APPENDIX 2

STAGE 3 (CUBE) HANDOUTS AND LECTURE MATERIALS

cube 5

Discover Cube
the GIS Transportation Planning System

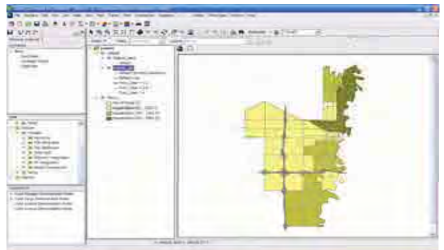
citilabs

cube 5

Introduction to Cube

What's Cube?

- Cube is a complete travel forecasting family of software products that provides exceptional and easy to use capabilities for the comprehensive planning of transportation systems
- Made of many Functional Libraries:
  - Cube Base
  - Cube Voyager
  - Cube Analyst
  - Cube Avenue
  - Cube Dynasim
  - Cube Cargo
  - Cube Land
- Developed by **Citilabs** (Usa/UK)



citilabs

citilabs
TRANSPORTATION MODELING SOFTWARE

Citilabs & its clients

- Citilabs was created at the beginning of 2002 via merger of UAG (a software company based in the US) and the software division of MVA (UK company)
- Company is owned by the French engineering group **Systra** (RATP/SNCF)
- Main offices:
  - Tallahassee
  - London
  - Milan
  - Paris
  - Mumbai
  - Beijing


Cube Base

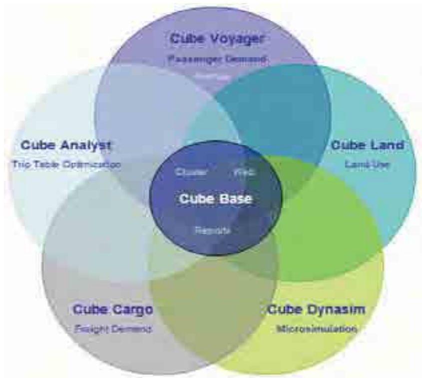

Cube Dynasim

Citilabs & its clients

- With More than 2500 users in 70 countries, Cube is one the most used travel forecasting systems in the world
- Advanced methods:
  - Cube is a modular, integrated, full featured product line for the transportation planning process (passenger demand, freight demand, microsimulation, air quality, reporting).
  - provides answers to all the planning questions from testing new public transit alternatives to road pricing strategies.

Introduction to Cube

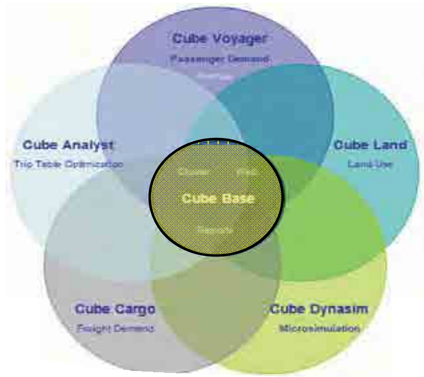
A comprehensive transportation planning system



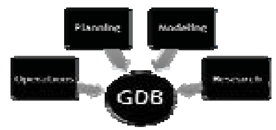
- Cube:
  - A common interface (Cube BASE)
  - Functional libraries
- Cube allows to:
  - add functions as required...
  - ...without the need to learn a new interface or create multiple planning databases
- Cube is seamlessly integrated with:
  - ArcGIS software (ESRI),
  - Microsoft Office
  - User programs

Introduction to Cube

A comprehensive transportation planning system

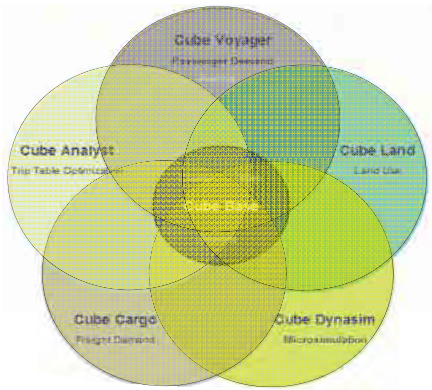


- Cube Base – build, edit, run, present:
  - Cube GIS
  - Application Manager
  - Scenario Manager
- Integrated with ESRI ArcGIS



Introduction to Cube

A comprehensive transportation planning system



- Cube Functional libraries
  - Voyager: passenger forecasting
  - Land: integrated land use – transport forecasting
  - Analyst: Matrix Estimation
  - Dynasim: dynamic, multimodal micro simulation
  - Cargo: commodity based freight forecasting
  - Avenue: mesoscopic assignment

Cube in detail

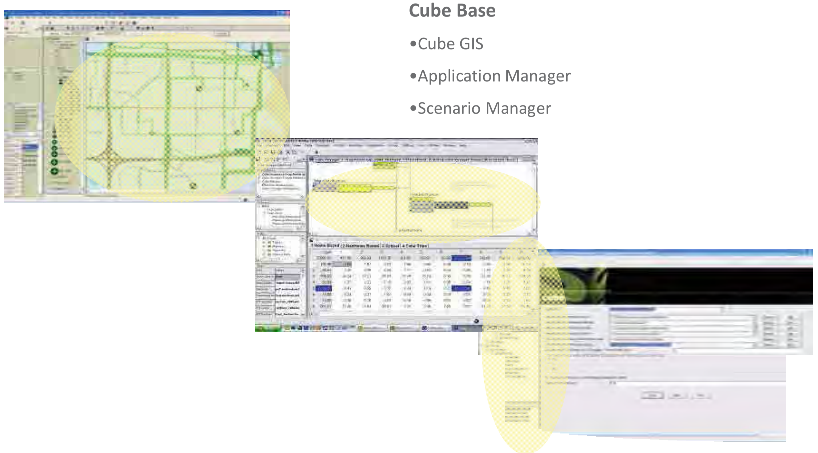
citilabs
TRANSPORTATION MODELING SOFTWARE

cube

Cube Base

Cube Base

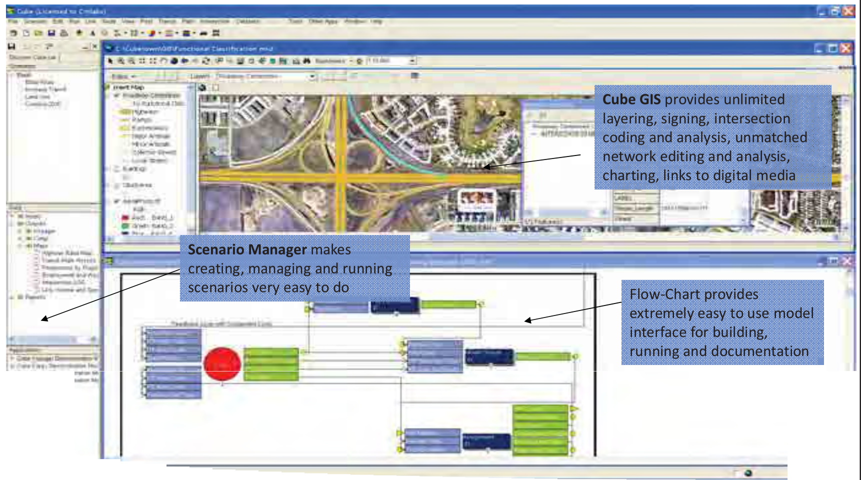
- Cube GIS
- Application Manager
- Scenario Manager



citilabs

cube

Cube Base



citilabs

Cube Base: Application Manager

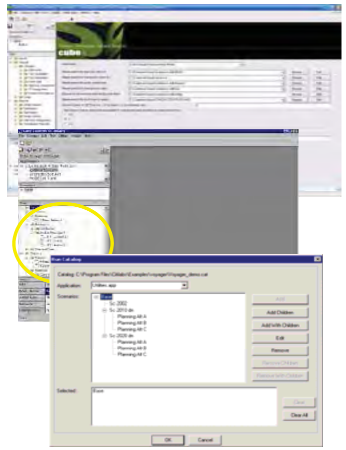
Application Manager uses a flow-chart system for designing, coding, documenting and running the model. Input and output data is shown in the context of the model and can be immediately viewed or edited by double-clicking on it.

- Application Manager makes it easy to present a model in a clear way to others. Features include
  - A pull-down menu to choose your model functions. Each model step has input data on the left, output data on the right.
  - File Linkage (networks, zonal data, etc.) and files creations for intermediate steps, then drag and link data from one model step to another.
  - A complete data editing and visualization system.
  - An easy to use interface to run part of the model or the whole process

Cube Base: Scenario Manager

Scenario Manager highlights key model parameters and data for easy creation and testing of scenarios.

- A set of input data is a Scenario, and "Scenario Manager" is the graphic interface for scenario creation, editing and management
- Forecasting has never been easier. Highlight the parameters and data in your model that you wish to change between scenarios.
- Cube automatically locates these values and creates menu prompts. An easy-to-use graphical interface allows users to run specific or all scenarios with no additional intervention.



Cube Voyager

Advanced methods for passenger forecasting

Cube Voyager combines the latest in Citilabs' technologies for the forecasting of personal travel.

Cube Voyager uses a modular and script-based structure allowing the incorporation of any model methodology ranging from standard four-step models, to discrete choice to activity-based approaches.

Advanced methodologies provide junction-based capacity restraint for highway analysis and discrete choice multipath transit pathbuilding and assignment. Cube Voyager includes highly flexible network and matrix calculators for the calculation of travel demand and for the detailed comparison of scenarios.

Cube Voyager was designed to provide an open and user-friendly framework for modeling a wide variety of planning policies and improvements at the urban, regional and long-distance level.

Cube Voyager: Network & Highway

Creates detailed representations of roadway segments, intersections and ramps.

This module creates a comprehensive roadway and junction database

Capability to estimate point-to point paths and associated travel times, costs, and distances.

Highways: used to estimate zone to zone paths and matrices of impedances for use in analysis and in demand models.

- Paths can be built using a variety of methodologies: All-or-Nothing, All-shortest paths, Stochastic
- Capacity restraint: intersection, link or both using standard processes or user supplied curves or equations.
- Junction-based assignment: Cube Voyager includes methods for performing intersection constrained assignments. This process provides the best estimate of travel flows, delays and queues in urban locations

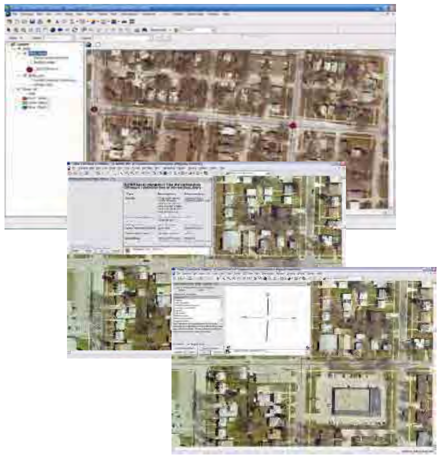
Cube Voyager: Highway

A key element in Cube Voyager is the ability to represent junctions as:

- Signaled and non signaled controlled intersections
- Priorities
- Stop
- Roundabouts

Cube calculate junction capacity on the base of:

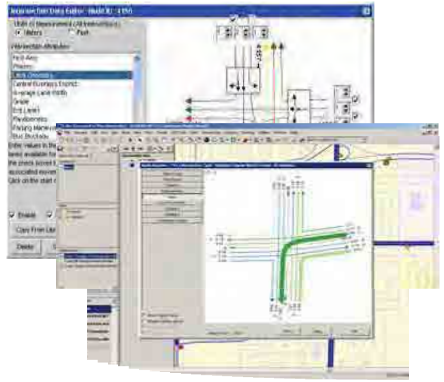
- Geometric characteristics (number of lanes per approach, lane sharing, circle diameter…)
- Functional characteristics (signal type, cycle duration, green/red times…)
- HCM Settings (Visibility, grade, pedestrian and bus…)



Cube Voyager: Highway

At the end of the assignment process, for each detailed junction, Cube gives information as

- Total delay per approach and for the junction as a whole
- Maximum and effective capacity
- Volume/Capacity ratio per single approach and for the junction as a whole
- HCM Level of service
- Maximum assigned flow
- Maximum and cumulative queue



Cube Voyager: Highway

One of the most innovative Cube characteristics' is the "path analysis" tool. A "path" file contains all the relevant information about the path building process and the assignment process. It is then possible:

- To understand the modifications among different iterations in a capacity restraint assignment process
- To fully analyze different classes behavior and network use
- select link/node/path "on screen"
- Best path and path optimization
- ...

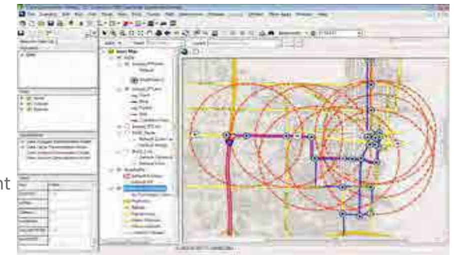
Cube Voyager: Public Transport

Cube Public Transit functional library provides advanced functionality for the study of public transit systems.

- provides the capacity to study even the largest and most complex public transport systems
- Unlimited lines, unlimited links and unlimited modes.
- Automated processes for creating walk, automobile and transfer links between services.
- Ability to represent infrequent and time coordination.

Point to point paths are found using a variety of techniques:

- Discrete route paths based on
  - All or-nothing,
  - Stochastic multi-path
- The Discrete route process of Cube allows to retain all the routing information for subsequent analysis



Cube Voyager: Public Transport

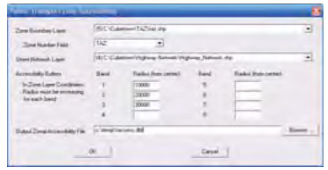
PT Modeling Features

- Advanced fares modeling
  - Combining fare systems for modes, operators, lines
- Advanced crowd/congestion modeling
- User control over all aspects of the public transport model
  - Preparation of a PT Network for Public Transport's modeling functionality
  - Generation of the non-transit element of the PT network, i.e. access, egress, transfer and park and ride legs
  - Demand stratification by user class
- Results/Analyses
  - Skimming, network-wide and mode specific, composite and average journey costs, and components of costs
  - Loading analyses - transfers between modes, operators, lines, entry-exit stations

Cube Voyager: Public Transport

PT Modeling Features

- Transit access geo-processing:
  - calculate and save zone-level access to transit stops
- Get selected link/node trip tables
  - Link Specific
  - Node Specific
  - Mode specific
  - Line specific
  - Operator Specific
  - Any combination of the above
- Useful for evaluating project demand
- Can identify/output percent or probability of trips IJ using the selection



citilabs
TRANSPORTATION MODELING SOFTWARE

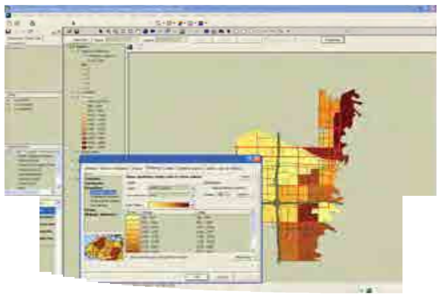
cube 5

Cube Voyager: Demand Modelling

Cube Voyager Demand processes zonal data and matrices according to user specified expressions. Zonal data and matrices are input, and matrices and reports are output.

There are no default processes. The powerful scripting language combined with user-friendly wizards, allows for the application of all types of commonly used demand processes:

- multinomial and hierarchical choice models
- cross classification and regression models
- gravity models
- matrix frataring



citilabs

cube 5

Cube Analyst

citilabs

Matrix Estimation: Cube Analyst

One of the most valuable pieces of data in travel demand forecasting is the matrix representing existing travel. It is the basis for forecasting and for almost all important comparative analyses

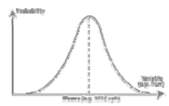
- Cube Analyst is the Cube functional library developed specifically for estimating and updating base year automobile, truck and public transit trip matrices.
- Cube Analyst enables the user to exploit a wide variety of data that contribute to matrix updating and matrix development.
- Cube Analyst has been used successfully on many and varied studies around the world.
- The trip matrices are estimated on a cell-by-cell basis using the supplied data.
- Statistical summaries are output giving indicators of the quality of the estimation.
- Extensive reporting options enable users to establish their own confidence in the results

Matrix Estimation: Cube Analyst

Rigorous Methodology
- Cube Analyst uses the maximum likelihood statistical method.
- A powerful optimizer allows individual cells to be estimated with precision the calculation is self-calibrating

Data Preparation
- The type and quantity of data input to the estimation process is left to the user to determine.
- the more data provided, the more accurate the resulting estimated matrix will be, but it is possible to achieve worthwhile results with limited data.

Integral Quality Assurance
- effects and implications on the estimated matrix of different input data may be studied
- specialist tools indicate the quality of the estimated matrix
- quality analysis of estimated matrix guides cost effective and selective data surveys when required



Cube 5

Cube Dynasim

citilabs

Cube 5

Microsimulation model: Cube Dynasim

Cube Dynasim is a microsimulation system developed by Citilabs and Dynalogic (France) in partnership. Dynalogic specialized in microsimulation for 10 years and now Dynasim is fully integrated in Cube software suite

- Cube extension for the microsimulation and visualization of vehicle and pedestrian flows.
- Used 'in-house' in France and Italy since the early 90's
- Marketed worldwide as a part of the Cube System since 2003
- Currently is used in many countries for microsimulation: Australia, China, Czech Republic, France, Hong Kong, Italy, Singapore, Switzerland, Portugal, Spain, Thailand, United Kingdom, United States...

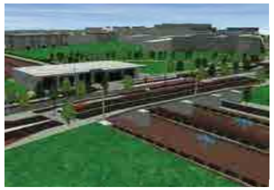
citilabs

citilabs
TRANSPORTATION MODELING SOFTWARE

cube

Microsimulation model: Cube Dynasim

- Scenario-based Simulation
  - Only one Dynasim project for all simulation alternatives
  - Eliminates redundancy
  - Ensures consistency
- Analysis of Multiple Runs inherent to the system
  - Automatically performs multiple runs and summarizes results
  - Ensures a robust analysis with no additional burden on the user
- Interactive Results
  - Completed simulations may be exported to a DynaViews program
  - Interactive Animations with the same features as Dynasim
  - Freely distributable



citilabs

cube

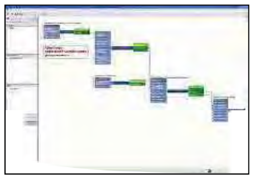
Microsimulation model: Cube Dynasim

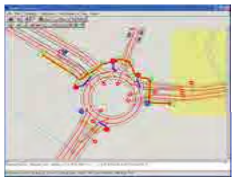
Cube Dynasim captures all of the intricacies of traffic behavior and is able to perform detailed operational analysis of complex traffic flows while realistically emulating the flows of automobiles, trucks, buses, rail, bicycles and pedestrians.

citilabs

Dynasim: Integration within Cube

- Cube Voyager calculates vehicle flows and paths for multiple vehicle classes
- Cube Dynasim takes flows, paths, transit, and intersection information for microsimulation
- Voyager may import data from Cube Dynasim for feedback or visualization
- Allows for a fully integrated modeling system





Dynasim: Integration within Cube

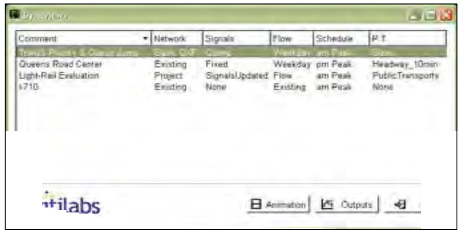
The passage from the macro model to the micros model requires in Cube just a few steps:

1. Identify sub-area and run analysis (flows and routing)
2. Use Cube Analyst to correct for projections, counts, or surveys
3. Load background layers (images, shape, dxf)
4. Confirm geometric information (Junction Editor)
5. Confirm control information (Junction Editor, Import)
6. Extract sub-area network to Dynasim
7. Extract and append additional scenarios
8. Clean-up the imported data
9. Simulate scenarios
10. Extract output data back into CUBE for visualization

Cube Dynaviews

Share and export microsimulation results

- Simulations' animations and data may be exported
- Exported simulations may be freely distributed
- Play animations at any speed
- Zoom, pan, and navigate within the animations



Cube Avenue

citilabs
TRANSPORTATION MODELING SOFTWARE

cube

Why "mesoscopic" models?

Transport modelling is mostly done on a strategic 'macro' level

Can cover a very large area, but...

some "inability" to model the required level of detail in congested areas.

For traffic engineering /traffic control/intelligent traffic management 'micro' models have become very popular and useful

Capture perfectly extreme level of details, but...
suffer data hungriness and can hardly be applied at a wide scale

citilabs

cube

How mesoscopic/macroscopic models differs

Traditional transportation models use macroscopic techniques to study the flow of traffic from point to point volume of traffic between an Origin and Destination = single unit

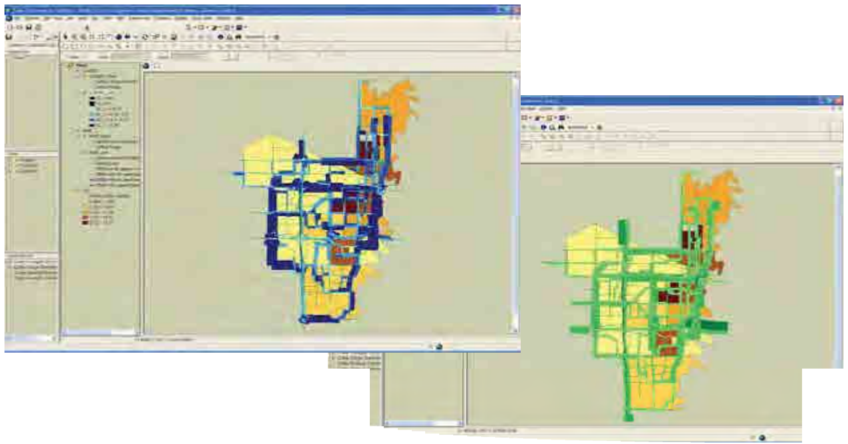
Compute the lowest-cost path for the traffic volume

Compute congestion effects (through volume-capacity ratios and resulting speeds)

citilabs

citilabs TRANSPORTATION MODELING SOFTWARE

Highway Traffic Flow



How mesoscopic/microscopic models differs

Microscopic techniques present the most detail

Microsimulation tools model each vehicle explicitly and captures detailed movements and interactions

Tools for detailed studies, but...

detailed results require extremely detailed inputs (rubbish in/rubbish out more than ever!)

Mesoscopic Models

Mesoscopic techniques can study traffic flows over time

Planner specifies the level of detail for Vehicle, Time, Network details

Mesoscopic model in Cube Avenue computes the lowest-cost path for each vehicle unit , based on its departure time

Computes interactions among vehicle units

Requires less-detailed inputs than Microscopic

Provides better details than Macroscopic

Mesoscopic Models

A mesoscopic model allows to complete new types of analyses:

- Quantify impacts of upstream traffic congestion
- Measure queuing at intersections and merge points in a network
- Isolate secondary impacts from one intersection through another
- Evaluate the benefits of ITS (intelligent transportation system) projects
- Simulate alternative infrastructure, operational, and policy changes to optimize
- Emergency evacuation plans and strategies
- Test strategies to improve arrival and departure from stadiums and other special event facilities
- ...

Mesoscopic Modeling in Cube Avenue

Cube Avenue is a dynamic equilibrium assignment model

Loads and tracks the movement of vehicle packets throughout the highway network. It models:

- traffic signals,
- roundabouts,
- stop-controlled intersections,
- ramp merges.

Vehicle packets move, stop, and queue through upstream roads and intersections. Cube Avenue calculates optimal network conditions

Mesoscopic Modeling in Cube Avenue

Dynamic bandwidths:

- flows
- queues
- Vol/Cap ratio
- ...

Dynamic data at junctions:

- Level of service and delays
- Queues
- Turning Volumes
- ...

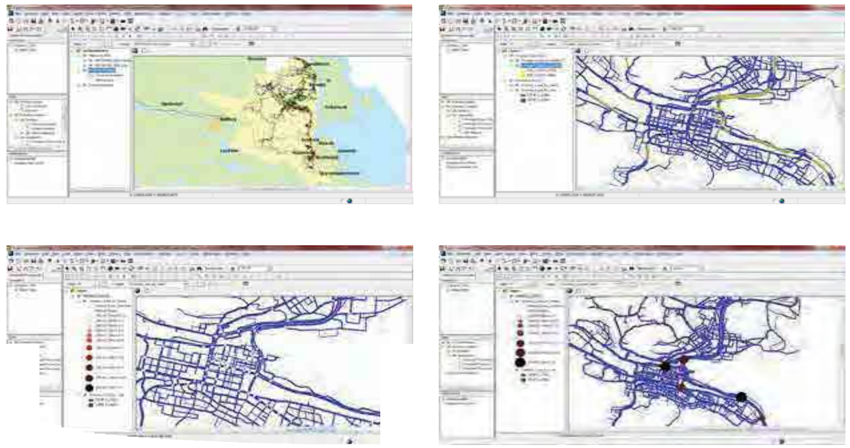
Mesoscopic Modeling in Cube Avenue

Packets of vehicles can be animated

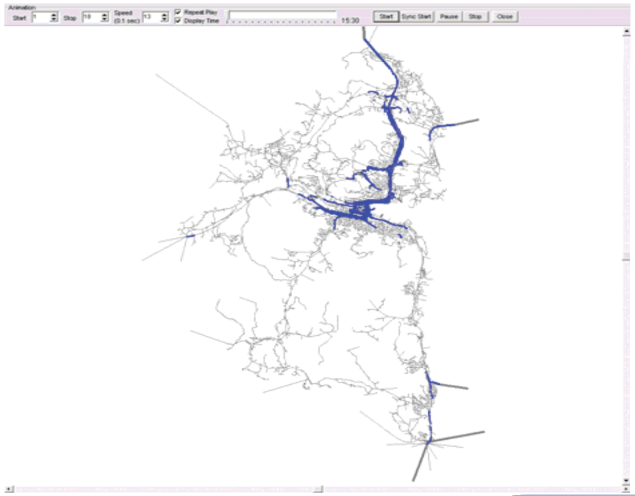
It is possible to set selections based on:

- Use of links/nodes
- Origins
- Destinations
- Time Period

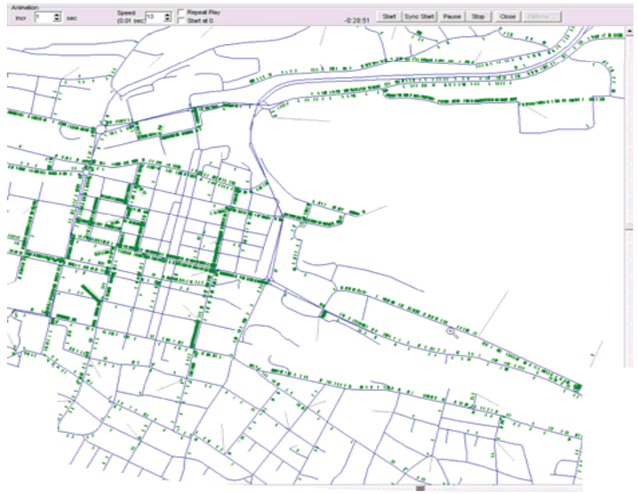
Cube Avenue application: Sundsvall Municipality



Cube Avenue application: Sundsvall Municipality



Cube Avenue application: Sundsvall Municipality



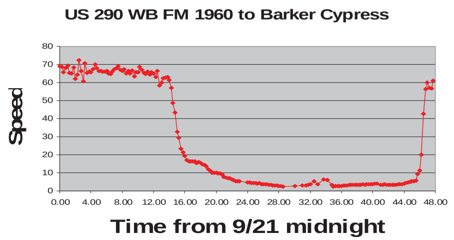
citilabs
TRANSPORTATION MODELING SOFTWARE

Cube Avenue application: Houston evacuation plan

In September 2005, Hurricane Rita landed east of Houston

Well over 1 million people attempted to evacuate from the eight county region

Severe congestion as a results



Cube Avenue application: Houston evacuation plan

cube

Cube Avenue application: Houston evacuation plan

Evacuation routes became "parking lots".

Some people spent more than 18 hours on the evacuation routes

Fatal accidents, abandoned cars, and other safety issues

citilabs

cube

Cube Avenue application: Houston evacuation plan

Why NOT use **traditional (Static) models**?

- No impact of queues
- No ability to deal with upstream impacts
- Links do not directly affect each other
- Not conducive to time-series analysis

Why NOT use traffic **micro-simulation models**?

- Study area of interest too large and complex
- Too much data and memory required
- Too many uncertainties to model accurately

citilabs

Cube Avenue application: Houston evacuation plan

Challenge - Model Size

- 8-county region with 4.7 million population in 2000.
- 3000 zones and 43,000 links
- Around 14,000,000 daily trips modeled

Cube Land

Land Use modeling with Cube Land

Cube Land is a functional library that models land use, prices and household / firm location by simulating real estate markets:

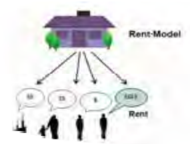
- Cube Land is economic land-use modeling software designed especially for land-use and transport interaction models
- Based upon the MUSSA II framework
- Developed by Dr. Francisco Martinez and researchers at the University of Chile
- Citilabs distributes Cube Land as a library coupled with the powerful Cube Base interface for modeling and GIS

Land Use modeling with Cube Land

- Cube Land can generate extensive information about the market under different scenarios.
- It processes supply, demand, and space in a disaggregated manner, based on the characteristics that describe:
  - Activities to be localized
  - Real estate supply
  - Location of said activities on the properties
  - Values of the resulting land uses
- You can adapt Cube Land to study any area by defining the proper zoning and city-planning policies, such as government incentives or regulation

Land Use modeling with Cube Land

- Cube Land can forecast the location of many different types of agents. For example, Cube Land can forecast
  - the locations of homes, which have different socioeconomic characteristics,
  - the locations of firms, which have different industrial activities.
- With Cube Land, you can include possible changes to the transportation system and study how such changes affect population location and the composition of the region



Cube:
Making the intelligent choice

citilabs
TRANSPORTATION MODELING SOFTWARE

cube

Cube – the intelligent choice

1. Complete Functionality
2. Completely Open
3. Advanced Modeling Functions
4. The Power of GIS
5. Easy-to-Use
6. Transparent
7. Speed
8. Reliable
9. Made and Supported by Modelers
10. Based on Industry-Standards

citilabs

cube

Cube – the intelligent choice

Complete Functionality

Cube packs technically advanced methods within an integrated environment for the forecasting and simulation of personal travel, freight and environmental impacts. Its scripting language provides the ultimate in user flexibility for representing even the most complex methodologies.

Completely Open

Cube allows external or user-developed applications to be seamlessly integrated inside the model. They appear on the pulldown menu and have the same look and feel as all other Cube Extensions, making the development of customized Cube models simple and easy.

Advanced Modeling Functions

Cube Extensions use the latest in methodologies for the forecasting of passenger and freight flows. Features range from an innovative multipath and multimode approach for modeling public transit pathbuilding and assignment to dynamic traffic assignment to a true commodity-based freight system.

citilabs

citilabs
TRANSPORTATION MODELING SOFTWARE

cubes

Cube – the intelligent choice

The Power of GIS

Cube Base seamlessly links data between the model and ArcGIS from ESRI, the world leader in GIS technology.

Easy-to-Use

From its fresh, new look to its intuitive scenario-based design, Cube makes it easier than ever to develop, produce and compare scenarios. You'll be able to get more done in less time with integrated scenario and application tools, manage your data in a snap, and arrange projects and data in a way that makes sense to you. Simply put, Cube helps you work smarter.

Transparent

Say goodbye to the black box. While Cube maintains the industry standard step-by-step approach to travel forecasting, the architecture of the model is clearly shown as well as the flow of data. Double-click on the model parameters and easily understand how the forecasts are made.

citilabs

cubes

Cube – the intelligent choice

Speed

Cube runs applications faster. In most cases, entire sets of scenarios will be completed more quickly.

Reliable

By building on proven forecasting systems, Cube delivers a reliable foundation you can count on to produce comprehensive and accurate forecasts.

Made and Supported by Modelers

Developed and supported by transportation planners and engineers, experts in the areas of passenger demand forecasting, commodity forecasting, microsimulation and environmental impacts.

Based on Industry-Standards

From data formats to software architecture to forecasting methodologies, Cube either sets or follows the standards. Input and output data can be exchanged in ESRI, Excel, and dBase formats.

citilabs

Integrate GIS and Transportation Planning

What's GIS – introduction to GIS

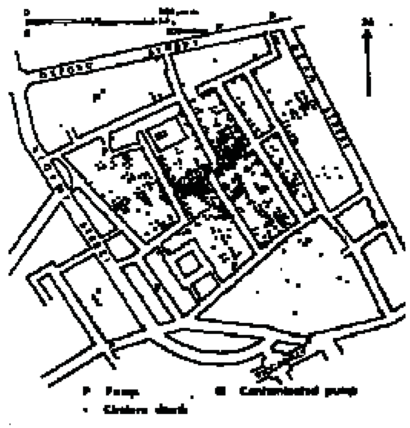
- GIS: Geographic information system
- A geographic information system (GIS), geographical information system, or geospatial information system is any system that
  - Captures,
  - stores,
  - analyzes,
  - manages,
  - presents data that are linked to location

What's GIS – introduction to GIS

- GIS is the merging of cartography, statistical analysis, and database technology.
  - GIS systems are used in cartography, remote sensing, land surveying, public utility management, natural resource management, photogrammetry, geography, urban planning, emergency management, navigation, and localized search engines.
- In a general sense, the term GIS describes any information system that integrates, stores, edits, analyzes, shares, and displays geographic information for informing decision making.

What's GIS – introduction to GIS

- The possibly earliest use of the geographic method: the cholera outbreak in London (John Snow – 1854)
- Modern GIS technologies use digital information



Drawn by Dr John Snow about 1854; shown in Stamp, L. D. 1964. *A Geography of Life and Death.*

What's GIS – introduction to GIS

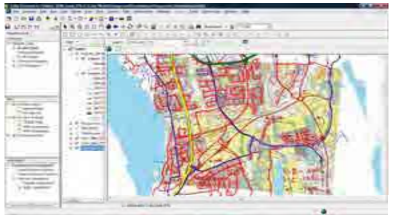
- A GIS helps you answer questions and solve problems by looking at your data in a way that is quickly understood and easily shared:
  - Map Where Things Are
  - Map Quantities
  - Map Densities
  - Find What's Inside
  - Find What's Nearby
  - Map Change

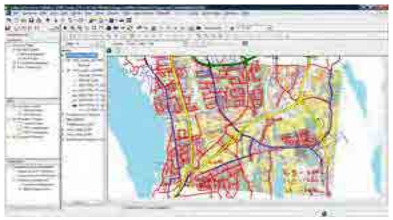
Integrate GIS and Transportation Planning

- Transportation professionals over the world have discovered and embraced GIS as an important tool in:
  - Managing
  - Planning
  - Evaluating
  - Maintaining transportation systems
- GIS for Transportation, named GIS-T by the American Association of State Highway and Transportation Officials, has been used for diverse purposes, such as:
  - Modeling travel demand
  - Improving transit service
  - Evaluating new road scheme

Integrate GIS and Transportation Planning

- In order to predict changes to travel demand, we build models
- Main Inputs:
  - Data/changes in demographics
  - Economic development
  - Social development
  - Policies





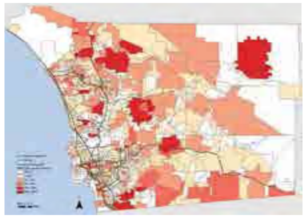
Integrate GIS and Transportation Planning

- Changes to travel demand:
  - Alternative developments
    - land use (GIS)
    - trip production (GIS)
    - trip distribution (GIS)
  - Alternative policies
  - Alternative solutions for:
    - roads (GIS)
    - public transport networks (GIS)

cubes

Integrate GIS and Transportation Planning

- Planning:
  - need to accommodate demand on public infrastructure / responsibility to preserve quality of life and environmental sustainability
  - GIS provides a framework to inform models
  - Shed light on the various transport alternatives

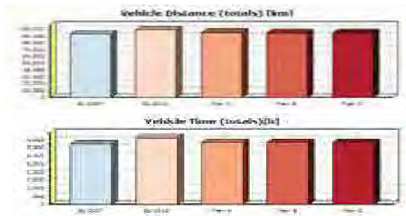


citilabs

cubes

Integrate GIS and Transportation Planning

- What – If ? analyses:
  - Planners develop alternative solutions
  - Models used for testing the alternative solutions
  - What if we choose solution A?
  - How much better is A compared to B, C and D?

