"زمن": مقاربة لنظام إدارة قواعد معطيات زمنية

د. محي الدين مراد* **م. معاذ عثمان****

م. أحمد عمايري***

الملخص

ظهر خلال نصف القرن المنصرم العديد من تطبيقات قواعد المعطيات التي يؤدي الوقت فيها دوراً مهمّاً، فكشفت عن نقص في دعم الدلالات الزمنية ضمن نظم إدارة قواعد المعطيات الحالية، إذْ يقع على عاتق التطبيقات الزمنية إعطاء البيانات دلالات زمنية مرتبطة بها، والتحقق من القيود الزمنية لضمان الاتساق، لذلك اتجهت البحوث إلى تضمين هذه الدلالات والقيود الزمنية في نظام إدارة قواعد المعطيات نفسه، وتوفير لغة استعلام جديدة مناسبة توصف بأنها زمنية.

يقدم هذا البحث مقاربة تعتمد على تغليف نظام إدارة قواعد معطيات علائقي بطبقة برمجية تدعم الدلالات والقيود الزمنية، وتؤمن واجهة للتخاطب مع التطبيقات الزمنية بلغة استعلام زمنية شبيهة ومتوافقة زمنياً مع SQL-92 أطلقنا عليها اسم zSQL، وصولاً إلى نظام إدارة قواعد معطيات زمنية أطلقنا عليه اسم "زمن".

الكلمات المفتاحية: قواعد المعطيات- القيود الزمنية

* قسم هندسة البرمجيات ونظم المعلومات - كلية الهندسة المعلوماتية – جامعة دمشق
** قسم هندسة البرمجيات ونظم المعلومات – كلية الهندسة المعلوماتية – جامعة دمشق
*** قسم هندسة البرمجيات ونظم المعلومات – كلية الهندسة المعلوماتية – جامعة دمشق

مقدمة

إن معظم تطبيقات قواعد المعطيات زمنية بطبيعتها، مثل التطبيقات المالية كبرامج المحاسبة والمصارف، وتطبيقات حفظ السجلات كالتطبيقات الطبية(1)، وتطبيقات الجدولة كتطبيقات حجز التذاكر في المطارات والقطارات والحجز الفندقي وإدارة المشاريع، والتطبيقات العلمية مثل مراقبة الطقس(2).

يضطر مطور وهذه التطبيقات إلى الاعتماد على نظم إدارة قواعد معطيات لازمنية بطبيعتها ومعالجة مدلولات التطبيق الزمنية برمجياً ضمن كل تطبيق على حدة، مما يؤدي إلى تكرار الجهد والابتعاد عن المعايير والمقاييس، لذا لجأ بعضهم إلى عزل المكونات البرمجية هذه ضمن مكتبات برمجية تُضمّن في التطبيقات الزمنية، إلا أن ذلك لا يمنع التطبيقات غير الزمنية من الولوج إلى قاعدة المعطيات وإحداث تغييرات في البيانات تخل بهذه الدلالات الزمنية.

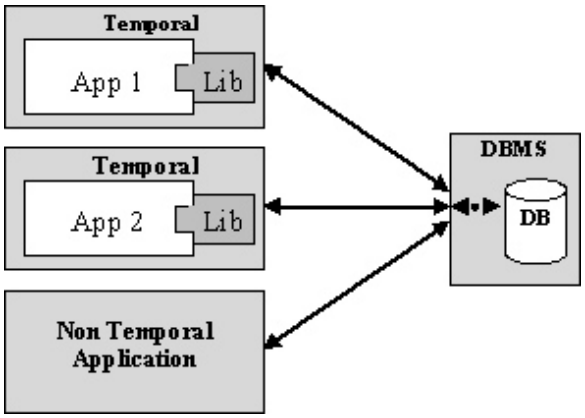
لحل المشكلة المذكورة وتخفيف العبء عن التطبيقات جاءت فكرة نقل وظائف المكتبة البرمجية المذكورة إلى نظام إدارة قواعد المعطيات، ينتج عن ذلك نوع جديد من أنظمة إدارة قواعد المعطيات يدعى بأنظمة إدارة قواعد المعطيات الزمنية.

يعرف هذا النوع الجديد من أنظمة إدارة قواعد المعطيات دلالاتٍ جديدةً لدعم الزمن، لذلك فهو بحاجة إلى تعريف لغة استعلامٍ جديدة تدعم هذه الدلالات، يطلق عليها اسم لغة استعلام زمنية.

سنعرض فيما يأتي شرحاً موجزاً عن الأبعاد الزمنية وأنواع الجداول المرتبطة بها ونماذج المعطيات العلائقية المستعملة لتمثيلها ثم نقدم لمحة عن العمليات والتوابع ذات الدلالات الزمنية ونلقي نظرة على لغات الاستعلام الزمنية المقدمة في البحوث السابقة، أخيراً نعرض مقاربة لنظام إدارة قواعد معطيات زمنية أطلقنا عليه اسم "زمن" وللغة الاستعلام الزمنية الخاصة به والتي أطلقنا عليها اسمzSQL.

1- الأبعاد الزمنية

البعد الزمني هو واصفة زمنية للمعلومة تحمل دلالة مميزة، تركز معظم البحوث على بعد زمني وحيد اصطلح عليه اسم زمن الصلاحية Valid Time، في حين تضيف البحوث الأكثر توسعاً بعداً آخرَ هو زمن المناقلةTransaction Time.



1-1- زمن الصلاحية

زمن الصلاحية لمعلومة ما هو الزمن الذي تكون فيه هذه المعلومة حقيقية في العالم المنمذج(3)، مثل صلاحية عقد إيجار أو تاريخ الانتخابات. إن الحقائق جميعها تملك زمن صلاحية حسب التعريف السابق، ولكن هذا الزمن قد لا يسجل ضمن قاعدة المعطيات لعدة أسباب، منها أن يكون زمن الصلاحية غير معلوم، أو كبيراً بما يكفي لعدِّ المعلومة صالحة مطلقاً، يحدد المستخدم قيمة زمن الصلاحية في أثناء إدخال المعلومة إلى قاعدة المعطيات.

