Design of a Map-Based Transit Itinerary Planner

Christopher Cherry, University of California-Berkeley
Mark Hickman, University of Arizona
Anirudh Garg, CoreWeb, Inc.

Abstract

Geographic Information Systems (GIS) have provided a platform to present information over the Internet to potential users of public transportation. The advantage of using a GIS is that it allows the user to select an origin and destination on a map, easing the task of inputting information to the itinerary-planning process. In addition, the mapping features of GIS can provide a user-specific map showing the route(s) used in the itinerary, as well as local access, egress, and bus stop information. In this article, the design issues associated with the use of GIS in itinerary generation are discussed. Specific design principles are articulated, based on existing knowledge of requirements for the human-computer interface (HCI). In application of these principles, this article describes the implementation of an ArcIMS GIS-based itinerary planner for the Sun Tran bus network in Tucson, Arizona. This system provides users the option of selecting their origin or destination on the map, manually entering an address, or selecting a landmark from a pull-down menu. The routing algorithm then finds the optimum path, and the output is presented to the user both in text and on the map. This is unique from other itinerary planners because it provides an interactive point-and-click map feature that can be implemented using commercially available GIS software.

Introduction

Transit agencies have always struggled to attract riders in a highly competitive transportation market. Potential riders have a large number of options available to them that would encourage use of other modes of transportation. One of the major problems associated with transit ridership is the presentation of information. Abdel-Aty (2001) performed a survey of the effect of advanced transit information and stated, "About 38% of non-transit users indicated that they might consider transit use if appropriate transit information was available to them" (p. 276).

Historically, transit agencies offered transit information including trip planning through call centers. One of the limitations of this system is that graphical information was difficult to relay through a telephone call. Recently, transit agencies have made their service information available on the Internet, using maps, schedules, and on-line automated trip planners. The Internet provides certain benefits when presenting information but cannot replace call centers for more complex trip-planning requests or for people without Internet access. Hence, rather than replacing call centers, the Internet can serve some passengers' routine trip-planning needs, relieving call centers of some of this traditional workload. Radin et al. (2002) provides an excellent report on the current state-of-the-practice of on-line trip planners. There have been several approaches to create on-line trip planners. One of the more advanced applications is the introduction of interactive maps using Geographic Information Systems (GIS) software. This software provides functionality that allows a map to be created and accessed that is personalized to the user's preferences and route choices.

Several researchers have described itinerary-planning systems and have recommended functions and features that should be provided. Trépanier et al. (2002) indicate that the itinerary calculation process includes:

- origin, destination, and circumstance specifications,
- access and egress calculations,
- path calculation, and
- schedule integration.

When developing a trip planner, the developer must consider the user's needs and provide tools that allow the user to plan his/her trip using the same decision-making factors that he/she would use without the trip planner. Several authors have identified some options that mimic this decision-making process. Huang

and Peng (2001) note that important trip-planning options should include the shortest travel time, the minimum amount of transfers, the minimum fare, and/or the least amount of walk time. Donovan (1998) recommends allowing the user to choose which mode he/she prefers and allow the user to choose his/her maximum walking time.

Trip planners can also provide a large amount of information to developers and planners related to the origins, destinations, and timing of trips, and on the use of the trip planner software. Trépanier et al. (2005) show that people using map-based trip planners employ a mixture of methods when determining their origins and destinations, including clicking on the map, entering a landmark or intersection, or any combination of those for both the origin and destination. The authors also demonstrate that users often plan more than one trip during their visit to the transit website. These are important considerations when developing an on-line trip planner.

Text-Based Trip Planners

Many times the input and output of the trip planner is a text-based interface. The user enters his/her address at the origin and destination and time requirements including day of travel in text fields. These time inputs can include the time at the origin or the time at the destination (Huang and Peng 2002). The information is sent to the web server, which sends it to a routing algorithm. The completed itinerary is returned to the web server and back to the browser. This has been the state-of-the-practice of on-line transit itinerary planners since their evolution from call centers (Radin et al. 2002).

Peng and Huang (2000) recognize some problems associated with text-based trip planners. One major problem is that, many times, the user either does not know the exact address of both the origin and the destination or the user does not enter the address correctly. Another problem is that sketch maps often do not provide enough detail or scale to be useful when attempting to plan a trip. Peng and Huang state that a solution to this problem is to allow users to pick their origins and destinations from pull-down menus and provide interactive maps during the itinerary-planning process. This is reinforced by a large literature on human-computer interface design (Brinck et al. 2002; Shneiderman 1998).

Map-Based Trip Planners

To incorporate maps into this process, Smith (2000) identifies the two major functions of a map-based trip planner. The first function is that the planner must make

spatial decisions. This function requires that a planner find all the nearby transit stops within walking distance of a desired origin or destination. The trip planner must also be able to make temporal decisions; that is, the planner must be able to link schedule times to the origin and destination as well as to determine total trip time and maximum time allowed for the trip. Smith (2000) explained that the goal of a map-based trip planner is to minimize total trip time subject to spatial, temporal, and system constraints.

Peng and Huang (2000) indicate that on-line, map-based trip planners have a three-tier architecture. The first tier of the architecture contains a user interface on the web browser, which is the client-server tier. The second tier is a web server, which is the server tier. The third tier is an application server that contains a GIS application server and/or a database server, which is the application tier. As an example of this third tier, Karimi et al. (2004) developed an address geocoding and routing application that finds the route with the minimum number of transfers between an origin and destination and displays them on a GIS-based map.

