



TALLER DE ANÁLISIS DE SISTEMAS

TALLER BI

Certified



Corporation

ABRIL 2021

Marcos Contreras Fuentes.



LA IMPORTANCIA DE LA INFORMACIÓN

**CUÁNTO PAGARÍAS POR EL DIARIO DE
AYER?**

POR EL DE HOY?

POR EL DE MAÑANA?

**LA
INFORMACIÓN
ES PODER**

QUE ES BUSINESS INTELLIGENCE

BI es un proceso interactivo para explorar y analizar información estructurada sobre un área, para descubrir tendencias o patrones, a partir de los cuales derivar ideas y extraer conclusiones.

El proceso de Business Intelligence incluye la comunicación de los descubrimientos y efectuar los cambios. Las áreas incluyen clientes, proveedores, productos, servicios y competidores.



CONCEPTOS

- Proceso interactivo
- Explorar
- Analizar información estructurada
- Área de análisis
- DataWarehouse
- Comunicar los resultados
- Efectuar los cambios

Disposición de la información correcta en el momento adecuado para la toma de decisiones:

- No es necesario solicitar a diferentes departamentos, con los consiguientes tiempos de espera, la información que se requiere para tomar decisiones.
- La información está almacenada en un único lugar.
- La extracción se realiza de manera sencilla y en tiempo real

Define indicadores que permiten medir el desempeño del Negocio:

- Información de calidad y confiable.
- Reportes Operacionales, Tácticos y Estratégicos.

Provee la capacidad de evaluar distintos escenarios al mismo tiempo:

- Permite analizar diferentes situaciones que pudiesen afectar el negocio.
- Permite adelantar las posibles decisiones estratégicas.
- Permite reaccionar rápidamente a eventos inesperados.
- Permite que una tendencia negativa se convierta en acciones positivas para la organización.
- Permite realiza análisis predictivos

¿DATA WAREHOUSE ES BI?

Tener un Data Warehouse no significa hacer Business Intelligence, así como la ausencia de éste tampoco significa no hacerla.

Es un concepto que **facilita y sirve de base** para las demás técnicas y herramientas del Business Intelligence.

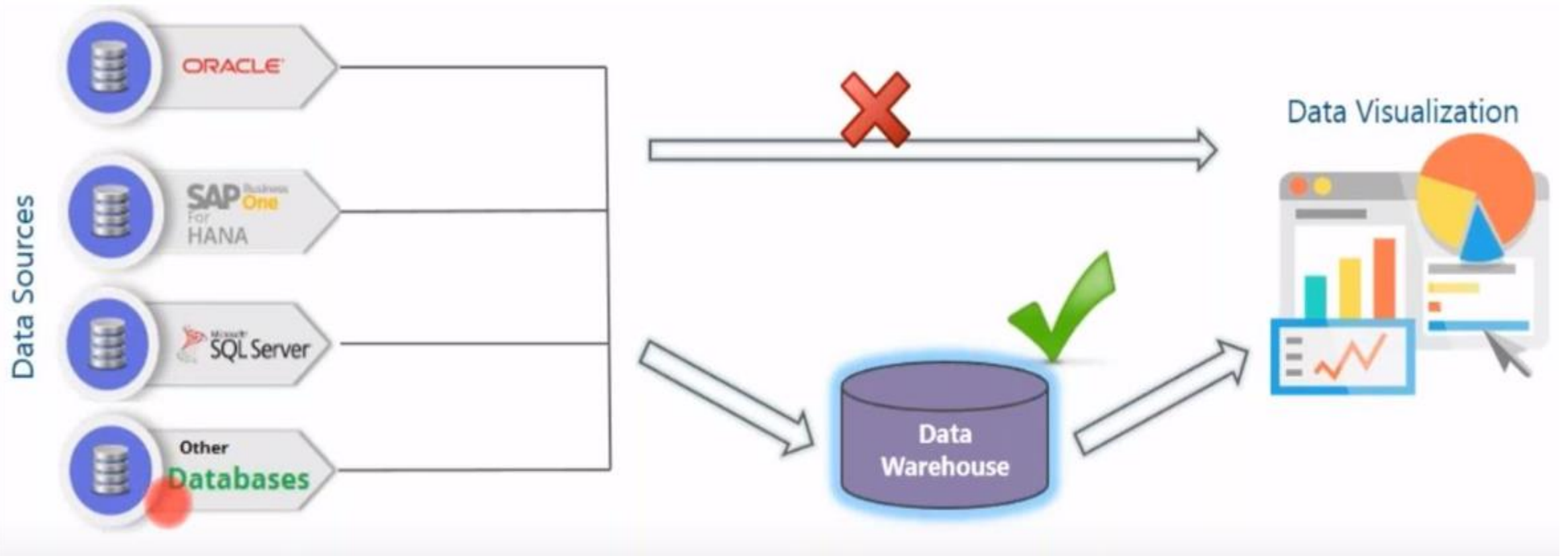
Data Warehouse trata de guardar cantidades masivas de datos para la posteridad. Debe servir para guardar datos por un largo periodo de tiempo y estos datos se deben poder usar para poder encontrar cuestiones interesantes para el negocio.

CARACTERÍSTICAS

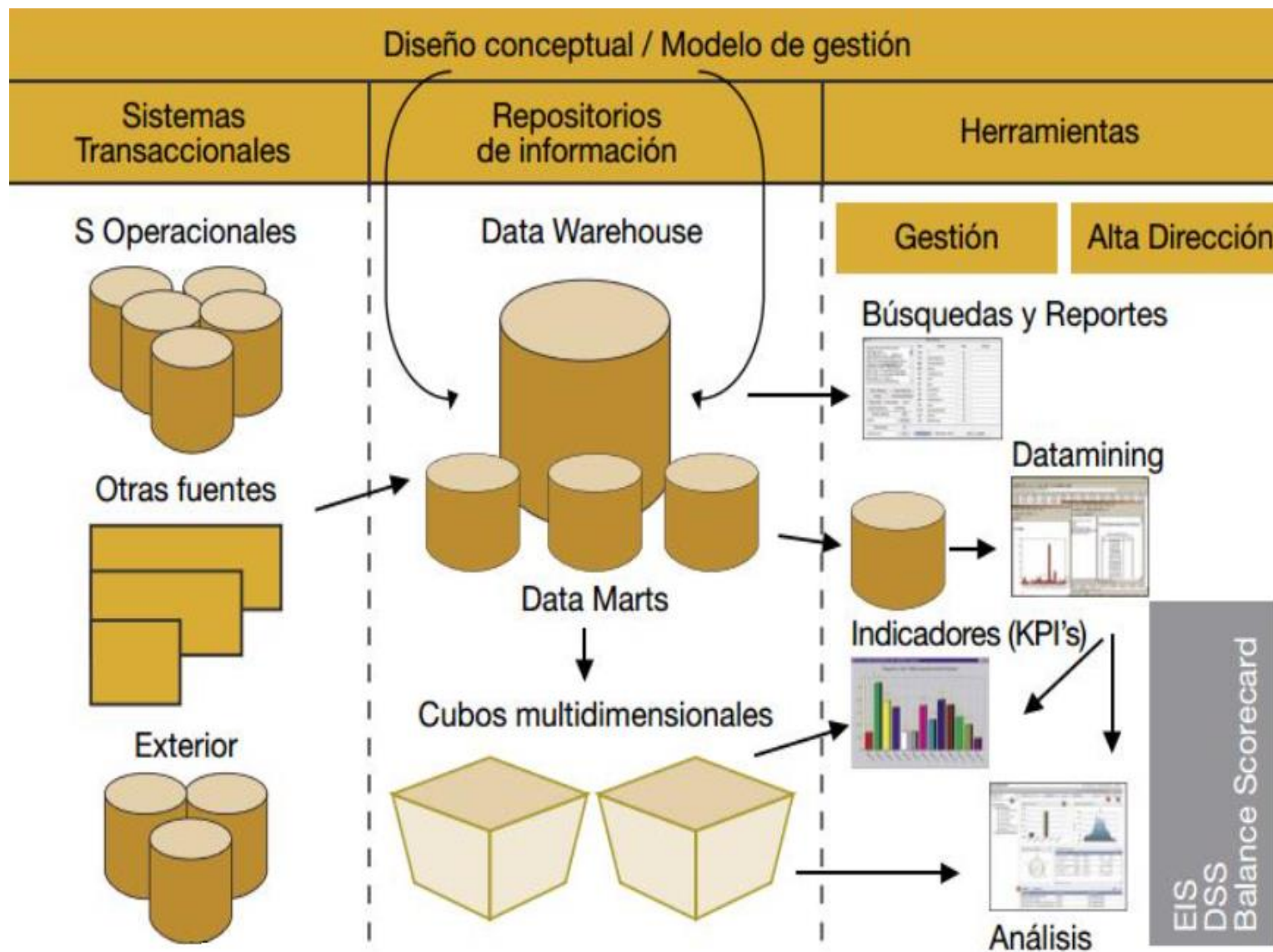
- Orientado a la Información Relevante
- Se diseña pensando en los procesos de la empresa y su cadena de valor: ventas, compras, producción.

