Etak Innovations

I. Early History

Etak and the Etak Navigator™ were the brainchild of Stan Honey, a bright, entrepreneurial engineer whose passion was transoceanic sailboat racing. His engineering training and familiarity with hardware and software let him apply innovative techniques to the art of ocean navigation and it was this experience that led him to conceive a unique concept for a land navigation device.

In 1983, while working as a research engineer at SRI International, he had the opportunity to crew as Navigator for Nolan Bushnell in the TransPac Race between Los Angeles and Honolulu. Nolan, inventor of Pong, the first successful video game, was flush with money after selling his game company, Atari. Nolan offered Stan seed money to start Etak and develop his navigation concepts.

A Different Concept for Navigation

In the early 80's several efforts around the world were underway to build navigation devices for automobiles. This was before GPS were in full constellation and long before GPS receivers were small, low power and cheap. The aerospace industry had solved the problems of accurate air and space navigation without GPS; why not on land for the automobile? Their general approach was to reduce the cost of inertial navigation sensors. Stan Honey had a different concept. He understood the revolutionary force of the microprocessor and believed that cheap dead reckoning sensors, along with a good map, adequate computational power and smart software would work best. After all, Moore's law was proving that memory and computer speed were unique in their ability to rapidly improve while at the same time come down in price.

Stan's experience in sea navigation had given him powerful insights into land navigation. He understood how ancient sailors had relied on dead reckoning (DR) for their position determination and how they would take advantage of sightings (e.g. stars, lighthouses, land features, etc.) to update their dead reckoned position and reduce its accumulated uncertainty and error. He also knew how catastrophic an erroneous update could be and how careful a good sailor was before position correcting based on such added information.

Unlike the open sea, land navigation has a wealth of additional navigational aids namely, the streets that cars drive on. A navigation system could store a map of streets and computerized "Map Matching" could then be used to update the DR position and reduce its inherent accumulated error. The idea was simply that a car is likely to be driving on a road and not slightly off the road, knocking down fences, pedestrians etc. Map matching was not unique to Etak, several European and Japanese companies were trying to develop land navigation based on the general concept of matching the current estimate of the car's position to some object in a map. Stan and the Etak team brought in significant innovations within the concepts of probabilistic map matching. Stan remembered how ships could get into serious trouble by making a blunder (updating their estimated position based on erroneous outside information and assuming that their accuracy had improved when it actually had gotten worse). From this, Etak developed a form of map matching that was far superior to contemporary methods, including that developed for EVA.

Importance of the Digital Map

Etak's map matching approach required an accurate map with the complete geometry of roads. But in the early 80's such a digital map was not available and indeed its creation, maintenance, storage and use held considerable challenge. Etak embraced these challenges while others, like Bosch, sought to avoid them. For Bosch it is evident that their EVA system was designed around minimal use of a digital map. They believed that their target audience, high-end German car manufactures, would not want to have a detailed map display distract their customers; it would be sufficient to have directions by icon display or simply announced audibly. So, EVA utilized only limited map data for its map matching - only the position of road intersections, road headings at the intersections and distances to the next intersection. Routing was devoid of most geometry and there was no map display. Finally, EVA had no map data at all to support destination finding, forcing the user to wade through an external address book. Even with this minimalist view of maps, Bosch had difficulty as they sought to scale their research project to areas larger than the small city of Hildesheim.

In contrast, Etak took on the tasks of developing new mapping technology to make, maintain, store, retrieve and apply large volumes of map information. Stan Honey understood that there were hard problems to solve and did a nationwide search to find experienced talent to lead such an effort. That resulted in the employment of Marvin White who had spent the past decade on the team that developed the US Census Bureau's national digital map - the DIME file. Starting in the 70's, the Census Bureau decided it needed a digital map of the entire US to help manage its vastly growing spatial database. Based on a strong mathematical (topology) foundation, Marv and his colleagues developed methods for editing and storing the large database on mainframe computers. Not only did Marv White come to Etak with all the knowledge gained at the Census Bureau but he was ready to tackle the task of extending those concepts to work in resource-limited real-time environments, like a navigation system. Specific innovations in map making and the novel formats used for storing and searching these large databases (Database and Storage (DB&S) Technology) were instrumental in many aspects of a land navigation device as will be discussed further below.

