KNOWLEDGE MANAGEMENT IN THE ENHANCED TRAFFIC MANAGEMENT SYSTEM

by

Christopher Gabriel Rodarte

B.S., Electrical Engineering and Computer Science
Massachusetts Institute of Technology, 1997

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Electrical Engineering and Computer Science

at the Massachusetts Institute of Technology

August 24, 1998

Author____
Department of Electrical Engineering and Computer Science
August 24, 1998

Certified by____
Amar Gupta
Co-Director, Productivity from Information Technology Initiative
Thesis Supervisor

Accepted by____
Arthur C. Smith
Chairman, Department Committee on Graduate Thesis

KNOWLEDGE MANAGEMENT IN THE ENHANCED TRAFFIC MANAGEMENT SYSTEM

by
Christopher Gabriel Rodarte

Submitted to the
Department of Electrical Engineering and Computer Science

August 24, 1998

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

ABSTRACT

The Enhanced Traffic Management System (ETMS) functions as the primary Database Management System (DBMS) for real-time flight information administered by the Federal Aviation Administration. The ETMS is a mission critical system responsible for facilitating air traffic control throughout the United States. The design of the ETMS represents a pinnacle achievement of modern data management. This thesis project will investigate the development of the ETMS and will identify several recent design decisions that illustrate a paradigm shift in data management techniques. These design decisions portray the initial implementation of a Knowledge-Base Management System (KBMS) wherein the system architecture shifts focus from data management toward knowledge management. DBMS and KBMS technologies will be introduced and compared. Specific implementations of each technology will be discussed and identified in the Enhanced Traffic Management System. Suggestions for further improvements in the ETMS design architecture will be entertained and several alternative design scenarios will be introduced.

Thesis Supervisor: Amar Gupta
Title: Co-Director, Productivity from Information Technology Initiative,
MIT Sloan School of Management

ACKNOWLEDGEMENTS

This thesis represents the culmination of my learning and research experiences at the Volpe Transportation Systems Center in Cambridge, Massachusetts. This work, and indeed most of my educational successes, would not have been possible without the support, guidance, and assistance from a great number of people. I would like to acknowledge Rick Oiesen and Marv Todd at the Volpe Center. Their insights and mentorship helped shape much of the information that is included in this thesis. I am particularly indebted to my thesis advisor, Dr. Amar Gupta, for providing me with an exceptional learning experience. His comments and feedback provided the encouragement necessary to complete this document.

I would also like to thank the friends whom I have met during my five years at MIT. They have truly made my college experience one of the most enjoyable times of my life. Thanks go out to the Delts, who are by far some of the nastiest guys that I have ever lived with. Thanks for the fond memories and the unbelievable stories. Special thanks go out to Chappy, Chhabra, and Krebs for making life as a graduate student more fun than I thought it would be. I would especially like to thank Domino and Johnny K. for being the best friends that I have ever had. I wish you both the best that life has to offer. I am also greatly indebted to John Rusnak, my fellow course 6 major and friend, who helped me through one of the toughest degree programs at MIT.

Most importantly I would like to thank my family for helping me through hard times and good times. Mom and Dad, thank you for all the support and encouragement that has helped me accomplish feats that I could never have done alone. I am also grateful for all the prayers and words of encouragement from my relatives. Together, they have helped shape who I am today and who I will be tomorrow.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

1 INTRODUCTION

The task of controlling the National Airspace System is quite complex. A myriad of diverse information must be evaluated in order to direct the flight path of each airplane entering or leaving the United States. There are over 47,000 permanent employees working for the Federal Aviation Administration (FAA) whose mission is to ensure the safe and efficient use of the nation's airspace. In a given hour, there may be as many as 4,000 commercial flights departing from any of six thousand airports in the United States [16]. Traffic Management Specialists are responsible for overlooking the status of all en route flights and those scheduled for departure. Their goal is to minimize delays and react to any critical situations arising from congestion. Traffic congestion can result at almost any given moment and is often caused by holiday travelers, adverse weather conditions, or peak business commute hours to name a few. Ideally, the Traffic Management Specialists are able to identify these problems quickly and make the corresponding flight adjustments to lower the traffic demand on a given airport or flight route. In order to accomplish this task, the Traffic Management Specialists rely on an extensive flight information database that includes other information pertaining to the National Airspace System (NAS).

1.1 Enhanced Traffic Management System

The Enhanced Traffic Management System (ETMS) acts as the primary interface between the management experts and the flight information database. This system serves several purposes. The ETMS supports a graphical interface that can access live flight information. This interface includes a geographical map of the United States and displays live flight information as airplane icons traversing the map. The user interface provides the option to display additional information including airports, runways, weather radar maps, and congested airspace locations to name a few. The ETMS is also responsible for maintaining and updating several distributed databases. Flight, weather, and geographical data are submitted from external sources that provide real-time and historic information. Thus, the ETMS must keep this information in a consistent state while providing the services that help the traffic managers predict and rectify congestion problems.

1.2 The Need for Automation

There are currently about 17,000 controllers who are responsible for directing air traffic operations in the United States. Each controller has a specific responsibility based on his or her position in a three-level hierarchical organization [4]:

1. The Air Traffic Control System Command Center (ATCSCC) is the supreme authority in the management of nationwide traffic problems and coordinates the actions taken by the other traffic management facilities.

2. The Traffic Management Units (TMUs) at the Air Route Traffic Control Centers (ARTCCs) are the middle level managers who resolve traffic problems within the scope of the ARTCC. There are currently twenty-one ARTCCs in the National Airspace System.

3. The TMUs at the Terminal Radar Approach Control (TRACON) facilities are the lower level managers who solve problems specific to the terminal(s) under their control. There are currently twenty-five TRACON traffic management sites that control over two hundred terminal sites.

Together, these control centers direct the nation's air traffic flow 24 hours a day, 365 days a year. Each Traffic Management Specialist must be able to co-ordinate his or her actions between the traffic managers at the other sites and the radar control operators who implement their requests by communicating with the pilots of the airplanes involved. Traffic managers have the authority to delay, redirect, and/or cancel any given flight under the control of the FAA. Thus, their actions must be relayed to the respective airlines who operate the affected flights. Airlines utilize this information to reschedule flights and take the appropriate action to minimize customer inconvenience. The mere scope of this information distribution requires automation at several levels. For example, airlines submit flight plans for each scheduled departure to the ETMS. This information is then automatically converted to a detailed list of flight events that correspond to a "road map" like data structure of the National Airspace

System. Traffic congestion can then be projected into the future so that management experts have warning as to which "highways in the sky" will be crowded.

1.2.1 Additional Traffic Demand

The current level of automation in the ETMS is quite impressive. There are several information components, in addition to the flight scheduling, that are automated in the ETMS. For example, there are currently automated weather observation stations at many towered and non-towered airports that provide real-time weather conditions at locations throughout North America [16]. There are also automated switching facilities that control communications between traffic control centers, airline flight plan processing centers, and aircraft pilots. Unfortunately, the current level of automation is not sufficient to deal with the projected increase in air traffic flow. The current system is not capable of providing the most efficient flow of traffic and it is possible that incremental improvements will fail to offset the increase in traffic demand over the next several years. It is estimated that domestic commercial air traffic will increase by 30 percent through the year 2008 while international passenger traffic will rise 5.7 percent annually [28].

1.2.2 Move Toward Free Flight

The need for increased flight capacity will drive a necessity for additional flight planning and information services. This will undoubtedly require additional automation within the ETMS. However, the FAA has adopted a policy that will push the capabilities of the Enhanced Traffic Management System even further. The FAA Office of Air Traffic Systems Development (AUA) believes that future air traffic control should allow airplane pilots the freedom to select their own flight paths and aircraft speed at will:

> "AUA's vision for future air traffic control and management centers on a collaboratively-managed system in which airspace users take a more active role in system control and FAA restrictions on flight planning and scheduling are reduced. This system will eventually lead to *free flight*.[1]" [16]

---

[1] The FAA defines free flight as "a safe and efficient flight operating capability under instrument flight rules in which the operators have the freedom to select their path and speed in real time Air traffic restrictions are only imposed to ensure separation, to preclude exceeding airport capacity, and to ensure safety of flight. Restrictions are limited in extent and duration to correct the identified problem." [16]

The concept of free flight introduces a paradigm shift to the present air traffic control architecture. Pilots have very little freedom in the flight paths they choose today. In fact, most aircraft are under the constant control and supervision of radar tracking stations, airport control towers, traffic management units, and airline flight plan processing centers. These centers are the primary users of the Enhanced Traffic Management System and thus the design architecture of the ETMS focuses on controlled flight and facilitating those individuals who supervise the National Airspace System. Accordingly, increased automation will not be a sufficient solution to the goal of free flight. The ETMS architecture must be revised such that pilots have a more active role in air traffic control. This will involve a corresponding paradigm shift in the manner in which the ETMS handles information. Namely, the system architecture must shift focus from data management toward knowledge management. System analysis will reveal that the most feasible solution towards the goal of free flight involves the creation of a Knowledge-Base Management System within the ETMS.

1.3 Developing New Technologies

The FAA recognizes the need to implement new technologies to improve the management of the National Airspace System. Currently, the FAA is focusing their efforts on developing additional automation techniques in several areas of traffic management. Development teams are working on programs that will support traffic flow managers in making decisions with respect to congestion problems. More importantly, the FAA has also taken steps to develop knowledge management tools within the ETMS architecture. These tools remove the need for human interaction at various levels in the acquisition and distribution of information. This thesis will investigate the utilization of knowledge management in the Enhanced Traffic Management System and will suggest other areas where knowledge management would improve system capabilities.

1.4 Thesis Structure

This chapter has introduced several concepts that relate to the traffic management of the National Airspace System. The reader should now have a general understanding of the need for traffic management, the systems involved, and the goals for future traffic control. Chapter 2 will discuss current technologies utilized in data management and will introduce several new concepts that are currently being deployed in information management systems. Namely, Database Management Systems will be compared and contrasted to Knowledge-Base Management Systems. Chapter 3 will describe the current architecture of the Enhanced Traffic Management System referring to the concepts explained in Chapter 2. The flight information database will be a major topic of interest and its design will be correlated to the present hierarchy of air traffic control. The automation of traffic demand alerts will also be discussed in detail as it involves the use of a simple knowledge base. Chapter 4 will identify deficiencies in the present system architecture and will investigate complaints from traffic managers utilizing the system at various field sites. Solutions to these problems will be delineated in Chapter 5. Several alternative design scenarios will be introduced and their tradeoffs will be discussed. Chapter 6 will provided the conclusions reached by this investigation and will recommend further implementations for Knowledge-Base Management Systems.

Chapter 2

2 BACKGROUND

This chapter focuses on the fundamental reason why the paradigm of Database Management Systems should be considered separate from the paradigm of Knowledge-Base Management Systems. Both paradigms have many similarities and some consider them as being one and the same.[2] Indeed, the same software technologies are used to develop both DBMS and KBMS programs. However, the fundamental goal that each system serves to accomplish is quite distinct. Database Management Systems focus on the storage, retrieval, and manipulation of *data* whereas Knowledge-Base Management Systems focus on the acquisition and manipulation of *information*. The former requires a computational theory (i.e. query languages, report generators, etc.), while the later requires a semantic theory (inference engines, lexical representation, symbolic constraints, etc.) [6]. A concise overview of Database Management Systems and Knowledge-Base Management Systems will be given, followed by a discussion of similar technologies that exemplify recent innovations in data management. Several new airport facilities contain the latest information management technologies and important lessons can be learned from the problems that were encountered during their implementation. This chapter will serve as a foundation for the subsequent analysis of the Enhanced Traffic Management System.

2.1 Database Management Systems

The need for effective and efficient data management is ubiquitous in today's information society. Businesses, governments, schools, and ordinary people utilize some form of data management on a daily basis. Where there is a computer, there is a need for data management. Database Management Systems emerged as the primary tool for processing operational data in the early 1960s [27]. Since then, DBMSs have been adapted to handle many diverse end user needs. Regardless of the role that a specific DBMS serves, there are common features that all

---

[2] For example, Reimer and Hahn introduce a database model that "is based on the idea that there is a far reaching intersection between both (knowledge representation languages and data models) and that they should be treated under a unifying approach." They go on to say that, "we use the terms 'data model' and 'knowledge representation language' as well as 'database' and 'knowledge base' as synonyms." [36]

Database Management Systems share. These features include provisions for the following: data independence, data integrity, data concurrency and consistency, recovery, access control, controlled data redundancy, centralized control, and data maintenance [22]. In order to better understand the differences between DBMSs and KBMSs, it is important to elaborate on the aforementioned concepts.

2.1.1 Data Independence

Perhaps one of the most common features between various DBMSs is their role in separating data from other applications. Put simply, Database Management Systems act as the primary interface between computer applications and the data that they use. This separation makes it possible to reorganize the physical data without any serious affect on the programs that utilize the data. For example, a business may choose to merge several disparate databases into a single central database without the need to rewrite all of their existing applications. This is possible even if their legacy databases were built by different commercial vendors and/or implemented on different computer systems [27].

Given that DBMSs provide the resources necessary for data independence, they are inherently centered on the problems concerned with the computational theories necessary for realizing databases on physical machines. Thus, Data Base Management Systems are often focused on the implementation of robust, reliable, efficient, and secure database applications [6]. Several businesses (Oracle, Sybase, etc.) develop such technologies and devote many resources to improve data management capabilities. However, one must realize that the majority of current DBMSs deal only with data at an operational level. That is to say that database systems rarely focus on the content of the data they control. Their main function is to deliver and store data that other applications and/or people use. Hence, it is the end user of the DBMS who is concerned with the actual content of the database that it is a part of.

2.1.2 Data Integrity

Database Management Systems usually provide facilities that help ensure that the database does not become corrupted. This gives the end user some assurance that their data are reliable and, in case of disk failures or other critical system failures, the DBMS will provide a certain level of redundancy such that data integrity is maintained. The DBMS will also verify that incoming data are in an acceptable form to ensure that other applications are not creating corrupt data. At this level, DBMSs are concerned with the physical storage of data and the integrity of the machines where the data are deposited.

2.1.3 Data Concurrency and Consistency

There are often multiple applications and users who access a database concurrently. For example, ATM networks often access personal bank account balances to determine money available, withdraw funds, and update balances. The transactions that occur must be atomic and the balances that are given must reflect a consistent state of the database. For instance, an ATM withdrawal should occur only if the funds are available and should update the current balance. Database Management Systems must be able to handle situations where multiple users wish to access the same data concurrently while processing any requests to update and/or change specific data values. There may be a situation where a bank customer wishes to withdraw funds while a check is being cleared. Both processes will access the same data and try to make changes at the same time. However, only one process should be allowed to change the data value or else the database would most certainly become corrupted. In other words, both processes will attempt to write a new balance to the database, which would not reflect the true balance of the account. Atomic transactions remedy these race conditions by halting one process while the other updates the balance. The process that is halted is then given the new balance after the first process has finished its transaction.

A consistent database guarantees that every user views the same picture of data values. There are other instances where the consistency of data is a central issue. Distributed databases involve computer systems where the data reside on multiple machines and multiple copies of the data may exist. Thus, concurrency control becomes a more complex task to implement in distributed databases. Issues such as communications between sites and the uncertainty of the order of events (such as data updates) make it difficult to provide data consistency [21]. Nonetheless, distributed database systems provide the same level of consistency that other databases offer by utilizing centralized control. Each copy of data is monitored by a centralized process that ensures every copy is updated when a change occurs. The process also has the ability to lock certain data structures so that other applications never receive stale data. Further discussion of distributed databases will be provided in a later section.

2.1.4 Recovery

System failures are often inevitable. Database Management Systems must provide provisions that allow system recovery. This often requires a backup copy of the database that represents the system state at a given point in time. If the database becomes corrupted or if data are lost, the backup copy can be used to restore the lost information. This is accomplished by logging all the changes that were made to the database after the backup copy was created. Thus, backups need not occur frequently while minimizing the risk of lost data.

2.1.5 Summary

Database Management Systems focus largely on the tasks associated with controlling and manipulating data. Implementing a DBMS often involves creating a set of computational tools that allow efficient storage and retrieval of data on physical machines. Figure 2-1 illustrates the role that a DBMS serves in a banking business environment. Note that the various applications receive data from the database via the DBMS. For instance, bank customers might access their account at ATMs, bank teller windows, and/or at department stores where they wish to purchase merchandise by drawing from their account. These applications access and update the database by telling the DBMS to carry out certain actions. These actions may then be logged in various files as debits, credits, transfers, etc. so that the state of the database can be restored in case of a system failure.

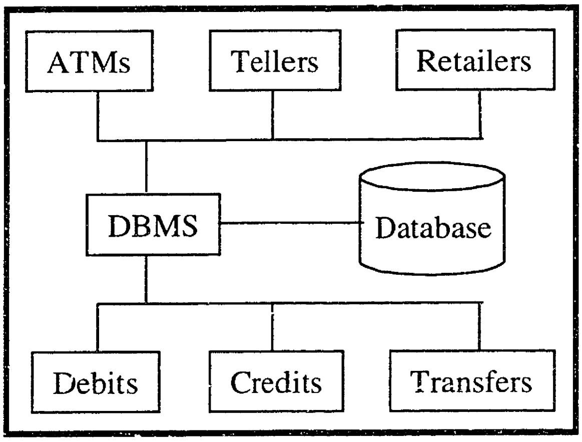


Figure 2-1: Example of a DBMS Architecture

As one might conclude, the DBMS has a very low-level role in the overall system architecture. The DBMS provides data services to the applications that utilize the database in carrying out transactions. Hence the DBMS does not deal with the information content of the database it controls. Rather, the bank customers and/or employees utilize the DBMS to make decisions regarding the information content of the database.

2.2 Distributed Database Management Systems

Distributed databases have emerged with the advent of the personal computer and the networked computing environment. Economics have driven many corporations to abandon the mainframe computer in favor of smaller and more affordable microprocessors. These smaller computers are often linked together over a Local Area Network (LAN). This configuration allows distribution and sharing of data and other computing resources thus lowering the overall operating costs of the company [5]. Distributed database systems offer several additional benefits:

- Performance benefits over centralized computer systems
- Higher reliability when compared to single processor systems
- Application specific solutions to tasks that are inherently distributed
- Modularity that allows incremental addition of enhanced machines and/or software

Today, distributed databases are almost as ubiquitous as centralized database systems. Distributed databases are often utilized in mission critical systems where reliability is a major concern. The distributed approach utilizes multiple copies of the same database on multiple computers. Thus, if one computer crashes, a second computer can quickly take over its responsibilities and continue processing data such that a user would not even realize that there was a computer failure. For example, the CIRRUS banking network utilizes a distributed database system to carry out ATM transactions. Several large consumer banks in the United States originally implemented this network: Baybank of Boston, First Chicago, Manufacturers Hanover of New York, Mellon Bank of Pittsburgh, and National Bank of Detroit [18].

These individual banks did not have the resources to install ATMs at major cities throughout the US (at the time, the installation of a single ATM cost $100,000 with a yearly maintenance cost of $50,000). Thus, these banks decided to create a cooperative banking network that would allow their customers to use ATMs at other banking institutions. Each bank controlled its own database for its set of customers. Consequently, the banks needed to create a distributed database that would allow them to share these data such that they could authorize transactions by customers from a peer institution. The schematic network for this distributed database is shown in Figure 2-2.

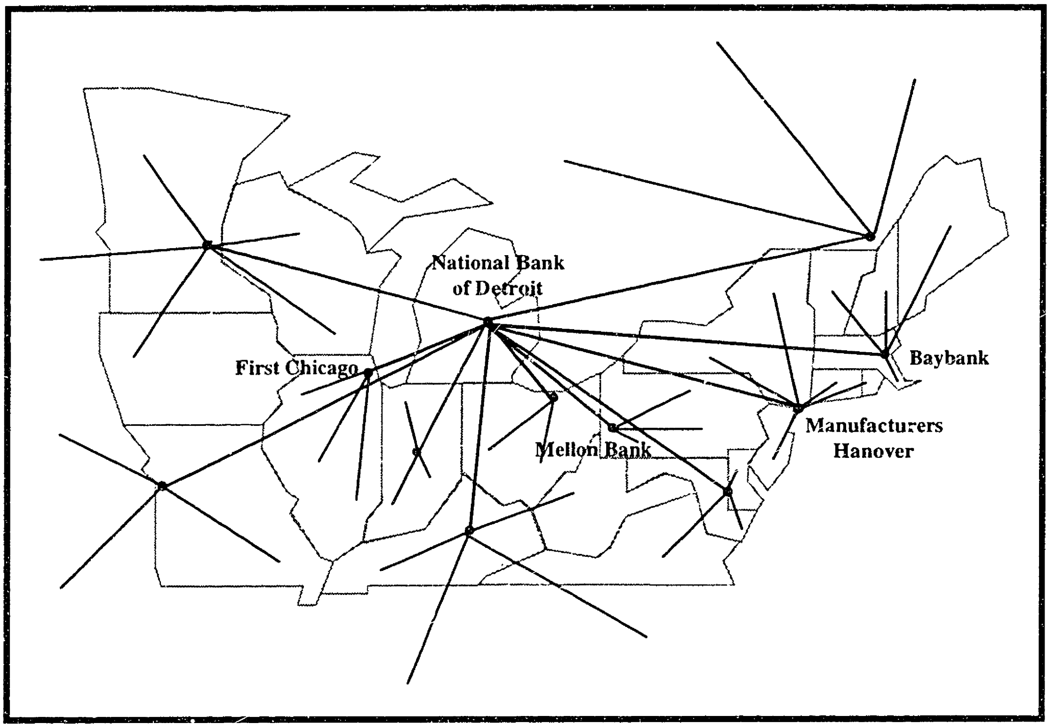


Figure 2-2: CIRRUS Network Illustrating Node Connections

Note that every node is linked through a primary switching node at the National Bank of Detroit. The primary node is connected via dedicated high-speed lines to several secondary nodes that act as regional access points. CIRRUS has grown much larger since the original implementation shown above, but this figure helps illustrate the primary concepts involved in a Distributed Database Management System (DDBMS). This particular DDBMS utilizes a single switching node to carry out all transactions. Thus, if a Baybank Customer were to withdraw funds from a Manufacturers Hanover ATM in New York, this transaction would be processed via the primary switching node at the National Bank of Detroit. Since this node handles all transactions, it keeps a log file to record all account credits, debits, and transfers. This provides centralized control and consistency between the various account databases that exist at each participating bank.

One obvious deficiency to the centralized control implementation is that the primary switching node can become overloaded with transactions. The original CIRRUS switching node could handle only 2 transactions per second [18]. However, the network generally had less than 500,000 transactions per month and an average transaction would take less than 15 seconds to dispense cash (from time of request to time of approval). The switching node contained six separate hard disk drives. Two disks were used to store the transaction log file with one acting as the primary and the second as a backup. Another two disks were used as system file storage devices. These disks contained the same copies of all system files in case one of the disks failed. The final two disks were used to store operational files such as bank routing tables and processor status files. Again, both disks served as dual copies in case of a disk failure.

The log files kept on the primary switching node would be used at the end of the day to settle all inter-banking customer transactions. Each institution also kept a record of its customer transactions so that the account balances would be up to date. This level of redundancy ensured that every transaction was properly logged and that any single system failure would not prove fatal. The primary switching node had a second computer that reflected the same state as the operational computer. Thus, if the primary computer failed, the backup would step in without any loss in availability. The communications network also contained a secondary phone line that could be used in case the dedicated line went down. The distributed database used by the CIRRUS network represents a highly reliable database management system. ATM networks are also inherently distributed in their nature; thus a DDBMS is an obvious solution for their implementation.

The CIRRUS network addresses several design issues that form an integral element of distributed database systems. These design issues emphasize that DDBMSs should appear no different from DBMSs. They also address many of the advantages that a DDBMS should possess over a DBMS. Namely, these issues include transparency, flexibility, reliability, and performance [39]. The CIRRUS network will be analyzed to determine how well it meets these design criteria.

