

Izkušnje uporabe podatkovnih zbirk časovnih vrst v Elektro Ljubljana

Tomaž Dolenc, Daniela Maksimović

Elektro Ljubljana, Ljubljana, Slovenija

tomaz.dolenc@elektro-ljubljana.si, daniela.maksimovic@elektro-ljubljana.si

V zadnjem desetletju smo priča izjemnemu razvoju tehnologij interneta stvari (Internet of Things – IoT). Uporaba IoT naprav in pametnih števec je v celotnem energetskega sektorju (in širše) povzročila ekspanzijo podatkov, ki se jih lahko pridobiva v skoraj realnem času iz različnih merilnih naprav, ki so na voljo in so instalirani neposredno pri končnih odjemalcih in/ali na samem omrežju oziroma v našem primeru na distribucijskem energetskega omrežju DO, ki je v upravljanju na Elektro Ljubljana. Infrastrukture za zbiranje, prenos, shranjevanje in obdelavo ogromnih količin podatkov se spreminja z uvedbo novih tehnologij na področju vele podatkov in sledi tehnološkim trendov. Na Elektro Ljubljana smo primerjali različne odprtokodne podatkovne zbirke časovnih vrst in primerjali njihove lastnosti. V tem članku bomo predstavili naše izkušnje pri zbiranju, prenosu, shranjevanju in obdelavi ogromnih količin podatkov iz merilnih naprav v našem omrežju. S pomočjo primerjalnih testiranj smo pokazali, da je ClickHouse ena najhitrejših analitičnih baz podatkov na trgu, pri čemer se je v realnih primerih uporabe skozi leta pokazalo znatno izboljšanje zmogljivosti pri izvajanju tipičnih poizvedb.

Ključne besede:

časovne serije,
meritve napetosti in energij,
Clickhouse,
TimescaleDB,
Elektro Ljubljana.

1 Uvod

V zadnjih letih je tehnološki napredek na področju interneta stvari (Internet of Things – IoT) bistveno spremenil način delovanja sodobnih elektroenergetskih sistemov. Sodobni elektroenergetski sistemi se vse bolj naslanjajo na digitalne tehnologije za zbiranje, prenos in obdelavo podatkov, pri čemer igrajo ključno vlogo pametni števeci in druge IoT naprave, ki so nameščene bodisi pri končnih odjemalcih bodisi neposredno v omrežju [1,2].

Uvedba teh tehnologij je povzročila eksponentno rast količine operativnih podatkov, ki jih je mogoče zbirati v skoraj realnem času. Ti podatki vključujejo informacije o trenutni porabi električne energije, napetostnih profilih, kakovosti oskrbe in drugih ključnih parametrih, ki so bistveni za učinkovito in zanesljivo delovanje distribucijskega omrežja [3]. V primeru distribucijskega omrežja (DO), ki je v upravljanju podjetja Elektro Ljubljana, to pomeni celovito spremljanje stanja omrežja, identifikacijo motenj, optimizacijo obratovanja ter boljše načrtovanje investicij in vzdrževalnih del.

Povezovanje merilnih naprav v pametna omrežja (angl. smart grids) omogoča prehod iz reaktivnega v proaktivno upravljanje, saj operaterji omrežij pridobijo možnost sprotnega prilagajanja obratovalnih parametrov glede na zaznane razmere v omrežju [4]. Tako se zmanjšujejo izgube v omrežju, izboljšuje kakovost oskrbe in povečuje energetska učinkovitost, kar je ključno v kontekstu prehoda na nizkoogljično družbo ter vključevanja obnovljivih virov energije.

Z vidika informacijsko-komunikacijskih tehnologij (IKT) je izziv predvsem v zagotavljanju ustrezne infrastrukture za zbiranje, prenos, shranjevanje in obdelavo ogromnih količin podatkov, kar zahteva uporabo naprednih metod strojnega učenja, podatkovnega rudarjenja ter orodij za analitiko velikih podatkov (big data) [5]. IoT naprave so tako postale ključni element digitalne transformacije energetskega sektorja in predstavljajo osnovo za nadaljnji razvoj inteligentnih in odzivnih elektroenergetskih sistemov.

