Trails and Tribulations of High Performance Computing

Mr. David B. Roden, Senior Consulting Manager

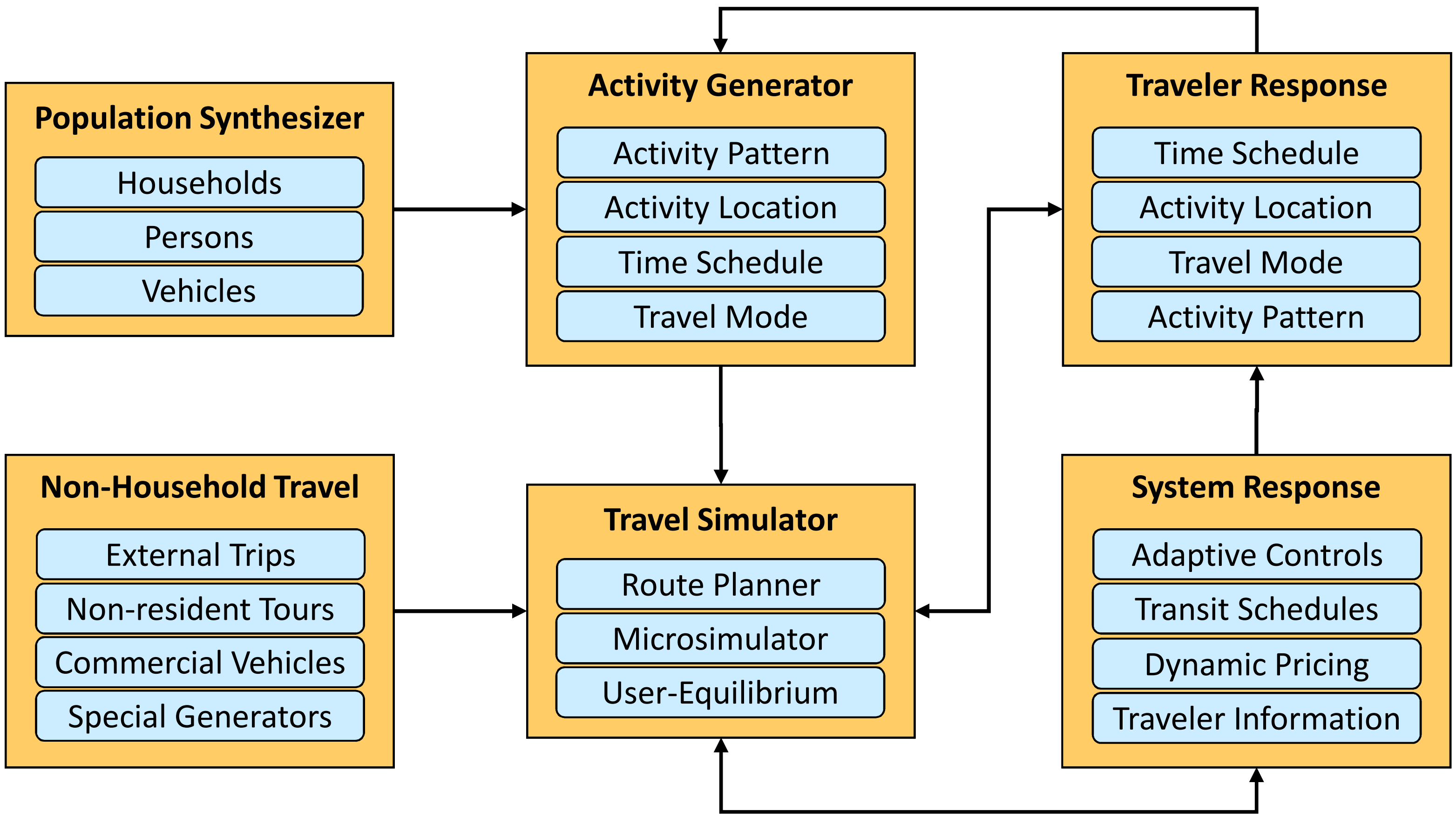
AECOM

April 29, 2012

Transportation Modeling

- Transportation modelers are charged with preparing credible forecasts of:
  - The impacts and benefits of proposed highway, transit, pedestrian, and bicycle improvements
  - Mobile source emissions for conformity analysis and for quantifying greenhouse gases (GHG), energy consumption, and particulates
  - The impacts of policy decisions and pricing or operational strategies designed to influence or manage travel demand, system performance, land-use development, special events, and economic or financial viability

Advanced Practice Modeling



Analysis Requirements

- Finer resolution of space and time dimensions
  - Network-based locations; 15 minutes or less
- Traveler decisions based on household activities
  - Coordinated person travel for one or more days
- The operations of specific streets and facilities
  - Time dependent networks with dynamic operations
- Regional simulation of individual vehicles and persons to evaluate system performance
  - Detailed forecasts of speeds, queues, flows, riders, etc.
  - By time of day, vehicle/user type, lane, train, etc.

The Computational Challenge

- The generally "acceptable" computer processing time for traditional TDF models is ~24 hours
  - Large regions must trade-off model detail and complexity against computer hardware/software costs and complexity OR do most regional modeling in-house or through a service center
- Advanced practice models are significantly more complex and computationally demanding
  - High performance computing is required for "feasible" processing times
    - Most MPOs won't accept run times over 48-60 hours

Case Studies

- Computational solutions and challenges for large traditional TDF models
  - MWCOG / WMATA
- Advanced demand models integrated with traditional network models
  - DRCOG / RTD
- Advanced demand models integrated with regional simulation models
  - SHRP2-C10 Jacksonville

MWCOG / WMATA – DC/VA/MD

- MWCOG version 2.3 model
  - Traditional TDF model using Cube software
  - Expanded zone structure to 3,722
  - 25 million person trips, 1.6 million transit trips
  - New mode choice models use 22 transit paths
  - Assigns 4 periods, 6 user classes, to 10^{-4} or better
  - 5 global speed feedback loops
- WMATA post processing model
  - Models each trip purpose by time of day
  - Park-&-ride capacity constraints

Model Run Times

- Tested several options to gage run times
  - Congestion level (year), assignment algorithm, and distributed processing through Cube Cluster
    - Most regional agencies limited to 4 core computers

