

doi:10.15199/48.2024.06.19

Przetwarzanie danych temporalnych dla wymiaru czasu transakcyjnego na platformie IBM Db2

Streszczenie. Celem artykułu jest wskazanie zakresu implementacji obsługi danych temporalnych dla czasu transakcyjnego w środowisku IBM Db2 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 IBM Db2.

Abstract. The aim of the article is an indication the scope of the implementation of temporal data support for transaction time in the IBM Db2 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 IBM Db2 environment. (**Processing of temporal data for the transaction time on the IBM Db2 platform**).

Słowa kluczowe: temporalne bazy danych, temporalne tabele, temporalne operatory, czas transakcyjny, SQL:2011, IBM Db2.

Keywords: temporal databases, temporal tables, temporal operators, transaction time, SQL:2011, IBM Db2.

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 temporalne w RDBMS [2], a także możliwościom składowania i przetwarzania danych temporalnych na platformach MS SQL Server i Azure SQL Database [3], Oracle [4,5], składowaniu i przetwarzaniu danych temporalnych na platformie MariaDB w wymiarze czasu transakcyjnego [6,7] i czasu rzeczywistego [8,9] oraz składowaniu danych temporalnych na platformie IBM Db2 w wymiarze czasu transakcyjnego [10]. Artykuł ten jest kontynuacją rozważań nt. możliwości składowania i przetwarzania danych temporalnych oferowanych przez serwery SQL. W całości poświęcony został on możliwościom przetwarzania danych temporalnych na platformie IBM Db2 z uwzględnieniem wymiaru czasu transakcyjnego. Przeprowadzona została analiza zakresu możliwości przetwarzania danych temporalnych dostępnych na platformie IBM Db2, dla czasu transakcyjnego. 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 transakcyjnego w środowisku IBM Db2.

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 [11-13]:

- 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 IBM Db2

Możliwość składowania i przetwarzania danych temporalnych w środowisku IBM Db2 została wprowadzona w 2012 roku, w wersji 10.1 (wówczas jeszcze IBM DB2) [14]. Od samego początku wprowadzenia rozszerzeń temporalnych standardu SQL:2011 środowisko IBM Db2 obsługuje zarówno tabele temporalne wersjonowane systemowo dla wymiaru czasu transakcyjnego oraz tabele temporalne wersjonowane aplikacyjnie dla wymiaru czasu rzeczywistego. Możliwe jest także utworzenie tabel bitemporalnych z jednoczesną obsługą obu wymiarów czasu [15].

Czas transakcyjny na platformie IBM Db2 nazywany jest czasem systemowym SYSTEM_TIME. Aktualne wersje rekordów przechowywane są w tabeli temporalnej wersjonowanej systemowo, zaś ich poprzednie wersje umieszczane są w tabeli historii (o identycznym schemacie jak tabela temporalna), która jest powiązana z tabelą temporalną. Dzięki takiemu rozwiązaniu możliwe jest tworzenie zapytań temporalnych bazujących zarówno na danych bieżących jak również danych historycznych. Schemat tabeli temporalnej wersjonowanej systemowo zawiera trzy dodatkowe pola. Dwa z nich przechowują wartości graniczne czasookresu, w którym wersja rekordu była wersją aktualną. Trzecie pole zawiera wartość identyfikatora początku transakcji, co pozwala w sposób jednoznaczny identyfikować rekordy temporalne modyfikowane przez konkretną transakcję. Wartości przechowywane w tych polach są automatycznie generowane i zarządzane przez system. Użytkownik w żaden sposób nie może ingerować w te wartości. Zmiana tych wartości automatycznie zachodzi w chwili dodania nowego rekordu temporalnego do tabeli temporalnej wersjonowanej systemowo bądź też w momencie jego modyfikacji lub też usunięcia [16].

Zestawienie wybranych cech temporalnych zawartych w standardzie SQL:2011 oraz cech temporalnych zaimplementowanych na platformie IBM Db2, w odniesieniu do obsługi danych temporalnych w wymiarze czasu transakcyjnego przedstawiono w tabeli 1.

Platforma IBM Db2 wykazuje praktycznie pełną zgodność ze specyfikacją standardu SQL:2011 w kontekście obsługi danych temporalnych, w wymiarze czasu transakcyjnego. Wyjątkiem jest tutaj brak implementacji systemowego typu danych INTERVAL. Możliwe jest tworzenie tabel temporalnych wersjonowanych systemowo zarówno z użyciem jawnych jak i niejawnych

znaczników czasowych. Ponadto zaimplementowane zostały wszystkie trzy predykaty temporalne uwzględnione w zapisach standardu SQL:2011, które pozwalają na przetwarzanie danych temporalnych składowanych w tabelach temporalnych wersjonowanych systemowo.