2 Baze časovnih serij na Elektro Ljubljana

Zadnjih 10 let na trgu postajajo vse bolj popularne time-series podatkovne baze, kot ne-relacijske podatkovne baze ali celo kot kombinacija in združitev relacijskih in ne-relacijskih baz pri čemer so na voljo vse funkcionalnosti od obeh baz in omogočajo za uporabnika dostop do podatkov na že ustaljen standarden način. V ta namen smo preverili delovanje in performanse ter primerjali med seboj timeseries baze ClickHouse in TimescaleDB [6].

2.1. Clickhouse baza časovnih serij

Clickhouse baza je odprtokodna OLAP podatkovna baza, zasnovana za visoko zmogljivo analitiko nad podatkovnimi nabori v petabajtnem obsegu z visokimi hitrostmi vnosa podatkov. Njen sloj za shranjevanje združuje podatkovni format, ki temelji na tradicionalnih drevesih z zlivanjem dnevniških struktur (LSM – log-structured merge) z inovativnimi tehnikami za neprekinjeno preoblikovanje (npr. agregacija, arhiviranje) zgodovinskih podatkov v ozadju. Poizvedbe so zapisane v priročnem SQL dialektu in jih obdeluje najmodnejši vektorizirani mehanizem za izvajanje poizvedb, z možnostjo kompilacije kode.

Z arhitekturnega vidika baze podatkov sestavljata (vsaj) sloj za shranjevanje podatkov in sloj za obdelavo poizvedb. Medtem ko je sloj za shranjevanje odgovoren za shranjevanje, nalaganje in vzdrževanje podatkov v tabelah, je sloj za obdelavo poizvedb zadolžen za izvajanje uporabniških poizvedb. V primerjavi z drugimi bazami podatkov ClickHouse uvaja inovacije v obeh slojih, kar omogoča izjemno hitre vstavljanja podatkov (inserts) ter poizvedbe (Select queries).

Clickhouse je posebej prilagojena za delo z velepodatki. Ima številne prilagoditve, na primer lahko uporabljamo standardni SQL, lahko pa uporabimo tudi funkcije podatkovne zbirke, ki pohitrijo izvajanje SQL-a. S stališča

skrbništva podatkovne zbirke je upravljanje poenostavljeno, saj sama podatkovna zbirka skrbi za partitioniranje tabel, stiskanje podatkov v tabelah in pa za samodejno odstranitev podatkov, ko le-ti niso več potrebni. Podatkovna zbirka omogoča številne prilagoditve in optimiranja, kar je opisano v nadaljevanju.

ClickHouse agresivno uporablja tehnike obrezovanja (pruning), da se izogne obdelavi nepomembnih podatkov med izvajanjem poizvedb. Druge sisteme za upravljanje podatkov je mogoče vključiti na ravni funkcij tabel, mehanizmov tabel ali celo na ravni mehanizmov celotnih podatkovnih baz.

Na zmogljivost baze podatkov poleg usmerjenosti podatkov vpliva še mnogo drugih dejavnikov. V nadaljevanju bomo podrobneje razložili, zakaj je ClickHouse tako hiter, zlasti v primerjavi z drugimi stolpično usmerjenimi podatkovnimi bazami.

2.2. TimescaleDB

TimescaleDB je razširitev baze PostgreSQL, z enostavno namestitvijo in poznanim načinom uporabe. Je odprtokodna baza podatkov, ki uporablja standardni poizvedovalni jezik SQL. Izkorišča številne prednosti PostgreSQL, kot so zanesljivost, varnost in povezljivost s široko paleto orodij drugih proizvajalcev (številni različni programski jeziki, za katera so na voljo gonilniki, ; številne podprte platforme, Linux, Windows, OS X, AIX, Solaris, itd.). Izkazuje se kot izjemno zanesljiva in učinkovita rešitev za poizvedovanje nad ogromnimi časovnimi vrstami, hkrati pa jo je moč porazdeliti med več strežnikov in zagotoviti hitro poizvedovanje [6].

