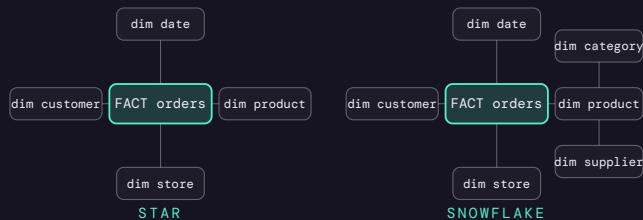


Star schema vs snowflake, in one image.

D8A.ACADEMY/RESOURCES/STAR-VS-SNOWFLAKE

THE TWO SHAPES



SIDE BY SIDE

	STAR	SNOWFLAKE
dimensions	denormalized, flat	normalized, split
joins per query	fewer	more
storage	a bit more	a bit less
BI tool fit	excellent	clunkier
maintenance	simpler	stricter

PICK STAR WHEN

BI dashboards	fewer joins, faster reads
small team	simpler to reason about
cloud warehouse	storage is cheap anyway

PICK SNOWFLAKE WHEN

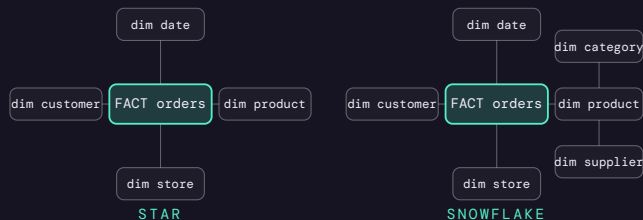
shared hierarchies	geo or product trees reused
strict consistency	update one place only
storage-bound	rare in the cloud era

→ default to star; snowflake only with a reason you can say out loud

WORTH MEMORIZING

01 The two shapes

One fact table in the middle. Dimensions stay flat, or keep splitting.



→ Fact = events and measures (orders, clicks). Dimensions = context (who, what, where, when).

02 Trap: snowflaking for storage

You save cents on storage and pay in joins on every query, forever.

the saving	normalized dims: a few cents per GB
the bill	extra joins in every single dashboard query
the rule	in cloud warehouses, flatten. Star wins by default.

→ Normalization is for OLTP systems. Warehouses trade redundancy for read speed.

03 The vocabulary

Four words that carry every modeling interview.

fact table	events and measures: orders, sessions, payments
dimension	context: customer, product, date, store
grain	what one fact row means. Decide it first.
surrogate key	warehouse-made id joining facts to dims