Tabela 1. Spełnienie wybranych wymogów standardu SQL:2011, dla obsługi danych temporalnych w wymiarze czasu transakcyjnego

Wymagania	SQL:2011	IBM Db2 11.5.9
Definicja czasookresu	✓	✓
Tabele wersjonowane systemowo	✓	✓
Implementacja predykatów czasowych TWS ¹	✓	✓
Jawne znaczniki czasowe	✓	✓
Niejawne znaczniki czasowe	✓	✓
Typ danych data	DATE, TIME, TIMESTAMP, INTERVAL	DATE, TIME, TIMESTAMP

Manipulacja danymi temporalnymi

Instrukcja INSERT pozwala na dodanie rekordów do tabeli temporalnej wersjonowanej systemowo. Nowo dodany rekord jest jednocześnie aktualną wersją rekordu [17]. Poniżej przedstawiono przykład dodania kilku rekordów do tabeli temporalnej z obsługą czasu transakcyjnego z użyciem jedno i wielopoleceniowej postaci instrukcji INSERT. Rekordy zostały dodane do tabeli temporalnej w różnych chwilach czasowych.

```
INSERT INTO
pracownik_tt(IdPracownika,Imie,Nazwisko,Miejscowosc)
VALUES (1,'Jan','Kowalski','Łódź');
```

```
INSERT INTO
pracownik_tt(IdPracownika,Imie,Nazwisko,Miejscowosc)
VALUES (2,'Joanna','Kowalewska','Kraków');
```

```
INSERT INTO
pracownik_tt(IdPracownika,Imie,Nazwisko,Miejscowosc)
VALUES
(3,'Katarzyna','Nowakowska','Olkusz'),
(4,'Adam','Grzegorzczak','Gdynia'),
(5,'Alicja','Jarebska','Jelenia Góra'),
(6,'Julia','Widawska','Kraków');
```

```
INSERT INTO
pracownik_tt(IdPracownika,Imie,Nazwisko,Miejscowosc)
VALUES (7,'Wiktoria','Koszevska','Zgierz');
```

Na rysunku 1 przedstawiono zawartość tabeli temporalnej po dodaniu rekordów. Oprócz wartości dodanych za pomocą instrukcji INSERT uwidocznione zostały także wartości jawnych znaczników czasowych, wygenerowane automatycznie przez silnik bazodanowy dla pól przechowujących granice czasookresu SYSCZASSTART i SYSCZASKONIEC oraz wartość identyfikatora rozpoczęcia transakcji SYSTRANID.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRANID
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olkusz	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Koszevska	Zgierz	2024-02-13 17:58:10.954954	9999-12-30 00:00:00.0	2024-02-13 17:58:10.954954

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

¹ TWS-tabela wersjonowana systemowo

Kolumna SYSCZASSTART przechowuje wartość daty i czasu odpowiadającą momentowi dodania rekordu do tabeli temporalnej. W kolumnie SYSCZASKONIEC składowana jest maksymalna wartość dopuszczalna dla systemowego typu danych TIMESTAMP pomniejszona o jeden dzień (w celu uniknięcia przepełnienia do 5-cyfrowego roku przy ewentualnej konwersji tej wartości na inną strefę czasową). Wartość identyfikatora rozpoczęcia transakcji pokrywa się z wartością kolumny SYSCZASSTART dla nowo dodanego rekordu. Jeżeli w ramach danej transakcji dodawanych jest kilka rekordów, wówczas wszystkie z nich mają przypisaną tą samą wartość SYSTRANID (można to zaobserwować dla grupy rekordów o wartości identyfikatora z zakresu 3÷6).

Instrukcja UPDATE pozwala na przeprowadzenie aktualizacji bieżących wersji rekordów w tabeli temporalnej wersjonowanej systemowo, które stają się jednocześnie nowymi aktualnymi wersjami rekordów. Poprzednie wersje rekordów umieszczane są automatycznie w powiązanej z tabelą temporalną tabeli historii. Historyczna wersja rekordu jest generowana w ramach tej samej transakcji, która aktualizuje dotychczasowy rekord bieżący w tabeli temporalnej. Jeżeli pojedyncza transakcja przeprowadza kilka aktualizacji w obrębie tego samego rekordu, wówczas w tabeli historii generowany jest tylko jeden rekord historyczny, który odzwierciedla stan rekordu przed wprowadzeniem jakichkolwiek zmian przez tą transakcję [18]. Poniżej przedstawiono przykład modyfikacji pojedynczego rekordu składowanego w tabeli temporalnej:

```
UPDATE pracownik_tt SET Miejscowosc='Szklarska Poręba'
WHERE IdPracownika=7;
```

Na rysunku 2 przedstawiono aktualną wersję rekordu oraz jego wersję poprzednią sprzed aktualizacji. Ponadto uwidoczniona została zawartość kolumn definiujących czasookres oraz wartość identyfikatora rozpoczęcia transakcji.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRANID
7	Wiktoria	Koszevska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954
7	Wiktoria	Koszevska	Szklarska Poręba	2024-02-13 18:21:47.874217	9999-12-30 00:00:00.0	2024-02-13 18:21:47.874217

Rys.2. Aktualna i poprzednia wersja rekordu sprzed i po jego aktualizacji, źródło: opracowanie własne

W kolumnie SYSCZASSTART dla nowej wersji rekordu generowana jest wartość znacznika czasowego odpowiadająca chwili czasowej modyfikacji tego rekordu. W kolumnie SYSCZASKONIEC generowana jest maksymalna wartość dla typu TIMESTAMP (pomniejszona o jeden dzień). Poprzednia wersja rekordu jest utrwalana w tabeli historii. Wartość kolumny SYSCZASKONIEC dla wersji historycznej rekordu odpowiada wartości SYSCZASSTART dla wersji aktualnej rekordu. Podobnie wartość identyfikatora rozpoczęcia transakcji SYSTRANID dla nowej wersji rekordu odpowiada wartości SYSCZASKONIEC dla jego poprzedniej wersji. Konwencjonalna postać zapytania o dane bieżące, w swoich wynikach będzie uwzględniać zaktualizowaną, bieżącą wersję rekordu. Poprzednia wersja rekordu sprzed modyfikacji, będzie dostępna dla zapytań, których konstrukcja zawiera specjalizowane predykaty temporalne.