1-2 زمن المناقلة

تخزن المعلومة في قاعدة المعطيات في لحظة ما وتبقى فعالة حتى يتم حذفها منطقياً (حذفها أو تعديلها)، يعبّر زمن المناقلة لمعلومة ما عن الزمن الذي تكون فيه "حالية" في قاعدة المعطيات والذي يمكن جلبها خلاله(3)، (أي إنَّ قاعدة المعطيات تعترف بالمعلومة على أنها

حالية)، لذا فإن زمن المناقلة هو مدة زمنية وليس لحظة، وتتلاءم أزمنة المناقلة في قاعدة المعطيات مع ترتيب المناقلات، لذا لا يمكن أن يبدأ زمن مناقلة ما في المستقبل، كما لا يمكن تعديل بداية زمن مناقلة سابق، ويُحسَبُ زمن المناقلة آلياً بمعزل عن المستخدم.

لا يرتبط زمن المناقلة وزمن الصلاحية بأية علاقة وهما مستقلان Orthogonalعن بعضهما تماماً.

ولتمثيل الزمن ضمن قاعدة المعطيات يمكن استعمال نموذج الزمن المستمر Continuous Time Model الذي ينظر إلى الزمن على أنه دفق مستمر من اللحظات التي يسعى الفارق بينها إلى الصفر، إلا أن أجهزة الحاسوب لا تتعامل مع القيم التماثلية وتقوم دوماً بالتكميم Quantization، كما أن أغلب التطبيقات الزمنية لا تحتاج إلى الدقة التي يوفرها النموذج المستمر، لذا ظهر نموذج الزمن المتقطع Discrete Time Model الذي ينظر إلى الزمن على أنه تتالٍ لعدد لا نهائي من المقاطع الزمنية Chronons، إذْ إنَّ مقطع الزمن هو مدة Period محددة (لحظة بداية ولحظة نهاية)، ولا يمكننا تعقب التغيّرات في القيمة إذا حصل التغيير ضمن المقطع الزمني للتغيير السابق نفسه، ولن نتمكن من معرفة القيمة قبل التغيير.

يمكن أن يكون طول المقطع Interval ثابتاً، فيدعى بالحبيبة الزمنية Granularity مثل: شهر أو أربعة أيام أو سنتين، كما يمكن أن يكون طول المقطع غير ثابتٍ، حيث تُعرف المقاطع ضمن نظام التقويم Calendar(4).

2- أنواع الجداول في قاعدة المعطيات الزمنية

يختلف نوع الجدول بحسب عدد الأبعاد الزمنية التي يحويها ونوعها(5)، فيمكن للجدول أن يحوي بعد زمن الصلاحية فقط، أو بعد زمن المناقلة فقط، أو كلا البعدين، ويمكن ألا يحوي أياً منهما، كما أن طبيعة زمن الصلاحية تختلف، فقد يكون مدة محصورة بين لحظتين معينتين مثل زمن صلاحية عقد الإيجار، ويمكن أن يكون لحظة وحيدة مثل يوم الانتخابات. ومن ثمَّ فإن هناك ستة أنواع مختلفة من الجداول قي قواعد المعطيات الزمنية وهي:

2-1 جدول صالح-حدثValid-Event Table

وهو الجدول الذي يخزن البيانات المرتبطة بلحظة زمنية وحيدة، أي إنَّها لاتملك لحظة بداية ونهاية، بل تملك لحظة حدوث، ومع أن معظم الحقائق تتميز بلحظة بداية ونهاية إلا أننا كثيراً مانشير إليها بتوقيت واحد على اعتبارها أحداث، فقد يشار إلى حرب على أنها حدثت في سنة كذا بدلاً من تحديد تاريخ بدايتها ونهايتها ،ومثل ذلك أيام المناسبات التي لا تتصف بلحظة بداية ونهاية.

2-2 جدول صالح-حالة Valid-State Table

وهو الجدول الذي يخزن البيانات المرتبطة بمدة زمنية (لحظة بداية ولحظة نهاية)، مثل مهمة موظف، أو عقد إيجار.

2-3 جدول مناقلة Transaction Table

وهو الجدول الذي يخزن البيانات التي من المهم متابعة أزمنة التغييرات التي تحصل عليها، ويمكن أن يجيب هذا الجدول علن أسئلة من الشكل: "ماذا كانت قيمة الواصفة في قاعدة المعطيات في اللحظة X؟".

2-4 جداول ثنائية الزمن Bitemporal Table

وهي الجداول التي من المهم متابعة التغييرات الحاصلة عليها، أي إنَّها جداول تحوي زمن مناقلة، فضلاً عن أنها تخزن زمن الصلاحية الخاص بكل معلومة ضمنها، أي إنَّها جداول زمن صلاحية، ومن ثمَّ لها نوعان:

2-4-1 جدول ثنائي الزمن -حدث Bitemporal-Event Table

وهو الجدول الذي يخزن البيانات التي تحمل زمن صلاحية على شكل لحظات زمنية بالإضافة إلى زمن المناقلة.

2-4-2 جدول ثنائي الزمن -حالة Bitemporal - State

وهو الجدول الذي يخزن البيانات التي تحمل زمن صلاحية على شكل مدد زمنية بالإضافة إلى زمن المناقلة.

2-5 جدول لازمنيSnapshot Table

وهو الجدول الذي يخزن البيانات التي لا تحمل زمن مناقلة ولا زمن صلاحية مثل الجداول في قواعد المعطيات الحالية.

