|  |  |  |
| --- | --- | --- |
|  | **Univerzitet u Nišu, Elektronski fakultet**  **Katedra za računarstvo** |  |

**Seminarski rad**

**Interna struktura i organizacija skladišta podataka kod InfluxDB baze podataka**

Darko Jotev br. ind. 1727

**Predmet: Sistemi za upravljanje bazama podataka**

**Mentor: Doc. dr Aleksandar Stanimirović**

**Niš, 2025**

**Interna struktura i organizacija skladišta podataka kod InfluxDB baze podataka**

**Darko Jotev br. ind. 1727**

**Predmet:**

Sistemi za upravljanje bazama podataka

**Mentor:**

Doc. dr Aleksandar Stanimirović

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Sadržaj**

[1. Uvod 4](#_Toc208850542)

[1.1. Opis teme i cilj seminarskog rada 4](#_Toc208850543)

[2. Teorijska osnova 4](#_Toc208850544)

[2.1. Istorijat InfluxDB 4](#_Toc208850545)

[2.2. InfluxDB 3 Core 5](#_Toc208850546)

[2.3. Karakteristike InfluxDB 3 Core 5](#_Toc208850547)

[2.4. Instalacija InlfuxDB 3 Core 6](#_Toc208850548)

[2.5. Organizacija skladišta podataka 8](#_Toc208850549)

[2.6. Podešavanje InlfuxDB 3 Core 11](#_Toc208850550)

[2.6.1. Podešavanje autorizacije 13](#_Toc208850551)

[3. InfluxDB 3 Core interna struktura 14](#_Toc208850552)

[4. Praktični deo 16](#_Toc208850553)

[4.1.1. Pokretanje baze 16](#_Toc208850554)

[4.1.2. Podešavanje InfluxDB Explorera 16](#_Toc208850555)

[4.2.1.3. Upis podataka 17](#_Toc208850556)

[4.1.4. Pregled upisanih podataka 18](#_Toc208850557)

[5. Zaključak 19](#_Toc208850558)

[5. Literatura 20](#_Toc208850559)

# Uvod

## 1.1. Opis teme i cilj seminarskog rada

Ovaj seminarski rad bavi se detaljnom analizom interne strukture i organizacije skladišta podataka u InfluxDB 3 Core, NoSQL bazi podataka namenjenoj za čuvanje vremenskih serija i događajnih podataka. Rad će prikazati kako baza podataka interno organizuje podatke, uključujući strukturu datoteka, upotrebu objektnih skladišta i arhitekturu bez diska, koja omogućava efikasno čuvanje i pristup velikim količinama podataka.

Cilj rada je da kroz teorijski deo objasni osnovne principe i mehanizme organizacije podataka u InfluxDB 3 Core, dok će praktični primeri demonstrirati konkretan rad sa bazom – kreiranje serija, upis podataka, čuvanje i pristup podacima, kao i pregled interne strukture kroz alate baze. Ovaj pristup omogućava jasnije razumevanje kako InfluxDB 3 Core skladišti i organizuje podatke, što je od ključnog značaja za sve koji žele da proučavaju ili primenjuju ovu bazu u realnim scenarijima.

# 2. Teorijska osnova

## 2.1. Istorijat InfluxDB

**InfluxDB 1.x**

* Prva stabilna verzija InfluxDB-a, fokusirana na **vremenske serije podataka**.
* Arhitektura: jednostavna server-klijent, lokalno skladištenje u TSDB formatu.
* Podržava **line protocol**, tagove i polja, upite preko InfluxQL jezika.
* Retention policy (politika zadržavanja podataka) za automatsko brisanje starih podataka.
* Ograničena skalabilnost i visoka dostupnost kroz vanjske alate (npr. Kapacitor, Chronograf).

**InfluxDB 2.x**

* Reorganizacija i unapređenje arhitekture, uvodi **buckets** umesto klasičnih baza i retention policy-ja.
* Integracija **InfluxQL** i **Flux** jezika za upite i analizu podataka.
* Ugrađeni web interfejs (UI) za upravljanje bazom, upite i vizualizaciju podataka.
* Podrška za **token-based autorizaciju**, jednostavnija administracija i bolje skaliranje.
* Cilj: olakšati rad sa vremenskim serijama i omogućiti moderni DevOps pristup.

