CHAPTER 37

Interactive Analysis and Decision Support with MATSim

Alexander Erath and Pieter Fourie

37.1 Basic Information

Entry point to documentation:

`http://matsim.org/extensions` → travelsummary

Invoking the module:

`http://matsim.org/javadoc` → travelsummary → `RunEventsToTravelDiaries`

Selected publications:

This chapter is largely based on work in Erath et al. (2013), where the interested reader will find references for further reading.

37.2 Introduction

Agent-Based Simulation Means Lots of Data Agent-based transport demand models require managing and integrating data sources several orders of magnitude larger than traditional aggregate models. In a truly disaggregate demand description, as seen in our MATSim implementation for Singapore, spatial data represents individual buildings and land parcels, not zones; travel demand takes the form of a full activity diary with connecting trips for every individual, based on their personal demographic attributes, instead of an aggregate number of trips from zone to zone for a specific time period. For this reason, input data for an aggregate four-step (or related) demand model can generally be edited on a laptop, using standard spreadsheet software, whereas agent-based modeling requires the manipulation and synthesis of large stores of structured, hierarchical data, frequently exceeding most personal computer capacity.

How to cite this book chapter:

Erath, A and Fourie, P. 2016. Interactive Analysis and Decision Support with MATSim. In: Horni, A, Nagel, K and Axhausen, K W. (eds.) *The Multi-Agent Transport Simulation MATSim*, Pp. 253–258. London: Ubiquity Press. DOI: http://dx.doi.org/10.5334/baw.37. License: CC-BY 4.0

How MATSim Stores Data MATSim stores and retrieves data from XML, because XML reflects objects' hierarchical structure in the simulation and is readable. However, performing general exploratory analysis of large XML data stores is usually poorly supported by most data analysis software packages, especially GIS-based systems. To perform analyses, expert knowledge of XML querying technologies like XPath and XQuery is required (or Java, if one performs more specialized analysis on the objects themselves). In our experience, this specialized knowledge is lacking in transport and urban spatial planning practice. Therefore, in most MATSim applications so far, authorities, and other interested parties, must formulate their desired analysis in advance and have expert consultants perform the analysis. Any queries resulting from the analysis require another consultation cycle and the client's perceived value declines, due to both lack of interactivity and model ownership feeling. We believe this lack of a broadly supported exploratory data analysis interface, and the customer experience the interface can create, presents a considerable barrier to entry for many authorities and operators interested in using MATSim.

How Customers Interact With Data: Relational Databases, GUI-Driven Interaction Most transport and urban spatial planning customers rely on mature, GUI-driven software, such as ArchGIS (ESRI, 2011), EMME/3 (INRO, 2015), the PTV (PTV, 2009) transport planning suite, or even Microsoft Excel; all of these connect to relational databases and perform queries on large data sets. Many analysts can explicitly query databases using the SQL (Structured Query Language); the ODBC (Open Database Connectivity) standard allows software to connect to any relational database regardless of the actual technology driving it. Importantly, many interactive exploratory data analysis software suites, like Tableau, Tibco Spotfire, SAS and the open source R project, support relational databases and ODBC.

37.3 Requirements of a Decision Support Interface to MATSim

The event stream produced by the MATSim mobility simulation represents the transport simulation process at the atomic level. It could be fed into a relational database; an analyst fluent in procedural languages could process it in arbitrary ways. But we expect more general use case scenarios, where most analysts will perform general tasks that can be standardized. To this end, we set about compiling requirements specifications for potential audiences and their use case scenarios, to come up with a general interactive analysis framework and decision support to satisfy most requirements. We developed a set of Java classes to process MATSim input and output, producing tables in a relational database, and an entity relationship diagram that should be intuitive and useful to a large user audience.

37.3.1 Users

This chapter presents a decision support tool geared to decision makers and researchers in the fields of transport planning and operations, spatial planning and spatial economics and geography. Generally speaking, it should serve professionals interested in mobility and spatial analysis, who understand transport modeling principles, but do not have the expertise to operate an agent-based transport simulation directly. Currently, we envision the following stakeholders and some hypothetical questions for a decision-support system—a non-exhaustive list that, we expect, will grow with time:

Transport planners: How many trips occur where, when and what is the activity purpose? What are the socio-demographic characteristics of people performing these trips and activities?