3- النماذج المستعملة لتمثيل قواعد المعطيات الزمنية

عُرِّفَتْ عدة نماذج لتمثيل الدلالات الزمنية ورُبِطَتْ بالمعطيات، وقد تجاوزت العشرين نموذجاً(6)، يكتفي أغلبها بدعم زمن الصلاحية فقط، في حين يقوم بعضها بدعم زمن المناقلة أيضاً، وتعدُّ هذه النماذج ناضجة بما فيه الكفاية (7)، منها ما يعتمد النموذج العلائقي Relational Model ومنها ما يعتمد النموذج غرضي التوجه Object-Oriented Model، سنقتصر بشرحنا هنا على النماذج العلائقية لبساطتها وقربها من واقع التطبيق، وأهم هذه النماذج:

3-1 النموذج المفهومي للمعطيات ثنائية الزمنBCDM

استعمل النموذج المفهومي للمعطيات ثنائية الزمن Bitemporal Conceptual Data Model لتوحيد النماذج العلائقية في نموذج واحد مفهومي يدعم كلاً من زمن المناقلة وزمن الصلاحية دون الخوض في تفاصيل طريقة التخزين المنطقية أو الفيزيائية(8)، ينظر هذا النموذج –مثل نموذج Codd(9)- إلى المعلومة على أنها تتكون من مجموعة أبعاد، إلا أنه يميز من هذه الأبعاد بعدي زمن المناقلة وزمن الصلاحية لدلالتيهما المميزتين، ويسمي باقي الأبعاد بالواصفات الصريحةExplicit، إذْ يعدُّ المعطيات التي تشترك بقيم الواصفات الصريحة نفسها مجتمعةً تعبّر عن معلومة وحيدة.

يمكن تقريب هذا المفهوم بتمثيل المعطيات ضمن فضاء ثلاثي البعد، أبعاده:

1. زمن المناقلة ($\overrightarrow{TT}$) Transaction-Time؛
2. زمن الصلاحية ($\overrightarrow{VT}$) Valid-Time؛
3. قيم الواصفات الصريحة مجتمعة.

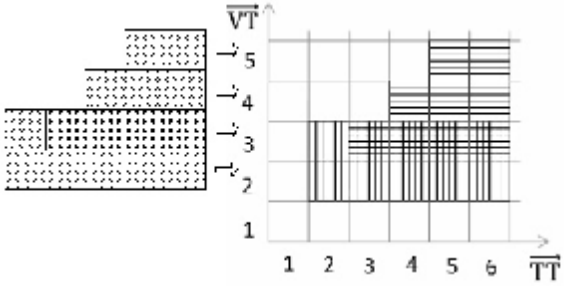
. Explicit-Values.($\overrightarrow{EV}$)

توسم كل معلومة بمجموعة مجالات كل منها مغلق أو نصف مفتوح من اليمين على المحور $\overrightarrow{TT}$، يعبّر كل منها عن لحظة بداية المناقلة ولحظة نهايتها إن وجدت (تنتهي المناقلة عند الحذف أو التعديل)، كما توسم بمجموعة مجالات كل منها مغلق أو نصف مفتوح من اليمين على المحور $\overrightarrow{VT}$، يعبّر كل منها عن المدة التي كانت فيها هذه المعلومة صالحة في العالم المنمذج، يشكل المجالان معاً مستطيلاً مغلقاً أو مفتوحاً من اليمين أو من الأعلى أو من اليمين والأعلى معاً ضمن المستوي $(\overrightarrow{TT}.\overrightarrow{VT})$، نصطلح تسمية هذا المستوي (أو المستويات الموازية له) بالفضاء الزمني *Temporal Space*.

من الواضح أن التسجيلات التي تحمل قيمة الواصفات الصريحة نفسها مجتمعة تحمل نفس الإحداثي على المحور $\overrightarrow{EV}$، لذا فقد تحدد على مستوي الفضاء الزمني أكثر من مستطيل، يعرف النموذج المفهومي نمطاً جديداً من المعطيات يدعى "العنصر الزمني Temporal Element" يعبّر عن اجتماع هذه المستطيلات، ويضيف إلى كل جدول عموداً واحداً يحوي العنصر الزمني المرتبط بالتسجيلة الذي يحوي زمن الصلاحية وزمن المناقلة في آنٍ معاً.

يظهر الشكل التالي جدولاً زمنياً يحوي معطيات تأجير غرف ضمن فندق باستعمال النموذج المفهومي ثنائي، ويعرض الشكل الذي يليه تمثيلاً بيانياً لهذه المعطيات.

Explicit Attrib. Values		Bitemporal Timestamp
Room	Customer	T
7	12	(2,2), (3,2), (4,2), (5,2),…,(UC,2) (2,3), (3,3), (4,3), (5,3),…, (UC,3)
15	34	(3,3), (4,3), (5,3),,(UC,3) (4,4), (5,4),…,(UC,4) (5,5),…,(UC,5)



3-2 الختم الزمني على مستوى التسجيلة Tuple Timestamping

نضيف في هذا النموذج إلى كلّ تسجيلة أربع واصفات وهي: زمن بداية المناقلة وزمن نهاية المناقلة، وزمن بداية الصلاحية، وزمن انتهاء الصلاحية، وفي حال تغيّرت قيمة واصفة صريحة واحدة أو أكثر ضمن تسجيلة معينة أو تغيّر زمن صلاحيتها نقوم بتعديل زمن انتهاء المناقلة في التسجيلة المعنية؛ ثم نضيف تسجيلة جديدة تحوي المعلومات التي عُدِّلَتْ في التسجيلة السابقة (فضلاً عن المعلومات التي بقيت كما هي)، ويكون في هذه التسجيلة زمن بداية المناقلة هو زمن نهاية المناقلة في التسجيلة السابقة، ويجدر القول: إنَّ هذا النموذج يستخدم في النماذج التي تتبع للصيغة القياسية الأولى 1NF.

من إيجابيات هذا النموذج أنه أسهل في التحقيق من غيره من النماذج.

من سلبياته أنه يسبب تكراراً في المعطيات، حيث إنَّه إذا تغيّرت قيمة واصفة واحدة ضمن تسجيلة تحوي على عدد كبير من الواصفات فسيتم إنشاء تسجيلة جديدة تحوي قيم التسجيلة السابقة نفسها مع تعديل قيمة الواصفة التي تغيّرت فقط.

يمثل الجدول الآتي تطبيقاً لهذا النموذج على جدول يحوي المعطيات نفسها في المثال السابق (حجوزات غرف في فندق).

Explicit Attrib. Values		Timestamp			
Room	Customer	V-Start	V-End	T-Start	T-End
7	12	2	4	2	∞
15	34	3	4	3	4
15	34	3	5	4	5
15	34	3	6	5	∞

3-3 الختم الزمني على مستوى الواصفة Attribute Timestamping

نضيف في هذا النموذج بداية الصلاحية ونهايتها وبداية المناقلة ونهايتها إلى كل واصفة من واصفات الجدول ضمن قاعدة المعطيات (6)، يتغلب هذا النموذج على تكرار المعطيات في نموذج الختم الزمني على مستوى التسجيلة، ولا يتبع الصيغة القانونية1NF (يوصف أنهNFNF)، لذ يجب لتحقيقه استعمال الأغراض المركبة Complex Objects (في قاعدة معطيات من النوع Object-Relational).

