

# SECCIÓN PERSONA 1 - Introducción y Contexto (5 minutos)

---

## SLIDE 1: PORTADA

[Mostrar título principal con ícono de avión]

Hoy vamos a presentar el Capítulo 12 del libro de Kimball sobre Data Warehouse, que trata sobre Transporte. Somos un grupo de 5 personas y cada uno cubrirá una sección específica del capítulo."

---

## SLIDE 2: ¿QUÉ SON LOS VIAJES EN DW/BI?

"Empecemos con lo básico: ¿Qué entendemos por 'viajes' en el contexto de Data Warehousing?

**Concepto fundamental:** Un viaje ocurre cuando una persona o cosa se traslada de un punto a otro, posiblemente con paradas en el medio.

Aunque suena simple, este concepto es absolutamente fundamental para muchas industrias:

- **Aerolíneas:** Transportan pasajeros entre aeropuertos
- **Logística y Carga:** Mueven paquetes y mercancías
- **Telecomunicaciones:** Las llamadas 'viajan' a través de redes telefónicas entre números de origen y destino
- **Transporte marítimo:** Barcos que llevan contenedores entre puertos

Lo interesante aquí es que **los principios de diseño dimensional que vamos a ver son aplicables a todas estas industrias**. No se limitan solo a aerolíneas."

---

## SLIDE 3: ¿POR QUÉ UNA AEROLÍNEA COMO CASO DE ESTUDIO?

"Kimball eligió una aerolínea como caso de estudio por varias razones estratégicas:

**Primera razón - Familiaridad Universal:** Todos hemos volado o al menos conocemos cómo funciona una aerolínea. Esto hace que los conceptos sean más fáciles de entender y relacionar.

**Segunda razón - Complejidad Real:** Las aerolíneas tienen desafíos de diseño dimensional muy reales y complejos. No es un ejemplo simplificado; es un caso de negocio real con problemas reales.

**Tercera razón - Múltiples Granularidades:** Como veremos más adelante, en aerolíneas existen diferentes niveles de detalle para los mismos datos. Algunos usuarios necesitan ver cada despegue y aterrizaje individual, otros necesitan ver el vuelo completo, y otros el viaje completo del pasajero.

**Cuarta razón - Alcance Global:** Las aerolíneas operan en múltiples países y zonas horarias, lo que nos permite explorar temas avanzados como localización, múltiples calendarios y manejo de fechas/horas internacionales."

---

## SLIDE 4: MATRIZ DE BUS DE LA AEROLÍNEA

"Ahora, hablemos de una herramienta fundamental en el diseño de Kimball: la Matriz de Bus.

**¿Qué es una Matriz de Bus?** Es una herramienta de planificación que muestra:

- En las **filas**: Los diferentes procesos de negocio (como Reservas, Boletos Emitidos, Actividad de Vuelos)
- En las **columnas**: Las dimensiones conformadas que se comparten entre estos procesos

**¿Por qué es importante?** Porque nos permite visualizar cuáles dimensiones son compartidas. Por ejemplo, la dimensión de Fecha se usa en TODOS los procesos. La dimensión de Pasajero se usa en la mayoría. Esto asegura que cuando construimos el data warehouse, estas dimensiones sean **conformadas** - es decir, significa lo mismo en todos los procesos.

### [Mostrar tabla de matriz de bus]

En esta matriz vemos procesos como:

- Reservas
- Boletos Emitidos
- **Actividad de Vuelos** ← Este es nuestro foco inicial
- Ingresos No Devengados
- Créditos de Viajero Frecuente
- Atención al Cliente

El equipo de negocio y el equipo de DW/BI decidieron que el **primer entregable** debe enfocarse en **Actividad de Vuelos**. ¿Por qué? Porque Marketing tiene preguntas específicas que necesita responder sobre este proceso."

---

## SLIDE 5: CONCEPTOS CLAVE DEL NEGOCIO

"Antes de entrar en el diseño técnico, necesitamos entender algunos conceptos clave del negocio de aerolíneas:

### Concepto 1: Ingresos Devengados vs No Devengados

Esto es fundamental en contabilidad de aerolíneas:

- Cuando alguien **compra un boleto**, ese dinero aún NO es ingreso devengado para la aerolínea. Es una obligación futura.
- Los ingresos se **devengan** (se reconocen contablemente) cuando el pasajero **realmente toma el vuelo** entre el origen y destino.

### Concepto 2: Viajeros Frecuentes

Son los clientes más valiosos de la aerolínea:

- Ganan **millas** por volar
- Tienen **niveles de estatus**: Básico, Nivel Medio, Élite, etc.
- Pueden tener **membresías de club** aeroportuario
- La aerolínea rastrea su **millaje acumulado de por vida**

Lo interesante es que estos viajeros están **motivados a mantener sus datos actualizados** porque quieren asegurarse de recibir el crédito de millaje apropiado.

### Concepto 3: Estructura de Tarifas

Cuando compras un boleto, el precio incluye:

- **Tarifa base**
- **Cargos por instalaciones**
- **Impuestos aeroportuarios y gubernamentales**
- **Tarifas auxiliares** (equipaje, asientos, etc.)

Las preguntas que Marketing quiere responder:

- ¿Qué vuelos toman los viajeros frecuentes?
- ¿Con qué base de tarifas pagan?
- ¿Con qué frecuencia viajan?
- ¿Cómo ganan y redimen sus millas?
- ¿Responden a promociones de tarifas especiales?
- ¿Cuánto duran sus estadías nocturnas?

- ¿Qué proporción tiene estatus oro, platino, etc.?"
- 

## **SLIDE 6: KEY TAKEAWAY - CONCEPTO CLAVE**

"Para cerrar mi sección, quiero enfatizar el concepto clave que deben recordar:

**El diseño dimensional es flexible y aplicable a múltiples industrias relacionadas con transporte y movimiento de recursos.**

Los principios que vamos a ver hoy no se limitan a aerolíneas. Si trabajan en:

- Transporte de carga
- Logística
- Servicios de viajes
- Telecomunicaciones
- Cadena de suministro
- Cualquier industria donde algo o alguien se mueve de un punto A a un punto B

...entonces estos principios les serán útiles.

---

**FIN DE SECCIÓN PERSONA 1**

**Tiempo estimado: 5 minutos**

# SECCIÓN PERSONA 2 - Múltiples Niveles de Granularidad (6 minutos)

---

## SLIDE 7: INTRODUCCIÓN A LA GRANULARIDAD

" Ahora voy a hablarles sobre uno de los conceptos más importantes en el diseño de data warehouses: la **granularidad**.

### ¿Qué es la granularidad?

La granularidad define el **nivel de detalle** de cada fila en la tabla de hechos. Es la respuesta a una pregunta muy simple pero crucial: *'¿Qué representa exactamente UNA fila en mi tabla de hechos?'*

Por ejemplo:

- ¿Una fila representa un vuelo completo?
- ¿O representa cada despegue y aterrizaje individual?
- ¿O representa el viaje completo de un pasajero?

**Principio fundamental de Kimball:** 'Diferentes usuarios del negocio necesitan ver los datos a diferentes niveles de detalle.'

**El problema común:** Muchas organizaciones intentan crear UNA ÚNICA tabla que sirva para todos los usuarios y todos los propósitos. Esto rara vez funciona bien.

**La solución de Kimball:** Crear **múltiples tablas de hechos** a diferentes niveles de granularidad, cada una optimizada para un conjunto específico de usuarios y preguntas de negocio.

En el caso de aerolíneas, identificamos **4 niveles diferentes de granularidad**. Vamos a verlos uno por uno."

---

## SLIDE 8: NIVEL 1 - TRAMO (El más granular)

"El primer nivel es el **TRAMO**, en inglés 'Leg'. Este es el nivel **más atómico, más detallado** de los datos.

**Definición de Tramo:** Un tramo representa un avión **despegando en un aeropuerto y aterrizando en otro aeropuerto** sin paradas intermedias.

