

doi:10.15199/48.2023.06.22

Przetwarzanie danych temporalnych dla wymiaru czasu rzeczywistego na platformie MariaDB

Streszczenie. Celem artykułu jest wskazanie zakresu implementacji obsługi danych temporalnych dla czasu rzeczywistego w środowisku MariaDB oraz określenie stopnia zgodności tej implementacji z zapisami dotyczącymi temporalnych rozszerzeń języka SQL zawartych w standardzie ISO/IEC 9075 w wersji SQL:2011, a także prezentacja możliwości przetwarzania danych temporalnych w środowisku MariaDB.

Abstract. The aim of the article is an indication the scope of the implementation of temporal data support for valid time in the MariaDB environment and determining the degree of compliance of this implementation with the provisions on temporal extensions of the SQL language of the ISO/IEC 9075 standard in the SQL: 2011 version, as well as to present possibility processing of temporal data by MariaDB environment. (**Processing of temporal data for the valid time on the MariaDB platform**).

Słowa kluczowe: temporalne bazy danych, temporalne tabele, temporalne operatory, czas rzeczywisty, SQL:2011, MariaDB.

Keywords: temporal databases, temporal tables, temporal operators, valid time, SQL:2011, MariaDB.

Wstęp

Artykuł ten stanowi kolejny cykl rozważań na temat obsługi danych temporalnych w systemach baz danych opartych o relacyjny model danych. Poprzednie artykuły poświęcone zostały modelowaniu danych temporalnych w relacyjnym modelu danych [1], rozwojowi języka SQL i standardu ISO/IEC 9075 ze szczególnym uwzględnieniem składni pozwalającej składować i przetwarzać dane w RDBMS [2], a także możliwościom składowania i przetwarzania danych temporalnych na platformach MS SQL Server i Azure SQL Database [3], Oracle [4,5] oraz składowaniu i przetwarzaniu danych temporalnych w wymiarze czasu transakcyjnego na platformie MariaDB [6,7] a także składowaniu danych temporalnych w wymiarze czasu rzeczywistego na platformie MariaDB [8]. Artykuł ten jest kontynuacją rozważań nt. możliwości składowania i przetwarzania danych temporalnych oferowanych przez serwery SQL. W całości poświęcony został on przetwarzaniu danych temporalnych na platformie MariaDB z uwzględnieniem wymiaru czasu rzeczywistego. Przeprowadzona została analiza zakresu możliwości przetwarzania danych temporalnych dostępnych na platformie MariaDB, dla czasu rzeczywistego. Celem tej analizy było zidentyfikowanie stopnia zgodności obsługi danych temporalnych z wymaganiami standardu ISO/IEC 9075, w szczególności z wersją standardu SQL:2011. Ponadto zaprezentowane zostały możliwości i przykłady przetwarzania danych temporalnych dla wymiaru czasu rzeczywistego w środowisku MariaDB.

Podstawowe wymagania składowania i obsługi danych temporalnych na podstawie specyfikacji standardu SQL:2011

Główne elementy dotyczące obsługi danych temporalnych, które zostały wprowadzone w standardzie SQL 2011 to [9-11]:

- definicja okresu czasu,
- tabele temporalne wersjonowane aplikacyjnie lub systemowo,
- tabele bitemporalne (wersjonowane aplikacyjnie i systemowo),
- możliwość aktualizacji i usuwania rekordów z określonego przedziału czasowego,
- temporalne ograniczenie klucza podstawowego,
- temporalne ograniczenie integralności referencyjnej,
- nowe predykaty czasowe dla interwałów czasowych.

Składowanie danych temporalnych w MariaDB

Obsługa danych temporalnych w środowisku MariaDB została wprowadzona w wersji 10.3.4, w 2017 roku. Dotyczyła ona wymiaru czasu transakcyjnego w odniesieniu do tabel wersjonowanych systemowo. Rok później została zaimplementowana obsługa wymiaru czasu rzeczywistego w wersji 10.4.3, dla tabel wersjonowanych aplikacyjnie. W tej wersji także dostępna stała się obsługa tabel bitemporalnych [12].

Dane temporalne z uwzględnieniem wymiaru czasu rzeczywistego składowane są w tabelach temporalnych wersjonowanych aplikacyjnie. Tabele wersjonowane aplikacyjnie dla obsługi czasu rzeczywistego przechowują dodatkowo wartość czasookresu, w którym dany rekord był prawdziwy (był aktualny w modelowanym świecie rzeczywistym). Wartości czasookresu przechowywane są w dwóch odrębnych kolumnach składających odpowiednio początek i koniec zdefiniowanego przez użytkownika czasookresu. W przeciwieństwie do tabel wersjonowanych systemowo, to użytkownik jest odpowiedzialny za zdefiniowanie wartości granic czasookresu, dla konkretnego rekordu składowanego i przetwarzanego w tabeli wersjonowanej aplikacyjnie.

Platforma MariaDB realizuje większość wymagań zdefiniowanych w standardzie SQL:2011, w odniesieniu do wymiaru czasu rzeczywistego. W aspekcie przetwarzania danych temporalnych dostępny jest mechanizm czasowej aktualizacji i czasowego usuwania rekordów składowanych w tabeli wersjonowanej aplikacyjnie. Niestety niezaimplementowane zostały temporalne predykaty do obsługi zapytań na danych składowanych w tabelach wersjonowanych aplikacyjnie. Odpytywanie ich możliwe jest za pomocą standardowej konstrukcji instrukcji wybierającej, jak w przypadku klasycznej tabeli.

Szczegółowe zestawienie wybranych cech temporalnych standardu SQL:2011 oraz tych zaimplementowanych na platformie MariaDB, w odniesieniu do obsługi danych temporalnych dla wymiaru czasu rzeczywistego przedstawiono w pracy [8].