4-3 الختم الزمني على مستوى الحدث Event Timestamping

تمثل العلاقات ثنائية الزمن في هذا النموذج عن طريق سجلات خلفية تتبع للصيغة القانونية الأولى 1NF.

الفرق الأكثر أهمية بين هذا النموذج والنماذج السابقة أن التسجيلات في السجلات الخلفية لا يمكن تحريرها أبداً ولكن تتم الإضافة إليها، فهو الأنسب لاستعمال أقراص الكتابة مرة واحدة رخيصة السعر، وهو أمر مهم لما كانت المعطيات الزمنية دائمة الزيادة في الحجم مما ينتج

عنه علاقات كبيرة جداً.

تمثل العلاقة ثنائية الزمن $R = (A_1, A_2, \ldots, A_n, T)$ بسجل خلفي R كما يأتي(6):

$$R = (A_1, A_2, \ldots, A_n, V_s, V_e, T, OP)$$

في هذا النموذج تخزن الواصفتان V_s, V_e بداية زمن الصلاحية ونهايتها على الترتيب، في حين تخزن الواصفة T زمن المناقلة وهو اللحظة التي تم فيها إدراج التسجيلة في السجل الخلفي.

تُصنف التسجيلات - التي يصطلح على تسميتها طلبات التغيير - إلى طلبات إدراج وطلبات حذف، ويُحَدَّد ذلك بإعطاء الواصفة OP القيمة I في حالة الإدراج أو القيمة D في حالة الحذف.

تُعدُّ المعلومة المدرجة "حالية" في قاعدة المعطيات بدءاً من ختم زمن المناقلة الخاص بها وحتى ورود طلب حذف بقيم الواصفات الصريحة نفسها وزمن الصلاحية.

تخزن التعديلات على شكل زوج من الطلبات: طلب حذف وطلب إدراج يحمل كلاهما زمن المناقلة T نفسه يوضح الجدول الآتي مثالاً على هذا الختم:

Explicit Attrib. Values		Timestamp			
Room	Customer	V-Start	V-End	T	OP
7	12	2	4	2	*I*
15	34	3	4	3	*I*
15	34	3	4	4	*D*
15	34	3	5	4	*I*
15	34	3	5	5	*D*
15	34	3	6	5	*I*

5-3 مقارنة بين النماذج

وفيما يأتي جدول يعرض مقارنة لبعض النماذج مع المراجع

اسم نموذج المعطيات	المرجع	زمن الصلاحية	زمن المناقلة	تمثيل زمن الصلاحية	تمثيل زمن المناقلة
-	(10)	✓	✓	مدة – ختم زمني على مستوى التسجيلة	مقطع زمني – ختم زمني على مستوى مجموعة من التسجيلات
Temporally Oriented Data Model	(11)	✓	✖	حدث – ختم زمني على مستوى التسجيلة	-
Time Relational Model	(12)	✓	✓	مدة – ختم زمني على مستوى التسجيلة	ثلاثة مقاطع زمنية – ختم زمني على مستوى التسجيلة
TQuel	(13)	✓	✓	مدة – ختم زمني على مستوى التسجيلة	مدة (زوج من المقاطع الزمنية) – ختم زمني على مستوى التسجيلة
Postgres	(14)	✓	✖	-	مدة (زوج من المقاطع الزمنية) – ختم زمني على مستوى التسجيلة
HQuel	(15)	✓	✖	مدة – ختم زمني على مستوى الواصفة	-
Accounting Data Model	(16)	✓	✓	مدة – ختم زمني على مستوى الواصفة	مقطع زمني – ختم زمني على مستوى مجموعة من التسجيلات
Time Oriented Data Base Model	(17)	✓	✖	مدة – ختم زمني على مستوى التسجيلة	-

4- العمليات الزمنية

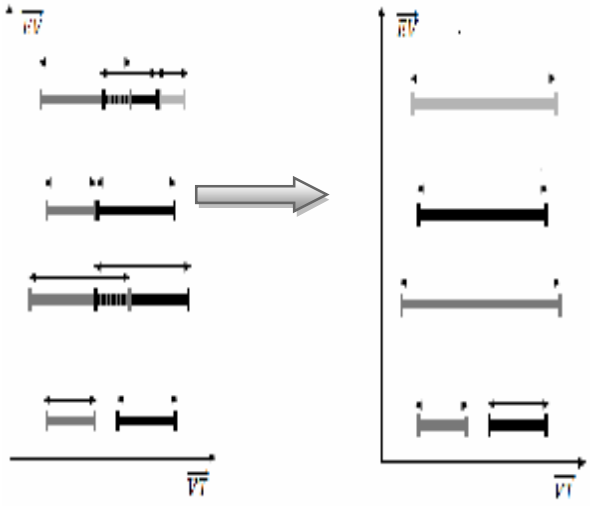
1-4 التضامّ Coalescing

هوعملية أحادية تأخذ كدخل علاقة زمنية، وتعيد علاقة زمنية أخرى لها البنية نفسها في الخرج، الغرض منها هو تطبيق أحد أنواع الاستنظام Normalization على العلاقة الزمنية بالنسبة إلى بعد زمني أو أكثر(18).

