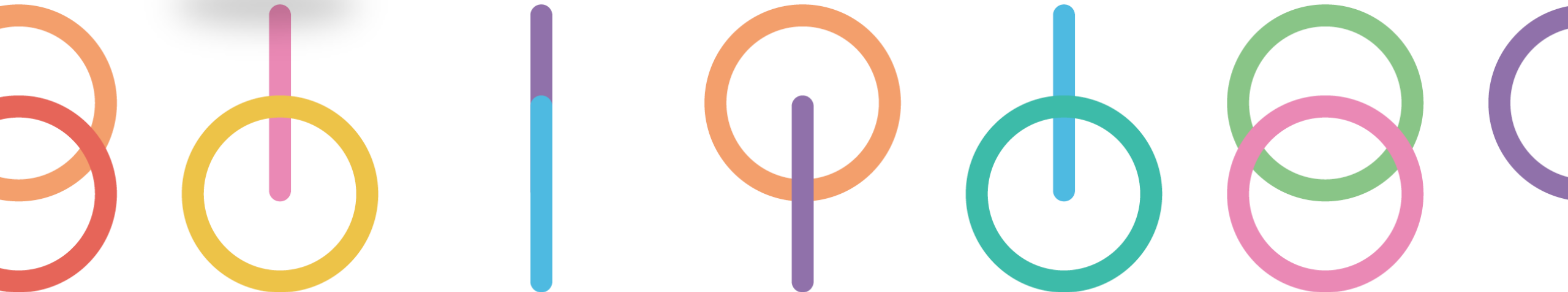




Scenari Event-Driven con SQL Server



Alberto Acerbis
Software Architect - Trainer



Ogni giorno Query Quinn si sveglia con l'idea in testa che i dati dei vicini sono più verdi dei suoi!

