



Omnis Developer Conference

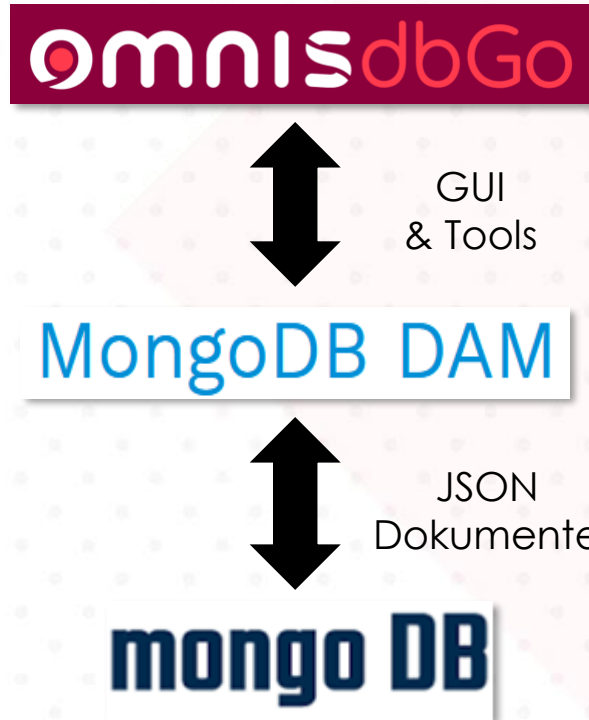
Vereinfachen. Virtualisieren. Entwickeln.

**Einführung von Omnis dbGo - die
Überholspur zur MongoDB Entwicklung**

Götz Krija

Omnis dbGo

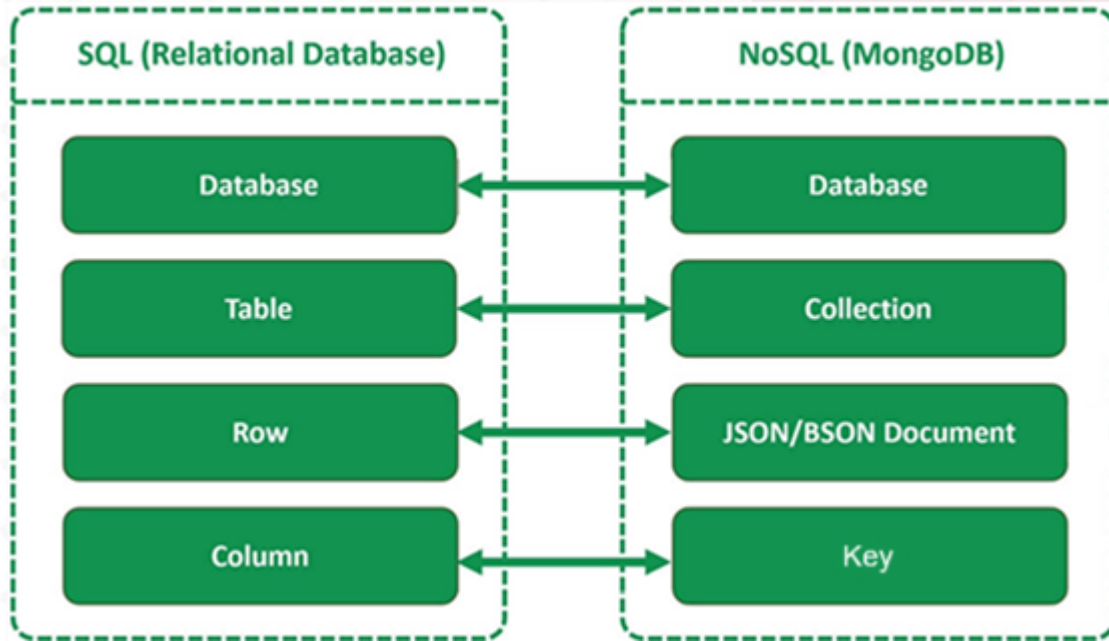
Was ist dbGo?



