



L1Calo Phase 2 Upgrade

Murrough Landon
26 March 2009

- Baseline Concept
- Geometries
- Granularity Options
- Architecture and Fanout
- Calo Issues
- Summary

Introduction to L1Calo Phase 2 Work

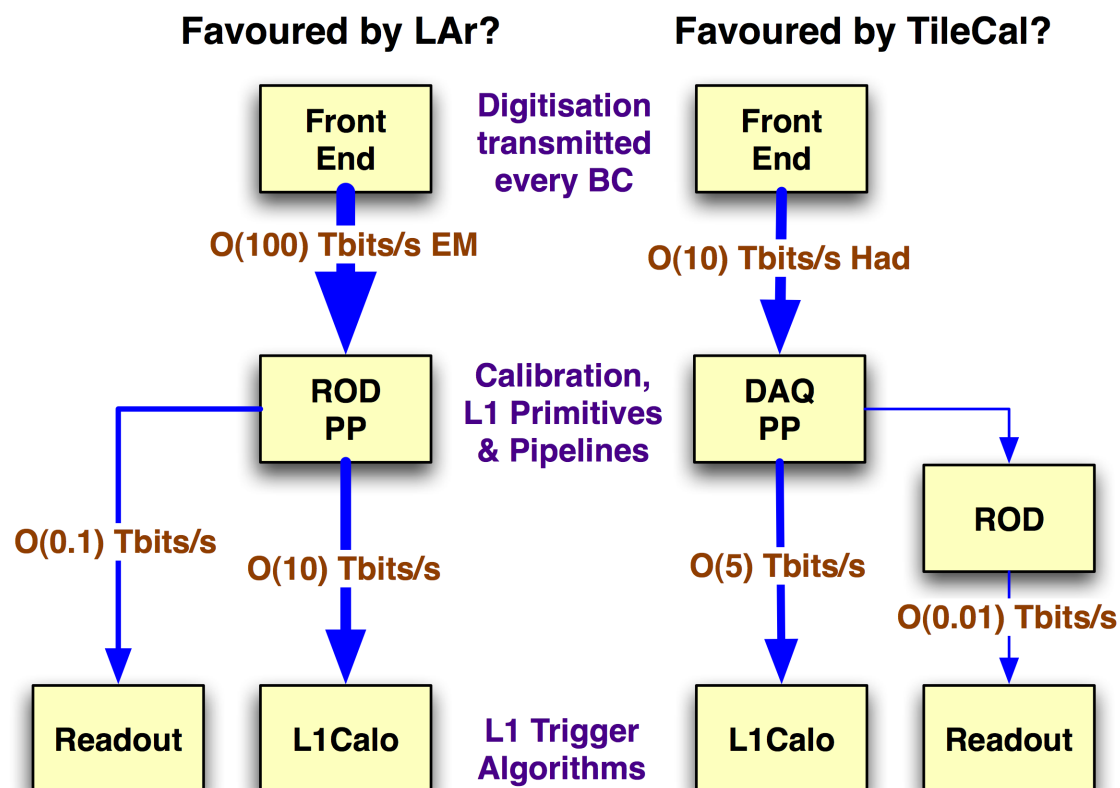
- Work done so far (a little)
 - Survey of ATCA, links and other technology
 - Survey of HLT algorithms we might steal
 - Thoughts on general L1Calo phase 2 architecture
 - Issues relating to mappings from FE and RODs
- Work not done yet (lots more!)
 - Simulation, simulation, simulation
 - More simulation - especially of pile up
 - Detailed thinking about
 - Granularities
 - Algorithms
 - Architecture
 - Bandwidths
 - etc

Challenges of Phase 2 Upgrade

- Huge increase in pileup
 - But no increase in basic detector granularity
- L1 rate similar to the present one
 - But prefer not to increase thresholds for physics objects
- Need better L1 algorithms
 - Borrow ideas from present L2?
- Use finer granularity in eta, phi and depth at L1
 - New trigger tower (L1 primitive?) more than an Et sum
 - Lateral and/or depth profile and position information
 - Quality bits?
- Not yet clear what granularity we need
 - For what efficiency, fake rate, threshold sharpness
- Nor where or how to use it

Baseline Phase 2 Concept

- Strong preference to digitise and transmit all cells every BC from the front end to off-detector pipelines
- ROD/preprocessor:
 - generates Level 1 primitives (towers++) from calibrated Et for the correct BC
 - sends them to a separate L1Calo trigger processor
 - different LAr/Tile views of the ROD?
 - and still changing



NB bandwidths are **very** approximate!

Implications of On-Detector Digitisation

- Allows much more sophistication in forming “towers”
 - New digital “L1 Primitive” could be a bit field with
 - Et (to greater precision than before if required)
 - Depth and lateral shower profile information, quality flags?
 - Better handling across boundaries?
 - Finer granularity (in EM layer)
 - Different granularities possible in EM and Hadronic layers
- Single calibration for trigger and main readout
- But brings trigger and readout closer together
 - Present architecture allows each branch complete freedom to optimise the organisation of their own system
 - Phase 2 upgrade will impose trigger constraints on layout of RODs and mapping of FE to ROD links
 - Also no (completely) independent readout path