**Ejemplo simple:** Un vuelo de San Francisco a Denver es UN tramo.

Si lo visualizamos: **SFO** ✈️ **DEN** - eso es un tramo completo.

### ¿Quién necesita datos a nivel de tramo?

Hay tres grupos principales de usuarios:

1. **Planificadores de capacidad:** Necesitan saber cuántos asientos hay disponibles en cada tramo para calcular el 'factor de carga' - es decir, qué porcentaje de los asientos están ocupados. Si un avión tiene 180 asientos y vuelan 150 pasajeros, el factor de carga es 83%.
2. **Programadores de vuelos:** Necesitan optimizar rutas y horarios basándose en datos de cada despegue y aterrizaje individual.
3. **Operaciones:** Monitorean el rendimiento operacional de cada tramo.

### ¿Qué métricas se capturan a nivel de tramo?

Métricas muy operacionales:

- **Duración del vuelo:** Tiempo real en el aire
- **Minutos de retraso en la salida:** Si despegó tarde
- **Minutos de retraso en la llegada:** Si aterrizó tarde
- **Número de asientos en el avión**
- **Peso de combustible al despegar y al aterrizar**
- **Factor de carga:** Porcentaje de ocupación

Estas son métricas que importan para operaciones, no necesariamente para marketing o finanzas."

---

## SLIDE 9: NIVEL 2 - SEGMENTO (El nivel elegido)

"El segundo nivel es el **SEGMENTO**, y este es el nivel que el equipo eligió para empezar. Les voy a explicar por qué.

**Definición de Segmento:** Un segmento se refiere a un **número de vuelo único** operado por una sola aeronave. Los segmentos pueden tener uno o más tramos asociados.

**Ejemplo 1: Un segmento con UN tramo** Si tomas un vuelo directo de San Francisco a Minneapolis:

- **SFO** ✈️ **MSP**
- Esto es: 1 segmento, 1 tramo
- Vuelo directo, sin escalas

**Ejemplo 2: Un segmento con DOS tramos** Si tomas el vuelo DL0040 (Delta vuelo 40) de San Francisco a Minneapolis, pero el avión hace una parada en Denver:

- **SFO** ✈️ **DEN** ✈️ **MSP**

- Mismo número de vuelo (DL0040)
- Misma aeronave (no cambias de avión)
- Pero son DOS tramos dentro de UN segmento

La clave aquí es: **NO cambias de aeronave, NO cambias de número de vuelo**. Es el mismo vuelo operacional.

### ¿Por qué el equipo eligió este nivel?

Hay tres razones fundamentales:

**Razón 1: Es el nivel más bajo con métricas de ingresos significativas** A diferencia del tramo, que es muy operacional, en el segmento se determinan:

- Los **ingresos de los pasajeros**
- El **crédito de millaje** que gana el viajero frecuente
- Los **impuestos y cargos** aplicables

**Razón 2: Balance perfecto** El segmento ofrece suficiente detalle para análisis profundo, pero sin la complejidad excesiva de trabajar a nivel de tramo.

**Razón 3: Es donde trabajan Marketing e Ingresos** Los grupos de marketing e ingresos (revenue management) trabajan naturalmente a este nivel. Ellos piensan en términos de 'el vuelo DL0040', no en términos de cada despegue y aterrizaje individual.

**La granularidad específica de la tabla de hechos:** 'Una fila en la tabla de hechos por cada **pase de abordaje** recopilado de los pasajeros.'

Esto significa que si 200 pasajeros abordan el vuelo DL0040, habrá 200 filas en la tabla de hechos, una por cada pasajero en ese segmento."

---

## SLIDE 10: NIVEL 3 - VIAJE y NIVEL 4 - ITINERARIO

"Ahora veamos los dos niveles más agregados: el Viaje y el Itinerario.

### NIVEL 3: EL VIAJE

El viaje proporciona una imagen precisa de **la demanda real del cliente**.

**¿Qué es un viaje?** Es lo que el pasajero **realmente quería** hacer, independientemente de las paradas técnicas o escalas que la aerolínea necesite hacer.

**Ejemplo concreto:** Imaginen que un pasajero quiere ir de San Francisco a Minneapolis. Pero la aerolínea necesita hacer una escala en Denver para reabastecer combustible. El pasajero no pidió ir a Denver; la aerolínea lo necesita por razones operacionales.

Entonces:

- **Viaje del pasajero:** SFO → MSP (lo que él quería)
- **Realidad operacional:** Podría involucrar 2 segmentos en 2 aeronaves diferentes

Los segmentos esconden la verdadera naturaleza del viaje. Por eso los **analistas de ventas y marketing** están muy interesados en datos a nivel de viaje - porque refleja la demanda real origen-destino.

**Métrica clave del viaje:** Demanda verdadera entre pares de ciudades.

#### NIVEL 4: EL ITINERARIO

El itinerario es el nivel más agregado.

**¿Qué es un itinerario?** Es equivalente al **número completo del boleto** de la aerolínea o la confirmación de reservación.

**Ejemplo:** Un pasajero compra un boleto de ida y vuelta:

- **Ida:** SFO → NYC (podría ser 2 segmentos)
- **Vuelta:** NYC → SFO (podría ser 1 segmento)
- **Total:** 1 itinerario que contiene 3 segmentos

El boleto completo tiene un número único, por ejemplo: 0062345678901

**¿Quién usa datos a nivel de itinerario?**

- **Finanzas:** Para contabilidad de ingresos totales
- **Contabilidad:** Para reconciliación de pagos

**Nota importante:** Aunque estos cuatro niveles existen, el equipo decidió **empezar con SEGMENTO**. Los otros niveles se pueden agregar en fases posteriores, y cuando eso suceda, podrán **reutilizar las dimensiones conformadas** que construyeron en la primera fase."

---



---

## SLIDE 11: KEY TAKEAWAY - CONCEPTO CLAVE

"Para concluir mi sección, el concepto clave que quiero que recuerden es:

**Diferentes niveles de granularidad sirven a diferentes usuarios del negocio con necesidades específicas.**

**No existe 'una tabla para gobernarlas a todas'.**

Si intentas crear una única tabla que sirva para operaciones, marketing, ventas Y finanzas, terminarás con:

- Una tabla demasiado detallada para algunos usuarios (lenta, confusa)



- Una tabla no suficientemente detallada para otros usuarios (no pueden hacer su trabajo)
- Métricas que no tienen sentido a ciertos niveles

**La solución correcta:** Múltiples tablas de hechos a diferentes granularidades, cada una optimizada para su propósito específico, pero todas compartiendo las mismas **dimensiones conformadas**.

Esto permite que un usuario de marketing pueda trabajar cómodamente a nivel de segmento, mientras que un ingeniero de operaciones puede profundizar hasta nivel de tramo cuando lo necesite, y ambos están hablando el mismo idioma sobre las dimensiones (fechas, aeropuertos, aeronaves, etc.).

# SECCIÓN PERSONA 3 - Diseño Dimensional del Esquema (7 minutos)

---

## SLIDE 12: INTRODUCCIÓN AL ESQUEMA ESTRELLA

Ahora voy a explicarles cómo se implementa técnicamente el diseño dimensional usando el esquema estrella.

### ¿Qué es un Esquema Estrella?

El esquema estrella es el diseño dimensional fundamental en data warehousing. Consiste en:

- Una **tabla de HECHOS** central
- Múltiples **tablas de DIMENSIONES** que rodean la tabla de hechos

Si lo visualizan desde arriba, parece una estrella - de ahí el nombre.