2.2.1 Transparency

Transparency is a feature that allows DDBMSs to imitate normal centralized Database Management Systems. The user should not be aware that the system is implemented on multiple machines that exist throughout the United States. For example, the Baybank customer who withdraws funds from a Manufacturers Hanover bank in New York should not need to do anything differently from what he or she would normally do to withdraw funds at home. Thus, the CIRRUS network should present a uniform interface to users at all locations. This is accomplished by ensuring transparency at several different levels in the system.

- Location transparency masks the location of the physical machines that the DDMS is implemented on.
- Migration transparency provides contingencies so that physical machines can be relocated or removed without disruption to the larger system.
- Replication transparency allows applications to make copies of files and other resources without the users noticing.
- Concurrency transparency ensures that users will not notice other users of the system.

The CIRRUS network provides location transparency because users experience the same services regardless of where they withdraw funds. The network maintains the same efficiency level at ATMs throughout the United States and Canada. The network is also migration transparent because individual ATMs can be removed without any disruption in service at other locations. Even the main switching node can be removed without disruption to service. Recall that the switching node has a backup computer that can take over its responsibility in case of failure. The switching node is also replication transparent because it often makes multiple copies of system, log, and application files without any delay in service. Finally, the CIRRUS network provides concurrency transparency because users need not worry about other users of the system. This is true because the network always portrays the then current balance on a given account so that a user can rest assured that they have funds available to withdraw when they view their balance at an ATM.

2.2.2 Flexibility

Distributed Database Management Systems offer flexibility because their implementations are modular. Flexibility also suggests that a DDBMS can easily evolve over time. Database management systems often require upgrades in system software and hardware. Traditional databases were implemented on mainframe computer systems [5]. Thus, if the database grew or if newer computer hardware was needed, the entire mainframe computer would have to be replaced. Distributed databases are often implemented on multiple machines with each machine assigned a specific task in the maintenance of the database. This makes it much easier to upgrade components of the database piecemeal. It also allows the database software to be upgraded without the need to rewrite the entire system.

There are many examples of how the CIRRUS network maintains flexibility. ATMs can easily be upgraded without the need to reorganize the entire network. Additional regional banks can be added to the network without much additional work. The routing tables would need to be updated with the new bank members and the transaction record files would be expanded to include additional entries. The majority of the burden would rest with the bank that wishes to be added to the CIRRUS network. They would need to provide an interface from their proprietary accounts database to the CIRRUS network. This would not affect the CIRRUS implementation, but it may be necessary for the new bank to change their internal database design to adhere to the CIRRUS authorization protocol. Hence, the CIRRUS network exemplifies a flexible Distributed Database Management System.

2.2.3 Reliability

Recall that distributed database systems were originally implemented to provide a higher degree of reliability than centralized database systems. Reliability is directly related to availability. Therefore, a system that is implemented on multiple machines is generally more reliable than a system implemented on a single machine. Multiple machines provide greater availability because if one crashes, the others will generally carry on without any loss in their performance. If the system is designed such that no single machine fault causes any major complications, then the system is said to be fault tolerant. This is generally impossible with single machines. Imagine a database that resides on a single machine. There are an endless number of possible faults that would render the database unusable. For example, processor failures, power outages, disk crashes, network downtime, or system reboots could all cause the database to be unavailable for a given period of time. However, this would not be the case if the database was distributed on multiple machines each with their own backup system.

The CIRRUS network demonstrates reliability throughout the various components. The node computer exemplifies redundancy at all levels. Recall that each storage component has a backup system. For example, the system software, database records, and transaction processes are all kept on separate disks each with their own backup disk. The primary switching node also has multiple CPUs. Each CPU computes the same instructions and their results are compared. This provides error checking as well as redundancy to ensure that the programs continue to execute. The node computer also has network backup capabilities in case the dedicated transmission lines fail. These measures help ensure that the most centralized component of the system does not fail and, if it were to fail, the backup node would be capable of taking over the primary switching node's responsibilities.

The CIRRUS network maintains reliability at other levels as well. For instance, each ATM logs its own transactions in the event that its data were lost over the network. These contingencies illustrate that there are many copies of the same data that exist on multiple machines and disk drives. One could eliminate several machines without any loss in the overall state of the distributed database. The CIRRUS network is also capable of providing service to a subset of the original users. For example, if the BayBank regional provider were to go down, CIRRUS would be able to provide service to its remaining customers. Obviously these customers would not be able to get CIRRUS service at a BayBank ATM due to that network's failure, but they would not be affected at any of the other CIRRUS ATMs. Thus, the CIRRUS network does not rely on any particular provider to guarantee overall service. The only exception being the fact that the National Bank of Detroit owns and operates the primary node switch and reserves the right to cancel service to any particular regional provided. However, this is more of a marketing and control issue than a system design issue.

2.2.4 Performance

Distributed Database Management Systems often provide greater performance because individual database processes can be allocated their own host machines. This argument does not attempt to claim that multiple processor machines are faster than single processor machines (the latter having a faster processor than the former), but instead emphasizes the theory that database performance can be improved by modularizing its architecture and dedicating machines for each of its components. This often allows the entire database to be stored in memory thus expediting the access of data. However, the performance of a distributed database is often lowered due to the need of communications between modules. This can be a limiting factor of the system's architecture as is apparent in the CIRRUS network. The primary switching node is a major data bottleneck within the CIRRUS network. Recall that all inter-bank transactions must pass through this node. The major delay in transaction processing arises because messages must be sent through this particular computer regardless of where the customer is physically using the ATM.

Returning to the Baybank customer who wishes to use a Manufacturers Hanover ATM in New York, it is evident that physical distance for message relay is not optimal. The request for cash dispensation passes from New York to Detroit and then to Boston. The authorization message is sent from the regional network in Boston to Detroit and then finally back to New York. If we assume the message travels at the speed of light, this will add approximately .6 μsec to the authorization process. This may seem insignificant, but when other delays such as switching time, processing time, and network traffic is considered, the delay can easily become very noticeable in the system. Inter-process communication is a major detriment to the performance of distributed databases. At times, delays in message passing can often lower the performance of distributed databases below that of centralized databases.

2.2.5 Summary

Distributed Database Management Systems have many benefits over normal Database Management Systems. They are often utilized in mission critical systems where reliability and performance are of utmost concern. DDBMSs are also a natural choice for systems that must be implemented in a distributed environment. The CIRRUS network example illustrates that the distributed approach is a natural choice for the implementation of a heterogeneous database in a large geographical environment. The CIRRUS network also exemplifies the need for DDBMSs to be transparent to the user while maintaining dynamic capabilities so that the system can expand when the needs of the user expand.

2.3 Knowledge-Base Management Systems

The field of Artificial Intelligence (AI) has recently developed new theories for issues concerning data and information management. These theories provide a deeper understanding of data manipulation and derive new knowledge about the information that is contained in various databases. The discipline is still debating many of these theories and thus, there are often conflicting points of view and different areas of concentrated study. For example, the terms expert systems, knowledge representation, knowledge acquisition, rule-based systems, business intelligence, and knowledge management are various topics that AI researchers are investigating [41]. Each specialized area of study utilizes a unique approach to model human problem solving abilities. Knowledge-Base Management Systems exemplify research efforts that attempt to model the way humans represent and utilize information in decision-making processes. There are several issues that KBMSs attempt to address and these issues best differentiate Knowledge-Base Management Systems from many similar technologies.

2.3.1 Knowledge Representation

A very important component of a KBMS is its knowledge representation scheme; put simply, a knowledge representation tells a KBMS how to view the world and emphasizes those properties that it should deem important. Davis et al argue that a knowledge representation can be defined by considering five roles that a representation serves [10]. These roles help distinguish Knowledge-Base Management Systems from Database Management Systems. As we will see, a knowledge representation provides a deeper understanding of the content of the data that resides with the system instead of the simple computational tasks that exemplify a database system. Davis et al claim that a knowledge representation:

1. acts as a substitute for the objects that exist in the real world

2. is a set of ontological commitments that focus attention toward the important characteristics of the real world

3. is a partial theory of intelligent reasoning that recommends inferences based largely on modeling human expert reasoning
4. acts as a medium for efficient computation
5. acts as a medium of human expression

These criteria suggest that knowledge representation attempts to model both physical and abstract objects in the real world that mirror how humans think and act. Thus, Knowledge-Base Management Systems deal with information in much the same manner that people would. For example, knowledge representation can encapsulate notions like causality, actions, beliefs, and assumptions, which make it possible to reason and respond to situations in the real world. This representation allows the system to make inferences and thus, the knowledge representation makes it possible to easily acquire new information about the world.

The first role that knowledge representation serves is fundamental to many applied sciences. Chemistry, Physics, Biology, Mathematics, etc. attempt to describe the world by modeling the objects that exist. Chemists view the world in terms of molecules that interact with one another. Physicists treat the world in terms of point masses and how they are affected by outside forces. Biologists are often concerned with cells and how they interact in a larger organism. Mathematicians focus on logical reasoning and formalisms. Each discipline creates a model that is a substitute for the physical object they wish to study in the real world. One can view these different sciences in terms of the knowledge representations that they use. This is the second important role that a knowledge representation fulfills.

Every knowledge representation is concerned with only a limited subset of the real world. Complexity often limits the amount of knowledge that a given representation can embody. It would be impossible to capture an accurate model of everything that exists in the real world. Thus, the next best thing is to accurately model a small subset of those objects and concepts that are deemed to be important for a specific task. For example, the MYCIN system utilizes a knowledge representation that is concerned with the diagnosis of infectious diseases. MYCIN's knowledge is limited to medicine, drugs, bacterial organisms, logical rule chaining, and probabilistic analysis [9]. The system is capable of diagnosing bacterial infections by asking basic data about the patient (i.e. age, sex, blood culture results, sickness complaints, etc.). Note that MYCIN is only capable of treating bacterial infections and that the system suggests treatment via probabilistic analysis. The system's view of the world is limited and the scope of its knowledge representation is finite. It would be very difficult to devise a system that is capable of diagnosing every possible disease or ailment that a patient might have.

The third component of a knowledge representation focuses on the task of acquiring new information about the present state of the world. In a way, knowledge representation provides a foundation for a specific framework of intelligent reasoning. This thesis does not attempt to define what intelligent reasoning consists of, but rather introduces alternative theories that have proven effective in imitating human decision making processes. A knowledge representation is obviously concerned with a very narrow view of the physical world. Thus, a knowledge representation is only a partial theory of the human decision making process. The term "Expert System" is often used to refer to such knowledge representation implementations. Expert Systems typically embody knowledge that is derived from human experts in the field. For instance, MYCIN suggests treatments based on what a doctor would normally prescribe given the same information. However, MYCIN goes a step further than many human experts in that it also has a large amount of probabilistic knowledge embedded in its knowledge representation.

Davis et al suggest that there are three design issues that must be addressed when developing the inference engine for a knowledge representation system. Specifically, a system designer must address what method will be used to make intelligent inferences, what inferences will be allowed, and what inferences will be recommended [10]. The solutions to these design issues largely depend on the system to be implemented. MYCIN is capable of making both logical and probabilistic inferences. For example, if the patient is male, MYCIN can logically infer that he is not pregnant. Thus, MYCIN can choose from a larger set of medication for treatment (e.g. there is no need to worry about drugs affecting an unborn child). MYCIN also utilizes probabilistic inferences to diagnose a specific type of infection (i.e. if a blood culture is gram positive, there is a certain likelihood that the patient may be infected with ecoli) [9]. Together, these methods embody MYCIN's intelligent inference process.

An inference engine must also be capable of distinguishing what inferences should be allowed and what inferences should be recommended. Given MYCIN's large amount of information about diseases, the inference engine is capable of suggesting numerous possible disease conditions and corresponding treatments. However, the system would be rather useless if it only generated a large list of possible treatments. MYCIN utilizes probabilistic information to determine what the most likely disease is and recommends treatment to remedy this ailment. Thus, MYCIN's inference engine is also concerned with what inferences should be recommended given those that it could make.

The fourth role that a knowledge representation serves is concerned with the computational feasibility of the implementation. Earlier arguments suggest that the complexity of the real world limits the scope of a knowledge representation. It is also true that the computational resources available are a limiting factor in the representation's depth of information. The inference engine can only make a finite number of inferences over a given time period. The number of inferences is largely dependent on the computational power available. Consequently, the knowledge representation is limited in the number of inferences it can make and should never rely on reasoning methods that involve a very large number of inferences.

The fifth criterion of knowledge representation states that it should be a medium of human expression. This role follows naturally from the previous discussion. Expert systems try to encapsulate the knowledge that professionals utilize to carry out their job in the real world. Thus, a knowledge representation captures the essential reasoning processes that people use to solve problems. These problem-solving skills are enumerated in the knowledge representation and one can conclude that the representation is also a natural medium for human expression.

The representation utilized in a Knowledge-Base Management System plays an important role in defining the information content of the system. The five criteria used to define a knowledge representation emphasize the level of intelligent reasoning that the system is capable of achieving. These criteria will be used to analyze the knowledge representation utilized in the Enhanced Traffic Management System. They will help to distinguish DBMS structures from KBMS structures and will provide insight as to how the overall architecture might be improved.

2.3.2 Self-Understanding

Ideally, a KBMS would include facilities that make it possible to query the system as to why it made certain inferences or decisions. Given that a KBMS is a medium for human expression, it should also be a medium for self-understanding. If you ask an expert why he/she made a certain suggestion, he/she would most certainly be able to give you a list of reasons that outline their intellectual reasoning process. This should also be true for a Knowledge-Base Management System. For example, MYCIN is capable of listing the inferences it makes in an order that follows its logical and probabilistic reasoning methodology. The user is able to step through this list to see if it makes sense to them much the same way they would query a person.

By providing an explanation mechanism, Knowledge-Base Management Systems also provide a convenient mechanism for knowledge transfer. The knowledge that goes into a KBMS should easily be conveyed to a person or application that exists outside of the system. Thus, a KBMS is concerned with the transfer of knowledge from one information base to another, be it people or other knowledge-base applications. The world consists of heterogeneous knowledge bases. Each knowledge base would benefit from the capability of sharing information with other knowledge bases. This should be possible even if the individual knowledge bases utilize different knowledge representations and/or are implemented on different types of computer systems.

2.4 Comparison and Discussion

As one might imagine, it is often difficult to distinguish between DBMS and KBMS implementations. There are often conflicting research papers that attempt to define Knowledge-Base Management Systems in a very precise manner. In fact, several papers suggest that Expert Systems should be considered separately from KBMSs while others suggest they should be considered one and the same [22]. There is also a large amount of literature that is devoted to addressing the differences between DBMS and KBMS architectures [6][22][10]. These references also contain conflicting points of view and focus on different aspects of each technology. Instead of dwelling on the inconsistencies, this section will discuss those opinions that are largely agreed upon.

2.4.1 Distinguishing Data and Knowledge

Most KBMS architectures will need some level of data management. Many researchers agree that many current KBMS implementations are developed on top of DBMS architectures. Hence, Knowledge-Base Management Systems will generally contain both data and knowledge content. Data content is identified by its computational nature (i.e. it is easily compiled from objects that exist in the real world and needs little or no human intervention). Knowledge is identified by its semantic nature and the need to acquire this information from human experts (i.e. a system designer will need to interact with the person to enumerate their expertise into the knowledge base). Knowledge also embodies generalized concepts and is easily applied to new situations whereas data is very specific and is useful in a limited number of settings.

The concept of a knowledge level is useful to distinguish between knowledge and data in a heterogeneous environment. The lower level is concerned with the storage and manipulation of data. The next level builds on the previous level and eventually a knowledge level will be formed. This is where intelligent reasoning occurs and the level where inferences are made. These decisions will often involve data, but they do not need human intervention as they would in a purely DBMS architecture. Although this thesis uses the terms data and information loosely (for grammatical variety), it is very important to realize that the concepts of data and information are not interchangeable. Thus, emphasis will be given when it is necessary to distinguish knowledge level information from simple data structures. Systems that exhibit knowledge management will be explained in terms of the knowledge level that they utilize.

2.4.2 Choosing an Implementation

There are several possible choices in implementing a Knowledge-Base Management System. As was mentioned earlier, there are different areas of concentrated study that are focusing on specific theories in knowledge management. For example, there are knowledge management systems that utilize logical inference. Other systems rely heavily on probability theory (e.g. MYCIN). Still, others are founded on the belief that intelligent reasoning is mainly a human cognitive process and that all forms of intelligent reasoning must be derived by modeling the human decision-making process. Each system has its strengths, and there are certain situations where one implementation works better than the others. From an architectural point of view, one of the primary concerns is the overall design of the system. There has been much debate surrounding how best to design a KBMS. The majority consensus is that Knowledge-Base Management Systems should be designed from the ground up whenever possible. Other solutions are feasible, but most researches believe that these solutions should only be used as a stepping stone to the development of a complete KBMS architecture.

The demand for knowledge management in the business environment has created the need to update many existing corporate databases. These databases are often very large and consist of heterogeneous data. Thus, it is often undesirable to completely overhaul the existing systems due to economic and reliability concerns. Instead, KBMS components are added incrementally to the DBMS such that the system attains a higher level of automation. The Enhanced Traffic Management System is an example of a DBMS that is being incrementally updated to include knowledge base components. Unfortunately, these systems are inherently limited in the knowledge level that they are capable of attaining. This is due to the fact that they must continue to use the old semantics of the system instead of creating a knowledge representation from scratch. The power of a KBMS stems from the fact that it is designed from the ground up to solve a particular problem at hand. Thus, the integration of KBMS and legacy DBMS systems constrain many of the benefits that an original KBMS architecture has to offer.

2.5 Contrasting Traffic Management Systems

The United States relies primarily on the ETMS for air traffic management operations. In fact, the ETMS is responsible for managing air traffic operations throughout the US, parts of Canada and Mexico, and over 80 percent of the world's controlled oceanic airspace [16]. The ETMS is a unique system in that it has evolved over a period of several decades. Traditional database components are integrated with newer autonomous systems that utilize knowledge management techniques. Thus, the ETMS consists of many different hardware architectures and software programs that have been assimilated over the years to aid in the management of the National Airspace System. New technologies are adopted using an incremental approach. Safety is of utmost concern and many of the original systems remain active to ensure operational availability. This often constrains the ability to implement state of the art technology in the Enhanced Traffic Management System. Instead, newer systems are employed slowly while the original architecture maintains most of its duties.

This section will compare the ETMS to other traffic management systems. Several countries have recently installed knowledge management systems at new airport facilities and are adopting the newest technologies for the management of their airspace. These technologies represent the future of air traffic management and reveal new methods that aid in reducing traffic congestion. However, many of these newer systems encountered serious problems during their implementation that caused severe traffic delays. The lessons learned from these high tech systems should not be forgotten when developing new automation facilities for the ETMS. Indeed, there are risks to be considered when deciding to implement automation and decision making support utilities in mission critical systems.

2.5.1 Traffic Management in Asia

Systems for air traffic management in Asia are somewhat different from the systems implemented in the Untied States. Several Asian countries have recently built new airport facilities and have also implemented new traffic management systems. These newer systems are radically more advanced than many of the systems implemented in the United States. This largely stems from the fact that most of the air traffic control systems in the U.S. consist of legacy based architectures. The United States originally implemented many of the computerized traffic control systems on IBM mainframe computers. These computers are out of date, but the replacement costs are so high that they are still running much of the system's software. Thus, the air traffic control systems in the U.S. must interface with existing components and this often limits the ability to install radically new technologies. However, this is not the case in Asia where there are often a very limited number of traffic management systems. Traffic management in Asia has recently become an important issue for many of the airlines with flights in the region. New control systems are being installed to cope with the expected increase in air traffic throughout Asia.

Much of the airspace over Asia is still unmanaged by air traffic control systems because of the prohibitive costs that characterize modern air traffic systems. For example, the United States is in the process upgrading 31 Terminal Radar Sites, 10 Long-Range (En Route) Radar Sites, 96 Radome Radar Sites, and 161 NEXTRAD Dobbler Radar Sites [16]. This represents only a small fraction of the overall radar-tracking infrastructure within the United States. The total costs incurred by the maintenance and upgrade of these traffic systems are expected to approach $5 billion a year [38]. Many of the poorer Asian countries are not prepared to pay such a high price for the implementation of air traffic control facilities within their territories. Unfortunately, air traffic demand is projected to increase faster in Asia than in the rest of the world. The increase in traffic has put additional strain on the region's limited air traffic management systems. In turn, this has resulted in severe traffic congestion, which, according to an estimate for non-stop flights between Southeast Asia and Europe, is costing the airline industry $100 million annually [25].

The International Air Transport Association (IATA) and the International Civil Aviation Organization (ICAO) have recently adopted plans to change air traffic control systems from ground to satellite based. Together, these organizations represent some 234 airlines that frequently conduct international flights [14]. They introduced a new air traffic control system, dubbed the Future Air Navigation System (FANS), at the Tenth Air Navigational Conference in Montreal in 1991. FANS is currently operational and is being utilized by several major airlines. The system allows pinpoint location accuracy of commercial aircraft flying at any location in the world. FANS is intended to become the predominate global air traffic management system. However, there are still many issues that must be resolved before developed countries adopt the Future Air Navigation System and the current implementation is primarily used in the South-Pacific on a trial basis.

FANS utilizes the U.S. based Global Position System (GPS) and the Russian Global Navigation System (Glonass) to track the location of aircraft. Both the United States and Russian governments have pledged the free use of these systems to aid in the development of an international air traffic control system [43]. FANS consists of three main elements including a satellite navigation system, a datalink communication network between the aircraft and ground, and better air traffic management provisions for the controllers. The Future Air Navigation System provides a convenient solution for the inadequate air traffic management systems in Asia. Traditional radar tracking methods require the installation of multiple radar sites throughout the controlled airspace region. This often leads to high maintenance costs, which are generally offset by taxing the airspace users. However, there are many air traffic regions in Asia that are not easily accessible by traditional ground based radar installations. There are also concerns regarding which countries will allow international traffic to fly over their domestic airspace. These issues have made it difficult to fund and implement a centralized air traffic management system in Asia.

The Future Air Navigation System has proven to be successful in reducing traffic congestion over many parts of Asia and many countries intend to adopt the system as their primary air traffic control system. Traditional ground based systems generally separate aircraft by 80 nautical miles to ensure safety. Due to the improved aircraft location accuracy of FANS, this separation can be reduced to 30 nautical miles or less [2]. This effectively increases the number of aircraft that can simultaneously fly over regions that lack ground based radar surveillance equipment and can even increase the traffic load in radar controlled airspace regions. For example, the Bay of Bengal was traditionally a heavily congested region for flights travelling between Europe and Asia. Since the implementation of FANS, this region has become much less congested and several airlines have noted greater profitability due to more efficient routing, less en route holding, less departure delays, and increased fuel efficiency [25].

India, Malaysia, North and South Korea, The Philippines, and Singapore have all pledged to adopt their air traffic control systems to utilize the Future Air Navigation System. This is an important step toward the development of an integrated traffic management system in Asia. Coordination is necessary in regions where there are many smaller countries that each have their own domestic traffic control system. Recall that the United States utilizes a three-level hierarchical traffic management command structure. This approach provides coordinated decision making between the traffic managers at various regional sites. Given the number of smaller countries in Asia (in terms of geographic size), cooperation between different nations will help resolve traffic congestion issues. One of the recent milestones in better cooperation between nations was emphasized by the agreement between North Korea and South Korea to allow one another to use each other's air space for civilian aviation [24]. Such agreements will allow flights to travel more directly to their destination decreasing flight time as well as lowering airspace demand.

