

doi:10.15199/48.2021.12.15

Obsługa danych temporalnych dla wymiaru czasu rzeczywistego na platformie ORACLE

Streszczenie. Celem artykułu jest wskazanie zakresu implementacji obsługi danych temporalnych dla czasu rzeczywistego w środowisku ORACLE 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 obsługi danych temporalnych w środowisku Oracle.

Abstract. The aim of the article is an indication the scope of the implementation of temporal data support for valid time in the ORACLE environments 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 of handling temporal data by Oracle environment. (**Support of temporal data on the ORACLE platform**).

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

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

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 oraz Azure SQL Database [3] i Oracle z obsługą czasu transakcyjnego [4]. Artykuł ten jest kontynuacją rozważań nt. możliwości składowania i przetwarzania danych temporalnych oferowanych przez serwery SQL. W całości poświęcony został on obsłudze danych temporalnych na platformie Oracle z uwzględnieniem wymiaru czasu rzeczywistego. Przeprowadzona została analiza zakresu możliwości obsługi danych temporalnych dostępnych na platformie Oracle 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 obsługi danych temporalnych z użyciem czasu rzeczywistego w środowisku Oracle.

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 [5,6,7]:

- 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 ORACLE

Częściowa obsługa danych temporalnych w środowisku Oracle stała się możliwa wraz z wprowadzeniem mechanizmu Oracle Flashback w wersji 9i serwera Oracle. Technologia ta została znacznie rozszerzona w wersji 10g serwera Oracle. W wersji 11g serwera Oracle wprowadzone zostały kolejne rozszerzenia tej technologii. Dodano wówczas m.in. kolejny moduł Flashback Data Archive, który obsługiwał tylko czas transakcyjny (transaction time). Moduł ten został szerzej zaprezentowany w pracy [4] Pełna obsługa danych temporalnych na platformie Oracle dostępna jest od wersji 12c (dokładnie od 12.1.01). W tej wersji zaimplementowana została obsługa czasu rzeczywistego (valid time) oparta na komponencie Temporal Validity.

Komponenty dostępne w środowisku Oracle do obsługi danych temporalnych

Środowisko Oracle udostępnia dwa moduły do obsługi danych temporalnych:

- Flashback Data Archive (FDA),
- Temporal Validity (TV).

Temporal Validity

Funkcjonalność ta została wprowadzona w wersji 12c serwera Oracle i wspiera obsługę czasu rzeczywistego. Pozwala skojarzyć z tabelą temporalną jeden lub kilka wymiarów czasu rzeczywistego. Możliwe jest przetwarzanie danych w zależności od ich ważności (aktualności) na dany moment w czasie [8].

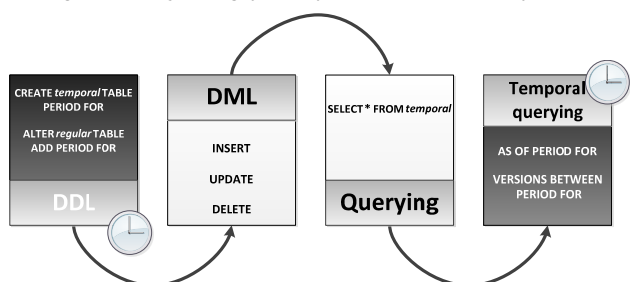
W celu implementacji obsługi czasu VT do tabeli temporalnej dołączane są dwie dodatkowe kolumny (typu date lub timestamp) będące odpowiednio początkiem i końcem czasookresu, dla których dane składowane w poszczególnych rekordach są obowiązujące.

Wymiar czasu rzeczywistego do tabeli temporalnej może być wprowadzony w sposób jawny poprzez wprowadzenie dodatkowych kolumn typu data i godzina lub też w sposób niejawny, wówczas kolumny takie tworzone są automatycznie. W przypadku nieokreślenia wprost kolumn składających wartości graniczne czasookresu, kolumny takie zostają tworzone automatycznie przez bazę Oracle z domyślnymi nazwami: valid_time_column_START i valid_time_column_END. Ponadto kolumny definiujące czasookres mogą być widoczne lub niewidoczne dla użytkownika [9].

Obsługa wymiaru czasu VT jest szczególnie istotna z punktu widzenia procesu zarządzania cyklem życia informacji ILM (ang. Information Lifecycle Management) a także w każdej aplikacji, w której istotna jest wiedza, kiedy określone dane (z punktu widzenia aplikacji) są ważne (aktualne na dany moment) lub staną się ważne i kiedy przestaną być ważne-jeśli w ogóle stracą swoją ważność.

Jako przykład zastosowania można wymienić m.in.: daty zatrudnienia i zwolnienia (wypowiedzenia umowy o pracę) pracownika, data wejścia w życie ochrony polisy ubezpieczeniowej, okres obowiązywania cen akcji na giełdzie, kursu walut, czy też data wejścia w życie zmiany adresu klienta.