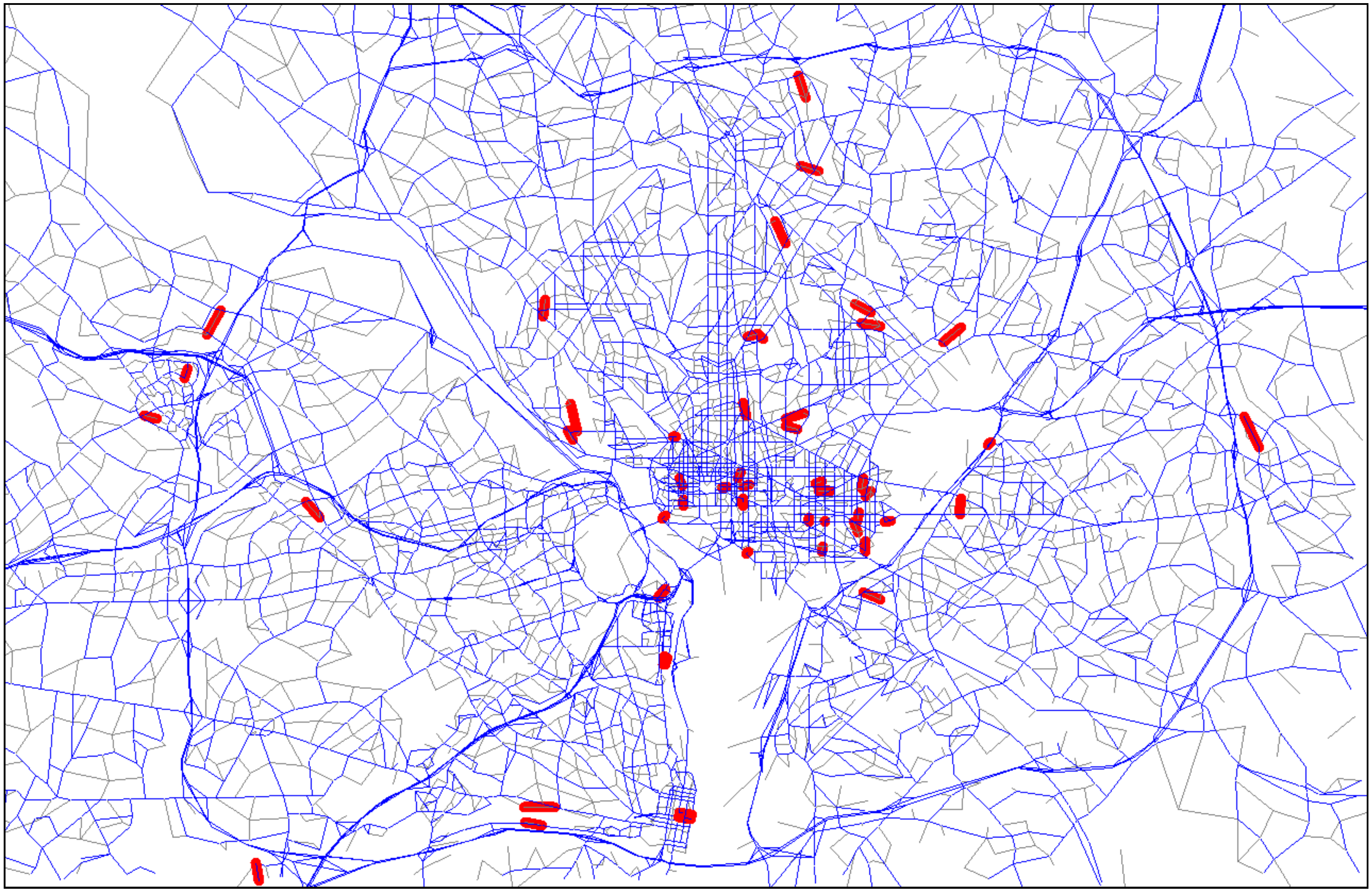
Year	Algorithm	Cores	Hours	Days
2007	Frank-Wolfe	1	95	4.0
2040	Frank-Wolfe	1	109	4.6
2007	Conjugate FW	1	77	3.2
2007	Bi-conjugate FW	1	75	3.1
2007	Conjugate FW	4	37	1.5
2007	Bi-conjugate FW	4	33	1.4
2040	Bi-conjugate FW	4	48	2.0

Effect of Cube Cluster on Results

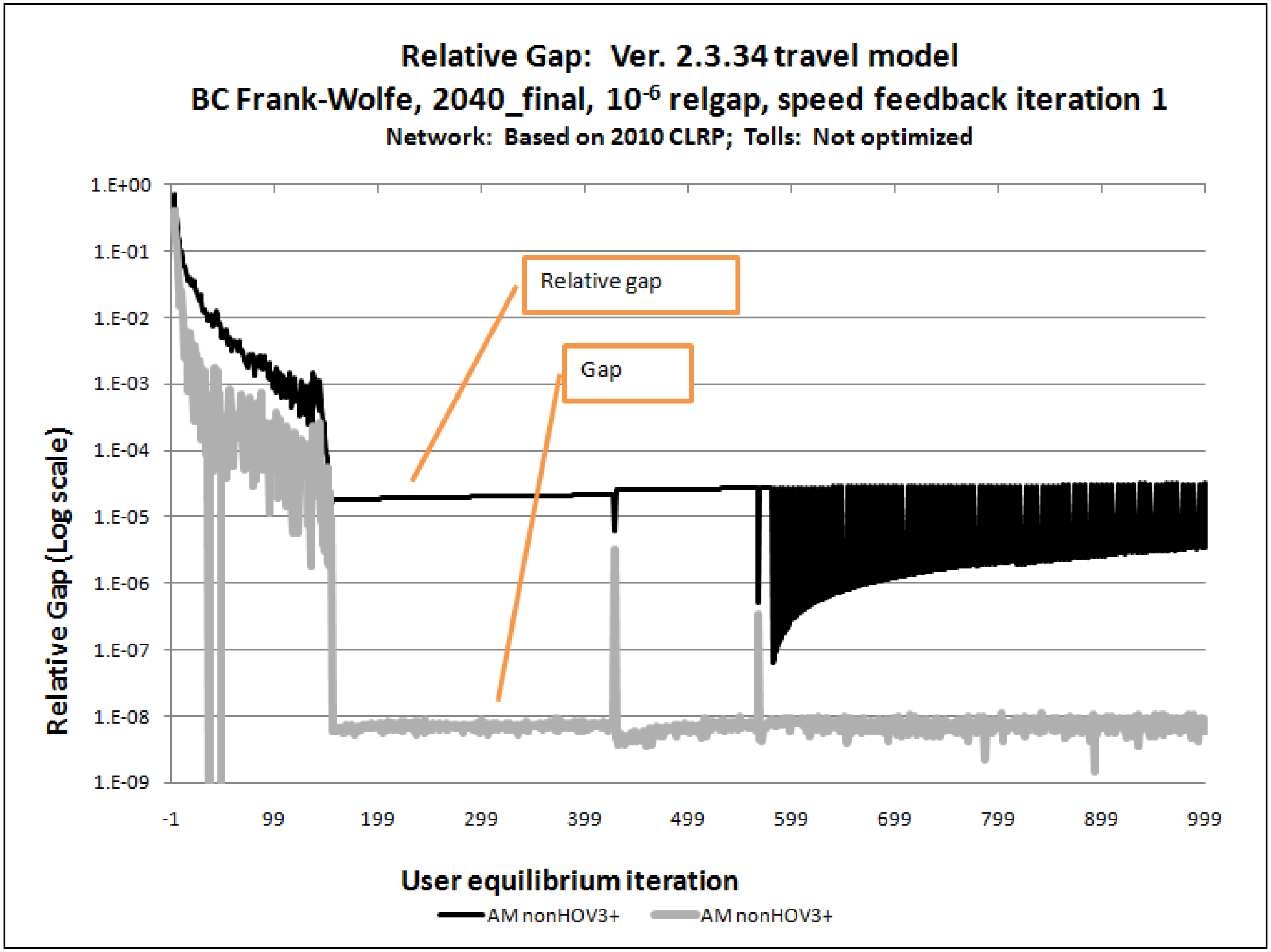
- Parallel assignments generated small VMT differences; some significant volume differences; and illogical convergence results
  - "Standardize" to 4 core computing as a result