**InfluxDB 3 Core**

* Fokus na **real-time monitoring** i upravljanje aktuelnim podacima.
* Koristi **diskless arhitekturu** i objekt skladište (Parquet format), može raditi samo sa lokalnim diskom ili cloud objekt skladištem.
* Primarni ključ u tabelama: redosled tagova + vreme.
* Token-based autorizacija po defaultu; admin i operator tokeni.
* InfluxDB 3 Enterprise nadograđuje Core sa podrškom za istorijske podatke, read replike, visoku dostupnost i dodatne administrativne funkcije.

A diagram of a cloud

AI-generated content may be incorrect.

Slika 2.1. Istorijat InfluxDB-a

## 2.2. InfluxDB 3 Core

InfluxDB 3 Core je baza podataka kreirana za prikupljanje, obradu, transformaciju i čuvanje događaja i vremenskih serija podataka, i idealna je za slučajeve upotrebe koji zahtevaju obradu podataka u realnom vremenu i brze odgovore na upite radi izgradnje korisničkih interfejsa, nadzora i automatizovanih rešenja.

Uobičajeni slučajevi upotrebe uključuju:

* Praćenje podataka sa senzora
* Nadzor servera
* Praćenje performansi aplikacija
* Nadzor mreže
* Analitiku finansijskih tržišta i trgovanja
* Analitiku ponašanja korisnika

InfluxDB je optimizovan za scenarije gde je praćenje podataka gotovo u realnom vremenu ključno, a upiti moraju da se izvrše brzo kako bi podržali korisničko iskustvo poput kontrolnih tabli i interaktivnih korisničkih interfejsa.

## 2.3. Karakteristike InfluxDB 3 Core

InfluxDB 3 Core je open source verzija InfluxDB 3.

Glavne karakteristike Core verzije uključuju:

* Arhitektura bez diska sa podrškom za objektni storage (ili lokalni disk bez dodatnih zavisnosti)
* Brzi odgovori na upite (ispod 10ms za upite poslednje vrednosti, ili 30ms za upite sa jedinstvenim metapodacima)
* Ugrađeni Python VM za dodatke (plugins) i okidače (triggers)
* Čuvanje podataka u Parquet fajlovima
* Kompatibilnost sa InfluxDB 1.x i 2.x write API-jima

Enterprise verzija dodaje sledeće funkcionalnosti Core verziji:

* Mogućnost istorijskih upita i indeksiranje pojedinačnih serija
* Visoka dostupnost (High Availability)
* Read replike
* Poboljšana bezbednost (uskoro dostupno)
* Podrška za brisanje po nivou reda (uskoro dostupno)
* Integrisan administratorski UI (uskoro dostupno)

## 2.4. Instalacija InlfuxDB 3 Core

**Skladište objekata (Object storage):**

Ključna karakteristika InfluxDB 3 je upotreba skladišta objekata za čuvanje podataka vremenskih serija u Apache Parquet formatu. Možemo odabrati da ove fajlove čuvamo na svom lokalnom fajl sistemu. Performanse na lokalnom fajl sistemu će verovatno biti bolje, ali skladište objekata ima prednost u tome što ne može ostati bez prostora i može mu se pristupiti sa drugih sistema preko mreže. InfluxDB 3 Core nativno podržava Amazon S3, Azure Blob Storage i Google Cloud Storage. Takođe možemo koristiti mnoge lokalne implementacije skladišta objekata koje obezbeđuju S3-kompatibilan API, kao što je Minio.

**Instalacija**

InfluxDB 3 Core radi na Linux-u, macOS-u i Windows-u.

Da bismo instalirali InfluxDB 3 Core na Linux-u ili macOS-u, možemo preuzeti i pokrenuti skriptu za brzu instalaciju – na primer, koristeći **curl** za preuzimanje skripte:

curl -O https://www.influxdata.com/d/install\_influxdb3.sh \

&& sh install\_influxdb3.sh

Skripta za brzu instalaciju se ažurira sa svakim izdanjem InfluxDB 3 Core, tako da uvek instalira najnoviju verziju.

**Preuzimanje i instalacija najnovijih build artefakata**

Možemo takođe direktno preuzeti i instalirati build artefakte InfluxDB 3 Core:

* **Linux binary**
  + Linux | AMD64 (x86\_64) | GNU
  + Linux | ARM64 (AArch64) | GNU
* **macOS binary**
* **Windows binary**
  + Windows (AMD64, x86\_64)

**Pull Docker image-a**

Pokrenemo sledeću komandu da pull-ujemo **influxdb:3-core** image, dostupan za x86\_64 (AMD64) i ARM64 arhitekture:

docker pull influxdb:3-core

Docker automatski pull-uje odgovarajući image za arhitekturu našeg sistema.