TimescaleDB je bila najprej naša prva izbira časovne baze. Največjo prednost smo videli v tem da se podatki samodejno razdelijo po času in prostoru (čas + druga dimenzija npr. naprava, uporabnik). Zato smo razdelili podatke po času in po različnih IoT naprav (števcu, ki merijo električno energijo ter druge merilne krmilne naprave. Medtem ko so podatki znotraj same baze razporejeni optimalno v t.i. Hipersegmentirane tabele (Hypertables) uporabnik tega ne opazi (ne rabi se prilagoditi) dela kot s "navadno" tabelo in piše navadne SQL poizvedbe, ki imajo odlično zmogljivost za časovne poizvedbe. Prav tako tudi insert podatkov je dokaj hiter in učinkovit v koliko se pravilno nastavi in uporablja optimizirani indeksi za časovno razvrščene podatke.

3 Karakteristične operacije z podatkovnimi zbirkami časovnih vrst

3.1. Polnjenje podatkovne zbirke

Operacija polnjenje podatkovne baze je ena osnovnih operacij, ki jo izvajamo ob uporabi podatkovnih zbirk podatkov. V našem primeru podatki so specifični sami po sebi ker niso navadni relacijsko povezani podatki ali transakcijski podatki tem več so podatki časovnih vrst oz. podatki so v zaporedju, ki so zabeležene ali opazovane v določenih časovnih intervalih. Vsaka podatkovna točka ima časovni žig (timestamp), ki označuje, kdaj je bila izmerjena.

Pomembno je razumevanje arhitekture procesa nastajanja in obdelave podatkov v podatkovni zbirki. To posebej velja ko so podatki časovne serije in si med seboj sledijo. Dodatno narava podatkov v našem primeru ko gre za meritve električne energije (meritve energije in napetosti) iz IoT naprav je taka, da se jih prepíše/dopolni vsak dan.

Proces polnjenja podatkovne zbirke se prilagodi izvornemu sistemu (na Elektro Ljubljana ostajata 2 glavna vira merilnih podatkov) ter velja za standarden način prenosa podatkov med sistemi (ETL). Vzpostavljen ETL proces vsebuje tudi napredne validacije podatkov ter analize podatkov v času prenašanja ('on the fly') do, pod pogojem, da pri tem nista motena pretok podatkov oziroma ni zmanjšana hitrost samega pretoka. Dodatno si pri tem želimo uporabiti čim več naprednih funkcij, ki se nanašajo na časovnih vrst in omogočajo dopolnitev manjkajočih podatkov z vnaprej določenimi vrednostmi (z uporabo pravil za nadomeščanje, ki jih bomo dobili iz vsebinskih poznavalcev/uporabnikov izvornega sistema, na podlagi statistični analiz podatkov iz prejšnjih nekaj let ter na podlagi algoritmov za strojno učenje, ki se nanašajo na dopolnitev manjkajočih podatkov).

Schema Name: measurements						
Tables	Table Name	Table Type	Schema	Table size	Row Count	Engine
Views	currents	TABLE	measurements	16G	3,276,290,169	MergeTree
Procedures	energies	TABLE	measurements	326G	64,382,171,405	MergeTree
Data Types	MerilnaMestaLPG	TABLE	measurements	5M	350,160	MergeTree
	mp_serialNo	TABLE	measurements	13M	850,257	MergeTree
	voltages	TABLE	measurements	256G	43,667,651,693	MergeTree

Slika 1: Velikost ClickHouse baze.

Na Sliki 1 za ilustracijo pokažemo velikost in organiziranost baze časovnih serij na določen dan na Elektro Ljubljana. Velikost baze se spreminja oz. narašča vsak dan. Podatki so grupirani po tipih meritev, ki jih IoT naprave beležijo na terenu.

V našem primeru ko smo delali inicialne load podatkov v obeh bazah, ki imajo enako sistemsko arhitekturo je bil ClickHouse tri krat hitrejši v eno letnem zapisu podatkov meritev energije in napetosti za vsa (cca. 350 000) merilna mesta na Elektro Ljubljana.