### LA TABLA DE HECHOS - El centro de la estrella

La tabla de hechos contiene:

- **Métricas numéricas (hechos):** Son las medidas que queremos analizar - ingresos, millas, impuestos, cantidades
- **Claves foráneas a dimensiones:** Conexiones a las tablas de dimensiones
- **Granularidad específica:** En nuestro caso, una fila por pase de abordaje
- **Muchas filas:** Estamos hablando de millones, potencialmente billones de filas

### LAS TABLAS DE DIMENSIONES - Los puntos de la estrella

Las dimensiones contienen:

- **Atributos descriptivos:** Texto, categorías, descripciones
- **Información de contexto:** El 'quién, qué, dónde, cuándo, cómo, por qué'
- **Estructuras jerárquicas:** Por ejemplo: aeropuerto → ciudad → estado → país
- **Relativamente pocas filas:** Miles, decenas de miles, raramente millones

### La diferencia clave:

- Hechos = números que SUMAMOS, PROMEDIAMOS, CONTAMOS
- Dimensiones = contexto por el cual FILTRAMOS, AGRUPAMOS, SEGMENTAMOS

Por ejemplo:

- 'Ingresos de \$50,000' es un HECHO
- 'En el aeropuerto de Atlanta' es una DIMENSIÓN

- 'El 15 de marzo de 2024' es una DIMENSIÓN
  - 'Para vuelos internacionales' es una DIMENSIÓN"
- 

## SLIDE 13: ESQUEMA DE ACTIVIDAD DE VUELOS

"Ahora veamos el esquema específico para Actividad de Vuelos a nivel de segmento.

**Recordatorio de granularidad:** Una fila por cada pase de abordaje recopilado de los pasajeros.

### LA TABLA DE HECHOS CENTRAL: HECHOS\_ACTIVIDAD\_VUELO

Esta tabla contiene múltiples claves foráneas (FK) que apuntan a las dimensiones:

#### Claves Foráneas (Dimensiones de Tiempo):

- fecha\_salida\_key → DIM\_FECHA
- hora\_salida\_key → DIM\_HORA
- fecha\_llegada\_key → DIM\_FECHA (otra instancia)
- hora\_llegada\_key → DIM\_HORA (otra instancia)

#### Claves Foráneas (Dimensiones de Ubicación):

- aeropuerto\_origen\_key → DIM\_AEROPUERTO
- aeropuerto\_destino\_key → DIM\_AEROPUERTO (otra instancia)
- aeropuerto\_origen\_viaje\_key → DIM\_AEROPUERTO (otra instancia)
- aeropuerto\_destino\_viaje\_key → DIM\_AEROPUERTO (otra instancia)

#### Claves Foráneas (Dimensiones de Negocio):

- pasajero\_key → DIM\_PASAJERO
- perfil\_pasajero\_key → DIM\_PERFIL\_PASAJERO
- aeronave\_key → DIM\_AERONAVE
- clase\_servicio\_key → DIM\_CLASE\_SERVICIO
- base\_tarifa\_key → DIM\_BASE\_TARIFA
- canal\_ventas\_key → DIM\_CANAL\_VENTAS

#### Los HECHOS (métricas numéricas):

- ingresos\_tarifa\_base (dinero)
- cargos\_instalaciones (dinero)
- impuestos\_aeroportuarios (dinero)
- impuestos\_gubernamentales (dinero)
- tarifas\_auxiliares (dinero)
- millas\_segmento\_voladas (distancia)
- millas\_segmento\_otorgadas (distancia - puede ser diferente si hay mínimo)

### Dimensiones Degeneradas (números operacionales sin tabla):

- numero\_itinerario
- numero\_boleto
- numero\_vuelo
- numero\_secuencia\_segmento

Estas son dimensiones que no tienen su propia tabla porque son identificadores únicos sin atributos descriptivos adicionales.

Noten algo muy importante: varias claves foráneas apuntan a la **misma tabla física de dimensión**. Esto es lo que llamamos **dimensiones de rol**, y es una técnica fundamental que voy a explicar ahora."

---

## SLIDE 14: DIMENSIONES DE ROL

"Las dimensiones de rol son un concepto clave en el diseño dimensional de Kimball.

### ¿Qué son las Dimensiones de Rol?

Ocurren cuando la **misma dimensión se usa múltiples veces** en la tabla de hechos, pero con diferentes propósitos o 'roles'.

### Ejemplo perfecto: La Dimensión AEROPUERTO

En nuestra tabla de hechos tenemos 4 referencias a aeropuertos:

1. Aeropuerto de ORIGEN del segmento
2. Aeropuerto de DESTINO del segmento
3. Aeropuerto de ORIGEN del viaje completo
4. Aeropuerto de DESTINO del viaje completo

### La pregunta es: ¿Necesitamos 4 tablas físicas diferentes de aeropuertos?

¡NO! Tenemos **UNA tabla física** con todos los aeropuertos, y creamos **4 vistas lógicas** con nombres diferentes.

### Tabla Física Única: DIM\_AEROPUERTO

Contiene columnas como:

- aeropuerto\_key (clave primaria)
- codigo\_aeropuerto (ej: 'ATL', 'SFO', 'JFK')
- nombre\_aeropuerto (ej: 'Hartsfield-Jackson Atlanta International')
- ciudad (ej: 'Atlanta')
- estado (ej: 'Georgia')
- pais (ej: 'Estados Unidos')

- region (ej: 'Sureste')
- codigo\_iata (ej: 'ATL')
- latitud, longitud
- zona\_horaria
- etc.

Esta tabla tiene todas las características descriptivas de cada aeropuerto.

#### Las 4 Vistas Lógicas (Roles):

1. **Vista: Aeropuerto\_Origen**
  - Interpretación: De dónde SALE el segmento
  - Uso en query: "WHERE ciudad\_origen = 'San Francisco'"
2. **Vista: Aeropuerto\_Destino**
  - Interpretación: A dónde LLEGA el segmento
  - Uso en query: "WHERE ciudad\_destino = 'Minneapolis'"
3. **Vista: Aeropuerto\_Origen\_Viaje**
  - Interpretación: Dónde COMENZÓ el viaje completo del pasajero
  - Uso en query: "WHERE ciudad\_origen\_viaje = 'San Francisco'"
4. **Vista: Aeropuerto\_Destino\_Viaje**
  - Interpretación: Dónde TERMINARÁ el viaje completo del pasajero
  - Uso en query: "WHERE ciudad\_destino\_viaje = 'Minneapolis'"

#### Implementación técnica:

En la tabla de hechos tenemos 4 claves foráneas:

- aeropuerto\_origen\_key (apunta a DIM\_AEROPUERTO)
- aeropuerto\_destino\_key (apunta a DIM\_AEROPUERTO)
- aeropuerto\_origen\_viaje\_key (apunta a DIM\_AEROPUERTO)
- aeropuerto\_destino\_viaje\_key (apunta a DIM\_AEROPUERTO)

Todas apuntan a la misma tabla física, pero con nombres diferentes y significado diferente en el contexto del negocio.

#### Beneficios de las Dimensiones de Rol:

1. **Mantenimiento más fácil:** Una sola tabla para actualizar
2. **Dimensión conformada:** La misma definición de aeropuerto en todos lados
3. **Queries intuitivos:** Los usuarios pueden filtrar por 'ciudad\_origen' o 'ciudad\_destino' naturalmente
4. **Reduce almacenamiento:** No duplicamos la información de aeropuertos
5. **Consistencia garantizada:** Si el nombre de un aeropuerto cambia, cambia automáticamente en todos los roles

Lo mismo sucede con Fecha y Hora - tenemos una DIM\_FECHA física, pero la usamos en múltiples roles: fecha\_salida, fecha\_llegada, etc."

---

## SLIDE 15: DIMENSIÓN DE PASAJERO

"Ahora hablemos de una dimensión muy importante y con características especiales: la Dimensión de Pasajero.

### Características de DIM\_PASAJERO:

**Característica 1: Volumen Enorme** Para una aerolínea grande, esta dimensión puede tener:

- Decenas de millones de filas
- Potencialmente cientos de millones si incluyen pasajeros ocasionales

**Característica 2: Atributos Ricos** Especialmente para los viajeros frecuentes más valiosos, capturamos mucha información:

- Información personal: nombre, email, teléfono, dirección
- Información demográfica: fecha de nacimiento, género
- Información geográfica: ciudad, estado, código postal, país
- Preferencias: asiento preferido, preferencias de comida
- Información del programa de lealtad

**Característica 3: Focus en VIPs** Los viajeros frecuentes son los clientes más valiosos. La aerolínea invierte en capturar información detallada sobre ellos.

