12 Transportation

Voyages occur whenever a person or thing travels from one point to another, perhaps with stops in the middle. Obviously, voyages are a fundamental concept for organizations in the travel industry. Shippers and internal logistical functions also relate to the discussion, as well as package delivery services and car rental companies. Somewhat unexpected, many of this chapter's schemas are also applicable to telecommunications network route analyses; a phone network can be thought of as a map of possible voyages that a call makes between origin and destination phone numbers.

In this chapter we'll draw on an airline case study to explore voyages and routes because many readers are familiar (perhaps too familiar) with the subject matter. The case study lends itself to a discussion of multiple fact tables at different granularities. We'll also elaborate on dimension role playing and additional date and time dimension considerations. As usual, the intended audience for this chapter should not be limited to the industries previously listed.

Chapter 12 discusses the following concepts:

- Bus matrix snippet for an airline
- Fact tables at different levels of granularity
- Combining correlated role-playing dimensions
- Country-specific date dimensions
- Dates and times in multiple time zones
- Recap of localization issues

Airline Case Study and Bus Matrix

We'll begin by exploring a simplified bus matrix, and then dive into the fact tables associated with flight activity.

Figure 12-1 shows a snippet of an airline's bus matrix. This example includes an additional column to capture the degenerate dimension associated with most of the bus process events. Like most organizations, airlines are keenly interested in revenue. In this industry, the sale of a ticket represents unearned revenue; revenue is earned when a passenger takes a flight between origin and destination airports.

	Date	Time	Airport	Passenger	Booking Channel	Class of Service	Fare Basis	Aircraft	Communication Profile	Transaction ID #
Reservations	X	X	X	X	X	X	X	X		Conf #
Issued Tickets	X	X	X	X	X	X	X	X		Conf # Ticket #
Unearned Revenue & Availability	X	X	X			X		X		
Flight Activity	X	X	X	X	X	X	X	X		Conf # Ticket #
Frequent Flyer Account Credits	X		X	X	X	X	X			Conf # Ticket #
Customer Care Interactions	X	X	X	X					X	Case # Ticket #
Frequent Flyer Communications	X	X	X	X					X	
Maintenance Work Orders	X	X	X					X		Work Order #
Crew Scheduling	X	X	X			X		X		

Figure 12-1: Subset of bus matrix row for an airline.

The business and DW/BI team representatives decide the first deliverable should focus on flight activity. The marketing department wants to analyze what flights the company's frequent flyers take, what fare basis they pay, how often they upgrade, how they earn and redeem their frequent flyer miles, whether they respond to special fare promotions, how long their overnight stays are, and what proportion of these frequent flyers have gold, platinum, aluminum, or titanium status. The first project doesn't focus on reservation or ticketing activity data that didn't result in a passenger boarding a plane. The DW/BI team will contend with those other sources of data in subsequent phases.

Multiple Fact Table Granularities

When it comes to the grain as you work through the four-step design process, this case presents multiple potential levels of fact table granularity, each having different associated metrics.

At the most granular level, the airline captures data at the leg level. The leg represents an aircraft taking off at one airport and landing at another without any intermediate stops. Capacity planning and flight scheduling analysts are interested in this discrete level of information because they can look at the number of seats to calculate load factors by leg. Operational aircraft flight metrics are captured at the leg level, such as flight duration and the number of minutes late at departure and arrival. Perhaps there's even a dimension to easily identify on-time arrivals.

The next level of granularity corresponds to a segment. Segments refer to a single flight number (such as Delta flight number 40 or DL0040) flown by a single aircraft. Segments may have one or more legs associated with them; in most cases segments are composed of just one leg with a single take-off and landing. If you take a flight from San Francisco to Minneapolis with a stop in Denver but no aircraft or flight number change, you have flown one segment (SFO-MSP) but two legs (SFO-DEN and DEN-MSP). Conversely, if the flight flew nonstop from San Francisco to Minneapolis, you would have flown one segment as well as one leg. The segment represents the line item on an airline ticket coupon; passenger revenue and mileage credit is determined at the segment level. So although some airline departments focus on leg level operations, the marketing and revenue groups focus on segment-level metrics.

