



CLOUD ANALYTICS CONFERENCE

LONDON



[@SnowflakeDB](https://twitter.com/SnowflakeDB) [#CloudAnalytics17](https://twitter.com/CloudAnalytics17)

Enabling the Agile Data Warehouse

Steve Herskovitz

VP Sales Engineering, Snowflake Computing



Enabling the Agile Data Warehouse - Agenda

- Agile Warehouse Scaling
 - Separation of Workloads
 - Virtual WH Scaling Techniques
- Agile Data Lifecycle
 - Cloning
- Agile Data Analytics
 - Time Travel
- Real Customer Story
 - GTA – Gulliver's Travel Associates



Agile Warehouse Scaling Separation of Workloads



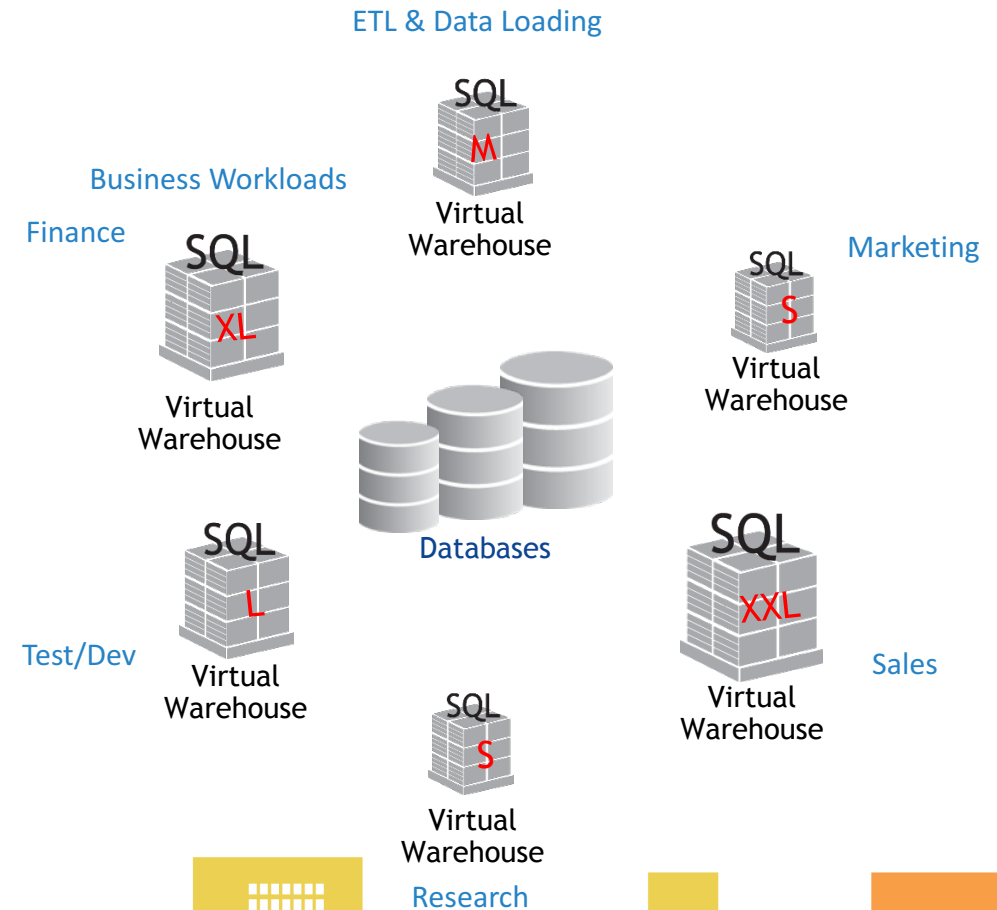
Separation of Workloads

- Struggle – multiple workloads sharing a fixed resource
- Overnight batch ETL
 - ETL must complete before business workloads start
 - Planned or unexpected data surges can cause ETL to run late
 - Worse yet, overnight in US is exactly the UK business day
 - ETL and business workloads impact each other's SLAs
- Competing business workloads
 - Sales, Marketing, Finance, Data Science
- Conventional solutions
 - Divide fixed resource into timeslots for each workload
 - Complex workload prioritization schemes
 - Periodic or seasonal surges handled by locking out some users



Separation of Workloads

- Struggle – multiple workloads sharing a fixed resource
- Snowflake solution
 - Assign each workload its own Virtual WH
 - At a minimum, two WH: ETL and Business
 - ETL can run continuously if it makes sense for the business
 - ETL / Business workload contention is eliminated
 - Further subdivide business workloads into own clusters as needed
 - Eliminate contention between Sales, Marketing, Finance, Data Science workloads
 - International groups can operate clusters on their own local time schedules
 - Permits departmental chargebacks



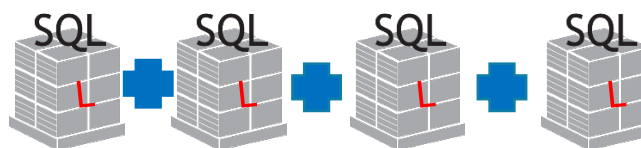
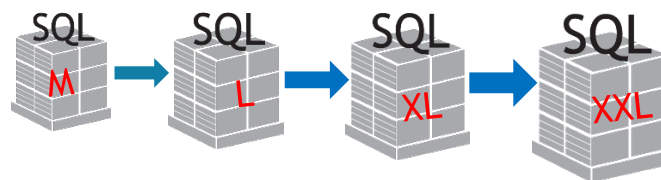
CLOUD ANALYTICS
CONFERENCE