Manipulacja danymi temporalnymi

Instrukcja INSERT skutkuje dodaniem nowego rekordu do tabeli temporalnej wersjonowanej aplikacyjnie. Poniżej przedstawiono przykład dodania kilku rekordów do tabeli temporalnej z obsługą czasu rzeczywistego za pomocą jedno i wielopoleceniowej instrukcji INSERT [13]:

```

INSERT INTO pracownik_vt
VALUES
(1, 'Jan', 'Kowalski', 'Warszawa', 'asystent', '2000-10-01', '2008-11-30');
INSERT INTO pracownik_vt
VALUES (1, 'Jan', 'Kowalski', 'Warszawa', 'adiunkt', '2008-12-01', '9999-12-31');
INSERT INTO pracownik_vt
VALUES
(2, 'Katarzyna', 'Nowakowska', 'Olkusz', 'adiunkt', '2012-09-30', '2014-09-30');
INSERT INTO pracownik_vt
VALUES (3, 'Adam', 'Grzegorzczak', 'Gdynia', 'starszy wykładowca', CURDATE(), '9999-12-31');

INSERT INTO pracownik_vt
VALUES
(4, 'Alicja', 'Jarebska', 'Jelenia Góra', 'asystent', '2022-10-01', '9999-12-31'),
(5, 'Julia', 'Widawska', 'Kraków', 'profesor', CURDATE()+INTERVAL 7 MONTH, CURDATE()+INTERVAL 2 YEAR);

```

Na rysunku 1 przedstawiono zawartość tabeli temporalnej po dodaniu rekordów, z widoczną zawartością kolumn dla jawnych znaczników czasowych.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2008-11-30
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-29	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-29

Rys.1. Przykładowa zawartość tabeli temporalnej wersjonowanej aplikacyjnie, po dodaniu rekordów, źródło: opracowanie własne

W tabeli temporalnej zostały umieszczone rekordy, które były aktualne w przeszłości, rekordy, które nadal są aktualne, rekordy, które staną się aktualne w nadchodzącej przyszłości, których status wygaśnie w konkretnym dniu, w przyszłości lub też ich status jest otwarty na chwilę obecną, a więc nie są one ograniczone żadną konkretną datą ważności. Wartość daty składowana w kolumnie `ApliCzasKoniec` odpowiadająca maksymalnej dopuszczalnej wartości dla systemowego typu danych `DATE`, stanowi informację, iż rekord ten jest rekordem aktualnym na chwilę obecną, dopóki wartość ta nie ulegnie zmianie na nową wiążącą datę (np. w skutek odejścia pracownika z pracy lub też jego awansu na nowe stanowisko).

Poniżej przedstawiono przykład dodania rekordu do tabeli temporalnej z wykluczeniem nakładania się okresów czasowych:

```

INSERT INTO pracownik_vt_wo
VALUES
(1, 'Jan', 'Kowalski', 'Warszawa', 'profesor', '2022-10-01', '2024-12-31');

```

Próba dodania rekordu awansującego pracownika na nowe stanowisko, na okres 2 lat zakończy się komunikatem o błędzie z uwagi na fakt, iż usiłowano zmienić stanowisko pracy pracownika przed wygaśnięciem poprzedniego stanowiska. Poniżej przedstawiono postać komunikatów zwracanych przy próbie dodania rekordu do tabeli temporalnej wersjonowanej aplikacyjnie z wykluczeniem nakładania czasookresów, który naruszają tą zasadę. Pierwszy komunikat generowany jest podczas próby dodania rekordu do tabeli temporalnej z wykluczeniem nakładania czasookresów implementowanym za pomocą ograniczającego indeksu `UNIQUE`. Druga postać komunikatu generowana jest przy próbie dodania rekordu do tabeli temporalnej z wykluczeniem nakładania

czasookresów implementowanym w ograniczeniu klucza podstawowego [13].

BŁĄD SQL 1062: Duplicate entry '1-profesor-2024-12-31' for key 'IdPracownika'

BŁĄD SQL 1062: Duplicate entry '1-profesor-2024-12-31-2022-10-01' for key 'PRIMARY'

Rekordy do tabeli bitemporalnej dodawane są w identyczny sposób, jak dla klasycznej tabeli nietemporalnej, czy też tabeli wersjonowanej systemowo lub tabeli wersjonowanej aplikacyjnie. Na rysunku 2 przedstawiono przykładową zawartość tabeli bitemporalnej. Zawartość kolumn `SysCzasStart` i `SysCzasKoniec` odpowiada odpowiednio momentowi dodania danego rekordu do tabeli bitemporalnej, a także, chwili czasowej, do kiedy dany rekord był aktualną wersją. Maksymalna dopuszczalna wartość dla typu systemowego `TIMESTAMP` w kolumnie `SysCzasKoniec` oznacza, że jest to bieżąca wersja rekordu. Wartości te nie pokrywają się z faktycznym czasookresem, gdy dany rekord był prawdziwy w zadanym przedziale czasowym w przeszłości.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec	SysCzasStart	SysCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2008-11-30	2022-07-29 22:25:37.943064	2038-01-19 04:14:07.999999
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31	2022-07-29 22:26:11.026234	2038-01-19 04:14:07.999999
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30	2022-07-30 11:01:02.648057	2038-01-19 04:14:07.999999
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-30	9999-12-31	2022-07-30 11:05:26.046731	2038-01-19 04:14:07.999999
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31	2022-07-30 13:08:13.418560	2038-01-19 04:14:07.999999
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30	2022-07-30 13:08:13.418560	2038-01-19 04:14:07.999999

Rys.2. Przykładowa zawartość tabeli bitemporalnej, po dodaniu rekordów, źródło: opracowanie własne

Rekordy składowane w tabeli temporalnej wersjonowanej aplikacyjnie mogą być poddawane aktualizacji za pomocą standardowej postaci instrukcji `UPDATE`. Na platformie MariaDB zaimplementowano rozszerzenie `FOR PORTION OF`, które umożliwia temporalną aktualizację rekordów. Temporalnej aktualizacji podlegają rekordy zawarte w przedziale czasowym podanym, jako parametr temporalnego rozszerzenia instrukcji `UPDATE` [14].

W przypadku istnienia rekordów, dla których punkt początkowy lub końcowy przedziału czasowego wykracza poza granicę czasookresu zdefiniowanego w klauzuli `FOR PORTION OF`, takie rekordy są dzielone odpowiednio na dwa lub trzy ciągłe rekordy (zależnie od zakresu pokrywania się czasookresów zdefiniowanych dla wybranego rekordu oraz określonego w klauzuli `FOR PORTION OF`), w tym rekord, którego czasookres całkowicie zawiera się w przedziale zdefiniowanym w klauzuli `FOR PORTION OF` [2,10].

Poniżej przedstawiono przykład aktualizacji rekordu w tabeli temporalnej wersjonowanej aplikacyjnie:

```

UPDATE pracownik_vt FOR PORTION OF APLI_TIME
FROM '2005-01-01' TO '2007-12-31'
SET Miejscowosc='Łódź'
WHERE IdPracownika=1

```

Na rysunku 3 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po przeprowadzeniu aktualizacji rekordu.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2005-01-01
1	Jan	Kowalski	Łódź	asystent	2005-01-01	2007-12-31
1	Jan	Kowalski	Warszawa	asystent	2007-12-31	2008-11-30
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-29	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-29

Rys.3. Przykładowa zawartość tabeli temporalnej, po temporalnej aktualizacji rekordów, źródło: opracowanie własne

Z uwagi na fakt, iż punkt początkowy przedziału czasowego rekordu leży poza granicą przedziału czasowego zdefiniowanego w klauzuli PORTION OF, rekord ten został podzielony na trzy rekordy, w tym rekord, którego czasookres w całości zawiera się w przedziale czasowym określonym w klauzuli PORTION OF.