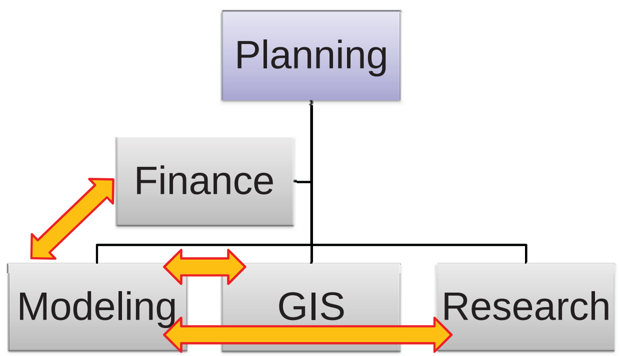


citilabs

Integrate GIS and Transportation Planning

- What We Do With GIS
  - Authoring & editing geographic data
  - Managing spatial databases
  - Analyzing & modeling spatial relationships
  - Basic mapping & cartography
- What We Do With Cube
  - Developing models of transportation systems
    - Developing input data
    - Designing processes
    - Writing scripts
  - Applying existing models to analyze scenarios
    - Editing input data
    - Running alternatives
    - Viewing results

Integrate GIS and Transportation Planning



Integrate GIS and Transportation Planning

- ArcGIS Desktop
  - ArcView
  - ArcEditor
  - ArcInfo
  - ArcPublisher/ArcReader
  - (ArcView, ArcEditor, ArcInfo: *ArcMap, ArcCatalog, ArcToolbox*)
- Server GIS (replaces ArcSDE)
  - ArcGIS Server
  - ArcGIS Image Server
  - ArcIMS (web delivery)
- Geodatabase types
  - Enterprise
  - Workgroup
  - Personal
  - File
- Developer Tools
  - Scripting (Python etc...)
  - Automation (VBA etc...)
  - Extensions
  - Applications

- Microsoft Office
  - Access
  - Excel
  - VBA
- SQL Server
- Visual Studio

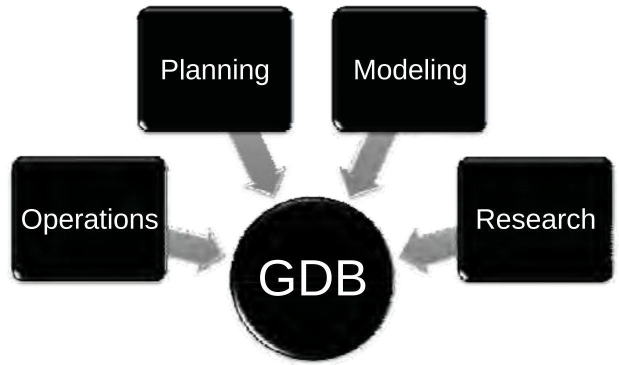
- Cube Base
  - Cube Graphics
  - Application Manager
  - Scenario Manager
- Cube Voyager
  - Cube Cluster
  - Cube Avenue
  - ...Cube Land
- Cube Analyst
- Cube Cargo
- Cube Dynasim

Microsoft Software & Standards

ESRI ArcGIS

Citilabs Cube

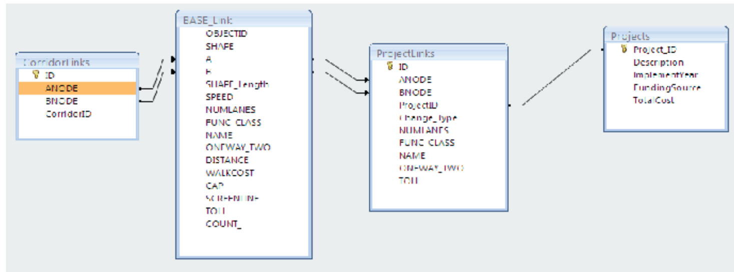
Integrate GIS and Transportation Planning



Integrate GIS and Transportation Planning

- Planning Agencies develop long-range plans under fiscal constraint
- Maintain a database of projects with description, implementation year, funding source, and total cost
- Want to map system alternatives by year and generate reports relating performance to cost
- Add “Project network” with node, link changes and common Project_ID
- Also add Corridor_ID and relate to feature class to tabulate/map by system segment

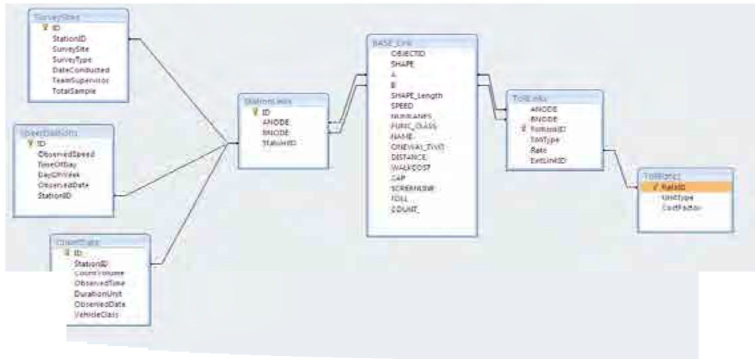
Integrate GIS and Transportation Planning



Integrate GIS and Transportation Planning

- Consultants are charged with forecasting revenues for financial planning & analysis
- Studies can be data-intensive and complex
  - Origin-destination surveys
  - Traffic counts by time of day, day of week, month of year
  - Speed and delay studies (GPS data collection)
  - Independent socio-economic data compilation & review
- Many alternative toll configurations & rates
- Add a Station_ID field to relate data collection sites to network links / system segments
- Store toll system (plazas/gates/sections) in geodatabase to streamline revenue tabulations

Integrate GIS and Transportation Planning

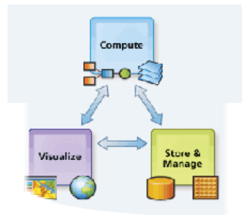


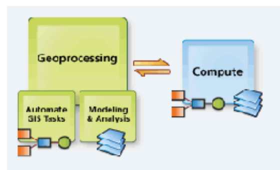
Integrate GIS and Transportation Planning

- Adverse effects of transportation projects on other systems (air, water, ecological, social) must be identified and addressed
- Many disparate issues and themes must be related to transportation project of interest
- Use geodatabase to enable exploratory analysis of interactions between different systems
  - Aerial and satellite imagery (raster data)
  - Natural resource inventories (wetlands, habitats, protected)
  - Historical landmark places
  - Socially and economically disadvantaged areas
  - Health and safety hazards
  - Air quality model outputs

Integrate GIS and Transportation Planning

- Geoprocessing : how you compute with data, connecting data to tools to derive new information
- GIS based systems:
  - They store and manage data.
  - They let you visualize data in a variety of ways.
  - They make it easy to compute with data





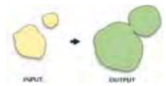
Integrate GIS and Transportation Planning

- For all users – both newbies and "old pros"
  - Wrangling herds of data from one format to another
  - Using a sequence of operations to model and analyze complex spatial relationships
- A typical geoprocessing tool performs an operation on an GIS dataset (such as a feature class, raster, or table) and produces a new dataset as the result of the tool.

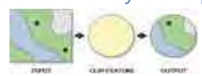
- Each geoprocessing tool performs a small yet essential operation on geographic data. ArcGIS includes hundreds of such geoprocessing tools.

Integrate GIS and Transportation Planning

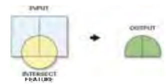
- Buffer



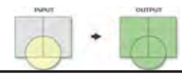
- Clip - Extracts input features that overlay the clip features



- Intersect - Computes a geometric intersection of the Input Features. Features or portions of features which overlap in all layers and/or feature classes will be written to the Output Feature Class.



- Union - Computes a geometric intersection of the Input Features. All features will be written to the Output Feature Class with the attributes from the Input Features which it overlaps.



Integrate GIS and Transportation Planning

- We can use geoprocessing in transpiration planning in analysis such as:
  - Buffer – Use the buffer tool to calculate proximity to stop nodes/parking areas/transit lines…
  - Multi Ring Buffer – Use the multi-ring buffer tool to calculate proximity ranges.
  - Clip – Use the clip tool to extract features or parts of features from a feature class, such as resident population in a range of a specific buffer
  - Intersect – Use the intersect tool for overlay analysis of feature classes.
  - Union – Use the union tool for overlay analysis of feature classes.
  - Multi to Single Part – Use the multi to single part tool to convert a multipart feature class into a single-part feature class supported by Cube Base.

Cube: real models, real GIS

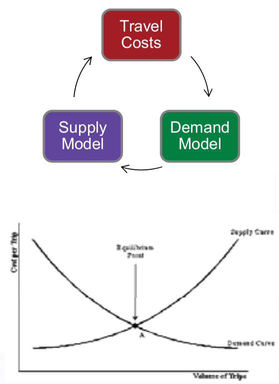
- At the core of the GIS is an embedded version of ESRI's market leading ArcGIS, known as ArcGIS Engine.
- Citilabs developed a specialized application of this technology for transportation modeling by adding transportation topology rules fully exploited within its geodatabase, and a large number of transportation-specific editing and analysis tools.
- The GIS in Cube is an extremely powerful transportation GIS system that is directly compliant with ESRI technologies and provides many of ArcGIS's capabilities, for example, on-the fly projections.

Introduction to Transportation Planning

What is transport planning?

- all useful processes to define and analyze project scenarios
- definition of strategies and intervention policies
- directing the transport system towards optimum configurations

- Always remember that, no matter the efforts we put in our analysis:
  - *All forecasts are wrong, some forecasts are more wrong than others.*
  - *All forecasts are wrong, some forecasts are useful.*



Source: *Transport Models: TAG Unit 3.1.2.* Department for Transport, June 2005

What does transport planning aim at?

- Defining whether a system/service is actually useful
- Quantifying the possible number of users
- Dimensioning the service
- Choosing the transport system - examples:
  - by road: building a road: infrastructure, toll, maintenance
  - Public Transport by road: buses management: vehicles, personnel, maintenance
  - Public Transport by rail: infrastructure, fare, vehicles, management, maintenance
- Defining the route or lines
- Defining service levels
  - number of lanes
  - type
  - stops
  - timetable
  - frequency
  - number of vehicles

What does transport planning aim at?

- defining costs
  - building costs or initial investment
  - yearly management costs
  - recurrent modernization costs
- defining revenues
  - revenues from fares (tickets, car parks, …)
  - other revenues (advertising, …)
  - funding
  - contributions

citilabs
TRANSPORTATION MODELING SOFTWARE

Who benefits from transport planning?

- **Public institutions**
  - Municipalities
    - *Urban Traffic Plans/Mobility Plans*
    - *Analysis of traffic for special interventions, ex.*
    - *shopping centers, traffic restriction in historic centers, PT services, etc.*
    - *parking plans for historic centers*
    - *special services plans, ex. in touristic areas*
  - Provinces, regions
    - *Transport Regional Plan*
    - *Viability Regional Plan*
    - *Feasibility studies*
    - *Analysis of the transport demand*
  - Ministries
    - *Transport and Logistics General Plan*
    - *Road Safety National Plan*

Who benefits from transport planning?

- **Private or Public Operators**
  - infrastructures owners or managing companies (such as motorway concessionaires)
    - *feasibility studies*
    - *technical/economic analysis of a new infrastructure*
  - Public Transport operators
    - *Analysis of the transport demand*
    - *technical/economic analysis of a new service*
  - Tourist boards, …
    - *feasibility of dedicated transport services*
  - trading operators, tour operators, etc.
    - *technical/economic analysis of dedicated services*
    - *analysis of traffic required by institutions (ex. shopping centres, tourist parks, multiplex cinemas, etc.)*
    - *links with stations/airports and tourist destinations*

Who carries out transport planning?

- The institution who is interested directly
- Specialized consulting offices
- Specialized institutions
- Operators who are directly interested

Carrying out a transport system

- Planning
  - quantification of traffic flows on the network
  - analysis of alternative solutions
  - feasibility analysis
  - drawing up of outline plans
- Dimensioning
  - definition of project specifications
- Design
  - drawing up of the executive plan
- Contract, construction
  - carrying out and implementation of the system

The planning procedure

- Phase 1 - Analysis of the current situation
  - delimitation of the plan and study areas
  - zoning of the areas
  - analysis of the physical, geological and geotechnical environment of the plan area
  - survey of the regional planning
  - survey of the transport infrastructures and services offered
  - analysis of the socio-economic planning
  - estimate of the mobility demand for passengers and goods
    - ***importance of relations between places of origin and destinations***
    - ***importance of relations according to means of transport, reason, time span***
    - ***importance of traffic flows on transport networks, importance of critical elements***

The planning procedure

- Phase 2 - Definition of intervention scenarios
  - definition of the future tendency socio-economic planning
  - representation of the future transport demand with no interventions
  - simulation of the non-intervention planning and definition of critical elements
  - definition of objectives and restraints
  - definition of intervention strategies
  - design of project alternatives of intervention – intervention scenarios
  - estimate of implementation costs for intervention scenarios
  - estimate of the resources available

The planning procedure - **Scenarios**

- **Current scenario**
  - current planning of the transport-territory system (T-T)
  - load conditions of networks
- **Non-intervention scenario (Zero Scenario)**
  - Future planning of the T-T system with no interventions
  - effect of the tendential evolution of: territory, activity ® demand
  - effect of interventions on the system
- **Intervention Scenario**
  - future planning of the T-T system after plan alternatives
  - prescriptive and organizational modifications, new infrastructures

The planning procedure

- Phase 3 - Simulation of interventions and evaluation of their effects
  - simulation of the several intervention alternatives
  - interaction demand-offer
  - functional evaluation of effects on modal networks
  - economic-financial evaluation – analysis of costs and benefits
  - evaluation of the effects on environment and consumption of resources
  - evaluation of the effects on business and employment
  - evaluation of the effects on safety
  - comparison between project alternatives
  - definition of priorities

The planning procedure

- Phase 4 – Carrying out of interventions
  - Prescriptive level
    - *fulfillment of administrative and decisional procedures, comparison of interested parts*
    - *approval by executive branches: Ministries, Council of Ministers, local councils*
    - *approval by legislative branches: Parliament, Regional, Provincial, Municipal Councils*
  - Infrastructural level
    - *final project*
    - *impact evaluations*
    - *executive project*
    - *contract, site setting up, construction, tests*
  - Organizational level
    - *definition of prices and fares*
    - *definition of maintenance plans*
    - *implementation of user information services*
    - *…*

The planning procedure

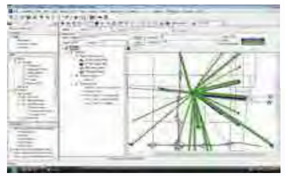
- Phase 5 – Survey of effects
  - Survey of intervention effectiveness
  - Improvement of the plan process
    - *recalibration and validation of simulation models*
    - *comparison between expectations and real changes of the system*
    - *evaluation of the decision-making process*
  - Characteristics of the environment external to the transport system
  - Characteristics of the transport offer
  - Characteristics of the mobility demand
  - Quality of the balance demand/offer

The planning procedure

- Phase 6 – Definition of corrective interventions
  - evaluation of possible problems during the decision-making process
  - evaluation of social agreement and/or disagreement phenomena
  - involvement of the parts in the decision-making process
  - readjustment of contrasting needs of planners, decision makers, collectivity
  - modification of elements in the planning iterative process
  - redefinition of objectives, restraints, type of intervention
  - redefinition of interventions

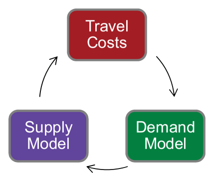
Basic concepts in model development

- We define a "model" as a set of data and processes that represent a real-world system and describe its behavior under alternative circumstances.
- This is distinct from the software used to implement a model, e.g. Cube Voyager.
- Models are applied to analysis of future conditions to create *forecasts*.
- When applying a model it is always important to use common sense and exercise appropriate judgment in interpreting results.



General Framework of Transport Modelling

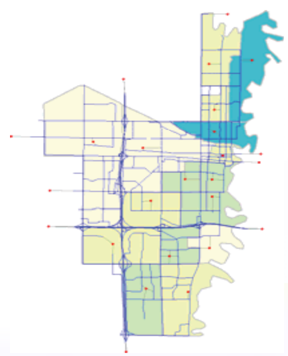
- Roots in transportation economics and theory of supply / demand equilibrium.
- Transportation supply = roads, bridges, trains, buses, airports, boats, planes, …
- Transportation demand is derived from demand for activities (goods & services).
- Transportation costs = expected time, distance, money, and other penalties associated with using some travel option
  - The combination of these costs is called "Generalised Cost"
- Costs increase with demand – congestion.
- Demand decreases with cost – diversion.



Source: *Transport Models: TAG Unit 3.1.2.* Department for Transport, June 2005

Transportation Model Databases

- Transport Analysis Zones (TAZ)
  - Study area definition
  - Internal zonal data (socio-economic factors)
  - External stations
- Transport networks
  - Directed graph (node-link connectivity)
  - Centroids and connectors
  - Junctions / intersections
  - Public transport lines and stops
- *Skimming* is the process of tracing (summing) link values across the network along the minimum cost path between each zone pair (origin-destination)



Components of the system

- Offer system
  - infrastructural components
  - road networks, railway networks, airline networks, sea networks, LPT networks
  - car parks, road junctions, stations and calls, branches
  - interchanges, intermodal terminals, freight terminals, airport hubs, port centres
- organizational components
  - road traffic, railway, airline and sea regulations
  - - systems to control traffic and safety
  - - routes, lines, timetable and fares of collective transport
  - - user information systems
  - - emergency management systems

Components of the system

- Demand system: all users who use the service offered by a transport system over a prearranged time span and with specific characteristics
- Traffic system: traffic flows travelling on every element of the offer system derived from the interactions between demand and offer

Representation of the offer

Representation of the offer

- Physical components
  - network structure/Transportation Supply
- Functional components
  - cost functions

transport network

⇓

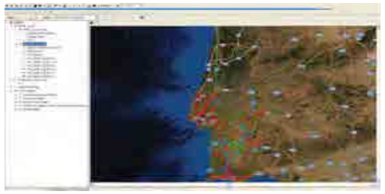
offer model

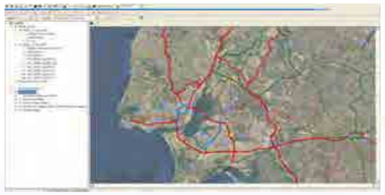
⇓

all mathematical relations that correlate transport costs and flows of the links of a network

Transportation Supply

- Graph of the network
  - graphical scheme associated to a database
- Components of the graph
  - links
    - linear elements of the graph that reproduce the basic sections of real transport infrastructures
  - Nodes
    - points on the graph that reproduce the intersections and terminals definition of node: null balance of flows



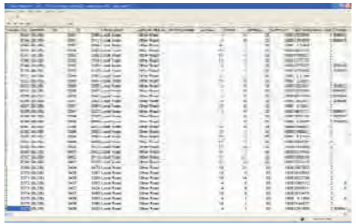


Transportation Supply

- **Centroids**
  - fictitious nodes representing OD zones
  - Centroid: balance of flows ≠ 0
  - Aim: correspondence between graph and structure of the OD matrix
  - internal centroid: centroid of a zone inside the study area
  - external centroid: centroid of a zone outside the study area
- **Connecting links**
  - fictitious elements that connect Centroids to physical nodes
  - they do not have any physical or functional characteristics

Transportation Supply

- Information associated to a graph
  - physical (or geometrical) characteristics of links
    - Length, section, number of lanes
    - …
  - functional characteristics of links
    - hierarchical level, administrative competence, direction, no entries
    - Speed, Capacity, traffic Flow curve
  - Physical characteristics of nodes
    - type of intersection: right of way, traffic light, roundabout,…
    - number of branches, number of lanes, no turning or no left turns…
  - Functional characteristics of nodes
    - speed of lanes, capacity of lanes, traffic lights timing, green percentage
  - ***A full, extensive Geodatabase***



Transportation Supply

- Public transport
  - Information that describe the PT system:
    - Services or data about PT Lines
    - Line Paths description
    - Transit Stops description
  - Frequency of collective transport services
  - Capacity of collective means of transport
  - "Not Transit" Links ("NT Legs"), with information for each PT mode regarding:
    - Enter a Public Transport System
    - Quit a Public Transport System
    - Change from a Public Transport System to another one.
  - ***A full, extensive Geodatabase***

Elements of the offer

- Transport Cost/General cost
  - Cost to cover a distance or section
  - Made of several components that represent the uselessness associated to the covering of a section
- Public transport
  - pedestrian access and leaving time to and from stops
  - waiting time at stops
  - journey time on vehicles
  - cost of fare (Fares modeling)
  - Lack of comfort due to crowding (Crowd modeling)

Elements of the offer

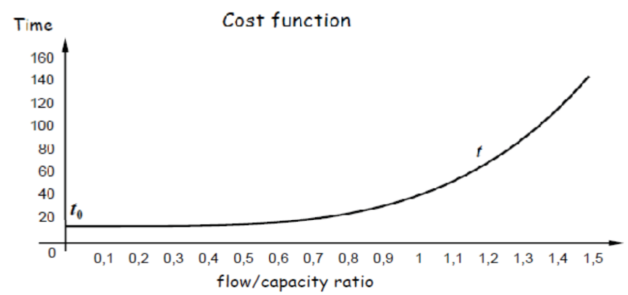
- Individual transport
  - monetary cost of fuel
  - cost of motorway tolls
  - cost of paid parking areas
  - journey time
  - time to find a parking space
  - time spent in a queue
  - risk of an accident
- Cost functions
- Discharge curves

Elements of the offer

- **Cost function**
  - cost c charged to the user when travelling on a certain link l
  - it can be a function of the flow f that travels on this link and possibly of the flow travelling on adjacent branches
- $c_l (f) = c_{ol} + c_{vl} (f)$
  - f = vector of flows
  - col = fixed cost independent of flow, ex: toll
  - cvl (f) = variable cost due to congestion
- **Transport (simulated) network**: graph whose elements are associated to a cost function or another quantitative characteristic

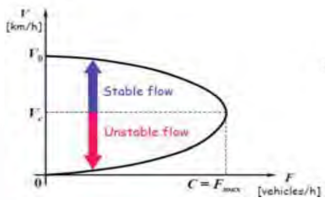
Elements of the offer

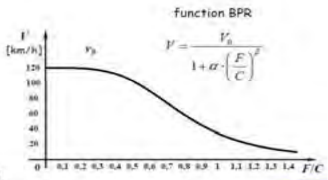
- Cost function: connection between
  - flow or flow/capacity
  - time, speed (inverse cost) or other components



Elements of the offer

- representation of congestion





The Cost Function

- Cost of the link

$$C_l = \beta_1 \times t_l + \beta_2 \times c_{ml}$$

- C_l = general transport cost for the link l
- t_l = journey time on the link l
- c_{ml} = monetary cost of the link l
- β_1 and β_2 = coefficients of reciprocal substitution (Homogenization): reduction of the cost to only one scaled quantity
- **Cost of the route**

$$C_k = \Sigma C_l$$

Cost functions - Homogenization

- Cost charged to a user: quantities x_i *that are not homogeneous*
- reduction of the cost to only one scaled quantity
- homogenization by means of dimensional coefficients
- $C_i = \Sigma_i (\beta_i \cdot x_i)$
- dimension of coefficients: inverse with respect to the dimension of variables

- **VOT: Value of Time**
  - $C_{general} = t + c_m$
  - $C_g = \beta_1 \times t + \beta_2 \times c_m$
  - $\beta_1 = [h_{-1}]$ and $\beta_2 = [€_{-1}]$
  - $C_g = VOT \times t + \beta_2 \times c_m$

Reason of the journey	VOT 2004 [€/h]
Business	27,10
Home-job	6,13
Other reasons	4,09
Goods with light vehicle	40,89
Goods with heavy vehicle	43,85

Cost functions - Highway

$$C = \beta_1 \cdot t_r + \beta_2 \cdot t_{nf} + \beta_3 \cdot c_m$$

- t_r = journey time
- t_{nf} = waiting time at the final node
- c_m = monetary cost

$$C = \beta_{11} \cdot t_p$$

- Journey of a pedestrian
- t_p = journey time

Cost functions - Highway

- Traffic flow curves: they represents a connection between
  - flow or flow/capacity
  - clearing average speed
- Typical curves: BPR

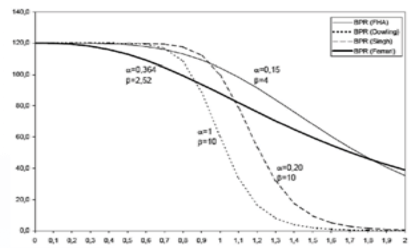
$$V = \frac{V_0}{1 + \alpha \left(\frac{F}{C} \right)^{\beta}}$$

Road type	α	β	Source
Motorways	0,016	4	Federal Highway Administration
	1	10	Dowling e Skabardonis, 1992
	0,364	2,52	Ferrari, 1992
	0,2	10	Singh, 1995
Extra-urban roads	2,62	5	Steenbrink, 1974
	0,747	4	Akcelik, 1978
Urban expressways	0,05	10	Skabardonis e Dowling, 1997
	1	7,5011	C.I.Su.T. - S.T.A., 1999
Residential urban roads	1	2,80 5	C.I.Su.T. - S.T.A., 1999
Local urban roads	1	2 - 3	C.I.Su.T. - S.T.A., 1999

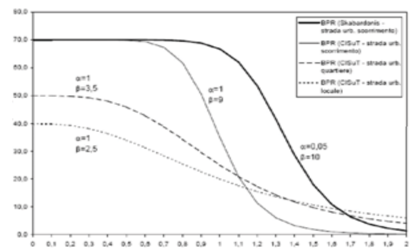
(*) values calibrated for motorway branches having a capacity C = 4000 vehic/h

Cost functions - Highway

Motorways



Urban Roads



Cost functions – Public Transport

$$C = \beta_4 \cdot t_{acc/leav} + \beta_5 \cdot t_s + \beta_6 \cdot t_w + \beta_7 \cdot t_b + \beta_8 \cdot t_t + \beta_9 \cdot t_d + \beta_{10} \cdot c_m$$

- $t_{acc/leav}$ = access/leaving time
- t_s = getting on time
- t_w = waiting time at a stop
- t_b = journey time on board
- t_t = transfer time
- t_d = getting off time
- c_m = monetary cost

Cost functions – Public Transport

$$C = \beta_4 \cdot t_{acc/leav} + \beta_5 \cdot t_s + \beta_6 \cdot t_w + \beta_7 \cdot t_b + \beta_8 \cdot t_t + \beta_9 \cdot t_d + \beta_{10} \cdot c_m$$

- Access/leaving time: time necessary for the user to:
  - reach the stop to get on from the origin of his/her journey
  - reach the destination of his/her journey from the stop where he/she gets off

$$t_{acc/leav} = \frac{L_{acc/leav}}{V_{c,acc/leav}}$$

- $L_{acc/leav}$ = access/leaving distance
- $V_{c,acc/leav}$ = commercial speed of the means of transport chosen by a user along access/leaving branches

Cost functions – Public Transport

$$C = \beta_4 \cdot t_{acc/leav} + \beta_5 \cdot t_s + \beta_6 \cdot t_w + \beta_7 \cdot t_b + \beta_8 \cdot t_t + \beta_9 \cdot t_d + \beta_{10} \cdot c_m$$

- Getting off/on time: time necessary for the user at a stop to get on/off a vehicle

$$C = \beta_4 \cdot t_{acc/leav} + \beta_5 \cdot t_s + \beta_6 \cdot t_w + \beta_7 \cdot t_b + \beta_8 \cdot t_t + \beta_9 \cdot t_d + \beta_{10} \cdot c_m$$

- Waiting Time: time necessary for the user at a stop waiting for a service that is from the instant he/she arrives until the instant when he/she starts getting on

Cost functions – Public Transport

$$C = \beta_4 \cdot t_{acc/leav} + \beta_5 \cdot t_s + \beta_6 \cdot t_w + \beta_7 \cdot t_b + \beta_8 \cdot t_t + \beta_9 \cdot t_d + \beta_{10} \cdot c_m$$

- Journey time or time on board: time spent by the user during the journey on board from the end of the getting on time to the beginning of the getting off time

$$t_t = \frac{L}{v_m}$$

- *L = distance run on board*
- *v_m = commercial speed of the transport system*

Cost functions – Public Transport

$$C = \beta_4 \cdot t_{acc/leav} + \beta_5 \cdot t_s + \beta_6 \cdot t_w + \beta_7 \cdot t_b + \beta_8 \cdot t_t + \beta_9 \cdot t_d + \beta_{10} \cdot c_m$$

- Monetary Cost: price paid by the user to buy the right to travel, for example: fare, pass, etc.

Representation of the Demand

Representation of the Transport demand

- System of activities – functions distributed over the territory

- Residence
  - quantity of people on a given portion of territory
  - type and density of houses
- Settlements
  - type and dimensions of settlements of production or tertiary
  - working activities (factories, shops, offices)
  - number of local units
  - number of employees
- Services (commercial, financial, social…)
  - type and number of shops, bank branches, surgeries, etc.
  - hospital beds

Representation of the Transport demand

- Education
  - type and dimensions of schools of several type and level
  - number and dimensions of universities
- Entertainment
  - type and dimensions of gyms, cinemas, etc.
- Tourism
  - type and dimensions of tourist attractions, etc.

- Related disciplines
  - town planning
  - science of territory
  - statistics
  - economy

Representation of the Transport demand

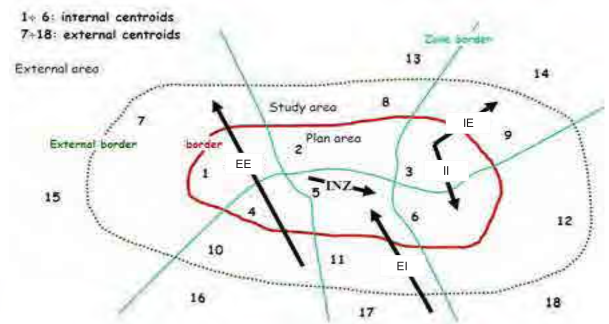
- Characterization of the demand
  - Origin and Destination of the journey
  - Category of user that makes the journey (User Classes):
    - *Worker*
    - *Student*
    - *Tourist.*
  - Reason of the journey
  - Topicality of the journey
    - *current: it takes place at the same time as the analysis of the demand*
    - *potential: it supposedly takes place under certain conditions of the offer to simulate and assess*
    - *future: it takes place at a future time reference*

Representation of the Transport demand

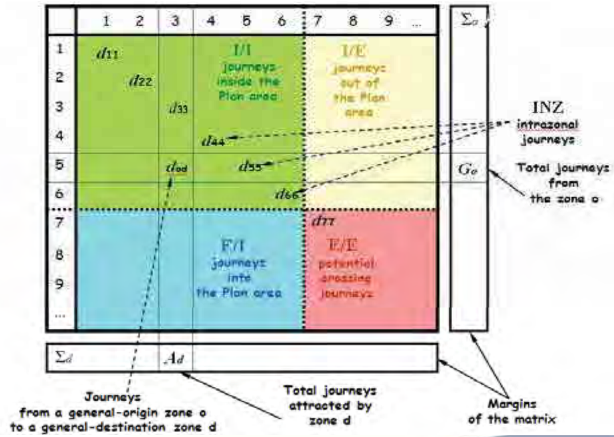
- Characterization of the demand
  - Time or time band of the journey
    - *morning rush hour, evening rush hour*
    - *afternoon moderate flow hour, …*
  - Recurrence of the journey
    - *commuting, irregular, occasional*
  - Frequency of the journey
    - *several times a day*
    - *every day of the week, every working day*
    - *sometimes in a month, ...*
  - Type of transport used for the journey
    - *collective transport*
    - *individual transport*
  - Vehicles used for each type
    - *tram, underground, train*
    - *car, motorbike, bicycle*
  - Route followed

Origin-Destination (OD) Matrix

- Zoning of the area



The structure of a OD Matrix



Components and characteristics of the OD matrix

- structure of the matrix
  - OD zones: structuring of the territory and of the data of the analysis
- sectors of the matrix
  - progressive numbering of zones inside/outside
  - journeys inside the area of intervention (II)
  - outwards journeys (IE)
  - inwards journeys (EI)
  - crossing journeys (EE)
  - intrazonal journeys – main diagonal ($zone_I = zone_J$)
- margins of the matrix
  - total per line and column (*Trip Ends*)
  - total journeys generated from the zones of origin (*Productions*)
  - total journeys attracted by the zones of destination (*Attractions*)

Path building and Assignment

cube

Path building / Assignment

- A typical assignment process:
  - builds paths based upon link costs (impedances)
  - assigns trips to those paths for each origin zone.
- After all origin zones have been processed, link costs are updated based upon the level of congestion on each link.
- The entire path and assignment process is repeated until some criteria for termination is reached. Different criteria are used to determine when enough iterations have been performed.
- The volumes from each iteration are combined to form a weighted assignment.
- Almost all of the operations follow a fixed pattern, and are driven by basic parameters. Various options are usually available to provide the user with additional outputs.
- *Specific courses and literature available…*

citilabs

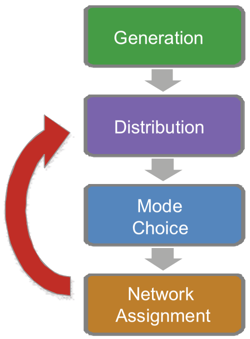
cube

The Four-Step Modelling Process

citilabs

The Four-Step Modelling Process

- One (extremely common) method of forecasting travel demand.
- Trip ends (productions and attractions) are generated based upon socio-economic and demographic factors.
- These are distributed between zones based upon aggregate travel costs.
- Logit models are used to split person trips between different travel modes.
- Trips by mode are factored by time of day and assigned to specific network paths.
- Modern versions of this process feedback costs from assignment to earlier steps.