Two new airports were recently opened in Asia that demonstrate the latest technologies in airport management systems. Malaysia's Kuala Lumpur airport and Hong Kong's Chek Lap

Kok airport both utilize advanced systems developed by companies in the United States. These systems attempt to integrate various data and information based components into a centralized control architecture. For example, Kuala Lumpur International Airport (KLIA) utilizes a sophisticated KBMS, dubbed the Total Airport Management System (TAMS), that cost $167 million to complete [3]. This system integrates 29 airport sub-systems including flight information display, gate collection, card access, meteorological data, air traffic management, and baggage handling to name a few. Dr. Ling Lieng Sik, Malaysia's Transport Minister, explains that "various functions need to be integrated to enable the airport to operate. For instance, the ticketing system goes hand in hand with, among others, the baggage handling and flight information systems." [42]

In comparison, Chek Lap Kok's $71 million dollar system integrates a myriad of airport sub-systems to a central airport operational database, which disperses information submitted by the various components to one another. The interface to this system, called the Flight Information Display System (FIDS), allows passengers, airlines, maintenance crews, luggage crews, pilots, and all other pertinent people to view and update the information contained in the system. FIDS facilitates airport management by automating many of the tasks normally carried out by humans. For example, FIDS is capable of automatically assigning arrival gates based on the most efficient configuration to keep the building heated or cooled [3]. In addition, the arrival gates are assigned by considering those passengers who have connecting flights. FIDS provides a myriad of other decision support tools that help provide efficient airport operation.

The complexity of these systems makes it crucial for each component to operate correctly. The level of integration and lack of system redundancy make it possible for single point failures to cause drastic side effects. Indeed, both Kuala Lumpur and Chek Lap Kok experienced major system failures shortly after opening. Kuala Lumpur's Total Airport Management System crashed on the airport's opening day after inexperienced check-in agents entered incorrect data to the system. The system was overloaded when the check-in agents continued to mistype ticket information when they could not figure out how to produce boarding passes [3]. Since the system was also in charge of automatically carrying out other various airport management tasks, Kuala Lumpur was effectively shut down while people tried to remedy the situation. Boarding passes eventually had to be issued by hand while others kept track of flights and arrival gates by using paper board. It took several days before the system was operational again and many businesses were affected due to the disruption to flight services.

Similar problems occurred at Chek Lap Kok when air traffic managers entered incorrect flight arrival times. The Flight Information Display System then dispersed this false information to all other subsystems. The result was complete confusion. People were directed to board their flight at the wrong departure gate while other arriving aircraft were mistaken for different flights. The baggage crews were also given incorrect information as to where luggage should be placed and thus, many flights departed without their luggage. However, these problems do not necessarily represent an inherent fault in FIDS. The system operated as specified and the majority of the blame was focused on the individual who incorrectly entered the arrival flights data. Nonetheless, important lessons can be learned from both of these system problems.

The problems encountered at Kuala Lumpur appear to be inherently embedded in the system design. The Total Airport Management System (TAMS) consists of a large Distributed Database Management System. There are over 20 different airport operation systems that all connect to TAMS. In fact, the airport contains 434 miles of fiber-optic cable that was installed by the U.S. networking company Harris Corporation [3]. Recall that one of the major limiting factors of a DDBMS is the delay associated with communications between modules. Thus, performance is often lowered at times of high network traffic. For example, the CIRRUS primary switching node is only capable of handling 2 transactions per second, which effectively limits the number of customers that can simultaneously use the system. TAMS has similar constraints and heavy network communication activity could likely have resulted in the system failure on opening day.

TAMS' communication network was likely overloaded when the check-in agents could not figure out how to issue boarding passes. The agents panicked and, instead of asking for help, continued to enter incorrect information into the system. This resulted in an unexpected network load that eventually paralyzed the entire system [3]. Apparently, TAMS failed to meet one of the primary design goals of a Distributed Database Management System. Recall that DDBMSs should provide concurrency transparency. This means that the users of a distributed system should not be affected by other system users. However, this obviously was not the case with TAMS. The check-in agents prevented everyone else from using the system when they created too much demand on the communications network.

The problems at Chek Lap Kok are somewhat different from those encountered at Kuala Lumpur. Although both failures were mainly caused by human error, Chek Lap Kok's Flight Information Database System continued to perform its duties as specified. However, the system did fail to recognize that an air traffic manager entered incorrect information into the system. Philip Bruce, public-relations manager for the airport, defended FIDS' actions by emphasizing the system can only be so smart, "If someone keys in the flight is coming at 11 and it's [actually] coming in at 12, that's not a system failure." [3]

Nevertheless, there is still an important message to take away from the problems encountered with FIDS. Humans are prone to make errors and mission critical systems should take this fact into consideration. Recall that Database Management Systems often verify the legitimacy of incoming data before storing it to the database. This helps to ensure that the database does not become corrupted. Knowledge Base Management Systems should also provide a certain level of error checking such that serious problems can be avoided. Evidently, FIDS did not provided enough oversight to ensure that the flight arrival times were entered correctly.

2.5.2 European Traffic Management

Traffic management in Europe mirrors what Asia's future air traffic management system will be. Europe is another area where there are many smaller countries that must coordinate their air traffic management efforts. The European nations decided to establish Eurocontrol in 1960 to resolve concerns over the management of the airspace above its 20-member states [23]. Eurocontrol is based in Brussels and, until recently, was responsible for flights traveling over Belgium, the Netherlands, Luxembourg and northern Germany. Eurocontrol illustrates the complex issues involved in the implementation of an international air traffic management system. Many similar issues will be faced during the continued development of Asia's integrated control system and it is important to highlight the challenges of developing a global air traffic management system.

Eurocontrol is in the process of creating a centralized control center that will eventually manage flights over all of its member states. The need for centralized traffic management stems from the projected increase in European traffic demand as well as the apparent inefficiencies of air traffic control over Europe. European traffic demand is expected to double between the years 1990 and 2003 [20]. This increase is greater than that of the United States, but similar to the expected increases in Asia. Air traffic in Europe is notorious for having some of the worst flight delays. For example, it is estimated that delays cost airlines between $2 billion and $2.6 billion in 1995 alone. Another estimate reveals that 18 percent of European flights were delayed by more than 15 minutes that same year [23]. These figures are substantially higher than those in the United States.

Many of these delays and congestion problems can be traced back to the systems that are responsible for air traffic control over Europe. There are some 32 different computer systems responsible for aiding the traffic managers at 42 control centers spread across Europe [7]. Communication and coordination between the control centers is not very efficient and this often leads to poor traffic management between the various European countries. Mr. Karl-Heinz Neumeister, the secretary-general of the Association of European Airlines, explains that "these centers operate with different levels of performance and technology, with no commonly agreed standards, and a lack of compatibility between the different systems." [7]

A study of air traffic control centers found that Europe spent nearly as much as the United States in air traffic control equipment and maintenance costs, but controlled less than a third of the flights in 1989. These figures prompted Eurocontrol to begin development of a centralized air traffic management system based in Brussels. A centralized flight database was one of the primary goals for implementing the original system [12]. Several years later, the Central Flow Management Unit (CFMU) now handles all the arrival and departure slots for airports in Western Europe [11]. The CFMU collects the flight schedule plans from each airport under its control. These flight plans are then used to determine the optimal order of aircraft departures and each flight is assigned a departure slot based on its proposed flight plan.

Since the inception of the CFMU, traffic delays have slowly improved over Europe. Yves Lambert, the Eurocontrol director general, reports that "only 7.4 percent of European flights were delayed for longer than 15 minutes in 1993 as opposed to 12.1 percent in 1992." [11]

Unfortunately, the CFMU has not been able to keep up with the increase in traffic demand over Europe. Recall that 18 percent of flights were delayed more than 15 minutes in 1995. Apparently, Europe must develop better traffic management systems to handle the expected increase in traffic demand over the next several years.

Chapter 3

3 SYSTEM ARCHITECTURE

The Enhanced Traffic Management System consists of two primary components. These components serve two distinct roles in facilitating the control of the National Airspace System. The first component consists of the distributed database that contains all the important data concerning the status of the National Airspace System and is responsible for relaying these data to and from sites throughout the United States. The second major component of the ETMS is the Traffic Situation Display (TSD). This graphical interface provides several tools that allow the traffic managers to view and update information contained in the ETMS. Each component will be studied in terms of the data and/or information management techniques that they utilize. This chapter will reveal that much of the ETMS is concerned with data management capabilities. Indeed, the need for real-time data access influences many of the design criteria. However, there are also several system applications that utilize knowledge management tools. These system components automate traffic management tasks at various levels within the ETMS architecture. The database management components will be introduced first, as they are the backbone of the ETMS. The knowledge management components are built on top of the database system and will be discussed after the primary ETMS architecture has been introduced.

3.1 Distributed Data Management

The ETMS utilizes a distributed database architecture that is somewhat similar to the CIRRUS banking network. Flight traffic management is another task that is inherently distributed in its nature. Airspace is divided into many different sections that are monitored by particular air traffic control centers. These centers are located throughout the United States and are often defined by geographic location based on their respective airport facilities. The processing of all data essential to the national airspace system is done at the Volpe Transportation Systems Center in Cambridge, Massachusetts. Each control center has a dedicated communications link to the FAA Tech Center located in Herndon, VA. The Tech Center facilitates data requests and updates from the Air Route Traffic Control Centers (ARTCCs) to the ETMS processors located at the Volpe Center. Thus, the Tech Center has a similar role to that of the primary switching node in the CIRRUS network. The main architectural difference is that the centralized database management is carried out at the Volpe Center instead of at the FAA Tech Center, which acts as the switching node for the various ARTCCs. The communications backbone of the ETMS is illustrated in Figure 3-1.

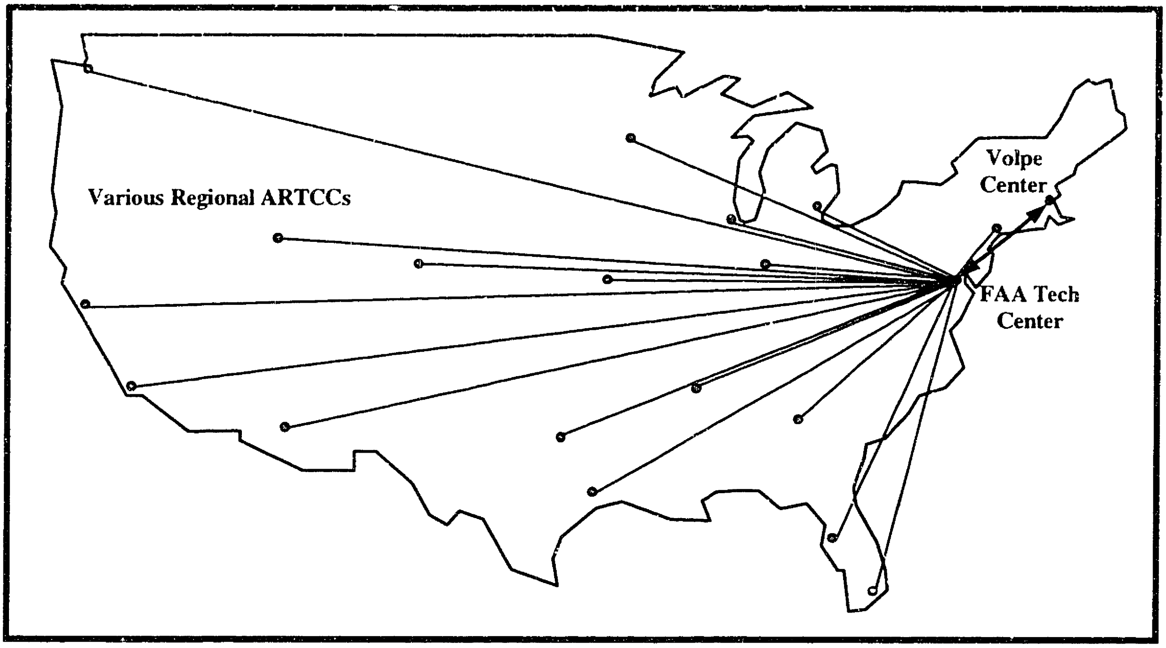


Figure 3-1: ETMS Communications Network

Recall that the regional ARTCCs act as the middle level traffic managers. They are responsible for controlling congestion problems within the scope of the ARTCC (defined in terms of geographic boundaries). The FAA Tech Center is located adjacent to the Air Traffic Control System Command Center. Recall that the ATCSCC is the top-level control center and coordinates the actions taken by the distributed ARTCC sites. Thus, the communications backbone of the Enhanced Traffic Management System follows the same hierarchical organization as the air traffic management hierarchy. This organizational approach also plays a significant role in the implementation of the ETMS's distributed database management system.

The Volpe Center acts as the centralized data control processor to ensure data consistency and integrity. The various ARTCCs access this database and create local copies to ensure efficient data access. The Volpe Center must keep track of these distributed copies to ensure that consistency is maintained. The FAA Tech Center can also place restrictions on data access and/or updates such that the ATCSCC maintains ultimate control over the Enhanced Traffic Management System. These control facilities illustrate the macro view of the ETMS distributed database architecture. There are also several database subsystems that are responsible for controlling certain data structures within the ETMS database. These subsystems represent lower level control facilities and are the components that carry out a majority of the data management within the ETMS. The discussion of these subsystems will provide a foundation for the subsequent analysis of the knowledge management tools implemented in the ETMS.

Recall that the Enhanced Traffic Management System is concerned with various data regarding the status of the National Airspace System. These data are divided into three main functional categories including:

- Maps data that include geographical information such as state boundaries, sector boundaries, jet routes, airports, control centers, and many other important geographical locations.

- Flights data that include the current en route airplanes, planned airplane departures and arrivals, airplane flight models, and other information that is critical to the status of flight congestion.

- Weather data that illustrate the current weather conditions such as precipitation, radar tops, lightning strikes, jet streams, and other relevant environmental conditions throughout the United States.

Together, these database structures capture the best possible picture of the overall traffic status in the U.S. and surrounding areas. These database structures are controlled by distinct processing facilities that will be explained in the following section. Each data structure also has a number of distinct data elements. These elements contain a vast amount of information that can be relayed to the traffic managers to aid in the efficient control of air traffic entering or leaving the U.S. Each of the major functional data structures will now be enumerated in more detail.

3.1.1 Maps Database

The maps database is primarily concerned with geographical information pertaining to aircraft navigation. This database includes geographic information from sources throughout the world. Most of the geographic data are concentrated in the United States. However, the ETMS contains a variety of other data due to its inherent role in controlling air traffic throughout the world. Flights often arrive in the U.S. from countries abroad and thus, it is important to have global geographic information to aid in managing international air traffic.

- Geographical boundaries including all country borders, state boundaries in the U.S. and provincial boundaries in Canada, boundaries for ARTCC control areas, boundaries for air control sectors within the ARTCCs, boundaries for international and oceanic control sectors, boundaries for special use airways (i.e. military and restricted areas), and boundaries for airspace arrival, departure and holding fixes.

- International airport locations for roughly 20,000 airports including the airport name, identifier, runway information, native state or country, latitude and longitude position, nearest city, native ARTCC (if in the U.S.), and the data source.

- International sector and fixes data including the identifier, position, native state or country, latitude and longitude, native ARTCC (if in the U.S.), type of fix or sector, data source, and a description of the element.

- International airway data including preferred routes, jets, victors, oceanic routes, the element identifier, latitude and longitude, data source, and description.

3.1.2 Flights Database

The flights database is perhaps the most important data structure of the Enhanced Traffic Management System. These data are used to display the live flight positions to the traffic managers. The flights database contains additional information including flight id, type of aircraft, current altitude and ground speed, origin, destination, estimated time to arrival, and a vast array of other information regarding the status of all en route flights. A detailed list of the flights database structure is included in Appendix A.1. There are several important points to emphasize about this database structure. Specifically, the data elements that it contains provide a foundation for higher-level knowledge management:

- Flight events provide a list of chronological geographic boundary crossings such that a detailed flight route is created. This includes airport origin and destination, ARTCCs crossed, sectors and fixes traversed, jetways and victors followed, and any other defining events throughout the flight path.

- Aircraft classification presents several measures to distinguish between different airspace uses. For example, flight category use (commercial, general aviation, military, etc.), general classification of aircraft (helicopter, jet, single piston prop, etc.), flight identification (call sign, country origin, model type, air carrier, etc.), and additional categorization for internal modeling.

- General status information regarding the progress of the aircraft's flight plan. This includes the airplane's estimated and actual departure time, estimated and actual latitude/longitude position, estimated and actual ground speed, requested ground speed and altitude, estimated and actual altitude, distance traveled, last flight event recorded, and a myriad of other data regarding flight status.

Together, these flight data provide essential information that aids traffic managers in prioritizing and redirecting air traffic during times of congestion. The flight events list provides the traffic managers with a detailed flight map for each active and proposed flight. The traffic managers utilize this information when redirecting traffic away from areas of congestion; they try to minimize any deviation from the original planned air route. The traffic managers also utilize the aircraft classification information to decide which flights should have priority and which fights can be redirected. The flight status data portray the current level of congestion throughout the national airspace system and help identify those areas where extended delays have occurred. The traffic managers utilize this information to improve the overall throughput of the NAS.

3.1.3 Weather Database

Perhaps one of the most unpredictable elements in managing traffic flow is the congestion caused by adverse weather conditions. In extreme conditions, weather can cause airports to shut down completely for extended periods of time. This obviously causes delays for flights scheduled to arrive or depart from closed airports, but it also causes delays for many other flights that travel through nearby areas. Inclement weather often causes a snowballing effect that backs up traffic throughout the entire National Airspace System. Thus, weather data are essential elements in the ETMS database. Traffic managers have access to a variety of weather data including:

- Terminal Forecasts and Surface Observations. These data are relayed to the traffic managers in a readable format and includes cloud cover, visibility, wind speed and direction, precipitation, and temperature. Recall that this information is often compiled by automated weather observation systems thus speeding the dissemination of airport meteorological conditions.

- Live radar weather data that include precipitation intensity, height, direction, and speed. The traffic managers can also access lighting ground strike information, jet stream patterns and speeds, and historical weather information. The managers use this information to make reroute decisions based on where and when inclement weather is expected to cause delays in flight paths.

- Grid winds reports data are also stored by the ETMS. This information is not available to the traffic managers, but is essential for calculating expected aircraft speeds throughout areas in the NAS. Jet steams can often speed up or slow down high flying jet aircraft thus making a significant impact on overall flight time.

The weather data compiled by the ETMS are used primarily by the traffic managers to predict possible areas of traffic delay. However, these data are now being used by automated processes to aid in the prediction of traffic flow. The grid winds are an example of a data structure that is utilized by a knowledge management component. This information is utilized when calculating the estimated time to arrival when taking into account the flight path of an aircraft. These calculations are done by dedicated data processors and this will be the topic of the next section.

3.1.4 ETMS Central Processor Functions

The Enhanced Traffic Management System utilizes a distributed data processing architecture. There are three primary processor components in this architecture and each contain lower level data management facilities that manipulate the data structures outlined in the previous section. These primary components consist of the ETMS Central Functions, the ETMS Auxiliary Support Functions, and the ETMS Field-site Functions. Both the Central and Support Functions are located at the Volpe Center while the Field-site Functions are distributed throughout the regional field sites. Each processor function plays a critical role in the maintenance and distribution of the maps, flights, and weather database structures.

The ETMS Central Functions control the majority of the distributed database structures. This design provides centralized control of important data and speeds the processing of incoming and outgoing data. Figure 3-2 illustrates the major sub-components of the Central Processor Functions. The main role of the Central Functions is to model the current and future status of air traffic travelling through the National Airspace System. Data are received frequently from external and internal sources by the Communication Links. These data then get relayed to the proper processing facility where they are analyzed to determine what affect they will have on the overall state of the NAS. The Traffic Modeling Functions are responsible for predicting the flight path of every scheduled and active flight. This information is utilized to determine present and future traffic congestion conditions, which are then relayed to the traffic managers.

There are several important points to illustrate in Figure 3-2. The External Communications Link handles all data messages that travel beyond the scope of the ETMS. This information includes incoming weather data, airline flight schedules and changes, active flight departures, arrivals, positions, and fuel advisories. The ETMS also sends data messages to the airlines and weather centers via the External Communications Link. These messages include requests for weather forecasts and replies for flight substitution requests. The Internal Communications Link provides a data connection from the central site to all the field sites (ATCSCC, ARTCCs, and other traffic management facilities throughout the world). This data link ensures that each copy of a data structure is consistent and also provides a protocol for data requests or updates.

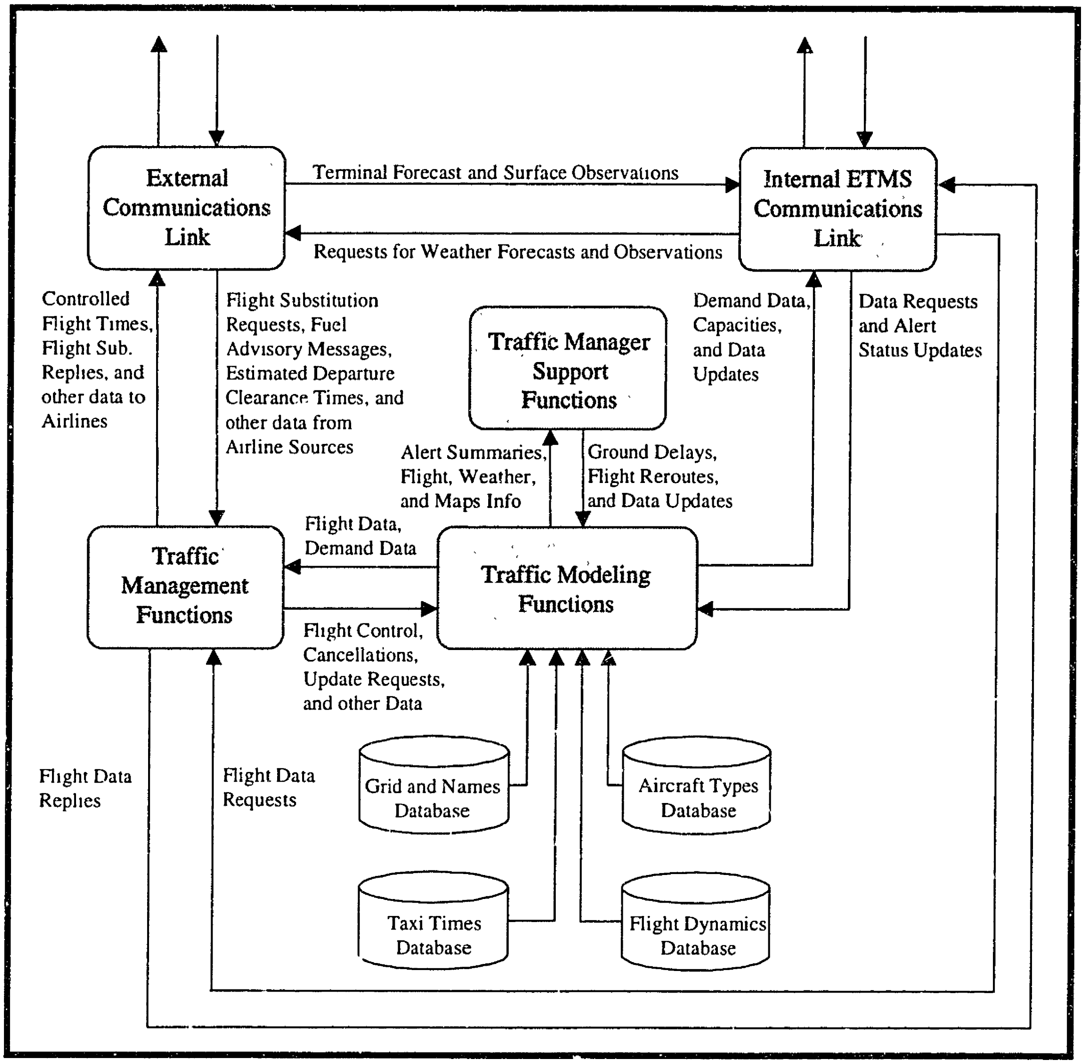


Figure 3-2: Diagram of ETMS Central Processor Functions[3]

The Traffic Management Functions control the actual departure time for each airplane scheduled to depart. This control is generally done at the highest level and any deviation from the planned departure time must generally be authorized by the ATCSCC. Scheduled departure times can be delayed by ground delay programs. Ground delay programs are used during extreme traffic congestion periods to alleviate overcrowded airways. These delayed flights can be replaced by other flights if an airline wishes to do so. Thus, controlled departure

---

[3] Adapted from [15].

times are relayed to the airlines to inform them of any schedule delays. The airlines will often reply with schedule substitution requests, which are then fed into the Traffic Management Functions. These functions then approve the airline's request if they meet certain criterion outlined by the FAA. Flight schedules are changed frequently and these changes must be relayed to the other system components to ensure that the ETMS continuously provides the most accurate flight information.