Urban Planners: What are the temporal usage patterns of buildings and the surrounding neighborhood? What is the flow from public transport stops to surrounding buildings?

Policy-Makers: What are the costs and benefits of a new public transport service? Who are the winners and losers when constructing a new road?
Public Transport Operators: What is the breakdown of specific bus lines' ridership?
Service Industry: Which customers are in catchment areas, separated by mode?

37.3.2 Functional Requirements

The decision support framework should facilitate classic transport appraisal methods, such as cost/benefit analysis and evaluation of transport infrastructure spatial impact and policy measures. The framework should allow any sort of spatial analysis, on the finest granularity level provided by the transport model; usually, individual buildings or parcels, as well as public transport stops and selected links, like count stations or tolled road segments. However, these geographic features should be indexed against transport zones, or other geographic areas of interes,t to allow customized results aggregation. Furthermore, it should capture all temporal aspects of the simulation; full temporal dynamics are a crucial part of the agent-based approach.

37.4 General Framework for Decision Support

Figure 37.1 shows the general framework as we envision it: data from various sources feeds into a spatially-enabled database, with all geodata transformed to use the same spatial reference system (ideally, using the same projection used for MATSim coordinates, allowing for simple distance calculations). Simple Java programs using the MATSim API and JDBC (Java Database Connectivity) produce XML input data for MATSim scenarios; events from these scenarios are fed back into the database. Analysts query the database to produce "data cubes", which are aggregations and queries across many database tables. These are designed for specific purposes, such as calibration and validation, location analysis, winner/loser analysis or other application-specific purposes.

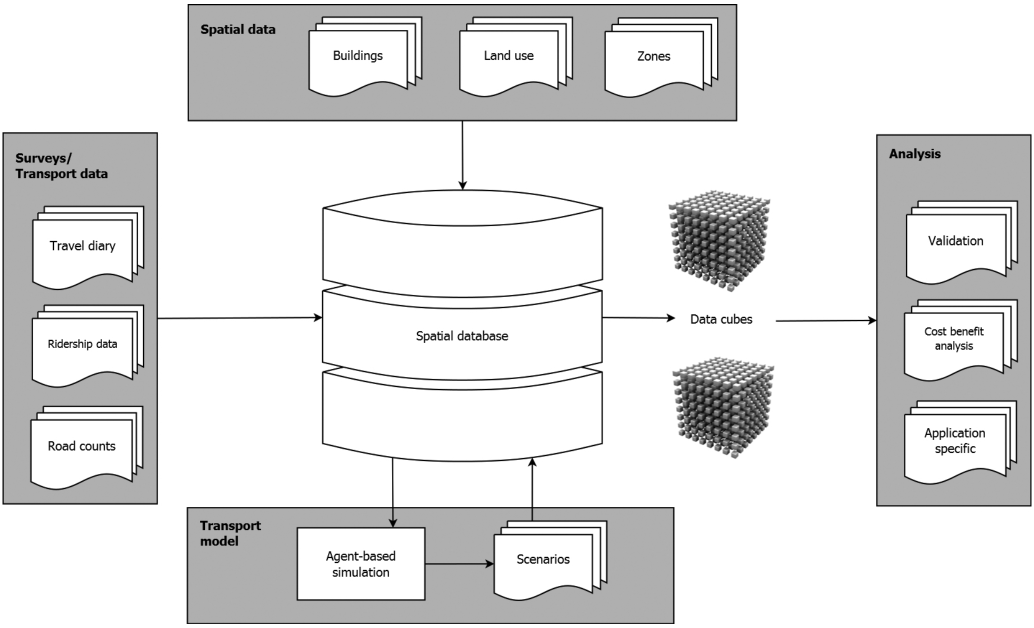


Figure 37.1: General framework of the decision support system.