Instrukcja DELETE umożliwia usunięcie bieżących wersji rekordów z tabeli temporalnej. Usunięty rekord z tabeli temporalnej zostaje przeniesiony do powiązanej z nią tabeli historii. Kopia usuwanego rekordu jest umieszczana w tabeli historii zanim rekord ten zostanie fizycznie usunięty z tabeli temporalnej. Jednocześnie poprzednia wersja rekordu nie jest dostępna dla klasycznej postaci zapytania wybierającego. Dostęp do jego poprzednich wersji możliwy jest za pomocą zapytania wybierającego z użyciem

predykatów temporalnych [19]. Poniżej przedstawiono przykład usunięcia rekordu z tabeli temporalnej.

```
DELETE pracownik_tt WHERE IdPracownika=7;
```

Na rysunku 3 przedstawiono poprzednie wersje rekordu sprzed aktualizacji, po jego aktualizacji i usunięciu. Ponadto uwidoczniła została zawartość kolumn definiujących czasookres oraz wartość identyfikatora rozpoczęcia transakcji.

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRANID
7	Witona	Koszevska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954
7	Witona	Koszevska	Sokulska Poreba	2024-02-13 18:21:47.874217	2024-02-13 18:50:11.639293	2024-02-13 18:21:47.874217

Rys.3. Temporalne wersje rekordu sprzed i po jego aktualizacji oraz po jego usunięciu, źródło: opracowanie własne

Wartość w kolumnie SYSCZASKONIEC modyfikowana jest na wartość odpowiadającą chwili czasowej usunięcia rekordu z tabeli temporalnej. Wartości kolumn SYSCZASSTART i SYSTRANID nie podlegają zmianie.

Obsługa zapytań

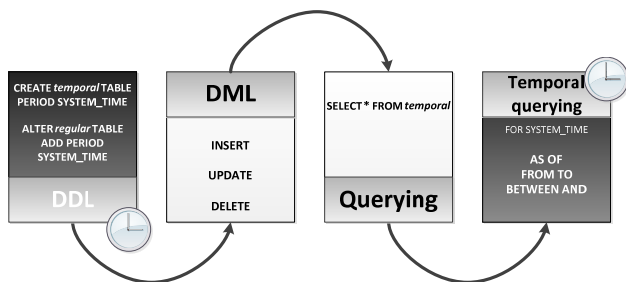
Tabele temporalne wersjonowane systemowo mogą być odpytywane za pomocą standardowej postaci instrukcji SELECT. W takim przypadku zawsze zwracane będą aktualne wersje rekordów. Natomiast użycie specjalnych predykatów temporalnych w połączeniu z instrukcją SELECT pozwala uzyskać wgląd również w poprzednie wersje rekordów. W zależności od rodzaju użytego predykatu temporalnego, w zapytaniu wybierającym wynikowy zbiór rekordów będzie mógł składać się z zarówno bieżących jak i historycznych wersji rekordów bądź też z tylko historycznych wersji rekordów. Wybór konkretnego predykatu temporalnego oraz jego odpowiednia parametryzacja będzie determinować obszar skanowania tabeli (tylko tabela temporalna, tylko tabela historyczna, tabela temporalna i tabela historyczna).

W środowisku IBM Db2 zaimplementowane zostały temporalne predykady do odpytywania tabel temporalnych wersjonowanych systemowo. Liczba i postać zaimplementowanych na platformie IBM Db2 predykatów temporalnych ściśle odpowiada liczbie i postaci predykatów temporalnych dla obsługi tabel temporalnych w wymiarze czasu transakcyjnego zawartych w zapisie standardu SQL:2011 [2,8,20]. Zestawienie predykatów temporalnych uwzględnionych w standardzie SQL:2011 oraz tych dostępnych na platformie IBM Db2 przedstawiono w tabeli 2.

Tabela 2. Rozszerzenia składniowe do przetwarzania danych składowanych w tabelach temporalnych wersjonowanych systemowo na platformie IBM Db2

SQL:2011	IBM Db2-TT
AS OF	AS OF
FROM TO	FROM TO
BETWEEN AND	BETWEEN AND

Na rysunku 4 przedstawiony został sposób pracy z tabelami temporalnymi wersjonowanymi systemowo w środowisku IBM Db2 z obsługą czasu transakcyjnego.



Rys.4. Sposób pracy z tabelami temporalnymi wersjonowanymi systemowo w IBM Db2, źródło: opracowanie własne

Zapytania na tabelach wersjonowanych systemowo z obsługą czasu transakcyjnego

Zapytania do tabel temporalnych wersjonowanych systemowo mogą być konstruowane za pomocą klasycznej postaci instrukcji wybierającej, bądź też postać ta może być rozbudowana o klauzulę FOR SYSTEM_TIME w połączeniu z jednym z predykatów temporalnych zestawionych w tabeli 2.