**Povlačenje za određenu sistemsku arhitekturu**

**Provera instalacije**

Nakon instalacije InfluxDB 3 Core, unesemo sledeću komandu da proverimo da li je uspešno instaliran:

influxdb3 --version

**Alati za korišćenje**

Sledeća tabela upoređuje alate koje možete koristiti za interakciju sa InfluxDB 3 Core.

A screenshot of a computer

AI-generated content may be incorrect.

Slika 2.2. Alati za upravljanje InfluxDB

## 2.5. Organizacija skladišta podataka

InfluxDB 3 Core model podataka organizuje vremenske serije u baze podataka i tabele. Jedna baza podataka može da sadrži više tabela. Tabele sadrže više tagova i polja.

* **Baza podataka (Database):** Imenovana lokacija gde se čuvaju podaci vremenskih serija. U InfluxDB 3 Core, baza podataka je sinonim za *bucket* u InfluxDB Cloud Serverless i InfluxDB TSM implementacijama.

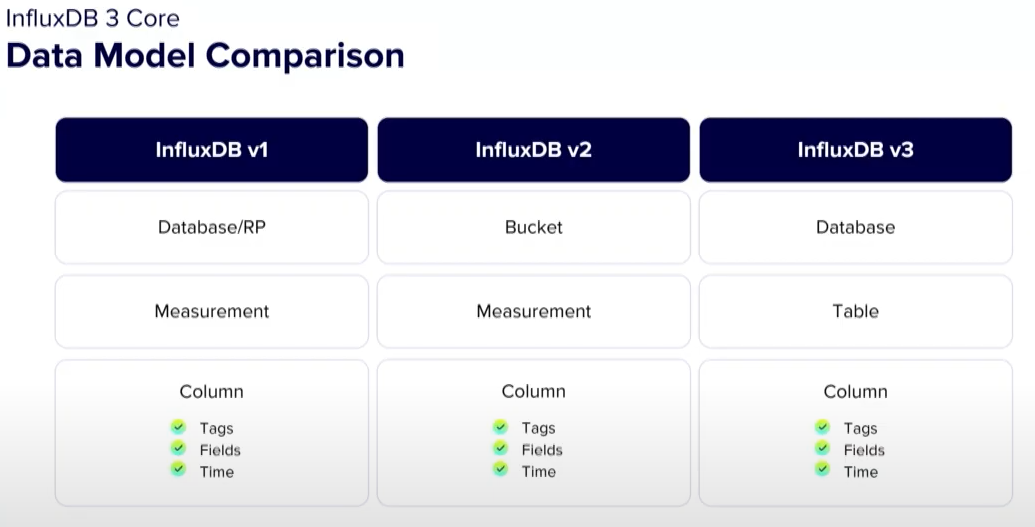
Jedna baza podataka može da sadrži više tabela.

* **Tabela (Table):** Logička grupa za podatke vremenskih serija. U InfluxDB 3 Core, tabela je sinonim za *measurement* u InfluxDB Cloud Serverless i InfluxDB TSM implementacijama. Sve tačke u jednoj tabeli treba da imaju iste tagove. Tabela sadrži više tagova i polja.
  + **Tagovi (Tags):** Ključ–vrednost parovi koji čuvaju string metapodatke za svaku tačku – na primer, vrednost koja identifikuje ili razlikuje izvor podataka ili kontekst, kao što su host, lokacija, stanica itd. Vrednosti tagova mogu biti *null*.
  + **Polja (Fields):** Ključ–vrednost parovi koji čuvaju same podatke za svaku tačku – na primer, temperaturu, pritisak, cenu akcije itd. Vrednosti polja mogu biti *null*, ali bar jedno polje u svakom redu mora imati vrednost koja nije *null*.
  + **Vremenska oznaka (Timestamp):** Vremenska oznaka povezana sa podacima. Kada se čuvaju na disku i kada se upituju, svi podaci su poređani po vremenu. U InfluxDB, vremenska oznaka je Unix timestamp u UTC formatu sa preciznošću na nanosekund. Vremenska oznaka nikada nije *null*.