37.4.1 Entity Relationship Diagram (ERD) for General Purpose Analysis

For entity relationships, we decided that a travel diary format is most suitable for the usual types of analysis, but works especially well for comparison with other data sources when validating simulation output. Most travel surveys take the form of a diary, recording travel time, purpose and mode, as well as aspects of the journey like number of stages, transfer walking and waiting time and in-vehicle time. Routines can be developed to transform survey data and public transport smart card records into the same format with consistent coding. Figure 37.2 shows the ERD (Entity Relationship Diagram) we propose, along with the primary/foreign key relationships between tables that facilitate aggregation and joining of e.g., personal/household attributes, such as income, with travel time experienced in the simulation.

37.4.2 Interactive Analysis Using Business Analytics Software

Modern business analytics software, like Tableau (Tableau Software, 2013), provide interactive data aggregation and visualization from relational databases. While basic analysis of individual tables

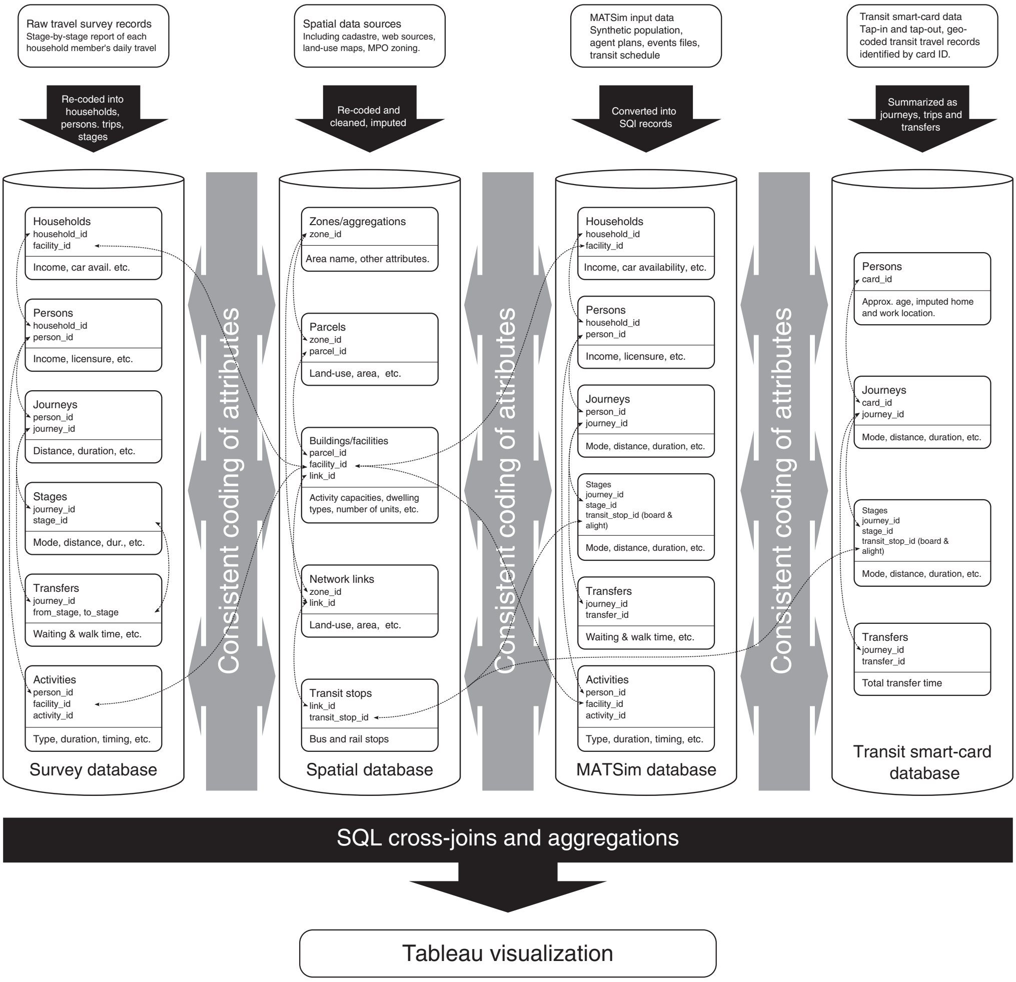

Raw travel survey records
Stage-by-stage report of each household member's daily travel
Spatial data sources
Including cadastre, web sources, land-use maps, MPO zoning.
MATSim input data
Synthetic population, agent plans, events files, transit schedule
Transit smart-card data
Tap-in and tap-out, geo-coded transit travel records identified by card ID.
Re-coded into households, persons. trips, stages
Re-coded and cleaned, imputed
Converted into SQl records
Summarized as journeys, trips and transfers
Households
household_id
facility_id
Income, car avail. etc.
Persons
household_id
person_id
Income, licensure, etc.
Journeys
person_id
journey_id
Distance, duration, etc.
Stages
journey_id
stage_id
Mode, distance, dur., etc.
Transfers
journey_id
from_stage, to_stage
Waiting & walk time, etc.
Activities
person_id
facility_id
activity_id
Type, duration, timing, etc.
Survey database
Consistent coding of attributes
Zones/aggregations
zone_id
Area name, other attributes.
Parcels
zone_id
parcel_id
Land-use, area, etc.
Buildings/facilities
parcel_id
facility_id
link_id
Activity capacities, dwelling types, number of units, etc.
Network links
zone_id
link_id
Land-use, area, etc.
Transit stops
link_id
transit_stop_id
Bus and rail stops
Spatial database
Consistent coding of attributes
Households
household_id
facility_id
Income, car availability, etc.
Persons
household_id
