

# UC Berkeley

## Research Reports

### Title

PLANTiS: Organization And Integration Of Modules

### Permalink

<https://escholarship.org/uc/item/67k0332k>

### Authors

Paramsothy, Thananjeyan  
Khattak, Asad  
Lovell, David  
et al.

### Publication Date

1995

**This paper has been mechanically scanned. Some errors may have been inadvertently introduced.**

CALJFORNIAPATHPROGRAM  
**INSTITUTE OF TRANSPORTATION STUDIES**  
**UNIVERSITY OF CALIFORNIA, BERKELEY**

## **PLANiTS: Organization and Integration of Modules**

**Thananjeyan Paramsothy**

**Asad Khattak**

**David Lovell**

**Adib Kanafani**

**California PATH Research Report**  
**UCB-ITS-PRR-95-18**

This work was performed as part of the California PATH Program of the University of California, in cooperation with the State of California Business, Transportation, and Housing Agency, Department of Transportation; and the United States Department of Transportation, Federal Highway Administration.

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the State of California. This report does not constitute a standard, specification, or regulation.

June 1995

ISSN 10551425

Institute of Transportation Studies  
University of California at Berkeley

**PLANITS: ORGANIZATION AND INTEGRATION OF MODULES**

Thananjeyan Paramsothy

**Asad Khattak**

David Love11

Adib Kanafani

109 McLaughlin Hall  
Berkeley, California 94720

Phone (510) 642-3585

Submitted to:

The California Department of Transportation

(MOU-52)

Draft Final Report

Volume 2

First submitted in December 1994

## **ACKNOWLEDGEMENT**

This research was conducted at the Institute of Transportation Studies, University of California at Berkeley. It was funded by the California Department of Transportation, as a part of the PATH Program. Current **PLANiTS** team members include: Adib Kanafani, **Asad** Khattak, Randall **Cayford**, David Lovell, Thananjeyan Paramsothy and Rosella **Picado**.

## **ABSTRACT**

This paper discusses the integration of **PLANiTS** (Planning and Analysis Integration for Intelligent Transportation Systems) components. The integration is achieved by defining a structure for representing transportation improvement actions, performance measures and environment in terms of spatial, temporal and user dimensions. We also discuss key issues related to computer implementation including data dimension, a data dictionary, data modeling, and data transformations. In the paper, we develop a basis for the next implementation of **PLANiTS**.

**Key Words:** Intelligent Transportation Systems, Transportation Planning, Policy, Decision Support System, Software Engineering, System Integration.

## EXECUTIVE SUMMARY

The objective of this paper is to present the organization and integration of various PLANiTS (Planning and Analysis Integration Transportation Systems) modules using the object-oriented design and programming methodology. A key advantage of the object-oriented methodology is that it allows the development process to be divided into manageable independent parts.

PLANiTS modules include: policy base, action base, database, case-based reasoner, expert system, methods base, deliberation tools and help tools. They are controlled by the PLANiTS main module that can draw information from and feed information to these foundation modules. The Main Module is the backbone of PLANiTS and bears the responsibility for leading PLANiTS users through analysis and deliberation. It uses the Planning Vector (PV) that consists of actions, performance measures and environment, defined in terms of spatial, temporal and user dimensions to inform the user of possible evaluation strategies and predict the outcomes of the selected strategies.

An object-oriented data model is used to represent the relationship between PLANiTS modules. Modules will communicate with other modules using messages. When a module receives a message from another module, it will take the necessary actions and respond back with the appropriate reply message. Each module is also a complex conglomeration of many sub-objects. Conceptually, the PV is represented as a table consisting of the following attributes: actions, **performance** measures, and environment. The network data model is an object-oriented data model based on the spatial relationship between zones, streets, intersections etc. The foundation modules include many sub-objects in the form of tables. The relationships between tables are illustrated using an example from the policy base in this report.

One of the major challenges of this project is the transferring of data and results between PLANiTS modules; in particular, transferring data between the models in the methods base.

Since transportation planning and operational models need data in different dimensions, data transformation rules have to be identified for each possible transformation. Issues related to data transformations are also addressed in this report. Instead of defining transformation rules for each possible attribute-dimension combination, the complexity of this problem can be reduced by classifying the attributes into different categories and identifying transformations rules for each category-dimension combination. The complexity can be further reduced by identifying a few base dimensions and conducting all data transformations through these base dimensions.



## LIST OF FIGURES AND TABLES

	Page
ACKNOWLEDGEMENT	ii
ABSTRACT	iii
EXECUTIVE SUMMARY	iv
1. INTRODUCTION	1
2. PLANiTS MODULES	1
2.1 PLANiTS Main Module	6
2.2 The Database	8
2.3 The Action Base	9
2.4 The Policy Base	10
2.5 The Model Base	11
2.6 The Deliberation Tool	12
2.7 The Help Tool	13
2.8 The Expert System	13
2.9 The Case Based Reasoner	13
3. INTEGRATION OF PLANiTS MODULES	14
3.1 OOPS Concepts and Notation	14
3.2 Data Structure and Modelling	16
4. SUMMARY	30
REFERENCES	31
APPENDIX	32

## LIST OF FIGURE AND TABLES

<b>List of Figures</b>	<b>Page</b>
Figure 1. PLANiTS Architecture	5
Figure 2. PLANiTS Analysis Process	7
Figure 3. Object Notation	15
Figure 4. PLANiTS Module Structure	17
Figure 5. Network Data Model	22
Figure 6. Example of Data Model for the Policy Base	24
Figure 7. Methods for Data Transformations	28
<b>List of Tables</b>	
Table 1. Planning Vector Definition: High Occupancy Vehicle Example	19

## **1. INTRODUCTION**

To address urban transportation planning problems, we have developed a methodology called **PLANiTS** (Planning and Analysis Integration for Intelligent Transportation Systems). A detailed description of **PLANiTS** appears in Kanafani, Khattak, Crotty and Dahlgren (1993) Kanafani, Khattak and Dahlgren (1994) and Vlahos, Khattak, Kanafani, and Manheim (1994). This new planning methodology integrates transportation planning and operations models with knowledge-based systems and electronic group decision support.

This report describes the organization and development of the **PLANiTS** computer model. The main emphasis is on the general **PLANiTS** idea and the integration of various modules. It presents the **PLANiTS** modules and their functions, notation used in the development of the computer model, data transformations and other related issues. The primary objective is to enhance the ideas conceived during **PLANiTS** prototype development (Cayford, Khattak, and Kanafani 1995) and develop ideas for the next versions. Some of the ideas presented here are not implemented in the current **PLANiTS** prototype. The methodology presented in the paper forms the foundation for our next **PLANiTS** model.

First, we present the complete structure of **PLANiTS**. It elaborates the functionalities of the **PLANiTS** modules. The second part presents the issues related to implementation of the model in a computer, including the theoretical layout of the modules and issues related to integration of the modules. Key issues such as data modeling, notation, rules for data transformations and data dictionary are also discussed.

## **2. PLANiTS MODULES**

From the **PLANiTS** architecture described in Kanafani et al. (1994), we have chosen the components that will be implemented in our next version(s) of **PLANiTS**. In the remainder of this report, we will discuss the issues related to this narrowed down version. The main reason for limiting our scope is to develop a practical **PLANiTS** tool within the time and resource limitations.

**PLANiTS** consist of four main elements:

1. Database and **Knowledge Base**
2. Strategy and Action Base
3. Policy and Goals Base
4. Methods and Tools Base

Deliberative planning and analysis processes occur in the Planning Vector, **PV**, which has three sub-vectors:

- Action Vector, **A**, contains the proposed set of actions that are the subject of the planning process.
- Environment Vector, **E**, contains the descriptors of the context that are relevant to the subject actions.
- Criteria Vector, **Y**, contains the measures of performance representing the goals for which the actions are proposed, and the criteria by which they are evaluated.

The planning vector is then, **PV = [A, E, Y]**

We can divide the four **PLANiTS** elements into the following independent modules:

- Data Base (DB)
- Action Base (AB)
- Policy Base (PB)
- Model Base (MB)
- Case Base Reasoner (CBR)
- Expert System (ES)
- Help Tool (HT)
- Deliberation Tool (DT)

Residing in a layer above these modules is the **PLANiTS** Main Module (PMM). The PMM can draw information from, and feed information to, these foundation modules. The means by which this is done is determined in part by the type of action, measures of performance and environment requested by the user, and by other choices made by the user regarding the specific direction of the analysis. The PMM is the backbone of **PLANiTS**, and bears sole responsibility for leading the user from a Planning Vector of interest to a set of predicted outcomes. The PMM is always aware of the status of the Planning Vector, and uses this information to decide its subsequent activities. It can also modify the Planning Vector, with the permission of the user, based on deliberation and information obtained from the Policy and Action Bases.