**Característica 4: Auto-mantenimiento** Algo único: los viajeros frecuentes están **motivados a mantener sus datos actualizados** porque quieren asegurarse de recibir el crédito de millaje apropiado. Esto ayuda a la calidad de datos.

### Atributos de ejemplo en DIM\_PASAJERO:

pasajero\_key (PK) ← Clave primaria  
numero\_viajero\_frecuente  
nombre\_completo  
email  
telefono  
fecha\_nacimiento  
genero  
direccion\_completa  
ciudad  
estado  
codigo\_postal  
pais  
fecha\_inicio\_programa  
preferencia\_asiento  
preferencia\_comida  
...

Y aquí vienen los atributos problemáticos (los marco en rojo en la presentación):

...

nivel\_viajero\_frecuente ← CAMBIA con el tiempo  
aeropuerto\_origen\_preferido ← CAMBIA con el tiempo  
estado\_membresia\_club ← CAMBIA con el tiempo  
millas\_acumuladas\_vida ← CAMBIA constantemente  
...

**\*\*El Problema:\*\***

Algunos atributos cambian con el tiempo:

- Un pasajero empieza como 'Básico', luego sube a 'Nivel Medio', luego a 'Élite'
- Se muda de ciudad, cambia su aeropuerto de origen
- Se une al club aeroportuario
- Acumula más millas

**\*\*Pregunta crítica: ¿Usamos SCD Tipo 2 (Slowly Changing Dimension Type 2)?\*\***

SCD Tipo 2 significa que cada vez que un atributo cambia, creamos una nueva fila con:

- Nueva clave
- Fecha efectiva desde/hasta
- Indicador de fila actual

**\*\*El problema con SCD Tipo 2 aquí:\*\***

Con **\*\*millones de pasajeros\*\***, y estos atributos cambiando regularmente:

- La tabla EXPLOTARÍA en tamaño
- Tendríamos decenas o cientos de millones de filas
- Las queries se volverían muy lentas
- El mantenimiento sería una pesadilla

Necesitamos una solución más inteligente."

---

## ## SLIDE 16: MINI-DIMENSIÓN DE PERFIL DE PASAJERO

"La solución inteligente es usar una **\*\*Mini-Dimensión\*\***, también conocida como técnica SCD Tipo 4.

**\*\*Concepto de Mini-Dimensión:\*\***

Extraemos los atributos que cambian frecuentemente a una **\*\*dimensión separada más pequeña\*\*** que contiene todas las combinaciones posibles de esos atributos.

**\*\*División de la información:\*\***

**\*\*DIM\_PASAJERO (tabla grande, atributos estables):\*\***

...

pasajero\_key (PK)  
numero\_viajero\_frecuente  
nombre\_completo  
email  
telefono  
fecha\_nacimiento  
genero  
direccion\_completa  
ciudad, estado, pais  
fecha\_inicio\_programa  
preferencias (asiento, comida)  
...

Estos son atributos que NO cambian frecuentemente.

**\*\*DIM\_PERFIL\_PASAJERO (mini-dimensión, atributos que cambian):\*\***

...

perfil\_pasajero\_key (PK)  
nivel\_viajero\_frecuente  
aeropuerto\_origen  
estado\_membresia\_club  
nivel\_millas\_acumuladas

Solo los 4 atributos que cambian frecuentemente.

### ¿Cómo funciona?

En lugar de tener millones de combinaciones, pre-calculamos todas las combinaciones **posibles** de estos atributos:

- Niveles de viajero frecuente: 4 opciones (Básico, Medio, Élite, Súper Élite)
- Aeropuertos principales: digamos 50 aeropuertos principales
- Estado membresía club: 2 opciones (Miembro, No Miembro)
- Nivel millas acumuladas: 5 rangos (< 100K, 100K-500K, 500K-1M, 1M-2M, 2M+)

Total de combinaciones posibles:  $4 \times 50 \times 2 \times 5 = \mathbf{2,000 \text{ filas}}$

¡De millones de filas a solo 2,000 filas!

**Ejemplo de filas en DIM\_PERFIL\_PASAJERO:**

Key	Nivel VF	Aeropuerto	Estado Club	Nivel Millas
-----	----------	------------	-------------	--------------



1	Básico	ATL	No Miembro	< 100K
2	Básico	ATL	Miembro	< 100K
3	Básico	BOS	No Miembro	< 100K
...	...	...	...	...
789	Nivel Medio	ATL	No Miembro	100K-500K
790	Nivel Medio	ATL	Miembro	100K-500K
...	...	...	...	...
2468	Élite	ATL	Miembro	1M-2M
2469	Élite	ATL	Miembro	2M-3M
...	...	...	...	...

### ¿Cómo se usa en la tabla de hechos?

Cada fila de la tabla de hechos tiene DOS claves foráneas:

1. **pasajero\_key** → apunta a DIM\_PASAJERO (quién es)
2. **perfil\_pasajero\_key** → apunta a DIM\_PERFIL\_PASAJERO (su perfil en ese momento)

Cuando un pasajero sube de nivel, no modificamos su fila en DIM\_PASAJERO.

Simplemente las nuevas filas en la tabla de hechos apuntan a un **perfil\_pasajero\_key** diferente.

### Beneficios de la Mini-Dimensión:

1. **Tabla pequeña:** 2,000 filas vs. millones
2. **Tracking histórico:** Podemos ver cómo evolucionó el perfil del pasajero a lo largo del tiempo
3. **Queries más rápidos:** Menos filas para escanear
4. **Análisis independiente:** Marketing puede analizar solo la mini-dimensión sin tocar la tabla enorme de pasajeros
5. **Mantenimiento simple:** Pre-calcular las combinaciones una vez

### Caso de uso real:

Marketing pregunta: '¿Cuántos vuelos tomaron pasajeros de nivel Élite desde Atlanta en el último trimestre?'

Con la mini-dimensión:

```
sql
SELECT COUNT(*)
FROM HECHOS_ACTIVIDAD_VUELO h
JOIN DIM_PERFIL_PASAJERO p ON h.perfil_pasajero_key = p.perfil_pasajero_key
JOIN DIM_FECHA f ON h.fecha_salida_key = f.fecha_key
WHERE p.nivel_viajero_frecuente = 'Élite'
      AND p.aeropuerto_origen = 'ATL'
      AND f.trimestre = 'Q1-2024'
```

Query simple, rápido, sin tocar la tabla enorme de pasajeros."

---

## SLIDE 17: OTRAS DIMENSIONES CLAVE

"Déjenme hacer un repaso rápido de las otras dimensiones importantes en nuestro esquema:

### **DIM\_AERONAVE (Dimensión de Aeronave)**

Información sobre cada avión de la flota:

- aeronave\_key (clave)
- numero\_cola (ej: 'N12345')
- tipo\_aeronave (ej: 'Boeing 737-800', 'Airbus A320')
- fabricante (ej: 'Boeing', 'Airbus')
- año\_fabricacion
- capacidad\_asientos\_total
- configuracion\_asientos (ej: '16 Primera, 30 Ejecutiva, 134 Económica')
- edad\_aeronave
- estado (activa, en mantenimiento, retirada)

### **DIM\_CLASE\_SERVICIO (Clase en que voló)**

Describe la clase de servicio:

- clase\_servicio\_key
- nombre\_clase
- tipo (Primera, Ejecutiva, Premium Económica, Económica)
- descripcion\_servicio
- amenidades\_incluidas
- prioridad\_embarque

### **DIM\_BASE\_TARIFA (Términos de la tarifa)**

Los términos y condiciones de la tarifa pagada:

- base\_tarifa\_key

- codigo\_tarifa (ej: 'Y', 'B', 'M')
- tipo\_tarifa (ej: 'Sin restricciones', 'Compra anticipada 21 días')
- restricciones (texto descriptivo)
- penalidades\_cambio (monto o porcentaje)
- reembolsable (Sí/No)
- dias\_anticipacion\_requeridos
- promocion\_especial (Si es parte de una promo)
- estadia\_minima\_requerida
- estadia\_maxima\_permitida

Esta dimensión es crucial para revenue management y pricing.

