TRANSWAY: Planning with the Tabu Search Algorithm

J. Nibecker, D. Taylor, R. Chambers, H. Larsen, K. Cudworth, C. Warren, M. Porczak, and J. Pohl

Collaborative Agent Design Research Center (CADRC)
California Polytechnic State University (Cal Poly)
San Luis Obispo, California, USA

Abstract

Military deployment and distribution responsibilities call for intelligent collaborative tools in support of strategic and operational planning functions involving the sustainment and movement of military forces. The sustainment requirement is generated at the operational level and is dynamic. It is composed of shifting priorities responding to changes in commander's intent and changes in the operational situation.

The TRANSWAY software application is designed as a set of intelligent collaborative tools supporting operators performing planning and re-planning tasks in a dynamically changing decision-making environment. TRANSWAY includes several agents with strategic and operational planning and re-planning capabilities. The principal agent is based on the Tabu Search algorithm, with the intent of finding an optimum plan for the delivery of supplies from multiple origins, through multiple routes, with different kinds of conveyances, to multiple destinations, within specified time and resource constraints.

The TRANSWAY System Architecture

The TRANSWAY system has a three-tier, service-oriented architecture, implemented using the Integrated Cooperative Decision Making (ICDM) ontology-based software development framework and the Hibernate object/relational persistence and query service. Figure 2.1 provides an illustration of the key components within each of these tiers (i.e., *presentation*, *information*, and *logic* tiers).

TRANSWAY incorporates an internal information model (i.e., ontology) consisting of objects, their characteristics, and the relationships among those objects. The information model is a virtual representation of the real world domain under consideration and is designed to provide adequate context for software agents (typically rule-based) to reason about the current state of the virtual environment. Since information-centric software has some 'understanding' of what it is processing it normally contains tools rather than predefined solutions to predetermined problems. These tools are commonly software agents that collaborate with each other and the human user(s) to develop solutions to problems in near real-time as they occur. Communication between information-centric applications is greatly facilitated since only the changes in information need to be transmitted. This is made possible by the fact that the object, its characteristics and its relationships are already known by the receiving application.

The *presentation* tier interfaces with human operators through a Graphic User-Interface (GUI) comprised of a menu system, map display, agent display, and various reports. The main TRANSWAY GUI is based on the Generic Space Generator (GSG) framework employing Java Bean technology and offering high performance map and graphics management. The map display supports a variety of map formats (e.g., CADRG, satellite imagery, etc.) and provides standard map interaction functionality (i.e., zoom, pan, highlight, layer management, etc.). In addition, due to its objectified nature theater and operational entities (e.g., tracks, operations centers, routes, planned activities, etc.) can be presented within the map display and interrogated through direct operator interaction. The agent display shows various concerns and recommendations generated by the agents for the operator to inspect.

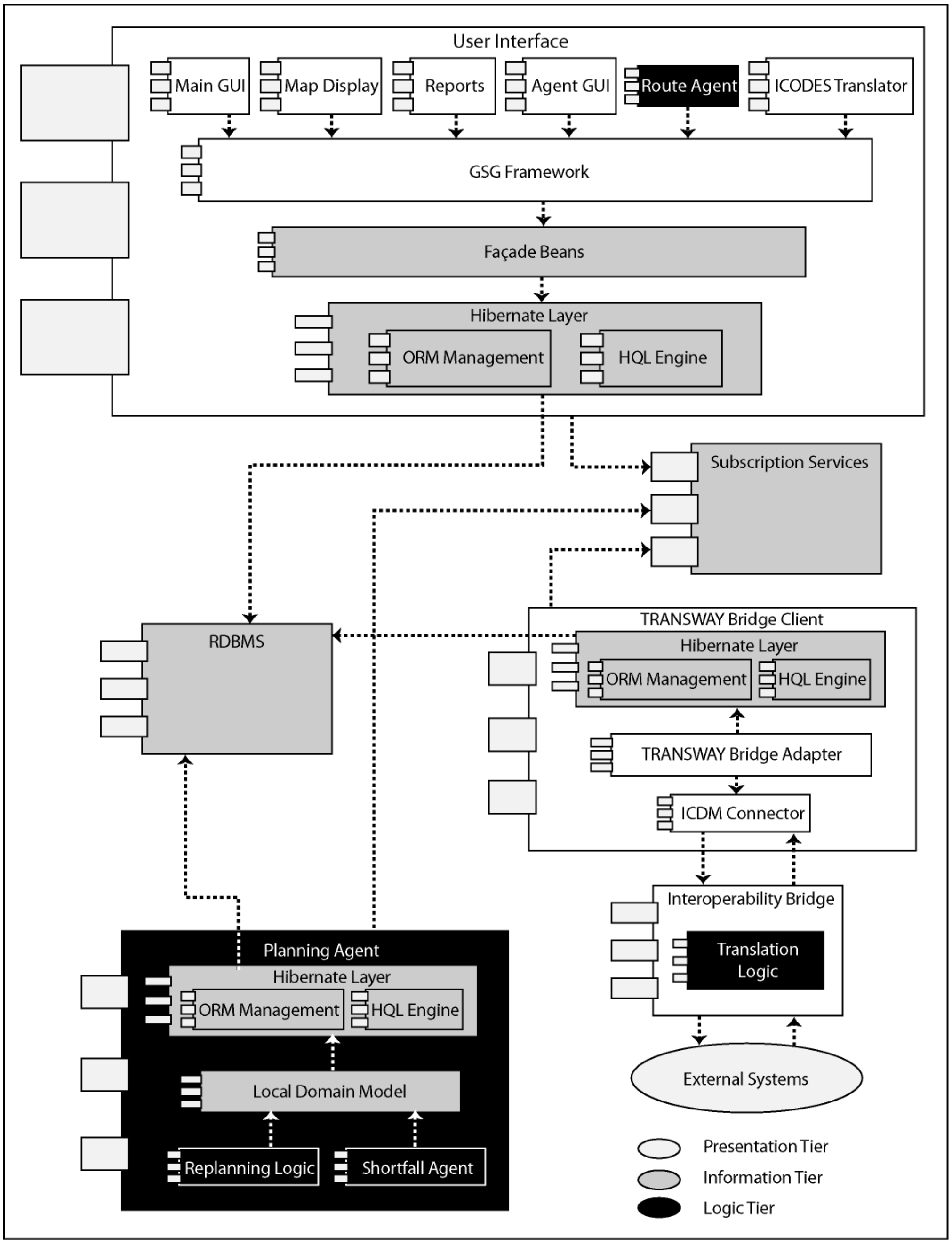


Figure 2.1: The TRANSWAY system architecture

Presentation and interaction with external systems is provided through the ICDM Interoperability Bridge supporting complex translation among potentially disparate system representations

(Leighton et al. 2004). Such translations can be specified as Extensible Style-sheet Language Transforms (XSLT) or rule-based logic. The underlying interaction metaphor supported by the Interoperability Bridge is that of *remote service calls* issues between bridge clients (i.e., interoperating systems).

The *information* tier utilizes an information-based ontology that provides relationship-rich descriptions of the concepts, notions, and entities relevant to the domains over which the system operates. These information-centric descriptions form the means by which intelligent decision-support agents analyze the evolving common operational picture. To support high degrees of extensibility, flexibility, referential integrity, and representational accuracy the TRANSWAY ontology employs numerous well-established analysis patterns such as operational-knowledge separation, contextual roles, and so on (Fowler 2003, Fowler 1997, Fowler and Scott 1997) as the basis for many of the concepts and entities it represents.

Information within the TRANSWAY system is persisted in a standard Relational Database Management System. To support the object-oriented nature inherent in the ontology structure a Hibernate object-to-relational mapping (ORM) layer is inserted within each client. It is through this object access layer that clients (e.g., GUI, agents, Interoperability Bridge) interact with the ontology. Collaboration among system entities is empowered through the use of the ICDM *Subscription Service* to register ontology-based interests and the Hibernate Query Language (HQL) facilities. Using these two mechanisms, TRANSWAY clients employ a decoupled collaboration model interacting with other parts of the system via the changes that occur in the ontology. This type of interaction model parallels the well-established *blackboard architecture* prominent in artificial intelligence-oriented systems. A further advantage of this type of decoupled collaborative architecture is that since clients need not know of each other's existence it is possible to attach and detach clients based on evolving system and operational needs.

The *logic* tier is comprised of technologies derived from both the artificial intelligence (AI) and operations research disciplines, in the form of software agents. The agents take the form of Java applications or other AI-based languages that collaborate via the information tier in accordance with a standard *blackboard* model. Agents provide the reasoning capabilities in TRANSWAY in several forms. Planning agents utilize proven planning algorithms that produce quality plans according to set criteria. Other monitoring agents utilize symbolic reasoning to recognize complex patterns representing specific situations that require the attention of the operator.

On the symbolic reasoning side, rule-based agents are employed to analyze theater and operational context providing alerts and recommendations (e.g., entire plans, or reacting to changing circumstances, or alternative actions that can be incorporated into existing plans). Another type of agent employed in the TRANSWAY system is based on the Tabu Search algorithm (Karaboga and Pham 2000, Glover and Laguna 1997). Unlike symbolic reasoning, the Tabu approach evolves toward solutions to complex problems (i.e., scheduling, etc.) by applying an extended greedy search algorithm that employs forms of adaptive memory to avoid pre-mature isolation in local optima with respect to the effective solution space. By employing two historically disparate technologies the TRANSWAY agents take advantage of the precision and definability of symbolic reasoning and the performance of a greedy search, while minimizing each of their respective limitations.

To aid in development and management of decision-support systems such as TRANSWAY, the ICDM toolkit provides framework generation tools capable of automatically processing the UML representation of an ontology into a platform specific implementation (Leighton et al. 2004). The ability to quickly and iteratively move from model to implementation promotes a development environment where agility to changing requirements and evolving knowledge acquisition are significantly improved over more manual approaches.

The Underlying Ontology

The representation of data and its interpretation for decision-support systems must be complex by necessity due to the very nature of the decision-support process. This complexity may be defined either in the interpretation of the data or it may be placed in the data representation itself. By placing the complexity in the data representation, less work is required to be performed to interpret the data. Additionally, this complex representation may more accurately reflect the real nature of the problem to be analyzed and may in fact more directly represent the knowledge that is proposed to be captured.

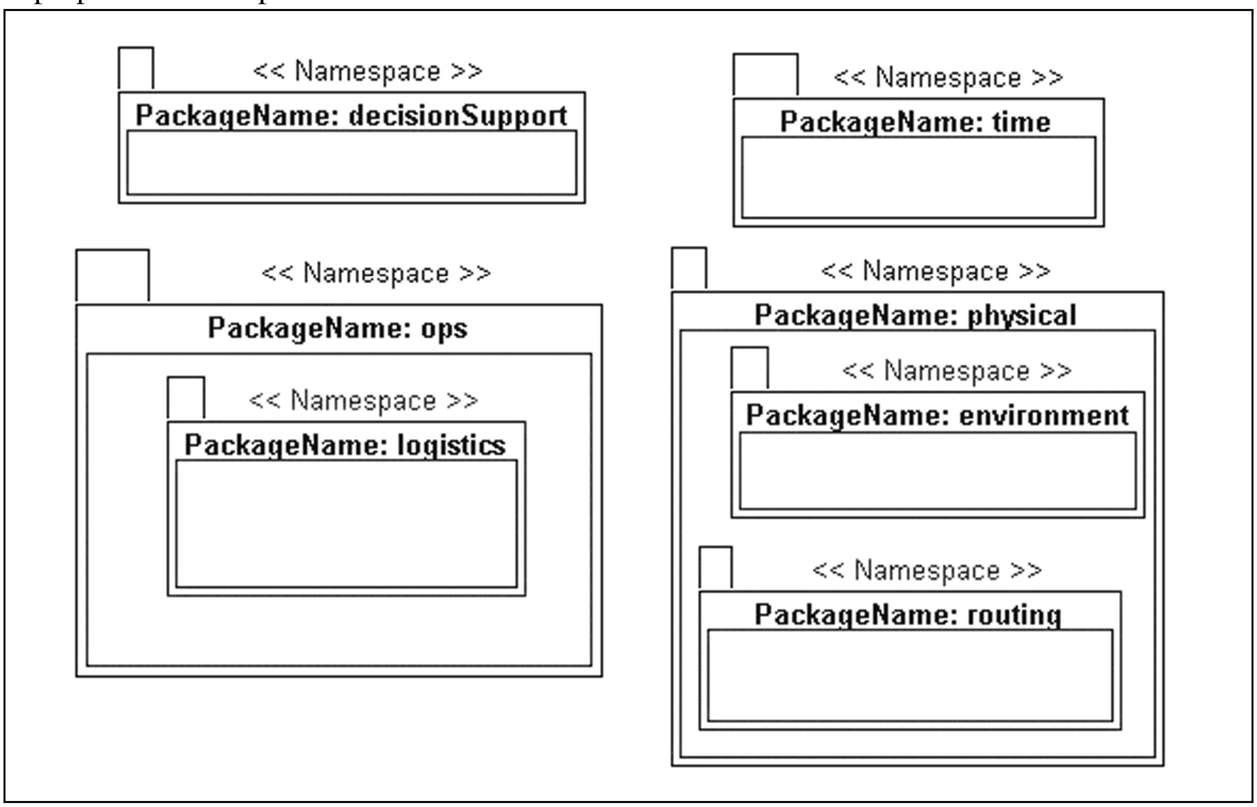


Figure 5.1: TRANSWAY ontology domains

An ontology can be characterized as an explicit specification of a conceptualization. The term is borrowed from philosophy, where an ontology is a systematic account of existence. For a software application, what "exists" is that which can be represented. When the information and knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called the universe of discourse. This set of objects, and the describable relationships among them represents all the information and knowledge that can be known in the context of the applications that employ them. In such an ontology, definitions associate the names of entities in the universe of discourse (e.g., classes, relations, functions, or other objects) with human-readable text describing what the names mean, and formal axioms that constrain the interpretation and well-formed use of these terms.