تُجرى هذه العملية بتجميع أكبر قدر ممكن من التسجيلات متكافئة القيمة في تسجيلة واحدة مكافئة لهم بالقيمة تحمل ختماً زمنياً أكبر، ولعملية التضام ثلاثة أنواع:

1-1-4 تضام زمن الصلاحية

نرمز للواصفات الصريحة Explicit Values بـ $\overrightarrow{EV}$

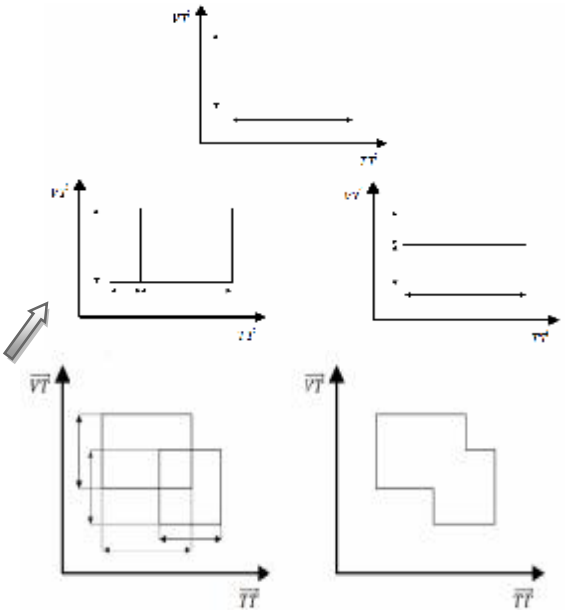


تضام زمن المناقلة

يتبع قواعد تضام زمن الصلاحية نفسها ولكن على بعد زمن المناقلة.

2-1-4 التضام ثنائي الزمن

يوضحه الشكلان الآتيان:



4- التشريح الزمني Temporal Slicing

عملية التشريح الزمني تأخذ معاملين:

الأول: علاقة زمنية؛

الثاني: لحظة أو مدة زمنية.

وتعيد علاقة زمنية أو لازمنية، تحوي العلاقة الخرج جزءاً من المعطيات الموجودة في العلاقة الدخل وتنقص عنها في الأبعاد الزمنية بعداً أو أكثر، ولعملية التشريح ثلاثة أنواع:

3-1-4 تشريح زمن الصلاحيةValid-Time Slicing

تأخذ هذا العملية علاقة زمن صلاحية أو علاقة ثنائية الزمن وتعيد علاقة لازمنية أو علاقة زمن مناقلة على الترتيب، كما تأخذ لحظة أو مدة زمنية وحيدة.

تحوي العلاقة الخرج على المعطيات التي يحوي أو يتقاطع زمن صلاحيتها مع اللحظة أو المدة المحددة، وتلغي بعد زمن الصلاحية.

تعبّر العلاقة الخرج عن المعلومات التي اعتقدت (أو مازالت تعتقد) قاعدة المعطيات بصلاحيتها في اللحظة أو ضمن المدة المحددة.

4-1-4 تشريح زمن المناقلة Transaction-Time Slicing

تأخذ هذا العملية علاقة زمن مناقلة أو علاقة ثنائية الزمن وتعيد علاقة لازمنية أو علاقة زمن صلاحية على الترتيب، كما تأخذ لحظة أو مدة زمنية وحيدة.

تحوي العلاقة الخرج على المعطيات التي يحوي أو يتقاطع زمن مناقلتها مع اللحظة أو المدة المحددة، وتلغي بعد زمن المناقلة.

تعبّر العلاقة الخرج عن المعلومات التي كانت حاليةً (أو مازالت) في قاعدة المعطيات في اللحظة أو ضمن المدة المحددة.

5-1-4 التشريح ثنائي الزمن

تأخذ هذا العملية علاقة ثنائية الزمن وتعيد علاقة لازمنية، كما تأخذ معاملاً آخرَ هو لحظة أو مدة زمنية وحيدة.

تحوي العلاقة الخرج على المعطيات التي يحوي أو يتقاطع كل من زمن مناقلتها وصلاحيتها مع المعامل الثاني، وتلغي الأبعاد الزمنية.

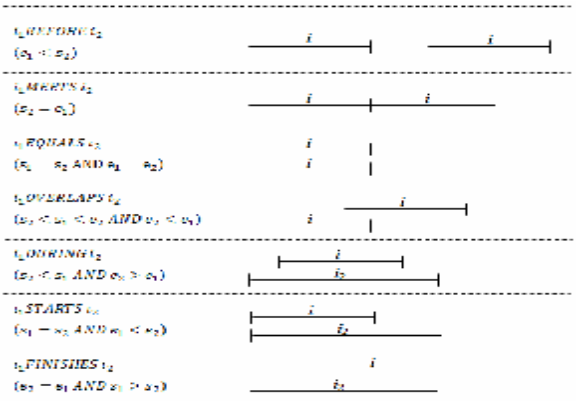
تعبّر العلاقة الخرج عن المعلومات التي كانت حاليةً (أو مازالت) في قاعدة المعطيات في اللحظة أو ضمن المدة المحددة، وفي الوقت نفسه كانت قاعدة المعطيات تعتقد (أو مازالت تعتقد) بصلاحيتها في اللحظة أو ضمن المدة المحددة أيضاً.

التوابع الزمنية بين المدد

بفرض لدينا المدتان i_1، i_2 بحيث يكون:

$$i_1 = [s_1, e_1];\ i_2 = [s_2, e_2]$$

يوضح الجدول الآتي التوابع الزمنية بين هاتين المدتين:



وفيما يأتي جدول يوضح بعض التوابع الزمنية بين المدد في بعض لغات الاستعلام الزمنية، ونورد في بدايته اللغة التي قمنا بتطويرهاzSQL المستعملة في نظامنا"زمن".

Allen(19)	zSQL	TSQL	TSQL2	HSQL	TQuel
a before b	a < b or b > a	a BEFORE b or b AFTER a	a PRECEDES b	a PRECEDES b	a precede b and not (end of a equal begin of b)
a meets b equal begin of b	ENDOF(a) = STARTOF(b)	a PRECEDES b or b FOLLOWS a	END(a) = BEGIN(b)	a MEETS b	end of a equal begin of b
a overlaps b	a OVERLAPS b	a OVERLAP b AND a.TIME-END PRECEDES b.TIME-END	a OVERLAPS b AND END(a) PRECEDES END(b)	a OVERLAP b AND a.TO < b.TO	a overlap b and end of a precedes end of b
a during b	a DURING b	a DURING b	BEGIN(b)PRECEDES BEGIN(a) AND END(a) PRECEDES END(b)	b CONTAINS a	begin of a overlap b and end of a overlap b and not (a equal b)
a meets b or b meets a	a MEETS b	a ADJACENT b	END(a) = BEGIN(b) or END(b) = BEGIN(a)	a ADJACENT b	end of a equal begin of b or end of b equal begin of a
a starts b	a STARTS b	(a.TIME-START EQUIVALENT b.TIME-START) AND (a.TIME-END PRECEDES b.TIME-END)	BEGIN(a) = BEGIN(b) AND END(a) PRECEDES END(b)	a.FROM = b.FROM AND a.TO < b.TO	begin of a equal begin of b and end of a precede end of b
a finishes b	a ENDS b	(a.TIME-START AFTER b.TIME-START) AND (a.TIME-END EQUIVALENT b.TIME-END)	a.FROM>b.FROM AND a.TO = b.TO		begin of a precede begin of b and end of a equal end of b