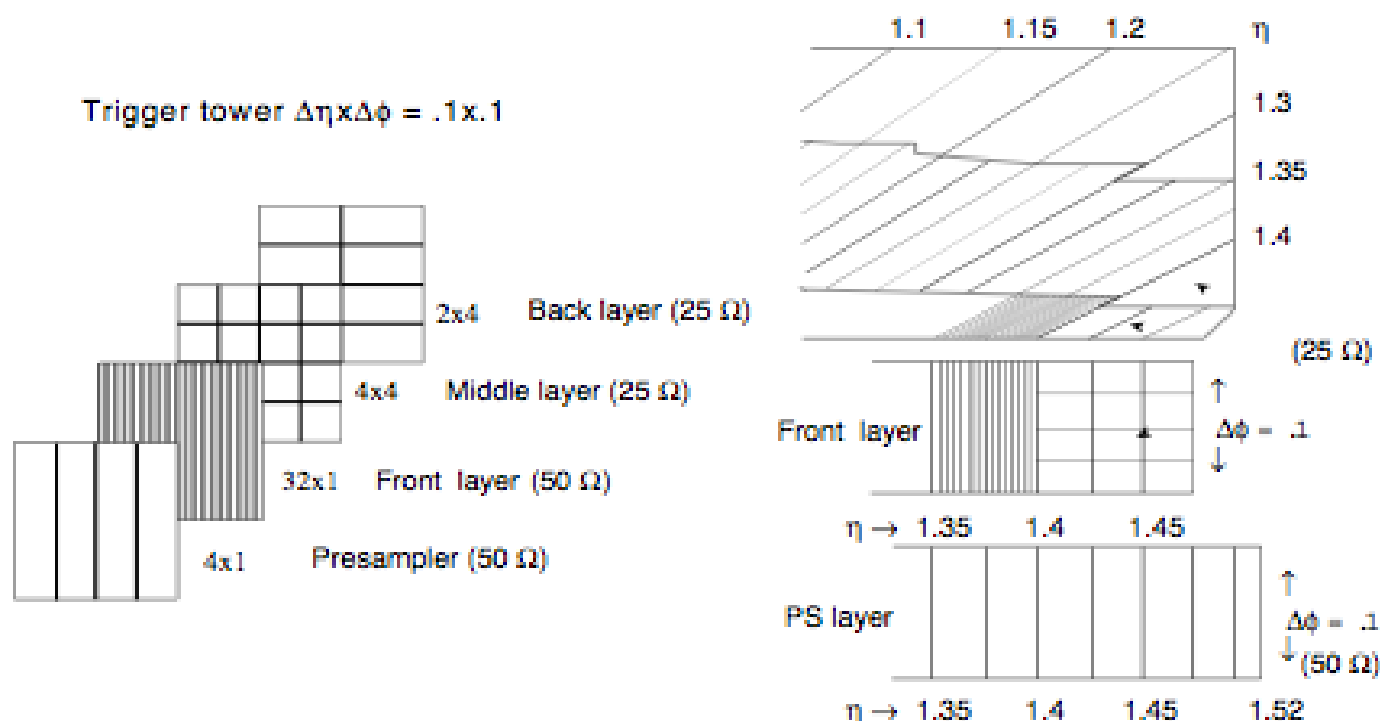
Latency

- Main upgrade scenarios:
 - No L1 track trigger: small latency increase possible to $3\mu\text{s}$ (constraint from muons?)
 - With L1 track trigger: need fast L0 seed from calo+muon (latency same as now, ie $2\mu\text{s}$?)
 - Independent L1 track trigger (no L0 requirement): greatly increased latency, up to $6\mu\text{s}$
- Until any decision, assume that latency is still critical
- No unnecessary deserialisation/reserialisation
 - Significant latency penalty (6 BCs?) at each such step
- Unavoidable: FE→ROD, ROD→L1, L1→Merger/CTP(?)
- Avoidable: everything else!
 - Eg between RODs, between L1 modules

EM Barrel Geometry

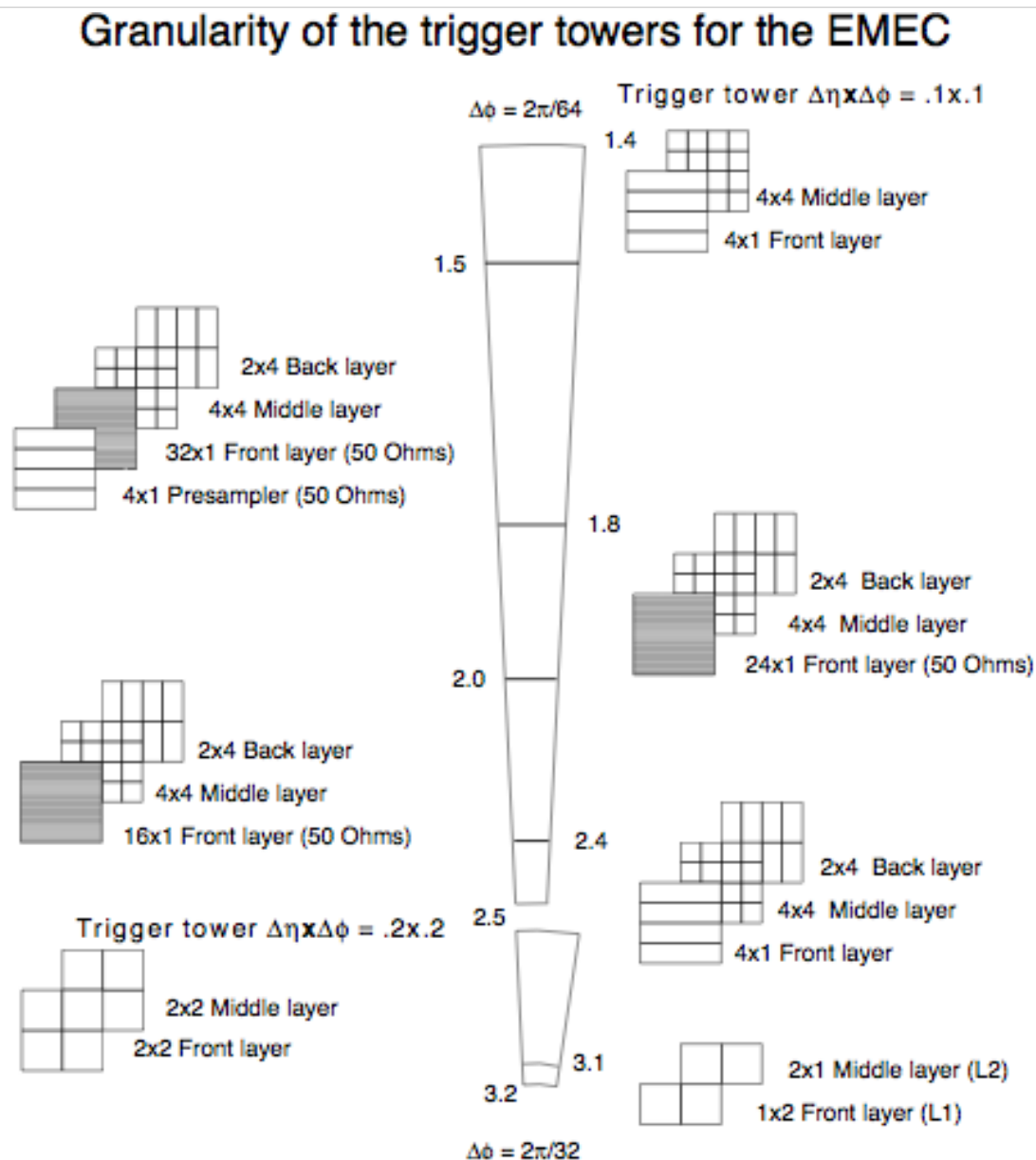
- Each layer has a different geometry
 - Uniform in eta, except for barrel/endcap transition region
- Middle and back compatible with 0.05×0.05 minitowers
- But front (strips) and presampler (PS) cover 0.1 in phi