The TRANSWAY ontology is divided into logical domains that can be described using the Unified Modeling Language (UML) methodology (Figure 5.1). These domains, or namespaces, are indicated by UML package symbols and named accordingly. Within each domain exist definitions of the various concepts and entities relevant to the representation and analysis of key aspects of each domain. Classes located within package symbols are defined within that domain. These classes may relate to classes defined in other domains through either *inheritance* or *associations*. In both cases, referenced classes are identified by their symbols existing outside the primary package symbol with some type of relationship symbol connecting them to package elements. Domains themselves may be related to each other in either a sibling or parent/child relationship. Such connections are an indication of the particular scope and inter-domain visibility. Following are brief textual descriptions and UML-based illustrations describing each domain. The names of the classes currently supported by TRANSWAY and some typical class descriptions are included in the Appendix.

The Tabu Agents

The current version of TRANSWAY includes several agents built around the Tabu search algorithm (Karaboga and Pham 2000, Glover and Laguna 1997). Tabu Search is a local search method for exploring a solution space (OpenTS 2005). It is best suited for combinatorial solution spaces where a certain combination of atomic entities is considered a solution.

The TRANSWAY agents need to be highly responsive to system events, so that they can adjust their plan generation strategies dynamically as the user makes changes to the visual environment. For example, if a route becomes unavailable due to weather or an enemy threat the agents should be informed of the disabled route and respond appropriately. A common practice for supporting this level of responsiveness in a Java development environment is to use Java Beans. A Java Bean provides a strategy for event-driven programming. By encapsulating all of the properties of an object into a bean and notifying listeners when properties change it is possible to create the necessary event-driven environment.

Since the TRANSWAY system incorporates many small agents that perform specific computational tasks, threading and synchronization required particular attention. Often several of these computational tasks need to be performed in parallel or, more accurately stated, cannot be performed serially. An example of this requirement for concurrency is the need for one agent to monitor the current demand for supplies, while another agent continually calculates the all-pairs shortest path algorithm.

Separation of Trip and Plan Generation: The literature describes many different approaches to combinatorial problems of the type encountered in trip routing (Talbi 2002). Based on a review of this literature it was decided early on in the design of the TRANSWAY agents to treat trip and plan generation as separate problems. It was noted that most of the approaches cited in the literature utilize not one but several strategies for solving the combinatorial problem. While the different strategies are normally domain specific, the commonality that appears to exist among most of the approaches is to limit the search space of the problem by taking advantage of the known constraints of the system. This criterion was adopted as an important design feature of the TRANSWAY Tabu agents, to limit the number of trips produced so that the combinations of trips that make up a better (i.e., more optimal) plan can be found more quickly.

Selection of Search Methodology: After the separation of trip and plan generation the planning part becomes primarily a search problem. As new trips are generated they need to be considered as possible components of a recommended plan. However, even with the limitation of the search space through the application of constraints, the combination of generated trips into valid plans is likely to be time consuming. It was therefore decided that the TRANSWAY user should be provided with some means for controlling the number of plans generated by the agents. In the current version of TRANSWAY this is accomplished by allowing the user to set a time limit at the beginning of the plan generation process, and by allowing the user to terminate the search process at will. Several different search methods were considered, as follows:

Simulated Annealing: This method is essentially a simulation of the annealing process in metals. A temperature value that simulates a cooling effect much like annealing is defined. This value eventually becomes cold enough to force the searching to find a close local optimum.

Genetic Algorithms: This method involves breeding solutions and applying random mutations to evolve a population of 'best fit' solutions.

Constraint Logic Programming: This method involves using a search algorithm with discrete domains to find values that satisfy the given constraints (e.g., backward chaining).

Tabu Search: This method is based on the concept that new solutions should not revisit portions of the solution space previously considered.

The Tabu Search method was selected because it is particularly suitable for the type of vehicle routing and scheduling problem encountered by TRANSWAY (Crino 2002). However, there was still a need to translate the mathematical representation of the Tabu search algorithm into the object-oriented environment of the TRANSWAY architecture. For example, in the case of trip representation, each trip contains a reference to a conveyance object and a list of 'trip legs' representing each journey that the conveyance will embark on, together with its associated cargo.

Another theoretical notion that required translation was the concept of a *move* (Crino 2002). In the Tabu environment a move is typically defined as replacing one trip in the solution with another trip. However, a trip cannot be replaced by just any other trip. Crino (2002) uses the conveyance as a convenient identifier, so that one trip can be replaced by another trip if they share the same conveyance. This is not acceptable in the case of TRANSWAY because conveyances should be able to make more than one trip. Therefore, in TRANSWAY trips are identified by the degree to which the demand for supplies is satisfied. Accordingly, a set of trips can be replaced by another set of trips that satisfies all or a subset of the demands.

Tabu Search Strategies: In the TRANSWAY implementation the Tabu agent attempts to find the best combination of trips that together form reasonable planning recommendations. The trips in this case are the atomic entities. The Tabu agent tries to add or remove trips during each iteration of the algorithm based on several strategies. It will first attempt to add trips to the current solution. If it cannot add more trips to its current solution it will remove trips and begin again.

One fundamental aspect of a Tabu search is the use of adaptive memory. By maintaining a list of taboo choices the Tabu agent is capable of diversifying its approach through the combinatorial solution space. When Tabu examines the various choices or trips that can be added to the current plan it first checks the taboo list to see if that solution has already been examined and chooses the best non-taboo option as the new incumbent solution. This approach allows the algorithm to search through a large combination of trips, while considering solutions that hold the most promise relatively quickly.

Using the Tabu agent TRANSWAY is able to find reasonable plans in a short amount of time and more optimal plans if it is allowed to continue running. Once some ending criterion has been reached the algorithm will stop and report the best solution that has been found. In the current version of TRANSWAY reporting occurs on a continuous basis as better and better solutions are found. The user may stop the search at any time.

Principal Design Components: The implementation of the Tabu algorithm in TRANSWAY can be best described in terms of two principal design components, namely *services* and *agents*. In respect to *services*, an event manager receives events from the TRANSWAY ontology through the ICDM-based subscription service. Agents acting as listeners are able to register interest in these events, which are treated as services. The following *services* have been implemented in the current version of TRANSWAY:

Request Service: This service maintains the locations, quantities, priorities, time windows, and types of supplies requested.

Conveyance Service: This service maintains the current locations and capabilities of all of the conveyances within the AOR.

Supply Service: This service maintains the locations, quantities, and types of supplies available.

Routing Service: This service listens to changes within the graph-like structure of nodes and route segments. A shortest path matrix is maintained for each type of route traversal such as air, water, and land. Accordingly, agents are able to ask the routing service whether one or more routes exist between two nodes and, if yes: What is the shortest route? Agents may also ask the routing agent to compute shortest routes based on a maximum range between refueling stops.

Several kinds of *agents* with different functional responsibilities have been implemented in TRANSWAY to collaboratively develop strategic planning solutions, as follows:

Generic Trip Generation Agents: These agents generate a set of all possible trips that satisfy all of the business rule constraints. In this regard a generic trip is composed of a vehicle traveling to a supply depot, picking up supplies, delivering those supplies to another location, and returning to its home base. However: a conveyance cannot exceed its range without refueling; a conveyance must travel on a route of its traversal type; a conveyance should try and take the shortest path when available; and, an impediment may cause the need for alternate routes.

Convoy Building Agent: This agent is responsible for constructing convoys out of trucks. The convoy then acts as another conveyance for the other agents to work with.

Advanced Trip Generation Agents: These agents take the single trips that have been generated and determine whether combining two or more of these trips could lead to greater efficiency. For example, two trips could be combined when they use the same conveyance and their time constraints are compatible.

The conveyance scheduling and routing problem falls into a class of problems that are *NP-complete*. This means that these problems grow in complexity quite fast, and it is unreasonable to try and examine every possible solution to a sizable scenario. The Tabu algorithm addresses this problem by providing good heuristics to guide searching.

A Typical TRANSWAY Scenario

The main TRANSWAY screen (Figure 3.1) is divided into two principal areas. On the left side, moving from the top down, below the main option bar the user will find: three agent icons; objects that may be placed on top of the map (the right side of the screen); a tree-structure that provides quick and convenient access to the data that the system is currently populated with; and, at the bottom a command window for the Tabu agent. On the right side of the screen is a geo-referenced map that allows the user to pan to any part of the world and, subject to the availability of maps, zoom down to street level if desired. Objects representing nodes (e.g., SAAs, APODs, etc.), route segments, impediments, and areas of interest may be moved from the left side of the screen to the right side by simple *click to locate* actions. Alternatively, the user may specify latitude-longitude locations and the selected object will be automatically placed on the map in the correct location. These objects, whether entered by the user or pre-initialized in the system, have attributes that relate to TRANSWAY's internal ontology and provide the necessary context for automated agent actions.

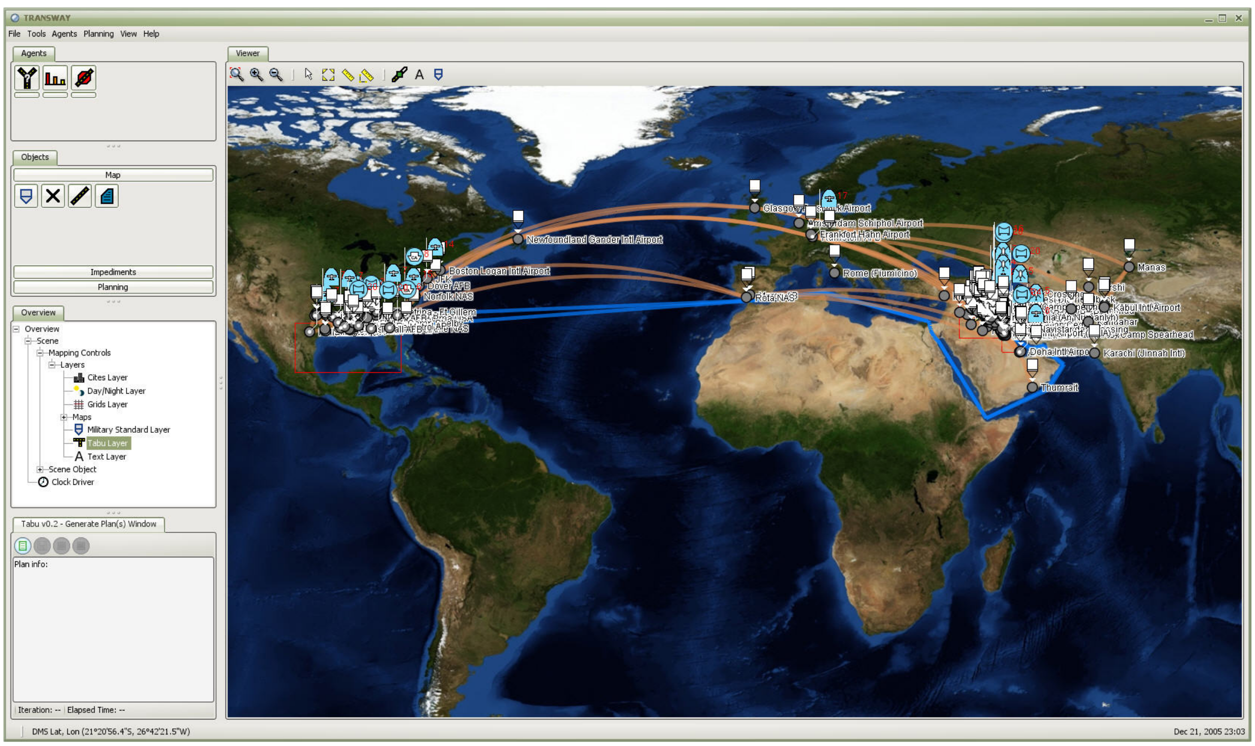


Figure 3.1: Main TRANSWAY screen

TRANSWAY is by no means limited to the current set of attributes. With the contractual goal of this first version of a prototype system to demonstrate the typical capabilities of an ontology-based multi-agent system, attributes were selected in a fairly generic fashion based on the feedback that the development team received during early demonstrations, perusal of military documents, and in-house experience with other logistic planning systems.

Supply Center Summary

Supply Center	MOG Parking	MOG Working	Throughput	Short T...	Fixed Wing	Rotary Wing	Vessels	Vehicles
Charleston AFB	15	10	0.0	8,289.2	15	0	8	0
Dover AFB	15	10	0.0	4,361.3	14	0	0	0
Ramstein AFB	15	10	0.0	966.1	17	0	0	0
Ash Shuaybah	20	10	0.0	966.1	0	0	0	24
Kuwait Intl Airport (KCIA)	6	5	0.0	966.1	4	5	0	20
Al Taqaddum AB	2	2	0.0	966.1	0	4	0	10
Al Udeid AB	2	2	0.0	966.1	8	5	0	0
Bagram AB	2	2	0.0	966.1	0	0	0	0
Balad Southeast/Camp Anaconda	2	2	0.0	966.1	1	7	0	16
Kandahar	2	2	0.0	966.1	0	0	0	0
Barksdale AFB	15	10	0.0	432.5	7	0	0	0
JTF Katrina - Ft Gillem	10	6	0.0	432.5	7	0	0	0
Kelly/Lackland	10	6	0.0	432.5	7	0	0	0
Fort Worth NAS	15	6	0.0	432.5	7	0	0	0
Jacksonville NAS	0	0	0.0	57.7	0	0	6	0
Dallas-FT Worth IAP	15	10	0.0	0.0	0	0	0	0
England AFB	15	10	0.0	0.0	0	0	0	0
George Bush IAP	15	10	0.0	0.0	0	0	0	0
Glasgow Prestwick Airport	15	10	0.0	0.0	0	0	0	0
JFK	15	10	0.0	0.0	0	0	0	0
Maxwell AFB	15	10	0.0	0.0	0	0	0	0
Newfoundland Gander Intl Airport	15	10	0.0	0.0	0	0	0	0
Tydall AFB	15	10	0.0	0.0	0	0	0	20
Mobile Regional AP	10	6	0.0	0.0	0	0	0	20
NAS Meridian	10	6	0.0	0.0	0	0	0	0
Rota NAS	10	6	0.0	0.0	0	0	0	0
William P. Hobby	10	6	0.0	0.0	0	0	0	0
Camp Najaf	10	5	0.0	0.0	0	0	0	0
Camp Navistar	10	5	0.0	0.0	0	0	0	0
Camp Scania (An Numanlyh)	10	5	0.0	0.0	0	0	0	0
Louis Armstrong IAP	0	4	0.0	0.0	0	0	0	0
Al Asad AB	5	3	0.0	0.0	0	0	0	0
Baton Rouge Metro	5	3	0.0	0.0	0	0	0	20
Ellington, TX	5	3	0.0	0.0	0	0	0	0
Gulf Port IAP	5	3	0.0	0.0	0	0	0	0
JTF FWD - Camp Shelby	5	3	0.0	0.0	0	0	0	20
Karshi	5	3	0.0	0.0	0	0	0	0
Lafayette IAP	5	3	0.0	0.0	0	0	0	20
Manas	5	3	0.0	0.0	0	0	0	0
Thumrait	5	3	0.0	0.0	0	0	0	0
Kessler AFB	0	2	0.0	0.0	0	0	0	0
Al Sahra AB/Camp Speicher	3	2	0.0	0.0	0	0	0	0
Al Qayyarah West	4	2	0.0	0.0	0	0	0	0
Herat	4	2	0.0	0.0	0	0	0	0
Kirkuk AB	4	2	0.0	0.0	0	4	0	0
Mosul AB/Camp Diamondback	4	2	0.0	0.0	0	0	0	0
Tallil AB/Camp Cedar	4	2	0.0	0.0	0	0	0	0
Ali Al Salem AB	2	1	0.0	0.0	0	0	0	0
Baghdad Intl Airport (BIAP)	2	1	0.0	0.0	2	8	0	20
Kabul Intl Airport	4	1	0.0	0.0	0	0	0	0
Ad Diwaniyah	0	0	0.0	0.0	0	0	0	0

