مجلة جامعة تشرين للدراسات و البحوث العلمية _ سلسلة العلوم الهندسية المجلد (27) العدد (3) 2005
Tishreen University Journal for Studies and Scientific Research- Engineering Sciences Series Vol. (27) No (3) 2005

مساهمة في أسس تصميم محركات قواعد معطيات لغة التأشير القابلة للتوسيع XML، ومعايير تقييم واختبار أدائها

الدكتور غسان فلوح*

(قبل للنشر في 2005/11/27)

□ الملخّص □

أصبحت لغة التأشير القابلة للتوسيع XML هي اللغة الأكثر استخداماً في مجال توليد وتبادل المعلومات والوثائق بين التطبيقات المختلفة. ونظراً إلى أهمية تخزين واسترجاع هذه الوثائق وتبادلها بشكل فعال وموثوق فإن هذا البحث يقدم في قسمه الأول مساهمة في أسس تصميم محركات قواعد معطيات XML الأصيلة، وذلك من خلال وضع وتوصيف المكونات الأساسية لهذه المحركات وبيان وظائف كل منها وطرق ارتباطها بقواعد المعطيات. ثم يضع البحث في قسمه الثاني بعض المعايير الأساسية في طرق اختبار ومحاكاة محركات قواعد معطيات XML، وذلك كي يتسنى التعرف على أدائها وتحسين كفاءة عملها ومدى استجابتها للاستعلامات في التطبيقات المتعاظمة في مجال الأعمال. ثم يقدم البحث النتائج التي تم الحصول عليها نتيجة الاختبارات العملية لمحاكاة البنى التي جرى اقتراحها والمعايير الموضوعة لاختبار محرك قواعد معطيات XML، ومن ثم تعميم واستخلاص النتائج وتعميمها.

أثبت الجانب العملي من هذا البحث أن النتائج التي جرى الحصول عليها ذات أهمية كبيرة، حيث تبين وجود تحسن كبير في زمن احتساب الاستفسارات عندما تقارن النتائج مع تلك الخاصة بقواعد المعطيات التقليدية. كما بين البحث أن المعايير التي وضعت في هذا البحث لتقييم أداء محركات قواعد معطيات XML التي جرى المساهمة في بعض أسس تصميمها يمكن أن تسهم إلى حد بعيد في تحسين أداء هذه قواعد المعطيات.

الكلمات المفتاح:

لغة التأشير القابلة للتوسيع XML، هيئة، محرك قاعدة معطيات، بنية معطيات شجرية، النموذج الغرض للوثيقة، معطيات مُتزّفعة، غرض

* أستاذ مساعد – قسم هندسة الحاسبات والأتمتة – كلية الهندسة الميكانيكية والكهربائية – جامعة دمشق– دمشق– سوريا.

مجلة جامعة تشرين للدراسات و البحوث العلمية _ سلسلة العلوم الهندسية المجلد (27) العدد (3) 2005
Tishreen University Journal for Studies and Scientific Research- Engineering Sciences Series Vol. (27) No (3) 2005

Contribution to the Design Principles of XML Database Engines and the Criteria for Their Performance Evaluation

Dr. Ghassan Fallouh*

(Accepted 27/11/2005)

□ ABSTRACT □

XML language has become the most widely used tool for data and document exchange among different applications. This research presents in its first part a contribution to the design principles of XML database engines in light of describing the main components that constitute them as well as the their functionalities with respect to databases.

The second part of this research lays down the essential foundations and methodologies for testing and simulating XML database engines in order to evaluate and improve the criteria for their performance evaluation. This paper is concluded with the results obtained from the experimental studies of the simulation of the proposed database engine based on the suggested design principles and criteria for evaluation.

The outcome of this research proved to be valuable in terms of the results obtained from the experimental part. The improvement of processing time during query evaluation was considerable and shown to bypass traditional database engines. Also, the foundation put together for the performance evaluation of the design principles of XML database engines has shown to be an essential part of the design process.

Key Words: XML, tags, database engine, tree data structure, Document Object Model (DOM), meta-data, object

* **Associate Prof, Department Of Computer Engineering & Automation, Faculty Of Mechanical & Electrical Engineering, Damascus University, Damascus, Syria.**

مقدمة:

أصبحت لغة التأشير القابلة للتوسيع Extensible Markup Language (XML) بشكل متسارع لغة التبادل الإلكتروني للمعطيات بين التطبيقات. وقد بدأ العمل بهذه اللغة في صيف 1996 بعد أن كانت تدعى لغة التأشير المعممة القياسية Standard Generalized Markup Language (SGML) والتي جاءت كنتاج لفعاليات World Wide Web Consortium (W3C) حيث قدمت هذه اللغة حلولاً عملية للإشكالات التي تعاني منها لغة HTML أو التطبيقات التي تستخدم SGML ,[Bray 2000] يمكن أن نذكر منها على سبيل المثال ما يلي:

1- إن عناصر تعريف المتحولات التي يمكن للمبرمج استخدامها ليست محدودة وإنما يمكن دوماً إنشاء هيئات Tags وأنماط Types جديدة قابلة لإعادة الاستخدام.

2- تجاوزت XML التعقيدات الكبيرة والمتعددة للغة SGML جاعلة عملية إنشاء أنواع الملفات أكثر سهولة ة ومرونة.

3- أصبح من الممكن استخدام الهيئات Tags الموصفة من قبل المبرمج في توصيف الملفات والمعطيات مما جعل تحليل parse وتبادل المعطيات بين أنواع التطبيقات المتوفرة على منصات Platforms متباينة أمراً ممكناً.

إن قواعد المعطيات التي تستطيع تخزين معطيات XML والتعامل معها يمكن أن تصنف وفق ما يلي:

أ- قواعد معطيات XML أصيلة native.

ب- قواعد معطيات تقبل التعامل مع XML (أي XML-enabled).

وبذلك يمكن لنا أن نعرّف قاعدة المعطيات الأصيلة لبيئة XML بأنها تلك المصممة خصيصاً لحفظ وثائق XML والاستفسار (الاستعلام) عنها. كما أن تصميم قاعدة لمعطيات يجب أن يحوي على بنى معطيات تستطيع أن تصون البنى الهرمية للغة XML وتوظف معرفتها في أمثلة optimization معالجة الاستفسارات. وهذا يعني بالضرورة أن قواعد المعطيات التي تقبل التعامل مع XML يجب أن يتوفر فيها طبقات إضافية كي تربط أنواع معطيات XML مع بنى المعطيات المتوفرة فيها.

ولا بد من التنويه إلى أن توفر معطيات XML أصيلة في قواعد المعطيات يعطيها دفعاً قوياً في تطبيقات الأعمال الإلكترونية e-business حيث أصبحت XML المعيار الواقعي لتبادل المعلومات في تطبيقات الأعمال التي تعرف بـ Business to Business Applications (B2B) [Moller 2001]، وذلك لأسباب عدة قد يكون أهمها أن ملفات XML مفتوحة وقابلة للاستخدام والمشاركة بين نظم حاسوبية وبرمجية متباينة من حيث البنية والنوعية.

يقدم هذا البحث في قسمه الأول مساهمة في أسس تصميم محركات قواعد معطيات XML وذلك من خلال وضع المكونات الأساسية لهذه المحركات وبيان وظائف كل منها وطرق ارتباطها بقواعد المعطيات.

ثم يضع البحث في قسمه الثاني بعض المعايير الأساسية في طرق اختبار ومحاكاة محركات قواعد معطيات XML وذلك كي يتسنى التعرف على أدائها وتحسين كفاءة عملها ومدى استجابتها للتطبيقات المتعاظمة في مجال الأعمال. ثم يقدم البحث النتائج التي تم الحصول عليها نتيجة اختبار محاكاة البنى التي جرى اقتراحها والمعايير الموضوعة لاختبار محرك قواعد معطيات XML المصمم ومن ثم تعميم هذه النتائج وآفاق البحث المستقبلية.