Granularity of the trigger towers for the EMB



EM Endcap Geometries

- Seven different layouts between $\eta=1.4$ and $\eta=3.2$
- Many different ways cells are grouped into Front End Boards (FEBs)
- NB two granularities in the EM barrel
- One in the FCAL
- (Plus similar in the hadronic layer)



Granularity Options (1)

- **Minitowers * depth samplings**
 - Send 0.05×0.05 EM towers (0.025×0.05 PS, Front layers?)
 - Still 0.1×0.1 in hadronic layer (detector limit)
 - All depth samplings separate
 - Less need to organise cells in RODs
 - LAr data “reduction”: 60 cells $\rightarrow 4+8+4+2 = 18$ minitowers
 - Only factor 3 bandwidth reduction to L1Calo
 - But must then multiply by required phi fanout, up to factor 2?
 - 150 Tbits/s to LAr RODs: something like 50 to 100 Tbits/s to L1Calo
 - Plus 10% for hadronic layer
 - L1Calo expands to 8 crates in phi octant layout?
 - $O(100)$ modules with $O(0.5 \text{ to } 1)$ Tbit/s per module
- **Alternative: minitowers summed in depth**
 - Less bandwidth, but needs more cell organisation in RODs
 - How to match PS & Front layers to middle and back layers?

Granularity Options (2)

- Semitowers?

- Intermediate granularity: 0.05 in η * 0.1 in ϕ
- Add detailed lateral and depth profile information
- Shower position within the semitower
- Requires cells to be organised into towers in ROD FPGAs
- Modest increase in present bandwidth to L1Calo
 - 25 bits/semitower \Rightarrow total $O(10)$ Tbits/s

Granularity Options (3)