Wersja rekordu dla pracownika o IdPracownika=1 została podzielona na wersję rekordu z przedziałem czasu 2000-10-01÷2005-01-01, gdy pracownik mieszkał w tym czasie w Warszawie. Kolejna wersja rekordu odpowiada czasookresowi 2005-01-01÷2007-12-31, gdy pracownik mieszkał w Łodzi. Jest zarazem wersją rekordu, którego czasookres w pełni zawiera się w przedziale czasowym zdefiniowanym w klauzuli PORTION OF. Trzecia wersja rekordu odpowiada czasookresowi 2007-12-31÷2008-11-30, gdy pracownik z powrotem przeprowadził się do Warszawy. Wersja rekordu, w której odnotowano awans pracownika na stanowisko adiunkta nie została poddana modyfikacji z uwagi na fakt, że jej czasookres całkowicie leży poza wartościami czasookresu zdefiniowanego w klauzuli PORTION OF.

Poniżej przedstawiono przykład temporalnej aktualizacji rekordu, dla przypadku, gdy dolna granica czasookresu zawartego w klauzuli PORTION OF poprzedza początek czasookresu modyfikowanego rekordu:

```
UPDATE pracownik_vt FOR PORTION OF APLI_TIME
FROM '1999-01-01' TO '2007-12-31'
SET Miejscowosc='łódź'
WHERE IdPracownika=1
```

Na rysunku 4 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnej aktualizacji rekordów dla przypadku, gdy dolna granica czasookresu zdefiniowanego w klauzuli PORTION OF poprzedza początek czasookresu aktualizowanego rekordu.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Łódź	asystent	2000-10-01	2007-12-31
1	Jan	Kowalski	Warszawa	asystent	2007-12-31	2008-11-30
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-30	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30

Rys.4. Przykładowa zawartość tabeli temporalnej, po temporalnej aktualizacji rekordów, dla przypadku, gdy dolna granica czasookresu zdefiniowanego dla klauzuli PORTION OF poprzedza początek czasookresu aktualizowanego rekordu, źródło: opracowanie własne

W tym przypadku wersja rekordu została podzielona na dwie wersje. Pierwsza, gdy pracownik od samego początku zatrudnienia, jako asystent, do dnia 2007-12-31 zamieszkiwał w Łodzi. Druga wersja rekordu, gdy od dnia 2007-12-31 do dnia 2008-11-30 mieszkał w Warszawie, nadal będąc zatrudnionym na stanowisku asystenta.

Kolejny przykład dotyczy temporalnej aktualizacji rekordu dla przypadku, gdy górna granica czasookresu zawartego w klauzuli PORTION OF wykracza poza koniec czasookresu modyfikowanego rekordu:

```
UPDATE pracownik_vt FOR PORTION OF APLI_TIME
FROM '2005-01-01' TO '2011-09-30'
SET Miejscowosc='łódź'
WHERE IdPracownika=1
```

Na rysunku 5 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnej aktualizacji rekordów dla przypadku, gdy górna granica czasookresu zdefiniowanego w klauzuli PORTION OF przekracza koniec czasookresu aktualizowanego rekordu.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2005-01-01
1	Jan	Kowalski	Łódź	asystent	2005-01-01	2008-11-30
1	Jan	Kowalski	Łódź	adiunkt	2008-12-01	2011-09-30
1	Jan	Kowalski	Warszawa	adiunkt	2011-09-30	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-30	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30

Rys.5. Przykładowa zawartość tabeli temporalnej, po temporalnej aktualizacji rekordów, dla przypadku, gdy górna granica czasookresu zdefiniowanego dla klauzuli PORTION OF przekracza koniec czasookresu aktualizowanego rekordu, źródło: opracowanie własne

W tym przypadku podziałowi uległa wersja rekordu, gdy pracownik był zatrudniony, jako asystent oraz wersja, gdy pracownik był zatrudniony, jako adiunkt. Pierwsza wersja obowiązuje do dnia 2005-01-01, gdy pracownik zatrudniony na stanowisku asystenta zamieszkiwał w Warszawie. Druga wersja rekordu, gdy pracownik od dnia 2005-01-01 do końca okresu zatrudnienia na stanowisku asystenta mieszkał w Łodzi. Kolejna wersja rekordu od dnia awansu pracownika na stanowiska adiunkta do dnia 2011-09-30, gdy pracownik nadal mieszkał w Łodzi. Ostatnia wersja rekordu, dotyczy sytuacji, gdy pracownik będąc już zatrudnionym na stanowisku adiunkta, w dniu 2011-09-30 zmienił ponownie miejsce zamieszkania na Warszawę.

Poniżej przedstawiono przykład temporalnej aktualizacji rekordu dla przypadku, gdy dolna granica czasookresu zawartego w klauzuli PORTION OF poprzedza początek czasookresu tabeli temporalnej, a górna granica wykracza poza koniec czasookresu aktualizowanego rekordu:

```
UPDATE pracownik_vt FOR PORTION OF APLI_TIME
FROM '1999-01-01' TO '2011-09-30'
SET Miejscowosc='łódź'
WHERE IdPracownika=1
```

Na rysunku 6 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnej aktualizacji rekordów dla przypadku, gdy dolna granica czasookresu zawartego w klauzuli PORTION OF poprzedza początek czasookresu modyfikowanego rekordu, a górna granica wykracza poza koniec czasookresu modyfikowanego rekordu.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Łódź	asystent	2000-10-01	2008-11-30
1	Jan	Kowalski	Łódź	adiunkt	2008-12-01	2011-09-30
1	Jan	Kowalski	Warszawa	adiunkt	2011-09-30	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-30	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30

Rys.6. Przykładowa zawartość tabeli temporalnej, po temporalnej aktualizacji rekordów, dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF leżą poza granicami czasookresu aktualizowanego rekordu, źródło: opracowanie własne

W wersji rekordu odpowiadającej okresowi zatrudnienia pracownika na stanowisku asystenta uległa tylko zmiana miejscowości zamieszkania z Warszawy na Łódź. Wersja rekordu, gdy pracownik był zatrudniony, jako adiunkt uległa podziałowi na dwie wersje. Pierwsza wersja od dnia awansu pracownika na stanowisko adiunkta, do dnia 2011-09-30, czyli do dnia, gdy zamieszkiwał w Łodzi. Druga wersja rekordu, gdy w dniu 2011-09-30 pracownik będąc zatrudnionym na stanowisku adiunkta przeprowadził się do Warszawy.