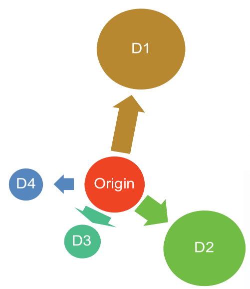


Trip Generation

- Answers the question "how many trips are produced by and attracted to each zone".
- Productions and attractions are a *function* of socio-economic attributes of the zone, such as households and employment.
- Categories of trips:
  - Home-based-work (commute)
  - Home-based-other (e.g. social/recreational)
  - Non-home-based (e.g. mid-day errands)
- Typical model forms:
  - Cross-classification (average trip rates by household and employment category)
  - Linear regression (least-squares estimation)
  - Ordered logit (discrete choice)

Trip Distribution

- Answers the question "where do trips produced by zones go, and where do trips attracted by zones come from?"
- Function of activity concentrations and zone-to-zone travel costs.
- Process calibrated to match an observed trip length distribution (e.g. from surveys).
- Typical forms:
  - FRATAR – iterative proportional fitting (IPF)
  - Gravity model – IPF with friction factors
  - Destination choice – random utility model (logit)



Mode Split

- Answers the question, "how do trips get from productions to attractions, given the available set of network options?"
- Possible modes: public transit, personal vehicle, non-motorized transport, as well as detailed path/vehicle type options.
- The probability of selecting a given mode is a function of the relationship between its cost and the cost of competing modes.
- Typical forms:
  - Multinomial logit (random utility model)
  - Nested or hierarchical logit
  - Incremental logit (based on changes in cost)

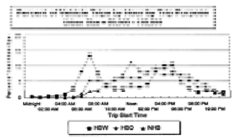
$$P_n(i) = \frac{e^{V_{in}}}{\sum_{j \in C_n} e^{V_{jn}}}$$

Time-of-Day Factoring

- Answers the question "when do trips between origins and destinations occur?"
- Trip Generation, Distribution, and Mode Split create square trip tables in production-attraction format, where each cell represents both outbound and return.
- To translate into origin-destination format, the trip tables must be transposed, added together, and divided by two.
- Time-of-day factors are simultaneously applied, representing probability that the outbound or return portion of a trip occurs during the time period to be analyzed.

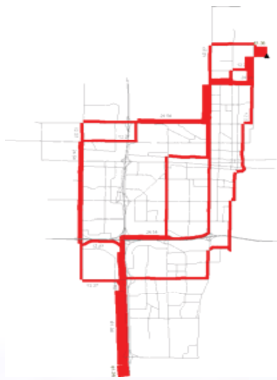
Time Segment N

	D_1	D_i	D_z
O_1	$T_{1,1}$	$T_{1,i}$	$T_{1,z}$
O_i	$T_{i,1}$	$T_{i,i}$	$T_{i,z}$
O_z	$T_{z,1}$	$T_{z,i}$	$T_{z,z}$



Highway and Transit Assignment

- Answers the question "what specific routes or links are used by trips, and at what level of intensity?"
- Function of interaction between travel demand and transportation supply including congestion and crowding effects.
- Equilibrium: all used paths have equal and minimum travel cost; no trip can unilaterally divert without increasing cost.
- Typical forms:
  - All-or-nothing (shortest paths only)
  - User equilibrium (convex combinations)
  - Route choice (transit multi-path analysis)



Activity Based Models

Limitations of Trip Based Models

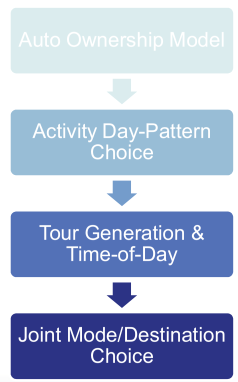
- With Person-trips as the unit of analysis:
  - No interactions between trips made in the same trip chain
  - No interactions between trip chains made during the same day
  - No interactions between the trips made by people in the same household
- Spatial aggregation of Trips:
  - Trip origins and destinations modeled as if they are located at the same point in space
- Demographic aggregation:
  - All households within a given zone are treated as identical or segmented along a few dimensions
- Temporal aggregation:
  - Only a few periods of the day are considered
  - Proportion of trips made in each period treated as constant

Activity Based Models

- Early recognition that travel is a derived demand
  - derived from a person's desire to engage in activities that are spatially separated
  - Focus of the model should be on the underlying behavior: What people want to do, not where people want to go
- Early attempts at implementing tour based models
  - San Francisco Bay Area, The Netherlands, Boise Idaho, Stockholm, New Hampshire, Italy
- Current implementations of activity-based travel demand model systems
  - Portland OR, San Francisco County CA, New York City, Columbus OH, Atlanta, San Francisco Bay Area (MTC)

Activity and Tour Based Modeling

- Alternative to four-step modeling approach popular in the academic transportation research community and becoming more common in practice (although still less than 4SM)
- Disaggregate simulation using synthetic populations based upon micro-data
- Complete tours, or chains of trips, are analyzed, rather than individual trips
  - e.g. Home > Work > Shop > Home
- Activity location and scheduling models
- Mode choice applies to entire tour
- Ideally suited for dynamic traffic assignment and meso-simulation

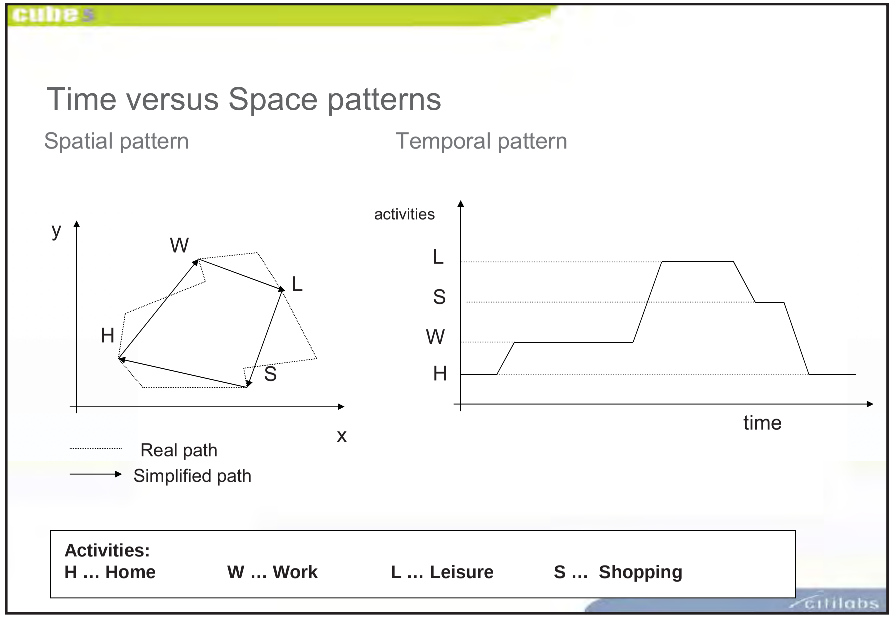


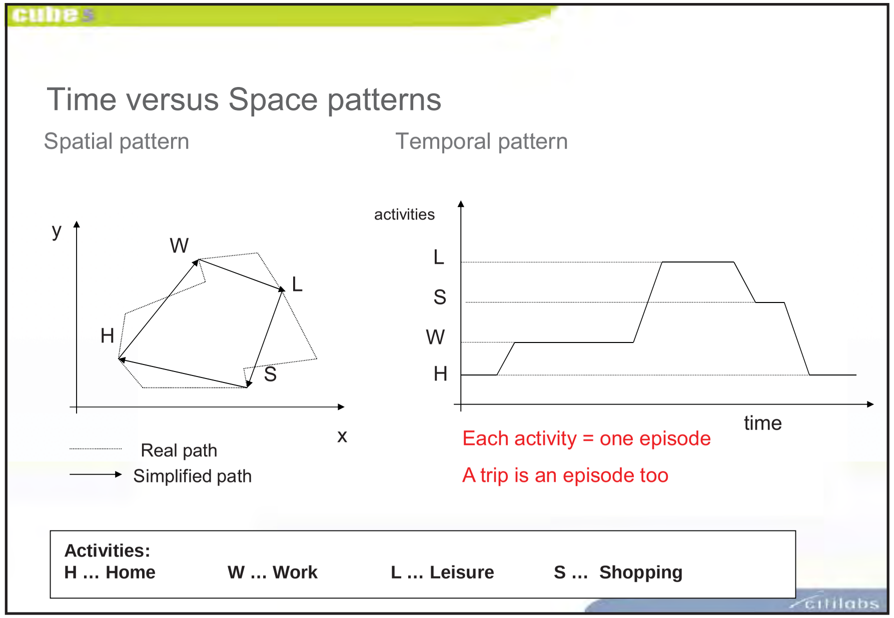
Activity-Based Approach(es)

- Activity-Based Approach
  - Think and model activities first (the motivation)
  - Consider interactions among activities and agents (people)
  - Derive travel as a result of activity participation (derived demand)
  - Consider linkages among activities and trips (interactions)
- Demand for activities <-> time allocation
  - By definition a dynamic relationship with feedbacks
  - Most approaches imply thinking in terms of temporal hierarchies

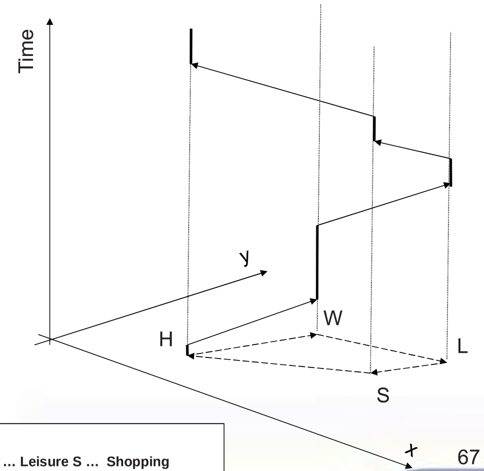
Activity Patterns (Schedule)

- A sequence of activities, or a schedule, defines a path in space and time
- What defines a person's activity pattern?
  - Total amount of time outside home
  - Number of trips per day and their type
  - Allocation of trips to tours
  - Allocation of tours to particular HH members
  - Departure time from home
  - Arrival time at home in the evening
  - Activity duration
  - Activity location
  - Mode of transportation
  - Travel party
  - What else?


cube
Time versus Space patterns
Spatial pattern
Temporal pattern
y
x
W
L
H
S
Real path
Simplified path
activities
L
S
W
H
time
Activities:
H ... Home
W ... Work
L ... Leisure
S ... Shopping
citilabs


cube
Time versus Space patterns
Spatial pattern
Temporal pattern
y
x
W
L
H
S
Real path
Simplified path
activities
L
S
W
H
time
Each activity = one episode
A trip is an episode too
Activities:
H ... Home
W ... Work
L ... Leisure
S ... Shopping
citilabs

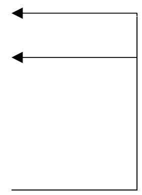
Activities in Time and Space



Activities:
H ... Home W ... Work L ... Leisure S ... Shopping

67

A typical Model Structure



1. Population synthesizer
2. Zonal accessibility measures
3. Activity and travel simulator
4. Travel aggregator
5. Traffic assignment
6. Feedback loop / equilibration

Conclusions

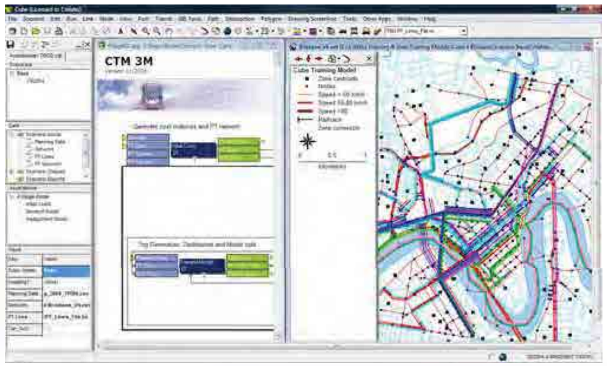
The typical model development workflow

1. Comprehend the model purpose
2. Gather and code data
   - Transportation networks: roadway centrelines, intersection definitions, public transit routes
   - Define TAZ boundaries and summarize demographic and economic factors (households and employment) by zone
3. Define model functions
   - Obtain travel behavior inventory or survey data
   - Specify, estimate and calibrate mathematical relationships using statistical tools & methods
4. Link data and processes with interface
   - Lay out application groups and programs
   - Sequence steps and link common data files
   - Create catalog; run, review, revise as needed

Introduction to Cube

Cube Base – model application and data

Overview of Cube Base



- Catalog area
  - Scenarios
  - Data
  - Applications
- Flow chart (apps)
  - The model
- Data window
  - Network/PT Lines
  - Reports
  - Databases
- Keys
  - Scenario specific inputs

Network window based on Cube4 style

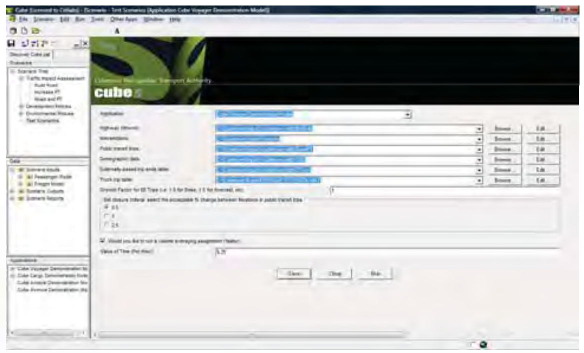
citilabs
TRANSPORTATION MODELING SOFTWARE

Overview of Cube Base

- Catalog area
  - Scenarios
  - Data
  - Applications
- Flow chart (apps)
  - The model
- Data window
  - Network/PT Lines
  - Reports
  - Databases
- Keys
  - Scenario specific inputs

Network window based on CubeGIS

Overview of Cube Base

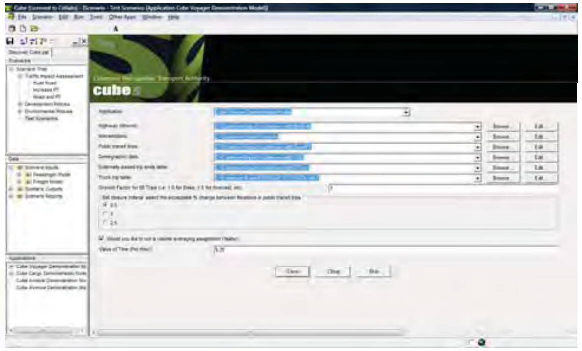


- Scenario edit menu
  - Menu for scenario inputs
  - Data files and tables
  - Parameters
  - User choices

Overview of Scenario Editing and Network Development Steps

- 1 ; in catalog area, 'Scenarios', right click on 'Base' scenario and choose 'Add Child'
- 2 ; name the new scenario, e.g. 'Year 2004' and put in scenario code, e.g. '04' and description, then OK
- 3 ; open input network from the 'Edit' button and choose to copy parent network/open as appropriate
- 4 ; check and change the 'Edit' … 'Options' settings (from Cube menu …)
- 5 ; check the 'View' … 'Layer Information' settings (from Cube menu …)
- 6 ; create workspace (data layers, colours, styles, legends etc.)
- 7 ; network editing; changing data for individual links/nodes
- 8 ; network editing; using link computations, update mode, using polygons
- 9 ; network editing; adding links, changing links, moving nodes, using copy and paste
- 10 ; intersections; quick look at the intersection editing
- 11 ; PT; quick look at PT editing

Scenario development steps

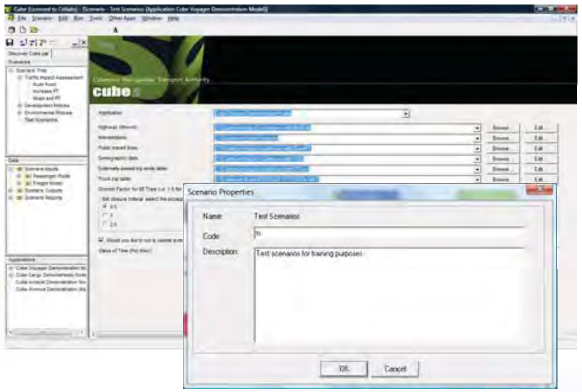


Sc STEP 1

- In catalog area, 'Scenarios', right click on 'Scenario Tree' and choose 'Add Child'

Training – Cube Base v5

Scenario development steps

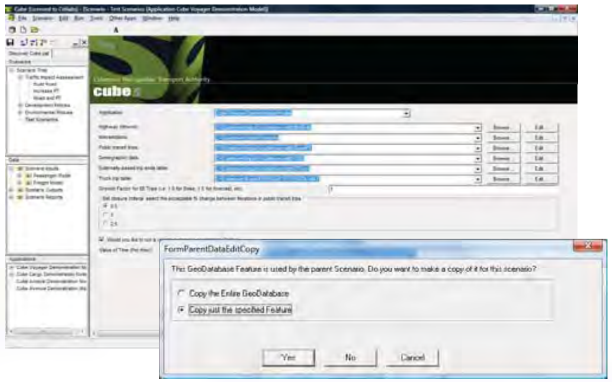


Sc STEP 2

- Name the new scenario, e.g. 'Test Scenarios'
- Put in scenario code, e.g. 'ts' and description
- Click OK

Training – Cube Base v5

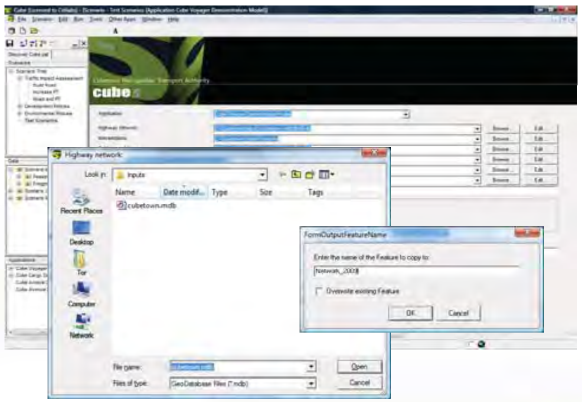
Scenario development steps



Sc STEP 3

- Open input network from the 'Edit' button and choose from copy options

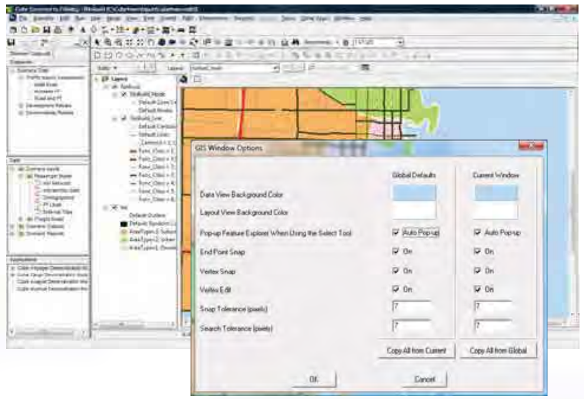
Scenario development steps



Sc STEP 3

- Open input network from the ‘Edit’ button and choose to copy parent network
- Give the new network a new name, e.g. Brisbane2008
- If the network name already differs from parent, this will just open

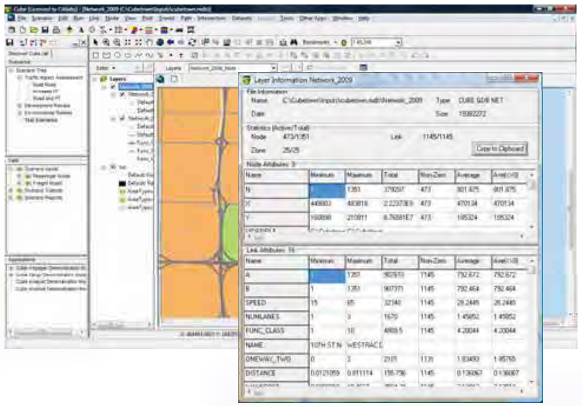
Scenario development steps



Sc STEP 4

- Check and change the ‘Edit’ … ‘Options’ settings from menu
- Click ‘Close’ button to accept changes

Scenario development steps

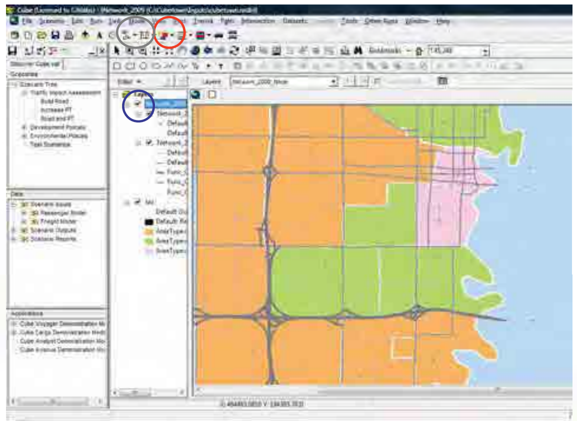


Sc STEP 5

- Check the 'View' … 'Layer Information' settings from menu

Training – Cube Base v5

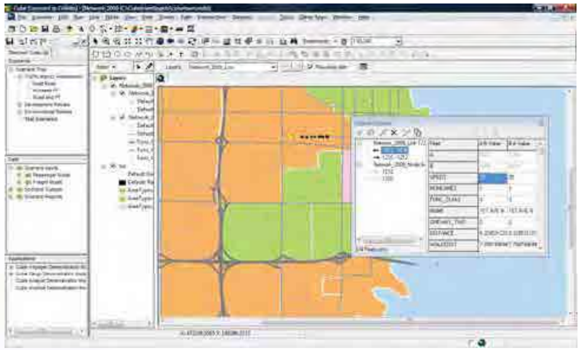
Scenario development steps



Sc STEP 6

- Create workspace:
- Data layers by right clicking on the 'TOC' frame (blue circle)
- Colours, styles, legends etc. by clicking on the 'Link/Line Color' button (red circle)
- Text annotation by clicking on 'Post Link' and 'Post Node' buttons (grey circle)

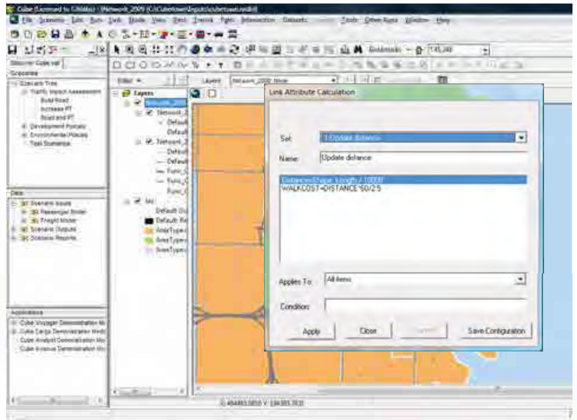
Scenario development steps



Sc STEP 7

- Network editing ...
  ... Editor - start editing
  ... Layers - choose layer to edit
  ... With editing tool - click on link or node
- Yellow diamond indicates A-node
- Black circle indicates B-node
- Small black diamonds indicates vertex points

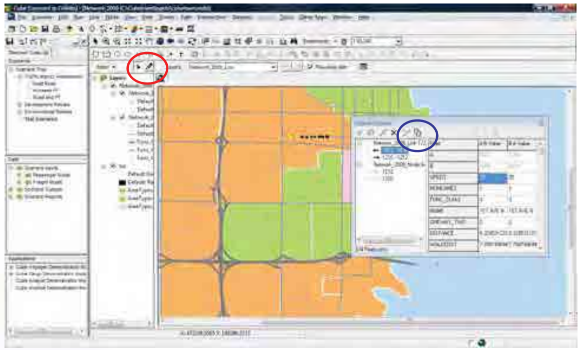
Scenario development steps



Sc STEP 8

- Network editing :
- Using link computations from 'Link' ... 'Compute...' menu
- Put in name and right click in middle area to 'insert' ... 'equation'
- Right click to define equation
- Used for single or selected items, items selected by graphics

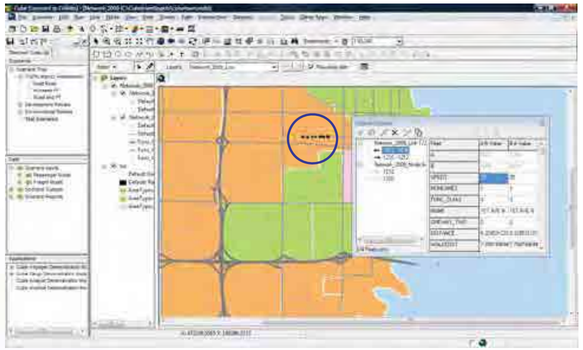
Scenario development steps



Sc STEP 9

- Network editing :
- Add links from 'Link' … 'Add …' menu
- Add links by clicking on a link and then use copy button in feature explorer (blue circle) and then the create tool (red circle)

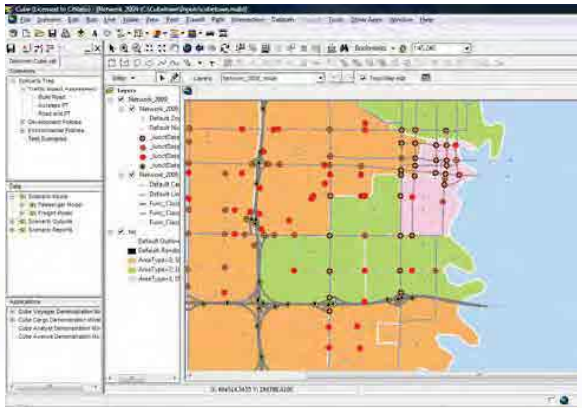
Scenario development steps



Sc STEP 9

- Network editing :
- Changing links by clicking on link and then move one of the nodes (yellow diamond or black circle)
- Moving nodes by clicking on node and move

Scenario development steps



Sc STEP 10

- Intersection editing :
- Read in intersection data from 'Intersections' menu
- Post intersection symbols from 'Node/Color' button (must have been set up)

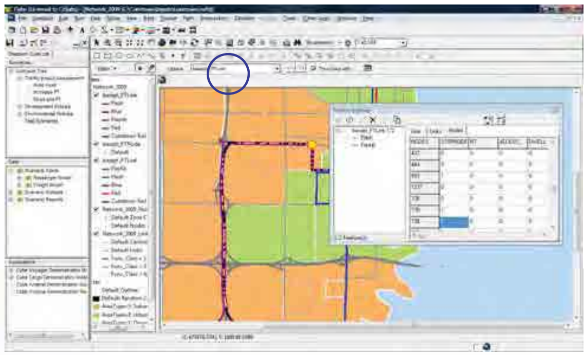
Training – Cube Base v5

Scenario development steps

Sc STEP 10

- Intersection editing :
- Click on 'Show input intersection data' button (blue circle) to open intersection editor
- Move through the various fields to see and set/edit data

Scenario development steps

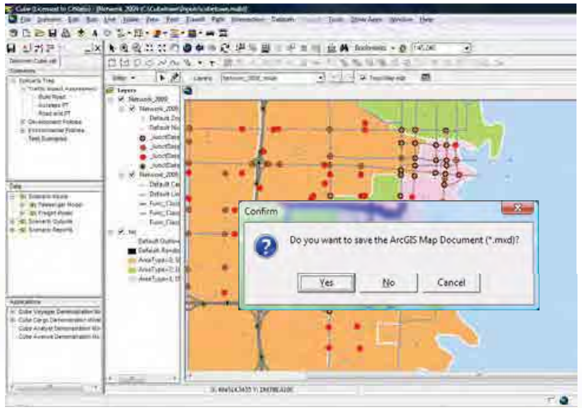


Sc STEP 11

- Quick look at PT editing:
- Open PT data from either the data section of the catalog or by adding PT data from geodatabase (right click on TOC)
- Add PT System data from the 'Transit Line Manager' button (blue circle)

cube Training – Cube Base v5

Scenario development steps



Sc STEP 12

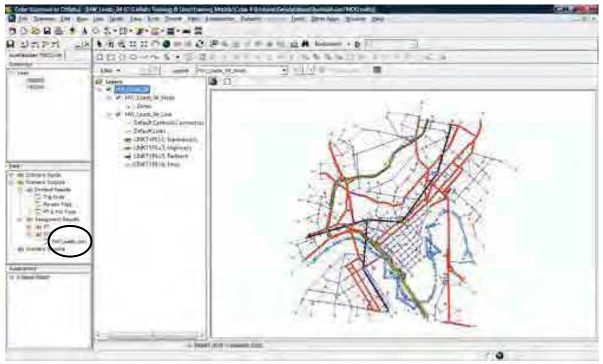
- Close network and save workspace as MXD
- The various layers will get display settings saved in VPR file with same name as geodatabase

citilabs

Overview of Results Display and Comparison Steps

- 1 ; in catalog area, 'Data', open the various output files by clicking on the '+' and then the data files, open the assigned highway network first, then the demand matrix
- 2 ; matrix view, look at matrix menu
- 3 ; goto network window, then add data to map with a right click in the TOC (table of contents)
- 4 ; link matrix to network from the 'Node' menu
- 5 ; in the network window, post link information, bandwidths etc. using the 'Post' menu and/or buttons
- 6 ; perform select link analysis from 'Path' … 'Use Path File' menu
- 7 ; display matrix charts and desire lines
- 8 ; in catalog area, 'Scenario Reports', create reports by right clicking on 'Scenario Reports' and choose 'Create Report'

Results Display and Comparison Steps

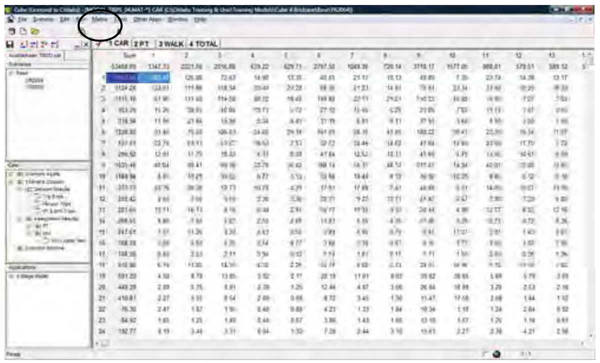


Screenshot based on Cube5 style

RC STEP 1

- Open the output network from the data pane on the left
- You find it under 'Scenario Outputs', 'Assignment results' under 'HW'
- Open the matrix from the 'Demand Results' section

Results Display and Comparison Steps

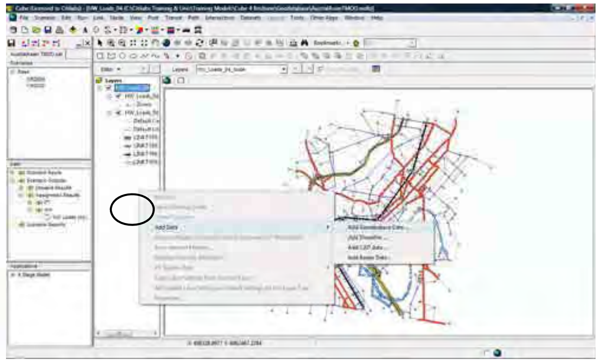


RC STEP 2

- Study the various options in the matrix view
- Use the 'Matrix' menu

Training – Cube Base v5

Results Display and Comparison Steps



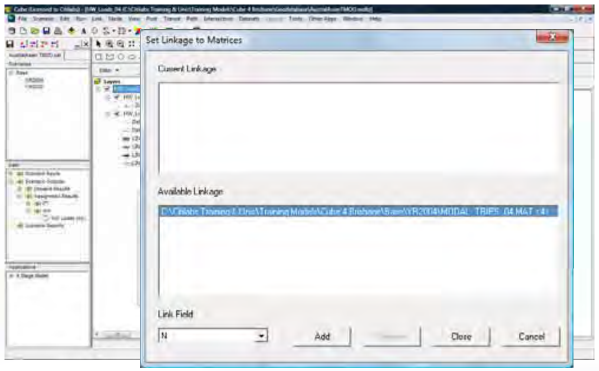
RC STEP 3

- Add more data to your map by right clicking in the TOC area as indicated
- Add from the Geodatabase
- Select the river data

Screenshot based on Cube5 style

citilabs

Results Display and Comparison Steps

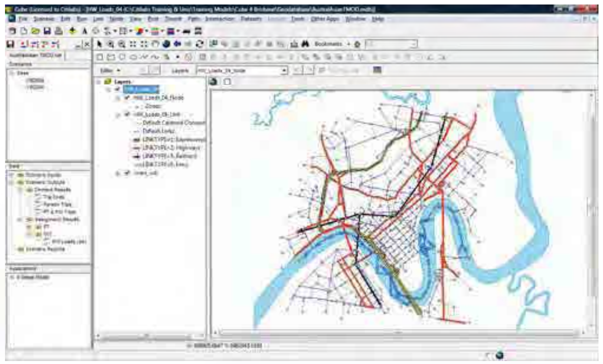


RC STEP 4

- Link matrix to network from the ‘Node’ menu
- Add the matrix to the network and close the dialog

Screenshot based on Cube5 style

Results Display and Comparison Steps



RC STEP 5

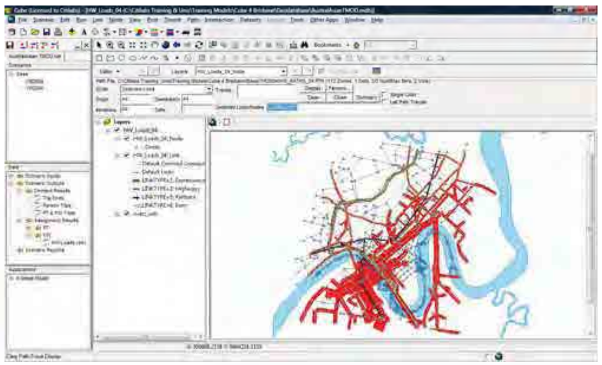
- In the network window, post link information, bandwidths etc. using the ‘Post’ menu and/or buttons

Screenshot based on Cube5 style

citilabs
TRANSPORTATION MODELING SOFTWARE

Training – Cube Base v5

Results Display and Comparison Steps



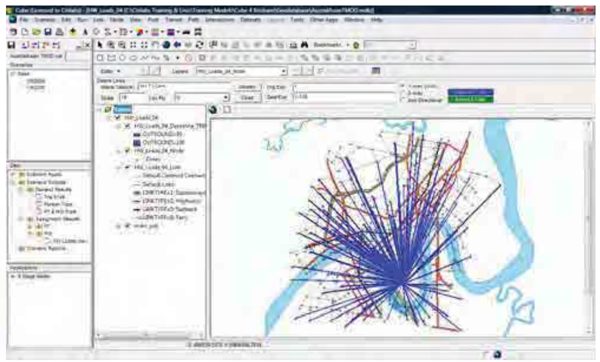
RC STEP 6

- Perform select link analysis from ‘Path’ … ‘Use Path File’ menu
- Follow instructions on screen

Screenshot based on Cube5 style

Training – Cube Base v5

Results Display and Comparison Steps

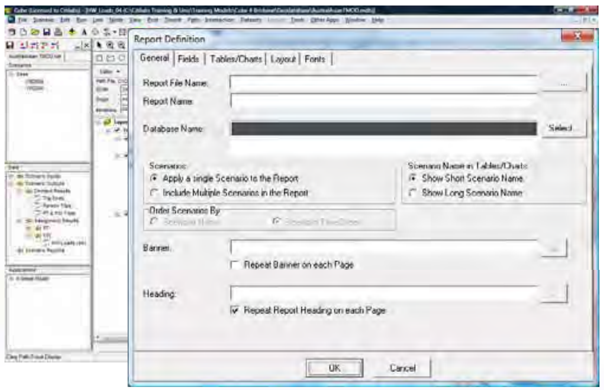


RC STEP 7

- Display matrix charts and desire lines
- Chart display require adding of node variables for tripend data
- All done from the ‘Node’ menu

Screenshot based on Cube5 style

Results Display and Comparison Steps



RC STEP 8

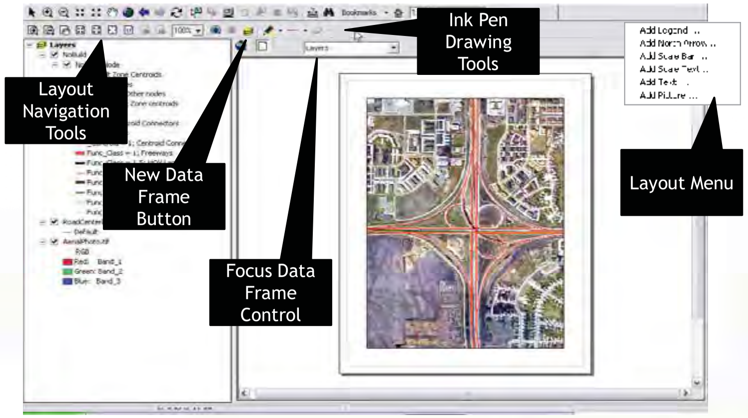
- In catalog area, 'Scenario Reports', create reports by right clicking on 'Scenario Reports' and choose 'Create Report'
- Go through the tabs in the dialog to create your report

