Trigger Architecture Studies

- Take "architecture" to encompass:
  - Definition of trigger objects
  - Object ID and data reduction algorithms
  - Data flow {into; out of; within} system
- Factorisation of problem
  - Production of trigger primitives (see other talks in this session)
  - Trigger architecture and algorithms
  - Optimisation of data reduction and cuts (future work)
  - Probably not a good factorisation... large interference
- Wider problem than just tracking trigger
  - Probably, necessarily so
  - At some stage in the L1, must combine objects from all subsystems
  - Can take inspiration from current HLT – though constraints are different
- As usual, more questions than answers

Dave.Newbold@cern.ch
University of BRISTOL

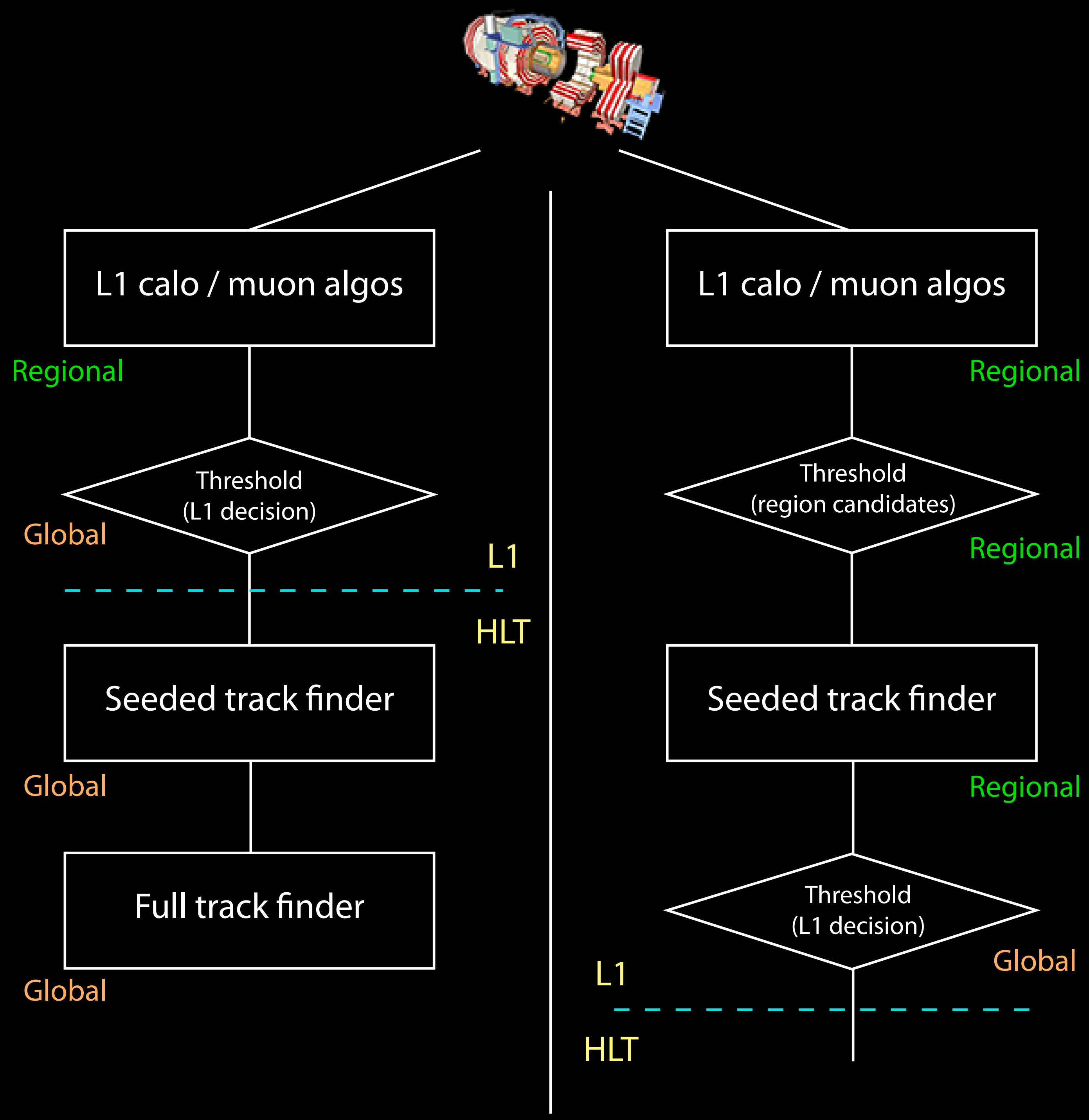
Trigger Strategy

- LHC trigger strategy
  - Trigger objects: leptons, photons, jets, global energy sums
    - p_t thresholds used for control of rate
    - Multi-object triggers used wherever possible
- SLHC trigger strategy
  - Current strategy should be efficient for rare heavy states
  - Physics constraints (W mass, heavy quark b/g spectrum) unchanged from LHC?
  - Exclusive multi-object triggers will be important
  - Need to find robust object ID algorithms against increased background
  - Rate /efficiency targets are the same as LHC (better would be nice, e.g. for tau)
- Tracking information
  - Use to back up the existing lepton ID algorithms
  - Can be used for veto purposes (local jet activity), and for object track match