person_id
Income, licensure, etc.
Journeys
person_id
journey_id
Mode, distance, duration, etc.
Stages
journey_id
stage_id
transit_stop_id (board & alight)
Mode, distance, duration, etc.
Transfers
journey_id
transfer_id
Waiting & walk time, etc.
Activities
person_id
facility_id
activity_id
Type, duration, timing, etc.
MATSim database
Consistent coding of attributes
Persons
card_id
Approx. age, imputed home and work location.
Journeys
card_id
journey_id
Mode, distance, duration, etc.
Stages
journey_id
stage_id
transit_stop_id (board & alight)
Mode, distance, duration, etc.
Transfers
journey_id
transfer_id
Total transfer time
Transit smart-card database
SQL cross-joins and aggregations
Tableau visualization

Figure 37.2: Simplified entity relationship diagram showing shared keys across tables.

in our proposed ERD could already provide valuable insight to MATSim simulations, much richer analysis is possible when tapping relationships between different tables in the database. With the help of graphical query building software, little or no knowledge is required to construct SQL scripts that create customized data cubes. These cubes are fed into the business analytics software, which is designed with a relatively programming-agnostic audience in mind. Relying on the familiar paradigm of drag-and-drop interaction in a simple, well-documented GUI, the user constructs “dashboards” summarizing information and allowing interactive aggregation, or drilling-down across multiple dimensions.

Figure 37.3 shows a Tableau visualization comparing public transport ridership from a MATSim simulation to actual smart card data records (transformed into the travel diary format specified in the ERD). Figure 37.4 shows the SQL query used to produce the data frame driving the Tableau analysis. The query exploits the primary/foreign key relationships in the database to perform rapid joins between the different tables.

37.5 Diaries from Events

In the package `contrib.analysis.travelsummary` (Chapter 38), the reader can find a set of classes that will transform their MATSim simulation results into a set of travel diary tables, like those discussed in the preceding section. The package contains a simple GUI class that can be run to specify input data XML files, the location to save output CSV (Comma-Separated Values) files and other information such as a subscript appended to the end of file names to identify different scenarios. These CSV files can be read into a relational database of choice, or directly queried in Tableau, or other interactive analysis software.

