

Data Integrity in Travel Management Reporting

The travel management reports produced by back office systems and third-party reporting tools are often plagued with omissions, errors and inconsistencies. Although the output of a system is generally no better than the information entered into it, **CrossFire** goes to great lengths to improve the quality of the information it receives. "Raw" data from the "source systems" is validated and corrected in order to provide meaningful travel management reports.

CrossFire is known for its ability to improve the quality of the data it receives and stores in its database. The Prism Group, an organization used by several airlines to analyze corporate contract data has developed an approach to rating the quality and completeness of data provided by over 80 sources. The rating system scores data on a scale of 0 – 1060 points. A summary of the top scoring providers as of June 1, 2003 is shown below.

PRISM Xport Certification Summary

System Name	Points	Grade
CrossFire	1,060	A+
SABRE VantagePoint	1,020	A+
TravCom	1,015	A+
World Span WorldLedger	1,015	A+
BTI - U.S. (World Travel Partners)	1,015	A+
Amadeus GlobalMax	1,010	A+

© PRISM Group, Inc. 2001

Note that CrossFire is the only system of the eighty products evaluated that received a perfect score!

What are the Data Integrity issues?

Travel management data typically includes air, hotel and car information that usually is provided by a travel agency back office system. To understand some of the data integrity issues, it is important to understand the shortcomings of the systems that provide data to the travel management systems.

Global Distribution Systems

The primary sources of information for travel management reporting are the Global Distribution systems (GDS). These systems were developed as booking systems for the airlines and little consideration was given to the need for producing management reports. Therefore, there are several problems with the GDS as the provider of quality information.

Data Integrity in Travel Management Reporting

- The primary problem is that there are no standard locations in the Passenger Name Record (PNR) for storing the various pieces of information needed for reporting (such as department number, authorization codes, employee numbers, project codes, low fares, full fares, etc). A number of free-form fields have been provided in the PNR and it is up to the travel agency to determine where the reporting information should be stored.
- The reporting information that is usually the same from booking to booking such as department number or employee number is often stored in a profile so that it can easily be moved into a new PNR. However, the use of profiles is not mandatory so this information may not be included in every booking.
- The reporting information that varies from booking to booking is typically entered into the free-form fields by the reservation agent. Although most travel agencies' practice is to use "scripts" to ensure that all required data is entered, PNR's can be created without using the scripts.
- It is not a standard feature of the GDS to validate that the required reporting information is entered in the PNR. Therefore, it is possible for PNRs to be created without accurate and complete reporting information.

Mid-Office Systems

In an attempt to address the validation shortcomings of the GDS, many travel agencies use "mid-office" quality control" software products to review PNRs that have been created. These systems can be programmed to check PNRs for completeness and accuracy. PNRs with errors can be queued for manual correction. Although these systems can correct some of the data quality problems, they tend not to do a comprehensive job. The major problems with the mid-office systems are:

- They may not have the sophistication or flexibility to identify and validate all required reporting data.
- They often cannot correct the data but must queue the PNR to an agent for manual correction. The agent may not make the correction in a timely manner.
- With an increasing number of fares requiring almost immediate ticketing, the window of opportunity for the mid-office system to detect and correct problems prior to the PNR being sent to the travel agency back office has become very small.

Data Integrity in Travel Management Reporting

Back Office Systems

Most travel agencies use back office systems that receive PNR information from the GDS when the ticket is issued. Back office systems are primarily used by travel agencies for internal accounting and ARC reporting. These systems have inherent problems related to providing accurate and complete travel management reporting information.

- Much like the GDS, many back office systems were developed without regard to the requirements of travel management reporting and the editing and validation required to provide accurate information is not in place. To overcome this problem, many travel agencies prepare “audit reports” to search the data looking for illogical information that must be manually corrected. This proves to be a time-consuming and often incomplete process.
- Many back office systems are old systems that cannot be easily changed to keep up with the changes in the travel industry. For example, the increased usage of airline contracts has resulted in an increase in the number of split tickets (multiple tickets created from a single PNR). In many cases, the back office system is unable to detect when the hotel and car bookings for a PNR that has been split are associated with each of the tickets created. This common situation results in duplicate hotel and car bookings.
- Back office systems have limited reporting capabilities and are usually unable to produce the sophisticated reports required by travel managers.

