**44.-EL NIVEL INTERNO**

El tercer nivel de la arquitectura es el nivel interno. La vista interna es una representación de bajo nivel de toda la base de datos y consiste en muchas ocurrencias de cada uno de los diversos tipos de registros internos. "Registro interno" es el término de ANSI/SPARC para la construcción que hemos venido llamando registro almacenado (y que seguiremos utilizando). Por lo tanto, la vista interna está todavía distante del nivel físico, ya que no tiene que ver con términos como registros físicos —también denominados bloques o páginas—ni con ninguna consideración específica de los dispositivos, como el tamaño de los cilindros o de las pistas. En otras palabras, la vista interna en efecto da por hecho un espacio de direcciones lineal infinito; los detalles de cómo el espacio de direcciones se asocia con el almacenamiento físico, son en gran medida específicos del sistema y se omiten deliberadamente de la arquitectura general. Nota: El bloque o página, es la unidad de E/S; es decir, es la cantidad de datos transferidos entre el almacena-miento secundario y la memoria principal en una sola operación de E/S. Los tamaños típicos de página son 1 KB, 2 KB o 4 KB (1 KB = 1024 bytes).La vista interna se describe por medio del esquema interno, el cual no sólo define los di-versos tipos de registros almacenados sino que especifica también qué índices existen, cómo están representados los campos almacenados, en qué secuencia están dichos registros, etcétera. El esquema interno está escrito utilizando otro lenguaje más de definición de datos: el DDL interno. Nota: En este libro usaremos normalmente los términos más intuitivos "estructura de almacenamiento" o "base de datos alma-cenada" en lugar de "vista interna", así como el término "definición de la estructura de almacena-miento" en lugar de "esquema interno".

Para terminar, señalamos que, en ciertas situaciones excepcionales, a los programas de aplicación —en particular, las aplicaciones de utilería se les podrían permitir operar directamente en el nivel interno en vez del nivel externo. Sobra decir que no es recomen-dable esta práctica, pues representa un riesgo para la seguridad (ya que se ignoran las restricciones de seguridad) y un riesgo para la integridad (debido a que, de igual manera, se ignoran las restricciones de integridad). Además, para iniciar, el programa será dependiente de los datos; aunque, en ocasiones, ésta podría ser la única forma de obtener la funcionalidad o el rendimiento requeridos (tal como le sucede al usuario de un lenguaje de programación de alto nivel que ocasionalmente tendría que descender al lenguaje ensamblador para satisfacer ciertos objetivos de funcionalidad o rendimiento).

(Date C. , 2001, pág. 40)