Screenshot based on Cube5 style

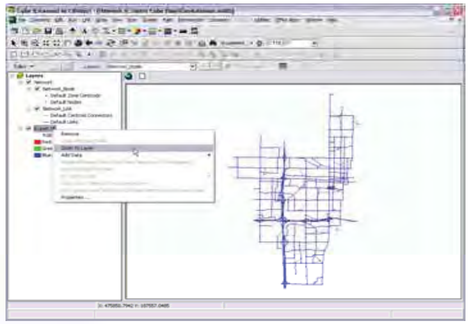
Overview of Mapping & Reviewing Data

- 1; The Layout View
- 2; Example Maps
  - Street Base Mapping
  - Transit Mapping
  - Node/Point Chart Maps
  - Intersection Level Of Service Maps
  - Multi-Bandwidth Maps
  - Desire Line Maps
- 3; Path File Select Link Analysis
- 4; Saving Settings in the VPR file
- 5; Sharing maps using MXD files

The Layout View

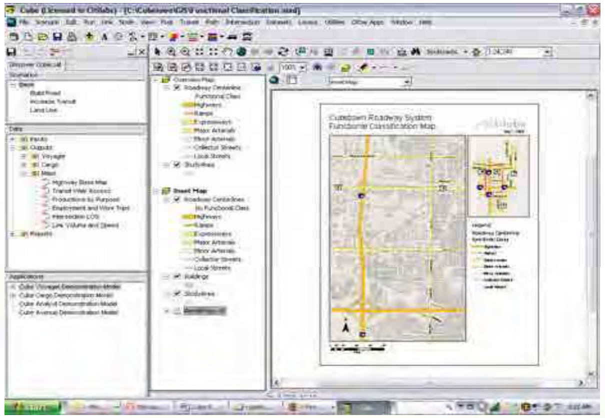


Add An Aerial Photo to Map



- Go to Datasets > Add Raster Data…
- Browse to C:\Intro Cube Base\Export.tif
- Move Network layer to top of Table of Contents
- Zoom in to downtown

Example Maps

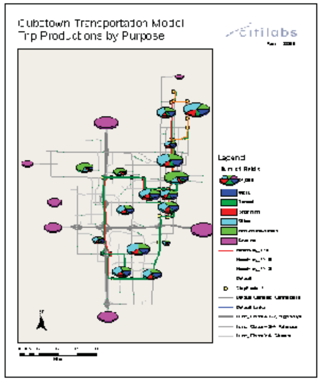


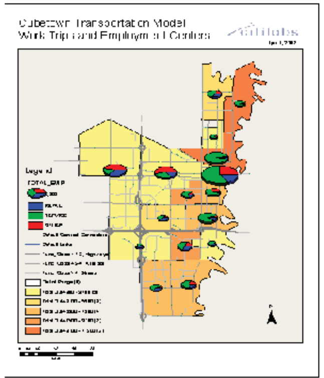
Training – Cube Base v5

Transit Walk Access Map

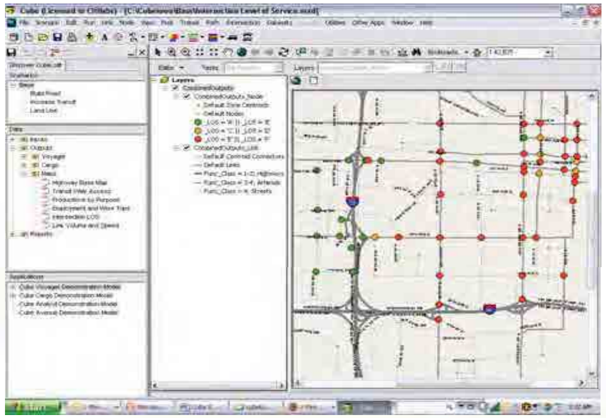
citilabs
TRANSPORTATION MODELING SOFTWARE

Node/Point Chart Maps



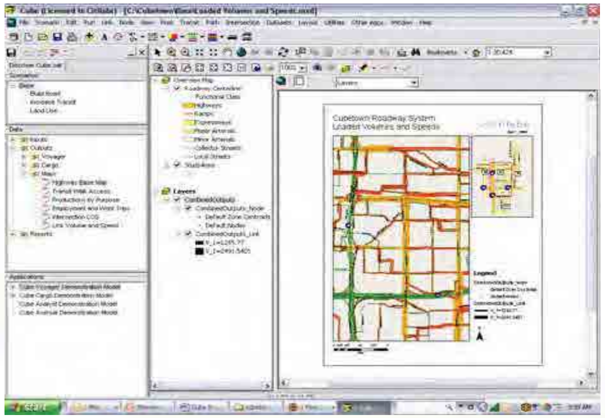


Intersection LOS Map



Training – Cube Base v5

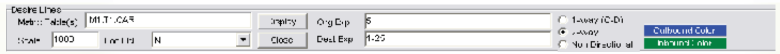
Link Volume and Speed



Training – Cube Base v5

Creating a Desire Lines Map

1. Open the output Mode Trips matrix
2. Open the output HW Loads network
3. From the Node menu, select Link to Matrix
4. Double-click on the Available Linkage and click Close
5. Go to Post > Desire Lines
6. Enter M1.T1.Car in Matrix Tables, 1000 in Scale, 5 in Org Exp, 1-25 in Dest Exp, and select 2-way
7. Click on the Display button to view desire lines



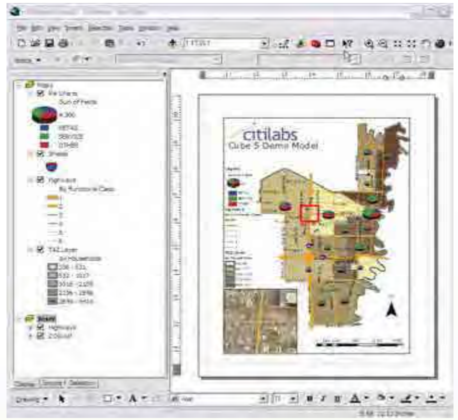
The roles of VPR and MXD files

- The Visual Project (VPR) file is still used in to track and store settings made in Cube 5, including:
  - Line/Node/Area color and symbol sets
  - Attribute posting and label symbol & style sets
  - Selection sets
  - Other network options
- A VPR is created for each MDB, with the same name as the MDB
- You can import settings from an existing VPR file for another MDB
- The MXD file is an ArcGIS-compatible map document, containing a “snapshot” of the current symbol style settings, with no link to VPR
- Changes stored in the VPR do not affect the MXD and vice versa!

Training – Cube Base v5

Share With GIS Staff

You (or other GIS users) can also create maps for Cube 5 using ArcView 9.2 or 9.3. Add advanced elements such as multiple inset frames with extent rectangles or semi-transparent layers, and specify detailed symbol style and legend settings. Once your map is saved as an *.mxd file it can be opened in Cube 5 as well.

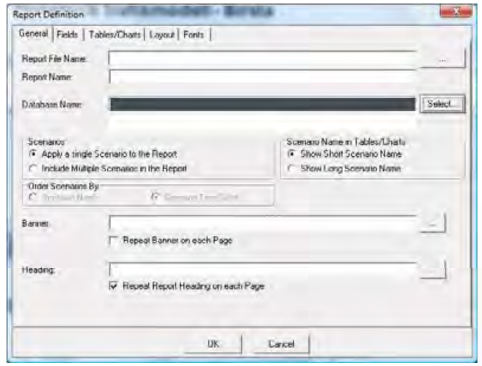


Overview of Report Creation Steps

- 1 ; overview of Cube Reports
- 2 ; creating charts
- 3 ; creating tables
- 4 ; copying to other programs such as MS Office

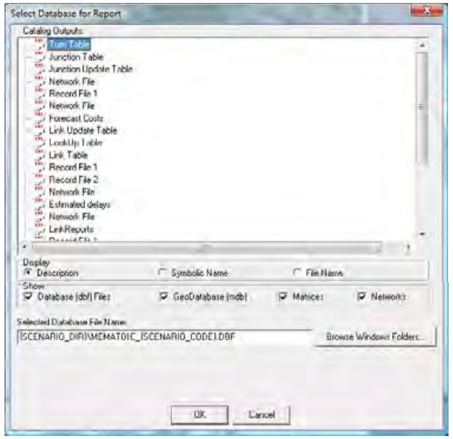
Overview of Cube Reports

- Create from Data panel with a right click on Scenario Reports
- Select database/file (all outputs defined for the model are available)
- Choose file with path to save to
- Choose single or multiple scenarios
- Create the chart and table elements for the report
- Pie charts, histograms, scatter plots, standard tables, cross tabulation tables
- Choose layout and styles



Cube Reports - data

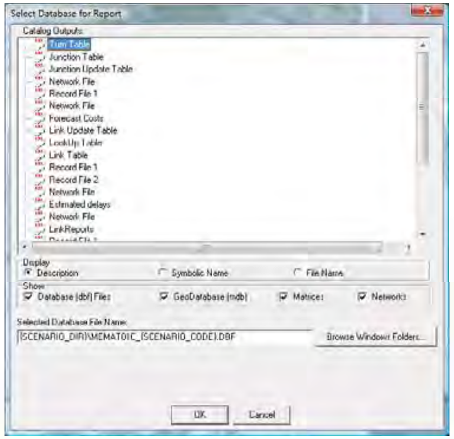
- Choose output table/file
- DBF files, geodatabases, matrices, networks
- Files can also be picked up from outside the catalog



Training – Cube Base v5

Cube Reports - data

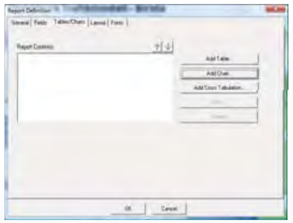
- Choose output table/file
- DBF files, geodatabases, matrices, networks
- Files can also be picked up from outside the catalog



Cube Reports – variables and chart types

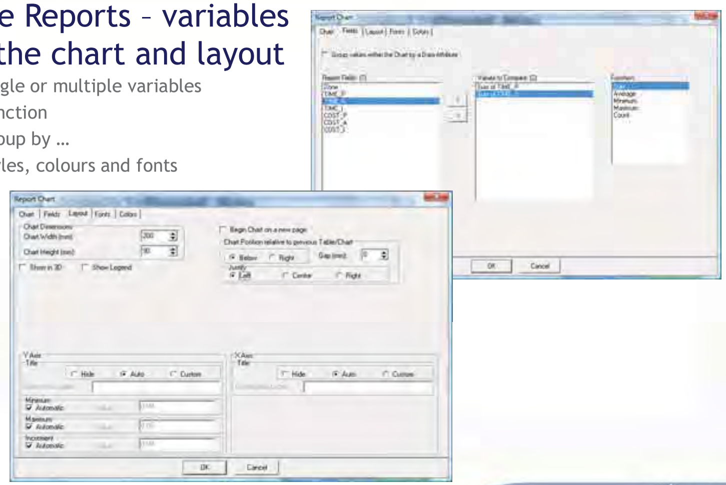
- Alias names for variables
- Choose what chart or table to create





Cube Reports – variables for the chart and layout

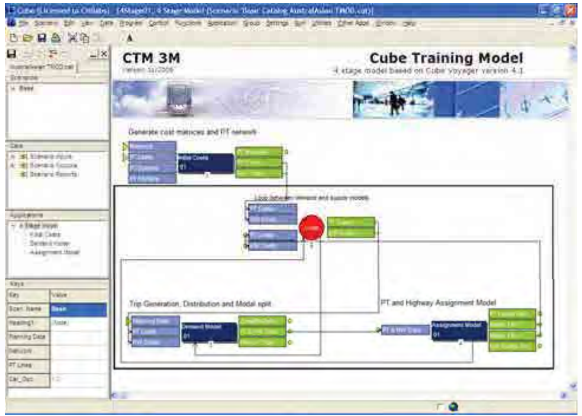
- Single or multiple variables
- Function
- Group by …
- Styles, colours and fonts



Overview of Model Development Steps

- 1 ; open model, make new subgroup ‘Assignment’ and open this new group
- 2 ; choose program ‘HIGHWAY’ and ‘PUBLIC TRANSPORT’ from the Cube menu, programs
- 3 ; link in input files
- 4 ; define keys (network and PT lines)
- 5 ; create output files
- 6 ; define scenario specific output files, make some ‘public’ and add some to the catalog
- 7 ; put in headings for flow chart (AM) and run menu
- 8 ; set model to application mode

Overview of the model

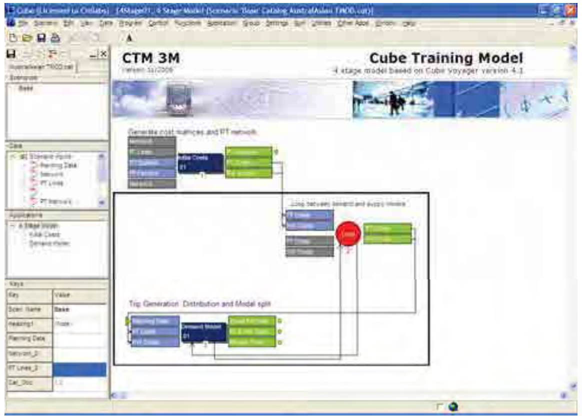


- Full 4 Stage model
  - Initial cost calculation
  - Generation, distribution and mode split
  - Highway and PT assignment
  - Loop between demand and supply
- Full integration in Cube Base

TRANSPORTATION MODELING SOFTWARE

Training – Cube Base v5

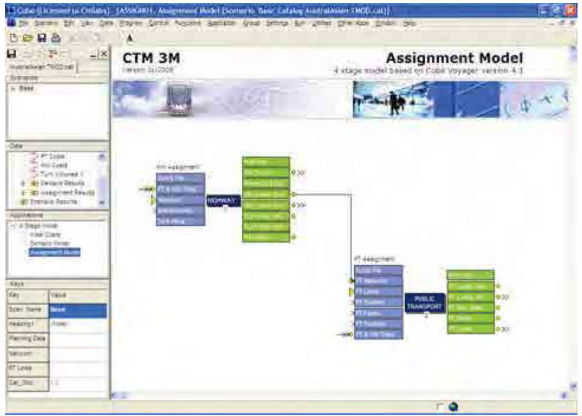
Model development steps



D STEP 1
- Open Cube
- Open Training Model from welcome screen
- Open application
- Make new sub group from 'Application' menu
- Name it 'Assignments' and save the group in the 'Application' folder

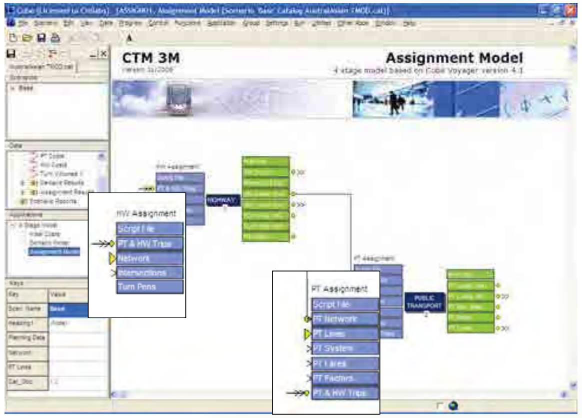
Training – Cube Base v5

Model development steps



D STEP 2
- Pull in the programs HIGHWAY and PUBLIC TRANSPORT from the 'Program' menu
- Link in necessary files (move mouse over to see which)

Model development steps

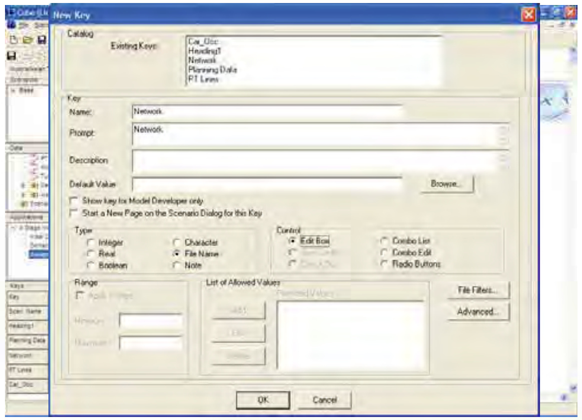


D STEP 3

- Link in necessary files (move mouse over to see which)
- Right click on the various input files to do this
- Then define keys ('Network' and 'PT Lines) by clicking in 'Keys' area, top left and choose 'Add'
- (please note that this will be a right click in Cube)

cube Training – Cube Base v5

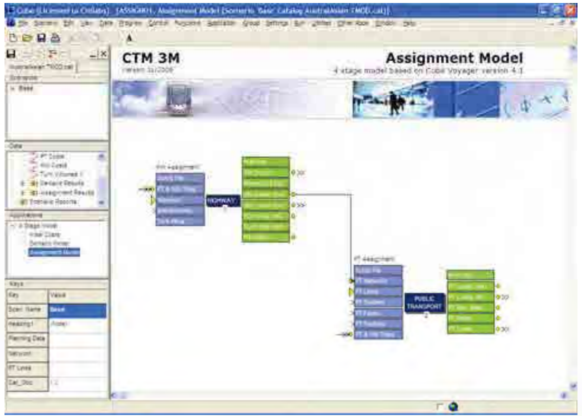
Model development steps



D STEP 4

- Define keys 'Network' and 'PT Lines' (see key dialog ...)
- Then define necessary output files (click on 'OK' button to continue)

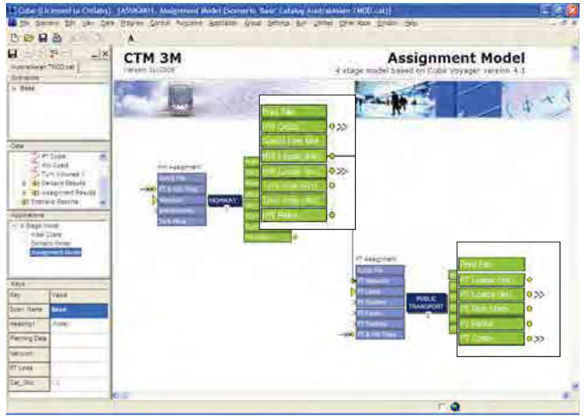
Model development steps



D STEP 5

- Create output files by right clicking on the various files (move mouse over to see which)

Model development steps

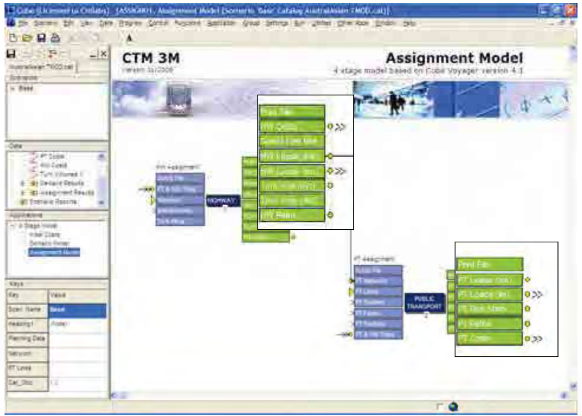


D STEP 5

- Create output files by right clicking on the various files

citilabs
TRANSPORTATION MODELING SOFTWARE

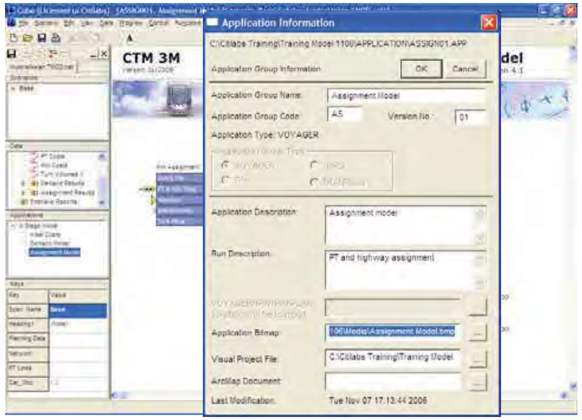
Model development steps



D STEP 6

- ‘Make File Scenario Specific’ by right clicking on the appropriate files (the ones with circles)
- ‘Make File Public’ (by right clicking) and add some to data catalog

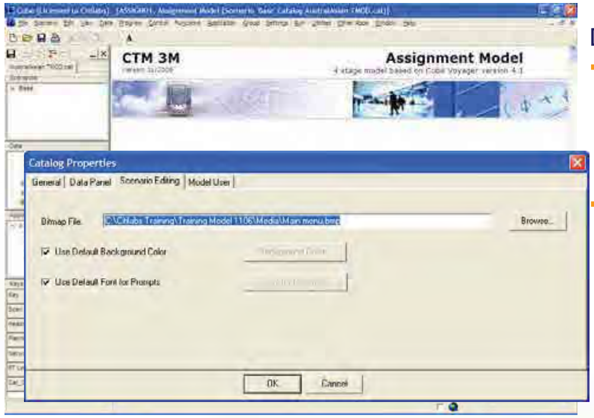
Model development steps



D STEP 7

- Add heading (bitmap) to flow chart from ‘Application’ … ‘Properties’ menu

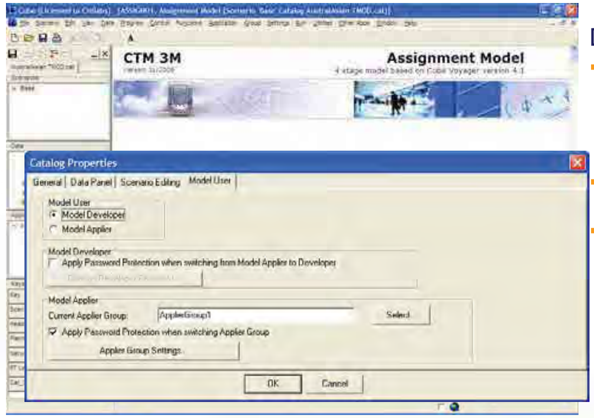
Model development steps



D STEP 7

- Add heading to run menu by right clicking on catalog tab (far top left) and choose 'Properties'
- Do this from 'Scenario Editing' tab

Model development steps



D STEP 8

- Change model user to 'Model Applier' by right clicking on catalog tab (far top left)
- Go to 'Model User' and check 'Applier'
- Then you are done with the development!

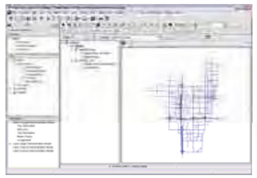
citilabs
TRANSPORTATION MODELING SOFTWARE

Introduction to Cube

Cube Voyager – functionality and programming

Essential Operations in Cube Voyager

- Pilot:
  - Control process flow (looping, conditional flow)
  - Call Voyager modules and run shell commands
  - Compute, store and retrieve “global” variables
- Matrix:
  - Compile and process matrices and tabular data
  - Compute and summarize zone-to-zone information
  - Manipulate tables, spreadsheets, and databases
- Network:
  - Compile and process multi-modal transport networks
  - Convert, merge, and prepare model input networks
  - Post-process and summarize model outputs



citilabs
TRANSPORTATION MODELING SOFTWARE

Demand Forecasting Tools in Cube Voyager

- Generation:
  - Applies linear equations or lookup tables to forecast/compute total trip ends (productions and attractions) by purpose and zone
- Distribution:
  - Distribute trips between productions and attractions based upon skims
- Fratar:
  - Adjust a matrix to match row/column targets or growth factors
- Matrix (XCHOICE):
  - Hierarchical LOGIT choice model application, including joint destination-mode choice and incremental modelling

Supply / Assignment Modeling Tools

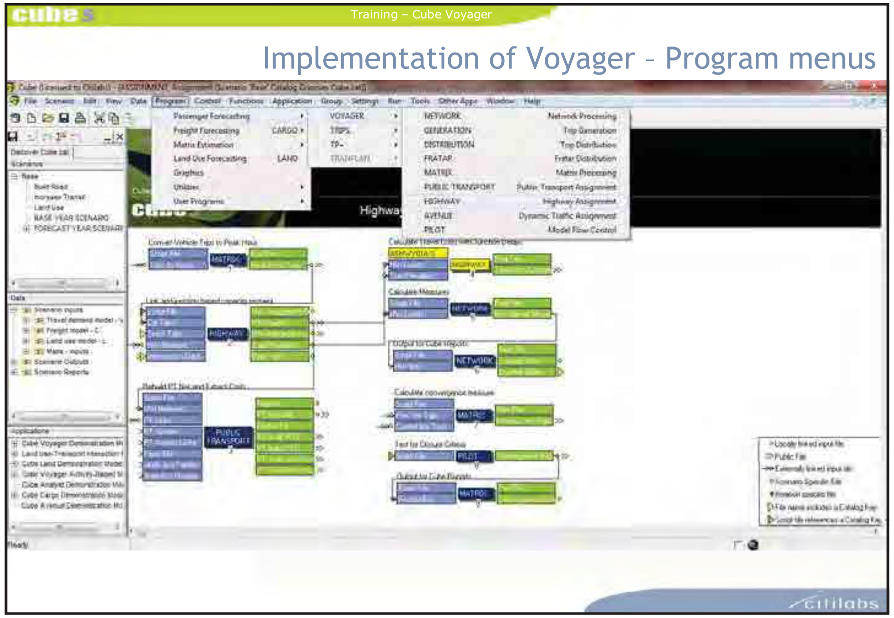
- Highway:
  - All zone-to-zone network path analysis operations for continuously available transportation services
  - Flexible convex combinations algorithms for equilibrium network traffic assignment
  - Explicit consideration of turn delays and detailed intersection analysis
  - AVENUE: An add-on to HIGHWAY for dynamic traffic assignment with meso-scopic simulation, including queue propagation and flow metering.
- Public Transport:
  - All scheduled transit service functions
  - Access, transfer, and egress analysis and network development
  - Multi-path route enumeration and evaluation including sub-mode choice
  - Transit fare analysis and crowd modeling
  - Limitations: up to 1,000 modes/operators/fare systems; 255 wait curves, crowding curves, and/or vehicle types; 10 user classes; 32,000 zones, 10^6 nodes/links, unlimited lines

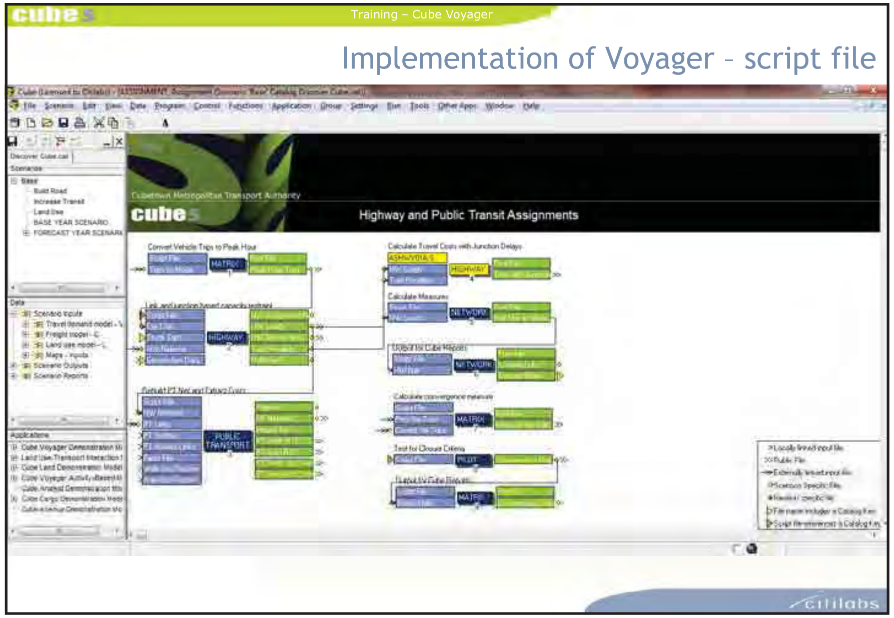
Transport supply and infrastructure data

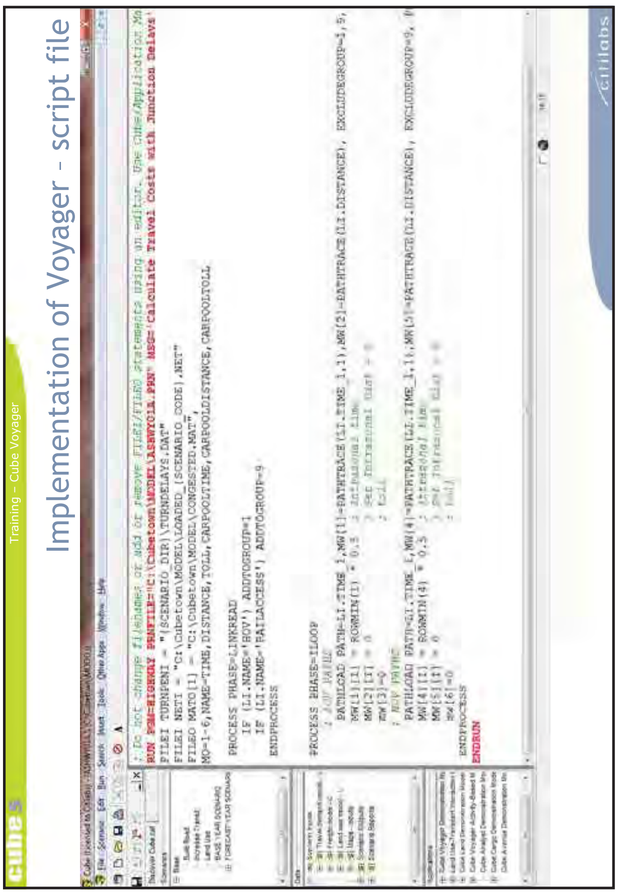
- “Highway” networks
  - 32,000 zones; 999,999 nodes; 999,999 links; unlimited attribute tables
  - Native binary Citilabs format for compression & efficiency
  - ESRI custom personal geodatabase feature dataset for GIS
  - ASCII (CSV) text and DBF for exchange
- Intersection / junction data
  - ESRI personal geodatabase formats
  - ASCII text format (Cube Voyager syntax)
  - Roundabout, priority, fixed-time and adaptive signals, two-way and all-way stops
  - Used to apply HCM 2000 or saturation flow lane group capacity and delay
- Public transport services
  - ESRI personal geodatabase format: lines, non-transit legs
  - ASCII text format (Cube Voyager syntax): lines, system data, fares, factors
  - Binary format to consolidate data with underlying multimodal network

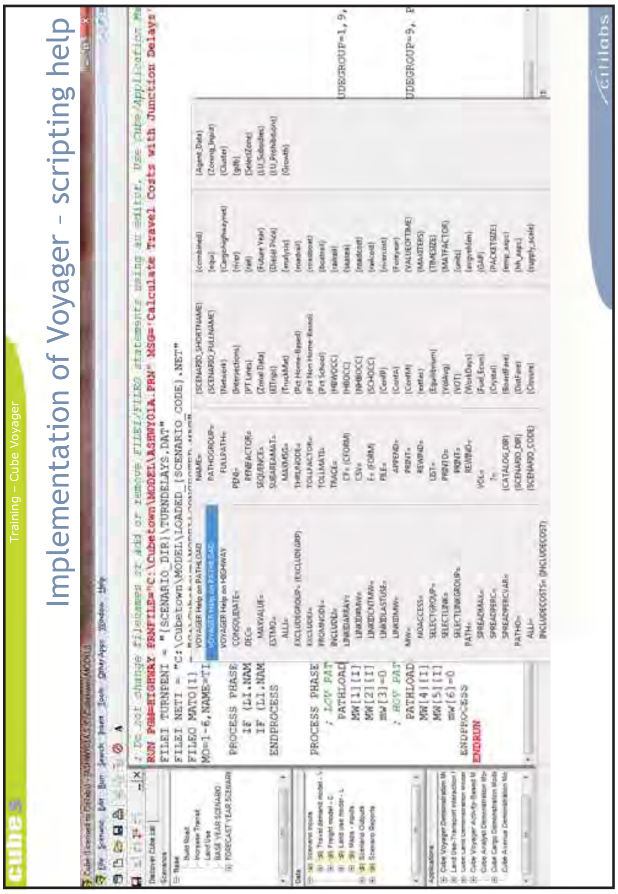
Demand Data and Other Tables

- Zonal Data
  - Stores socio-economic & demographic information, trip ends
  - Exchange formats: ASCII (e.g. CSV) text, DBF (SHP)
  - Best GIS format: geodatabase polygon feature class
- Matrix Data
  - Zone-to-zone information of any kind e.g. trips, costs
  - Up to 255 tables per file; 32,000 zones
  - Native binary MAT format for compression, efficiency
  - ASCII (e.g. CSV) text, DBF formats for exchange
- Record and Database Files
  - ASCII (e.g. CSV) text, DBF, MS Access 2003, ESRI personal geodatabase formats
- Lookup Tables
  - ASCII (e.g. CSV) text, DBF formats


Training – Cube Voyager
Implementation of Voyager – Program menus


Training – Cube Voyager
Implementation of Voyager – script file
Highway and Public Transit Assignments


Training – Cube Voyager
Implementation of Voyager – script file


Training – Cube Voyager
Implementation of Voyager – scripting help

Training – Cube Voyager

Cube Voyager Syntax and Structure

- All statements follow the same general structure:
  - COMMAND KEYWORD=VALUE SUBKEYWORD = VALUE
- KEYWORDs are always followed by an equals sign and a VALUE
- The VALUE may be the result of an expression or computation
- Continuation characters (commas, equals signs, any operator) distribute statements across multiple lines
  - E.g.: COMMAND,
    KEYWORD=VALUE,
    SUBKEYWORD=VALUE

Training – Cube Voyager

Cube Voyager Expressions

- Operators:

+	addition	||	logical OR
-	subtraction	&&	logical AND
*	multiplication	==, =	equals
/	division	!=, <>	does not equal
%	modulus	>=	greater than or equal to
^	exponentiation	<=	less than or equal to
()	parentheses	>	greater than
		<	less than

- Numeric functions:
  - ABS, INT, ROUND, MAX, MIN
    EXP, LN, LOG, POW, SQRT
    RAND, RANDOM, RANDSEED

Cube Voyager Expressions

- Trigonometric functions:
  - COS, ARCCOS, SIN, ARCSIN, TAN, ARCTAN
- Character functions:
  - DELETESTR(s1,n1,n2), DUPSTR(str,n), FORMAT(x,w,d,str), INSERTSTR(s1,s2,n), LEFTSTR(s1,n), LTRIM(str), REPLACESTR(s1,s2,s3,n), REPLACESTRIC(s1,s2,s3,n), REVERSESTR(s1), RIGHTSTR(s1,n), STR(v,w,d), STRLEN(str), STRLOWER(str), STRPOS(str,str2), STRPOSEX(s1,s2,n1), STRUPPER(str), SUBSTR(str,b,n), TRIM(str), VAL(str)
- Special functions:
  - CmpNumRetNum(V1,OP,V2,R1,R2)
  - Compare number V1 to number V2 based on OP and return R1 if result is true and R2 if result is false.