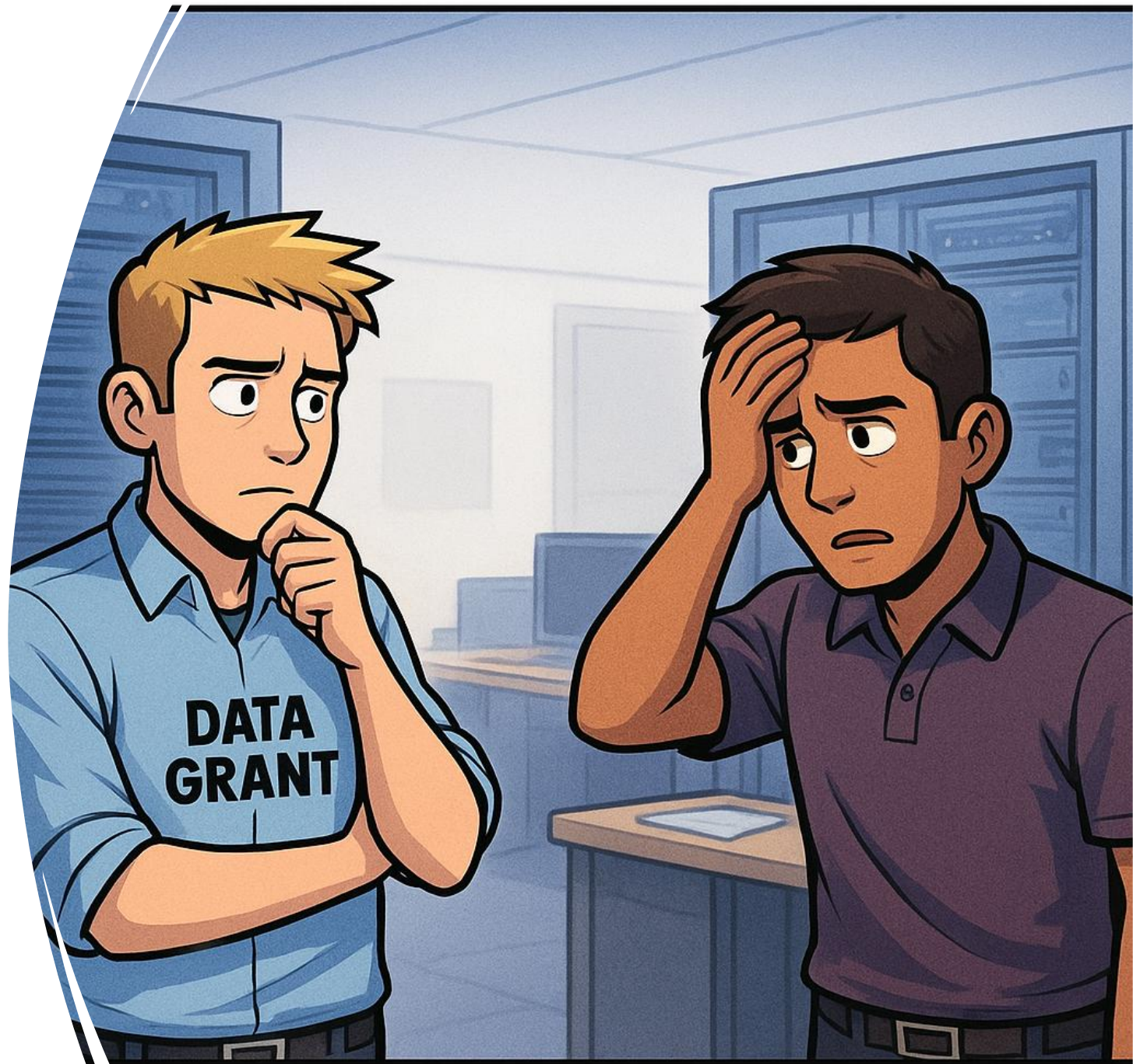
**I NEED DATA
FROM YOUR
DATABASE**



-
- Ogni giorno Data Grant si sveglia con la necessità di creare nuovi ETL per il suo collega Query Quinn!

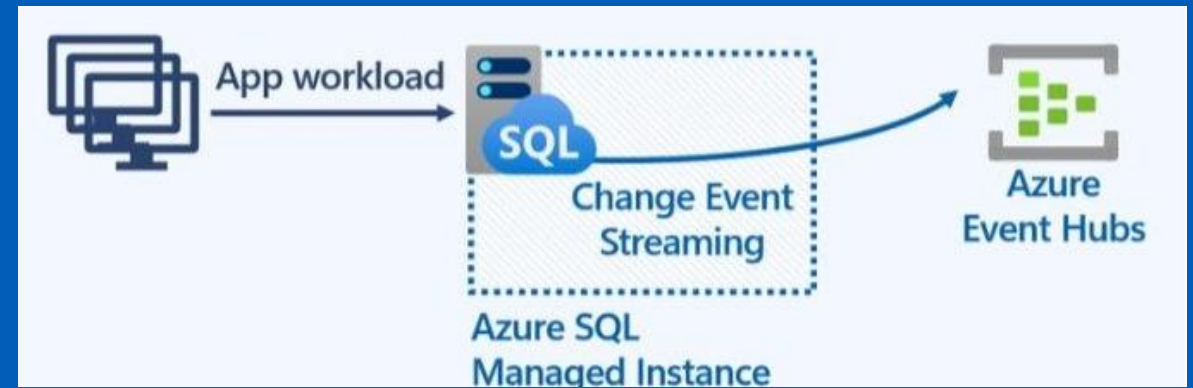


How can we
solve the
problem?



Change Event Streaming

- Change Event Streaming è una funzionalità di integrazione fornita da SQL Server 2025.
- Acquisisce e pubblica le modifiche incrementali dai dati in tempo reale (quasi).
- I dettagli delle modifiche vengono inviati ad Azure Event Hub sotto forma di CloudEvent

