REQUIREMENTS OF THE NAVY'S
TOMAHAWK THEATER MISSION PLANNING SYSTEM
RELATING TO OBJECT-ORIENTED TECHNOLOGY

by

John D. Bozarth

Major paper submitted to the Faculty of the Virginia Polytechnic
Institute and State University in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

in

Systems Engineering

APPROVED:

[Signatures]

Dr. Edward A. Fox
Chairman

Professor Benjamin S. Blanchard

Dr. Bruce R. Copeland

November 1992

Blacksburg, Virginia
c.2

LD
5655
V851
1992
B693
c.2
REQUIREMENTS OF THE NAVY'S
TOMAHAWK THEATER MISSION PLANNING SYSTEM
RELATING TO OBJECT-ORIENTED TECHNOLOGY

by

John D. Bozarth

Committee Chairman: Dr. Edward A. Fox
Computer Science

(ABSTRACT)

It was discovered during the Gulf war that the current time required to plan a Tomahawk Land Attack Missile (TLAM) mission was too long and that the current mission planning system design was limited in its capabilities.

The possibility of incorporating object-oriented technology into the TLAM Planning System (TPS) was investigated in order to reduce the time required to plan a particular mission and to increase the capabilities of mission planning.

The current time to plan a Tomahawk mission is approximately over three hours. Utilizing Object-Oriented Technology (OOT) within the TPS will reduce this time significantly. OOT also allows for the use of complex data transactions and data types such as voice, video, graphics, image and text. Utilizing complex data types in mission planning will increase mission capabilities and performance.
Results of this investigation indicated that the best way to begin to include OOT into mission planning would be to use both a relational and an object-oriented DBMS server. This approach would begin to include OOT into mission planning and at the same time it would allow those databases that are more suited for a relational format to remain within the relational DBMS.
## TABLE OF CONTENTS

LIST OF ILLUSTRATIONS ........................................ vi

TABLE LIST .................................................. vii

1. INTRODUCTION ............................................ 1
   1.1 Tomahawk Weapon System Description .................. 3
   1.2 Theater Mission Planning System Overview .......... 4
      1.2.1 Function and Capability ....................... 7
      1.2.2 Hardware and Software ....................... 11
      1.2.3 Typical Planning Session ................. 12
   1.3 TPS Data Bases ...................................... 15
      1.3.1 Geographical Data Bases ................... 16
      1.3.2 Defense Data Bases ....................... 21
      1.3.3 Terrain Data Bases ....................... 25
      1.3.4 Mission Data Bases ....................... 31
   1.4 Object-Oriented Database Management Background
      Material ........................................... 34
      1.4.1 OODB Features ............................. 35
      1.4.2 ObjectStore ................................ 37

2. PROBLEM DEFINITION ..................................... 39
   2.1 Shortcomings of Existing TMP System ............ 39
      2.1.1 INFORMIX On-Line .......................... 39
      2.1.2 Constraints ............................... 41
   2.2 Need for OOT Inclusion in TMP ................... 42
      2.2.1 Object-Oriented Data Base Design .......... 42
      2.2.2 Advantages/Disadvantages ............... 43

3. SYSTEM REQUIREMENTS .................................. 47
   3.1 Mission Definition ............................... 47
   3.2 TPS Use Requirements ............................ 47
   3.3 OODBMS Requirements ............................. 49
   3.4 Safety Requirement .............................. 51
   3.5 Security Requirements ........................... 51
   3.6 Reliability Requirements ....................... 51
   3.7 Maintainability Requirements ................... 52
   3.8 Configuration Management Requirements .......... 53
   3.9 Personnel Requirements .......................... 53

4. DESIGN FOR THE TPS UTILIZING OOT ................. 55
   4.1 ERDBMS Approach .................................. 55
      4.1.1 Advantages/Disadvantages .................. 57
      4.1.2 Major Design Issues ...................... 58
      4.1.3 Feasibility ................................ 59
   4.2 RDBMS and OODBMS Approach ..................... 59
      4.2.1 Advantages/Disadvantages .................. 61
LIST OF ILLUSTRATIONS

Figure 1 TMPC High-Level System Configuration ................. 2
Figure 2 VL/T Configuration ........................................ 5
Figure 3 ABL Configuration .......................................... 6
Figure 4 TLAM Planning System (TPS) Functional Configuration ... 8
Figure 5 Mission Planning and Execution ......................... 13
Figure 6 Geographic Data Bases .................................... 17
Figure 7 Defense Data Bases ........................................ 22
Figure 8 Terrain Data Bases ......................................... 26
Figure 9 Mission Data Bases ........................................ 33
Figure 10 ERDBMS Architecture .................................. 56
Figure 11 RDBMS and OODBMS Architecture ...................... 60
Figure 12 Geographic DB Grid Example ............................ 65
Figure 13 Class Definitions .......................................... 66
<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table I</td>
<td>54</td>
</tr>
<tr>
<td>Table B-1</td>
<td>77</td>
</tr>
<tr>
<td>Table B-2</td>
<td>78</td>
</tr>
<tr>
<td>Table B-3</td>
<td>79</td>
</tr>
<tr>
<td>Table B-4</td>
<td>80</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

Tomahawk Theater Mission Planning (TMP) provides the United States Navy with the capability to develop pre-planned missions for the Tomahawk Land Attack Missile (TLAM) either at a land based Theater Mission Planning Center (TMPC) or at sea on an Afloat Planning System (APS). These pre-planned missions are used by the Tomahawk Cruise Missile Guidance Set (CMGS) to guide a Tomahawk cruise missile to its intended target. The TMPC and APS are currently being upgraded to accommodate the evolving requirements of the Tomahawk program. One of the changes being considered by the Navy for the upgrade is to incorporate Object-Oriented Technology (OOT).

The need for the inclusion of OOT into the TLAM Planning System (TPS) evolved from analysis that was done after the Gulf War on the use of the Tomahawk Weapon System (TWS) during the War. The analysis indicated that it was impossible to update data bases, re-plan missions and get the new missions to the fleet quickly. Results indicated that even though the TWS was effective during the Gulf War, there was room for improvement in mission planning time and logistics. The purpose of this project will be to provide an approach for the inclusion of OOT into the TPS for the TMPC and APS upgrades.

Both the TMPC and the APS are composed of three subsystems: the TPS, the Digital Imagery Workstation Suite (DIWS) and the Mission Data Distribution System (MDDS). The functionally equivalent Tactical Data Distribution
Figure 1 TMPC High-Level System Configuration

System (TDDS) replaces the MDDS on the APS. Figure 1 contains a high-level depiction of the TMPC system configuration. The TPS is the subsystem that maintains the General Service (GENSER) classified data bases and conducts route planning. The DIWS processes digital imagery in support of mission planning and maintains the Sensitive Compartmented Information (SCI) data bases. The MDDS and TDDS are the accounting agents for TMPC and APS respectively. The major difference is that TDDS will not distribute the Data Transport Devices (DTDs) on which the missions are stored. Communication links connect the three subsystems so that taskings and data can be passed between them. To better understand the purpose, significance and functions of the TMPC and APS, a brief description of the Tomahawk Weapon System (TWS) is required.

1.1 Tomahawk Weapon System Description

The Tomahawk Weapon System (TWS) for the current fielded configurations comprises hardware and software that provide for status monitoring, launch preparation, and launch execution of missiles from two types of launching systems: the Armored Box Launcher (ABL) and the Vertical Launching System (VLS). The ABL supports the AN/SWG-2A, destroyer and cruiser platforms, while the VLS supports the AN/SWG-3A and AN/SWG-3B cruiser platforms. The TWS is structured around a Track Control Group (TCG) and Launch Control Group (LCG). Figures (2) and (3) present a view of the LCG's for both the VLS and ABL configurations.
The TCG has one primary processor with another identical processor assigned to function as a ready spare for backup capability. This computer is known as the secondary processor for the Track Data Processing Set (TDPS). The TCG allows the operator to create the Over-The-Water (OTW) trajectory path of the missile's flight. This data then gets sent to the LCG where it is appended onto the front end of the mission data that the LCG maintains.

The LCG comprises the other side of the TWS. In similar fashion to that of the TCG, the LCG has a primary processor called the Central DPS (CDPS) and a dedicated backup processor known as the Dedicated DPS (DDPS). The LCG is the single element where warhead prearming, formatting, transmission of missile mission and alignment data and missile launch are initiated.

1.2 Theater Mission Planning System Overview

TMP is performed by the Cruise Missile Support Activities (CMSAs) either at the shore based TMPCs or on an APS. Currently, there are two land bases within the United States. One is located in Honolulu, Hawaii and the other is located in Norfolk, Virginia. The APS is a ship based version of the TMPC's and is still undergoing laboratory integration testing. The primary purpose of the CMSA and the APS is to prepare large numbers of missions to support theater-level contingency/operational plans. In the case of the TWS, the TPS prepares the missions that the missiles will be programmed to fly. Figure (4) (TLAM Planning System
Launch Control Group

OIDT - Operator Interactive Display Terminal
DSCC - Data Storage Control Center
RASS - Random Access Storage Set
VLS - Vertical Launching System

Figure 2 VL/T Configuration
Launch Control Group

Graphics Control Center

OIDT #1

DSCC
RASS

OIDT #2

Launch Processing Control Center

To ABL

OIDT - Operator Interactive Display Terminal
DSCC - Data Storage Control Center
RASS - Random Access Storage Set
ABL - Armored Box Launcher

Figure 3 ABL Configuration
(TPS) Functional Configuration) illustrates the high-level functions of the TPS. There are four main components of the TPS: the analysis modules, the Mission Data Preparation System (MDPS), the Data Preparation and Maintenance (DPM) system and the Interactive Route and Development Analysis (IRDA) system. The analysis modules provide interactive capabilities to increase data integrity, the MDPS incorporates data into defined missions and also performs data checks, the DPM performs all accesses of data bases and the IRDA performs all the remaining TPS functions. Note in Figure 4 that once a mission is approved it is place into a separate data base by the MDPS.

Once a mission is developed by TPS, it is passed to the MDDS. The MDDS is responsible for the Configuration Management (CM) of the missions and for installing the missions on the DTDs. The TWS will read a particular mission off of the DTD during launch sequence processing. Once the missions are installed, the DTD is passed onto the DTD Certification Processor (DCP). The DCP's purpose is to certify that the missions were installed properly and without any corruption of data.