Next, you can analyze flight activity by trip. The trip provides an accurate picture of customer demand. In the prior example, assume the flights from San Francisco to Minneapolis required the flyer to change aircraft in Denver. In this case, the trip from San Francisco to Minneapolis would entail two segments corresponding to the two involved aircraft. In reality, the passenger just asked to go from San Francisco to Minneapolis; the fact that she needs to stop in Denver is merely a necessary evil. For this reason, sales and marketing analysts are also interested in trip level data.

Finally, the airline collects data for the itinerary, which is equivalent to the entire airline ticket or reservation confirmation number.

The DW/BI team and business representatives decide to begin at the segment-level grain. This represents the lowest level of data with meaningful revenue metrics. Alternatively, you could lean on the business for rules to allocate the segment-level metrics down to the leg, perhaps based on the mileage of each leg within the segment. The data warehouse inevitably will tackle the more granular leg level data for the capacity planners and flight schedulers at some future point. The conforming dimensions built during this first iteration will be leveraged at that time.

There will be one row in the fact table for each boarding pass collected from passengers. The dimensionality associated with this data is quite extensive, as illustrated in Figure 12-2. The schema extensively uses the role-playing technique. The multiple date, time, and airport dimensions link to views of a single underlying physical date, time, and airport dimension table, respectively, as we discussed originally in Chapter 6: Order Management.

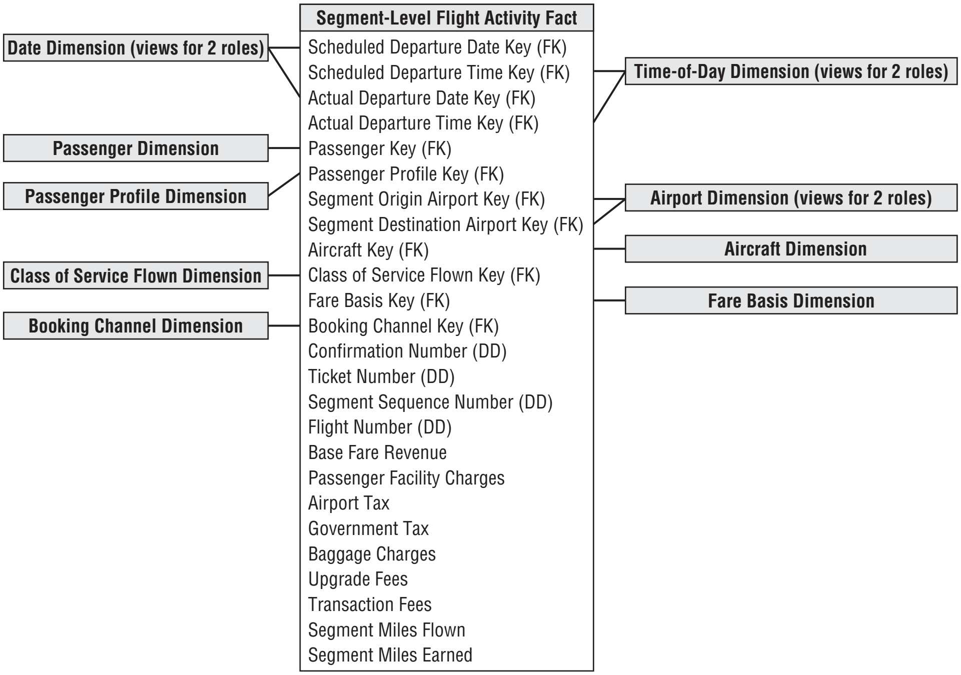

Segment-Level Flight Activity Fact
Scheduled Departure Date Key (FK)
Scheduled Departure Time Key (FK)
Actual Departure Date Key (FK)
Actual Departure Time Key (FK)
Passenger Key (FK)
Passenger Profile Key (FK)
Segment Origin Airport Key (FK)
Segment Destination Airport Key (FK)
Aircraft Key (FK)
Class of Service Flown Key (FK)
Fare Basis Key (FK)
Booking Channel Key (FK)
Confirmation Number (DD)
Ticket Number (DD)
Segment Sequence Number (DD)
Flight Number (DD)
Base Fare Revenue
Passenger Facility Charges
Airport Tax
Government Tax
Baggage Charges
Upgrade Fees
Transaction Fees
Segment Miles Flown
Segment Miles Earned
Date Dimension (views for 2 roles)
Passenger Dimension
Passenger Profile Dimension
Class of Service Flown Dimension
Booking Channel Dimension
Time-of-Day Dimension (views for 2 roles)
Airport Dimension (views for 2 roles)
Aircraft Dimension
Fare Basis Dimension

Figure 12-2: Initial segment flight activity schema.