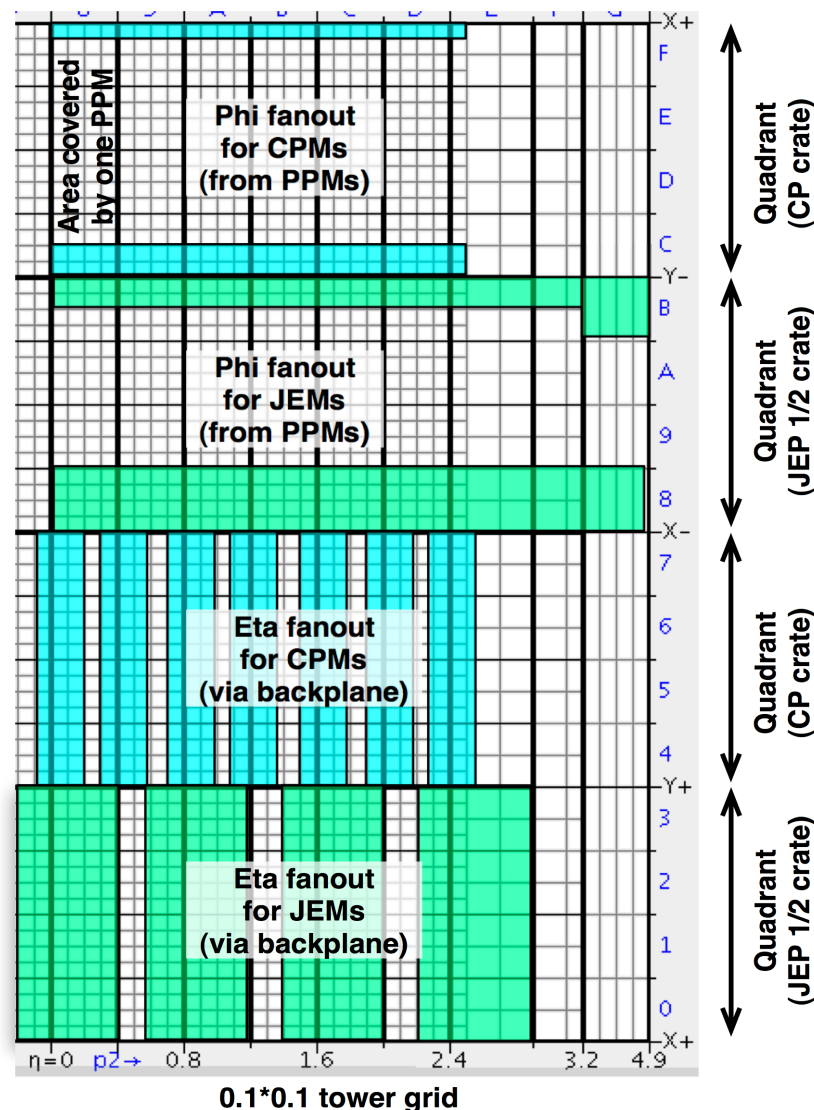
- Supertowers, miniL1Calo

- EM shower is well contained in existing 0.1×0.1 tower
- Future LAr ROD FPGA might cover 0.2×0.2 with full depth?
 - 40 GBT links equivalent to two whole FEBs or 256 cells
 - Would need a lot of organisation of links into RODs
- Run mini sliding window at full granularity in each ROD FPGA
 - L2 quality if shower is contained within one ROD FPGA
 - Option increasingly attractive as FPGAs & links get bigger & faster
- Send found electrons or half electrons to L1Calo (at 0.1×0.1)
 - Another sliding window algorithm in L1Calo to fix up boundaries
- Total bandwidth to L1Calo maybe 5-10 Tbits/s?
- Might squeeze low granularity L1Calo into a single crate
 - No phi fanout required for single crate L1Calo
- Issue: part of the L1 algorithm moves into LAr ROD domain
 - How to collaborate on development?

Present L1Calo Architecture

- Separate EM/Tau and Jet/Energy processors
- Sliding window algorithms
 - Requirement for environment
- Phi quadrant layout
 - $O(30\%)$ fanout at source (PPM)
 - $O(75\%)$ fanout at CPMs/JEMs
 - Strong requirement on eta, phi shape covered by all modules
 - NB orthogonal to detector layout
- Many remapping stages
 - Receivers (20 mapping variants), patch panels, PPMs

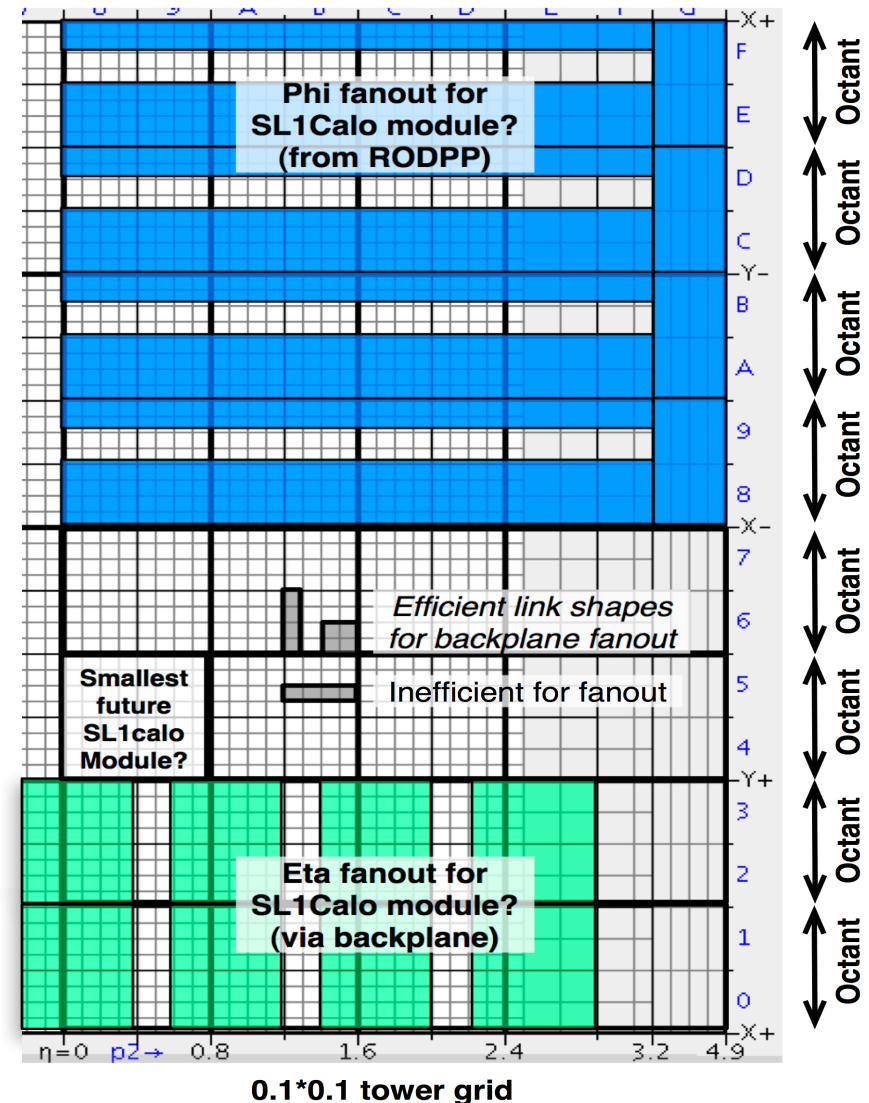
Phi quadrant architecture and fanout



Possible Phase 2 Architectures (1)