Funkcjonalność udostępniona przez moduł Temporal Validity znacznie redukuje także złożoność kodu aplikacji, dostarczając prosty deklaracyjny interfejs, za pomocą którego aplikacje mogą zarządzać poprawnością rekordów.



Rys.1. Sposób pracy z tabelami czasowymi wersjonowanymi aplikacyjnie w Oracle, źródło: opracowanie własne

Zestawienie wybranych cech temporalnych standardu SQL:2011 oraz tych zaimplementowanych na platformie Oracle przedstawiono w tabeli 1.

Tabela 1. Spełnienie wybranych wymogów standardu SQL:2011

Wymagania	SQL:2011	ORACLE 12c
Definicja czasookresu	✓	✓
Typ danych PERIOD	-	-
Tabele wersjonowane aplikacyjnie	✓	✓
Tabele wersjonowane systemowo	✓	✓
Tabele bitemporalne	✓	✓
Aktualizacja i usuwanie rekordów z zadanego przedziału czasowego	✓	-
Temporalne ograniczenie PK	✓	-
Temporalne ograniczenie FK REFERNCES	✓	-
Implementacja predykatów czasowych TWA ¹	✓	✓/-
Implementacja predykatów czasowych TWS ²	✓	✓/-
Jawne znaczniki czasowe	✓	✓
Niejawne zanczniki czasowe	✓	✓
Typ danych data	DATE, TIME, TIMESTAMP, INTERVAL	DATE, TIMESTAMP

Niestety w kolejnych wersjach serwera Oracle (18c i 19c) nie zostały wprowadzone dalsze udogodnienia obsługi i przetwarzania danych temporalnych. Nadal brakuje

¹ TWA-tabela wersjonowana aplikacyjnie

² TWS-tabela wersjonowana systemowo

obsługi temporalnej aktualizacji i temporalnego usuwania rekordów, integralności temporalnej, złączeń temporalnych, temporalnych ograniczeń nakładanych na pola klucza głównego PK i klucza obcego FK, czy też agregacji temporalnych. Typ danych PERIOD także nie jest zaimplementowany.

Tworzenie tabeli temporalnej z obsługą czasu rzeczywistego w ORACLE

Każda tabela temporalna wersjonowana aplikacyjnie musi posiadać dwa pola typu: date lub timestamp. W tabeli 2 zestawiono wykaz systemowych typów danych kategorii data i czas dostępnych na platformie Oracle wraz z zakresem składowanych wartości [10].

Tabela 2. Systemowe typy danych kategorii data i czas dostępne na platformie ORACLE

Typ danych	Zakres	Dokładność
Date	1.1.4712 p.n.e +31.12.9999	1s
Timestamp	1.1.4712 p.n.e +31.12.9999	0s+1ns
Timestamp with time zone	1.1.4712 p.n.e +31.12.9999.	0s+1ns
Timestamp with local time zone	1.1.4712 p.n.e +31.12.9999	0s+1ns

Typy danych Timestamp przechowują wiek, rok (w postaci czterocyfrowej), miesiąc, dzień, godzinę (w formacie 24 godzinnym), minuty i sekundy. Rozszerzenie tego typu o klauzulę: *with time zone* dodatkowo przechowuje strefę czasową. Rozszerzenie: *with local time zone* przechowuje strefę czasową dla konkretnej wartości daty i czasu przekonwertowaną do lokalnej strefy czasowej bazy danych. Jest ono szczególnie przydatne w zastosowaniach globalnych tzn. w przypadkach gdzie dany system informatyczny jest używany w różnych częściach świata, dzięki czemu lokalni użytkownicy widzą wartości daty i czasu dostosowane do swoich indywidualnych stref czasowych.

W tabeli 3 zestawiono wykaz systemowych typów danych interwałów czasowych dostępnych na platformie Oracle wraz z zakresem składowanych wartości [10].

Tabela 3. Systemowe typy danych interwałów dostępne na platformie ORACLE

Typ danych	Dokładność
Interval year to month	Precyzja lat (liczba całkowita) 0-9
Interval day to second	Precyzja dni (liczba całkowita) 0-9, Precyzja części ułamkowej sekundy 0-9

Interwały czasowe rejestrują pewien okres w odróżnieniu od typów danych wymienionych w tabeli 2, które składują konkretną datę i czas. Interwał: *Interval year to month* składa się z interwału czasowego wyrażonego w latach i miesiącach. Interwał: *Interval day to second* składa się z interwału czasowego wyrażonego w dniach i sekundach. Możliwe jest składowanie dodatnich i ujemnych interwałów czasowych.