The Traffic Modeling Functions utilize the flights schedule data and the active flights data to generate the information necessary for the flights database structure. The complete database structure is listed in Tables A-1 through A-4 of the Appendix. Note that the Traffic Modeling Functions utilize several additional database structures to calculate the corresponding data in the flights database structure. These databases include the Aircraft Types, Flight Dynamics, Taxi Times, Grid and Names Databases. These data provide a semantic structure for higher-level knowledge management. The knowledge management components of the Traffic Modeling Functions will be referred to in a later section, but it is important to introduce the database structures that are utilized by the knowledge base.

Flight schedule information is combined with aircraft type and route information to calculate accurate travel times and location points. The Grid and Names Database contains an exhaustive list of every possible geographic location and the corresponding name that might be included in a flight plan. The Taxi Times Database includes estimated aircraft taxi times to calculate the predicted time to takeoff after departing from a given departure gate. The Aircraft Types Database includes a standardized classification for every aircraft that might come under the control of the FAA. The Flight Dynamics Database provides a corresponding aircraft ascent and descent profile for each aircraft type classified by the FAA. These dynamics provide a flight altitude profile for each scheduled flight that aids in projecting traffic demand throughout the NAS.

This information is then relayed to the Traffic Management Support Functions that provide an interface to the ETMS Data for the Traffic Management Specialists. The majority of the Traffic Management Support Functions are located at the field sites and are controlled by the Field-site Functions. However, the Volpe Center contains these same facilities to support the development and enhancement of the various support functions. The primary component of the Traffic Manager Support Functions is the Traffic Situation Display. This interface provides many knowledge management tools that aid traffic managers in controlling the National Airspace System. Section 3.2 provides a detailed discussion of the TSD.

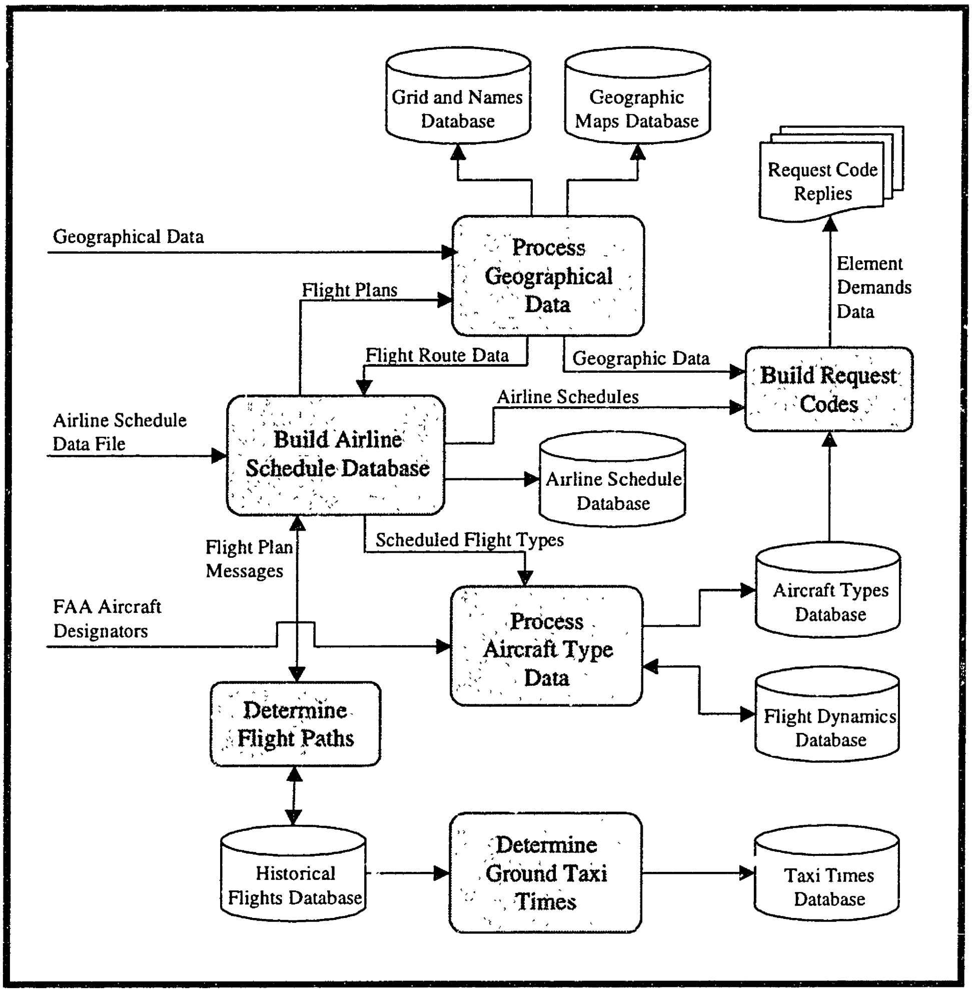


Figure 3-3: Diagram of ETMS Auxiliary Processor Functions[4]

3.1.5 ETMS Auxiliary Support Processor Functions

The ETMS Auxiliary Support Functions provide routine data maintenance utilities. They update various data structures when information is submitted from external sources. They also service certain data requests and store historic flight information for later analysis. Figure 3-3 illustrates the main components in the ETMS Auxiliary Functions. Note that several of the databases manipulated by the Auxiliary Functions are also used by the Central Functions.

---

[4] Ibid.

The Auxiliary Functions generally update these databases whereas the Central Functions tend to read information from them.

The Auxiliary Processes receive data from the National Flight Data Center, Adaptation Controlled Environmental System Center, and the National Oceanic Service once every 56 days. These centers provide updated geographic information that must be relayed to the Maps, Grid and Names Databases. The Process Geographical Data application is responsible for verifying the integrity of the data received by the centers. After checking for errors, the new geographic data are then written to aforementioned databases. The updated information is then transmitted to the field sites that have local copies of geographic data.

The Auxiliary Functions also receive airline schedule information from the Official Airline Guide. The airline schedules are passed to the Build Airline Schedule routine that relays this information to several other processes. The Process Geographical Data application receives airport information that might not be included in the data files that are received every 56 days. The Determine Flight Paths stores the schedule information in the Historical Flights Database for approximately one week. The historic information is useful for estimating airport ground taxi times based on previous flight demands. The historic database is also useful for determining past flight performances between various airports throughout the NAS. After distributing the necessary information, the Build Airline Schedule routine updates the Airline Schedule Database so that it reflects the most recent scheduled flight departures and arrivals in the next 12 hours.

The Process Aircraft Type function receives data from the FAA regarding airplane classifications. The Process Aircraft Type matches an aircraft dynamic profile for each type of aircraft that is classified by the FAA. However, if the Official Airline Guide contains an aircraft type that is not included in the current database, the Build Airline Schedule routine must notify the Process Aircraft Type of a new aircraft classification. This classification information will be included in the data received by the Official Airline Guide.

The Build Request Codes service receives data requests from the traffic managers. These requests may include demand information for particular map elements, flight schedule data, aircraft classification information, and/or historic flight data. The Build Request Codes service provides the traffic managers with information that is not included in the field site databases. This information is usually not run-time essential and thus, the traffic managers must make requests for this data to the ETMS via the Traffic Situation Display.

3.1.6 ETMS Field-Site Processor Functions

The ETMS Field-Site Functions are responsible for maintaining the data needed by the Traffic Situation Display. They are also responsible for the communications between the various field sites and the FAA Tech Center. Recall that all data communications to the Volpe Center are relayed by the FAA Tech Center. The field sites often receive updated data from the central site. The traffic managers may also make requests for data that are not directly available at the field site. These requests must be relayed to the central site where the Build Request Codes service replies to the field site with the necessary information. Consequently, many of the Field-Site Functions are concerned with data communication with the central site. Figure 3-4 illustrates the primary components at the various field sites.

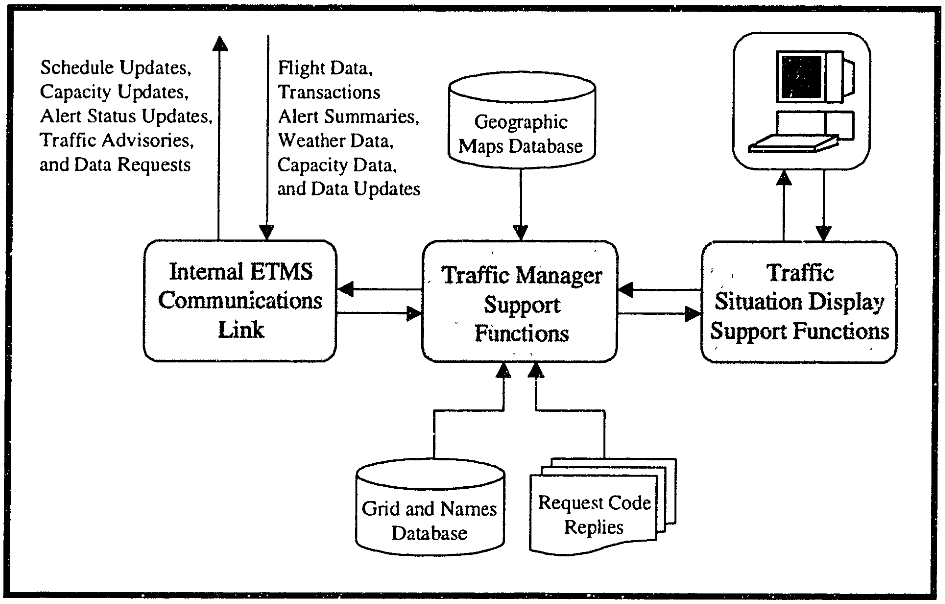


Figure 3-4: Diagram of ETMS Field-Site Processor Functions[5]

Note that each field site has a duplicate copy of the Geographic Maps, Grid and Names Databases. The Traffic Situation Display must have quick access to these databases in order to

---

[5] Ibid.

provide a real-time interactive display. These databases tend to be quite large and the local copies prevent the communications network from becoming crowded with requests for Map Overlay information from the TSD. These local database copies are updated by the Auxiliary Functions every 56 days or when any updates are made at the central site. The Traffic Manager Support Functions determine if data requests can be satisfied by retrieving information from the local database copies and, if necessary, send requests to the central cite via the Internal ETMS Communications Link.

3.1.7 Summary of DDBMS Components in the ETMS

The ETMS contains many distributed database management components that serve distinct roles in manipulating and transmitting data throughout the system. The Central Site Functions and the Auxiliary Support Functions are responsible for maintaining the master copies of all data structures and process all incoming data. This information is then relayed to the Field-Site Functions that make a local copy of the most frequently used data. The overall DDBMS architecture of the ETMS ensures optimal performance, reliability, transparency and flexibility throughout the system.

The Internal Communications Link provides system transparency to the end users. The traffic managers receive updated and real-time data from the central site that is stored on a local machine. Thus, the traffic managers receive the same level of service that they would receive at any other regional sites or even at the central site. The Field-Site Functions and Central Functions ensure that each database is consistent and thus, the ETMS provides replication transparency for the traffic managers. Migration transparency is provided by the FAA Tech Center, which is responsible for all communications between the regional sites and the central site. It would be possible to move the central site without affecting database communications.

The ETMS also ensures a high level of reliability. Redundant copies of each data structure are made and each critical node has a backup system. The Internal Communications Link has multiple dedicated lines to each site and satellite links can be used should ground communications fail. These additional communication links can also be used in times of heavy message traffic, thus providing a higher level of system performance. There are also redundant nodes that can take over if a critical node were to fail. Together, these features ensure that the ETMS will continue running even with multiple component failures. System modularity adds flexibility to the ETMS as well. Node machines can be upgraded without any loss in system availability. The division of data functions also makes it easy to update software components in the ETMS. In fact, many additional system functions have been added to the ETMS during its development. The next section will introduce the Traffic Situation Display and some of the newer knowledge management tools that aid in efficient air traffic control.

3.2 Traffic Situation Display

The Traffic Situation Display provides a graphical interface to the various data contained in the Enhanced Traffic Management System. The TSD also provides several tools that aid traffic managers in controlling the traffic flow throughout the National Airspace System. Figure 3-5 demonstrates several features of the TSD. The zoom level and location has been set such that the TSD displays the contiguous United States and nearby areas. Note that the TSD has also been set to display all large en route jets as small dot icons (a small subset of all flights). There are also three flights that have been set to show block data and thus their icons resemble jets instead of dots. The state boundaries, major airports, and ARTCCs are also shown in this display setting. For example, the Minneapolis-St. Paul (MSP) international airport is visible near the Minnesota-Wisconsin border. Recall that ARTCCs are responsible for controlling large portions of airspace over their respective region. For instance, the ZMP ARTCC is depicted as a label and border controlling the airspace over parts of North and South Dakota, Nebraska, Iowa, Wisconsin, Michigan, and all of Minnesota.

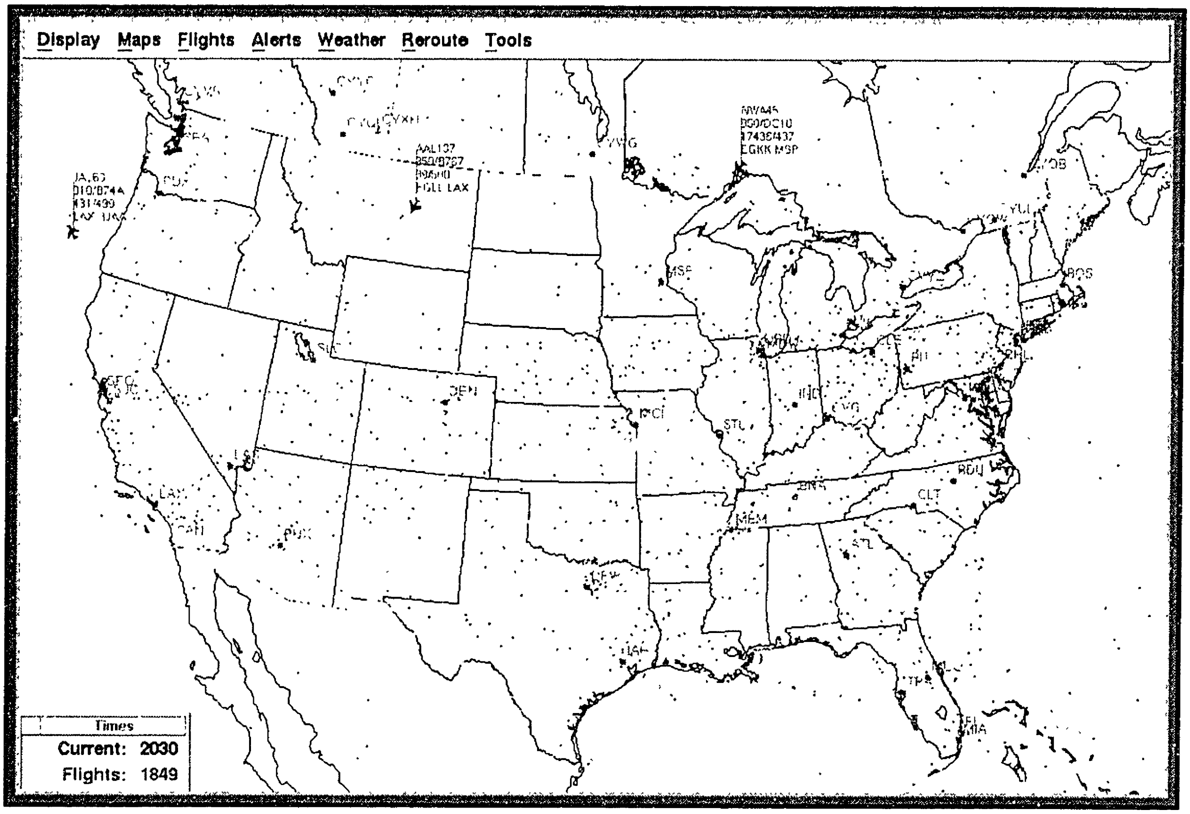


Figure 3-5: Graphical Depiction of the TSD[6]

Flight block data shows flight id, aircraft altitude, type, speed, departure airport, and arrival airport. For example, American Airlines Flight 137 (AAL137) is a Boeing 767 traveling at 350 hundred feet with a ground speed of 500 knots heading to Los Angeles (LAX) international airport from London's Heathrow Airport (EGLL). The TSD can also draw the flight route to depict the exact path that the airplane will follow. The graphical depiction of flights traversing the various control sectors helps the traffic managers identify and redirect traffic that is congesting the NAS. This information can be supplemented with traffic demands data for any given controlled airspace or airport. The traffic demands data and the alerts database are the quintessential information elements provided to the traffic managers by the ETMS. These information databases utilize knowledge management tools to derive traffic congestion levels throughout the National Airspace System.

---

[6] TSD Build 6, Version 5.10 at the Volpe Transportation Systems Center.

3.2.1 Traffic Modeling Functions

The Traffic Modeling Functions are a component of the Central Processing Functions. These functions utilize several databases to project flight traffic over a 12-hour horizon. The Traffic Modeling Functions utilize active flight and scheduled flight data to calculate the traffic demand at every sector, fix, and airport throughout the United States. The active flights portray the current level of traffic congestion and thus present situations that must be dealt with immediately. However, the overall goal of air traffic management is to minimize traffic congestion throughout the NAS. This would be impossible without the ability to project air traffic flow into the future. The Traffic Modeling Functions are capable of projecting traffic demand by utilizing several low-level databases. These include the Aircraft Types, Aircraft Dynamics, Historic Flight, Jet Streams, Grid and Names databases.

The Aircraft Types database provides detailed information about the nominal flight characteristics of a given airplane. This includes cruising altitude, maximum cruising speed, flying range, and other general aircraft characteristics. The Aircraft Dynamics database provides a flight model for typical ascent and descent patterns for the given aircraft type. This information is essential for determining the sectors that the flight will travel through from takeoff to landing. The Historic Flight database provides best estimates for the routes that will be followed between certain airports. Recall that the flight plans generally contain this information, but it is often the case that different routes will be utilized during various times throughout the year. Together, these databases provide a general estimate as to what path a given flight will follow and the position of the airplane throughout its flight.

The Traffic Modeling Functions also utilize the Jet Streams database to provide a better estimate of the overall flight performance for a given route. The Grid and Names database contains detailed definitions of every airspace location that a flight might traverse. The Grid and Names database provides a fine-grained definition for the entire NAS. Each grid is then analyzed to determine what effect that the jet stream will have on a flight traveling through a particular grid. Thus, the addition of jet steams conditions provides a more accurate projection of where a flight will be at any given moment.

3.2.2 Traffic Demands Functions

The Traffic Demands Functions utilize the information generated by the Traffic Modeling Functions to determine the traffic load at every sector, fix, and airport in the National Airspace System. If the traffic load is more than a predefined limit, a message is sent to the Alerts Functions. The Alerts Functions then notify the traffic managers of projected traffic crowding so that they can correct any problems. The Traffic Demands Functions also provide the traffic managers with demand information to aid them in redirecting traffic flow to avoid congestion problems. This information is essential because the traffic managers must monitor where additional flight load can be placed during times of congestion. The demands information allows the traffic managers to optimize traffic flow such that the National Airspace System stays below crowded levels throughout the various elements.

3.2.3 Traffic Alerts Functions

The Traffic Alerts Functions provide the traffic managers with a warning system so that they can be forewarned about potential traffic crowding in elements under their control. Figure 3-6 illustrates several different alerts occurring along the northeastern control areas. The TSD is displaying several different types of alerts simultaneously. The red areas illustrate sectors that have currently exceeded traffic capacity levels. For example, sectors ZID82 and ZME63 have currently exceeding traffic capacity. The yellow areas illustrate sectors that are projected to exceed traffic levels over a given time horizon. The traffic managers are able to set their own time window for alerts data (this example illustrates an alerts window of 2 hours). There are a number of sectors that are projected to exceed capacity, including ZME26 and ZME62 over Tennessee, ZAU34 and ZAU46 over Indiana, etc. The green areas portray sectors that traffic managers are working to rectify the projected or active congestion. Sectors ZAU75, ZID89, and ZBW38 are the three sectors where traffic managers are redirecting traffic to avoid any congestion.

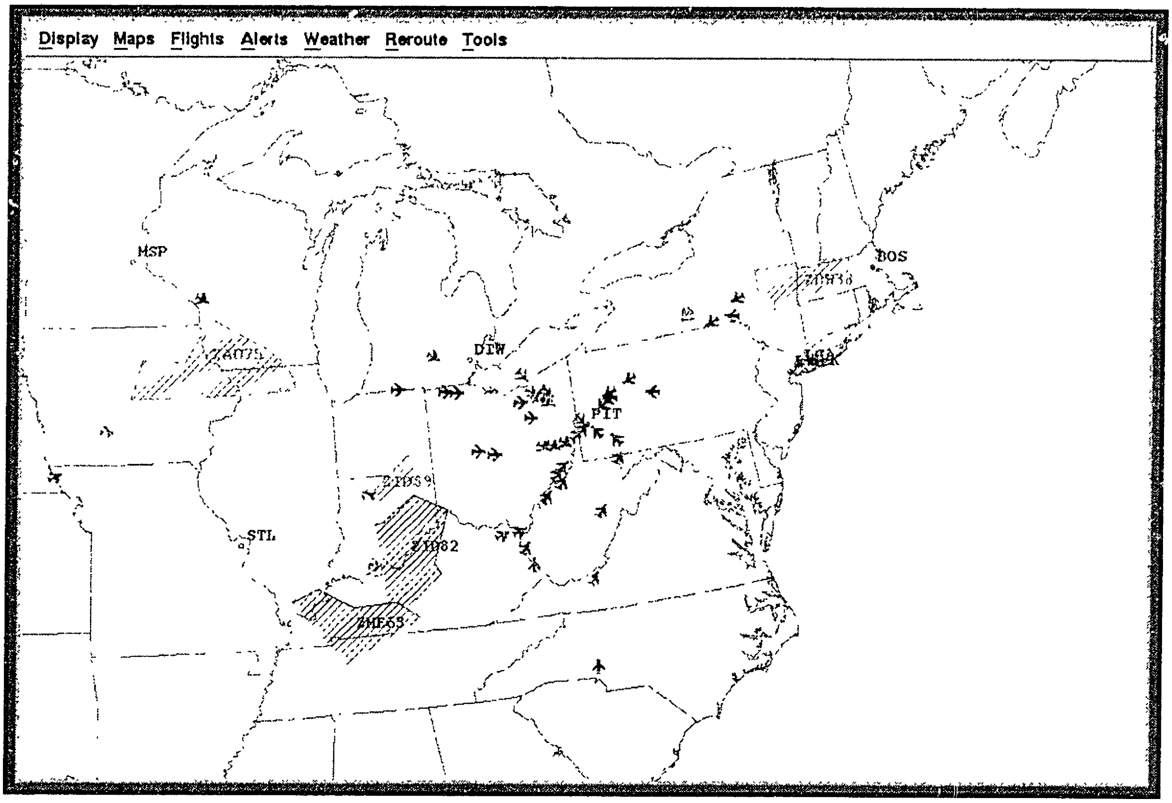


Figure 3-6: TSD Illustrating Alerts Data[7]

Figure 3-6 also illustrates several airports that have or will exceed flight capacity. The same coloring scheme is used to identify congested airports and fixes. For example, MSP, STL, DTW, PIT, LGA, and JFK airports have currently exceeded traffic demand. The yellow airports are projected to exceed demand, including ORD, MDW, CLE, ATL, and CLT to name a few. As one might conclude, the level of traffic congestion along the East Coast is apparently quite high in this example. The Alerts Functions are combined with the Demands Functions through the TSD. Note that this snapshot of the TSD is also displaying the flights arriving at Pittsburgh airport. A traffic manager can access the present and projected demands at this airport through the Alert Tools embedded in the TSD.