To date, there are very few map-based trip planners in which the user can interact with the map while choosing origins and destinations (Caliper Corporation 2003; STM 2003), particularly those that use commercial off-the-shelf GIS tools. Lee et al. (1999) discuss the challenges of developing a map-based trip planner using commercial off-the-shelf GIS tools. The authors outline data needs and difficulties associated with identifying accurate locations on a map using GIS software.

The objective of this research is to develop some general design principles for a GIS-based itinerary-planning tool. To illustrate these principles, the research also implemented a prototype GIS map-based itinerary-planning website for a transit agency, using readily available software. This paper documents the design principles and subsequent development of the prototype website. The second section of the article provides more detail about the design principles for a map-based trip planner. The process of data acquisition and development for such a trip planner are discussed in the third section. The fourth section presents the development of a prototype map-based itinerary planner using ArcIMS and transit schedule information. Finally, the fifth section offers conclusions and recommendations to improve the functionality and performance of such itinerary-planning tools.

System Design Principles

The goal of a map-based itinerary-planning system is to provide the transit rider with an interactive, easy-to-use, trip-planning application on the Internet. This system must be GIS map-based to provide interactivity. Its tools must be intuitive and easy to use, yet it must include powerful GIS functions to perform the analysis.

User Interface Properties

The user interface serves two different purposes: (1) it provides a common interface by which information is collected from the user as inputs to the itinerary planner; and (2) it provides the interface by which the specific itinerary is communicated back to the user. Both of these purposes require specific considerations on the user interface.

When developing the trip planner, one important element in the design is to determine what input from the GIS is needed for the routing algorithm and then decide how to provide the data to the routing algorithm. Most commonly, the primary inputs provided by the user are the origin and the destination of the trip as well as the circumstances of the trip, implying the day of week, the time at which the trip begins, the time when the trip should end, or both. More specifically, routing algorithms require that the user enter in unique locations, in the form of land parcels, addresses, or landmarks, as origins and destinations. The GIS analysis must then deliver nearby bus stop identifiers to the routing algorithm. Finally, more sophisticated itinerary-planning tools ask the user for specific information such as how far they are willing to walk or drive to get to and from a transit stop, what preferences they may have for transfers, whether they prefer bus or rail modes, and whether they would like to minimize cost (fares) or travel time.

While there has been substantial research on the design of interactive websites, of relevance here is the use of map information in this input process. Specifically, when designing the display of the system, consideration must be made to determine what cartographic information should be presented that is relevant to trip planning in specifying a trip origin and a trip destination. When deciding how users will determine the origin and destination bus stops, the developer must provide the options that the rider normally uses when planning a trip. Many people desiring to use transit either know the address of their origin or destination, or they know the name of a large landmark where they would like to leave from or go to. However, this does not constitute all of the ways that a transit rider would plan a trip. Many riders plan trips by looking at maps and identifying their approximate

location based on street intersections or other orientation landmarks. From that point, they decide which bus route to take.

In attempting to provide this same level of information, many types of GIS-based cartographic information are available, including specific land parcels or locations (e.g., organized by street address), landmark data, and intermodal (transportation network) data. Landmark and intermodal data contain map features that would aid in the orientation of the users, as landmarks and street networks can be major inputs to the cognitive process of navigation. Landmarks show the user points of interest or large trip generators served by the transit network. Landmarks and the local street network can also aid the user in orienting themselves on the map. In addition, through the intermodal data, the user can also determine what modal options are available within or outside of the transit network.

Overall, all features of the map must have some purpose that will aid the rider when determining the trip plan. GIS interfaces have the advantage of displaying different levels of detail and information depending on the scale or zoom level of the map. The primary challenge in designing the interface for user input is to determine how much information to display on the map. More cartographic information (e.g., additional landmarks, a background aerial photograph, a more detailed street network, etc.) may be very helpful to the user to assist in orientation and to find specific locations that serve as a trip origin and/or a trip destination. Conversely, more information can provide greater visual clutter, possibly confusing the user. In addition, more information in the display means that there can also be substantial latency in generating and manipulating the maps through a GIS-based system. As a result, a trade-off must be made in determining the type of information provided in the interface.

Once the system has generated an itinerary, this information must then be presented to the user. Again from a cartographic perspective, the same map features for user input may also be useful for the map output. The display should also provide a clear indication of the recommended itinerary. Specifically, transit route(s) to be used can be accentuated by highlighting specific bus stop locations and paths of access from origin to bus stop and from bus stop to destination.

Key Functions

A GIS-based interface provides all of this functionality for identifying origin and destination bus stops: an address search function, a landmark search function, and a function that allows the user to click (or select) any point on the map, allowing

the user to select specific landmarks, specific locations, and/or individual land parcels (specific addresses).

Figure 1 displays the core series of events that occurs when a user runs a map-based itinerary planner, specifically over the Internet. With the GIS functionality, the user can have numerous options to select the trip origin and destination. If the user knows the address or landmark, they can enter that directly as a text field. The GIS system can then conduct a query for that location (feature) in the database. Once the feature is found, it can be highlighted on the map and a buffering tool in the GIS can be used to find all of the bus stops within the a given (maximum) walking or driving distance. Those bus stops are then sent to the routing algorithm.