The Data Base contains all factual information available to **PLANiTS**. This information can be related to physical entities, such as the transportation network and improvement actions, to performance measures, such as the current values of various attributes, to the environment, or to historical facts, such as the results of previous analytical studies or implementations.

The Action Base contains a list of transportation improvement actions and provides additional information in response to user selected actions contained in the Planning Vector. This includes the suggestion of alternate actions, associated actions, or linked actions. For example, the action tool may inform the user that all ramps must be metered in certain areas to achieve certain goals. The Action Tool can also be used for exploratory purposes or as a help tool while creating and modifying the Planning Vector.

The Policy Base is a dictionary of goals and requirements that are sensitive to the type of user (organization/agency or group) using **PLANTS**, the type of funding intended for the candidate projects, and the specific type of project being proposed (e.g., roadway or transit). The Model base is a set of models and transformations that help bridge the gap between existing data and predicted measures of performance for a particular action in an environment. Each of these

modules is directly accessible to the user via a browser. For instance, the Data Base browser will include a Geographic Information System or GIS-type of facility to view the transportation network on the screen, while the Policy Base browser will present policies and explanations in a more traditional text-based data format.

PMM can evaluate the impacts of actions in terms of performance measures based on data and models provided by the AB, PB, MB, and DB. In addition, it is also connected to four support agents, each of which can be invoked any time by the user. These are the Case-Based Reasoner (CBR), the Expert System (ES), the Help Tool (HT), and the Deliberation Tool (DT). They serve the purpose of providing additional information to the user that may guide choices among alternatives presented by the PMM. At any point during the analysis, if there is some ambiguity regarding how to proceed or another reason to involve the user, the PMM will leave the specific choice up to the user. **PLANiTS** user may choose to use one or more of these support agents to facilitate decision-making, or may use resources already at his or her disposal.

The Case-Based Reasoner is a tool that searches the Data Base for case histories and performs a matching process to identify similar projects. The Help Tool can be invoked from within the PMM, and will provide context-sensitive help on the current activity. The Deliberation Tool enables the user to send and receive information from other users, to compare and contrast alternatives, and to seek consensus on decisions. The Expert System is a sort of “watchdog” that records significant events during the analysis process, and can be queried as to the best approach to a particular problem, and will give answers based on the data it has collected. There are expert systems that can provide advice on enhancing the planning vector or exploring whether an action will be feasible in the current context.

The architecture of **PLANiTS** is illustrated in figure 1. Each of the foundation modules are objects that can be used independently. Except for the **PLANiTS** Main Module and the Help Tool, each of the higher-level modules can be invoked independently, but will not allow invoking another object from these objects. For example, users can browse through the model base,

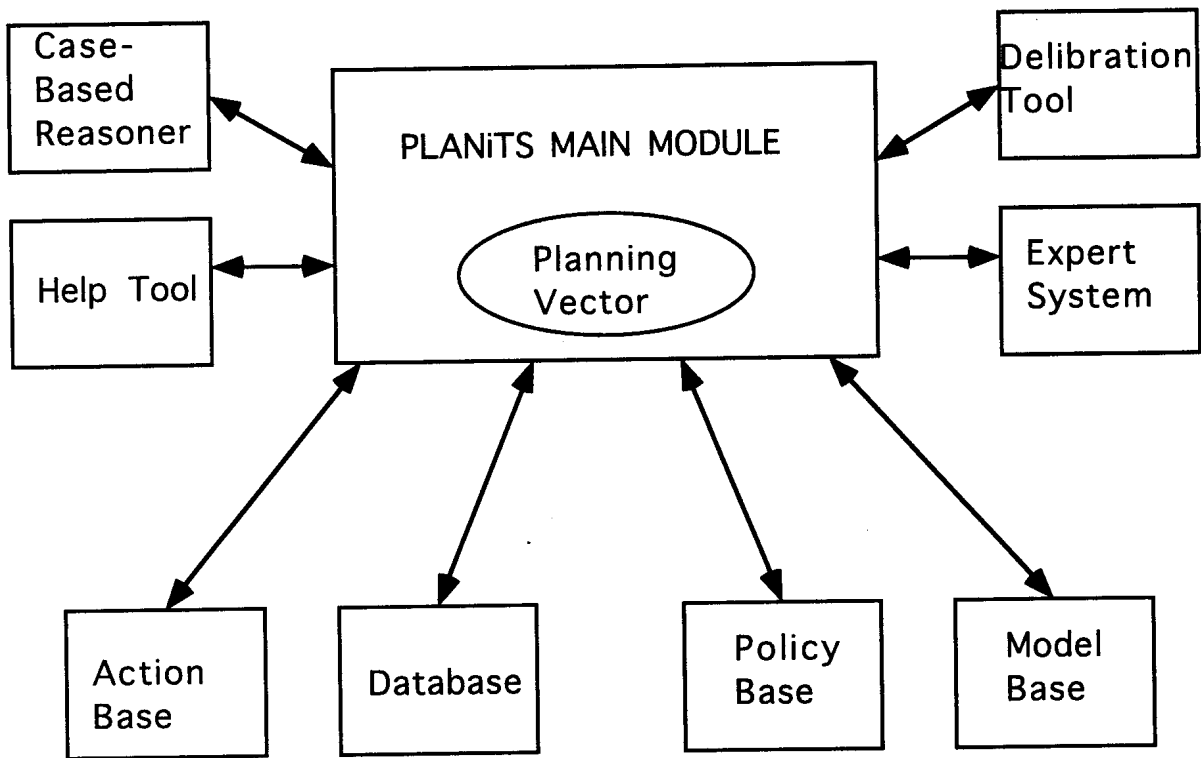


Figure 1. PLANiTS Architecture

add/delete cases in the Case Based Reasoner, and deliberate about a policy **without any** analysis, i.e., without specifying actions, measures of performance and an environment (without specifying a Planning Vector). But the AB, DB, PB, MB, CBR, DT, and ES cannot be invoked from within each other. The Help Tool also can be invoked independently and also from within any of the modules any time during any process. From within PMM, all of the above modules can be invoked. However, when other modules are invoked from the PMM, the invocation will be context sensitive, i.e., it will respond to the contents of the Planning Vector found in the PMM.

We will describe briefly the basic functionalities of **PLANiTS** modules (more detailed functionalities appear in Khattak and Kanafani 1995a). The PMM is presented first, as it is the primary agent in **PLANiTS**. Then the modules upon which the PMM is built are described. Other **PLANiTS** reports present further details and concentrate on these modules when invoked from the PMM (see Picado, Khattak, Love11 and Kanafani 1995 for a description of the methods base).

## **2.1 PLANiTS Main Module**

The **PLANiTS** Main Module (PMM) conducts all analyses in **PLANiTS** with Actions, Environments, and Measures of Performance. For example, the user may wish to find the impacts and implications of certain actions in some environment such as adding an HOV (High Occupancy Vehicle) lane on a freeway between two predetermined locations. As a first step in the analysis process, the user will create a Planning Vector (PV) specifying the context of application. An example of the analysis process is illustrated in figure 2.

In the planning vector, the user will indicate the action(s), environment and the measure(s) of performance of interest. Preliminary lists of Actions, Environments, and Measures of Performance are presented in appendix 1. These Actions, Measures of Performance and Environment will be entered through a series of menus and/or graphical screens. These will be specified in terms of spatial, temporal and user dimensions. First, **PLANiTS** will check the database to see whether the required results (performance measure estimates with and without the