Datos Integrados

- Se integran distintas fuentes de información:
 - Sistemas Operacionales
 - Datos inestructurados internos y/o externos
 - Información del mercado.
- Información no Volátil: Los datos almacenados no son actualizados, solo incrementados.
- Variable en el tiempo.
 - Los sistemas transaccionales muestran el estado actual de los hechos.



ARQUITECTURA KIMBALL - Bottom-up



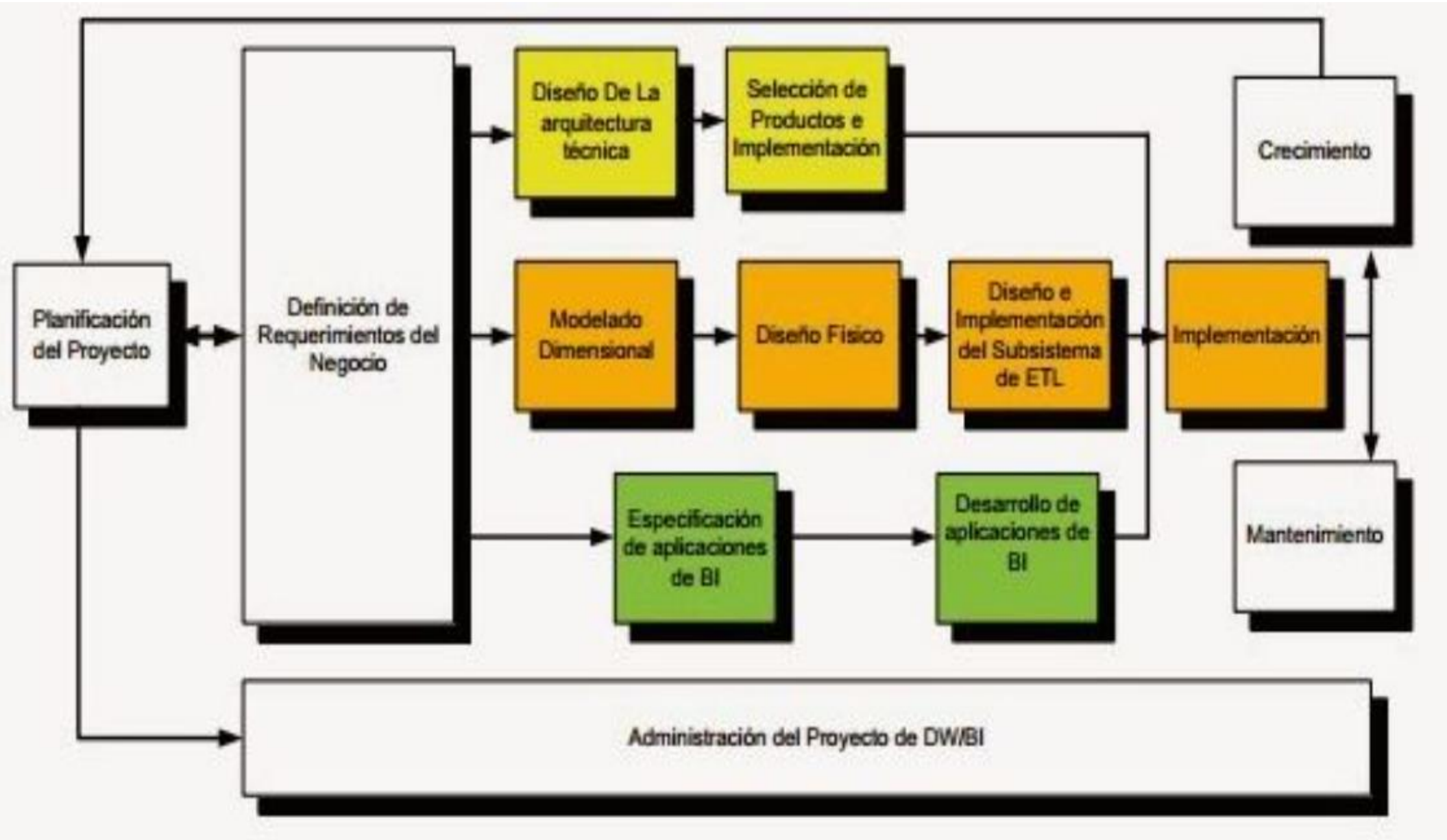
Su filosofía se centra en que, en la mayoría de las organizaciones, la construcción de un DataWarehouse se origina por el interés y esfuerzo de un departamento. Es por esto por lo que en su primera versión este DataWarehouse no es más que un Datamart departamental.

A medida que otros departamentos necesiten sus propios Datamart, éstos se irán combinando con el primero manteniendo una metodología de estandarización mediante lo que Kimball denomina “dimensiones conformadas”, que serán las dimensiones comunes entre los diferentes departamentos. La clave radica en que estas dimensiones han de ser compartidas por los distintos Datamart que existan en la organización, garantizándose así la integridad de los mismos y dando lugar al conglomerado de estructuras que para Kimball conforman el DataWarehouse.

Para lograr este resultado, es importante que estas dimensiones conformadas tengan un diseño consistente y apto para todos los Datamart, de forma que al crearse uno nuevo, reutilice las dimensiones ya definidas, pudiendo incluir o no otras dimensiones nuevas.

La principal ventaja de este enfoque de almacén de datos es que, al estar formado por pequeños Datamart estructurados en modelos de datos dimensionales (esquemas de estrella o copo de nieve), especialmente diseñados para la consulta y generación de informes, el DataWarehouse al completo puede ser explotado directamente por las herramientas de reporting y análisis de datos sin la necesidad de estructuras intermedias.

A este tipo de arquitectura Kimball lo denomina como “Data Warehouse Bus Architecture” y los cuatro pasos fundamentales que se han de seguir para construir este tipo de base de datos son, en primer lugar, la identificación del proceso de negocio, la definición de la granularidad de los datos, la selección de las dimensiones y atributos y, por último, la identificación de los hechos o métricas.



ARQUITECTURA INMON



Un DataWarehouse ha de entenderse como un almacén de datos único y global para toda la empresa.

En este modelo, la premisa es que la información se almacene al máximo nivel de detalle (garantizando la futura exploración de los datos), permaneciendo invariable y no volátil, de manera que los cambios que sufran los datos a lo largo del tiempo queden registrados sin que puedan modificarse o eliminarse.

Claves de la arquitectura:

Estas son las claves fundamentales de la arquitectura defendida por Inmon, conocida como 'Corporate Information Factory (CIF)', donde el DataWarehouse centraliza todos los datos de la compañía para alimentar, a continuación, pequeños Datamart temáticos, que serán los puntos de acceso para las herramientas de reporting. En este sentido, cada departamento tendrá su propio Datamart, abastecido con la información del DataWarehouse, listo para su análisis y explotación.