Figure 1. Flowchart of Events Required for the Trip Planner
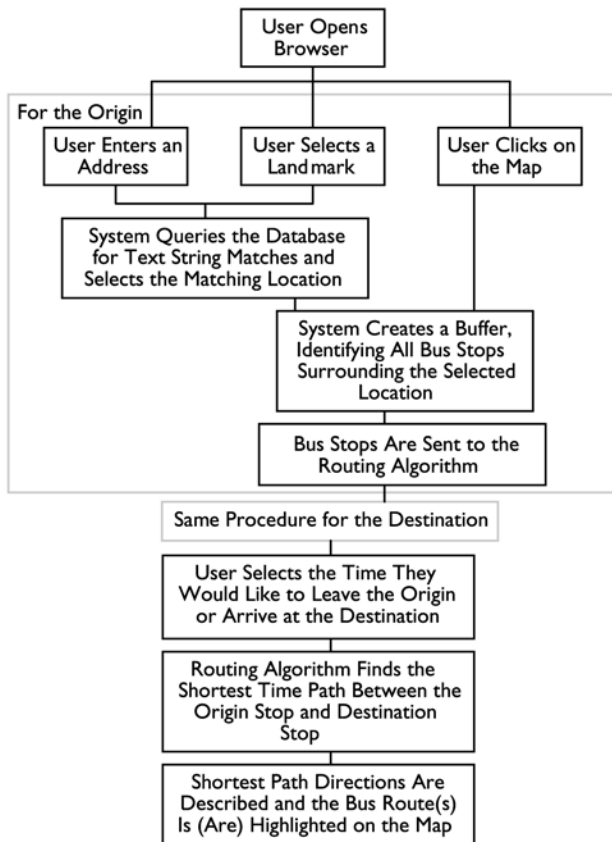
If the user prefers, he/she can utilize the point-and-click capability of the GIS. With this function, the user can simply click a point on the map and the selected location is highlighted (and can even be displayed in text form). With this location, the system then identifies the same buffer around that parcel and the bus stops are sent to the routing algorithm. Essentially, manually inputing an address or landmark performs the same task as clicking a point on the map. All of the query tools access the same database for the location and generate a buffer around that location.

The final critical input required by the routing algorithm is the desired time of departure from the origin, or the desired time of arrival at the destination. This information can be input by using a simple text form or pull-down menu. All of this information is sent to the routing algorithm, and the routing algorithm returns an itinerary to the viewer in the form of text directions and a highlighted route map.

The main purpose of these functions is to provide an interface that can be accessed through a standard web browser, allowing everyone who has Internet access to use the system. It is also designed to provide all of the functionality of the system through a single, easy-to-use interface. As a design principle, all of this functionality can be provided by modifying the tools of existing GIS software to create a custom application. The adaptation of existing GIS tools is important in that many of the existing GIS resources that a transit agency may have can be applied directly to the system without the additional cost of accommodating a propriety system.

Software and Data Requirements

In a GIS-based itinerary system, spatial and temporal decisions are required and likewise, spatial and temporal data are required to make those decisions. The spatial data are usually stored as georeferenced shapefiles, covering the usual sets of shapes (points, lines, polylines, polygons, etc.) with associated database files. The temporal data required are schedule data for the bus stops. With these data, the GIS software, a routing algorithm, and itinerary planner can be developed.

Software

Based on the functions described in the previous section, the primary functions that might be considered internal to the GIS software tools include the ability to:

- generate maps for the World Wide Web (i.e., of reasonable size, electronically);
- pan, zoom, refresh, and otherwise manipulate the map;
- change display features based on the map scale;
- select using point-and-click functionality on the map;
- query for landmarks, addresses, and other points of interest;
- generate buffers for determining passenger access and egress distances to and from bus stops; and
- select and highlight portions of a route (bus route, street network, etc.) for display with the itinerary.

In addition, the ability to generate a shortest path (itinerary) through the network can also be seen as a function that may be included directly in the GIS. Some commercial GIS software have this capability today.

Spatial Data

The spatial data used in the GIS can be categorized into three groups: those required for routing functionality, those complimenting an intermodal transportation network, and those landmarks that might aid in the use of the program. The basic routing functionality within the itinerary-planning problem requires bus route information, bus stop information, as well as a parcel layer that provides complete coverage of the transit network. The bus route information is used within the GIS to display available routes to the user when requesting input, and to display the recommended itinerary after it has been generated. The bus stop information is used to locate bus stops in the vicinity of the origin and destination (a set of parcels), typically using some sort of buffering function. Most GIS systems have a simple Euclidean distance buffer tool. More sophisticated buffering tools calculate distance along the street network, which is more appropriate for walking trips, but also more difficult to implement because the physical representation of access (existence of sidewalks, bicycle lanes, physical barriers, turning movements, etc.) requires a significant amount of additional data and computation.

The intermodal data can be valuable to the user in determining potential access and egress routes, as well as providing a general spatial orientation to the map. Data in this category include the street network (at perhaps various levels of detail, such as major or minor streets, local roads, etc.), public parking lots or park-and-ride lots, transit centers, bike routes, walking paths, public trails, and other

such transportation facilities. Similarly, landmark information can include any number of possible forms to orient users to the map and to provide salient locations that serve as origins and destinations: major employment centers, shopping centers, schools, hospitals, parks, post offices, libraries, bodies of water, or other topographic features. The list of potential landmarks can be extensive. However, one is reminded of the potential trade-off between additional information for the user, and the creation of visual clutter, confusion, and longer computer processing times that might also be caused by the higher information display.