Poniżej przedstawiono przykład temporalnej aktualizacji rekordu dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF pokrywają się z granicami czasookresu modyfikowanego rekordu:

```
UPDATE pracownik_vt FOR PORTION OF APLI_TIME
FROM '2000-10-01' TO '2008-11-30'
SET Miejscowosc='Łódź'
WHERE IdPracownika=1
```

Na rysunku 7 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnej aktualizacji rekordów dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF pokrywają się z granicami czasookresu aktualizowanego rekordu.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Łódź	asystent	2000-10-01	2008-11-30
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-30	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30

Rys.7. Przykładowa zawartość tabeli temporalnej, po temporalnej aktualizacji rekordów, dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF pokrywają się z granicami czasookresu aktualizowanego rekordu, źródło: opracowanie własne

W tym przypadku wersja rekordu nie uległa podziałowi na kilka wersji. Zmianie uległa miejscowość, w której zamieszkiwał pracownik przez cały okres swojego zatrudnienia na stanowisku asystenta. Zmianie nie uległa także wersja rekordu odnotowująca awans pracownika na stanowisko adiunkta.

Przeprowadzenie temporalnej aktualizacji dla przypadku, gdy przedział czasowy zdefiniowany w klauzuli PORTION OF całkowicie jest położony poza granicami czasookresu rekordu składowanego w tabeli temporalnej, nie zmienia zawartości wersji rekordów temporalnych składowanych w tabeli temporalnej wersjonowanej aplikacyjnie.

W odniesieniu do kolumn definiujących granice czasookresu nie można odwoływać się za pomocą operatora SET. Tym samym kolumny te nie mogą podlegać aktualizacji.

Aktualizacja rekordów składowanych w tabeli bitemporalnej odbywa się w identyczny sposób jak dla tabeli wersjonowanej aplikacyjnie. Dodatkowo rejestracji podlega data i godzina aktualizacji rekordu w wymiarze czasu transakcyjnego. Ponadto archiwizacji podlega poprzednia wersja rekordu sprzed jego aktualizacji.

Kolejny przykład dotyczy temporalnej aktualizacji rekordu składowanego w tabeli bitemporalnej, dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF pokrywają się z granicami czasookresu modyfikowanego rekordu:

```
UPDATE pracownik_bt FOR PORTION OF APLI_TIME FROM
'2000-10-01' TO '2008-11-30'
SET Miejscowosc='Łódź'
WHERE IdPracownika=1
```

Na rysunku 8 przedstawiono zawartość tabeli bitemporalnej, po aktualizacji rekordu, dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF pokrywają się z granicami czasookresu modyfikowanego rekordu.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec	SysCzasStart	SysCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2008-11-30	2022-07-29 22:25:37.945866	2022-10-23 11:12:52.152271
1	Jan	Kowalski	Łódź	asystent	2000-10-01	2008-11-30	2022-10-23 17:12:52.152271	2038-01-19 04:14:07.999999
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31	2022-07-29 22:26:11.006234	2038-01-19 04:14:07.999999
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30	2022-07-30 11:01:02.648857	2038-01-19 04:14:07.999999
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-30	9999-12-31	2022-07-30 11:09:26.546733	2038-01-19 04:14:07.999999
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31	2022-07-30 13:08:13.418560	2038-01-19 04:14:07.999999
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30	2022-07-30 13:08:13.418560	2038-01-19 04:14:07.999999

Rys.8. Przykładowa zawartość tabeli bitemporalnej, po temporalnej aktualizacji rekordów, dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF pokrywają się z granicami czasookresu aktualizowanego rekordu, źródło: opracowanie własne

W odniesieniu do aktualizacji na tabeli wersjonowanej aplikacyjnie, wersjonowaniu poddana została poprzednia wersja rekordu wraz z utwaleniem chwili czasowej tej zmiany w systemie.

Rekordy składowane w tabeli temporalnej wersjonowanej aplikacyjnie mogą być usuwane za pomocą standardowej postaci instrukcji DELETE. Na platformie MariaDB zaimplementowano rozszerzenie PORTION OF także w odniesieniu do instrukcji DELETE. Dzięki temu możliwe jest temporalne usuwanie rekordów z tabeli temporalnej, a więc rekordów, zawartych w przedziale czasu podanym w klauzuli PORTION OF. Wielokrotne usuwanie nie jest obsługiwane [15].

Usuwane są rekordy całkowite zawierające się w zdefiniowanym czasookresie. W przypadku istnienia zbioru rekordów, dla których czasookres wykracza poza granice „ważności” wykonywana jest akcja wstawienia tych rekordów do tabeli temporalnej [2,10].

Poniżej przedstawiono przykład temporalnego usuwania rekordu dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF zawierają się w granicach czasookresu rekordu składowanego w tabeli temporalnej:

```
DELETE FROM pracownik_vt FOR PORTION OF APLI_TIME FROM
'2005-01-01' TO '2007-12-31'
WHERE IdPracownika=1;
```

Na rysunku 9 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnym usunięciu rekordów dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF zawierają się w granicach czasookresu rekordu składowanego w tabeli temporalnej.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Łódź	asystent	2000-10-01	2005-01-01
1	Jan	Kowalski	Łódź	asystent	2007-12-31	2008-11-30
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-30	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30

Rys.9. Przykładowa zawartość tabeli temporalnej, po temporalnym usunięciu rekordów, dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF zawierają się w granicach czasookresu rekordu składowanego w tabeli temporalnej, źródło: opracowanie własne

W tym przypadku, wersja rekordu została podzielona na dwie wersje. Do tabeli temporalnej dodana została wersja rekordu z przedziału czasowego 2000-10-01÷2005-01-01, a więc poniżej dolnej granicy czasookresu zawartego w klauzuli PORTION OF. Tak, więc, dane z tego przedziału czasowego nie podlegają usunięciu. Druga wersja rekordu dotyczy przedziału czasowego 2007-12-31÷2008-11-30, a więc okresu czasu do końca zatrudnienia pracownika na stanowisku asystenta. Wersja rekordu przechowująca dane na temat awansu pracownika na stanowisko adiunkta nie uległa żadnym przekształceniom z uwagi, na fakt, iż granice czasookresu tego rekordu całkowicie leżą poza granicami przedziału czasowego uwzględnionego w klauzuli PORTION OF dla usuwanych rekordów.

