WHAT IS CLAIMED IS:

1. A distributed data warehouse system, comprising:

a plurality of compute nodes, implementing:

one or more storage devices providing storage for a columnar relational database table, wherein the one or more storage devices comprise a plurality of data blocks;

a multi-column key generator, configured to:

identify at least two columns of a plurality of columns of the columnar relational database table;

generate a multi-column index for the columnar relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein said multi-column index provides a respective index value for each entry of the columnar relational database table;

a write module, configured to:

direct the one or more storage devices to store the entries of the columnar relational database table sorted according to the respective index value for each entry in one or more of the plurality of data blocks; and

direct the one or more storage devices to store metadata indicating multi-column index value ranges corresponding to the index values of the entries stored in each of the one or more data blocks.

2. The system of claim 1, wherein to generate the multi-column index for the columnar relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, the multi-column key generator is configured to compress data from one or more of the at least two columns according to an order preserving compression technique.

3. The system of claim 2, wherein to compress data from the one or more of the at least two columns according to an order preserving compression technique, the multi-column key generator is further configured to:

determine a column data hierarchy for a particular one of the one or more columns from one or more other columns of the columnar relational database table; and

apply the column data hierarchy to the particular one of the one or more columns in order to group data in the particular one of the one or more columns according to the column data hierarchy.

4. The system of claim 3, further comprising:

a read module;

a query engine, configured to:

receive an indication of a query directed to one or more of the at least two columns or to one or more of the one or more other columns of the columnar relational database table for select data;

evaluate the indication of the query to determine one or more predicate data values based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein the one or more predicate data values identify the select data;

in response to receiving and evaluating the indication of the query:

analyze the metadata indicating the multi-column index value ranges for the one or more predicate data values for each of the one or more data blocks to determine particular ones of the one or more data blocks to be read in order to service the query for the select data; and

direct the read module to read the particular ones of the one or more data blocks storing data for the columnar relational database table.

5. A method, comprising:

performing, by one or more computing devices:

identifying at least two columns of a plurality of columns of a relational database table;

generating a multi-column index for the relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein said multi-column index provides a respective index value for each entry of the relational database table; and

storing the entries of the relational database table according to the respective index value for each entry.

6. The method of claim 5, wherein said generating the multi-column index for the relational database table, comprises applying an order-preserving compression technique in order to compress data for one or more of the at least two columns.

7. The method of claim 6, wherein said applying an order-preserving compression technique in order to compress data for the one or more of the at least two columns, comprises applying a column data hierarchy to a particular one of the one or more columns in order to group data in the particular one of the one or more columns according to the column data hierarchy.

8. The method of claim 5, wherein the respective index value for each entry of the relational database table is a distribution key value, wherein said storing the entries of the relational database table according to the respective index value for each entry comprises distributing the entries of the relational database table to be persisted among a plurality of different persistent storage devices based, at least in part, on the distribution key values for the entries of the relational database table.

9. The method of claim 5, wherein the respective index value for each entry of the relational database table is a sort key value, wherein said storing the entries of the relational database table according to the respective index value for each entry comprises storing the entries of the relational database table sorted according to the sort key value for the entries of the relational database table.

10. The method of claim 9, further comprising:

receiving one or more additional entries to be stored for the relational database table; and

generating one or more sort key values, based, at least in part, on the interleaving of data bits for selectivity from the identified at least two columns for the additional entries.

11. The method of claim 9, wherein the entries of the relational database table are persistently stored among a plurality of data blocks, and wherein the method further comprises maintaining metadata indicating multi-column sort key value ranges corresponding to the sort key values of the entries stored in each of the one or more data blocks.

12. The method of claim 11, further comprising:

receiving an indication of a query directed to one or more of the at least two columns of the relational database table for select data;

evaluating the indication of the query to determine one or more predicate data values based, at least in part, on the interleaving of data bits for selectivity from the identified two columns, wherein the one or more predicate data values identify the select data; and

in response to receiving and evaluating the indication of the query:

analyzing the multi-column sort key value ranges for the one or more predicate data values for each of the one or more data blocks to identify particular ones of the plurality of data blocks to be read in order to service the query for the select data.

13. The method of claim 5, wherein the one or more computing devices are part of a larger collection of computing devices implementing a data warehouse cluster storing data for one or more clients in a distributed database system, wherein the one or more computing devices together implement a compute node of the data warehouse cluster, and wherein said identifying the at least two columns of the plurality of columns of the relational database table, comprises receiving an indication of client-selected columns as the at least two identified columns.