Temporal Data

The other major component of the trip-planning data is the temporal data, in the form of transit schedule data. Schedule information gives the departure time at each time point (and preferably every bus stop) along each route for each run (bus trip from terminal to terminal). Schedule data must be included for all operating schedules, such as weekday, Saturday, Sunday, and holiday schedules. In some cases, some routes or route segments are not operated on some days (e.g., express bus service is not operated on weekends); in these cases, separate data for routes and bus stops must be maintained, conditioned on the day of the week.

In some instances, schedule data for specific bus stops is not available directly from the transit agency. In these cases, it is necessary to interpolate to determine the estimated bus departure times from these intermediate stops. This in and of itself can be done directly in the GIS, if a linear reference model has been created that gives the locations of bus stops and time points along a route. A less precise method simply uses the preceding time point when determining a passenger's departure time from the origin, and the subsequent time point is used when determining a passenger's arrival time at the destination.

Prototype Design and Implementation

Under these design principles, a prototype system was developed with commercial off-the-shelf GIS software and existing spatial and temporal data in Tucson, Arizona. This prototype system serves as a simple test of the functionality for Sun Tran, the fixed-route bus transit operator in Tucson.

Developing a Custom Website

The software used for this project was ArcIMS (Arc Internet Map Server) distributed by the Environmental Systems Research Institute (ESRI 2003). ArcIMS provides the functionality to present GIS information on the Internet or an intranet,

allowing users who do not have GIS software or who are not familiar with GIS operations to use the system. In addition, ArcIMS has all of the features described earlier in this article in terms of built-in functionality for producing a map-enabled itinerary-planning website.

ArcIMS provides a simple interface with potential for customization that will allow both users and developers to easily use the system. To develop a website using ArcIMS, the developer must go through three stages of site development. The steps within ArcIMS are:

1. ArcIMS Author—develop an initial GIS map.
2. ArcIMS Administrator—create a map service using the map created in Author.
3. ArcIMS Designer—create a web browser interface in which to display the map.

Authoring a Map. The first step in developing a website is to create a map using Author. This step allows the most customization of the map display. Once all of the shapefile data are collected, the desired layers are added to the map, and displayed on the map and in a list. The layers are essentially stacked on top of each other; because of this the developer must take care to order the layers to avoid obscuring an important layer.

This project contains 15 shapefiles, provided and maintained by the City of Tucson and Pima County. The shapefiles, which are ordered according to desired visibility, are:

- parcels,
- bus routes,
- bike routes,
- major streets,
- minor streets,
- bus stops,
- park-and-ride lots,
- public trails,
- schools,
- post offices,

- parks,
- malls and colleges,
- libraries,
- lakes, and
- hospitals.

From this list, one may see that the parcels are the top layer; this was done to enhance user selection of origins and destinations. While the bus routes and stops can be considered critical to understanding the itinerary, the parcel layer was placed at the top to provide user point-and-click functionality for origin and destination locations. ArcIMS's default setting requires that any spatial queries be conducted on the top layer in the map. The only data required to operate the algorithm are bus stops, bus routes, and parcel coverage layers. All other layers are included at the discretion of the developer and primarily provide spatial information to orient the user and have no application to the algorithm generating the itinerary.

Major transportation intermodal data form the next several layers, covering bus routes and stops, streets, bike routes, park-and-ride lots, and public trails. Curiously, bike routes were near the top of the layers, as Tucson is considered a very bicycle-friendly community and there are a large percentage of bus users who access the bus using a bicycle.

Finally, the lower layers include major landmarks, such as schools, post offices, parks, malls, colleges, libraries, lakes, and hospitals. In this particular layer, many other potential landmarks were omitted to limit the total information provided in the map interface. While a user may not see their desired landmark in this list, the constraints on computational time had a significant effect on the design of the prototype. Further experimentation on the selection of landmarks would be useful, but was not in the scope of this research.

Once these layers are added, the map is made into a map service through Administrator. For this application, an additional overview map was created displaying only the bus routes layer; this was added to simplify presentation of the bus network. In this way, users can have a very simple spatial representation of the bus service provided in the community. This can be seen in Figure 2.