Figure 3-7 illustrates the timeline generated by calling the Examine Alerts command in the Alerts menu of the TSD. Note that the alerts timeline displays both the number of aircraft arriving and departing from Pittsburgh airport. The timeline depicts a two-hour window of flight demands starting at 20:45 GMT and ending at 22:45 GMT. In this example, the current time is 20:45 GMT. Flight demands are always calculated for each sector, fix, and airport in 15-minute segments. For example, there are currently 21 flights scheduled to arrive and 18 flights scheduled to depart Pittsburgh airport. Also note that it is the number of active arrivals that has generated the alert status for this airport. The number of active and scheduled departures is within capacity until 21:45 GMT.

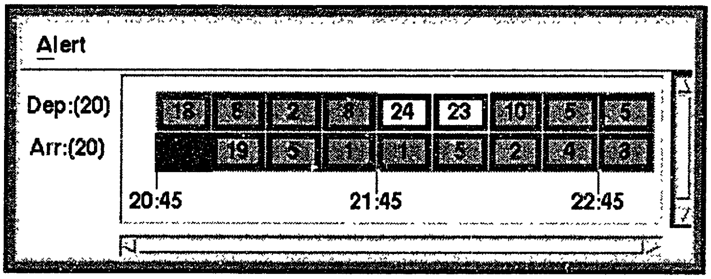


Figure 3-7: Alerts Timeline for Pittsburgh Airport

Traffic managers can also produce a bar chart to view the active and proposed flights at a given airport, sector, or fix. This provides additional information that can be used to reroute flights. Figure 3-8 illustrates the arrival demands at Pittsburgh airport for the same time period as the alerts timeline. Note that the bar chart displays the airport arrivals capacity and the number active and proposed flights. Given the level of traffic congestion, managers can decide to delay certain flights so that the traffic demands are distributed more evenly throughout the 15-minute intervals. However, the traffic managers must also ensure that any delayed flights do not cause additional congestion at other areas in the NAS. The Alerts Functions also provide a tool that lists all flights involved in any given traffic alert. This aids the traffic managers in determining which flights should be rerouted and which flights must maintain schedules. The traffic managers must also consider which flights should have priority over other flights given the aircraft's classification (i.e. military vs. commercial vs. general aviation) and those flights that must maintain a strict departure schedule.

---

[7] Ibid.

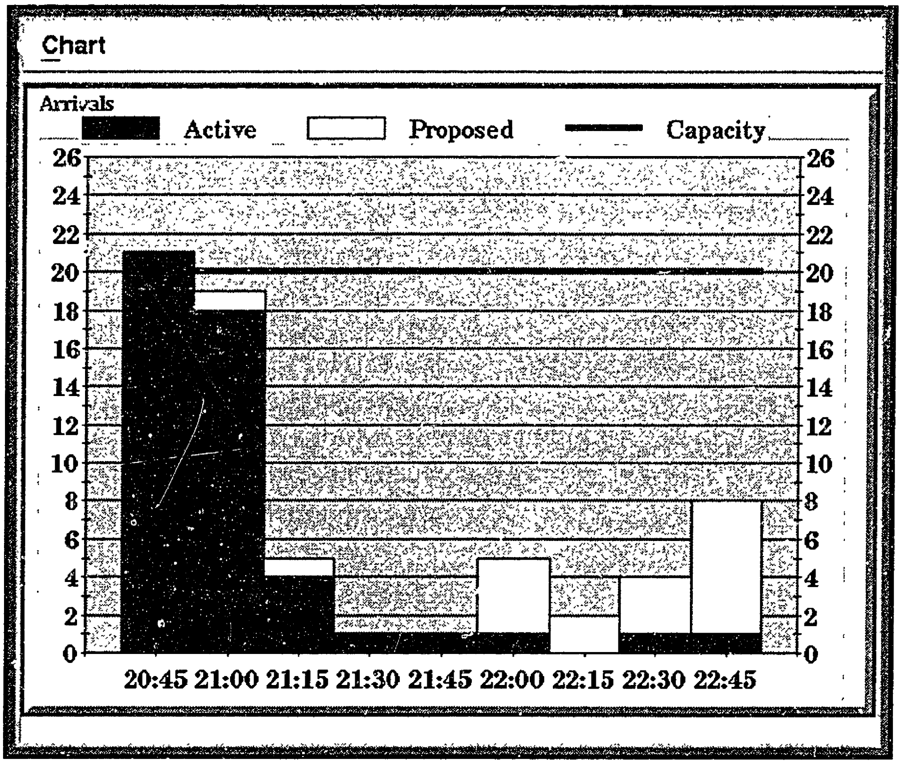
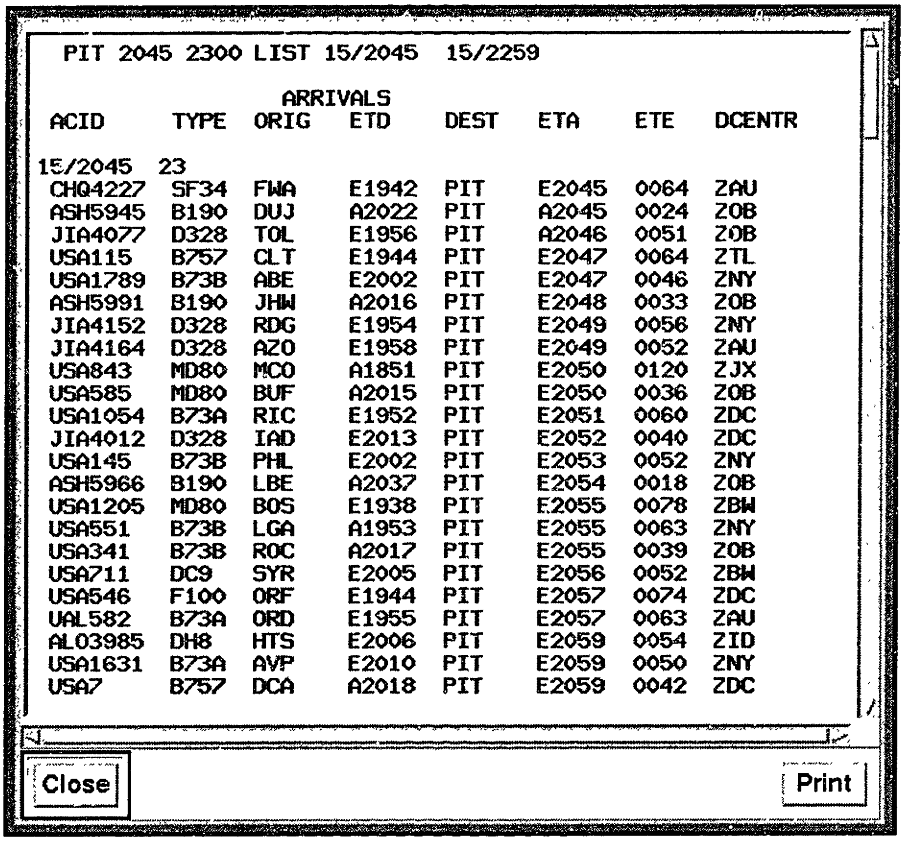


Figure 3-8: Arrival Demands Chart for Pittsburgh Airport

Figure 3-9 illustrates the flight arrivals list for Pittsburgh airport. Note that the flights contained in this list are those arriving in the 15-minute time period that is alerted in Figure 3-7. The alerts list contains the flight id, aircraft type, departure time, destination airport, arrival time, time en route, and the ARTCC from which the aircraft departed. The traffic manager can utilize this information to make additional requests to the flights and/or alerts databases to make decisions regarding flight delays and/or reroutes. This specific example is somewhat unique in that the Terminal Traffic Management sites control the aircraft departing and/or arriving at any given airport including Pittsburgh. Thus, they concentrate on flights in the immediate vicinity of the airport. The Terminal traffic managers do not have much flexibility in rearranging active or proposed flights. These decisions come from a higher management level and tend to involve the ARTCCs or the ATCSCC.



```
 PIT 2045 2300 LIST 15/2045  15/2259

                  ARRIVALS
ACID     TYPE  ORIG  ETD     DEST   ETA     ETE   DCENTR

15/2045  23
CHQ4227  SF34  FWA   E1942   PIT    E2045   0064  ZAU
ASH5945  B190  DUJ   A2022   PIT    A2045   0024  ZOB
JIA4077  D328  TOL   E1956   PIT    A2046   0051  ZOB
USA115   B757  CLT   E1944   PIT    E2047   0064  ZTL
USA1789  B73B  ABE   E2002   PIT    E2047   0046  ZNY
ASH5991  B190  JHW   A2016   PIT    E2048   0033  ZOB
JIA4152  D328  RDG   E1954   PIT    E2049   0056  ZNY
JIA4164  D328  AZO   E1958   PIT    E2049   0052  ZAU
USA843   MD80  MCO   A1851   PIT    E2050   0120  ZJX
USA585   MD80  BUF   A2015   PIT    E2050   0036  ZOB
USA1054  B73A  RIC   E1952   PIT    E2051   0060  ZDC
JIA4012  D328  IAD   E2013   PIT    E2052   0040  ZDC
USA145   B73B  PHL   E2002   PIT    E2053   0052  ZNY
ASH5966  B190  LBE   A2037   PIT    E2054   0018  ZOB
USA1205  MD80  BOS   E1938   PIT    E2055   0078  ZBW
USA551   B73B  LGA   A1953   PIT    E2055   0063  ZNY
USA341   B73B  ROC   A2017   PIT    E2055   0039  ZOB
USA711   DC9   SYR   E2005   PIT    E2056   0052  ZBW
USA546   F100  ORF   E1944   PIT    E2057   0074  ZDC
UAL582   B73A  ORD   E1955   PIT    E2057   0063  ZAU
AL03985  DH8   HTS   E2006   PIT    E2059   0054  ZID
USA1631  B73A  AVP   E2010   PIT    E2059   0050  ZNY
USA7     B757  DCA   A2018   PIT    E2059   0042  ZDC
```

Figure 3-9: Flights Arrival List for Pittsburgh Airport

3.3 Identifying KBMS Components in the ETMS

The ETMS consists of a heterogeneous data and knowledge management architecture. Several database structures and processor functions were introduced in Section 3.1. These components were primarily Database Management Systems. They were concerned with the manipulation of data and the communication of data to various elements in the ETMS. The next level of information management consists of the Traffic Modeling Functions, Traffic Demands Functions, and the Traffic Alerts Functions. These components utilize a semantic knowledge level such that they can be classified as Knowledge-Base Management Systems. They also utilize self-understanding capabilities such that traffic managers can query them to explain their conclusions. For example, traffic managers can investigate any alert notification to determine why a specific element has exceeded traffic demand limits and find which flights are involved in the congestion problems. These KBMS components are embedded in the

Traffic Situation Display such that traffic managers have easy access to the information contained in the Enhanced Traffic Management System.

3.3.1 Knowledge Representation in the ETMS

The Traffic Modeling Functions utilize a knowledge representation to project traffic demands into the future. This knowledge representation is concerned with aircraft dynamics and the airspace that flights travel through. Recall that a knowledge representation is a partial theory of intelligent reasoning that recommends inferences based largely on modeling human expert reasoning. The Traffic Modeling Functions contain many concepts that are modeled after human experts in the field of air traffic control. These functions utilize a semantic structure that reflects how air traffic controllers view the National Airspace System. Namely, the Traffic Modeling Functions utilize the same naming scheme that people use to identify various sectors, fixes, and airports throughout the NAS. Thus, the Traffic Modeling Functions use a knowledge representation that is also a medium of human expression. People can easily extract information from these functions because they utilize a similar representational scheme.

The internal computational semantics of the Traffic Modeling Functions vary somewhat from the normal use of sectors, fixes, airports and other geographic locations. This is necessary for efficient computation. The Grids and Names Database defines sectors, fixes, airports, and all other map elements in terms of the latitudes and longitudes that they span. The Grids Database divides the world into cells that are each 5 minutes longitude by 5 minutes latitude. Note that 1 minute corresponds to 1 nautical mile, thus each cell is 5 nautical miles by 5 nautical miles or approximately 33 square miles in area. Each geographic element is thus defined by the grid cells that it is composed of in additional to the altitudes that it spans. These definitions are then used to determine flight paths and the sectors, fixes, or airports that a flight will travel through.

Figure 3-10 illustrates how the Flight Modeling Functions process flight plan information. For example, a flight plan might consist of the list [MSP...V82...LVN...V82...SYN]. This represents a list of named elements that the flight route will intersect and/or follow. The flight will depart from Minneapolis-St. Paul Airport, follow victor V82 until it reaches LVN airport, then it will rejoin victor V82 until it arrives at SYN airport, its final destination. The Flight Modeling Functions will look up the named elements and determine the grid cells in which they reside. For instance, MSP resides in grid 44°'50'N/93°10'W, as does V82. Thus, the flight processor knows where the flight will start following the victor V82. This victor will be followed until it intersects with LVN at 44°35'N/93°05'W. Then V82 will be followed from LVN to SYN where the flight reaches its final destination at 44°20'N/92°55'W.

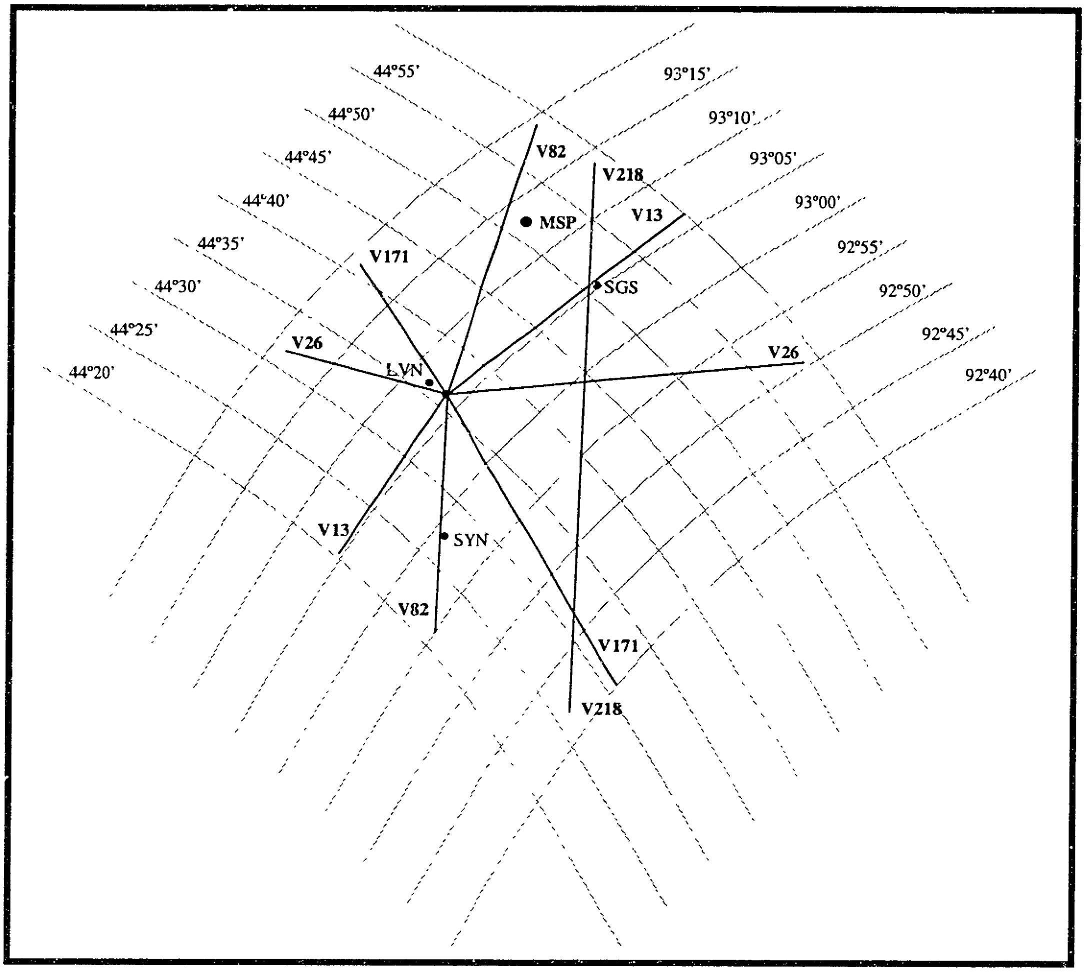


Figure 3-10: Internal Semantic Processing

The Grids Database is also utilized by the Traffic Demands and Alerts Functions. The Traffic Modeling Functions determine the flight path for each scheduled flight. The Traffic Demands Functions use this information in addition to the active flight information to predict the traffic demand at each grid cell. If these cells are located in a sector, fix, or at an airport, the traffic flow is summed to determine the overall demand for a given element. For example, let's introduce a sector ZLVN which spans the grid cells 44°35'N/93°05'W, 44°30'N/93°05'W, 44°30'N/93°00'W, and 44°35'N/93°00'W. In reality, these grid cells are part of a much larger sector ZMP07, which includes hundreds of cells that are not worth listing in this example. In order to calculate the current traffic demand for the sector ZLVN, the Traffic Demands Functions would simple sum the number of aircraft in all four of the aforementioned grid cells. If the number of flights in these grids exceeds a predetermined capacity, then the sector ZLVN would be alerted by the Traffic Alerts Functions.

The calculation of current traffic demands is somewhat trivial. The latitude/longitude position of each active flight is known and thus the flight's current grid cell location is also known. The Traffic Demands Functions simple calculate the demand at each grid cell and then use these numbers to calculate the demand for every sector, fix, and airport. However, the more important capability of the Enhanced Traffic Management System is its ability to calculate traffic congestion levels for a certain window of time into the future. The ETMS is generally capable of projecting traffic demands over the next 12-hour time horizon. This information is an essential component of traffic management and allows the traffic managers to reduce traffic flow before it reaches critical crowding levels. The ETMS also contains detailed information about the flight dynamics for every aircraft type that might come under the control of the FAA. This knowledge aids the Traffic Modeling Functions in the ability to project the routes of aircraft that are scheduled to depart in the Official Airline Guide. The Traffic Modeling Functions then relay the scheduled routes information to the Traffic Demands Functions.

The Traffic Demands Functions utilize the projected flight route information to determine grid cell demands over the next several hours. If these demands exceed capacity, the information is relayed to the Traffic Alerts Functions which then generate alerts. These alerts are relayed to the traffic managers via the TSD. However, none of these alerts would be identifiable without the information processing capabilities of the Traffic Modeling Functions. The Aircraft Dynamics Database contains flight profile information obtained from experts in the aviation industry (See Appendix A.2). These profile characteristics give a nominal flight path for each aircraft type by considering the airplane's maximum thrust, cruising altitude, and maximum air speed. A flight profile generally has three pieces of essential information. This includes the flight's ascent profile, descent profile, and normal cruising altitude and speed. Both the ascent and descent profiles are functions that specify aircraft speed and altitude given a distance from the departure and arrival airports respectively. Thus, it is possible to estimate the entire route that a scheduled flight will follow given its flight plan and aircraft type.

Together, the Traffic Modeling Functions, Traffic Demands Functions, and the Traffic Alerts Functions compose a knowledge representation system. The semantics utilized in this system meet the five criteria introduced by Davis et al. Recall that the Grid and Names Database contains information that is both a medium for efficient computation and human expression. The grid cells provide an internal computational scheme while the Names Database relates these cells to the common element reference names used by traffic managers. Also note that the Aircraft Dynamics Database attempts to model the overall flight characteristics for any given aircraft. This database acts as a substitute for the real flight characteristics of the active flights. Thus, air traffic can be projected into the future utilizing flight schedule information from the Official Airline Guide. These semantics portray a set of ontological commitments that focus on aircraft flight through the National Airspace System. In summary, they provide vital information content to the Traffic Management Specialists that aid them in controlling traffic congestion.

3.3.2 Self-Understanding in the ETMS

The information content provided to the traffic managers is in the form of Traffic Alerts and Traffic Demands. The Enhanced Traffic Management System also demonstrates a certain level of self-understanding. The knowledge representation utilized in the ETMS allows the users to query why certain traffic elements are alerted. For active flight alerts, the queries are somewhat trivial. The user can simply identify the flights that are currently positioned over the alerted element. However, without the Traffic Demands Functions, the traffic managers would be at a loss for determining which proposed flights have caused projected traffic congestion. The Traffic Modeling Functions communicate directly with the Traffic Demands Functions and thus each component knows which particular flights have caused any given alert. This information is especially useful to the traffic managers because they can identify flights that have caused any particular alert as well as determine all of the alerts that a given flight has caused. This allows the traffic managers to make strategic decisions regarding the redirection of traffic flow in the National Airspace System.

The Knowledge-Base Management Components of the ETMS provide high-level knowledge content to the traffic managers. This information is relayed to the Traffic Management Specialists via the Traffic Situation Display. The TSD utilizes several distributed knowledge management components to derive the information that is passed on to the traffic managers. In turn, these higher-level system components utilize many low-level distributed databases to make inferences regarding the status of air traffic flow. The next chapter will identify the major limiting factors of the present system. Namely, the knowledge management components fail to accurately model the real world. However, this is a recognizable fault inherent in all Knowledge-Base Management Systems. Recall that a knowledge representation is a substitute for objects in the real world and thus, it is impossible to accurately model the real world with absolute precision. Consequently, the performance of knowledge-base systems depends largely on the accuracy of the models that they utilize and this will be the major topic of the next chapter.

Chapter 4

4 DESIGN LIMITATIONS

The Enhanced Traffic Management System has proven to be an essential component of air traffic control throughout the United States, Guam, Puerto Rico, and in the U.S.-controlled oceanic airspace. Recently, the ETMS has been expanded to aid with air traffic control in Mexico, Canada, Great Britain, France, and Russia [16]. Indeed, the scope of the ETMS is easily expandable and the current implementation already tracks all international flights arriving in or departing from the United States. Even with the monumental success of the current system, there are several major design flaws that have raised concern from many Traffic Management Specialists. These design flaws have not created serious problems to date, but they have often impeded the underlying task that the system was developed to accomplish. Namely, the ETMS often generates false traffic congestion warnings and/or fails to recognize traffic congestion that has occurred. This chapter will investigate these design flaws and will identify the limiting factors of the system that may cause false inferences. This chapter will also study the current role of the ETMS in air traffic control and will identify what changes must occur for the system to reach the goal of *free flight*. The present level of automation and knowledge management is likely insufficient for the projected traffic demands in the future, let alone the goal of attaining free flight.

4.1 Traffic Model Inaccuracies

There have been multiple reports that the Enhanced Traffic Management System produces inaccurate traffic demands information. This erroneous information leads to false alerts that distract the traffic managers from carrying out more important tasks. Regardless of the accuracy of any alerted element, the traffic managers must investigate the problem and submit an acknowledgement to the Command Center that the problem has been rectified. These false alerts obviously burden the traffic managers, but more importantly, the Traffic Alerts Functions often fail to identify traffic congestion that has already occurred. Thus, traffic managers sometimes fail to identify traffic crowding before it reaches a critical level. This causes unnecessary delays and often creates situations that can easily lead to additional traffic congestion problems. These design flaws must be eliminated so that the Enhanced Traffic Management System can provide an optimal level of support to the Traffic Management Specialists.

4.1.1 Traffic Alerts Problems

The Traffic Alerts Functions have occasionally produced questionable data since their inception. These inaccuracies can often be traced back to certain sectors or fixes in various ARTCCs. The frequencies of the erroneous data and their non-random nature suggest that there is a fundamental problem in the methodology used to identify alerted elements. The evidence stems largely from Traffic Management Specialists in the field. This feedback suggests that the Traffic Modeling Functions do not accurately model the routes of proposed flights. Specifically, Traffic Management Specialists have suggested that the flight profiles used to model aircraft ascent and descent are not entirely accurate.