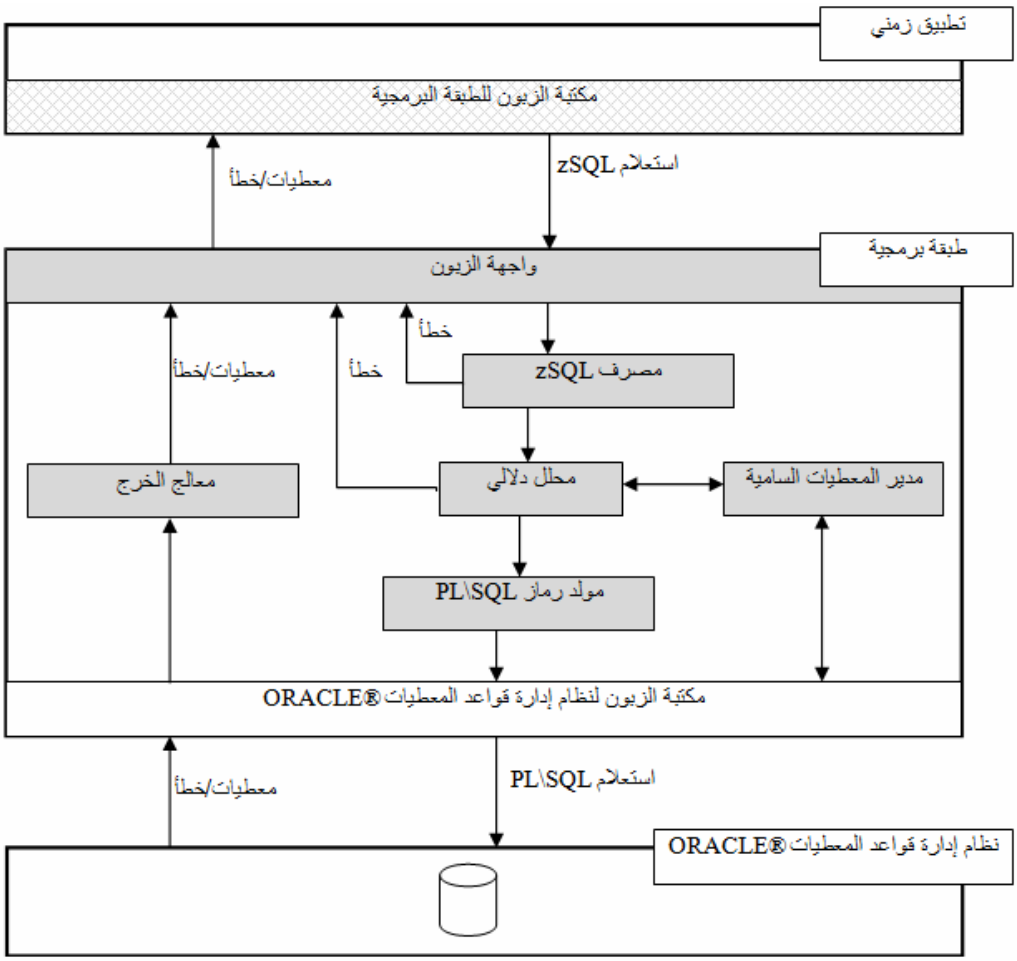
5-نظام إدارة قواعد المعطيات الزمنية المنجز "زمن"

يهدف نظام "زمن" إلى توفير بيئة برمجية مساعدة لعمل التطبيقات ذات الدلالات الزمنية، من خلال تطوير طبقة برمجية تغلف نظام إدارة قواعد معطيات علائقي، مدعمة بلغة استعلام زمنية قمنا باشتقاقها من SQL-92 بحيث تكون متوافقة معها ليستعملها التطبيق الزمني الزبون لمخاطبة هذه الطبقة، ويتكون من جزئين:

1. طبقة برمجية (محرك) تغلف نظام إدارة قواعد المعطيات الموجود.
2. مكتبة برمجية تحوي بنى المعطيات الزمنية وتسمح للزبون بإنشاء اتصال مع الطبقة البرمجية المذكورة.

كما يمكن توفير أداة برمجية مساعدة للمطور لإرسال الاستعلامات مباشرة إلى قاعدة المعطيات الزمنية.

سنعرض فيما يأتي تفصيلاً لجزئي النموذج عند تطبيقه على النظام.



تمثل المستطيلات البيضاء (□) مكتبات أو تطبيقات برمجية متوافرة، ويمثل المستطيل المظلل (▨) مكتبة الزبون، فيحين تمثل المستطيلات الرمادية (■) أهم أجزاء الطبقة البرمجية الواجب تحقيقها، وهي:

5-1 مصرف لغة الاستعلام الزمنية

يأخذ عبارات مكتوبة بلغة الاستعلام الزمنية التي يدعمها النظام، وهي لغة zSQL ويقوم بتحليلها إملائياً ونحوياً، ثم يعيد النتيجة –شجرة التركيب النحوي المجرد AST- إلى المحلل الدلالي، كما يقوم بالكشف عن الأخطاء الإملائية أو النحوية -إن وجدت- ويُعلم التطبيق الزبون بها.

بُنِيَ المصرف باستعمال JavaCC 4.2.