Klasyczne odpytanie tabeli temporalnej, bez użycia rozszerzenia składniowego o predykady temporalne zwraca tylko i wyłącznie aktualne wersje rekordów. Poniżej przedstawiono przykład użycia klasycznej postaci zapytania wybierającego w odniesieniu do tabeli temporalnej.

```
SELECT * FROM pracownik_tt;
```

Na rysunku 5 przedstawiono zestaw wyników zwracanych przez powyższe zapytanie. Dostępne są tylko aktualne wersje rekordów. Niedostępny jest rekord pracownika o IdPracownika=7, który został usunięty. Przykład ten dotyczy wyników zwracanych z tabeli temporalnej z jawnymi znacznikami czasowymi.

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRANID
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:52.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:52.766122
3	Katarzyna	Nowakowska	Olkusz	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186

Rys.5. Zestaw danych zwróconych przez zapytanie wybierające w klasycznej postaci, bez użycia predykatów temporalnych. Usunięty rekord jest niewidoczny, źródło: opracowanie własne

W przypadku użycia zapytania wybierającego do tabeli temporalnej z niejawnymi znacznikami czasowymi, w zestawie wyników końcowych domyślnie nie są dostępne pola znaczników czasowych oraz identyfikatora rozpoczęcia transakcji, niezależnie od tego czy użyta została klasyczna postać instrukcji SELECT, czy też w połączeniu z temporalnymi predykatami. Poniżej przedstawiono przykład zapytania wybierającego bez użycia rozszerzeń temporalnych na tabeli temporalnej z niejawnymi znacznikami czasowymi.

```
SELECT * FROM pracownik_tt_ih;
```

Na rysunku 6 przedstawiono rezultat działania zapytania wybierającego w klasycznej postaci na tabeli temporalnej z niejawnymi znacznikami czasowymi.

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC
1	Jan	Kowalski	Łódź
2	Joanna	Kowalewska	Kraków
3	Katarzyna	Nowakowska	Olkusz
4	Adam	Grzegorzczak	Gdynia
5	Alicja	Jarebska	Jelenia Góra
6	Julia	Widawska	Kraków

Rys.6. Domyślna postać zwracanych wyników z tabeli temporalnej z niejawnymi znacznikami czasowymi, źródło: opracowanie własne

W celu zwrócenia zawartości pól znaczników czasowych oraz identyfikatora rozpoczęcia transakcji należy na liście pól w instrukcji SELECT podać bezpośrednio ich nazwy. Poniżej przedstawiono przykład zwracający aktualne wersje rekordów z tabeli temporalnej z niejawnymi znacznikami czasowymi, z uwzględnieniem wartości składowanych w kolumnach znaczników czasowych oraz kolumnie identyfikatora rozpoczęcia transakcji.

```
SELECT *,SysCzasStart,SysCzasKoniec,SysTranID FROM pracownik_tt_ih;
```

Na rysunku 7 przedstawiono rezultat działania zapytania wybierającego w klasycznej postaci na tabeli temporalnej z niejawnymi znacznikami czasowymi, z jednoczesnym

zwróceniem zawartości kolumn znaczników czasowych oraz identyfikatora rozpoczęcia transakcji.

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olsztyn	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186

Rys.7. Postać zwracanych wyników z tabeli temporalnej, z niejawnymi znacznikami czasowymi, z jednoczesnym zwróceniem zawartości kolumn znaczników czasowych oraz identyfikatora rozpoczęcia transakcji, źródło: opracowanie własne

Za pomocą predykatu temporalnego AS OF możliwe jest uzyskanie stanu rekordów na zadany moment w czasie, który odpowiada wartości argumentu tego predykatu. Zwracane są wszystkie wersje rekordów, dla których wartość początkowa czasookresu jest mniejsza lub równa zadanej wartości (przypisanej jako argument tego predykatu), a wartość końcowa czasookresu jest od niej większa [2,8,20]. Poniżej przedstawiono przykład zapytania temporalnego z użyciem klauzuli AS OF dla chwili czasowej sprzed modyfikacji nazwy miejscowości pracownika o IdPracownika=7.

```
SELECT * FROM pracownik_tt FOR SYSTEM_TIME AS OF '2024-02-13 18:20:00'
```

Na rysunku 8 przedstawiono rezultat powyższego zapytania. Widoczna jest poprzednia wersja rekordu, która była wersją bieżącą na dany moment w czasie, którego wartość została podana jako argument predykatu AS OF. Ponadto zwrócone zostały bieżące wersje rekordów, które na chwilę obecną nie zostały poddane żadnym przekształceniom. Kolumna SYSCZASKONIEC przechowuje wartość stempla czasowego, do kiedy dana wersja rekordu była wersją obowiązującą.