The passenger dimension is a garden variety customer dimension with rich attributes captured about the most valuable frequent flyers. Interestingly, frequent flyers are motivated to help maintain this dimension accurately because they want to ensure they're receiving appropriate mileage credit. For a large airline, this dimension has tens to hundreds of millions of rows.

Marketing wants to analyze activity by the frequent flyer tier, which can change during the course of a year. In addition, you learned during the requirements process that the users are interested in slicing and dicing based on the flyers' home airports, whether they belong to the airline's airport club at the time of each flight, and their lifetime mileage tier. Given the change tracking requirements, coupled with the size of the passenger dimension, we opt to create a separate passenger profile mini-dimension, as we discussed in Chapter 5: Procurement, with one row for each unique combination of frequent flyer elite tier, home airport, club membership status, and lifetime mileage tier. Sample rows for this mini-dimension are illustrated in Figure 12-3. You considered treating these attributes as slowly changing type 2 attributes, especially because the attributes don't rapidly change. But given the number of passengers, you opt for a type 4 mini-dimension instead. As it turns out, marketing analysts often leverage this mini-dimension for their analysis and reporting without touching the millions of passenger dimension rows.

Passenger Profile Key	Frequent Flyer Tier	Home Airport	Club Membership Status	Lifetime Mileage Tier
1	Basic	ATL	Non-Member	Under 100,000 miles
2	Basic	ATL	Club Member	Under 100,000 miles
3	Basic	BOS	Non-Member	Under 100,000 miles
...	...	...	...	...
789	MidTier	ATL	Non-Member	100,000-499,999 miles
790	MidTier	ATL	Club Member	100,000-499,999 miles
791	MidTier	BOS	Non-Member	100,000-499,999 miles
...	...	...	...	...
2468	WarriorTier	ATL	Club Member	1,000,000-1,999,999 miles
2469	WarriorTier	ATL	Club Member	2,000,000-2,999,999 miles
2470	WarriorTier	BOS	Club Member	1,000,000-1,999,999 miles
...	...	...	...	...

Figure 12-3: Passenger mini-dimension sample rows.

The aircraft dimension contains information about each plane flown. The origin and destination airports associated with each flight are called out separately to simplify the user's view of the data and make access more efficient.

The class of service flown describes whether the passenger sat in economy, premium economy, business, or first class. The fare basis dimension describes the terms surrounding the fare. It would identify whether it's an unrestricted fare, a 21-day advance purchase fare with change and cancellation penalties, or a 10 percent off fare due to a special promotion.

The sales channel dimension identifies how the ticket was purchased, whether through a travel agency, directly from the airline's phone number, city ticket office, or website, or via another internet travel services provider. Although the sales channel relates to the entire ticket, each segment should inherit ticket-level dimensionality. In addition, several operational numbers are associated with the flight activity data, including the itinerary number, ticket number, flight number, and segment sequence number.

The facts captured at the segment level of granularity include the base fare revenue, passenger facility charges, airport and government taxes, other ancillary charges and fees, segment miles flown, and segment miles awarded (in those cases in which a minimum number of miles are awarded regardless of the flight distance).

Linking Segments into Trips

Despite the powerful dimensional framework you just designed, you cannot easily answer one of the most important questions about your frequent flyers, namely, "Where are they going?" The segment grain masks the true nature of the trip. If you fetch all the segments of a trip and sequence them by segment number, it is still nearly impossible to discern the trip start and endpoints. Most complete itineraries start and end at the same airport. If a lengthy stop were used as a criterion for a meaningful trip destination, it would require extensive and tricky processing at the BI reporting layer whenever you try to summarize trips.

The answer is to introduce two more airport role-playing dimensions, trip origin and trip destination, while keeping the grain at the flight segment level. These are determined during data extraction by looking on the ticket for any stop of more than four hours, which is the airline's official definition of a stopover. You need to exercise some caution when summarizing data by trip in this schema. Some of the dimensions, such as fare basis or class of service flown, don't apply at the trip level. On the other hand, it may be useful to see how many trips from San Francisco to Minneapolis included an unrestricted fare on a segment.