Propiedades de un Data Warehouse

"A Data Warehouse is a **subject-oriented**, **integrated**, **time-variant** and **nonvolatile** collection of data in support of management's decision-making process." -Bill Inmon, Father of Data Warehousing

Orientado a Temas

No a usos o aplicaciones
específicas

Integrado

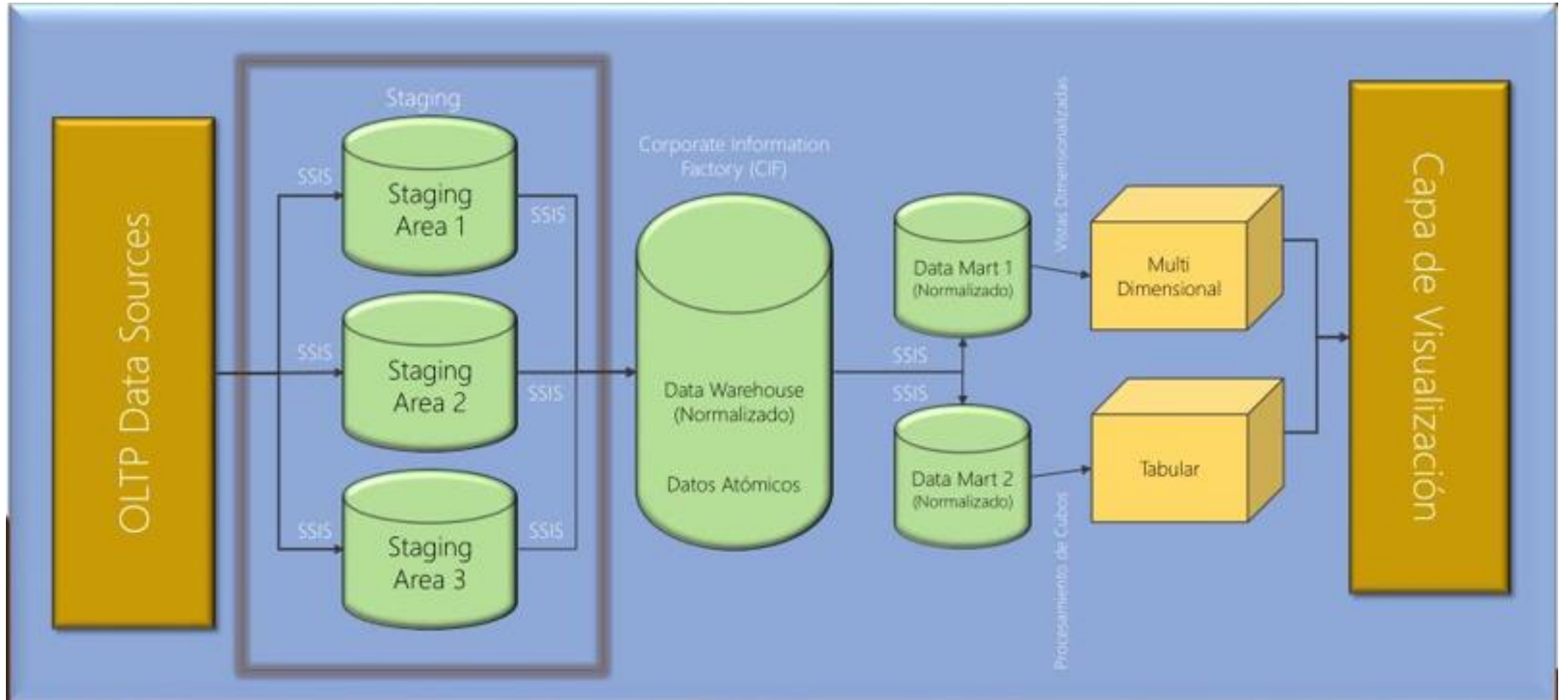
entre distintas fuentes

Varía en el Tiempo

almacena historia

No Volátil

la historia no debería ser
actualizada o borrada





Este enfoque de Inmon suele denominarse como una metodología de trabajo ‘Top-Down’, ya que se centra primero en una visión global de la compañía, para ir desmembrándola en pequeños sets de datos departamentales. Así, con esta arquitectura, todos los Datamart de la organización están conectados al DataWarehouse, evitándose la aparición de incongruencias y anomalías al comparar los datos entre distintos departamentos.

En cuanto a la estructura interna del DataWarehouse, para Inmon la prioridad es que el modelo de datos esté construido en tercera forma normal. Por dar una breve explicación de lo que esto significa, el proceso de normalización consiste en aplicar una serie de reglas o normas a la hora de establecer las relaciones entre los diferentes objetos dentro de la base de datos.

Con este proceso de normalización se consiguen muchos beneficios, como evitar la redundancia de los datos, mantener su integridad referencial, facilitar el mantenimiento de las tablas y disminuir el tamaño de la base de datos. Sin embargo, a diferencia de los DataWarehouse desnormalizados, las consultas exigen el empleo de queries mucho más complejas, lo que dificulta el análisis directo de la información y el uso de las herramientas de reporting. De ahí, la necesidad de construir los Datamart que, como ya comenté, están basados en modelos dimensionales de estrella o copo de nieve, diseños fácilmente explotables por estas herramientas de análisis de datos.

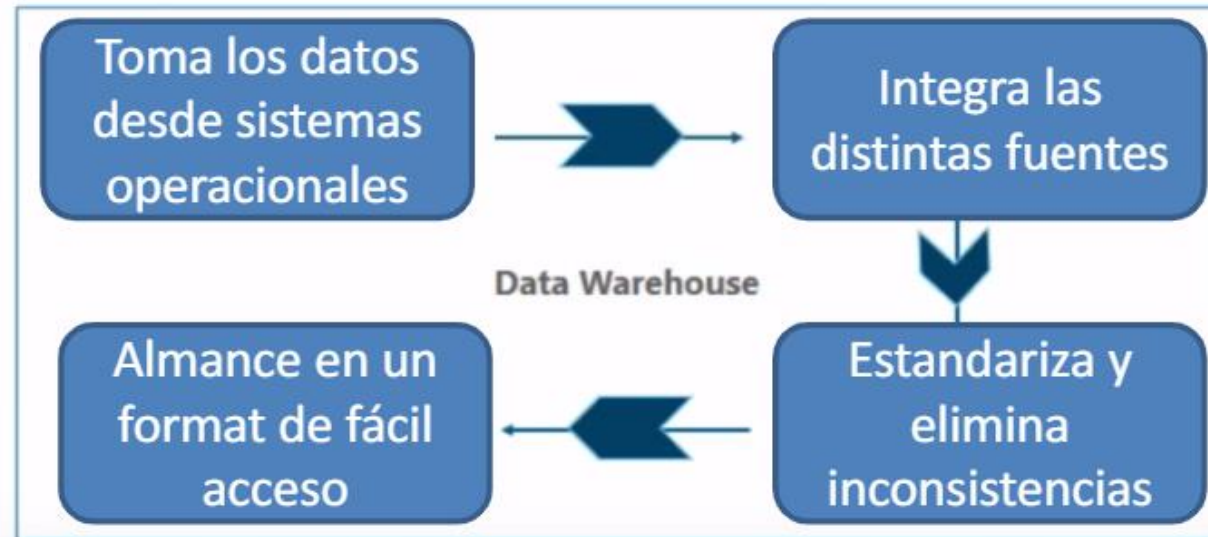
Principales ventajas

Permite responder preguntas estratégicas, estudiar tendencias e incluso realizar predicciones.

Acceso simple y rápido a los datos.