Third Party Reporting Systems

Most corporate travel agencies and Corporate Travel Departments (CTD) have licensed or developed systems that produce reports either directly from the back office system’s database or from proprietary databases that accept data from the back office. The first approach has limited capability because the back office databases often are not designed to support sophisticated reporting.

The most effective means of providing sophisticated travel management reports is to store the travel data in a separate database designed specifically for travel reporting. Typically, this reporting system is developed and maintained by a third-party vendor or a larger travel agency. However, there are two types of third party systems:

Data Integrity in Travel Management Reporting

o Travel Reporting Systems

These systems accept data from back office systems and store it a proprietary database with little or no change to the information. The inaccuracies of the data in the back office system are brought into the reporting database. Some systems offer the ability to perform “quality” queries after the data is stored in the reporting database that identify and correct various basic data integrity such as missing low fares and full fares. It is often the responsibility of the user to define and develop these queries (usually in a cryptic SQL language). The sophistication and completeness of the “quality query” approach for data integrity improvement is directly related to the technical sophistication of the end-user.

o Travel Management Reporting Systems

This type of reporting system (**CrossFire**) incorporates sophisticated validation and logic checking directly into the process of importing the back office data into the reporting database. The logic built into the system is much more sophisticated and complete than the after-the-fact “quality query” approach used by the other type of systems. **CrossFire** allows the users to define their own logic by filling a simple screen instead of building complicated SQL queries. The advantages of the **CrossFire** approach are significant:

- o The loading of the reporting database is a one-step activity because the validation logic is integrated into the process. For example, **CrossFire** performs over 100 standard validation checks (plus any user-defined validation checks) as the data is processed into the system. Systems in the other category would require a large number of separate “quality queries” to be processed individually after the data was loaded to perform similar results.
- o The logic used to perform the validations can be much more sophisticated that the typical end-user would be able to develop in an SQL language that requires in-depth knowledge of the reporting database and its component parts.
- o **CrossFire** stores the results of the validation process in an Error database that can be used to for tracking the source of the error so that corrective action can be taken. For example, summary reports of the types of errors by reservation agent can be produced so that operations management can determine if there are training issues that can be

Data Integrity in Travel Management Reporting

addressed. This allows errors to be eliminated instead of merely being cleaned up by the "quality queries".

What are the Specific Data Quality Issues?

Travel management reporting usually includes air, hotel and car information that initially was stored in the PNR, was transmitted to the travel agency back office system and finally loaded into the reporting database. Each type of information can suffer from various data integrity problems. These problems can either be the result of omitted or illogical data in the PNR (**Validation problems**) or can be the result of problems inherent to the way that the data is stored and processed in the GDS and back office systems (**System problems**).

Air-related Validation Issues

Due to the number of important reporting items that are stored in free-form fields in the PNR and in back office systems and because the GDS and back office systems do not validate these items, it is highly likely that a number of errors, due to missing or illogical entries, will be found in the reporting data. The most important errors identified and/or corrected by **CrossFire** are:

- Missing or invalid Client codes
- Missing Full Fares (for calculating "savings")
- Missing Low Fares (for calculating "missed savings")
- Full Fare < Fare Paid
- Low Fare > Fare Paid
- Missing or invalid savings codes. (In some cases, savings codes are unique to a single client and are validated accordingly)
- Missing or invalid Branch codes. (If travel agency branch-related reporting is to be done).
- Missing or invalid Agent codes. (If travel agent-related reporting is to be done).
- Missing free-form field information. (Important reporting items such as department, project, and employee number are often stored in free-form fields in the PNR. If these items are required for individual clients, **CrossFire** will verify that an entry is present for each item. It is likely that the entries will vary by client.)
- Valid free-form field information. (**CrossFire** can store client-specific lists of reporting items such as department, project, and employee numbers. During the process of loading the database, errors can be created if an entry is present but is not one of the valid values).
- Missing or invalid Credit card information .
- Missing or invalid Validating Carrier information.