In addition to linking segments into trips on the segment flight activity schema, if the business users are constantly looking at information at the trip level, rather than by segment, you might create an aggregate fact table at the trip grain. Some of the earlier dimensions discussed, such as class of service and fare basis, obviously would not be applicable. The facts would include aggregated metrics like trip total base fare or trip total taxes, plus additional facts that would appear only in this complementary trip summary table, such as the number of segments in the trip. However, you would go to the trouble of creating this aggregate table only if there were obvious performance or usability issues when you use the segment-level table as the basis for rolling up the same reports. If a typical trip consists of three segments, you might barely see a three times performance improvement with such an aggregate table, meaning it may not be worth the bother.

Related Fact Tables

As discussed earlier, you would likely create a leg-grained flight activity fact table to satisfy the more operational needs surrounding the departure and arrival of each flight. Metrics at the leg level might include actual and blocked flight durations, departure and arrival delays, and departure and arrival fuel weights.

In addition to the flight activity, there will be fact tables to capture reservations and issued tickets. Given the focus on maximizing revenue, there might be a revenue and availability snapshot for each flight; it could provide snapshots for the final 90 days leading up to a flight departure with cumulative unearned revenue and remaining availability per class of service for each scheduled flight. The snapshot might include a dimension supporting the concept of "days prior to departure" to facilitate the comparison of similar flights at standard milestones, such as 60 days prior to scheduled departure.

Extensions to Other Industries

Using the airline case study to illustrate a voyage schema makes intuitive sense because most people have boarded a plane at one time or another. We'll briefly touch on several other variations on this theme.

Cargo Shipper

The schema for a cargo shipper looks quite similar to the airline schemas just developed. Suppose a transoceanic shipping company transports bulk goods in containers from foreign to domestic ports. The items in the containers are shipped from an original shipper to a final consignor. The trip can have multiple stops at intermediate ports. It is possible the containers may be off-loaded from one ship to another at a port. Likewise, it is possible one or more of the legs may be by truck rather than ship.

As illustrated in Figure 12-4, the grain of the fact table is the container on a specific bill-of-lading number on a particular leg of its trip. The ship mode dimension identifies the type of shipping company and specific vessel. The container dimension describes the size of the container and whether it requires electrical power or refrigeration. The commodity dimension describes the item in the container. Almost anything that can be shipped can be described by harmonized commodity codes, which are a kind of master conformed dimension used by agencies, including U.S. Customs. The consignor, foreign transporter, foreign consolidator, shipper, domestic consolidator, domestic transporter, and consignee are all roles played by a master business entity dimension that contains all the possible business parties associated with a voyage. The bill-of-lading number is a degenerate dimension. We assume the fees and tariffs are applicable to the individual leg of the voyage.

Travel Services

If you work for a travel services company, you can complement the flight activity schema with fact tables to track associated hotel stays and rental car usage. These schemas would share several common dimensions, such as the date and customer. For hotel stays, the grain of the fact table is the entire stay, as illustrated in Figure 12-5. The grain of a similar car rental fact table would be the entire rental episode. Of course, if constructing a fact table for a hotel chain rather than a travel services company, the schema would be much more robust because you'd know far more about the hotel property characteristics, the guest's use of services, and associated detailed charges.