Figure 2. Basic ArcIMS HTML Viewer
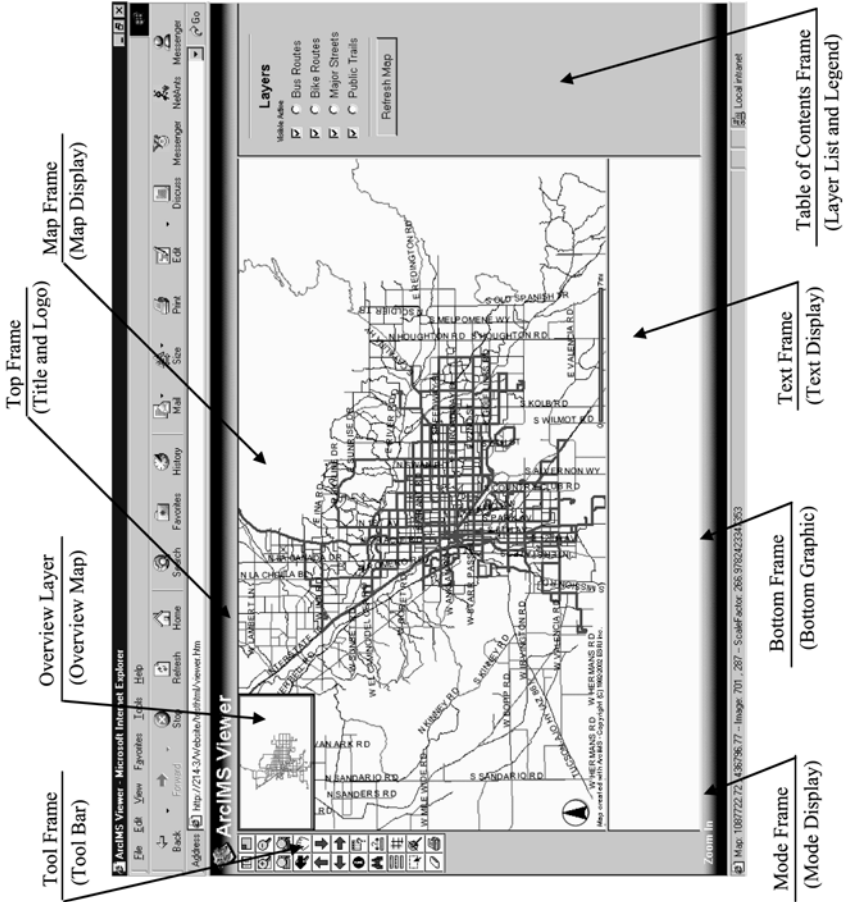
Administering a Service. The next step for the creation of this website is to create map services. These services actually generate the relevant map(s), upon request from a browser for a specific map. Creation of map services is done through the Administrator component of ArcIMS. In the case of this website, two map services must be created: one for the main map and one for the overview map. After these map services are administered, they can be accessed by the web server and displayed on a web browser.

Designing a Website. Developing a browser interface that displays the map services is done through the Designer component of ArcIMS. See Figure 2 for a default display of the main map and the overview map. One of the goals of this project is to make a powerful yet simple interface in which a user, who is assumed to have no GIS experience and does not understand the use of GIS tools, can easily navigate through the system without confusion. Because of this, we have to customize the interface to eliminate or combine some of the advanced GIS tools that are inherent in ArcIMS so that the user only needs to perform the least number of steps.

Customization of the Interface

The GIS interface provided by ArcIMS is very powerful but is also too complicated for the average Internet user. This project therefore requires that the interface be simplified in such a way as to provide all of the functionality required for the trip-planning application, yet not so complicated as to discourage use of the system. Simplification must occur in the interface of the browser as well as in the functionality of several of the tools provided by the ArcIMS software.

Interface Customization. The interface of the software provides some features that could be confusing to a casual user and might discourage use of the system. As shown in Figure 2, the original viewer contains three columns: toolbar frame, map frame, and table of contents frame. The viewer also contains three rows: title frame, map frame, and bottom frame. The text frame is contained within the map frame and is a fixed height. The map fills in the space above the text frame.

To make the website more user friendly, several of the frames were moved to aid in the simplicity of the interface. The intent was to provide basic functionality with the map (visualization, pan and zoom, point-and-click, etc.), and input of other text, but without providing visual clutter or otherwise overstimulating the user. Several of the frames had little meaning to a user with no GIS experience, so these were moved or modified to maximize the size of the map. A color-coded legend was included to ease the use of the map. Tools that are not relevant to the func-

tions of this website were also taken out of the toolbar. Toolbar icons were also changed to text to be more intuitive to a casual user. Finally, a simple HTML frame was included that contained all of the text input fields such as origin address, destination address, and travel time for users of the system who choose not to use the map-based system.

Tool Customization. The main function required from the GIS software is to determine the location of the bus stops surrounding the user's origin or destination on the map. In this case, since the user is looking for addresses, landmarks, and points on the map, all of the analysis tools are used on the parcel layer. To provide the point-and-click functionality that is desired, the Buffer tool was used to find bus stops. Because of its complexity, this tool was modified in such a way as to automatically execute the buffer when the Select by Rectangle tool was used.

Since the only feature that the routing algorithm needs is the bus stops, the bus stops layer is chosen as the highlighted feature for buffering. The user is given the option of choosing an acceptable walking distance, and the bus stops within that Euclidean distance are selected. The display shows the highlighted parcel surrounded by a circle with a radius of whatever walking distance the user has specified. Within this circle, all of the bus stops are highlighted. Other metrics, such as a right-angle metric for a grid street network, could also be used to generate the buffer. Such an approach would approximate the actual walking distance more closely than the Euclidian distance, although it still may not adequately capture the walking times in the actual street network. As mentioned previously, a full representation of the street network in this step generates additional data and computational requirements to the itinerary planner.

A similar technique is used for the user's option of entering an address or selecting a landmark. Instead of using the Select by Rectangle tool, the user can take advantage of the Query tool. With the modified interface, the format of the query is automatically generated so that the user does not have to develop a query on their own. In the case of entering an address or landmark, the user enters his/her exact address in the field or selects a landmark from a pull-down list. The system then queries the parcel database for a match. At this point, the Buffer tool is immediately run and the surrounding bus stops are selected. Again, those bus stops are used in the routing algorithm as potential origins and destinations.