Traffic managers at the Albuquerque ARTCC complained about the Monitor Alerts Function back in September of 1995. For example, the super-high sector ZAB65 reportedly receives sector counts that are too high. This sector is southwest of Phoenix Airport and consists of airspace at and above 33,000 feet. One of the traffic managers noted that the problem was likely due to the ascent profiles of aircraft traveling from Phoenix [29]. Apparently, the aircraft scheduled to depart from Phoenix Airport never reach an altitude of 33,000 feet at the proper time to be included in the ZAB65 demand count. Thus, the Traffic Demands Functions compute that aircraft should be in a sector too quickly after takeoff. The traffic manages have learned to ignore these false alerts, but they must still take the time to ensure that each alert is indeed a false alarm.

A few of the traffic managers also noted that the Traffic Alerts Functions do not work well when flights deviate around a thunderstorm. Apparently, the Traffic Alerts Functions do not recognize that the active flights have changed their flight path to avoid adverse weather. There are often situations where flight crowding is not identified by the ETMS when multiple flights have been rerouted to avoid thunderstorms. The ETMS accurately portrays the scheduled flights that maintain their flight path, but the addition of redirected traffic is not included in the traffic demands. Thus, the traffic managers at the Albuquerque ARTCC have identified situations where the ETMS has failed to identify traffic congestion as well as situations where false crowding has been introduced. These problems suggest that major improvements need to be made to the methods used to identify traffic congestion in the National Airspace System.

Over three years have passed since the Albuquerque complaints and the original alert problems still exist. Traffic managers at the Boston ARTCC have recently complained that the Traffic Alerts Functions generate invalid information. For example, one of the traffic managers estimates that he receives 6 to 20 alerts per day with a 60% to 70% inaccuracy rate [37]. In addition, there are certain sectors that always contain false alerts data (i.e. one high sector over Cap Cod was mentioned). The traffic managers have also suggested that the descent profiles are inaccurate. Flights are modeled to descent too quickly and at too step an angle. They believe that the flight models are the primary reason why the ETMS generates false traffic demands information.

The traffic managers at the Boston ARTCC also brought up concerns about adverse weather. One traffic manager concluded that weather is the biggest issue of non-scheduled traffic management concerns. The Command Center often creates standard reroutes and/or ground delays for adverse weather conditions. These reroutes are implemented by the various ARTCCs depending on where the reroutes are to occur. However, the traffic managers note that heavy air traffic cannot be easily changed. These traffic reroutes often have a domino effect and cause additional traffic congestion at locations far away from the original location of the adverse weather conditions. Unfortunately, the Command Center is only partially aware of this additional traffic crowding due to the fact that the ETMS does not accurately identify traffic congestion caused from weather reroutes. Thus, inclement weather will frequently cause serious traffic congestion problems throughout the National Airspace System that costs the airline industry millions of dollars [37]. It is obviously in the best interest of the FAA to remedy the problems with the Traffic Alerts Functions.

4.1.2 Aircraft Dynamics Deficiencies

The Traffic Management Specialists tend to agree that the underlying problem with the inaccurate alerts information rests with the modeling of the aircraft ascent and descent profiles. The Aircraft Dynamics Database provides essential information to the Traffic Modeling Functions (refer to Appendix A.2). The aircraft dynamics for a scheduled flight dictate the projected altitudes and aircraft speeds that will be encountered during the actual flight. This information plays a large role in estimating the projected demands for all of the grid cells located in the National Airspace System. The traffic demands at each grid cell are then measured to determine the resultant affect on the various sectors, fixes, and airports in the NAS. Thus, the Traffic Modeling Functions must provide accurate flight path models to ensure that the projected traffic demands are reliable. Any inaccuracies in the predicted flight routes will eventually lead to false traffic demands information, thus causing erroneous alerts to be generated.

The aircraft dynamics database models descent profiles based largely on a study that focused on reducing fuel costs by varying approach angles at arrival airports. This study was conducted at Denver Airport and all flight approach models were constructed from aircraft arriving at this airport [31]. These models are currently applied universally to all scheduled airport arrivals generated by the Official Airline Guide. Thus, all flights are modeled using the same descent path. However, aircraft type has a large role in determining the final approach vector and aircraft landing speed. Consequently, the final descent path onto the airport runway is determined by the information enumerated in Appendix A.2. This information includes mach over indicated airspeeds for a variety of different aircraft types. These airspeeds indicate the rates at which aircraft slow down as they descend from their cruising altitudes and thus provide the necessary information needed to model the final approach path of various aircraft.

Figure 4-1 illustrates the original descent profile utilized by the Traffic Modeling Functions. This profile assumes that all aircraft arriving at an airport intersect a specified descent angle at their cruising altitude. The descent angle intersect point is calculated based on aircraft altitude and distance from the airport. Upon intersecting this descent angle, the aircraft then follows a constant 4-degree descent vector until it reaches an altitude of 12,000 feet where it levels off for 10 nautical miles. The aircraft continues to slow its airspeed at this level altitude until it intersects a second descent angle. This descent vector is then followed until the aircraft touches down on the runway at its specified landing speed. It is important to note that this model assumes all aircraft follow the same general approach path. However, the rate of descent and final landing speed vary between aircraft and thus, each aircraft will transverse this path at varying time rates.

For example, a heavy jet with a cruising altitude of 35,000 feet and a speed of 487 knots will take 22.9 minutes to touchdown utilizing the descent profile shown below. In comparison, a slower commercial jet cruising at 35,000 feet and 403 knots will take 26.3 minutes to touchdown [32]. These differences have a large affect on the traffic demands at various airspace locations near the airport. Thus, any inaccuracies in the flight descent profile will cause incorrect projected traffic demands at the sectors and fixes surrounding the airport. Given the complaints from the Traffic Management Specialists, the flight descent profile is a prime source of error for generating incorrect traffic demands data. Indeed, studies have verified that the flight descent profile models aircraft as descending at too steep an angle and too quickly [31]. Consequently, alternative models for the flight descent profiles have been proposed to correct these problems.

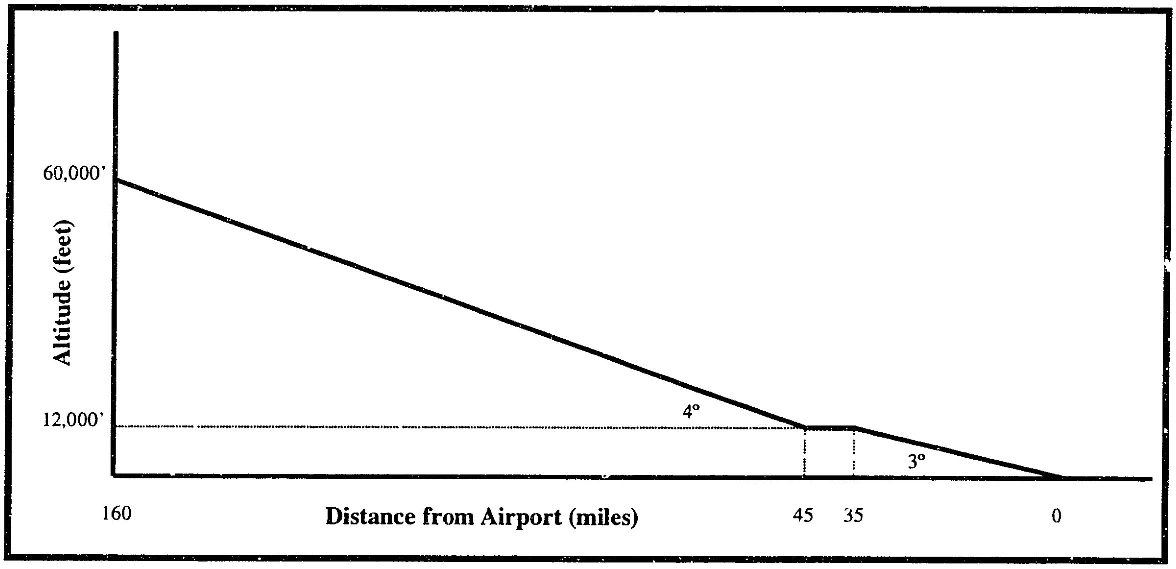


Figure 4-1: Flight Descent Profile[8]

---

[8] Adapted from [31].

Two new flight profiles were introduced in August of 1995 to correct the problems identified by the traffic managers at the Albuquerque ARTCC [40]. The proposed flight profiles contained the following changes:

- The initial glideslope in the first proposed flight descent profile was changed from 4 degrees to 3 degrees.
- The second proposed flight descent profile utilized a 3-degree glideslope and changed the level-out altitude from 12,000 feet to 10,000 feet.

These proposed flight profiles model aircraft descending at a slower rate than the original descent profile. The descent angle is not as steep as the original one and thus, it takes aircraft a longer time to transverse the new descent path. For example, the heavy jet mentioned earlier will take 26.9 minutes to touchdown using the first proposed descent profile. Thus, the new profile effectively adds 4 minutes to the flight's descent route. The commercial jet will transverse this new descent profile in 30.9 minutes. This represents a difference of 4.6 minutes from the original flight profile [32]. These changes attempt to correct the observed problems outlined by the traffic managers, but they do not address the fundamental inaccuracy of the flight profile model.

The proposed flight descent profiles are an adhoc solution to the inaccuracies of the Traffic Model Functions. The generalized flight profile model is inherently limited in its ability to calculate descent routes for aircraft landing at a myriad of different airports utilizing varying approach paths. Recall that the approach speeds and final aircraft landing speeds were compiled based on a study at Denver Airport. These speeds are applied universally to all airports using the same descent profile. One should be somewhat skeptical as to the resultant accuracy of this methodology. Aircraft tend to have different performance characteristics at different altitudes and Denver Airport is at a greater elevation than many other airports in the United States. Thus, data compiled at Denver Airport are not necessarily the best data to use when creating flight descent characteristics for each airport in the National Airspace System. Subsequent analysis of the proposed descent profiles reveals that there are indeed many circumstances where the aircraft descent models have proven grossly inaccurate.

4.1.3 Analysis of Flight Descent Profiles

The proposed flight descent profiles were analyzed for accuracy by comparing the modeled flight routes to real time flight data compiled by the ETMS. Data from various jets were collected by monitoring the Position Update Messages (TZs) generated by the Terminal Radar Approach Control facilities (TRACONs). Recall that the Flights Database maintains an entry of the last actual latitude, longitude, altitude, and speed for each active aircraft in the National Airspace System (refer to Appendix A.1). This database also contains a detailed listing of the aircraft type and flight plan. Position Update Messages occur approximately once per minute and the corresponding information is relayed to the Flights Database. Thus, it is possible to store a history of the Flights Database to examine certain flight data, which reveal the true descent path that various aircraft follow when landing at different airports. This data can then be compared to the descent profiles generated by the Traffic Model Functions to determine their accuracy.

Several airports were selected as test cases to analyze the accuracy of the proposed flight descent profile that utilized a 3-degree glideslope angle and a level-out altitude of 12,000 feet. The airports included Atlanta, Boston, Washington Dulles, Denver, Dallas Fort Worth, New York John F. Kennedy, Los Angles, Chicago O'Hare, and St. Louis. Data were collected by determining the most frequently used Standard Arrival Routes (STARs) for each airport. Recall that a flight route is defined by the Traffic Modeling Functions as a list of flight events that designate the sectors, fixes, and airports that the flight will transverse. Thus, the accuracy of the descent profile can be verified by comparing the aircraft's actual latitude, longitude, altitude, and speed to those generated by the Traffic Modeling Functions.

To better understand the methodology involved, the AVE-SADDE6-LAX STAR will be examined to illustrate a concrete example. Jet aircraft frequently use this arrival route during their final approach to Los Angeles Airport. The route follows jet J1 and victor V107 from northern California to Los Angles Airport (LAX). Flights often intersect fixes DERBB, REYES, PIRUE, FIM, GINNA, SYMON, and SADDE during their approach to LAX. Figure 4-2 illustrates the various NAS elements that comprise the AVE-SADDE6-LAX Standard Arrival Route. Note that jet J1 and victor V107 are indicated by dashed lines while the fixes are illustrated by triangles.

Aircraft flight data were collected from the ETMS by selecting flights that utilized the aforementioned arrival route. These flights were then analyzed to determine their actual approach route to LAX. The descent profile for each aircraft was compared to the actual position of the aircraft at each flight event point along its descent path. The resultant data for large jets utilizing the route illustrated below are given in Appendix A.3.

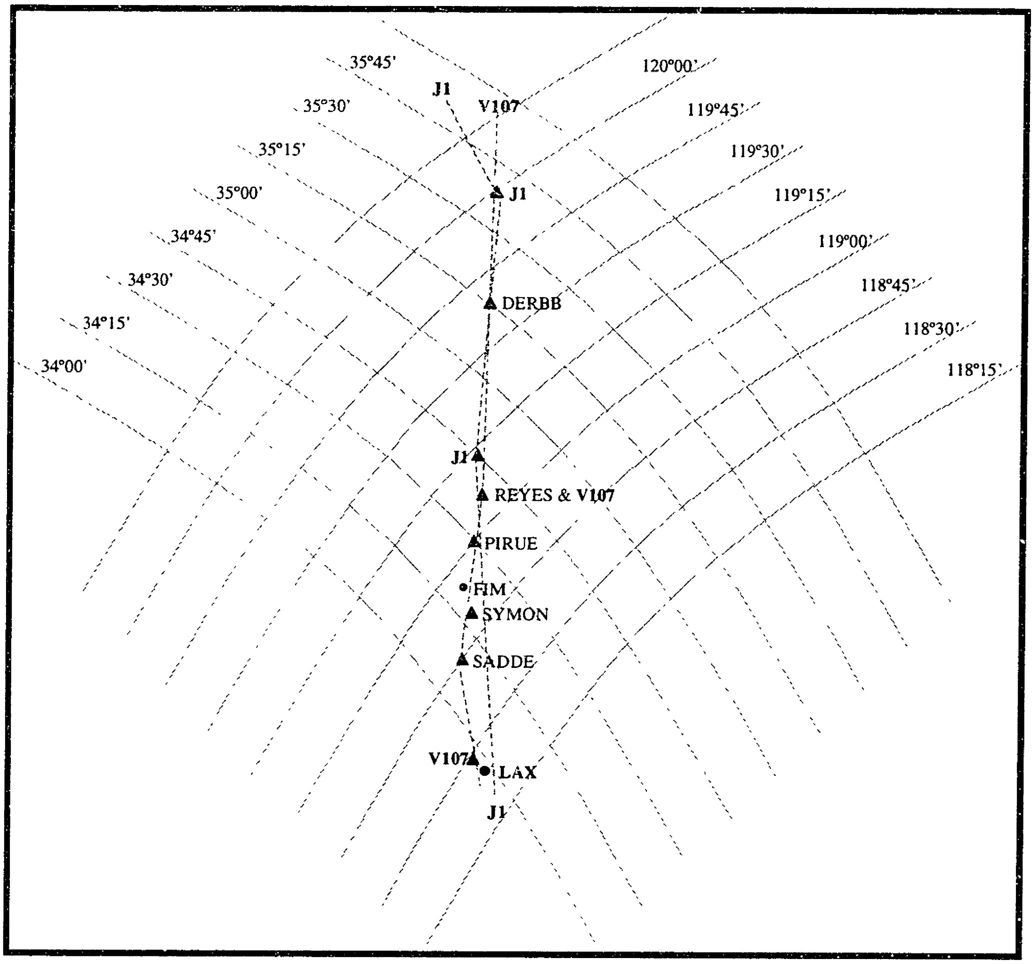


Figure 4-2: AVE-SADDE6-LAX Standard Terminal Arrival Route

There are several important trends to note from Tables A-6 through A-8. The first two tables include data from 47 flights arriving at LAX. The last table contains data from 26 flights. These flight data indicate that the proposed descent profile models aircraft at too high an altitude away from the airport and at too low an altitude just prior to touchdown. From REYES to SADDE, the modeled descent profile is from 2,500 to 6,750 feet lower than the actual altitude of the aircraft. From V107 to LAX, the modeled descent profile is from 100 to

550 feet higher than the actual altitude of the aircraft. Hence, the proposed flight descent profile models aircraft descending at too shallow an angle and at too low an altitude. Another interesting trend in the data is that every aircraft failed to intersect the J1, DERBB, ZLA26, and ZLA13 flight events. Flights that came within 6 nautical miles of the flight event were included in the model analysis. In other words, they were considered to have intersected the flight event. These data suggest that the modeled flight path is also inaccurate and this will be the topic of discussion in the next section.

4.2 Flight Routing Inadequacies

The ETMS utilizes historic flight data to determine the most likely route scheduled flights will follow. The flight paths, cruising speeds, and cruising altitudes are all calculated based on a city pair and airline entry in the Official Airline Guide. For example, the city pair JFK-LAX and the airline NWA represent a flight scheduled by Northwest Airlines that will depart New York's John F. Kennedy Airport and arrive at Los Angeles International Airport. Recall that the ETMS Auxiliary Support Functions maintain a Historic Flight Database that stores flight data for approximately one week. This database is used to determine the most likely route a scheduled aircraft will follow based on the departure and arrival cities as well as the airline that has proposed the flight. Note that aircraft type is also included in the OAG and this information is used to determine the flight's modeled cruising altitude and speed. The modeled route is determined probabilistically by considering the previous routes that were most often used between the city pair. The modeled altitude and speed are calculated based on the previous week's average values monitored for the proposed route given the aircraft type. The modeled route, cruising altitude, and speed are then all used by the flight ascent and descent profiles to attain a detailed list of flight events, which comprise the proposed flight route.

4.2.1 Flight Database Updates

After the scheduled flights are modeled, the resultant information is provided to the Flight Database. Recall that this database provides detailed information regarding the active and proposed flights in the NAS (see Table A-1 through A-4). In addition, the database maintains all flight information for the past 12 hours and projects flight data 12 hours into the future. The Flight Database receives the most up to date information regarding scheduled and active flights. Active flights are tracked by ground-based radar in the NAS and their position is updated once per minute. These updates provide crucial information to the Traffic Demands Database because they portray the active demand load at each element in the NAS. However, traffic managers have complained that the traffic alerts do not always recognize traffic congestion that is occurring in real time. These complaints suggest that the Traffic Demands Database does not receive proper flight update information for the active flights. There are several possible sources of error that may cause the active traffic demands to be calculated incorrectly. Furthermore, errors in flight updates can also lead to incorrect flight route information thus promoting additional errors in the projected traffic demands.

Proposed flights are relayed to the Flight Database via Scheduled Flight Plan (FS) messages created by the Traffic Model Functions. These messages are created for all flights scheduled to depart within a 12-hour window. They contain flight id, aircraft type, cruising speed, departure airport, scheduled departure time, cruising altitude, flight path, scheduled time en route, and Julian departure date. These data fields are immediately entered into the Flight Database as a new active flight so that it will be considered during traffic demands calculations. The flight is considered "live" when a Departure (DZ) message is received by the Flight Database. This message contains the flight id, aircraft type, departure point, activation time, destination, and estimated time to arrival (ETA). Upon receipt of this message, the Traffic Situation Display creates an aircraft icon to display the flight's progress to the Traffic Management Specialists (see Figure 3-5). Position Updates (TZs) are transmitted from the radar tracking stations throughout the U.S. to the ETMS such that the exact latitude/longitude, altitude, and speed are known for every live flight. Together, these flight update messages provide the information that is utilized to calculate the traffic demands throughout the NAS.

The primary source of inaccurate flight route information appears to reside in the methodology that is used to calculate the scheduled flight routes. Scheduled Flight Plan messages contain flight paths that are created based on the previous week's most frequently used routes. This method does not give additional weight to the routes being utilized between the arrival and departure cities during the day of scheduled departure. For example, adverse weather might cause normal inter-city routes to be changed so that flights can avoid severe turbulence. Furthermore, airports might utilize different approach routes given the runways that are in operation and the direction of takeoffs and landings. Thus, the original flight paths generated by the Traffic Model Functions generally contain errors that should be corrected utilizing additional information available in the ETMS. Unfortunately, the ETMS does not take full advantage of the data available in the system and often relies on adhoc methods to calculate projected traffic demands.

Another source of inaccurate flight route information may reside in the way that the ETMS handles flight plan information. Flight Plan (FZ) messages are received for all proposed flights that contain a detailed list of the planned flight route prior to departure. Recall that the airline flight plan processing centers are responsible for generating the flight routes that their aircraft will follow. These messages attempt to relay the most accurate Flight Plan, but there are many circumstances that often force the aircraft to deviate from the proposed flight route. Flight Plan Amendment (AF) messages try to address this issue by allowing changes in the original flight plans submitted by the airline flight plan processing centers. However, the air traffic controllers and/or the airlines that have made changes to the original flight route must submit these amendment messages. Traffic congestion, adverse weather, and flight substitutions often cause the original flight route to be changed. Unfortunately, these changes do not always get relayed to the Flight Database thus causing additional inaccuracies in the flight route information.

4.2.2 Adverse Weather Contingencies

Adverse weather is one of the primary causes of traffic delay in the United States. Severe weather can often shut down airports for extended periods of time thus causing delays throughout the entire National Airspace System. In less severe circumstances, poor visibility, caused by fog and/or precipitation, forces air traffic controllers to increase the separation between flights to maintain safety. The increased aircraft separation can greatly reduce the number of aircraft capable of landing airports which, in turn, often leads to traffic congestion and delays near the affected airports. The Enhanced Traffic Management System has a number of automated systems to deal with these circumstances. However, these systems provide limited ability for the airspace users to take their own course of action to reduce the adverse consequences of severe weather. Airlines are often forced to cancel flights while airlines with active flights are forced to land their aircraft at airports far away from their original destination. These circumstances cost the airlines hundreds of millions of dollars every year and can be blamed partly on the limited role that the airlines are given when flights must be redirected during severe weather [16].

Recall that ground delay programs are used to reduce the traffic demand by controlling the time when flights are allowed to depart. This often entails the alteration of scheduled departure times for flights listed in the Official Airline Guide. The ATCSCC creates ground delay programs by issuing Estimated Departure Clearance Time (EDCT) messages to the ETMS for the flights that are to be delayed. These messages are relayed to the Flight Database where the new controlled departure times are entered into the active flight record. This causes the Traffic Model Functions to generate a new flight route that corresponds to the new departure time. Consequently, the Traffic Demands Database is updated such that the traffic load is reduced according to the number of EDCTs issued. These messages are also relayed to the affected airlines so that they may take the appropriate action to minimize any adverse consequences that the departure delays might cause.

If the airlines have multiple flights that are affected by ground delay programs, they have the option to substitute flights for their allotted arrival times. This allows them to prioritize their delayed flights such that they have some flexibility during severe weather conditions. For example, suppose an airline has been issued delays for flights arriving at Chicago O'Hare due to snow. One EDCT message delays a scheduled flight from Boston Logan Airport for 30 minutes while another message delays a flight from Los Angels International Airport for 1 hour. Furthermore, there are many passengers from LAX who must make a connecting flight to London Heathrow Airport at O'Hare. Flight Substitutions (SUBs) allow the airline to prioritize the flights arriving at O'Hare so that the passengers from LAX don't miss their connecting flight. This is done by substituting the Boston flight's arrival slot with the Los Angeles flight's arrival slot. The SUB messages are handled automatically by the ETMS and the corresponding changes are made for each flight entry in the Flight Database.

Upon receipt of a Flight Substitution request, the ETMS verifies that the substitution meets certain criteria. For instance, both flights in the SUB must have been affected by a ground delay program and they must also be scheduled to arrive at the same airport over a given window of time. These criteria help ensure that the ground delay program carries out its intended task and also ensures that airlines do not abuse their ability to make adjustments to delayed flights. If the SUB is approved, the ETMS makes the necessary amendments to the Flight Database and Traffic Demands Database. Unfortunately, this is the extent to which airlines are capable of reacting to adverse traffic situations caused by severe weather. Airlines have virtually no control over the reroutes and/or delays that occur to the active flights that they control. For example, a Traffic Management Specialist at the Boston ARTCC noted that the only time he corresponds with airline personnel are when they call him to complain about delays that have occurred to the flights under his supervision [37].