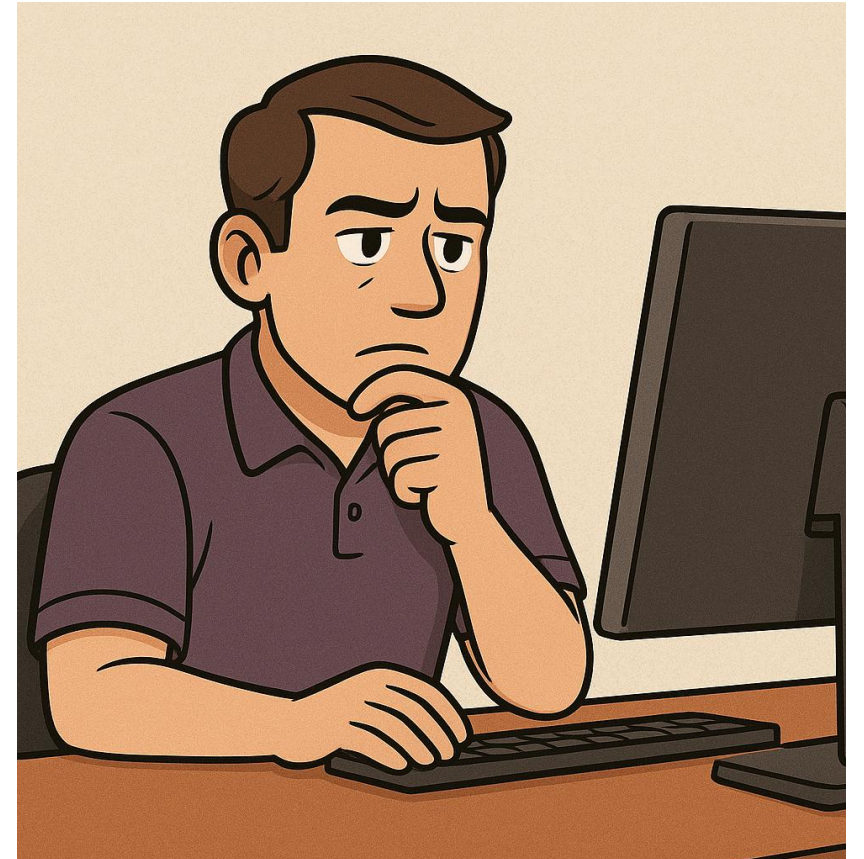


A specification
for describing
event data in a
common way



Come si attiva?

- Generazione SAS Token.
- Abilitazione a livello di Database.
- Creazione di un EventHub Namespace.
- Creazione di un EventHub.
- Creazione di un Gruppo in SQL Server.
- Associazione Gruppo-Tabella



Demo Time



Main advantages

- **Scalabilità:** Supporto per throughput elevato.
- **Disaccoppiamento:** Maggiore flessibilità significa manutenzione più semplice.
- **Multi-consumer:** Azure Event Hubs consente a più consumer di accedere allo stesso flusso.
- **Integrazione Real-time:** Consente l'integrazione fra sistemi OLTP e fra diversi database.





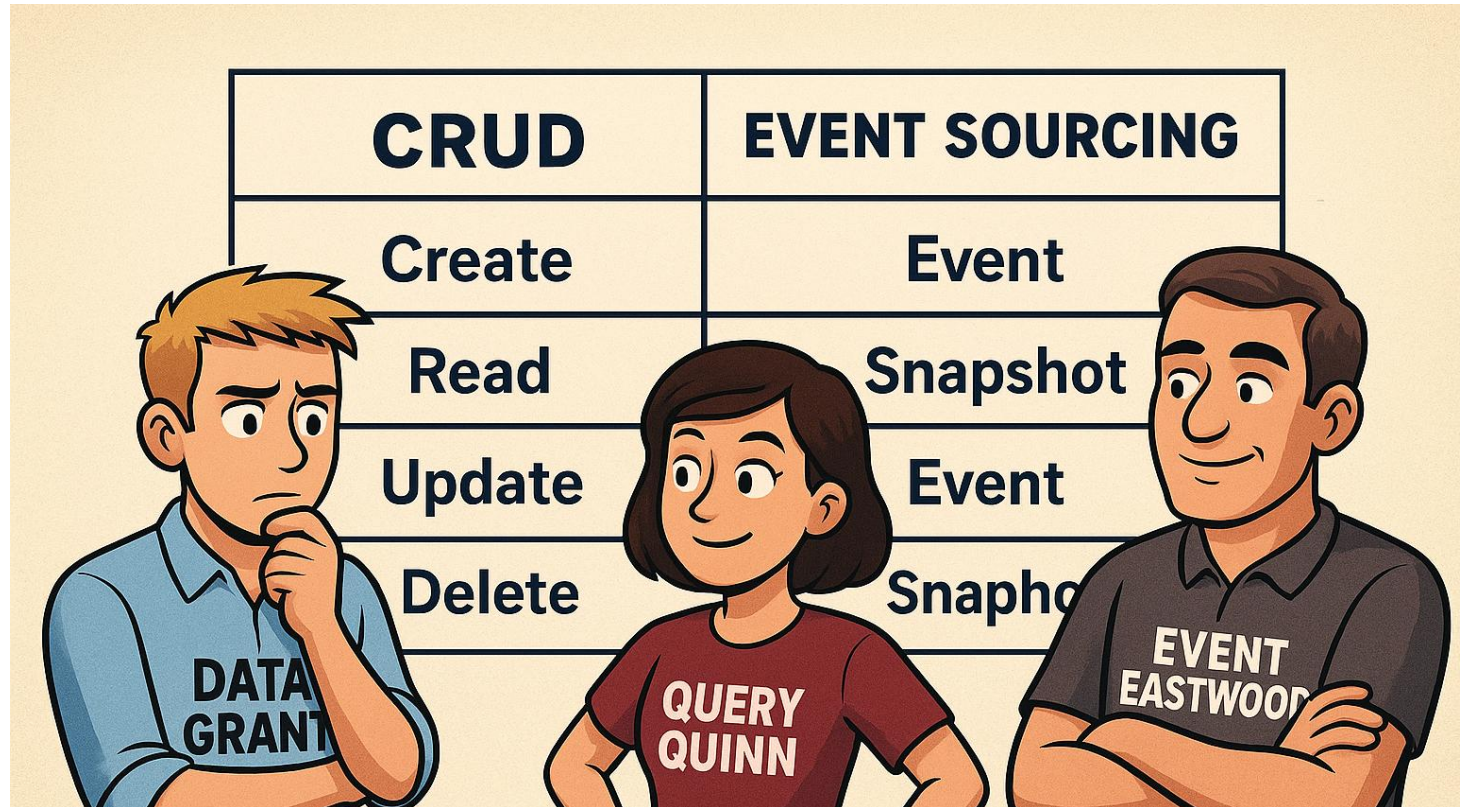
Why don't we create
an EventStore?