- Single processor module?
 - For all objects: EM, tau, jet
- Still use sliding windows
 - Unless there is a better idea?
- Fewer remapping stages?
 - May want fibre ribbon PPs?
- Consider phi octant layout?
 - Similar fanout in L1 modules
 - Unless modules wider in eta?
 - $O(75\%)$ fanout from RODs
 - Fewer restrictions on eta, phi shape covered by RODs
 - But still need regularity

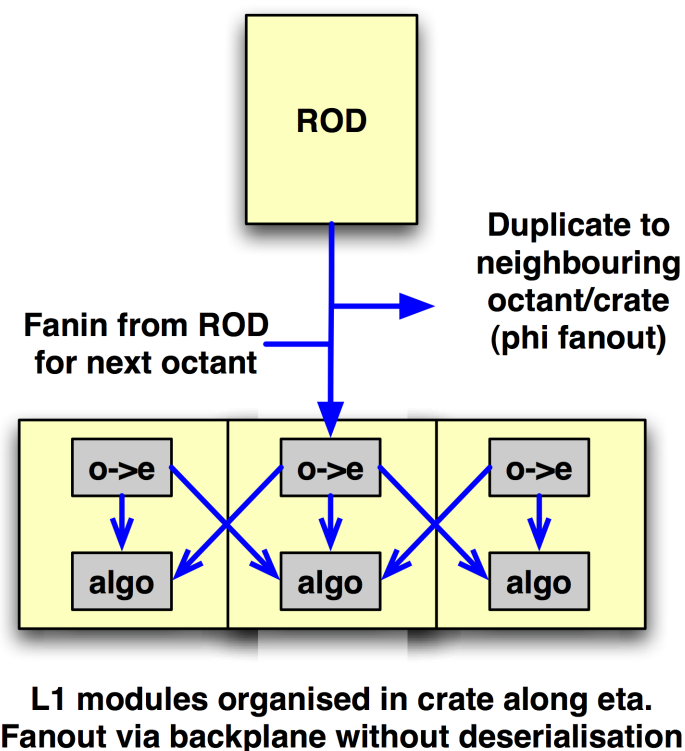
Phi octant architecture and fanout



Possible Phase 2 Architectures (2)

- Links from RODs duplicated to neighbouring octants
 - Either on ROD or intermediate fanout step?
- Links to neighbouring L1 modules duplicated via crate backplane without reserialisation
 - Most efficient if links cover regular, squarish $\eta^*\phi$ areas

Fanout of data from RODs to L1



Downstream from L1Calo...

- Topology and the CTP?
 - Phase 1 upgrade proposes additional L1Calo (plus muon) topological processor passing extra bits to the current CTP
- What would be appropriate for phase 2?
 - Keep separate topological layer combining calo, muon and track trigger
 - Or combine topology with new CTP?
 - Another case of phase 2 boundaries possibly being different from now

Granularity: Links

- ROD→L1 links

- Sliding window algorithms require lots of fanout
- For phi octant layout, this is most efficient if L1 link contains contains more “towers” in phi (2^n) but is narrower in eta
 - Not how the detector is organised (especially TileCal)
- Small number of towers per link easier to handle
 - But greater number of serial streams to fanout
- 1 Gbit/s is 25 bits at 40 MHz
 - Roughly one EM tower with Et and profile bits?
 - 6 Gbit/s would easily cover 4 towers (or mini-towers)
 - Likely possible to cover eta,phi space with 0.2×0.2 links

- FE→ROD links

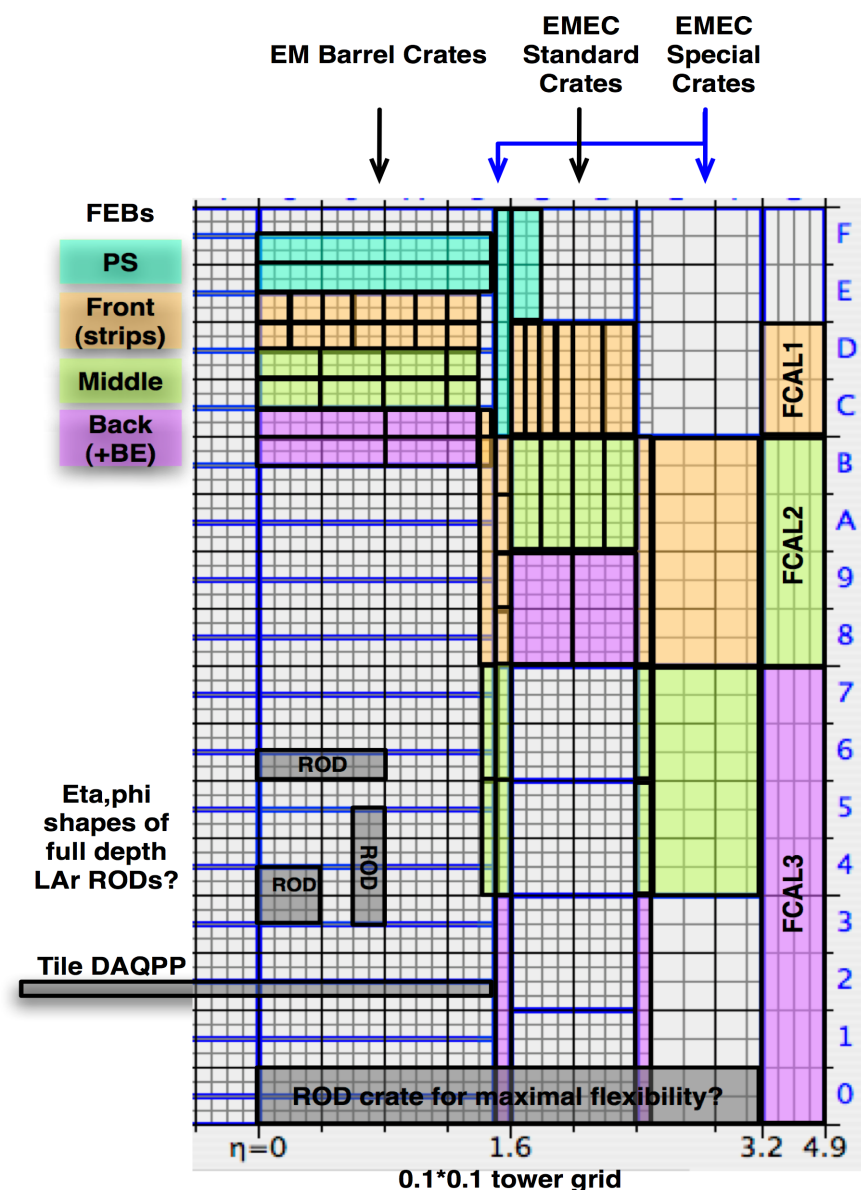
- Group together cells onto links by towers if possible
 - Follow existing tower builder or Tile adder layouts?
 - Projective geometry in TileCal, not division by z?