النفاذ إلى قواعد المعطيات في بيئة الوب:

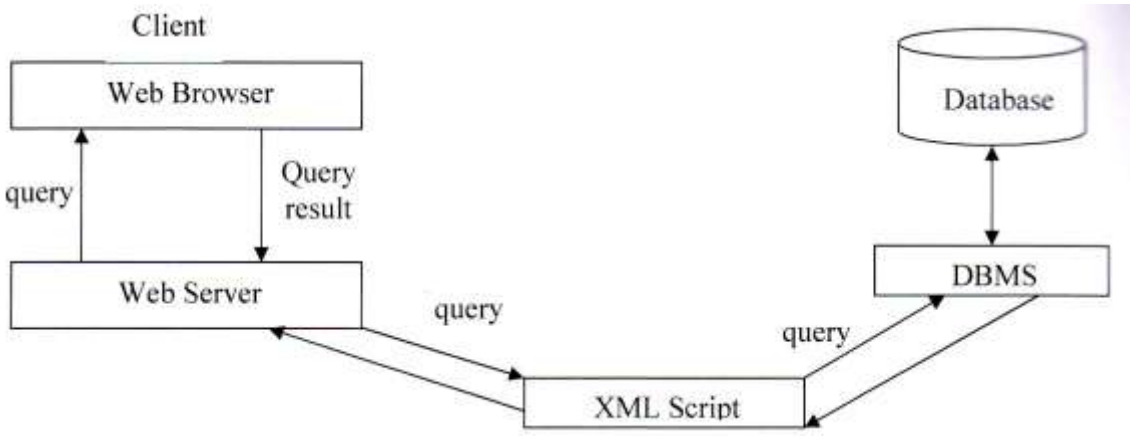
إن البيئة الأساسية للنفاذ إلى قواعد المعطيات المعروفة باسم بيئة زبون – مخدم client – server طرأ عليها حديثاً عدد من التطورات الهامة أخذت تنهج نحو بيئة أكثر قدرة على التعامل مع محيط الوب. هذه البيئة الحديثة أصبحت تعرف باسم زبون – شبكة client-network في إشارة إلى شبكة الوب العالمية والتي يجري خلالها النفاذ من وإلى الشبكة عبر مخدم وب web server كان في وقت قريب يتوفر عليه تطبيقات مكتوبة بلغة HTML التي توفر عدداً من الأدوات التي تسمح للمبرمجين بإنشاء صفحات وروابط للمواقع المختلفة إضافة إلى التعامل مع الوسائط المتعددة multimedia من صوت وصورة وفيديو [Halaschek 2003].

وقد أعطى ذلك للمستخدمين قدرات كبيرة للبحث والتنقيب عن المعلومات التي تكون عادة موزعة على العديد من المواقع الموجودة في أماكن جغرافية متباعدة.

إلا أن التقنيات الحديثة وضمن تطورها المتسارع أصبحت بحاجة إلى التعامل ليس فقط مع معلومات ساكنة static بشكل دائم وإنما أيضا مع معلومات ديناميكية متغيرة تتوافر في قواعد معطيات تتمثل في كل لحظة instance. وقد كانت إحدى المحطات الهامة في إطار هذه التطورات استخدام واجهة البوابة المشتركة Common Gateway Interface (CGI) التي عملت كبرمجيات وسيطة بين قواعد المعطيات المختلفة ونظم التشغيل أو التطبيقات البرمجية.

تقوم البرمجيات الوسيطة CGI بتنفيذ البرامج للحصول على المعلومات الديناميكية التي تعيدها إلى المخدم على شكل HTML حيث تظهر بعدئذ للمستثمر ضمن المتصفح. وقد ظهرت حاجات ملحة مع تطور الوب بتحوله إلى وسط لا متناه من المعلومات المختلفة الأنواع والأنماط والسماح للمشغلين من النفاذ ليس إلى النظم فحسب وإنما إلى قواعد المعطيات ومعالجة الاستفسارات وتوليد التقارير ...ألخ.

يستدعي كل ما سبق اللجوء إلى نظام توظف فيه معايير جديدة ولغات برمجية لتبادل المعلومات مع الانترنت. إحدى أهم الوسائل الحديثة هي لغة التأشير القابلة للتوسيع XML التي تسمح بتوسيع أو تعديل التركيب النحوي للغة syntax ودلالاتها semantics. ويبين الشكل (1) إحدى طرق النفاذ إلى قواعد المعطيات من خلال متصفح وب.



الشكل (1): النفاذ إلى قواعد المعطيات في بيئة الوب باستخدام لغة XML

البنية الأساسية لقاعدة معطيات XML:

إن المكونات الأساسية لقاعدة معطيات XML أصيلة هي:

1- أداة استفسار Query Tool

2- معالج استفسارات query processor

3- محرك قاعدة المعطيات database engine

حيث يترتب توفير واجهة موائمة بيانية GUI للتعامل مع المشغلين users بشكل تفاعلي interactive، تتضمن هذه الواجهة أدوات مثل:

1- إضافة مخطط قاعدة معطيات Database Schema إلى مجموعة الجداول التي تتضمن وصفاً لبنية قاعدة المعطيات metadata catalog،[Miller 2000] .

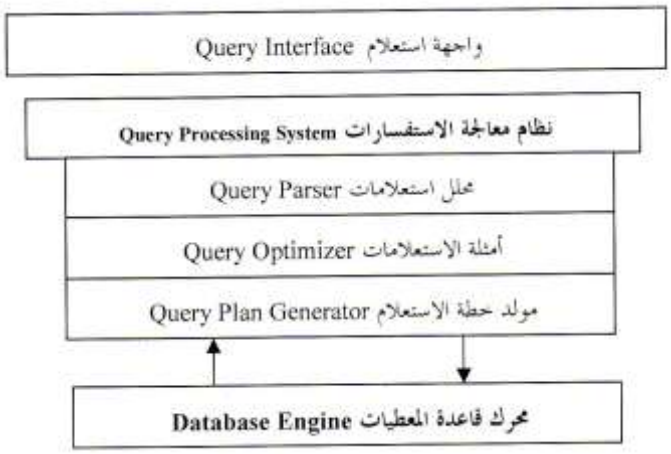
2- إضافة وثائق documents إلى حافظة المعطيات data depository.

3- إجراء الاستفسارات.

4- تخزين المعلومات في قاموس المعطيات data dictionary.

يقوم معالج الاستفسارات بتحليل parse الاستعلامات ثم يولد خطة تقيم أمثلية لتنفيذها.

يبين الشكل (2) بنية قاعدة معطيات XML وبنية مقترحة لمحرك قاعدة المعطيات.



الشكل (2): بنية قاعدة معطيات XML

الوظائف الأساسية المطلوبة لمحرك قاعدة معطيات XML:

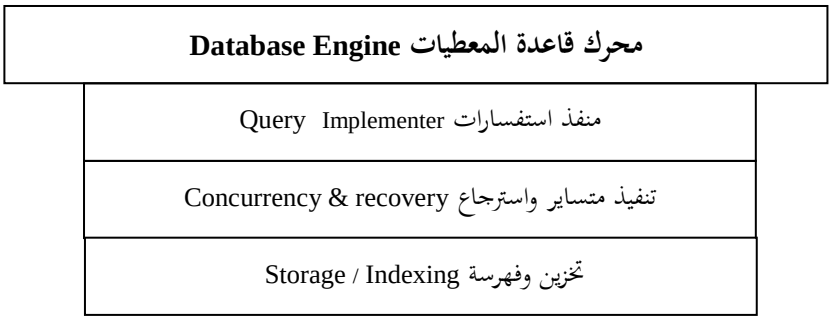
إن محرك قاعدة المعطيات هو الجزء الأساس من قاعدة المعطيات الذي يصون بنى التخزين ويقوم بعمليات الفهرسة، إضافة إلى إدارة المناقلات transactions واسترجاع المعلومات recovery وتوفير تكاملية integrity وأمن وسلامة المعلومات من خلال حجب أو منح الصلاحيات.