1.2.1 Function and Capability

The TPS is designed to assist a planner in preparing preplanned missions for the Tomahawk Sea Launched Cruise Missile (SLCM). Many factors are taken into account during this process. They include national policy, military tactics, weapon performance, resource availability and coordination requirements. The missions which TPS plans include both
**Figure 4 TLAM Planning System (TIPS) Functional Configuration**
nuclear and conventional land-attack Tomahawk missiles. In addition to the actual missions which are produced, TPS also provides command and control documentation for the Tomahawk "platforms".

Missions for the TLAM are a composite of two segments: an overland segment and an overwater segment. The overland segment of the mission is the part which is planned by TPS. The overwater segment is prepared by an individual on the launch platform designated the Track Operator (TO). The remainder of this discussion and the conceptual data base design do not address the overwater portion of the mission. The overland mission segments are stored on a mass storage device, such as the current DTD, or on a Removable Interchangeable Media Module (RIMM) which is planned for use with Mass Media Storage Device/Tomahawk (MMSD/T), during the Advanced Tomahawk Weapon Control System (ATWCS) upgrade. These mass storage devices interface with the TWCS and contain a mission for every possible target in a platform's intended operating area.

Mission planning specifies the ground track and vertical profile of a cruise missile mission. Various constraints may be imposed on the intended route because of physical limitations of the missile and its guidance system, navigational aids (Terrain Contour Matching (TERCOM) maps, Digital Scene Matching Area Correlator (DSMAC) scenes), characteristics of the terrain the missile must cross, locations and types

---

2 Platform - Either surface ship, submarine or aircraft.
of enemy defenses, areas to be avoided (for example cities), and other limitations. The most difficult part of creating a route (ground track and elevation profile) is evaluating trade-offs of performance versus probability of success (i.e., arriving at the intended target) in light of these constraints. TPS provides not only a convenient means of specifying the ground track and vertical profile, but also allows the planner to perform trade-offs of performance versus probability of success using the analysis modules.

The planner at the Mission Planning Center (MPC) lays out the route from the nominal launch area to the target, considering the prescribed tactics, inherent constraints, and mission objectives. This is done by laying out a series of straight-line route segments from launch to target. These segments are identified by waypoints, a set of latitude and longitude coordinates from which and toward which a missile flies. Another way to define a waypoint is that it is the intersection of two straight-line route segments. The missile will not fly directly over waypoints but begins turning prior to reaching them so that when the turn is complete the missile will roll out (wings level) on the next route segment and will be heading toward the next waypoint. Waypoints are selected to define the route from launch to target using both planner experience and external data (terrain elevation data, threat locations, vertical obstructions, etc.). The missile guidance system must be updated to correct for inherent drift in the navigation system using either TERCOM or DSMAC if the required accuracy at the target is to be achieved.
TERCOM maps, that hold terrain elevation data describing a geographic area on the earth, can be created with different granularity or sizes. The size of the area is determined by the expected drift of the navigation system. Usually, to minimize computer storage requirements on the missile, larger maps are used which have a greater granularity. The granularity of the terrain elevation data determines the accuracy of the navigation system update. In addition, the size and the time (range flown) since the last update are strong factors in establishing whether the map will actually be overflown by the missile.

The DSMAC is a feature applicable only to conventional land-attack Tomahawk missiles. DSMAC provides navigation updates to the missile guidance system for the terminal guidance phase. DSMAC provides the capability to acquire camera scene images of the ground directly below the missile at pre-determined geographical coordinates during the missile flight, and to compare those scenes with stored reference scene data. The resulting correlation will be a determination of downrange and crossrange position errors. The guidance set will then compensate for these errors to achieve greater accuracy.

1.2.2 Hardware and Software
The TPS is hosted on a system built around dual VAX 11/785 processors. There are three specially designed planning stations called Dedicated Planning Terminals (DPTs). Each DPT has both an AYDIN Controls Model 8026 high resolution 19-inch color graphics terminal (for display of graphical
data), an AYDIN Controls Model 8037 high resolution 19-inch monochrome alphanumeric terminal (for output of alphanumeric data), an AYDIN Controls Model 5116 Keyboard, a function keyboard with backlighting capabilities for selected function keys, an AYDIN Controls Model 5576 trackball, and a Tektronix 4632 video/hard copy unit. These DPTs allow three mission planners to simultaneously plan independent missions.

The TPS operates in a UNIX environment. A commercial Relational Data Base Management System (RDBMS), INFORMIX, is called by the TPS applications software to process data base operations. Approximately two-thirds of the TPS applications software is written in Ada, with the bulk of the remainder being Fortran and INFORMIX Structured Query Language (SQL). The TPS data bases primarily reside on magnetic disk drives.

At a high-level view, the Mission Planning Center (MPC) can be separated into two major parts: one part containing the TPS and the second part containing the MDDS, DCP and the Command and Control Launch Packages (CCLPs). Please refer to Figure 5. The TPS receives its assignments, plans the missions and then sends the planned missions to the MDDS. The MDDS builds the disks with the preplanned missions, Maps and Operational Flight System (OFS) and distributes the disks to the fleet.

1.2.3 Typical Planning Session

Up to three planners may plan independent missions simultaneously. Each planner may only plan one mission at a time. A typical planning session
Figure 5 Mission Planning and Execution

consists of system log on, mission definition, route editing, and termination. Mission definition may be accomplished in one of three ways.

1. Define a new mission by specifying a mission name, a launch point, a vehicle type (missile type) and a Designated Ground Zero (DGZ).
2. Recall an existing mission from the missions data base by specifying the mission name.
3. Recall the last mission worked on by using the autofile restart option.

Once the launch point and target are defined through mission definition, the planner must specify the route the missile is to fly. While there are a number of methods for planning a cruise missile route, a common practice is to layout an initial path in the horizontal plane, refine the route using the various horizontal displays and analysis results and fine tune the route by adjusting the vertical profile. The TPS assists the planner in defining the route by providing displays showing information contained in the various data bases, such as available TERCOM map sets, DSMAC scenes, defenses, and terrain information. The planner is further assisted in defining the route by four analysis modules; the Performance Analysis Module (PAM), the Navigation Accuracy Module (NAM), the Clobber Analysis Module (CAM) and the Defense System Analysis Module (DSAM). Once the planner is satisfied with the ground track and vertical profile of a mission, it is submitted to the Off-Line Verification Process (OLVP) for detailed analysis and verification. OLVP runs more comprehensive versions of PAM, NAM, CAM and DSAM, and generates three reports; the mission, the
action point and, if requested by the operator, the vertical obstacles summary reports. After the mission has been verified, it may be submitted to the Mission Data Preparation System (MDPS). MDPS will incorporate the necessary information from the various data bases, such as TERCOM map data, and will perform data format, data range and data completeness checks. Additionally, a flight simulation will be performed if specified by the planner.

The planner has two methods for specifying most of the information pertaining to the missile route. Data may be entered through the alphanumeric keyboard, or the cursor on a particular display may be positioned through the use of a trackball to select a waypoint or map, or to specify the vertical flight path. The planner may also request a checkpoint to be taken at any point in the planning process. A checkpoint is a "snapshot" at a current point in the planning process from which the operator can restart the planning process. This allows the planner to try various options to achieve the route with the greatest probability of success.

1.3 TPS Data Bases

TPS data bases can be divided into three types depending on their source: primary, derived and support. Primary data bases are mission planning related and are not dependent upon any other TPS data base for their

---

4 Snapshot - Current State of the system.
creation. Derived data bases are mission planning related, and are produced from primary data bases before, during, or as a result of mission planning activities. Support data bases are not mission planning related and tend to be system administrative in nature. For this project the data bases were divided into four areas of concentration: geographical, defense, terrain and mission. Appendix B contains a table of each area of concentration defining the acronym, the file structure (INFORMIX or Binary-Direct-Access (BDA)), the type and the security classification.

1.3.1 Geographical Data Bases
The following data bases portray geographical information for the planner. These data bases are primarily used for graphics displays to aid the planner. The total size of these data bases is approximately 250MB.

1. The Geography Data Base is extracted from the Central Intelligence Agency's (CIA's) World Data Bank II. It provides river, lake, coastline, and political boundary information. It is displayed concurrent with other data to provide mission planners with a frame of reference.

2. The Geographic Boundaries (GEO BNDS) Data Base is designed to augment the geography data base and consists of labeled boundary segments that are input and/or updated by the Data Base Administrator (DBA).

3. The City Data Base, produced by the Defense Mapping Agency (DMA), provides boundary information on cities. City outlines are displayed on the TPS terminal during mission planning to alert the
planner of their location. The Cities data base consists of two parts: a static display file and an INFORMIX RDBMS data base file to contain updates. The Cities display file contains only source data from the Defense Mapping Agency (DMA), while the INFORMIX RDBMS Cities file contains cities entered by the user (the DBA). Cities are stored by name with latitude and longitude coordinate pairs.

4. Vertical Obstruction Data Base (VODB) is produced and provided by the DMA. It can also be produced by the DIWS. The Vertical Obstruction data base contains elevation and location point, line, and area vertical obstructions (mostly manmade) that the missile may encounter in flight. These features are maintained because they will not show up in digital terrain data such as Digital Terrain Elevation Data (DTED). VODB is used for display and calculation of possible collisions (clobber).

5. The Prohibited Areas Data Base (PADB) is unique to each user site. Prohibited Areas are determined by the U.S. Commander In Chief (USCINC) for each CMSA. PADB is used to define restricted flight areas or corridors.

6. The Launch Points Data Base (LPDB) defines the locations of the launch sites, and the initial launch site parameters. Launch points are determined by each CMSA and are input by the user (the DBA). Launch points are used only in simulations and are not included in actual missions that are sent to the fleet.

7. The Launch Areas Of Interest Data Base (LAIDB) contains geographical areas for mission planning from which actual
operational missile launches could be made. Launch areas are determined by the USCINC responsible for each CMSA.

8. The Digital Feature Analysis Data (DFAD) Data Base contains terrain and manmade features of importance to the military (key roads, buildings, utility right-of-ways, vehicular obstructions, etc.). These features are stored in latitude and longitude coordinate pairs and attribute codes. DFAD is an input to the derivation of Threat Site Characteristics (TSC).

9. The Point Positional Data Base (PPDB) is produced by the Defense Mapping Agency and consists of stereo aerial photography pairs marked with geodetic control points. Geodetic control points are easily identified features of known latitude, longitude and elevation. The PPDB are used to establish precise positions for features of interest and are also used to extend geodetic control to uncontrolled imagery. Incoming source PPDBs are received by the DIWS, which sends the data to the TPS.

10. The Environment Data Base (EDB) is distributed by the United States Air Force (USAF) Environmental Technical Analysis Center (USAFTAC). Environmental data consists of averaged annual and monthly wind and temperature data. The EDB is static in nature, with no scheduled updates. It is used in the derivation of Autorouter Data Dangerspace ARD(DANGER) data.