- Leistungsstarkes Tool auf Basis des **MongoDB DAMs** (NoSQL)
- Bietet zusätzliche Funktionen und **GUI mit visuellen Tools**, die es ermöglichen, MongoDB Datenbanken zu verwalten und Abfragen zu erstellen
- Ermöglicht schnell und einfach **Web- und Desktop Applikationen zu erstellen**, die auf den MongoDB Daten und Abfragen basieren

RDBMS vs MongoDB (NoSQL)

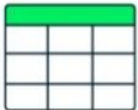
Datenmodell und Terminologie



- Relationale DBs speichern Daten in Tabellen (mehrere Zeilen)
- MongoDB speichert Daten in Collections (mehrere JSON-Dokumente)
- Zeile in einer Tabelle entspricht einem JSON-Dokument in einer MongoDB Collection
- Eine Spalte in einer Tabelle ist vergleichbar mit dem Schlüssel eines Schlüssel-Werte-Paares in einem JSON-Dokument

RDBMS vs MongoDB (NoSQL)

Tabellen vs Collections



Relational
Database

User table

ID	first_name	last_name	cell	city
1	Leslie	Yepp	8125552344	Pawnee

Hobbies table

ID	user_id	hobby
10	1	scrapbooking
11	1	eating waffles
12	1	working



MongoDB

User Collection (JSON Document)

```
{
  "_id": 1,
  "first_name": "Leslie",
  "last_name": "Yepp",
  "cell": "8125552344",
  "city": "Pawnee",
  "hobbies": ["scrapbooking", "eating waffles", "working"]
}
```

- Relationale DBs speichern Daten in Tabellen mit Spalten und Zeilen
- MongoDB speichert Daten als Schlüssel/Wert-Paare in einem JSON-Dokument
- Keine Notwendigkeit für Daten-Normalisierung
- Keine Notwendigkeit für Joins

RDBMS vs MongoDB (NoSQL)

SQL vs MongoDB Query Language(MQL)

SQL

```
SELECT * FROM users WHERE age > 30;
```

```
INSERT INTO users (name, age) VALUES  
( 'Alice', 30 );
```

```
UPDATE users SET age = 31 WHERE name =  
'Alice';
```

```
DELETE FROM users WHERE age < 20;
```

MongoDB

```
db.users.find({ age: { $gt: 30 } })
```

```
db.users.insertOne({ name: "Alice", age: 30 })
```

```
db.users.updateOne({ name: "Alice" }, { $set: {  
age: 31 } })
```

```
db.users.deleteMany({ age: { $lt: 20 } })
```

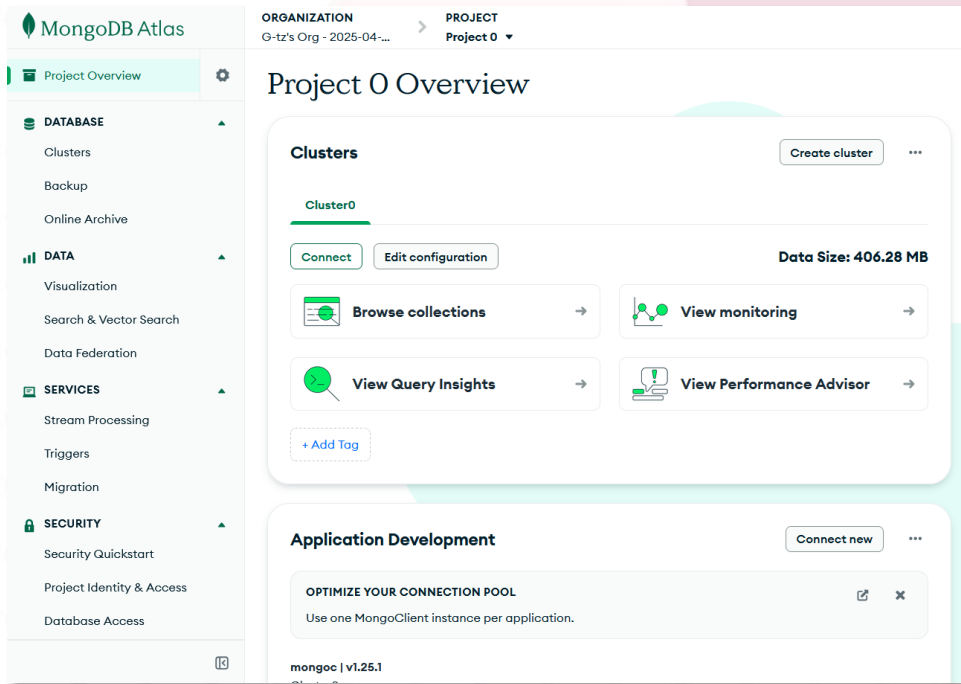
Verwendung einer eigenen proprietären Abfragesprache