Tworzenie tabeli temporalnej z obsługą czasu rzeczywistego (VT-valid time) jest realizowane przez uwzględnienie czasookresu w definicji tabeli. Podstawowa instrukcja tworzenia tabeli została rozszerzona o klauzulę PERIOD FOR. Wartość NULL ma specjalne znaczenie w modelu temporalnym na platformie Oracle. W odróżnieniu od platformy MS SQL Server, dopuszczalne jest użycie wartości NULL dla kolumn definiujących granice czasookresu. Użycie wartości NULL dla początku przedziału czasowego oznacza „od początku pomiaru

czasu" („od zawsze”), zaś dla końca przedziału czasowego oznacza „na zawsze”-dopuszczalną wartość maksymalną dla danego typu danych skojarzanego z tą kolumną. Kolumny te mogą być zdefiniowane w sposób jawny lub niejawny [11-13].

Poniżej przedstawiono przykład tworzenia tabeli temporalnej wersjonowanej aplikacyjnie, z jawnie zdefiniowanymi kolumnami dla czasookresu:

```
CREATE TABLE pracownik
(
  IdPracownika int CONSTRAINT PK_IdPrac PRIMARY KEY NOT NULL,
  Imie VARCHAR2(15) NOT NULL,
  Nazwisko VARCHAR2(30) NOT NULL,
  Miejscowosc VARCHAR2(50) NOT NULL,
  ApliCzasStart DATE NULL,
  ApliCzasKoniec DATE NULL,
  PERIOD FOR APLI_TIME (ApliCzasStart,ApliCzasKoniec)
);
```

Poniższe zapytanie pozwala zapoznać się ze strukturą utworzonej tabeli temporalnej:

```
SELECT COLUMN_NAME, DATA_TYPE, COLUMN_ID AS COL_ID,
SEGMENT_COLUMN_ID AS SEG_COL_ID, INTERNAL_COLUMN_ID AS INT_COL_ID,
HIDDEN_COLUMN FROM USER_TAB_COLS WHERE
TABLE_NAME='PRACOWNIK';
```

Na rysunku 2 przedstawiono strukturę tabeli temporalnej po jej utworzeniu. Widoczna jest dodatkowa ukryta kolumna dla tworzonego czasookresu APLI_TIME.

❖ COLUMN_NAME	❖ DATA_TYPE	❖ COL_ID	❖ SEG_COL_ID	❖ INT_COL_ID	❖ HIDDEN_COLUMN
1 APLI_TIME	NUMBER	(null)	(null)	1	YES
2 IDPRACOWNIKA	NUMBER	1	1	2	NO
3 IMIE	VARCHAR2	2	2	3	NO
4 NAZWISKO	VARCHAR2	3	3	4	NO
5 MIEJSCOWOSC	VARCHAR2	4	4	5	NO
6 APLICZASSTART	DATE	5	5	6	NO
7 APLICZASKONIEC	DATE	6	6	7	NO

Rys.2. Struktura tabeli temporalnej Pracownik z jawnie zdefiniowanymi kolumnami czasookresu APLI_TIME

Poniżej przedstawiono kod tworzenia tabeli temporalnej wersjonowanej aplikacyjnie z niejawnie zdefiniowanymi kolumnami dla czasookresu APLI_TIME:

```
CREATE TABLE pracownik1
(
  IdPracownika int CONSTRAINT PK_IdPrac1 PRIMARY KEY NOT NULL,
  Imie VARCHAR2(15) NOT NULL,
  Nazwisko VARCHAR2(30) NOT NULL,
  Miejscowosc VARCHAR2(50) NOT NULL,
  PERIOD FOR APLI_TIME
);
```

Na rysunku 3 przedstawiono strukturę tabeli temporalnej z niejawnie zdefiniowanymi kolumnami czasookresu APLI_TIME. Kolumny te przyjmują domyślne nazwy będące połączeniem nazwy czasookresu z sufiksami _START i _END.

❖ COLUMN_NAME	❖ DATA_TYPE	❖ COL_ID	❖ SEG_COL_ID	❖ INT_COL_ID	❖ HIDDEN_COLUMN
1 APLI_TIME_START	TIMESTAMP(6) WITH TIME ZONE	(null)	(null)	1	YES
2 APLI_TIME_END	TIMESTAMP(6) WITH TIME ZONE	(null)	(null)	2	YES
3 APLI_TIME	NUMBER	(null)	(null)	3	YES
4 IDPRACOWNIKA	NUMBER	1	1	3	NO
5 IMIE	VARCHAR2	2	2	4	NO
6 NAZWISKO	VARCHAR2	3	3	5	NO
7 MIEJSCOWOSC	VARCHAR2	4	4	6	NO

Rys.3. Struktura tabeli temporalnej Pracownik z niejawnie zdefiniowanymi kolumnami czasookresu APLI_TIME

Istnieje także możliwość dodania kolejnych wymiarów czasu VT do istniejącej tabeli temporalnej. Poniższy fragment kodu wprowadza do już istniejącej tabeli temporalnej Pracownik (z jawnie zdefiniowanymi kolumnami czasookresu) drugi wymiar czasu VT APLI_2_TIME:

```
ALTER TABLE pracownik ADD PERIOD FOR APLI_2_TIME;
```

Na rysunku 4 przedstawiono strukturę tabeli temporalnej Pracownik po jej rozszerzeniu o dodatkowy wymiar czasu VT (z niejawnie zdefiniowanymi kolumnami czasookresu).