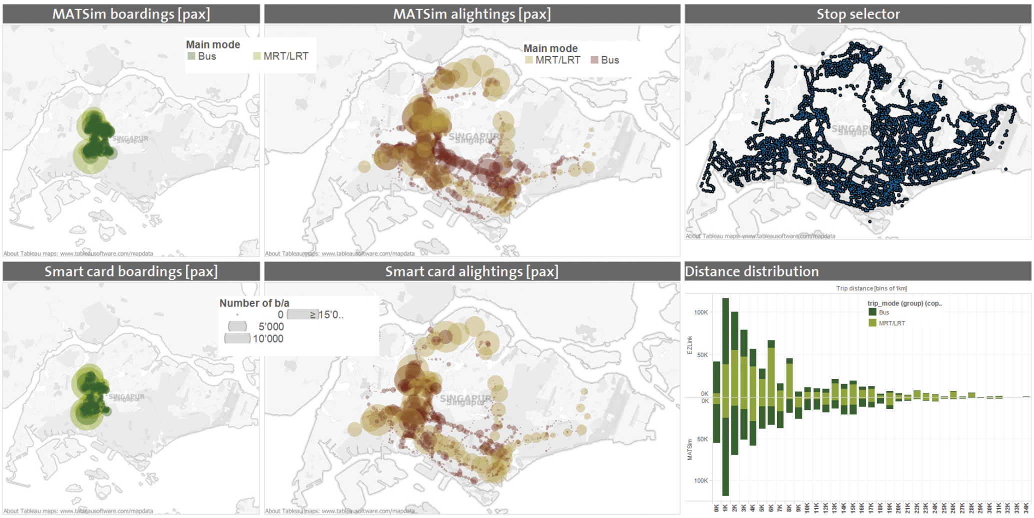
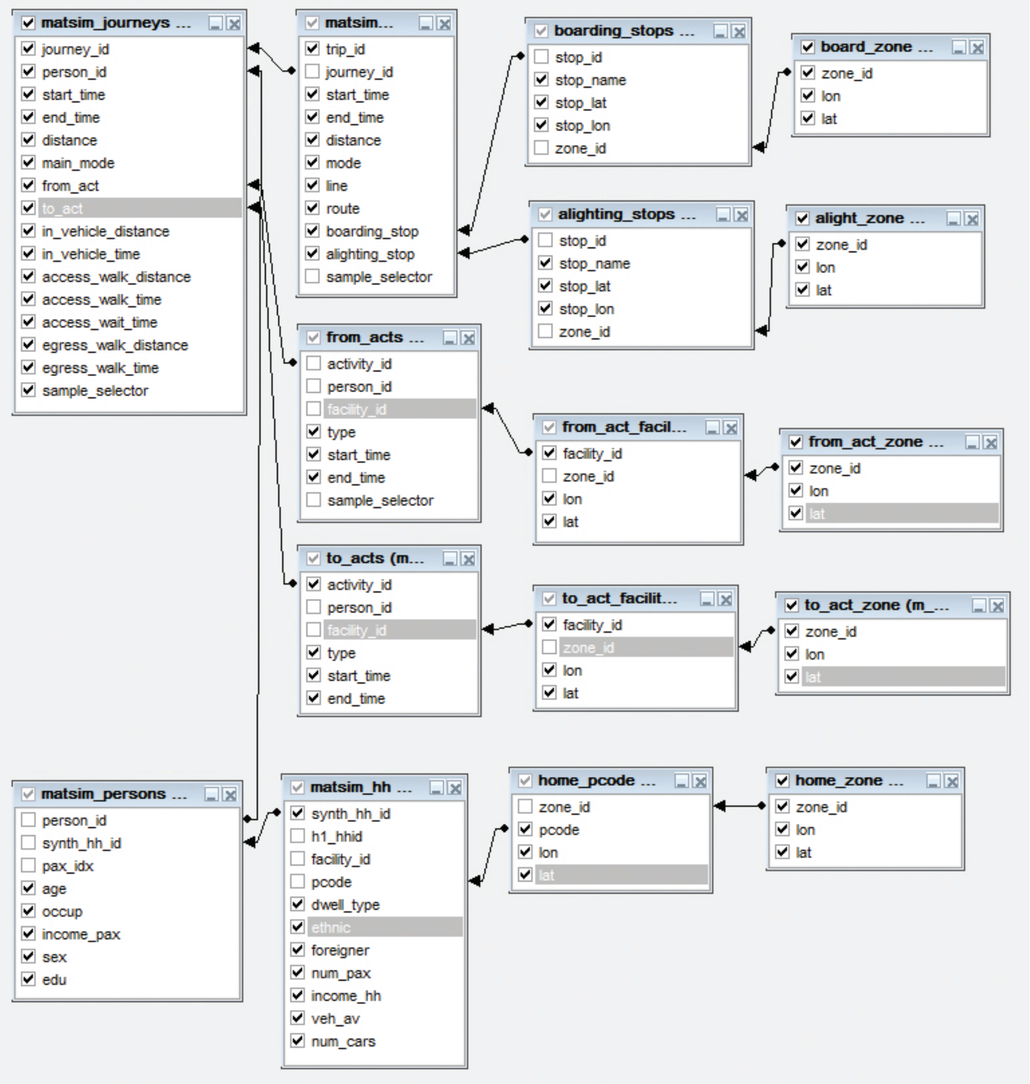