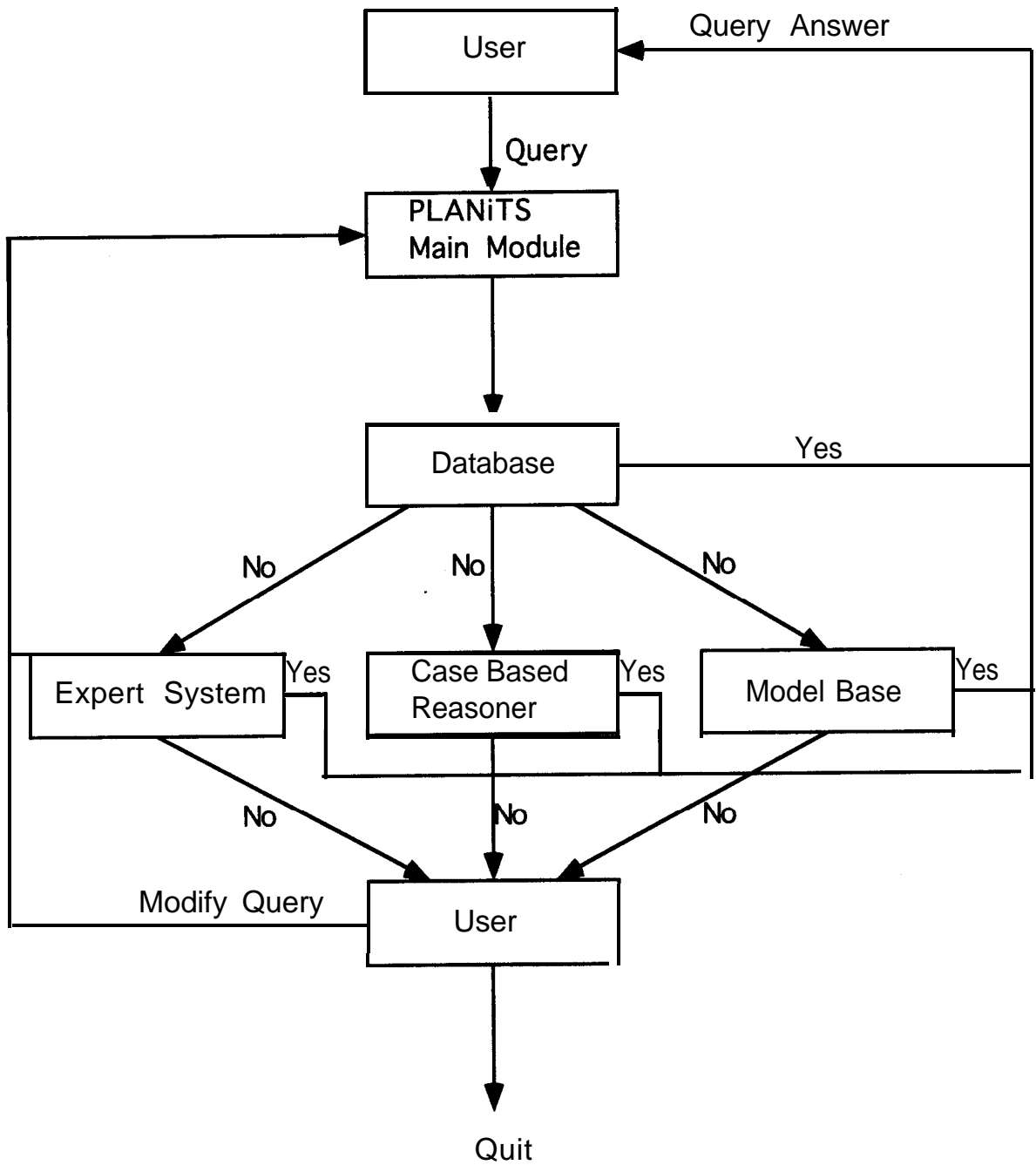


Figure 2. PLANiTS Analysis Process

actions) are already available. If it finds them, it will present the results to the user. Otherwise, it will then present a set of alternate methods (for example, a list of models and/or the Case Based Reasoner) by which the impact of the action can be evaluated. Once the user has selected the method of analysis, the PMM will take the necessary step(s) and evaluate the measures of performance contained in the PV. If PLANITS could not find an appropriate model, it may inform the user to modify the query and re-execute the analysis. At every step, when more than one feasible method is available, **PLANITS** will prompt the set of options from which the user can select the method of analysis. The best method, as perceived by **PLANITS**, will be highlighted, if such information is available from the Expert System.

Any other main module can be invoked from within the PMM. These can be invoked before or after the analysis phase, and the invocation is context sensitive, that is, the modules will respond to the contents of the PV.

The PMM also provides extensive diagnostic information, i.e., alerting the user to an inconsistent network, insufficient data and/or violations of policies based on the contents of the PV. Besides its own set of tests, this function may invoke the diagnostic tools found in the Policy Base, Action Base, Model Base and Data Base to run diagnostic tests. This tool can provide diagnostic messages based on both the initial PV and the results of the PMM analysis (obtained from traffic models, Case Base Reasoner, etc.).

## **2.2 The Database**

The **PLANITS** Data Base (DB) is the main repository of all network, trip and socioeconomic data. The database functions are similar to a GIS with the data kept at fine detail. The primary reason for such disaggregate levels is the need to translate the data for use by various models. In order to run various transportation models, **PLANITS** must import data and transform it to suit the models (present it in model input format). It must also translate the model outputs to **PLANITS** understandable format. Since data can be aggregated to suit the needs of models with

less/no errors than disaggregating them, data will be kept at the most disaggregate level required (data transformations are discussed in the next section). To simplify data input and queries, most interaction screens will be enhanced by graphical displays with windowing systems. The DB will support the following functionalities.

- **CREATE/EDIT:** Create/edit and modify the contents of DB including network, trips and socioeconomic data.
- **DISPLAY/QUERY:** Display the network, socioeconomic and trip tables, and display results of simple queries.
- **DIAGNOSTIC TOOL:** Check for consistencies in network, trip and socioeconomic data.
- **IMPORT:** Import data from ASCII test files into DB.
- **EXPORT:** Output data from DB into ASCII files.
- **HELP TOOL:** Invoke the main Help Tool.

### **2.3 The Action Base**

The Action Base (AB) is like a thesaurus of action definitions and includes an intelligent agent. Given a particular requested action, particularly with some additional context supplied by the Environment and Measure of Performance elements of the Planning Vector, the Action Base can return many suggestions to enhance and develop the Planning Vector. The following functions are supported:

- **SUGGEST ALTERNATIVE ACTIONS:** Actions that accomplish the same goals, in the same type of environment, will be suggested.
- **SUGGEST ASSOCIATED ACTIONS:** Actions that are typically complementary to the proposed action will be suggested. For instance, if HOV or bus-only lanes are proposed, then the Action Base might suggest the construction of park-and-ride lots, or the inception

of a real-time rideshare-matching program.

- **SUGGEST LINKED ACTIONS:** It may be possible to achieve performance improvements in discrete, chronological steps. If the proposed action could be followed easily by a similar action on a larger scale, this would be suggested for the long-term. If largely similar results can be achieved by a smaller project, then this might be suggested for the short-term.
- **DIAGNOSTIC TOOL:** This function would ensure compatibility between the proposed action(s) and other elements of the Planning Vector.
- **HELP TOOL:** Invoke the main Help Tool.

## **2.4 The Policy Base**

The PB primarily consists of a list of performance measures, a database of cases and an intelligent agent that can be used to determine the policy implications of Actions, Environments, and Measures of Performance. For example, the user may wish to find out the policy and funding implications of adding an HOV lane on a certain section of a freeway or the user may wish to include a certain performance measure related to a policy goal. The PB can be invoked from the PMM or independently. When invoked from the PMM, the PB will respond to the attributes specified in the PV. Using keywords as identifiers, the PB will retrieve the relevant policies from the PB repository that can be used to enhance and develop the PV. If invoked independently, the PB will display an empty input screen requesting the user to input the context of application. The PB performs the following functions.

- **BROWSE:** List the relevant policies retrieved from the database.
- **SELECT:** Select the highlighted policy and include it in the current analysis. For example, this tool can be used to include additional policies that were not mandated, but the user wishes to conform to (i.e., include policies that were not retrieved based on the keywords

in the PV).

- **SELECT POLICY FACTORS:** There are operationalized policy factors that users can check to ensure that the project is policy relevant. For example, if the user is interested in **ISTEA** (Intermodal Surface Transportation Efficiency Act), “congestion relief” factor, then **PLANiTS** would suggest delay as a performance measure.
- **SHOW POLICY:** Display more details of the highlighted policy.
- **ADD:** Add a new policy to the database. Once a policy has been added, it will be also used in future analysis by the PB.
- **DELETE:** The delete function will have two options: (1) Delete the selected policy from the current analysis, (2) Delete from the PB data base. If a policy is deleted from the current analysis, it will still be stored in the database, and can be used in other analyses. However, if a policy is deleted from the database, the policy will not be available for future analysis of **PLANiTS**.
- **DIAGNOSTIC TOOL:** The tool compares the attributes of the selected policies with the elements specified in the PV. This function enables the comparison of the retrieved policies with the current PV so the user can make informed decisions regarding relevant Actions and Measures of Performance. The Diagnostic Tool can be specified to respond to both the results of the analysis and the initial attributes of the PV.
- **HELP TOOL:** Invoke the main Help Tool.