❖ COLUMN_NAME	❖ DATA_TYPE	❖ COL_ID	❖ SEG_COL_ID	❖ INT_COL_ID	❖ HIDDEN_COLUMN
1 APLI_2_TIME	NUMBER	(null)	(null)	10	YES
2 APLI_TIME	NUMBER	(null)	(null)	1	YES
3 IDPRACOWNIKA	NUMBER	1	1	2	NO
4 IMIE	VARCHAR2	2	2	3	NO
5 NAZWISKO	VARCHAR2	3	3	4	NO
6 MIEJSCOWOSC	VARCHAR2	4	4	5	NO
7 APLICZASSTART	DATE	5	5	6	NO
8 APLICZASKONIEC	DATE	6	6	7	NO
9 APLI_2_TIME_START	TIMESTAMP(6) WITH TIME ZONE	(null)	(null)	7	YES
10 APLI_2_TIME_END	TIMESTAMP(6) WITH TIME ZONE	(null)	(null)	8	YES

Rys.4. Struktura tabeli temporalnej Pracownik z niejawnie zdefiniowanymi kolumnami czasookresu APLI_2_TIME

Możliwe także jest dodanie wymiaru czasu VT do istniejącej tabeli nietemporalnej. W ten sposób tabela taka staje się tabelą temporalną. Poniżej przedstawiono kod dodania wymiaru czasu VT do istniejącej tabeli nietemporalnej Pracownik2:

```
ALTER TABLE pracownik2 ADD PERIOD FOR APLI_TIME
(ApliCzasStart, ApliCzasKoniec);
```

W przypadku dodawania wymiaru czasu z jawnie określonymi kolumnami czasookresu, należy zmodyfikować definicję tabeli Pracownik2 przez dodanie do niej dodatkowych pól o odpowiednim typie danych.

Tworzenie tabeli bitemporalnej w Oracle

W środowisku Oracle table bitemporalne tworzone są przez jednoczesne użycie obsługi czasu transakcyjnego oraz czasu rzeczywistego. Dla tabeli wersjonowanej systemowo dołączany jest wymiar czasu rzeczywistego VT lub też tabela wersjonowana aplikacyjnie włączana jest do archiwum FDA. Obsługa wymiaru czasu transakcyjnego na platformie Oracle została opisana w pracy [4].

Manipulacja danymi temporalnymi

W celu dodania rekordów do tabeli temporalnej wersjonowanej aplikacyjnie używana jest instrukcja INSERT w klasycznej postaci (podobnie jak dla tabel temporalnych wersjonowanych systemowo). W trakcie dodawania rekordów do tabeli temporalnej wersjonowanej aplikacyjnie można odwoływać się do pól dla czasookresu zarówno zdefiniowanych w sposób jawny, jak i niejawnie. Pola te mogą być również pominięte na liście pól po klauzuli INSERT VALUES.

Poniżej przedstawiono przykład z kilkoma wariantami dodania rekordów do tabeli temporalnej wersjonowanej aplikacyjnie z obsługą czasu rzeczywistego:

```
INSERT INTO pracownik
VALUES (1, 'Jan', 'Kowalski', 'Łódź', TO_DATE('01-02-2021', 'DD-MM-YYYY'), NULL);
INSERT INTO pracownik
VALUES (2, 'Katarzyna', 'Nowakowska', 'Kraków', TO_DATE('30-03-2018', 'DD-MM-YYYY'), TO_DATE('30-09-2020', 'DD-MM-YYYY'));
INSERT INTO pracownik
VALUES (3, 'Stanisław', 'Grzegorzcyk', 'Wrocław', NULL, TO_DATE('30-11-2021', 'DD-MM-YYYY'));
INSERT INTO pracownik
VALUES (4, 'Alicja', 'Jarzębska', 'Jelenia Góra', NULL, NULL);
INSERT INTO pracownik (idpracownika, imie, nazwisko, miejscowosc)
VALUES (5, 'Julia', 'Widawska', 'Sanok');
```

Na rysunku 5 przedstawiono przykładową zawartość tabeli temporalnej Pracownik, po dodaniu do niej kilku rekordów.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCOWOSC	APLICZASSTART	APLICZASKONIEC
1	Jan	Kowalski	Łódź	21/02/01	(null)
2	Katarzyna	Nowakowska	Kraków	18/03/30	20/09/30
3	Stanisław	Grzegorzczak	Wrocław	(null)	21/11/30
4	Alicja	Jarzębska	Jelenia Góra	(null)	(null)
5	Julia	Widawska	Sanok	(null)	(null)