Kolejny przykład przedstawia temporalne usuwanie rekordu dla przypadku, gdy dolna granica czasookresu zawartego w klauzuli PORTION OF leży poniżej dolnej granicy czasookresu rekordu składowanego w tabeli temporalnej:

```
DELETE FROM pracownik_vt FOR PORTION OF APLI_TIME FROM
'1999-01-01' TO '2007-12-31'
WHERE IdPracownika=1;
```

Na rysunku 10 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnym usunięciu rekordów dla przypadku, gdy dolna granica czasookresu zawartego w klauzuli PORTION OF położona jest poniżej dolnej granicy czasookresu rekordu składowanego w tabeli temporalnej.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2007-12-31	2008-11-30
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-31	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-31

Rys.10. Przykładowa zawartość tabeli temporalnej, po temporalnym usunięciu rekordów, dla przypadku, gdy dolna granica czasookresu zdefiniowanego dla klauzuli PORTION OF leży poniżej dolnej granicy czasookresu rekordu składowanego w tabeli temporalnej, źródło: opracowanie własne

W tym przypadku usunięte zostały dane za okres czasu odpowiadający początkowi zatrudnienia pracownika na stanowisku asystenta do dnia 2007-12-31, a więc do dnia stanowiącego górną wartość czasookresu zawartego w klauzuli PORTION OF. W tabeli temporalnej pozostała wersja danych z wartościami granicznymi czasookresu odpowiadającymi górnemu zakresowi przedziału czasowego zawartego w klauzuli PORTION OF oraz dacie wygaśnięcia zatrudnienia pracownika na stanowisku asystenta. Podobnie jak w poprzednim przypadku wersja rekordu składująca czasookres zatrudnienia na stanowisku adiunkta nie uległa zmianie.

Poniżej przedstawiono przykład temporalnego usuwania rekordu dla przypadku, gdy górna granica czasookresu zawartego w klauzuli PORTION OF leży powyżej górnej granicy czasookresu rekordu składowanego w tabeli temporalnej:

```
DELETE FROM pracownik_vt FOR PORTION OF APLI_TIME FROM '2005-01-01' TO '2011-09-30' WHERE IdPracownika=1;
```

Na rysunku 11 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnym usunięciu rekordów dla przypadku, gdy górna granica czasookresu zawartego w klauzuli PORTION OF położona jest powyżej górnej granicy czasookresu rekordu składowanego w tabeli temporalnej.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2005-01-01
1	Jan	Kowalski	Warszawa	adiunkt	2011-09-30	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-31	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-31

Rys.11. Przykładowa zawartość tabeli temporalnej, po temporalnym usunięciu rekordów, dla przypadku, gdy górna granica czasookresu zdefiniowanego dla klauzuli PORTION OF leży powyżej górnej granicy czasookresu rekordu składowanego w tabeli temporalnej, źródło: opracowanie własne

Czasookres odpowiadający zatrudnieniu pracownika na stanowisku asystenta został zredukowany do dnia 2005-01-01, a więc do początku zakresu przedziału czasowego zawartego w klauzuli PORTION OF. Podobnie czasookres odpowiadający zatrudnieniu pracownika na stanowisku adiunkta także uległ skróceniu. Początek czasookresu dla tej wersji rekordu odpowiada końcowi zakresu przedziału czasowego zawartego w klauzuli PORTION OF.

Poniżej przedstawiono przykład temporalnego usuwania rekordu dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF całkowicie zawierają granice czasookresu rekordu składowanego w tabeli temporalnej:

```
DELETE FROM pracownik_vt FOR PORTION OF APLI_TIME FROM '1999-01-01' TO '2011-09-30' WHERE IdPracownika=1;
```

Na rysunku 12 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnym usunięciu rekordów dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF całkowicie zawierają granice czasookresu rekordu składowanego w tabeli temporalnej.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	adiunkt	2011-09-30	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-31	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-31

Rys.12. Przykładowa zawartość tabeli temporalnej, po temporalnym usunięciu rekordów, dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF całkowicie zawierają granice czasookresu rekordu składowanego w tabeli temporalnej, źródło: opracowanie własne

W tym przypadku wersja rekordu obejmująca cały okres zatrudnienia pracownika na stanowisku asystenta została usunięta z tabeli temporalnej. Czasookres odpowiadający zatrudnieniu pracownika na stanowisku adiunkta także uległ skróceniu. Początek czasookresu dla tej wersji rekordu odpowiada końcowi zakresu przedziału czasowego zawartego w klauzuli PORTION OF.

Poniżej przedstawiono przykład temporalnego usuwania rekordu dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF pokrywają się z granicami czasookresu rekordu składowanego w tabeli temporalnej:

```
DELETE FROM pracownik_vt FOR PORTION OF APLI_TIME FROM '2000-10-01' TO '2008-11-30' WHERE IdPracownika=1;
```

Na rysunku 13 przedstawiono zawartość tabeli temporalnej wersjonowanej aplikacyjnie po temporalnym usunięciu rekordów dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF pokrywają się z granicami czasookresu rekordu składowanego w tabeli temporalnej.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30
3	Adam	Grzegorzczak	Gdynia	starszy wykładowca	2022-07-31	9999-12-31
4	Alicja	Jarebska	Jelenia Góra	asystent	2022-10-01	9999-12-31
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-31

Rys.13. Przykładowa zawartość tabeli temporalnej, po temporalnym usunięciu rekordów, dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF pokrywają się z granicami czasookresu rekordu składowanego w tabeli temporalnej, źródło: opracowanie własne

W tym przypadku wersja rekordu obejmująca cały okres zatrudnienia pracownika na stanowisku asystenta została usunięta z tabeli temporalnej. Natomiast w odróżnieniu od poprzedniego przykładu, modyfikacji nie uległ czasookres odpowiadający zatrudnieniu pracownika na stanowisku adiunkta.

Temporalne usuwanie rekordów dla przypadku, gdy przedział czasowy zdefiniowany w klauzuli PORTION OF całkowicie jest położony poza granicami czasookresu rekordu składowanego w tabeli temporalnej, nie zmienia wersji rekordów temporalnych składowanych w tabeli temporalnej wersjonowanej aplikacyjnie oraz nie usuwa żadnego rekordu.

Usuwanie rekordów składowanych w tabeli bitemporalnej odbywa się w identyczny sposób, jak dla tabeli wersjonowanej aplikacyjnie. Dodatkowo rejestracji podlega data i godzina usunięcia rekordu w wymiarze czasu transakcyjnego. Ponadto archiwizacji podlega poprzednia wersja rekordu sprzed jego usunięcia [16].