Agile Warehouse Scaling Techniques



Warehouse Scaling Techniques

- Increase T-shirt size
 - More data being analyzed
 - More complex queries
 - Get some concurrency boost
- Multi-cluster WH is best for concurrency
 - Workload queries have usual weight
 - But more of them, e.g. 20 dashboard users rather than the usual 5
- Combine these two techniques for best effect



MCWH Scaling Algorithms – Scaling Up

- Scheduler checks if it should spin up another cluster
 - Queries must queue for > 30 seconds
 - Spinning up cluster is often immediate (for XXL or smaller)
 - Queries begin to get load-balanced across new clusters
 - One-minute rule: 60 seconds of load balancing before next queuing interval
- Repeat up to maximum clusters configured for warehouse
- Designed to
 - Balance responsiveness against cost
 - Ensure SLAs



MCWH Scaling Algorithms – Load Balancing

- Scheduler checks for clusters to distribute queries
 - Cluster is active (not quiesced)
 - Cluster has latest version of software (in case system update in progress)
 - Cluster has head-room for more queries
 - Cluster is the least busy
 - Session affinity breaks any ties (for cache)
- Designed to maximize
 - Individual query performance
 - Overall throughput
 - Overall concurrency



MCWH Scaling Algorithms – Scaling Down

- Scheduler checks if it can spin down a cluster
 - One-minute rule: 60 seconds of load balancing before check if WH underloaded
 - Check if WH with one less cluster could have handled the load over 15 minutes
 - Quiesce cluster: finish current queries but accept no new queries
 - Wait another 15 minutes before checking if can quiesce another cluster
- Repeat down to minimum clusters configured for warehouse
- Designed to
 - Maximize the value of the running clusters
 - Minimize the cost of the MCWH



Warehouse Scaling – Best Practices

- Anticipated surges
 - Explicitly increase WH nodes (T-shirt size) when expecting more data
 - Explicitly increase MCWH minimum clusters when expecting more queries
 - Can do both at once with ALTER WAREHOUSE
 - Use cron or other scheduling/orchestration tool
- Unanticipated surges
 - Rely on MCWH maximum clusters for some extra headroom
- Maximize
 - Responsiveness for users
 - Throughput and value extracted from variable compute power
- Minimize
 - Cost and administrative overhead



Agile Data Lifecycle



Agile Data Lifecycle

- Separation of Workloads
 - Individual virtual warehouse for each dev/test/prod functional area
- CLONE for dev/test
 - Full logical copy of the data, but uses no extra storage
 - Test/dev operations against clone have no effect on original data
 - Security
 - RBAC limits dev/test access to clone and not production data
 - Secure Views permit role- or user-based obfuscation / masking / projection
 - Clone to TRANSIENT reduces storage usage by dev/test operations
 - TRANSIENT tables can have retention period set to 0 days if time-travel is not part of your app
- Business Impact – better quality code
 - Dev and test teams are working on data at scale, see true app performance
 - Full range of values means fewer surprises when app encounters live data