Data Integrity in Travel Management Reporting

- Missing or illogical Contract Fare information.
- Missing or illogical Comparison Fare information.
- Missing or illogical "Special Services" Fare information.
- Illogical use of specific Savings Codes. (For example, if Savings Code 'L' is used when the traveler accepts the lowest available fare, the low fare should always match the fare paid. **CrossFire** allows the user to define specific tests for individual savings code and can take corrective action, if necessary).
- Illogical missed savings. (The **CrossFire** user can establish a limit for the maximum missed savings amount that is acceptable. Errors can be created for invoices that exceed this limit so that the items can be researched to determine if there was a data input error).
- Illogical savings. (The **CrossFire** user can establish a limit for the maximum savings amount that is acceptable. Errors can be created for invoices that exceed this limit so that the items can be researched to determine if there was a data input error).
- E-ticket indicator is missing or invalid.
- Service Fees missing. (Certain clients may require that a Service Fee invoice be created for each ticket issued).
- Flight Number is missing.
- Fare Basis is missing.
- Flight cost is missing.
- Airline code is missing or invalid.
- Flight Mileage is missing.
- Illogical Airport Pair. (In some cases when flight information is manually entered into the back office system, illogical data is entered. For example, an airport pair may be required and an entry such as MSP-MSP is entered just to complete the form. **CrossFire** will identify and report any flights between the same airport or airports in the same city).

Air-related Validation Issues

As previously stated, the systems that provide information to the travel reporting software were not designed to meet travel reporting requirements. In order to produce accurate and complete reports, the source data must be analyzed and corrected as the information is loaded into the reporting database. In most cases, the logic required to correct the data is too complicated to be processed in the "quality queries" processed by the Travel Reporting Systems described above. Only Travel Management Reporting Systems (such as **CrossFire**) can incorporate the logic necessary to correct these significant problems.

Data Integrity in Travel Management Reporting

- o **Incorrect stop over and connection codes**

A major reporting requirement is to produce reports showing the true origin and destination (O & D) of the travel. To accomplish this, the reporting system usually relies on the connection code provided by the back office system to determine if a stop over or connection has taken place. Connecting flights are combined to create the overall O & D. For example, consider the following itinerary:

4/14/2003	ORD – LAX	X
4/14/2003	LAX – HKG	O
4/16/2003	HKG – ORD	O

The passenger has flown from Chicago to Hong Kong with a connection in Los Angeles. To return, the passenger has flown directly from Hong Kong to Chicago. The reporting system should report this travel as two O & D's (Chicago – Hong Kong and Hong Kong – Chicago).

However, our research has shown that in some cases, the connection codes provided by the back office system are incorrect. This situation can happen when airline tickets are manually priced. The result is that the connection codes tend to indicate a stopover for each flight. The problem appears more frequently with international tickets.

If the problem occurred with the itinerary above, the back office data (and the data in most reporting systems) would show the following:

4/14/2003	ORD – LAX	O
4/14/2003	LAX – HKG	O
4/16/2003	HKG – ORD	O

The reporting system would incorrectly show three O & D's (Chicago – Los Angeles, Los Angeles – Hong Kong, and Hong Kong – Chicago)

CrossFire resolves the problem by comparing the arrival date and time of one flight to the departure date and time of the next flight. The length of the time between flights determines whether the flight is a connection or a stop over. In the case of the itinerary above, **CrossFire** would change the connection code in the first flight from "O" to "X" and would correctly report the two O & D's.

Data Integrity in Travel Management Reporting

- o **Incorrect Origin/Destination determination**

Some back office systems, particularly those used by travel agencies outside of the U.S and Canada, do not store connection codes, arrival dates and arrival times. Therefore, all flights appear to be stopovers. Typical Travel Reporting Systems will report O & D's accordingly.

In this situation, **CrossFire** has developed a means of analyzing the itineraries and determining where stopovers are likely to have occurred and will more accurately report the actual O & D's. The differences between **CrossFire** reports and those of other reporting systems can be significant.