Year	Algorithm	Cores	VMT	% Diff
2007	Conjugate FW	1	156,698,908	
2007	Conjugate FW	4	156,653,683	-0.03%
2007	Bi-conjugate FW	1	156,697,741	
2007	Bi-conjugate FW	4	156,674,456	-0.01%

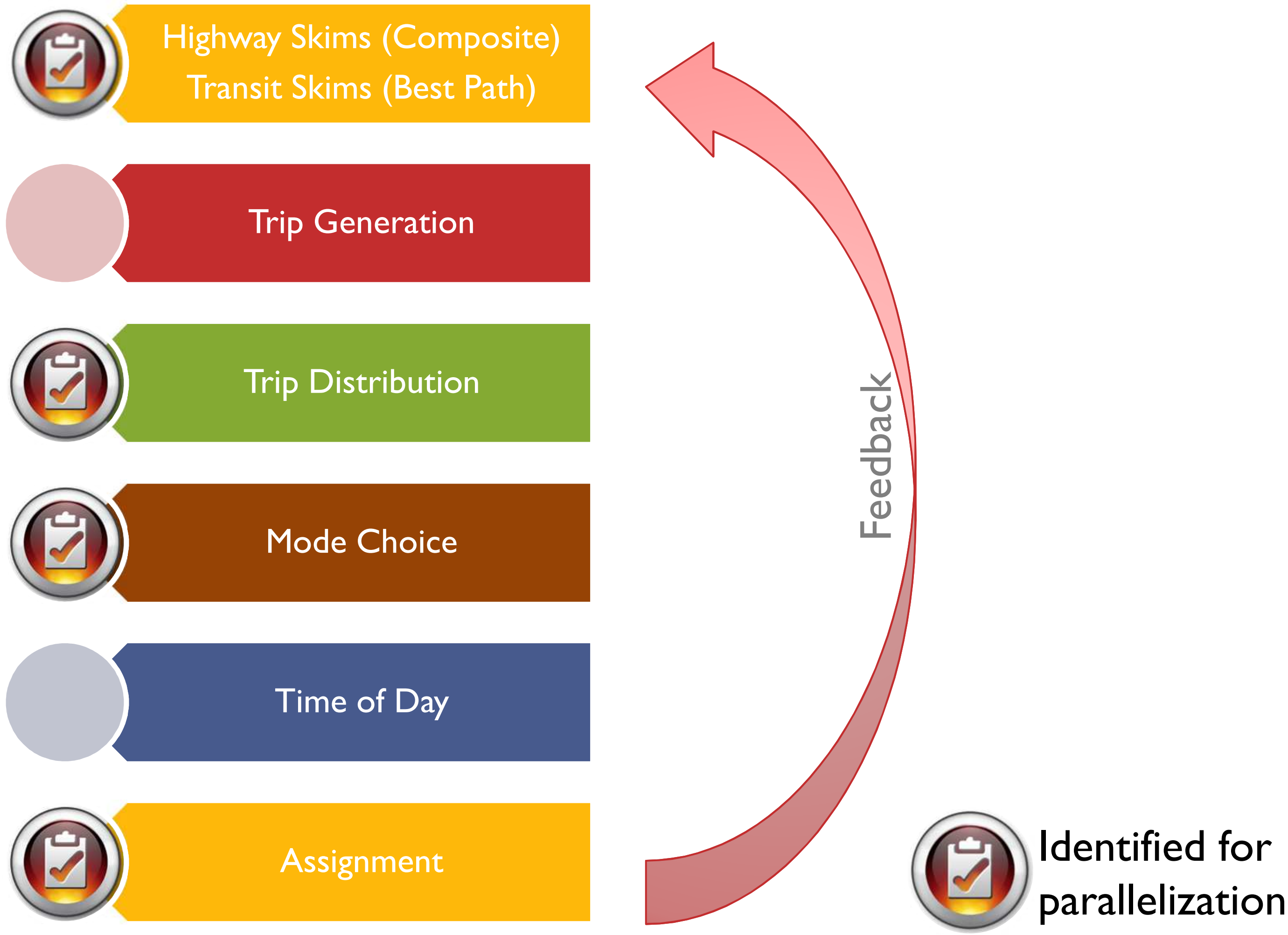
Daily Volume Different by > 20%



Assignment Convergence Problem

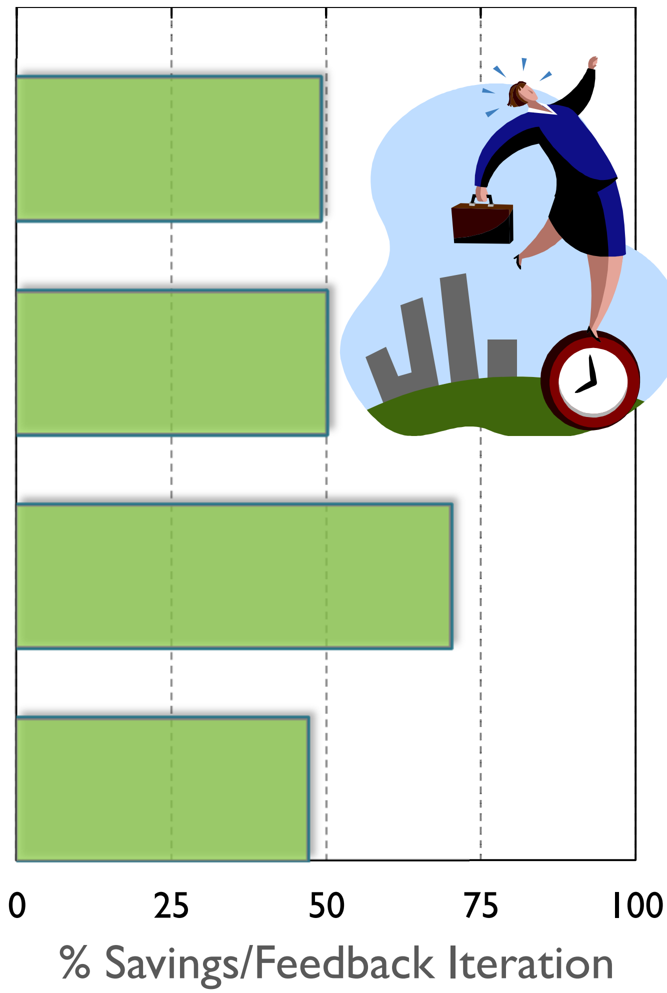


Performance Improvement Task



Time Savings from Parallelization

- Highway and Transit Skims
  - Process time periods together
- Trip Distribution
  - Process trip purposes together
- Mode Choice
  - Process trip purposes together
- Highway/Transit Assignment
  - Process time periods together
  - Combine non-HOV and HOV
- ~40% savings for all steps



Relaxing the 4 Core Standard

- Parallelizing the WMATA post-processor