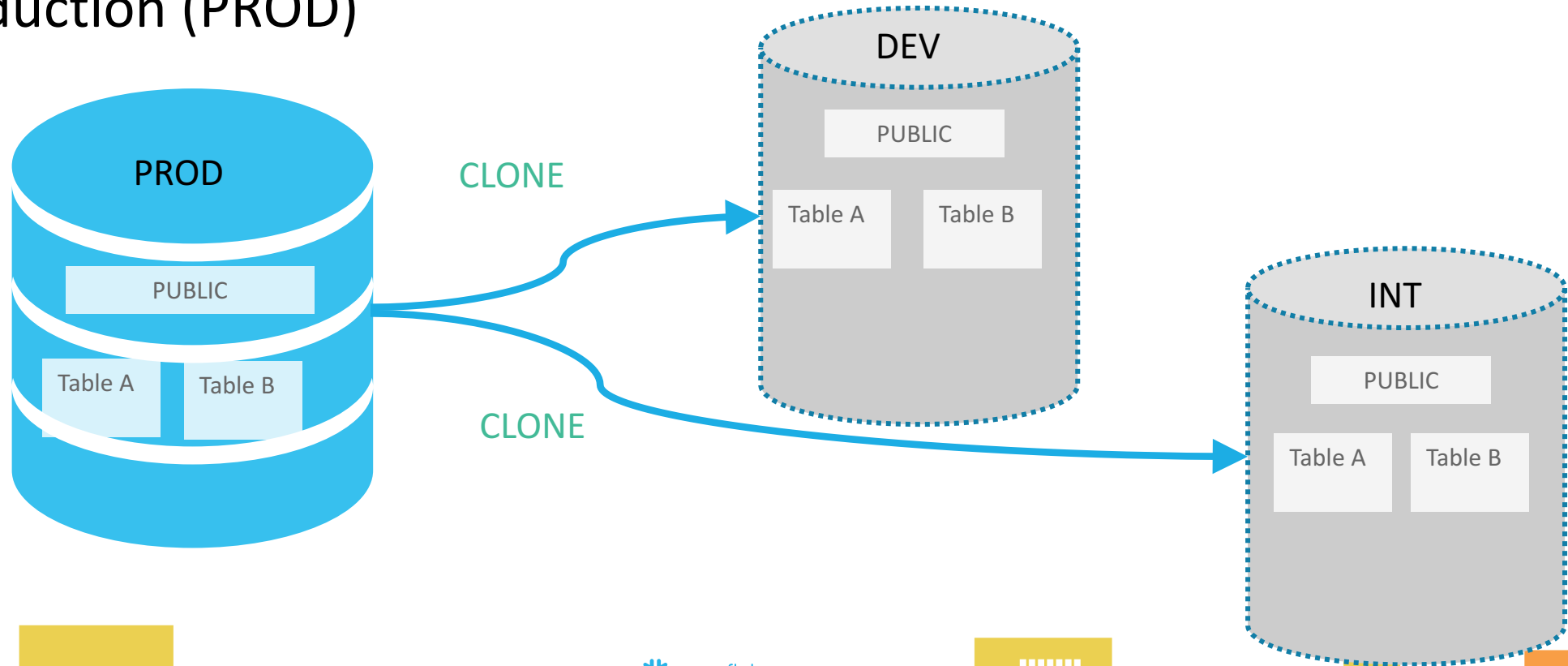
CLOUD ANALYTICS
CONFERENCE

Demo



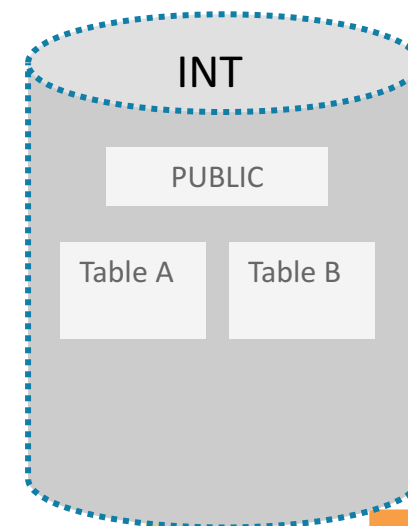
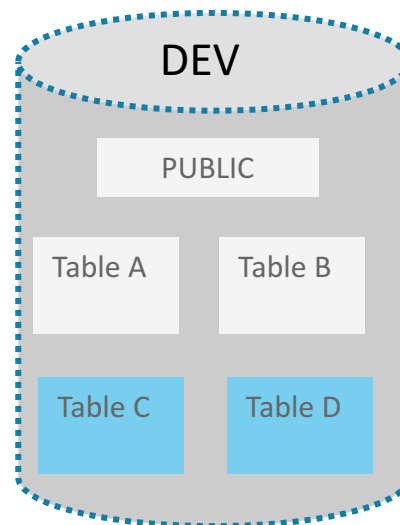
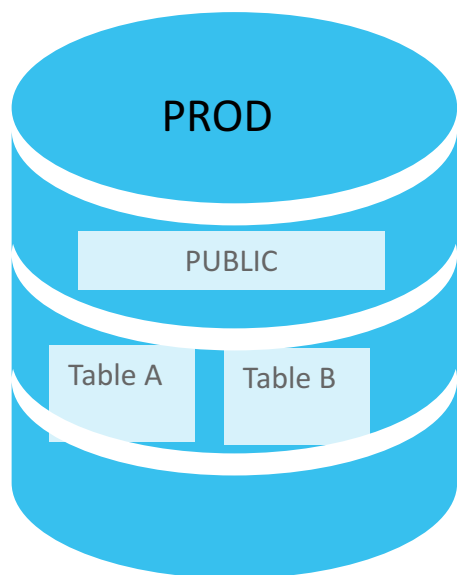
Scenario 1

- Create development (DEV) and integration (INT) databases from production (PROD)



