**On Knowing What You Have When You Get It: Building Activity Models and Learning Activity Modeling Theory**

Erik Sabina, Jennifer Malm, Suzanne Childress
Denver Regional Council of Governments (DRCOG)
1290 Broadway, Suite 700
Denver, CO 80203
303-455-1000 (voice), 303-480-6790 (fax)
esabina@drcog.org, jmalm@drcog.org,
schildress@drcog.org

John Bowman
Transportation Systems and Decision Sciences
28 Beals Street
Brookline, MA 02446
617-232-8189 (voice)
John_L_Bowman@alum.mit.edu

DeVon Culbertson
DeVon Culbertson, LLC
1156 Opal Street #202
Broomfield, CO 80020
720-936-5848
DevonCulbertson@yahoo.com

12th TRB National Transportation Planning Applications Conference, Houston, TX, May 17th – 21st, 2009

## Abstract

Following closely behind the schedule of some early adopters in the US, a lengthening list of planning organizations are taking first steps toward development of tour/activity-based regional travel models. One of the key issues raised by organizations contemplating such projects concerns how they can make the educational leap from their current trip-based modeling expertise to equivalent expertise in tour/activity-based models. The experience of DRCOG in its activity modeling project provides a useful example to other MPOs in the planning stages of their own projects. By project conclusion, DRCOG will have conducted approximately half of the technical work of the project in-house, with the other half conducted by consultants. DRCOG staff have been involved in every aspect of the project, including estimation data development, choice model estimation, software development, and model calibration and validation.

This paper addresses a list of issues, beginning with the theoretical expertise organizations should have before they commence such a project, and the trade-offs inherent in learning as the project progresses versus keeping to schedule. Detailed discussion is provided of DRCOG's experience in model estimation, including: what models were estimated by DRCOG and what by consultants; in retrospect, other models DRCOG could have estimated; easier versus harder models for DRCOG to handle itself; practical details in estimation from the MPO point of view; examples of model estimation from DRCOG's experience; advantages and disadvantages of doing the work in-house; and the success of the "consultant coach" model. Detailed discussion also is provided of DRCOG's software approach: the basic details of the design; thoughts about what modelers need to know about their software; and up and down sides to DRCOG's approach. Conclusions and recommendations for others are provided, including DRCOG's perception of future training it will require as it continues to work with and enhance its new model.

## Introduction

When DRCOG began in 2002 to design a project to develop a next-generation model for the Denver metropolitan area, our knowledge of activity-based modeling (ABM) structures was quite limited, and there was only one fully-operational ABM in the United States. However, our view of how well we wanted to understand the model we would eventually build was well settled: DRCOG staff were widely agreed that we wanted to understand the new model very well, at least as well as we understood our current trip-based model. The motive behind this need was similar to the motive to build the model in the first place: the kinds of questions our customers were asking us. The Denver region has experienced rapid growth for many years, and many complex planning issues and approaches have been debated in the context of response to that growth. The need for DRCOG staff to assist in that debate has required us to develop a quite detailed understanding of the technical tools we employ, and that demand shows every sign of increasing in the future. The approach we took to software development for model implementation, which is unique in several respects, also required us to understand the model very well, as we would have been unable to design and build the software without a high level of model understanding. This paper is not intended to tell others how they should conduct their own projects. Rather, it is intended to provide one "data point" for others contemplating such a project, showing what we learned about our model, how we learned it as the project has progressed, and why we felt we needed to learn it.

**Where We Started**

While DRCOG staff possessed little knowledge of ABM structure at the beginning of the project, staff knowledge of the fundamentals of model design and development was reasonably well-developed. Key area of expertise, and our experience in them included:

- Software design and programming – one extremely strong software design lead, and three other project staff with well-developed programming skills.
- Discrete choice model estimation – two project staff with good educational background in the subject, though relatively limited application experience.
- Economics, mathematics, and statistics – five or six staff with well developed skills in one or more of these areas.
- Trip-based modeling – two staff with strong experience in this area.

It should be emphasized that, while we could have conducted the project with lower skill levels than we possessed, our ability to conduct some project activities in-house was facilitated by this expertise, which in turn strongly facilitated our ability to understand the model as it was being built.