11. The Imagery Cover Data Base (IMGCOV) is an inventory file of all the imagery available within the TMPC or APS system. The TPS receives information from the DIWS to update IMGCOV, since the DIWS
is responsible for receiving and cataloging incoming imagery. Imagery types within the data base include: national and tactical reconnaissance imagery, PPDB and multispectral imagery. The TPS does not store the actual image but does store the following data for each image:

a. Imagery ID alias (from DIWS)
b. Basic encyclopedia number
c. Image geometric descriptors
d. A list of the TMPC products the image can support
e. Acquisition parameters

12. The Operational Areas (OP AREAS) Data Base contains regions of the world divided up by theaters of operations. Each TMPC and APS has specific operational areas assigned to it for mission planning purposes. Each OP AREA will have a name, dangerspace ARD(DANGER) cell size, four latitude and longitude coordinate pairs defining area limits, ten altitudes, priority, area active status, autorouter active and derived data statuses.

13. The Geographic Search Capability (GSC) Data Base contains the GSC Ids, hierarchical characteristics, and spatial composition of all geo-searchable data bases. The GSC acts as an index to allow rapid, multiple searches on all objects within the following data bases

a. DDB                      Defenses Data Base
b. MHF                      Mission History File
c. SS_AREA                  Scene Suitability Area
d. LPDB      Launch Points Data Base  

e. TERCOM    Terrain Comparison map directory  

f. IMGCOV    Imagery Coverage  

g. PPDB      Point Positional Data Base  

h. GEO BNDS  Geographic Boundaries  

i. DSMAC     Digital Scene Matching Area Correlator  

j. PADB      Prohibited Areas Data Base  

k. TGTC      Target Complex  

The GSC algorithm uses three algorithms to ascertain whether or not there is any data in an area: search by sector, search by Minimum Bounding Rectangle and search by MBRs.  

14. The Probabilistic Vertical Obstruction Data Base (PVODB) is produced by the DMA. PVOD differs from the VODB since it does not contain each and every vertical obstruction, but delineates areas with high vertical obstruction possibilities for the mission planner to avoid.  

15. The Image History File (IHF) Data Base contains a record for each image received by the DIWS. Data within the record includes: creation date, image ID, on-line/off-line status and date received. The DIWS sends IHF data to the TPS for storage.  

1.3.2 Defense Data Bases  

1. The Defense Data Base (DDB) contains files that describe locations and types of defenses. The Air Order of Battle (AOB), Missile Order of Battle (MOB), and Radar Order of Battle (ROB) files
are maintained within the DDB and are collectively referred to as Fixed Defenses (FD). The Tactical Surface to Air Missile (SAM) Order of Battle (TOB) and Polygon files are also maintained within the DDB and are referred to as Area Defenses (ADs). The AOB, ROB, MOB and TOB are obtained from the Joint Services Theater Planning System (JSTPS). The total size of these data bases is approximately 30MB.

2. The Defense Systems Characteristics and Capabilities (DSCC) Data Base is maintained from information provided by JSTPS. The DSCC data base defines the characteristics and performance of fixed SAMs, airborne interceptors, Ground Control Intercept (GCI), tactical SAMs, polygon defenses and radar cross-section profiles. There is one DSCC file for each type of cruise missile.

3. The Scene Suitability Area (SS_AREA) Data Base contains data that is used in the evaluation of terrain for DSMAC maps. Scene Suitability is performed by the DIWS for terminal route areas.

4. The Scene Suitability Cell (SS_CELL) Data Base is another terminal area product that is produced by the TPS as input to the autorouter program, which requires regularly gridded data. Each cell is assigned a numerical weight based on the Digital Terrain Elevation Data (DTED) roughness, the presence of imagery, the suitability of the imagery and the presence of DSMAC scenes within the cell.

5. The Fighter Launch Acceptability Range (FLAR) Data Base is produced on the TPS by the Threat Model Trajectory Analysis (TMTA)
program. FLAR data consists of aspect angles and effective aircraft attack standoff ranges for specified enemy fighter types. Neither FLAR data nor FLAR data base coverage can be displayed on the TPS workstation.

6. The Radar Signal/Noise Interference Ratio (Radar_S/I) Data Base is produced on the TPS by the Threat Model ALARM (TM(ALARM)) program. Radar_S/I data is generated for the Tomahawk onboard radar system, and includes signal/interference ratios based on radar type, range, altitude, clutter and roughness coefficients.

7. The Surface To Air Missile Time Of Flight (SAM(TOF)) Data Base is calculated from defenses data by the Threat Model Enhanced SAM simulation (TM(ESAMS)) program. SAM(TOF) data consists of flight times from SAM launch to Tomahawk interception, and are calculated for all known sites based on SAM type.

8. The Surface To Air Probability Of Kill (SAM(PK)) Data Base is also calculated from defenses data by the TM(ESAMS) program. TM(ESAMS) outputs the probability of Tomahawk kill for a specific SAM for all combinations of clutter and roughness levels at several different altitudes.

9. The Near-Real-Time Threats (NRTT) Data Base is currently under design by the Navy and will provide data to enhance the DDB. Defenses data will be transferred from the MDDS or TDDS to the TPS as mission planning activities require it.

10. The Threat Site Characteristics (TSC) Data Base is derived from the DDB and the Digital Feature Analysis Data (DFAD). TSC data
characterizes the physical nature of threat sites and is used as input to calculate roughness clutter. Physical characteristics that might be included in the TSC data base include: ground composition, vegetation description, and building materials used in the site.

11. The Anti-Aircraft Artillery Estimated Hit (AAA_EH) Data Base produces data from the Threat Model Radar Directed Gun Simulation (TM(RADGUNS)) program. Input data to the simulation includes weapons parameters, clutter, roughness coefficients, altitude or dive angle and crossrange.

1.3.3 Terrain Data Bases

The Terrain data bases provide information based on the elevation of the terrain. They are primarily used for calculation of clobber. The size of the TERCOM data base is about 40MB. The total size of the remaining data bases is approximately 2000MB.

1. The Digitized Terrain Elevation Data (DTED) Data Base is provided by DMA, but can also be generated by the DIWS. The DTED data base provides the mission planner with the terrain detail necessary to produce low altitude missions. DTED consists of terrain elevations (in meters above mean sea level) posted to latitude and longitude grid intersections. The elevations are gridded into cells. Each cell contains terrain elevation data for 1 degree squares of the earth's surface.

2. The DTED Constant Elevation Data (DTED(CE)) Data Base is a subset of the DTED data base. It can also be produced by DMA or the DIWS.
Figure 8 Terrain Data Bases
The DTED(CE) contains DTED cells that are in constant elevation across their entire area (such as water cells) and are stored within the data base as one elevation value, not a matrix of different elevation values.

3. The Terrain Contour Matching (TERCOM) Data Base contains terrain maps that are provided by DMA or produced by the DIWS. The TPS stores the name, location, bearing, and accuracy data for each map in the TERCOM Map Set Header data base. TERCOM maps are based on cell size and differ from DTED data in that elevations are not assigned to the intersections of latitude and longitudes on a grid, but to the actual grid squares or cells. Since a cell is an area, the elevation assigned to it is representative of that area. There are four TERCOM map types: landfall, midcourse, enroute and terminal. TERCOM map coverage can be displayed on the TPS workstation. To accomplish coverage display, there is a TERCOM display file containing map set Ids, selection points and drawing instructions. The TERCOM map data base contains only data pertaining to TERCOM maps.

4. The Radar Terrain Mask Altitude Matrix (RTMAM) Data Base is derived from DTED and defense data. It provides the altitudes for each Ground Controller Intercept (GCI) and strategic SAM installation. This data allows the missile to fly hidden from enemy radar.

5. The Radar Terrain Mask (RTM) Data Base is the display data base for the RTMAM data base. It provides the option to select the
defense type, status (hostile or non-hostile) and terrain masking altitude.

6. The Terrain Contours Data Base (TCDB) is derived from the DTED data base and is displayable on the TPS during mission planning to highlight and accent terrain features.

7. The Digital Terrain Matrix (DTM) Data Base is produced by the DIWS and is stored on the TPS. The data contained within the DTM data base is gridded elevation data. DTED is a DTM with specific grid spacing and azimuth. When new imagery data is received at the DIWS, DTM is updated.

8. The Terminal Terrain Profiles (TTP) Data Base is produced by the DIWS for target areas. The TTP consists of elevations at specific latitude and longitude coordinates. The TPS provides the DIWS with the coordinates in a Terrain Profile Generation Parameter File and the DIWS returns the elevations of those coordinates.

9. The Autorouter Parameter Algorithms (ARP(ALG)) Data Base contains values that are used by the Dynamic Programming Algorithm (DPA) to either accept or reject arcs. The ARP(ALG) parameters include airborne vehicle penalty, LaGrange multipliers, Probability of Survival (Ps) threshold, Probability of Clobber (Pc) and Probability of Kill (Pk) threshold. The ARP(ALG) data base is supplied by McDonnell Douglas Government Aerospace (MDGA).

10. The Autorouter Parameter Accessibilities Quota (ARP(AQ)) Data Base contains accessibility limits from one route object to another for use by the autorouter program. Route objects contain TERCOM and
DSMAC maps. The ARP(AQ) data base is supplied by MDGA.

11. The Autorouter Parameter Clobber (ARP(CLOB)) Data Base contains parameters used by the clobber analysis model in recreating or regenerating Autorouter Data Dangerspace (ARD(DANGER)) from the Autorouter Characterized Terrain (ARD(CHART)) data base. The ARP(CLOB) data base is supplied by MDGA.

12. The Autorouter Parameter TLAM Navigation (ARP(NAV)) Data Base contains Tomahawk missile navigation data used by the autorouter program to calculate the optimal route for a Tomahawk missile from a specified launch point to a specified target. The ARP(NAV) data base is supplied by MDGA.

13. The Autorouter Parameter Switches (ARP(SW)) Data Base contains a set of program options that the user may select for alternate route hypothesis testing. The ARP(SW) data base is supplied by MDGA.

14. The Autorouter Parameter Target Evaluation (ARP(TE)) Data Base contains parameters used in automatic route generation and acceptance. These parameters include: minimum and maximum Height Of Burst (HOB) Above Ground Level (AGL), probability of Tomahawk missile loss (attrition), best route criteria, Minimum Mapset Quality Indicator (MSQI) and MSQI bias values, number of table priorities, arming priority threshold and arming scheme number. The ARP(TE) data base is supplied by MDGA.