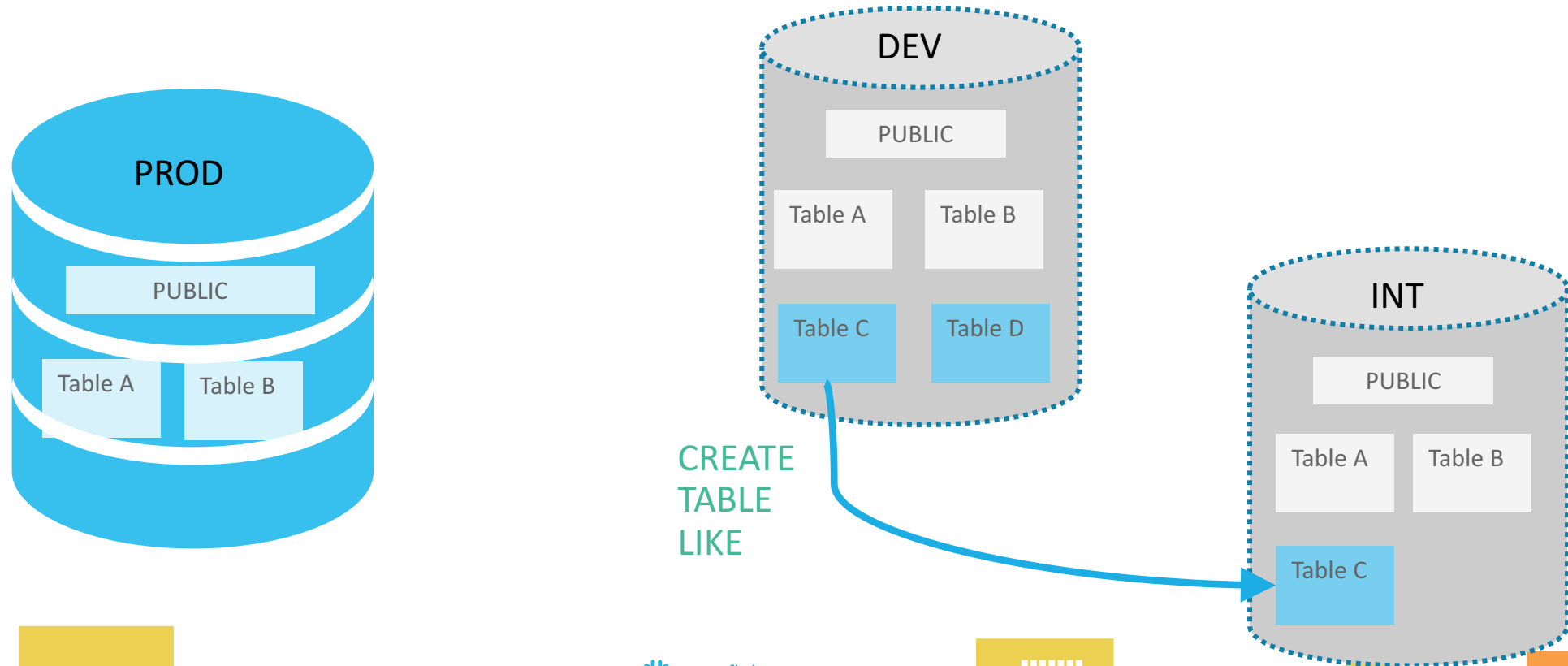
Scenario 2: new development

- Create two new tables, C and D, in the development (DEV) database



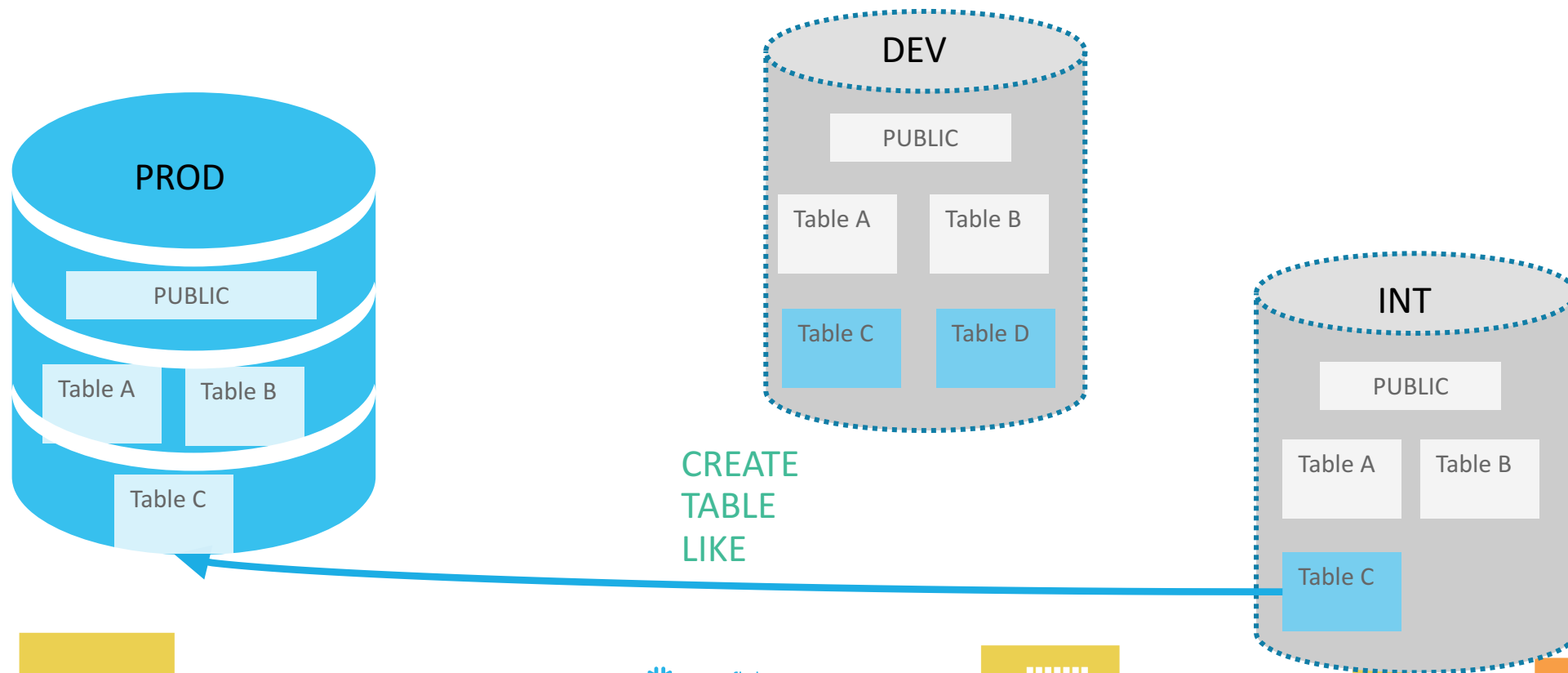
Scenario 2: new development

- Mini-release: promote table C for integration testing



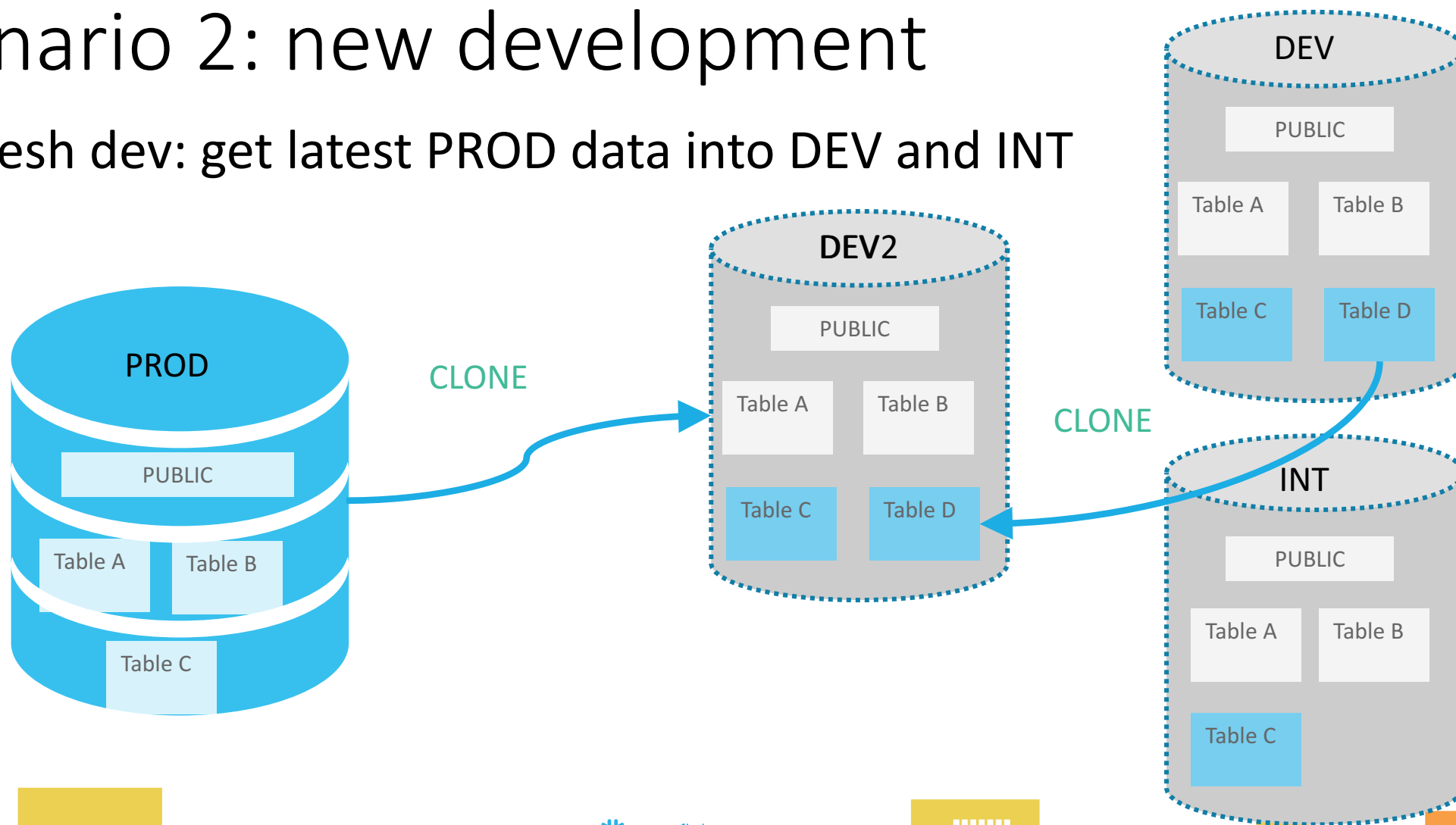
Scenario 2: new development

- Deploy to production: promote table C to PROD database



Scenario 2: new development

- Refresh dev: get latest PROD data into DEV and INT



Agile Data Lifecycle

- CLONE for Data Scientists
 - Quick and Safe sandbox for discovery and testing
 - Combine with own virtual warehouse for complete isolation
 - Business Impact – better data science
 - More fine-grained data over longer time intervals
 - Deeper insights, better forecasting, more monetizable results
- CLONE for Compliance
 - Monthly, quarterly, annual clones – financial reporting, auditing requirements
 - Business Impact – simpler compliance
 - Your "backups" are live and immediately available