Typische Abfrage besteht aus drei Hauptbestandteilen:

1. Collection: `db.users`
2. Methode: `find()`
3. Bedingung mit Operatoren im JSON-Format:

```
{ { age: { $gt: 30 } } }
```

MongoDB Atlas



Schneller Einstieg mit MongoDB Atlas (Cloud Lösung):

<https://www.mongodb.com/products/platform/cloud>

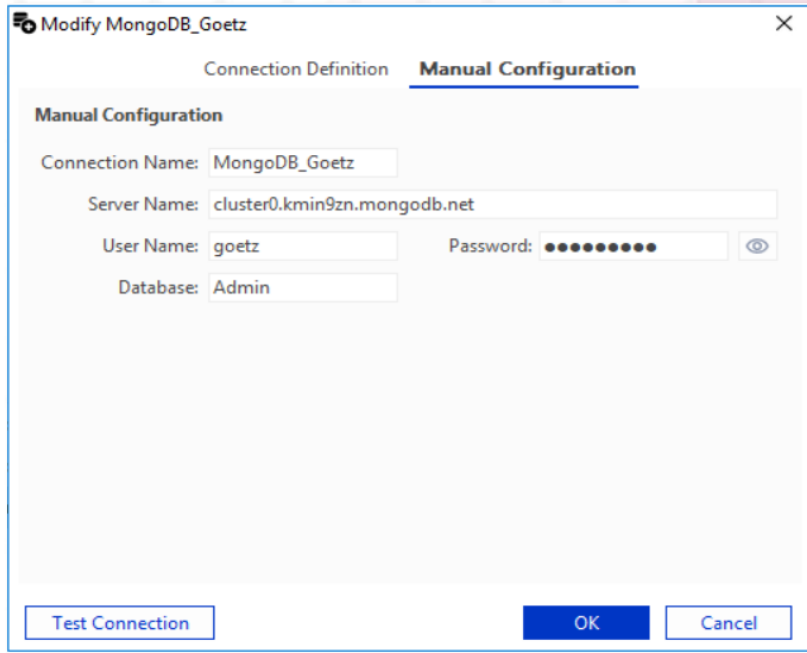
- Einfache Anmeldung
- Bietet kostenlose Stufe mit Beispiel-Collections und Daten
- Eigene Host-URI, Benutzernamen und Passwort

Omnis dbGo

Features

Omnis dbGo Features

Connection Manager



The screenshot shows a dialog box titled 'Modify MongoDB_Goetz' with a close button (X) in the top right corner. It has two tabs: 'Connection Definition' and 'Manual Configuration', with the latter being selected. The 'Manual Configuration' section contains the following fields:

- Connection Name: MongoDB_Goetz
- Server Name: cluster0.kmin9zn.mongodb.net
- User Name: goetz
- Password: [masked with dots] and a toggle icon
- Database: Admin

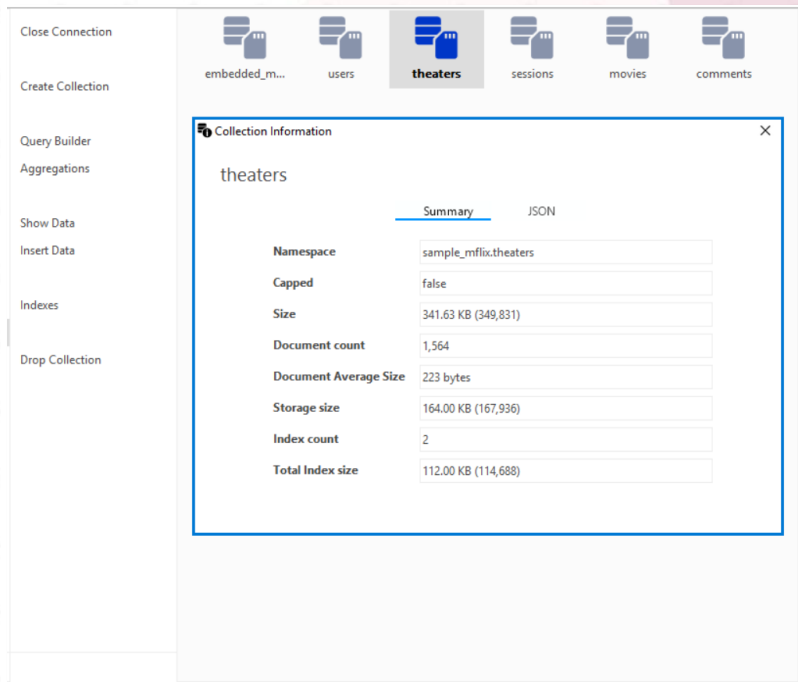
At the bottom of the dialog, there are three buttons: 'Test Connection', 'OK', and 'Cancel'.