Figure 3.2: Summary of supplies and available conveyances at supply centers

The report shown in Figure 3.2 provides a summary of supplies (short tons) and available conveyances (i.e., fixed wing aircraft, helicopters, ships, and trucks (in convoys)) at most supply centers currently initialized in the system for this particular demonstration scenario. Details of supplies at Charleston and Al Udeid are shown in Figures 3.3 and 3.4 (in terms of supply Class, number of pallets, number of items per pallet, and short tons), respectively.

Inventory Supply Summary

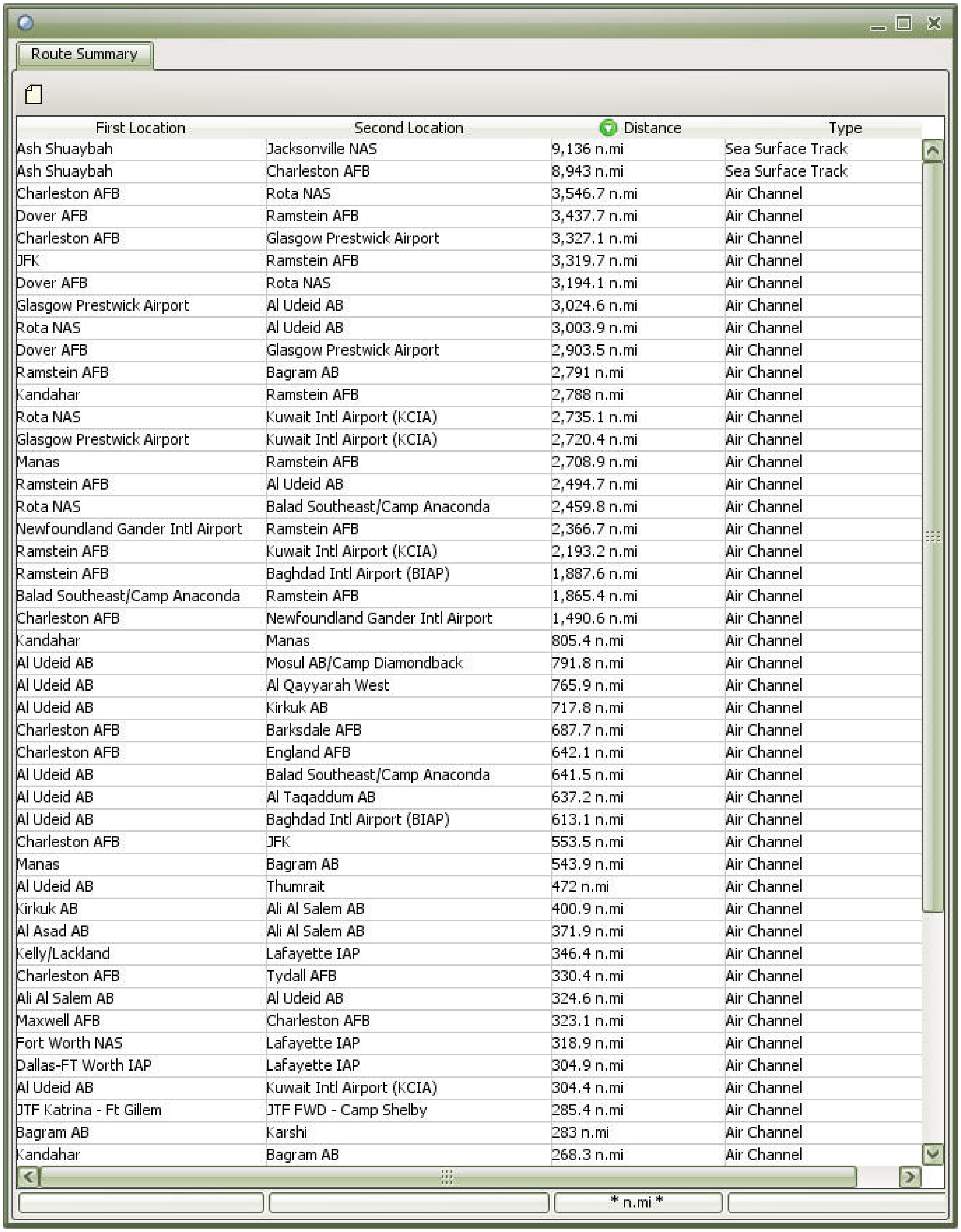
Name	NSN	Supply Class	# Pallets	Items/Pallet	Short Tons
BALLISTIC WINDSHIELD (Group)	2510-01-523-4504	IX	2	8	34.9
FMTV LOW SIGNATURE ARMORED CAB (LSAC) KIT (Group)	2510-01-523-0059	IX	206	1	716.4
FMTV RADIAN ARMORED CAB (RAC) KIT (Group)	2540-01-519-0377	IX	558	1	1,842.8
HEMTT A/C KIT (Group)	4120-01-526-9153	IX	477	12	943.3
HEMTT ARMOR KIT (Group)	2540-01-520-6821	IX	477	1	1,086.4
HET A/C KIT (Group)	4120-01-505-4149	IX	2	1	0.9
HET ARMOR KIT (Group)	2540-01-520-6826	IX	2	1	6.0
HMMWV AIR CONDITIONERS (Group)	4130-01-523-3966	IX	2	8	18.7
HMMWV GSIE 2 DOOR KIT (Group)	2510-01-514-9688	IX	2	1	1.4
HMMWV GSIE 4 DOOR KIT (Group)	2510-01-514-9710	IX	2	1	2.0
HMMWV O GARA HESS 2 DOOR KIT (Group)	2510-01-524-2948	IX	2	1	2.2
HMMWV O GARA HESS 4 DOOR KIT (Group)	2510-01-524-2937	IX	2	1	2.8
M915A2 CABS (Group)	2540-01-523-1336	IX	2	2	5.6
M939 ARMOR CABS (Group)	2540-01-522-3749	IX	1009	1	3,180.9
M998 TROOP CARRIER KITS (Group)	2540-01-525-3585	IX	2	2	4.6
MEAL READY TO EAT - MRE CASE (Group)	8970-00-149-1094	I	100	395	432.5
PLS A/C KIT (Group)	4120-01-526-9158	IX	2	6	3.1
PLS ARMOR KIT (Group)	2540-01-520-6819	IX	2	1	4.8

Figure 3.3: Details of supplies at Charleston

Inventory Supply Summary

Name	NSN	Supply Class	# Pallets	Items/Pallet	Short Tons
CARTRIDGE, .50 CAL (Group)	1305-00-028-6603	V	140	48	276.8
CARTRIDGE, 120MM (Group)	1315-01-232-4638	V	50	30	65.1
CARTRIDGE, 5.56MM (Group)	1305-01-155-5459	V	180	48	321.4
MEAL READY TO EAT - MRE CASE (Group)	8970-00-149-1094	I	70	395	302.8

Figure 3.4: Details of supplies at Al Udeid



Route Summary

First Location	Second Location	Distance	Type
Ash Shuaybah	Jacksonville NAS	9,136 n.mi	Sea Surface Track
Ash Shuaybah	Charleston AFB	8,943 n.mi	Sea Surface Track
Charleston AFB	Rota NAS	3,546.7 n.mi	Air Channel
Dover AFB	Ramstein AFB	3,437.7 n.mi	Air Channel
Charleston AFB	Glasgow Prestwick Airport	3,327.1 n.mi	Air Channel
JFK	Ramstein AFB	3,319.7 n.mi	Air Channel
Dover AFB	Rota NAS	3,194.1 n.mi	Air Channel
Glasgow Prestwick Airport	Al Udeid AB	3,024.6 n.mi	Air Channel
Rota NAS	Al Udeid AB	3,003.9 n.mi	Air Channel
Dover AFB	Glasgow Prestwick Airport	2,903.5 n.mi	Air Channel
Ramstein AFB	Bagram AB	2,791 n.mi	Air Channel
Kandahar	Ramstein AFB	2,788 n.mi	Air Channel
Rota NAS	Kuwait Intl Airport (KCIA)	2,735.1 n.mi	Air Channel
Glasgow Prestwick Airport	Kuwait Intl Airport (KCIA)	2,720.4 n.mi	Air Channel
Manas	Ramstein AFB	2,708.9 n.mi	Air Channel
Ramstein AFB	Al Udeid AB	2,494.7 n.mi	Air Channel
Rota NAS	Balad Southeast/Camp Anaconda	2,459.8 n.mi	Air Channel
Newfoundland Gander Intl Airport	Ramstein AFB	2,366.7 n.mi	Air Channel
Ramstein AFB	Kuwait Intl Airport (KCIA)	2,193.2 n.mi	Air Channel
Ramstein AFB	Baghdad Intl Airport (BIAP)	1,887.6 n.mi	Air Channel
Balad Southeast/Camp Anaconda	Ramstein AFB	1,865.4 n.mi	Air Channel
Charleston AFB	Newfoundland Gander Intl Airport	1,490.6 n.mi	Air Channel
Kandahar	Manas	805.4 n.mi	Air Channel
Al Udeid AB	Mosul AB/Camp Diamondback	791.8 n.mi	Air Channel
Al Udeid AB	Al Qayyarah West	765.9 n.mi	Air Channel
Al Udeid AB	Kirkuk AB	717.8 n.mi	Air Channel
Charleston AFB	Barksdale AFB	687.7 n.mi	Air Channel
Charleston AFB	England AFB	642.1 n.mi	Air Channel
Al Udeid AB	Balad Southeast/Camp Anaconda	641.5 n.mi	Air Channel
Al Udeid AB	Al Taqaddum AB	637.2 n.mi	Air Channel
Al Udeid AB	Baghdad Intl Airport (BIAP)	613.1 n.mi	Air Channel
Charleston AFB	JFK	553.5 n.mi	Air Channel
Manas	Bagram AB	543.9 n.mi	Air Channel
Al Udeid AB	Thumrait	472 n.mi	Air Channel
Kirkuk AB	Ali Al Salem AB	400.9 n.mi	Air Channel
Al Asad AB	Ali Al Salem AB	371.9 n.mi	Air Channel
Kelly/Lackland	Lafayette IAP	346.4 n.mi	Air Channel
Charleston AFB	Tydall AFB	330.4 n.mi	Air Channel
Ali Al Salem AB	Al Udeid AB	324.6 n.mi	Air Channel
Maxwell AFB	Charleston AFB	323.1 n.mi	Air Channel
Fort Worth NAS	Lafayette IAP	318.9 n.mi	Air Channel
Dallas-FT Worth IAP	Lafayette IAP	304.9 n.mi	Air Channel
Al Udeid AB	Kuwait Intl Airport (KCIA)	304.4 n.mi	Air Channel
JTF Katrina - Ft Gillem	JTF FWD - Camp Shelby	285.4 n.mi	Air Channel
Bagram AB	Karshi	283 n.mi	Air Channel
Kandahar	Bagram AB	268.3 n.mi	Air Channel

* n.mi *

Figure 3.5: Summary report of air channels and sea routes

Figure 3.5 provides information about the air channels and sea routes that the system has been initialized with for this particular demonstration scenario. In each case the two end-points and the distance in nautical miles is indicated.

Detailed information about the current compliment of conveyances can be obtained by selecting the appropriate report. Typical examples for various fixed wing aircraft, trucks and ships are shown in Figures 3.6 to 3.11, below. The reason that the *speed* and *bearing* attributes in each table are zero is because the conveyances are not currently in-transit.