### **DIM\_CANAL\_VENTAS (Cómo se compró)**

Cómo el pasajero compró el boleto:

- canal\_ventas\_key
- nombre\_canal
- tipo\_canal (Web, Agencia de Viajes, Call Center, Kiosko Aeropuerto, App Móvil)
- proveedor\_servicios (ej: 'Expedia', 'Booking.com', 'Directo')
- comision\_pagada (porcentaje)
- categoria\_canal (Directo, Indirecto)

### **Dimensiones Degeneradas:**

Recuerden que también tenemos números operacionales que NO tienen tabla de dimensión porque son identificadores únicos sin atributos adicionales:

- Número de Itinerario
- Número de Boleto
- Número de Vuelo
- Secuencia del Segmento

Estas se almacenan directamente en la tabla de hechos.

### **El resultado final:**

Tenemos aproximadamente:

- 10-12 dimensiones con tablas
- 4-5 dimensiones degeneradas
- Múltiples hechos numéricos

Esta combinación nos da un **análisis extremadamente poderoso y flexible.**"

---

## **SLIDE 18: KEY TAKEAWAY - CONCEPTO CLAVE**

"Para cerrar mi sección, el concepto clave que quiero que recuerden:

**El uso de mini-dimensiones y dimensiones de rol es esencial para manejar grandes volúmenes y múltiples perspectivas.**

**Principios clave:**

1. **No todas las dimensiones deben ser enormes:** Separa lo que cambia de lo que es estable
  - DIM\_PASAJERO: Estable, grande
  - DIM\_PERFIL\_PASAJERO: Cambia, pero pequeña
2. **Las dimensiones de rol evitan duplicación:** Una tabla física, múltiples roles lógicos
  - DIM\_AEROPUERTO física
  - Roles: origen, destino, origen\_viaje, destino\_viaje
3. **La granularidad de la tabla de hechos debe ser clara y específica:**
  - Nuestra granularidad: Una fila por pase de abordaje
  - Esto dicta qué hechos tienen sentido capturar
4. **Las dimensiones conformadas son la clave para la integración:**
  - La misma DIM\_FECHA se usa en todos los procesos
  - La misma DIM\_AEROPUERTO se usa en múltiples contextos

**Resultado:** Un diseño dimensional robusto que:

- Maneja grandes volúmenes eficientemente
- Permite análisis flexible
- Es fácil de entender para usuarios de negocio
- Se puede extender en el futuro

# SECCIÓN PERSONA 4 - Técnicas Avanzadas (7 minutos)

---

## SLIDE 19: INTRODUCCIÓN - EL PROBLEMA DE LOS VIAJES

Ahora voy a hablarles sobre algunas técnicas avanzadas que resuelven problemas reales del negocio.

A pesar del poderoso esquema dimensional que acabamos de diseñar, hay un problema fundamental que aún no podemos resolver fácilmente.

### El Problema:

Marketing quiere responder una de las preguntas más importantes sobre los viajeros frecuentes:

**'¿A dónde van realmente nuestros pasajeros?'**

Parece una pregunta simple, pero con el grano a nivel de **segmento**, es casi imposible responderla correctamente.

### ¿Por qué es difícil?

Déjenme explicarles con un ejemplo concreto:

### Caso: Pasajero Juan Pérez

Juan compra un boleto y toma estos vuelos:

- Segmento 1: SFO → DEN (salida 11:00 AM, llegada 2:00 PM)
- Segmento 2: DEN → MSP (salida 4:00 PM, llegada 7:00 PM)

### Pregunta: ¿Cuál es el viaje REAL de Juan?

- ¿Es SFO → DEN? No, claramente Denver es solo una escala.
- ¿Es DEN → MSP? No, Juan no empezó su viaje en Denver.
- ¿Es SFO → MSP? ¡Sí! Este es el viaje real que Juan quería hacer.

### Los Desafíos Técnicos:

1. **Itinerarios circulares:** La mayoría de los itinerarios completos empiezan y terminan en el mismo aeropuerto (viajes redondos). Entonces si vemos SFO → DEN → MSP → SFO, ¿cómo sabemos que el destino real era MSP?
2. **Identificar destinos vs. escalas:** ¿Cómo diferenciamos una escala técnica de 2 horas de un destino significativo?
3. **Procesamiento complejo en cada query:** Si dejamos esto para que lo resuelvan las aplicaciones de BI, cada analista tendría que escribir lógica compleja para:
  - Recuperar todos los segmentos de un viaje
  - Ordenarlos por secuencia
  - Identificar paradas prolongadas
  - Determinar origen y destino reales
4. **Usuarios no técnicos no pueden hacerlo:** La persona de marketing que solo sabe usar herramientas de BI point-and-click no puede escribir esta lógica.
5. **Inconsistencia:** Diferentes aplicaciones implementarían la lógica de manera diferente, resultando en reportes inconsistentes.

### Principio Fundamental de Kimball:

'La complejidad debe resolverse en el ETL (Extract, Transform, Load), NO en la capa de reportes.'

Los usuarios de BI deben poder hacer queries simples y obtener respuestas correctas."

---

## SLIDE 20: SOLUCIÓN - DIMENSIONES ADICIONALES DE VIAJE

"La solución elegante es agregar dos **dimensiones de rol adicionales** que se calculan durante el proceso de ETL.

### La Solución:

Agregamos a la tabla de hechos de segmentos:

- **aeropuerto\_origen\_viaje\_key** → DIM\_AEROPUERTO
- **aeropuerto\_destino\_viaje\_key** → DIM\_AEROPUERTO

Estas son dimensiones ADICIONALES. NO reemplazamos las dimensiones de origen y destino del segmento.

Entonces, nuestra tabla de hechos ahora tiene:

### Dimensiones de Segmento (ya las teníamos):

- **aeropuerto\_origen\_key** → De dónde sale este segmento específico
- **aeropuerto\_destino\_key** → A dónde llega este segmento específico

### Dimensiones de Viaje (NUEVAS):

- aeropuerto\_origen\_viaje\_key → Dónde COMENZÓ el viaje completo del pasajero
- aeropuerto\_destino\_viaje\_key → Dónde TERMINARÁ el viaje completo del pasajero

### ¿Cómo se determinan estas dimensiones?

Durante el proceso de ETL, el sistema:

1. Recupera todos los segmentos de un boleto/itinerario
2. Los ordena por secuencia
3. Busca paradas de **más de 4 horas**
4. Aplica la definición oficial de la aerolínea de 'escala' vs. 'destino'

**Criterio clave:** Parada de más de 4 horas = Destino significativo del viaje

Este es el estándar oficial de la aerolínea para distinguir escalas de destinos reales.

### Ejemplo con Datos Reales:

Segmento	Origen Seg	Destino Seg	Origen VIAJE	Destino VIAJE	Duración Escala
1	SFO	DEN	<b>SFO</b>	<b>MSP</b>	2 horas
2	DEN	MSP	<b>SFO</b>	<b>MSP</b>	-

Noten que AMBOS segmentos tienen el mismo origen\_viaje (SFO) y destino\_viaje (MSP), porque esos son los puntos verdaderos de inicio y fin del viaje completo.

Si hubiera una parada de más de 4 horas en Denver, entonces Denver sería considerado un destino intermedio del viaje.

### Beneficios de esta Solución:

1.  **Query simple:**

sql

```
SELECT COUNT(*)
```

```
FROM HECHOS_ACTIVIDAD_VUELO
```

```
WHERE aeropuerto_origen_viaje = 'SFO'
```

AND aeropuerto\_destino\_viaje = 'MSP'

2. **Usuarios no técnicos pueden usarlo:** Solo necesitan arrastrar y soltar campos en su herramienta de BI
3. **Se calcula UNA VEZ:** En el ETL, no en cada query
4. **Consistencia garantizada:** Todos los reportes usan la misma lógica
5. **Mantiene granularidad de segmento:** No perdemos detalle, solo agregamos contexto

### Consideración Importante:

Cuando resumimos datos por viaje usando este esquema, debemos tener cuidado. Algunas dimensiones aplican al viaje completo (como origen\_viaje y destino\_viaje), mientras que otras aplican solo al segmento individual (como clase de servicio o base de tarifa).

