

Odysseus Home

Odysseus Documentation

On this pages we present the Odysseus project. Odysseus is a data stream management framework that can be used to create efficient applications that require data stream or event based processing.

Some more information can be found at <http://odysseus.informatik.uni-oldenburg.de>

Look here for detailed documentation:

Getting Started and Installation

- [How to install Odysseus](#)
- [How to install new features](#)
- [How to update Odysseus](#)
- [Enable extended logging](#)
- [How to report a bug](#)
- [ODYSSEUS_HOME](#)
- [Odysseus.conf](#)
- [Run with Docker](#)
- [Use odysseus console](#)
- [WebStudio](#)
- [Grafana](#)

Tutorials

- [Odysseus Studio](#)
- [Getting Started with Nexmark](#)
- [Simple Query Processing](#)
- [Selection, Projection and Map](#)
- [Aggregation and Window](#)
- [Join](#)
- [Run Nexmark Example](#)
- [Working with Charts](#)
- [How to access sources with different input inside a input single stream](#)
- [Connecting Odysseus and Telegram](#)
- [Reading RSS Feed with Odysseus](#)
- [How-to articles](#)

Accessing Odysseus Server

- [REST interface V2](#)
- [Reading Query Results via REST/Websocket](#)
- [Rest Information Services](#)

Use Cases

- [Patterns for Realtime Streaming Analytics](#)
- [ACM DEBS Grand Challenge](#)
- [Odysseus for Moving Object Processing](#)
- [RabbitMQ use case](#)
- [Adaptive Query Processing](#)

Query Data Input

- [LineSender](#)

Query Languages

- [Odysseus Script](#)
- [Procedural Query Language \(PQL\)](#)
- [Continuous Query Language \(CQL\)](#)
- [SASE](#)

Data Types

Operators

- [Base operators](#)
- [Advanced operators](#)
- [Source operators](#)
- [Sink operators](#)
- [Database operators](#)

- Enrich operators
- Pattern operators
- Mining operators
- Anomaly Detection Operators
- Recommender System operators
- Probabilistic operators
- Order operators
- Plan operators
- Processing operators
- Transform operators
- Benchmark operators

MEP: Functions and Operators

- Bit Functions
- Command
- Counter
- Datatype Functions
- Date Functions
- Encryption Functions
- Graph (Graph Server Feature)
- Image Functions
- ImageJCV Functions
- Interval Functions
- KV Store Function
- List Functions
- Mathematical Functions
- Matrix Functions
- MDA store functions
- Miscellaneous Functions
- Probabilistic Functions
- Spatial Functions
- String Functions
- Systemload Function
- Text Functions

Meta data

- Data rate
- Latency
- Priority
- Probabilistic
- Systemload Metadata
- TimeInterval

Access framework

- Protocol handler
- Transport handler
- Data handler
- Datatypes

Scheduling

Odysseus User Management

Features

- Core Features
- Incubation Features
- OdysseusNet
- Currently not (fully) available features

Development with Odysseus

- Odysseus Development Startup
- Target Platform
- Add a new Bundle and Feature
- Add Bundle with Ressources/Libraries
- Tycho builds for Odysseus
- The Odysseus Operator Framework
- Adding features to products
- Creating new Operators
- Creating a new Wrapper for Odysseus
- Access Operators
- Integration Tests
- The Odysseus Operator Test Framework

- [The Odysseus Procedural Query Language \(PQL\) Framework](#)
- [Developing DashboardParts](#)
- [MEP Functions](#)
- [The Odysseus Rule Engine](#)
- [Programmer's FAQ](#)
- [Extending OdysseusScript](#)
- [Jenkins](#)
- [The Command Architecture](#)
- [Create an Odysseus Docker Container](#)
- [Create New Aggregation Function](#)
- [Creating Metadata types](#)
- [Customizing Operator Graph Visualization \(Odysseus Studio\)](#)
- [Image Manager and Image Sets](#)
- [Getting Parameter from OdysseusScript into Odysseus](#)
- [Odysseus Development Best practices](#)
- [Odysseus Architectural overview](#)
- [Create a new language](#)
- [Create a new Datatype](#)
- [Extending PQL](#)
- [Output schema](#)

Odysseus FAQ

Odysseus Cheat Sheet

A generated cheat sheet with all operators, functions and data types can be found [here](#).