## **2.5 The Model Base**

The Model Base (MB) contains the list of all transportation models that can be used by **PLANiTS**. The MB can also be invoked from the PMM or independently. When invoked from the PMM, the model base will respond to the contents of the PV and will run the requested model and return the results to the PMM. (The import and export of data files is done automatically.) The user cannot add or delete models in the MB. However, the user may disable certain models. The

disabled models will not be run by the PMM. The MB will have the following functions.

- **BROWSE:** List the models available in the model base. While browsing, the user may disable or include models that may be used by the PMM.
- **RUN:** Run the selected model in context to the contents of the PV. If the MB was not invoked from the PMM, then MB will run the models independently. In such cases, the user must provide input files in the specified formats. (The import or export of data files is done automatically if invoked through PMM.) However, the user may be able create data input files and import output files using **CREATE DATA** and **IMPORT DATA**.
- **DIAGNOSTIC TOOL:** Check the **PLANiTS** database for the availability of data to run the selected model and list the assumptions made in data transformations, etc., for importing and exporting data. When more than one transformation is possible, list the choices so that the user can specify the required method of data transformation
- **CREATE DATA:** Create data files. for the selected model(s) from the **PLANiTS** data base.
- **IMPORT DATA:** Import the data files or result files from model(s) into the **PLANiTS** Data Base.
- **HELP TOOL:** Invoke the main Help Tool.

## 2.6 The Deliberation Tool

The Deliberation Tool (DT) is intended to facilitate a group decision-making process (Vlahos et al. 1994). At the most rudimentary level, it is an electronic mail system, which allows concurrent users of **PLANiTS** to send and receive messages regarding their current or planned activities and PV specifications. In this manner, they can make suggestions and seek advice from other users. This is a means of ensuring communication regarding proposed projects.

At a higher level, the Deliberation Tool provides the means to seek consensus between participants on a particular issue. This is accomplished via the presentation of all alternatives,

ranking and voting by each individual user, and then ranking and selection of an overall preferred alternative. Any issue that arises in PLANiTS is subject to group deliberation; a specific example might be the choice of coefficients used to calibrate a particular planning and operations model.

## **2.7 The Help Tool**

The Help Tool (HT) is a global help function that can be invoked at any point during the PLANiTS session. The Help Tool will contain indexes and explanations on all attributes of the PV, PMM, CBR, PB, MB, DB, and Deliberation Tool. The help tool also includes directions and examples of usage for all functions available in PLANiTS.

## **2.8 The Expert System**

The Expert System (ES) will record statistics about various choices made by users while performing PLANiTS operations, and can then in the future make recommendations based on these past observations. The Expert System can also be programmed exogenously to suggest certain courses of action under certain circumstances. This would occur when a certain alternative is preferred to others (but not based on information provided by the Case Based Reasoner or the Policy Base). The system can provide advice on whether to implement an action or on how to modify the planning vector.

## **2.9 The Case Based Reasoner**

Case Based Reasoner (CBR) is an intelligent agent that can be used to estimate the measures of performance based on similar studies or data collected elsewhere (Khattak and Kanafani 1995b). For example, the user may wish to determine the savings in travel time due to the implementation of HOV lanes based on results of other similar studies. In the CBR, the user will specify similarity thresholds (e.g., High, Medium, Low) to control the cases selected for analysis. The CBR will then perform appropriate statistical analysis based on the selected cases

and provide estimates of measures of performance.

### 3. INTEGRATION OF PLANiTS MODULES

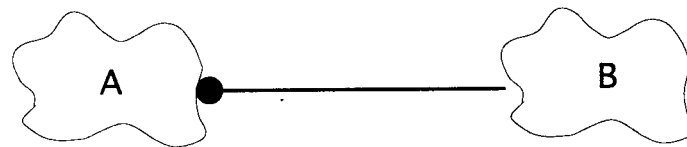
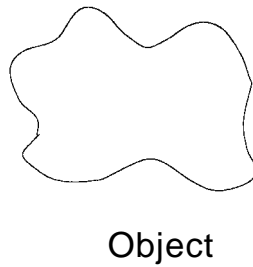
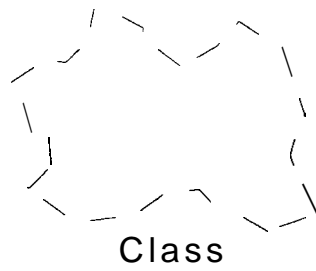
In this section, we present the issues and ideas related to the integration of the various modules. We will also introduce some basic concepts and notation in Object-Oriented Programming (OOP) such as classes, objects, messages and data models. A key advantage of the OOP methodology is that it allows the development process to be divided into several small independent parts. OOP has been developed in the Computer Science literature and we will only provide brief definitions of the OOP terminology necessary to express the ideas in this paper. For the development of PLANiTS, we will use the concepts and notation developed by **Booch** (1993) and White (1994).

#### 3.1 OOPS Concepts and Notation

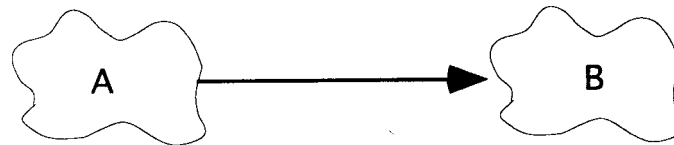
Some important **Booch** (1993) notations are illustrated in figure 3 and explained below.

- **Class:** A class captures the common structure and common behavior of a set of objects. A class is an abstraction of real-world items.
- **Object:** An object is an instance of a class. An object has state, behavior, and identity.
- **Association Relationship:** An association represents a semantic connection between two classes. The main two relationships used in this report are (1) Has Relationship and (2) Inherits relationship. The has relationship shows the whole-and-part relationship. The cardinality of the relationship is shown at the applicable ends of the relationship. An inherits relationship between classes shows that the subclass shares the structure or behavior defined in one or more of the super classes.
- **Link:** A link is an instance of a relationship or association.
- **Message:** Objects communicate with other objects through messages.





Has Relationship (Object A Contains Object B)



Inherits Relationship (Object A Inherits Object B)



Message Passing (Object A Sends Message to Object B)

Figure 3. Object Notation (Source: Booch 1993)

Although the OOP methodology is introduced in this section, we have used some of these ideas in the previous sections. For example, figure 1 illustrates that the **PLANiTS** has nine independent objects, and each of them communicates through messages. To evaluate the PV, the PMM may send a message to the model base to run certain models and supply the results back to the PMM. The PMM may then send the results to the database for storage. All objects in **PLANiTS** will function similarly. For instance, when the PMM sends a message  $M_1$ , to the CBR, the intelligent agent will interpret the message and may send another message  $M_2$  to the CBR database requesting a list of some specified cases. When the CBR data base returns the cases, it will perform statistical computations based on those cases and return them to the PMM.

The integrated structure of **PLANiTS** is presented in figure 4. The **PLANiTS** structure is complicated and must be developed as independent objects with an established protocol of message passing between objects. To integrate the various models, **PLANiTS** must contain a detailed data structure (the organization of various fields in the computer), particularly the PMM. The PMM must interpret the PV, and perform various tasks, and send and receive messages (requests, commands, data, and results) to the various objects. In particular, communication with the models is a difficult task, since it is a collection of many models that have various forms of input and output formats. To achieve this in **PLANiTS**, we propose to develop a data structure (at a block level) so that most data can be aggregated to various model specifications with least errors. The data structure and modeling, and data transformations are discussed in the next sections.

## **3.2 Data Structure and Modeling**

### **3.2.1 Planning Vector**

The Planning Vector is the main input module of the PMM. During the initial phase of the analysis, the user will specify the context of application by creating a PV with the following information: (1) Actions, (2) Environments, and (3) Measures of Performance. Since this is the