إضافة لما سبق، فإن أهمية محرك قاعدة معطيات XML تتمثل في الدور الجديد الذي بدأ يأخذه في الابتعاد عن التعامل مع زبائن خدمات الوب الذين يستخدمون متصفحات browsers للتفاعل مع هذه الخدمات، والاقتراب من مفهوم "خدمات تطبيق إلى تطبيق في محيط الوب" application to application Web service والتي تستدعي وجود لغات برمجية مبنية على XML تقوم بدور الوسيط بين الزبائن والمخدمات التي تتوافر عليها قواعد معطيات XML.

إن هذا الدور المتنامي سوف يمكِّن واجهات تخاطبية ومحركات قواعد معطيات XML ليس فقط في البحث عن المعلومات وإنما تبادلها واستخلاصها أخرى وتحويلها إلى صيغ قابلة للتخزين في قاعدة معطيات XML.

إن الوظائف الأساسية المقترحة في هذا البحث لمحرك قاعدة معطيات XML الموضحة في الشكل (3) هي:

1- نظام تنفيذ الاستفسارات Query Implementation System:



الشكل (3): الوظائف الأساسية لمحرك قاعدة معطيات XML

يجب أن يتمتع نظام تنفيذ الاستفسارات الخاص بتصميم محرك قاعدة معطيات XML المقترح ذي الخصائص العامة بالمواصفات الآتية:

- القدرة على العمل مع لغات الاستعلام المعروفة والتي تتوافق مع مقترحات W3C.
- توفر محلل parser ذي واجهة مواءمة لبرمجة التطبيقات API.
- يجب أن يتمكن هذا المحلل من التعامل مع النموذج الغرضي للوثائق Document Object Model (DOM) [Horse 2000]، والذي يقوم على بنية شجرية tree structure تعمل في بيئة API. كما يجب أن يقبل التعامل مع النموذج المبسط للتطبيقات البرمجية (SAX) [Megginson 2001].

2- تعريف بنى المتحولات المختلفة Variable structures:

لقد وفرت لغة XML بنى معطيات تستطيع أن تمثل أنواع المعطيات التي تحويها الوثائق كافة. يشمل الإعلان عن المتحولات نوع المجموعة المحرفية المعيارية Character set المستخدمة (مثل ISO 8859-1 to9)، وتعريف نوع الوثيقة Document Type Definition (DTD)، كما يترتب توفير وعاء برمجي يحوي العناصر البرمجية الأخرى كافة في الوثيقة root elements ويكون بمثابة البنية الأعلى فيها.

إن أنواع المتحولات المتوافرة التي تحاكي الواقع هي تلك الخاصة بالمتحولات العددية (صحيحة وحقيقية)، والمحارف وأشرطتها (characters & strings)، والمتحولات المنطقية، والمتحولات من نوع محدد الموارد النظامية Uniform Resource Identifier **(URI)**. يقوم هذا الأخير بتحديد العناصر والواصفات attributes الخاصة بها من خلال مخطط XML Schema.

سنبين لاحقاً في هذا البحث كيفية تمثيل وتعريف أنواع المتحولات المختلفة من خلال نموذج الاختبار الذي تم بناؤه.

3- توفير بنى التخزين Storage structures:

لقد جرى اقتراح العديد من طرق ربط معطيات XML إلى قواعد المعطيات العلائقية Relational Database Systems [Arpinar 2001] ، حيث يخزن محرك قاعدة المعطيات XML المعطيات من نوع DTD/XSD (Document Type definition & XML Schema Definition)كمعطيات عن المعطيات فيما يطلق عليه اسم معطيات مُترّفعة meta–data، حيث يستخدمها في تخزين وثائق XML وإجراء الاستعلامات عليها.

تخزن قواعد المعطيات بشكل فيزيائي كملفات تسجيلات record files يمكن استرجاعها كلما تطلب التطبيق ذلك أو كلما ظهرت الحاجة إلى تحديثها أو معالجتها.

تتكون كل تسجيلة من مجموعة من المعطيات والقيم التي تأخذها الكينونات وواصفاتها والعلاقات فيما بينها، وتخزن التسجيلات بطريقة تسهل عملية الاسترجاع بهدف المعالجة والتعديل أو الحذف.

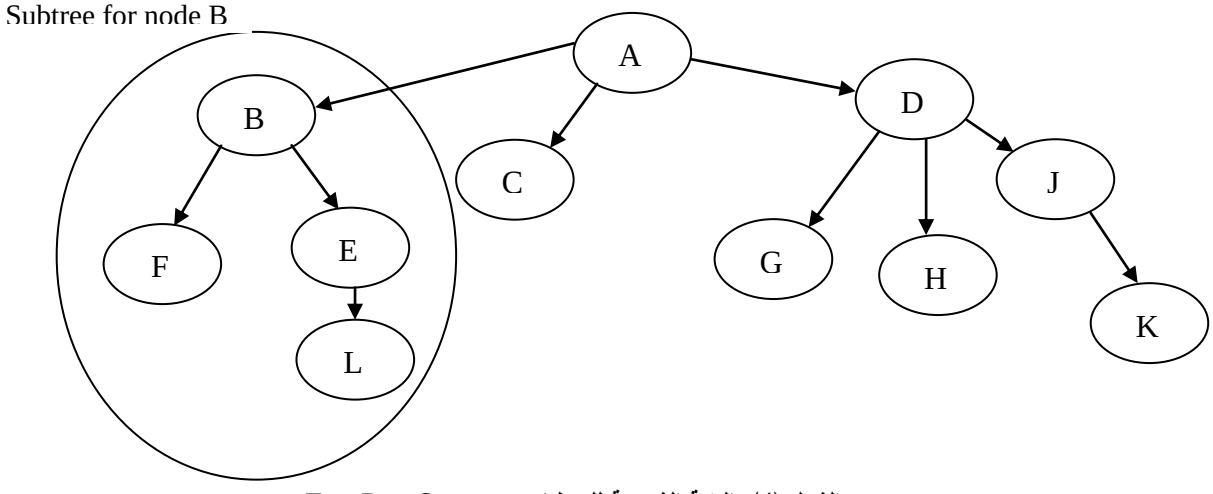
وهناك عدة طرق لتخزين الملفات نذكر منها:

أ- طريقة التخزين غير المنتظمة Unordered files (أو الكومة heap) حيث تخزن المعطيات على وسط التخزين دون أي ترتيب حيث يتم إضافة تسجيلات جديدة إلى نهاية الملف.

ب- طريقة التخزين التسلسلية للملفات sequential files (أو الملفات المفروزة sorted files) التي تبقي على الملفات مفروزة بناء على حقل يطلق عليه اسم حقل الفرز sort key.

ت- طريقة التخزين باستخدام ملفات التلبيد hash files والتي تستخدم توابع عددية لتحديد موقع التسجيلة على وسط التخزين.

ونظراً للطبيعة الخاصة لقواعد معطيات XML فقد تبين لنا نتيجة هذا البحث أن بنية التخزين المناسبة، هي تلك التي توظف البنية الشجرية tree structure لتنظيم تخزين الملفات، حيث توفر هذه البنية إمكانيات التخزين والاسترجاع التسلسلي sequential أو العشوائي random، وهذا الاستنتاج يتوافق بشكل عام مع العديد من التطبيقات العملية العديدة المتوفرة في محيط الوب. تقوم هذه الطريقة على توظيف بنية شجرية تستخدم عملياً لأغراض التخزين والبحث عن التسجيلات للوصول إلى القيم أو للمواقع المناسبة، كما هو مبين في الشكل (4).



