4. Base de datos XML nativa: Marklogic

XML ha ganado con el paso de los años protagonismo a la hora de trabajar con la información. Su lenguaje fuertemente tipado permite la comunicación entre distintas entidades, facilitando el intercambio de información de manera eficiente e intuitiva.

Ya que existe un lenguaje potente que permite trabajar cómodamente con los datos, el siguiente paso era establecer un sistema de almacenamiento de estos datos de manera masiva, que sustituyera o conviviese con los sistemas ya existentes. A la luz de la notoriedad que iba alcanzando XML aparecieron varias soluciones que daban respuesta a la idea básica planteada anteriormente, pero muchas de ellas han caído en desuso, y su mercado no es muy amplio. Sin embargo, algunas soluciones, principalmente comerciales, se han mantenido y ofrecen un servicio de calidad que permiten trabajar con la ingente cantidad de datos que demanda la industria tecnológica hoy en día

4.1 Introducción

Marklogic es una base de datos operacional, pensada para lo que hoy en día llamamos Big Data. Basada en almacenamiento de documentos XML de forma nativa, proporciona las herramientas necesarias para trabajar con datos estructurados, semiestructurados, o desestructurados.

El hecho de trabajar con información sin la necesidad de guardar una rígida estructura predefinida e inamovible permite que Marklogic sea una herramienta muy veloz y fácilmente escalable, dos de los requisitos que son imprescindibles para empresas que trabajan con grandes volúmenes de datos.

Marklogic tiene tras de sí un importante soporte, tanto técnico como a nivel de comunidad de internet, y soporta varios drivers para poder trabajar construyendo aplicaciones con distintos lenguajes.

Cuenta con varias herramientas que facilitan el trabajo con los documentos recurrentes con los que un administrador o desarrollador debe tratar. Cuenta con un área de administración que permite configurar las diferentes áreas con las que cuenta: servidores, bases de datos, host, privilegios… y a la misma vez cuenta con una herramienta para trabajar en sí mismo con los documentos, la consola XQUERY, que ayuda a realizar en este tipo de bases de datos las mismas tareas que pueden llevarse a cabo en otro tipo de bases de datos clásicas, gracias al lenguaje de consulta específico para XML, *xquery*.

4.2 Características

4.2.1 Orientado a documentos

Gracias a la arquitectura de Marklogic, puede soportar el trabajo con un gran número de documentos, utilizando XML o HTML como herramientas encapsuladoras de los mismos, dotando de soporte para una comunicación ágil y segura. Se muestran a continuación algunos de los formatos soportados, información suministrada por la propia compañía [6]

Archivos soportados Marklogic	Extensión
	.RPM .RAR .DMG .ISO .JAR .EXE .GZ .TAR .ZIP
	.3GP .SWF .FLV .AVI .MP3 .MPG .MP4 .MOV
	.WAV .WMA .WMV .TXT .HTM .XML .HTML

Figura 13: Archivos soportados por Marklogic, más destacados

Además de estos archivos y documentos soportados, hay muchos más que puede consultarse en [6].

El hecho de soportar todos estos formatos de documentos viene dado por la naturaleza integradora de XML, que permite embeber en cierta forma todo tipo de datos dentro de unas etiquetas predefinidas. Si dos equipos que quieren conectarse conocen cómo está diseñada la estructura de un documento XML, entonces podrán aprovechar este conocimiento para compartir información a través de estas etiquetas. Las bases de datos XML nativas están especialmente diseñadas para trabajar de esta forma, si bien las bases de datos XML enabled también pueden trabajar con XML, pero más como una capa por encima de su arquitectura habitual, que como un propósito en sí mismo.

Con Marklogic es posible generar salidas en diferentes formatos, lo que hace de ésta, una herramienta de gran versatilidad.

Documentos de Salida
XML
HTML
RSS
PDF
JSON
Microsoft Office
Adobe InDesign
Quark Express

Figura 14: Documentos de salida soportados por Marklogic

4.2.2 Xquery

Xquery es un lenguaje de consulta específicamente indicado para trabajar con documentos XML. En este caso particular, permite interactuar perfectamente con Marklogic, cumpliendo con el estándar del WC3.

Xquery permite la consulta, recuperación y manipulación de documentos XML; actualmente Marklogic soporta las siguientes variantes de Xquery:

- 1.0: Es una implementación de Xquery 1.0 sin extensiones
- 1.0-ml: Es una versión completa de Xquery con varias extensiones, algunas específicas para Marklogic
- 0.9-ml: Basada en la especificación de 2003 de Xquery, se mantiene por asuntos de compatibilidad.