**Key Project Challenges**

There is not space to treat here in exhaustive fashion all of the key challenges of building an ABM, especially when taking a strongly "in-house" approach. A few comments only will be offered concerning a list of technical challenges others are likely to face who initiate such a project. Project management challenges are likely to be significant as well, but in the interest of brevity, we will address only the broadest of them: that building an ABM is a "high risk" project. The Project Management Institute lists two key characteristics of high risk projects: that they are large and technical, and that it is not possible to write a detailed scope of work at project initiation (Project Management Institute, 2004.) ABM projects possess both of these characteristics, and project managers are advised to build in significant safety margins both of cost and schedule.

Key technical challenges include:

*Selecting the software environment* – the list of feasible software environments is very long, which by itself can make this a difficult decision to make. Typically, travel modelers' knowledge of the pros and cons of various hardware and software options is somewhat limited, making the decision process all the more difficult. Finally, if it is desirable to consider options beyond the products offered by existing travel modeling software vendors, the project likely will need to bring in additional participants, such as the agency's IT staff. This can bring much-needed expertise to the project, but can also increase the number of stakeholders, adding to the level of effort required to reach consensus on an approach.

*Knowledge exchange between software designers and modelers* – in DRCOG's case, the IT group was brought in at the beginning of the process, and brought invaluable understanding of software architectural design. However, the project overhead involved in using IT specialists to lead the software design effort is very great: a time-consuming two-way exchange of information is required, with the IT specialists educating the modelers in software design (and in our case, a programming language that was new to the modelers), and modelers educating the IT specialists in many technical details of travel modeling (such as the mathematics of discrete choice analysis.)

*Understanding how logsums are actually used in ABMs* – most presentations refer to "downward" and "upward" integration of travel model components, and talk of "feeding logsums upward from the later models to the earlier." While we do not pretend to have exhaustively examined every possible ABM approach, none of the models with which we're familiar actually feed variables from later model components to earlier ones (no one having yet invented a time machine component!) Logsums that are used in a component early in the model stream (say, a workplace location choice model) are actually generated by an approximate version of the tour mode choice model (whose full version is run much later in the model sequence), using information available early in the model sequence. This, and other subtleties of logsums, presented some of the biggest challenges to our project team as we worked to understand the theoretical structure of our ABM system.

*Other subtleties of ABM theory and design* – there are very many of these, and it is hard to know which are important to know and which not. However, in our experience, we have not yet encountered an element of theoretical/structural knowledge of our model that we felt was not useful to us. A few examples:

- The daily activity pattern (DAP) and tour time of day (TOD) models have large choice sets, cut down from even larger sets, based on feasibility constraints and on observed patterns from the travel survey used for estimation (for example, very few people made more than three tours in a day.)
- Work-based subtours are treated as separate tours in our ABM. However, they are not generated as part of the initial DAP set of tours, but are generated afterward, and only if the initial daily activity pattern contains a work tour.

- Speed of performance of the ABM software is a huge challenge. Consider our workplace location choice model, with approximately 2,800 zones in the choice set, and 35 variables in each utility function, run in the base year for approximately 1.3M workers. As one of these variables is a disaggregate mode choice logsum, the total calculations necessary to run this model is on the order of one trillion.

*Computer memory limitations* – many of the models are run on a very large number of "decision agents" (people, households, etc.) and the most straightforward way to run them is to bring simultaneously into memory all information on all decision agents. For example, the workplace location choice model must be run for 1.3M workers in our base year scenario. This approach has challenged the memory capacity of current desktop computers. Similarly, some of the model components (especially the mode choice models) require many skim variables. If all these skims can be held in memory simultaneously, speed of performance is enhanced. However, DRCOG's large zone system means that each skim matrix has approximately 8M cells, which also challenges the capacity of typical computer memory.

*Theoretical issues in location choice model estimation* – DRCOG elected to estimate in-house two location choice models and the auto availability model. When location choice models are applied to zones, they actually are aggregate alternatives models, and make use of size variables, which are treated in estimation in a manner quite different from variables in fully-disaggregate models (such as mode choice models.) DRCOG staff had to go through a learning process to be capable of estimating such models, as our previous experience had been limited to fully disaggregate models. DRCOG modelers made use of several standard references in enhancing our knowledge in this area (Ben Akiva & Lerman, 1985, Koppelman and Bhat, 2006.)