  - Possibility of multi-object vertex match is very interesting
  - Tracking input needs to be 'good enough' for rate control, not perfect

Constraints / Drivers

- Trigger upgrade: key technical constraints
  - Additional material in tracker volume
  - On-detector processing in tracker (power, inter-layer communication)
  - Bandwidth from tracker to off-detector systems
  - IO density in off-detector trigger logic
    - Logic density should not be a major constraint
  - Latency
- First thinking on architecture
  - Need simple, robust approach for trigger primitives with p_t-cut
    - Stacked tracking one promising approach Complexity? Robustness? Power requirements?
  - Bandwidth reduction is key
    - "Fixed object count" approach; multiple steps of on-detector data reduction
    - Keep in mind that we are looking for isolated objects
  - Drive track-finding from calo / muon objects
    - Cuts down complexity of tracking algorithms + inter-system IO
  - These ideas are now quite old – alternatives exist

Exemplar Concept



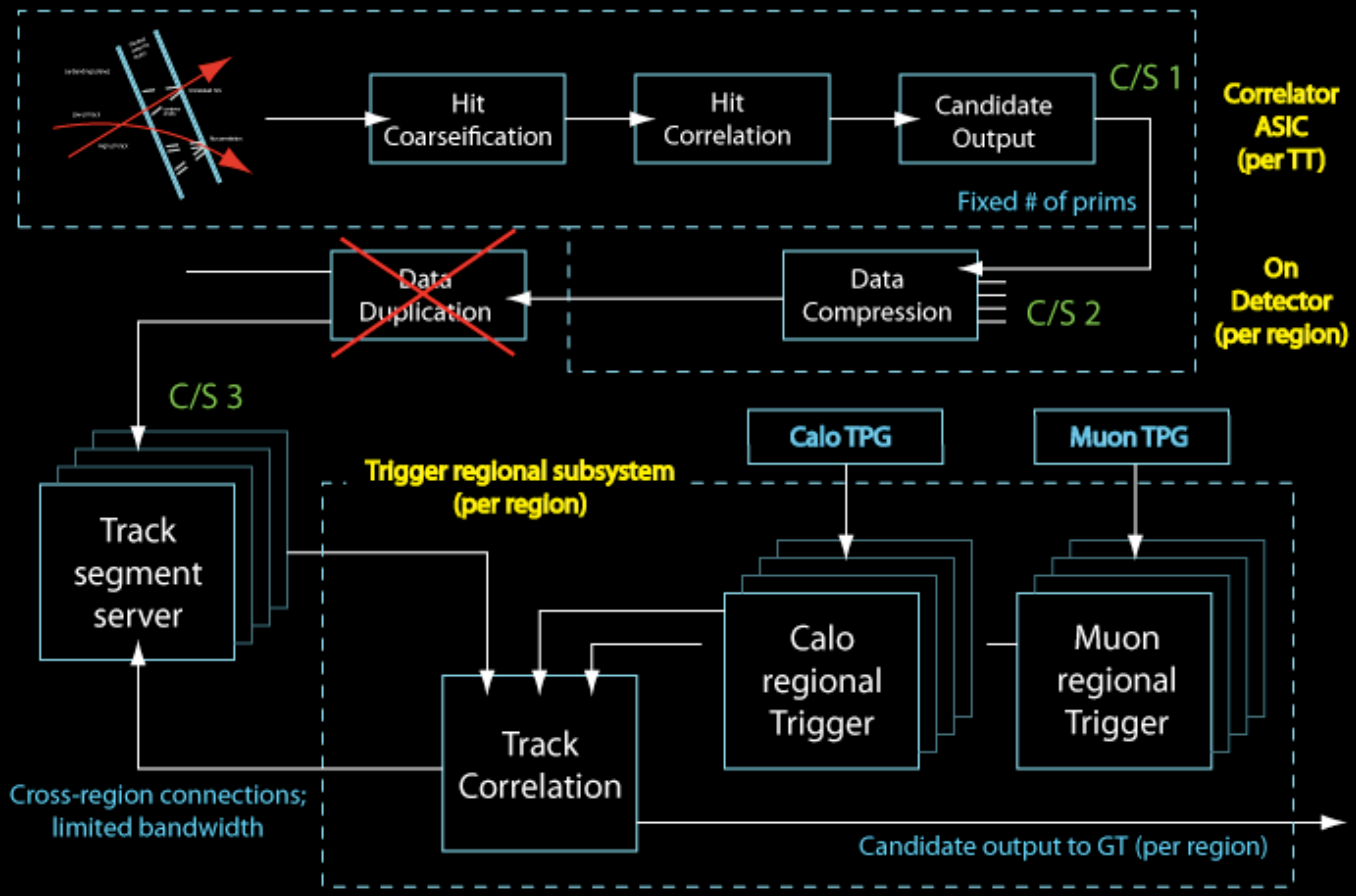
- Region-of-interest
  - Local tracking also used in HLT
  - Regional approach at L1 is similar
  - => regional segmentation in phi
    - As per existing trigger segmentation
- HLT has global calo info
  - So brem recovery possible, unlike L1
  - Effect on electron matching?
- Alignment / beam posn
  - Similar concerns at L1 as in HLT
  - We can only rely on mechanical alignment & crude beam posn
  - Robustness must be shown