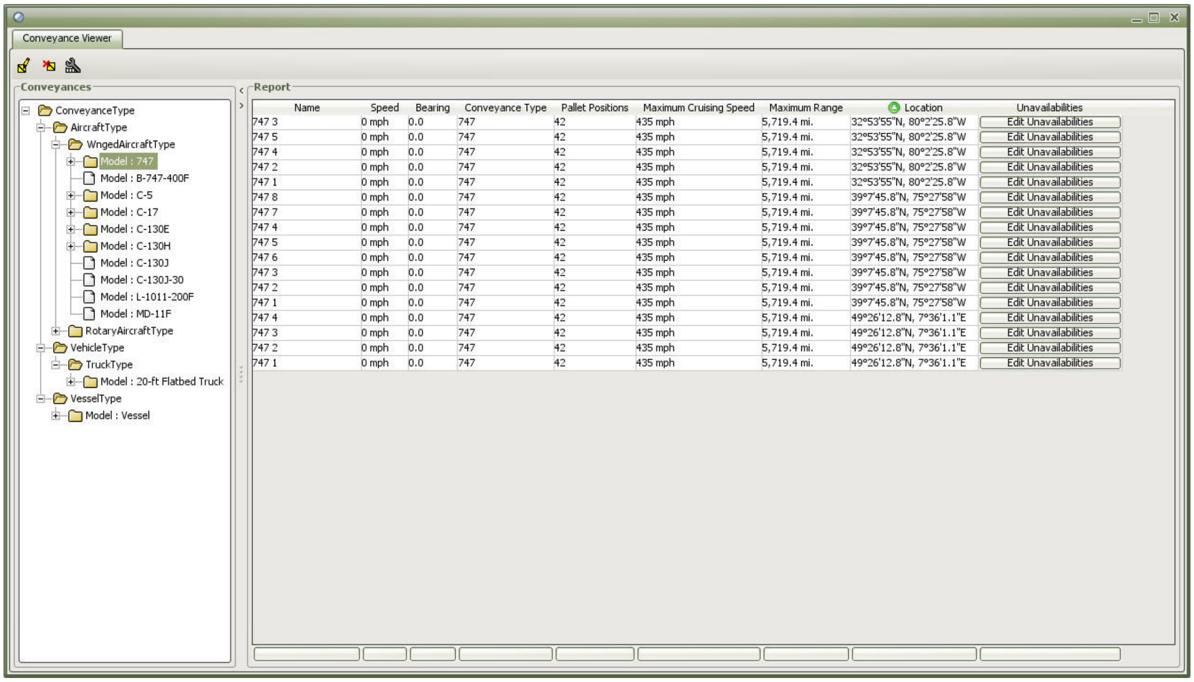


Figure 3.6: Boeing 747 aircraft attributes

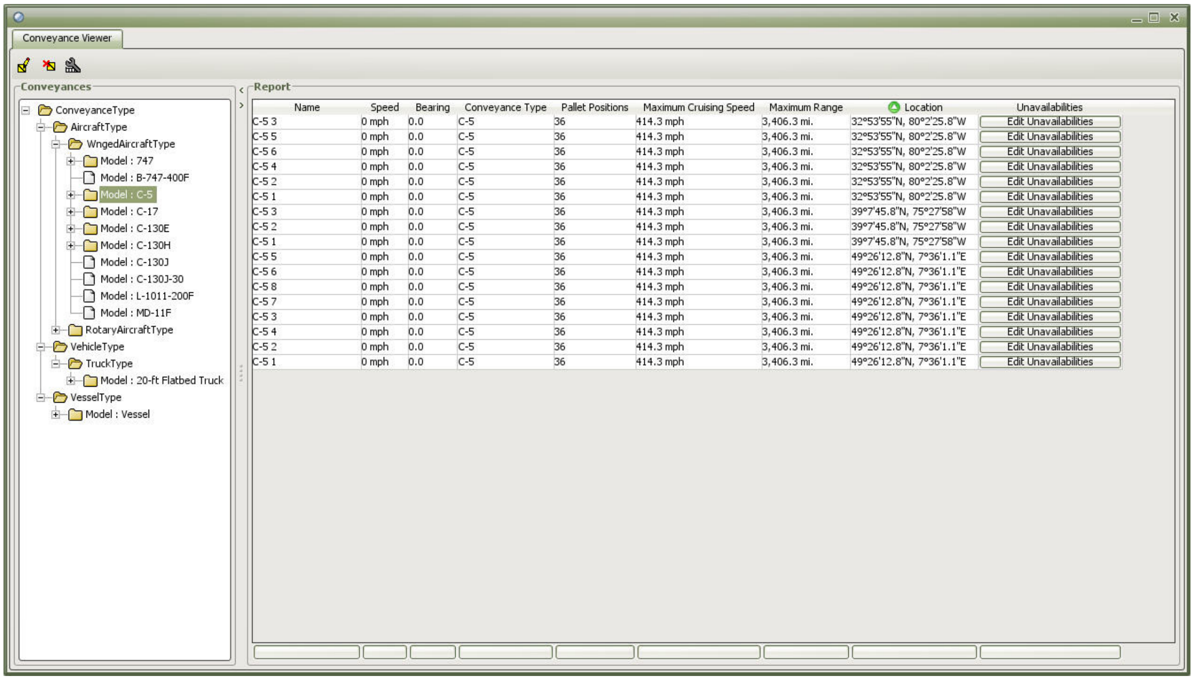
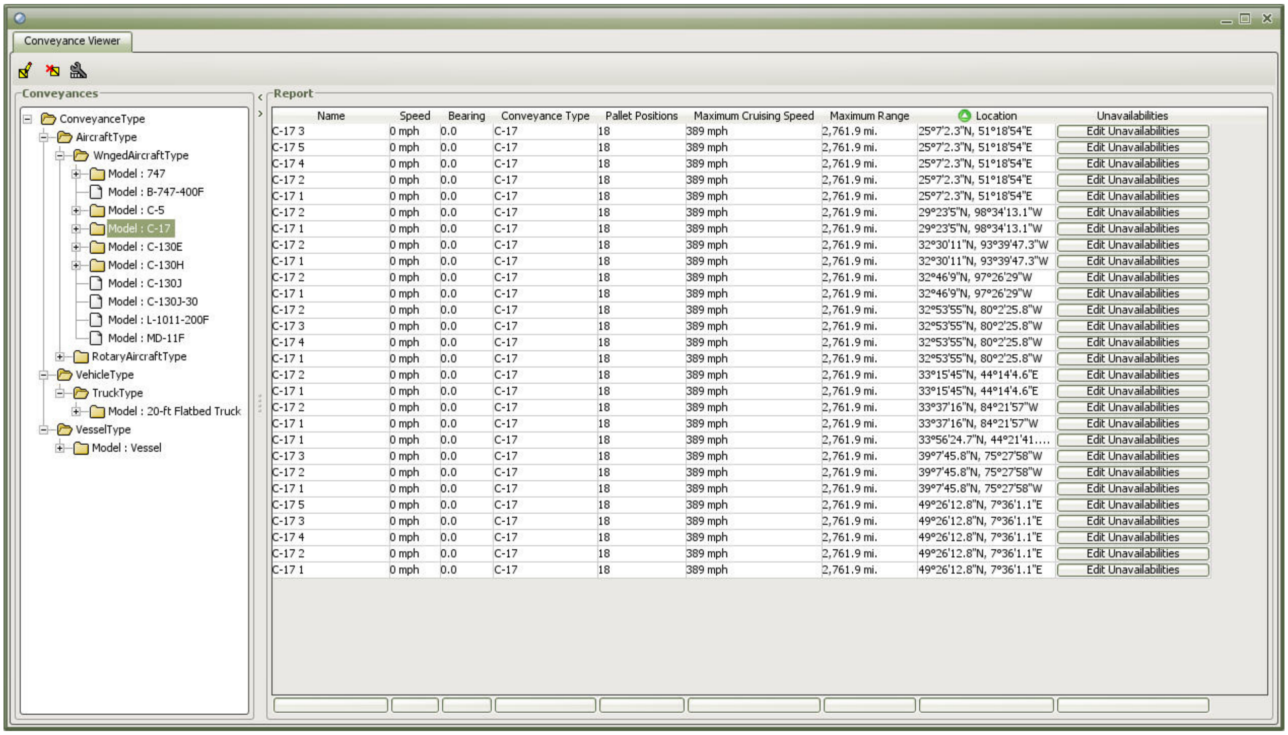
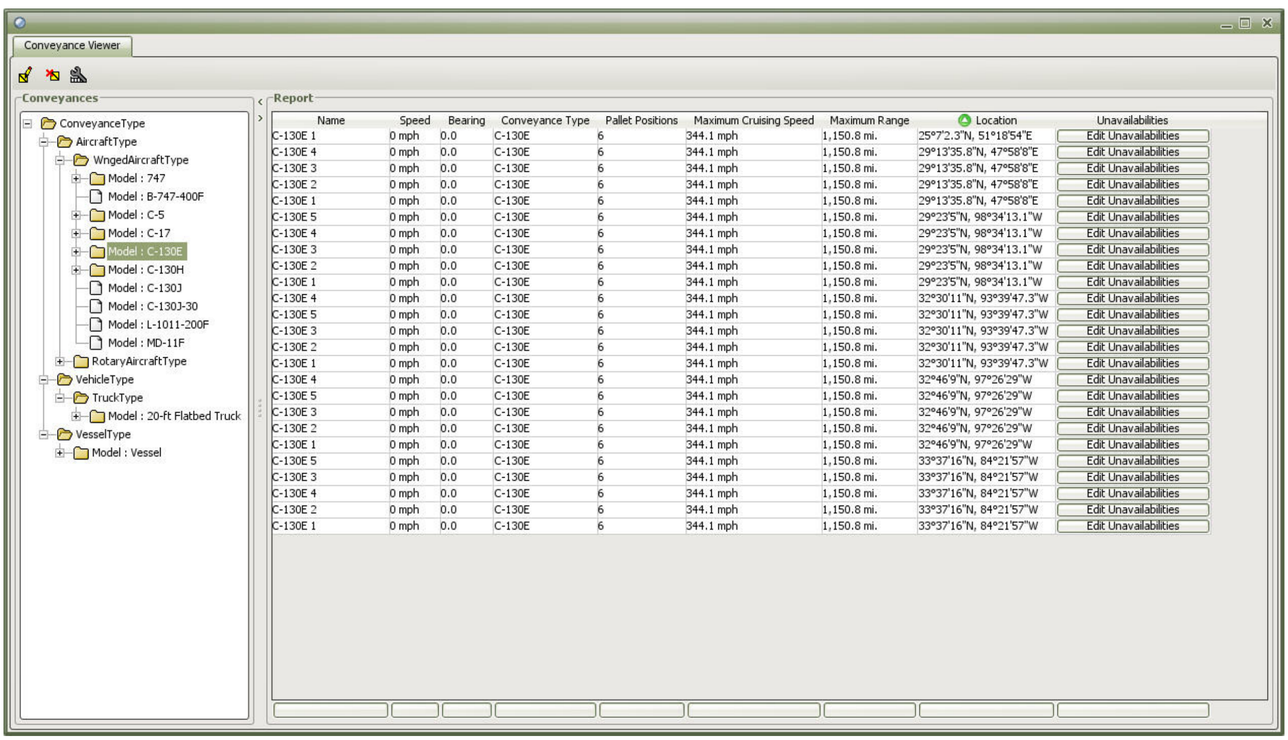


Figure 3.7: C5 aircraft attributes



Name	Speed	Bearing	Conveyance Type	Pallet Positions	Maximum Cruising Speed	Maximum Range	Location	Unavailabilities
C-17 3	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	25°7'2.3"N, 51°18'54"E	Edit Unavailabilities
C-17 5	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	25°7'2.3"N, 51°18'54"E	Edit Unavailabilities
C-17 4	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	25°7'2.3"N, 51°18'54"E	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	25°7'2.3"N, 51°18'54"E	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	25°7'2.3"N, 51°18'54"E	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	29°23'5"N, 98°34'13.1"W	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	29°23'5"N, 98°34'13.1"W	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°30'11"N, 93°39'47.3"W	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°30'11"N, 93°39'47.3"W	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°46'9"N, 97°26'29"W	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°46'9"N, 97°26'29"W	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°53'55"N, 80°2'25.8"W	Edit Unavailabilities
C-17 3	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°53'55"N, 80°2'25.8"W	Edit Unavailabilities
C-17 4	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°53'55"N, 80°2'25.8"W	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	32°53'55"N, 80°2'25.8"W	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	33°15'45"N, 44°14'4.6"E	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	33°15'45"N, 44°14'4.6"E	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	33°37'16"N, 84°21'57"W	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	33°37'16"N, 84°21'57"W	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	33°56'24.7"N, 44°21'41....	Edit Unavailabilities
C-17 3	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	39°7'45.8"N, 75°27'58"W	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	39°7'45.8"N, 75°27'58"W	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	39°7'45.8"N, 75°27'58"W	Edit Unavailabilities
C-17 5	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	49°26'12.8"N, 7°36'1.1"E	Edit Unavailabilities
C-17 3	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	49°26'12.8"N, 7°36'1.1"E	Edit Unavailabilities
C-17 4	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	49°26'12.8"N, 7°36'1.1"E	Edit Unavailabilities
C-17 2	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	49°26'12.8"N, 7°36'1.1"E	Edit Unavailabilities
C-17 1	0 mph	0.0	C-17	18	389 mph	2,761.9 mi.	49°26'12.8"N, 7°36'1.1"E	Edit Unavailabilities

Figure 3.8: C17 aircraft attributes



Name	Speed	Bearing	Conveyance Type	Pallet Positions	Maximum Cruising Speed	Maximum Range	Location	Unavailabilities
C-130E 1	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	25°7'2.3"N, 51°18'54"E	Edit Unavailabilities
C-130E 4	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°13'35.8"N, 47°58'8"E	Edit Unavailabilities
C-130E 3	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°13'35.8"N, 47°58'8"E	Edit Unavailabilities
C-130E 2	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°13'35.8"N, 47°58'8"E	Edit Unavailabilities
C-130E 1	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°13'35.8"N, 47°58'8"E	Edit Unavailabilities
C-130E 5	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°23'5"N, 98°34'13.1"W	Edit Unavailabilities
C-130E 4	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°23'5"N, 98°34'13.1"W	Edit Unavailabilities
C-130E 3	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°23'5"N, 98°34'13.1"W	Edit Unavailabilities
C-130E 2	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°23'5"N, 98°34'13.1"W	Edit Unavailabilities
C-130E 1	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	29°23'5"N, 98°34'13.1"W	Edit Unavailabilities
C-130E 4	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°30'11"N, 93°39'47.3"W	Edit Unavailabilities
C-130E 5	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°30'11"N, 93°39'47.3"W	Edit Unavailabilities
C-130E 3	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°30'11"N, 93°39'47.3"W	Edit Unavailabilities
C-130E 2	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°30'11"N, 93°39'47.3"W	Edit Unavailabilities
C-130E 1	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°30'11"N, 93°39'47.3"W	Edit Unavailabilities
C-130E 4	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°46'9"N, 97°26'29"W	Edit Unavailabilities
C-130E 5	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°46'9"N, 97°26'29"W	Edit Unavailabilities
C-130E 3	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°46'9"N, 97°26'29"W	Edit Unavailabilities
C-130E 2	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°46'9"N, 97°26'29"W	Edit Unavailabilities
C-130E 1	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	32°46'9"N, 97°26'29"W	Edit Unavailabilities
C-130E 5	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	33°37'16"N, 84°21'57"W	Edit Unavailabilities
C-130E 3	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	33°37'16"N, 84°21'57"W	Edit Unavailabilities
C-130E 4	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	33°37'16"N, 84°21'57"W	Edit Unavailabilities
C-130E 2	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	33°37'16"N, 84°21'57"W	Edit Unavailabilities
C-130E 1	0 mph	0.0	C-130E	6	344.1 mph	1,150.8 mi.	33°37'16"N, 84°21'57"W	Edit Unavailabilities