15. The Autorouter Parameter Vehicle Data (ARP(VEH)) Data Base contains specific Tomahawk performance specifications which includes
maximum climb rate, missile weight without fuel, maximum fuel weight, nominal fuel flow, minimum and maximum turn radius and target maneuver radius. The ARP(VEH) data base is supplied by MDGA.

16. The Autorouter Parameter Violation Impact Factors (ARP(VIF)) Data base contains defense violation ranges and the penalties (Tomahawk missile damage) at each range for each defense. The ARP(VIF) data base is supplied by MDGA.

17. The Autorouter Data Accessibility (ARD(ACCESS)) Data Base contains all forward reachable nodes from a specific route object for use by the autorouter in both operational areas and terminal area processing. The ARD(ACCESS) data base is derived by the TPS.

18. The Autorouter Data Arcs (ARD(ARCS)) Data Base contains mission specific arc series that specify the best starting and ending points for a mission route based on planning constraints, clobber threat, clobber danger and available route objects. The ARD(ACCESS) data base is derived by the TPS.

19. The Autorouter Data Characterized Terrain (ARD(CHART)) Data Base is derived from DTED. It provides the day type and missile performance data to calculate ARD(CLOB), which then becomes input to ARD(DANGER).

20. The Autorouter Data Dangerspace (ARD(DANGER)) Data Base contains probabilities of Tomahawk missile clobber and attrition from terrain and defenses. ARD(DANGER) is derived by the TPS.

21. The Digital Scene Matching Area Correlator (DSMAC) Data Base terrain maps produced by the DIWS are used for terminal area
navigational updates in conventional missions; nuclear missions do not require DSMAC maps. DSMAC maps are compared by the missile's computer against real-time data collected by its own optical scanner. A DSMAC map is produced by gridding imagery and assigning each grid cell an intensity value based upon the average brightness of the entire cell.

22. The Potential DSMAC Scene Locations (PDSL) Data Base contains potential DSMAC scene locations produced by the DSMAC scene locator program. The program identifies areas with imagery coverage that might support DSMAC map production. The program accomplishes this by evaluating imagery based on scene suitability criteria. The scene suitability data bases are used as inputs to these programs.

23. The Image Products (IMAGE PROD) Data Base contains digital imagery processed by the DIWS in response to TPS tasking. The imagery in the IMAGE PROD data base is displayable on the TPS.

24. The Catalog Data Base contains the media ID, storage shelf location, data type and coverage of each media item received by the TPS. This includes export/import media, DIWS transfers and source media. The Catalog is internal to and maintained by the TPS.

1.3.4 Mission Data Bases

1. The Mission History File (MHF) Data Base contains the missions planned on the TPS. This data base contains approximately 160MB of data. The average mission size is currently around 20KB. It is anticipated that the average mission size will increase to 25-30KB.
because of the DSMAC map upgrades. The MHF holds all the missions produced by each TMPC or APS, with the work and data that went into the production of each one.

2. The Target Complex (TGTC) Data Base is produced from DIWS imagery data and is used by mission planning to determine the terminal target approach route. The TGTC data base contains target complex, target object, and target segment data. Re-evaluation of existing targets is driven by the arrival of new imagery data to the DIWS. The target complex, object, and segment data are displayable on the TPS workstation.

3. The Planning and Implementation Rules (PI RULES) Data Base will contain official mission planning guidance from the USCINC for each mission planning activity. This data base is currently under design.

4. The Activity Journal/Log (ACTIV) Data Base contains a log of TPS activities for traceability purposes. Log entries are drawn from a file of standard activity messages by activity code. The default status for this capability is operational, but the log can be turned off by the user. This data base is also still under design by the Navy.

5. The Global Positioning System (GPS) Data Base will be used only to increase Tomahawk navigation capabilities, since the signals can be blocked. GPS data will be received by the MDDS within the TMPC and the TDDS within APS. The GPS data base will be installed globally.
Figure 9 Mission Data Bases

MISSION DATA BASES

PUBLISHER

SOP

WIF
1.4 Object-Oriented Data Base Management Background Material

The computer industry has seen many generations of data management, starting with indexed files, and later network and hierarchical Data Base Management Systems (DBMSs). More recently, relational DBMSs revolutionized the industry by providing powerful data management capabilities based on a few simple concepts. Now we are on the verge of another generation of data base systems, object-oriented DBMSs. Object-Oriented Data Bases (OODBs) are becoming increasingly popular in the industry. This popularity is attributed to their ability to map real-world objects and their relationships directly to computer representations. Applications such as Computer Aided Design (CAD) and Computer Aided Manufacturing (CAM), Computer Aided Software Engineering (CASE), multimedia, and expert systems require data bases that can support a wide variety of types with the ability to express complex relationships among objects. Object-oriented data bases can meet these requirements, while conventional data bases such as network, hierarchial, and relational are inadequate for these types of applications. The goal of this new type of DBMS is to support a much wider range of data intensive applications in engineering, medicine, and science and in other areas where traditional DBMS's have proven inadequate. In 1990, International Data Corporation, of Framingham, Massachusetts, predicted that "the OODB market would grow from $20 million in 1990 to $270 million by 1994."5 These figures are not unrealistic with the current rate of growth for the computer industry.

---

OODBs are needed to handle complex data relationships and complex data types such as voice, video, graphics, image and text. The market for OODBMS will continue to expand well into the future.

1.4.1 OODB Features

An OODB is a DBMS built specifically to store, retrieve and manipulate objects rather than simple data types. Object data bases also offer numerous features to the user that are unavailable in conventional type data bases. These features include: Objects, Object Identity, Encapsulation, Classes, Inheritance, Multimedia Data, Versioning and Extensibility.

Objects - are entities that are used to represent abstract or concrete real-world things in the application domain. Objects have a state, exhibit behavior and have their own identity. An example might be a battleship in a military application.

Object Identity - refers to the idea that an object has an existence which is independent of its value. This means that two objects can have the same value, but can be distinguished by their identity. Object identifiers in OODBs and record pointers in hierarchial data bases are similar; the major difference is that object identifiers are logical pointers and record pointers are physical pointers.

Encapsulation - simplifies the interaction between objects and prevents data corruption by hiding the internal structure of the object; objects communicate by sending messages to other objects. Objects that are encapsulated help to form the foundation for
modular programming.

**Classes** - A class is a means of assembling objects that share the same structure and behavior. An object is not a class, but a class may be an object.

**Inheritance** - allows the user to derive a new class from an existing class. This new class inherits all the attributes and methods of the original class and may define additional attributes and methods. Inheritance reduces the need to specify redundant information thereby simplifying updates and modifications. It is also used to create objects that are almost like other objects except with a few alterations. Inheritance is important since it makes it possible to declare that certain specifications are shared by multiple parts of a program. This allows programs to be smaller and more tightly organized.

**Multimedia Data** - allows a database to handle Binary Large Objects (BLOBs) such as voice, video or text.

**Versioning** - adds Configuration Management (CM) capabilities to a database. Most OODBMSs provide capabilities to track multiple versions or states of objects.

**Extensibility** - allows the addition of new classes of objects into the database hierarchy that can be used and managed in the same manner as the classes in the original hierarchy.

In addition to these OODB features to support complex data, most OODBMSs provide close integration with Object-Oriented Programming Languages (OOPL), such as C++ or Smalltalk.
1.4.2 ObjectStore

Choosing a DBMS is a long-term investment. New hardware can be obsolete within a year, while software applications typically have a life cycle of five or more years. According to Dr. Bruce Copeland at the Naval Surface Warfare Center (NSWC), Dahlgren, Virginia, ObjectStore was the object-oriented DBMS chosen by the Navy to be used in research and development for the incorporation of OOT into the mission planning system. ObjectStore, which was developed by Object Design Incorporated, was chosen by the Navy because of its ability to process larger more complex databases containing data types such as voice, video, graphics, image and text; user friendliness; expressive power; reusable code and tight integration with the host environment. Even more important is its ability for high performance. With requirements being placed upon mission planning to assemble TLAM missions as quickly as possible, a DBMS had to be selected that could reduce the overhead processing that is associated with mission production.

ObjectStore is a high performance, UNIX-based OODBMS for storing complex data structures and non-record information such as bit map images, vector graphics, video and voice. ObjectStore features distribution and transaction management for concurrent access to objects among members of a workgroup. It also offers state of the art database functions such as data integrity, versioning, query processing, and a complete multi-client/multi-server architecture. The Navy's plan for ObjectStore in the TPS environment is to have ObjectStore run on SUN workstations using UNIX.
ObjectStore has three principle components: the ObjectStore runtime system, the ObjectStore applications interfaces and the ObjectStore C++ development tools. Each of these components are designed to integrate smoothly into existing development environments using third-party tools. The ObjectStore runtime system provides complete distributed DBMS services. The application interface provides access from the user's program to the runtime system using a choice of a "C" library interface or a C++ library interface.
2. PROBLEM DEFINITION

2.1 Shortcomings of Existing TMP System

After the war in the Persian Gulf, the Navy began to evaluate the performance of the weapon systems that were utilized. They were looking for deficiencies, and were concerned with finding ways to improve their systems based on their experiences in the Gulf. Prior to the Gulf War, the TWS had undergone numerous laboratory and field tests in order to prepare it for war time use. Although the weapon system was extremely effective during the crisis, there were a few areas where improvement was needed. The main concern regarding mission planning was the inability to get updated TLAM missions to the theater of operations when needed. This was not only a logistics concern, but a mission production concern. The logistics concern was that the MPC's were far away from the theater of operations and the mission production concern was that updated data needed to be input into the mission planning data bases and new missions needed to be developed. To get around these problems, the concept of an APS was developed and a MPC upgrade was considered. The APS would allow TLAM missions to be planned within the theater of operations and the mission planning upgrade would give the Navy the opportunity to improve upon their TPS.

2.1.1 INFORMIX On-Line

INFORMIX On-Line is the relational DBMS that is being used along with several binary-direct-access files to support mission planning today. On-Line is developed by INFORMIX Software, Menlo Park, California and offers
a wide range of benefits to its users including: performance, availability, data consistency, multimedia support and compliance with industry standards.

On-Line maintains its high performance by increasing disk access speed through disk management, reducing disk input/output and execution time through the use of shared memory, adding multi-processor features, and efficient data base optimization. On-Line provides users with high availability by having a dependable uninterrupted flow of data, and by minimizing the amount of downtime that can reduce productivity. In Mission Planning's case this could be critical when there is a demand for immediate TLAM mission production. On-Line ensures that data is kept correct and consistent despite the large numbers of users accessing it at the system and transaction levels, without posing any unnecessary barriers to information access and smooth performance. On-Line combines data base performance with the ability to store documents and images. On-Line's multimedia capabilities allow it to store up to two gigabytes of text or binary data.