  - Using a 64 core server
- Implemented outside of Cube Cluster

Process	Steps	Sequential	Parallel
Access Links	4 steps	35 minutes	2 minutes
Path Building	22 paths	66 minutes	3 minutes
Transit Fares	22 updates	60 minutes	3 minutes
Mode Choice	6 models	60 minutes	10 minutes
Assignment	22 tables	66 minutes	3 minutes
Total		227 minutes	21 minutes

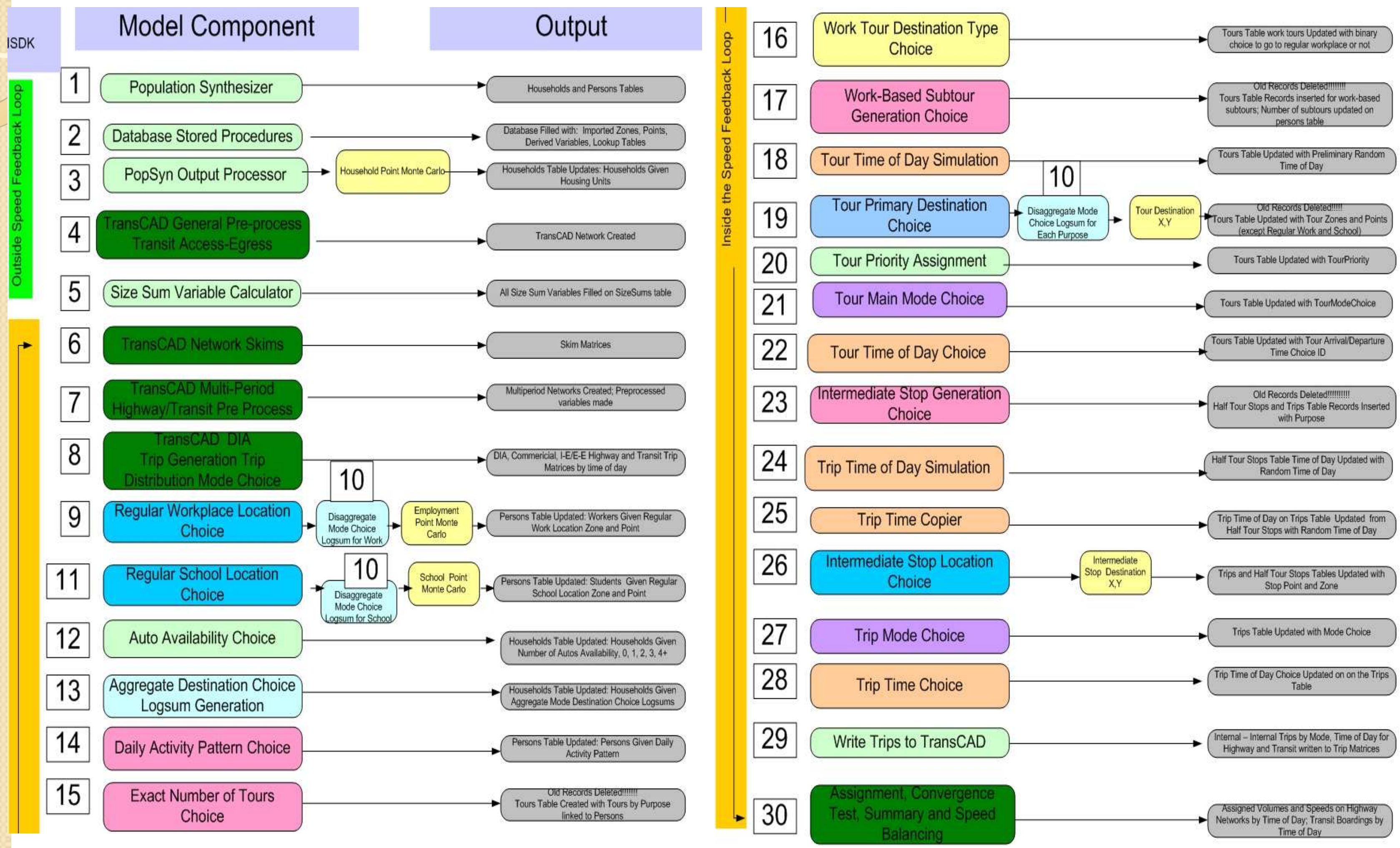
Lessons and Findings

- Significant computational saving can be achieve by simple parallel processing
  - The basic concept and processing mechanism is not well understood by traditional TDF modelers
  - Highway assignment is still the major bottleneck
- Changing the computing configuration should not change the model results
  - Single CPU, multi-threaded or computer cluster