Comments and style guidelines

- Include plenty of comments—for yourself and others
  - Everything following a semi-colon on any line is a comment
  - Use ; comments to explain intent behind coded commands
- Everything between /* and */ is a control block (ignored by parser)
  - Use control blocks to turn off sections of script without deleting
- Use full COMMAND KEYWORD=VALUE syntax when possible
- Write system commands, keywords, and functions in UPPERCASE, reserve lower and proper case for user-defined variables and names
- NEVER have your script file open when you are linking new files into a Voyager Module. Input and output files will not get updated!
- Try right-clicking when you are unsure what should come next

Program module structure

- Cube Voyager program modules may contain one or more “phases”
- Each phase performs a user-specified sequence of operations on each element of a data structure, such as links in a network
- Typical program script:

```
;Comments preceded by semicolon
RUN PGM=name
FILEI … ;specify input files
FILEO … ;specify output files
PARAMETERS … ;global settings not in any phase
PROCESS PHASE=…
 COMMAND KEYWORD=…
 ;more commands…
ENDPROCESS
;more phases…
ENDRUN
```

- Additionally, some programs have an iterative looping framework that will repeat certain phases until a convergence criterion is met

General Setup

```
Comments                  ;Used to document your model
                          ; whole lines or
                          ; end (;) of command line
RUN Program               ; start of program run/module
FILES                     ; input and output files as specified in Flow
                          ; Chart (we do not need to type this in
                          ; Application Manager does this automatic for us).
PARAMETERS                ; setting values for variables
PROCESS PHASE =           ; start a calculation process
   COMMAND keyword=       ; document/explain the command
   ….more commands
ENDPROCESS                ; end the calculation process
…. more processes
END                       ; end of program run
```

Functions Used in All Modules

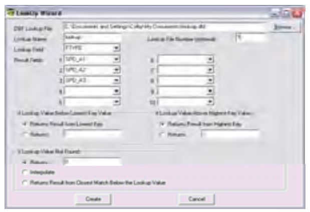
- IF Statement
  - IF(k > l)…
  - = == > < >= <= != <>
  - & && | || !
  - Can compare item with list: IF(I=1-10,21,25-29)
  - IF (condition) COMMAND
  - IF (condition)
    - …
  - ENDIF

Functions Used in All Modules

- PRINT Control Statement
  - LIST= a,b,c
  - FORM= fieldWidth&format sets default format
  - FILE=filename if writing to file
  - PRINTO=# ; Refer to print output file
  - CSV=T ; Automatically sets the output to a CSV format
  - Items in LIST may have their own format: VariableName(format)
  - Formats:
    - field width 10: (10)
    - field width 8, 2 decimal places: (8.2)

Functions Used in All Modules

- LOOKUP table
  - A lookup table can be used anywhere in Voyager script
  - There are multiple forms; however the most common is a DBF file with several columns, some of which represent lookup fields, and others representing result fields
  - You can automate lookup table coding using the wizard at Insert > Lookup Table
  - Example:
    speed by area type and facility type code
  - Limitation: only one lookup value



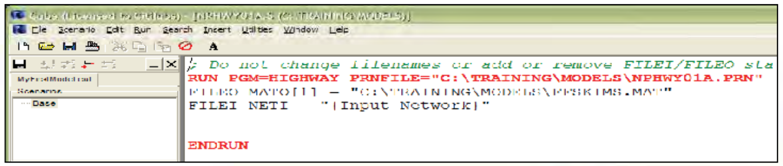
```
FILEI LOOKUPI[1]="lookup.dbf"
LOOKUP LOOKUPI=1,
 NAME=lookup,
  LOOKUP[1]=FTYPE,RESULT=SPD_A1,
  LOOKUP[2]=FTYPE,RESULT=SPD_A2,
  LOOKUP[3]=FTYPE,RESULT=SPD_A3,
       FAIL[3]=0
; example of use: v=lookup(3,25)
; look for 25 in the FTYPE field
; and returns the SPD_A3 value
```

Modelling TIPS

- Always write scripts in CAPITALS, certain control statements are case sensitive
- Include Lots of Comments to your scripts (Don't Forget the ;)
- NEVER have your script file open when you are linking new files into a Voyager Module: input and output files will not get updated
- Use the right click when unsure what comes next

Generated Script Files

- A script file is created and maintained by AM
- AM automatically generate the basic structure
- AM keeps track of all the input and output files
- The user will add Parameters, Process blocks, Calculations
- The user should NEVER edit the file names and locations
- All changes to the file names and locations should be done in AM



Cube Voyager
Network

Training – Cube Voyager

Network

- Building, Comparing and Manipulating Highway Networks
- Module Basics
  - Inputs:
    - Up to 10 Link Files
    - Up to 10 Nodes Files
    - Up to 10 Networks
  - Outputs:
    - 1 Network (Citilabs Binary/Geodatabase);
    - 1 Link and 1 Node (text, dbf, binary Citilabs, geodatabase table)
    - Up to 30 Print Files
  - Variables:
    - Unlimited Link & Node, 15 character limit, referencing=LI.#.name or NI.#.name, working variable=_varname
    - E.g. LI.2.distance ;Read in from link input file #2 the distance variable

Training – Cube Voyager

Network

- Module Structure – Linkloop/Nodeloop/Phases
  - INPUT:
    - Read ASCII and DBF files, re-code values from any input files specifically designated.
  - NODEMERGE:
    - Read all node data and organize it.
  - LINKMERGE:
    - Read all link data and process it (main phase).
    - The process compares corresponding A-B links across all input networks. It then selects link variables from each network as per your specification. The default is numbers from the first input link file if nothing is specified.
  - SUMMARY:
    - Report results of LINKMERGE phase

citilabs
TRANSPORTATION MODELING SOFTWARE

cube Training – Cube Voyager

Cube Voyager
Matrix

cube Training – Cube Voyager

Matrix: Demand Modelling and Matrix Manipulation

- Matrix is primarily a calculator. It simply processes matrices, zonal data, or text records according to user specified expressions.
- Inputs may include:
  - Matrices
  - Zonal data files
  - Record Data files
- Outputs may include:
  - Matrices
  - Print Data files
  - Record files
- Various file formats for both input and output are supported
- User is responsible for specifying what is to be accomplished

citilabs
TRANSPORTATION MODELING SOFTWARE

Training – Cube Voyager

Matrix: Demand Modeling

- Module Basics
  - Inputs:
    - Up to 20 matrix files (w/ 255 tables each)
    - Up to 10 zonal data files
    - 1 record data file of ascii/dbf data (Record Processing)
    - Up to 10 Database files (BDI Processing)
    - Up to 20 lookup tables
  - Outputs:
    - Up to 20 trip matrix files (w/ 255 tables each)
    - Up to 15 record data files (dbf format)
    - Up to 10 print files
- Module Structure – ILOOP/JLOOP
- Allows up to 999 internal tables

Training – Cube Voyager

Matrix – Common Uses

- Commonly used to:
  - Calculate new matrix values
  - Convert and merge matrices between various formats
  - Report values from matrices and zonal data by:
    - Selected rows
    - Marginal summaries (trip ends, etc.)
    - Frequency distributions
  - Transpose matrices
  - Generate matrices
  - Renumber, aggregate, and disaggregate matrices
  - Process Record data

TRANSPORTATION MODELING SOFTWARE

Cube Voyager
Highway

Purpose and function of Highway module

- Function is to build paths using network links and either extract path cost/impedance matrices or assign trips
- Input includes highway network, zonal matrices, zonal data, and turn penalties
- Output includes a loaded network, new matrices, turning volumes and reports
- Basic default operations available with the user controlling much of the process

Training – Cube Voyager

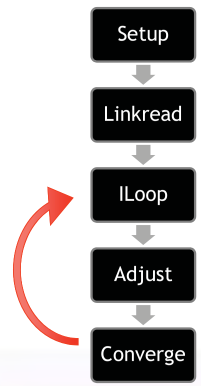
Highway module – input and output data types

- Inputs:
  - Up to 20 trip matrices
  - 1 highway network
  - 1 turn penalty file
  - 1 junction file
  - Up to 10 zonal data file
  - 1 SubArea network
  - Up to 10 toll matrices
  - Up to 30 lookup files
- Outputs:
  - Up to 20 LOS matrices
  - 1 SubArea matrix
  - 1 assigned network
  - Up to 5 Turn volume files
  - Up to 10 Path Files
  - 1 junction volume file
  - 1 intercept file for Cube Analyst
  - 1 screeenline file for Cube Analyst
  - Up to 10 print files

Training – Cube Voyager

Highway module process structure

- Phases – multiple iterative loops
  - SETUP: initialize certain variables and/or arrays
  - LINKREAD: obtain required initial values that and compute link values referenced elsewhere.
  - ILOOP: loop across all zones, building and loading minimum paths as requested
  - ADJUST: revise the link variable values for output or use in the next iteration
  - CONVERGE: check to determine whether additional iterations are necessary
- Methods – convex combinations
  - Multi-user class equilibrium, average or weighted assignment,
  - incremental assignment, all-or-nothing,
  - multi-user class link and intersection constrained equilibrium assignment,
  - user defined…



Path Building in Voyager

- Paths are built by origin zone to ALL destinations zones in the ILOOP Phase
- Path building:
  - initiated with the PATHLOAD statement
  - built on a fixed set of link costs for current iteration and origin zone
  - The PATH= keyword defines link costs used for path building
- Multiple path sets can be built using multiple PATHLOAD statements
- Specific links can be excluded from path sets
- Skims of any current link attribute can be created as paths are built
- OD Trips can be loaded to PATH sets by volume set

Cube Voyager Generation

Generation

- Primary function is to process zonal data and generate arrays of productions and attractions
- Calculations and balancing functions are user defined

- Input may include
  - Up to 10 zonal data files

- Output may include
  - Up to 10 production and attraction files

Generation PHASE=ILOOP

- In this step the stack of calculations are performed on each zone.
- The user usually accessed a Zonal Data File and supplies regression equations.
- Script:
  - P[1]=3.2*ZI.1.HOUSEHOLDS
  - P[2]=2.5*ZI.1.HOUSEHOLDS
  - P[3]=1.2*ZI.1.HOUSEHOLDS
  - A[1]=1.6*ZI.1.TOTAL_JOBS
  - A[2]=34*ZI.1.TOTAL_JOBS
  - A[3]=19*ZI.1.TOTAL_JOBS

Generation PHASE=ADJUST

- This phase is used to balance the attraction and production totals.
- The user can use the 'BALANCE' function to set:
  - Production Totals to Attraction Totals (P2A)
  - Attraction Totals to Production Totals (A2P)
  - Attraction Totals to Production Totals then the Number of Productions set to the Number of Attractions (NHB)
- This can be accomplished using 'math' or the 'Balance' function.
- Script:
  - BALANCE A2P=1,3 NHB=2

Cube Voyager Distribution

Distribution

- Uses the number of trip ends in each zone as the starting point. These ‘margin’ totals are distributed to the rows and column of a generated matrix.
- The distribution is weighted by the ‘Impedance’
- The impedance is calibrated with a Friction Factor Curve.
- Most common process is the "gravity" model, but there is no automatic, or default, trip distribution process.
- Input may include:
  - Up to 20 matrix files, 1 text data file of data (friction factors). Up to 10 production and attraction files
- Output may include:
  - Up to 32 trip matrix files

Voyager Matrix for Modal Choice

Mode Split – Choice Modelling

Training – Cube Voyager

```
XCHOICE  Command
ALTERNATIVES=...   ; Define each alternative choice (eg. Car, PT)
DEMANDMW=...       ; Define input trip matrix to be split
COSTSMW=...        ; Define cost matrices for each alternative
ODEMANDMW=...      ; Define working matrices to store output trip
                          matrices for each alternative
STARTMW=...        ; Define a working matrix for internal calculation
SPLIT= ...         ; Define the choice model (i.e. structure and
                   ; scale)
```

Choice Models that can be developed in Cube include:

Simple binary or multinomial; hierarchic; destination choice

Absolute or incremental

Cost or utility based

Mode Split – Absolute Model

Training – Cube Voyager

- Logit Curve Formula (cost-based model):

$Pr_{(PT)} = exp(-\lambda \times GC_{(PT)})/[exp(-\lambda \times GC_{(PT)}) + exp(-\lambda \times GC_{(Car)})]$

$Pr_{(Car)} = exp(-\lambda \times GC_{(Car)})/[exp(-\lambda \times GC_{(PT)}) + exp(-\lambda \times GC_{(Car)})]$

- Where

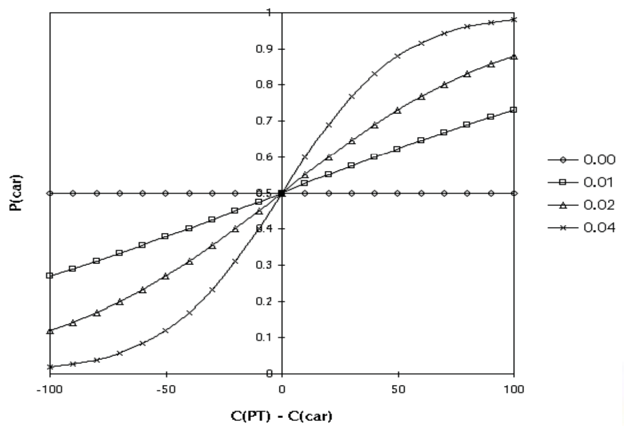
  PR= Probability of use (Mode)

  GC= Generalised Cost (Mode – includes mode-specific constant)

  λ = Lambda ; defines "slope" or sensitivity to cost

Training – Cube Voyager

Mode Split – Logit Curve Example



Training – Cube Voyager

Public Transport Assignment Procedures

citilabs
TRANSPORTATION MODELING SOFTWARE

cube Training – Cube Voyager

Public Transport

- Primary function is to assign trips to public transport network services
- Input includes highway network, zonal matrices, fare structure and route choice factors
- Output includes a loaded network, new matrices, PT patronage and reports

citilabs

cube Training – Cube Voyager

Public Transport

- Inputs:
  - 1 highway network
  - Up to10 trip matrices
  - 1 system file
  - Up to 10 factor files
  - Up to 15 line files
  - Up to 5 NTLEG files
  - Up to 10 route files
  - 1 Screenline data file
  - Up to 5 lookup files
- Outputs:
  - 1 line file
  - 1 network,
  - Up to 4 DBF link file
  - Up to 10 matrices,
  - 1 NTLEG File,
  - 1 Report File,
  - Up to 10 Route Files
  - 5 DBF Stop to Stop file
  - Up to 20 Print files
  - 1 intercept file for Cube Analyst
  - 1 screenline file for Cube Analyst

citilabs

Public Transport

- Processes
  - Construct PT Network
  - Perform Route Enumeration - Is a process of finding one or more discrete routes between zone pairs, which have some probability of being used by passengers to travel between the zones
  - Perform Route Evaluation - Is a process to calculating the 'probability of use' of each of the enumerated routes between zone pairs
  - Skim costs
  - Assign PT Trip

Public Transport

- Route Evaluation Choice Models for determining alternative PT route Probabilities are:
  - Walking Choice Model
    - Is a logit model to determine the proportion of demand when alternative walk choices are available
  - Service Frequency Model
    - Determines the proportion of demand for competing PT services based upon service frequency
  - Service Frequency and Cost Model
    - Determines the proportion of demand for competing PT services based upon service frequency and travel cost
  - Alighting Model
    - Is a logit model to determine the proportion of demand when there are two or more valid alighting points

citilabs TRANSPORTATION MODELING SOFTWARE

Public Transport

- Assignment Types:
  - Multi-routing
  - Multi-routing with crowd modelling
  - All or Nothing (best path)

- Note: All or Nothing cannot be used in conjunction with Crowd Modelling

Public Transport – PT Lines Input File

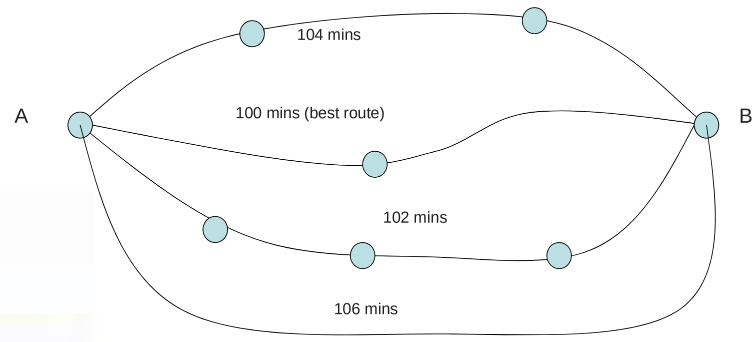
- PT Lines File summarises PT service information using the LINE Control statement and the following keywords:

- MODE=num ; defines the service's mode, eg. Train, bus
- HEADWAY[num]=num ; defines the service's headway in minutes, up to five alternative headways can be stored
- VEHICLETYPE=num ; defines vehicle type characteristics as specified in the system file
- NODE=num ; specifies the list of sequential nodes that the PT service travels along
- Node has many the sub keywords including:
- SPEED=num ; defines speed between nodes
- DELAY=num ; adds additional link time between two nodes
- Cube base summarises the information in a graphic format for easy editing.

Public Transport – System Input Files

- PT System File contains information that defines:
  - Public transport modes (i.e. mode ID and name)
  - Operators of public transport (i.e. bus and train companies)
  - Vehicles types (including, vehicle number ID, seat capacity, crush capacity, crowding curve, load distribution factor for each vehicle type)
  - Wait curves for both initial boarding and transfers
  - Crowding adjustment curves for crowing modelling
- Node: Cube Base provides a graphic interface for defining this data

Public Transport – Example



citilabs
TRANSPORTATION MODELING SOFTWARE

Introduction to Freight Forecasting With Cube Cargo

Benefits of Freight Modeling

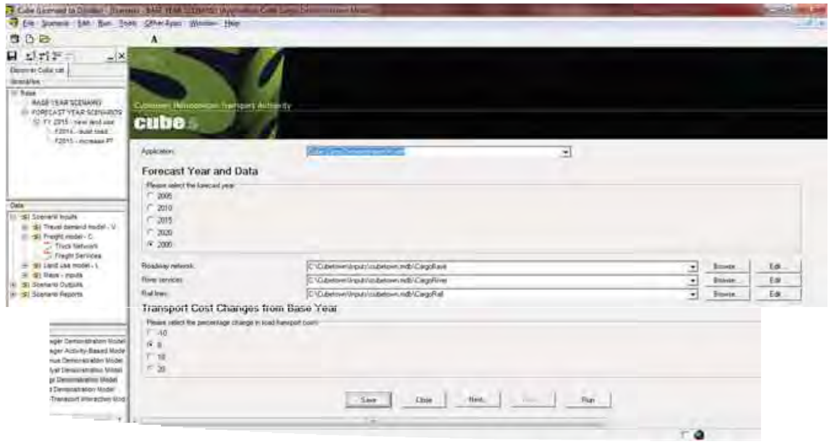
- To Answer Policy Questions
  - Effects of alternative growth scenarios on freight movement
    - What if regional development patterns change?
    - What if major freight facilities are developed?
  - Effects of alternative policies on freight movement
    - What if tolls were increased?
    - What if the price of fuel continues to increase?
  - Impacts of major projects on freight movement
    - What if a two-lane US highway was widened to four lanes?
    - What if major access improvements to a region were advanced?

Introducing Cube Cargo

- Generation: estimates annual tons of commodities produced and consumed by zone by commodity class
- Distribution: distributes goods by commodity class
- Mode Choice: estimate modal shares of long-haul flows
- Logistics Nodes Model: partitions the long-haul matrices by mode and commodity class into direct flows and transport chain flows
- Fine Distribution Model: for each of the matrices redistributes from coarse zones to the fine zones
- Vehicle Model: converts the estimated annual commodity flows by truck into number of heavy trucks and light trucks
- Service Model: estimates daily urban service truck trips

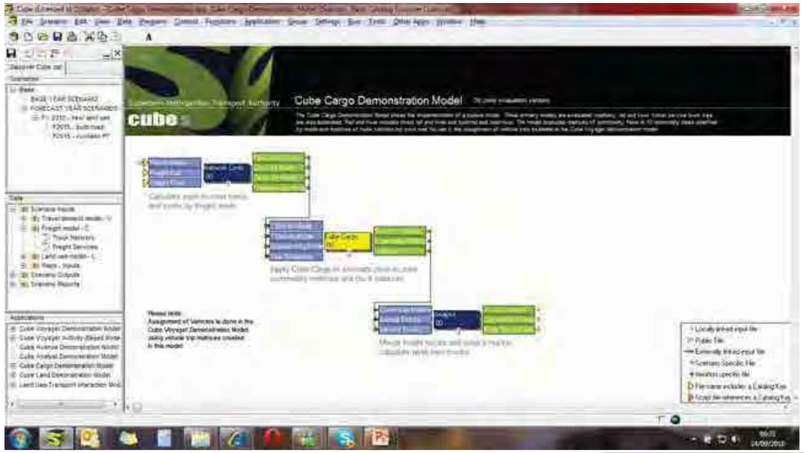
The Cube Cargo Model

- Main Interface for Running the Cube Cargo Model to test alternative scenarios



The Cube Cargo Model

- Cube Cargo model with Voyager modules for preparing and analysing data and results

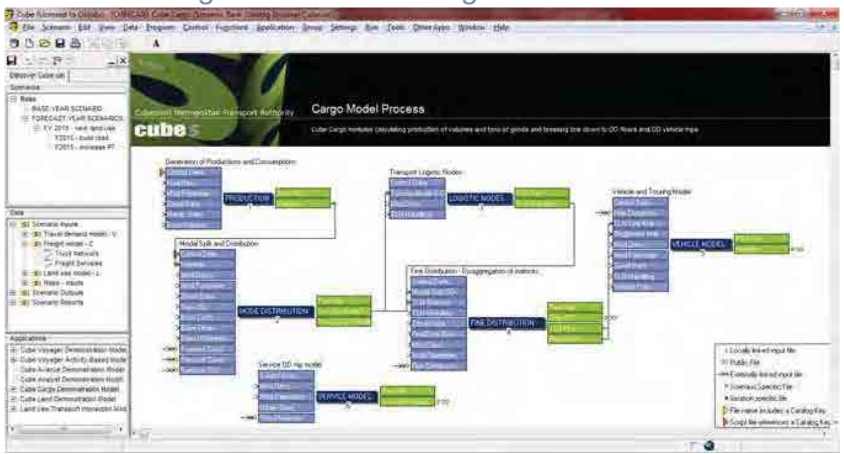


Network Data Pre-Processing

- Voyager modules (HIGHWAY, NETWORK, PUBLIC TRANSPORT, MATRIX) can be used to evaluate network inputs to Cargo

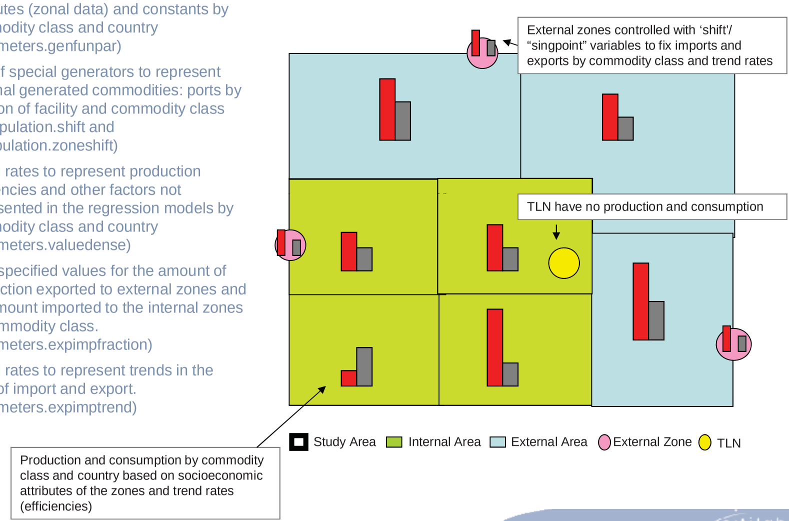
Cube Cargo Model Process

- Cube Cargo modules calculating production of volumes and tons of goods and breaking this down to OD flows



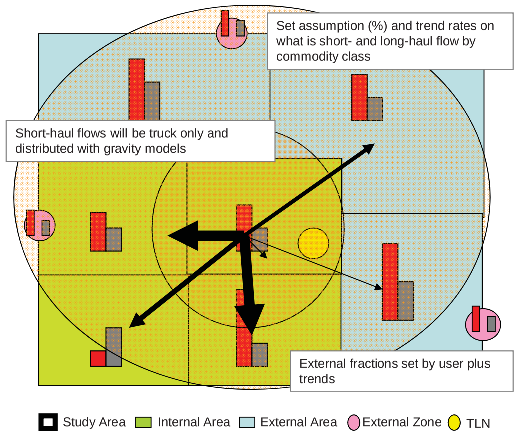
Generation

- Regression models on socioeconomic attributes (zonal data) and constants by commodity class and country (parameters.genfunpar)
- Use of special generators to represent external generated commodities: ports by location of facility and commodity class (manipulation.shift and manipulation.zoneshift)
- Trend rates to represent production efficiencies and other factors not represented in the regression models by commodity class and country (parameters.valuedense)
- User specified values for the amount of production exported to external zones and the amount imported to the internal zones by commodity class. (parameters.expimpfraction)
- Trend rates to represent trends in the level of import and export. (parameters.expimptrend)



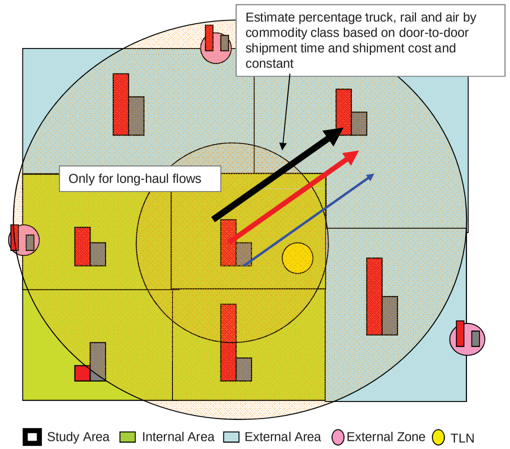
Distribution

- User assumption on percentage of goods that are to be considered short-haul and long-haul by commodity class (parameters.nearfraction)
- Trend rates to represent changes in short-haul and long-haul percentages by commodity class (parameters.nearfractrend).
- Short-haul trips are considered to be transported by truck and are distributed using gravity models by commodity class Impedance is cost. (parameters.neardistribution).
- Segments the remaining long-haul flows into those remaining ‘internal’ and those remaining ‘external’ by commodity class. (zonal data.foreignfraction)
- Adjusts internal and external fractions by user assumption on trends by commodity class (zonal data.foreigntrend).
- Distributes internal, import and export long-haul flows using gravity models by commodity class. (parameters.longdistribution) Impedance is a generalized cost using a linear combination of time, distance and cost by mode weighted by the mode choice coefficients. (parameters.mdsppar).



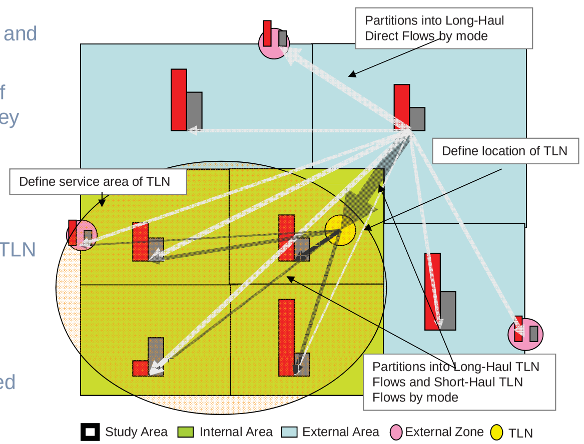
Mode Choice

- For Long-Haul Only
- Multinomial logit models by commodity distance class
- Choice models use by mode and commodity class
  - Time
  - Cost
  - Constant



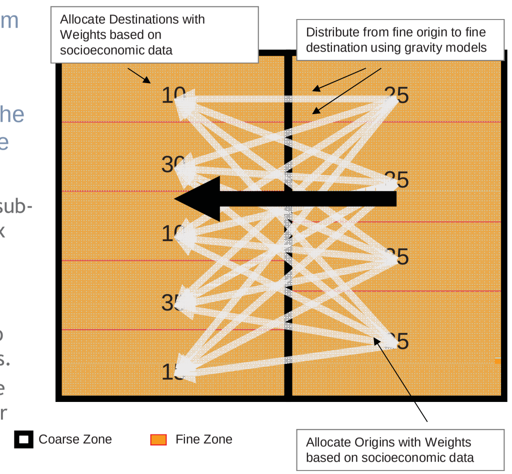
Logistics Node Model

- Partitions the long-haul matrices by mode and commodity class into direct and TLN flows
- definition of zone location of TLNs and the zones that they serve (tln.tlntable); (tln.tlnservedzones); (tln.tlnserved outerzones).
- definition of directionality of TLN flows and selection of TLN using tlnweightroad, tlnweighttrain and tlnweightship.
- product is matrices by commodity group segmented into:
  - Long-haul direct flows by mode (truck, rail and air)
  - Long-haul to/from TLN flows by mode (truck, rail and air)
  - Short-haul to/from TLN flows by truck



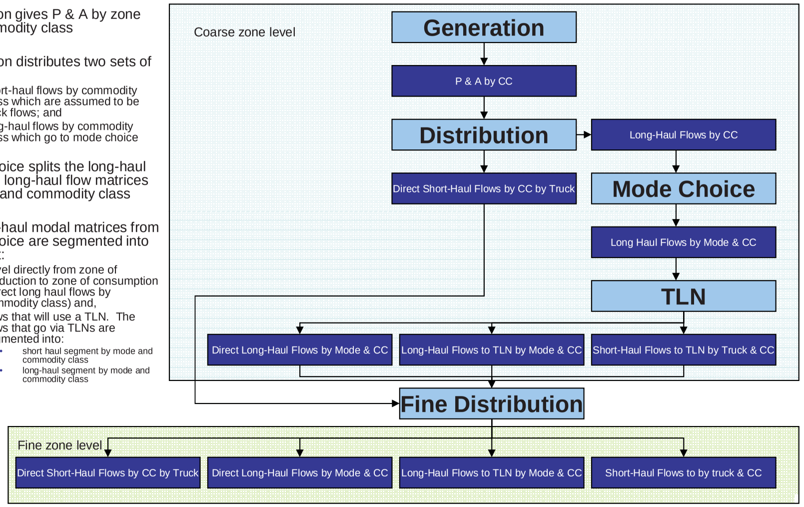
Fine Distribution Model

- Distributes models from coarse zone system to fine zone system
- The fine zones are smaller and nested under a coarse zone. These flows are distributed to the fine zones encompassed by the coarse zone using:
  - a weight to establish a small sub-matrix of the fine zone matrix based on parameters and fine zone level zonal data.
  - and a gravity model using, distance as the impedance, to infill the individual cell values.
  - It is possible to override these models to represent particular points in the system.



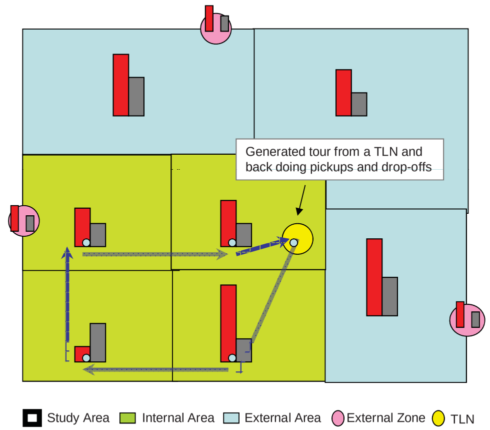
From flows to vehicle models

- Generation gives P & A by zone and commodity class
- Distribution distributes two sets of matrices:
  - short-haul flows by commodity class which are assumed to be truck flows; and
  - long-haul flows by commodity class which go to mode choice
- Mode Choice splits the long-haul flows into long-haul flow matrices by mode and commodity class
- The long-haul modal matrices from Mode Choice are segmented into flows that:
  - travel directly from zone of production to zone of consumption (Direct long haul flows by commodity class) and,
  - flows that will use a TLN. The flows that go via TLNs are segmented into:
    - short haul segment by mode and commodity class
    - long-haul segment by mode and commodity class



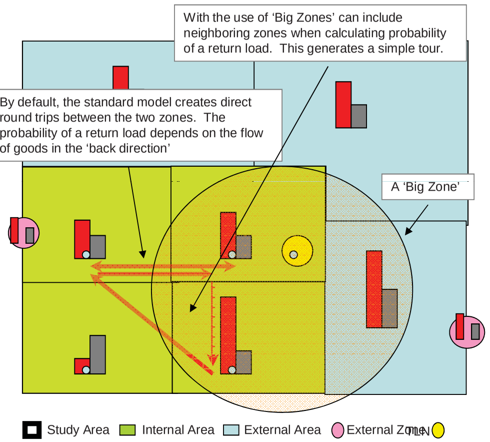
Touring Vehicle Model

- Vehicles are assumed to have the same start and end zone but make intermediate stops to load and unload.
- Heavy computations so limit use to TLN and selected zones.
- Estimates number of vehicles based at the origin using the flows from that zone and average load factors.



Standard Vehicle Model

- Model assumes that all vehicles make trips of the form A-B-A.
- Return load is a function of the commodity flow in the opposite direction.
- Can modify using 'big zones' enlarging the area considered for a return load.



Vehicle Models

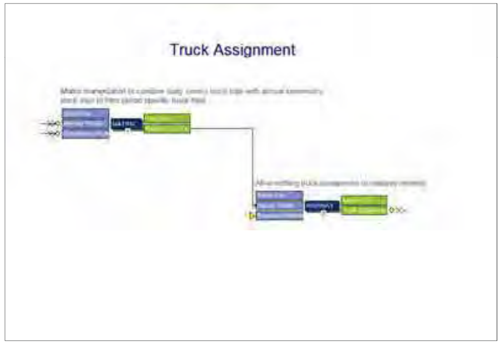
- The vehicle models can provide three truck matrices for assignment to a roadway network:
  - Heavy long-haul trucks
  - Heavy short-haul trucks
  - Light short-haul trucks
- By default, these matrices are annual truck flows since we estimate annual commodity flows. Matrix manipulation can be used to estimate daily and hourly flows by season if so desired.

Service Trips

- All modeling to this point concerns the movement of goods.
- The service model is used to estimate urban service truck trips such as:
  - Repair men (e.g. elevator repair)
  - Small shopkeeper taking goods from a wholesaler to a local restaurant..etc.
- Used directly on the fine zone system
- Estimates generation using regression models based on zone type and socioeconomic data
- Trips are distributed using gravity models.