14. A system comprising:

one or more processors;

one or more memories, the memories having stored thereon program instructions that when executed by one or more computing devices cause the one or more processors to implement a relational database system that implements:

identifying at least two columns of a plurality of columns of a relational database table;

generating a multi-column index for the relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein said multi-column index provides a respective index value for each entry of the relational database table; and

directing storage of the entries of the relational database table according to the respective index value for each entry.

15. The system of claim 14, wherein, in said generating the multi-column index for the relational database table, the program instructions cause the database system to implement applying an order-preserving compression scheme in order to compress data for one or more of the at least two columns.

WHAT IS CLAIMED IS:

1. A distributed data warehouse system, comprising:

a plurality of compute nodes, implementing:

one or more storage devices providing storage for a columnar relational database table, wherein the one or more storage devices comprise a plurality of data blocks;

a multi-column key generator, configured to:

identify at least two columns of a plurality of columns of the columnar relational database table;

generate a multi-column index for the columnar relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein said multi-column index provides a respective index value for each entry of the columnar relational database table;

a write module, configured to:

direct the one or more storage devices to store the entries of the columnar relational database table sorted according to the respective index value for each entry in one or more of the plurality of data blocks; and

direct the one or more storage devices to store metadata indicating multi-column index value ranges corresponding to the index values of the entries stored in each of the one or more data blocks.

2. The system of claim 1, wherein to generate the multi-column index for the columnar relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, the multi-column key generator is configured to compress data from one or more of the at least two columns according to an order preserving compression technique.

3. The system of claim 2, wherein to compress data from the one or more of the at least two columns according to an order preserving compression technique, the multi-column key generator is further configured to:

determine a column data hierarchy for a particular one of the one or more columns from one or more other columns of the columnar relational database table; and

apply the column data hierarchy to the particular one of the one or more columns in order to group data in the particular one of the one or more columns according to the column data hierarchy.

4. The system of claim 3, further comprising:

a read module;

a query engine, configured to:

receive an indication of a query directed to one or more of the at least two columns or to one or more of the one or more other columns of the columnar relational database table for select data;

evaluate the indication of the query to determine one or more predicate data values based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein the one or more predicate data values identify the select data;

in response to receiving and evaluating the indication of the query:

analyze the metadata indicating the multi-column index value ranges for the one or more predicate data values for each of the one or more data blocks to determine particular ones of the one or more data blocks to be read in order to service the query for the select data; and

direct the read module to read the particular ones of the one or more data blocks storing data for the columnar relational database table.

5. A method, comprising:

performing, by one or more computing devices:

identifying at least two columns of a plurality of columns of a relational database table;

generating a multi-column index for the relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein said multi-column index provides a respective index value for each entry of the relational database table; and

storing the entries of the relational database table according to the respective index value for each entry.

6. The method of claim 5, wherein said generating the multi-column index for the relational database table, comprises applying an order-preserving compression technique in order to compress data for one or more of the at least two columns.

7. The method of claim 6, wherein said applying an order-preserving compression technique in order to compress data for the one or more of the at least two columns, comprises applying a column data hierarchy to a particular one of the one or more columns in order to group data in the particular one of the one or more columns according to the column data hierarchy.

8. The method of claim 5, wherein the respective index value for each entry of the relational database table is a distribution key value, wherein said storing the entries of the relational database table according to the respective index value for each entry comprises distributing the entries of the relational database table to be persisted among a plurality of different persistent storage devices based, at least in part, on the distribution key values for the entries of the relational database table.

9. The method of claim 5, wherein the respective index value for each entry of the relational database table is a sort key value, wherein said storing the entries of the relational database table according to the respective index value for each entry comprises storing the entries of the relational database table sorted according to the sort key value for the entries of the relational database table.

10. The method of claim 9, further comprising:

receiving one or more additional entries to be stored for the relational database table; and

generating one or more sort key values, based, at least in part, on the interleaving of data bits for selectivity from the identified at least two columns for the additional entries.

11. The method of claim 9, wherein the entries of the relational database table are persistently stored among a plurality of data blocks, and wherein the method further comprises maintaining metadata indicating multi-column sort key value ranges corresponding to the sort key values of the entries stored in each of the one or more data blocks.

12. The method of claim 11, further comprising:

receiving an indication of a query directed to one or more of the at least two columns of the relational database table for select data;

evaluating the indication of the query to determine one or more predicate data values based, at least in part, on the interleaving of data bits for selectivity from the identified two columns, wherein the one or more predicate data values identify the select data; and

in response to receiving and evaluating the indication of the query:

analyzing the multi-column sort key value ranges for the one or more predicate data values for each of the one or more data blocks to identify particular ones of the plurality of data blocks to be read in order to service the query for the select data.