Query
Result



	Inmon	Kimball
Presupuesto	Coste inicial alto	Coste inicial bajo
Plazos	Requiere más tiempo de desarrollo	Tiempo de desarrollo inferior
Expertise	Equipo con especialización alta	Equipo con especialización media
Alcance	Toda la compañía	Departamentos individuales
Mantenimiento	Fácil mantenimiento	Mantenimiento más complejo

Data Warehouse vs. Data Mart

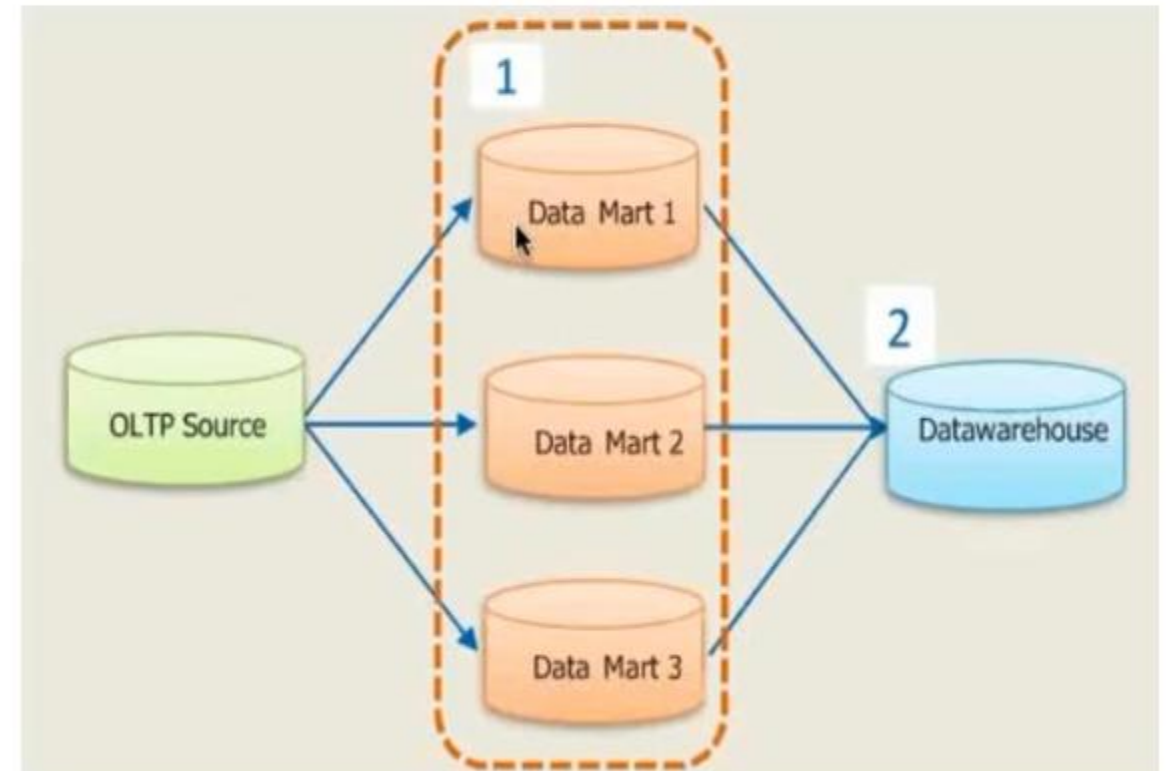
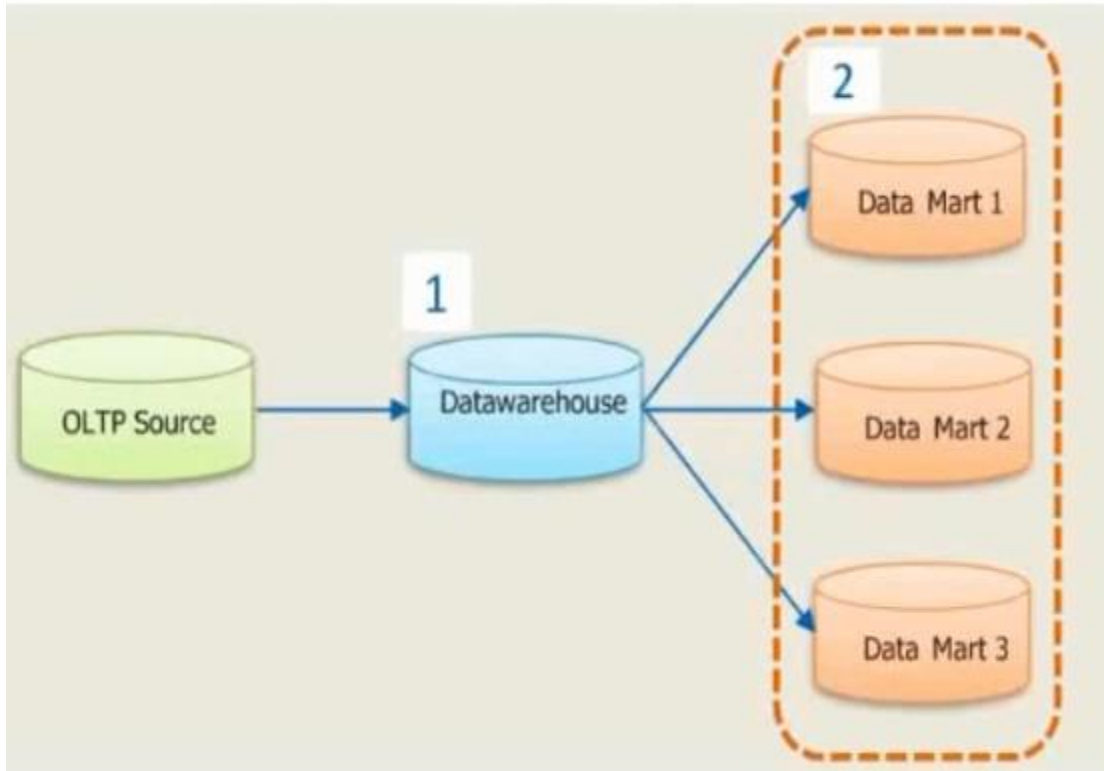
	DATA WAREHOUSE	DATA MART
ALCANCE	Independiente de las aplicaciones que lo utilicen	Dependiente de alguna aplicación específica
DATA	Histórica y detallada	Puede o no ser detallada y además estar o no resumida
TEMAS	Múltiples	Acotados o únicos
FUENTES	Varias	Pocas
ESTRUCTURA	Una única y compleja	Varias estructuras más simples

ETL: Extract, Transform & Load

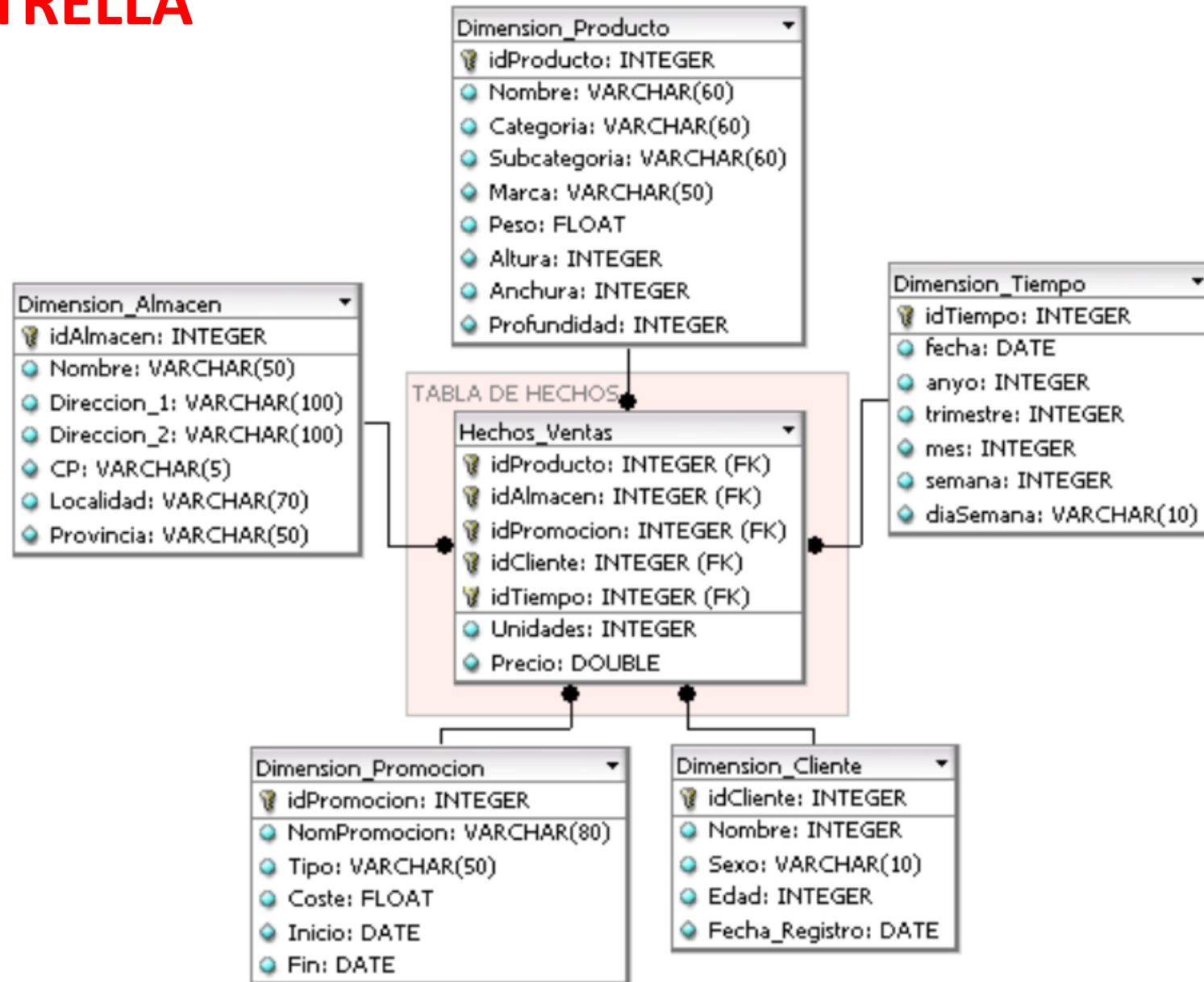
Es el proceso de extraer datos desde varias fuentes, transformarlos a un estado requerido y luego almacenarlos en un data warehouse.