Marv rapidly extended the conceptual basis of topologically structured digital maps to include: organization by carriers, a hierarchy of carriers and level of detail/generalization concepts. These, along with other innovations such as the use of effective space filling curves and efficient search algorithms, allowed Etak to solve the problems inherent in using large maps on small microprocessors with slow and limited storage. This resulted in Etak's ability to go to market in 1985 with a state-of-the-art navigation product: the Etak Navigator™. This was so impressive that it resulted in technology licenses with General Motors of the US, Clarion of Japan and Bosch of Europe. The latter aggressively sought to build a product incorporating virtually the entire Etak Navigator™ functionality and technology: the TravelPilot.

Over the following years and under licenses that progressively increased Bosch's access to more and ultimately all of Etak's technology, engineers from both companies worked closely together to increase the TravelPilot's performance and functionality. The following will go into some detail about key innovations made by Etak and taught to Bosch. Indeed, Bosch had started its EVA project well before Etak but as with any new endeavor, false starts and technical blind alleys abound. As will be apparent in the discussion below, Etak's long list of innovations were instrumental in teaching Bosch how to resolve fundamental problems throughout the technical spectrum involved in land vehicle navigation systems.

II. Etak's Technical Innovations

Dead Reckoning (DR)

For centuries DR was the prime means by which sailors navigated. With a compass for heading and a distance (= speed x time) measuring device one can draw a vector from a known starting point to a new estimated current position. That position can be used as the next starting point and the process iterated. It was simple and easy to use but it had one major weakness: it accumulates error with its attendant uncertainty. Each measurement is not exact so the new estimated position has a small error yet it is the start of the next iteration. Over time (and distance), the errors and uncertainty accumulate.

Etak chose the least costly DR sensors. For distance, they mounted wheel sensors on the car's two non-driven wheels. Two sensors have the advantage of being able to measure the amount of turn. This is called a relative heading measurement because it does not know where absolute North (True or Magnetic) is, only the amount of turn. Etak even improved on this detail by developing a compensation for the foreshortening of the track of the front wheels due to Ackerman steering (US Patent 4,788,645).

Unlike EVA, Etak added a compass to its Navigator. Etak realized that the differential input from the two wheel sensors was not a robust estimator of heading. Tire drift, calibration errors, change in pressure and wear, loading, deformation, road crown and slippage around turns etc. all added up to significant errors. Furthermore, without a measure of True or Magnetic North, these errors could cause the DR position to rapidly accumulate error. Because, unlike EVA, Etak's map matching was performed on road segments, not just at intersections but everywhere, and because one of the key matching parameters was a comparison of the heading of the vehicle with the directions of the nearby road segments, it was necessary to have a reliable and accurate estimate of heading. Consequently, Etak developed an electronic, flux gate compass (US Patent 4,646,015) to be incorporated into the Navigator (and later into the Bosch TravelPilot IDS).

Most designers of land navigation efforts of the time frowned on adding a compass because it also could be subject to significant errors. However, Etak realized that the two measures of heading (differential wheel sensors and compass) had characteristically different and independent types of errors and that they could be combined to form a single more robust and accurate heading measurement as described in US Patent 4,734,863. In addition, Etak developed a calibration technique, effective sensor health monitors, a sensor to detect and software to eliminate magnetic effects of the window defogger and even an emergency auto calibrate function. The combination of all these innovations made for a dramatically better DR position. All of these innovations were built into the TravelPilot IDS[1].

Map Matching

The key to Etak's map matching lies in its ability to NOT update to the map at each opportunity. Rather, the algorithm would continue to DR until such time that the map and the DR position, heading and other parameters (e.g. connectivity) matched sufficiently well to be considered, with high probability, an accurate update. This was embodied through the concept of a Contour of Equal Probability (CEP), described in US Patent 4,796,191. The CEP allowed Etak to model the fact that the DR position was not an exact point but rather the center of an area that contained, with a certain probability, the location of the vehicle. As the vehicle continued to move and a next DR position was computed by adding the next sensor-derived measurements to the previous DR position, the uncertainty and error of this next DR position would be larger than the uncertainty and error of the last position. That is, each DR position came with its own sensor errors and statistically, these errors accumulate. The CEP was a way to encompass the area around the DR position that contained all possible places where the vehicle was likely to be, say within a certainty of 95%.