Poniżej przedstawiono przykład temporalnego usunięcia rekordu składowanego, w tabeli bitemporalnej, dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF pokrywają się z granicami czasookresu usuwanego rekordu.

```
DELETE FROM pracownik_bt FOR PORTION OF APLI_TIME FROM
'2000-10-01' TO '2008-11-30'
WHERE IdPracownika=1;
```

Na rysunku 14 przedstawiono zawartość tabeli bitemporalnej, po usunięciu rekordu, dla przypadku, gdy obie granice czasookresu zawartego w klauzuli PORTION OF pokrywają się z granicami czasookresu usuwanego rekordu.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec	SysCzasStart	SysCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2008-11-30	2022-07-29 22:25:37.945866	2022-10-23 17:12:52.152271
1	Jan	Kowalski	Łódź	asystent	2000-10-01	2008-11-30	2022-10-23 17:12:52.152271	2022-10-23 17:45:05.449627
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31	2022-07-30 13:08:11.026234	2038-01-19 04:14:07.999999
2	Katarzyna	Niewiadomska	Opole	adiunkt	2012-09-30	2014-09-30	2022-07-30 11:01:02.648817	2038-01-19 04:14:07.999999
3	Adam	Czerniecki	Gdynia	starszy wykładowca	2022-07-30	9999-12-31	2022-07-30 11:05:26.046733	2038-01-19 04:14:07.999999
4	Alicja	Jędrska	Jelenia Góra	asystent	2022-10-01	9999-12-31	2022-07-30 13:08:11.018360	2038-01-19 04:14:07.999999
5	Julia	Widawska	Kraków	profesor	2023-02-28	2024-07-30	2022-07-30 13:08:11.018360	2038-01-19 04:14:07.999999

Rys.14. Przykładowa zawartość tabeli bitemporalnej, po temporalnym usunięciu rekordów, dla przypadku, gdy obie granice czasookresu zdefiniowanego dla klauzuli PORTION OF całkowicie pokrywają się z granicami czasookresu rekordu składowanego w tabeli bitemporalnej, źródło: opracowanie własne

W odniesieniu do usuwania rekordu z tabeli wersjonowanej aplikacyjnie wersjonowaniu poddana została poprzednia wersja rekordu sprzed jego usunięcia, wraz z utwaleniem chwili czasowej tej zmiany w systemie.

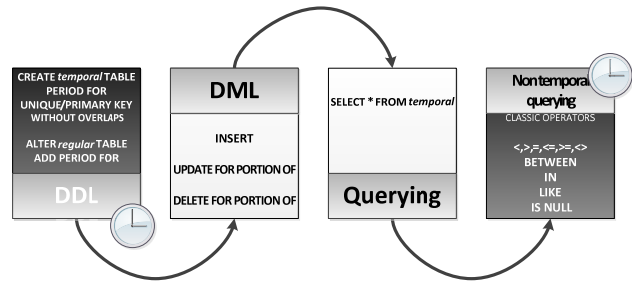
Obsługa zapytań

Standard SQL 2011 wprowadza siedem predykatów temporalnych do obsługi zapytań temporalnych dla wymiaru czasu rzeczywistego. Niestety na chwilę obecną w środowisku MariaDB, nie zostały one zaimplementowane, nawet fragmentarycznie. Tabele temporalne wersjonowane aplikacyjnie odpytywane są w tradycyjny sposób, jak w przypadku klasycznej tabeli. W tabeli 1 zestawiono wykaz predykatów temporalnych uwzględnionych w standardzie SQL:2011, celem odpytywania tabel temporalnych wersjonowanych aplikacyjnie [2].

Tabela 1. Rozszerzenia składniowe do przetwarzania danych składowanych w tabelach temporalnych wersjonowanych aplikacyjnie, dla wymiaru czasu rzeczywistego uwzględnione w standardzie SQL:2011

SQL:2011	Interpretacja
[t1,t2] CONTAINS [t3,t4]	t1<t3 n t2>t4
[t1,t2] OVERLAPS [t3,t4]	t1<t3 n t2>t3,
[t1,t2] EQUALS [t3,t4]	t1=t3 n t2=t3
[t1,t2] PRECEDES [t3,t4]	t2<t3
[t1,t2] SUCCEEDS [t3,t4]	t1>t4
[t1,t2] IMMEDIATELY PRECEDES [t3,t4]	t2=t3
[t1,t2] IMMEDIATELY SUCCEEDS [t3,t4]	t1=t4

Na rysunku 15 przedstawiony został sposób pracy z tabelami temporalnymi wersjonowanymi aplikacyjnie w środowisku MariaDB, z obsługą czasu rzeczywistego.



Rys.15. Sposób pracy z tabelami temporalnymi wersjonowanymi aplikacyjnie w środowisku MariaDB, źródło: opracowanie własne

Zapytania na tabelach wersjonowanych aplikacyjnie z obsługą czasu rzeczywistego

Ze względu na fakt, iż na platformie MariaDB nie zaimplementowano rozszerzeń temporalnych do obsługi zapytań czasowych operujących na temporalnych tabelach wersjonowanych aplikacyjnie, zapytania te realizowane są za pomocą instrukcji wybierającej w jej klasycznej postaci, jak dla standardowej tabeli.

Poniżej przedstawiono przykład zapytania na tabeli temporalnej wersjonowanej aplikacyjnie z użyciem operatora BETWEEN:

```
SELECT * FROM pracownik_vt WHERE IdPracownika=1 AND
ApliCzasStart BETWEEN '2000-01-01' AND '2008-12-31';
```

Na rysunku 16 przedstawiono efekt działania powyższego zapytania.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2008-11-30
1	Jan	Kowalski	Warszawa	adiunkt	2008-12-01	9999-12-31

Rys.16. Zestaw danych zwróconych przez zapytanie wybierające w klasycznej postaci, z użyciem operatora BETWEEN, operujące na tabeli temporalnej wersjonowanej aplikacyjnie, źródło: opracowanie własne.

Zwrócone zostały rekordy pracownika o identyfikatorze IdPracownika= 1, gdzie czas rozpoczęcia zatrudnienia pracownika zawiera się w zakresie dat podanych, jako argument użytego operatora.

Kolejny przykład zwraca rekordy dotyczące stanowisk, na których był zatrudniony pracownik do dnia 2022-09-30 (po tej dacie pracownik nie był już zatrudniony na tym stanowisku).

```
SELECT * FROM pracownik_vt WHERE IdPracownika=1 AND
ApliCzasKoniec<='2022-09-30';
```

Na rysunku 17 przedstawiono efekt powyższego zapytania.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
1	Jan	Kowalski	Warszawa	asystent	2000-10-01	2008-11-30

Rys.17. Zestaw danych zwróconych przez zapytanie wybierające w klasycznej postaci, operujące na tabeli temporalnej wersjonowanej aplikacyjnie, wskazujące stanowiska obejmowane przez pracownika do dnia określającego kryterium przeszukiwania, źródło: opracowanie własne

W wynikach końcowych nieuwzględnione zostało zatrudnienie pracownika na stanowisku adiunkta z uwagi na fakt, że pracownik jest zatrudniony na tym stanowisku na czas nieokreślony, tym samym nie spełnia postawionego

kryterium ograniczającego. Zapytanie zwraca tylko historyczne, nieaktualne stanowiska pracownika.