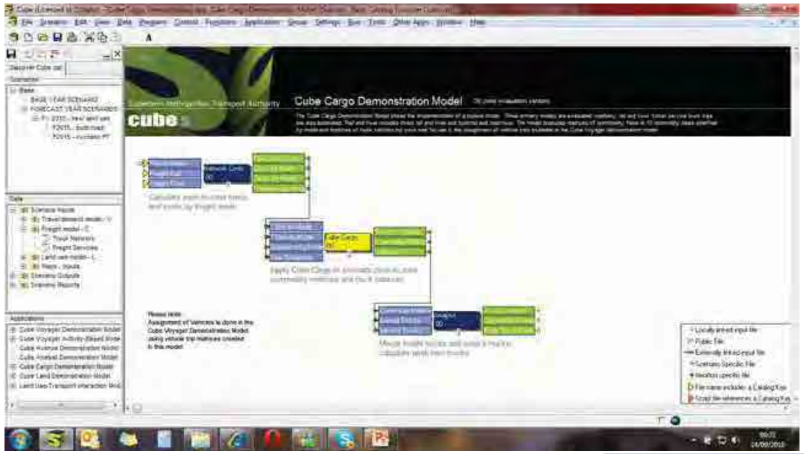
Assignment

- Matrix manipulation to estimate daily or hourly truck trips for assignment
- All of nothing assignment on cargo roadway network

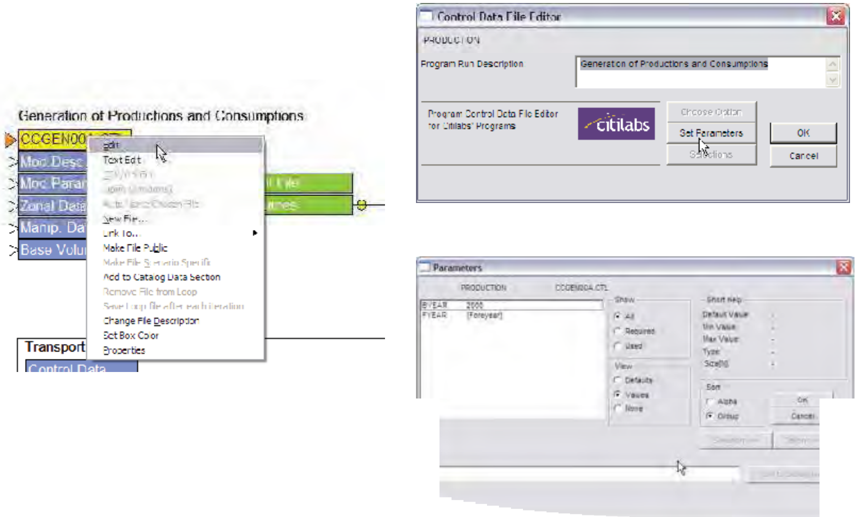


Cube Cargo – Model Interfaces

- Model structure set up in Cube's Application Manager



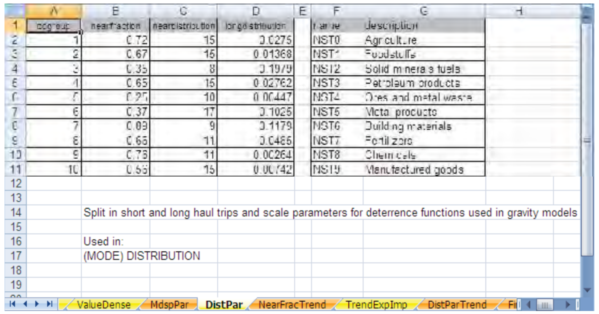
Cube Cargo – Model Data and Parameters



cube 5

Cube Cargo - Model Data and Parameters

- MS Excel tables used to store & input model parameters, options, and system information
- Cargo expects multiple sheets in each Excel file
- Required column labels must be in first worksheet row



	nearFraction	nearDistribution	longDistribution	Name	Description
1	0.72	15	0.0275	NST0	Agriculture
2	0.67	16	0.01368	NST1	Foodstuffs
3	0.35	8	0.1979	NST2	Solid mineral fuels
4	0.65	16	0.02762	NST3	Petroleum products
5	0.25	10	0.00447	NST4	Ores and metal waste
6	0.37	17	0.1026	NST5	Metal products
7	0.89	9	0.1179	NST6	Building materials
8	0.65	11	0.0486	NST7	Fertilizers
9	0.73	11	0.00264	NST8	Chemicals
10	0.55	15	0.00742	NST9	Manufactured goods

Split in short and long haul trips and scale parameters for deterrence functions used in gravity models

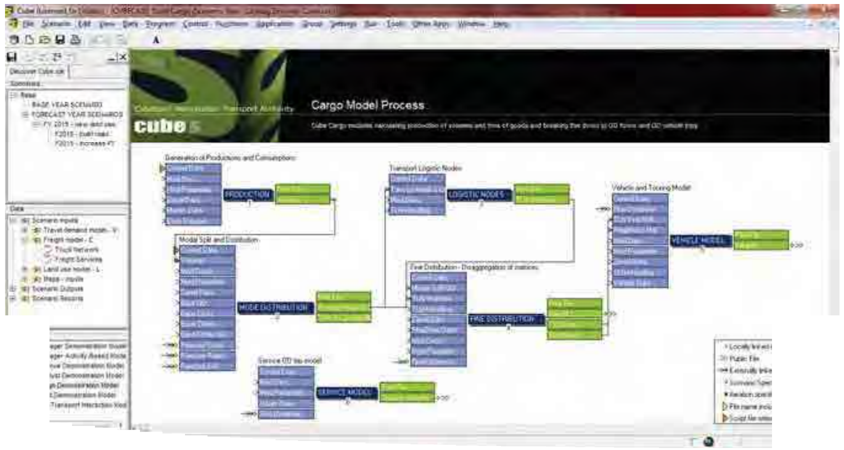
Used in:
(MODE) DISTRIBUTION

citilabs

cube 5

Cube Cargo – Detailed Descriptions

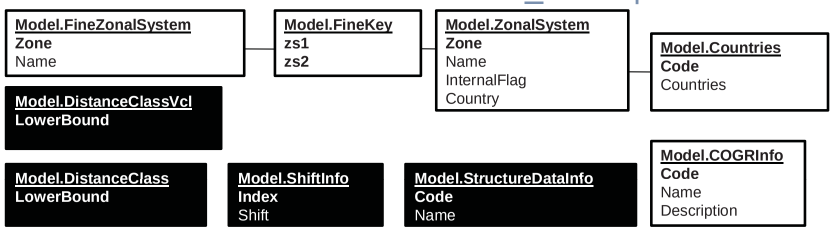
- Methodology and data inputs for the Cargo programs



citilabs

Cube Cargo – Data in MS Excel files

- MAIN MODEL DEFINITION – Model_Descriptions.XLS



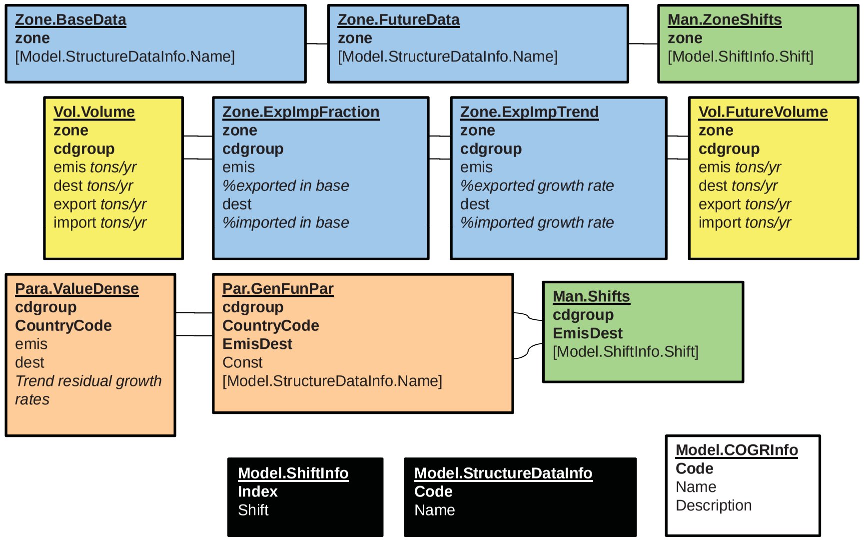
- XLS file inputs to Cargo programs
  - MODEL_DESCRIPTIONS.XLS - Inputs to ALL CARGO PROGRAMS
  - MODEL_PARAMETERS.XLS - Inputs to ALL CARGO PROGRAMS
  - ZONAL_DATA.XLS - Inputs to ALL CARGO PROGRAMS except vehicle and service models
  - MANIPULATION_DATA.XLS - Inputs to PRODUCTION
  - BASE_VOLUMES.XLS - Inputs to PRODUCTION
  - TLN_HANDLING.XLS - Inputs to LOGISTIC NODES
  - FINEZONE_DATA.XLS - Inputs to FINE DISTRIBUTION
  - VEHICLE.XLS - Inputs to VEHICLE MODEL
  - URBAN.XLS - Inputs to SERVICE MODEL

Cargo PRODUCTION - Estimating Base Trip Ends

$$T_{cej} = \left(k_{cel[j]} + \sum_{v} D_{cejv} P_{cel[j]v} + \sum_{s} Z_{cejs} \right) \left(1 + \frac{g_{ce}}{100} \right)^{y}$$

- Tcej = The initial estimate for the total trip end (tons per year) for commodity group c, direction e, and coarse internal zone j.
- kcel[j] = The constant for commodity group c, direction e, and country l[j].
- Dcejv = The value of the variable called v for commodity group c, for direction e, for zone j.
- Pcel[j]v The value of the coefficient associated the variable called v for commodity group c, for the direction e, for the country l[j].
- Zcejs = The value of the shift variable called s for commodity group c, for direction e, for zone j.
- gce = the volume growth factor for commodity group c, for direction e, for country l[j].
- y = The time, in years, from the base year to the year for which Cube Cargo is being run.

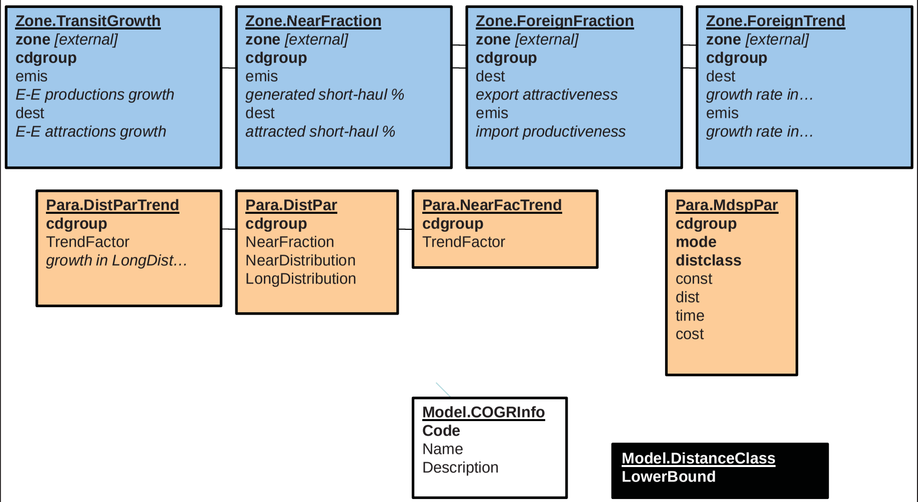
Cargo PRODUCTION - Input Data Structure



Cargo MODE DISTRIBUTION - Gravity Model

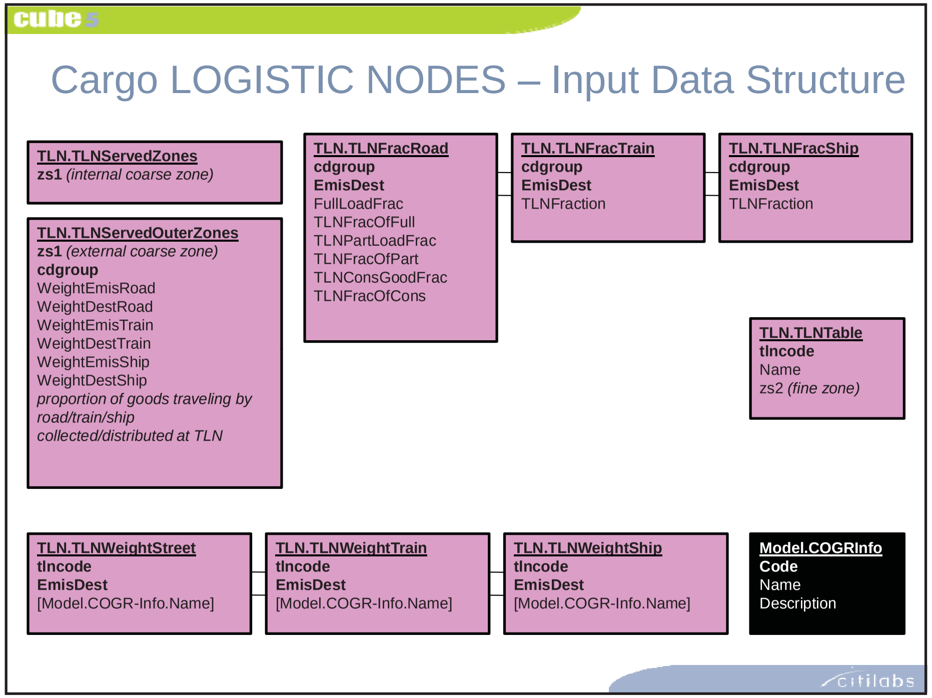
- Short-Haul Deterrence: $F_c(d) = e^{\left(\frac{-d^2}{2 p_c^2}\right)}$
  - c = A commodity group
  - e = The base of the natural logarithm function
  - d = The distance
  - Fc(d) = The deterrence function of distance used in the gravity model
  - pc = the calibration parameter for commodity c
- Long-Haul Deterrence: $\Phi_c(G_c) = e^{-P_c\left(1+\frac{\Gamma_c}{100}\right)^y G_c}$
  - Gc = The Cube Cargo generalized cost: this is the composite cost derived from the logit model used for modal split;
  - Pc = The calibration parameter for commodity c
  - Γc = growth factor for the calibration parameter for commodity c.
  - y = The time, in years, from the base year to the year for which Cube Cargo is being run.

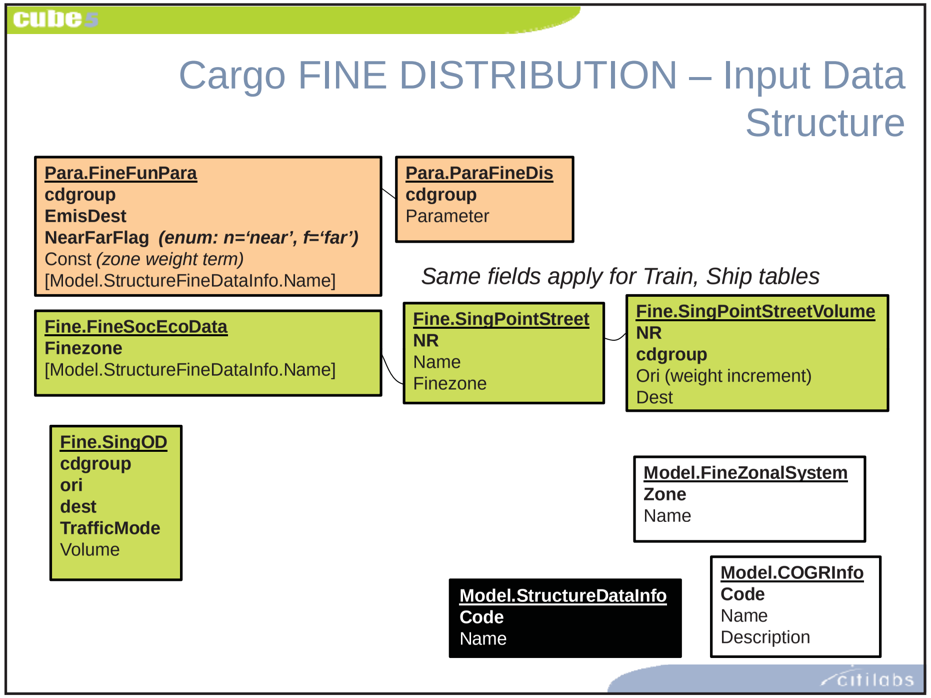
Cargo MODE DISTRIBUTION - Input Data Structure



Cargo MODE DISTRIBUTION - Mode Choice Model

- Generalized cost: $\Gamma_{cm}(d,\tau,\chi) = \kappa_{0cm} - \kappa_{1cm} d + \kappa_{2cm}\tau + \kappa_{3cm}\chi$
- c = A commodity group
- m = A mode
- d = The total journey length in miles.
- t = The total journey time in minutes.
- x = The direct monetary cost
- k0cm = The constant term from para.MdspPar.const
- k1cm = The coefficient from para.MdspPar.dist
- k2cm = The coefficient from para.MdspPar.time
- k3cm = The coefficient from para.MdspPar.cost


cube
Cargo LOGISTIC NODES – Input Data Structure
TLN.TLNServedZones
zs1 (internal coarse zone)
TLN.TLNServedOuterZones
zs1 (external coarse zone)
cdgroup
WeightEmisRoad
WeightDestRoad
WeightEmisTrain
WeightDestTrain
WeightEmisShip
WeightDestShip
proportion of goods traveling by road/train/ship collected/distributed at TLN
TLN.TLNFracRoad
cdgroup
EmisDest
FullLoadFrac
TLNFracOfFull
TLNPartLoadFrac
TLNFracOfPart
TLNConsGoodFrac
TLNFracOfCons
TLN.TLNFracTrain
cdgroup
EmisDest
TLNFraction
TLN.TLNFracShip
cdgroup
EmisDest
TLNFraction
TLN.TLNTable
tlncode
Name
zs2 (fine zone)
TLN.TLNWeightStreet
tlncode
EmisDest
[Model.COGR-Info.Name]
TLN.TLNWeightTrain
tlncode
EmisDest
[Model.COGR-Info.Name]
TLN.TLNWeightShip
tlncode
EmisDest
[Model.COGR-Info.Name]
Model.COGRInfo
Code
Name
Description
citilabs


cube
Cargo FINE DISTRIBUTION – Input Data Structure
Para.FineFunPara
cdgroup
EmisDest
NearFarFlag (enum: n='near', f='far')
Const (zone weight term)
[Model.StructureFineDataInfo.Name]
Para.ParaFineDis
cdgroup
Parameter
Same fields apply for Train, Ship tables
Fine.FineSocEcoData
Finezone
[Model.StructureFineDataInfo.Name]
Fine.SingPointStreet
NR
Name
Finezone
Fine.SingPointStreetVolume
NR
cdgroup
Ori (weight increment)
Dest
Fine.SingOD
cdgroup
ori
dest
TrafficMode
Volume
Model.FineZonalSystem
Zone
Name
Model.StructureDataInfo
Code
Name
Model.COGRInfo
Code
Name
Description
citilabs

citilabs
TRANSPORTATION MODELING SOFTWARE

Cube Analyst
Matrix Estimation

Agenda

- Introduction
- Cube Analyst Program
- (Mathematical Background)
- Calibration of Estimation Process

Cube Analyst - Introduction

- Cube Analyst allows user to update an “old matrix” using several (traffic counts, etc.).
- Takes the adjacent inputs with corresponding confidences to produce an OD matrix that best fits the input observations using a maximum likelihood technique, coupled with an optimization procedure.
- Cube Analyst estimates one matrix at a time, therefore your inputs must be consistent with the modelling period that you are trying to estimate.

Cube Analyst - Introduction

- Input Data:
  - A prior (existing) trip matrix
  - Traffic generations and attractions of zones
  - Traffic counts on links
  - Turning Count Movements
  - Modelled (multiple) paths between zones
  - Cost of travel between zones
  - Parameters of a calibrated trip distribution function
- Output Data:
  - The estimated O-D matrix.
  - Summary reports on differences between input data and corresponding values implied by the estimated matrix.
  - A set of files with information on : *Model Parameter values*
    A log of the optimization steps
    Internal Gradient Search and Intercept data

Cube Analyst - Estimation Process

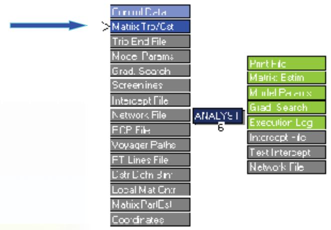
Cube Analyst - Input Data

Input data	Required
Trip Ends	Yes/no
Prior matrix	Yes/no
Costs matrix	Yes/no
Screenlines	Always
Paths (or IPC File)	Always

Cube Analyst - Data Preparation

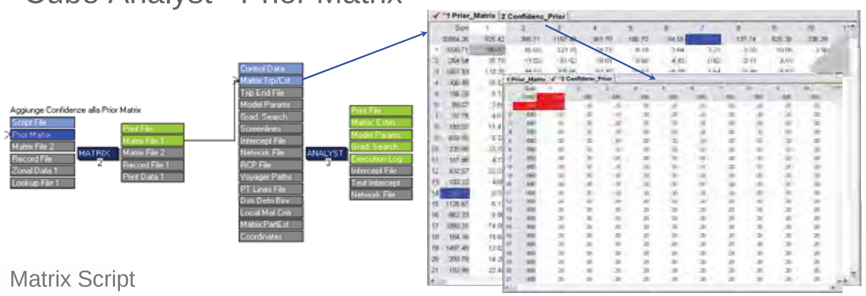
- INPUT Files to be developed in Voyager:
  - Existing Matrix (“Prior Matrix”) with Confidence Levels (using MATRIX).
  - Cost Matrix (using HIGHWAY).
  - Trip Ends File (using MATRIX).
  - Screenlines File (using HIGHWAY/PT or MATRIX).
  - Voyager Path File (using HIGHWAY) - Analyst can use this file to generate its own intercept file, storing routes for OD pairs.
  - Intercept File (using HIGHWAY/PT) - Stores routes for OD pairs that cross each screenlines.

Cube Analyst - Prior Matrix



- It is the “old” trips matrix.
- It is one of the most important input data for the estimation process.

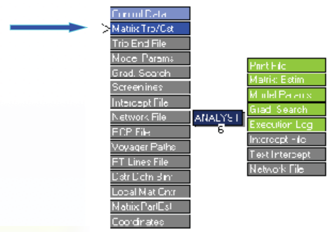
Cube Analyst - Prior Matrix



Matrix Script

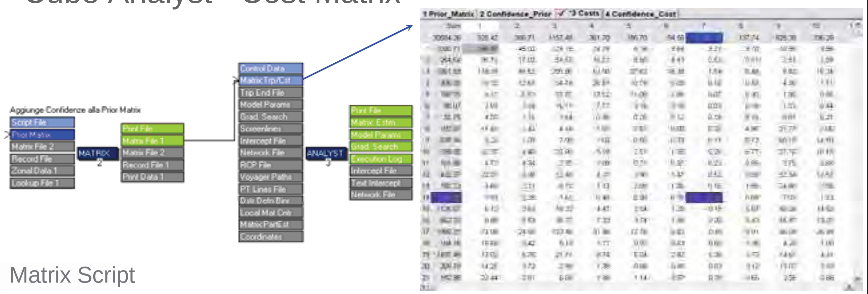
```
; Do not change filenames or add or remove FILEI/FILEO statements using an editor. Use Cube/Application Ma
RUN PGM=MATRIX PRNFILE="C:\DATA\TRAINING\01 - CUBE TRAINING\ITALIAN\MOD_VOYAGER_BZ\CVMAT00A.PRN"
FILEI MATI[1] = "{SCENARIO_DIR}\MATRICE_{SCENARIO_CODE}.MAT"
FILEO MATO[1] = "C:\DATA\TRAINING\01 - CUBE TRAINING\ITALIAN\MOD_VOYAGER_BZ\CVMAT00A.MAT",
  MO=1-4, NAME=Prior_Matrix, Confidence_Prior, Costs, Confidence_Cost

Zones=25

MW[1] = MI.1.1  ; It saves prior matrix in the first table of the output matrix
MW[2] = 20      ; It saves prior confidences matrix in the second table of the output matrix

ENDRUN
```

Cube Analyst - Cost Matrix



- The Cost Matrix contains information about generalized costs. Each *i-j* cell contains the cost value to travel from origin *i* to destination *j*.
- If it is in input, it is in the same file of the Prior Matrix.

Cube Analyst - Cost Matrix



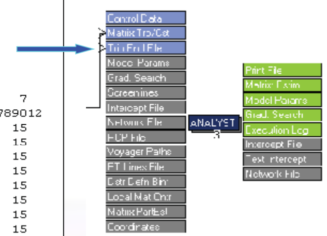
Matrix Script

```
; Do not change filenames or add or remove FILEI/FILEO statements using an editor. Use Cube/Application .
RUN PGM=MATRIX PRNFILE="C:\DATA\TRAINING\01 - CUBE TRAINING\ITALIAN\MOD_VOYAGER_BZ\CVMAT00A.PRN"
FILEI MATI[1] = "{SCENARIO_DIR}\MATRICE_{SCENARIO_CODE}.MAT"
FILEO MATO[1] = "C:\DATA\TRAINING\01 - CUBE TRAINING\ITALIAN\MOD_VOYAGER_BZ\CVMAT00A.MAT",
  MO=1-4, NAME=Prior_Matrix, Confidence_Prior, Costs, Confidence_Cost

Zones=25

MW[1] = MI.1.1   ; It saves prior matrix in the first table of the output matrix
MW[2] = 20       ; It saves prior confidences matrix in the second table of the output matrix
MW[3] = MI.2.1   ; It saves cost matrix in the third table of the output matrix
MW[4] = 5        ; It saves cost confidences matrix in the fourth table of the output matrix

ENDRUN
```

Cube Analyst - Trip Ends

- Trip Ends are the generated and attracted trips for each zones during the simulation time period.

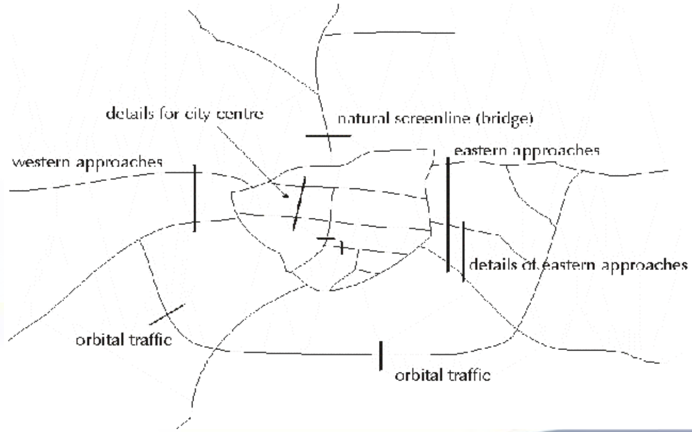
```
******************* TRIP ENDS FILE FORMAT ********************
*
* Columns      Type        Contents
*  1   10      Integer     Zone Number
* 11 - 20      Real        Trips Generated
*  6 - 50      Real        Trips Attracted
* 15 - 60      Integer     Generation Confidence Level
* 61 - 70      Integer     Attraction Confidence Level
*
*
*         1         2         3         4         5         6         7
*234567890123456789012345678901234567890123456789012345678901234567890123456789012
         1      3219         0         0      2220        15        15
         2      1940         0         0      2187        15        15
         3       682         0         0      1358        15        15
         4      3207         0         0      4186        15        15
         5      1770         0         0      3205        15        15
         6      3836         0         0      6711        15        15
         7      1794         0         0      1807        15        15
         8      2973         0         0      2121        15        15
         9      2182         0         0      2771        15        15
        10      2271         0         0      2152        15        15
        11      2431         0         0      3162        15        15
```



Cube Analyst - Traffic Count Data and Screenlines

- In addition to the Prior Matrix, Screenlines are the other important input data for the matrix estimation process.
- The traffic count sections must be located on the corridors among O-D couples.

Cube Analyst – Screenlines location

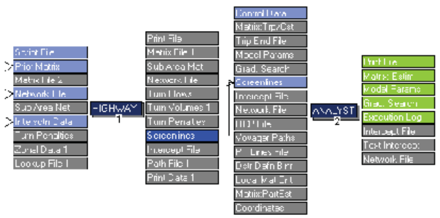


Cube Analyst – Screenlines location

- Location rules:
  - **OD covering rule** - traffic counting points on a road network should be located so that a certain portion of trips between any OD pair will be observed.
  - **Maximal flow fraction rule** - for a particular OD pair the traffic counting points on a road network should be located at the links so that the flow fraction between this OD pair out of flows on these links is as large as possible.
  - **Maximal flow-intercepting rule** - under a certain number of links to be observed the chosen links should intercept as many flows as possible.
  - **Link independence rule** - the traffic counting points should be located on the network so that the resultant traffic counts on all chosen links are not linearly dependent.

Cube Analyst - Screenlines File

- Screenlines file can be generated automatically by HIGHWAY program.



```
* File generated by HWYLOAD Wed May 16 12:50:52 2007
* Screenline Record Format
*
* Columns    Contents
*       1    S(Volume) or  T(Toll) based record
*  2 - 4     Screenline Number
*  5 - 14    Anode
* 15 - 23    Bnode
*
* If S in col 1:-
*
* 24 - 33    Link volume count
* 34 - 40    Confidence Level
*
*         1         2         3         4         5         6
*2345678901234567890123456789012345678901234567890123456789012
S   1       974       966       700      80 SL 974-966        1
S   1       967       767       605      80 SL 967-767        1
S   1       968       768      1455      80 SL 968-768        1
S   1       978       765      1270      80 SL 978-765        1
S   2       966       974       700      80 SL 966-974        1
S   2       768       968      1455      80 SL 768-968        1
S   2       764       [illegible]         [illegible]          1
S   2       767       967       605      80 SL 767-967        1
S   3       764       [illegible]         [illegible]          1
S   3       739      1238      2570      80 SL 739-1238       1
S   3       762      1213       610      80 SL 762-1213       1
S   3       741      1014       640      80 SL 741-1014       1
S   3       740      1272       390      80 SL 740-1272       1
```

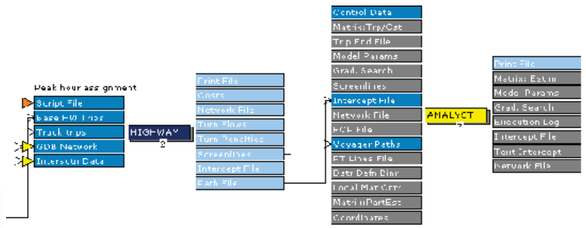
Highway Script

```
; Fase di lettura dei dati
PROCESS PHASE=LINKREAD

; Crea il file delle screenline
    IF (LI.screenline>0)
      LW.count=LI.COUNT
      LW.screenline = li.screenline
    ENDIF
```

Cube Analyst - Screenlines File

- Screenlines file can be generated by user as .DAT file (manually or using MATRIX program).

Cube Analyst – Route information (Path or ICP File)



- ICP file (both for PT and HW)
- PATH file (only for HW)
- Differences:
  - PATH FILE – whole set of routes (all the zones and user classes)
  - INTERCEPT – only the routes that cross the screenlines and by user class

Cube Analyst – Route information (Path or ICP File)

- ICP file – Script

```
FILEO ESTMICPO = "C:\Cubetown511\MODEL\ASHWI01A.ICP",
   var=lw.count, screenline=li.screenline
...
   PATHLOAD ... ESTMO=T
```

- PATH file - Script

```
   PATHLOAD ...
            PATHO=1, NAME='SOV_Paths',ALLJ=T,
            INCLUDECOSTS=T,PENI=1, ESTMO=T
```

Cube Analyst - Confidence Level

- "Confidence level" is a weighted value for input data used to estimate new trips matrix.
- It depends on reliability and variability of data:
  - Inconsistencies in what the different data suggest that the estimated matrix should be.
  - The inherent variability means that collected data items are merely a sample, and hence the values, (even of simple traffic counts) may only be considered to fall within a range (a distribution). The width of this range is a reflection of the confidence that may be placed in particular items.
  - ...
- Confidence level is a weight factor for each element.

citilabs
TRANSPORTATION MODELING SOFTWARE

Cube Analyst - Confidence Level

- Define confidence levels properly:
  - Asking: How old is the prior matrix? What is the level of sample from the surveys?
  - Create a classification of relevance.
  - Starting to apply a set of general values.
  - Analyze results and decide.
  - …

Cube Analyst - Estimation Process

- The estimation model is given by following equation:

$$\mathbf{T_{ij}} = a_i b_j \mathbf{t_{ij}} \prod_K X_K^{R_{ijk}}$$

T_{ij} Estimated Trips from origin *i* to destination *j*
t_{ij} Prior Trips from origin *i* to destination *j*
R_{ijk} Probability of trips between zones *i* and *j*
a_i, b_j, X_K Model Parameters

Cube Analyst - Estimation Process

- If there is no prior observation for movements between some O-D (and if there is an input cost matrix), t_{ij} may be calculated by:

$$t_{ij} = c_{ij}^{\alpha} e^{-\beta c_{ij}}$$

t_{ij}	Prior Trips from origin *i* to destination *j*
c_{ij}	Generalized cost of travel between zones *i* and *j*
α, β	Model Parameters

NOTE: The equation above is borrowed from the gravity model that makes the behavioural assumption that people prefer lower cost journeys to cost ones, but are influenced by level of trips generated by and attracted to different zones. (It is not a rigorous approach but it may be used where no other source of prior matrix data is available.

Cube Analyst - Estimation Process

- The main features of the Cube Analyst calculations are:
  - *Optimization* – Changing the parameters of the estimation model to optimize them
  - *Evaluation* – Defining if the result is the best

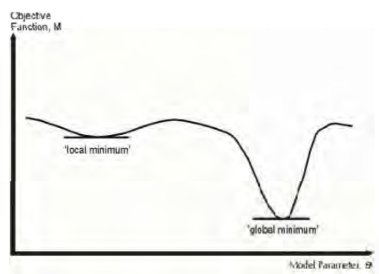
Maximum likelihood objective function:

$$M = \sum_{H} \lambda_H H - \lambda_H H \log(\lambda_H h)$$

h	Estimated data
H	Observed data
λ_{ij}	Confidence level associated with *H*

Cube Analyst - Estimation Process

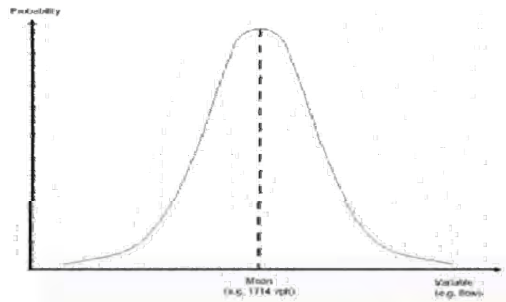
- Maximum likelihood theory shows that the most likely values are indicated when M, which is negative, reaches its minimum possible value. (For reason of computational convenience, Cube Analyst minimizes the negative of the "log-likelihood" objective function, rather than maximizing the positive version, as the name "maximum likelihood" might suggest.)



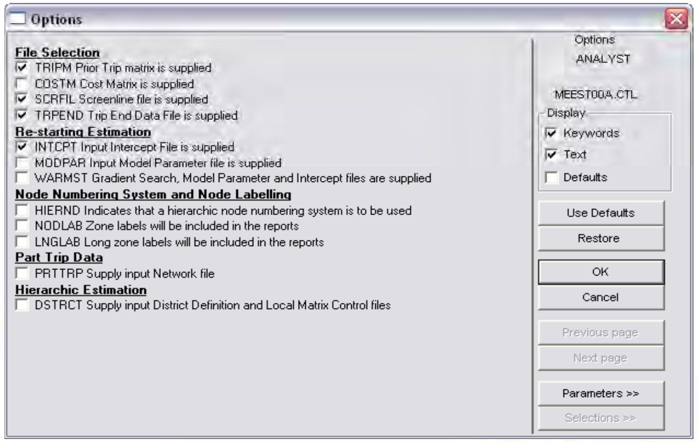
$$\frac{\partial M}{\partial \vartheta} = 0$$

Cube Analyst - Estimation Process

- The maximum likelihood method assumes that each item of input data represents an observation from a random distribution of possible values, but where the variation of values may be described by a probability distribution function (Poisson Probability Distribution Function).



Cube Analyst - Options



Cube Analyst - Standard user control Parameters

- TABLES — The input matrix numbers to be used.
  They are respectively the prior trip matrix and confidence levels, and the cost matrix and confidence levels.
  (example: *TABLES=101,102,0,0*)
- PSETS — Applies only when a VOYAGER path file is input.
  It defines the path sets to apply when building the intercepts for the screenlines. (example: *PSET=1*)
- PVOLS — Applies only when a VOYAGER path file is input.
  It defines the volumes to apply when building the intercepts for the screenline. (example: *PVOLS=1*)
- WIDEND — Specifies the format of the Screenline File. (0 = automatic)
- MFORM — Indicates the format of the output matrix. (0 = as in input)
- DEC — Defines the precision for storing values in output matrix.
- ...

Cube Analyst - Secondary user control Parameters

- MAXITERS The maximum number of iterations (*Default = 3000*)
- UTOL The accuracy tolerance in detecting convergence or failure. (*Default = 0.0001*)
- ITERH The number of iterations between recalculations of the estimated Hessian matrix. (Default = generated by ME)
- IHTYPE Type of optimization process used by Cube Analyst

Methods of optimization

Optimization process	Value of IHTYPE	Comments
Steepest Descent	0	"Simple searching"
Quasi Newton	1,2,3	1 – HJ set to unit matrix 2 – HJ read from file (warm start) 3 – HJ computed every iteration
Newton/Hybrid Newton	4	Hessian calculated regularly, according to setting of ITERH

- IREP Reporting level for the optimization log file
- ...