The CEP was computed at each map matching cycle which occurred every other DR cycle, approximately every two seconds. The CEP was used to find roads within the map that intersected with it. Those were the only roads close enough to be considered candidates for map matching. Each such road was then tested using multiple parameters as is described in the above cited patent. If more than one such candidate was found, the best candidate was selected and the vehicle was updated to a point along that street and the CEP was reduced (because sensor errors grew at different rates the CEP was not circular

[1] Inferred or directly found in RGS05 are C-language or Assembler modules and functions from Etak Code Directory and Sub-directories of "NAV\DR…"; BIG_GAUS.C; CAL_CDEV.C; CAL_CRAD.C; CEPLIMIT.ASM; DEAD_RECK.C; DEFOG.ASM; DEV_CAL.C: DEVCORR.C; DRCALC.ASM; HP_FILT.ASM; NEW_CNTR.C; RDTGSENS.C; SENSORS.C; TRACK.ASM; V_TRACK.ASM. Functions dr; cep_grow.

and reductions were made only in the direction of the update, i.e. orthogonal to the direction of travel). But if no such candidate was found (or no "best" candidate could be chosen), then no map update was made and the navigation continued to DR and the CEP was grown to reflect the further accumulation of error.

While Bosch claims to have already used map matching in EVA, the Etak map matching was profoundly different. The fact that Etak tested every two seconds by matching to actual road geometry and the fact that Etak used the concept of the CEP to test the quality of the match and continue DR navigation even when a map matching cycle failed to find a match, made for a much more accurate and robust position determination. The reality is that a map is never a perfect model of where cars drive and Etak's more flexible algorithm enabled cars to drive into parking lots, driveways or on new roads without forcing the algorithm to make a blunder; that is an erroneous map update. Moreover, Etak's ability to update on the road geometry between intersections enabled the algorithm to constantly limit its accumulation of error and naturally deal with such maneuvers as a car taking a U-turn and returning to the node it last visited.

Within Etak's map matching were further innovations such as a connectivity tree, a running update, correlation of car heading changes with curves in the road and multiple dynamic calibration routines. All these were built into the Etak Navigator™, later on used for the Bosch TravelPilot IDS and available for Bosch to use subsequently[2]. When GPS became commercially viable, Etak extended its position determination to incorporate GPS as described in its patents US 5,311,195 and later US 5,948,043.

Moving Map Heading Up Display

Perhaps the most distinctive innovation Etak developed was its intuitive map display. Back then digital maps were new and even those applications that were available on big machines were slow to respond. For most companies developing navigation systems, it was simply not possible to think of dynamically updating a map display in a resource constrained navigation system. By solving map storage and retrieval problems (explained later on), Etak was able to make fundamental innovations in the map presentation.

Early navigation devices used transparent map overlays. The vehicle's position was indicated by the CRT beam which moved and the driver had to change overlays as the beam left the screen. More advanced systems displayed a map and would change it as the vehicle's position got close to the screen's edge. EVA had no map display, only presenting icons to indicate when to make the next maneuver. All these approaches were either cumbersome to use or extremely limited in the information they provided. In contrast, Etak's map display proved to be of great help to a driver even without a destination entered or route computed.

Specifically, Etak made several user interface innovations, explained in its patent US 4,914,605. The display for the Etak Navigator™ and for the early TravelPilot IDS was a monochrome CRT. It offered the map at different scales so the driver could zoom to a scale that best matched his needs at the time. Roads in the map were classified as to level of importance such as residential, arterial, highway and the like. At large scales (smaller area) all the roads were shown in full detail. At progressively smaller scales (larger areas) smaller roads would not be shown and roads of higher road class would be plotted with less detail (generalized) thereby reducing storage. These measures prevented the display from becoming overly complex while still giving the driver the information that was most appropriate for that scale.