Figure 3.9: C130 aircraft attributes

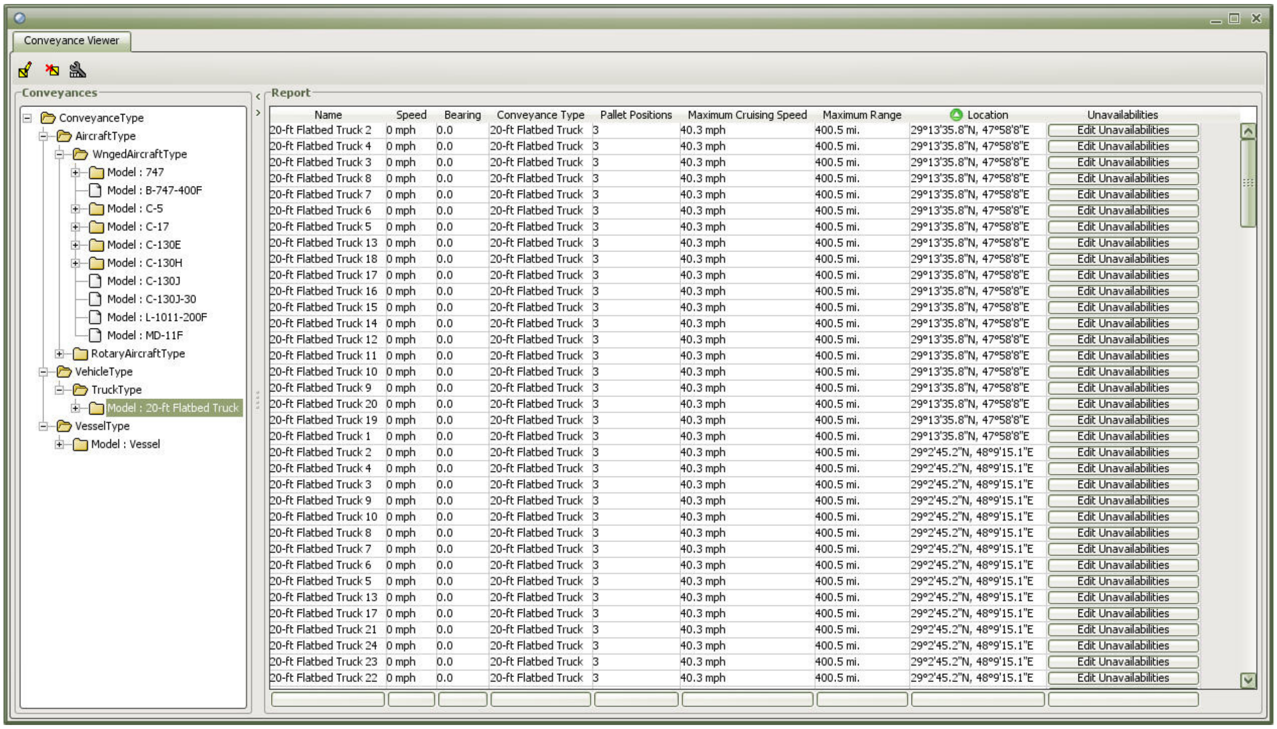


Figure 3.10: Truck convoy attributes

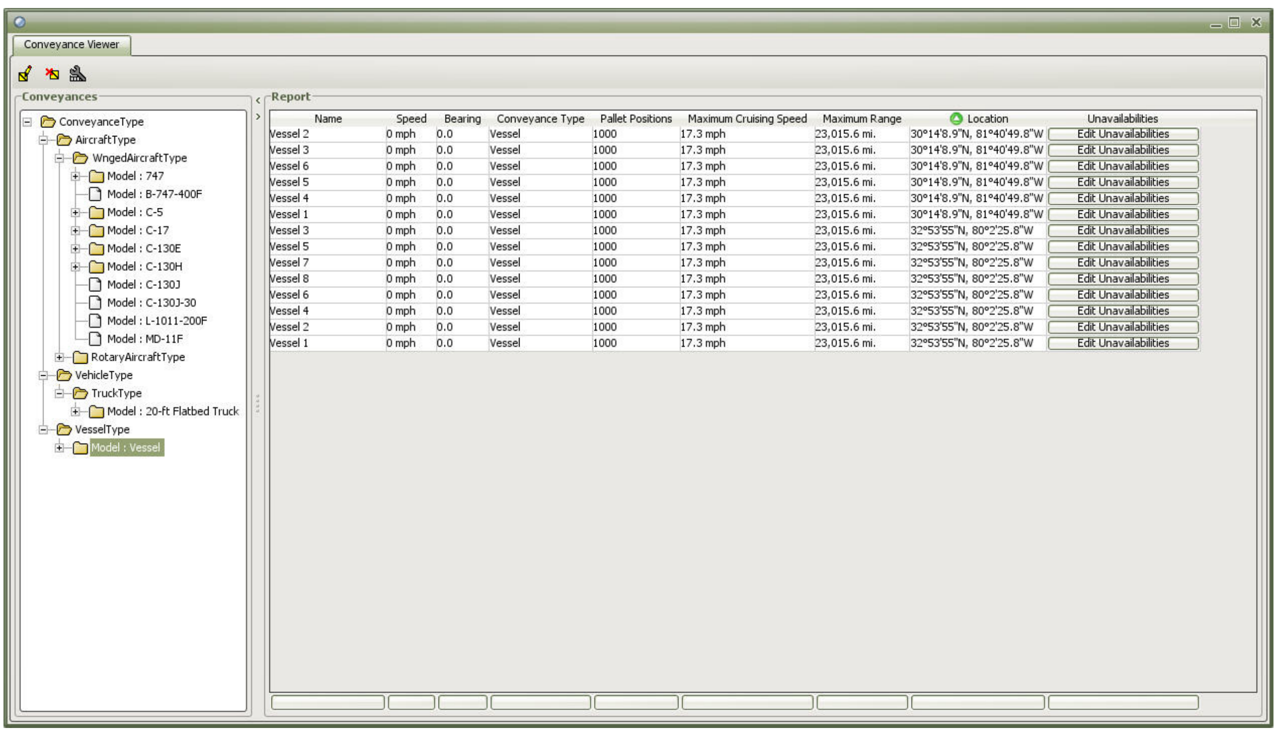


Figure 3.11: Typical ship attributes

A typical request for *add on armor* is shown in Figure 3.12. It requires deliver to Al Udeid, with a *high* priority and an earliest and latest time for delivery window of 25 to 31 December 2005.