Datenbankverbindungen verwalten

- Neue MongoDB-Verbindungen einrichten und öffnen
- Zuvor angelegte Verbindungen öffnen

Omni dbGo Features

dbGo Browser

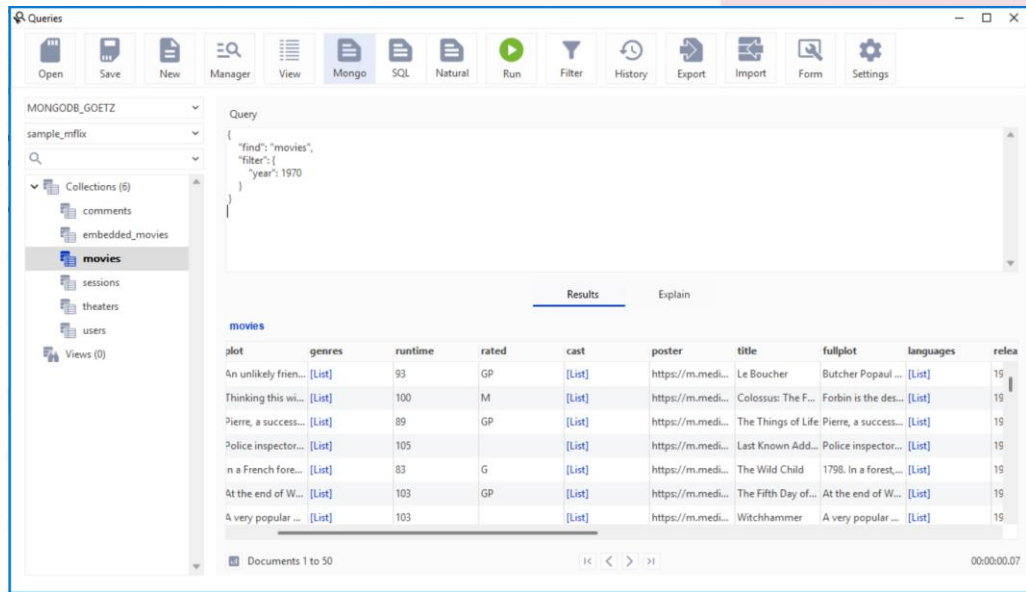


Datenbanken, Collections und Dokumente verwalten

- Datenbanken, Collections und Dokumente anzeigen
- Collections erstellen & löschen
- Neue JSON-Dokumente einfügen oder importieren
- Daten in Dokumenten anzeigen und aktualisieren
- Indizes hinzufügen, bearbeiten und löschen