Por ejemplo, un viaje podría tener:

- Segmento 1: Tarifa económica sin restricciones
- Segmento 2: Tarifa económica con descuento

¿Cuál es 'la tarifa del viaje'? No hay una respuesta única - depende de la pregunta de negocio."

---

---

## SLIDE 21: COMBINANDO DIMENSIONES CORRELACIONADAS - INTRODUCCIÓN

"Ahora quiero hablarles sobre una técnica avanzada que es una **excepción a las reglas normales** de diseño dimensional.

### La Regla General (que aprendimos antes):

Si existe una relación muchos-a-muchos entre dos grupos de atributos de dimensión, deben modelarse como **dimensiones separadas con claves foráneas separadas** en la tabla de hechos.

Por ejemplo:

- Un pasajero puede volar en muchas aeronaves
- Una aeronave puede transportar muchos pasajeros
- Por lo tanto: DIM\_PASAJERO separada, DIM\_AERONAVE separada

Esta es la regla que seguimos en el 99% de los casos.

### Las Excepciones (muy específicas):



En situaciones MUY específicas, podemos combinar múltiples dimensiones relacionadas en una sola tabla en lugar de mantenerlas separadas.

### ¿Cuándo considerar combinar dimensiones?

Solo cuando se cumplen estas condiciones:

#### Condición 1: Volumen de datos extremadamente pequeño

- Estamos hablando de docenas o cientos de filas, no miles o millones
- El producto cartesiano de las dimensiones resulta en una tabla manejable

#### Condición 2: Se necesitan atributos adicionales que dependen de la combinación

- Hay atributos descriptivos que solo tienen sentido cuando consideras AMBAS dimensiones juntas
- Estos atributos no 'pertenecen' naturalmente a ninguna de las dimensiones individuales

#### Condición 3: Las dimensiones están altamente correlacionadas

- Hay una relación natural entre las dimensiones
- Los usuarios de negocio piensan en ellas como una unidad

#### Importante:

Esta es una **excepción, NO la regla**. La mayoría de las veces, dimensiones separadas es la respuesta correcta. Pero Kimball identifica situaciones donde combinarlas tiene más sentido.

Veamos dos ejemplos del caso de aerolíneas donde combinar dimensiones es la solución correcta."

---

## SLIDE 22: CASO 1 - CLASE DE SERVICIO COMBINADA

"El primer caso es la Clase de Servicio.

#### El Requerimiento de Negocio:

Marketing dice: 'Queremos analizar la actividad según:

1. La **clase que el pasajero COMPRÓ** (lo que pagó originalmente)
2. La **clase en que el pasajero realmente VOLÓ** (donde terminó sentado)
3. Si hubo una **mejora o degradación** de clase'

#### Opción 1: Dimensiones Separadas (enfoque tradicional)

Tendríamos:

- **dim\_clase\_comprada** (4 filas: Primera, Ejecutiva, Premium Eco, Económica)
- **dim\_clase\_volada** (4 filas: Primera, Ejecutiva, Premium Eco, Económica)
- **dim\_indicador\_cambio** (3 filas: Mejora, Degradación, Sin Cambio)

En la tabla de hechos: **3 claves foráneas separadas**

#### Problemas con este enfoque:

1. **Lógica compleja en queries:** La aplicación de BI tendría que identificar escenarios como:
  - Económica → Premium Eco = Mejora
  - Económica → Ejecutiva = Mejora
  - Económica → Primera = Mejora
  - Premium Eco → Económica = Degradación
  - Ejecutiva → Económica = Degradación
  - ... y así sucesivamente
2. **Tres claves foráneas:** Más complejidad en la tabla de hechos
3. **No hay un "hogar" natural para el indicador de cambio**

#### Opción 2: Dimensión Combinada (solución elegida)

Crear una sola **dim\_clase\_servicio** con el producto cartesiano:

- 4 clases compradas × 4 clases voladas = **16 filas total**

#### Estructura de DIM\_CLASE\_SERVICIO:

Key	Clase Comprada	Clase Volada	Grupo	Indicador Cambio
-----	----------------	--------------	-------	------------------

1	Económica	Económica	Eco-Eco	Sin Cambio
2	Económica	Premium Eco	Eco-Premium	<b>Mejora</b> ↑
3	Económica	Ejecutiva	Eco-Ejecutiva	<b>Mejora</b> ↑
4	Económica	Primera	Eco-Primera	<b>Mejora</b> ↑
5	Premium Eco	Económica	Premium-Eco	<b>Degradación</b> ↓
6	Premium Eco	Premium Eco	Premium-Premium	Sin Cambio
7	Premium Eco	Ejecutiva	Premium-Ejecutiva	<b>Mejora</b> ↑
8	Premium Eco	Primera	Premium-Primera	<b>Mejora</b> ↑
9	Ejecutiva	Económica	Ejec-Eco	<b>Degradación</b> ↓
10	Ejecutiva	Premium Eco	Ejec-Premium	<b>Degradación</b> ↓
11	Ejecutiva	Ejecutiva	Ejec-Ejec	Sin Cambio
12	Ejecutiva	Primera	Ejec-Primera	<b>Mejora</b> ↑
13	Primera	Económica	Primera-Eco	<b>Degradación</b> ↓

14	Primera	Premium Eco	Primera-Premium	<b>Degradación</b> ↓
15	Primera	Ejecutiva	Primera-Ejecutiva	<b>Degradación</b> ↓
16	Primera	Primera	Primera-Primera	Sin Cambio

#### **Ventajas de la Dimensión Combinada:**

1. **Solo 1 FK en tabla de hechos** en lugar de 3
2. **Indicador pre-calculado:** El ETL determina si fue mejora/degradación
3. **Queries simples:**

sql

```
SELECT COUNT(*), SUM(ingresos)
```

```
FROM HECHOS_ACTIVIDAD_VUELO h
```

```
JOIN DIM_CLASE_SERVICIO c ON h.clase_servicio_key = c.clase_servicio_key
```

```
WHERE c.indicador_cambio = 'Mejora'
```

4. **16 filas = volumen totalmente manejable**
5. **Atributo adicional tiene hogar natural:** El 'Grupo Comprada-Volada' describe la relación

#### **Consideración Importante:**

Otras tablas de hechos de la aerolínea (como disponibilidad de inventario o compras de boletos) que solo conocen UNA clase invariablemente referenciarían una tabla de dimensión de clase conformada con solo 4 filas.

Entonces podrías tener:

- **dim\_clase** (4 filas) para la mayoría de procesos
- **dim\_clase\_servicio** (16 filas) específicamente para actividad de vuelos

O podrías incluir filas 'N/A' en la tabla de 16 filas para que sea conformada."

---

## **SLIDE 23: CASO 2 - RUTA DE PAR DE CIUDADES**

"El segundo caso es más sutil pero igualmente importante: los atributos de ruta.

### El Requerimiento de Negocio:

Los usuarios necesitan responder preguntas como:

- '¿Cuántos vuelos tenemos entre San Francisco y Denver?'
- '¿Cuál es la distancia de cada ruta?'
- '¿Es una ruta doméstica o internacional?'
- '¿Es transatlántica, transpacífica, o no oceánica?'

Y quieren la respuesta **sin importar la dirección**: SFO→DEN o DEN→SFO deben contarse juntos.