Glede celotne zasedenosti prostora na disku tega ranga podatkov je bistvena prednost ponovno bila tudi na strani ClickHouse-a.

3.2. Obremenitev podatkovne zbirke

Tudi če imamo zelo zmogljiv računalnik, ne moremo s preprosto akcijo zmeriti in/ali pospešiti odzivnost podatkovne zbirke. Hitrost oziroma odzivnost podatkovne zbirke je omejena z drugimi dejavniki, kot so hitrost shranjevanja, omrežna pasovna širina ali zasnova same podatkovne zbirke. Če se pričakuje, da se bo odzivnost podatkovne zbirke povečala za n -krat, potem je potreba temeljita analiza vsega kar se dogaja v podatkovni zbirki. V primeru relacijskih baz te že ponujajo svoja lastna orodja s pomočjo katerih se lahko spremlja in analizira obremenitev podatkovnih zbirk. To žal ne velja za bazah časovnih serij in to je analiza prepuščena končnemu uporabniku ter odvisna od njegovih scenarijev uporabe podatkov in načina pisana SQL poizvedb. Pri uporabi smo se omejili in dva »use case« in zanj posledično načrtovali in optimizirali vse operacije nad podatki: poizvedovanje vseh meritev za posamezno merilno mesto ter poizvedovanje veh meritev za manjšo/večjo skupino merilnih mest združeno. Agregacije podatkov in/ali agregatne operacije nad podatkih niso naši najbolj pogosti scenarije uporabe.

3.3. Odstranitve podatkov

Zaradi prostorske omejitve moramo poskrbeti za brisanje/uničenje podatkov in zato je bilo potrebno dobro razmisliti, kako se bodo podatki ustrezno odstranili iz podatkovne zbirke. Sprva smo načrtovali, da bi ročno odstranjevali podatke po časovnih obdobjih, ko jih ne bomo rabili več ampak v izogib ročnemu delu smo vzpostavili avtomatski mehanizem, ki sem briše podatke, ko so te starejše od določenega datuma.

3.4. Sprememba podatkov

Zaradi izvajanja sprememb podatkov, kar odstopa od običajnih praks v okviru časovnih vrst, nismo uspeli najti ustreznih spletnih virov za uspešno izvedbo procesa sprememb (update) podatkov. Zato smo bili primorani izvesti testiranja delovanja podatkovne zbirke, da bi prepoznali ustrezne, neoptimalne in neprimerne funkcionalnosti. Proces spreminjanja podatkov smo začeli tako, da smo najprej pobrisali podatke, ki smo jih morali spremeniti in jih ponovno vstavili v podatkovno zbirko. Ta metoda je dokaj hitra ker jo omejimo, na časovni okvir v katerega se

zgodijo spremembe vhodnih podatkov in sicer 40 dni za nazaj. Metoda, ni optimalna ne tehnično optimizirana ampak v našem primeru zadošča glede načina uporabe podatkovne zbirke in je v skladu z izvirnim sistemom na Elektro Ljubljana, ki je vir za naše podatke časovnih vrst.

3.5. Poizvedovanje in število povezav do podatkovne zbirke

Uporabe podatkovne zbirke je razvidna preko uporabe poizvedb v specifičnem časovnem okvirju. Število povezav je odvisno od števila posameznih uporabnikov oz. poizvedb, ki se izvedejo teh od njihove kompleksnosti in trajanja.

Največkrat poizvedovanje se zgodi po potrebi in sicer tako da ga sproži uporabnik sam (on demand oz. ad-hoc). Običajno je govora o poizvedovanju večkrat na dan za eno specifično merilno mesto ali manjše/večje skupine merilnih mest, ki imajo določene skupne lastnosti. Pri večjih na primer pri letnih oz. večletnih poizvedb na majhno skupino merilnih mest običajno gre za poizvedovanje, ki je specifično/optimizirano za časovne vrste.

