SERIÖS GROUP

Building Modern Data Platforms with Seriös ONE

Part Three

Table of Contents

Delivering Data for Analytics 2
Gold Layer: Kimball with SQL Flexibility 3
Building Dimension Tables 3
Building Fact Tables 5
Configuration as Code 5
Making Dimensional Modelling Easier to Maintain 6
What's Next 6

©Seriös Group | Company Number. 12456390
Studios 2 & 6, The Old Forge, Hoults Yard, Newcastle Upon Tyne NE6 2HL

Delivering Data for Analytics

This is the third in a short series of articles where Jake Wedge, Senior Product Engineer, explores how modern data platforms can be built in a faster, more consistent and more maintainable way using DataOps.

The Gold layer is where data becomes consumable for reporting, dashboards and analytics. Whilst Data Vault provides a scalable foundation for integrating and governing data in the Silver layer, business users need data organised into structures that are easy to query and understand. This is where dimensional modelling comes in.

In this article, we'll explore how Seriös ONE supports Kimball-style dimensional modelling through a metadata-driven approach, combining automation for common patterns with the flexibility of Spark SQL.

Gold Layer: Kimball with SQL Flexibility

The Gold layer is where business users live. They need data organised into familiar star schemas: fact tables with measures and dimension tables with descriptive attributes. Seriös ONE Core makes this transformation straightforward whilst giving data engineers complete freedom to express complex business logic.

Here's the crucial design choice: whilst the framework provides standardised handling for Kimball patterns (surrogate keys, SCD types, default rows, incremental merges), you define transformations using **Spark SQL**. This gives you the full expressiveness of SQL for complex joins, window functions, conditional logic, and aggregations.

Continue reading to explore how Seriös ONE handles dimension and fact tables, automates common Kimball patterns and uses metadata to support lineage and AI across the Gold layer.

Building Dimension Tables

For dimension tables, you create a configuration file specifying:

- The natural keys that define each dimension member
- The surrogate key name
- The slowly changing dimension type (Type 1 or Type 2)
- A SQL file that transforms Silver layer data into dimension attributes

```
{
    "DestinationEntityName": "Dim_Customer",
    "DestinationEntityType": "DW_Dimension",
    "DestinationEntityLayerName": "Gold",
    "SurrogateKeyName": "CustomerKey",
    "NaturalKeyFields": ["customer_id"],
    "SCDType": 2,
    "SQLFile": true,
    "SQLFileName": "Dim_Customer.sql"
}
```

```
SELECT
    hub.customer_id,
    sat.customer_name,
    sat.email,
    sat.segment,
    CASE
        WHEN sat.lifetime_value > 10000 THEN 'High Value'
        WHEN sat.lifetime_value > 5000 THEN 'Medium Value'
        ELSE 'Standard'
    END as value_category,
    sat.registration_date
FROM silver.hub_customer hub
JOIN silver.hub_sat_customer sat ON hub.CustomerHashKey = sat.CustomerHashKey
WHERE sat.LoadEndDate IS NULL
```


©Seriös Group | Company Number. 12456390
Studios 2 & 6, The Old Forge, Hoults Yard, Newcastle Upon Tyne NE6 2HL

The framework then:

- Executes your SQL transformation
- Generates integer surrogate keys automatically
- Creates the standard -1 and -2 default rows (for unknown and not applicable)
- Handles SCD Type 2 effective dating if configured
- Manages incremental loading based on watermarks
- Fills NULL values with Kimball-standard defaults

Building Fact Tables

For fact tables, the pattern is similar:

```
{
    "DestinationEntityName": "Fact_Sales",
    "DestinationEntityType": "DW_Fact",
    "DestinationEntityLayerName": "Gold",
    "DestinationLoadType": "ComparisonMerge",
    "NaturalKeyFields": ["transaction_id"],
    "SQLFileName": "Fact_Sales.sql",
    "HashComparison": true
}
```

The framework then:

Your Fact_Sales.sql contains the business logic to join Hubs, Satellites, and Links into the fact grain, lookup foreign keys to dimensions, and calculate any derived measures. The framework handles the merge logic, incremental processing, table optimisations, grain enforcement and enforcing the fact type.
This approach gives you the best of both worlds. You express business logic in familiar SQL whilst the framework ensures consistent Kimball patterns across your warehouse. No need to manually code surrogate key lookups, SCD logic, or merge operations. No risk of inconsistent patterns between different fact tables.

©Seriös Group | Company Number. 12456390
Studios 2 & 6, The Old Forge, Hoults Yard, Newcastle Upon Tyne NE6 2HL

Configuration as Code

The SQL files live in Git alongside your JSON configurations, enabling proper code review, version control, and CI/CD deployment. Change a transformation? Commit it, test it, deploy it through your standard pipeline. The framework ensures the deployment is safe and consistent.
The metadata pays off for AI here too: because every transformation and its lineage are captured declaratively, agents can read across the Gold layer to explain exactly how any fact or dimension is constructed, trace it back through Silver and Bronze, or draft the SQL for a new one.

Making Dimensional Modelling Easier to Maintain

Kimball modelling remains one of the most widely adopted approaches for analytics because it provides a structure that is easy for reporting tools and business users to consume. The challenge is maintaining consistency across dimensions, facts and transformations as a platform grows.
By standardising common modelling patterns whilst allowing engineers to define transformations in SQL, Seriös ONE helps teams build and maintain dimensional models without repeatedly implementing the same supporting logic. This keeps development focused on business requirements whilst ensuring a consistent approach across the warehouse.

What's Next

In the final article of this series, we'll look at how the metadata-driven approach behind Seriös ONE helps engineering teams work more effectively. We'll explore how configuration as code, built-in best practices, platform flexibility and support for agentic AI help organisations deliver data platforms faster whilst maintaining quality and governance.
If you'd like to find out more about Seriös ONE click here or speak to one of our data solutions experts.

©Seriös Group | Company Number. 12456390
Studios 2 & 6, The Old Forge, Hoults Yard, Newcastle Upon Tyne NE6 2HL

SERIÖS GROUP

Studios 2 & 6, The Old Forge,
Hoults Yard, Walker Road,
Newcastle Upon Tyne, NE6 2HL

seriosgroup.com
hello@seriosgroup.com

© Seriös Group 2026
Company Number. 12456390