### Problema con Dimensiones Separadas de Origen y Destino:

Tenemos `dim_aeropuerto_origen` y `dim_aeropuerto_destino` separadas. Esto funciona bien para la mayoría de los análisis, pero presenta desafíos:

#### Desafío 1: Query complejo para análisis bidireccional

sql

WHERE (aeropuerto\_origen = 'SFO' AND aeropuerto\_destino = 'DEN')

OR (aeropuerto\_origen = 'DEN' AND aeropuerto\_destino = 'SFO')

Esto funciona, pero no es intuitivo para usuarios no técnicos.

**Desafío 2: Etiquetas inconsistentes** Diferentes aplicaciones podrían etiquetar la misma ruta como:

- "SFO-DEN"
- "DEN-SFO"
- "San Francisco-Denver"
- "Denver-San Fran"
- "SF-Denver"

Kimball dice: **'Sería una lástima tomarse la molestia de crear un data warehouse y luego permitir que el código de aplicación implemente etiquetas de reportes inconsistentes.'**

#### Desafío 3: Atributos que dependen del PAR

Hay atributos que solo tienen sentido cuando consideras AMBOS aeropuertos:

- **Distancia:** SFO a DEN son 1,235 millas. Esta distancia es un atributo de la RUTA, no de SFO ni de DEN individualmente.
- **Tipo de ruta:** La ruta SFO-DEN es 'Doméstica'. La ruta SFO-LGW (Londres) es 'Internacional Transatlántica'.

- **Nombre direccional vs. no direccional:** SFO→DEN es diferente de DEN→SFO en dirección, pero ambos son la ruta 'SFO-DEN' en sentido bidireccional.

### **Solución: Dimensión de Ruta de Par de Ciudades**

Agregar una **dimensión adicional** (no reemplazar origen/destino) con atributos específicos de la ruta:

`dim_ruta_par_ciudades"`

---

---

## **SLIDE 24: KEY TAKEAWAY - CONCEPTO CLAVE**

"Para cerrar mi sección, el concepto clave:

**Combinar dimensiones correlacionadas es una EXCEPCIÓN, no la regla, pero cuando se aplica correctamente resuelve problemas reales de usabilidad y rendimiento.**

### **Cuándo combinar dimensiones (recordatorio):**

1. Volumen extremadamente pequeño (decenas/cientos de filas)
2. Necesitas atributos que dependen de la combinación
3. Dimensiones altamente correlacionadas
4. Mejora significativa en usabilidad para usuarios de negocio

### **Cuándo NO combinar (la mayoría de los casos):**

1. Volumen grande (miles o millones de filas potenciales)
2. Relación muchos-a-muchos verdadera sin correlación
3. Las dimensiones son independientes conceptualmente
4. No hay atributos adicionales que dependan de la combinación

### **Ejemplos de nuestro caso:**

#### **Combinar:**

- Clase Comprada + Clase Volada = 16 filas + indicador de mejora
- Origen + Destino (en dim\_ruta) = miles de filas + distancia y tipo

#### **NO combinar:**

- Pasajero + Aeronave = millones × cientos = explosión combinatoria sin sentido
- Fecha + Aeropuerto = no hay atributos que dependan de la combinación

### **El balance correcto:**

El arte del diseño dimensional está en reconocer cuándo seguir las reglas y cuándo aplicar las excepciones apropiadas. La guía siempre debe ser:

1. ¿Simplifica las queries de los usuarios?
2. ¿Mejora el rendimiento?
3. ¿El volumen es manejable?
4. ¿Tiene sentido conceptual para el negocio?

# SECCIÓN PERSONA 5 - Consideraciones Internacionales y Cierre (5 minutos)

---

## SLIDE 25: CONSIDERACIONES INTERNACIONALES

Para cerrar esta presentación, voy a hablarles sobre los desafíos que enfrentamos cuando diseñamos data warehouses para empresas que operan globalmente.

Las aerolíneas son inherentemente negocios globales:

- Vuelan a múltiples países
- Operan en diferentes zonas horarias
- Sirven a clientes de diferentes culturas e idiomas
- Deben cumplir con regulaciones locales

Esto introduce complejidades adicionales que debemos manejar en nuestro diseño dimensional.

**Los principales desafíos internacionales que cubriremos:**

- 1. Fecha y Hora en Múltiples Zonas Horarias**
  - ¿Cuándo ocurrió realmente un evento?
  - ¿Hora local o hora estándar?
- 2. Calendarios Específicos por País**
  - Diferentes feriados
  - Diferentes definiciones de 'temporada alta'
  - Múltiples sistemas calendarios
- 3. Resumen de Localización**
  - Multimoneda
  - Multiidioma
  - Interfaces de usuario localizadas

Kimball dedica considerable atención a estos temas porque son fundamentales para empresas multinacionales."

---

## SLIDE 26: EL PROBLEMA DE LAS ZONAS HORARIAS

"Empecemos con uno de los problemas más complejos: las zonas horarias.



## El Dilema:

Cuando operamos en múltiples países o zonas horarias, surge una pregunta fundamental:

¿Deberíamos capturar la fecha y hora **relativas a la medianoche local** de cada zona horaria, o deberíamos expresar el período temporal en relación con un **estándar**, como:

- La fecha/hora de la sede corporativa
- GMT (Greenwich Mean Time)
- UTC (Tiempo Universal Coordinado)
- Zulu time (como lo llaman en aviación - es equivalente a UTC)

## Ejemplo del problema:

Un vuelo sale de San Francisco a las 11:00 PM hora local del Pacífico. Al mismo tiempo, en la sede de Atlanta (hora del Este), son las 2:00 AM del día siguiente.

**Pregunta:** ¿Qué fecha y hora registramos?

- ¿23:00 del día 15 (hora local de SFO)?
- ¿02:00 del día 16 (hora de la sede en Atlanta)?
- ¿06:00 del día 16 (UTC)?

## La creencia errónea común:

Mucha gente piensa: 'Bueno, hay 24 zonas horarias en el mundo (correspondientes a las 24 horas del día), así que solo necesito almacenar un offset de +/- horas respecto a UTC.'

## La realidad es MUCHO más compleja:

Existen **MÁS de 24 zonas horarias** en el mundo. Veamos algunos ejemplos:

## Ejemplos de complejidad:

1. **China:** Una SOLA zona horaria para todo el país
  - A pesar de su enorme extensión geográfica (equivalente a 4-5 zonas horarias de EE.UU.)
  - Todo el país usa UTC+8
2. **India:** Una zona horaria con offset de MEDIA HORA
  - UTC+5:30 (5 horas y 30 minutos)
  - No es un múltiplo de hora completa
3. **Australia:** Tres zonas horarias, una con offset de media hora
  - Zona Estándar: UTC+10
  - Zona Central: UTC+9:30 (media hora)
  - Zona Occidental: UTC+8
4. **Nepal:** Offset de CUARTO DE HORA
  - UTC+5:45 (5 horas y 45 minutos)
5. **Islas Chatham (Nueva Zelanda):** UTC+12:45
6. **Horario de Verano (Daylight Saving Time):**
  - No todos los países lo usan

- Los que lo usan cambian en diferentes fechas
- Algunos países del hemisferio sur tienen horario de verano cuando el norte tiene horario estándar

### ¿Por qué no podemos simplemente almacenar un offset?

Porque el offset UTC de un aeropuerto **no es constante**:

- Depende de la **ubicación** (obviamente)
- Depende de la **fecha** (por horario de verano)
- Puede cambiar por decisiones políticas

Por lo tanto, NO puedes almacenar el offset:

- Solo en la tabla de hechos (¿qué offset usar? ¿del origen? ¿del destino?)
- Solo en la tabla de aeropuertos (cambia con el tiempo por horario de verano)
- Solo en la tabla de fechas (no sabes de qué ubicación)

El offset depende de la combinación de **ubicación Y fecha**."

## SLIDE 27: SOLUCIÓN - FECHAS Y HORAS LOCALES Y ESTÁNDAR

"La solución correcta es reconocer que necesitamos AMBAS perspectivas temporales.