Następny przykład zwraca pracowników zatrudnionych na stanowisku adiunkta w ostatniej dekadzie.

```
SELECT * FROM pracownik_vt WHERE stanowisko='adiunkt'
AND ApliCzasStart>=(NOW()-INTERVAL 10 YEAR);
```

Na rysunku 18 przedstawiono efekt działania powyższego zapytania.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec
2	Katarzyna	Nowakowska	Olkusz	adiunkt	2012-09-30	2014-09-30

Rys.18. Zestaw danych zwróconych przez zapytanie wybierające w klasycznej postaci, operujące na tabeli temporalnej wersjonowanej aplikacyjnie, wskazujące rekordy pracowników zatrudnionych na stanowisku adiunkta w ostatnim dziesięcioleciu, źródło: opracowanie własne

Poniżej podano zmodyfikowany przykład zapytania, zwracający pracowników zatrudnionych na stanowisku adiunkta w ostatniej dekadzie, którzy na dzień dzisiejszy nadal pracują na tym stanowisku:

```
SELECT * FROM pracownik_vt WHERE stanowisko='adiunkt'
AND ApliCzasStart>=(NOW()-INTERVAL 10 YEAR) AND
ApliCzasKoniec>NOW();
```

Powyższe zapytanie zwraca pusty zbiór rekordów z uwagi na brak rekordów spełniających postawione kryterium przeszukiwania. Brak jest rekordów pracowników, którzy rozpoczęli pracę na stanowisku adiunkta nie dalej jak 10 lat temu i nadal są zatrudnieni.

Poniżej podano przykład odpytania tabeli bitemporalnej:

```
SELECT *,SysCzasStart,SysCzasKoniec FROM pracownik_bt
FOR SYSTEM_TIME BETWEEN '2022-07-30 12:00:00' AND
'2022-07-31 12:00:00'
WHERE ApliCzasStart>NOW();
```

Na rysunku 19 przedstawiono efekt działania powyższego zapytania.

IdPracownika	Imie	Nazwisko	Miejscowosc	Stanowisko	ApliCzasStart	ApliCzasKoniec	SysCzasStart	SysCzasKoniec
4	Alija	Jarebka	Jelenia Gora	asystent	2022-10-01	9999-12-31	2022-07-30 13:08:13:418580	2038-01-19 04:1407:999999
5	Julia	Widawska	Krakow	profesor	2023-02-28	2024-07-30	2022-07-30 13:08:13:418550	2038-01-19 04:1407:999999

Rys.19. Zestaw danych zwróconych przez zapytanie wybierające w klasycznej postaci, operujące na tabeli bitemporalnej, źródło: opracowanie własne

Zwrócone zostały rekordy pracowników, których wakant na dzień bieżący jeszcze się nie rozpoczął (obejmą go w przyszłości) oraz wersje rekordów, które były aktualne w danym przedziale czasu systemowego. W chwili uruchomienia zapytania funkcja NOW() zwracała datę z sierpnia 2022 roku, tym samym, wakant dla pracownika o IdPracownika=4 był na ten moment w czasie jeszcze wakatem nieaktywnym.

Podsumowanie

Artykuł ten dotyczy przetwarzania danych temporalnych dla wymiaru czasu rzeczywistego, w środowisku MariaDB. Platforma MariaDB obsługuje oba wymiary czasu, zarówno czas transakcyjny i czas rzeczywisty, co umożliwia także obsługę tabel bitemporalnych.

Dodanie rekordu do tabeli temporalnej wersjonowanej aplikacyjnie przebiega w identyczny sposób, jak dla klasycznej tabeli. Tabela ta zawiera dodatkowo dwie kolumny typu data lub data i czas, określające granice czasookresu dla każdego rekordu w niej

przechowywanego. Wartości składowane w tych kolumnach określają, czy jest to rekord bieżący, czy też historyczny.

W odniesieniu do rekordów składowanych w tabeli temporalnej wersjonowanej aplikacyjnie może być użyta standardowa postać instrukcji UPDATE. Jednakże możliwa jest także temporalna aktualizacja rekordów za pomocą rozszerzenia FOR PORTION OF. Temporalnej aktualizacji podlegają rekordy zawarte w przedziale czasowym podanym, jako parametr temporalnego rozszerzenia instrukcji UPDATE. W przypadku, gdy któryś z punktów granicznych przedziału czasowego dla danego rekordu temporalnego wykracza poza granice czasookresu zdefiniowanego w klauzuli FOR PORTION OF, rekord taki jest dzielony na dwa lub trzy ciągłe rekordy, w tym rekord całkowicie zawierający się w przedziale zdefiniowanym w klauzuli FOR PORTION OF.

W celu usunięcia rekordów z tabeli wersjonowanej aplikacyjnie można zastosować standardową postać instrukcji DELETE. Możliwe jest też użycie temporalnego rozszerzenia tej instrukcji FOR PORTION OF. Umożliwia ono temporalne usuwanie rekordów z tabeli. Usunięcie podlegają wówczas rekordy całkowicie zawierające się w czasookresie zdefiniowanym w klauzuli FOR PORTION OF. W przypadku rekordów, dla których punkt graniczny przedziału czasowego wykracza poza granice czasookresu określonego w rozszerzeniu temporalnym instrukcji DELETE, wykonywane jest wstawianie takich rekordów do tabeli temporalnej.

Realizacja zapytań na tabelach temporalnych wersjonowanych aplikacyjnie możliwa jest tylko za pomocą standardowej postaci instrukcji wybierającej. Na platformie MariaDB nie zostały zaimplementowane predykaty temporalne uwzględnione w zapisach standardu SQL:2011, umożliwiające temporalne odpytywanie tabel temporalnych z obsługą czasu rzeczywistego.

Aktualizacja i usuwanie rekordów składowanych w tabeli bitemporalnej składniowo wygląda identycznie, jak w przypadku działań DML na tabeli temporalnej wersjonowanej aplikacyjnie. Różnica polega na tym, że dodatkowo rejestrowane są zmiany przeprowadzonych modyfikacji tj. data i czas wprowadzenia modyfikacji. Archiwizacji podlega także wcześniejsza wersja rekordu sprzed jego modyfikacji.

Odpytywanie tabeli bitemporalnej możliwe jest przy użyciu temporalnego rozszerzenia FOR SYSTEM_TIME w odniesieniu do kolumn składających granice czasookresu, dla wymiaru czasu transakcyjnego oraz przy użyciu klasycznych operatorów, w odniesieniu do kolumn składających granice czasookresu dla wymiaru czasu rzeczywistego.