الشكل (4): البنية الشجرية للمعطيات Tree Data Structure

شروط عمل محرك قاعدة معطيات XML المقترح:

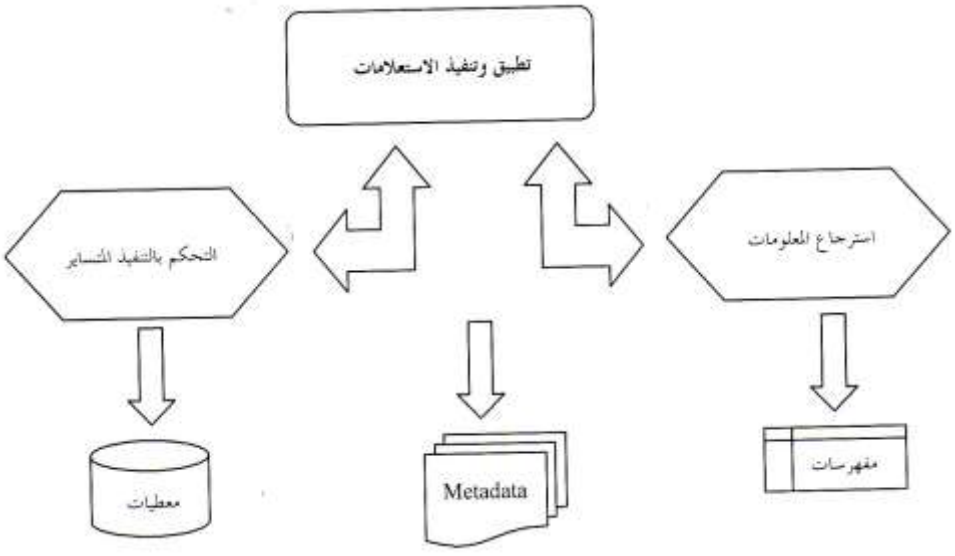
لقد بينا سابقاً أن المكونات الأساسية اللازمة لمحرك قاعدة معطيات XML من بنى تخزين وواجهة لبرمجة التطبيقات API والقيام بالاستفسارات وعمليات الفهرسة indexing والتحكم بالعمليات المتسايرة concurrency control، توفر بيئة ديناميكية لتبادل المعلومات ضمن الأشكال الأصيلة لها.

يجب أن يتوفر في محرك قاعدة معطيات XML تمثيل مبيت لأنواع المعطيات الأساسية وهياكل لتوسيع إمكانيات تعريف أنماط جديدة، وطرق تخزين تتناسب مع تلك الهرمية hierarchical للعلاقات بين العناصر elements في وثائق XML. ولا بد من التنويه أيضا إلى أن قاعدة معطيات XML أصيلة native يجب أن تتمكن من احتواء أنماط وثائق من النوع DTD أو XSD [Choi 2002] التي تتفوق على سابقتها بأنها توفر تنميطاً أقوى للمعطيات والواصفات Attributes.

إن تخزين المعطيات بأنماط مثل XSD/DTD يسهم في إغناء عمليات تنفيذ الاستفسارات ضمن بيئة API. كما يجدر بمحرك قاعدة المعطيات التعامل مع النموذج الغرضي للوثائق Document Object Model (DOM) الذي يحدد بنية المعلومات الهرمية للوثيقة والعلاقات من نوع أب – أبن parent-child بين العناصر الموجودة فيها. كما أن التنفيذ المتساير concurrency الذي يجب أن يتوفر في محرك قاعدة المعطيات تؤمن حل العديد من المشاكل التي قد تفرض تخزين ومعالجة المعلومات في بيئة DOM.

البنية التفصيلية للمكونات الأساسية الخاصة بتصميم محرك قاعدة معطيات XML

إن الوظائف الرئيسة لمحرك قاعدة المعطيات تتمثل في تخزين المعطيات المترفعة والمعلومات المتعلقة بها meta-data وتقديم التسهيلات للتنفيذ المتساير concurrency، ومعالجة الاستفسارات Query Processing وفق ما هو مبين في الشكل (5).



الشكل (5): بنية مقترحة لمحرك قاعدة معطيات XML

وبناء على الوظائف المبينة والمطلوبة من محرك قاعدة المعطيات فيترتب على هذا المحرك تخزين المعطيات كعناصر غرضية Element Objects من خلال جداول تلبيد hash tables، وذلك بما يتوافق مع ما افترضته لغة

XML، كما ويترتب على محرك قاعدة المعطيات مهمات إضافية مثل تخزين معطيات مُترّفعة (معلومات المعطيات) meta-data كأغراض أنماط العناصر Element type Objects.

كما لا بد من توفير متحولات مرجعية تشير إلى حقول الفهرسة الخاصة بالجداول وقيمها. إن تخديم التنفيذ المتساير يتطلب وجود مدير للمناقلات Transaction Manager يحتفظ في كل لحظة بمحددات المناقلات النشطة وقائمة بكافة العمليات التي جرت من قبلهم.

تخزن المعلومات الخاصة بكل عملية مثل (تحديث المعلومات update، الانتخاب select، إقحام المعطيات insert، الحذف delete) من خلال نمط يمكن أن يطلق عليه اسم غرض عمليات operation object.

كما يترتب على محرك قاعدة المعطيات XML أن يتوفر فيه آلية للقيام بعملية الإقفال locking على التسجيلات التي يتم الولوج إليها بغرض معالجة المعلومات (تحديث، حذف، ...).

العمليات التي يجب على محرك قاعدة معطيات XML دعمها:

إن طبيعة بنية معطيات XML التي تتصف بكونها غير بنيوية على غرار تلك التي توفرها قواعد المعطيات العلائقية تجعل من مسألة معالجة المعلومات من خلال لغات للاستفسارات أمراً صعباً. وقد جرى تصميم العديد من لغات الاستعلام التي يمكن تبنيها في هذا السياق، ويبين [Chinwala 2001] دراسة مرجعية لهذه اللغات.

وبناء على ما سبق يمكن تلخيص العمليات التي يترتب على محرك قاعدة المعطيات القيام بها وهي:

1. بناء وتعريف العناصر elements ومعالجتها.
2. توفير التوابع functions و operators.
3. توفير الشروط والعمليات على المتحولات.
4. توفير عبارات البرمجة التقليدية مثل عبارة for في الحلقات، عبارة let للمتحولات، عبارة where للشروط، عبارة return للبرامج الجزئية، الخ...

وبناء على كل ما سبق يمكن لنا الآن كمثال وضع خوارزمية توصيف احتساب وتنفيذ الاستعلامات التي يترتب على محرك قاعدة معطيات XML المنشود القيام بها:

1. تحليل الاستعلام query parsing إلى مكوناته الأساسية وتوليد شجرة القواعد syntax tree.
2. تحويل شجرة syntax tree إلى شجرة جبرية algebraic tree.
3. القيام بتوليد خطة الاستعلام المنطقي الخاصة بمحرك قادة معطيات XML.
4. تحويل خطة الاستعلام المنطقي إلى خطة استعلام فيزيائي.

إن مهمة تنفيذ الخوارزمية السابقة ووضع آلية لمحرك قاعدة المعطيات تسمح له بموجبها القيام بتقييم خطة الاستعلام الفيزيائية التي هي محل عدد من البحوث والدراسات التي قدمت بعض الحلول التي نذكر منها الطريقة العودية – التكرارية recursive-iterative [McHugh 1999] وطريقة معالجة الاستفسارات query processing [Mani 1996] التي تقوم بالتنفيذ على مرحلتين، كما يجدر بالذكر طريقة الفهرسة البنيوية structured indexing المقدمة في [Joseph 1999].

إن القاسم المشترك الأعظم للطرق المبينة أعلاه في تقييم وتنفيذ الاستعلام على محرك قاعدة معطيات XML هو الوصول إلى المعطيات بأسرع وقت اقتصادي ممكن مما يجعل المحرك ذا فعالية ووثوقية عاليتين وأداء متميز.