4.2.3 Gestión de Datos

El modelo de indexado de datos permite a Marklogic trabajar con rapidez, pudiendo acceder a la totalidad de los datos de manera eficiente. Es posible indexar palabras, conjuntos de palabras, estructuras, valores o incluso tamaños. Destacan además los siguientes modelos de indexación:

- *Indexación de Colecciones*: Al igual que MongoDB, Marklogic server puede tratar con conjunto de datos que comparten características comunes, son las colecciones. Cada documento puede pertenecer a varias colecciones.

- *Indexación de Directorios*: Marklogic trabaja de manera interna con directorios que actúan como sistemas de ficheros, pudiendo contener ficheros, o subdirectorios. Es similar a una colección.
- *Indexación de seguridad*: Cada consulta que se lleva a cabo tiene una restricción asociada a la configuración de seguridad del usuario; la seguridad en Marklogic está basada en los roles, y según cada usuario, se le asignará uno o varios roles que seguirán unas directivas de seguridad que darán permisos a unos módulos y restringirán otros.

El sistema basado en Marklogic podrá tener varias bases de datos asociadas, sin embargo, cada consulta se hará para una base de datos específica. Cada Base de datos contiene una o varias colecciones de documentos implementados en el disco duro como un directorio físico. Estas colecciones son llamadas *Forest*, cada uno de los cuales mantiene un conjunto de documentos y todos sus índices. Estos conjuntos se comprimen como archivos binarios conteniendo subconjuntos de datos, son los llamados *Stands*, que pueden encontrarse, o no, en los *Forest.*

Cuando queremos eliminar o modificar un documento, esto no sucede inmediatamente, sino que previamente serán marcados para un posterior tratamiento. En el caso de una eliminación de documentos, primeramente se marca este documento para eliminar, y en la posterior ejecución ya se habrá hecho efectiva la operación. Para modificar un documento, se marca la versión antigua para ser eliminada, y se inserta una nueva versión modificada con los datos deseados, en la siguiente ejecución la versión antigua será retirada.

4.2.4 Escalabilidad y Seguridad

Cuando los datos crecen y la carga asociada a las consultas se hace demasiado pesada, es necesario dividir los documentos entre distintos equipos, de forma que puedan trabajar en paralelo y el peso de la carga se vea reducido. Marklogic proporciona un servicio nativo de Clustering que evita esfuerzos económicos extras en la contratación de equipos adicionales o de Ingenieros de Sistemas que ayuden a optimizar el equipamiento Software y Hardware.

La clave con la que Marklogic trata la escalabilidad se encuentra en los nodos E y D, Evaluadores y Gestores de Datos, respectivamente.

- Si aumenta la carga de las consultas, añadimos nodos-E, que escuchan permanentemente a través de sockets, analizando peticiones y generando respuestas. Es necesario configurar el

servidor para escuchar peticiones a través de un puerto determinado.

- Si crece el tamaño de los datos, añadimos nodos-D, que mantienen los datos con sus índices asociados e interactúan con los nodos-E proporcionándoles los datos que necesitan en sus consultas. Es necesario configurar el servidor para que pueda manejar datos dentro del sistema.

Gracias al intuitivo sistema gráfico de Marklogic, crear un clúster es tan sencillo como presionar un botón en el proceso de configuración del sistema.

Para proveer de seguridad al sistema de datos, Marklogic implementa algunas soluciones que permiten seguir adelante a pesar de la caída de algún servidor.

Marklogic utiliza un algoritmo de sondeo para determinar si un equipo está caído, de manera que si falla una máquina, el Clúster pueda saberlo de manera inmediata y trabajar como si el equipo indispuesto no formara parte de él, consiguiendo así una mejora del rendimiento en caso de fallo. Si falla un nodo-E, que trabaja con las consultas, sólo fallaran las peticiones dirigidas hacia éste, pero el resto podrá funcionar normalmente, mediante balanceo de carga y enrutamiento a través de los diferentes equipos. En el caso de fallo de un nodo-D el problema es mayor, y requiere de replicación para poder asegurar la integridad en el funcionamiento de Marklogic, para lo que habría que utilizar almacenamiento en discos locales o discos de datos compartidos, para poder hacer uso de ellos en caso de caída de los datos de un nodo-D, que estarían inaccesibles en su caso.

En el capítulo 4 se ha introducido Marklogic como ejemplo de base de datos XML nativa, conociendo su particular arquitectura diseñada para el trabajo en origen con documentos XML. Se ha mostrado cómo se organizan sus nodos para dar el servicio prometido, y cómo permite una ágil escalabilidad que ayuda a organizaciones a superar los retos asociados a un aumento en el número de datos con los que trabajar. A continuación, la memoria se centra en las bases de datos NoSQL, que forman un conjunto de tecnologías que parten de teorías formuladas muchos años atrás, revisándolas y adaptándolas a las necesidades actuales, centrando el estudio en la base de datos MongoDB, desarrollada por 10gen.