Calorimeter Boundaries

- Worst case (by far) is EM barrel/endcap transition
 - Anything we can possibly do will probably be needed
 - Sum cells across EMB/EMEC before making L1 primitives?
 - Add in crack scintillators? (Currently read out via Tile EB)
 - Upgrade being considered in that region
- Cant do anything about crack at $\eta=0$
- Next worst is Tile LB/EB transition
 - Currently cells are deliberately misorganised to adjacent η bins to avoid analogue summing across the boundary
 - Upgrade Tile ROD could handle it properly
 - Add in the gap/crack scintillators?
- Least worst is Tile EB/HEC transition
 - Again, currently misorganised - could do better digitally

LAr Front End Board Layout

- Diagram shows eta,phi sizes of FEBs in different regions
 - Sketched on 0.1*0.1 tower grid
- Barrel, endcap & FCAL have many different geometries between (and within them)
- Transition regions span boundaries in both eta & phi
- Bring all layers to one ROD
 - requires splitting some FEBs between two or four RODs
 - Is this a problem (in principle)?



Summary

- (Much) more simulation and thought required to:
 - Identify optimal and workable algorithms
 - What can be implemented in firmware and in whose FPGA?
 - Derive viable L1Calo architecture in more detail
 - Whats in a tower? What extra information (apart from Et)?
 - What granularity do we need?
 - What bandwidth can we (and the RODs) handle?
 - Discard any unnecessary “worst case” scenarios
- Need to discuss implications of FE and ROD layouts
 - What is desirable/acceptable/undesirable/unacceptable to the LAr and Tile groups?
 - Some options have significant impact on ROD organisation
 - Issues of boundaries of responsibility?

Backup Slides

Channel and Link Organisation (1)

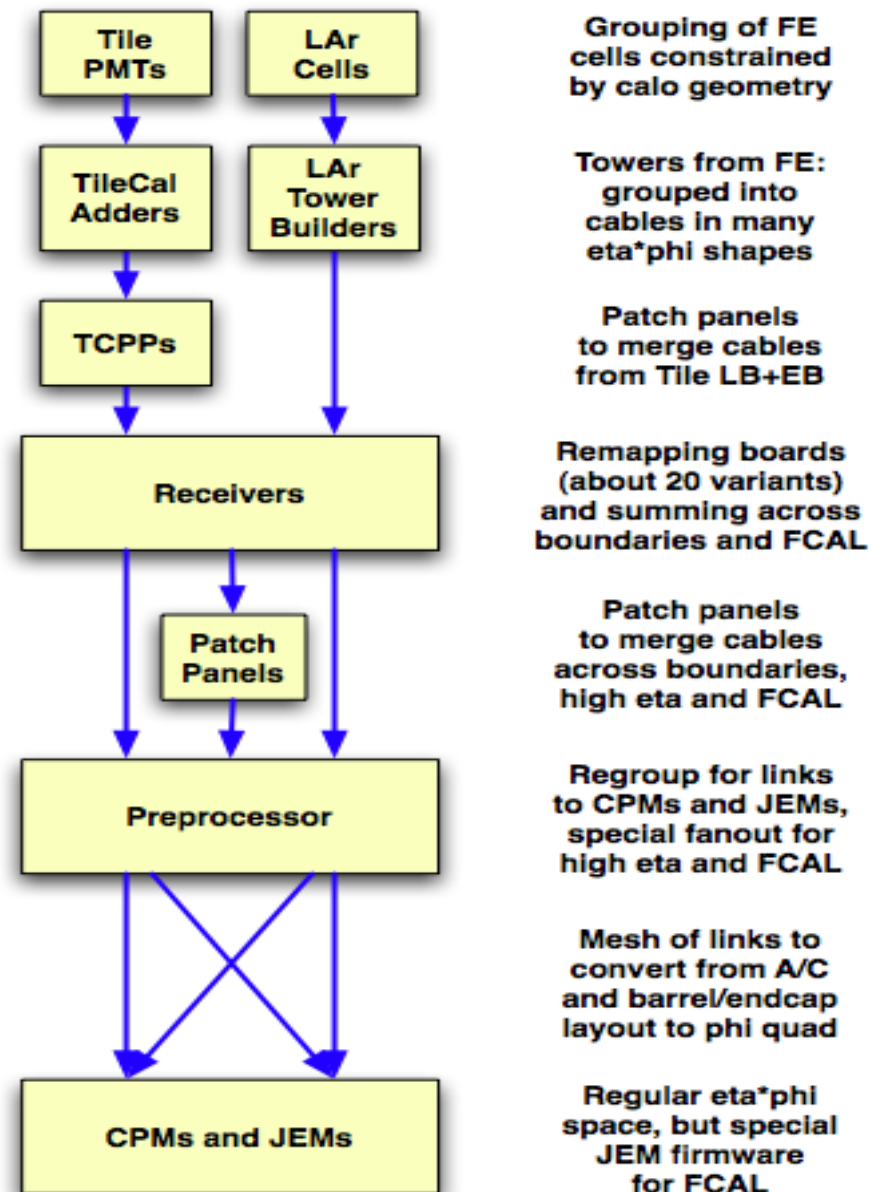
- Lessons from existing L1Calo
 - Worry about the difficult areas early in the design process
 - It only gets worse later (and don't forget about the FCAL!)
 - Do as much as possible at the first stage in the chain
 - Irreducible constraints from calo geometry will hit later
- Link organisation
 - Data processed together needs to be brought the same chip!
 - Best to bring links directly to the right chip
 - If not, at least to the same module
 - Or from a module in the same crate (fast parallel transfer)
 - Avoid need for high latency serial transfers
 - Either between modules in the same crate or different crates
- L1 constraints affect the layout of RODs and FE links