وضمن هذا الإطار ولأغراض دراسة الأداء ومعاييره المختلفة على محرك قاعدة معطيات XML يحقق شروط التصميم المبينة في هذا البحث فإنه يترتب القيام بما يلي:

أ- إنشاء قاعدة معطيات تحاكي أحد التطبيقات العملية مثل شركة، مؤسسة، مشفى...

ب- القيام بإدخال معطيات حقيقية إلى هذه القاعدة على مراحل مما يسمح بعملية الاختبار لكل الحجوم المتزايدة لقاعدة المعطيات (أي مثلاً زيادة أعداد المرضى أو طلبيات الأدوية في تطبيق قاعدة معطيات المستشفى ...).

ت- القيام بتنفيذ الاستعلامات على قاعدة المعطيات باستخدام محاك للمحرك الذي جرى وضع شروط تصميمه وذلك من خلال اتباع أحد استراتيجيات تنفيذ الاستعلامات المبينة أعلاه.

بناء النموذج التجريبي والاختبارات العملية:

النموذج التجريبي:

نظراً لمحدودية بعض إمكانات الاختبارات في محيط العمل المحلي من حيث منصات العمل عالية الأداء (ذات المعالجات المتعددة أو البنى العنقودية clustered أو غير ذلك)، فقد تم إجراء الاختبارات على بنى مادية متوسطة الأداء ومتوفرة مخبرياً بشكل عام وبحيث ترتب بعد ذلك القيام بعمليتي تقييس scaling واستيفاء extrapolation على هذه النتائج مما أدى إلى تحقيق هدفين هما:

1- التعرف على تأثير الأسس الناظمة لعمل محركات قواعد معطيات XML التي قمنا بمساهمة في وضع بعض معايير أسس تصميمها.

2- استنباط واستخلاص المعايير الخاصة باختبار وتقييم أداء هذه المحركات.

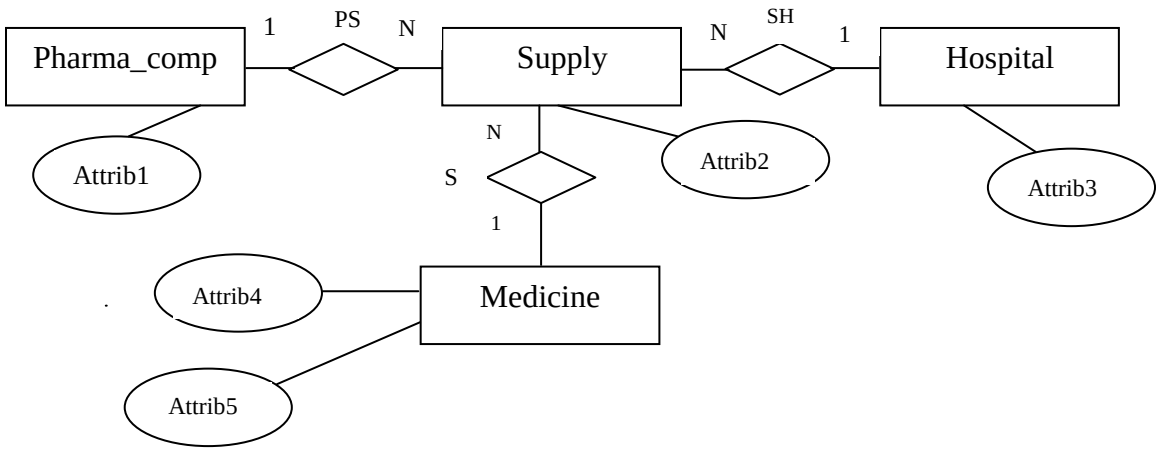
قامت عملية الاختبار العملي في محاكاة عمل محرك XML واستخلاص المعايير المناسبة للتقييم على بناء قاعدة معطيات ذات حجم يزيد عن 50000 تسجيلة record وبحيث تتضمن هذه القاعدة عدداً من الجداول tables مما يسمح باختبار العلاقات الممكنة بين حقول الجداول المختلفة ويقترب بطبيعة الحال من التطبيقات العملية التي يمكن أن تتواجد على منصات مختلفة مرتبطة أو غير مرتبطة بمحيط الوب. تمثلت البيئة الاختبارية في بناء شبكة حاسوبية محلية يرتبط بها عدد من محطات العمل (<20 محطة) ومخدم تتوفر فيه إمكانات معالجة وتخزين عاليتي المستوى. بدأت عملية المحاكاة ببناء قاعدتي معطيات علائقية والأخرى XML أصيلة على المخدم ومن ثم جرى الولوج إليهما بطريقتين الأولى محلية من خلال محطات الشبكة المحلية والثانية من محطة عمل بعيدة ضمن مبدأ RAS مما اقترب إلى حد ما من التطبيقات العاملة في محيط شبكة واسعة ذات تطبيقات عاملة في محيط الوب.

وكما بينا سابقاً فقد جرى في بحثنا هذا توظيف البنية الشجرية (التي تسمى أحياناً الهرمية) في تخزين معطيات XML اعتماداً على التوجهين الآتيين:

- بناء واستخلاص المعطيات من قاعدة معطيات علائقية بهدف توليد وثائق XML documents وإجراء الاستعلامات من خلال محاكاة الأسس الموضوعة لبناء محرك قاعدة معطيات XML.
- تخزين المعطيات في قاعدة معطيات XML أصيلة ومن ثم استخلاص المعلومات من خلال إجراء الاستعلامات.

(لا بد من التنويه إلى الدور الذي لعبته قواعد لغة ورقة التنسيق القابلة للتوسيع Extensible Stylesheet Language (XSL) في هذا البحث من خلال تحويل مستندات XML إلى صيغ قابلة للفهم والاستخدام فيما يمكن أن يكون شكلاً من أشكال المواءمة مع الحاسب).

وظفت في هذا البحث قاعدة معطيات مختزلة ومبسطة لدورة إنتاج وتأمين الأدوية لمستشفى، حيث تم استخدام محرك قاعدة المعطيات في استخلاص وثائق XML كما هو مبين في الشكل (6):



الشكل (6): مخطط ER مبسط لقاعدة معطيات أدوية مستشفى واللازمة لاستخلاص وثيقة XML

وللتعرف على محددات الأداء من خلال المعايير التي سيتم استخلاصها في هذا البحث فقد قمنا باستخلاص معلومات XML من قاعدة معطيات المستشفى والتي تتمثل في بعض واصفات الكينونات (pharma_comp, hospital, medicine) وهي في هذه الحالة أسماء وكميات، الخ... على سبيل المثال فإننا في حقيقة الأمر ولأغراض هذا البحث نحتاج إلى مجتزءات من هذه الواصفات (أي أجزاء من المعلومات).

يمكن ولأهداف المعالجة استخلاص عدد من البنى الشجرية من قاعدة المعطيات المجتزأة وبحيث يكون الجذر root فيها مبنياً على الطريقة التي نرغب ببناء قاعدة معطيات XML منها. أحد الاحتمالات هو الشكل (7) الذي يكون الجذر فيه على سبيل المثال هو الشركة المنتجة للدواء والتي ينجم عنها ظهور اسم الشركة عدداً من المرات يتناسب مع الأدوية المنتجة من قبلها والتي يقتنيها المستشفى عادة.