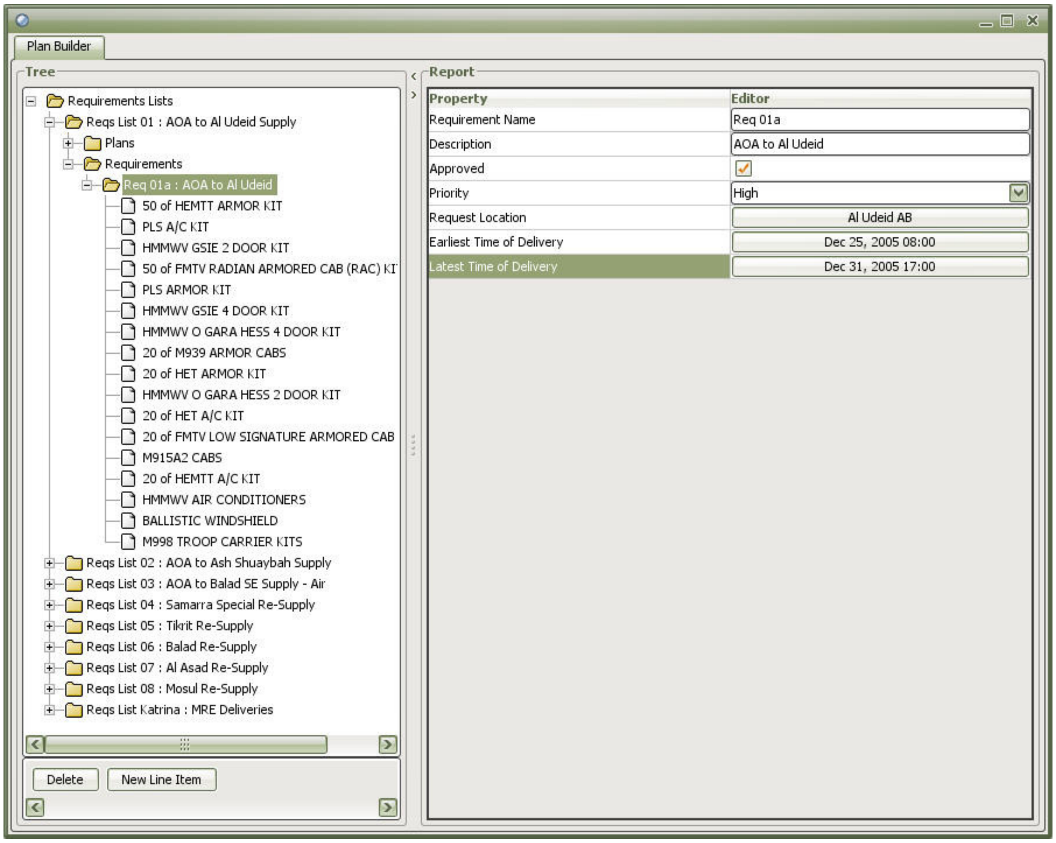


Figure 3.12: Add-on-Armor (AOR) request for delivery to Al Udeid

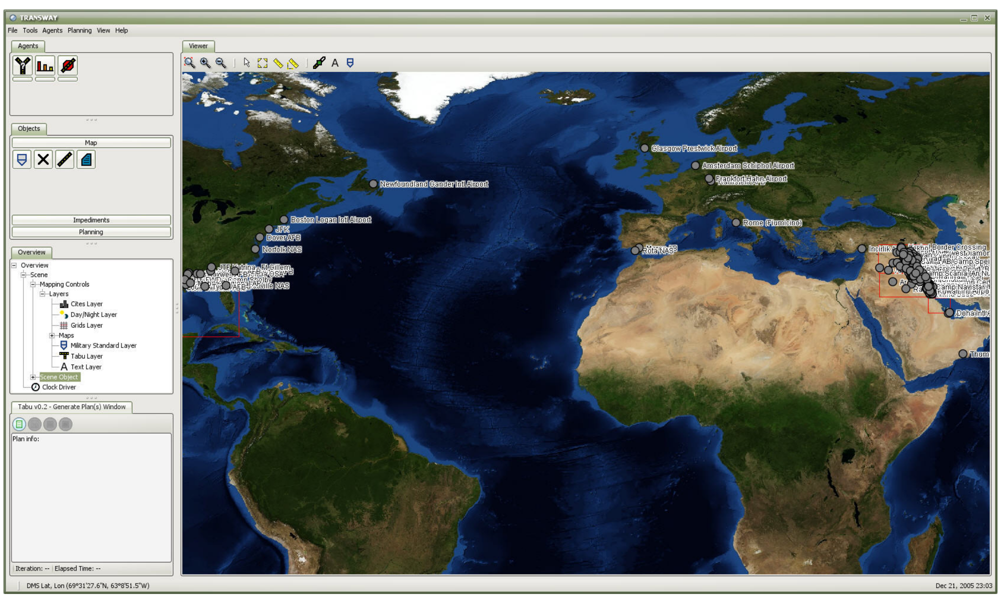


Figure 3.13: User zooms in on map to reduce clutter

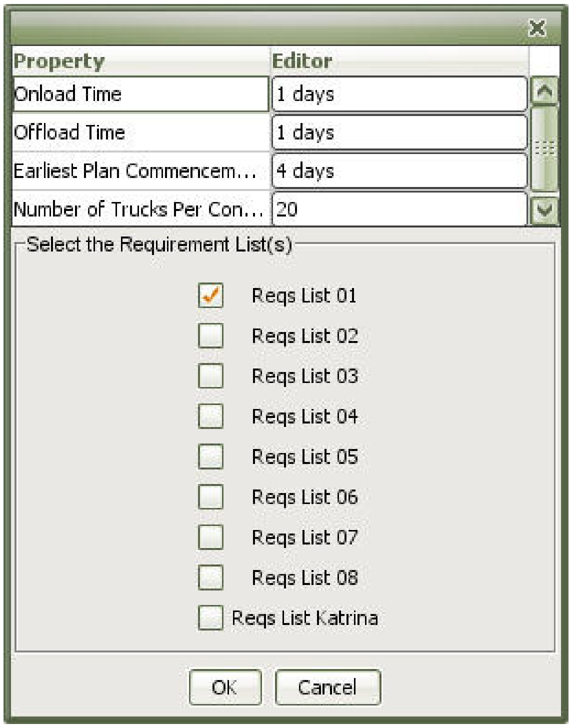


Figure 3.14: Tabu agent interface

Figure 3.15: Control of search duration

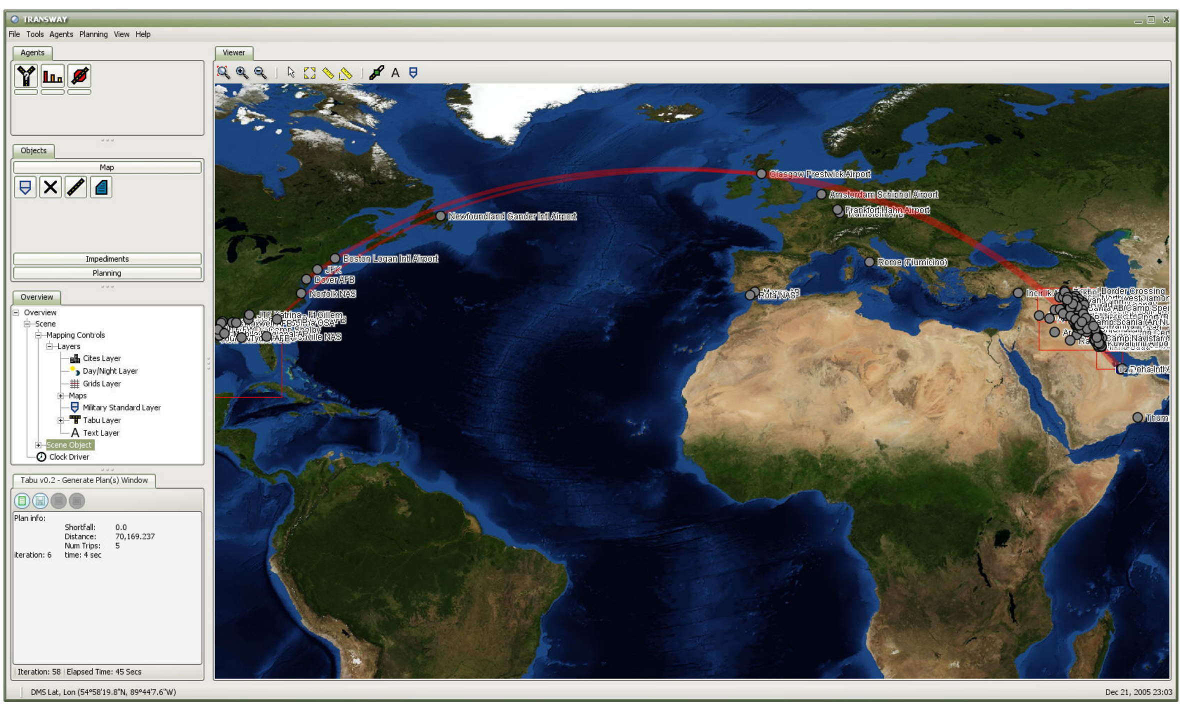


Figure 3.16: Completed first plan showing routes

To fulfill the request for the shipment of *add-on-armor* to Al Udeid (Figure 3.12) the user activates the Tabu agent and selects the appropriate *requirement* from the displayed Requirement Lists (Figure 3.14). In this case the Al Udeid *requirement* is Requirement List 1. Since the Tabu agent has the ability to continue its search for an optimum delivery plan even after it has found a way of satisfying the *requirement*, the user has the option of either setting a maximum time for the planning activity (Figure 3.15) or allowing the agent to continue until all alternatives have been explored. Of course it is not expected that the user would ever want to wait for that length of time and therefore the option for the user to simply stop the agent is available. In future versions of TRANSWAY, particularly if the Tabu agent were to be implemented in an opportunistic mode (i.e., in a manner that would activate the planning process without user involvement as soon as the conditions on which an existing plan were originally based have changed), it would be a relatively simple matter to restrict the extensiveness of the search for an optimum plan. For example, the search could be automatically aborted if after either a specified period of time or a given number of generated plans no better plan has been found.

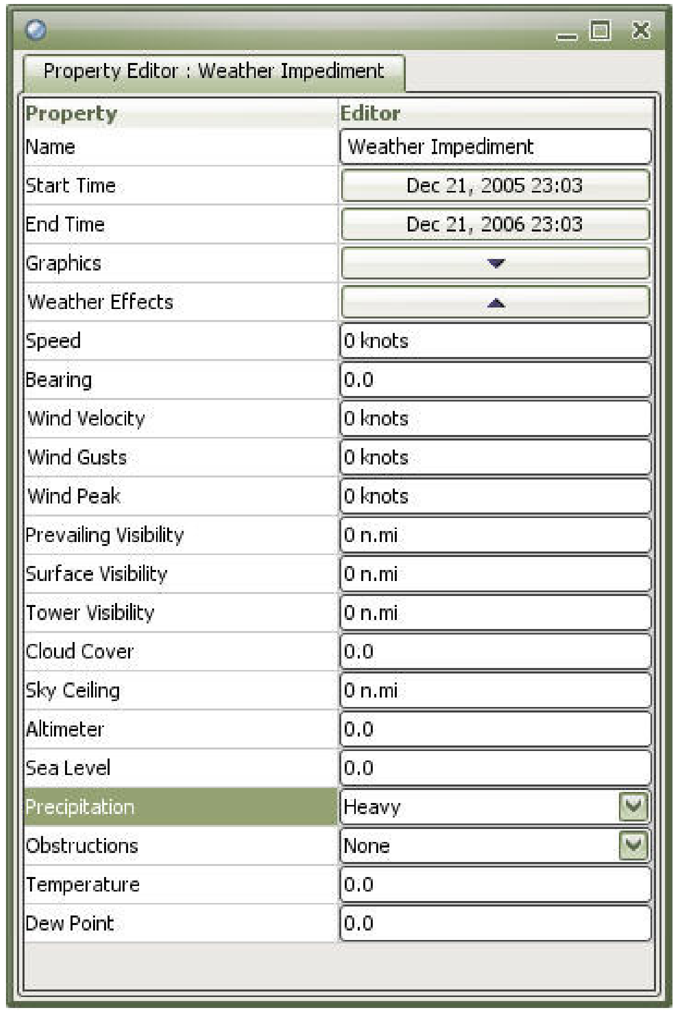


Figure 3.17: Weather impediment

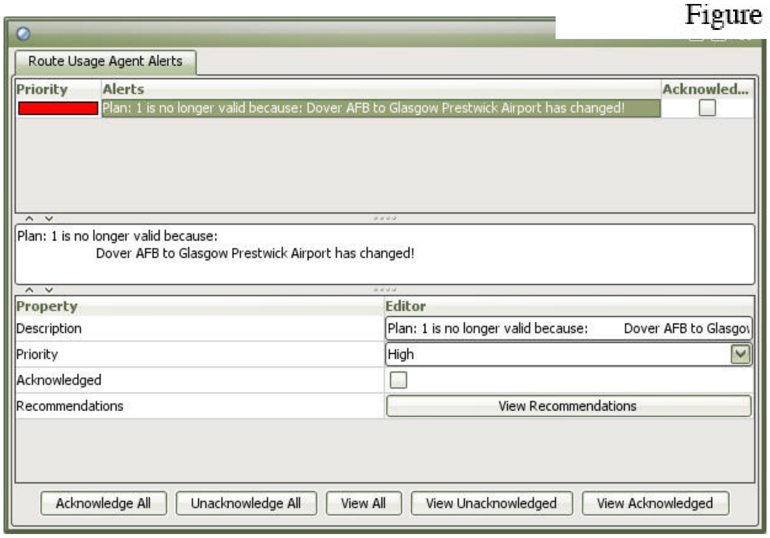


Figure 3.18: Impediment agent alert

For the completed plan the route is shown in Figure 3.16 by means of a red line. Next the user enters an impediment in the form of an adverse weather report that essentially eliminates

Glasgow as a refueling stop (Figure 3.17). Immediately, the Impediment agent alerts the user and suggests that re-planning is in order (Figure 3.18). Again, also in the case of impediments, this first version of TRANSWAY provides only one type of generic impediment (i.e., a weather condition), with the objective of demonstrating the kinds of causes that would require re-planning that could be easily implemented in subsequent versions of the system, based on user preferences and priorities.

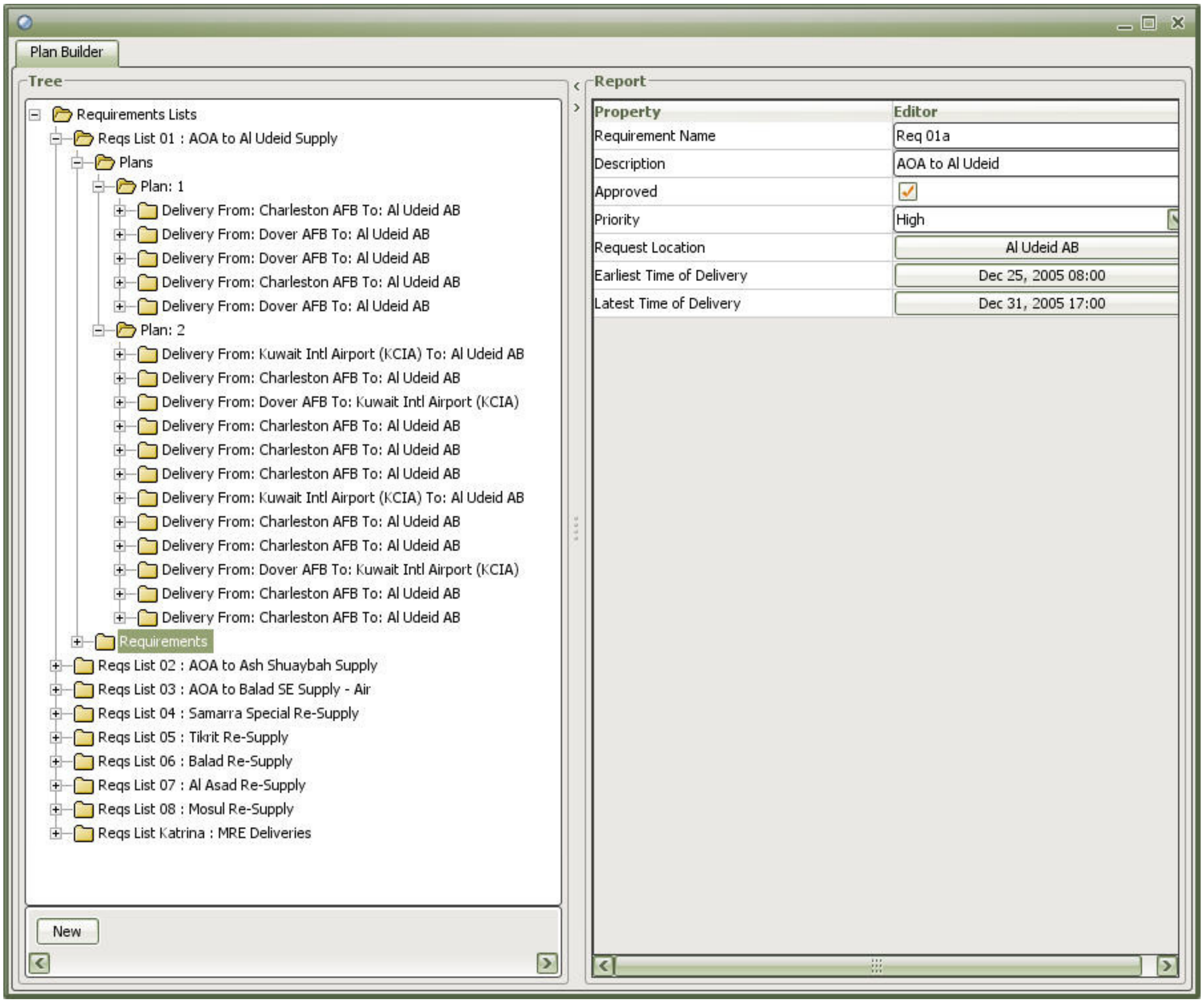


Figure 3.19: Summary of deliveries for the first and second plans

To initiate a re-planning action the user proceeds in the same manner as described previously for the generation of the first plan (Figures 3.14 to 3.16). The user will notice that during the generation of each plan the routes that are being explored by the Tabu agent are dynamically indicated on the map display. Temporarily displayed green lines indicate drop-off points that are being considered. Red lines indicate actual delivery routes with the thickness of the red line providing a proportional indication of the volume of supplies being transported along that particular route. Summary lists of the deliveries involved in both plans are shown in Figure 3.19. Even thought this first test-bed version of TRANSWAY is purposely limited in scope it does allow the user to explore the details of each delivery plan (i.e., start and end locations, conveyances and routes used, start and end times, and duration of each trip), as shown in Figures 3.20 to 3.23.