[2] Inferred or directly found in RGS05 are C-language or Assembler modules and functions from Etak Code Directory and Sub-directories of "NAV\MM…"; CK_ANGLE.C; CONNECT.C; CORRELATE.C; DISTUPDT.C; DST_CALC.ASM; GEO_MAG.C; HOP_CEP.C; INCEP.C; INCLUSIV.C; MAG_UPDT.C; MCBUF.C; NEWDRPSN.C; NPAM.C; NX_BRNCH.C; PA_BUILD.C; PA_QUAL.C; PA_UPDST.C; QEP_EXP.C; SEG_ANGL.C; SEG_LENG.C; SFINCLSV.C; SFINTRST.C; SWAPNODE.C; TR_ADD.C; TR_ALOC.C; TR_BURN.C; TR_DALOC.C; TR_DEL.C; TR_DEPTH.C; TR_FIND.C; TR_GROW.C; TR_INIT.C; TR_MERGE.C; TR_OLDRT.C; TR_PATH; TR_PLANT.C; TR_POP.C; TR_PRNXT.C; TR_RIGHT.C; TR_SPLIT.C; TR_STORE.C; TR_UPDWN.C; TURN_LEN.C; UPNORM.C; VCTRGEOM.C. Functions: bcorcalc; nav; run_fix.

Etak developed dynamic labels to annotate street names. These labels were the same size, regardless of scale, were always more or less right-side-up regardless of the direction of the map, did not collide with each other, and were not truncated by the edge of the map – changing position as needed. Finally, an algorithm selected only those labels that were most likely to be important to the driver.

Most importantly, Etak provided a map display that was centered on the current estimated position of the car and remained oriented with the current estimated heading of the car in the upward direction – a "heading-up" map display. This was controversial at the time. Some were afraid it would be distracting and disorienting. But in fact the opposite was true. In the environment of the car, you were moving and changing directions and the "moving" map stayed in perfect synchronization with you. It was the intuitive method of display. This was easily demonstrated to skeptics by setting the map on North-Up display and asking them which way to turn towards a given destination as the car headed southward. Invariably, this prompted confusion. The process was repeated using the display in Heading-Up mode and the problem disappeared. Not only did this innovation become part of the TravelPilot, it basically became part of every navigation device since.

Geocoding

Before a navigation system can help you in reaching your destination, you have to tell it where your destination is. That is not easy. Often, you can only represent your destination by some description such as an address. Sometimes you may only know a street intersection or a single street name. And sometimes addresses are ambiguous (e.g. more than one 100 Main Street). Using its map DB&S innovations, Etak was the first to offer a method to directly enter a street address into a navigation unit and have the device find its location. So, unlike EVA with its unwieldy and unscalable external destination entry book, Etak offered a convenient way to directly enter addresses in a variety of formats. The interface is described in US patent 4,914,605. Etak modified the basic Administrative Descriptors and made the appropriate changes so that the same geocoding function would work in the TravelPilot in Germany, and later in other countries as well.

Routing

The initial Etak Navigator™ provided many features that helped the driver get from his current location to his destination. Several of these features are considered part of an overall routing function such as showing the current position of the vehicle as well as the destination and the streets in between. Also, showing the distance and direction to the destination and dynamically updating these as the car moves forward are important to any routing function. Etak did not provide a best path computation nor directions output in the Etak Navigator™. The complete algorithms were not fully developed as of 1985. The processor was too resource constrained and the acquisition and subsequent maintenance of routing-specific attribution such as one-ways were time consuming and costly and not justified for the anticipated small initial market. Etak decided that while best-path computation would eventually become a necessary feature of navigation systems (as would color display, GPS, business listings, real time traffic and a host of other enhancements), it was better to go to market with the long list of innovations already available for its revolutionary new device than to delay and increase device cost. Bosch made the same decision and selected Etak's innovative technology as its starting platform into the navigation market.

