Prueba corta #5 - 6

Max Richard Lee Chung - 2019185076

1. Explique en que consiste un clustered index y cuál es la diferencia entre este y un índice non-clustered que utiliza INCLUDE para agregar columnas al índice. (25 pts)

- Clustered index: El índice está incrustado en los datos/archivos. Arma el árbol en el disco o una estructura de datos para evitar el viaje de abrir el disco en un puntero y sólo se puede definir un índice por tabla ya que se seleccionan las columnas para organizar las llaves. Al cambiar las columnas, gasta mucho la organización, por lo que se debe de modificar lo menos posible. La estructura tiene tiempo de respuesta rápido, sin embargo, si se añaden otras columnas de otras tablas, el tiempo de búsqueda aumenta mucho por el órden de las columnas a consultar. Además, si se realiza un ORDER BY con columnas fuera de la tabla (INNER JOIN), se tiene que realizar un ordenamiento secuencial para revisar las dos tablas. Por último, se debe de utilizar dependiendo de la carga de trabajo.
- Non-clustered index: Son archivos separados con datos, de los cuales tienen punteros con el siguiente archivo con la información (archivo apunta otro archivo) y puede tener "n" cantidad de índices aunque se considere un mal diseño. Funciona como un heap de datos, es decir, entra datos y se pega al final. Puede tener datos/archivos duplicados y en las "hojas" de los árboles. Sin embargo, si se utiliza el INCLUDE, el tiempo de respuesta a consultas regulares disminuye mucho ya que se encuentran los datos almacenados en memoria. Además, entre más pequeño sea el índice, es mejor dado que las consultas son rápidas.

2. Explique el concepto de memory footprint y como afecta este la creación de índices. ¿Cuál es la relación entre un memory footprint alto y la paginación a disco? (25 pts)

El memory footprint es la cantidad de memoria principal que una base de datos utiliza mientras está en ejecución. Permite el fácil acceso de lectura a los datos de los índices dado que están almacenados en memoria y no en disco. Si la cantidad de memoria es alta, la cantidad de información que puede tener una página es mucho mayor para el movimiento de datos entre la memoria y la base de datos, por lo que se puede tener más capacidad de respuesta rápida en comparación del uso de una memoria baja. Sin embargo, si las consultas tienen muchos índices, afecta la disponibilidad del memory footprint por la cola de consultas en espera.

3. FASTantic Inc es una empresa especializada en optimización de búsquedas sobre datos, está a sido contratada por la empresa TooSlow para ayudarle a organizar 40 billones de registros, los registros tienen las siguientes columnas: country: este es un código de país, city: está es una ciudad en un país específico, date: está es la fecha en que el registro fue agregado a los datos y payload: es un documento JSON que contiene el evento. FASTantic Inc debe optimizar la búsqueda sobre las columnas country, city y date. Explique la mejor forma de organizar los datos para incrementar la velocidad de búsqueda, actualmente se hace un scan sobre todos los datos. Asuma que no existe una base de datos mencione estructuras de datos que utilizará. ¿Que tipo de base de datos recomendaría a TooSlow para almacenar sus datos? (50 pts)

Dependiendo de cómo se quiera realizar la búsqueda, se pueden aplicar los siguientes estructuras.

- Clustered Index: Como se mencionó anteriormente, este método se utiliza dependiendo de la cantidad de tablas y cargas de trabajo. Asumiendo que no se volverá a cambiar la cantidad de columnas de los registros, se puede obtener respuestas rápidas con este método, sin embargo, el tiempo puede aumentar ligeramente si se quieren hacer ORDER BY's. Dado que se solicita la búsqueda de las columnas country, city y date, se puede realizar una consulta de esta manera.
- Non-clusterd Index: Como se mencionó anteriormente, este método puede tener más de un índice (n cantidad), por lo que puede aumentar su variabilidad de cúal(es) columna(s) pueden ser el índice a consultar. Se puede crear una variable como índice con un INCLUDE que contenga las 3 columnas mencionadas para realizar la lectura. Sin embargo, si se quieren escribir, tiene un alto costo de rendimiento dado que se debe de escribir, guardar y actualizar en diferentes archivos.
- Hash Index: Este método es el más utilizado en sistemas no distribuidos ya que busca una mejor organización de datos. Tiene una función que crea los índices únicos (identificadores primarios) de cada dato dependiendo de lo que se requiera (como por ejemplo, el código del country puede ser dividido o implementado en su totalidad como índice). Además, se pueden crear más índices para mejorar la especificación de la organización (este método se aplica analógicamente como un árbol B+). Además, tiene la característica de tener un alto rendimiennto.