Time Travel



Agile Data Analytics

- CLONE operates on metadata repo
 - Table micro-partitions are tracked by pointers in metadata repo
 - Cloning copies pointers only, not the micro-partitions
- Time Travel leverages metadata pointers
 - Pointer has millisecond-granularity timestamp
 - Snowflake knows which micro-partitions are active in your table at any moment



Agile Data Analytics

- Data Retention for Time Travel
 - Default is 24 hours, maximum 90 days
 - Configurable per-table by table owner
 - Uses more storage because keeps the micro-partitions around longer
- Simple SQL syntax
 - `SELECT cols... FROM t1 AT (TIMESTAMP => timestamp);`
 - `CREATE obj2 CLONE obj1 BEFORE (STATEMENT => query-id);`



Agile Data Analytics

- `SELECT count(*)
FROM lineitem AT(TIMESTAMP => '2020-01-01 12:00:00'::timestamp);`
- `SELECT
(SELECT count(*) FROM lineitem AT(OFFSET => -60*2)) before_etl,
(SELECT count(*) FROM lineitem) after_etl;`



Demo



Agile Data Analytics

- Business Impact of Time Travel
 - UNDROP — table, schema, database
 - Un-TRUNCATE table (CLONE table, then swap names)
 - Recover from ETL/ELT update (CLONE database, then swap names)
 - Temporal queries
 - For example, what was inventory on a given date?
 - Type 2 slowly changing dimensions
 - up to 90-day running window
 - Fast prototype for longer-window Type 2 analytics
 - Test predictive models against historical data
 - Don't need to make and store daily backups



CLOUD ANALYTICS
CONFERENCE

Agile Data Warehouse – Summary

- Separation of Workloads
 - Individual virtual warehouse for each dev/test/prod functional area
- Virtual Warehouse scaling
 - T-shirt sizes, number of WHs, and MCWH
- CLONE for dev/test and other uses cases
 - Full logical copy of the data, but uses no extra storage
- Time Travel and CDP
 - SELECT "as of" for testing of predictive models, type 2 changing dimensions
 - Easy "undo" of updates – UNDROP, un-TRUNCATE, CLONE "as of"



Customer Story – GTA



Cast Study: Enabling the Agile Data Warehouse with Snowflake

Adam Slader

adam.slader@gta-travel.com



Data Warehousing projects...

Complex
Long = **Risky**
Change

- **How Snowflake helped GTA be more Agile.**
 1. **Focus on Value:** Choosing the right tools
 2. **Our plumbing:** Pipeline transparency
 3. **Agile Development:** Iteration, prototyping & testing
 4. **Scaling:** start small, remain flexible

1. Choosing your tools

- What would let us focus on adding value quickest?
- What is going to give highest productivity?
- What is lowest risk – but still future proof?



1. Our new stack...

.



Golden Gate

- Don't re-invent the wheel



EC2 + S3

- Standard, proven, & skills availability
- Lower dependency on IT



Airflow

- Python, open source



Snowflake

- SQL
- On-demand – get going quickly
- Zero admin – we lost our DBA
- **Managed platform**

2. Plumbing

Captain Obvious is here to say something *obvious*...