Etak continued to work on its path finding and directions after the launch of the Etak Navigator™ and the TravelPilot. In fact, Etak was under a contractual obligation to deliver to Bosch a working demonstration of its routing by 1988 and did so. Etak first adapted a Sun Sparc so it could operate in the trunk of a car which was used to demonstrate real-time routing. This was followed by porting its routing technology to work on a TravelPilot with extended memory (1MB instead of 500kB). Etak also demonstrated routing within its latest navigation system called the Etak NAV IV.

As Etak was working on these developments, Bosch was struggling with its own routing problems. For the best path algorithm, Bosch was using Bellman, Ford and Moore while Etak was using Moore/ Dijkstra. These optimum shortest path search algorithms were public domain and needed to be adapted to the map application with its enormous number of nodes and links. The real challenge was in designing an algorithm on a solid foundation of data structures and retrieval methodology that would allow computing a path in real time on a memory constrained system, since the straight forward use of these classic algorithms was not feasible. Etak chose to reduce data complexity by employing a purely data-driven multiple-level hierarchy embodied in its carrier tree. Bosch had used a two-level search strategy where its coarse map level was constructed based on criteria other than natural map hierarchy. With this approach, they had difficulty with size imbalance in the sectioning of the map which therefore had to be done, at least partially, manually. Furthermore, an adjustment to one section might well have required an adjustment to another section so the process was iterative. While this was adequate for EVA as a research project applied to the small city of Hildesheim, it simply was implausible to scale this to a production system covering all of Germany, let alone future coverage of Europe. On the other hand, as will be discussed below, Etak had developed a fully automated data formatting compiler that could segment the map at multiple levels with different amounts of detail, along with methods to search down to find the most details, to search up to continue the path on only higher class roads, and at any level to search across to branch the search out at a consistent class of roads.

Etak and Bosch agreed to meet, share their insights and to mutually find a best way to proceed. After an intense several-day meeting that involved the key path and map developers from both companies, including Mr. Neukirchner from Bosch and Mr. White and Mrs. Kuznetsov from Etak, agreement on a common method was not obtained. Bosch wished to continue using their algorithm. However, they had clearly learned a considerable amount regarding how to format the map data for the efficient retrieval needed to find routes extending across Germany or even across a sizable city within Germany. The "existence proof" for the feasibility of this method was also demonstrated. In analyzing the RGS05 data files, Tele Atlas has found evidence that Bosch subsequently used Etak's topological and hierarchical concepts and extended their two-level map to a multiple level map hierarchy (with varying levels of detail) as described in Etak's US Patent 5,694,534. These changes were not just cosmetic but reflect a transformation in their thinking about the problem which led them out of their technical blind alley towards a feasible solution.

Real Time Multi-tasking Kernel and Math Utilities

As can be seen by the above descriptions, even early navigation systems involved complex real-time processes, many of them event driven. For example, the display required refresh at a minimum rate of 30 times per second; less would result in flicker. DR involved sensors that needed to be measured at rates as high as five times per second. Map matching had to be performed about every other second. Also, the system had to be responsive to button presses and computations such as geocoding, changing display etc. All these individual tasks needed to happen simultaneously and asynchronously. Today, such system requirements are generally handled by a sophisticated operating system but none existed in the mid to late 80's that was designed to work within the limited environment of an embedded processor.

Etak struggled with this problem when developing the Etak Navigator™ and used an interrupt driven time-slicing algorithm which allocated a certain amount of time to each process or interrupt. This was an inefficient use of the processor and was cumbersome to maintain as it needed adjustment when the software team changed or added tasks. To eliminate these difficulties, Etak developed its own operating system – a real time multi-tasking kernel. This software had the intelligence to recognize what task needed resources and to swap tasks in and out of execution as needed. It handled all the intricacies of the computer stack, memory and interrupt system. The software team only needed to define a task in terms of resources, interrupts and priority and the kernel would handle the details. This innovation was an effective tool in allowing new tasks, like routing, traffic or business listings to be added with little concern about how they interfaced into the complexity of the total real-time processing task. Etak installed its kernel in Bosch's TravelPilot IDS and it became the backbone of the TravelPilot product line for years to come[3].