**Promene u nomenklaturi kod InlfuxDB Core**

U ranijim verzijama InfluxDB, InfluxDB Cloud (TSM) ili InfluxDB Cloud Serverless, sretali smo pojmove *bucket* i *measurement*:

* *Bucket* u InfluxDB v2 ili InfluxDB Cloud Serverless je sinonim za bazu podataka (Database) u InfluxDB 3 Core.
* *Measurement* u InfluxDB v1, v2 ili InfluxDB Cloud Serverless je sinonim za tabelu (Table) u InfluxDB 3 Core.



Slika 2.3. Poređenje vezija InfluxDB

**Primarni ključevi**

Kod podataka vremenskih serija, primarni ključ za jedan red obično je kombinacija vremenske oznake i drugih atributa koji jedinstveno identifikuju tačku podataka. U InfluxDB 3 Core, primarni ključ reda je kombinacija vremenske oznake i seta tagova (skup ključeva i vrednosti tagova za datu tačku). Set tagova u primarnom ključu ne uključuje tagove čija je vrednost *null*.

**Model podataka**

Server InfluxDB 3 Core sadrži logičke baze podataka; baze podataka sadrže tabele; a tabele se sastoje od kolona.

Kolone u tabeli predstavljaju vreme, tagove i polja. Kolone mogu biti jednog od sledećih tipova:

* String dictionary (tag)
* int64 (field)
* float64 (field)
* uint64 (field)
* bool (field)
* string (field)
* timestamp (vreme sa nanosekundskom preciznošću)

U InfluxDB 3 Core svaka tabela ima primarni ključ – uređen skup tagova i vreme – za svoje podatke. Primarni ključ jedinstveno identifikuje svaki red i određuje redosled sortiranja za sve Parquet fajlove koji se odnose na tabelu. Kada kreirate tabelu, bilo eksplicitnim pozivom ili upisom podataka u tabelu prvi put, primarni ključ se postavlja na tagove redosledom kojim su stigli. Iako je InfluxDB i dalje baza tipa *schema-on-write*, definicije tag kolona za tabelu su nepromenljive.

Tagovi treba da sadrže jedinstvene identifikatore kao što su **sensor\_id**, **building\_id** ili **trace\_id**. Svi ostali podaci treba da se čuvaju kao polja (*fields*).

InfluxDB 3 mehanizam za skladištenje podržava beskonačan broj tag vrednosti i serija (cardinality). Za razliku od prethodnih verzija InfluxDB, kardinalnost tag vrednosti ne utiče na ukupne performanse baze podataka.

## 2.6. Podešavanje InlfuxDB 3 Core

**Preduslovi**

Da bi se započelo sa radom, potrebno je imati sledeće:

* **InfluxDB 3 Core**: Instalirati i proveriti najnoviju verziju na svom sistemu.

**Pokretanje InfluxDB-a**

InfluxDB 3 Core se pokreće pomoću komande:

influxdb3 serve

Prilikom pokretanja, treba obezbediti sledeće parametre:

* **--node-id**: Tekstualni identifikator koji razlikuje pojedinačne instance servera. Ovo čini poslednji deo putanje skladišta: <CONFIGURED\_PATH>/<NODE\_ID>.
* **--object-store**: Tip skladišta objekata koje će se koristiti. InfluxDB podržava sledeće tipove:
  + file: lokalni fajl sistem
  + memory: u RAM-u (bez trajnog čuvanja)
  + memory-throttled: slično memorijskom, ali sa kašnjenjem i propusnim opsegom koji podsećaju na cloud skladište
  + s3: AWS S3 i S3-kompatibilni servisi kao Ceph ili Minio
  + google: Google Cloud Storage
  + azure: Azure Blob Storage

Ostali parametri zavise od izabranog tipa skladišta. Na primer, ako se koristi S3, neophodno je navesti ime bucket-a i pristupne akreditive.

**Diskless arhitektura**

InfluxDB 3 podržava arhitekturu bez diska, koja može raditi isključivo sa skladištem objekata, čime se eliminiše potreba za lokalno priključenim diskovima. Takođe, InfluxDB 3 Core može raditi i samo sa lokalnim diskom kada je to potrebno.

U ovom vodiču, koristićemo file objekt store da bismo trajno čuvali podatke na lokalnom disku.