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olsztyn	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Kozłowska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954

Rys.8. Zestaw wyników zwróconych przez temporalne zapytanie z użyciem predykatu AS OF dla zadanej chwili czasowej. Widoczne są dane bieżące rekordów oraz poprzednia wersja rekordu sprzed jego aktualizacji (aktualna w zadanej chwili czasowej), źródło: opracowanie własne

Poniżej zaprezentowano przykład zapytania temporalnego z użyciem predykatu AS OF dla chwili czasowej po przeprowadzeniu modyfikacji rekordu.

```
SELECT * FROM pracownik_tt FOR SYSTEM_TIME AS OF '2024-02-13 18:22:00';
```

Na rysunku 9 przedstawiono zestaw wyników zwróconych przez zapytanie temporalne, odzwierciedlający stan rekordów na zadany moment w czasie, po zmianie nazwy miejscowości dla pracownika o IdPracownika=7. Informacja o chwili modyfikacji tego rekordu utrwalona została w kolumnie SYSCZASSTART. Jednocześnie jest to wersja rekordu sprzed jego usunięcia z tabeli temporalnej. Moment usunięcia tego rekordu został utrwalony w kolumnie SYSCZASKONIEC.

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olsztyn	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Kozłowska	Śląska Poręba	2024-02-13 18:21:47.874217	2024-02-13 18:50:11.639293	2024-02-13 18:21:47.874217

Rys.9. Zestaw wyników zwróconych przez temporalne zapytanie z użyciem predykatu AS OF dla zadanej chwili czasowej. Widoczne

są dane bieżące rekordów oraz wersja rekordu po jego aktualizacji, źródło: opracowanie własne

Kolejny przykład przedstawia postać zapytania temporalnego z użyciem predykatu AS OF dla chwili czasowej po usunięciu rekordu.

```
SELECT * FROM pracownik_tt FOR SYSTEM_TIME AS OF '2024-02-13 18:55:00';
```

Na rysunku 10 przedstawiono zestaw wyników zwróconych przez zapytanie temporalne, odzwierciedlający stan rekordów na zadany moment w czasie, po usunięciu pracownika o IdPracownika=7. W zestawie zwróconych rekordów usunięty rekord jest niedostępny. Widoczne są tylko bieżące wersje rekordów. W tym konkretnym przypadku, tylko te rekordy, które nie były poddane na chwilę obecną żadnym przekształceniom ani modyfikacjom.

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olsztyn	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186

Rys.10. Zestaw wyników zwróconych przez temporalne zapytanie z użyciem predykatu AS OF dla zadanej chwili czasowej, po usunięciu rekordu. Widoczne są dane bieżące rekordów na dany moment w czasie, źródło: opracowanie własne

Wszystkie poprzednie wersje usuniętego rekordu wraz z wartościami granicznymi czasookresu ich obowiązywania znajdują się w tabeli historycznej. Dostęp do niej można uzyskać za pomocą podstawowej konstrukcji zapytania wybierającego. Poniżej przedstawiono przykład odpytania tabeli historii.

```
SELECT * FROM pracownik_tt_hist;
```

Na rysunku 11 przedstawiono rezultat działania powyższego zapytania. Widoczna jest wersja rekordu sprzed modyfikacji nazwy miejscowości oraz po modyfikacji nazwy miejscowości. W kolumnie SYSCZASSTART widnieje stempel czasowy, którego wartość odpowiada odpowiednio momentowi pojawienia się rekordu w tabeli (dla pierwszej wersji) oraz jego modyfikacji (dla drugiej wersji rekordu). Kolumna SYSCZASKONIEC zawiera stempel czasowy, którego wartość odpowiada chwili czasowej, do kiedy dana wersja była wersją aktualną (dla pierwszej wersji rekordu) oraz kiedy rekord został usunięty z tabeli temporalnej (dla drugiej wersji rekordu).

IDPRACOWNIKA	IMIE	NAZWISKO	MIJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
7	Wiktoria	Kozłowska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954
7	Wiktoria	Kozłowska	Śląska Poręba	2024-02-13 18:21:47.874217	2024-02-13 18:50:11.639293	2024-02-13 18:21:47.874217

Rys.11. Zawartość tabeli historycznej, przechowującej wszystkie poprzednie wersje rekordu. Dostępne są tylko i wyłącznie historyczne wersje rekordu, źródło: opracowanie własne

Za pomocą predykatu temporalnego FROM TO możliwe jest uzyskanie rekordów z konkretnego przedziału czasowego, których wersje były wersjami aktualnymi w danym przedziale czasowym. Należy zaznaczyć, że w wynikowym zbiorze rekordów nieuwzględniane są wersje rekordów, które są aktualne dla chwili czasowej równej górnej granicy czasookresu. Zwracane są wszystkie wersje rekordów, dla których wartość początkowa czasookresu jest mniejsza od górnej wartości granicznej przekazanej jako argument predykatu. Ponadto wartość końcowa czasookresu jest większa od dolnej wartości granicznej przekazywanej jako pierwszy argument predykatu. W przypadku, gdy dolna wartość graniczna zadana jako argument tego predykatu jest większa od górnej wartości granicznej tego predykatu lub gdy te wartości są sobie równe, zapytanie temporalne nie zwróci żadnych rekordów

[2,8,20]. Poniżej przedstawiono przykład zapytania temporalnego z użyciem klauzuli FROM TO.

```
SELECT * FROM pracownik_tt FOR SYSTEM_TIME FROM '2024-02-13 18:00:00' TO '2024-02-13 18:05:00'
```