Additionally, Etak developed specialized mathematical functions, needed in positioning routines and elsewhere and converted those that were particularly time consuming, into Assembler modules. Many of them have already been traced to Bosch's RGS05 TravelPilot[4].

Mapping Technology

Basically, there are two types of digital maps: an **image map** – generally, a scanned paper map, and a **vector map** - a database that contains various spatially related information including geometric positions which can be drawn as vectors on a screen. While image maps were quick to obtain (by making a digital copy of an existing paper document), they require significant memory space and have no object or attribute intelligence. In contrast, a vector map was a database that could be easily annotated with additional information and could be used in many computer applications. The vectors can be appropriately labeled at any scale and orientation. Information associated with the map objects such as geometry, street names, addresses, road classifications, turn restrictions, speed limits, traffic delays etc. can be used in conjunction with software applications for map matching, geocoding, map display and routing tasks like shortest path search.

But making and maintaining a vector map is cumbersome. Typically, in the mid-80's individual positions were measured from a paper map on a "digitizing table." These positions were put into a database and formed the vector line work that depicted the map geometry that could be displayed on a graphical computer workstation. But the human map maker (digitizer) had to constantly look between the source map on the table and the digital vector map shown on the computer screen, mentally comparing them. This was slow, tedious and prone to error.

Etak made two major innovations in production mapmaking. First, using a scanning microdensitometer, it eliminated the digitizing table by scanning the source data into the computer and using it as a background image over which the line vectors of the digital map could be displayed. This put the new digital lines directly in the same space as the source data which enabled the digitizer to rapidly and accurately create or reposition geographic data. This worked as well for paper maps as for aerial or satellite images. Second, Etak introduced on-line topology edits to the digitizing process. The mathematics of topology offers a sound basis for testing the consistency of a map. Etak developed a mapping process that applied the rules of topology in real time, after each map edit was made. This had the effect of catching many errors immediately and allowing the human to make a correction when the information about this edit was still fresh in his/her memory.

[3] Inferred or directly found in RGS05 are C-language or Assembler modules and functions from Etak Code Directory and Sub-directories of "MT…" or "SYS…"; EXEC_PER.C FEVENTS.ASM; FRSRC.ASM; FTASKING.ASM; INTRCD.ASM; INTRCOMM.ASM; K_ALLOCA.ASM; K_CHKSTK.ASM; KSHUTDWN.ASM; KSWINTR.ASM; KUTILS.ASM; FTIME.ASM; KINTRPTS.ASM; KSHRDATA.ASM; KSTARTUP.ASM; KEVENTS.ASM: KRSRC.ASM; KTASKING.ASM; KTIME.ASM. Functions: comm_ports_init; file_sys_init; initdev2; iomonitor; sys_start; sys_close; sys_init; sys_onexit.

[4] Inferred or directly found in RGS05 are C-language or Assembler modules from Etak Code Directory "MATHUTIL": ADD_6BYT.ASM; CMPS_FAR.ASM; DIV32_16.ASM; DIV48_32.ASM; DIV_SI_F.ASM; DIVLGINT.ASM; DIVS48S1.ASM; DIVU32U1.ASM; DRPUPDT.ASM; IATAN2.ASM; ISMUL.ASM; ISQRT.ASM; LAT_LEN.ASM; LAT_POL.ASM; LDIFPRDS.ASM; LSAR8.ASM; LSMUL.ASM; LSUMUL.ASM; MUL16_32.ASM; MUL32.ASM; MUL_SI_F.ASM; MUL_UL_F.ASM; PRD48_32.ASM; RATIO.ASM; SFADDSUB.ASM; SFABS.ASM; SFMUL.ASM; SFCMP.ASM; SFDIV.ASM; SFLTOSF.ASM; SFRSFTS.ASM; SFSFTOL.ASM; SFSFTOS.ASM; SFSQRT.C; SFSTOSF.ASM; SHIFT16.ASM; SINCOS.ASM; SMUL3216.ASM; UDIV.ASM

Etak considered its heads-up digitizing and on-line topology testing to be trade secrets and did not file patents. These techniques were built into a multi-seat Geographical Information System (GIS) for map production that enabled digital vector maps with all their related attribution to be developed at speeds estimated to be 3 times faster than contemporary commercial mapping operations. This system sustained Etak's mapping operations with approximately 50 seats operating in 3 shifts around the clock and was also ported to an even larger production facility in India. While Etak, under contract from Bosch, made a first map of Germany, Bosch licensed this mapping technology and replicated it so that they could maintain their own map of Germany and elsewhere yet remain completely consistent with Etak's maps.