For example:

CrossFire loaded travel information that was exported from the database of a competitive product. This product is a Travel Reporting System (that is, it does not significantly enhance the quality of the back office data). Since the back office system that provides the data does not store connection information, the other system reports all flights as stopovers. **CrossFire** converts stopovers to connections when appropriate. Top O & D airport pair reports were produced that highlight the differences in the results:

Zurich Top 25 City Pairs				
System A				Cross Fire
Rank	From	To	Count	Rank
1	ZRH	LHR	472	1
2	ZRH	HAM	287	2
3	ZRH	SOU	222	3
4	ZRH	LGW	87	4
5	ZRH	FRA	80	10
6	ZRH	MUC	77	8
7	ZRH	FCO	74	6
8	ZRH	MAN	66	7
9	ZRH	IAD	63	5
10	ZRH	MXP	54	9
11	ZRH	EVR	46	36
12	ZRH	BHX	34	13
13	ZRH	EDI	34	12
14	ZRH	HEL	30	16
15	ZRH	OSL	32	14
16	ZRH	MAD	29	17
17	ZRH	ZRH	23	23
18	ZRH	LCY	22	18
19	ZRH	AMS	19	73
20	ZRH	BCN	19	20
21	ZRH	GVA	19	20
22	ZRH	CPH	18	44
23	ZRH	STR	18	21
24	ZRH	WAW	18	22
25	ZRH	ORD	17	25

Cross Fire				System A
Rank	From	To	Count	Rank
1	ZRH	LHR	405	1
2	ZRH	HAM	286	2
3	ZRH	SOU	217	3
4	ZRH	LGW	82	4
5	ZRH	IAD	71	9
6	ZRH	FCO	66	7
7	ZRH	MAN	68	8
8	ZRH	MUC	64	6
9	ZRH	MXP	48	10
10	ZRH	FRA	41	5
11	ZRH	PHX	40	*
12	ZRH	EDI	36	13
13	ZRH	BHX	34	12
14	ZRH	OSL	30	15
15	ZRH	IST	25	26
16	ZRH	HEL	24	14
17	ZRH	MAD	24	16
18	ZRH	LCY	22	18
19	ZRH	BCN	20	20
20	ZRH	GVA	18	21
21	ZRH	STR	18	23
22	ZRH	WAW	17	24
23	ZRH	BRU	15	17
24	ZRH	NBO	14	48
25	ZRH	CLE	13	64

The other system (referred to as System A) has some very interesting deficiencies.

Data Integrity in Travel Management Reporting

- The total for the top airport pair (ZRH – LHR) is inflated because 67 of the flights in the System A total were actually connections to other destinations. **This is an overstatement of 15%.**
- The client has major facilities in Zurich and Phoenix. **CrossFire** shows ZRH – PHX as the 11th most frequently flown O & D. **System A does not show this O & D at all!** The reason is that there are no direct flights between Zurich and Phoenix used by the client. The majority of the travel between these two cities connected in London. This contributes to the overstatement of the top O & D by System A
- The System A shows Zurich – Chicago as the 25th most frequently flown O & D. **CrossFire does not show ZRH – ORD as an O & D at all! CrossFire** shows that every flight between Zurich and Chicago was a connection to Dallas, Phoenix or Houston.

In summary, the System A overstates the top O & D's, does not report any travel between the client's two major facilities and erroneously lists Chicago as a major destination.

- **Inaccurate flight cost calculation**

Flight costs received from back office systems are usually base costs (excluding taxes and fees) while the total cost of the airline ticket usually contains all taxes and fees. Therefore, the sum of the flight costs will not match the price of the ticket. The difference usually is the taxes and fees, however, in some cases the entire flight cost is missing. The reporting system should correct this discrepancy.

Because most Travel Reporting Systems do not correct these problems as the data is stored in the database, their reports must make adjustments as they are processed. A common method, used by System A mentioned above, is to:

- calculate an average cost per mile by dividing the total cost for the ticket by the total miles flown.
- multiply the number of miles for each flight by the calculated cost per mile to determine the cost of the flight.