Na primer ko poizvedujemo meritve za število dni (1000) in skupino merilnih mest MM (100) in izračunamo število podatkov, ki jih pri povpraševanju lahko dobimo v matriki merilnih podatkov (glede na 15 min frekvenco podatkov tokov in energije):

$$(1000 \times 96 \times 100) \times 4 + (1000 \times 120 \times 100) \times 3 = 100.000 \times (4 \times 96 + 120 \times 3) = 74.400.000$$

kar pomeni poizvedba po 74,4 milijonov podatkov. Tako poizvedovanje v klasičnih relacijskih baz običajno traja neprimerno dolgo.

Testi so ponovno pokazali večjo prednost na strani ClickHouse v povprečju je čas trajanje poizvedb bil za 3.5 krat hitrejši na ClickHouse kot na TimescaleDB za enak obseg podatkov, ki so v bazah shranjene v podobni strukturi, ki izkorišča posamezne bazne karakteristike.

V primeru ko hočemo pisati kompleksne poizvedbe, ki grupirajo podatke in izvajajo agregat operacije nad njimi pa hitro TimescaleDB pridobi v prednosti. Prednost je večja takoj ko se začne z uporabo specifičnih funkcij za TimescaleDB kot so `time_bucket`, `last`, `first` v kombinaciji z relacijskimi funkcijami.

3.6. Zaklepanja na podatkovni zbirki

Pri večjih spremembah podatkov v podatkovni zbirki se lahko hitro zgodi, da prihaja do zaklepanja objektov podatkovne zbirke. V standardnih relacijskih bazah se to pogosto zgodi, na primer če se posamezni SQL-i izvajajo več kot nekaj milisekund. Dlje traja posamezen SQL, večja je verjetnost, da bo prišlo do medsebojnega zaklepanja objektov na podatkovni zbirki.

ClickHouse nima vidno na uporabnika zaklepanje. Uporablja asinhrono operacije i notranje mehanizme za konkurentno dostope.

TimescaleDB ima to prednost nad PostgreSQL da uporablja hypertable, ki so narejeni da izvajajo particioniranje in s tem v osnovi preprečijo zaklepanje.

3.7. Particioniranje

Največja tabela v našem okolju je velika 800 GB (130 bilijon zapisov), kar predstavlja podatke energij za vseh merilnih mest na Elektro Ljubljana za obdobje pet pol. Vsak dan v podatkovni zbirki nastane dodatne 150 milijonov zapisov. ClickHouse uporablja MergeTree engine in s tem zagotavlja particioniranje. To je proces, kjer logično tabelo razdelimo na več fizičnih tabel. S stališča aplikacije obstaja ena tabela, fizično pa jih je lahko več

tisoč. Pomembno je, da pravilno nastavimo »particioniranje« tabele, ki se pri podatkih časovne vrste praviloma nastavi na časovno značko nastanka dogodka.

TimescaleDB nujno rabi, da niso ti podatki v eni sami veliki tabeli, ker bi SQL-i delovali občutno prepočasi ter nujno rabi dodatne mehanizme. Zato se za vsako tabelo ustvari indeks za pohitritev iskanja podatkov in ta indeks ne sme biti prevelik, da bi deloval optimalno. Posamezna TimescaleDB particionirana tabela je pomembno da ni kljub temu prevelika. Potrebno je izračunati priporočene velikosti particionirane tabele glede na frekvence polnjenja podatkov. Uporabili smo razmejitev tako, da so za vsak dan podatki v svoji partijski tabeli in na podlagi tipa meritev (energija ali napetost).

3.8. Indeksiranje

Z vidika uporabnika TimescaleDB izpostavlja objekte, ki izgledajo kot klasične tabele, imenujejo pa se hipertabele (ang. hypertables). Pravzaprav so te abstrakcije ali virtualni pogled na številne manjše tabele, ki vsebujejo podatke, imenovane kosi (ang. chunks). Kosi so ustvarjeni z razdelitvijo podatkov hipertabele na eno ali več dimenzij. Vse hipertabele so razdeljene glede na časovni interval, dodatno pa jih lahko delimo še z izbiro ključa, npr. ID naprave, lokacija, ID uporabnika, itd. Temu rečemo porazdelitev glede na čas in prostor (ang. Partitioning across time and space).