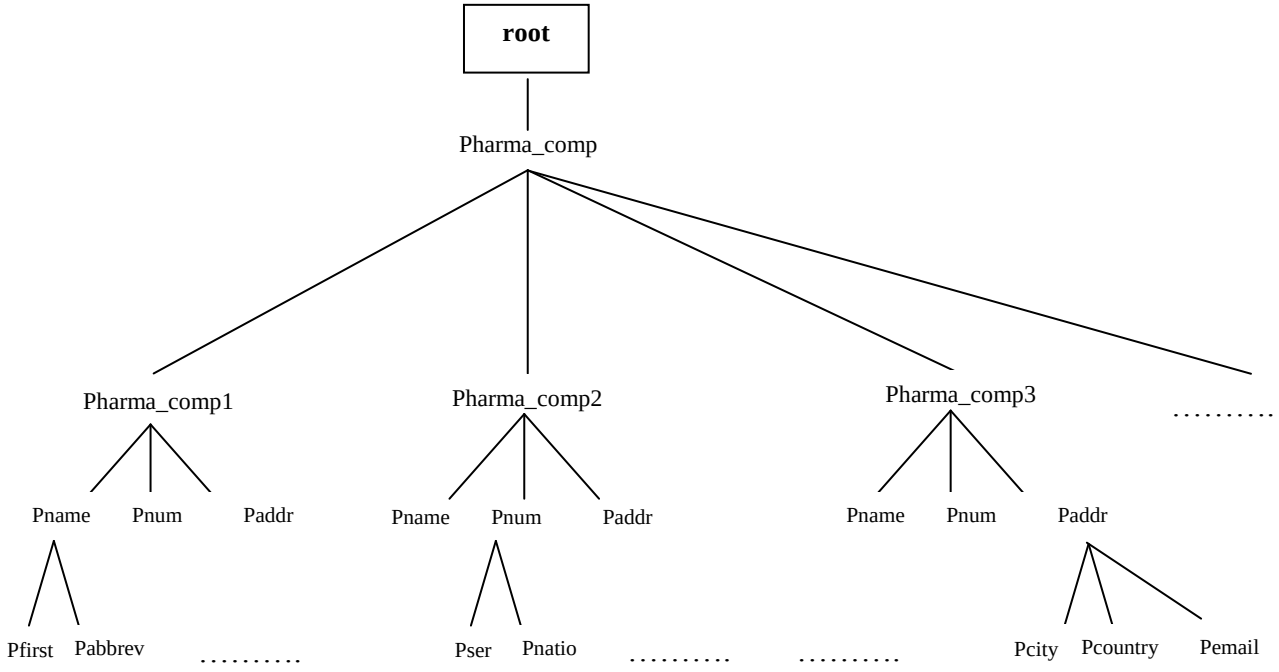
توجت عملية بناء قواعد المعطيات بالنوعين المنوه عنهما أعلاه بإجراء مجموعة من الاستعلامات للوصول إلى استنتاج عدد من المعايير اللازمة للقيام بتقييم واختبار أداء محركات قواعد معطيات XML وبما يتناسب مع عدد العناصر (العقد) التي سنبين نتائجها لاحقاً للوصول إلى النتائج الخاصة بهذا البحث.

نذكر في هذا السياق أنه قد قدم حتى الآن عدد من المقترحات حول لغات الاستعلام الخاصة بـ XML، حيث تبين وجود توجهين معياريين لذلك هما:

1- لغة استعلام XPath التي يجري فيها الوصول إلى العناصر elements والواصفات attributes والمعلومات المحرفية التي تسمى بشكل عام العقد في وثائق XML.

2- لغة استعلام XQuery التي اعتمدتها توصيات W3C في الاستفسارات الخاصة بقواعد معطيات XML.

ونظراً لسهولة طريقة الاستعلام الأولى فقد تقرر استخدامها لأغراض هذا البحث وذلك من حيث التعامل مع العقد nodes كما تقرر أيضاً حصر عملية الاستعلام في الأبناء، الأحفاد وواصفاتها وأن تكون الاستعلامات من النوعين top-down و down-up وبحيث يمر مسار البحث إما أفقياً (عرضانياً) breadth أولاً أو عمودياً depth ما يؤمن تبسيط كل من المعالجة والتحليل.



الشكل (7):البنية الشجرية لوثيقة XML

النتائج العملية:

يبين الجدول (أ) معاملات الاختبارات التي تم استخدامها في التجارب العملية على محاكي محرك قاعدة المعطيات من حيث عدد العناصر (أو العقد) وعدد المستويات في البنية الشجرية ونسبة طول توزع الاستعلامات على عدد العناصر.

الجدول (أ): معاملات الاختبارات المستخدمة في التجارب

المعامل	القيمة الافتراضية	مجال القيم المستخدمة
عدد العناصر (أو العقد)	40	130 - 20
عدد المستويات الشجرية	7/3	7 - 0
عدد الاستعلامات	75	-
نسبة طول توزع الاستعلامات على عدد العناصر	75% موجودة في القاعدة، 25% عشوائية	0- 25%

الاختبارات:

قمنا بإجراء عدد من الاختبارات للتعرف على مردود عملية الاستعلام باستخدام المثال المبين أعلاه وهو قاعدة معطيات أدوية مستشفى، وذلك من خلال القيام ببناء عدد من الاستفسارات على ضوء عدد العقد التي يترتب المرور بها للوصول إلى المعلومة المطلوبة إن توفرت. تمثل العقد كينونات معلومات أو واصفات أو معطيات محرفية توصيفية خاصة بوثائق XML.

وعلى سبيل المثال للوصول إلى اسم المدينة في عنوان الشركة المنتجة لدواء ما فإنه يترتب المرور بأربع عقد، وللحصول على العنوان البريدي لشركتين تنتجان الدواء نفسه، فإنه يترتب المرور بثماني عقد مع التنويه إلى أن عملية الاستعلام هذه هي عملية join .

قمنا بإجراء عدد من التجارب على قاعدة معطيات متزايدة تبدأ من 2000 تسجيلة وتصل إلى أكثر من 50000 تسجيلة أو وثيقة XML، وذلك لقياس عدد من العوامل نورد منها في هذا البحث تأثير كل من عدد عناصر elements الوثيقة، وعدد مستوياتها وطول الاستعلام على الأزمنة اللازمة لمعالجة الاستعلام ونبين بعضاً من النتائج التي جرى الحصول عليها والتي نبينها في الجداول الآتية:

أ- تأثير عدد عناصر الوثيقة:

تم في هذه التجربة دراسة تأثير عدد عناصر وثيقة XML document المستخرجة من قاعدة المعطيات بنوعيها العلائقي والأصيل، حيث تراوح عدد عناصر الوثيقة الواحدة بين 10 و 140 عنصراً. تبين في هذه التجربة أن كلاً من البنية الهرمية ونوع مؤشر الفهرسة لهما تأثير مباشر من حيث طريقة تحديد مسار البحث أفقياً breadth أو عمودياً depth. يبين الجدول (ب) زمن المعالجة لكل من حالتي البحث الأفقي والعمودي، حيث نجد أن زمن المعالجة في حالة الفهرسة العمودية (bottom-up) هو أفضل نسبياً من حالة الفهرسة (top-down) ويعود ذلك إلى احتمال توفر المعلومات المبحوث عنها ضمن هذا التوجه في الاستعلام بشكل أسرع.