*Consistency between estimation code and implementation code* – in any model estimation effort, code is created in the estimation software package, as variables are generated, tested, and used in the final model. The model application software must replicate this code, and it is easy to end up with application language syntax that is different from the scripting languages used in many of the estimation software packages (especially if the application language is object-oriented.) In trip-based modeling systems, which have very few variables used in their utility functions, this is a small effort. Our ABM has approximately 350 variables, making this a large task.

**Seven Knowledge Areas**

As discussed in the introduction above, it is not the purpose of this paper to take a position on how others intending to conduct an ABM development project should structure their models or pursue development of them. Nevertheless, we feel it appropriate to venture a few observations on the areas of ABM knowledge that are more easily acquired, and perhaps those that are more important to the successful use of such models by MPOs, transit authorities, or state departments of transportation. They are presented below, in what seems to us ascending order both of difficulty and importance of in-house knowledge:

*Model sensitivities* – one of the main points of building an ABM is the enhanced sensitivity to regional state variables that such models permit. DRCOG embarked on its ABM project because we were being asked to evaluate the effects on key outputs (VMT, air pollutant emissions, transit ridership, etc.) of a range of proposals, including extensive expansion of the rapid transit system, various approaches to an urban growth boundary, focusing development in urban and transit-oriented centers, emphasis on "new urbanist" development, and many other initiatives. An example of the superior sensitivity of our ABM is the variables in our mode choice model describing mixed-use density and intersection density (a proxy for pedestrian friendliness), which provide the ability to quantify some aspects of such development patterns and observe their effects on travel behavior. While the greatly expanded set of input variables represent a huge opportunity to answer such questions more effectively, it also represents a challenge, as there simply is more to understand in the new models. The complexity (on the "cost" side) and richness (on the "benefit" side) of the task increases more than linearly, as the new variables can interact in complex ways that take considerable analysis to understand fully. These realities strongly suggest that agencies developing ABMs should devote significant effort to understanding the models' input variables as completely as possible. DRCOG developed this knowledge by in-house staff writing the majority of the code to create the variables, including GISDK code in TransCAD, C# code for many of the model components implemented in our custom model implementation software, and SQL-Server database code. It should be noted that the project consultant team also is writing variable-generating code for model components assigned to them on the project.

*Limitations and approximations* – it goes without saying that all models have limitations, and are subject to numerous approximations. A model as complex as an ABM, while it is much more faithful to reality than a trip-based model, still has many approximations built in. Effective operation of any model is greatly aided by understanding these approximations. Some of these are related to small details of how a given variable is generated. Examples in our model include:

- A simpler approach to developing distance skims for the tour mode choice logsums, used as accessibility variables in other model components, than for the tour mode choice model itself, in order to save processing time.
- An at-times simplified approach to skimming for tour mode choice models. For a tour mode choice model, the most accurate way to generate skims is to skim the O-D direction for the out-bound time period of the tour, and the D-O direction for the home-bound direction. For distance skims, there is almost no difference between them, so we used twice the O-D distance.

Some approximations are related to larger model design issues. Examples in our model include:

- Estimation of sequential tour time of day and mode choice models. There are good theoretical reasons to estimate these two models as one joint model. However, there are practical drawbacks, such as difficulties in calibrating joint models.

- Modeling of intra-household interactions for the most part implicitly, rather than explicitly.  Explicit modeling might, for example, involve calculating an under-sixteen school child's mode choice, and if that mode was "passenger", then at that point generating/assigning a "serve passenger" tour.  However, this approach can significantly increase model complexity and run time, compared to a more implicit modeling of such interactions.

DRCOG staff developed detailed familiarity with these approximations again by coding the majority of the input variables in-house, which also required us to develop a detailed understanding of the theoretical structure of the model.

*Fixing and updating variables* – in DRCOG's experience, on-going operation of any model requires small but regular adjustments of various kinds, including how variables are generated. While some MPOs may have consultants handle such adjustments, it can be faster and easier to handle (at least) the smaller ones in-house.  Examples of types of such adjustments in the DRCOG ABM include:

- Changes in the TransCAD GISDK code to modify the development of skims (relatively easy.)
- Changes in the C# code to modify variables that are generated in their final form there (also usually not difficult.)  For example, development in the school location choice model of an "older sibling school location" variable had to be done in the C# code of the model, and so required at least reasonable facility with that programming language. Changes to variables residing in our SQL-Server data engine may also be required in similar circumstances, requiring functional knowledge of that software package.
- Changes to the more fundamental operation of the code (for example, changing the operation of the logit solver to improve its performance.)  This is more challenging, requiring, in this example, more knowledge of the mathematics of the logit model.