Na rysunku 12 przedstawiono rezultat działania powyższego zapytania. Zwrócone zostały wersje rekordów, które były aktualne w zadanym przedziale czasu i nadal są wersjami aktualnymi (dotyczy rekordów 1÷6) oraz historyczna wersja ostatniego zwróconego rekordu (sprzed jego modyfikacji), która była wersją bieżącą w zadanym przedziale czasu.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCO...	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olszusz	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Koszevska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954

Rys.12. Wynik działania zapytania na tabeli temporalnej z użyciem predykatu FROM TO, źródło: opracowanie własne

Poniżej przedstawiono przykład zapytania temporalnego z użyciem klauzuli FROM TO z poszerzonym przedziałem czasowym, który obejmuje obie wersje rekordu dla pracownika o IdPracownika=7. Ponadto zwracane są także aktualne wersje rekordów, które są jednocześnie wersjami bieżącym w terażniejszości.

```
SELECT * FROM pracownik_tt FOR SYSTEM_TIME FROM '2024-02-13 18:00:00' TO '2024-02-13 18:25:00';
```

Na rysunku 13 przedstawiono rezultat działania powyższego zapytania. Zwrócone zostały wersje rekordów, które były aktualne w zadanym przedziale czasu i nadal są wersjami aktualnymi (dotyczy rekordów 1÷6) oraz historyczne wersje rekordu (sprzed i po jego modyfikacji), które były wersjami bieżącymi w zadanym przedziale czasu. Czas ważności dla każdej z tych wersji widoczny jest w kolumnie SYSCZASKONIEC.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olszusz	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Koszevska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954
7	Wiktoria	Koszevska	Suklerska Poręba	2024-02-13 18:21:47.874217	2024-02-13 18:50:11.639293	2024-02-13 18:21:47.874217

Rys.13. Wynik działania zapytania na tabeli temporalnej z użyciem predykatu FROM TO, źródło: opracowanie własne

Kolejny przykład dotyczy sytuacji, w której górna granica predykatu temporalnego FROM TO pokrywa się z górną granicą czasookresu rekordu temporalnego składowaną w kolumnie SYSCZASKONIEC, a więc chwili czasowej, w której doszło do modyfikacji rekordu.

```
SELECT * FROM pracownik_tt FOR SYSTEM_TIME FROM '2024-02-13 18:00:00' TO '2024-02-13 18:21:47.874217';
```

Na rysunku 14 przedstawiono rezultat działania powyższego zapytania. Jak można zaobserwować w zestawie wyników końcowych nie została uwzględniona wersja rekordu z chwili czasowej, w której rekord został poddany modyfikacji. Jak wspomniano wcześniej jest to cecha działania predykatu temporalnego FROM TO.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCO...	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olszusz	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Koszevska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954

Rys.14. Wynik działania zapytania na tabeli temporalnej z użyciem predykatu FROM TO, źródło: opracowanie własne

Za pomocą predykatu temporalnego BETWEEN AND możliwe jest uzyskanie zbioru rekordów z zadanego przedziału czasowego. Jednakże w odróżnieniu od predykatu FROM TO w zbiorze wynikowym uwzględniane są także wersje rekordów, które są aktualne dla chwili czasowej odpowiadającej górnej granicy czasookresu. Zwracane są wersje rekordów, dla których czasookres pokrywa się z dowolnym punktem w czasie zdefiniowanym pomiędzy dolną i górną wartością graniczną predykatu BETWEEN AND. Wersje rekordów są zwracane, gdy wartość początkowa czasookresu jest mniejsza lub równa górnej wartości granicznej predykatu oraz gdy wartość końcowa czasookresu jest większa lub równa dolnej wartości granicznej predykatu [2,8,20]. Poniżej przedstawiono przykład zapytania temporalnego z użyciem klauzuli BETWEEN AND, gdzie górna wartość graniczna predykatu pokrywa się z górną wartością graniczną czasookresu rekordu.

```
SELECT * FROM pracownik_tt FOR SYSTEM_TIME BETWEEN '2024-02-13 18:00:00' AND '2024-02-13 18:21:47.874217';
```

Na rysunku 15 przedstawiono rezultat działania powyższego zapytania. W przeciwieństwie do użycia predykatu FROM TO, w zestawie wyników końcowych uwzględniona została również wersja rekordu z chwili czasowej, będącej jednocześnie górną wartością graniczną czasookresu, a więc wtedy, gdy rekord został poddany modyfikacji.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olszusz	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Koszevska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954
7	Wiktoria	Koszevska	Suklerska Poręba	2024-02-13 18:21:47.874217	2024-02-13 18:50:11.639293	2024-02-13 18:21:47.874217

Rys.15. Wynik działania zapytania na tabeli temporalnej z użyciem predykatu BETWEEN AND, źródło: opracowanie własne