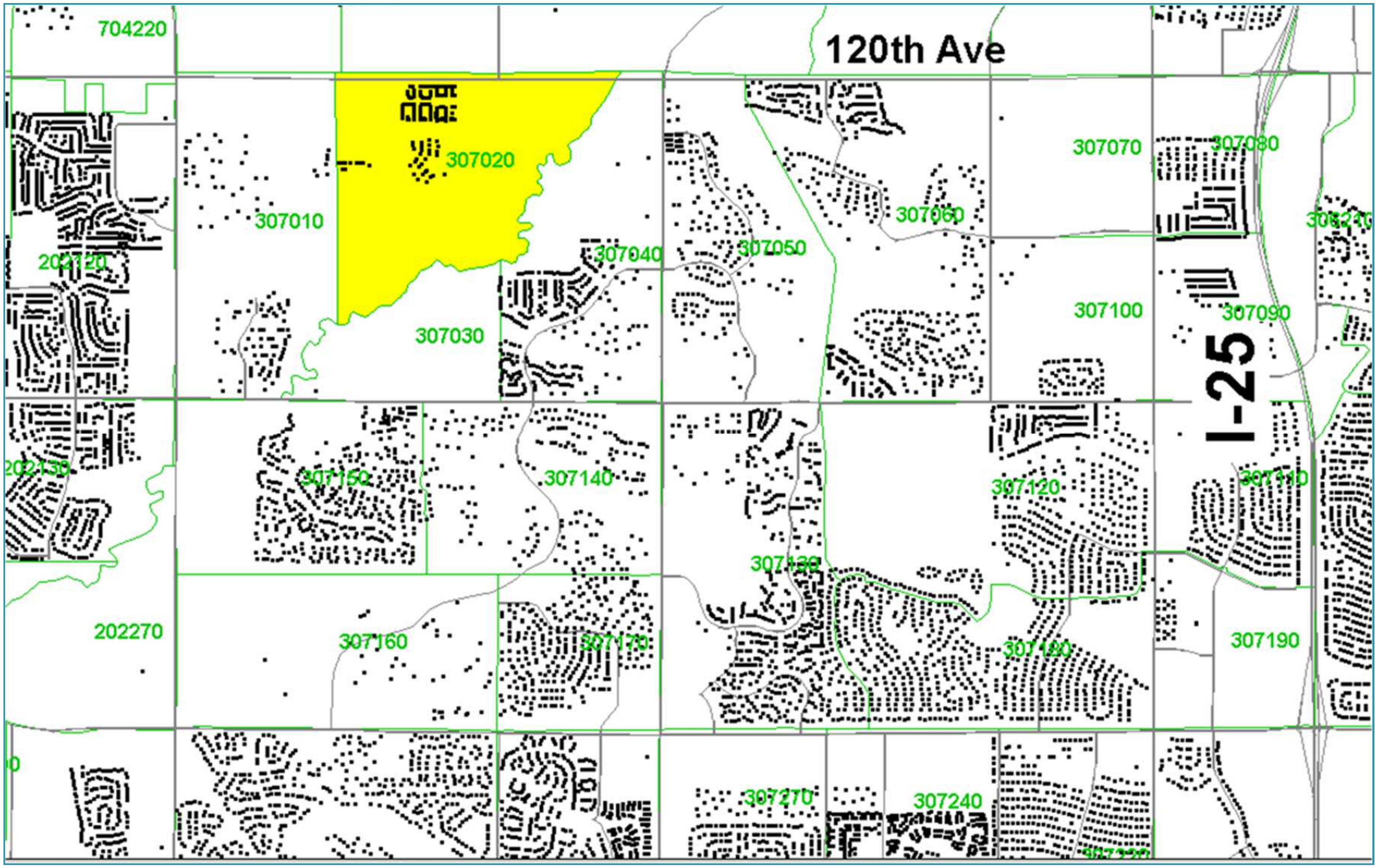
DRCOG / RTD – Denver

- DRCOG developed tour-based FOCUS model
  - TransCAD 5.0, C# and SQL Server
  - Windows Enterprise Server, 32 CPUs, 64GB memory
  - 2,832 zones and activity points
  - 8 modes, 6 tour purposes – DaySim family (CS)
  - 10 highway time periods, 4 transit time periods
  - 3 feedback loops → 60 hour runs
  - Includes simplifying compromises to reduce run times
- RTD still uses older trip-based COMPASS model for FTA New Starts work

Basic Modeling Steps



Zone Points


120th Ave
I-25
704220
307020
307010
307070
307080
307060
307040
307050
202120
307100
307030
307090
307150
307140
307120
307110
202270
307160
307190
307270
307240

Model Migration Plans and Issues

- Seeking FTA acceptance for New Starts work
  - How / where to freeze the "trip tables" for SUMMIT
    - Tour Mode Choice appears to be the best option
  - Improve model consistency throughout the process
  - Remove simplifying compromises in transit models
- Hardware/software distribution issues
  - Expensive purchase – ~$35,000 hardware/software
  - Hard to install and operate – locked server room
- Address run time bottlenecks
  - Current process is only using about 10% of CPUs

Optimize Database Interface

- Standard database software is not well suited to transportation applications
  - Designed for fast queries and interactive editing
  - Updating all data records by sequential or multi-threaded writes is expensive
    - Record locks and index/relationship maintenance is time consuming
  - Faster to re-create the database using a bulk load and relaxed relationship checks
    - Create a "trusted" partnership between the transportation models and the database software
    - Update indices and relationships "offline"