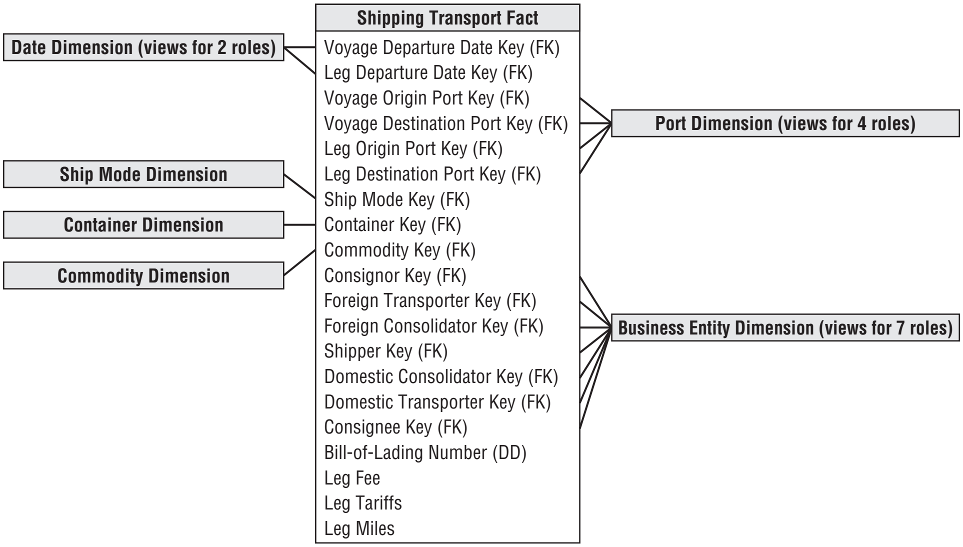

Shipping Transport Fact
Voyage Departure Date Key (FK)
Leg Departure Date Key (FK)
Voyage Origin Port Key (FK)
Voyage Destination Port Key (FK)
Leg Origin Port Key (FK)
Leg Destination Port Key (FK)
Ship Mode Key (FK)
Container Key (FK)
Commodity Key (FK)
Consignor Key (FK)
Foreign Transporter Key (FK)
Foreign Consolidator Key (FK)
Shipper Key (FK)
Domestic Consolidator Key (FK)
Domestic Transporter Key (FK)
Consignee Key (FK)
Bill-of-Lading Number (DD)
Leg Fee
Leg Tariffs
Leg Miles
Date Dimension (views for 2 roles)
Ship Mode Dimension
Container Dimension
Commodity Dimension
Port Dimension (views for 4 roles)
Business Entity Dimension (views for 7 roles)

Figure 12-4: Shipper schema.

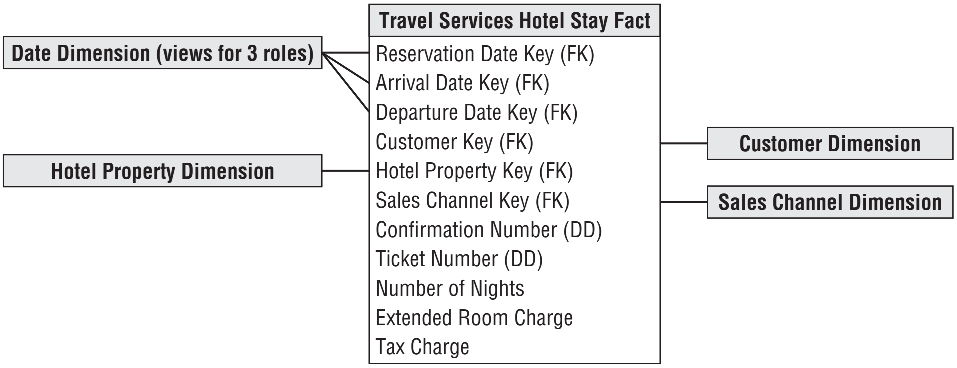


Figure 12-5: Travel services hotel stay schema.

Combining Correlated Dimensions

We stated previously that if a many-to-many relationship exists between two groups of dimension attributes, they should be modeled as separate dimensions with separate foreign keys in the fact table. Sometimes, however, you encounter situations where these dimensions can be combined into a single dimension rather than treating them as two separate dimensions with two separate foreign keys in the fact table.

Class of Service

The Figure 12-2 draft schema includes the class of service flown dimension. Following a design checkpoint with the business community, you learn the users also want to analyze the booking class purchased. In addition, the business users want to easily filter and report on activity based on whether an upgrade or downgrade occurred. Your initial reaction might be to include a second role-playing dimension and foreign key in the fact table to support both the purchased and flown class of service. In addition, you would need a third foreign key for the upgrade indicator; otherwise, the BI application would need to include logic to identify numerous scenarios as upgrades, including economy to premium economy, economy to business, economy to first, premium economy to business, and so on. In this situation, however, there are only four rows in the class dimension table to indicate first, business, premium economy, and economy classes. Likewise, the upgrade indicator dimension also would have just three rows in it, corresponding to upgrade, downgrade, or no class change. Because the row counts are so small, you can elect instead to combine the dimensions into a single class of service dimension, as illustrated in Figure 12-6.