Niektórzy dostawcy RDBMS oferują w swoich rozwiązaniach ponadstandardowe predykaty temporalne, które nie zostały uwzględnione w standardzie SQL:2011. Jednym z takich temporalnych rozszerzeń jest predykat ALL, który jest dostępny m.in. na platformach MS SQL Server, Azure SQL Database [3,21] czy MariaDB [7,22]. W środowisku IBM Db2 nie zostały zaimplementowane żadne dodatkowe rozszerzenia temporalne wykraczające poza zakres predykatów temporalnych zawartych w zapisie standardu SQL:2011. Niemniej jednak istnieje możliwość własnej implementacji działania predykatu ALL. Predykat ten zwraca wszystkie wersje rekordów zarówno te historyczne jak i aktualne. Nie wymaga także określania żadnego argumentu w postaci wartości typu data i czas. Poniżej przedstawiono sposób implementacji sposobu działania predykatu temporalnego ALL.

```
SELECT * FROM pracownik_tt
UNION
SELECT * FROM pracownik_tt_hist;
```

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCOWOSC	SYSCZASSTART	SYSCZASKONIEC	SYSTRAND
1	Jan	Kowalski	Łódź	2024-02-10 23:28:53.754805	9999-12-30 00:00:00.0	2024-02-10 23:28:53.754805
2	Joanna	Kowalewska	Kraków	2024-02-10 23:44:53.766122	9999-12-30 00:00:00.0	2024-02-10 23:44:53.766122
3	Katarzyna	Nowakowska	Olszusz	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
4	Adam	Grzegorzczak	Gdynia	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
5	Alicja	Jarebska	Jelenia Góra	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
6	Julia	Widawska	Kraków	2024-02-13 17:54:53.547186	9999-12-30 00:00:00.0	2024-02-13 17:54:53.547186
7	Wiktoria	Koszevska	Zgierz	2024-02-13 17:58:10.954954	2024-02-13 18:21:47.874217	2024-02-13 17:58:10.954954
7	Wiktoria	Koszevska	Suklerska Poręba	2024-02-13 18:21:47.874217	2024-02-13 18:50:11.639293	2024-02-13 18:21:47.874217

Rys.16. Wynik działania zapytania będącego odpowiednikiem zapytania z użyciem predykatu ALL, źródło: opracowanie własne

Na rysunku 16 przedstawiono rezultat działania powyższego zapytania. Zwracana jest suma zbiorów

rekordów pochodzących z tabeli temporalnej (wszystkie wersje aktualne rekordów) oraz tabeli historycznej (wszystkie wersje historyczne rekordów).

Podsumowanie

Artykuł ten poświęcony został możliwościom przetwarzania danych temporalnych w wymiarze czasu transakcyjnego, w środowisku IBM Db2.

Dodanie rekordu do tabeli temporalnej wersjonowanej systemowo odbywa się w identyczny sposób jak dla klasycznej postaci tabeli nietemporalnej. Nie ma możliwości zaproponowania własnych wartości dla kolumn znaczników czasowych przechowujących wartości graniczne czasookresu, w którym dana wersja rekordu była wersją bieżącą a także wartości identyfikatora rozpoczęcia transakcji, która utworzyła lub zmodyfikowała dany rekord. Wartości te generowane są automatycznie przez system, bez udziału użytkownika, który nie ma żadnej możliwości ingerowania w te wartości. Bieżące wersje rekordów składowane są w tabeli temporalnej [17].

Aktualizacja jak również usuwanie rekordów składowanych w tabeli temporalnej wersjonowanej systemowo przeprowadzane są za pomocą standardowej postaci instrukcji UPDATE i DELETE. W celu zachowania zgodności z wytycznymi zawartymi w zapisie standardu SQL:2011 aktualizacji i usuwaniu podlegają wersje bieżące rekordów temporalnych, które wskutek tych działań stają się wersjami historycznymi. Każda akcja modyfikacji lub usunięcia rekordu w tabeli temporalnej powoduje przeniesienie wersji tego rekordu sprzed modyfikacji do powiązanej z nią tabeli historii [18,19]. Usunięcie historycznych wersji rekordów skutkowałoby utratą ciągłości historii zmian, a także byłoby sprzeczne z założeniami standardu SQL:2011, w odniesieniu do magazynowania i przetwarzania danych temporalnych w wymiarze czasu transakcyjnego. Domyślnie tabela historii usuwana jest w momencie usuwania tabeli temporalnej, z którą jest powiązana. W uzasadnionych przypadkach tabela historii może pozostać w systemie, w momencie usuwania tabeli temporalnej, jeżeli została ona utworzona z opcją RESTRICT ON DROP.

Z punktu widzenia zgodności i zakresu implementacji rozszerzeń temporalnych zawartych w zapisach standardu SQL:2011 dla wymiaru czasu transakcyjnego platforma IBM Db2 umożliwia tworzenie tabel wersjonowanych systemowo z jawnymi i niejawnymi znacznikami czasowymi. Zaimplementowane zostały wszystkie predykaty temporalne do obsługi zapytań temporalnych zawarte w standardzie SQL:2011. Obsługiwane są także trzy spośród czterech typów danych (przechowujących wartości data i czas) poza typem INTERVAL. W oparciu o kolumny przechowujące początek i koniec przedziału czasowego tworzona jest definicja czasookresu [2,8,20].