2-5 المحلل الدلالي

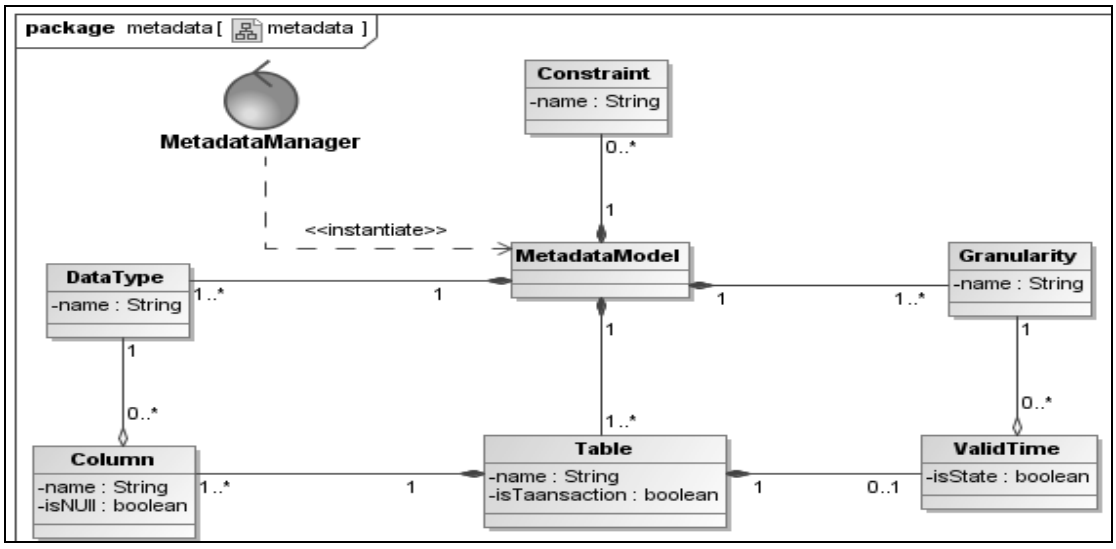
يعالج المحلل الدلالي خرج المصرف بالاستعانة بمدير المعطيات السامية ليحدد معنى أو دلالة العبارات المرسلة من التطبيق الزبون، ثم يعيد النتائج إلى مولد الرماز، كما يقوم بالكشف عن الأخطاء الدلالية -إن وجدت- ويعلم التطبيق الزبون بها، ويولد رسائل التحذير عند الحاجة لذلك.

لا تقتصر علاقة المحلل الدلالي مع مدير المعطيات السامية بالقراءة فقط، بل إنه يقوم بالكتابة في المعطيات السامية عند استلام تعليمات DDL.

3-5 مدير المعطيات السامية

بوجود مفاهيم جديدة مثل الحبيبة الزمنية والقيود الزمنية التي لا يقوم نظام إدارة قواعد المعطيات التجاري بتأمينها ظهرت الحاجة لبناء مدير لهذه المعطيات، حيث يتولى مدير المعطيات السامية حفظ المعطيات المرتبطة بالأغراض ضمن قاعدة المعطيات كالجداول والحقول وأنماط المعطيات وغير ذلك، ويستعين بنظام إدارة قواعد المعطيات الهدف لتخزين هذه المعطيات السامية.

نعرض في الشكل الآتي مخطط الصفوف المفهومي لبنى المعطيات السامية:



مولد الرماز

يحصل مولد الرماز على دخله من المحلل الدلالي، ويقوم بتوليد رماز بلغة الاستعلام اللازمنية التي يدعمها نظام إدارة قواعد المعطيات الهدف، وهي SQL\PL بما يتناسب مع دلالة العبارات الواردة بلغة الاستعلام الزمنية وهي zSQL - التي عُرِّفَتْ وهي لغة شبيهة باللغة الزمنية المعيارية TSQL2، إلا أن zSQL متوافقة زمنياً مع SQL-92 على عكسTSQL2-، ثم يقوم بإرسالها إلى نظام إدارة قواعد المعطيات الهدف.

وصُمِّمَ النظام بحيث يمكن تغيير نظام إدارة قواعد

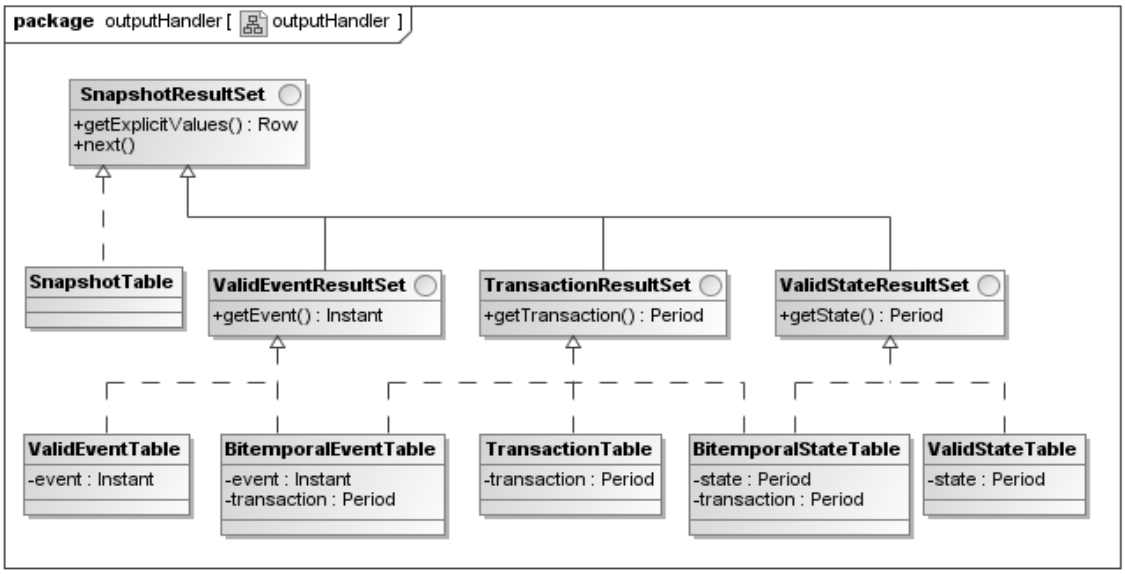
المعطيات الهدف بسهولة، وذلك بتغيير مكوّن توليد الرماز فقط دون الحاجة إلى أي تغيير آخر.