Park-&-Ride Partially Implemented

- Only Tour Mode Choice
  - No intermediate stops or trip mode choice → P-A loading
  - Impedance based on 2*outbound path
- Not modeled like other modes
  - TransCAD path building limitations
- Suggested Improvements
  - Build return trip skims through the outbound parking lot
    - Park-n-ride (O-P-D) + walk-transit (D-P) + drive (P-O)
  - Use the walk-transit and drive skims from the return trip time period
    - For example: AM outbound → PM return
  - Assign in O-D format like other modes

TransCAD Performance Issues

- Reading O→D and D→O matrix cells from 100+ tables is prohibitive
  - Transpose D→O and attach to O→D rows
    - AM depart – AM, MD, PM, EL return
- Implementation options
  - GISDK with cell reads/writes → 30 hours
  - GISDK with vector reads/writes → 3+ hours
  - Custom software using CaliperMTX.dll → 8 minutes
    - Read tables into memory, merge, and write
- Streamline other model components

Lessons and Findings

- Databases are nice for analyzing results, but need to be optimized for in-line modeling
  - Store data in memory or flat files and do bulk loads to the database at the end of the process or off-line
- "Standard" GISDK interfaces are not always the most efficient approach to TransCAD models
  - Manipulate files for processing efficiency
  - If you have memory, use it

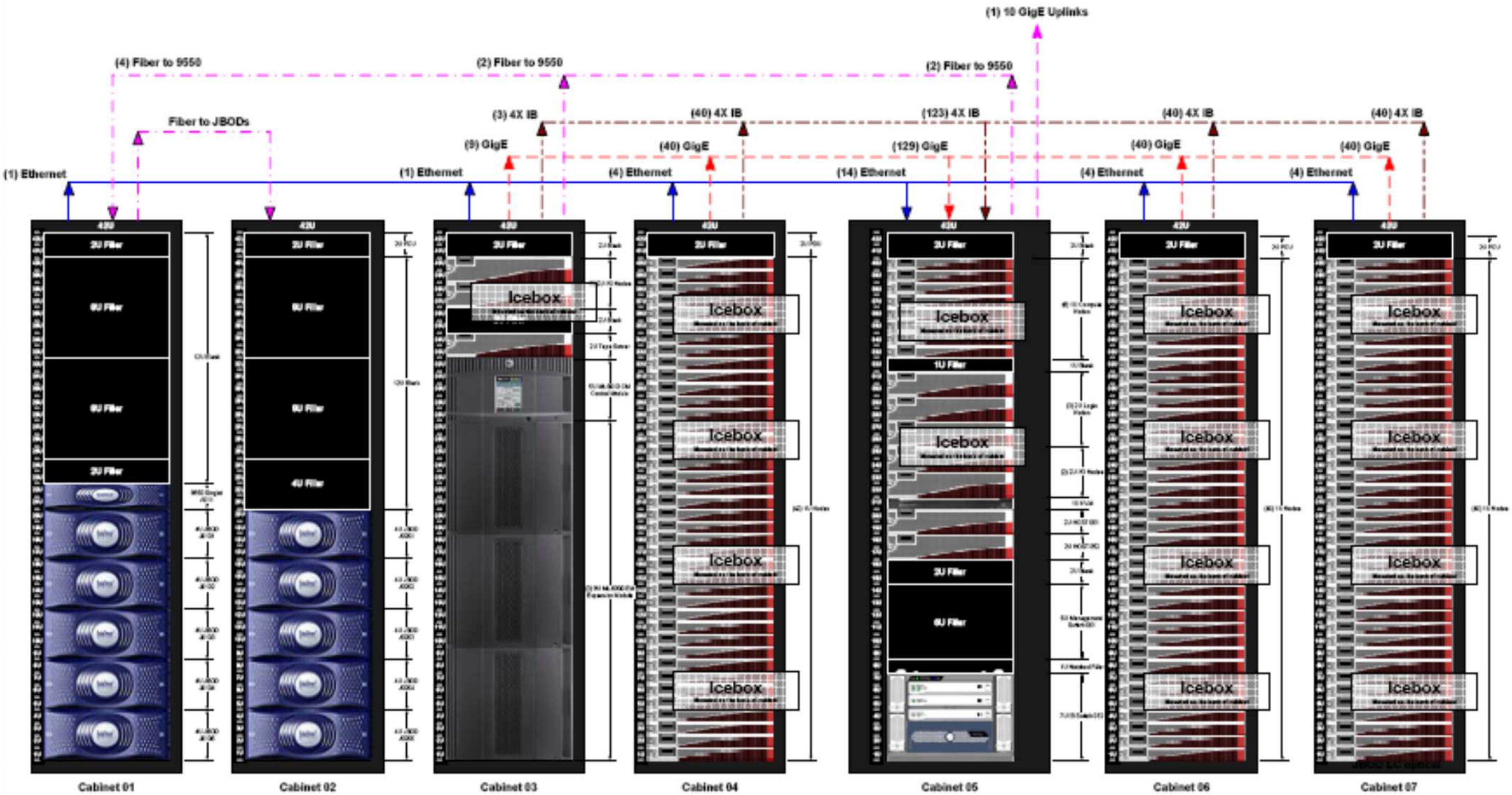
SHRP2-C10 – Jacksonville

- DaySim tour-based demand model
  - Parcel-based, 30 minutes activity schedules
    - 22 time periods (30 mins. in peaks) by1335 zone skims
  - Outputs person activities assigned to one minute schedules and TRANSIMS activity locations
- TRANSIMS regional simulations
  - Dynamic user-equilibrium simulation
    - One second time steps for 27 hour day
    - Trip gap and link gap convergence criteria
  - Generates zone-to-zone skims using 5 minute increments of link and turning movement delays