Routing Interface

Transit routing algorithms are very different than standard street network algorithms. These algorithms are more complex because they generally operate on fixed routes rather than standard street networks. The intricacies of the schedule's timings, especially in the case of transfers, adds to the complexity. As a result, rather than using existing network tools in the ESRI suite of software, the ArcIMS map-based system was designed to connect with a back-end, separate routing algorithm. This interface sends the selected bus stops to the routing algorithm, and, after the routing is done, presents the results of that algorithm to the user back through ArcIMS. Therefore, the main focus of this application from ArcIMS is to provide the existing routing algorithm with the information that it needs to find the shortest path, and, once the path is found, to present the path to the user.

Routing Algorithm. The itinerary-planning algorithm of Hickman (2002) was used for this application. This is a forward-searching algorithm (from the origin to the destination) can also be implemented from the destination back to the origin. It finds the shortest path between a given origin-destination pair of bus stops, starting at the origin at a certain time and working toward the destination. An itinerary can be generated for every origin-destination bus stop pair, using any one of the buffered bus stops at the origin and any buffered bus stop at the destination. With each bus stop pair, the algorithm traces a minimum possible time path from node to node along the transit network; these nodes are actually the time points in the Sun Tran schedule. If the system cannot find a path to the destination bus stop on the same route as the origin, it then looks for a transfer node. The transfer nodes are also schedule time points serving several routes. Once the algorithm finds a transfer node, it begins performing a similar search along the new route toward the destination. The algorithm repeats this process until it finds a path to the destination bus stop. All possible routes are considered and the one with the shortest travel time is selected. This algorithm avoids the computationally difficult task of finding all possible origin-destination paths by pruning paths with excessive travel times, as the paths are generated (see Hickman 2002).

This particular routing algorithm gives the user the option of choosing whether to determine the optimum path from the existing schedule or from historic bus arrival time data. Historic bus arrival and departure data are taken from an archive of vehicle location data from an automatic vehicle location (AVL) system. These data are used to find the optimum path based on the total trip time as well as

the probability of the passenger arriving at the destination by a particular time. In this way, reliability of service can be considered. For instance, a bus that has high variance in arrival times might produce an itinerary with a low probability of making scheduled transfers and of arriving at the destination by a certain time. Alternately, the user may choose to find a shortest path using only the static schedule. The user then can choose the deterministic schedule-based shortest path or the nondeterministic, historic shortest path.

Data Input. For the routing algorithm, the data input required is the time point that the rider would like to board, the time point that the rider would like to alight, and the time of day the user would like to leave (or arrive) on a given day (weekday, Saturday, Sunday, or holiday). In our case, the schedule data do not correspond to actual bus stops; they only correspond to time points. For the purposes of the prototype, rather than interpolating the schedules for all bus stops, we used time points in the following way. In the selection of origin and destination, the user clicks the approximate locations on the map, inputs an address in the text field, or chooses a landmark from a pull-down list for both the origin and destination. The database is queried and a parcel corresponding to that click location, address, or landmark is identified; then, the bus stops within the buffer distance of the origin or destination parcel are found. Many of the bus stops found are not actually time points for the routing algorithm. In the case of the origin bus stop, the nearest upstream time point is found. In the case of the destination bus stop, the nearest downstream time point is found. The program then runs the routing algorithm. By using these time points, the algorithm delivers a conservative itinerary. However, since the time taken for a bus to traverse between time points is often rather small, this overestimation should not make a large impact on the overall estimated travel time.

Output

The output of the routing algorithm must be presented to the user in a manner that is clear and easy to understand. This is done through text directions and a map display. Once the routing algorithm has determined an optimum path between time points, it sends the results back in the form of text directions to an HTML frame that lists the arrival time at the originating bus stop, and the probabilities of the bus arriving at that time if the user chooses the historic arrival times. Specifically, the user can be given the probability of arriving at the destination by a specific time; they can also see, historically, what the earliest and latest times of arrival at the destination have been for that specific itinerary. The HTML frame

also lists the transfer nodes that are required and the bus stop where the rider will alight. The name of the intersection where the bus stop exists is also sent to the display and presented to the user. This is done for the origin and destination bus stops as well as the transfer points in the itinerary. Because of the additional data requirements to generate local walk directions, our prototype does not include text walk directions from the origin and destination to the corresponding bus stops; however, the user can use the map to navigate to or from their destination or origin. The time that the bus arrives at each of those stops is also displayed. Presenting this information gives the user complete and specific directions.

Map Display. The map display is one of the unique features of this application; it includes a display of a personalized map of the highlighted transit routes that will be taken. This highlighting process is done using the existing functionality provided by ArcIMS. The Query tool automatically highlights all of the features that have been queried; in the prototype, the query is run on the bus routes and they are highlighted on the map. Figure 3 shows the final output screen, displaying the map and the text directions to the user. The user can then use GIS tools to navigate the map.

Another improvement that could be included is a more detailed map output. Currently, the map simply has the routes highlighted that will be used. A more desirable output would be to highlight the route from the actual origin to the actual destination, graphically displaying relevant bus stops and transfer nodes as well as relevant times along the route.