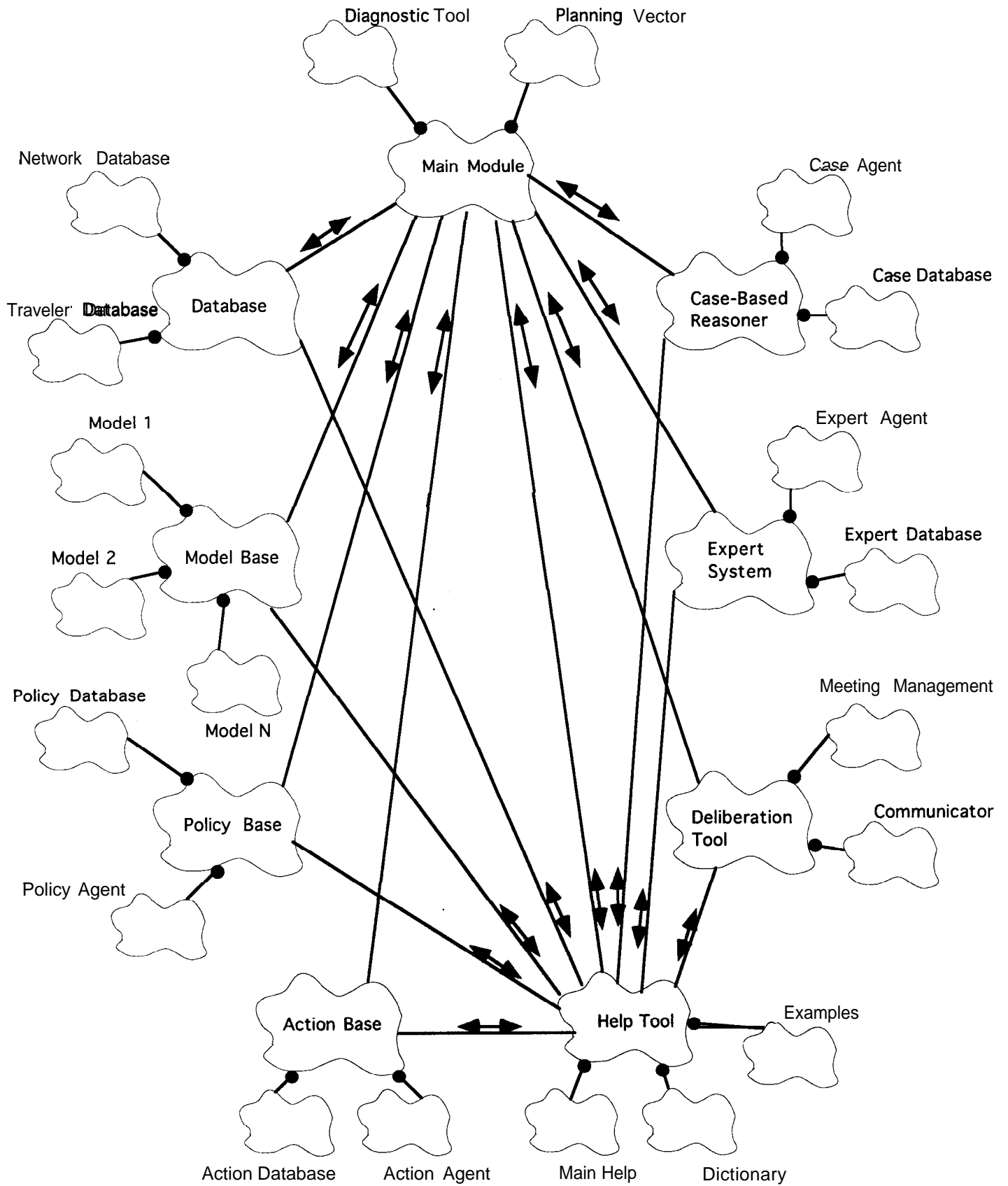


Figure 4. PLANITS Module Structure

only module in which the user will specify all required data, it is essential that this module be defined and structured properly to enable the user to model the problem without ambiguity.

In the simplest form, the PV can be imagined as a table consisting of A, E, and Y and their Space, Time and User group dimensions. For example, the implementation of an HOV lane and the associated reduction in delays can be represented as shown in table 1. During the input process, the user will define the problem by creating the PV and specifying the attributes and dimensions of A, E, and Y. The actual input process will be simplified by graphical and visual displays appropriate for the intricate nature of the transportation data.

### **3.2.2 Transportation data characteristics**

Most geographic (spatial) data can be classified in the following groups:

- **A-spatial:** A-spatial data are attribute-based data, e.g., the name of a street.
- **Spatial:** Spatial data describe the locational properties of objects, e.g., the boundary of a city or the outline of a street.
- **Graphical:** Graphical data are used to represent actual pictures of objects. (Although in GIS systems actual pictures may need to be manipulated, in **PLANiTS** we will be mostly interested in generic features and we will often derive these from other data.)

Transportation data are complex. In addition to the above, they also include time element, user group, and interaction of user groups. Specifying the A, E, and Y in the basic dimensions is

**Table 1: Planning Vector Definition:High Occupancy Vehicle Example**

	Space	Time	User Group
Actions	HOV Location	Time of Operation	Occupancy Threshold
Environment Modeling	Location	Modeling Period	Auto Users
Measures of Performance	Delay Location	Time of Delay	HOV users and non-users

difficult. Often, we will need to create complex forms of data. For example, to specify a particular section of highway network, we need to know not only the spatial extent but also the type of network. Although in reality, the type of network is specified by the location, this is unsuitable for computer representation and analysis because the type of network is not readily identified based on location only. Based on the usage of transportation data, in **PLANiTS**, we will represent the A, E, and Y in the following five dimensions (subject to modification later):

- Spatial Dimensions

- Geographical Area

- Zone (e.g., county, city, census tracts and blocks)
- Network type (e.g., freeways and arterials)

- Transportation Mode (e.g., auto, rail, and bus)

- Temporal Dimensions

- Time

- Time Units (e.g., minutes and hours)
- Time Range (e.g., AM peak period, PM peak period, Daily)
- Time Projection

- User Dimensions

- User Group

- User Unit (e.g., persons and vehicles)
- User Group (e.g., household and employment characteristics)

- Spatial interaction (number of trips between two zones)

- Trip purpose

It is clear from the contents of the PV that **PLANiTS** will need data at various levels of

above dimensions (multi-dimensions) for many of its operations. However, it is impossible to collect or keep all the data in the computer. The main challenge of this part of the project is to determine a feasible data structure that can adequately represent the dimensions of transportation data, and data transformation rules; and that can be used to transform data within these dimensions with minimum loss of accuracy.

### **3.2.3 Data Dictionary**

The purpose of the data dictionary is to collect and organize all transportation related keywords. For the purposes of **PLANiTS**, we have collected more than 500 words that need to be incorporated. However, in our next **PLANiTS** model, we have chosen to limit our modeling effort to highway transportation only. The data items that will be incorporated in our next versions of the **PLANiTS** are listed in appendix 2. This subset of data items was selected to represent the highway transportation system based on practical considerations for the next version. Descriptions, explanations and examples of these terms will also be included in the **PLANiTS** Help Tool.

### **3.2.4 Data Model**

Most of the geographic data information can be grouped into three generic spatial classes: (1) Point (2) Line (3) Area (Region). For the next version of **PLANiTS**, we will extend these structures to include other spatial elements, time elements and user dimensions. The data model also uses the OOP concepts mentioned previously.

The class data model for **PLANiTS** is shown in figure 5. By specifying the attributes in the data dictionary, we will derive the class structure for the PMM. The class model does not represent actual objects but only the conceptual structure. The actual objects are obtained by instantiating classes, i.e., deriving objects **from** these classes. For example, I-880 Freeway in the Bay Area is an instance of the class street that has many links. The GIS type of model discussed above has been used in several transportation applications, but these models are very limited and