At first glance this appears to be an effective approach, however, there is a major shortcoming. The calculation assumes that all flights booked have the same cost per mile which implies that all flights in a ticket are booked using the same fare basis. This means that in a round-trip itinerary with first class

Data Integrity in Travel Management Reporting

and discounted coach flights, the cost of the out-bound flight will be exactly the same as the cost of the return trip in spite of the fact that the first class flight may have actually cost significantly more than the discounted coach ticket. A test on a sample of 31,000 PNR's with more than one flight shows that only 300 (10%) have only one fare basis.

CrossFire corrects the cost discrepancy using a method that retains the cost differences between flights booked in different fare bases. The following example illustrates the difference in the two approaches:

Ticket Price:		\$1,349.00					
			System A		CrossFire		
Itinerary	Class	Miles	CPM	Cost	Cost	CPM	
MSY	SLC	Y	1,425	\$ 0.3722	\$530.44	714.36	\$ 0.5013
SLC	BIL	Y	387	\$ 0.3722	\$144.06	194.14	\$ 0.5017
BIL	SLC	M	387	\$ 0.3722	\$144.06	94.08	\$ 0.2431
SLC	MSY	M	1,425	\$ 0.3722	\$530.44	346.42	\$ 0.2431
		3,624		\$1,349.00	\$1,349.00		
<p>System A cost is determined by calculating an overall cost per mile by dividing fare paid (\$1,349.00) by total miles (3,624). The cost of each segment is calculated by multiplying the miles of each flight by the average cost per mile. All classes of service in an invoice have the same cost per mile.</p>							
<p>Cross Fire cost is determined by adding to the base cost of each segment a prorated portion of the taxes and surcharges. Cost per mile is calculated by dividing the flight cost by the flight miles. Each class of service in an invoice has a different cost per mile which reflects the relative cost of the flight.</p>							

In summary, the System A method shows that the price of a Y-class flight and of an M-class flight from New Orleans to Billings is \$674.52. **CrossFire** shows that a Y-class flight from New Orleans to Billings is \$908.50. The M-class flight between the same cities is \$440.50.

o Exchanged Ticket Processing

The dramatic increase in the number of exchanges processed presents a challenge to the reporting system. The problem is a matter of timing. Consider the following scenario:

In January, a traveler books a round trip from Chicago to Paris in first class. The ticket cost \$6,000. The reporting database is updated with the following itinerary:

<u>Date</u>	<u>Routing</u>	<u>Cost</u>
01/15/2003	ORD – CDG	\$3,000
01/21/2003	CDG – ORD	\$3,000

Data Integrity in Travel Management Reporting

The traveler's trip is postponed and in February, the original ticket is exchanged for a new ticket. An exchange fee of \$100.00 is charged. The reporting database is updated with the following itinerary:

<u>Date</u>	<u>Routing</u>	<u>Cost</u>
02/15/2003	ORD – CDG	\$50
02/21/2003	CDG – ORD	\$50

At this point, the reporting data base shows 4 flights between Chicago and Paris. The average cost of a first class flight is calculated to be \$1,525. (\$6,100.00 / 4 flights).

Most Travel Reporting Systems do little to correct this problem. **CrossFire** adjusts the cost of the flights in the new ticket by reversing the cost of the original flights and adding it into the cost of the new flights. The systems stays in balance and the average cost of a first class flight is calculated to be \$3,050. (\$6,100.00 / 2 flights).

- o **Duplicate Flights in Split Tickets**

Increases in the number of split tickets also create challenges for reporting systems. When multiple tickets are invoiced from a single PNR, the ticketing agent must be careful to identify to the accounting system which flights belong to each ticket. If not done properly, all flights in the PNR may appear on each of the ticket transactions stored in the accounting system. These duplicate flight records are then transferred to the reporting system. If the reporting system does not identify and eliminate these duplicates, O & D counts will be overstated and average cost per mile will be understated.

CrossFire identifies the duplicate flights in a PNR and eliminates those that it can. In cases where it is not possible to determine which flights to delete, an error record is created and reported to the user.