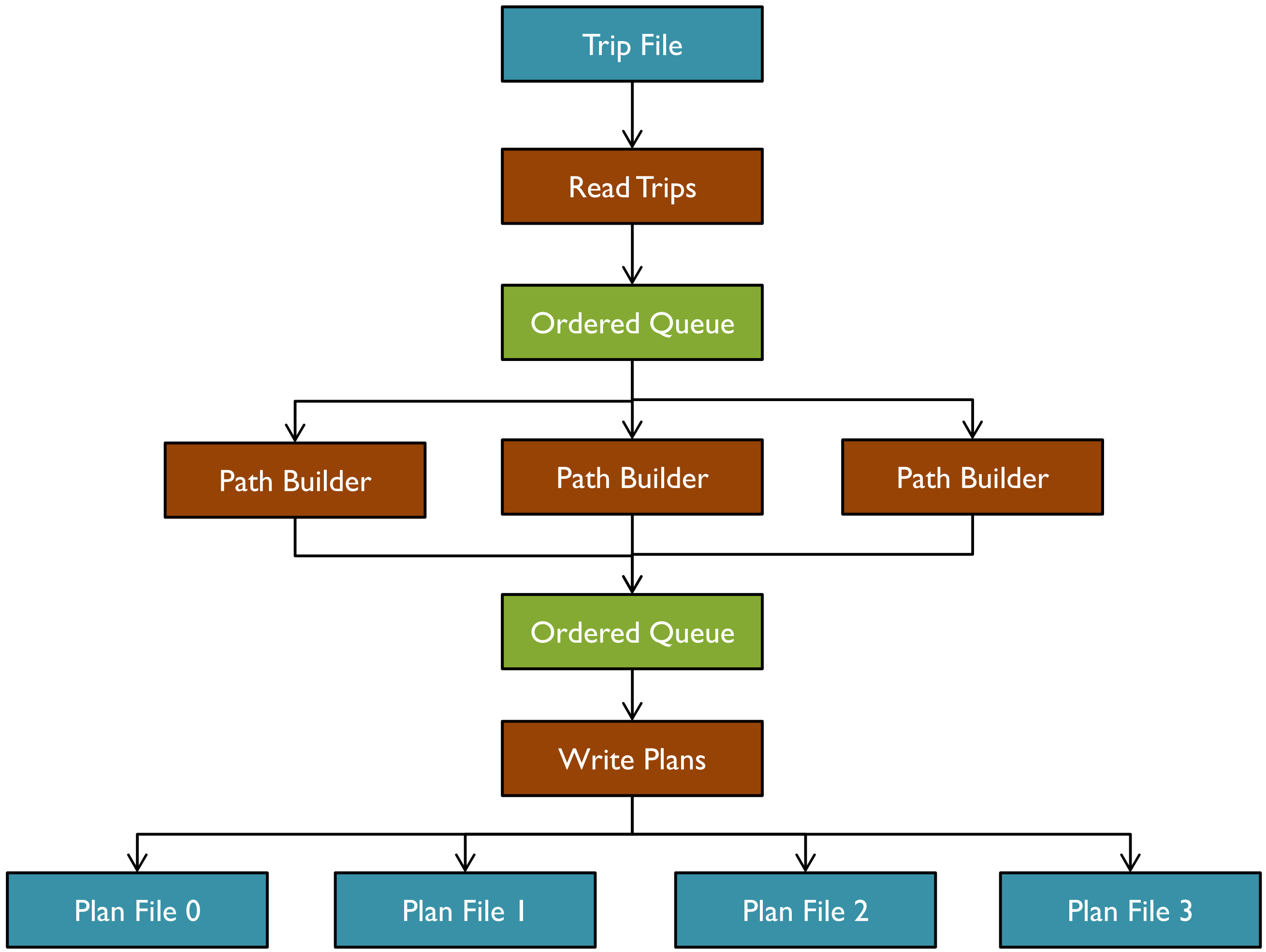
Processing Time

- TRANSIMS (v4.0) Router – Microsimulator
  - Typically run on the TRACC Linux cluster using parallel processing of single thread executables
    - 16 or more plan partitions for Router and Plan processing
    - Single CPU for PlanSum and Microsimulator
  - Each iteration takes ~3.5 hours
  - Network stabilizes in about 20 iterations – ~3 days
  - RSG gap criteria requires ~60 iterations – ~9 days
  - RSG does 4 global iterations – ~36 days
- Upgrade to TRANSIMS (v5.0) for MPI and multi-threaded software performance