Channel and Link Organisation (2)

- Little guidance yet from simulation
 - Assume the worst cases (from FE and ROD viewpoint) and look at the implications
- Assume L1 primitives formed from all depth samplings
 - For EM and hadronic layers separately (at this point)
 - Sending separate depth samplings to L1 is easier for organising links
- Assume L1 primitives must cross calo boundaries
 - Process Barrel/Endcap cells together in same chip
 - Assume crack scintillators for EMB/EMEC boundary
 - Implies LAr and Tile RODs sharing crates
- Worst possibility? Full EM+hadronic depth summing
 - Inevitable latency penalty, high degree of convergence between EM and hadronic RODs (and shared crates)

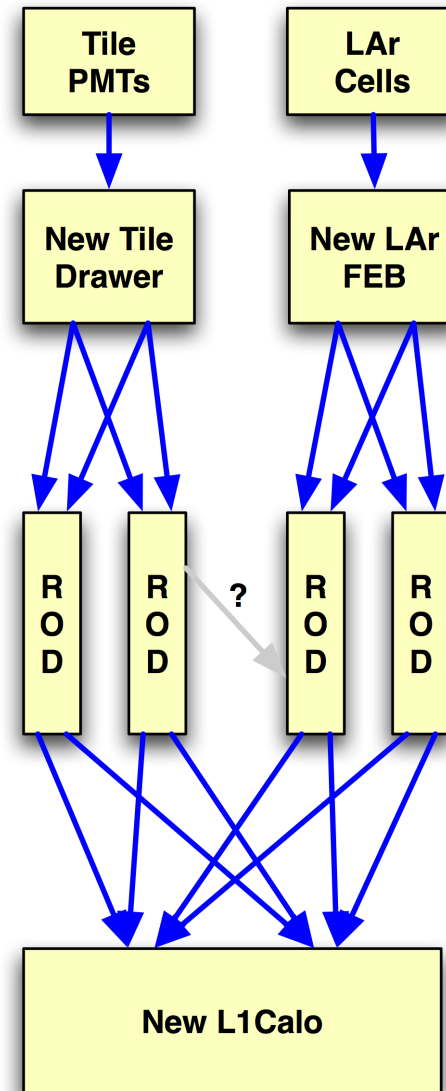
Present Mapping Stages

- Many stages
- Lots of patch panels
 - Humble TCPP is ~2Gbit/s remapping device with ~0 latency and power!
- Easy areas regularised in one step at receivers
- Tricky areas needed many successive steps
- Never really managed it with the FCAL



Possible Upgrade Mapping Stages?

- Fewer steps available?
 - Unless we add latency with an additional reorganisation
- Start with FE boards
 - May need several different channel to link mappings?
- Remap FE to ROD links
 - Signals in depth and across boundaries to same place
- Minimal (low latency) transfers in ROD crates?
- Regroup (and duplicate) ROD to L1Calo links



Grouping of FE cells constrained by calo geometry

Add remapping board on LAr FEB to regroup cells on links?

Regroup inputs to RODs?
Split/merge fibre ribbons?