- o **Incorrect Comparison Fares on refunds**

When tickets are booked, it is common practice to document in the PNR, comparison fares such as the full coach fare, the lowest unrestricted fare and the low fare offered. These fares are stored in the reporting database and used to calculate savings and missed savings.

A problem arises when tickets are refunded because quite frequently the comparison fares that were entered into the original PNR are not recorded in the refund transaction. In many cases, the comparison fares in the refund

Data Integrity in Travel Management Reporting

are set equal to the amount of the refund. The following example illustrates the problem this causes:

A ticket is invoiced with the following fares:

<u>Fare Paid</u>	<u>Low Fare</u>	<u>Full Fare</u>
\$500.00	\$300.00	\$1,000.00

The ticket is later refunded, however, the comparison fares are not stored in the refund record. The following fares are stored in the reporting database:

<u>Fare Paid</u>	<u>Low Fare</u>	<u>Full Fare</u>
(\$500.00)	(\$500.00)	(\$500.00)

Although the two transactions should net out to zero, the following totals will be reported:

<u>Fare Paid</u>	<u>Low Fare</u>	<u>Full Fare</u>
\$0.00	(\$200.00)	\$500.00

CrossFire addresses this problem by determining the percentage of the original ticket that was refunded and changing the comparison fare entries in the refund by applying that percentage to the comparison fares in the original ticket. The **CrossFire** refund transaction would be:

<u>Fare Paid</u>	<u>Low Fare</u>	<u>Full Fare</u>
(\$500.00)	(\$300.00)	(\$1,000.00)

The **CrossFire** totals would be:

<u>Fare Paid</u>	<u>Low Fare</u>	<u>Full Fare</u>
\$0.00	\$0.00	\$0.00

Data Integrity in Travel Management Reporting

- o **Incorrect Refund counts (Gross vs Net)**

A common problem with Travel Reporting Systems is evident in the way in which transactions are counted. Most back office and Travel Reporting Systems count refunds as positive transactions with negative values. The following example demonstrates the problem with this approach:

Ten tickets costing \$1,000 each are issued. Two of the tickets are refunded. Clearly, the net result of these transactions should be 8 transactions totaling \$8,000. A Travel Reporting System may show the following:

	<u>Count</u>	<u>Total</u>	<u>Average Cost</u>
Tickets	10	\$10,000	\$1,000.00
Refunds	<u>2</u>	<u>(\$ 2,000)</u>	<u>(\$1,000.00)</u>
Total	12	\$ 8,000	\$ 666.67

CrossFire shows the proper totals:

	<u>Count</u>	<u>Total</u>	<u>Average Cost</u>
Tickets	10	\$10,000	\$1,000.00
Refunds	<u>(2)</u>	<u>(\$ 2,000)</u>	<u>\$1,000.00</u>
Total	8	\$ 8,000	\$1,000.00

Travel Reporting systems that exhibit this problem tend to mask the issue by producing the majority of their reports with "gross" totals meaning that refunds are excluded from the reports. However, effective travel management reports should show both "gross" and "net" totals.

Hotel-related Validation Issues

The data integrity issues inherent to travel management reports are not limited to air-related information. Common problems with hotel information on travel management reports include:

- o **Duplicate Hotel Bookings**

Frequently, hotel booking information in the back office system is duplicated. "Split ticketing" is a major cause of the problem. When multiple tickets are issued from a PNR that contain hotel bookings, those hotel bookings can be transmitted from the GDS to the back office system with each ticket. This

Data Integrity in Travel Management Reporting

causes the hotel information to be included in the back office system's data base with each ticket. Most Travel Reporting systems do little to correct this problem. With the increased number of split tickets issued, the number of duplications can be significant. The result is overstatement of hotel bookings and cost.

During the process of importing the back office data into its database, **CrossFire** identifies multiple hotel bookings and automatically eliminates the duplicates.

- o **Hotel Refunds**

When airline tickets are refunded, most back office systems **do not** attempt to cancel hotel bookings associated with the trip. Therefore, although the travel management reports will reflect the refunded air travel, back office reports and those of Travel Reporting Systems will not reflect the cancelled hotel bookings. Therefore, the hotel totals on reports from these types of systems will be overstated.