W kwestii zgodności z wytycznymi zawartymi w standardzie SQL:2011, odnośnie składowania i obsługi danych temporalnych dla wymiaru czasu rzeczywistego środowisko MariaDB zapewnia możliwość tworzenia i obsługi tabel wersjonowanych aplikacyjnie z jawnymi znacznikami czasowymi. Nie ma możliwości utworzenia takiej tabeli z niejawnymi znacznikami czasowymi. Możliwa jest także do przeprowadzenia temporalna aktualizacja rekordów oraz temporalne usuwanie rekordów. Niestety nie zostały zaimplementowane predykaty czasowe (zestawione w tabeli 1), które pozwoliłyby na temporalne przetwarzanie danych składowanych w tabelach wersjonowanych aplikacyjnie. Platforma MariaDB obsługuje większość typów danych istotnych dla przetwarzania danych temporalnych, które są uwzględnione w standardzie SQL:2011, oprócz typu danych INTERVAL.

Jednakże należy wspomnieć, że w innych środowiskach RDBMS tj. Oracle, IBM DB2 czy MS SQL Server także niezaimplementowane zostały predykaty czasowe,

uwzględnione w zapisie standardu SQL:2011 (zaprezentowane w tabeli 1), umożliwiające temporalne odpytywanie tabel wersjonowanych czasowo [17-22]. Ponadto platforma MS SQL Server w ogóle nie realizuje obsługi czasu rzeczywistego [3,17,21,22]. Na platformie IBM DB 2 możliwe jest odpytywanie tabel temporalnych za pomocą tej samej grupy predykatów, jak dla obsługi wymiaru czasu systemowego w połączeniu z klauzulą FOR BUSINESS_TIME [19-22]. Podobnie platforma Oracle pozwala na realizację zapytań w odniesieniu do tabeli temporalnej wersjonowanej aplikacyjnie za pomocą dwóch spośród trzech rozszerzeń składniowych uwzględnionych w standardzie SQL:2011: AS OF PERIOD FOR oraz VERSIONS BETWEEN PERIOD FOR [5,18,21,22]. W odróżnieniu od platformy Oracle, gdzie możliwe jest zdefiniowanie kilku okresów czasu, platforma MariaDB pozwala zdefiniować tylko jeden czasookres. MariaDB nie obsługuje wielokrotnego usuwania rekordów. W środowisku MariaDB niemożliwe jest zdefiniowanie czasookresu z niejawnymi znacznikami dla tabel wersjonowanych aplikacyjnie. Funkcjonalność taka dostępną jest w środowisku Oracle.

Artykuł ten poświęcony został przetwarzaniu danych temporalnych na platformie MariaDB ze szczególnym uwzględnieniem rejestracji zmian w wymiarze czasu rzeczywistego. Kolejnym etapem pracy będzie zbadanie stopnia realizacji założeń temporalnych rozszerzeń standardu SQL:2011, w wybranych komercyjnych i niekomercyjnych rozwiązaniach dostępnych na rynku oraz prezentacja zaimplementowanych w nich funkcjonalności temporalnych.

Autorzy: dr inż. Sebastian Łacheciński, Uniwersytet Łódzki, Instytut Logistyki i Informatyki, Katedra Informatyki Ekonomicznej i Medycznej, ul. Rewolucji 1905 r. 37, 90-214 Łódź, E-mail: sebastian.lachecinski@uni.lodz.pl

LITERATURA

- [1] Łacheciński S., Modelowanie danych temporalnych w relacyjnym modelu danych, *Informatyka Ekonomiczna*, 4(46) (2017), 90-107
- [2] Łacheciński S., Składowanie i przetwarzanie danych temporalnych w świetle wymagań standardu języka SQL ISO-IEC 9075, *Przegląd Elektrotechniczny*, 96 (2020), nr 10, 184-191
- [3] Łacheciński S., Obsługa danych temporalnych na platformie MS SQL Server i Azure SQL Database, *Przegląd Elektrotechniczny*, 96 (2020), nr 12, 95-101
- [4] Łacheciński S., Obsługa danych temporalnych dla wymiaru czasu transakcyjnego na platformie ORACLE, *Przegląd Elektrotechniczny*, 97 (2021), nr 11, 159-163
- [5] Łacheciński S., Obsługa danych temporalnych dla wymiaru czasu rzeczywistego na platformie ORACLE, *Przegląd Elektrotechniczny*, 97 (2021), nr 12, 86-91
- [6] Łacheciński S., Składowanie danych temporalnych dla wymiaru czasu transakcyjnego na platformie MariaDB, *Przegląd Elektrotechniczny*, 98 (2022), nr 10, 28-33
- [7] Łacheciński S., Przetwarzanie danych temporalnych dla wymiaru czasu transakcyjnego na platformie MariaDB, *Przegląd Elektrotechniczny*, 98 (2022), nr 11, 307-312
- [8] Łacheciński S., Składowanie danych temporalnych dla wymiaru czasu rzeczywistego na platformie MariaDB, *Przegląd Elektrotechniczny*, 98 (2022), nr 12, 124-128
- [9] Date C.J., Darwen H., Lorentzos N., *Time and relational theory Temporal Databases in the Relational Model and SQL*, 2014, Morgan Kaufmann
- [10] Kulkarni K., Jan-Eike Michels, 2012, *Temporal features in SQL:2011*, 34-43 <https://sigmodrecord.org/publications/sigmodRecord/1209/pdfs/07.industry.kulkarni.pdf>
- [11] Temporal extension SQL: https://link.springer.com/content/pdf/10.1007%2F978-1-4899-7993-3_80729-1.pdf
- [12] MariaDB Temporal Tables: <https://mariadb.com/kb/en/temporal-tables/>
- [13] MariaDB Without overlaps: <https://mariadb.com/kb/en/application-time-periods/#without-overlaps>
- [14] MariaDB Updating by portion: <https://mariadb.com/kb/en/application-time-periods/#updating-by-portion>
- [15] MariaDB Deletion by portion: <https://mariadb.com/kb/en/application-time-periods/#deletion-by-portion>
- [16] MariaDB Bitemporal tables: <https://mariadb.com/kb/en/bitemporal-tables/>
- [17] MS SQL Server Temporal tables: <https://docs.microsoft.com/pl-PL/sql/relational-databases/tables/temporal-tables?view=sqlserver-ver16>
- [18] Temporal Validity in Oracle Database 12c Release 1 (12.1): <https://oracle-base.com/articles/12c/temporal-validity-12cr1>
- [19] IBM DB2 10 Querying application-period temporal data: <https://www.ibm.com/docs/en/db2/10.1.0?topic=appt-querying>
- [20] IBM DB2 13 Querying temporal tables: <https://www.ibm.com/docs/en/db2-for-zos/13?topic=tables-querying-temporal>
- [21] Petković D., Support of Temporal Data in Database Systems, *International Journal of Computer Applications* (0975 –8887), Volume 152 –No.4, October 2016, 26-33
- [22] Petkovic D., Temporal Data in relational Database Systems: A Comparison, *Conference: WorldCIST (1) 2016, At Recife, Volume 1*