Class of Service Key	Class Purchased	Class Flown	Purchased-Flown Group	Class Change Indicator
1	Economy	Economy	Economy-Economy	No Class Change
2	Economy	Prem Economy	Economy-Prem Economy	Upgrade
3	Economy	Business	Economy-Business	Upgrade
4	Economy	First	Economy-First	Upgrade
5	Prem Economy	Economy	Prem Economy-Economy	Downgrade
6	Prem Economy	Prem Economy	Prem Economy-Prem Economy	No Class Change
7	Prem Economy	Business	Prem Economy-Business	Upgrade
8	Prem Economy	First	Prem Economy-First	Upgrade
9	Business	Economy	Business-Economy	Downgrade
10	Business	Prem Economy	Business-Prem Economy	Downgrade
11	Business	Business	Business-Business	No Class Change
12	Business	First	Business-First	Upgrade
13	First	Economy	First-Economy	Downgrade
14	First	Prem Economy	First-Prem Economy	Downgrade
15	First	Business	First-Business	Downgrade
16	First	First	First-First	No Class Change

Figure 12-6: Combined class dimension sample rows.

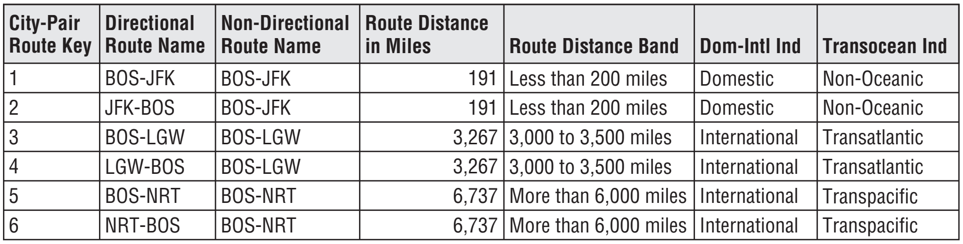
The Cartesian product of the separate class dimensions results in a 16-row dimension table (4 class purchased rows times 4 class flown rows). You also have the opportunity in this combined dimension to describe the relationship between the purchased and flown classes, such as a class change indicator. Think of this combined class of service dimension as a type of junk dimension, introduced in Chapter 6. In this case study, the attributes are tightly correlated. Other airline fact tables, such as inventory availability or ticket purchases, would invariably reference a conformed class dimension table with just four rows.

NOTE In most cases, role-playing dimensions should be treated as separate logical dimensions created via views on a single physical table. In isolated situations, it may make sense to combine the separate dimensions into a single dimension, notably when the data volumes are extremely small or there is a need for additional attributes that depend on the combined underlying roles for context and meaning.

Origin and Destination

Likewise, consider the pros and cons of combining the origin and destination airport dimensions. In this situation the data volumes are more significant, so separate role-playing origin and destination dimensions seem more practical. However, the business users may need additional attributes that depend on the combination of origin and destination. In addition to accessing the characteristics of each airport, business users also want to analyze flight activity data by the distance between the city-pair airports, as well as the type of city pair (such as domestic or trans-Atlantic). Even the seemingly simple question regarding the total activity between San Francisco (SFO) and Denver (DEN), regardless of whether the flights originated in SFO or DEN, presents some challenges with separate origin and destination dimensions. SQL experts could surely answer the question programmatically with separate airport dimensions, but what about the less empowered? Even if experts can derive the correct answer, there's no standard label for the nondirectional city-pair route. Some reporting applications may label it SFO-DEN, whereas others might opt for DEN-SFO, San Fran-Denver, Den-SF, and so on. Rather than embedding inconsistent labels in BI reporting application code, the attribute values should be stored in a dimension table, so common standardized labels can be used throughout the organization. It would be a shame to go to the bother of creating a data warehouse and then allowing application code to implement inconsistent reporting labels. The business sponsors of the DW/BI system won't tolerate that for long.

To satisfy the need to access additional city-pair route attributes, you have two options. One is merely to add another dimension to the fact table for the city-pair route descriptors, including the directional route name, nondirectional route name, type, and distance, as shown in Figure 12-7. The other alternative is to combine the origin and destination airport attributes, plus the supplemental city-pair route attributes, into a single dimension. Theoretically, the combined dimension could have as many rows as the Cartesian product of all the origin and destination airports. Fortunately, in real life the number of rows is much smaller than this theoretical limit because airlines don't operate flights between every airport where they have a presence. However, with a couple dozen attributes about the origin airport, plus a couple dozen identical attributes about the destination airport, along with attributes about the route, you would probably be more tempted to treat them as separate dimensions.



