Query Overview

Data exploration in SIFT begins with queries, the mechanism for defining which data to analyze and how to structure it for comparison.

In a nutshell, queries sit in the following order of operations:

Library → Query → Groups (which can contain multiple conditions) → Analysis

Groups

Querying data creates groups. Groups are SIFT's fundamental unit of analysis. All traces (or metrics) within a group are treated as belonging together, and SIFT's analysis features (such as SPM) operate by comparing groups against one another. How you structure your groups depends on the questions you want to ask.

Conditions

Within a group, multiple conditions allow you to specify distinct requirements for different traces while maintaining their relationship. Conditions define the different rules traces can meet to be part of the same group.

Building Effective Queries

Query design should reflect your analytical intent. Traces you plan to compare or analyze together belong in the same group. Traces representing different experimental conditions should be separated into distinct groups to enable meaningful comparison.

Working with Queries

Autopopulate Queries

Sift can generate queries automatically based on signals present in your library / the underlying CMZ. This approach surfaces all available data within a specified scope, making it well-suited for open-ended exploration when your direction is still taking shape.

When to use: Use 'Autopopulate Queries' for open-ended exploration and when the direction of the project is still taking shape. Refinements can be applied to filter results based on metadata values.

Learn More: Auto Populate Queries Dialog

Tutorial: See the Walk-through section of the Getting Started page

Custom Queries

Sift lets you build precise queries tailored to your specific requirements. Custom queries support replication of previous analyses and enable advanced refinements including time-normalization to event sequences and filtering based on signal characteristics.

When to use: Use Custom Queries when the signals of interest are known in advance, when a previous analysis needs to be replicated, or when precise control over time-normalization or signal filtering is required. Query definitions can be saved as .q3d files for reuse across sessions.

Learn More: Manual Query Dialog

Tutorial: See the Custom Queries section of the Getting Started Page

Example

In gait analysis, comparing knee joint angles across three walking speeds (SLOW, MEDIUM, FAST) might use three separate groups (one per speed condition). Each group would contain two conditions: left knee angle time-normalized from left heel strike (LHS)-LHS and right knee angle from right heel strike (RHS)-RHS, both refined by the appropriate speed tag. This structure enables direct comparison across speed conditions while properly handling bilateral symmetry within each condition.

Group 1: KneeAngleX_SLOW	**Group 2:** KneeAngleX_MEDIUM	**Group 3:** KneeAngleX_FAST
Condition A: Left Side **Signal**: LKneeAngleX **Events**: LHS-LHS **Tag**: SLOW	**Condition A:** Left Side **Signal**: LKneeAngleX **Events**: LHS-LHS **Tag**: MEDIUM	**Condition A:** Left Side **Signal**: LKneeAngleX **Events**: LHS-LHS **Tag**: FAST
Condition B: Right Side **Signal**:RKneeAngleX **Events**: RHS-RHS **Tag**: SLOW	**Condition B:** Right Side **Signal**:RKneeAngleX **Events**: RHS-RHS **Tag**: MEDIUM	**Condition B:** Right Side **Signal**:RKneeAngleX **Events**: RHS-RHS **Tag**: FAST

Tutorials

Refine Queries with Metadata

- Practical applications of query refinement

Analysis of Shoulder Angular Velocity between Elite Level and Average Collegiate Pitchers

- Build Necessary Queries for Normal Database
- Build 'Elite' Group based on Top 10 Pitchers determined by Pitch Speed