**Primer lokalnog skladišta objekata (file system)**

influxdb3 serve \

--node-id host01 \

--object-store file \

--data-dir ~/.influxdb3

**Docker sa montiranim file system objekt store-om**

Ako pokrećemo Docker imidž i želimo da podaci budu sačuvani na lokalnom fajl sistemu, potrebno je montirati volume:

docker run -it \

--volume /path/on/host:/path/in/container \

influxdb:3-core influxdb3 serve \

--node-id my\_host \

--object-store file \

--data-dir /path/in/container

Podrazumevani port za HTTP konekcije je **8181**. Mapiranje porta možemo prilagoditi pomoću opcije --http-bind ili promenljive okruženja INFLUXDB3\_HTTP\_BIND\_ADDR.

**Docker Compose primer**

Primer compose.yaml fajla za InfluxDB 3 Core:

services:

influxdb3-core:

image: influxdb:3-core

ports:

- 8181:8181

command:

- influxdb3

- serve

- --node-id=node0

- --object-store=file

- --data-dir=/var/lib/influxdb3/data

- --plugin-dir=/var/lib/influxdb3/plugins

volumes:

- type: bind

source: ~/.influxdb3/data

target: /var/lib/influxdb3/data

- type: bind

source: ~/.influxdb3/plugins

target: /var/lib/influxdb3/plugins

Pokretanje servera pomoću Docker Compose:

docker compose pull && docker compose up influxdb3-core

**S3 objekt store**

Za produkciona okruženja koja zahtevaju visoku dostupnost i trajnost podataka, koristi se S3-kompatibilno skladište. Potrebno je navesti ime bucket-a i pristupne akreditive:

influxdb3 serve \

--node-id host01 \

--object-store s3 \

--bucket OBJECT\_STORE\_BUCKET \

--aws-access-key AWS\_ACCESS\_KEY\_ID \

--aws-secret-access-key AWS\_SECRET\_ACCESS\_KEY

Za Minio ili druge open source S3-kompatibilne servise, dodaju se dodatni parametri:

influxdb3 serve \

--node-id host01 \

--object-store s3 \

--bucket OBJECT\_STORE\_BUCKET \

--aws-access-key-id AWS\_ACCESS\_KEY\_ID \

--aws-secret-access-key AWS\_SECRET\_ACCESS\_KEY \

--aws-endpoint ENDPOINT \

--aws-allow-http

**Memorijsko skladište objekata**

Podaci se čuvaju samo u RAM-u i ne traju nakon gašenja servera. Ovo je korisno za brzo testiranje i razvoj:

influxdb3 serve \

--node-id host01 \

--object-store memory

**Dodatne informacije**

Za više opcija i parametara servera, možemo koristiti CLI pomoć:

influxdb3 serve –help

Takođe, za vizuelno upravljanje bazama podataka, tokenima i pisanjem upita možemo koristiti **InfluxDB 3 Explorer**, web interfejs za upite i administraciju.

### 2.6.1. Podešavanje autorizacije

U InfluxDB 3 Core koristimo autorizaciju zasnovanu na tokenima kako bismo odobravali akcije u bazi podataka. Autorizacija je podrazumevano uključena kada pokrenemo server. Kada je autorizacija omogućena, moramo koristiti token prilikom izvršavanja influxdb3 CLI komandi ili HTTP API zahteva.

InfluxDB 3 Core podržava **admin tokene**, koji nam omogućavaju pristup svim CLI komandama i API endpoint-ima.

**Kreiranje operator tokena**

Nakon što pokrenemo server, kreiramo prvi admin token. Taj prvi admin token postaje naš **operator token** za server.

Komanda koju koristimo za kreiranje operator tokena je:

influxdb3 create token --admin

Ova komanda nam vraća token u obliku stringa, koji koristimo za autentifikaciju CLI komandi i API zahteva.

**Sigurno čuvanje tokena**

InfluxDB prikazuje token **samo kada ga kreiramo**. Zato ga moramo sačuvati na sigurnom mestu, jer kasnije ne možemo da ga preuzmemo iz baze podataka.

**Korišćenje tokena za autorizaciju**

Koristimo operator token da bismo se autentifikovali prilikom administrativnih zadataka, upisivanja i čitanja podataka.

Token možemo proslediti na jedan od sledećih načina:

1. **Kroz promenljivu okruženja** (preporučeno)
2. **Kroz opciju komande --token**

Primer korišćenja tokena sa CLI komandom:

influxdb3 show databases --token YOUR\_AUTH\_TOKEN

U komandi, **YOUR\_AUTH\_TOKEN** zamenjujemo stvarnim tokenom koji smo kreirali kao operator token.