Overall, this application performs the appropriate function of finding those bus stops within a predetermined distance from a parcel and converting those unique stop identification numbers into a format that could be used by the routing algorithm. The routing algorithm then finds a set of optimum routes based on the total travel time, and, if desired, the reliability of the travel time. The route is delivered to the user in text format and the routes are highlighted on the map. This map gives the user the same functionality as the original interactive map; the only difference is that the routes are highlighted, giving the user the option to identify landmarks near the origin, destination, or along the route.

Figure 3. Final Output Including Text Directions and Highlighted Map
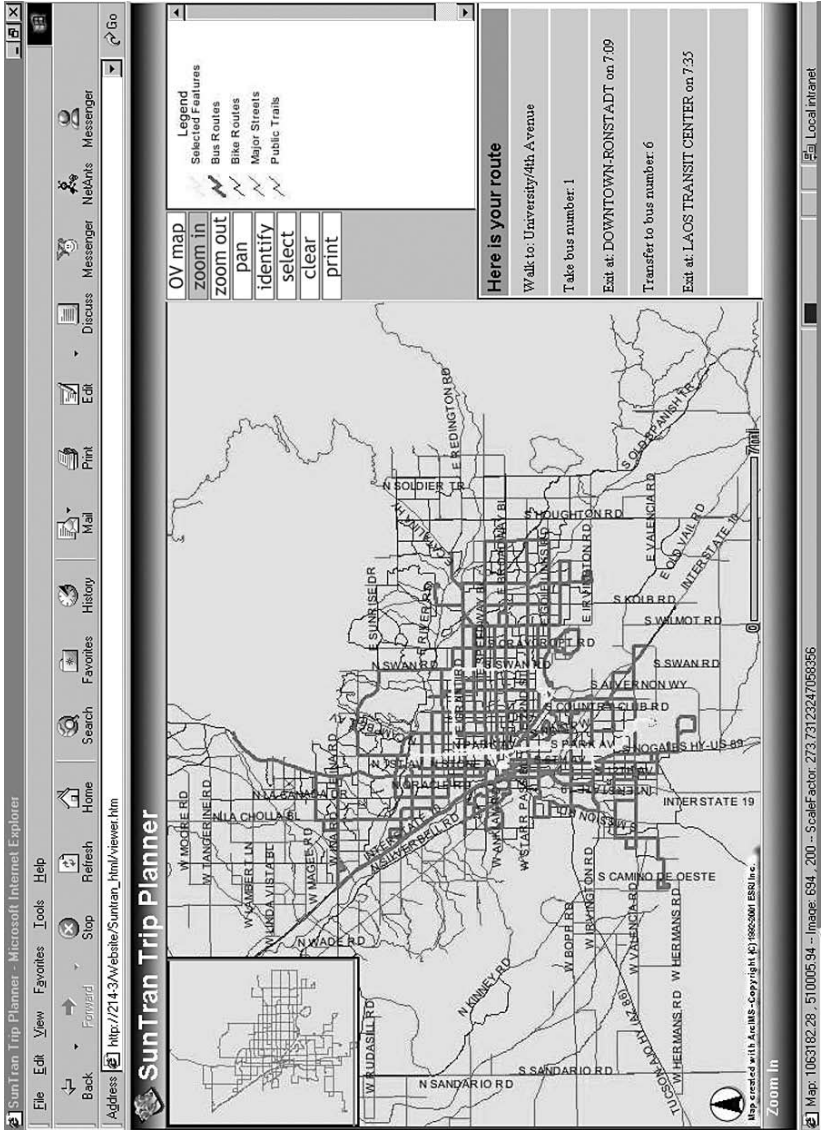
Conclusions and Recommendations

One of the major obstacles associated with transit is the complexity of trip planning. The goal of this research was to explore different design issues associated with map-based itinerary-planning tools. This will become more important as such map features are added to transit websites. In this way, a prototype was demonstrated to use and modify existing GIS tools and software to create an online map-based trip planner for Sun Tran, the transit agency of Tucson, Arizona. This trip planner contains much of the functionality that is currently available in state-of-the-practice websites, but it also contains an interactive map. The interactive map provides the functionality to point and click on a location for the origin and/or the destination, thus eliminating the need for the rider to know exact location information.

System Development

One of the main objectives of this research was to explore the design capabilities and tools that are provided within the readily available GIS software. This would aid future developers of similar projects by using a customized application of existing and widely available GIS software tools. The article has specifically documented system design considerations for a map-based itinerary planner in terms of: (1) interface requirements to allow users to interact with system maps, and associated trade-offs in the level of detail; (2) overall system function and information flows; (3) requisite GIS software functionality; and (4) spatial and temporal data requirements.

ArcIMS 4.0 was used for the prototype. This application was developed beginning with the standard steps through which to create a GIS website in ArcIMS, but the interface had to be customized to make it more intuitive to the user. Some of the tools that allow the user to change settings were removed and many of the tools were automated. For trip planning, a new input frame was added to allow the user to select origin, destination, and desired departure time and send those data to the routing algorithm. The routing algorithm finds the optimum path and sends the output to the browser in the form of text-based directions and a highlighted route map.

Potential Improvements

This project was developed as a prototype of a trip planner that could be implemented by transit agencies. There are several issues related to performance and user features. The main problem with the existing system is that the processing

time to perform one trip calculation is rather slow. Most of the processing occurs on the server, so increasing the processor speed and efficiency of the server would greatly reduce the processing times of each of the steps.