Rys.5. Przykładowa zawartość tabeli temporalnej Pracownik wersjijonowanej aplikacyjnie po dodaniu kilku rekordów

Istnieje także możliwość odwoływania się do niejawnych kolumn czasookresu podczas dodawania rekordów do tabeli temporalnej. Poniżej przedstawiono przykład dodania rekordu do tabeli temporalnej Pracownik1 z jednoczesnym odwołaniem do niejawnych kolumn czasookresu:

```
INSERT INTO pracownik1(IdPracownika, Imie, Nazwisko,
Miejscowosc, Apli_Time_Start, Apli_Time_End)
VALUES (1,'Kinga','Walewska','Łódź',TO_DATE('30-03-
2018','DD-MM-YYYY'),TO_DATE('30-09-2020','DD-MM-
YYYY'));
```

Na platformie Oracle nie zaimplementowano temporalnej obsługi aktualizacji oraz usuwania rekordów. Tym samym działania te będą także niedostępne dla wymiaru czasu rzeczywistego tabeli bitemporalnej. Natomiast możliwa jest aktualizacja zawartości kolumn składających wartości graniczne zdefiniowanego czasookresu-niezależnie od sposobu ich zdefiniowania (jawne/niejawne). Kolumny te mogą być także używane do tworzenia warunków ograniczających w klauzuli WHERE.

Obsługa zapytań

Odpytywanie tabel temporalnych realizowane jest za pomocą klauzuli SELECT, w sposób podobny dla klasycznych tabel. Na potrzeby zapytań temporalnych instrukcja SELECT rozbudowana została o stosowne predykaty temporalne.

W środowisku Oracle wprowadzone zostały klauzule do przetwarzania danych temporalnych składanych w tabelach temporalnych wersjijonowanych systemowo i aplikacyjnie. Są one zgodne z zapisem standardu SQL:2011 (który wprowadza trzy klauzule). Na platformie Oracle dostępne są dwie konstrukcje umożliwiające temporalne odpytywanie tabel wersjijonowanych aplikacyjnie z obsługą czasu rzeczywistego. Zestawienie tych konstrukcji zawarte zostało w tabeli 3.

Tabela 3. Rozszerzenia składniowe do przetwarzania danych składanych w tabelach temporalnych wersjijonowanych aplikacyjnie

SQL:2011	ORACLE-VT
AS OF	AS OF PERIOD FOR
FROM TO	
BETWEEN AND	VERSIONS BETWEEN PERIOD FOR

Za pomocą zestawionych w tabeli 3 predykatów temporalnych można manipulować danymi temporalnymi, dzięki czemu można np. pobierać rekordy, których wartość była aktualna w określonym przedziale czasowym, czy też prześledzić historię zatrudnienia danego pracownika.

Zapytania na tabelach wersjijonowanych aplikacyjnie z obsługą czasu rzeczywistego

Zapytania temporalne na tabeli wersjijonowanej aplikacyjnie z obsługą czasu rzeczywistego VT realizowane są za pomocą instrukcji SELECT rozszerzonej składniowo o predykaty AS OF PERIOD FOR i VERSIONS BETWEEN PERIOD FOR. Uwzględniają one definicję czasookresu za pomocą klauzuli PERIOD (w odróżnieniu do zapytań

temporalnych dla wymiaru czasu transakcyjnego TT) [11-13].

Zapytania na tabeli bitemporalnej realizowane są za pomocą temporalnych predykatów dla wymiaru czasu transakcyjnego [4] i czasu rzeczywistego.

Rozszerzenie AS OF PERIOD FOR zwraca rekordy, które są ważne (obowiązujące) na dany moment w czasie. Poniżej przedstawiono przykład zwracający pracowników, którzy byli aktualnie zatrudnieni w danym dniu:

```
SELECT * FROM pracownik AS OF PERIOD FOR APLI_time
'20/10/01';
```

Na rysunku 6 przedstawiono wynik powyższego zapytania w odniesieniu do pełnego zestawu rekordów tabeli temporalnej.

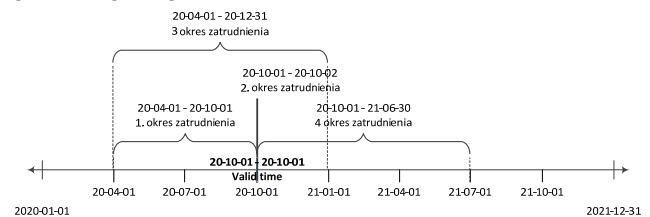
Rys.6. Zapytanie na tabeli temporalnej wersjijonowanej aplikacyjnie z użyciem predykatu AS OF PERIOD FOR, dla wybranej chwili czasowej

Jak można zauważyć, zapytanie temporalne zwróciło zestaw rekordów pomniejszony o 2 rekordy w odniesieniu do całkowitej zawartości tabeli temporalnej. Wynika to z faktu, że jeden z pracowników jeszcze nie był zatrudniony w tym okresie czasu, a drugi zakończył pracę przed datą podaną, jako argument zapytania.