On-Line also received full SQL certification from the National Institute of Standards and Technology (NIST) and exceeds the specifications for the American National Standards Institute (ANSI) 1989 level 2 standard for SQL data bases.
2.1.2 Constraints

INFORMIX On-Line is not one of the top relational DBMS's in the industry. Its limits can be strained by applications running intensive data operations or utilizing complex data types. TMP needs a system that can handle many overhead, complex data type and complex data transactions that will be placed upon it.

"A recent survey indicated that INFORMIX On-Line was behind other relational DBMSs, Oracle, Ingres, Rdb, and Sybase in various areas. The response base was fifty users for each DBMS. According to the survey, INFORMIX was limited in its on-line transaction processing, system administration functionality, transaction control, system monitoring capabilities, multi-level security features, performance in decision-support applications and in distributed updating, recovery and remote applications." Besides performance limitations, there were also concerns with maintenance and supportability. According to the survey, INFORMIX On-Line scored the lowest among its competitors "in responsiveness of vendor service, quality of vendor support, support for complex tables and processing reports, support for application development tools and support for standard SQL." These limitations indicate areas where there is obviously room for improvement in the TPS.

---

2.2 Need for OOT Inclusion in TMP

Today's TLAM Theater Mission Planning Centers (TMPCs) are capable of handling the production of numerous TLAM missions to support the needs of the Navy's deployed Tomahawk Weapon Systems (TWSs). However, with the world situations changing rapidly, the Navy is faced with a significant increase in the total number of TLAM missions that must be generated to fulfill the needs of the several different missile configurations and platforms, day-types, flight modes, launch areas and emerging targets. It is essential that rapid TLAM mission generation be pursued to allow the Tomahawk TMPCs to meet the ever-increasing workload. An improved TPS is needed to increase the Navy's capabilities, reduce their mission production time, and increase their quality and supportability. OOT integration into the TPS would provide the Navy with a state-of-the-art system that is much faster, easier to maintain, more efficient and more versatile.

2.2.1 Object-Oriented Data Base Design

An Object-Oriented DBMS allows the user to access every conceptual entity by using a single design concept: the object. Additionally, mechanisms such as aggregation and generalization let the user represent relationships among objects, and among object collections. Like any DBMS, an OODBMS must provide persistent storage for objects and their descriptors. The system must also provide a language schema definition for the manipulation of objects. In addition to these basic characteristics, an OODBMS usually includes: versioning, a query language,
and the necessary data base management mechanisms for access optimization, such as index and clustering, concurrency control, and authorization mechanisms for multiuser access and recovery.

"According to Martino and Bertino in their discussion of OODBMS concepts and issues an Object-Oriented Design (OOD) should consider the following five concepts:

(1) Each real-world entity is modeled by an object. Each object is associated with a unique identifier.

(2) Each object has a set of instance attributes (instance variables) and methods; the value of an attribute can be an object or set of objects.

(3) The attribute values represent the object’s status. This status is accessed or modified by sending messages to the object to invoke corresponding messages.

(4) Objects sharing the same structure and behavior are grouped into classes. A class represents a template for a set of similar objects. Each object is an instance of some class.

(5) A class is defined as a specialization of one or more classes. A class defined as a specialization is a subclass and inherits attributes and methods from its superclass or superclasses."

2.2.2 Advantages/Disadvantages

Today, most data resides in hierarchial or flat-file data bases. Programmers must specify how applications can retrieve data, which limits direct user access. For this reason, new data base sales are being dominated by relational DBMSs that let users access data by direct reference. In relational DBMSs, data is held in tables that can be

---

quickly scanned and joined together into more complex data structures. However, the relational model has its problems. With easier access, users can corrupt data. Complex queries hurt performance, and relational DBMSs have trouble with graphics and large data types created at intelligent workstations. Object-Oriented databases can overcome these limitations. Where conventional databases store records in tables or hierarchies, OODBs store objects. As a result, such data bases can store procedures as well as data, accommodate multimedia objects, handle high levels of abstraction and let users create structures of any complexity while still providing rapid access.

Relational DBMSs also allow data to be organized into records and can perform operations on many records at once. They are good at such functions as query and browse, but their limits are strained by such applications as Computer Aided Design (CAD), Computer Aided Manufacturing (CAM), graphical user interfacing, complex data transactions, and complex data types. Relational DBMS transactions and tables often have excessively large overhead for real-time engineering and multimedia applications. This is the case with the current TPS. The overhead is extremely high when databases are being accessed for data to construct missions. This in turn slows the mission production process down, which could be critical to fleet needs.

Some relational database vendors (INFORMIX, Ingres, Sybase and Oracle) are extending their products to support objects more directly, but will
likely require a tangle of SQL, C and Fourth Generation Languages (4GLs) to perform effectively. This could be an advantage to companies who have already invested significant amounts of money into their systems, but would significantly increase their maintenance and supportability requirements and cost. Also, extending the RDBMS does not increase the speed as much as a real OODBMS would.

Applications that need to store and share objects have needs that cannot be addressed by manipulating tables in a relational DBMS. An OODBMS allows applications to store and share data, while giving up table structure and some integrity checks. Object sharing reduces overall program size, reduces disk input/output and increases speed.

Extended relational data bases allow images such as Binary-Large-Objects (BLOBs) in their tables, but do little to help the user manipulate them. Object-Oriented data bases provide the user with greater flexibility when manipulating BLOBs. This flexibility includes image retrievals, modifications and storage.

A distinct disadvantage to OODBMSs is that they do not comply with the American National Standards Institute (ANSI) Structured Query Language (SQL), nor do they have a standard of their own. However, ANSI is currently developing an Object SQL (O SQL).
Relational database transactions also tend to be short and involve a tiny fraction of the objects in the database. Each data type has well-known operations and side effects. In contrast, OODBMSs must support a very large number of user types with well-defined operations and many possible side effects. Again, by sacrificing table structure and some integrity checks the TPS could be made to faster by using an OODBMS than staying with the current relational design. An OODBMS would also be smaller (less code) and tightly structured. This would make it more useful in handling the complex data transactions, and complex data types such as voice, video, graphics, multimedia and text, where relational DBMSs are simply too slow.

As an example, one of the few published benchmarks of OODBMS and RDBMSs, Cattell's Object Data Management, reports a comparison of a non-DBMS b-tree index access method, an unnamed RDBMS and an unnamed OODBMS in a billing of materials application.

"There are three tests in the benchmark: a lookup of 1000 random parts, a traversal of all parts (up to seven levels deep) connected to a randomly selected part, and an insert of 100 parts, each of which is connected to three other randomly selected parts. The benchmark was run with a database containing 20,000 parts and 60,000 connections. The measurements were done for a cold run where the buffers had not been populated and a fast run in which they had. The OODBMS was far faster on all tests, but especially so on the warm traversal, where it was almost 80 times faster."8

---

3. SYSTEM REQUIREMENTS

System requirements define a number of key questions that should be defined at the beginning of any program. Responses to these questions define a baseline from which system design and development evolve. Defining system requirements is especially difficult at the inception of a program because of the lack of system definition, but it is essential that requirements be in place prior to any design or development in order for a baseline of requirements to be established and built upon. The following paragraphs define the system requirements for the inclusion of OOT into the TPS. Table I at the end of this section summarizes all of the requirements mentioned here based on new ideas, and requirements carried over from the existing system, which will be referred to as baseline requirements. Table I also identifies which requirements are critical and which ones would be highly recommended in any object-oriented design of the TPS.

3.1 Mission Definition

The OODBMS required to support the TPS shall be capable of allowing the Navy to plan TLAM missions to support theater level operations in planning, tasking, deployment and employment of the Tomahawk Land Attack Missile.

3.2 TPS Use Requirements

3.2.a The TPS shall have the capability to perform automated and interactive data base operations on source data, derived data, tasking
information, route production files, route planning support data, mission history files, finished products and administrative information stored within the TPS data bases. These operations shall include configuration control, controlled access, recovery procedures and automatic backup with an audit trail.

3.2.b TPS data base operations involving digital imagery data should be limited to receipt, storage, management and staging for mission production.

3.2.c The TPS shall have the capability to transfer and utilize any TPS data base, in whole or in part.

3.2.d The TPS shall have the capability to automatically or manually generate backup files of any TPS data base in part or in full and shall maintain an archive of such backup files which may be recovered and incorporated in whole or in part.

3.2.e The TPS shall automatically track all activities and maintain historical information for traceability of all data bases and their contents.

3.2.f The TPS shall be able to query all data bases to automatically assess and report the availability of data required for planning TLAM missions in response to specific tasks. This report shall include data
that is missing or not yet processed, data that is outdated and the status of data on request.

3.2.g The TPS shall have the capability to query the mission data base for any descriptor field or fields, or to recall any pre-planned mission.

3.2.h The TPS shall have the capability to store and query a data base which contains scene suitability data in order to determine areas that are suitable for production of DSMAC maps.

3.2.i The TPS shall have the capability to query all imagery-related data bases and to automatically assess and display the availability and status of required imagery and support data.

3.3 OODBMS Requirements

3.3.a The OODBMS used in the TPS shall be capable of maintaining all data bases used in planning TLAM missions and shall be able to perform source data processing, cataloging, verification, formatting and storage.

3.3.b The OODBMS shall be able to monitor and control creation, maintenance and access to all TPS data bases and have knowledge about the content and location of all TPS data bases.

3.3.c The OODBMS shall have the capability to process large complex transactions and data types (i.e. voice, video, graphics, image and text).
3.3.d The OODBMS shall have the capability to accept and update its data bases with digital imagery data for neural networks.

3.3.e The OODBMS shall have the capability to accept and update its data bases with data regarding damage assessments during war time operations.

3.3.f The OODBMS shall be able to import and export portions of TPS data bases and shall provide archive, backup and recovery features.

3.3.g The OODBMS shall maintain configuration information on the state (version, creation-date, source, type, and classification) of its data bases.

3.3.h The OODBMS shall provide automatic and interactive operations on selected data bases. Data shall be stored and accessed so as to meet system performance, precision requirements.

3.3.i The OODBMS shall be able to interface with the DIWS and MDDS or the APS TDDS to import data bases in full and in part.

3.3.j The OODBMS shall provide a report generation capability and shall generate the reports required of the TPS.

3.3.k The OODBMS shall be capable of interfacing with all analysis modules (i.e. NAM, CAM, PAM and DSAM) in order to ensure mission integrity.
3.3.1 The OODBMS shall be designed and developed in accordance with the
Defense Software Development Standards specified in DoD-STD-2167A.