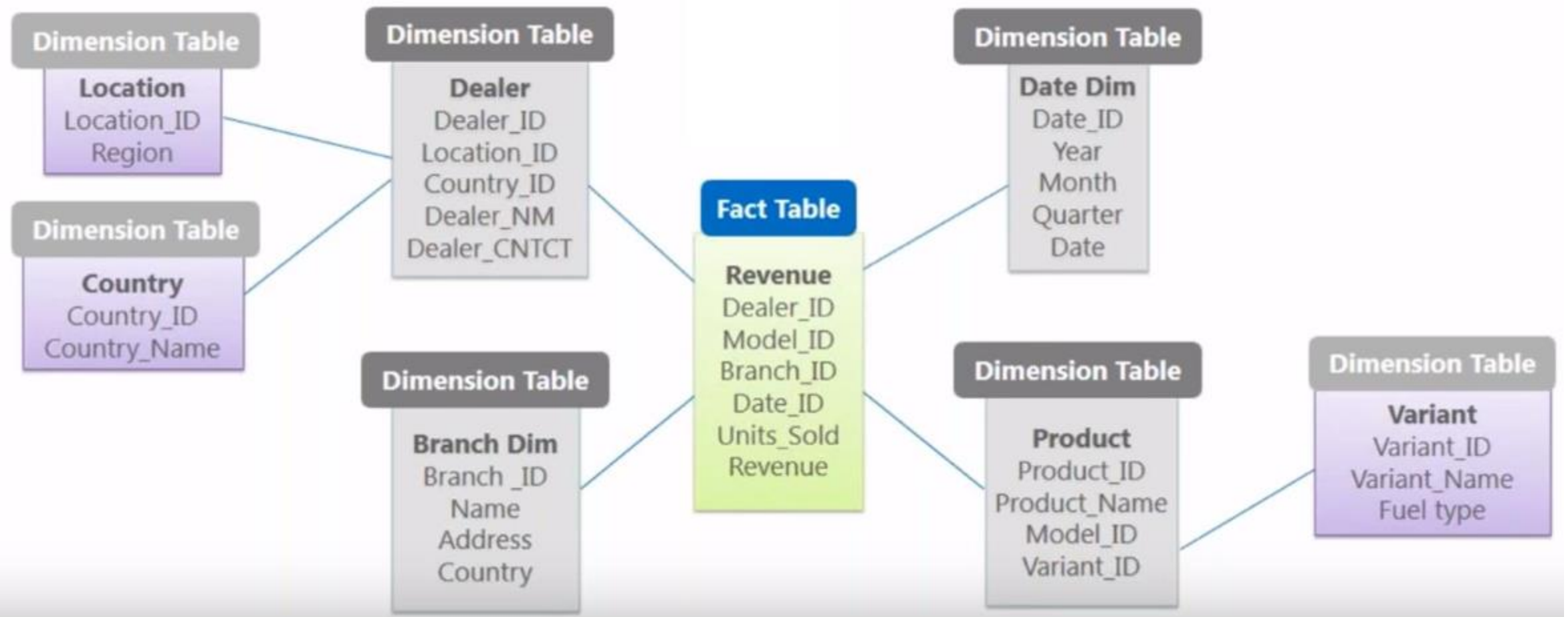
Top/Down vs Bottom/Up



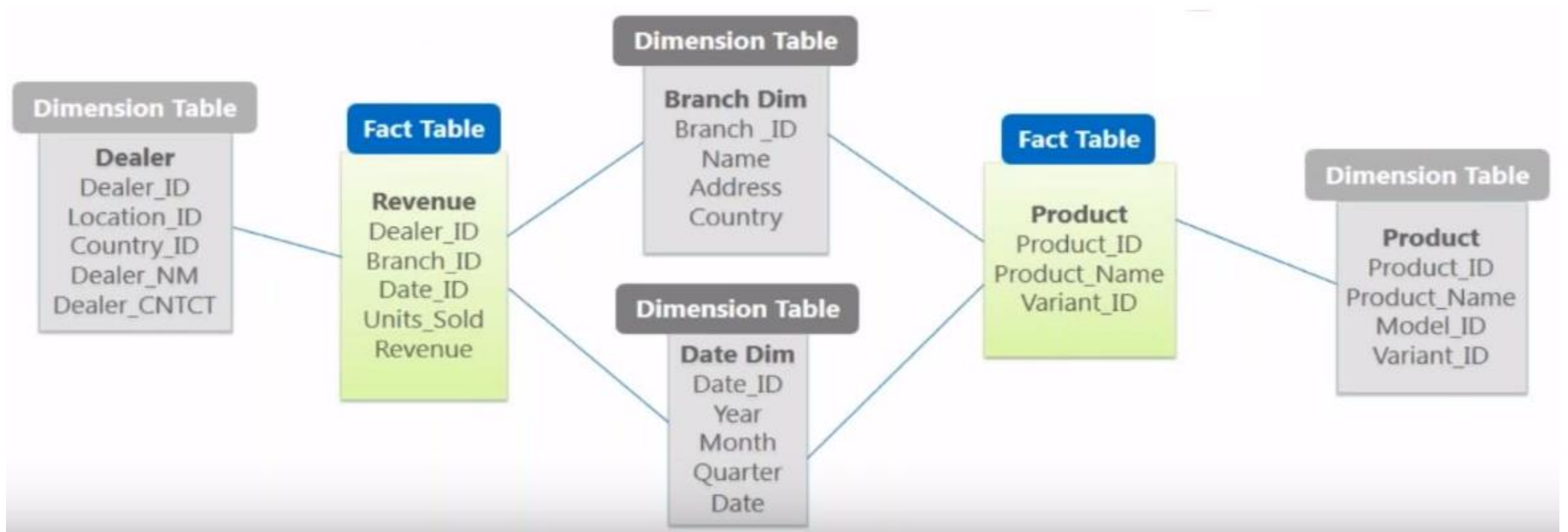
MODELO ESTRELLA



MODELO COPO DE NIEVE

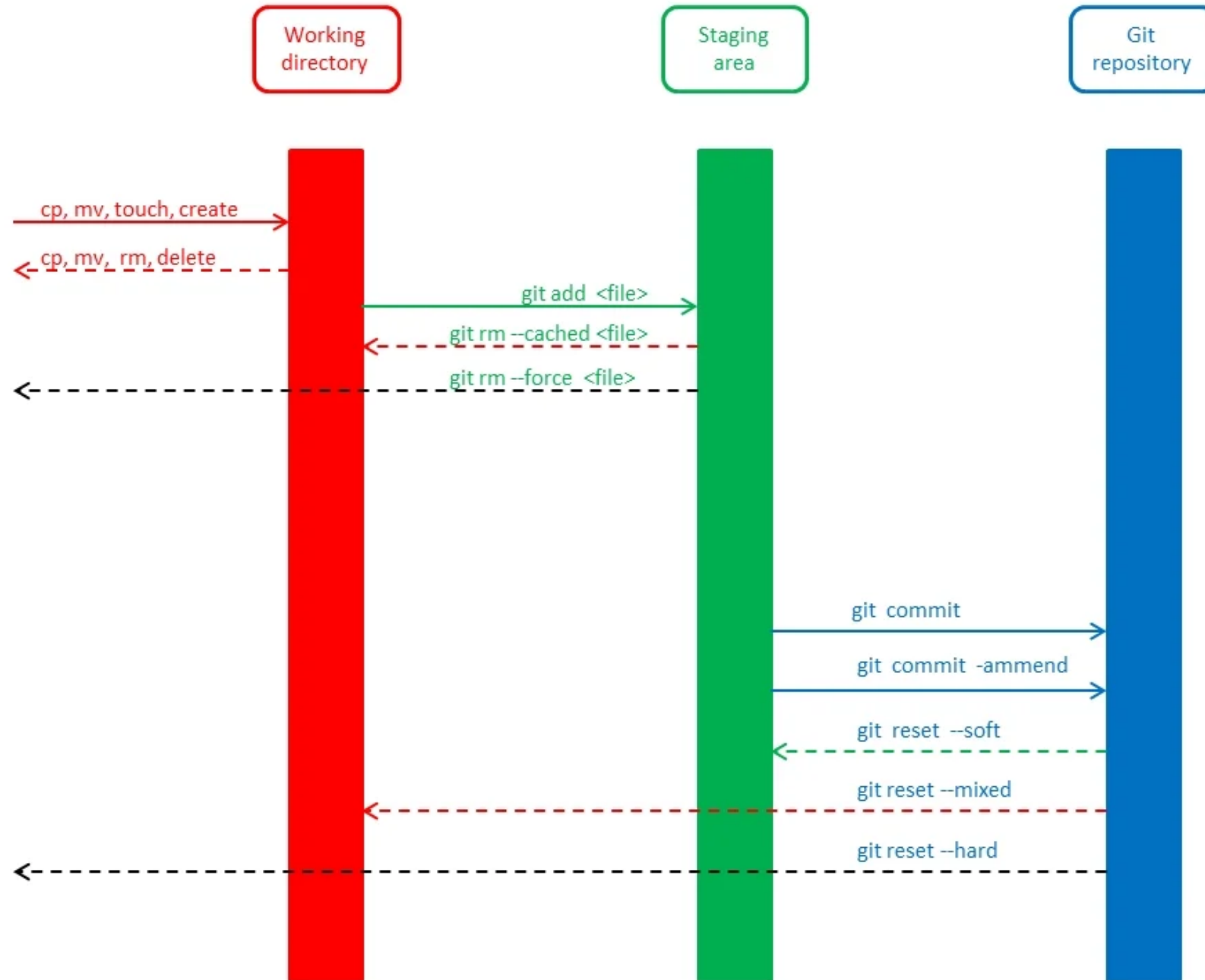


MODELO CONSTELACION / GALAXIA



- ¿ Qué es una tabla de dimensión ?
 - Tabla que corresponde a un objeto o concepto del mundo real
 - Ejemplo: consumidor, producto, día, empleados, regiones, tiendas, promociones, vendedores, proveedores, etc.
- Propiedades
 - Contienen varias columnas descriptivas
 - En general son tablas anchas (docenas de columnas)
 - Generalmente no tienen muchas filas
 - Al menos en comparación con las tablas de hechos
 - Usualmente < 1 millón de filas
 - Relativamente estáticas

- ¿ Qué es una tabla de hechos ?
 - Tabla que contiene mediciones acerca de un evento en un proceso de interés. Ej: venta, insumo, etc.
- Cada fila contiene 2 tipos de datos:
 - Columnas con valores numéricos o mediciones
 - Llaves a tablas de dimensiones
- Propiedades
 - Gigantes: A menudo millones o billones de filas
 - Angostas: A menudo pocas columnas
 - Cambian frecuentemente
 - Nuevos eventos en el mundo producen nuevas filas en la tabla
 - Típicamente las nuevas filas son sólo agregadas (no hay un ordenamiento especial)