Cube Analyst – Calibration (Analyzing Reports)

- Average Confidence Levels

```
AVERAGE CONFIDENCE LEVELS (EXCLUDING ZERO VALUES)
-------------------------------------------------

                                       Average  Maximum  Minimum  Number of
                                                                  Elements
Trip matrix confidence levels            22.7     30.0     20.0      490
Screen line confidence levels            87.7    120.0     80.0       26
Trip end (dest) confidence levels        31.6     50.0     30.0       25
Trip end (orig) confidence levels        30.4     35.0     30.0       25
```

- Prior/Estimated Matrix Totals

```
   Stima Matriciale

REPORTING PRIOR/ESTIMATED MATRIX TOTALS

    CONFIDENCE      PRIOR   ESTIMATED  ESTM-PRIOR  (ESTM-PRIOR)/PRIOR(%)
          22.7    40261.6     47628.9      7367.2         18.3%
```

Cube Analyst – Calibration (Analyzing Reports)

- Observed/Estimated Generations

REPORTING OBSERVED/ESTIMATED GENERATIONS AND ATTRACTIONS

	GENERATIONS				
ZONE NO	CONFIDENCE	OBSERVED	ESTIMATED	ESTM-OBSV	(ESTM-OBSV)/OBSV(%)
1	35.0	2456.5	2357.3	-99.3	-4.0%
2	35.0	1549.8	1495.3	-54.4	-3.5%
3	30.0	682.3	668.3	-14.0	-2.0%
4	30.0	2745.9	3271.8	526.0	19.2%
5	30.0	1118.3	1282.3	161.0	14.7%

- Observed/Estimated Attractions

	ATTRACTIONS				
ZONE NO	CONFIDENCE	OBSERVED	ESTIMATED	ESTM-OBSV	(ESTM-OBSV)/OBSV(%)
1	30.0	1836.3	1849.1	12.8	0.7%
2	30.0	1767.9	1792.2	24.3	1.4%
3	30.0	646.7	567.7	-79.0	-12.2%
4	30.0	3387.1	3239.0	-148.1	-4.4%
5	50.0	2323.3	2416.2	92.8	4.0%

Cube Analyst – Calibration (Analyzing Reports)

- Screenlines Report

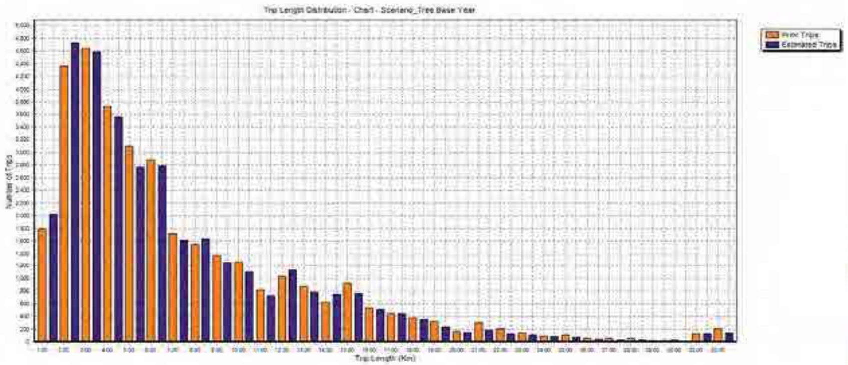
Stima Matriciale

REPORTING OBSERVED/ESTIMATED SCREEN LINE COUNTS

SCREENLINE NO & NAME	CONFIDENCE	OBSERVED	ESTIMATED	ESTM-OBSV	OBSV(%)	NO OF ODs
1 SL 974-966	120.0	4030.0	3831.9	-198.1	-4.9%	78
2 SL 966-974	120.0	4280.0	4090.5	-189.5	-4.4%	65
3 SL 764-772	120.0	5160.0	4814.4	-345.[illegible]	-6.7%	79
4 SL 772-764	120.0	5160.0	5278.2	118.2	2.3%	105
5 SL 717-988	120.0	2205.0	2688.5	483.5	21.9%	72
6 SL 988 717	30.0	2565.0	2895.3	330.3	12.9%	68
7 SL 786-790	30.0	2595.0	3079.1	484.1	18.7%	71

Cube Analyst – Calibration (Analyzing Reports)

- Check the structure of the matrix – usual TLD (trip length distribution)



Cube Analyst – Calibration (Analyzing Reports)

- Guidelines on acceptability

Criteria and Measures	Acceptability Guideline
1. **Assigned hourly flows compared with observed flows** • For flows < 700 vph – Individual flows within 100 vph; • For flows 700 – 2700 vph - Individual flows within 15%; • For flows > 2700 vph – Individual flows within 400 vph.	85% of all cases
2. Total screenline flows (normally > 5 links) to be within 5%.	All (or nearly all) screenlines
3. GEH statistic • GEH < 5 for individual flow; • GEH < 4 for screenline.	• 85% of cases • All (or nearly all) screenlines
4. **Modelled journey times compared with observed times.** Times within 15% or 1 min (if higher)	> 85% of routes

citilabs
TRANSPORTATION MODELING SOFTWARE

cube s

Dynamic Traffic Assignment with Cube Avenue

citilabs

cube s

Agenda

- Transportation Modelling and Cube Avenue

- Basic Principles of Cube Avenue
- Exercise: Dynamic Avenue Assignment Model

- Visualization and Analysis Tools
- Packet Log Analysis Techniques
- Equilibrium Methods for Large Urban Models

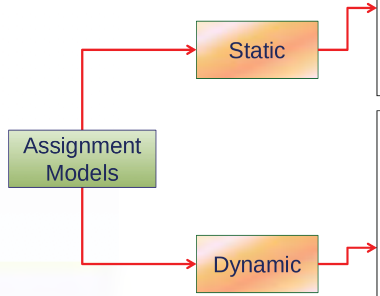
- Conclusion and news in Cube Avenue 5.1.0

citilabs

Transportation Modelling and Cube Avenue

Dynamic Traffic Assignment with Cube Avenue

Introduction



During the “model period” the following variables are constant and do not change:

- Origin-destination flows (travel demand)
- Routing and path proportions
- Link flows (vehicle volumes)
- Link costs (congested times)
- Path costs (origin-destination skims)

These models process time-varying inputs and outputs.

- Inputs:
  - Time-varying origin-destination travel demand (flow per time segment)
  - Average link costs by time segment
  - Capacities (max flow/period by segment)
- Outputs:
  - Dynamic path/link flows (total entering vehicles by time segment) and path/link costs
  - Simulation-based record of actual trajectories

Introduction

Assignment Models → Macroscopic / Mesoscopic / Microscopic

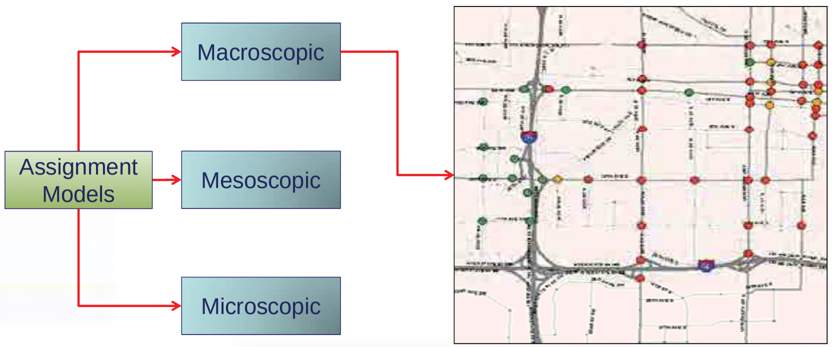
- Generally macroscopic models are used for “strategic” planning (medium and large areas)
- Macroscopic models consider the entire system and estimate routing and flows through a network for a time period
- Assumption 1: traffic congestion can be described by the static relationship between fundamental variables such as flow & speed
- Assumption 2: stable traffic flows between Origin and Destination
- Compute (1) minimum-cost path set for the traffic volume
- Compute (2) congestion effects (through volume-capacity ratios and resulting speeds)
- Fast iterative analysis for very large areas.

Introduction

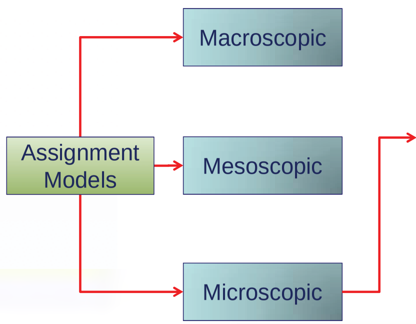
Assignment Models → Macroscopic / Mesoscopic / Microscopic

Traffic Flows

Introduction



Introduction



- Generally microscopic models are used to study infrastructure geometry, traffic control system, etc.
- Vehicles are analyzed individually, studying driver behavior and interactions

⬇

- Microscopic models are speciialized tools for detailed studies
- Detailed results require extremely detailed input data → microscopic models are used for small areas or corridors

Introduction

Assignment Models → Macroscopic
Assignment Models → Mesoscopic
Assignment Models → Microscopic →

Introduction

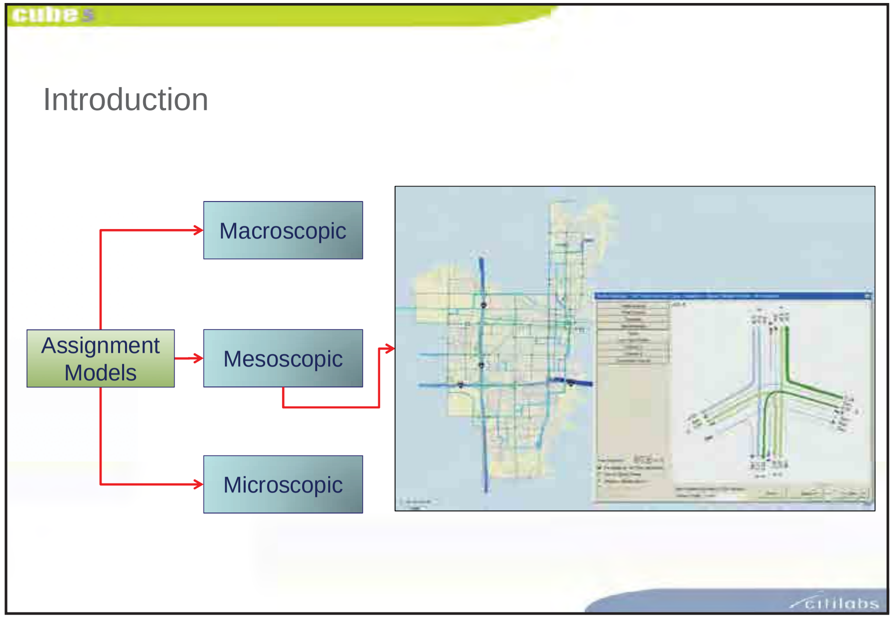
Assignment Models → Macroscopic
Assignment Models → Mesoscopic →
Assignment Models → Microscopic

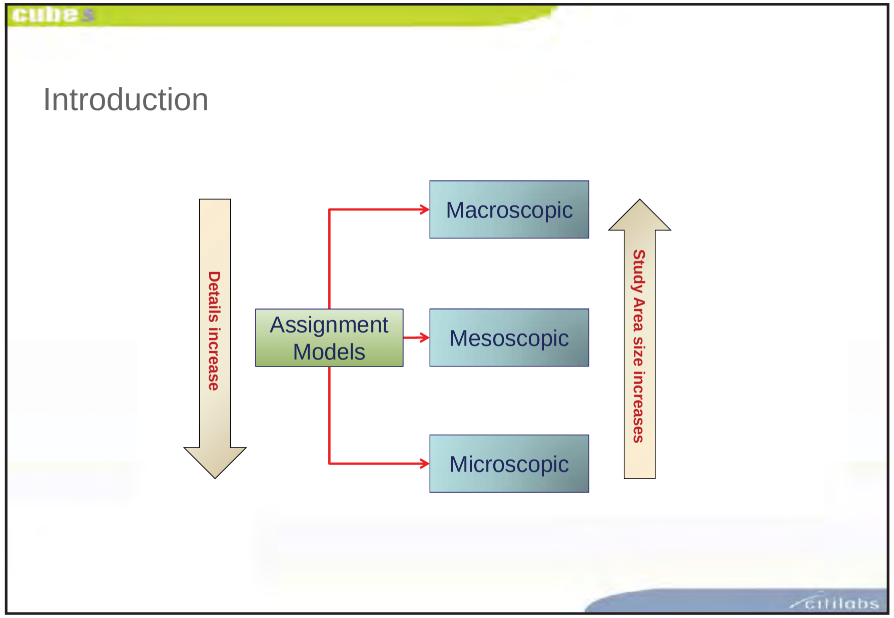
- Mesoscopic models try to find a middle group between macro and micro models
- Vehicles are analysed as "packets" of vehicles by studying fundamental variables (flow, speed, density)
- Mesoscopic models techniques can study traffic flows over time (Dynamic)

⬇

- Mesoscopic models the lowest-cost path for the traffic volume for each packet of vehicles
- Mesoscopic models compute congestion effect, through volume-capacity rations and also interaction among vehicles units ("packets of vehicles")

citilabs
TRANSPORTATION MODELING SOFTWARE





Introduction

- Macro-meso-micro methods can be most easily distinguished by how they represent *flow* and evaluate *congestion*
- Flow can be either continuous (streams) or discrete (vehicles/packets)
- Performance functions can be either aggregate (evaluated for a whole time interval) or disaggregate (evaluated for individual flow quanta)

Typology of assignment models

		Performance functions	
		Aggregate	*Disaggregate*
Flow Representation	*Continuous*	MACRO	N/A
	Discrete	MESO	MICRO

Transportation modelling tools

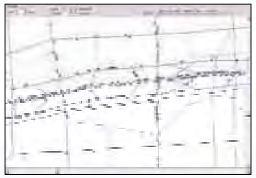
Macroscopic and Static Modelling — cube voyager

Mesoscopic and Dynamic Modelling — cube avenue

Microscopic and Dynamic Modelling — cube dynasim

What is Cube Avenue?

- Avenue is an optional add-on to Cube Voyager that enables *Dynamic Traffic Assignment* with *Mesoscopic Simulation*:
  - ***Dynamic Traffic Assignment***
    Routes and flow rates change during the model period based upon congested costs
  - ***Mesoscopic Simulation***
    Vehicles are grouped into homogenous "packets" and simulated as they move through the network

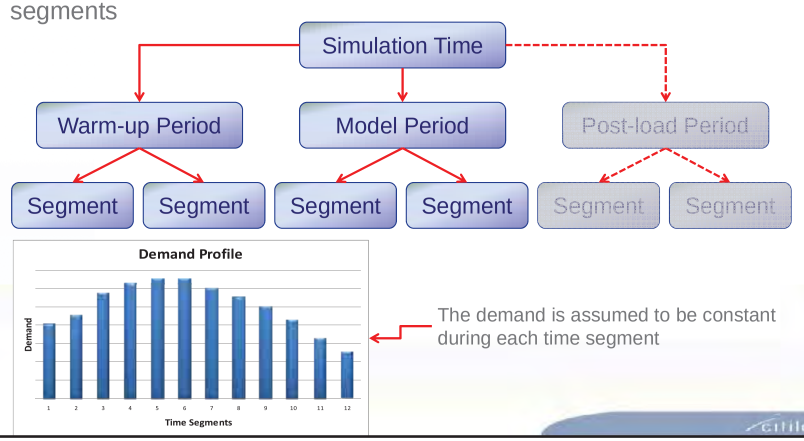


Cube Avenue – Dynamic Traffic Assignment (DTA)

- Method of system-level assignment analysis which seeks to track the progress of a trip through the network over time
- Accounts for formation and propagation of queues due to congestion
- A bridge between traditional region-level static assignment and corridor-level (micro-simulation)

Cube Avenue – Dynamic Traffic Assignment (DTA)

- The model duration is explicitly defined and divided into smaller time segments



Cube Avenue – Mesoscopic Modelling

- With mesoscopic models, it is still possible to quickly analyze larger areas with a more detailed model which overcomes the pitfalls of the macroscopic travel demand models.
  - Takes into account intersection configurations and controls
  - More detailed estimates of delay, travel time, and capacities
  - Enforces capacity limitations and the effects of queues 'blocking back'
  - Models flow curves and changing demand throughout an analysis period
  - Allows vehicles to respond to traffic conditions and change their route

Cube Avenue – Input Data

- Time-varying O-D travel demand (flow per time segment)
- Link properties (Capacity, T0, Storage, Junction Modelling, ...)

Time Segment N

	D_1	D_i	D_z
O_1	$T_{1,1}$	$T_{1,i}$	$T_{1,z}$
O_i	$T_{i,1}$	$T_{i,i}$	$T_{i,z}$
O_z	$T_{z,1}$	$T_{z,i}$	$T_{z,z}$

Cube Avenue – Output Data

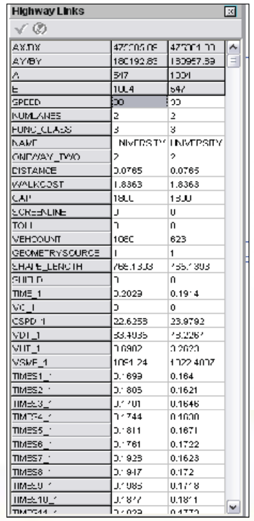
- Dynamic bandwidth:
  - Flows
  - Queues
  - Vol/Cap ratio
  - ...
- Dynamic data at junctions:
  - LOS and delays
  - Queues
  - Turning volumes
  - ...

Cube Avenue – Output Data

- Dynamic bandwidth:
  - Flows
  - Queues
  - Vol/Cap ratio
  - ...
- Dynamic data at junctions:
  - LOS and delays
  - Queues
  - Turning volumes
  - ...
- Packets animation

Cube Avenue – Output Data

- Dynamic bandwidth:
  - Flows
  - Queues
  - Vol/Cap ratio
  - ...
- Dynamic data at junctions:
  - LOS and delays
  - Queues
  - Turning volumes
  - ...
- Packets animation
- Numerical data regarding all these outputs



Cube Avenue – Mesoscopic Modelling Applications

- A mesoscopic model allows to complete new types of analyses:
  - Quantify impact of upstream traffic congestion
  - Measure queuing at intersection and merge points in a network
  - Isolate secondary impacts from one intersection through another
  - Evaluate the benefits of ITS (Intelligent Transportation System) projects
  - Simulate alternative infrastructure, operational and policy changes to optimise
  - Emergency evacuation plans and strategies
  - Test strategies to improve arrival and departure from stadiums and other special event facilities
  - ...

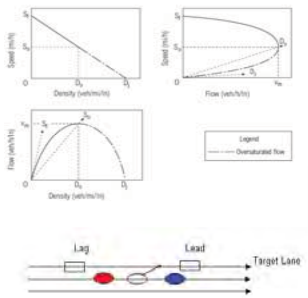
Basic Principles of Cube Avenue

Dynamic Traffic Assignment with Cube Avenue

citilabs
TRANSPORTATION MODELING SOFTWARE

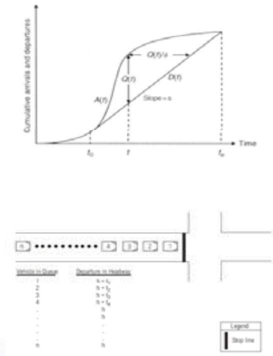
Congestion: the key to understanding model scale

- Macroscopic models typically estimate congestion using speed-flow curves – theoretically based on fundamental diagram(s) of traffic engineering
- Microscopic models simulate individual vehicle trajectories on a detailed network and use behavior models (e.g. car following, gap acceptance) to predict second-by-second driver responses to en-route events



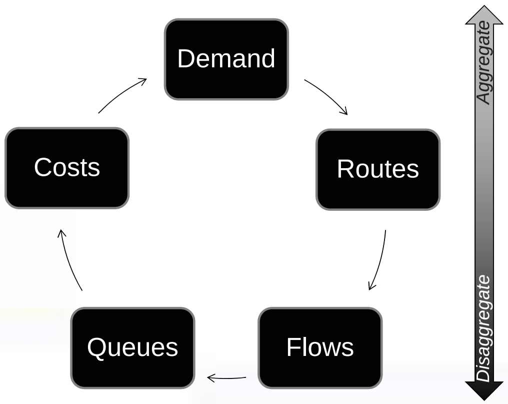
Mesoscopic Traffic Modelling

- Typically simulate movement of trips along their routes at some resolution of detail (vehicle, packet)
- Discretely model traffic *queues* in network (at intersections, ramps, tolls)
- Traffic *stream* performance is typically still evaluated using aggregate macroscopic (e.g. speed-flow) relationships
- Aggregation / dis-aggregation processes used to combine queue and stream performance measures



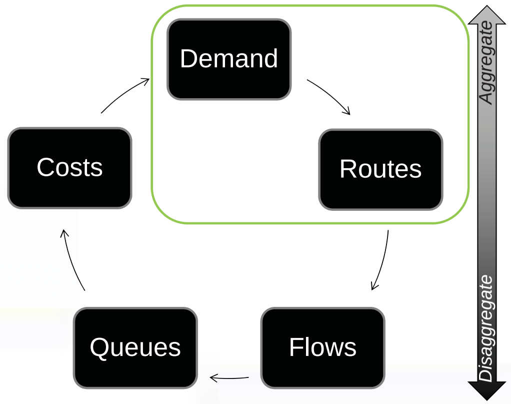
citilabs
TRANSPORTATION MODELING SOFTWARE

The temporal information cycle



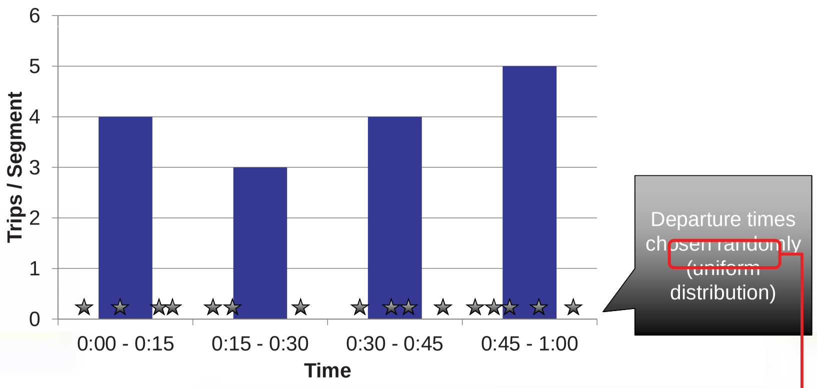
- Aggregation and disaggregation processes play a key role in meso-scopic modelling.
- As Cube Avenue processes the various kinds of information used in traffic assignment, it moves between different levels of aggregation.

From trips by segment to vehicle packets with paths



- Origin-destination demand is input at the most discrete level – the segment
- Trips disaggregated to packets given random departures within the segment
- Departures grouped and assigned paths based on link costs by time segment

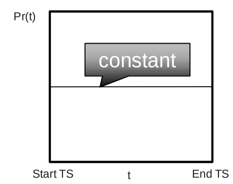
Example: Disaggregating trips to packets

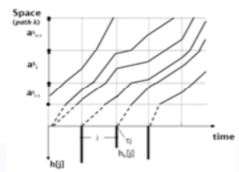


An internal random number generator randomly draws a departure time for each packet departing in a given interval.

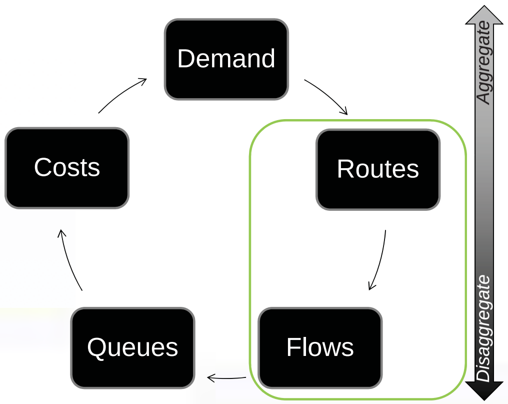
Packet routing and grouping

- Packet departure times are chosen at random from the time segment interval, assuming a uniform distribution
- Each unique departure faces a potentially distinct set of link costs (by time segment)
  - Implies that we could build a new path for every packet generated! (very expensive)
- We can reduce the number of path builds required by defining some Δt during which departures take effectively the same path
- PARAMETERS MAXPTHPERSEG:
  Upper bound on the number of path builds executed for a given time segment, iteration, and origin (Defaults to 5).





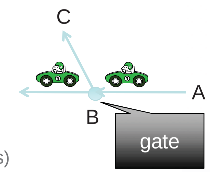
Translating packet paths into link flows



- Packet simulator moves packets along their path
- Packets traverse links according to speed-flow function output time
- Movement from one link to another is regulated by "gates"
- Packet volumes are added to link flow volumes upon entry

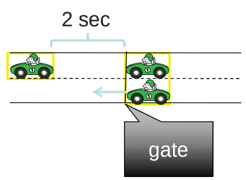
Event-based simulation

- Cube Avenue simulation processes events as they are encountered by packets moving along their paths
- Packets can be in one of two states:
  - Moving on a link
  - In queue (waiting on a link)
- A vehicle may have to wait if:
  - Cars leaving a link exceed its exit flow capacity (***Capacity*** Constraints)
  - Cars entering a link exceed its entrance flow capacity (***Capacity*** Constraints)
  - There is no room for it on the next link (***Storage*** Constraints)
- These criteria are evaluated **by A-B-C movement**
- Turn capacity is also checked if output by a junction model (intersection analysis, e.g. HCM 2000)
- Constraint is the minimum of constraints at node



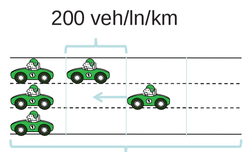
General Principles: Gates

- Capacity and storage constraints are maintained by “gates” on each link
- In practice, *minimum headway* is used rather than *maximum flow*
- Consider a 2-lane freeway link with per-lane flow capacity of 1800 vehicles per hour and total flow capacity of 3600 vehicles per hour:
  - This is equivalent to a headway (or gap) of one sec/vehicle
  - So if a packet with two vehicles arrives at the gate, it cannot leave the link any sooner than two seconds after the packet ahead of it.



General Principles: Queue Propagation

- The number of Vehicles In Transit (VIT) on a link is limited by the link storage
- VIT includes moving and queued vehicles
- VITS / STORAGE = link *occupancy*
- If there is no space remaining on a link, then entering packets will be queued
- This is referred to as a “horizontal” queue
- In practice, storage is the jam density at **minimum vehicle *spacing***
- PARAMETERS VEHPERDIST:
  Density per lane [vehicles/lane/distance]

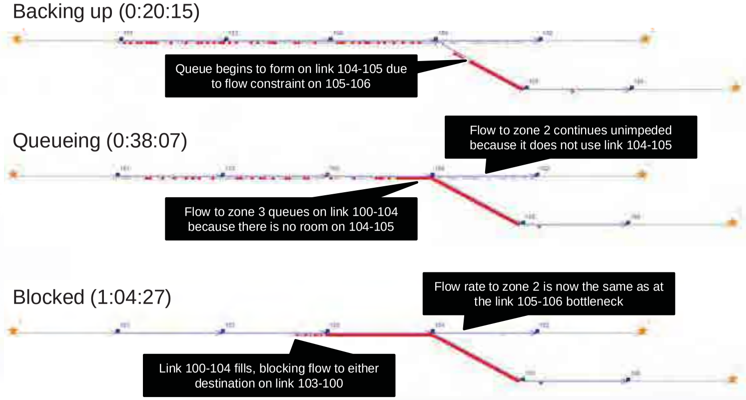


Distance = 0.01 km
Lanes = 3
Storage = 6 vehicles

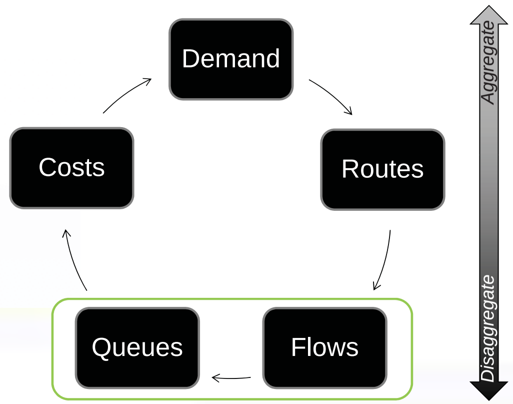
Backing up (0:20:15)

Queueing (0:38:07)

Blocked (1:04:27)

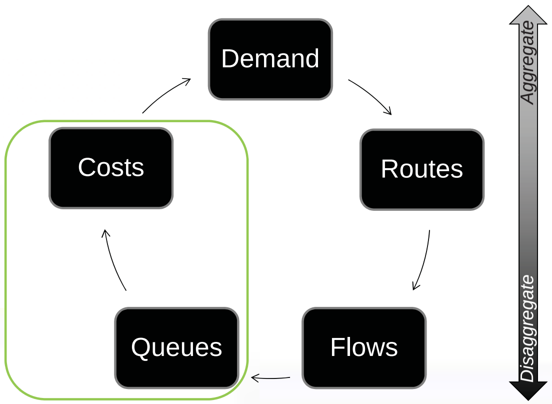


Queues: Expected vs. excess time



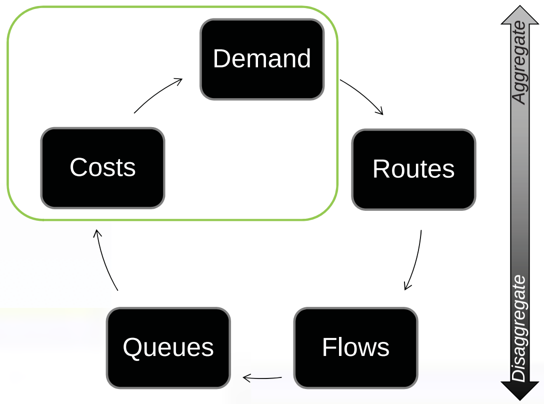
- The simulator tracks average time spent in queue, referred to as "excess" above the "expected" time given by speed-flow
- Volume-weighted average excess time by link and time segment is calculated after simulation finishes

Re-estimating link costs by time segment



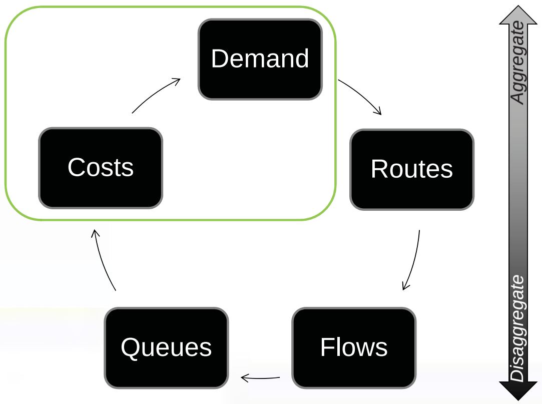
- The accumulated total link entry flows provide V for speed-flow function (TC[])
  - Factored to model period because C is in units of flow/MP
  - Factor = MP / TS
- Expected excess (queue) time is added to estimated time in flow stream
- Generalized COST function applied by time segment

Link-based cost feedback and demand scaling



- Congested link costs are used to build auxiliary paths for new iterations
- Once generated, a packet persists beyond its iteration and is re-simulated
- Flow-based MSA: assign a "copy" of initial packet set with 1/n flow scaling to paths

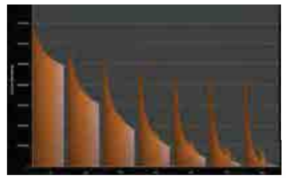
Route and departure time choice models (Optional)



- Path costs (skims) can also be extracted using TRACE() function
- Can extract actual simulated path costs from log file
- Behavioral route and departure time choice models follow readily

Dynamic Traffic Assignment Modeling Options

- Equilibrium – usually via path-based MSA
  - Flow-based implementation (COMBINE=AVE)
  - Variations… search, "k-worst", MSWA, "k-reset"
- Dynamic process models
  - Represents process of "learning" link costs
  - Within-day or between-days
- Incremental assignment
  - Conventional: divide trips into "fractions" and accumulate via successive loading
  - By-time: treat time segments as increments



Specific Output Data of Cube Avenue

- **TIME_1**: average travel time on a link during the model period
- **TIMES*t*_1**: average travel time on a link during the time segment ***t***
- **CSPD_1**: average travel speed on a link during the model period
- **SPEEDS*t*_1**: average travel speed on a link during the time segment ***t***
- **V*f*SMP_1**: volume of traffic entering a link (volume field ***f***) during the simulation period
- **V*f*S*t*_1**: volume of traffic entering a link (volume field ***f***) during the time segment ***t***
- **VSMP_1**: result of applying the V function to V1SMP_1, V1SMP_1, …
- **VS*t*_1**: result of applying the V function to V1S*t*_1, V2S*t*_1, …
  - Note: VS <= C (+ 2*PACKETSIZE)
  - Note: The extra 2 packets are due to certain exceptions that may arise at intersections

Specific Output Data of Cube Avenue

- **VITS*t*_1**: number of vehicles in transit on a link during the time segment ***t***
  - Note: VITS <= STORAGE (+ 2*PACKETSIZE)
- **QUEUEVSt_1**: average number of vehicles queuing on a link during the time segment ***t***
  - Note: QUEUEVS <= STORAGE (+ 2*PACKETSIZE)
- **BLOCKVS*t*_1**: number of vehicles in the queue in the time segment ***t*** that will remain in the queue at the end of the simulation
  - Note: BLOCKVS <= QUEUEVS
  - Note: link and path flows are not necessarily conserved because the model period may end during a packet's journey…

Comparison With Static Assignment

Static Assignment

- A vehicle exists everywhere along its route during period
- Variables do not change over the duration of the period to be modelled
- Capacity constraints not strictly enforced; V/C > 1
- No link storage constraint
- Link volumes and costs are separable and independent
- Time = Link Travel Time + Junction Delay

Cube Avenue

- Simulated packets can only be in one place at a time
- Model period divided into "time segments" with varying flow rates
- Capacity strictly enforced using "flow gates"
- Storage strictly enforced
- Simulation of queues affects preceding link volume, cost
- Time = Link Travel Time + Junction Delay + Queue Time

Comparison with Micro-Simulation

Micro-Simulation

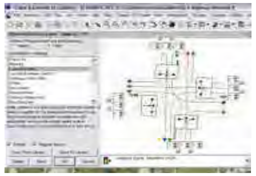
- Each vehicle is simulated individually
- Complex flow interactions like weaving and merging
- Explicit representation of facility lane geometry
- Produces 3D animations of output results
- Computationally intensive

Cube Avenue

- Vehicles can be grouped into homogenous packets
- Uses aggregate speed/flow relationships
- Run using unaltered regional model networks
- 2D maps & animations possible in Cube
- Much shorter run times

Relationship to Other Cube Features

- Junction modeling
  - Separate feature; it is not necessary to implement both
  - However, because Avenue strictly enforces capacity constraints it may improve some junction models
- GIS tools / Cube GIS
  - Allows you to ensure that link distances are based upon actual feature geometry (instead of arbitrary straight line links)
  - This will provide more accurate input distance values for Avenue, giving a better starting point for storage estimates
  - Therefore, using a GIS-enabled network is recommended



Scripting Dynamic Traffic Assignment With Cube Avenue

Dynamic Traffic Assignment with Cube Avenue

Dynamic Avenue Assignment Model

- Cube Avenue Parameters
- Specific Functions
- LINKREAD Phase
- ILOOP Phase
- ADJUST Phase
- Scripting Tips

Cube Avenue Parameters

- **COMBINE** → Combine type "EQUI" is not valid for Avenue. For most uses of AVENUE, "AVE" is a good choice of combine value.