Extract, Load, then Transform

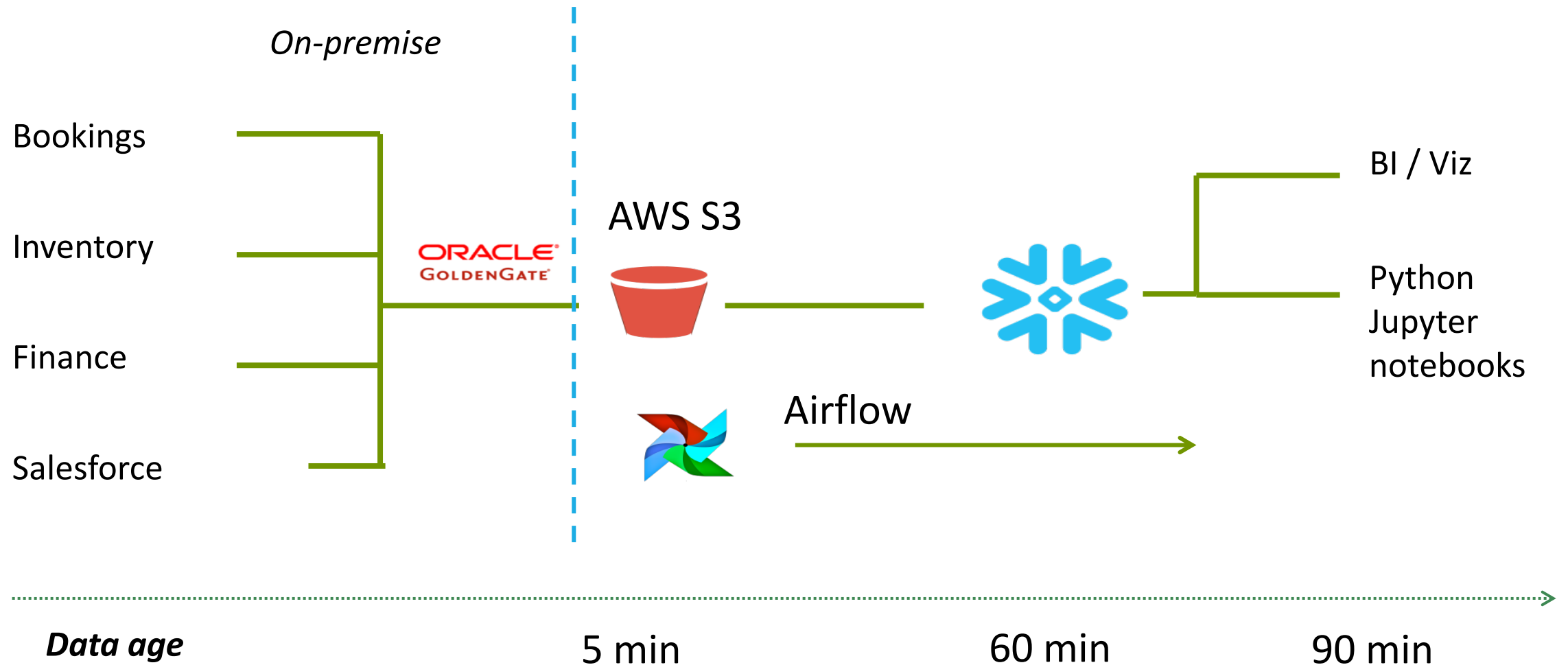
We do everything in Snowflake

- Replicate source
- Transparency!
- No ETL tool

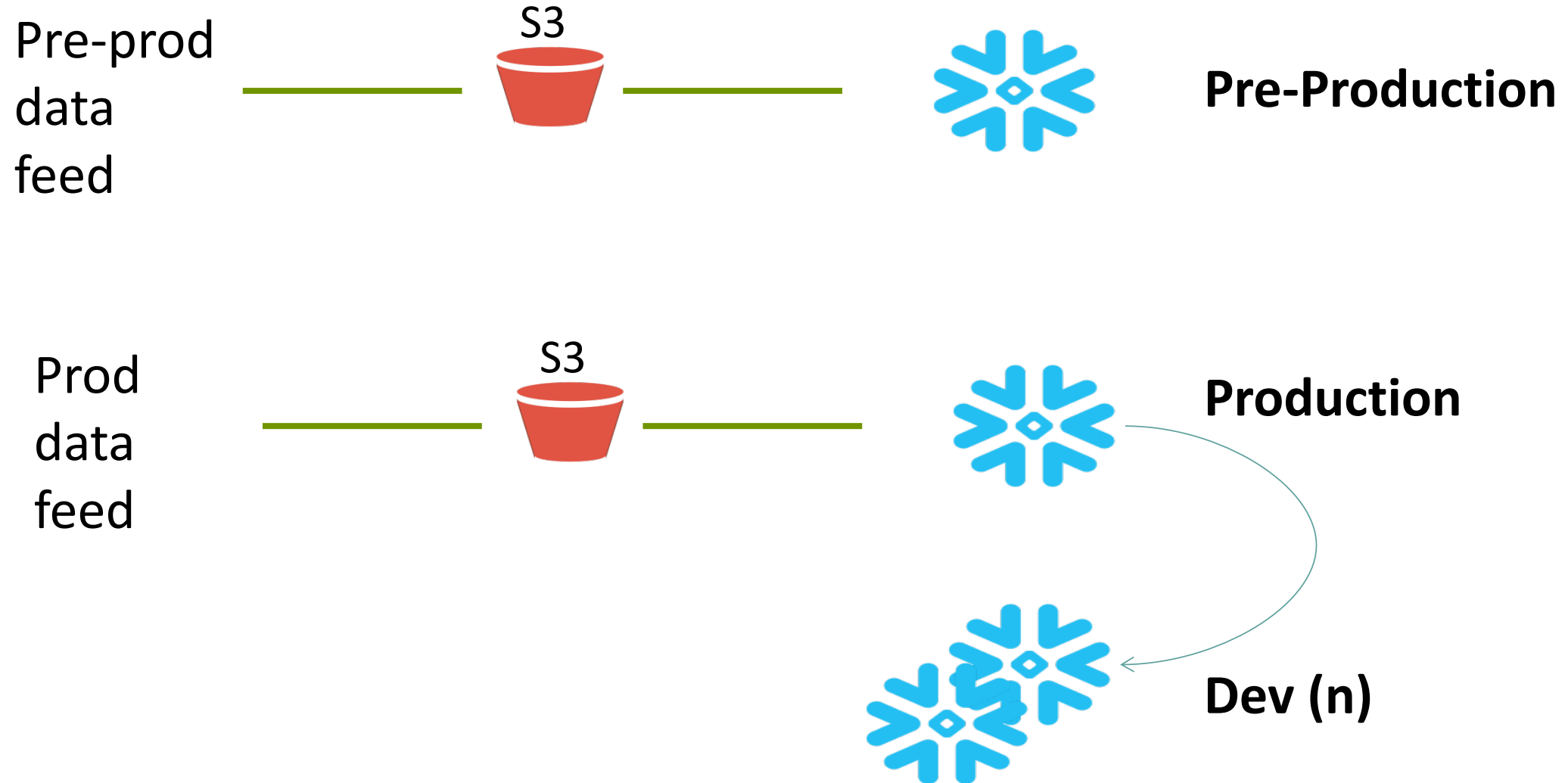


2. We ingest every 5 mins...

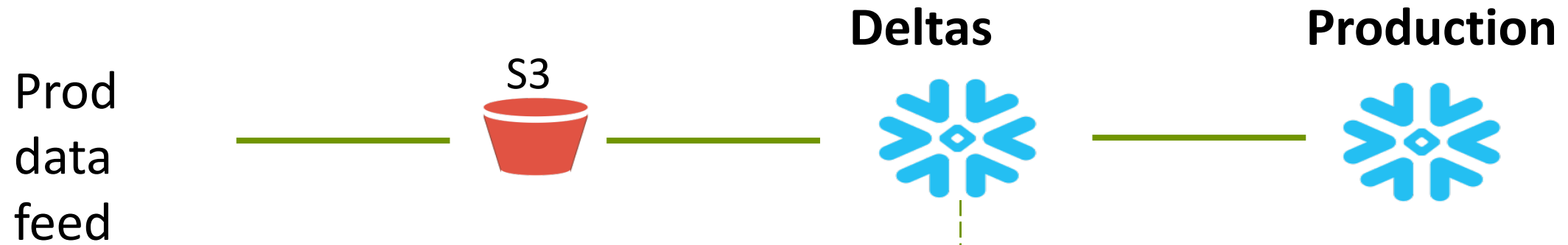
Data age: 1.5 days -> 1.5 hrs



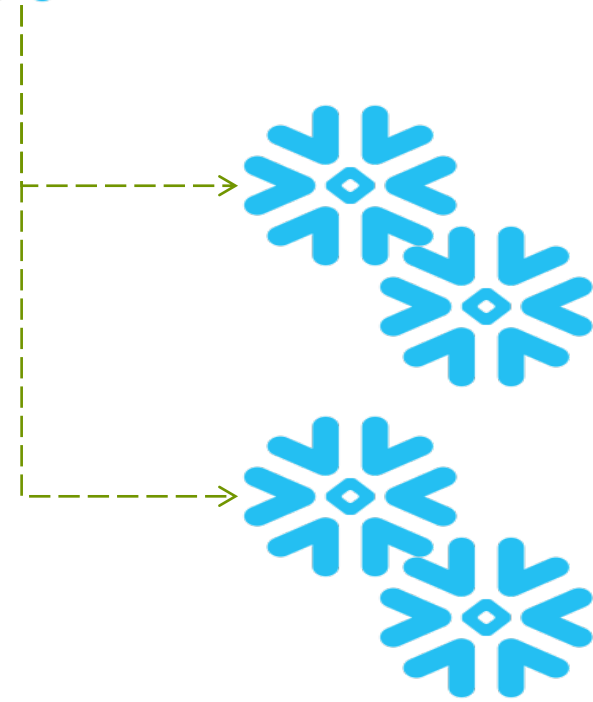
3. Agile Development - Cloning!



3. Using Cloning – next



- Maintain 1 feed
- (n) Dev & Test pairs
On-demand (*cost*)
- Live Deltas feed from point of cloning



4. Scaling – flexibility!

Before you start your project, how confident are you on:

- Load & concurrency?; Storage? Test & Dev.?
- Ad-hoc Analytics / Data Science pulls?

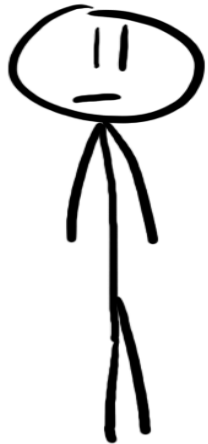
Started small, and scale up or out as needed:

- Direct BI querying on large data sets
- Project workloads
- Re-processing



4. Snowflake reduces opportunity cost to try things

My new pricing analysis code takes 3m to run...



Can I run it 7000 times?

***Snowflake scaling =
Reducing Opportunity costs for experimentation***

Agile tips

- **Agile modelling** on whiteboards with the business
- **Prototyping** – share early in excel & BI tool
- **Iterate** – 1st version in use early
- **Milestones** - no big bang
- **Verification** – plan time for this and tackle early

Takeaways on Snowflake + agile

- Choose tools that fit your team's skillset
- Choose tools that move you quickly to delivering business value
- Transform in your target environment
- Create an agile development environment
- Choose tools that are flexible and on-demand to start small

A tropical sunset landscape with palm trees and a white text box. The sky is filled with soft, golden light and wispy clouds. In the foreground, the dark silhouettes of palm trees are visible against the bright background. A large white rectangular box is positioned in the lower-left quadrant of the image, containing text.

Q&A

Adam Slader

adam.slader@gta-travel.com

Thank You to Our Partners

Platinum



Gold



CLOUD ANALYTICS
CONFERENCE