Zapytanie w tej postaci będzie uwzględniało następujący zbiór rekordów:

- pracownicy zatrudnieni w podanym dniu, i w tym samym dniu kończący zatrudnienie,
- pracownicy zatrudnieni w podanym dniu i pracujący nadal po tym dniu,
- pracownicy zatrudnieni przed podanym dniem, którzy w podanym dniu zakończyli zatrudnienie,
- pracownicy zatrudnieni przed podanym dniem, którzy nadal pracowali po tym dniu.

Na rysunku 7 przedstawiono interpretację graficzną działania zapytania temporalnego z użyciem predykatu AS OF PERIOD FOR.

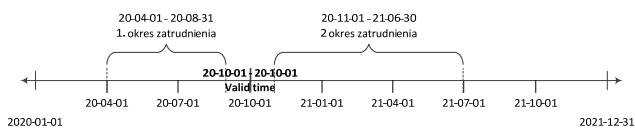


Rys.7. Interpretacja działania zapytania na tabeli temporalnej wersjijonowanej aplikacyjnie z użyciem predykatu AS OF PERIOD FOR, dla wybranej chwili czasowej

Natomiast w przypadku użycia tego predykatu nieuwzględnione zostaną następujące rekordy:

- pracownicy zatrudnieni po podanej dacie,
- pracownicy, którzy przestali pracować przed podaną datą.

Interpretacja graficzna tego przypadku została przedstawiona na rysunku 8.



Rys.8. Interpretacja działania zapytania na tabeli temporalnej wersjonowanej aplikacyjnie z użyciem predykatu AS OF PERIOD FOR, dla wybranej chwili czasowej, w odniesieniu do zbioru rekordów niespełniających postawione kryterium

Predykat VERSIONS BETWEEN PERIOD FOR zwraca zestaw rekordów, które były obowiązujące w zadanym przedziale czasu. Poniżej przedstawiono przykład zwracający pracowników, którzy byli zatrudnieni w danym przedziale czasu (jednocześnie nie jest to równoważne stwierdzeniu, że pracownicy ci byli zatrudnieni przez cały ten okres):

```
SELECT * FROM pracownik VERSIONS BETWEEN PERIOD FOR APLI_TIME BETWEEN ('20/09/30') AND ('21/09/30');
```

Na rysunku 9 przedstawiono wynik powyższego zapytania w odniesieniu do pełnego zestawu rekordów tabeli temporalnej.

IDPRACOWNIKA	IMIE	NAZWISKO	MIEJSCOWOSC	APLICZASZCZAS	APLICZASKONIEC
1	Jan	Kowalski	Lódź	21/02/01	(null)
2	3	Stanisław Grzegorzczak	Wrocław	(null)	21/11/30
3	4	Alicja Jarzębska	Jelenia Góra	(null)	(null)
4	5	Julia Widawska	Sanok	(null)	(null)
5	6	Julian Wysocki	Sandomierz	(null)	(null)

Rys.9. Zapytanie na tabeli temporalnej wersjonowanej aplikacyjnie z użyciem predykatu VERSIONS BETWEEN PERIOD FOR, dla wybranej chwili czasowej.

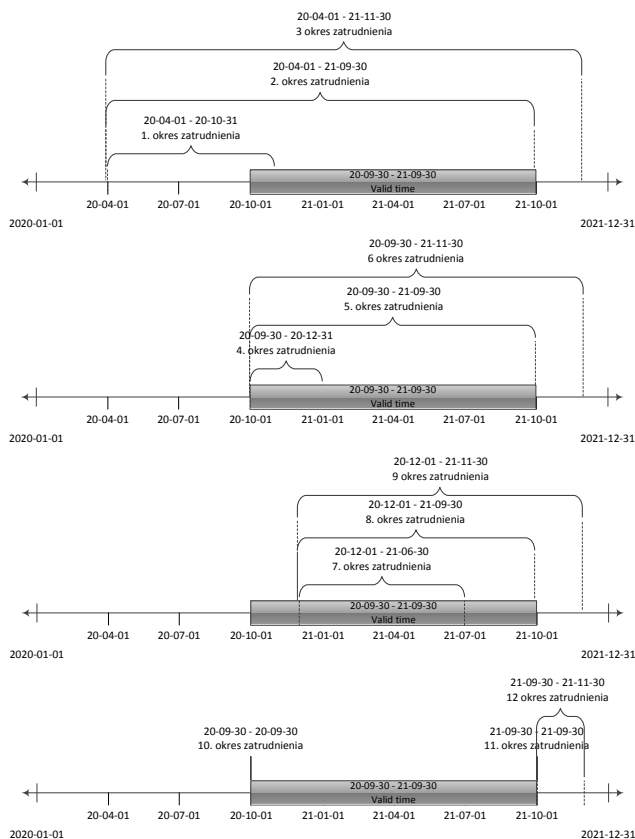
Zapytanie temporalne zwróciło zestaw rekordów pomniejszony o 1 rekord w odniesieniu do całkowitej zawartości tabeli temporalnej. Wynika to z faktu, że dla jednego z pracowników zatrudnienie ustąpiło przed początkiem zdefiniowanego czasookresu.