Method of Successive Averages (MSA) With Packet Splitting (PS)

1) Initialization:
   - Iteration number ($n = 0$)
   - Packet Volumes (V_0) = 0
   - Link Cost ($COST_0$) for free-flow conditions
2) Update iteration number: $n = n + 1$
3) All-or-nothing assignment (F_n) in the iteration n to $COST_{n-1}$ paths based on V_{n-1}
4) Update the Packet Volumes: $V_n = V_{n-1} + \phi(F_n - V_{n-1})$, where $\phi = 1 / n$
6) Update Link Costs ($COST_n$) given V_n (based on simulation)
7) If no convergence go to step (2).

Problem Size and RAM

With PACKETS=PS,

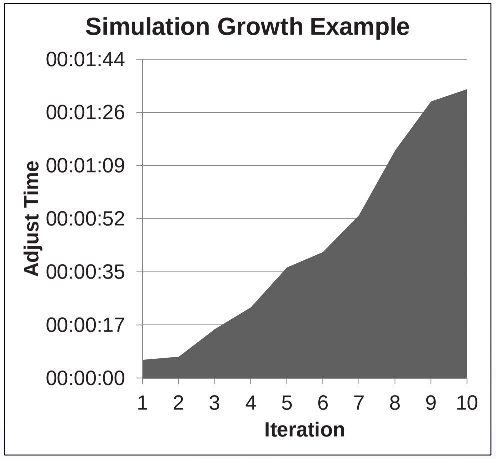
$p_n = p_1 \times n$,

where p_n is the number of packets being simulated during iteration *n*

Under extreme congestion, a large number of vehicles may remain queued in the system waiting to leave in subsequent time segments, further inflating the number of packets to be simulated

In a Windows 32-bit computing environment, any given process can address at most 2 GB of RAM.

If there are 2 million packets in the peak hour and 50-70% exiting the system during each time segment of the first iteration, it is easy to exceed the memory limitations of a typical desktop PC



Cube Avenue 5.1.0 – Additional Simulation Modes

- Avenue 5.1.0 includes enhancements to COMBINE=AVE.
- The methodology is still MSA, but just implemented differently:
  - Fixed number of packets for each iteration (this set of packets never changes; the only thing that changes is the path to which each packet is assigned)
  - These packets are allocated pseudo-randomly based on MSA probabilities for the paths generated during each iteration.
  - In other words, during each iteration, a packet is tested and has a ***1/k*** probability (where ***k*** is the iteration number) of being “moved” to the new path set, otherwise it is left on the path it was using before.
  - This results in much better memory usage due to only one set of packets being generated, allowing larger simulations with more iterations.
  - Accordingly, this new method (PACKETS=PA) is now the default!

Cube Avenue 5.1.0 – Packet Size Considerations

- Since the original PACKETS=PS simulation results in packet sizes being dropped by a factor of 1/*k* (where *k* is the iteration number), and the number of packets being a factor of *k*, it is sensible to start off with a fairly large PACKETSIZE parameter to DYNAMICLOAD. In later iterations the packet size will be PACKETSIZE/*k*, but because there is now a factor of *k* packets being simulated for the iteration, the memory requirements and time required for the simulation increase for each iteration.
- With the new PACKETS=PA simulation, only one set of packets is ever created for a time segment, and they stay the same size, but are moved between paths each iteration. After all time segments are loaded, this results in each iteration having the same number of packets as the previous iteration, meaning that simulation times are more consistent. It also means that for reasonable fidelity results, the packet size should be low, e.g. PACKETSIZE=1.

Cube Avenue 5.1.0 – Packet Size Considerations

Example:
- 100 packets
- 10 iterations
- Two available paths (path "A" and path "B")

PACKETS=PS	COMBINE=PA
In each iteration whole the packets are assigned to the best path → number of packets increases every iteration. At the 10th iteration: 100 x 6 = 600 assigned packets to path A 100 x 4 = 400 assigned packets to path B 600 + 400 = 1000 assigned packets! Packet size dropped by factor of ***1/k***: 600 x 1/10 = 60 packets volume on path A 400 x 1/10 = 40 packets volume on path B	Packets stay the same size and they are moved among paths each iteration. In each iteration there is the same number of packets as the previous iteration: 100 assigned packets! Packets are allocated based on MSA probabilities for the paths generated during each iteration: 100 x 6/10 = 60 packets volume on path A 100 x 4/10 = 40 packets volume on path B

Cube Avenue 5.1.0 – Keywords

- The new simulation mode is specified by using **COMBINE=AVE, PACKETS=PA** parameter setting (the default).
- There is a sub-key to PACKETS=PA:

  ITERLOADINC=n → specifies the iteration loading increment. This means that ***n*** iterations occur before adding the traffic specified in the next time segment.

  E.g, ITERLOADINC=3 indicates that the first time segment is loading during iteration 1, but that time segment 2 is loaded during iteration 4 and so on.

Cube Avenue Parameters

- **MODELPERIOD** → Length of the model period in minutes
- **SEGMENTS** → Time segments duration. SUM(Segments)≥Model Period
- **VEHPERDIST** → Maximum Density per Lane [vehicles/lane/distance]

```
; Do not change filenames or add or remove FILEI/FILEO statements using an editor. Use
RUN PGM=AVENUE PRNFILE="C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\APPLICATIONS\DAAVN00A.PRN"
FILEO NETO = "C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\OUTPUT\DYNAMIC_LOAD.NET"
FILEO PACKETLOG = "C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\OUTPUT\PACKET.LOG"
FILEI NETI = "C:\DATA\Cube_Models\Avenue_Seminar\Output\Network.mdb\Base"
FILEI MATI[1] = "C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\OUTPUT\DEMAND.MAT"

PARAMETERS COMBINE=AVE, GAP=0.005, MAXITERS=10
MODELPERIOD=90, SEGMENTS=15,15,15,15,15,15, ;30 min warm up
VEHPERDIST=250.00    ; average units (ft or mts) per vehicle
```

Example 1

```
PARAMETERS COMBINE=AVE, GAP=0.005, MAXITERS=10
MODELPERIOD=90, SEGMENTS=6*15, ;30 min warm up
VEHPERDIST=250.00    ; average units (ft or mts) per vehicle
```

Example 2

Cube Avenue Parameters

- **CAPLINKENTRY** → When CAPLINKENTRY=Y (default) the capacity of the link limits how quickly vehicles can enter or leave a link. When CAPLINKENTRY=N, the link capacity only limits how quickly they can leave the link. The primary difference between these two regimes is where the front of a queue occurs.
- **MAXPTHPERSEG** → Upper bound on the number of path builds executed for a given time segment, iteration, and origin (Defaults to 5).
- **PKTPTHSIZ** → Maximum number of nodes that a packet keeps in RAM. To save memory, packets can swap their route information between RAM and temporary disk files.

Specific Functions

- **RANDSEED(*n*)** → Initialize the random number generator with ***n***, where ***n*** is an integer between 1 and 2147483647 (so a repeatable series of random numbers can be generated).

Example

```
; Do not change filenames or add or remove FILEI/FILEO statements using an editor. Use
RUN PGM=AVENUE PRNFILE="C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\APPLICATIONS\DAAVN00A.PRN"
FILEO NETO = "C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\OUTPUT\DYNAMIC_LOAD.NET"
FILEO PACKETLOG = "C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\OUTPUT\PACKET.LOG"
FILEI NETI = "C:\DATA\Cube_Models\Avenue_Seminar\Output\Network.mdb\Base"
FILEI MATI[1] = "C:\DATA\CUBE_MODELS\AVENUE_SEMINAR\OUTPUT\DEMAND.MAT"


PARAMETERS COMBINE=AVE, GAP=0.005, MAXITERS=10
MODELPERIOD=90, SEGMENTS=6*15, ;30 min warm-up
VEHPERDIST=250.00   ; average units (ft or mtrs) per vehicle

x=RandSeed(123)  ; Inizialize random number generator for repeatability
```

LINKREAD Phase

- **DISTANCE** → If it is not set in script, it will initialized from LI.DISTANCE
- **LINKCLASS** → It functions as the index for functions TC and COST
- **LI.LANES** → Number of lanes
- **SPEED** → If it is not set in script, it will initialized from LI.SPEED (the defaulting behavior of SPEED is only significant if it is used to calculate a default for T0)
- **C** → Flow capacity of the link, in vehicles per model period. Scripts can use the DYNAMIC command to specify that the value of C varies by time segment
- **T0** → Free-flow time for the link (in minutes)
- **T1** → Link time to be used on the first iteration of assignment. It is very common to accept this value, although there may be some merit to using observed times where these are available
- **STORAGE** → Number of vehicles that can fill the link

LINKREAD Phase

Example

```
PROCESS PHASE=LINKREAD
   DISTANCE=LI.DIST
   IF(A<=ZONES || B<=ZONES)    ; connectors
      LINKCLASS=3
      STORAGE=999999
      C=999999
      T0=0
   ELSE
      C=LI.LANE_CAP*LI.LANES
      T0=LI.DIST/LI.SPEED*60     ; [min]
      STORAGE=250*LI.DIST*LI.LANES     ; [vehicle/km]
      IF(LI.SPEED=50)
         LINKCLASS=1
      ELSE
         LINKCLASS=2
      ENDIF
   ENDIF
ENDPROCESS
```

For Connectors:
- Endless Capacity
- Endless Storage
- T0 = 0

Example

```
...
DYNAMIC  C[3]=1000,1000,1200
```

Results in C taking the value 1800 during time segments 3, 1000 during time segment 4, 1200 during the time segment 5 and its usual value during all other time segments.

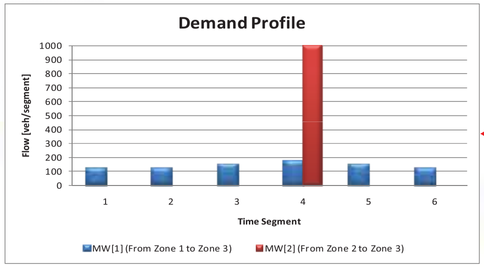
ILOOP Phase

- **DYNAMICLOAD** → DYNAMICLOAD is the dynamic analog of the static PATHLOAD statement. A conventional load (that is, a PATHLOAD statement) evaluates an expression (usually involving matrices) to determine the number of trips, builds paths according to some attribute minimization criterion, and then it loads the trips into the network's volume fields.
  - **PATH** → It may take, as its value, "TIME," "COST"
  - **PACKETSIZE** → It specifies the target number of vehicles per packet.
  - **DEMANDISHOURLY** → It determines whether the demand volumes for each time segment are supplied as an absolute value in vehicles or as a rate in vehicles per hour [by default it is FALSE]. For example, suppose that there is a segment of 15 minutes and in a cell of the matrix there is a value of 40. If DEMANDHOURLY=T, the demand is 30 veh/h and 10 vehicles depart the origin during the time segment. Otherwise, if DEMANDHOURLY=F, 40 vehicles depart the origin during the time segment, giving a departure rate of 160 veh/h.
  - …

ILOOP Phase

Example

```
PROCESS PHASE=ILOOP
  MW[1]=MI.1.1
  MW[2]=MI.1.2

  DYNAMICLOAD PATH=COST, PACKETSIZE=2, DemandIsHourly=T,
  VOL[1]=0.8*MW[1], 0.8*MW[1],   ; pre-load period
       MW[1], 1.2*MW[1]+4*MW[2], MW[1], 0.8*MW[1]  ; simulation period
ENDPROCESS
```



ADJUST Phase

- Similar to ADJUST phase for static assignment (HIGHWAY)

Example

```
PROCESS PHASE=ADJUST
function {
    tc[1]=t0*(1.0+1*((v/c)^8)) ; congested time function for major raods
    tc[2]=t0*(1.0+1*((v/c)^4))
    tc[3]=t0
    cost=Time*Time_Cost+DISTANCE*Distance_Cost) ; cost function
ENDPROCESS

ENDRUN
```

Scripting Tips

- Improving Performance
  - Link consolidation in Cube
  - **MaxPthPerSeg**: controls the number of discrete paths to be built per O/D pair per segment
  - **PktPthSiz**: Maximum number of nodes a packet keeps in RAM
- Useful Variables
  - **TimeSegment**: the current time segment number (0 during static)
  - **__TS__** suffix: arrays a variable by segment in expressions
  - **SegmentStart**: Time between start of period and current segment
  - **Period**: duration of current period

Slow way

```
VOL[1]=MI.1.1,
       MI.1.2,
       MI.1.3,
       MI.1.4,
       MI.1.5,
       MI.1.6,
       MI.1.7,
       MI.1.8,
       MI.1.9,
       MI.1.10,
       MI.1.11,
       MI.1.12,
       MI.1.13,
       MI.1.14,
       MI.1.15,
       MI.1.16,
       MI.1.17,
       MI.1.18
```

Fast way

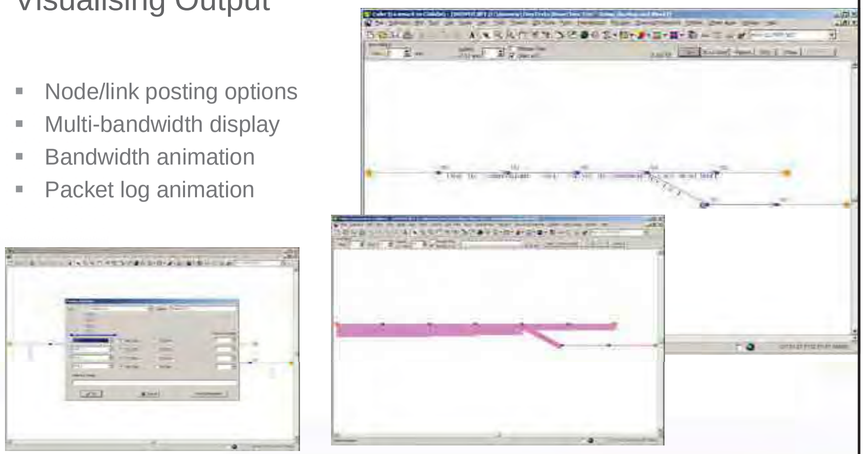
```
FILLMW MW[1]=MI.1.1(18)
...
VOL[1]=MW[__TS__]
```

Visualization and Analysis Tools

Dynamic Traffic Assignment with Cube Avenue

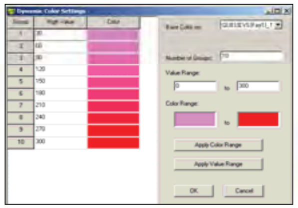
Visualising Output

- Node/link posting options
- Multi-bandwidth display
- Bandwidth animation
- Packet log animation



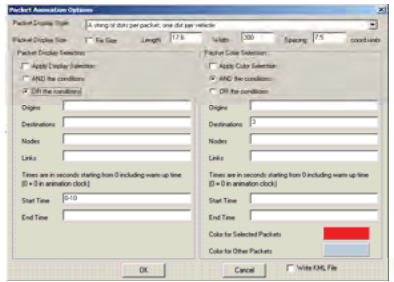
Bandwidth Animation

- Animation start/stop: key value range
- Speed: duration of each animation time step in tenths of a second
- Repeat play: cycle through key value range
- Display time: translate keys to segment time
- Start/Sync Start: trigger animation
- Pause/Resume: momentarily halt
- Stop/close: end animation



Packet Animation Options

- Packet Display Style: controls representation and placement of packets
- Packet Display Size: in coordinate units
  - Fix Size: dots do not scale after zooming in
- Packet Color Selection: apply the specified color to packets meeting the defined criteria
- Packet Display Selection: only show packets meeting the defined criteria



Packet Log Analysis Techniques

Dynamic Traffic Assignment with Cube Avenue

FILEO PACKETLOG Options

- ORIGIN/DESTINATION: Output packets with only listed origins or destinations.
- DEPARTTIME/ARRIVALTIME: Output only packets with departure/arrival times within the specified range of seconds.
- SELECTLINK: Output only packets using a specific link or list of links.
- SELECTGROUP: Output only packets using links of a particular group.
- MUSTMEETALL: If “F”, packets need not meet all select link criteria.
- AFTERITER: Delay writing the packet log until a specified iteration.
- FORMAT=BIN: Output a binary log instead of default text format.

The *.LOG text file

The text packet log follows a pseudo-XML format that is left intentionally open so as to permit additional parsing, processing and summary by the user.

Each file consists of a header record followed by one or more blocks of packet data.

Header Record

```
<!-- RUN="NETI.101585" PGM="AVENUE (v.10/18/2007 [4.2.0])"
    TIME="Thu Oct 18 18:47:46 2007" NPKT=133997
    PKTLNG=64         START=-0.500000  END=1.000000
    NVOL=2  VPD=250.000000 TimeSliceEnd=-0.416667,-
    0.333333,-0.250000,
    -0.166667,-
    0.083333,0.000000,0.083333,0.166667,0.250000,0.333333,
    0.416667,0.500000,0.583333,0.666667,0.750000,0.833333,
    0.916667,1.000000 -->
<p #=941,it=1>
    <v>
          <ix=0,f=0.357143>
          <ix=2,f=0.357143>
    </v>
    <r>
          <n=20,a=*,d=-0.499593>
          <n=430,a=-0.499593,d=-0.499529>
          <n=447,a=-0.472814,d=-0.472814>
...
          <n=475,a=-0.462416,d=-0.462416>
          <n=464,a=-0.458362,d=-0.458362>
          <n=476,a=-0.454521,d=-0.454521>
          <n=478,a=-0.448999,d=-0.448999>
          <n=484,a=-0.435480,d=-0.435480>
          <n=486,a=-0.432114,d=-0.432114>
          <n=17,a=-0.432114,d=*>
    </r>
</p>
```

Packet Data

Packet start record

This line begins the data relating to a packet. The '#=' gives the packet's identification number, in this case 941 (i.e. it was the nine hundred and forty first packet to be generated). The packet was generated during iteration 1 of the avenue run.

The data relating to this packet will continue until the corresponding </p> record. Both the <p> and the </p> records begin with the '<' character in column 1 but all the records enclosed will begin with at least one tab character.


```
<!-- RUN="NETI.101585" PGM="AVENUE (v.10/18/2007 [4.2.0])"
    TIME="Thu Oct 18 18:47:46 2007" NPKT=133997
    PKTLNG=64         START=-0.500000  END=1.000000
    NVOL=2  VPD=250.000000 TimeSliceEnd=-0.416667,-
    0.333333,-0.250000,
    -0.166667,-
    0.083333,0.000000,0.083333,0.166667,0.250000,0.333333,
    0.416667,0.500000,0.583333,0.666667,0.750000,0.833333,
    0.916667,1.000000 -->
<p #=941,it=1>
    <v>
          <ix=0,f=0.357143>
          <ix=2,f=0.357143>
    </v>
    <r>
          <n=20,a=*,d=-0.499593>
          <n=430,a=-0.499593,d=-0.499529>
          <n=447,a=-0.472814,d=-0.472814>
...
          <n=475,a=-0.462416,d=-0.462416>
          <n=464,a=-0.458362,d=-0.458362>
          <n=476,a=-0.454521,d=-0.454521>
          <n=478,a=-0.448999,d=-0.448999>
          <n=484,a=-0.435480,d=-0.435480>
          <n=486,a=-0.432114,d=-0.432114>
          <n=17,a=-0.432114,d=*>
    </r>
</p>
```

Volume start record

This line begins the data relating to the volume (ie number of vehicles) in the packet.

The data relating to volume will continue until the corresponding </v> record is found. The <v> and </v> records begin with one tab character and the enclosed records each begin with two tab characters.

```
<!-- RUN="NETI.101585" PGM="AVENUE (v.10/18/2007 [4.2.0])"
    TIME="Thu Oct 18 18:47:46 2007" NPKT=133997
    PKTLNG=64         START=-0.500000  END=1.000000
    NVOL=2  VPD=250.000000 TimeSliceEnd=-0.416667,-
    0.333333,-0.250000,
    -0.166667,-
    0.083333,0.000000,0.083333,0.166667,0.250000,0.333333,
    0.416667,0.500000,0.583333,0.666667,0.750000,0.833333,
    0.916667,1.000000 -->
<p #=941,it=1>
    <v>
        <ix=0,f=0.357143>
        <ix=2,f=0.357143>
    </v>
    <r>
        <n=20,a=*,d=-0.499593>
        <n=430,a=-0.499593,d=-0.499529>
        <n=447,a=-0.472814,d=-0.472814>
...
        <n=475,a=-0.462416,d=-0.462416>
        <n=464,a=-0.458362,d=-0.458362>
        <n=476,a=-0.454521,d=-0.454521>
        <n=478,a=-0.448999,d=-0.448999>
        <n=484,a=-0.435480,d=-0.435480>
        <n=486,a=-0.432114,d=-0.432114>
        <n=17,a=-0.432114,d=*>
    </r>
</p>
```

Volume set records

Each record gives the volume set number after 'ix=' and the volume (flow) after the 'f='.

There may be up to twenty volume sets numbered from 1 to 20, in addition to a virtual 'volume set zero' defined to be the sum of the other volume sets. The sets are listed in ascending order of index.

Records are not produced for sets that do not exist (ie that are not mentioned in the generation script). Records are not produced for sets that contain no vehicles.


```
<!-- RUN="NETI.101585" PGM="AVENUE (v.10/18/2007 [4.2.0])"
    TIME="Thu Oct 18 18:47:46 2007" NPKT=133997
    PKTLNG=64         START=-0.500000  END=1.000000
    NVOL=2  VPD=250.000000 TimeSliceEnd=-0.416667,-
    0.333333,-0.250000,
    -0.166667,-
    0.083333,0.000000,0.083333,0.166667,0.250000,0.333333,
    0.416667,0.500000,0.583333,0.666667,0.750000,0.833333,
    0.916667,1.000000 -->
<p #=941,it=1>
    <v>
        <ix=0,f=0.357143>
        <ix=2,f=0.357143>
    </v>
    <r>
        <n=20,a=*,d=-0.499593>
        <n=430,a=-0.499593,d=-0.499529>
        <n=447,a=-0.472814,d=-0.472814>
...
        <n=475,a=-0.462416,d=-0.462416>
        <n=464,a=-0.458362,d=-0.458362>
        <n=476,a=-0.454521,d=-0.454521>
        <n=478,a=-0.448999,d=-0.448999>
        <n=484,a=-0.435480,d=-0.435480>
        <n=486,a=-0.432114,d=-0.432114>
        <n=17,a=-0.432114,d=*>
    </r>
</p>
```

Start route record

The </v> line denotes the end of the volume data and the <r> record denote the beginning of the route data. The route data will continue until the corresponding </r> record.

Again the <r> and </r> records each begin with one tab character and the enclosed records begin with two tab characters.

```
<!-- RUN="NETI.101585" PGM="AVENUE (v.10/18/2007 [4.2.0])"
    TIME="Thu Oct 18 18:47:46 2007" NPKT=133997
    PKTLNG=64         START=-0.500000  END=1.000000
    NVOL=2  VPD=250.000000 TimeSliceEnd=-0.416667,-
    0.333333,-0.250000,
    -0.166667,-
    0.083333,0.000000,0.083333,0.166667,0.250000,0.333333,
    0.416667,0.500000,0.583333,0.666667,0.750000,0.833333,
    0.916667,1.000000 -->
<p #=941,it=1>
    <v>
        <ix=0,f=0.357143>
        <ix=2,f=0.357143>
    </v>
    <r>
        <n=20,a=*,d=-0.499593>
        <n=430,a=-0.499593,d=-0.499529>
        <n=447,a=-0.472814,d=-0.472814>
...
        <n=475,a=-0.462416,d=-0.462416>
        <n=464,a=-0.458362,d=-0.458362>
        <n=476,a=-0.454521,d=-0.454521>
        <n=478,a=-0.448999,d=-0.448999>
        <n=484,a=-0.435480,d=-0.435480>
        <n=486,a=-0.432114,d=-0.432114>
        <n=17,a=-0.432114,d=*>
    </r>
</p>
```

Node list records

The route itself consists of a sequence of node records in the order that they are visited on the route. Two times are listed for each node: an arrival time, denoted 'a=', and a departure time denoted 'd='.

The times are in **hours** relative to the beginning of the model period (so negative numbers refer to the warm up period). Where no time is available, an asterisk, '*', is placed in the field in place of the number. No arrival time is available at an origin and no departure time is available at a destination. No departure time is available for a node that the packet failed to reach by the end of the model period.


```
<!-- RUN="NETI.101585" PGM="AVENUE (v.10/18/2007 [4.2.0])"
    TIME="Thu Oct 18 18:47:46 2007" NPKT=133997
    PKTLNG=64         START=-0.500000  END=1.000000
    NVOL=2  VPD=250.000000 TimeSliceEnd=-0.416667,-
    0.333333,-0.250000,
    -0.166667,-
    0.083333,0.000000,0.083333,0.166667,0.250000,0.333333,
    0.416667,0.500000,0.583333,0.666667,0.750000,0.833333,
    0.916667,1.000000 -->
<p #=941,it=1>
    <v>
        <ix=0,f=0.357143>
        <ix=2,f=0.357143>
    </v>
    <r>
        <n=20,a=*,d=-0.499593>
        <n=430,a=-0.499593,d=-0.499529>
        <n=447,a=-0.472814,d=-0.472814>
...
        <n=475,a=-0.462416,d=-0.462416>
        <n=464,a=-0.458362,d=-0.458362>
        <n=476,a=-0.454521,d=-0.454521>
        <n=478,a=-0.448999,d=-0.448999>
        <n=484,a=-0.435480,d=-0.435480>
        <n=486,a=-0.432114,d=-0.432114>
        <n=17,a=-0.432114,d=*>
    </r>
</p>
```

citilabs
TRANSPORTATION MODELING SOFTWARE

cube

Processing the Packet Log Data

By applying record processing techniques to packet log output data, you can implement many advanced analyses with Avenue:

- Build origin-destination table from log file
- Select node/link analysis
  - Select link/node trip table: build table of trips using some node at some particular time
  - Check whether packets used a particular link/node and build a link table from the list of nodes used by these packets (2 passes of the file)
- Extract average queue for specific packets (departure minus arrival)
- Temporal disaggregation (e.g. build 15-minute matrices from peak hour simulation output based upon recorded departures)
- Peak spreading
  - Build packet table from log file, flag packets that failed to arrive at their destination
  - Shift packets to new departure time segment based upon logit or other decision rule
  - Re-build hourly trip matrix from packet table
- Other applications: ITS/VMS, parking, sub-area extraction, ME, etc…

citilabs

cube

Thank you!

Tor Vorraa – Citilabs, Regional Director
Alberto Brignone – Citilabs, Regional Director

citilabs

APPENDIX 3

STAGE 3 (GIS) TRAINING PROGRAM CONTENT

Training Report
For
ArcGIS Training
For
The Comprehensive Study
On
Master Plan for Nationwide Transport System in Egypt

Between

Oriental Consultants Co.,Ltd.

And

Quality Standards Information Technology (QSIT)

Training Report: ArcGIS

1 Outline of ArcGIS training

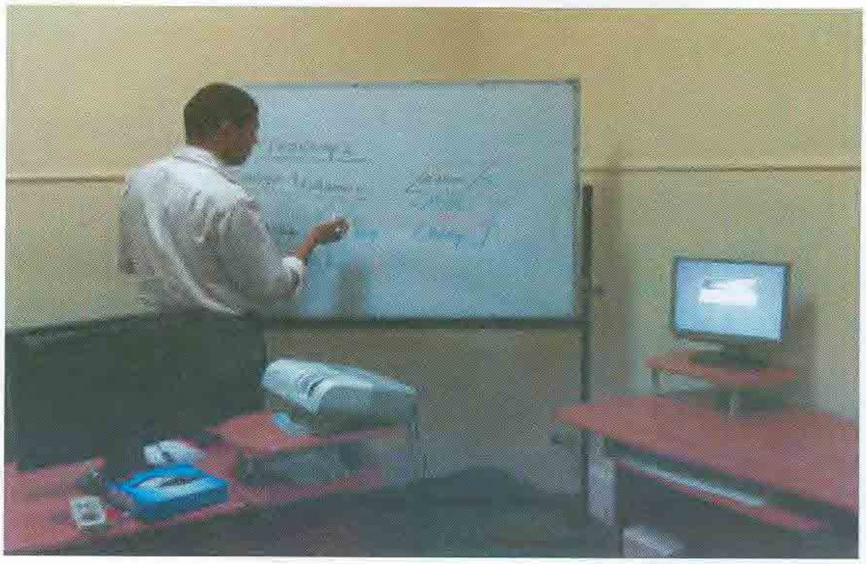
1.1 Schedule and Venue

GIS Basics Track		
Item	Course Name	No. of training days
1	Introduction to ArcGIS desktop I	2 days
2	Introduction to ArcGIS desktop II	3 days
3	On the Job Training	3 days
Total		**8 days**

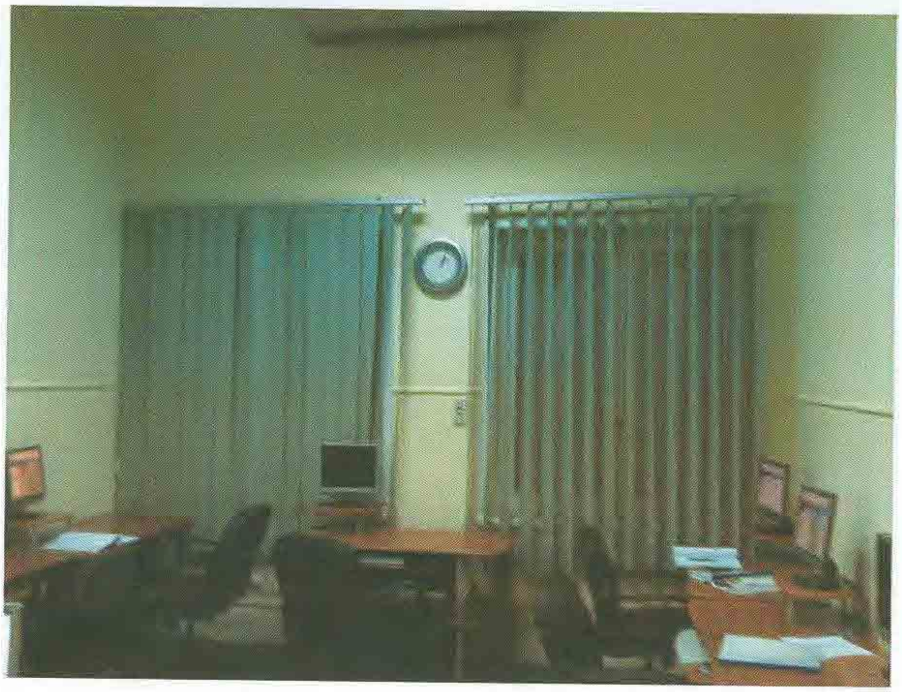
Regarding the training center Facilities & Images; many facilities are available:

- Training center Entrance:

- Presentation Facilities: (White board, Data Show,..etc)



- Training Lab



ESRI NeA Training Standards:

It is the desire of ESRI to establish and maintain a high level of quality in teaching ESRI courses through its International Distributors. The quality of education should reflect the standard of service offered by ESRI

And Since ESRI Northeast Africa (ESRI NeA) is recognized as one of the most significant, specialized, and qualified training providers in Egypt and the MENA region, we are always making sure that all ESRINeA Certified Instructors are following ESRI International course standards.

Focusing on knowledge-transfer to trainees, our instructors have hands-on experience in application development which enriches the training process.

As a standard procedure, ESRINeA training center provides the attendees with a high quality printed ESRI Training materials (Lectures & Exercises),Regarding the training, ESRI training courses start at 9:00 AM to 5:00 PM with a lunch break; all instructors are subject to evaluation by trainees. Being keen on employee's advancement, ESRI NeA's training staff is regularly appraised against performance to develop their skills and stretch their capabilities.

SiT

Training Center

Address: 2 El-Messaha Square, Dokki, Cairo. Egypt

Telefax: (202) 3748 5288 - 33387358

Website: www.esrinea.com

QSIT Head Office

Address: St 15 ,Free zone , Nasr City , Cairo

Tel: (202) 22719350 - 22719341

Fax: (202) 22719350

1.2 Training Materials

1- Each trainee is provided with exercise book and lecture book for each course.

2- For ESRI standard courses, one computer per trainee is provided in the classroom for exercises purposes.

3- Evaluation copy for the software is delivered to the customer upon request.

4- During the on job training, JICA trainees were grouped in order to make information sharing between them.

5- The course was delivered on ArcGIS Desktop 9.3 (ArcInfo floating) on 7 Pcs

2 Course description and results

2.1 Day 1 & 2: Introduction of ArcGIS desktop I

(1) **Goals**

- Be able to explain what GIS is and what it can do
- Learn how GIS maps are different from other maps
- Have fun making maps
- Use geographic data
- Answer geographic questions
- Analyze spatial relationships
- Solve a real-world problem with ArcGIS
- Explain how you will use GIS in your daily life

(2) **Topic Covered**

- Identify basic functions of a GIS
- Formulate a first draft definition of GIS
- differences between paper and GIS maps

- Describe features, layers, and data frames
- Describe characteristics of GIS maps
- the feature-attribute relationship
- Describe categorical vs. quantitative symbolization
- methods of symbolization
- the two main geographic data structures
- how each represents geographic objects
- that geographic data is acquired
- what a geodatabase is

(3) **Result**

- What GIS is and what it can do
- How GIS maps are unique
- How to make a map in ArcMap
- How to use geographic data
- How to query and analyze geographic data
- How to solve a problem using GIS

2.2 Day 3,4 & 5: Introduction of ArcGIS desktop II

(1) **Goals**

- Symbolize, classify, and label geographic data
- Apply coordinate systems to geographic data
- Create maps for presentation
- Manage table properties and relationships
- Edit map features and attributes
- Create a geographic database and new data
- Geocode addresses
- Solve geographic problems using analysis tools
- Customize the ArcGIS interface

(2) **Topic Covered**

- the structure of geographic data
- what a geodatabase is
- non-geodatabase file formats
- advantages of the geodatabase
- main purposes of ArcCatalog and ArcMap
- the relationship between data and layers

(3) **Result**

- Symbolize, classify, and label geographic data
- Apply coordinate systems to geographic data
- Create maps for presentation
- Manage table properties and relationships
- Edit map features and attributes

- Create a geographic database and new data
- Geocode addresses
- Solve geographic problems using analysis tools
- Customize the ArcGIS interface

2.3 **Day 6 & 7: On the Job Training**

(1) **Goals**

- Distinguish types of ArcGIS Desktop geodatabases
- Add data from other sources to a geodatabase
- Perform spatial and attribute validation in a geodatabase
- Edit features using geodatabase behaviors
- Build geoprocessing models using ModelBuilder
- Analyze GIS data in ArcGIS Desktop
- Solve real-world problems using GIS analysis

(2) **Topic Covered**

- Distinguish types of ArcGIS Desktop geodatabases
- Add data from other sources to a geodatabase
- Perform spatial and attribute validation in a geodatabase
- Edit features using geodatabase behaviors
- Build geoprocessing models using ModelBuilder
- Analyze GIS data in ArcGIS Desktop
- Solve real-world problems using GIS analysis

(3) **Result**

- Review ArcGIS terminology
- Define additional ArcGIS and geodatabase terms
- Describe advantages of using a file geodatabase
- Know what data formats can be converted
- Load data into existing feature classes
- Learn methods of data conversion and loading
- Work with projections and datums
- Attendees wrote the exact steps by themselves