CrossFire, while importing the back office data into the database, will determine if the ticket refunded had hotel bookings. If so, refund transactions for the hotel bookings will be automatically created. This approach improves the integrity of the travel management data and allows **CrossFire** to produce both "gross" and "net" hotel reports. Travel Reporting Systems can only produce "gross" reports which contain the overstated hotel totals.

Car-related Validation Issues

Car-related information on travel management reports has data integrity issues similar to the hotel problems. Common problems with car information on travel management reports include:

- o **Duplicate Car Bookings**

Frequently, car booking information in the back office system is duplicated. "Split ticketing" is a major cause of the problem. When multiple tickets are issued from a PNR that contain hotel bookings, those car bookings can be transmitted from the GDS to the back office system with each ticket. This causes the car information to be included in the back office system's data base with each ticket. Most Travel Reporting systems do little to correct this problem. With the increased number of split tickets issued, the number of

Data Integrity in Travel Management Reporting

duplications can be significant. The result is overstatement of car bookings and cost.

During the process of importing the back office data into its database, **CrossFire** identifies multiple car bookings and automatically eliminates the duplicates.

- **Car Refunds**

When airline tickets are refunded, most back office systems **do not** attempt to cancel car bookings associated with the trip. Therefore, although the travel management reports will reflect the refunded air travel, back office reports and those of Travel Reporting Systems will not reflect the cancelled car bookings. Therefore, the car totals on reports from these types of systems will be overstated.

CrossFire, while importing the back office data into the database, will determine if the ticket refunded had car bookings. If so, refund transactions for the car bookings will be automatically created. This approach improves the integrity of the travel management data and allows **CrossFire** to produce both "gross" and "net" car reports. Travel Reporting Systems can only produce "gross" reports which contain the overstated car totals.

- **Car Rates**

A common problem with car rates is the many reporting systems cannot recognize monthly, weekend and weekly rates. As a result, car reports in many systems can contain illogical totals caused by multiplying a monthly rate by the number of days the car was rented. Although the "data quality" queries performed by some systems can identify and possibly correct these problems, processing these queries is an added task that the user must perform.

CrossFire automatically converts multiple-day rates into daily rates as the back office data is imported into the database.

Data Integrity in Travel Management Reporting

The **CrossFire** Approach to Data Integrity and Validity Checking

CrossFire is unique in its ability to validate travel information and improve the integrity of the data. Basic Travel Reporting Systems import the back office data directly into their databases and require the user to process basic "Quality Queries" to identify some reporting errors. This approach can be ineffective and time-consuming because:

- Processing of "quality queries" is a separate task that must be performed each time that data is loaded into the reporting database. A separate "quality query" may be required for each data integrity issue. **CrossFire** automatically performs over 100 data quality checks as the data is loaded into the **CrossFire** database. No additional processing is required. It follows that over 100 separate "quality queries" may be required to duplicate the checking performed by **CrossFire**
- "Quality Queries" typically require proficiency in Structured Query Language (SQL) programming and an in-depth knowledge of the structure of the reporting systems database. Most travel agencies and Corporate Travel Departments lack the technical expertise to develop the SQL queries necessary. In fact, the sophisticated logic built into the **CrossFire** import process would be difficult for an SQL expert to duplicate. **CrossFire** provides an easy-to-use screen so that non-technical users can add customized validity checks.
- **CrossFire** allows the user not only to verify that the required data is present in the database, but can also determine if the reporting information is valid. For example, **CrossFire** can store lists of valid department numbers for key clients and can verify that the data entered is a valid number. This feature allows **CrossFire** to go beyond the capabilities of other systems. Invalid entries such as '9999' would be identified by **CrossFire** as being invalid while other systems that simply check to see if data is present would not detect the error.
- **CrossFire** creates an error database so that problems that cannot be automatically be corrected can be reviewed and corrected by the user. Error reports can be produced by client or by agent so that recurring data integrity problems can be identified and addressed. Basic Travel Reporting Systems do not offer this feature.