DRCOG staff developed familiarity with these aspects of our model by (with coaching/advice from our consultant team) developing the basic code that drives any discrete choice model in our ABM, with coaching and advice from our consultant team.

*Recalibrating the model* – it appears quite common for MPOs and other custodians/operators of travel models to leave recalibration to consultants.  However, it again may be convenient (faster, cheaper, easier) to perform some recalibration in-house.  Many types of recalibration may be necessary, and with a new model, and one as complex as an ABM, we expect our model to need frequent recalibration for some time to come.  A few hypothetical examples of recalibration we might expect include:

- Response to difference between ACS journey to work pattern and the ABM work location choice model results, perhaps showing too few people working at home.  In this

case, model coefficients in the “work at home” nest might need adjustment, or it might be that some of the person characteristics in the synthetic population need adjustment.
- Too few light rail boardings. If for example, the problem was found to be that there were too few university students riding the light rail, several issues could be causing this error. It might be the case that the university tour table was in error, with the average length of university trips too short. This could cause the university tour mode choice model to put too many trips on walk and bike modes. So the mode choice problem could be corrected by fixing the school location choice model.

DRCOG staff developed sufficient familiarity with our ABM model, as well as experience with model estimation/calibration, through estimation of several models (work and school location choice, and auto availability.)

*Re-estimate the model* – While in many cases, perhaps the majority of cases, it may be best to leave re-estimation to consultant experts, there are many advantages to developing and maintaining in-house the capability to re-estimate some model components. First, through developing this capability, modeling staff also will develop a much more complete understanding of their own model. Second, consultants often do not have the budget flexibility to respond to the new circumstances that may present themselves in model development projects, especially leading edge models like ABMs, in which new insights and changes of direction are common occurrences. For example, during DRCOG’s estimation of the school location choice models (see DRCOG 2008), we realized that the use in the k-8 and high school sub-models of a variable that represented the school location choice of an older sibling would be both highly desirable and practical. However, this variable had not been developed during creation of the estimation datasets. We also encountered a situation in which the initial estimation data was found to be in error (an occurrence that is very hard to avoid entirely.) As DRCOG was less constrained on budget than the consultant team, DRCOG simply created the new variable, made corrections to the erroneous data, and incorporated the new data into the estimation process. In the course of developing the capability to perform such work, DRCOG had to learn many subtleties of model estimation software syntax (see ALOGIT Software and Analysis, 2007) , and of the mathematics associated with aggregate choices and aggregate choice variables in discrete choice modeling, with invaluable assistance from one of our consultant team as a coach during this process. As frequent model improvements are expected to be necessary for some time to come, and as that need is likely to continue after the consultant’s budget is exhausted, such capability is of great value to the long term success of the model.

*Re-design the theoretical structure* – Of all the changes expected in our ABM in the future, this more than any other may be best left to consultant experts. Some of the more straightforward possible upgrades may be practical to handle in-house, such as substituting a new population synthesizer for the existing one, or perhaps even estimating and implementing a joint mode/time-of-day model in place of the existing sequential treatment of these choices. However, more explicit modeling of intra-household interactions can affect many aspects of the model, requiring numerous complex changes, and so requiring a very high-level grasp of model design. DRCOG spent considerable time working with our consultant team to understand the details of interactions between model components in our ABM; this was again necessary to code the model effectively, and to operate it effectively in the future. In implementing major upgrades in the future, we likely will make significant use of consultant assistance.

*Re-design software* – One of the essential values of the object-oriented approach to software development is the potential ease in upgrading the software over time. Well-written OO code is modular, and new functionality often can be substituted for old with relative ease. The novelty of ABM models, and the fact that much of DRCOG's code base has been developed "from scratch" makes such upgrades likely in future years. Upgrades may involve small, focused code elements such as the logit equation solver, or they may involve larger enhancements to the ability of the code execution to be distributed over multiple cores or servers. At minimum, knowledge of the code sufficient to "plug in" such upgraded elements will facilitate model maintenance over time. DRCOG anticipates completion of model code sufficient to conduct calibration/validation work in 2009. However, many code upgrades are expected to be necessary to develop a model version that is sufficiently user-friendly for roll-out to consultants and other users. Knowledge of our ABM code has been developed primarily by writing most of it in-house, again with invaluable coaching from members of the consultant team, and from former DRCOG staff.