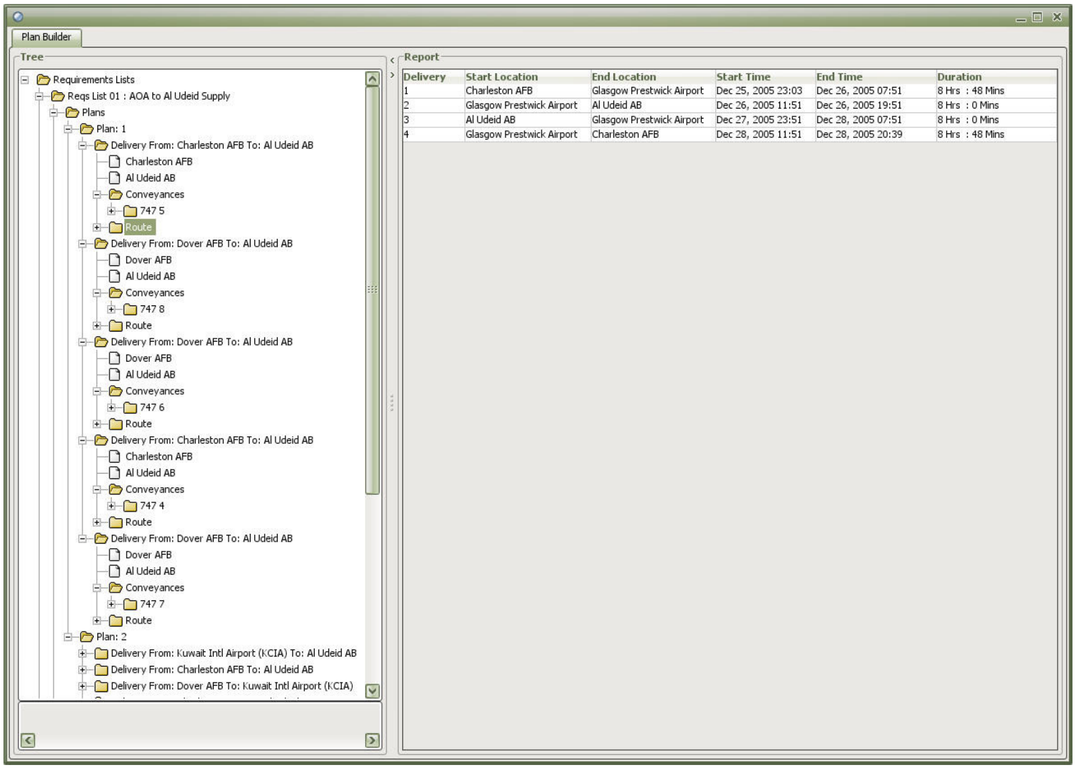


Figure 3.20: Typical drill-down details of the first plan

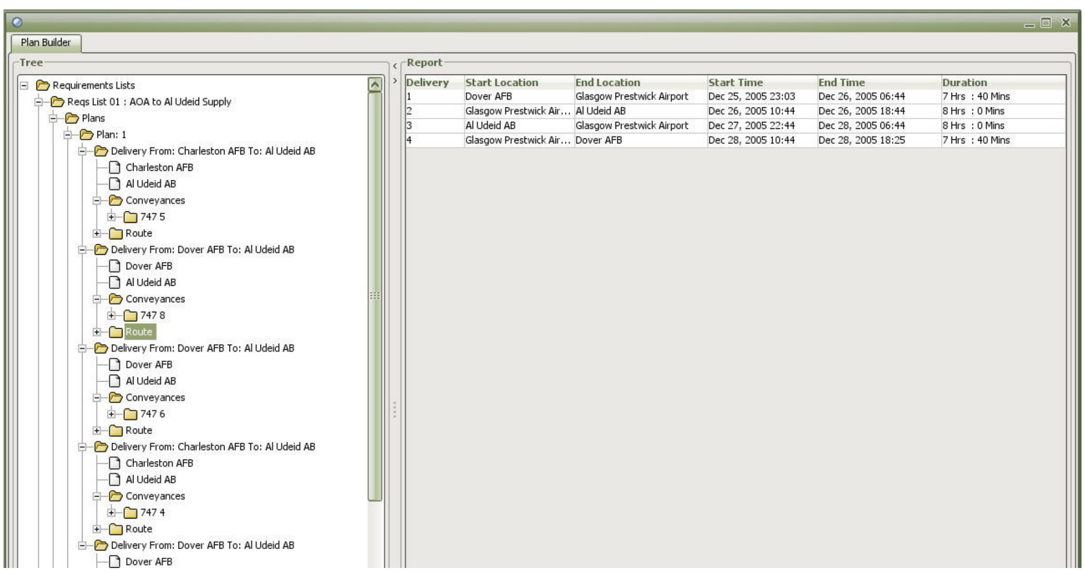


Figure 3.21: Typical drill-down details of the first plan

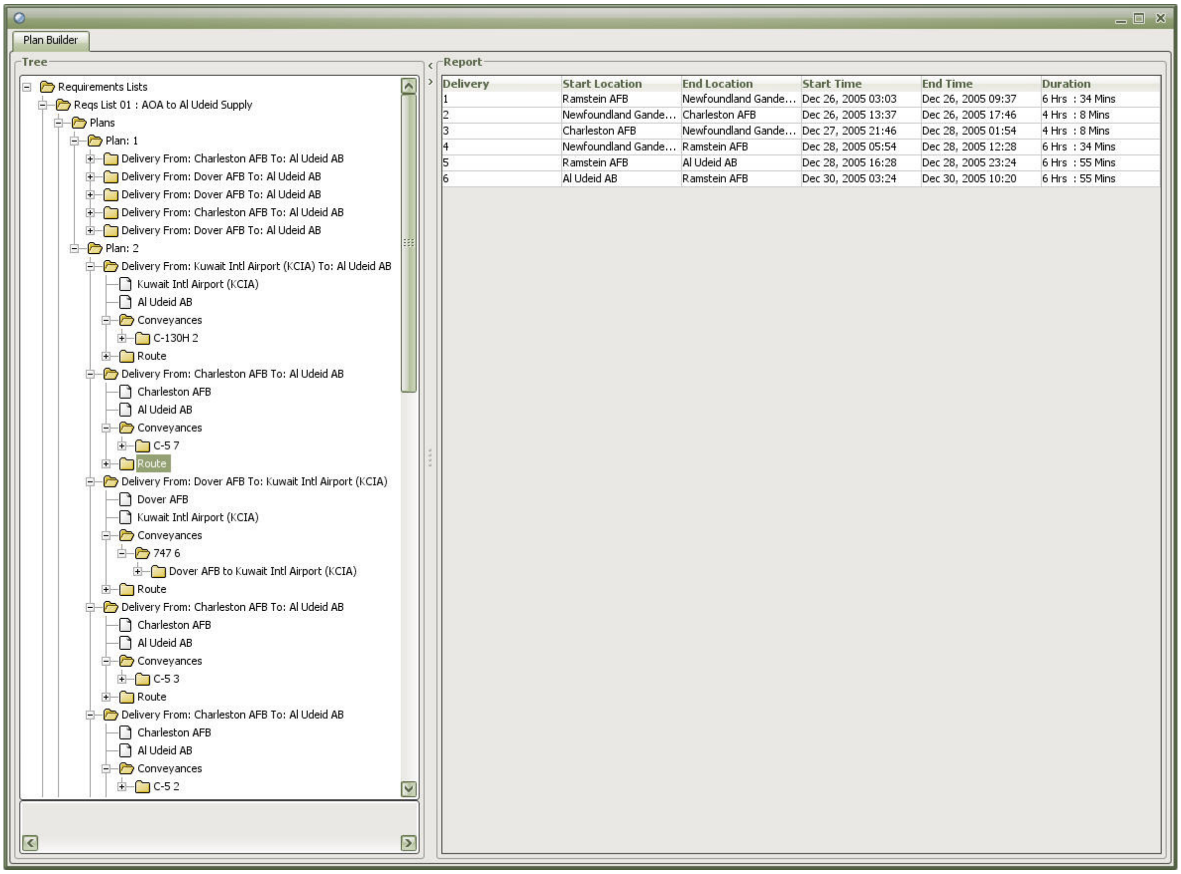


Figure 3.22: Typical drill-down details of the second plan

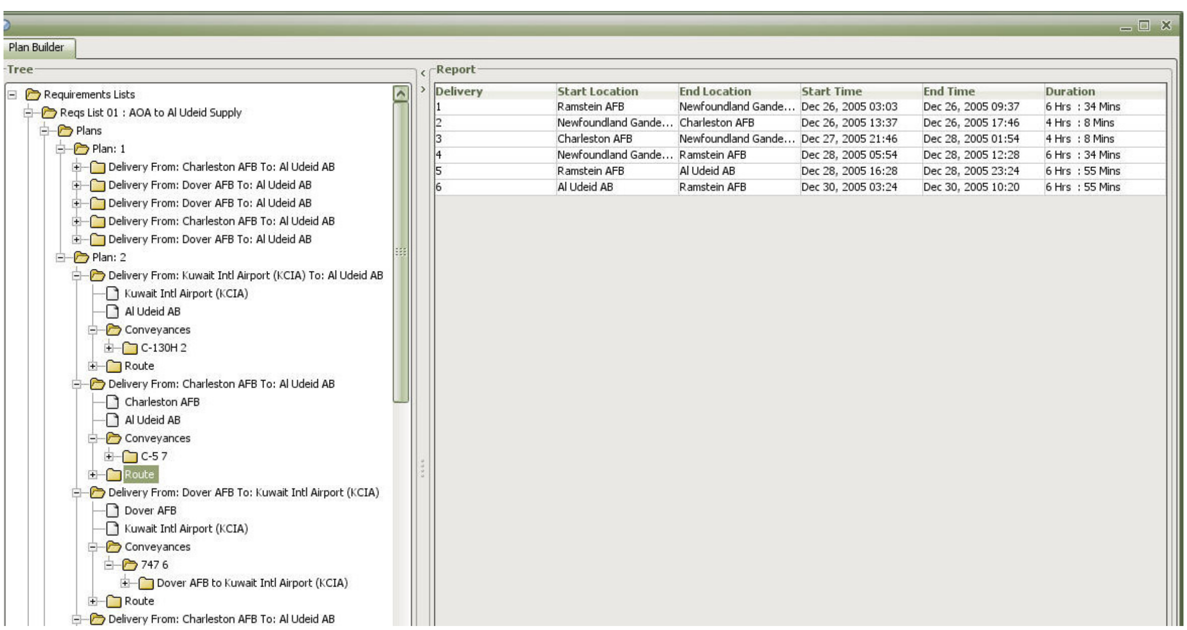


Figure 3.23: Typical drill-down details of the second plan

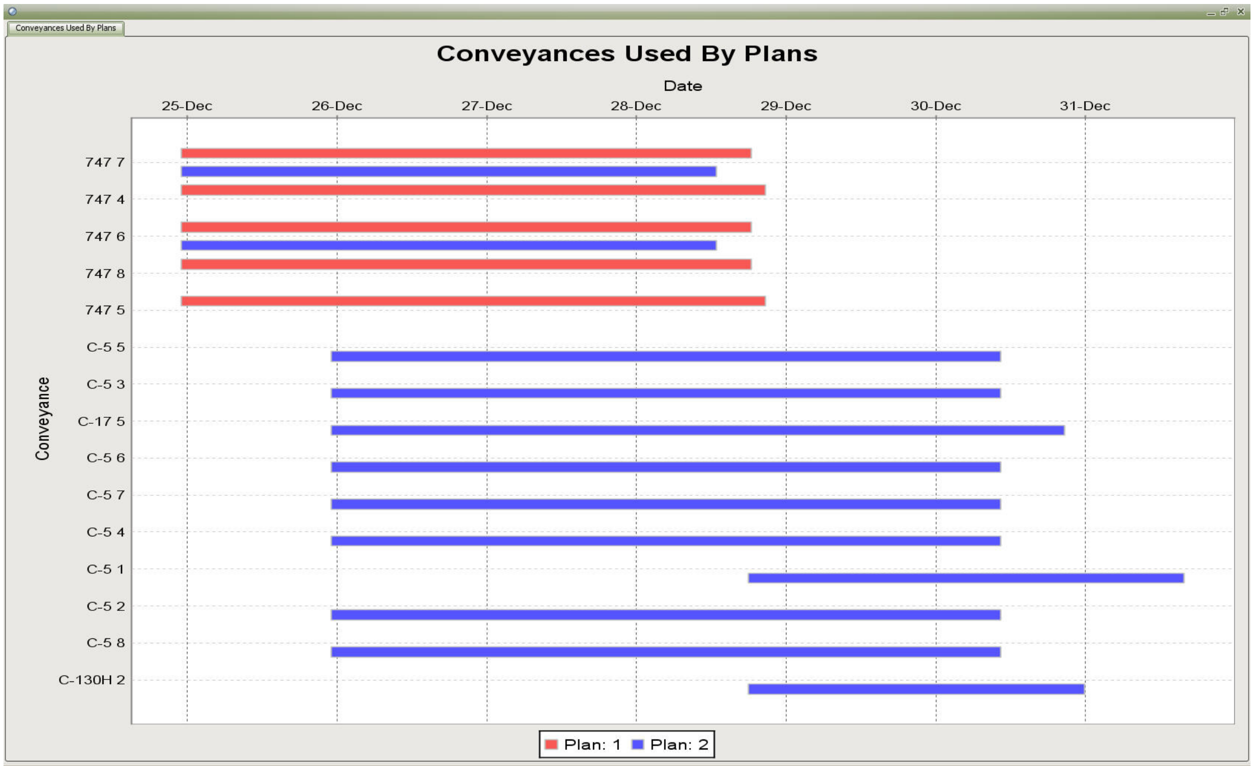


Figure 3.24: Comparison of conveyances needed in support of the first and second plans

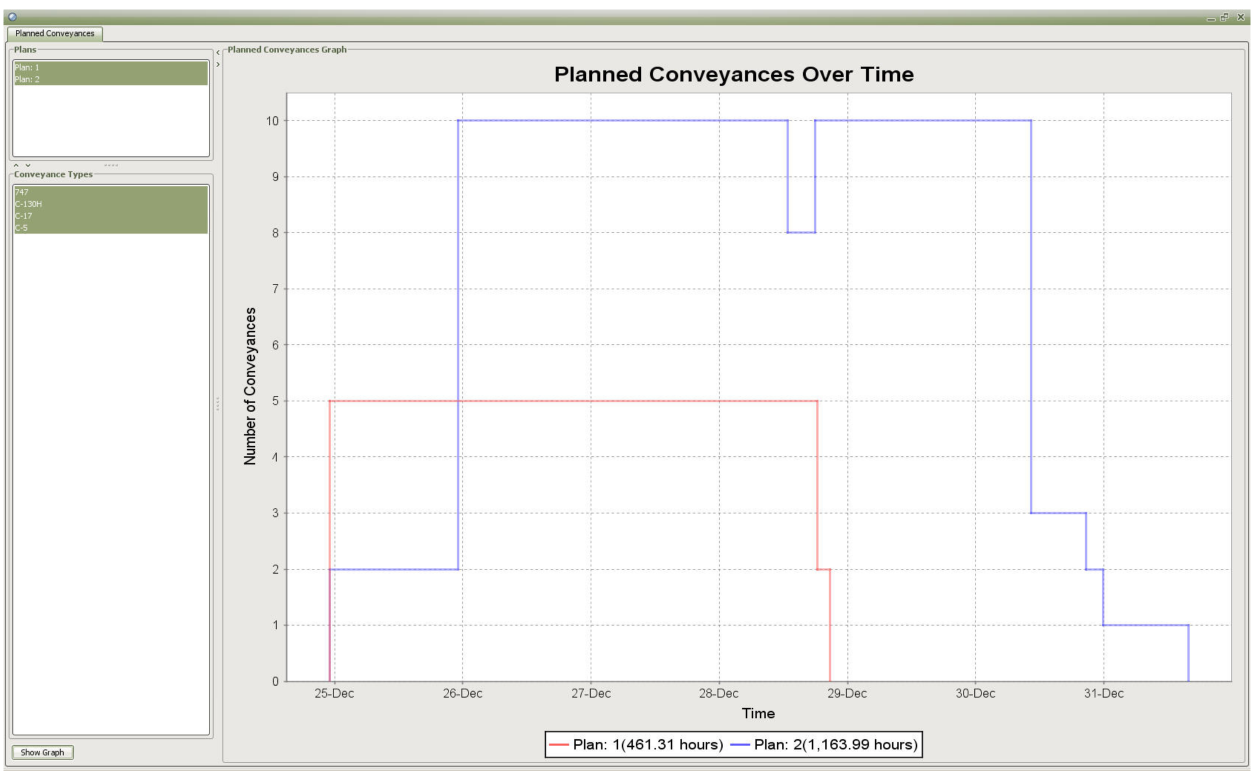


Figure 3.25: Comparison of overall lift requirements for the first and second plans

Apart from the ability of the user to drill down into the details of each delivery plan there are a number of comparative graphical reports available, such as the utilization of specific conveyances by each plan shown in Figure 3.24 and the number of conveyances that are required to support each plan over time shown in Figure 3.25.