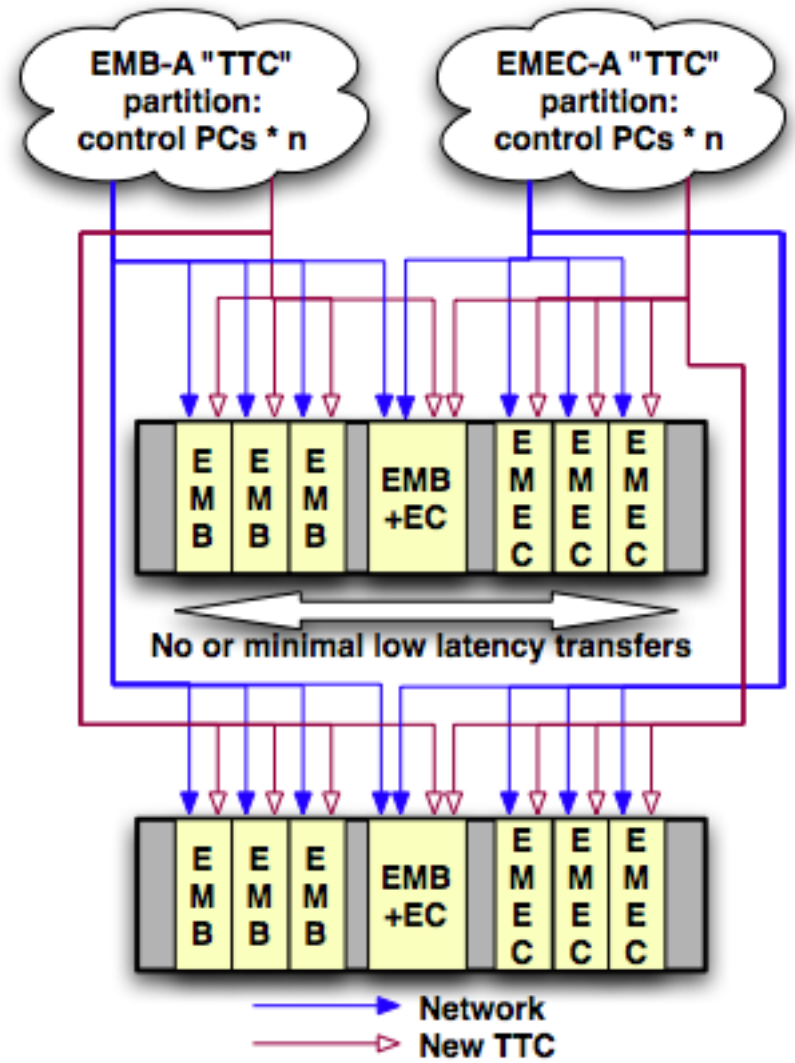
Little data transfer between RODs?
Regroup towers on links to L1Calo

Mesh of links to L1Calo (duplicate data for phi fanout)

Any remaining mapping issues resolved by firmware

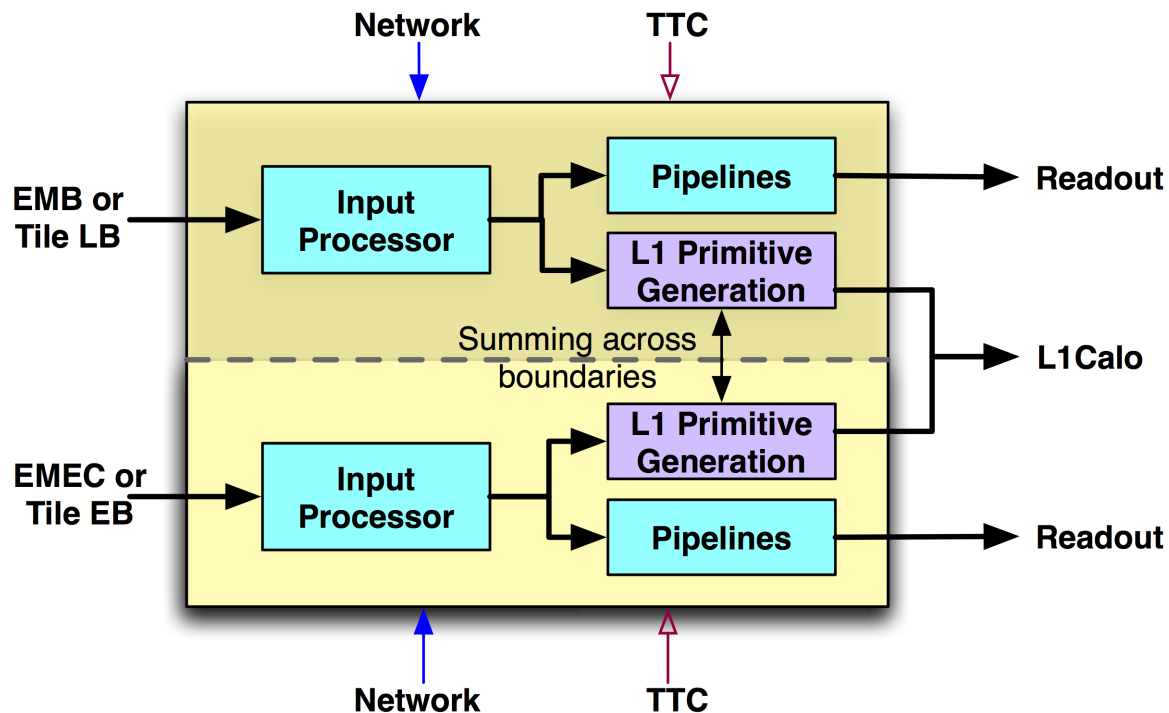
ATCA-based ROD Crate?

- New crate: new architecture for control/configuration?
- No crate CPU or control bus
- Separate network and TTC++ connection to each ROD
- Flexible and scalable set of PCs to configure N RODs/PC
- Different TTC partitions can (but need not) share crates
- Can run separate standalone partitions for calibration
- Many configurations possible



ROD Issues (1)

- Two TTC partitions in one ROD at boundaries
- Tile baseline (DAQPP) has all four partitions:
 - Q1 (curiosity): how to run partitions independently?
 - Q2 (request!): can LAr do the same?



ROD Issues (2)

- Granularity of FPGAs:
 - Probably want little or no transfer between chips?
 - If so, N links input to one chip define the η, ϕ space of n links output to L1
 - Can all depth layers be really brought together?
 - Depends on cells/link, links/chip, chips/ROD
- η, ϕ space covered by whole RODs:
 - Any need to transfer data between RODs requires congruent η, ϕ spaces covered by those RODs
 - Eg for Tile RODs to send crack scintillators to EMB/EMEC RODs
- Sparseness at higher η
 - Changing ratio of input cells per “tower” with η
 - Underutilised RODs or reconfiguration of input:output links