# InfluxDB 3 Core interna struktura

Kada upisujemo podatke u InfluxDB 3 Core, oni prolaze kroz više faza kako bi se osigurala trajnost, optimizovale performanse i omogućilo efikasno izvršavanje upita. Opcije konfiguracije u svakoj fazi utiču na ponašanje sistema, balansirajući pouzdanost i korišćenje resursa.

**Tok podataka prilikom upisa**

Kada se podaci upisuju u InfluxDB 3 Core, oni slede strukturisan put kako bi se obezbedila trajnost, efikasni upiti i optimizovano skladištenje.

A screenshot of a computer flowchart

AI-generated content may be incorrect.

Slika 3.1. Tok upisa, odgovora i ingest-a u InfluxDB 3 Core i Enterprise

Glavne faze uključuju:

1. Validacija upisa i memorijski bafer
2. Persistencija kroz Write-Ahead Log (WAL)
3. Dostupnost za upite
4. Skladištenje u Parquet formatu
5. Keš u memoriji

**Validacija upisa i memorijski bafer (Write validation and memory buffer)**

**Proces:** InfluxDB proverava dolazne podatke pre nego što ih prihvati u sistem.

**Uticaj:** Sprečava da nepravilni ili nepodržani podaci uđu u bazu.

**Detalji:** Baza proverava dolazne podatke i čuva ih u memorijskom **write buffer-u**. Ako je no\_sync=true, server odmah šalje potvrdu o upisu.

**Persistencija kroz Write-Ahead Log (Write-ahead log (WAL) persistence)**

**Proces:** Baza upisuje podatke iz memorijskog bafera u WAL svakog sekunda (podrazumevano).

**Uticaj:** Obezbeđuje trajnost podataka upisom u objekt skladište.Kompromis: Češće upisivanje povećava trajnost, ali povećava i I/O opterećenje.

**Detalji:** Svakog sekunda, podaci iz write buffer-a se upisuju u WAL radi trajnog čuvanja u objekt skladištu. Ako je no\_sync=false (podrazumevano), server šalje potvrdu o upisu.

**Dostupnost za upite (Query availability)**

**Proces:** Podaci se nakon WAL persistencije premještaju u **queryable buffer**.Uticaj: Omogućava brze upite nad najnovijim podacima.

**Kompromis:** Veći bafer ubrzava upite, ali povećava korišćenje memorije.

**Detalji:** Nakon završetka persistencije u WAL, podaci postaju dostupni za upite u queryable buffer-u. Po podrazumevanoj vrednosti, server čuva do 900 WAL fajlova (što odgovara 15 minuta podataka) u baferu.

**Skladištenje u Parquet formatu (Parquet storage)**

**Proces:** Podaci se svakih deset minuta (podrazumevano) trajno čuvaju u Parquet fajlovima u objekt skladištu.

**Uticaj:** Omogućava dugoročno i trajno čuvanje podataka.Kompromis: Češća persistencija smanjuje zavisnost od WAL-a, ali povećava I/O troškove.

**Detalji:** Svakih deset minuta, InfluxDB 3 Core upisuje najstarije podatke iz queryable buffer-a u Parquet format u objekt skladištu, dok najnovijih pet minuta podataka ostaje u memoriji.

**Keš u memoriji (In-memory cache)**

**Proces:** Nedavno sačuvani Parquet fajlovi se keširaju u memoriji.

**Uticaj:** Smanjuje kašnjenje pri upitima jer se ne mora odmah pristupati objekt skladištu.

**Detalji:** InfluxDB 3 Core smešta Parquet fajlove u **in-memory cache**, tako da upiti nad najnovijim podacima ne moraju ići u objekt skladište.

# Praktični deo

4.1.1. Pokretanje baze  
Bazu pokrećemo na sledeći način koristeći cmd:

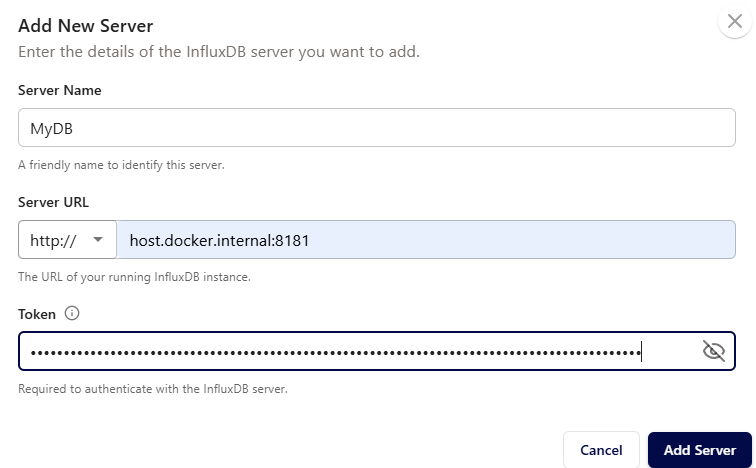
.\influxdb3.exe serve --node-id host01 --object-store file --data-dir ~/.influxdb3