Related to system design, much of the processing speed issues revolve around the very large parcel file that must be visible for most of the functions to work, particularly the point-and-click function. This results in the slow generation of thousands of polylines every time the map display is redrawn. Also, the processing time increases as the zoom level increases for many of the tools. Finally, because of the large number of parcels, much time is spent querying the parcel database for an address or landmark. Directly buffering the bus stops around a point on the map without using the parcel layer would likely improve the processing time.

Some improvements could be made to enhance the user features in any web-based itinerary system. One improvement would be to provide an alternative text-based trip-planning site that is accessible to those with visual impairments who utilize screen-reading software. This could be an alteration of the existing site, prompting the user to use the text-based fields only, or it could be a completely alternate website that executes the routing algorithm with only a text-based interface. As the need for accommodating these patrons continues to grow, provision of these text-only alternate sites may also become more critical.

The system design and prototype described here has some limitations, mostly with processing times and with the detailed features provided in the map display. Nonetheless, it is a successful prototype and provides exceptional map-based functionality that is rarely found with other on-line trip planners. This trip planner provides current state-of-the-practice functionality with its text-based search capability, but it also includes a point-and-click functionality that is very rare with other trip planners.

Acknowledgments

References

Abdel-Aty M. A. 2001. Using ordered probit modeling to study the effect of ATIS on transit ridership. *Transportation Research C: Emerging Technology* 9: 265–277.

Brinck T., D. Gergle, and S. Wood, 2002. *Usability for the web: Designing web sites that work.* San Francisco: Morgan Kaufmann Publishers.

Caliper Corporation. TransCAD Detroit DOT Bus Route Finder Example.2003. http://chardonnay.caliper.com/ddotweb/map_frame.asp, Accessed June 2, 2003.

Donovan, R. 1998. Transit trip planning on the Internet. *Compendium: Graduate Student Papers on Advanced Surface Transportation Systems.* Southwest University Transportation Center: 357–387.

ESRI. 2003. http://www.esri.com, Accessed February 14, 2003.

Hickman M. 2002. Robust passenger itinerary-planning using transit AVL data. *Proceedings of the IEEE 5th International Conference on Intelligent Transportation Systems.* Singapore.

Huang R., and Z. R. Peng, 2001. An integration of network data model and routing algorithms for online transit trip planning. *Proceedings of the 80th Annual Meeting of the Transportation Research Board* (CD-ROM). Washington DC.

Huang R., and Z. R. Peng, 2002. Schedule-based path finding algorithms for online transit trip planning. *Proceedings of the 81st Annual Meeting of the Transportation Research Board* (CD-ROM). Washington DC.

Karimi, H.A., Peachavanish, R., and Peng, J. 2004. Finding optimal bus service routes: Internet-based methodology to serve transit patrons. *Journal of Computing in Civil Engineering* 18 (2): 83–91.

Lee, J.S., J.A. Baumgartner, and Tschango, J.K. 1999. A Web-based bus information system. *ESRI User Conference Proceedings* (494).

Peng, Z. R., and R. Huang, 2000. Design and development of interactive trip planning for web-based transit information systems. *Transportation Research C: Emerging Technology* 8: 409–425.

Radin S., D. Jackson, D. Rosner, and S. Peirce, 2002, Trip planning state of the practice. Volpe National Transportation Systems Center, Federal Transit Administration Report # FTA-TRI-11-02.6, Federal Highway Administration.

Shneiderman B. 1998. *Designing the user interface: Strategies for effective human-computer-interaction* (3rd ed.). Reading, MA: Addison Wesley Longman.

Smith B.L. 2000. Using Geographic Information Systems and the World Wide Web for interactive transit-trip itinerary-planning. *Journal of Public Transportation* 3 (2): 37–50.

STM-Société de Transport de Montréal. 2003. http://www.stm.info/azimuts/, Accessed June 2, 2003.

Trépanier M., R. Chapleau, and B. Allard, 2002. Transit itinerary calculation on the Web: Based on transit user information system. *Journal of Public Transportation* 5 (3): 13–32.

Trépanier M., R. Chapleau, and B. Allard, B. 2005. Can trip planner log files analysis help in transit service planning? *Journal of Public Transportation* 8 (2): 79–103.

About the Authors

CHRISTOPHER R. CHERRY (*cherry@berkeley.edu*) is a Ph.D. student and graduate student researcher in the Department of Civil and Environmental Engineering at the University of California, Berkeley. He holds a master's degree in civil engineering from the University of Arizona. His research interests include public transit operations and planning, transit security, transportation policy, and transportation needs of developing countries.

MARK HICKMAN (*mhickman@engr.arizona.edu*) is an associate professor in transportation engineering in the Department of Civil Engineering and Engineering Mechanics at the University of Arizona. At the university, he has taught courses and performed research in transportation and traffic engineering, traffic flow and capacity analysis, and public transit planning and operations. His areas of research interest and expertise include public transit planning and operations, the

application of new information technologies in transportation, and transportation systems analysis and quantitative modeling.

ANIRUDH GARG (*anirudh@coreweb.com*) is a senior software engineer at Core-Web Inc. in Woburn, Massachusetts. He holds a master's degree in computer science from the University of Arizona. His interests include GIS, web-based mapping software, distributed computing, and web programming.