3.3.m The OODBMS shall provide a query language that is capable of
supporting interactive and pre-defined requests.

3.3.n The query language selected must be capable of meeting standards
specified for by the American National Standards Institute (ANSI).

3.4 Safety Requirement
The OODBMS for both the TMPC and APS shall be designed and developed in
accordance with the Military Handbook for Nuclear Weapon Systems Safety
Design and Evaluation, MIL-HDBK-272(OS) and in accordance with the
Tomahawk nuclear safety guidelines specified in PDA-14INST 8020.1A.

3.5 Security Requirements

3.5.a The OODBMS shall be able to process source data at the same
classification level at which the data was received.

3.5.b The OODBMS shall provide the capability to allow for subjects at
different security levels to have different views of a single entity or
object without causing a security violation.

3.6 Reliability Requirements

3.6.a TPS reliability engineering and analyses shall be based on a 24-hour
per day, seven days per week work cycle. Additional requirements for Mean Time Between Failure (MTBF) rates for the current TPS systems hardware and software can be found in the Test and Evaluation Master Plan (TEMP) for the TMPC/APS upgrade; (No. 1007-1).

3.6.b The TPS shall prompt or alert the operator to any errors that can degrade or prevent the accomplishment of functions, either directly or through interference with users or other programs and for which there is no reasonable work-around solution, at any level of program stress.

3.7 Maintainability Requirements

3.7.a The TPS system operational availability shall be at least 90% when measured over a randomly selected period of time during normal working hours. Additional requirements for the Mean Time To Repair (MTTR) rates for the current TPS system hardware and software can be found in TEMP No. 1007-1.

3.7.b The TPS shall allow for interactive data base maintenance and update, including storage, retrieval, modification, deletion and printing of records. The OODBMS shall have the following functions and/or capabilities:

   a. ability to modify data records
   b. ability to select records for printing
   c. ability to sort records
   d. ability to perform global updates
e. ability to generate and store statistical analysis information on data base activities
f. ability to perform file comparisons
g. ability to create support files
h. ability to create a partition in any selected data base to include elements with desired characteristics

3.8 Configuration Management Requirements

3.8.a The TPS shall have the capability to automatically record and report on request, the state of its data bases with respect to the data base version, creation date and data source.

3.8.b The TPS shall provide control of data base generation, regeneration, loading, editing and other aspects of the data base function.

3.9 Personnel Requirements

3.9.a The TPS shall be operable by military officers, enlisted personnel, government employees and contractors who have been trained in the specific operation of the functions of the system.

3.9.b Operation of the TPS shall be designed and developed in accordance with the human engineering standards specified in MIL-STD-1472C.
Table I Requirement Summary

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Source</th>
<th>Criticality</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2.a</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.b</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.c</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.d</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.e</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.f</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.g</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.h</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.2.i</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.a</td>
<td>New</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.b</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.3.c</td>
<td>New</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.d</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.3.e</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.3.f</td>
<td>New</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.g</td>
<td>New</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.h</td>
<td>New</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.i</td>
<td>New</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.j</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.3.k</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.3.l</td>
<td>New</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.m</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.3.n</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.4</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.5.a</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.5.b</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.6.a</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.6.b</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.7.a</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.7.b</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.8.a</td>
<td>New</td>
<td>Highly Recommended</td>
</tr>
<tr>
<td>3.8.b</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.9.a</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
<tr>
<td>3.9.b</td>
<td>Baseline</td>
<td>Critical</td>
</tr>
</tbody>
</table>
4. DESIGN FOR THE TPS UTILIZING OOT

Two approaches were examined for the redesigning of the TPS utilizing OOT. One approach is to extend the current RDBMS design into an Extended Relational RDBMS (ERDBMS), and the alternative is to incorporate both a RDBMS and an OODBMS into the TPS. These approaches are not meant to be the final result for the inclusion of OOT into the TPS, but present two ways that can help the current system begin to evolve into a new generation of technology.

4.1 ERDBMS Approach

The first appeared to offer the capabilities that the Navy was looking for with minimal impact. Additionally, INFORMIX Inc. recently announced that they would be releasing an extended version of INFORMIX On-Line during the first quarter of 1993 that would contain object-oriented features which would allow them to compete in the object-oriented market. The extended version of On-Line would add capabilities to the TPS to process BLOBs, enhance graphics and support objects more directly, without dramatically changing the system. Figure 10 contains a depiction of the ERDBMS architecture. The difference between the architecture depicted here and the current architecture is minimal. The ERDBMS server just replaces the RDBMS server. The basic concept would be that the ERDBMS server would replace the current RDBMS depicted by the dotted lines in figure 4. The ERDBMS would still be required to interface with the mission planners and user applications (i.e. IRDA, Analysis modules and the MDFS).
4.1.1 Advantages/Disadvantages

The main advantage to adding object-oriented features to the current RDBMS would be that it would add some speed to the TPS. It would also maintain the same DBMS engine and still be in compliance with industry standards for SQL. Additionally, it would also support the system requirements defined under TPS use, safety, reliability, maintainability and personnel requirements.

The main disadvantage to this approach is that even though OOD features would be used, the DBMS would still have a significant amount of overhead processing and still be to slow.

Another problem with adding object-oriented features to an existing RDBMS is the extensive redesign, to the existing RDBMS. The major barrier to RDBMSs adding the features of the OODBMSs is that the cost of doing so sometimes exceeds what will be received from it. The easiest parts to include would be to add data types, inheritance and encapsulation, along with collections that support fast linkages. The hardest parts would be the changes to the underlying physical access mechanisms to improve performance, especially in the use of Object Identifications (OIDs) and in memory traversals. An additional disadvantage is that several of the system requirements defined earlier under OODBMS, security and configuration management requirements, would not be met.
4.1.2 Major Design Issues

No matter which design approach is taken it is inevitable that data base modification or complete redesign will have to occur. In this approach, certain data bases would have to be selected for redesign. Data bases that would be selected would be ones which contain large images of data or BLOBs. This would include data bases such as TERCOM, DTED, DTED (CE), or DSMAC. These data bases would have to be modified to allow the ERDBMS to add, retrieve, modify, sort and delete data upon operator request in a timely fashion. Unfortunately, since these data bases are so large a mechanism would have to be devised to expedite these types of data transactions thereby reducing a tremendous amount of overhead that would otherwise be levied upon the DBMS.

Another design issue to consider is the implementation of data versioning. Data versioning would require the addition of more tables within data bases to keep track of versions, creation dates and sources. This capability needs to be included in order to satisfy the system requirements for configuration management.

Data integrity is another issue to consider. It is essential that the users have access to data that is current and accurate. One way to ensure data integrity would be to design locking mechanisms into the ERDBMS. This would allow data to be updated by one user without interruption by the other users. Once the data was updated or modified all users would
have access to it. Locking mechanisms could also be used to enhance data security.

4.1.3 Feasibility

This approach appeared to be logical since the INFORMIX RDBMS was already being used within TPS and since INFORMIX Inc. had announced that they would be releasing an extended version of INFORMIX On-Line during the first quarter of 1993. However, after examining the approach it appears that choosing this alternative would limit the capabilities of the TPS in regards to OOT. This approach only satisfies a selected number of system requirements defined earlier, is limited in its approach to OOD and limits growth and versatility. Plus, it is still utilizing a relational technology base that is over ten years old. Today’s Navy needs a system that can interface in real-time with the situation at hand. In a war situation, the system must accept real-time data inputs, update the data bases and be able to plan missions quickly. No matter how this design approach was modified or "molded" the versatility and growth would be missing.

4.2 RDBMS and OODBMS Approach

This approach, like the ERDBMS approach, builds off of the existing architecture. The difference is that instead of extending the existing RDBMS, a separate OODBMS would be added to the architecture. Figure 11 contains a high level depiction of the RDBMS and OODBMS architecture. In this approach the application software would have the option to talk to
Figure 11 RDBMS and OODBMS Architecture
both a relational and object-oriented data base server depending on the type of data the user wishes to access. According to Parker Hodges, a researcher at Object Design Incorporated, "the future will be applications that can talk to a number of servers."\(^9\)

The ObjectStore OODBMS would be used for storing the critical complex data structures and non-record oriented information such as bit map images, enhanced graphics and sound. The INFORMIX RDBMS would be used to handle the remaining record oriented information and simpler data types where data base access time is not as critical. This approach also would involve converting the data bases that are still using binary-direct access techniques into OODB's. The goal of this approach would be to begin the evolution of OODB use into the TPS.

\[ 4.2.1 \text{ Advantages/Disadvantages} \]

This approach would satisfy the system requirements defined earlier through increasing the capabilities of mission planning, reducing production time and increasing maintainability and supportability.

The capabilities of mission planning would be enhanced since the Navy would now have the option to store and update its OODB's with complex data relationships and data types such as voice, video, graphics, image and text. By being able to store and manipulate data of this type, in a

timely manner, new techniques for mission planning could be developed. These techniques might include adding data for neural networks, that could be used in future missile guidance set upgrades and in adding data base capabilities that would allow for rapid data base update and rapid mission production onboard an APS platform for war time damage assessment data.

Mission planning production time would be reduced since all of the data bases that required extensive overhead processing in a relational environment could be converted over into an object-oriented format. By giving up table structure an OODBMS can execute significantly faster than a RDBMS.

Data base quality and supportability would also be enhanced through ObjectStore's state of the art data base functions such as data integrity, versioning, and query processing. ObjectStore's client/server implementation allows one server to support many client workstations, and each workstation to simultaneously access multiple data bases.

The main disadvantage to this approach is that there currently is no industry standard query language for an OODBMS. This would not satisfy the system requirement that the OODBMS "must provide a query language that is capable of meeting the standards specified by ANSI". However, ObjectSQL (OSQL) is currently being developed by ANSI and could be used by the ObjectStore OODBMS.
Another concern for this approach is that the application code that interfaces with both DBMS's would require a more extensive redesign. Application software would need to be concerned with interfacing with two DBMS's now instead of one. Since the OODBMS would be much faster than the RDBMS there may be some timing problems with data being accessed from the RDBMS and being needed by the OODBMS.

Another advantage to this approach is that it would provide a more realistic mission planning data base environment. Design classes and objects within the OODBMS represent real-world design entities and provide a much better feel for the mechanics of the problem than do a set of flat tables as in the INFORMIX RDBMS. Also, there is no mechanism in a RDBMS to associate behavior with data, which is easily achieved with objects.