InfluxDB server se pokreće lokalno na portu 8181.

### 4.1.2. Podešavanje InfluxDB Explorera

Već imamo token koji smo generisali i prelazimo na InfluxDB 3 Explorer.

InfluxDB Exploreru pristupamo lokalno na portu 8888 i podešavamo detalje vezane za InfluxDB server:



Slika 4.1. Podesavanje InfluxDB Explorer-a

### 4.2.1.3. Upis podataka

Podatke upisujemo u bazu izvršavanjem sledeće komande:

influxdb3 write \

--token apiv3\_L0QIamllMG5Kg-3KHzXeNvmJoQcJGLU1ldvqt8qOiYVox3D1PUlxbcEdVcUlA9Pw2APKeZHHma1XNRo9GyVw \

--database AirTempDB \

'home,room=Living\ Room temp=21.1,hum=35.9,co=0i 1757836800

home,room=Kitchen temp=21.0,hum=35.9,co=0i 1757836800

home,room=Living\ Room temp=21.4,hum=35.9,co=0i 1757840400

home,room=Kitchen temp=23.0,hum=36.2,co=0i 1757840400

home,room=Living\ Room temp=21.8,hum=36.0,co=0i 1757844000

home,room=Kitchen temp=22.7,hum=36.1,co=0i 1757844000

home,room=Living\ Room temp=22.2,hum=36.0,co=0i 1757847600

home,room=Kitchen temp=22.4,hum=36.0,co=0i 1757847600

home,room=Living\ Room temp=22.2,hum=35.9,co=0i 1757851200

home,room=Kitchen temp=22.5,hum=36.0,co=0i 1757851200

home,room=Living\ Room temp=22.4,hum=36.0,co=0i 1757854800

home,room=Kitchen temp=22.8,hum=36.5,co=1i 1757854800

home,room=Living\ Room temp=22.3,hum=36.1,co=0i 1757858400

home,room=Kitchen temp=22.8,hum=36.3,co=1i 1757858400

home,room=Living\ Room temp=22.3,hum=36.1,co=1i 1757862000

home,room=Kitchen temp=22.7,hum=36.2,co=3i 1757862000

home,room=Living\ Room temp=22.4,hum=36.0,co=4i 1757865600

home,room=Kitchen temp=22.4,hum=36.0,co=7i 1757865600

home,room=Living\ Room temp=22.6,hum=35.9,co=5i 1757869200

home,room=Kitchen temp=22.7,hum=36.0,co=9i 1757869200

home,room=Living\ Room temp=22.8,hum=36.2,co=9i 1757872800

home,room=Kitchen temp=23.3,hum=36.9,co=18i 1757872800

home,room=Living\ Room temp=22.5,hum=36.3,co=14i 1757876400

home,room=Kitchen temp=23.1,hum=36.6,co=22i 1757876400

home,room=Living\ Room temp=22.2,hum=36.4,co=17i 1757880000

home,room=Kitchen temp=22.7,hum=36.5,co=26i 1757880000'

Uključuje simulirane podatke kućnih senzora na satnom nivou sa anomalnim očitavanjima senzora, radi demonstracije obrade i alarmiranja nad vremenskim serijama podataka.