### El Enfoque Recomendado por Kimball:

Incluir en la tabla de hechos **dimensiones separadas** de fecha y hora para:

1. **Tiempo LOCAL** (en la zona horaria del evento)
2. **Tiempo ESTÁNDAR** (UTC/Zulu time)

Para un vuelo, esto significa:

### Dimensiones de Salida (4 FKs):

- `fecha_salida_local_key` → DIM\_FECHA
- `hora_salida_local_key` → DIM\_HORA
- `fecha_salida_utc_key` → DIM\_FECHA
- `hora_salida_utc_key` → DIM\_HORA

### Dimensiones de Llegada (4 FKs adicionales):

- `fecha_llegada_local_key` → DIM\_FECHA
- `hora_llegada_local_key` → DIM\_HORA
- `fecha_llegada_utc_key` → DIM\_FECHA

- `hora_llegada_utc_key` → DIM\_HORA

**Total: 8 claves foráneas solo para tiempo**

Sí, son muchas claves foráneas, pero cada una tiene un propósito claro.

**Ejemplo Concreto:**

Vuelo de San Francisco (PST, UTC-8) a París (CET, UTC+1):

Evento	Fecha Local	Hora Local	Fecha UTC	Hora UTC
Salida	2024-03-15	23:00	2024-03-16	07:00
Llegada	2024-03-16	19:00	2024-03-16	18:00

**¿Por qué necesitamos ambas?**

**Tiempo LOCAL permite responder:**

- '¿Cuántos vuelos salen entre 6:00 AM y 9:00 AM hora local?' (hora pico de mañana)
- '¿Qué proporción de retrasos ocurren en vuelos nocturnos?' (definido por hora local)
- '¿Cuál es el patrón de demanda por hora del día en cada aeropuerto?'

**Tiempo ESTÁNDAR (UTC) permite responder:**

- '¿Cuántos vuelos estaban en el aire simultáneamente a nivel global?'
- '¿Cuál fue la secuencia real de eventos?' (cuando hay incidentes que afectan múltiples ubicaciones)
- '¿Cuánto tiempo REALMENTE tomó el vuelo?' (sin confusión por cambios de zona horaria)

**Recordatorio de DIM\_HORA:**

Como vimos en capítulos anteriores, la dimensión de hora del día permite:

- Agrupación por turnos (Matutino, Vespertino, Nocturno)
- Bloques horarios de alta demanda (rush periods)
- Categorización por parte del día (Madrugada, Mañana, Tarde, Noche)

Tener esto tanto en hora local como UTC es valioso para diferentes tipos de análisis."

---

## SLIDE 28: CALENDARIOS ESPECÍFICOS POR PAÍS

"Ahora hablemos sobre otro desafío: diferentes calendarios y definiciones de feriados.

### El Desafío:

Si el sistema de DW/BI sirve necesidades multinacionales, debemos manejar:

- Diferentes definiciones de feriados
- Diferentes temporadas (verano en el hemisferio norte es invierno en el sur)
- Diferentes semanas laborales (en algunos países, el fin de semana es viernes-sábado)
- Potencialmente diferentes sistemas de calendario (Gregoriano, Hebreo, Islámico, Chino)

### La Solución: Tabla Auxiliar (Outtrigger) de Calendario por País

Kimball propone mantener:

#### DIM\_FECHA Principal (atributos genéricos):

- fecha\_key (PK)
- fecha\_completa
- día, mes, año
- día\_semana
- semana\_año
- trimestre
- Calendarios múltiples si aplica:
  - Gregoriano: día, mes, año
  - Hebreo: día, mes, año
  - Islámico: día, mes, año
  - Chino: día, mes, año

Esta dimensión contiene atributos **independientes del país**.

#### DIM\_FECHA\_PAIS (tabla auxiliar/outtrigger):

- fecha\_key (PK, compuesta)
- pais\_key (PK, compuesta)
- nombre\_feriado
- tipo\_feriado (Nacional, Regional, Religioso)
- nombre\_temporada (Verano, Invierno, Temporada Alta Turística)
- dia\_laboral (Sí/No)
- tipo\_dia (Laborable, Fin de Semana, Feriado)

#### Estructura de Clave Compuesta:

La clave primaria es la **combinación** de fecha + país.

**Ejemplo de filas:**

Fecha	País	Nombre Feriado	Temporada	Día Laboral
2024-07-04	USA	Independence Day	Verano	No
2024-07-04	France	-	Verano	Sí
2024-07-14	USA	-	Verano	Sí
2024-07-14	France	Bastille Day	Verano	No
2024-12-25	USA	Christmas	Invierno	No
2024-12-25	Australia	Christmas	Verano	No

Noten que:

- El 4 de julio es feriado en EE.UU. pero día normal en Francia
- El 25 de diciembre es invierno en EE.UU. pero verano en Australia

**¿Cómo se usa?**

Puedes unir esta tabla a la dimensión de calendario principal como una tabla auxiliar (outrigger), o directamente a la tabla de hechos si incluyes una FK de país.

**Enfoque con Interfaz:**

Si proporcionas una interfaz que requiere que el usuario especifique un país, entonces los atributos del suplemento específico por país pueden considerarse lógicamente anexados a la tabla de fechas principal.

Esto permite visualizar el calendario desde la perspectiva de un solo país a la vez.

**Complejidades Adicionales:**

1. **Feriados locales:** En algunos países, diferentes regiones tienen feriados diferentes
  - En España: feriados autonómicos
  - En EE.UU.: algunos feriados estatales
2. **Feriados móviles:** Algunos feriados cambian de fecha cada año
  - Pascua (Easter): calculada algorítmicamente
  - Ramadán: basado en calendario lunar
  - Año Nuevo Chino: fecha variable en calendario gregoriano
3. **Cambios políticos:** Los feriados pueden agregarse o eliminarse por decisiones gubernamentales

Todo esto hace que la construcción de calendarios específicos por país sea compleja, pero Kimball argumenta que es necesaria para análisis multinacional correcto."

---

---

---

## SLIDE 29: KEY TAKEAWAY FINAL - CIERRE

"Para cerrar toda la presentación, el mensaje final que queremos que recuerden:

**El diseño dimensional de Kimball es FLEXIBLE y ADAPTABLE, pero requiere planificación cuidadosa y decisiones conscientes sobre:**

1. **Granularidad** - El nivel de detalle apropiado
2. **Dimensiones** - Qué compartir, qué separar, qué combinar
3. **Conformidad** - Mantener significado consistente
4. **Localización** - Planificar para alcance global desde el inicio
5. **Usabilidad** - Siempre pensar en el usuario final de BI

**El caso de aerolíneas nos enseñó:**

- Que los problemas complejos de negocio tienen soluciones dimensionales elegantes
- Que no existe 'una talla única' - diferentes usuarios necesitan diferentes vistas
- Que las técnicas avanzadas (mini-dimensiones, dimensiones combinadas) existen por razones válidas
- Que el diseño debe anticipar complejidad futura (internacional, múltiples granularidades)

**Estos principios se aplican mucho más allá de aerolíneas:**

Cualquier organización que:

- Mueva personas, productos o información entre ubicaciones
- Opere en múltiples geografías
- Tenga diferentes perspectivas de los mismos datos
- Necesite rastrear cambios históricos en atributos

...puede beneficiarse de estas técnicas.

### **Llamado a la Acción:**

Cuando diseñen su próximo data warehouse:

1. Empiecen con la matriz de bus
  2. Declaren la granularidad explícitamente
  3. Identifiquen oportunidades para dimensiones de rol
  4. Evalúen si necesitan mini-dimensiones
  5. Planifiquen para crecimiento internacional
  6. Y sobre todo: mantengan al usuario de negocio en el centro del diseño
- 

## **RESUMEN DE DISTRIBUCIÓN DE TIEMPOS:**

- **Persona 1 (Introducción y Contexto):** 5 minutos
- **Persona 2 (Granularidades):** 6 minutos
- **Persona 3 (Diseño Dimensional):** 7 minutos
- **Persona 4 (Técnicas Avanzadas):** 7 minutos
- **Persona 5 (Internacional y Cierre):** 5 minutos
- **Total:** 30 minutos