الجدول (ب): زمن المعالجة لكل من حالتي البحث الأفقي والعمودي

	زمن المعالجة (ثانية)	
عدد التسجيلات (records)	**مسار فهرسة** (top-down)	**مسار فهرسة** (bottom-up)
2000	13	9
4000	21	15
8000	29	21
10,000	40	27
20,000	75	61
30,000	106	89
40,000	142	125
50,000	174	144

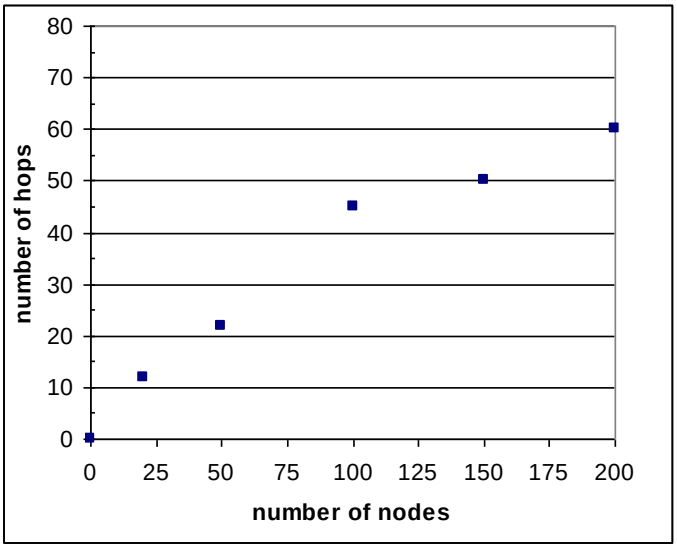
ب- تأثير عدد مستويات الوثيقة:

نظراً لكون كل عقدة تحتوي على توصيف منتظم لمحتويات الوثيقة والخدمات التي تؤديها في سياق تبادل المعلومات، لذلك فإن كل وثيقة يجب أن تفهرس بشكل فعال مما يعطي مردوداً كبيراً في الاستعلامات واسترجاع المعلومات. وقد تظهر الحاجة أحياناً إلى أن يتطلب أحد الاستعلامات من عقدة ما معلومات قد تكون لها علاقة بالمعلومات المتوفرة على عدد كبير من العقد المجاورة مما يحتم على محرك قاعدة معطيات XML

توفير آلية لإيجاد العقد التي تحتوي على معلومات ذات صلة بموضوع الاستعلام، بفعالية عالية.

جرى في هذه التجربة دراسة عمل نموذج محرك قاعدة معطيات XML من خلال عدد مستويات الوثيقة التي تحوي على عدد من العقد تراوح بين 20 و 200 عقدة. قمنا بقياس عدد القفزات hops التي يجريها الاستعلام للوصول إلى أول نتيجة يجري البحث عنها.

تبين النتائج التي حصلنا عليها أن الأداء ينخفض بشكل واضح عندما يتم تنفيذ الاستعلام للقيام بالبحث بشكل عرضاني أولاً بينما يتحسن أداء البحث من حيث عدد العقد التي يتم استعراضها والوقوف عند كل منها وصولاً إلى النتائج المطلوبة باستخدام الاستعلام العمودي شبه الخطي (depth first). كما نشير في هذا السياق إلى أن الحالة الأسوأ في عملية الاستعلام هي تلك التي يتساوى فيها عدد القفزات مع عدد العقد والذي يمكن أن نعزوه إلى عملية الرجوع backtracking. يبين الشكل (8) النتائج التي تم الحصول عليها:

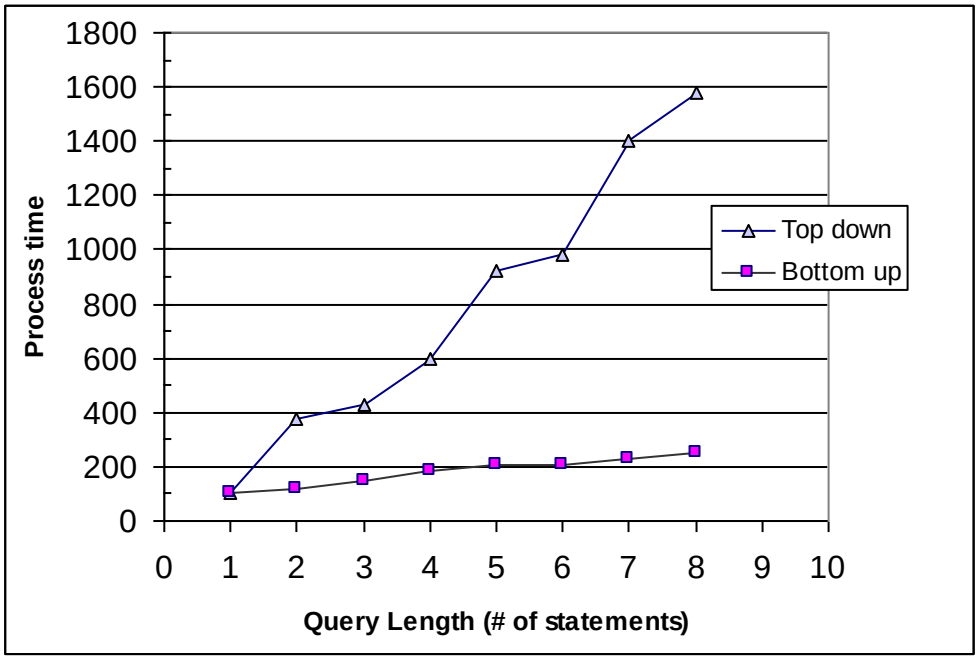


الشكل (8): أداء نموذج محرك قاعدة معطيات XML من خلال عدد العقد وعدد قفزات الاستعلام

جرى في هذه التجربة دراسة عمل نموذج محرك قاعدة معطيات XML من خلال عدد مستويات الوثيقة التي تحوي على عدد من العقد تراوح بين 20 و 200 عقدة. قمنا بقياس عدد القفزات hops التي يجريها الاستعلام للوصول إلى أول نتيجة يجري البحث عنها.

ج- تأثير طول الاستعلام:

قمنا في هذا الاختبار بدراسة تأثير طول الاستعلام (عدد العبارات statements) على زمن المعالجة من حيث عدد المستويات levels التي ينبغي المرور فيها للوصول إلى العقدة (أو العقد) المطلوبة والمبينة في الشكل (9). جرت الاختبارات على 50 عنصراً element وخمسة مستويات بنية شجرية. لوحظ في هذا الاختبار أن أداء محرك قاعدتي المعطيات (العلائقية والأصيلة) يتحسن بشكل كبير عند إجراء الاستعلام للوصول إلى أول العناصر المبحوث عنها، إن كانت موجودة فعلاً، إضافة إلى ضرورة توفير مفهرس مسار path index مناسب يشير إلى معلومات بنية قاعدة المعطيات، مما يساعد في الوصول إلى الحقول المطلوبة وذلك من خلال المرور بقاعدة المعطيات إما بشكل عرضاني أو من الأعلى للأسفل top-down.



الشكل (9): دراسة تأثير طول الاستعلام على زمن المعالجة

معايير تقييم واختبار أداء محركات قواعد لغة التأشير القابلة للتوسيع XML:

أ- عند تقييم الاستعلام الذي يتم من الأعلى إلى الأسفل، فإن المرور عبر المخطط الشجري يجري من الجذر إلى آخر فروع قاعدة المعطيات للحصول على المعلومات المطلوبة أو للتأكد من عدم وجودها. وخلال ذلك يتم وضع قفل lock على العقد الحاوية على التسجيلات كافة التي تقع ضمن إطار المعلومات المبحوث عنها وذلك باستخدام كل من بنية المفهرس الخاص بمحرك قاعدة المعطيات من جهة ومسار الفهرسة الخاص بالاستعلام نفسه. وعندما يكون عدد العناصر التي تم العثور عليها كبيراً، فإن ذلك يشكل عبئاً على محرك قاعدة المعطيات ويدعو إلى اللجوء إلى مبدأ التنفيذ بالطريقة المتسايرة concurrent، مما يحتم استخدام بنى مادية متطورة من جهة وقواعد معطيات عالية الأداء وقادرة على توظيف تلك البنى المادية واستخدامها على النحو المناسب. ويمكن أن نستخلص من ذلك أنه لتحسين أداء قواعد معطيات ذات حجوم كبيرة وتعمل في محيط الوب لابد من استخدام بنى مادية متطورة وقواعد معطيات XML أصيلة تستطيع توظيف التقانات المادية على أفضل وجه.