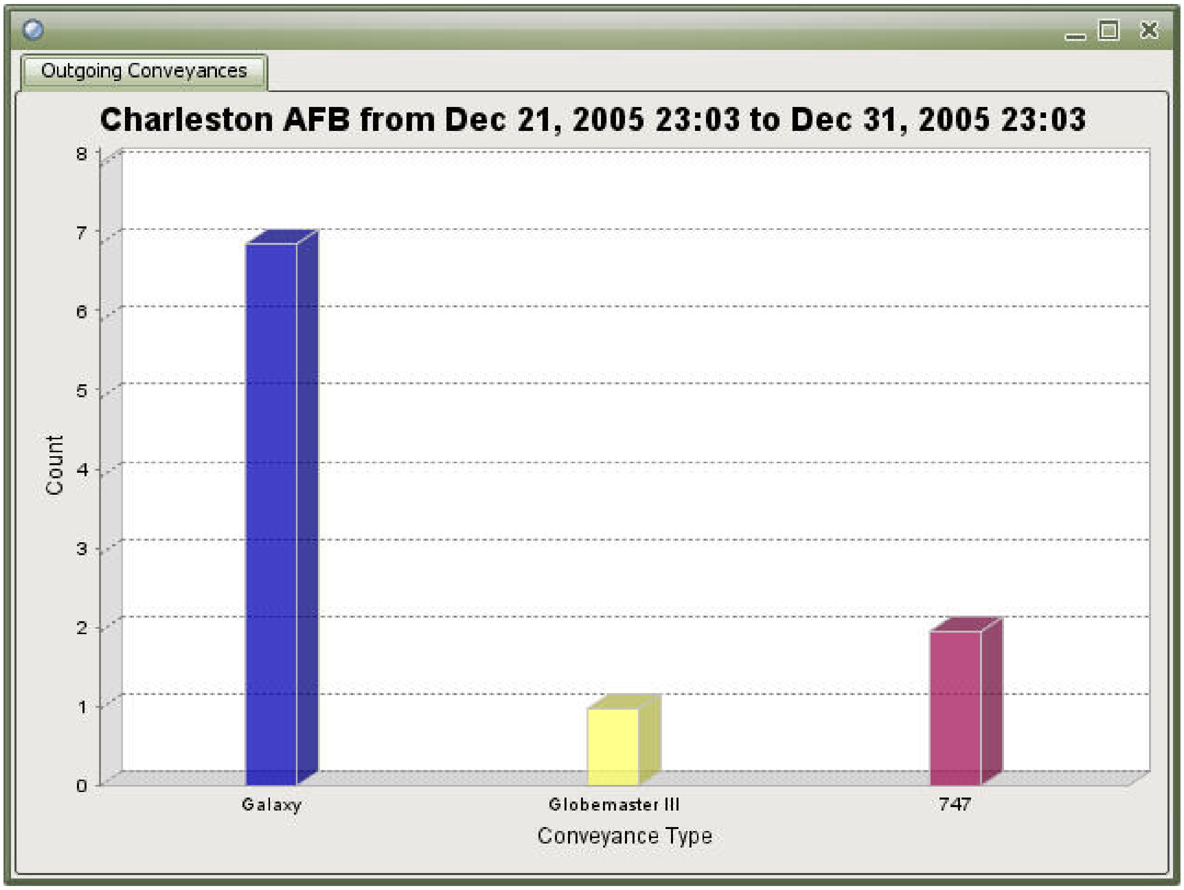


Figure 3.26: Departures from Charleston by conveyance type

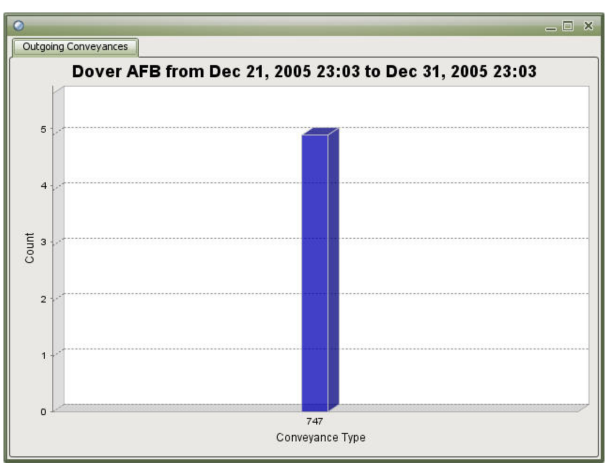


Figure 3.27: Departures from Dover

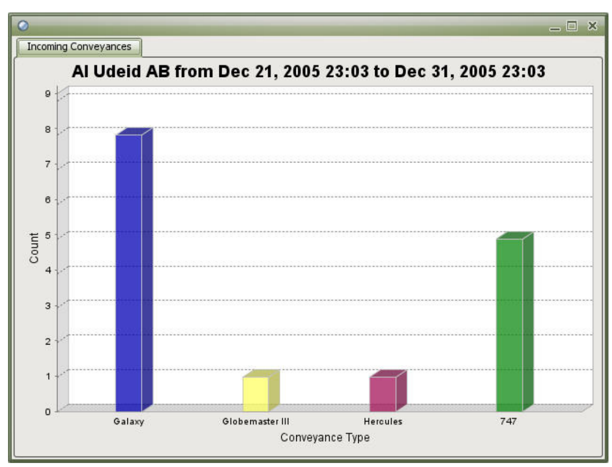


Figure 3.28: Departures from Al Udeid

Figures 3.26 to 3.28 show examples of conveyance departures from the Charleston, Dover and Al Udeid APODs, respectively. Similar reports are available for cargo transfers by date (Figures 3.29 to 3.30) in terms of what was lifted yesterday, the current inventory, and what is planned to be lifted during the next 72 hours. In this way the user is able to determine the expected volume of shipments from any particular APOD on a daily basis. The dates selected for the example bar chart reports shown in Figures 3.29 and 3.30 are December 23 to 26, 2005.

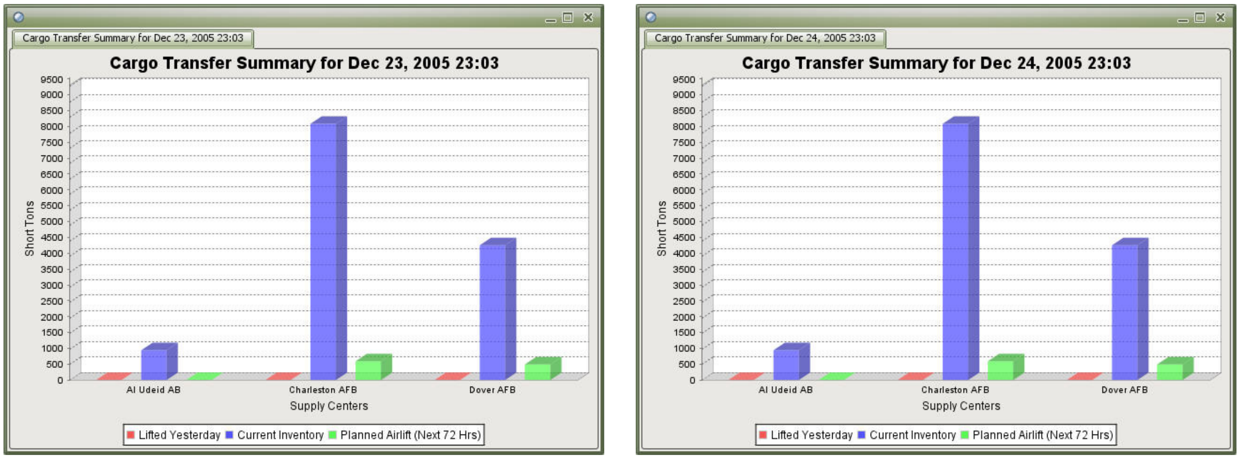


Figure 3.29: Typical cargo transfer history, status, and 72-hour projections

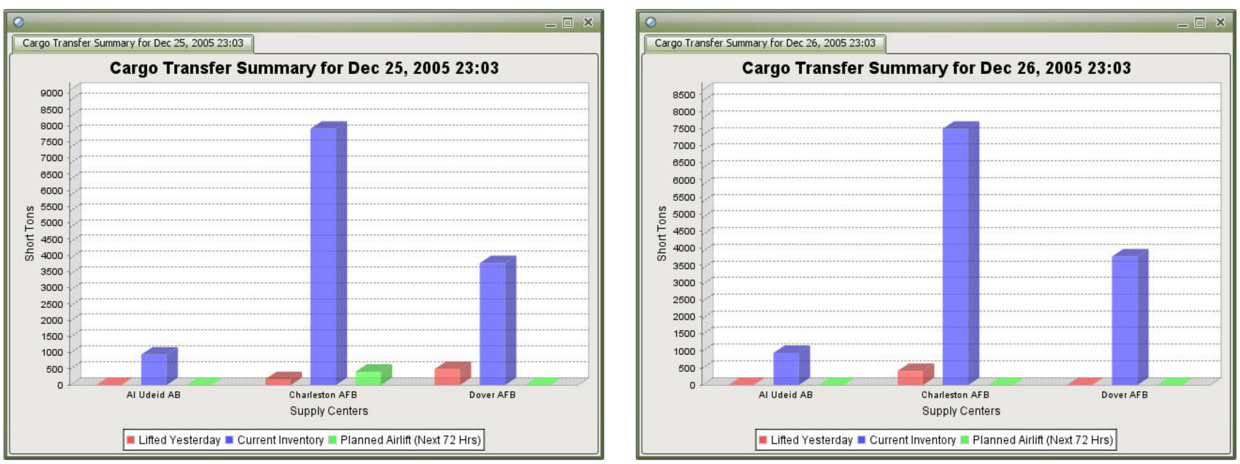


Figure 3.30: Typical cargo transfer history, status, and 72-hour projections

Again, these reports are intended to be examples of the kind of information that can be made available by TRANSWAY. The development team will be guided by feedback from users in future development cycles. The reporting capabilities of the system can be easily extended in any direction within the constraints of data availability.

References

Bauer C. and G. King (2005); 'Hibernate in Action'; Second Edition, Manning Publications, Greenwich, Connecticut.

Crino J. R. (Maj. USA) (2002); 'A Group Theoretic Tabu Search Methodology for Solving the Theater Distribution Vehicle Routing and Scheduling Problem'; Ph.D. thesis, Air Force Office of Scientific Research, Arlington, Virginia.

Fowler M. (2003); 'Patterns of Enterprise Application Architecture'; Addison-Wesley, Reading, Massachusetts.

Fowler M (1997); 'Analysis Patterns: Reusable Object Models'; Addison-Wesley, Reading, Massachusetts.

Fowler M. and K. Scott (1997); 'UML Distilled: Applying the Standard Object Modeling Language'; Addison-Wesley, Reading, Massachusetts.

Friedman-Hill E. (2003); 'Jess in Action'; Manning Publications, Greenwich, Connecticut.

Genesereth M. and N. Nilsson (1987); 'Logical Foundations of Artificial Intelligence'; Morgan Kaufmann, San Mateo, California.

Glover F. (1986); 'Future Paths for Integer Programming and Links to Artificial Intelligence'; Computers and Operations Research, Vol 13 (pp.533-5490.

Glover F. and M. Laguna (1997); 'Tabu Search'; Kluwer, Norwell, Massachusetts.

Hibernate API on-line: http://www.hibernate.org/hib_docs/v3/api.

Hibernate Web site: http://www.hibernate.org.

Karaboga D. and D. Pham (2000); 'Intelligent Optimization Techniques, Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks'; Springer Verlag, Berlin, Germany.

Leighton R., L. Vempati, A. Davis, M. Porczak and J. Pohl (2004); 'The ICDM Development Toolkit : Technical Description' ; Technical Report (CDM-18-04), CDM Technologies, Inc., San Luis Obispo, California.

NASA (1992); 'CLIPS 6.0 Reference Manual'; Software Technologies Branch, Lyndon B. Johnson Space Center, Houston, Texas.

OpenTS (2005); Web site: http://www.coin-or.org/OpenTS/docs/index.html

Pohl J., K. Pohl, R. Leighton, M. Zang, S. Gollery and M. Porczak (2004); 'The ICDM Development Toolkit: Purpose and Overview'; Technical Report (CDM-16-04), CDM Technologies, Inc., San Luis Obispo, California.

Stroustrup B. (1987); 'The C++ Programming Language'; Addison-Wesley, Reading, Massachusetts.

Talbi E. G. (2002); 'A Taxonomy of Hybrid Metaheuristics'; Journal of Heuristics, 8 (5), September (pp.541-564).

USTC (2003); 'Data Management Handbook'; USTRANSCOM (J6), Scott Air Force Base, Illinois.

Wright R. and M. Sweet (2000); 'OpenGL®: Super Bible'; Waite Group Press, Indianapolis, Indiana.