Zapytanie z użyciem tego predykatu temporalnego będzie uwzględniało następujący zbiór rekordów:

- pracownicy zatrudnieni przed rozpoczęciem podanego okresu, których staż zakończył się przed końcem podanego okresu lub dokładnie w ostatni dzień podanego okresu, bądź też pracowali dłużej niż górna granica podanego czasookresu,
- pracownicy, którzy rozpoczęli pracę dokładnie w dniu, określonym, jako początek zdefiniowanego czasookresu, lub w czasie późniejszym po nim, ale nie później niż górna granica czasookresu. Mogli pracować w całym zdefiniowanym czasookresie, zakończyć pracę przed jego górną granicą lub pracować dłużej niż

- wymieniona górna granica, bądź też zakończyć pracę w tym samym dniu, w którym została ona podjęta,
- pracownicy, którzy rozpoczęli pracę w ostatnim dniu czasookresu i w tym samym dniu przestali pracować lub pracowali dalej,
- pracownicy, którzy podjęli zatrudnienie dokładnie w pierwszym dniu czasookresu a zakończyli ją dokładnie w ostatnim dniu czasookresu.

Na rysunku 10 przedstawiono interpretację graficzną działania zapytania temporalnego z użyciem predykatu VERSIONS BETWEEN PERIOD FOR.

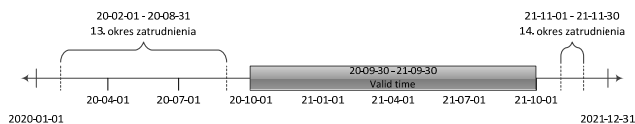


Rys.10. Interpretacja działania zapytania na tabeli temporalnej wersjonowanej aplikacyjnie z użyciem predykatu VERSIONS BETWEEN PERIOD FOR, dla wybranej chwili czasowej.

W przypadku działania tego predykatu nieuwzględnione zostaną rekordy:

- pracownicy, którzy rozpoczęli zatrudnienie po czasie zdefiniowanego czasookresu,
- pracownicy, którzy zakończyli zatrudnienie przed rozpoczęciem zdefiniowanego czasookresu.

Interpretacja graficzna tego przypadku została przedstawiona na rysunku 11.



Rys.11. Interpretacja działania zapytania na tabeli temporalnej wersjonowanej aplikacyjnie z użyciem predykatu VERSIONS BETWEEN PERIOD FOR, dla wybranej chwili czasowej, w odniesieniu do zbioru rekordów niespełniających postawione kryterium.

Podsumowanie

Na platformie Oracle nie zostały zaimplementowane wszystkie postulaty standardu SQL:2011 w zakresie temporalności danych. Brakuje temporalnych ograniczeń

dla klucza głównego, klucza obcego, ograniczeń referencyjnych, specjalnych rozszerzeń temporalnych dla akcji modyfikowania i usuwania rekordów z tabeli temporalnej wersjonowanej aplikacyjnie. Nie został także zaimplementowany jeden z predykatów temporalnych FROM TO zawarty w specyfikacji standardu SQL:2011. Brak także złączeń temporalnych czy też agregacji temporalnych (dla działań UNION, MINUS, INTERSECT). Jednakże Oracle obsługuje oba wymiary czasu, zarówno czas transakcyjny (transaction time) i czas rzeczywisty (valid time). Dzięki temu możliwa jest także obsługa tabel bitemporalnych. Przy czym należy zauważyć, że nie została zaimplementowana składnia tworzenia tabeli bitemporalnej zgodnie z wytycznymi zawartymi w standardzie SQL:2011. Każdy z wymiarów czasu obsługiwany jest za pomocą dwóch odrębnych technologii. Obsługa czasu transakcyjnego realizowana jest za pomocą technologii Oracle Flashback a w szczególności modułu Oracle Flashback Data Archive. Czas rzeczywisty obsługiwany jest za pomocą modułu Temporal Validity. Wprowadzenie obsługi obu wymiarów czasu na platformie Oracle znacznie uprościło proces tworzenia aplikacji, w których kluczowa jest obsługa wybranego lub obu wymiarów czasu.