Figure 37.3: Tableau visualization of public transport ridership from a MATSim simulation compared against actual smart card data records in Singapore.



☑ matsim_journeys ...
- ☑ journey_id
- ☑ person_id
- ☑ start_time
- ☑ end_time
- ☑ distance
- ☑ main_mode
- ☑ from_act
- ☑ to_act
- ☑ in_vehicle_distance
- ☑ in_vehicle_time
- ☑ access_walk_distance
- ☑ access_walk_time
- ☑ access_wait_time
- ☑ egress_walk_distance
- ☑ egress_walk_time
- ☑ sample_selector

☑ matsim...
- ☑ trip_id
- ☐ journey_id
- ☑ start_time
- ☑ end_time
- ☑ distance
- ☑ mode
- ☑ line
- ☑ route
- ☑ boarding_stop
- ☑ alighting_stop
- ☐ sample_selector

☑ boarding_stops ...
- ☐ stop_id
- ☑ stop_name
- ☑ stop_lat
- ☑ stop_lon
- ☐ zone_id

☑ board_zone ...
- ☑ zone_id
- ☑ lon
- ☑ lat

☑ alighting_stops ...
- ☐ stop_id
- ☑ stop_name
- ☑ stop_lat
- ☑ stop_lon
- ☐ zone_id

☑ alight_zone ...
- ☑ zone_id
- ☑ lon
- ☑ lat

☑ from_acts ...
- ☐ activity_id
- ☐ person_id
- ☐ facility_id
- ☑ type
- ☑ start_time
- ☑ end_time
- ☐ sample_selector

☑ from_act_facil...
- ☑ facility_id
- ☐ zone_id
- ☑ lon
- ☑ lat

☑ from_act_zone ...
- ☑ zone_id
- ☑ lon
- ☑ lat

☑ to_acts (m...
- ☑ activity_id
- ☐ person_id
- ☐ facility_id
- ☑ type
- ☑ start_time
- ☑ end_time

☑ to_act_facilit...
- ☑ facility_id
- ☐ zone_id
- ☑ lon
- ☑ lat

☑ to_act_zone (m_...
- ☑ zone_id
- ☑ lon
- ☑ lat

☑ matsim_persons ...
- ☐ person_id
- ☐ synth_hh_id
- ☐ pax_idx
- ☑ age
- ☑ occup
- ☑ income_pax
- ☑ sex
- ☑ edu

☑ matsim_hh ...
- ☑ synth_hh_id
- ☐ h1_hhid
- ☐ facility_id
- ☐ pcode
- ☑ dwell_type
- ☑ ethnic
- ☑ foreigner
- ☑ num_pax
- ☑ income_hh
- ☑ veh_av
- ☑ num_cars

☑ home_pcode ...
- ☐ zone_id
- ☑ pcode
- ☑ lon
- ☑ lat

☑ home_zone ...
- ☑ zone_id
- ☑ lon
- ☑ lat

Figure 37.4: A diagram showing how the tables from Figure 37.2 are joined together for visualization in business analytics software, e.g., Tableau, as shown in Figure 37.3.
Source: (Erath et al., 2013)