Perhaps the most frequent flight delays and reroutes are those that occur within the ARTCCs to avoid traffic congestion within the center. These delays and reroutes generally involve minor flight adjustments so that traffic demand can be reduced at various NAS elements. As long as the delays do not affect traffic flow outside of their region, the ARTCCs are given the authority to make any flight adjustment they deem necessary. This often involves adding "mile and trail" to the active flights in their region so that aircraft arriving at airports and arrival fixes are spread out to avoid heavy demand. The same Boston ARTCC Traffic Management Specialist noted that, "it's the [job of the airline's flight plan processing center] to put all of their aircraft over a given arrival fix at the same time and it's my job to spread all this traffic out." [37]

It is somewhat ironic to note that the airline's flight plan processing center seems to be working against the air traffic controller. However, airlines derive obvious benefits if their aircraft arrive at the airport very close to one another. Passengers often need to make connecting flights and crewmembers are often transferred from one flight to the next. In the airline industry, time is money and flights that are not airborne are not producing revenue for the company. Thus, the airlines attempt to coordinate their airplane fleets such that passenger, crew, and equipment transfers occur in the most efficient manner possible. FedEx, who has the world's largest fleet of Boeing 727 aircraft, utilizes efficient fleet coordination at Memphis International Airport [8]. In 1997, FedEx transferred 2.3 million metric tons of cargo flown into Memphis International Airport. This was accomplished by concentrating flight arrivals and departures over very short time intervals. These methods have proven very successful for FedEx, but they have had serious ramifications on traffic congestion at Memphis airport.

The FAA estimates that Memphis International Airport will experience a 49% increase in landings and takeoffs between the year 1995 and 2010. This would make Memphis airport one of the fastest growing airports in the United States [35]. The high traffic demands placed on airports such as Memphis often create traffic situations that must be remedied by making minor flight adjustments to the aircraft approach routes. These adjustments often make aircraft enter holding patterns at certain flight event locations to reduce the arrival traffic at the airport. These adjustments rarely get entered into the ETMS and thus, the Traffic Demands Functions are supplied with incorrect aircraft descent profiles for the adjusted flight routes. The situation is often much worse during adverse weather conditions. Heavy traffic demand cannot be easily diverted if an airport must close temporarily. Such conditions create havoc at the affected ARTCCs and Traffic Management Specialists are often overcome with tasks related to distributing traffic load.

Adverse weather often results in incorrect Traffic Demands Data because large quantities of air traffic must be redirected in a short period of time. The original flight routes submitted to the ETMS are rarely updated in such situations and thus, the Traffic Modeling Functions never create new flight routes that reflect the new flight plans of the redirected aircraft. Furthermore, the airlines are given very limited flexibility in their ability to control the delays and reroutes that their aircraft are assigned and this often forces additional delays while flights wait for their connecting passengers and/or flight crews. Most of these problems are due to the current design implementation of the ETMS. Flight Plan updates are not automated to the extent that all deviations from the original flight plan are amended to the entries in the Flight Database. Moreover, the ETMS is not designed so that airline flight plan processing centers can take an active role in aiding the Traffic Management Specialists in determining where to redirect their flights during severe weather. The FAA must compliment the ETMS to resolve these problems and this will be discussed in more detail in the next chapter.

Chapter 5

5 IMPROVING TRAFFIC MANAGEMENT

The current implementation of traffic management in the National Airspace System reflects an architecture that is largely founded on a centralized approach to aircraft control. Pilots do not have the flexibility to choose the routes that they wish to travel. Instead, airline flight plan processing centers generate the flight routes that their aircraft are scheduled to follow while the air traffic controllers are responsible for directing the active flights along their respective routes making changes only when necessary. This approach has largely influenced the overall implementation of the Enhanced Traffic Management System and the navigation, communications, and computer systems which comprise it. This chapter will recommend several changes to the ETMS that will help improve traffic management capabilities. The previous chapter revealed that the current design architecture is limited in its ability to aid traffic managers in reducing air traffic congestion. Indeed, the present ETMS will unlikely be capable of supporting the projected increase in global air traffic nor will it be possible to achieve free flight without radical improvements to the system architecture.

5.1 Alternative Knowledge Representations

One of the major limiting factors of the current system rests in the Traffic Model Functions' inability to properly project traffic demands given the scheduled flights in the Official Airline Guide. Recall that the Traffic Model Functions utilize a single ascent and descent profile that is applied to the scheduled flights universally. This methodology was shown to have limited success. The knowledge representation utilized by the Traffic Model Functions must be improved to remedy the inaccurate traffic demands generated by the ETMS. A number of differ knowledge representations were introduced in Chapter 2. This section will build on several of these representations and will recommend alternative design approaches that are capable of improving the traffic demands accuracy of the ETMS.

5.1.1 Probabilistic Analysis of Flight Profiles

Evidence pooling is a common probabilistic approach utilized in Knowledge-Base Management Systems. Recall that MYCIN utilizes a Bayesian inference network to calculate the probability that a patient has a disease given their symptoms [9]. The same methodologies have been extended to other systems. For example, the Microsoft Office Assistant also utilizes a Bayesian inference network to trouble shoot problems that Word and Excel users encounter. Probabilistic analysis can be very successful in systems that store historic data [34]. The ETMS is one such system and Chapter 4 discussed the methodologies used to compare historic data to the flight profile models generated by the Traffic Model Functions. These historic data can be used to develop new knowledge representations that utilize probabilistic inferences to generate flight profile models. Most of the necessary components are already available in the ETMS and the addition of a probabilistic inference engine would not be overly difficult.

The first step in creating a probabilistic knowledge representation involves analyzing the historic data utilizing Bayesian techniques. The essence of this analysis rests in the celebrated inversion formula given in Equation 5-1. The inversion formula states that the belief given to a hypothesis H upon obtaining evidence e can be computed by multiplying the previous belief in the hypothesis $P(H)$ by the likelihood $P(e|H)$ that e will materialize if H is true [34].

Equation 5-1: Bayes Conditional Probability Formula

$$p(H \mid e) = \frac{p(e \mid H)p(H)}{p(e)}$$

The a priori probability that the given hypothesis is true $P(H)$ can be found by considering the historic flight data for various elements in the National Airspace System. For example, a new flight descent profile for the AVE-SADDE6-LAX STAR could be generated by analyzing the data shown in Tables A-6 to A-8 of Appendix A.3. Recall that the original proposed descent profile models aircraft as descending at too low an altitude. For instance, large jets are, on average, 3,465 feet higher than their modeled altitude at the SADDE arrival fix [33]. These same statistics can be used to generate the a priori probability that an aircraft will be at a given altitude over a given position along the arrival route. A new descent profile can be suggested by considering different altitude ranges that a flight will most likely fall into.

Assuming that the historic flight altitude data are normally distributed, it is possible to calculate the probability that a flight will fall into a given altitude range at a given NAS element. This probability can easily be found by considering the relation of the mean altitude and altitude standard deviation to a normalized probability distribution with a mean of zero and a standard deviation of one. For example, the probability that a large jet will be between 11,293 feet and 9,637 feet at SADDE is determined by using the following relation:

Equation 5-2: Probability Distribution Calculation

$$p(x_1 \le x \le x_2) = p(z_1 \le z \le z_2) = F(z_2) - F(z_1)$$

Where x_1=9,637 and x_2=11,293. The z values correspond the normalized x values that can be found by utilizing a *z-score* conversion equation given below:

Equation 5-3: Standard *Z-Score* Calculation

$$z_i = \frac{x_i - \mu}{\sigma}$$

Note that μ represents the mean altitude value at the given NAS element and σ represents the altitude's standard deviation. Utilizing the data in Tables A-6 through A-8 for the SADDE fix, the mean altitude was found to be 10,465 feet with a standard deviation of 828.2 feet. Thus, z_1=-1.0 and z_2=1.0. $F(z)$ is often included in the appendix of a statistics book, such as [26], and the remaining calculation is trivial. $F(z_1)$=0.159 and $F(z_2)$=0.841, hence $F(z_2)$-$F(z_1)$=0.682, which represents the probability that a large jet will be between 11,923 and 9,637 feet at the SADDE arrival fix during its descent into Los Angels International Airport when utilizing the AVE-SADDE6-LAX Standard Terminal Arrival Route.

This altitude range was chosen such that it is plus or minus one standard deviation from the mean. This is why the z values were found to be -1.0 and 1.0. These methods can be extended to the other flight events such that the most likely altitude range can be calculated for the entire Standard Terminal Arrival Route. The resultant calculations for the data enumerated in Appendix A.3 are given in Table A-9 of Appendix A.4. These altitude ranges were chosen such that the a prior probability that a large jet will be between the high and low altitude range is always equal to 68.2%, which corresponds to plus or minus one standard deviation from the altitude mean at the given flight event location. If there are particular airspace regions that must be distinguished, e.g. low sector areas versus high sector areas, the altitude ranges can be adjusted to find the probability that the flight will fall into one or the other airspace regions.

The next step involved in creating a probabilistic knowledge representation entails analyzing the historic data using different evidence criteria. For example, the current flight descent profiles use aircraft type to determine the rate of descent that an aircraft will experience while following the descent angle. However, the aircraft types are broadly categorized into three groups (See Appendix A.2). Probabilistic analysis can be used to better categorize the aircraft such that each aircraft is assigned a descent profile that it will most likely follow. This is done by using Equation 5-1 to determine the altitude range an aircraft will fall into given its type as evidence. For instance, to calculate the probability that a Boeing 727 will be between 11,293 feet and 9,637 feet at SADDE, the historic data must be analyzed to determine $P(e)$ and $P(e|H)$. Recall that $P(H)$, or the a priori probability that a flight will be in the aforementioned altitude range, was calculated to be 68.2%.

The probability of observing the evidence $P(e)$ is simply $P(e|H)P(H) + P(e|{\sim}H)P({\sim}H)$. The information obtained in Tables A-6 to A-8 include data from 32 Boeing 727 aircraft. Of these 32 flights, 31 of them had an altitude reading between 11,293 feet and 9,637 feet at the SADDE fix. Only one of the Boeing 727's fell out of this range. Thus, $P(e|H)$=31/32=.96875 and $P(e|{\sim}H)$=1/32=.03125. Using this information, Equation 5-1 becomes:

Equation 5-4: Utilizing Bayes Formula for Evidence Pooling

$$P(H \mid e) = \frac{P(e \mid H)P(H)}{P(e)} = \frac{.96875 \times .682}{(.96875 \times .682)(.03125 \times .318)} = .985$$

Given the additional information that a flight is a Boeing 727, the probability that its altitude will be between 11,293 and 9,637 feet at the SADDE fix has increased from 68.2% to 98.5%. This example illustrates the benefits that can be derived by using Bayesian analysis. As more evidence is considered (i.e. flight aircraft type, flight origination, weather conditions, etc.), the probability that a flight will be in a given altitude range along its flight path can be refined such that its flight profile is extremely accurate. The historic data needed for this analysis already exists. The ETMS can be easily extended to incorporate a probabilistic inference engine that utilizes the calculations illustrated above to create precise flight profile models for various aircraft types landing at different airports throughout the NAS. The more accurate flight profile models would provide a corresponding increase in the accuracy of the Traffic Demands Database. These benefits would greatly enhance the capabilities of the ETMS and would also aid the Traffic Management Specialists in reducing traffic congestion in the NAS.

5.1.2 Increasing Knowledge Acquisition

There is a tremendous amount of information contained in the ETMS that is rarely utilized. The previous section illustrated one example of how additional knowledge representation can improve the overall performance of the ETMS. Indeed, there are many other techniques that can be used to increase the overall amount of information that is derived from the ETMS. Unfortunately, many organizations never utilize the full potential of the databases they employ. By some estimates, less than seven percent of the data collected in corporate databases are used [27]. The same is true of the ETMS. The Historic Flight Database is only used to determine the most likely route that a scheduled aircraft will follow and the estimated ground clearance times that will be experienced at various airports in the NAS. There is a tremendous amount of additional information that could potentially be derived from the Historic Flight Database.

Currently, the Historic Flight Database stores data for one week or less. Older data are purged from the database so that it does not become too large. However, additional knowledge could be derived from data stored longer than one week. For example, many corporations utilize data repositories to store historic data for extended periods of time. These data are used to find trends or relationships that are not evident in shorter periods of time. The historic data enumerated in Tables A-6 through A-8 of the appendix revealed that flights are much higher than their modeled descent profile. This is only one example of the type of information that can be found by analyzing historic data. By increasing the amount of historic data contained in the ETMS, additional traffic trends might be identified that could aid in the reduction of traffic congestion in the NAS.

Data mining techniques are often applied to large historic databases to search for specific trends in the data. For example, data mining could potentially reveal the most optimal ascent and descent profiles to apply at various airports in the NAS. This is similar to the probabilistic methods described in the previous section with the exception that data mining provides a more automated method in extracting the information contained in the data. Bayesian analysis requires detailed steps in formulating the probabilities that an aircraft will fall into a given altitude range. Data mining often uses a less structured approach and simply searches the data to find any predictive trends that can be utilized to solve a specific problem. These predictive trends provide unforeseen methods that might not have been realized without the use of data mining. Thus, there is a potential wealth of information that is still hidden to the users of the ETMS in the historic databases that track the progress of every flight in the NAS. The Historic Flight Database could prove even more useful if it stored data for an extended period of time.

5.2 Future Air Traffic Management

The FAA recognizes the need to improve traffic management capabilities to handle the projected demand in air traffic over the next several years. The FAA also acknowledges that the goal of free flight must be obtained by the implementation of better technologies and improved air traffic control efficiency in the National Airspace System. The Air Traffic Services division of the FAA believes that these goals can be achieved through the implementation of decision support tools, reduced aircraft separation minima, improved air/ground communications and coordination, and enhanced weather detection and reporting capabilities [1]. More importantly, these goals must be considered in the larger scope of the evolving air traffic systems around the world. Indeed, the future of air traffic management will involved the coordinated efforts of many countries and the emergence of a global air traffic management system.

5.2.1 Global Traffic Management

The International Civil Aviation Organization has been charged with the task of implementing a global communication, navigation, and surveillance air traffic system (CNS/ATM). Recall the Future Air Navigation System introduced in Chapter 2. This system is currently operational and has been used extensively throughout Asia. The adoption of FANS represents a cornerstone in the implementation of a global traffic management system. FANS provides needed improvement to the location accuracy of active aircraft. This system also promises to integrate many disparate traffic control systems that have been implemented around the world. Thus, important information regarding the status of planned and active flights can be easily relayed to air traffic managers, pilots, airline operations personnel, passengers, and all other relevant people in a quick and efficient manner.

Increased communications will provide benefits to the systems that are responsible for projecting traffic demand. Recall that one of the limiting factors of the ETMS is its inability to accurately predict the aircraft routes of scheduled and active flights. These problems stem, in part, from reroute adjustments made during flight, changes in proposed flights by the airlines, flight diversions made by the traffic managers to reduce congestion, and flight adjustments caused by adverse weather conditions. FANS offers the ability to relay any deviation in flight routes to the traffic demands system. Real-time trajectory updates reflect more accurate route profiles, thus eliminating many of the inaccurate traffic demand predictions. Improved communications are accomplished through a global aeronautical telecommunications network that links air traffic control centers worldwide. This network provides the swiftest possible transmission of aircraft location and route plans such that any center can track the aircraft's flight progress [13].

FANS also allows the flight crew the freedom to choose their own routes with limited intervention from the air traffic controllers on the ground. The automated air-to-ground data transmission links provide contingencies for advanced warning of nearby aircraft traffic. Thus, the controllers take action only if a flight is about to enter an area of high congestion. The air traffic managers are assisted by FANS because the system provides pinpoint aircraft location accuracy and advanced information regarding the flight's intended route. This allows the traffic managers to increase aircraft density over highly active airspace regions thus eliminating many flight delays and/or reroutes. In many areas of low traffic density, the aircraft pilots have total freedom to choose their route. However, in areas of higher traffic density, FANS must be improved before it will be capable of providing full air traffic management support.

5.2.2 Attaining Free Flight

Even with the improved traffic management that FANS promises to offer, free flight is unlikely to be feasible in the near future. The ETMS is by far the most efficient air traffic management system currently implemented. Indeed, the ETMS handles the greatest amount of traffic flow while providing the most efficient service available in the world. However, several design limitations have prevented the ETMS from giving the airlines the freedom to select their own flight routes at will. Several steps have been taken to provide additional cooperation between the Traffic Management Specialists and the airline flight plan processing centers. These steps attempt to rectify the coordination problems revealed in Chapter 4 and the best hope for free flight rests in the ability to create systems that allow airspace user input when deciding how to divert traffic flow to control congestion problems.

The FAA created the Collaborative Decision-Making (CDM) program in March of 1992 to improve communication between airlines and the Traffic Management Specialists [16]. CDM promises to give the airlines more flexibility during the implementation of ground delay programs. Recall that the current design implementation of the ETMS only allows the airlines to exchange delayed flight arrival slots with other delayed flights under their control. Thus, the airlines effectively have no control in the manner in which EDCT messages are issued. This often creates circumstances where the airline flight plan processing centers work at odds against the Traffic Management Specialists. Mike Nadon, the President of the Airline Dispatchers Federation (ADF), emphasized many of these shortcomings during his testimony before the National Civil Aviation Review Commission:

> "The ADF has been active in creating and promoting Collaborative Decision Making with the Air Traffic Management specialists at the FAA. When we began this process dispatchers saw the Air Traffic Community as just another force to be overcome in order to operate their flights as safely and efficiently as possible, just as they viewed thunderstorms, blizzards, and volcanic ash. There was almost an adversarial relationship and dispatchers delighted in 'gaming' the system to get some incremental advantage for their flight or airline." [16]

Mr. Nadon's statements underscore the lack of cooperation between the airline dispatchers and the traffic managers. Many of these inefficiencies can be traced back to the design architecture of the ETMS and the hierarchical structure of the various traffic control centers in the United States. Ground delay programs are often implemented by the ATCSCC. However, the Traffic Management Specialists at the ATCSCC often lack information pertaining to the actual congestion levels in the NAS due to the problems associated with the inaccurate flight routes produced by the Traffic Modeling Functions. Thus, ground delay programs often cause additional traffic delays that can be avoided by fixing the problems enumerated in Chapter 4. Specifically, the Flight Schedule Database must receive frequent updates such that it reflects the most accurate picture of current and proposed air traffic in the NAS. In turn, these updates will provide the Traffic Modeling Functions with the information needed to produce accurate flight routes so that the Traffic Demands Database reflects the true level of traffic congestion in the National Airspace System.

The Collaborative Decision-Making Program offers a significant improvement in the communications that occur between the airlines and the traffic managers. The Data Exchange program implemented by CDM provides the Traffic Management Specialists with real-time schedule information provided directly by the airline dispatchers at the flight plan processing centers. This schedule information provides the most recent updates in the airlines' proposed flights and chosen flight routes. Thus, Flight Plan Amendment messages can be entered into the ETMS so that the Flight Model Functions generate the appropriate flight paths for each scheduled flight. With these improvements, the Traffic Demands Database provides a much more accurate picture of traffic flow in the NAS and this greatly improves the Traffic Management Specialists' ability in controlling congestion.

The benefits derived from the CDM program are quite perplexing. The FAA has estimated that the first trial runs of CDM, from late 1997 to early 1998, have already saved the airlines and airline passengers $2.6 billion [16]. These savings are derived from reduced traffic delays through improved collaboration between the airlines and the traffic managers. CDM allows the airline dispatchers to view the traffic demands at various locations in the NAS and thus they can take a more active role in preventing their flights from entering areas of high congestion. For example, United Airlines estimates that their delays have been reduced by roughly 50 to 60 percent since the first implementation of CDM [16]. The CDM program also provides several enhancements to the ETMS that will aid in the development of free flight.

Recall that the current ETMS architecture is largely founded on a centralized approach to air traffic control. The traffic managers are responsible for directing the traffic flow throughout the NAS and they rarely accept outside input when traffic flow must be adjusted to control traffic congestion. The CDM program has initiated a paradigm shift in the present structure of air traffic control in the United States. Airlines can now access the same information that is available to the Traffic Management Specialists. Furthermore, the airline dispatchers can take an active role in deciding how to best avoid delays to the aircraft under their supervision. In turn, CDM offers more accurate information to the traffic managers so that they can make more informed decisions when deciding how traffic should be rerouted to avoid congestion. The next leap will be to provide the pilots with similar information and freedom so that they can take a more active role in determining the flight path that their aircraft will follow. Further improvements in the Collaborative Decision-Making Program could eventually provide more freedom to the end users of the NAS, thus bringing the FAA one step closer to the goal of free flight.

Chapter 6

6 CONCLUSIONS

This thesis has introduced various concepts related to the task of controlling air traffic in the National Airspace System. The Federal Aviation Administration relies primarily on the Enhanced Traffic Management System to facilitate air traffic control in the United States. The design architecture of this system was illustrated in Chapter 3. The ETMS relies on the heterogeneous unification of DBMS and KBMS components. Knowledge representation facilities and the database management components utilized by the ETMS were explained in relation to the paradigms introduced in Chapter 2. Indeed, the design architecture of the ETMS has slowly evolved from a purely data focused implementation to one that utilizes knowledge representation and information processing techniques. However, there are several limiting factors that the ETMS must overcome to maintain efficient traffic management in light of the projected increase in domestic and international air traffic demands. These limiting factors were revealed in Chapter 4 by analyzing historic flight data stored in the ETMS. The same historic data was then used to develop new knowledge representation schemes that could potentially be implemented in the ETMS. Chapter 5 discussed how these new representations could improve the abilities of the ETMS and suggested what the future might hold for air traffic management.

These chapters offer much insight toward the evolution of the technology utilized by the ETMS. First and foremost, the ETMS is a mission critical system that is responsible for aiding in the management of air traffic flow throughout the United States and surrounding areas. The ETMS was developed as a distributed database system to provided the most reliable services. In turn, the distributed approach offered the increased flexibility to add and upgrade system components as necessary. This has allowed the implementation of Knowledge-Base Management Components that facilitate in the identification of Traffic Alerts. These KBMS facilities are still in their early stage of development and many faults must be rectified before they are capable of providing optimal service. Fortunately, the ETMS differs from traffic management systems in Asia such that these faults have not caused the entire system to crash. Thus, additional components can be added to the ETMS in an incremental fashion such that the system maintains optimal reliability. This is perhaps one of the greatest advantages that the ETMS has to offer. However, these incremental additions may not be able to keep up with the rate of increase in projected international air traffic demand.

Alternative approaches to air traffic management were suggested by several international organizations. For example, FANS promises to be the first global air traffic management system, and is supported by both the International Air Transport Association and the International Civil Aviation Organization. This system offers radically advanced traffic management capabilities and provides the first steps necessary to attain free flight. However, the many disparate traffic management systems around the globe have prevented FANS from being adopted by a majority of the countries in the world. In the end, free flight will only be achievable if every nation adopts a single global traffic management system that is capable of supporting free flight. Otherwise, pilots will be limited in their ability to freely choose their flight path and will need to select a course where free flight is supported.

In conclusion, a truly global air traffic management system will need to rely on many of the technologies being developed for the ETMS and a successful global system will include advanced Knowledge-Base Management Systems. KBMS components provide increased system automation and efficiency such that less human intervention is required. The KBMS facilities exemplified in the ETMS underscore the important role that these components will undertake in future air traffic management systems. Indeed, the benefits derived from KBMS components are directly evident in the increased profitability that airlines have experienced in the United States. Many countries are attempting to model their domestic air traffic management systems after those implemented in the U.S. Working together, the international community is capable of developing a new global air traffic management system that may eventually mark the successful vision of free flight.