Dave.Newbold@cern.ch
University of BRISTOL

Some Possible Shortcuts

- High p_t tracks only
  - Cuts down search region for track finding (but charge ID harder?)
    - e.g. 4GeV/c p_t track bends though only +/- 1 trigger region
- Isolated objects only
  - If multiple primitives are found in a region, simply flag this
  - We are looking for regions with {1, 2, 3, many} high p_t tracks
  - High track count could be used in a jet veto algorithm
- Seeded track finding
  - Do not require an exhaustive track search
  - For muon ID, perhaps even a single point is enough
  - For electron ID, difficulty could be sensitive to layer placement - trade-off
- Vertex matching?
  - Is this feasible? Clearly cannot separate 400 vertices
  - May cut down background rate enough to be worthwhile

Exemplar Concept: Dataflow

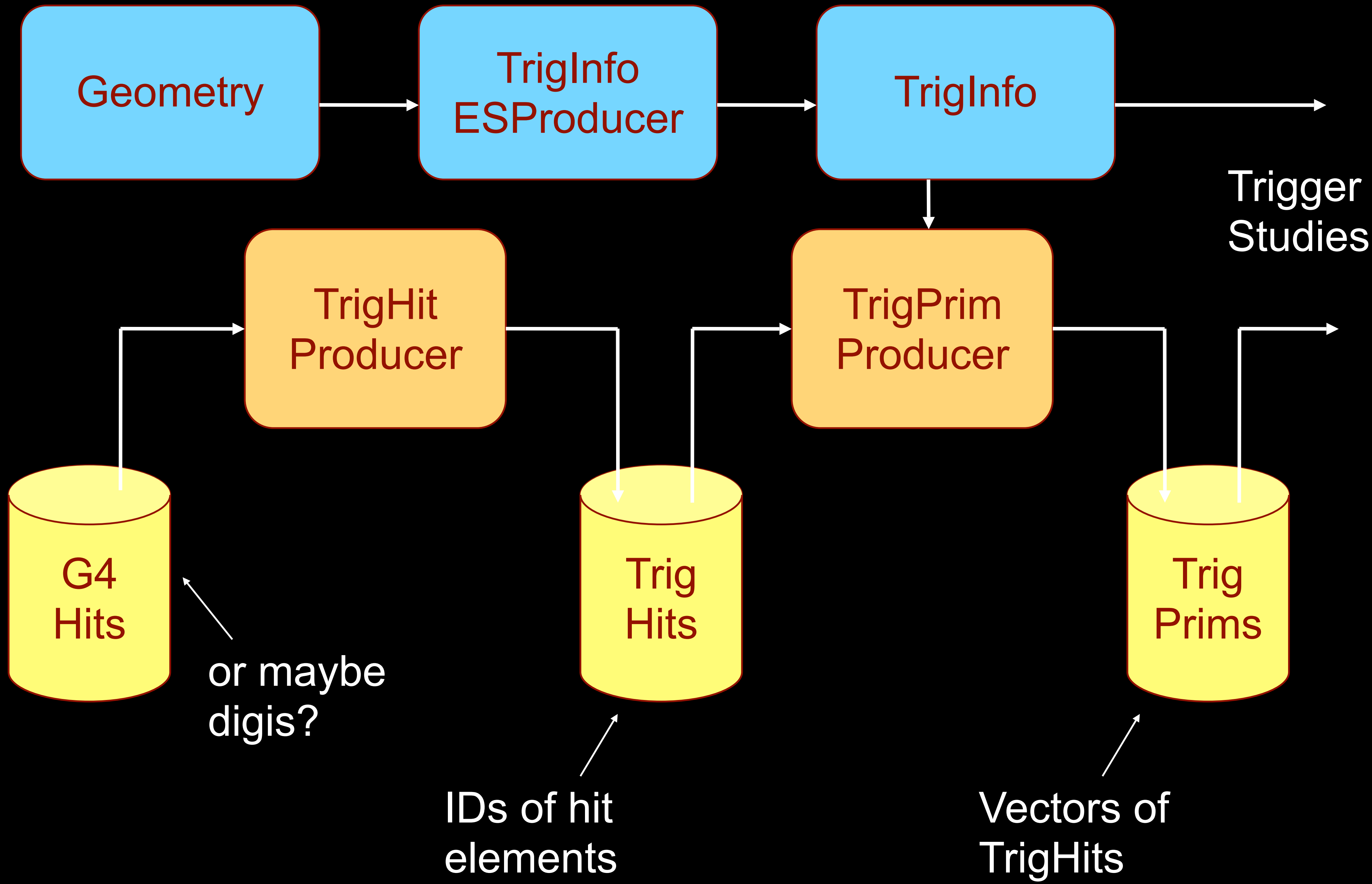


- What is the interplay with a Phase-1 upgrade?
  - e.g. remodularisation of the L1 hardware

Simulation Studies

- Do these (or similar) ideas work?
  - First architecture studies are under way, but progress has been slow
- Testing the basic ideas
  - Four-vector level simulation can answer many of the most basic questions
  - Allows a rapid feedback to tracker design
  - Can study gross changes – number of layers; number of objects; etc
  - Can estimate order-of-magnitude dataflows
- Introducing realism
  - When track primitives are available, can begin more realistic studies
  - Must find out where the points of uncertainty are for detailed simulation work
- Organisation
  - Requires input from *all* L1 subsystems (calo, muons, tracking)
  - Requires agreement on the basic approach fairly soon
  - Coherent trigger upgrade group now forming – more simulation effort needed

Example Code Structure



- Prototype track trigger code structure (Jim Brooke)
  - Concrete contributions now coming from several institutes

Track Trigger Simulation Roadmap

- Four-vector level studies