git init

Entorno de desarrollo personal

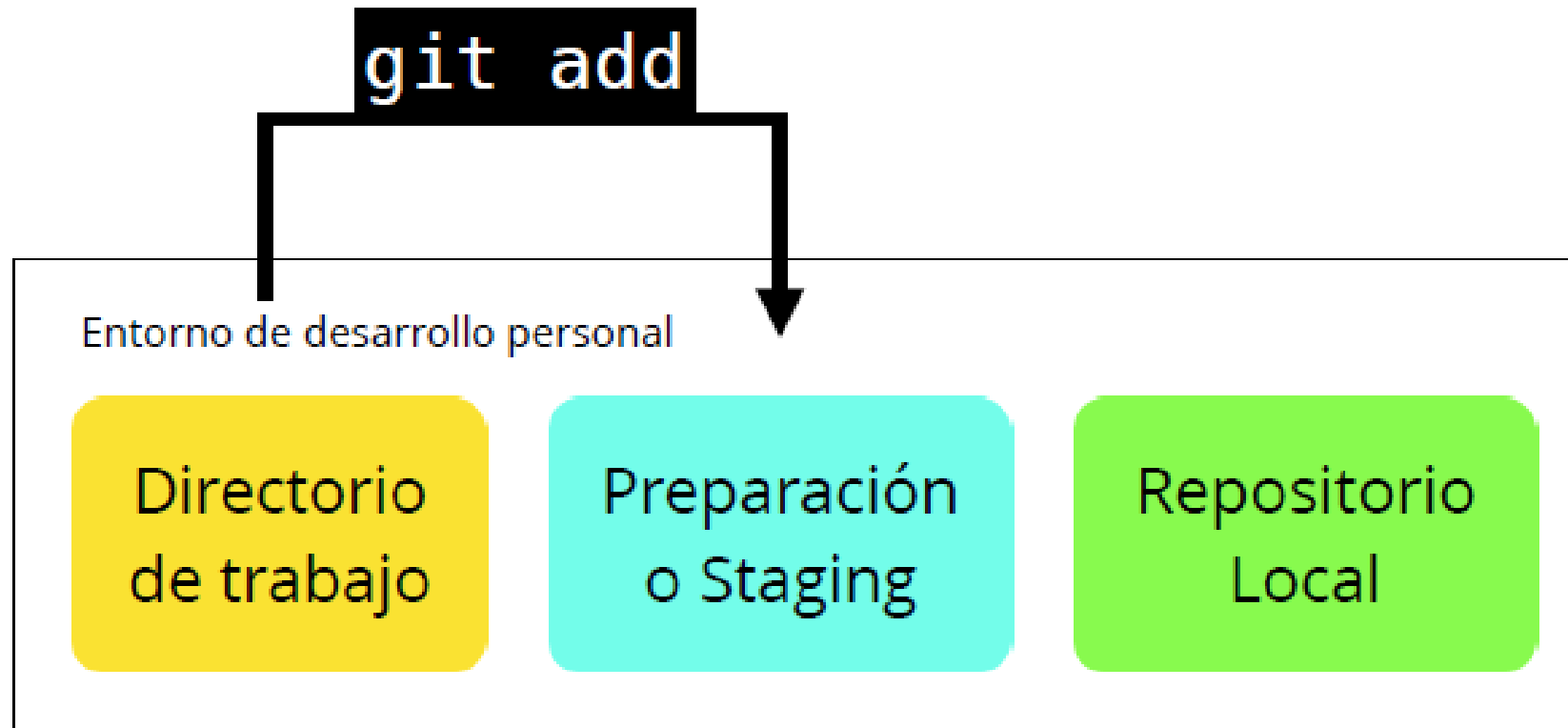
Directorio
de trabajo

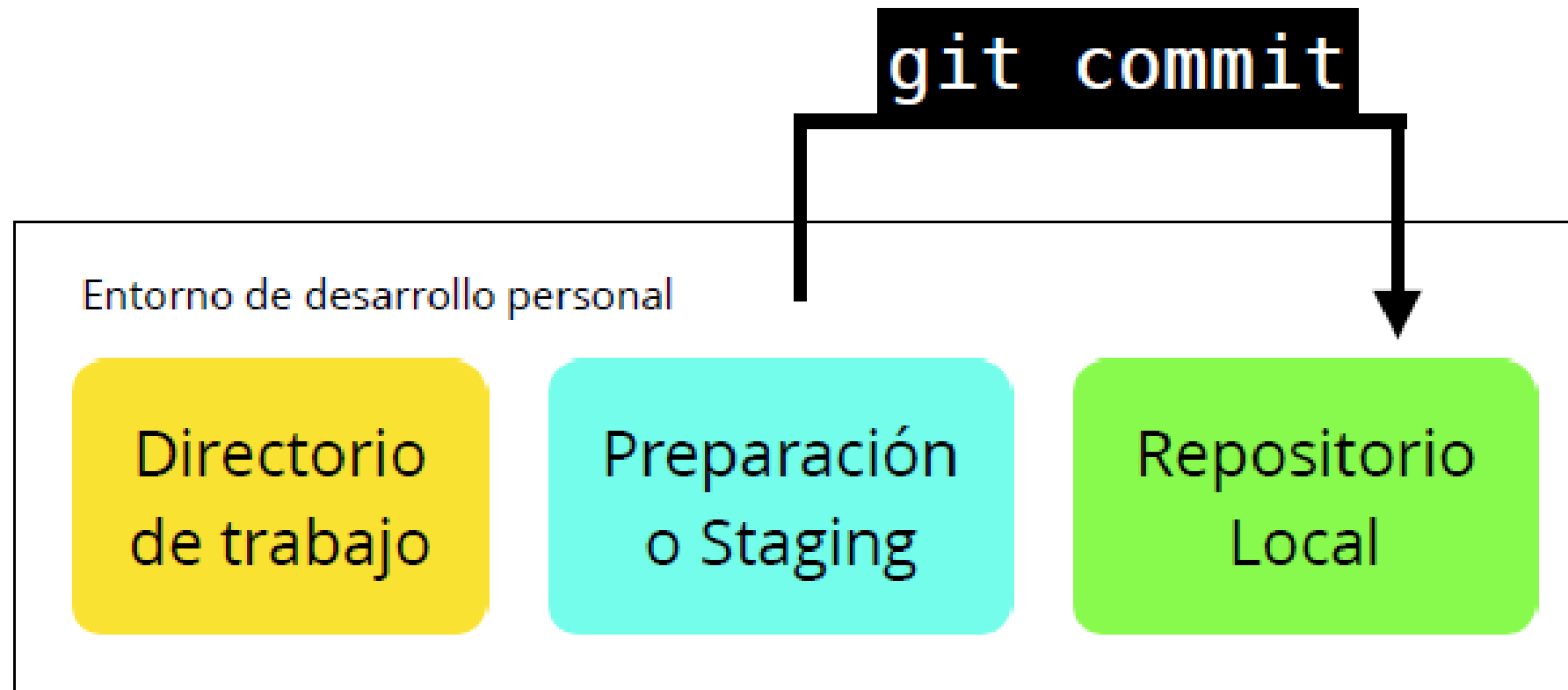
Entorno de desarrollo personal

Directorio
de trabajo

Preparación
o Staging

Repositorio
Local





Servidor remoto

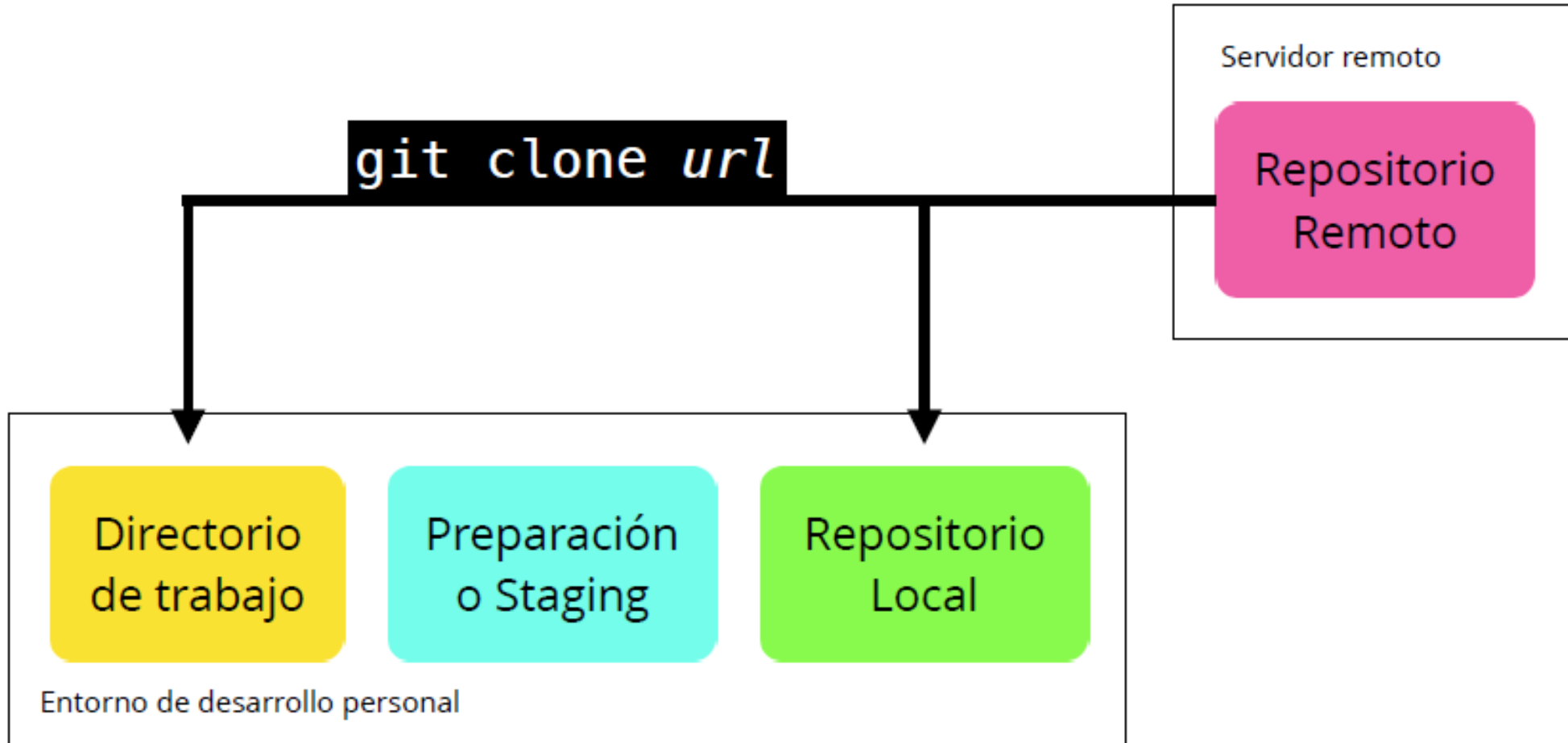
Repositorio
Remoto

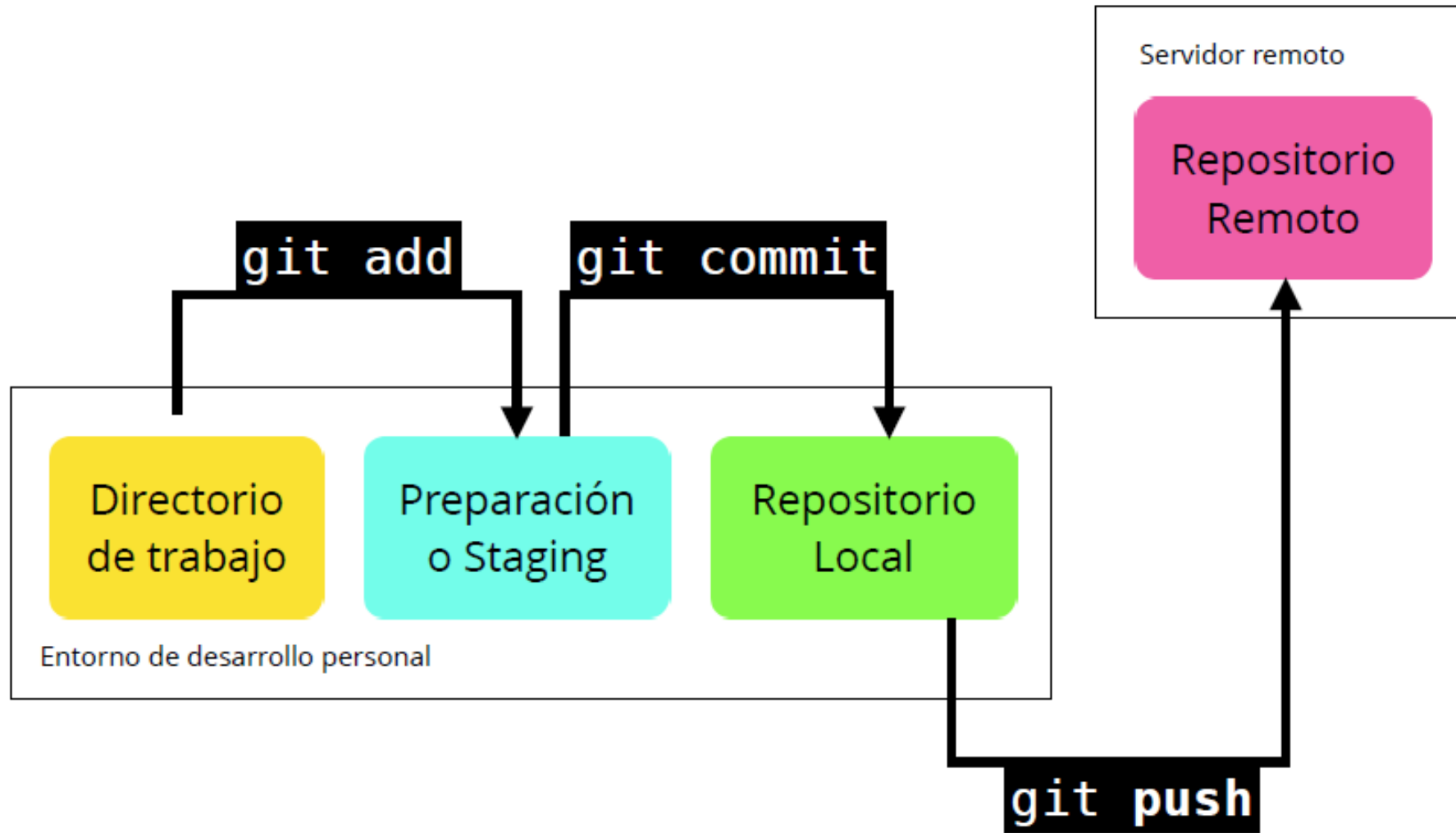
Entorno de desarrollo personal

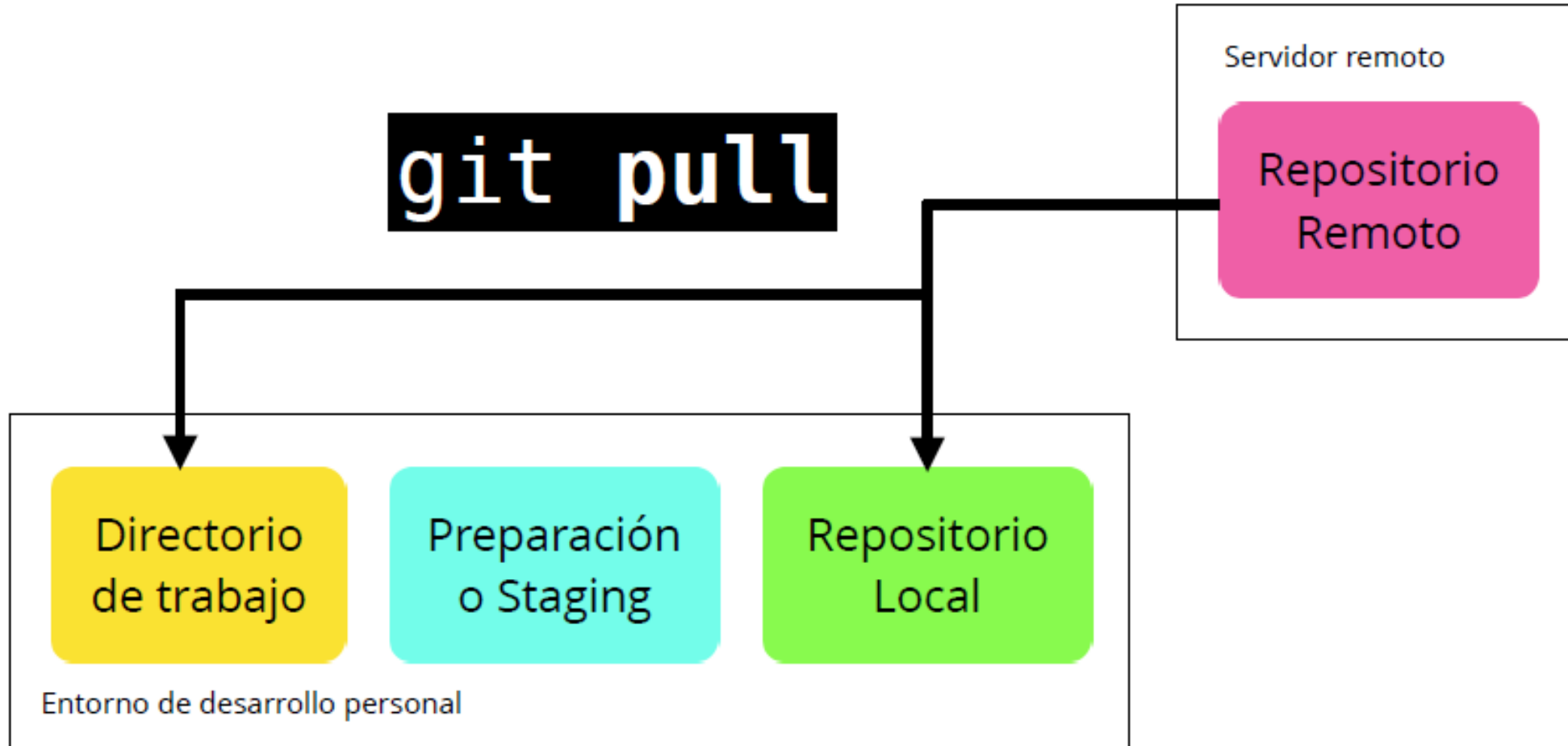
Directorio
de trabajo

Preparación
o Staging

Repositorio
Local







APUNTES CURSO GIT-GITHUB (REPOSITORIO REMOTO)

\$git remote add origin <https-url>	Establecer un origen remoto: sede del repositorio remoto para gestionar nuestro proyecto mediante conexión HTTPS