ClickHouse uporabljajo malo drugačen pristop baziran na mehanizmov MergeTree, ki so bili zasnovani in optimizirani za obdelavo ogromnih količin podatkov. Podatki se hitro zapisujejo v tabelo po chunkih, pri čemer se v ozadju uporabljajo pravila za združevanje chunk (delov). V ClickHouseu ima vsak chunk svoj primarni indeks. Ko se chunk združijo, se združijo tudi primarni indeksi združenih chunkov. Pri zelo velikem obsegu, za katerega je zasnovan ClickHouse, je ključnega pomena velika učinkovitost diska in pomnilnika. Zato namesto indeksiranja vsake vrstice ima primarni indeks za del en indeksni vnos (znan kot »oznaka«) na skupino vrstic (imenovano »granula«) – ta tehnika se imenuje redko indeksiranje.

TimescaleDB pridobi znatno z uporabo indeksov medtem ko (presenetljivo) ClickHouse zgubi pri performansah po dodajanju indeksov.

4 Zaključek

Uporaba IoT naprav in pametnih števcov je v celotnem energetskega sektorju (in širše) povzročila ekspanzijo podatkov, ki se jih lahko pridobiva v skoraj realnem času iz različnih merilnih naprav, ki so na voljo in so instalirani neposredno pri končnih odjemalcih in/ali na samem omrežju oziroma v našem primeru na distribucijskem energetskega omrežju DO, ki je v upravljanju na Elektro Ljubljana.

S stališča informacijsko-komunikacijskih tehnologij (IKT) je izziv predvsem v zagotavljanju ustrezne infrastrukture za zbiranje, prenos, shranjevanje in obdelavo ogromnih količin podatkov, kar zahteva uporabo naprednih metod strojnega učenja, podatkovnega rudarjenja ter orodij za analitiko velikih podatkov (big data) [5]. S tem namenom smo na Elektro Ljubljana preizkusili načrtovanje in uporabo odprtokodnih baz časovnih serij in jih primerjali med sabo. V predstavljam članku smo povzeli naše izkušnje ter poudarili prednosti in slabosti uporabe in umeščenosti v naši IKT podatkovni in sistemski infrastrukturi. Rezultat preizkusa je v prednosti uporabe ClickHouse baze.

IoT naprave še naprej bodo naš ključni element podatkovni in sistemski infrastrukturi. Njihov razvoj in razvoj tehnologij, ki jih podpirajo spreminja in nadgrajuje digitalne transformacije celotnega energetskega sektorja in predstavljajo osnovo za nadaljnji razvoj inteligentnih in odzivnih elektroenergetskih sistemov. Na Elektro Ljubljana sledimo tem razvojem in novitete vpeljujemo v naše procese in postopke dela na vse področjih.

Literatura

- [1] Gungor, V. C., Sahin, D., Kocak, T., Ergut, S., Buccella, C., Cecati, C., & Hancke, G. P. (2011). Smart grid technologies: Communication technologies and standards. *IEEE Transactions on Industrial Informatics*, 7(4), 529–539.
- [2] Mohassel, R. R., Fung, A., Mohammadi, F., & Raahemifar, K. (2014). A survey on advanced metering infrastructure. *International Journal of Electrical Power & Energy Systems*, 63, 473–484.
- [3] Amin, S. M., & Wollenberg, B. F. (2005). Toward a smart grid: Power delivery for the 21st century. *IEEE Power and Energy Magazine*, 3(5), 34–41.
- [4] Fang, X., Misra, S., Xue, G., & Yang, D. (2012). Smart grid – The new and improved power grid: A survey. *IEEE Communications Surveys & Tutorials*, 14(4), 944–980.
- [5] Zhou, K., Fu, C., & Yang, S. (2016). Big data driven smart energy management: From big data to big insights. *Renewable and Sustainable Energy Reviews*, 56, 215–225.
- [6] NOVAK, Matic, 2019, Uporaba tehnologije NoSQL za učinkovitejše izračunavanje regulatornih poročil v zavarovalništvu [na spletu]. 2019. [Dostopano 25 oktober 2022]. Pridobljeno <https://repozitorij.uni-lj.si/IzpisGradiva.php?lang=slv&id=110061>