13. The method of claim 5, wherein the one or more computing devices are part of a larger collection of computing devices implementing a data warehouse cluster storing data for one or more clients in a distributed database system, wherein the one or more computing devices together implement a compute node of the data warehouse cluster, and wherein said identifying the at least two columns of the plurality of columns of the relational database table, comprises receiving an indication of client-selected columns as the at least two identified columns.

14. A non-transitory, computer-readable storage medium, storing program instructions that when executed by one or more computing devices cause the one or more computing devices to implement a relational database system that implements:

identifying at least two columns of a plurality of columns of a relational database table;

generating a multi-column index for the relational database table based, at least in part, on an interleaving of data bits for selectivity from the identified at least two columns, wherein said multi-column index provides a respective index value for each entry of the relational database table; and

directing storage of the entries of the relational database table according to the respective index value for each entry.

15. The non-transitory, computer-readable storage medium of claim 14, wherein, in said generating the multi-column index for the relational database table, the program instructions cause the database system to implement applying an order-preserving compression scheme in order to compress data for one or more of the at least two columns.

16. The non-transitory, computer-readable storage medium of claim 15, wherein the program instructions further cause the database system to implement:

receiving one or more additional entries to be stored for the relational database table; and

generating one or more index values, based, at least in part, on the interleaving of data bits for selectivity from the identified at least two columns for the additional one or more entries.

17. The non-transitory, computer-readable storage medium of claim 16, wherein the program instructions devices further cause the database system to implement:

detecting a re-compression event for a particular one of the one or more columns based, at least in part, on selectivity of the particular one of the one or more columns;

modifying the order-preserving compression scheme based, at least in part, on the selectivity of the particular one of the one or more columns;

applying the modified order-preserving compression scheme to compress the data for the one or more columns including the additional one or more entries in order to generate re-compressed data; and

performing said generating the multi-column index for the relational database table based, at least in part, on the re-compressed data for the particular one of the one or more columns in order to update the respective index values for each of the entries of the relational database table including the additional one or more entries.

18. The non-transitory, computer-readable storage medium of claim 17, wherein the program instructions further cause the database system to implement:

maintaining the multi-column index for servicing read requests during the performance of generating the multi-column index for the relational database table based, at least in part, on the re-compressed data;

upon completion of generating the multi-column index for the relational database table based, at least in part, on the re-compressed data, servicing read requests based, at least in part, on the updated index values for the multi-column index based, at least in part, on the re-compressed data.

19. The non-transitory, computer-readable storage medium of claim 16, wherein in said applying the order-preserving compression scheme in order to compress data for one or more of the at least two columns, the program instructions devices further cause the database system to implement adding additional data bits in order to represent values for each of the one or more additional entries.

20. The non-transitory, computer-readable storage medium of claim 15, wherein, in applying the order-preserving compression scheme in order to compress the data for one or more of the at least two columns, the program instructions cause the database system to implement applying a column data hierarchy to a particular one of the one or more columns in order to group data in the particular one of the one or more columns according to the column data

hierarchy, wherein the column data hierarchy is determined from one or more other columns of the relational database table.

21. The non-transitory, computer-readable storage medium of claim 20, wherein the respective index value for each entry of the relational database table is a sort key value, wherein the entries of the relational database table are sorted according to the sort key value for the entries of the relational database table, wherein the entries of the relational database table are persistently stored among a plurality of data blocks, and wherein the program instructions cause the database system to further implement:

maintaining metadata indicating multi-column sort key value ranges corresponding to the sort key values of the entries stored in each of the one or more data blocks;

receiving an indication of a query directed to one or more of the at least two columns or to one or more of the one or more other columns of the relational database table for select data;

evaluating the indication of the query to determine one or more predicate data values based, at least in part, on the interleaving of data bits for selectivity from the identified two columns, wherein the one or more predicate data values identify the select data; and

in response to receiving and evaluating the indication of the query:

analyzing the multi-column sort key value ranges for the one or more predicate data values for each of the one or more data blocks to identify particular ones of the plurality of data blocks to be read in order to service the query for the select data.

22. The non-transitory, computer-readable storage medium of claim 14, wherein the respective index value for each entry of the relational database table is a distribution key value, and wherein, in said directing the storage of the entries of the relational database table according to the respective index value for each entry, the program instructions cause the database system to implement distributing the entries of the relational database table to be persisted among a plurality of different persistent storage devices based, at least in part, on the distribution key values for the entries of the relational database table.