Omni dbGo Features

Query Builder



Daten abfragen

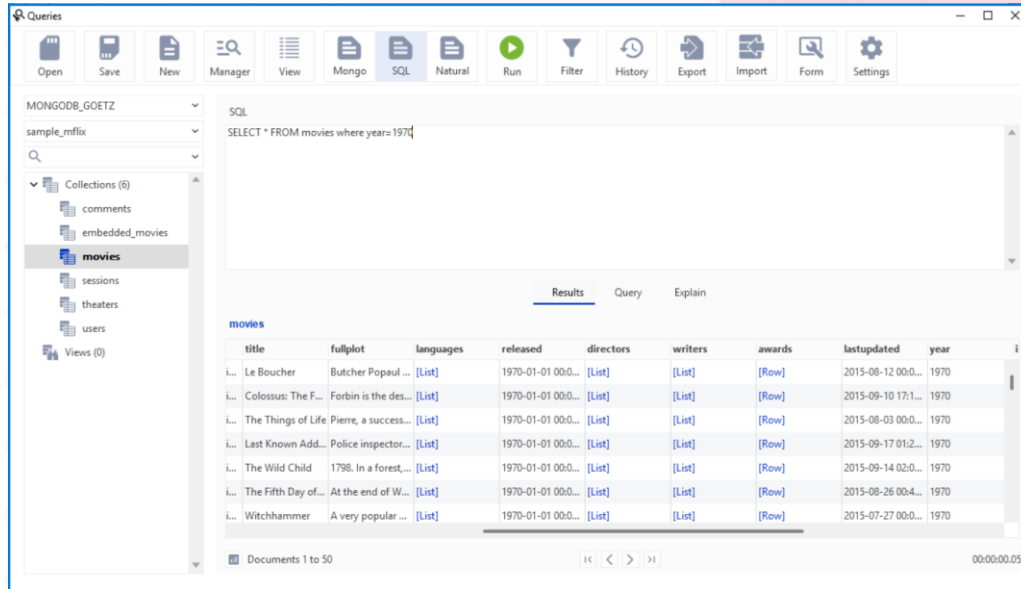
Erstellung und Ausführung nativer MongoDB-Abfragen im JSON-Format, z.B.:

```
{  "find": "movies",  "filter": {    "year": 1970  }}
```

Ergebnisdaten werden im Fenster angezeigt.

Omnis dbGo Features

Query Builder



Daten abfragen

Erstellung von Standard-SQL Abfragen:

```
SELECT * FROM movies where  
year=1970
```

Omnis dbGo “übersetzt” in eine native MongoDB Abfrage

Omnis dbGo Features

Query Builder

The screenshot displays two windows from the Omnis dbGo application. The top window, titled 'Natural Language', is in the 'AI Setup' stage. It contains instructions: 'To generate Mongo queries using Natural Language you must obtain an API key from your chosen AI Service provider. Enter this together with your configuration details and press Save, followed by OK to select.' Below the text is a table with columns: Name, Service, Model, and API Key. The table lists three providers: OpenAI (gpt-4o-mini), Anthropic (claude-3-5-sonnet-latest), and Gemini (gemini-1.5-flash-latest), all with 'YES' in the API Key column. The bottom window, titled 'Queries', shows the 'Natural Language' tab selected. The 'Name' field is 'MONGODB_GOETZ', 'Service' is 'sample_mflix', and 'Model' is 'collections (8)'. The 'API Key' field is empty. The 'Natural Language' input field contains the text 'Show all movies released in year 1970'. The 'Results' tab is active, displaying a table of movie data.

Name	Service	Model	API Key
OpenAI	OpenAI	gpt-4o-mini	YES
Anthropic	Anthropic	claude-3-5-sonnet-latest	YES
Gemini	Gemini	gemini-1.5-flash-latest	YES

plot	genres	runtime	rated	cast	poster	title	fullplot	languages	release
An unlikely frien...	[List]	93	GP	[List]	https://m.med...	Le Boucher	Butcher Popaul ...	[List]	19
Thinking this wi...	[List]	100	M	[List]	https://m.med...	Colossus: The F...	Forbin is the des...	[List]	19
Pierre, a success...	[List]	89	GP	[List]	https://m.med...	The Things of Life	Pierre, a success...	[List]	19
Police inspector...	[List]	105		[List]	https://m.med...	Last Known Add...	Police inspector...	[List]	19
n a French fore...	[List]	83	G	[List]	https://m.med...	The Wild Child	1798. In a forest...	[List]	19
At the end of W...	[List]	103	GP	[List]	https://m.med...	The Fifth Day of...	At the end of W...	[List]	19
4 very popular ...	[List]	103		[List]	https://m.med...	Witchhammer	A very popular ...	[List]	19

Daten abfragen

Erstellung von Abfragen in natürlicher Sprache (KI-gestützt):