**Some Details Concerning In-House Work**

*Model estimation* – While DRCOG estimated several model components in-house, from our current vantage point near the end of the project, we feel that we could practically have estimated several additional models in the context of this project. Figure 1 shows the major model components in DRCOG's ABM, with a grey-scale code showing those estimated by DRCOG, those transferred from our trip-based model, those best left to consultant experts, and those DRCOG did not estimate but could have. The models DRCOG could have estimated include mode choice models, which in some cases are comparatively simple, and location choice models, which are similar in many respects to other location choice models DRCOG staff estimated. It should be noted, however, that taking on estimation of the tour mode choice model puts a client modeling agency in the heart of the project's critical path, as that model typically supplies the accessibility variable logsums for many other models. This model therefore must be estimated first in the modeling process, so modeling organizations contemplating in-house estimation of some models may want to consider the very significant effects on project schedule that estimating this model in-house may have.

*Software development* – while DRCOG coordinated extensively with our consultant team concerning design of some aspects of the model software, the vast majority of it was designed and written by in-house staff. The software architecture (elements that may be thought of as the "wiring" of the model, since they pass information and signals between various functional elements) was written by DRCOG's IT lead (now working as a consultant.) Initial drafts of model components also were written by the IT lead; however, most of the work on model components, and on the SQL-Server data engine, has been conducted by travel modeling staff, with coaching from the consultant team and our former IT lead. To accomplish these tasks, travel modeling staff expended extensive effort improving our knowledge of logit mathematics, as well as IT tools such as object-oriented programming, specifically C#, and SQL Server.

Decisions surrounding software development for DRCOG's ABM were almost all difficult ones, and there is not space in this paper to go into them in detail. However, the following overarching issues, and our response to them, may be of use to modeling organizations considering initiating their own ABM projects:

- Vendor software versus other options. At the time DRCOG initiated its ABM project, ABM-supportive software from traditional transportation software vendors did not appear to be in a mature state. Since that time, several different types of ABMs have come under development or into operation. This situation appears to us to present challenges to software vendors, as it increases the difficulty of their decisions regarding what sort of product to bring to market. However, working closely with a traditional vendor is an option that should be given careful consideration, given their experience and expertise in software development, and the advantages such an approach provides for future software maintenance.
- Open source versus commercial software environments. The fully open source approach often seems to gravitate toward free software tools (e.g. Python or R languages, MySQL or PostgreSQL databases, etc.) Obviously these tools have the advantage of being free, and in some cases of having large communities of users participating in the open source development model. Advantages and disadvantages of this approach versus the use of commercial tools often are the subject of controversy; having said that, the Microsoft-based approach DRCOG has taken has provided us with a very robust and effective development environment (Visual Studio), and a powerful, widely-known database tool (SQL-Server) which, taken together, so far have supported a successful and efficient development and operational process.
- Broader industry-standard tools, versus tools specific to the regional modeling field. At the beginning of our ABM project, one of DRCOG's primary goals was to break the "stovepipe" that travel models and travel model data traditionally have inhabited. By "stovepipe" we mean an isolated status which made model input and output data difficult to prepare, access and distribute. Use of broad IT industry standard tools was a key part of the effort to break that isolation, as tools like SQL-Server permit relatively easy development of software that can automate access to information like the results of the population synthesizer, model-generated trips and tours, etc. The goal is to make the travel model much more relevant to our colleagues and customers than it has been in the past.