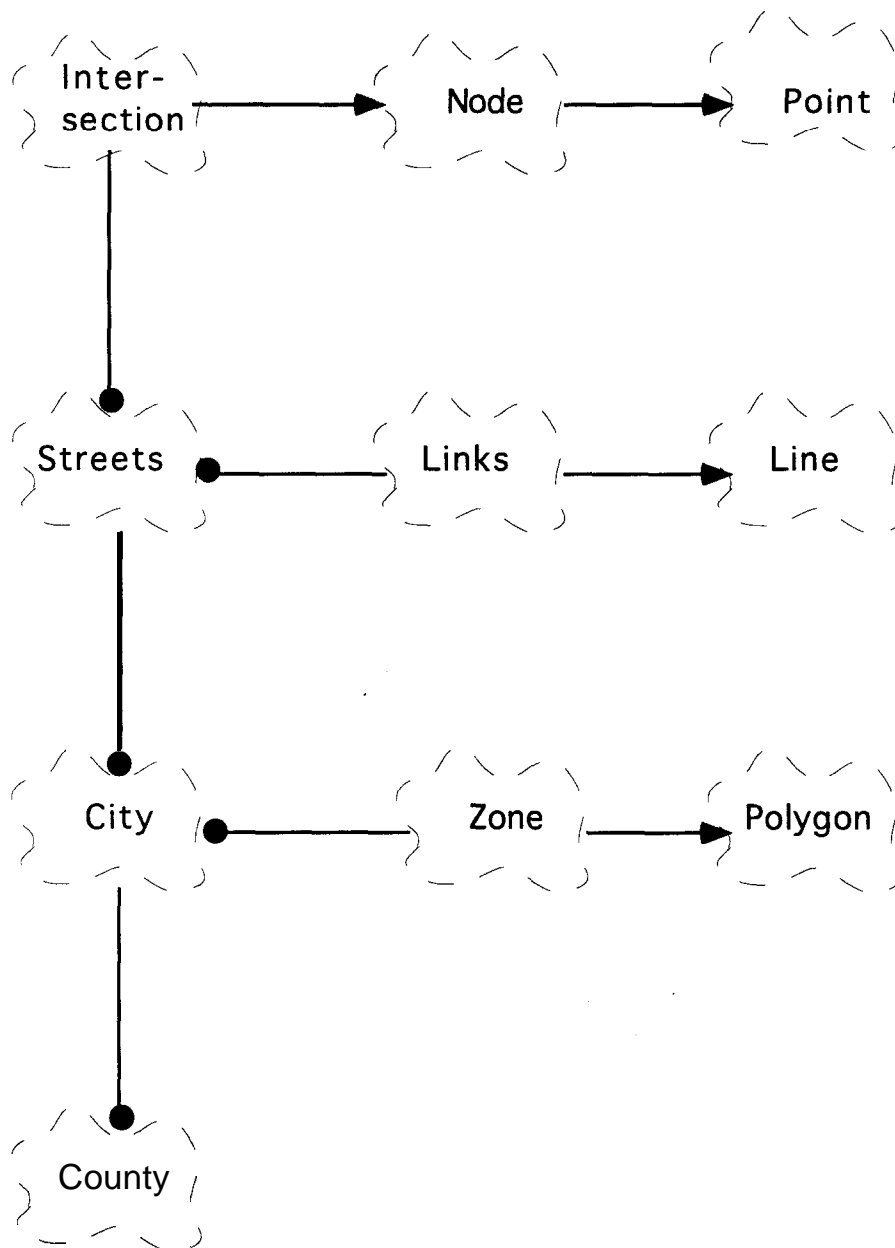


Figure 5. Network Data Model



none of the existing models are sufficient to provide all the information required by PLANiTS. We need a structure that can model transportation data and its complex dimensions.

The functions of the CBR, PB, MB and DT are different compared to the PMM, and require a different type of data organization. For example, the data model for the PB is shown in figure 6. (This is only an example and not the full data model of PB, and the same concepts are applicable to other modules.) The PB has the following main classes: policy, actions, performance measures, state and city. Many of these classes have an m-m association relationship. For example, consider the classes **policy** and **actions**. Each policy may be associated with many (i.e., zero or more) actions; and each action may be associated with many policies. Whenever there is an m-m relationship between classes, we will create another intermediate class describing the association between the objects in these two classes (e.g., the policy-action class). In the model presented in Figure 6, policy-actions, policy-mop, policy-state, and policy-city have m-m association relationship, and extra classes have been used to capture these associations. Note that state-city is an l-m relationship, i.e., one state can have many cities, but each city can belong to only one state. This association can be represented without additional classes. The tables next to each class show the contents of each class.

The purpose of using the structure described above is that now we can use relational set operators to get the results of user queries as illustrated below. The query could represent a message from the PMM or the user.

Example One:

Retrieve the set of policies associated with action At.

Step (1): Retrieve the set of policy ID matching At from the table action-policy.

(2): Using the resulting set from step (1), retrieve the policies from the table policy.

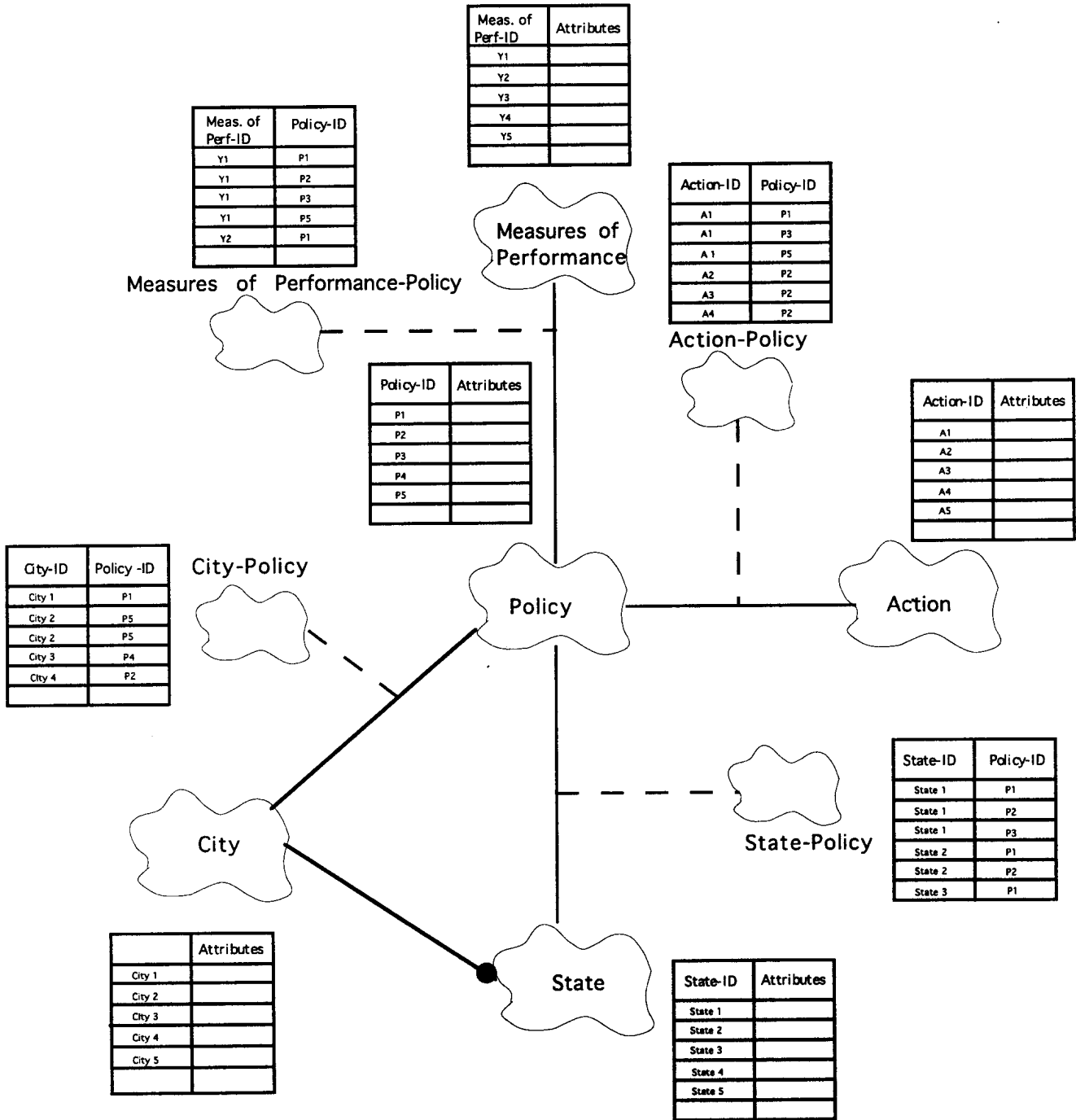


Figure 6. Example of a Data Model for the Policy Base

The answer to this example would be **P<sub>1</sub>, P<sub>3</sub>, P<sub>5</sub>**.

Example Two:

Retrieve the set of policies associated with action At **and Y<sub>1</sub>**.

- Step (1): Retrieve the set of policy ID matching At from the table action-policy.
- (2): Retrieve the set of policy ID matching **Y<sub>1</sub>** from the table measures of performance-policy.
- (3): Determine the intersection of the above sets.
- (4): Using the resulting set from step (3), retrieve the policies from the table policy.

The answer to this example would be **P<sub>1</sub>, P<sub>3</sub>, P<sub>5</sub>**.

Example Three:

Retrieve the set of policies associated with action At **and Y<sub>1</sub> and State<sub>1</sub> or City<sub>1</sub>**.

- Step (1): Retrieve the set of policy ID matching At from the table action-policy.
- (2): Retrieve the set of policy ID matching **Y<sub>1</sub>** from the table measures of performance-policy.
- (3): Determine the intersection of the above sets
- (4): Retrieve the set of policy ID matching **State<sub>1</sub>** from the table State-policy.
- (5): Determine the intersection of the sets resulting from step (3) and step (5)
- (6): Retrieve the set of policy ID matching **City<sub>1</sub>** from the table City-policy.
- (7): Determine the intersection of the sets resulting from step (3) and step (6)
- (8): Determine the union of the sets resulting from in step (5) and step(7)

(9): Using the resulting set from step (8), retrieve the policies from the table policy.