City-Pair Route Key	Directional Route Name	Non-Directional Route Name	Route Distance in Miles	Route Distance Band	Dom-Intl Ind	Transocean Ind
1	BOS-JFK	BOS-JFK	191	Less than 200 miles	Domestic	Non-Oceanic
2	JFK-BOS	BOS-JFK	191	Less than 200 miles	Domestic	Non-Oceanic
3	BOS-LGW	BOS-LGW	3,267	3,000 to 3,500 miles	International	Transatlantic
4	LGW-BOS	BOS-LGW	3,267	3,000 to 3,500 miles	International	Transatlantic
5	BOS-NRT	BOS-NRT	6,737	More than 6,000 miles	International	Transpacific
6	NRT-BOS	BOS-NRT	6,737	More than 6,000 miles	International	Transpacific

Figure 12-7: City-pair route dimension sample rows.

Sometimes designers suggest using a bridge table containing the origin and destination airport keys to capture the route information. Although the origin and destination represent a many-to-many relationship, in this case, you can cleanly represent the relationship within the existing fact table rather than using a bridge.

More Date and Time Considerations

From the earliest chapters in this book we've discussed the importance of having a verbose date dimension, whether at the individual day, week, or month granularity, that contains descriptive attributes about the date and private labels for fiscal periods and work holidays. In this final section, we'll introduce several additional considerations for dealing with date and time dimensions.

Country-Specific Calendars as Outriggers

If the DW/BI system serves multinational needs, you must generalize the standard date dimension to handle multinational calendars in an open-ended number of countries. The primary date dimension contains generic calendar attributes about the date, regardless of the country. If your multinational business spans Gregorian, Hebrew, Islamic, and Chinese calendars, you would include four sets of days, months, and years in this primary dimension.

Country-specific date dimensions supplement the primary date table. The key to the supplemental dimension is the primary date key, along with the country code. The table would include country-specific date attributes, such as holiday or season names, as illustrated in Figure 12-8. This approach is similar to the handling of multiple fiscal accounting calendars, as described in Chapter 7: Accounting.

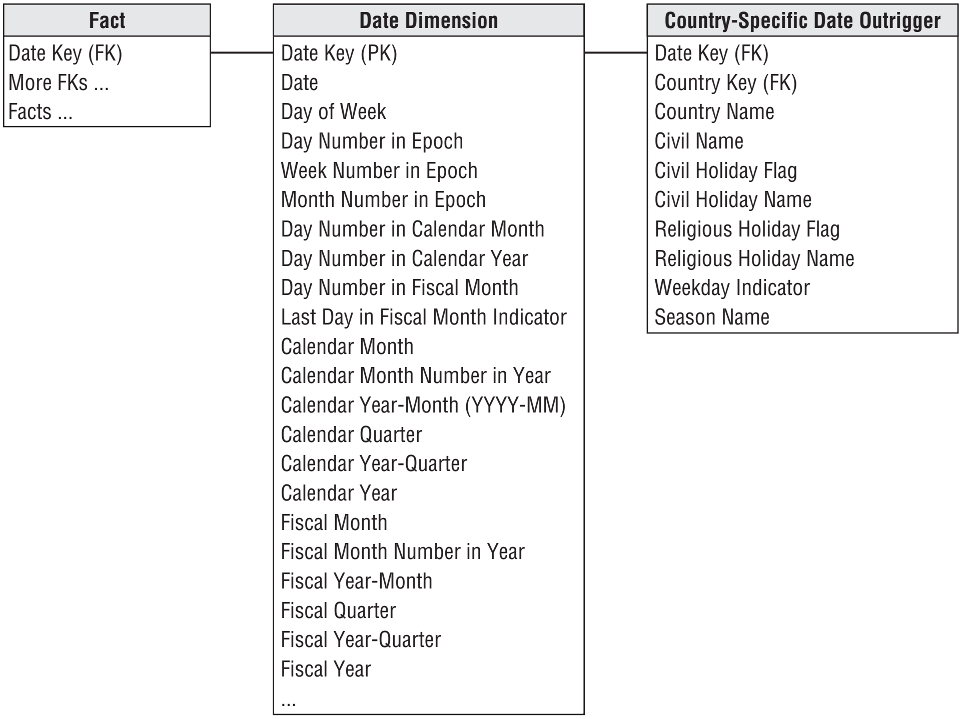


Figure 12-8: Country-specific calendar outrigger.