**Figure 1: Feasibility of In-House Model Estimation by Component**

**Dark Gray=Estimated by Consultants/ Could Not have been estimated by DRCOG**

**White=Estimated by DRCOG**

**Black=Used existing trip-based model component**

**Light Gray= Estimated by Consultants/Could have been estimated by DRCOG**

| **COMPONENT NAME** | |
|---|---|
| Synthetic Sample Generator | Tour Primary Destination Choice |
| Regular Workplace Location | |
| Regular School Location | Tour Main Mode Choice |
| Auto Ownership | |
| Area Type | Tour Time of Day Choice |
| Parking Cost | Intermediate Stop Generation |
| Daily Activity Pattern | Intermediate Stop Location |
| | Trip Mode Choice |
| Exact Number of Tours | Trip Departure Time |
| Work Tour Destination Type | |

## Future Training

As DRCOG goes into the future with its new ABM, and as we decide upon and implement upgrades to it (as we do regularly with our trip-based model now), we anticipate the need for an on-going training program. Some of that work, perhaps the most important, is going on now: the wider dissemination, in-house, of skills often acquired by particular individuals as they led a particular technical element of the project. Regular training classes in C# are given, as are classes in SQL-Server, and in discrete modeling theory. This training is crucial to DRCOG's ability to maintain and operate the new ABM over the long term.

In addition, DRCOG expects to implement improvements to the theoretical structure of the model in future years, and expects to seek additional training in a number of areas. Examples include:

- Better understanding of modeling approaches that more explicitly model intra-household interaction. Some existing models (such as that in Columbus, OH) already model such interactions more explicitly that DRCOG's ABM does. We expect to seek further information and training concerning such approaches, to help us evaluate the need for them in our region, and to decide how best to upgrade our model to implement them.
- Fully disaggregate location choice set elements. The Sacramento Area Council of Governments (SACOG) model operates its location choice models at the parcel/point level, whereas DRCOG's ABM operates at the TAZ level (and then assigns locations to points based on the Monte Carlo process.) DRCOG modelers expect to study the SACOG approach, and perhaps to implement it in our model in the future.
- At present, our location choice models are, in effect, singly constrained. The work tour productions, for example, match zonal totals by necessity, as each worker in a zone generates a tour. However, the attraction end has no constraint. There are several ways to implement such a constraint (for example, shadow pricing that reduces the utility of any given zone as more workers select it.) DRCOG staff also will be studying such methods in the near future, and will implement them as they seem necessary and practical.
- A number of researchers are implementing enhancements to population synthesizers, and DRCOG will be studying and monitoring that work, and may adopt new approaches in the near future.

DRCOG also is working on implementation of dynamic traffic assignment (DTA) and next-generation land use models in the region, and will be studying these approaches and their connection to ABMs in future years.

**Conclusion**

The most effective step DRCOG staff took to aid us in learning our new ABM model might be thought of as a very sharp "two edged sword": that is, we put ourselves in the critical path in several places. On one edge of the sword, we were forced to learn, and we did learn. Paraphrasing one of our project staff, "desperation is the mother of knowledge." There were many points in the project at which there was no one but DRCOG staff available to accomplish a given task, and we didn't know how to do the necessary work, so we learned. On the other edge of the sword, this approach had significant effect on the schedule of the project. At this point in the project, the trade-off of schedule for knowledge seems like a good one. However, the coaching model we employed was indispensible to DRCOG accomplishing several of these critical path tasks, and producing a good result in them. These included model estimation, C# coding, and SQL Server database development.

It may well turn out that, as more and more ABM projects are initiated and concluded, a better way will be found (or already has been found) to ensure that staff at modeling agencies understand the new models they must maintain and operate. It is possible that, in the wake of the first few ABM model projects, the next group of projects can be completed, and modeling agency staff be trained to use the models, in a way that takes advantage of the work of the earlier projects, and does not affect project schedule significantly. In any case, we believe that it is critical to the widespread and successful transition in modeling agencies from trip-based models to ABMs that such knowledge also becomes widespread. We look forward to watching others' projects, to see how they manage this necessary transition.

## REFERENCES

Project Management Institute, A Guide to the Project Management Body of Knowledge, Third Edition, 2004.

Ben Akiva, Moshe, Steven R. Lerman, Discrete Choice Analysis, Massachusetts Institute of Technology Press, 1985.

Koppleman, Frank, and Chandra Bhat, A Self Instructing Course in Mode Choice Modeling: Multinomial and Nested Logit Models, United States Department of Transportation, 2006.

ALOGIT Software and Analysis, Ltd., ALOGIT User Manual Version 4.2, 2007

Denver Regional Council of Governments, Regular School Location Choice Models Documentation – Draft 4, internal working draft, 2008.