REFERENCES

[1] Air Traffic Services, "ATS Concept of Operations for the National Airspace System in 2005," Federal Aviation Administration, September 30, 1997.

[2] Allen, D., "Boeing System Allows 747s to be Navigated By Satellite," *Journal of Commerce*, December, 1995.

[3] Arnold, W., "Human Error Causes System Glitches That Embarrass New Asian Airports," *The Wall Street Journal*, July 13, 1998.

[4] Automation Applications Division, "Enhanced Traffic Management System Design Document, Version 5.8," System Specification, U.S. Department of Transportation, May, 1996.

[5] Bray, O., "Distributed Database Management Systems," Lexington, Massachusetts, 1982.

[6] Brodie, M. and Mylopoulos, J., "On Knowledge Base Management Systems," New York, 1986.

[7] Buchan, D., "Airlines Demand One Traffic Control System for Europe," *Financial Times (London)*, September 7, 1989.

[8] Clancy, B., "Air Cargo - FedEx May Get Lighter Load Limits," *The Commercial Appeal*, July 15, 1997.

[9] Davis, R., Buchanan, B. and Shortliffe, E., "Production Rules as a Representation for a Knowledge-Based Consultation Program," *AI Magazine*, Fall, 1977.

[10] Davis, R., Shrobe, H. and Szolovits, P., "What Is a Knowledge Representation?" *AI Magazine*, Spring, 1993.

[11] Dempsey, M., "Europe Charts a New Flight Plan - Steps are being taken towards solving the problem of congested airspace," *Financial Times (London)*, January 31, 1995.

[12] Dickson, T., "EC to Have 40 Million Pounds Air Traffic Centre," *Financial Times (London)*, July 5, 1989.

[13] Donne, M., "Navigation: Safer, Faster, and Cheaper," *Financial Times (London)*, June 12, 1997.

[14] Elias, R., "IATA Reviewing Action Plan to Cut Congestion," *Shipping Times*, December, 1995.

[15] ETMS Development Committee, "Enhanced Traffic Management System," March 24, 1995.

[16] FAA Website, Federal Aviation Administration. http://www.faa.gov/

[17] Futer, A., "Tools to Monitor ETMS/CDM Data Base Consistency," Internal Memo, February 11, 1998.

[18] Gifford, D. and Spector, A., "The Cirrus Banking Network," *Communications of the ACM*, August, 1985.

[19] Hale, F., "Aircraft Performance, Selection, and Design," New York, New York, 1984.

[20] Johnstone, A., "Air Turmoil Looms on Radar," *The Herald (Glasgow)*, May 12, 1995.

[21] Kanellakis, P., "The Complexity of Concurrency Control for Distributed Databases," Doctoral Thesis, MIT Laboratory for Computer Science, September, 1981.

[22] Kerry, R., "Integrating Knowledge-Based and Database Management Systems," New York, 1990.

[23] Kinnock, N., "Air Traffic Reform Urged in EU," *Journal of Commerce*, March 7, 1996.

[24] Kotaite, A., "India: Satellite-based air navigation by 2010," *The Hindu*, 1997.

[25] Lee, D., "A Satellite System to Help Airlines Reduce Costs," *Shipping Times*, June, 1996.

[26] Loftus, G. & Loftus, E., "Essence of Statistics," New York, New York, 1988.

[27] Mao, Y., "Data Warehousing, OLAP, and Data Mining: An Integrated Strategy for Use at FAA," Masters Thesis, Massachusetts Institute of Technology, May 1998.

[28] Office of Aviation Policy and Plans, "FAA Aviation Forecasts, Fiscal Years 1997-2008," Federal Aviation Administration, March, 1996.

[29] Oiesen, R., "Trip Report to the Albuquerque ARTCC," Internal Memo, September 14, 1995.

[30] Oiesen, R., "Minutes of the TIM on the TSD at the Command Center," Internal Memo, March 10, 1998.

[31] Parkin, L., "New Descent Profiles," Internal Memo, March 8, 1994.

[32] Parkin, L., "Flight Profile Modification Description," Internal Memo, July 17, 1995.

[33] Parkin, L., "Flight Profile Assessment," Internal Memo, November 17, 1995.

[34] Pearl, J., "Probabilistic Reasoning in Intelligent Systems," Los Angeles, California, 1988.

[35] Perl, A., "Runway Readies Memphis For 21st Century Takeoff," *The Commercial Appeal*, April 12, 1998.

[36] Reimer, U. and Hahn, U., "A Formal Approach to the Semantics of a Frame Data Model," *Proc. IJCAI-83*, Karlsruhe, 1983.

[37] Rodarte, C. and Traffic Management Specialists, Interviews at the Boston ARTCC, July, 1998.

[38] Sweeny, L., "Making Friendly Skies Fair, Too", *Journal of Commerce*, January, 1997.

[39] Tanenbaum, A., "Modern Operating Systems," Amserdam, The Netherlands, 1992.

[40] Vinson, E., "Flight Descent Profile Flattening," Internal Memo, July 25, 1995.

[41] Winston, P., "Artificial Intelligence," Reading, Massachusetts, 1992.

[42] Yeow, J., "Monitoring KLIA Progress to Fix Opening Date," *Business Times (Malaysia)*, January 13, 1998.

[43] Zarocost, J. and Burgess, L., "Satellite Navigation Offers Big Savings ICAO Sets 2010 as Date for Debut of Global System," *Journal of Commerce*, October, 1994.

A APPENDIX

A.1 Database Structures

Table A-1: Flights Database Structure[9]

Data Item	Definition	Unit/Format	Variable Type
tdbentry	Does this flight exist in the TDB?	Not Applicable.	boolean
tdb_num_of_events	Number of events in the TDB for this flight.	Not Applicable.	int
id_of_flight	Flight identification.	1 or 3 letters followed by numbers.	string7
comp_id	Computer identification for a flight.	3 numbers.	string3
ftm_flight_tag	Date/time stamp for FTM and TDB transactions.	Packed julian date and time in minutes.	int
actype	NAS abbreviation for aircraft type.	Combination of letters and numbers	string4
user_category	Category of user.	An enumerated type: Commercial, military, etc.	usercat_t
flightreg_class	Flight regulations observed by the flight.	An enumerated type: VFR, IFR_CAT1, etc.	flight_regs_t
ac_general_class	General type of aircraft.	An enumerated type: Land, helicopter, etc.	actype_t
ac_cat_class	Specific type of aircraft.	An enumerated type: Civilian, jet, prop, etc.	ac_cat_t
ac_weight_class	Aircraft weight class.	An enumerated type: Small, large, etc.	ac_weight_t
numb_of_aircraft	Number of planes in flight.	Only filled for multiple plane flights.	integer
fstatus	Status of the flight.	An enumerated type: Scheduled, filed, etx.	status_of_flight_t
last_msg_time	Time of last NAS message receipt.	Year, month, day, hour, and minute.	cal_$timedate_rec_t
message_history	History of flight's NAS messages.	Packed format containing counts for each message type.	int
fdb_indicator	16-bit flag field used internally to the FDB.	Packed format, one bit for each of up to 16 flags.	int
current_ctr	ARTCC through which the flight is currently flying.	One letter code representing the ARTCC.	char
departure_subs	Subscript into the departure array.	A number from 0 to 191.	int
arrival_subs	Subscript into the appropriate arrival time array.	A number from 0 to 191.	int

---

[9] Adapted from [15].

Table A-2: Continuation of the Flights Database Structure

Data Item	Definition	Unit/Format	Variable Type
profile_info	Record containing values for flight modeling routinea.	Indices and distances to define aircraft dynamics.	profile_rec_t
tz_indicator	Flags for TZ processing.	16 one-bit fligs, keep track of past TZ processing.	int
tz_message_history	Number of TZ messages received.	A number.	int
TO_message_history	Number of TO messages received.	A number.	int
TA_message_history	Number of TA messages received.	A number.	int
TZ_delay_indicator	TZ delay information.	16 bits.	int
tz_time	Time when last TZ was received.	Year, month, day, hour, and minute.	cal_$timedate_rec_t
TO_time	Time when last TO was received.	Year, month, day, hour, and minute.	cal_$timedate_rec_t
TA_time	Time when last TA was received.	Year, month, day, hour, and minute.	cal_$timedate_rec_t
last_act_lat	Last actual latitude.	Radians times 10,000.	int
last_act_lon	Last actual longitude.	Radians times 10,000.	int
lat_pos_predicted	Predicted latitude.	Radians times 10,000.	int
lon_pos_predicted	Predicted longitude.	Radians times 10,000.	int
last_actual_event	Event number for plane's last reported position.	Number between 0 and 250.	int
next_goal_event	Event number for plane's next flight-plane position.	Number between 0 and 250.	int
fl_measured_heading	Flight heading measured between TZ messages.	Nadian time 10,000.	int
total_distance	Total distance covered by the event list.	Nautical miles.	int
total_distance_strline	Total straight line distance flown.	Nautical miles.	int
total_distance_passed	Total distance flown so far.	Nautical miles.	int
smooth_deviation	Smoothed deviation from route.	Nautical miles.	int
smooth_dev_slope	Smoothed slope of deviation value.	Nautical miles per minute.	int
time_smooth_made	Time of last reevaluation in TZ processing.	Year, month, day, hour, and minute.	cal_$timedate_rec_t
proposed_speed	Requested air speed.	Nautical miles per hour.	int

Table A-3: Continuation of the Flights Database Structure

Data Item	Definition	Unit/Format	Variable Type
reported_speed	Last reported ground speed.	Nautical miles per hour.	int
proposed_ait1	Requested altitude.	Hundreds of feet.	int
reported_alt1	Last reported altitude.	Hundreds of feet.	int
departure_ap	Airport of flight's origin.	Combination of 3 or 4 letters and numbers.	string4
arrival_ap	Destination airport of flight.	Combination of 3 or 4 letters and numbers.	string4
ground_time	Predicted number of minutes for flight to taxi.	Minutes.	int
dep_pushback	Predicted value for departure queue delay.	Minutes.	int
departure_date	Julian date for flight's departure.	Number of days since January 1, 1980.	int
proposed_dep_time	Flight's proposed departure time.	Minutes from midnight.	int
actual_dep_time	Flight's actual departure time.	Minutes from midnight.	int
sched_dep_time	Flight's scheduled departure time.	Minutes from midnight.	int
control_dep_time	Flight's controlled departure time.	Minutes from midnight.	int
first_event_time	Time from first event in the event list.	Minutes from midnight.	int
orig_dep_time	Lastest proposed or scheduled dep time before control.	Minutes from midnight.	int
proposed_arr_time	Flight's proposed arrival time.	Minutes from midnight.	int
init_arr_time	Intial prediction fro flight's arrival time.	Minutes from midnight.	int
curr_arr_time	Current prediction for flight's arrival time.	Minutes from midnight.	int
sched_arr_time	Flight's scheduled time of arrival.	Minutes from midnight.	int
orig_arr_time	Latest proposed or scheduled arrival time before control.	Minutes from midnight.	int
timestamp_offset	Time between time in DZ message and time stamp.	Seconds.	int

Table A-4: Continuation of the Flights Database Structure

Data Item	Definition	Unit/Format	Variable Type
ground_time_method	Mode used in ground time determination.	a=aircraft, c=category, t=controlled, d=default	char
dept_center	Departure center code.	Character or symbol.	char
arr_center	Arrival center code.	Character or symbol.	char
controllable	Indicates flight's ability to accept control programs.	Not Applicable.	boolean
arrival_fix_event	Event to describe the flight's arrival fix.	Location and time of arrival fix.	short_erect
numb_of_events	Size of flight's event list.	A number from 0 to MAXEVENTS.	int
elist_offset	Offset into EVDb for this flight's event list.	Number of bytes from beginning of EVDB map file.	int32
last_pos_message_to	Indicates whether last pos message was a TO.	Not Applicable.	boolean
proposed_altitude_type	Type of altitude: Block, OTP, AVR, etc.	A character.	char
proposed_alt2	Requested altitude 2 filed in FZ.	Hundreds of feet.	int
reported_alt2	Last reported altitude 2.	Hundreds of feet.	int

A.2 Aircraft Dynamics Database

The Aircraft Dynamics Database contains aircraft flight characteristics that are compiled from several sources. The Aircraft Types Database consists of the data contained in the *FAA Handbook 7340.* This document contains general information regarding the classification of aircraft. Refer to Section 3.1.2 for a more detailed discussion regarding aircraft classification. More importantly, the handbook also contains information about the climb and descent rates for each aircraft type listed. This information is utilized by the Flight Modeling Functions to determine the flight route that a particular aircraft will follow. Additional flight dynamics data are compiled in the *Integrated Noise Model.* Aviation experts determined detailed flight characteristics for some 43 common aircraft types to analyze the effects of airport noise in surrounding communities. This study analyzed flight ascent up to 10,000 feet considering trip length and aircraft category. The data was then extracted to span the entire category range of aircraft contained in the *FAA Handbook 7340.* Finally, to model all other flight characteristics that were not included in either of the aforementioned documents, the book *Aircraft Performance, Selection, and Design* was used to determine ascent profiles up to the absolute ceiling and the descent profiles from absolute ceiling [19]. The resultant descent data are given below.

Table A-5: Aircraft Dynamics for Descent Profiles[10]

Aircraft Types	Initial Descent (Mach/IAS)	Speed Leaving 12,000 feet (IAS)	Speed 10 Miles Out (IAS)	Landing Speed (IAS)
Heavy Jets, Fighters	.85/350 (kts) .80/330 .75/310 .70/290	250 (kts) 250 250 250	140 (kts) 140 140 140	140 (kts) 140 140 140
Large Jets	.85/350 .80/330 .75/310 .70/290	250 250 250 250	120 120 120 120	120 120 120 120
Turboprops, Piston props, Helicopters	.70/290	250	90	90

---

[10] Ibid.

A.3 Profile Analysis Results

Table A-6: Comparison of Actual Data and Descent Profile (Group 1)[11]

Event Name	Latitude/ Longitude	Modeled Altitude	Num. Misses	Num. Hits	Altitude Ave	Error Std	Distance Ave	Error Std
J1	03539N/11959W	35,000	47	0	0.0	0.0	0.000	0.000
DERBB	03515N/11959W	31,100	47	0	0.0	0.0	0.000	0.000
ZLA26	03453N/11917W	21,600	47	0	0.0	0.0	0.000	0.000
ZLA13	03453N/11917W	21,600	47	0	0.0	0.0	0.000	0.000
J1	03443N/11913W	18,400	43	4	2,907.5	1,297.7	5.325	0.836
REYES	03439N/11908W	16,500	39	8	3,893.8	1,823.0	1.384	1.354
V107	03439N/11908W	16,500	39	8	3,893.8	1,823.0	1.384	1.354
PIRUE	03430N/11900W	12,000	5	42	6,086.2	1,618.6	1.413	1.380
FIM	03421N/11853W	12,000	0	47	2,403.6	1,919.0	0.534	0.689
GINNA	03413N/11850W	10,200	0	47	2,331.5	769.7	0.552	0.799
ZLA13	03413N/11950W	10,200	0	47	2,331.5	769.7	0.552	0.799
SYMON	03410N/11849W	8,300	0	47	3,603.4	758.2	0.843	0.844
ZLA14	03410N/11849W	8,300	0	47	3,603.4	758.2	0.843	0.844
SADDE	03402N/11846W	7,000	0	47	3,406.0	680.4	1.814	1.110
ZLA14	03402N/11846W	7,000	0	47	3,406.0	680.4	1.814	1.110
ZLALA	03402N/11846W	7,000	0	47	3,406.0	680.4	1.814	1.110
LAX	03356N/11826W	1,300	0	47	-513.4	156.1	4.005	0.520
V107	03356N/11826W	1,300	0	47	-513.4	156.1	4.005	0.520
ZLALA	03357N/11824W	0	0	47	786.6	156.1	2.156	0.301
LAX	03357N/11824W	0	0	47	786.6	156.1	2.156	0.301

Notes: Data were compiled from Large Turbojets that came within 6 nautical miles of the event location. Modeled Altitude was obtained using the proposed descent profile with a 3-degree descent slope and a level-out altitude of 12,000 feet. Altitudes are in feet and distance values are in nautical miles. A positive altitude error indicates that the actual recorded altitude was higher than the modeled altitude at that event whereas a negative altitude error indicates that the actual altitude was lower than the modeled altitude at that event.

---

[11] Adapted from [33].

Table A-7: Comparison of Actual Data and Descent Profile (Group 2)[12]

Event Name	Latitude/ Longitude	Modeled Altitude	Num. Misses	Num. Hits	Altitude Ave	Error Std	Distance Ave	Error Std
J1	03539N/11959W	35,000	47	0	0.0	0.0	0.000	0.000
DERBB	03515N/11959W	31,100	47	0	0.0	0.0	0.000	0.000
ZLA26	03453N/11917W	21,600	47	0	0.0	0.0	0.000	0.000
ZLA13	03453N/11917W	21,600	47	0	0.0	0.0	0.000	0.000
J1	03443N/11913W	18,400	41	6	3,846.7	1,042.1	4.556	0.413
REYES	03439N/11908W	16,500	37	10	5,055.0	1,120.0	1.923	1.570
V107	03439N/11908W	16,500	37	10	5,055.0	1,120.0	1.923	1.570
PIRUE	03430N/11900W	12,000	10	37	6,731.4	1,598.5	2.017	1.899
FIM	03421N/11853W	12,000	0	47	3,485.3	1,288.8	0.499	0.815
GINNA	03413N/11850W	10,200	0	47	2,828.7	997.8	0.760	0.986
ZLA13	03413N/11950W	10,200	0	47	2,828.7	997.8	0.760	0.986
SYMON	03410N/11849W	8,300	0	47	4,058.9	972.8	1.070	1.197
ZLA14	03410N/11849W	8,300	0	47	4,058.9	972.8	1.070	1.197
SADDE	03402N/11846W	7,000	1	46	3,638.0	583.9	1.986	1.051
ZLA14	03402N/11846W	7,000	1	46	3,638.0	583.9	1.986	1.051
ZLALA	03402N/11846W	7,000	1	46	3,638.0	583.9	1.986	1.051
LAX	03356N/11826W	1,300	0	47	-545.7	155.8	3.984	0.332
V107	03356N/11826W	1,300	0	47	-545.7	155.8	3.984	0.332
ZLALA	03357N/11824W	0	0	47	754.3	155.8	2.119	0.314
LAX	03357N/11824W	0	0	47	754.3	155.8	2.119	0.314

Notes: Data were compiled from Large Turbojets that came within 6 nautical miles of the event location. Modeled Altitude was obtained using the proposed descent profile with a 3-degree descent slope and a level-out altitude of 12,000 feet. Altitudes are in feet and distance values are in nautical miles. A positive altitude error indicates that the actual recorded altitude was higher than the modeled altitude at that event whereas a negative altitude error indicates that the actual altitude was lower than the modeled altitude at that event.

---

[12] Ibid.

Table A-8: Comparison of Actual Data and Descent Profile (Group 2)[13]

Event Name	Latitude/ Longitude	Modeled Altitude	Num. Misses	Num. Hits	Altitude Ave	Error Std	Distance Ave	Error Std
J1	03539N/11959W	35,000	26	0	0.0	0.0	0.000	0.000
DERBB	03515N/11959W	31,100	26	0	0.0	0.0	0.000	0.000
ZLA26	03453N/11917W	21,600	26	0	0.0	0.0	0.000	0.000
ZLA13	03453N/11917W	21,600	26	0	0.0	0.0	0.000	0.000
J1	03443N/11913W	18,400	25	1	2,870.0	0.0	0.411	0.000
REYES	03439N/11908W	16,500	19	7	5,542.9	1,525.0	2.606	2.070
V107	03439N/11908W	16,500	19	7	5,542.9	1,525.0	2.606	2.070
PIRUE	03430N/11900W	12,000	1	25	6,750.8	1,264.2	1.523	1.842
FIM	03421N/11853W	12,000	0	26	3,282.3	1,135.4	0.418	0.662
GINNA	03413N/11850W	10,200	0	26	2,627.7	906.7	0.502	0.686
ZLA13	03413N/11950W	10,200	0	26	2,627.7	906.7	0.502	0.686
SYMON	03410N/11849W	8,300	0	26	3,818.8	755.8	0.735	0.857
ZLA14	03410N/11849W	8,300	0	26	3,818.8	755.8	0.735	0.857
SADDE	03402N/11846W	7,000	0	26	3,258.8	1,537.2	2.241	1.044
ZLA14	03402N/11846W	7,000	0	26	3,258.8	1,537.2	2.241	1.044
ZLALA	03402N/11846W	7,000	0	26	3,258.8	1,537.2	2.241	1.044
LAX	03356N/11826W	1,300	0	26	-111.9	1,706.0	3.996	0.315
V107	03356N/11826W	1,300	0	26	-111.9	1,706.0	3.996	0.315
ZLALA	03357N/11824W	0	0	26	733.1	164.3	2.102	0.370
LAX	03357N/11824W	0	0	26	733.1	164.3	2.102	0.370

Notes: Data were compiled from Large Turbojets that came within 6 nautical miles of the event location. Modeled Altitude was obtained using the proposed descent profile with a 3-degree descent slope and a level-out altitude of 12,000 feet. Altitudes are in feet and distance values are in nautical miles. A positive altitude error indicates that the actual recorded altitude was higher than the modeled altitude at that event whereas a negative altitude error indicates that the actual altitude was lower than the modeled altitude at that event.

---

[13] Ibid.

A.4 Probabilistic Profile Estimate Results

Table A-9: Flight Event Altitude Ranges

Event Name	Actual Mean Altitude	Altitude Std	Altitude High est.	Altitude Low est.	Modeled Altitude
J1	NA	NA	NA	NA	35,000
DERBB	NA	NA	NA	NA	31,100
ZLA26	NA	NA	NA	NA	21,600
ZLA13	NA	NA	NA	NA	21,600
J1	21,816	1,040.3	22,856.7	20,776.1	18,400
REYES	21,320	1,458.4	22,778.4	19,861.7	16,500
V107	21,320	1,458.4	22,778.4	19,861.7	16,500
PIRUE	18,476	1,526.3	20,001.8	16,949.2	12,000
FIM	15,018	1,502.4	16,520.0	13,515.3	12,000
GINNA	12,790	888.7	13,679.1	11,901.7	10,200
ZLA13	12,790	888.7	13,679.1	11,901.7	10,200
SYMON	12,128	841.7	12,970.2	11,286.7	8,300
ZLA14	12,128	841.7	12,970.2	11,286.7	8,300
SADDE	10,465	828.2	11,293.2	9,636.7	7,000
ZLA14	10,465	828.2	11,293.2	9,636.7	7,000
ZLALA	10,465	828.2	11,293.2	9,636.7	7,000
LAX	861	491.8	1,352.7	369.1	1,300
V107	861	491.8	1,352.7	369.1	1,300
ZLALA	762	157.8	920.1	604.6	0
LAX	762	157.8	920.1	604.6	0

Notes: Data were compiled from Large Turbojets that came within 6 nautical miles of the event location. Modeled Altitude was obtained using the proposed descent profile with a 3-degree descent slope and a level-out altitude of 12,000 feet. Altitudes are in feet. The altitude range was found by adding or subtracting one standard deviation from the mean. Thus, the probability that a flight will be between the Altitude Low estimate and Altitude High estimate is 68.2%.

THESIS PROCESSING SLIP

FIXED FIELD: ill. ____________ name ____________

index ____________ biblio ____________

▶ COPIES: [✓ Archives] Aero Dewey [✓ Eng] Hum

Lindgren Music Rotch Science

TITLE VARIES: ▶ ☐ ____________

NAME VARIES: ▶ ☐ ____________

IMPRINT: (COPYRIGHT) ____________

▶ COLLATION: ____________

▶ ADD: DEGREE: ________ ▶ DEPT.: ____________

SUPERVISORS: ____________

NOTES:

cat'r date

page

▶ DEPT: ____________ ▶

▶ YEAR: ____________ ▶ DEGREE: ____________

▶ NAME: ____________