This approach also offers a more powerful data base environment for mission planning. Object data representation is highly flexible and may be customized by users with little or no restriction. It allows data to be specialized and intelligent. The facilities that ObjectStore offers for inheritance and schema evolution would allow the design to grow incrementally. Problems that normally occur in a RDBMS, such as duplication of data, do not exist. Again, the complexity of the data and the intricacies of relationships that can be handled by objects is far superior to that supported by the RDBMS.
4.2.2 Major Design Issues

This design approach will also require extensive data base manipulation or complete redesign. The goal would be to begin a migration to a complete TPS OODBMS. This would be accomplished by removing the binary-direct access format from the TPS and converting selected INFORMIX relational data bases to OODBs. This would reduce the number of data base formats to two: INFORMIX and ObjectStore. All of the data bases that are currently accessed by using binary-direct access would be converted either to an INFORMIX or ObjectStore format based on their criticality to the system. Selected INFORMIX relational structured data bases would also be migrated over into ObjectStore object-oriented format based upon their criticality. Criticality for selection to one format or the other would be based on the capabilities expected of a particular data base. For example, data bases that are expected to support complex data types and complex data relationships such as TERCOM, DTED, DTED (CE), DSMAC, City, LPDB, VODB, PADB, GEO BNDS, and LAIDB would be candidates for conversion into OODB's. Data bases such as SAM(TOF), SAM(PK), SS_AREA, SS_CELL, the ARP data bases and ARD data bases would be candidates for the INFORMIX relational format.

Another design issue to consider is the details of the migration of the selected relational bases into an object-oriented format. An example of how this restructuring might occur is as follows. The grid depicted in Figure 12 represents a section within the county of Spotsylvania, Virginia. The grid square would be identified by its latitude and longitude coordinates. The square contains various geographic features:
Figure 12 Geographic DB Grid Example
<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment</td>
<td>Curve Object</td>
<td>River</td>
<td>Name Object</td>
</tr>
<tr>
<td></td>
<td>Vector Object</td>
<td></td>
<td>Segment Object</td>
</tr>
<tr>
<td>Node</td>
<td>Latitude_Deg_1 Object</td>
<td>Grid Object</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Longitude_Deg_1 Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Latitude_Deg_2 Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Longitude_Deg_2 Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Latitude_Deg_3 Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Longitude_Deg_3 Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Latitude_Deg_4 Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Longitude_Deg_4 Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSMAC</td>
<td>Map Object</td>
<td>Grid Object</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airport</td>
<td>Name Object</td>
<td>Easting Object</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Grid Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hanger Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tarmac Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Terminal Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Map</td>
<td>Grid Object</td>
<td>Development</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Filter Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scene Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vector</td>
<td>Descriptor Object</td>
<td>Curve</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Start_Pt Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>End_Pt Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Distance Object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ctr_Pt Object</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 13 Class Definitions
a section of the Rappahannock river, a section of the Richmond, Fredericksburg and Potomac Railroad, a section of U.S. Route 17, a section of a power line, Shannon Municipal airport, and various subdivision streets. Figure 13 illustrates how each class would be identified in object-oriented format for the grid depicted in Figure 12.

Modifications to the application software would have to be considered a major design issue. As mentioned earlier, there may be timing problems that will occur if the OODBMS has to wait on the RDBMS for data.

There may also be problems with transactions that include modifications to multiple data bases within separate DBMSs at the same time. INFORMIX On-Line has a two-phase commit protocol, where data consistency is guaranteed at a global level. In transactions that include modifications of more than one data base server, one of the servers assumes the role of coordinator for the global transaction. The coordinator routes the transaction work and tracks the progress of the transaction for all of the On-Line data base servers that participate in the transaction. This could create problems when one server will be an INFORMIX RDBMS server and the other will be an OODBMS server. Additional application software will need to be developed to ensure that when a data base transaction is required of both DBMSs at the same time, that the transaction will be executed in both DBMSs and not just one or the other. Having this monitor software at the application level will help to ensure data integrity.
4.2.3 Feasibility

This design approach appears to be the better of the two approaches discussed. It is a design that would allow the Navy to begin a gradual transition into object-oriented technology. This approach would satisfy the system requirements defined earlier, adds OOD to the TPS and allows growth and versatility. It provides the Navy with a state-of-the-art system that can begin to interface real-time with the situation at hand. Integrating the OODBMS with the RDBMS makes the TPS much faster, more efficient and more maintainable and supportable.
5. CONCLUSIONS

The current state of the world requires the Navy to be responsive and adaptable to changing situations at a moments notice. The upgrade to the TPS for both the TMPC's and the APS's will provide the Navy with the ability to enhance its weapon system, by possibly adding neural networks to the cruise missile guidance set, or by adding in-flight GPS update capabilities to the missile from reconnaissance reports on damage assessment. By beginning to incorporate OOD into the TPS these goals can be met. The incorporation of both a RDBMS and OODBMS into TPS will update the current TPS and provide the Navy with enhanced capabilities now, while opening the door for the eventual migration over to an entire TPS that would utilize only object-oriented technology.
BIBLIOGRAPHY


Theater Mission Planning Center (TMPC) System Specification, 1 February 1988, PDA-14 3900/2A.

# APPENDIX A

## Acronym List

### A

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA</td>
<td>Anti-Aircraft Artillery</td>
</tr>
<tr>
<td>ABL</td>
<td>Armored Box Launcher</td>
</tr>
<tr>
<td>ACTIV</td>
<td>Activity Journal/Log Data Base</td>
</tr>
<tr>
<td>AD</td>
<td>Area Defense</td>
</tr>
<tr>
<td>AGL</td>
<td>Above Ground Level</td>
</tr>
<tr>
<td>ANSI</td>
<td>American National Standards Institute</td>
</tr>
<tr>
<td>AOB</td>
<td>Air Order of Battle</td>
</tr>
<tr>
<td>APS</td>
<td>Afloat Planning System</td>
</tr>
<tr>
<td>AQ</td>
<td>Accessibilities Quota</td>
</tr>
<tr>
<td>ARD</td>
<td>Autorouter Data</td>
</tr>
<tr>
<td>ARD(ACCESS)</td>
<td>Autorouter Data(Accessibility)</td>
</tr>
<tr>
<td>ARD(ARCS)</td>
<td>Autorouter Data(Arcs)</td>
</tr>
<tr>
<td>ARD(CHART)</td>
<td>Autorouter Data(Characterized Terrain)</td>
</tr>
<tr>
<td>ARD(CLOB)</td>
<td>Autorouter Data(Clobber)</td>
</tr>
<tr>
<td>ARD(DANGER)</td>
<td>Autorouter Data(Dangerspace)</td>
</tr>
<tr>
<td>ARP</td>
<td>Autorouter Parameter</td>
</tr>
<tr>
<td>ARP(ALG)</td>
<td>Autorouter Parameter(Algorithms)</td>
</tr>
<tr>
<td>ARP(AQ)</td>
<td>Autorouter Parameter(Accessibilities Quota)</td>
</tr>
<tr>
<td>ARP(CLOB)</td>
<td>Autorouter Parameter(Clobber)</td>
</tr>
<tr>
<td>ARP(NAV)</td>
<td>Autorouter Parameter(Navigation)</td>
</tr>
<tr>
<td>ARP(SW)</td>
<td>Autorouter Parameter(Switches)</td>
</tr>
<tr>
<td>ARP(TE)</td>
<td>Autorouter Parameter(Target Evaluation)</td>
</tr>
<tr>
<td>ARP(VEH)</td>
<td>Autorouter Parameter(Vehicle)</td>
</tr>
<tr>
<td>ARP(VIF)</td>
<td>Autorouter Parameter(Violation Impact Factors)</td>
</tr>
<tr>
<td>ATWCS</td>
<td>Advanced Tomahawk Weapon Control System</td>
</tr>
</tbody>
</table>

### B

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLOB</td>
<td>Binary Large Object</td>
</tr>
<tr>
<td>BNDS</td>
<td>Boundaries</td>
</tr>
</tbody>
</table>

### C

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAD</td>
<td>Computer Aided Design</td>
</tr>
<tr>
<td>CADOB</td>
<td>Consolidated Aerospace Defense Order of Battle</td>
</tr>
<tr>
<td>CAM</td>
<td>Clobber Analysis Module</td>
</tr>
<tr>
<td>CASE</td>
<td>Computer Aided Software Engineering</td>
</tr>
<tr>
<td>CCLP</td>
<td>Command and Control Launch Packages</td>
</tr>
<tr>
<td>CDPS</td>
<td>Central Data Processing Set</td>
</tr>
<tr>
<td>CEP</td>
<td>Circular Error of Probability</td>
</tr>
<tr>
<td>CHART</td>
<td>Characterized Terrain Data</td>
</tr>
<tr>
<td>CIA</td>
<td>Central Intelligence Agency</td>
</tr>
<tr>
<td>CLOB</td>
<td>Clobber Data</td>
</tr>
<tr>
<td>CM</td>
<td>Configuration Management</td>
</tr>
</tbody>
</table>
CMGS  Cruise Missile Guidance Set
CMSA  Cruise Missile Support Activity

D
DBA  Data Base Administrator
DBMS  Data Base Management System
DCP  DTD Certification Processor
DDB  Defense Data Base
DDPS  Dedicated Data Processing Set
DEC  Digital Equipment Corporation
DFAD  Digital Feature Analysis Data
DIWS  Digital Image Workstation
DMA  Defense Mapping Agency
DoD  Department of Defense
DPA  Dynamic Program Algorithm
DPM  Data Preparation and Maintenance
DPT  Dedicated Planning Terminal
DSAM  Defense System Analysis Module
DSCC  Defense System Characteristics and Capabilities
DSMAC  Digital Scene Matching Area Correlator
DTD  Data Transport Device
DTED  Digital Terrain Elevation Data
DTED (CE)  Digital Terrain Elevation Data Constant Elevation
DTM  Digital Terrain Matrix

E
EDB  Environment Data Base
ERDBMS  Enhanced Relational Data Base Management System
EH  Estimated Hit
ESAMS  Enhanced SAM Simulation

F
FD  Fixed Defense
FLAR  Fighter Launch Acceptability Range
4GL  Fourth Generation Language

G
GCI  Ground Controller Intercept
GENSER  General Service
GEO  Geography
GPS  Global Positioning System
GSC  Geographic Search Capability