EventStore



Aspetto	Proprietà
Eventi immutabili	Ogni evento rappresenta qualcosa che è accaduto. Una volta scritto, non viene modificato!
Timeline completa	L'EventStore conserva <i>la storia degli eventi</i> , non solo l'ultimo stato. (Auditing, debugging, analytics, state builder)
Ricostruzione dello stato	Lo stato attuale si ottiene rileggendo e applicando tutti gli eventi associati ad un aggregato.
Perfetto per architetture event-driven	Si integra naturalmente con: Event Sourcing CQRS Microservizi

-
- The DBA dream team evaluates the transition from CRUD to Event Sourcing



CQRS

Command Query Responsibility Segregation

COMMAND
SIDE

- Netta separazione fra scrittura e lettura dei dati.
- Netta separazione dello storage (Write Model & Read Model).
- Write Model come “Source of Truth”.
- Possibilità di avere diverse “viste” dello stesso aggregato (Projection).



QUERY SIDE

Event Sourcing



Aspetto	CRUD	Event Sourcing
Concetto base	Memorizza <i>l'ultimo stato</i> di ogni entità	Memorizza <i>tutti gli eventi</i> che hanno portato a quello stato
Operazioni tipiche	Create, Read, Update, Delete	Append di nuovi eventi immutabili
Storia dei dati	Lo stato precedente è perso (a meno di audit log)	La storia completa è conservata
Consistenza e ricostruzione	Lo stato è sempre «qui e ora»	Lo stato è sempre derivato <i>riproducendo</i> gli eventi
Audit e tracciabilità	Richiede log o trigger aggiuntivi	Intrinseca nel modello
Scalabilità e integrazione	Centrico e sincrono	Distribuito e asincrono per natura
Casi d'uso reali	CRUD app, sistemi di gestione dati	Sistemi reattivi, CQRS, DDD, microservizi

Demo Time





Real-Time

Change Event Streaming introduce un nuovo modo per acquisire in tempo reale le modifiche dei dati

SQL evolution

SQL Server non è più soltanto una DB relazionale (JSON, Vector), ma un partecipante attivo nel flusso di eventi di sistema

Streaming nativo

Si integra nativamente con le piattaforme di eventi cloud



Alberto Acerbis

GRAZIE!



alberto.acerbis@intre.it



<https://github.com/Ace68/CES-Microservices>



INTR3 |

Domain-Driven Refactoring

A hands-on DDD guide to transforming monoliths into modular systems and microservices

Alessandro Colla and Alberto Acerbis