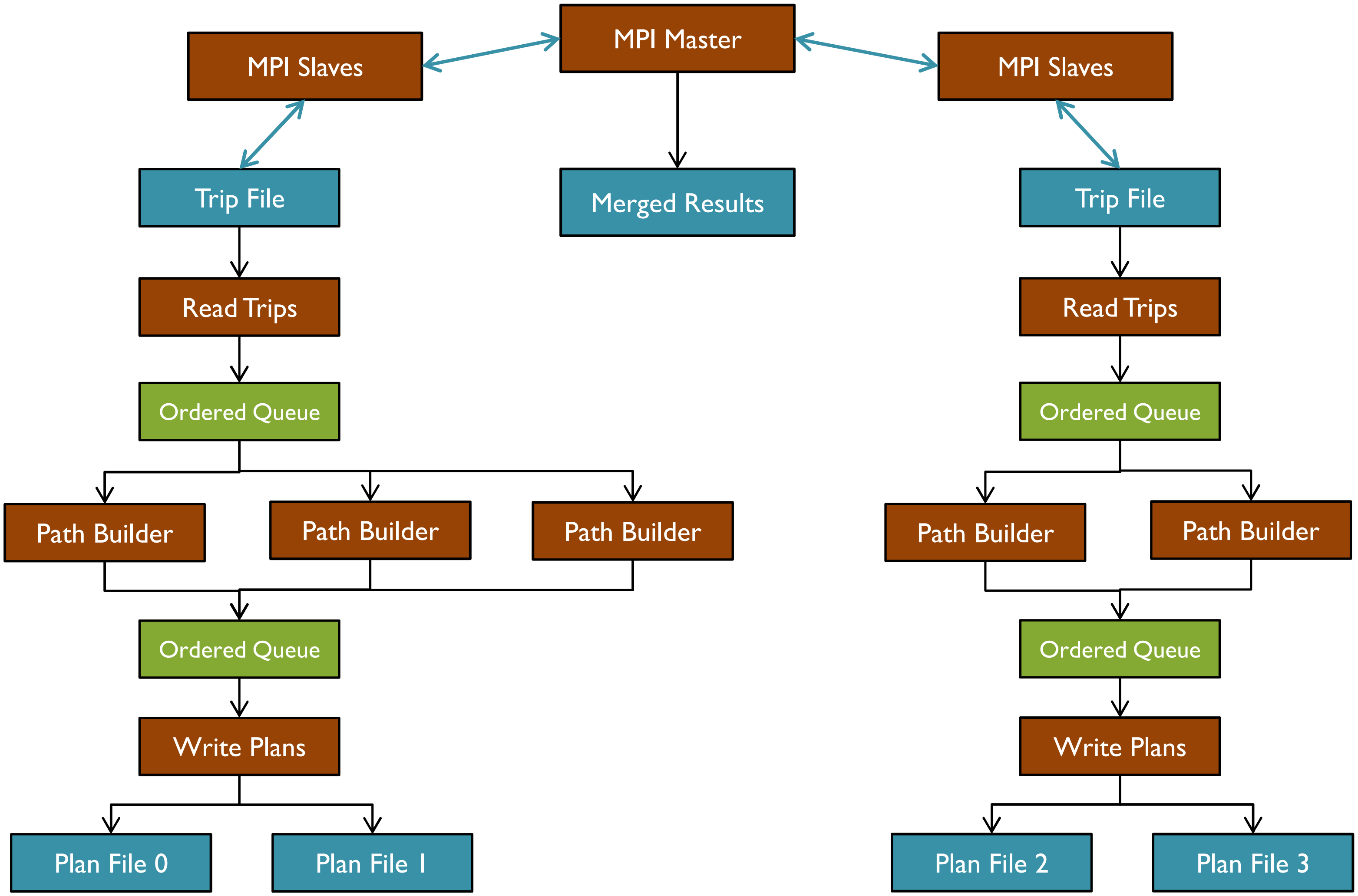
TRACC Computer Cluster



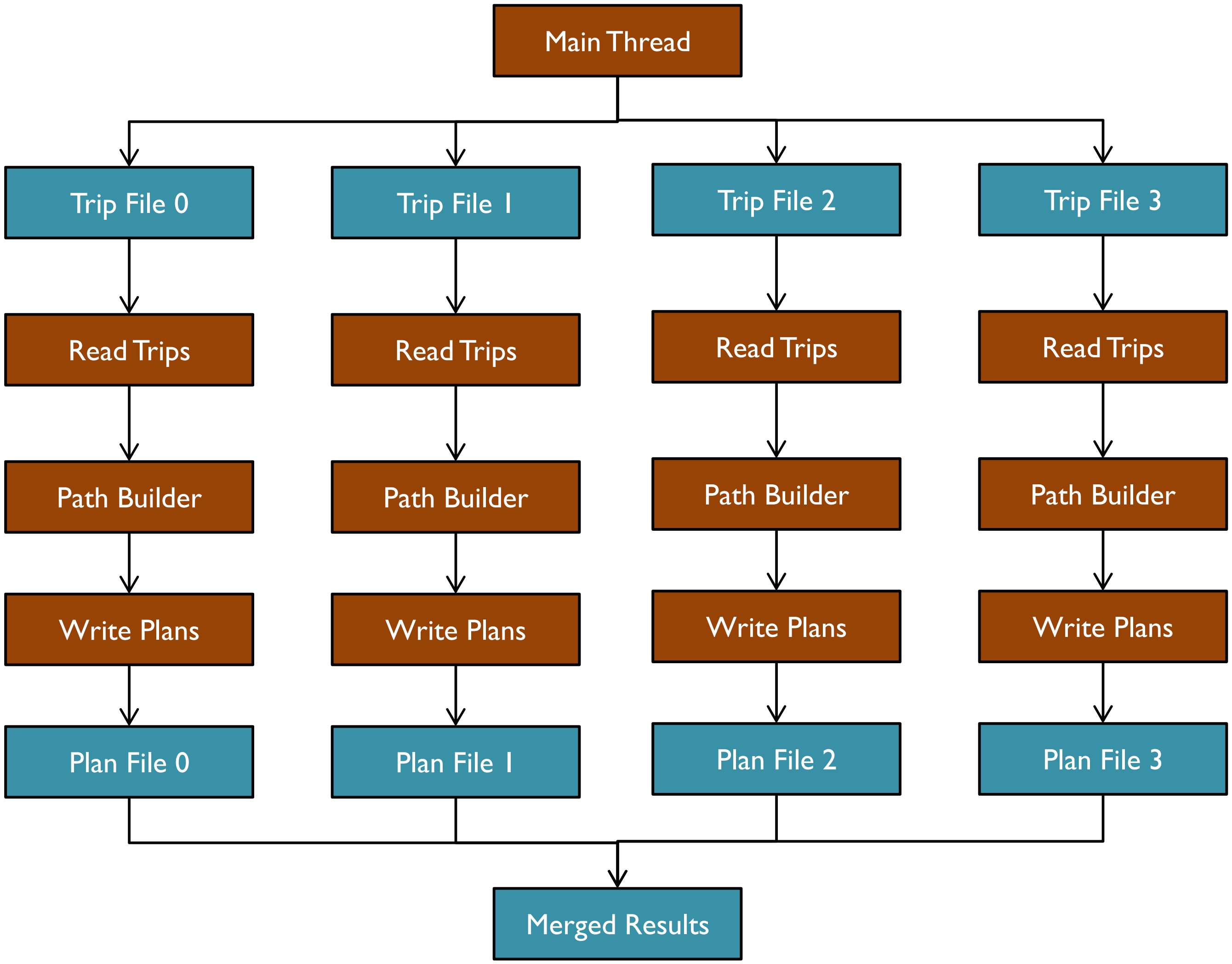
Router Multi-Threaded Design



Router MPI Design



Router Large Server Design



Integrated Computations

- File input/output and data manipulation is a huge component of run times
  - If the computer has sufficient memory and CPUs, keep data and iterations in memory
- Simulator processing and convergence
  - Time sorted, geographic partitions
  - Vary level of computational resolution by iteration
    - Time-based flows-speeds to macro, meso, micro-simulation
- Integrate Router and Simulator
  - Build paths that start at each time increment

Software Performance Lessons

- Isolate independent calculations
  - Processing threads or partitioned applications
- Preserve input/output order
  - Processing threads write to an ordered queue that an output thread uses to write to the output file
- Avoid file or data locks and input/output
  - Write to shared memory if data records are fixed and the thread has unique record ownership
  - Use private thread-based memory to hold data until a data exchange is required

Modeling Principals

- Changing the computing configuration should not change the model results
  - Single CPU, multi-threaded or computer cluster
- Changing the data partitioning or re-running the model should not change the model results
  - Random impacts should be consistent and reproducible
- Balance run times with computer requirements
  - Advanced models require high performance computing → application and staffing challenges