73
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDBK</td>
<td>Handbook</td>
</tr>
<tr>
<td>HOB</td>
<td>Height of Burst</td>
</tr>
<tr>
<td>ID</td>
<td>Identification</td>
</tr>
<tr>
<td>IHF</td>
<td>Image History File</td>
</tr>
<tr>
<td>IMGCOV</td>
<td>Imagery Cover Data Base</td>
</tr>
<tr>
<td>IRDA</td>
<td>Interactive Route Development and Analysis</td>
</tr>
<tr>
<td>JSTPS</td>
<td>Joint Services Theater Planning System</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>LAIDB</td>
<td>Launch Areas of Interest Data Base</td>
</tr>
<tr>
<td>LCG</td>
<td>Launch Control Group</td>
</tr>
<tr>
<td>LPDB</td>
<td>Launch Points Data Base</td>
</tr>
<tr>
<td>MB</td>
<td>MegaByte</td>
</tr>
<tr>
<td>MDDS</td>
<td>Mission Data Distribution System</td>
</tr>
<tr>
<td>MDGA</td>
<td>McDonnell Douglas Government Aerospace</td>
</tr>
<tr>
<td>MDPS</td>
<td>Mission Data Preparation System</td>
</tr>
<tr>
<td>MHF</td>
<td>Mission History File</td>
</tr>
<tr>
<td>MIL</td>
<td>Military</td>
</tr>
<tr>
<td>MMSD/T</td>
<td>Mass Media Storage Device/Tomahawk</td>
</tr>
<tr>
<td>MOB</td>
<td>Missile Order of Battle</td>
</tr>
<tr>
<td>MPC</td>
<td>Mission Planning Center</td>
</tr>
<tr>
<td>MSQI</td>
<td>Minimum Mapset Quality Indicator</td>
</tr>
<tr>
<td>MTBF</td>
<td>Mean Time Between Failure</td>
</tr>
<tr>
<td>MTTR</td>
<td>Mean Time To Repair</td>
</tr>
<tr>
<td>NAM</td>
<td>Navigation Accuracy Module</td>
</tr>
<tr>
<td>NSWC</td>
<td>Naval Surface Warfare Center</td>
</tr>
<tr>
<td>NCPS</td>
<td>National Contingency Planning System</td>
</tr>
<tr>
<td>NRTT</td>
<td>Near Real Time Threat</td>
</tr>
<tr>
<td>OFS</td>
<td>Operational Flight Software</td>
</tr>
<tr>
<td>OID</td>
<td>Object Identification Description</td>
</tr>
</tbody>
</table>
OLVP  Off-Line Verification Process
OODB  Object Oriented Data Base
OODBMS Object Oriented Data Base Management System
OOD  Object-Oriented Design
OOPL  Object-Oriented Programming Language
OOT  Object Oriented Technology
OP  Operational
OSQL  Object Structured Query Language
OTW  Over-The-Water

P
PADB  Prohibited Areas Data Base
PAM  Performance Analysis Module
PDSL Potential DSMAC Scene Locations
PI  Planning and Implementation
PK  Probability of Kill
PPDB  Point Positional Data Base
PVOD  Probabilistic Vertical Obstruction Data Base

R
RADAR_S/I  Radar Signal Noise Interference Ratio
RADGUNS  Radar Detected Gun Simulation
RASS  Random Access Storage Set
RDBMS  Relational Data Base Management System
RIMM  Removable Interchangeable Memory Module
ROB  Radar Order of Battle
RTM  Radar Terrain Mask
RTMAM  Radar Terrain Mask Altitude Matrix

S
SAM  Surface-To-Air Missile
SCI  Sensitive Compartmented Information
SLCM  Sea-Launched Cruise Missile
SQL  Structured Query Language
SS  Scene Suitability
STD  Standard
SW  Switch

T
TCG  Track Control Group
TCDB  Terrain Contours Data Base
TDODS  Tactical Data Distributions System
TDPS  Track Data Processing Set
TE  Target Evaluation
TEMP  Test and Evaluation Master Plan
TERCOM  Terrain Contour Matching
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TGTC</td>
<td>Target Complex</td>
</tr>
<tr>
<td>TLAM</td>
<td>Tomahawk Land Attack Missile</td>
</tr>
<tr>
<td>TM</td>
<td>Threat Module</td>
</tr>
<tr>
<td>TMP</td>
<td>Theater Mission Planning</td>
</tr>
<tr>
<td>TMPC</td>
<td>Theater Mission Planning Center</td>
</tr>
<tr>
<td>TPS</td>
<td>Theater Mission Planning System</td>
</tr>
<tr>
<td>TMTA</td>
<td>Threat Module Trajectory Analysis</td>
</tr>
<tr>
<td>TO</td>
<td>Track Operator</td>
</tr>
<tr>
<td>TOB</td>
<td>Tactical SAM Order of Battle</td>
</tr>
<tr>
<td>TPP</td>
<td>Theater Planning Package</td>
</tr>
<tr>
<td>TPS</td>
<td>TLAM Planning System</td>
</tr>
<tr>
<td>TSC</td>
<td>Threat Site Characteristics</td>
</tr>
<tr>
<td>TTP</td>
<td>Terminal Terrain Profile</td>
</tr>
<tr>
<td>TWCS</td>
<td>Tomahawk Weapon Control System</td>
</tr>
<tr>
<td>TWS</td>
<td>Tomahawk Weapon System</td>
</tr>
</tbody>
</table>

**U**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USAF</td>
<td>United States Air Force</td>
</tr>
<tr>
<td>USAFETAC</td>
<td>USAF Environmental Technical Analysis Center</td>
</tr>
<tr>
<td>USCINC</td>
<td>United States Commander In Chief (President)</td>
</tr>
<tr>
<td>USN</td>
<td>United States Navy</td>
</tr>
</tbody>
</table>

**V**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLS</td>
<td>Vertical Launching System</td>
</tr>
<tr>
<td>VL/T</td>
<td>Vertical Launch/Tomahawk</td>
</tr>
<tr>
<td>VOD</td>
<td>Vertical Obstruction Data</td>
</tr>
<tr>
<td>VPI &amp; SU</td>
<td>Virginia Polytechnic Institute and State University</td>
</tr>
</tbody>
</table>
# APPENDIX B

## Table B-1 Geographic Data Bases

<table>
<thead>
<tr>
<th>Acronym</th>
<th>File Structure</th>
<th>Type</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEOGRAPHY</td>
<td>INFORMIX</td>
<td>P</td>
<td>Unclassified</td>
</tr>
<tr>
<td>GEO BNDS</td>
<td>INFORMIX</td>
<td>P</td>
<td>Unclassified</td>
</tr>
<tr>
<td>CITY</td>
<td>INFORMIX</td>
<td>P</td>
<td>Unclassified</td>
</tr>
<tr>
<td>VODB</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>PADB</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>LPDB</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>LAIDB</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>DFAD</td>
<td>TBD</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>PPDB</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>EDB</td>
<td>TBD</td>
<td>P</td>
<td>Unclassified</td>
</tr>
<tr>
<td>IMGCOV</td>
<td>INFORMIX</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>OP AREAS</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>GSC</td>
<td>TBD</td>
<td>D</td>
<td>Top Secret</td>
</tr>
<tr>
<td>PVODB</td>
<td>TBD</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>IHF</td>
<td>INFORMIX</td>
<td>D</td>
<td>Secret</td>
</tr>
</tbody>
</table>

**Key:**

- **P** - Primary
- **D** - Derived
- **BDA** - Binary-Direct-Access
- **TBD** - To Be Designed
<table>
<thead>
<tr>
<th>Acronym</th>
<th>File Structure</th>
<th>Type</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDB</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>DSCC</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>SS AREA</td>
<td>INFORMIX</td>
<td>D</td>
<td>Top Secret</td>
</tr>
<tr>
<td>SS CELL</td>
<td>TBD</td>
<td>D</td>
<td>Top Secret</td>
</tr>
<tr>
<td>FLAR</td>
<td>TBD</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>RADAR S/I</td>
<td>TBD</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>SAM(TOF)</td>
<td>BDA</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>SAM(PK)</td>
<td>BDA</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>NRTT</td>
<td>TBD</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>TSC</td>
<td>TBD</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>AAA EH</td>
<td>TBD</td>
<td>D</td>
<td>Secret</td>
</tr>
</tbody>
</table>

Key:  
P - Primary  
D - Derived  
BDA - Binary-Direct-Access  
TBD - To Be Designed
Table B-3 Terrain Data Bases

<table>
<thead>
<tr>
<th>Acronym</th>
<th>File Structure</th>
<th>Type</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTED</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>DTED CE</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>TERCOM</td>
<td>INFORMIX</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>RTMAM</td>
<td>BDA</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>RTM</td>
<td>BDA</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>TCDB</td>
<td>TBD</td>
<td>D</td>
<td>Confidential</td>
</tr>
<tr>
<td>DTM</td>
<td>INFORMIX</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>TTP</td>
<td>TBD</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>ARP(ALG)</td>
<td>BDA</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>ARP(AQ)</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>ARP(CLOB)</td>
<td>BDA</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>ARP(NAV)</td>
<td>BDA</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>ARP(SW)</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>ARP(TE)</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>ARP(VEH)</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>ARP(VIF)</td>
<td>BDA</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>ARD(ACCESS)</td>
<td>BDA</td>
<td>D</td>
<td>Top Secret</td>
</tr>
<tr>
<td>ARD(ARCS)</td>
<td>BDA</td>
<td>D</td>
<td>Top Secret</td>
</tr>
<tr>
<td>ARD(CHART)</td>
<td>BDA</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>ARD(DANGER)</td>
<td>BDA</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>DSMAC</td>
<td>INFORMIX</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>POSL</td>
<td>TBD</td>
<td>D</td>
<td>Secret</td>
</tr>
<tr>
<td>IMAGE PROD</td>
<td>TBD</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>CATALOG</td>
<td>INFORMIX</td>
<td>D</td>
<td>Secret</td>
</tr>
</tbody>
</table>

Key: P - Primary
D - Derived
BDA - Binary-Direct-Access
TBD - To Be Designed
<table>
<thead>
<tr>
<th>Acronym</th>
<th>File Structure</th>
<th>Type</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>MHF</td>
<td>INFORMIX</td>
<td>D</td>
<td>Top Secret</td>
</tr>
<tr>
<td>TGTC</td>
<td>INFORMIX</td>
<td>P</td>
<td>Top Secret</td>
</tr>
<tr>
<td>PI RULES</td>
<td>TBD</td>
<td>P</td>
<td>Secret</td>
</tr>
<tr>
<td>ACTIV</td>
<td>BDA</td>
<td>D</td>
<td>Unclassified</td>
</tr>
<tr>
<td>GPS</td>
<td>TBD</td>
<td>P</td>
<td>Secret</td>
</tr>
</tbody>
</table>

Key: P - Primary  
D - Derived  
BDA - Binary-Direct-Access  
TBD - To Be Designed