ب- تظهر الحاجة في حالة الحوسبة المتوزعة pervasive computing إلى إيجاد العقد التي تحتوي على معلومات تتعلق بموضوع الاستعلام، مما يجعل من الضرورة بمكان بناء مفهرسات يخزن فيها معلومات ملخصة عن تلك العقد كما يترتب أن تحوي إمكانية لتوجيه routing الاستعلام إلى العقد الحاوية على المعلومات. وضمن هذا السياق نستخلص أنه لضمان أداء محرك قاعدة معطيات XML لا بد من وضع مفهرسات تأخذ بالحسبان توزع المعلومات عبر العقد من جهة، وتلخص معلومات وثائق XML من جهة ثانية في ظل احتمال وجود عدد كبير من تلك الوثائق يقع ضمن إطار البحث في قاعدة المعطيات والحاجة إلى الوصول إلى المعلومات المطلوبة بشكل سريع.

تورد البحوث ذات العلاقة عدداً من الطرق لفهرسة وثائق XML حيث توفر وسائل فعالية في تلخيص المعطيات والاستجابة إلى استعلامات معقدة وننوه في هذا السياق إلى تلك المذكورة في [Goldman 1977] والتي

تعمل وفق نموذج البنية الشجرية لوثائق XML، كما تستطيع تخزين معلومات إحصائية وعينات من المعطيات مما يساهم في أمثلة تنفيذ الاستعلامات.

ج- إن عملية استعادة قاعدة المعطيات للوضع الذي كانت عليه قبل الوقوع في خطأ ما يفرض وضع آلية مناسبة للقيام بعدد من المهام تضمن حصول ذلك بشكل شفاف. وقد جرى في هذا البحث بناء ملف تكتب فيه المناقلات transactions المثبتة كافة التي يجريها النظام ويطلع على محتويات هذا الملف محرك قاعدة المعطيات بشكل دائم.

تبين لنا أن أداء محرك قاعدة المعطيات يتأثر سلباً كلما اتسع حجم ملف المناقلات بشكل عام وبطريقة تحديث هذا الملف بشكل خاص، حيث تبين أن عملية تحديث صورة image عن الملف تعطي نتائج أفضل من التحديث التراكمي.

الخلاصة والنتائج:

قام هذا البحث على تحديد الوظائف الأساسية المطلوبة لمحرك قاعدة معطيات XML، وهي نظام تنفيذ الاستفسارات وتعريف بنى المتحولات المختلفة وتوفير بنى التخزين اللازمة. ثم بين البحث شروط عمل محرك قاعدة المعطيات وتوصيف البنية التفصيلية للمكونات الأساسية المقترحة لتصميم هذا المحرك ووظائف كل منها، وذلك من حيث توفير مخازن للمعطيات كعناصر غرضية وتوفير متحولات مرجعية تشير إلى حقول الفهرسة الخاصة بالجداول وقيمها، إضافة إلى مكون تخديم التنفيذ المتساير concurrent processing الذي يتطلب وجود مدير للمناقلات Transaction Manager ومكون معالجة الاستفسارات. وقد جرى أيضاً البحث في العمليات التي يجب على محرك قاعدة معطيات XML دعمها مثل تحليل الاستعلام إلى مكوناته الأساسية وتوليد شجرة القواعد وتحويلها إلى شجرة جبرية والقيام بتوليد خطة الاستعلام المنطقي الخاصة بمحرك قادة معطيات XML، ومن ثم تحويل خطة الاستعلام المنطقي إلى خطة استعلام فيزيائي.

أما الجانب التطبيقي من البحث فقد جرى فيه إنشاء قاعدة معطيات تحاكي أحد التطبيقات العملية بهدف التعرف على أداء محرك قاعدة معطيات XML من خلال محاكاة عمل هذا المحرك وفق أسس التصميم التي أسهمت بها هذا البحث. تم القيام بعدد من التجارب على قاعدة معطيات (> 50000 تسجيلة أو وثيقة XML) وبشكل متزايد، وذلك لقياس عدد من العوامل بينا منها في هذا البحث تأثير كل من عدد عناصر elements الوثيقة وعدد مستوياتها وطول الاستعلام على الأزمنة اللازمة لمعالجة الاستعلام، كما وجرى بيان بعضاً من النتائج التي تم الحصول عليها.

يمكن أن تستخلص من هذا البحث ومن خلال تقييم واختبار أداء محركات قواعد معطيات XML أنه من الواجب تصميم المكونات الأساسية الخاصة بمحركات قواعد معطيات XML وفق الأسس والمعايير التي تم وضعها في هذا البحث وتوظيف بنى مادية متطورة وقواعد معطيات عالية الأداء قادرة على توظيف تلك البنى المادية واستخدامها على النحو المناسب. كما جرى الإشارة في سياق هذا البحث ولضمان أداء محرك قاعدة معطيات إلى ضرورة بناء مفهرسات تخزن فيها معلومات ملخصة عن العقد الحاوية على المعلومات وتوجيه الاستعلام إلى تلك العقد مما يؤدي إلى الوصول إلى المعلومات المطلوبة بشكل سريع.

كما تبين لنا من خلال هذا البحث إلى أن أداء محرك قاعدة المعطيات يتأثر سلباً كلما اتسع حجم ملف المناقلات بشكل عام وبطريقة تحديث هذا الملف بشكل خاص، مما يحتم توظيف آلية تحديث مناسبة تختلف عن الطريقة التراكمية مما يعطي نتائج أفضل.

المراجع:

[Arpinar 2001]: I. B. Arpinar, J. A. Miller, A. B. Sheth, 2001 "An efficient data extraction and storage utility for XML documents", Proceedings of the 39th Annual ACM S.E Conference, Athens, Ga, USA.

[Bray 2000]: T. Bray, J. Paoli. C. M. Sperberger-McQueen, and E. Maler, "Extensible Markup Language (XML), 1.0", 2nd ed. W3C Recommendation, http://www.w3.org/TR/REC-xml. Oct. 2000 .

[Chinwala 2001]: M. Chinwala , R. Malhorta, 2001, "Progress towards standards for XML databases", Proceedings of the 39th Annual ACM Conference, Athens, Georgia, USA.

[Choi 2002]: B. Choi, 2002, "What are real DTD's like", Proceedings of the 5th Workshop on the Web and Databases, ACM Press, Madison, WI, USA.

[Goldman 1977]: R. Goldman and J. Widom, "DataGuides: enabling query formulation and optimization n semi-structured databases, In Proc. 23rd VLDB Conf.'97, Athens, Greece, pp. 436-445. Morgan Kaufmann, San Francisco, CA, USA.

[Halaschek 2003]: C. Halaschek, J. Miller, 2003, "Native XML databases today", XML Journal (XMLJ) 4 (1), 39-74.

[Hors 2000]: Hors A. L., P.L. Hegaret, L. Wood, Document Object Model (DOM) Level 2 Core Specification Ver. 1.0 W3C recommendation, 20001113, 2000.

[Joseph 1999]: A. Joseph, B. Zhao, 1999, "XSet: A Lightweight database for Internet applications", Ms.C. thesis, Computer Science Division, University of California, Berkeley, USA.

[Mani 1996]: M. Mani, N. Sundaresan, 1996, "Query processing using QuiXote", IBM research report.

[McHugh 1999]: J. McHugh, J. Widom, 1999, "Query Optimization for XML", Proceedings of the 25th International Conference on Very Large Data Bases (VLDB 1999), Edinburgh, Scotland.

[Megginson 2001]: D. Megginson, SAX 2.0, "The Simple API for XML", XML-DEV mailing list. Available from http://www.saxproject.org/.

[Miller 2000]: J. A. Miller, S. Sheth, 2000 "Querying XML documents", IEEE Potentials 19 (1), 24-26.

[Moller 2001]: A. Moller, M. I. Schwartzbach, " The XML Revolution Technologies for the Future WEB", Basic research in Computer Science, Univ. of Aarhus notes series NS-01-08, http://www.brics.dk/amoeller/XML/ , Dec. 2001.