Vremenski opseg: 2025-09-14T08:00:00Z do 2025-09-14T20:00:00Z

Šema:

home (tabela)

tagovi (Tags):

* room
  + Kitchen
  + Living Room

Polja (Fields):

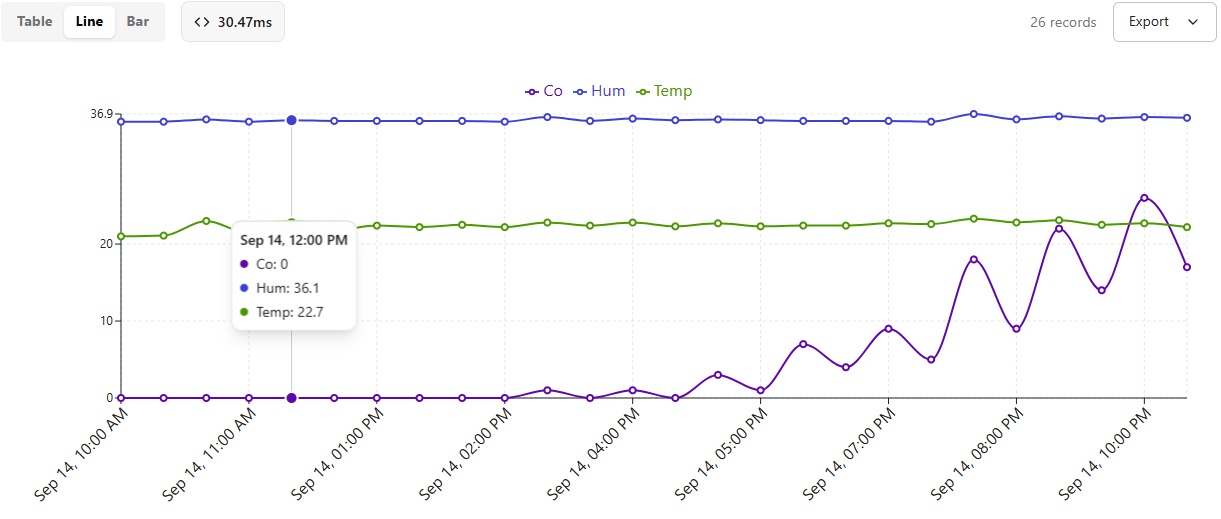
* co (integer)
* temp (float)
* hum (float)

### 4.1.4. Pregled upisanih podataka

Korišćenjem alata InlfuxDB 3 Explorer možemo da vidimo upisane podatke:  
  
A screenshot of a computer

AI-generated content may be incorrect.

Slika 4.2. Pregled upisanih podataka

Ili drugačije možemo promeniti prikaz da ne bude tabelarno, nego da nam prikaze grafik:  
  


Slika 4.3. Grafik na osnovu upisanih podataka

# 5. Zaključak

InfluxDB 3 Core predstavlja savremeno i moćno rešenje za rad sa vremenskim serijama podataka, koje kombinuje diskless arhitekturu, objekatno skladište i Parquet format kako bi obezbedilo efikasno, pouzdano i skalabilno upravljanje podacima. Interna organizacija zasnovana na bazama, tabelama, tagovima, poljima i vremenskim oznakama jasno pokazuje kako se podaci logički i fizički strukturiraju da bi omogućili brze upite i minimalno opterećenje sistema.

Kroz analizu procesa obrade podataka – od validacije i upisa u WAL, preko trajnog skladištenja, do keširanja i dostupnosti za upite – jasno je da je InfluxDB 3 Core optimizovan za aplikacije koje zahtevaju gotovo trenutni odziv i visoku pouzdanost.

Praktični deo rada demonstrirao je jednostavnost instalacije, podešavanja i upotrebe alata poput InfluxDB Explorera, što dodatno potvrđuje da je baza pogodna ne samo za kompleksna produkciona okruženja, već i za razvojne scenarije i edukaciju.

Može se zaključiti da InfluxDB 3 Core donosi značajan iskorak u odnosu na prethodne verzije, jer omogućava veću fleksibilnost, skalabilnost i integraciju sa modernim cloud okruženjima. Razumevanje interne strukture i načina organizacije skladišta podataka u ovoj bazi od suštinske je važnosti za projektovanje i implementaciju pouzdanih sistema za obradu vremenskih serija u realnom vremenu.

# 5. Literatura

1. InfluxData Documentation: <https://docs.influxdata.com/influxdb3/core/>
2. <https://docs.influxdata.com/influxdb3/core/get-started/>
3. [Inside InfluxDB 3 Core from the Creator Himself: Rust Rewrite, Object Storage, and More | GrafanaCON - YouTube](https://www.youtube.com/watch?v=mKu1PIhl2Ms)
4. <https://docs.influxdata.com/influxdb3/core/get-started/setup/>
5. <https://docs.influxdata.com/influxdb3/core/reference/internals/>
6. <https://docs.influxdata.com/influxdb3/core/reference/internals/durability/>