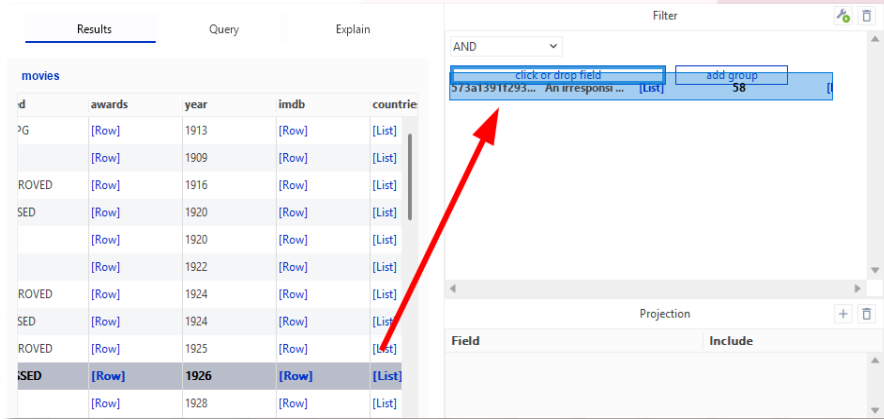
“Show all movies released in year 1970”

KI-Anbieter auswählen und API-Key hinzufügen:

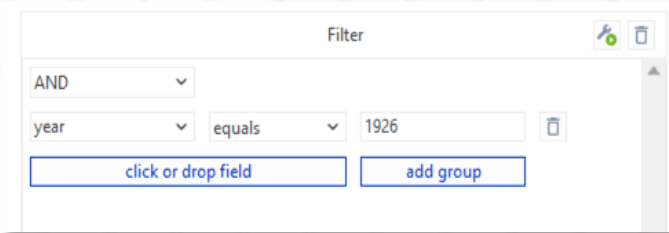
- OpenAI
- Anthropic
- Gemini

Omni dbGo Features

Query Builder



d	awards	year	imdb	country
OG	[Row]	1913	[Row]	[List]
	[Row]	1909	[Row]	[List]
ROVED	[Row]	1916	[Row]	[List]
SED	[Row]	1920	[Row]	[List]
	[Row]	1920	[Row]	[List]
	[Row]	1922	[Row]	[List]
ROVED	[Row]	1924	[Row]	[List]
SED	[Row]	1924	[Row]	[List]
ROVED	[Row]	1925	[Row]	[List]
SED	[Row]	1926	[Row]	[List]
	[Row]	1928	[Row]	[List]



Filter

AND

year equals 1926

click or drop field add group

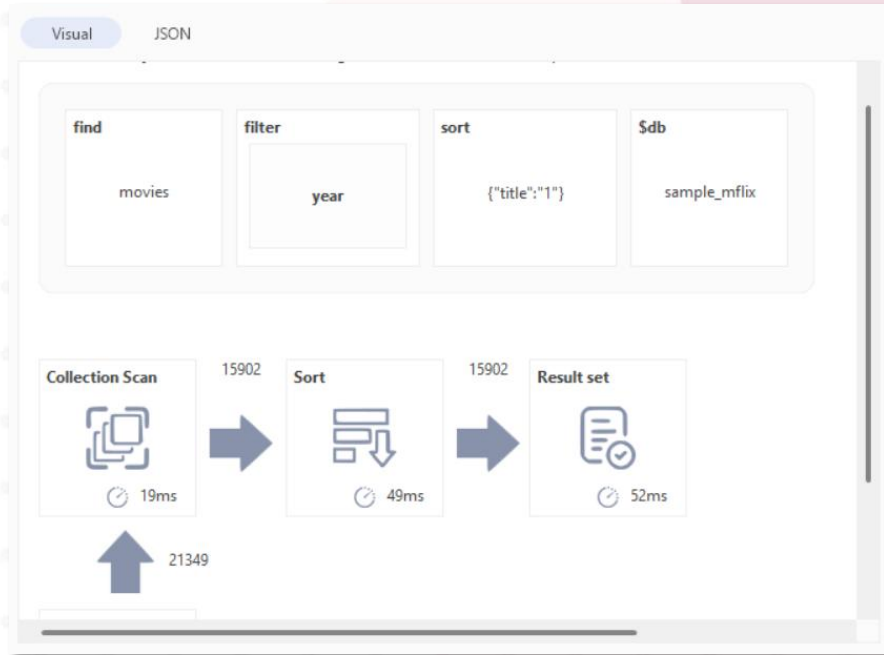
Daten abfragen

Verwendung eines **Filter-Panels** zur Erstellung von Abfragen:

- Hinzufügen einer oder mehrerer Bedingungen per Drag&drop
- Felder aus den Ergebnissen in den Filter-Panel ziehen und den Operator auswählen
- Felder im Ergebnis ein- oder ausschließen
- Ergebnis sortieren

Omni dbGo Features

Query Builder



Abfragen visualisieren

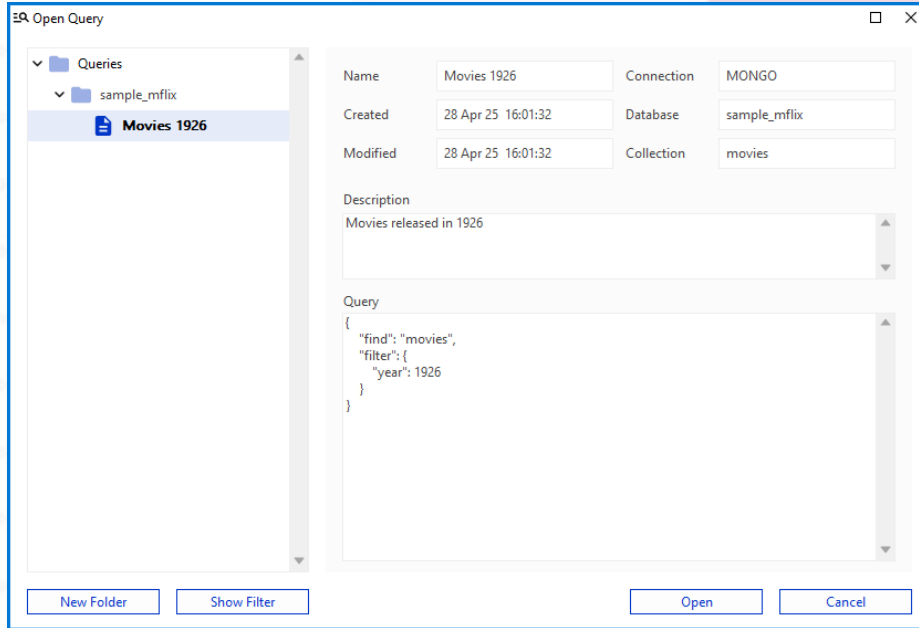
Visuelles **Explain-Tool**, das information zu Abfragen als Bild anzeigt:

- Ausführungsplan (wie MongoDB die Abfrage auflöst)
- Geschwindigkeit
- Anzahl gescannter und zurückgegebener Dokumente
- Verwendeter Index

Hilft dabei, Abfragen zu optimieren und die Performance zu verbessern.

OmniS dbGo Features

Query Manager



Verwaltung von Abfragen

- Abfragen speichern
- Zuvor gespeicherte Abfragen öffnen
- Gespeicherte Abfragen nach Stichworten durchsuchen
- Abfragen im JSON-Format exportieren

Omnis dbGo Features

Form Builder

The screenshot shows a web browser window with the address bar displaying "127.0.0.1:60094/jshtml/jsMovies.htm". The browser's address bar also shows "127.0.0.1:60094/jshtml/jsMovies.htm". The form contains the following fields:

- _id**: A text input field containing the value "573a1395f29313caabce26c4".
- plot**: A text input field containing the value "Repelled by the commercialism he sees around him, Charlie Brown tries to find the true meaning of Christmas."
- genres**: A list of text input fields containing the values "Animation", "Comedy", and "Family".
- runtime**: A text input field containing the value "25".
- cast**: A text input field containing the value "Ann Altieri".

At the bottom of the form, there are four buttons: "Update", "Insert", "Delete", and "Back".

Formulare erzeugen

- Erstellt Web- oder Desktop Applikation mit den Ergebnisdaten der Abfragen
- Erzeugt Library mit Window oder RemoteForm
- Generiert den Code
- Kann als Grundlage für weitere Entwicklung mit Omnis Studio genutzt werden

Versionen

dbGo Standard

- Standalone Version
- Für Kunden ohne Omnis Studio 11.2 Entwicklerlizenz
- Enthält alle Features, aber keine Möglichkeit zur Editierung oder Weiterentwicklung der erzeugten Form

dbGo Pro

- Add-on für Omnis Studio
- Für Kunden mit Omnis Studio 11.2 Entwicklerlizenz

Kostenlose, zeitlich begrenzte Demoversion verfügbar

Demo



Many thanks