The answer to this example would be  $P_1, P_3$

The purpose of this above example is to illustrate the process. The actual model in the PB will be more complex, and will involve many other classes and class relationships. The results of the queries may be different. For example, instead of returning a set of policies, the result may trigger a function to display warning messages. The queries may involve the results of some model runs in the model base, in which case the PB may return diagnostic messages based on the results. Furthermore, the above example uses the PB, but the same processes are applicable to other modules in **PLANiTS**.

The above examples may seem very simple. However, the large number of possible combinations of scenarios makes **PLANiTS** very labor intensive, as every possible situation must be defined, planned and programmed. A well-defined data model is important for both timely development and subsequent maintenance of **PLANiTS**.

### ***3.2.5 Data Transformation***

To evaluate the PV for all A, E and Y, the model inputs and outputs need to be transformed across many dimensions. Suppose the user intends to determine the total travel time saved per day by adding an HOV lane on a certain section of the freeway. The **2+** person HOV lane will be operated during the three-hour PM peak period on week days. However, the transportation model available in **PLANiTS** determines only hourly impacts. In this example, the data and results need to be transformed across three sub-dimensions of the time, namely PM peak hour, PM peak period and daily. It is challenging to develop a methodology that allows such transformations. Similarly, if **PLANiTS** has data only at the census tract level, but the selected model needed data at the block level, then this may add two sub-dimensions of spatial transformation (a methodology to do this

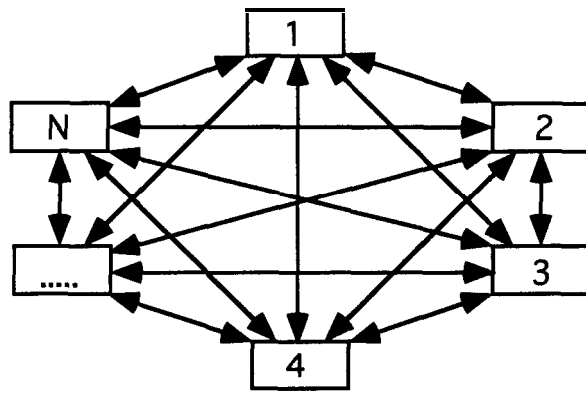
needs to be developed).

More generally, data transformations are needed due to the following reasons:

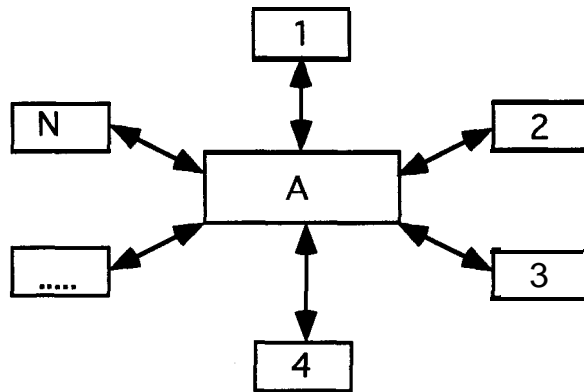
- Changes in dimensionality of data for providing inputs to models and providing statistics for outputs, e.g., calculating averages for performance measures.
- Interpolation of data to infer values, particularly for missing data, and extrapolation of data to forecast trends.

The number of data transformations needed in **PLANiTS** will be enormous. We had listed five main dimensions in the previous section. Each of them will have many sub-dimensions. Each object will have many attributes. Since it is impossible to obtain or maintain all required data at all dimensions, we need to define all possible data transformation (whenever it is legal). One approach is to define every legal transformation for every attribute (as illustrated in figure 7 (a)). If the forward and backward transformations are considered different, for a total of  $N$  different attribute-dimension combinations, we will need a total of  $N^2 - N$  different transformations. This is clearly impossible even for small  $N$  (say 30 to 40). The second approach is to do all transformations through a common base dimension as illustrated in figure 7 (b). For example, we could consider a base dimension (e.g., a block for one hour in person trips) and force all transformations through this base dimension. This will reduce the number of transformations to  $2N$ . However, this will introduce the following difficulties: first, we may not be able define a base dimension that contains a legal transformation for all attributes; and second, this will increase the errors in our analysis. The entire transformation issue is further complicated because often there are several methods of transformation between the same end points.

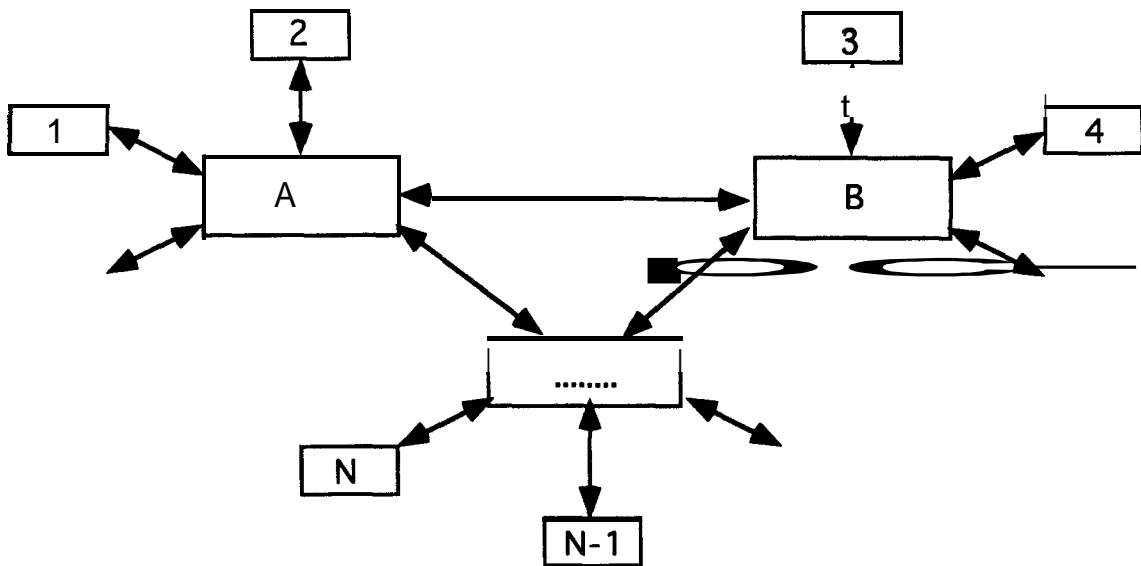
A hybrid approach that is a combination of the above two methods seems promising. Instead of considering each attribute separately, we will classify each attribute into one category, and specify transformation rules for each type of attribute. When there are multiple possibilities,



7(a)



7(b)



7(c)

Figure 7. Methods for Data Transformations

we will also rank these rules. Each data transformation can then be performed using the best rule depending on the type of attribute and the end dimensions. The attribute types can be classified according to the following criteria:

- Unique, e.g., the name of a freeway
- Representative, e.g., average daily traffic volumes on a section of freeway
- Range, e.g., average daily traffic that varies between limits x and y
- Maximum, Minimum, e.g., the maximum capacity of a freeway
- Derived, e.g., average person delay on a freeway
- Distributions, e.g., income distribution across a group of individuals
- Presence/Absence, e.g., whether an intersection is signalized
- Relative Degree, e.g., intersection level of service A to F
- Time Element, e.g., network flows during a certain period
- Narrative Descriptions, e.g., whether a street has high pedestrian activity at a certain location

The above classification of attributes reduces the complexity of the problem considerably. However, the task is still large and complex. The ten types of attributes, each combined with the many dimensions can still leave us with thousands of possible transformations. Although many of them may be very straightforward and many may be illegal (or unspecified), we still have to identify them and define them **unambiguously** (including the illegal ones).

The proposed approach to further simplify the rules of transformation is presented in figure 7(c). This approach combines the methods illustrated in figures 7 (a) and 7 (b). Instead of having one base dimension or having all direct transformations, we will define several base dimensions; for example, block-PM Peak hour-Vehicle-Auto mode and census tract-daily-Persons-Auto mode. Each attribute type-dimension combination can then be linked through the network of

transformation rules. By appropriately choosing the base dimensions (i.e., the number of base dimensions and their definitions), we can control loss of accuracy in data transformation and reduce the complexity of the problem.

Although data modeling and data transformations have been presented implicitly in the context of the PMM, these issues also apply to other modules of **PLANiTS**. Data modeling will be required in all modules and data transformations will be a major part of the Case Based Reasoner, Model Base and Data Base. Furthermore, the conceptual structures discussed here will be used in the intelligent agents and other objects found in the **PLANiTS** main modules. Details of each of these modules appear in other **PLANiTS** reports (Khattak and Kanafani 1995b; Picado, Khattak, Lovell, and Kanafani 1995).