Map Database and Storage Technology – DB&S

Being able to build and maintain an accurate and richly annotated digital map of continental proportions solved only part of the map challenge. Equally important was storing the map for use in small devices like navigation systems and organizing the map database in such a way that it could be speedily searched to pull in the necessary information to answer such diverse questions as: Where is this address? What streets intersect with the CEP? What pertinent streets are around me at a 2 mile scale? What businesses are close by? What is the address range of the street segment I am pointing to? What street segment corresponds to this received traffic link? And of course, what is the shortest path from my current location to the destination address I entered?

In the early days of digital maps, most applications stored their map on a mainframe or at least a large minicomputer such as a VAX. These were powerful computers with fast mass storage devices and large amounts of internal memory – at least in comparison to device microprocessors of that era. Even so, these types of questions generally necessitated the user to take a coffee break before an answer could be provided – clearly not reasonable performance for a real time navigation system. It was in this area that Etak developed one of its broadest and most important innovations in support of navigation systems. Marv White and his co-inventor, George Loughmiller, extended the map related topological concepts into a hierarchical data storage structure that was extremely compact and efficient for doing the basic spatial searches required to answer the questions listed above and more.

Up to this point, topology had been used to define the basic objects of a planar map (all map objects can be defined in two dimensions – even roads passing over other roads). Briefly, this topological data model consisted of zero cells (0-cells) representing points like the intersection of two roads: one cells (1-cells) representing roads or linear water or political boundaries etc, and two cells (2-cells) representing areas. (This model can be extended to three cells for volume but that is unnecessary for a planar map.) Topology provided strict relationships among these objects, for example a 1-cell was always bounded by two 0-cells (one at each end) and a 0-cell was always co-bounded by at least one 1-cell (and more if the road ending was an intersection). 2-cells were bounded by a series of 1-cells and every 1-cell was co-bounded by exactly two 2-cells (which could be the same). This short list of topological properties could be used to test and enforce a consistent map of arbitrary size and complexity. Attributes, such as street names and addresses, turn restrictions, transit times, administrative area and latitude and longitude could be attached to the most appropriate object class (or in a few rare cases such as a complex turn restriction, attachment was made to an adjacent sequence of objects). For roads and other linear objects that were not straight between their end points (0-cells), shape points were added. These points had no topological significance but were there simply to define the feature's geometry to within a position accuracy specification.

This basic map topology had already been worked out and Marv White helped apply it to creating the US Census Bureau's Dual Incident Map Encoded (DIME) File used in the 1980 census. But the Census Bureau had big mainframe computers and massive storage and their requirements were not real time.

What Marv White figured out when he came to Etak was a way to use the topological map as its own index file and successively group less detailed data in storage units that successively covered larger areas. Marv called each storage unit a carrier (as in data carrier) and successfully built a logical tree of carriers, where the tree was a hierarchical way to store and refer to data at different levels of detail. Part of the genius of the innovation was that Marv used the basic objects of the map data to act as the carrier boundaries, thereby eliminating the large amount of metadata and extra complexities associated with imposing some arbitrary hierarchical data structure.