  - Remainder of 2008 – window of opportunity
- "Perfect" detector studies (early 2009)
  - Parameterised response for upgraded calo / muon trigger
  - Track primitives from tracker upgrade simulation (under devt)
- Introduce realistic constraints (2009-10)
  - Fixed number of objects; coarse resolutions; limited intercommunication
- Realistic simulation (2010?)
  - Fast or full simulation route?
- Bit-level emulation (2011?)
  - Allows firmware design / debugging
- Schedule clearly depends on available effort

Dave.Newbold@cern.ch
University of BRISTOL

Software Issues

- We expect to encounter difficulties
- Performance issues
  - Full simulation of 10^35 lumi crossings not yet demonstrated
  - Trigger simulation requires ~all subdetectors in simulation, and large samples
    - e.g. imagine trying to study detailed behaviour of Ht trigger at SLHC: ~ impossible today.
  - What role can fast simulation play here, and at which stages?
- Software issues
  - A ground-up trigger simulation code is a significant software project
    - The current software took ~years to construct and validate
  - What can we reuse, what needs to be redone?
    - Opportunity to use a common approach for all subsystems?
    - What are the interfaces to the (upgraded) subdetector simulations?
  - Does this link into the possible use of common hardware?
- Again, significant (expert) effort needed here
  - We should avoid divergent approaches, even for early studies

Summary

- SLHC L1 trigger studies are starting in earnest
  - Some first ideas on architecture have been identified
- “Architecture” has strong interplay with detector upgrades
  - In particular, the tracker
  - Must decouple the work to some extent to make progress
- Four-vector studies may tell us a lot in the short term
  - But we will quickly need to move beyond this
- Need to start considering the software framework
  - Building upon the expertise gained in the current system
- More people welcome and needed!