You can join this table to the main calendar dimension as an outrigger or directly to the fact table. If you provide an interface that requires the user to specify a country name, then the attributes of the country-specific supplement can be viewed as logically appended to the primary date table, allowing them to view the calendar through the eyes of a single country at a time. Country-specific calendars can be messy to build in their own right; things get even more complicated if you need to deal with local holidays that occur on different days in different parts of a country.

Date and Time in Multiple Time Zones

When operating in multiple countries or even just multiple time zones, you're faced with a quandary concerning transaction dates and times. Do you capture the date and time relative to local midnight in each time zone, or do you express the time period relative to a standard, such as the corporate headquarters date/time, Greenwich Mean Time (GMT), or Coordinated Universal Time (UTC), also known as Zulu time in the aviation world? To fully satisfy users' requirements, the correct answer is probably both. The standard time enables you to see the simultaneous nature of transactions across the business, whereas the local time enables you to understand transaction timing relative to the time of day.

Contrary to popular belief, there are more than 24 time zones (corresponding to the 24 hours of the day) in the world. For example, there is a single time zone in China despite its latitudinal span. Likewise, there is a single time zone in India, offset from UTC by 5.5 hours. In Australia, there are three time zones with its Central time zone offset by one-half hour. Meanwhile, Nepal and some other nations use one-quarter hour offset. The situation gets even more unpleasant when you account for switches to and from daylight saving time.

Given the complexities, it's unreasonable to think that merely providing a UTC offset in a fact table can support equivalized dates and times. Likewise, the offset can't reside in a time or airport dimension table because the offset depends on both location and date. The recommended approach for expressing dates and times in multiple time zones is to include separate date and time-of-day dimensions corresponding to the local and equivalized dates, as shown in Figure 12-9. The time-of-day dimensions, as discussed in Chapter 3: Retail Sales, support time period groupings such as shift numbers or rush period time block designations.

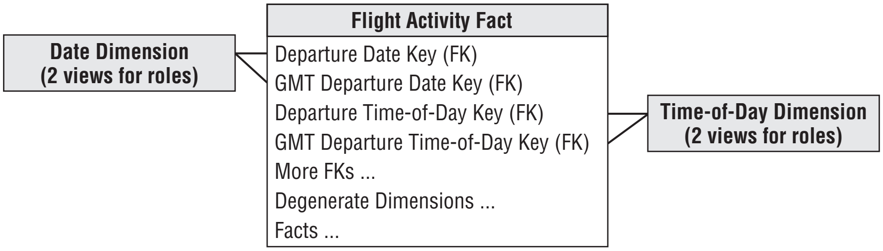


Figure 12-9: Local and equivalized date/time across time zones.

Localization Recap

We have discussed the challenges of international DW/BI system in several chapters of the book. In addition to the international time zones and calendars discussed in the previous two sections, we have also talked about multi-currency reporting in Chapter 6 and multi-language support in Chapter 8: Customer Relationship Management.

All these database-centric techniques fall under the general theme of *localization*. Localization in the larger sense also includes the translation of user interface text embedded in BI tools. BI tool vendors implement this form of localization with text databases containing all the text prompts and labels needed by the tool, which can then be configured for each local environment. Of course, this can become quite complicated because text translated from English to most European languages results in text strings that are longer than their English equivalents, which may force a redesign of the BI application. Also, Arabic text reads from right to left, and many Asian languages are completely different.

A serious international DW/BI system built to serve business users in many countries needs to be thoughtfully designed to account for a selected set of these localization issues. But perhaps it is worth thinking about how airport control towers and airplane pilots around the world deal with language incompatibilities when communicating critical messages about flight directions and altitudes. They all use one language (English) and unit of measure (feet).

Summary

In this chapter we turned our attention to airline trips or routes; we briefly touched on similar scenarios drawn from the shipping and travel services industries. We examined the situation in which we have multiple fact tables at multiple granularities with multiple grain-specific facts. We also discussed the possibility of combining dimensions into a single dimension table for cases in which the row count volumes are extremely small or when there are additional attributes that depend on the combined dimensions. Again, combining correlated dimensions should be viewed as the exception rather than the rule.

We wrapped up this chapter by discussing several date and time dimension techniques, including country-specific calendar outriggers and the handling of absolute and relative dates and times.