Marv used the road elements as the carrier boundaries. This was particularly helpful in building a carrier-based hierarchical tree since roads of different road classes could be used to form the boundaries of carriers at different levels of the tree. Roads were also a useful construction mechanism because they naturally created the multiple levels or layers of the tree that were needed to provide the necessary levels of detail for the various search algorithms. As carriers were built at higher levels up the tree, it was also natural to reduce the level of detail contained within those carriers. The lowest level had all detail of all map objects and attributes stored for a small patch of area (whose size naturally changed depending on the road density). Generally, roads at the next-to-lowest road class were then used as carrier boundaries. At the next level in the tree, the carriers may not have stored roads of the smallest road class and might reduce the amount of geometric accuracy for the roads that it did store (a process called generalization) and again the roads of the next highest road class were used as the boundary for these carriers. Resulting from this topologically consistent process, the area of any carrier exactly contained the area of a discrete number of carriers in the tree below it and its area was fully contained by one and only one carrier above it. This full and complete containment was essential for efficient storage of the data. By using pointers below, above and sideways any manner of search could be quickly performed.

For example in a best path algorithm one searches the roads around the destination for paths leading back to the current location. Since it is too time consuming to search at this low level across a large and detailed area, the algorithms generally limit their search when they encounter roads at a higher road class (allowing for faster transit times). Now the search proceeds at this higher level of road (stored at a higher level in the carrier tree hierarchy) until it again encounters roads of a higher road class and so forth until it must reverse the process to look for roads of progressively lower road class to reach the current position. (Variants on this search from the current position or search from both ends simultaneously.) This topological hierarchy and the three basic search algorithms are detailed in Etak's patent US 5,694,534. In this patent, the inventors used the real time map display application as an example but could also have used the best path application.

Another useful aspect of this innovation is that a hierarchy of many levels can be automatically built from the database (i.e. data driven) without manual adjustments based on path finding algorithm or other heuristics. The hierarchy could be built into several levels to store appropriate data in carriers needed for specific tasks (whether it be display of a map at a certain scale or path search of a map at roads of a certain class). Bosch, with only two map levels in EVA, had the problem that the storage blocks would invariably be too small for bringing in enough data or too large, containing too much data to efficiently process.

The topology based carrier hierarchy was responsible for a tenfold reduction in memory and an even larger speed enhancement over conventional map data structures at the time. It was this innovation that enabled Etak to make its intense use of maps in the Etak Navigator™ and Bosch's TravelPilot. Different data carriers stored different map objects for different searches (display, geocoding etc.) because the inherent data for each search was different. These separate map files worked synergistically to provide the overall map access performance built into Etak's navigation design.

Bosch chose not to use Etak's routing algorithm, so no direct software match is likely to be found in versions of the TravelPilot. But Tele Atlas has found that Bosch did make use of Etak's concepts of a purely data-driven map file construction. Bosch used the carrier technology and modified the carrier tree and the multiple level concept, to support their routing application. They chose administrative areas (a 2-cell attribute) instead of road classification to define their multi-leveled carrier-based structure (administrative areas also have an adequate number of levels to structure a tree with differing levels of detail). Such modifications (from road to administrative area based hierarchy) could be easily accomplished within the Etak technology. Carrier trees and multiple levels (many more than two) gave Bosch the technical insights necessary to overcome their mental blocks and progress from a research demonstration (EVA) to a production-level product. With a Bosch Header File still archived at Tele Atlas facility near Hildesheim and once used to compile maps for TravelPilots, Tele Atlas has been able to unpack the data files on the RGS05 CD to the point where the multi-level carrier bounding boxes can be displayed – consistent with Etak's DB&S technology.

III. Summary

Etak's innovations helped Bosch in many ways. Not all manifest themselves in ways that are easily detectable in versions of Bosch's products. Hardware such as the Etak compass played an important part and is easy to observe. Software was instrumental in Bosch's continuing product line as can be seen by source code comparisons in the dead reckoning/map matching/GPS position determination code, the heading up moving map/dynamic labeling display code, geocoding, utility libraries and even the real time multi-tasking kernel that is the operating system code. Mapping technology was key to making the map economically and flexibly adding and maintaining new features such as traffic attributes, but specific map making technology is not traceable from the final map product. And finally but perhaps, most importantly, Etak's Database and Storage (DB&S) Technology paved the way for Bosch to conceptually overcome the map related blind alley that would not let them scale their routing to large areas such as all of Germany. By using a Bosch data structure document the data files in the RGS05 can be unpacked and analyzed to show that Bosch did indeed make direct use of this technology for all its map related applications.