\$git remote -v	Verifica la existencia del origen remoto
\$git config -l	Permite ver los parámetros de configuración
\$git push origin master	Fusiona una copia del master local con el remoto
\$git pull origin master	Fusiona una copia del master remoto con el local
\$git config -l valores	Permite ver los parámetros de la configuración y sus valores
\$git config --global user.email "email"	Permite modificar el valor de la variable user.email en la configuración
\$git remote set-url origin <ssh-url> través de SSH (en lugar de HTTPS)	Configura git para conectar con el repositorio remoto a través de SSH (en lugar de HTTPS)

SSH: GENERACIÓN DE CLAVES PÚBLICA Y PRIVADA

\$ssh-keygen -t rsa -b 4096 -C "email"	Generación de claves de cifrado pública y privada
\$eval \$(ssh-agent -s)	Comprueba si el servicio de cifrado está activo
\$ssh-add <ruta-id_rsa> contiene la llave privada	Informa al sistema de la ubicación del archivo que contiene la llave privada

Posteriormente, hay que proporcionar a GitHub una copia de nuestra llave pública:
En la ruta: **Personal Settings/SSH and GPG keys/SSH keys/** hay que copiar el contenido del archivo **id_rsa.pub** que se generó con el comando ssh-keygen.



MUCHAS GRACIAS

Certified



Corporation[®]