#### **4. SUMMARY**

This report presents the functionalities and integration of the various **PLANiTS** components. The primary purpose of this research is to link the conceptual transportation issues to computer implementation of a new planning methodology. In particular, this report identifies and defines the functionalities of the next **PLANiTS** version. Key issues related to computer implementation such as data dimensions, data dictionary, data modeling, and data transformations are discussed in this report. The ideas presented in this paper will serve as the basis for the next **PLANiTS** computer model.



## REFERENCES

- Booch, G. 1993. **Object-Oriented Analysis and Design with Applications**, Redwood City, California: The Benjamin/Cummings Publishing Company.
- Cayford, R., A. Khattak, and A. Kanafani. 1995. **PLANiTS: Volume 5, A User's guide for the Prototype**. University of California at Berkeley, California: Partners for Advanced Transit and Highways (PATH), Institute of Transportation Studies. Research Report (Draft No. D95-04).
- Kanafani, A., A. Khattak, M. Crotty, and J. Dahlgren. 1993. **A Planning Methodology for Intelligent Urban Transportation Systems**. University of California at Berkeley, California: Partners for Advanced Transit and Highways (PATH), Institute of Transportation Studies. Research Report UCB-ITS-PRR-93-14.
- Kanafani, A., A. Khattak, M. and J. Dahlgren. 1994. A planning methodology for intelligent urban transportation systems. **Transportation Research, C** Vol. 2, No. 4.
- Khattak, A., and A. Kanafani. 1995a. **PLANiTS: A Functional Description**. University of California at Berkeley, California: Partners for Advanced Transit and Highways (PATH), Institute of Transportation Studies. Research Report UCB-ITS-PRR-95-7.
- Khattak, A., and A. Kanafani. 1995b. **PLANiTS: Case-Based Reasoning**. University of California at Berkeley, California: Partners for Advanced Transit and Highways (PATH), Institute of Transportation Studies. Research Report (in-preparation).
- Picado, R., A. Khattak, D. Lovell, and A. Kanafani. 1995. **PLANiTS: Volume 4, The Methods Base**. University of California at Berkeley, California: Partners for Advanced Transit and Highways (PATH), Institute of Transportation Studies. Research Report Draft No. D95-03).
- Vlahos, N., A. Khattak, M. Manheim, and A. Kanafani. 1994. The role of teamwork in a planning methodology for intelligent transportation systems. **Transportation Research, C** Vol. 2, No. 4.
- White, I. 1994. **Using the Booch Method - A Rational Approach**. Redwood City, California: The Benjamin/Cummings Publishing Company, Inc.

## APPENDIX 1

### LIST OF ACTIONS, ENVIRONMENTS, AND MEASURES OF PERFORMANCE

#### Actions

##### Highways

- Construct/remove a highway
- Add/remove mixed use lane(s)
- Add/remove HOV Lane(s)
- Improve interchange
- Construct HOV lanes
- Widen street width
- Add/remove on/off ramp(s)
- Add/remove HOV on ramp(s)
- Install/remove ramp metering
- Restripe lanes
- Improve freeway interchange(s)
- Install signals
- Change signal timings/phasing
- Add/remove left turn lanes, turning pockets
- Add/remove road side parking

##### Transit

- Add/remove bus transit lines
- Add/remove rail transit lines
- Add/remove ferry service
- Change bus service frequency
- Change rail service frequency
- Change ferry service frequency
- Establish transit information systems

##### Bicycle

- Add/remove bus lanes
- Add/remove bicycle storage
- Add/remove bicycle shuttles
- Add/remove bicycle facilities transport in buses
- Provide lockers and showers in work place

##### Pedestrians

- Add/remove pedestrian walkways
- Add/remove crossing lanes
- Add/remove pedestrian crossing signals

##### ATIS

- Variable message signs
- In-vehicle information systems
- At-home information systems

## TSM

- Reversible lanes
- Variable speed control measures
- Vehicle inspection programs to reduce breakdowns and accidents
- Freeway surveillance patrols
- Law enforcement and fines

## TDM

- Rideshare matching programs
- Park and ride programs
- Land-use restrictions
- Transportation impact fees

## Parking

- Add/remove off-street parking
- Add/remove on-street parking
- Change parking fees
- Change parking fines

## Economic Disincentives

- Congestion pricing
- Gas tax
- Vehicle registration fee
- Tolls for vehicles

## Air Quality

- Change emission standard for existing cars
- Change emissions standards for new cars
- Establish buy-out and scrap program of polluting cars
- Roadside monitoring for emissions
- HOV lane usage by low impact vehicles

## Mobility

- Paratransit
- Subsidized taxi service
- Jitney service
- Demand responsive transit

## Safety

- Improve safety of trucks
- Improve safety of cars
- Improve safety of transit facilities
- Improve pedestrian safety

## Aesthetics

- Sound walls
- Landscaping of highway facilities
- Transit station design and art
- Scenic highways, bikeways, and walkways

- Billboard ordinances

## **Environments**

### Geographical Area

- Country
- State(s)
- County(s)
- City(s)
- Census Tract(s)
- Block(s)
- User Defined Transportation Analysis Zone(s) (**TAZ**)

### Network

- Mode (auto, bus, rail, bicycle)
- Link Types(s)

### User Group

- Demographic distribution (income, age, sex, age, occupation,...)
- Household characteristics (size, auto ownership, ..)
- Disability
- Industry
- Employment

### Trip Characteristics

- Mode
- Purpose
- Origin(s) / destination(s)

### Time

- Year of analysis
- Time unit (annual, weekly, daily, hourly, AM peak hour/period, PM peak period, user defined time interval)
- Time Range

### User Unit

- Person
- Vehicle (auto, truck, bus, rail, bicycle)

## **Measures of Performance**

### Air Quality

- Emissions (HC, **SOx**, **NOx**, **COx**, particulate)
- Sources (mobile, stationary)
- Ambient

### Congestion

- Travel time
- Speed
- Delay
- Queue length

- Queue spillback
- Level of service
- Average speed
- Waiting time

#### Productivity

- Highway (vehicle miles traveled, person miles traveled, vehicle occupancy)
- Bus (ridership, waiting time, intermodal transfers, loading factor)
- Rail (ridership, waiting time, intermodal transfers, loading factor)
- Reliability
- Accessibility
- Mobility
- Convenience
- comfort

#### Energy Efficiency

- Fuel consumption
- Productivity
- Emissions

#### Environmental

- Noise
- Aesthetics
- Neighborhood traffic

#### Safety

- Incident
- Accidents
- Personal damage
- Property damage

#### Economic and Financial

- Cost (capital, operating, maintenance)
- revenue (farebox, toll)
- Effects on economy
- Effects on future growth
- Implementation feasibility

## APPENDIX 2

### DATA ITEMS FOR PLANiTS

#### Intersections

Attributes:

Location - X, Y Coordinates

**Zone ID**

Type of Control: e.g. Signal, Stop Sign (2-way, All-way), Yield Sign, None.

Matrix of Possible Turning Movements

1: Volumes (each 15 minute time slices for 24 hrs)

2: Type control for each movement

3: Vehicle composition

4: Capacity

5: Number of Lanes

6: Turn Penalty (a) Time (b) Cost (toll)

#### Streets

Attributes:

A node, B node (i.e. end intersections)

**Length**

Free-flow Speed

Type of Link (Freeway, Major Arterial, Minor Arterial, Collectors, Local, On Ramp, Off Ramp, Merge Lanes, Centroid Connectors)

Speed/Flow Curve

With or Without Shoulders

Travel times

Number of Lanes

For Each Lane:

Width

Capacity

Type (e.g. Mixed flow, Priority Lanes)

Flow on Each Lane: at 15 minute intervals for 24 hrs

Occupancy

#### Zones

Attributes:

Centroid Coordinates

Boundary ID (i.e. Polygon ID)

Area

Residential Population

Land Use --> Distribution by type of use

(CBD, Urban, Rural, Commercial, Industrial,  
Residential)

Income Distribution (25 groups-# of people in each group)  
Age Distribution (10 groups-# of people in each group)  
Occupation (Professional, Students, Agricultural, Trade,  
Manufacturing, Construction etc.)  
Households (Single Family, Multi Family, Apartments etc)  
Car Ownership ( 0, 1, 2, 3+)  
Retail Employment  
Total Employment

**Inter/Intra Zonal Activities (Represented by Trip Tables)**

Attributes:

Home-Based Shopping Auto Trips  
Home-Based Work Auto Trips  
Home-Based Other Auto Trips  
Non-Home Based Auto Trips  
Home Based Work Transit Trips