count might grow again for larger jet option
- Real-time circuitry has absolute precedence over non-real-time components
  - Happy to see RODs move off module
  - Any TTC solution must be minimum footprint
  - Power and ATCA control minimized in floor-plan shown
- Baseline design would probably allow for low-latency direct output into L1Topo (48 fibres total)
- Output consolidation possible at some latency penalty
- jFEX relies on “MicroPOD” high-density optical devices
- Aim at higher line rates (currently FPGAs support 13 Gb/s, Mini/MicroPOD 14 Gb/s)
- Would allow for even finer granularity or larger jets or smaller FPGA devices

Issues / questions

Many issues that came up recently are common to both FEXes. Concentrate on jFEX specific issues here.

- What is our default jet algorithm ?
- The 6.4 Gb/s baseline is not quite what we want to build
  - At module level there is enough environment for large jets
  - With FPGAs available on the market that cannot be exploited (excessive on-board duplication)
- Is 6.4 Gb/s / 0.9 x 0.9 ok, assuming we will not actually build the baseline but rather increase input rates 2-fold ?
- jFEX preferring 12.8G 8b/10b over eFEX preferred 64/66!
- What real-time output bandwidth is required ?
  - Board level merging (phase 1) ?
  - Phase 2 output to L1Calo ?

Power…

- Large FPGAs easily dissipate 30W each.
- Standard ATCA power modules rated 250W (ATC250)
- Might have to find non-standard solution
- High power crates today provide 450W/slot (power, cooling)
- Larger jet environment will inevitably increase power consumption
- microPODs are tiny and have a considerable local dissipation, likely to require custom mechanics/heat sinks

questions

- Can we agree on common components
  - IPbus
  - LAPP DIMM
  - Configuration (SysACE?)
  - Some TTC
    - Small footprint
    - Ideally firmware only
    - Just one clean clock plus an MGT link for data ?
- The ROD is understood to be conceptually similar between FEXes.
- Can we agree on something that doesn't require large local buffers on the processors ?