Porównując IBM Db2 do konkurencyjnych rozwiązań RDBMS pod kątem dostępnych predykatów temporalnych platforma IBM Db2 wypada korzystniej od ORACLE (gdzie zaimplementowane zostały dwa spośród trzech predykatów o nieco zmienionej składni) [4]. Natomiast platforma MariaDB dodatkowo udostępnia jeden ponadstandardowy predykat ALL [7,22], a środowisko MS SQL Server i Azure SQL Database udostępnia dwa ponadstandardowe predykaty temporalne ALL i CONTAINED IN [3,21].

Należy także zwrócić uwagę na fakt, że przechowywanie wszystkich poprzednich wersji rekordów temporalnych wiąże się z dużym przyrostem danych. W uzasadnionych sytuacjach można rozważyć usunięcie części danych historycznych, co pozwoli zmniejszyć zajętość przestrzeni dyskowej. Zazwyczaj takie usuwanie danych historycznych odbywa się według określonych

strategii. Strategia taka może polegać m.in. na: usuwaniu historycznych wersji rekordów starszych niż zadana wartość, zachowywaniu tylko zadanej liczby najnowszych wersji rekordów historycznych, usunięciu wszystkich wersji rekordów historycznych w momencie usunięcia ich wersji bieżącej z tabeli temporalnej. Działania takie są możliwe do przeprowadzenia pod warunkiem posiadania odpowiednich uprawnień. Podobne rozwiązania pozwalające zarządzać czasem retencji danych dostępne są także na innych platformach bazodanowych tj. MS SQL Server [3,25], ORACLE [4,26], MariaDB [6,7,27].

Kolejnym etapem pracy będzie zbadanie stopnia realizacji założeń rozszerzeń temporalnych standardu SQL:2011 na platformie IBM Db2 w kontekście obsługi danych temporalnych dla wymiaru czasu rzeczywistego oraz prezentacja zaimplementowanych 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] Łacheciński S., Przetwarzanie danych temporalnych dla wymiaru czasu rzeczywistego na platformie MariaDB, *Przegląd Elektrotechniczny*, 99 (2023), nr 6/2023, 111-118
- [10] Łacheciński S., Składowanie danych temporalnych dla wymiaru czasu transakcyjnego na platformie IBM Db2, *Przegląd Elektrotechniczny*, 100 (2024), nr 1/2024, 185-188
- [11] Date C.J., Darwen H., Lorentzos N., *Time and relational theory Temporal Databases in the Relational Model and SQL*, 2014, Morgan Kaufmann
- [12] Kulkarni K., Jan-Eike Michels, 2012, *Temporal features in SQL:2011*, 34-43 <https://sigmodrecord.org/publications/sigmodRecord/1209/pdfs/07.industry.kulkarni.pdf>
- [13] Temporal extension SQL: https://link.springer.com/content/pdf/10.1007%2F978-1-4899-7993-3_80729-1.pdf
- [14] IBM Db2 WhatsNewDB2: https://public.dhe.ibm.com/ps/products/db2/info/vr101/pdf/en_US/DB2WhatsNew-db2q0e1011.pdf?_gl=1*a37ez1*_ga*NTI0OTc3MzcxLjE2ODg5ODYyMzE.*_ga_FYECCCS21D*MTY5MTkyNDQ0NC4yMS4wLjE2OTE5MjQ0NTYuMC4wLjA
- [15] IBM Db2 Temporal Tables: <https://www.ibm.com/docs/en/db2-for-zos/12?topic=tables-temporal-data-versioning>
- [16] IBM Db2 Creating system period table <https://www.ibm.com/docs/en/db2-for-zos/12?topic=tables-creating-system-period-temporal-table>

- [17] IBM Db2 Inserting data into a system-period temporal table:
<https://www.ibm.com/docs/en/db2/11.5?topic=tables-inserting-data>
- [18] IBM Db2 Updating data in a system-period temporal table:
<https://www.ibm.com/docs/en/db2/11.5?topic=tables-updating-data>
- [19] IBM Db2 Deleting data from a system-period temporal table:
<https://www.ibm.com/docs/en/db2/11.5?topic=tables-deleting-data>
- [20] IBM Db2 Querying system-period temporal data:
<https://www.ibm.com/docs/en/db2/11.5?topic=tables-querying>
- [21] Microsoft Query data in a system-versioned temporal table:
<https://learn.microsoft.com/en-us/sql/relational-databases/tables/querying-data-in-a-system-versioned-temporal-table?view=sql-server-ver16>
- [22] MariaDB Querying Historical Data:
<https://mariadb.com/kb/en/system-versioned-tables/#queryinghistorical-data>
- [23] Petković D., Support of Temporal Data in Database Systems, International Journal of Computer Applications (0975 –8887), Volume 152 –No.4, October 2016, 26-33
- [24] Petkovic D., Temporal Data in relational Database Systems: A Comparison, Conference: WorldCIST (1) 2016, At Recife, Volume 1
- [25] MS SQL Server Retention:
<https://docs.microsoft.com/enus/sql/relational-databases/tables/manage-retention-ofhistorical-data-in-system-versioned-temporal-tables?view=sqlserver-ver15>
- [26] Database Development Guide, Using Oracle FlashbackTechnology:
https://docs.oracle.com/database/121/ADFNS/adfn_flashback.htm#ADFNS1008
- [27] MariaDB Removing old history:
<https://mariadb.com/kb/en/system-versioned-tables/#removing-old-history>