نقوم حالياً بالعمل على تطوير مولد الرماز الخاص بتوليد اللغة SQL\PL لرفع أدائه والاستفادة من التقنيات التي يمنحها محرك قواعد المعطيات ORACLE، كما نقوم حالياً بتطوير مولد رماز للعمل على نظام إدارة قواعد المعطيات SQL-Server وأيضاًMySQL .

4-5 معالج الخرج

يستلم معالج الخرج من نظام إدارة قواعد المعطيات الهدف نتائج الاستعلام المولد ويعيد تشكيلها لتتناسب مع نموذج المعطيات الزمني الذي يتوقعه الزبون.

نعرض في الشكل الآتي مخطط الصفوف المفهومي لمعالج الخرج:



واجهة الزبون

تعمل كخدمة في الخلفية Background Process (Daemon) وتنتظر طلبات الزبائن لإنشاء اتصال مع الطبقة، ثم تدير هذا الاتصال حتى انتهائه.

تؤمن الواجهة الحالية إمكانية التحقق من هوية المستخدم ولكنها لا تدعم إدارة السماحيات التي يتم العمل عليها حالياً وتؤمن إمكانية التحكم بالوصول المتزامن لأكثر من مستخدم، وبُنِيَتِ الواجهة باستعمال تقنيات Java Sockets.

5-5 مكتبة الزبون

يمثل المستطيل المظلل (▨) في الشكل مكتبة الزبون، وهي مكتبة برمجية تُوزَّع مع النظام ليقوم المطور بتضمينها ضمن التطبيق الزمني، تحوي هذه المكتبة بنى المعطيات الجديدة (الزمنية)، كما تمكن التطبيق من إنشاء اتصال مع الطبقة البرمجية لإرسال الاستعلامات الزمنية واستلام النتائج. مكتبة الزبون الحالية تعمل على Java ويتم العمل على تطوير مكتبة أخرى للعمل على بيئة .NET. .

6- الخلاصة

قمنا في هذا البحث باستعراض موجز لمقاربة مقترحة لبناء نظام إدارة قواعد معطيات زمنية، يحتاج هذا النظام إلى لغة استعلام زمنية خاصة تراعي الدلالات الزمنية وتحافظ على التوافقية مع الاستعلامات التقليدية، أطلقنا على هذه اللغة اسمzSQL وسنتناولها في مقالة قادمة.

المراجع:

1. *Extending Temporal Databases to Deal with Telic/Atelic Medical Data*. Terenziani, P., et al. 2, s.l. : Elsevier Science Publishers Ltd., 2007, Artificial Intelligence in Medicine, Vol. 39, pp. 113-126. ISSN:0933-3657.
2. Jensen, C. S. Introduction to Temporal Database Research. [book auth.] R. T. Snodgrass. *The TSQL2 Temporal Query Language*. 1995, pp. 1-27.
3. Jensen, C. S. and Dyreson, C. E., [ed.].*A Consensus Glossary of Temporal—February 1998 Version*. 21. 1998. pp. 367–405.
4. Snodgrass, R. T., et al. A TSQL2 Tutorial. [book auth.] R. T. Snodgrass. *The TSQL2 Temporal Query Language*. 1995.
5. Soo, M. D., Jensen, C. S. and Snodgrass, R. T. An Algebra for TSQL2. *The TSQL2 Temporal Query Language*. s.l. : Kluwer Academic Publishers, 1995, 27, pp. 505--546.
6. *Sematics of Time-Varying Information*. Jensen, C. S. and Snodgrass, R. T. 4, 1996, Information Systems, Vol. 21, pp. 311-352.
7. *Spatial, Temporal and Spatio-Temporal Databases - Hot Issues and Directions for PhD Research*. Roddick, J. F., et al. 2, s.l. : ACM, 2004, ACM SIGMOD Record, Vol. 33, pp. 126 - 131. ISSN:0163-5808.
8. *Unifying Temporal Data Models via a Conceptual Model*. Jensen, C. S., Soo, M. D. and Snodgrass, and R. T. 7, 1994, Information Systems, Vol. 19, pp. 513–547.
9. *Further Normalization of the Data Base Relational Model*. Codd, E. F. [ed.] Englewood Cliffs. 1972, Data Base Systems. Vol. 6 of Courant Computer Symposia Series.
10. *Temporal Databases*. Snodgrass, R. T. and Ahn, I. 9, 1986, IEEE Computer, Vol. 19, pp. 35-42.
11. *A Temporally Oriented Data Model*. Ariav, G. 4, 1986, ACM Transactions on Database, Vol. 11, pp. 499-527.
12. Ben-Zvi, J.*The Time Relational Model*. Computer Science Department, UCLA. 1982. Ph.D. dissertation.
13. *The Temporal Query Language TQuel*. Snodgrass, R. T. 2, 1987, ACM Transactions on Database Systems, Vol. 12, pp. 247-298.
14. Rowe, L. A. and Stonebraker, M. R.*The Postgres Papers. Memorandum UCB/ERL M86/85*. Electronics Research Laboratory, College of Engineering, University of California, Berkeley CA 94720. 1987.
15. *Adding Time Dimension to Relational Model and Extending*. Tansel, A. U. 4, 1986, Information Systems, Vol. 11, pp. 343–355.
16. Thompson, P. M.*A Temporal Data Model Based on Accounting Principles*. Department of Computer Science, University of Calgary, Calgary, Alberta, Canada. 1991. Ph.D. dissertation.
17. *Structured Organization of Clinical Data Bases*. Wiederhold, G., Fries, J.F. and Weyl, S. s.l. : AFIPS, 1975. AFIPS National Computer Conference. pp. 479-485.
18. *Coalescing in Temporal Databases*. Böhlen, M. H., Snodgrass, R. T. and Soo, M. D. s.l. : Morgan Kaufmann Publishers Inc, 1996. 22th International Conference on Very Large Data Bases. pp. 180-191. ISBN:1-55860-382-4.
19. *Maintaining Knowledge about Temporal Intervals*. Allen, J. F. 11, 1983, Communications of the Association of Computing Machinery, Vol. 26, pp. 832–843.

تاريخ ورود البحث إلى مجلة جامعة دمشق 2010/3/15.