W kolejnych edycjach serwera Oracle (18c i 19c) wprowadzonych zostało wiele ulepszeń i nowych funkcjonalności, lecz żadne z nich nie dotyczyło wprowadzenia brakujących elementów dotyczących obsługi danych temporalnych zgodnych z wytycznymi zawartymi w standardzie SQL:2011. Pewne ograniczenia dotyczą użycia klauzuli VERSIONS. Nie można jej użyć w zapytaniach flashback na tabelach tymczasowych, tabelach zewnętrznych, tabelach będących częścią klastra oraz na widokach. Dopuszczalne jest jednak jej użycie wewnątrz zapytania definiującego widok.

Pomimo niepełnej implementacji standardu SQL:2011 w zakresie obsługi danych temporalnych, zakres tej implementacji w środowisku Oracle wypada korzystniej niż w przypadku serwera MS SQL Server, gdzie zaimplementowana została tylko obsługa czasu transakcyjnego [3,14]. Jednakże na rynku dostępne są rozwiązania ze znacznie szerszą obsługą danych temporalnych. Najbardziej zaawansowaną obsługą danych temporalnych zgodnie ze standardem SQL:2011 zaimplementowano w rozwiązaniach IBM DB2 [15,16,19-21]. Zaawansowana obsługa danych temporalnych jest także zaimplementowana w rozwiązaniach Teradata, aczkolwiek opiera się ona innych fundamentach niż standard SQL:2011 [17-20]. Zarówno IBM DB2 jak i Teradata obsługują m.in. temporalną aktualizację i usuwanie rekordów w przeciwieństwie do Oracle.

Kolejnym etapem pracy będzie zbadanie stopnia realizacji założeń temporalnych rozszerzeń standardu SQL:2011, w wybranych wiodących produktach na rynku (nie tylko komercyjnych) oraz prezentacja zaimplementowanych w nich funkcjonalności temporalnych.

Autorzy: dr inż. Sebastian Łacheciński, Uniwersytet Łódzki, Instytut Logistyki i Informatyki, Katedra Informatyki Ekonomicznej, 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] Date C.J., Darwen H., Lorentzos N., *Time and relational theory Temporal Databases in the Relational Model and SQL*, 2014, Morgan Kaufmann
- [6] Kulkarni K., Jan-Eike Michels, 2012, *Temporal features in SQL:2011*, 34-43 <https://sigmodrecord.org/publications/sigmodRecord/1209/pdfs/07.industry.kulkarni.pdf>
- [7] Temporal extension SQL: https://link.springer.com/content/pdf/10.1007%2F978-1-4899-7993-3_80729-1.pdf
- [8] McLaughlin M., *Oracle 12c Programowanie w języku PL/SQL*, 2015, Helion
- [9] Temporal Validity in Oracle Database 12c Release 1 (12.1): <https://oracle-base.com/articles/12c/temporal-validity-12cr1>
- [10] Datetime Datatypes and Time Zone Support: https://docs.oracle.com/cd/B19306_01/server.102/b14225/ch4datetime.htm
- [11] Salvisberg P., Multi-temporal Features in Oracle 12c, <https://www.salvis.com/blog/2014/01/04/multi-temporal-database-features-in-oracle-12c/>
- [12] Implementing Temporal Validity: <https://www.oracle.com/webfolder/technetwork/tutorials/obe/db/12c/r1/ilm/temporal/temporal.html>
- [13] Gohil J., Dolia P., Checking and Verifying Temporal Data Validity Using Valid Time Temporal Dimension and Queries in Oracle 12C, *International Journal of Database Management Systems (IJDBMS)* Vol.7, No.4, August 2015, 11-22
- [14] MS SQL Server: <https://docs.microsoft.com/en-us/sql/relational-databases/tables/getting-started-with-system-versioned-temporal-tables?view=sql-server-ver15>
- [15] Saracco M., Nicola M., Gandhi L., 2012, A matter of time: Temporal data management in DB2 10 <https://www.yumpu.com/en/document/read/13882386/a-matter-of-time-temporal-data-management-in-db2-10-ibm>
- [16] IBM: <https://www.ibm.com/developerworks/data/library/techarticle/dm-1410temporal-tables-db2zos/index.html>
- [17] Snodgrass R.T., 2010, A Case Study of Temporal Data, Teradata Corporation https://cs.ulb.ac.be/public/_media/teaching/infh415/teradata_temporal_case_study.pdf
- [18] Teradata https://docs.teradata.com/reader/vi_E5DAzLxKE_Xhr8yNIWw/BuO5td4SJlxUXyFilG6Geg
- [19] Petković D., Support of Temporal Data in Database Systems, *International Journal of Computer Applications (0975 –8887)*, Volume 152 –No.4, October 2016, 26-33
- [20] Petković D., Temporal Data in Relational Database Systems: A Comparison, Conference: WorldCIST (1) 2016, At Recife, Volume: 1
- [21] Petković, D., Temporal Data in Enterprise Database Systems, The 7th International Conference on Information Technology, 2015, pp 276-282